Collaboration Manager Administration Guide

10*g* Release 3 (10.1.3.3.1)

October 2007

Collaboration Manager Administration Guide, 10*g* Release 3 (10.1.3.3.1)
Copyright © 2007, Oracle. All rights reserved.

Contributing Authors: Bruce Silver

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States 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, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Table of Contents

C

Chapter 1: Introduction

Chapter 2: Collaboration Manager Overview

Chapter 3: Setting Collaboration Manager Security

Chapter 4: Using a Collaboration Workflow

Chapter 5: Projects Administration

Appendix A: Security Details

Appendix B: Third Party Licenses

Index

Chapter

1

INTRODUCTION

OVERVIEW

This section covers the following topics:

- About This Guide (page 1-1)
- New Features (page 1-2)

ABOUT THIS GUIDE

This System Administration Guide provides conceptual, reference, and step-by-step task information for all of the administrative functions of Oracle Collaboration Manager.

- Audience (page 1-1)
- Conventions (page 1-1)

Audience

This guide is for people who administer Oracle Collaboration Manager and for everyone to whom the system administrator assigns sub administrator rights.

Conventions

- The notation <*install_dir*>/ is used to refer to the location on your system where Content Server is installed.

- Forward slashes (/) are used to separate the directory levels in a path name. This is true when referring to files on a Windows file system or on a UNIX system. A forward slash will always appear after the end of a directory name.
- Notes, technical tips, important notices, and cautions use these conventions:

Symbol	Description
💡	**Note:** Brings special attention to information.
⚙	**Tech Tip:** Identifies information that can be used to make your tasks easier.
❗	**Important:** Identifies a required step or required information.
✖	**Caution:** Identifies information that might cause loss of data or serious system problems.

NEW FEATURES

The `EnableForcedACLSecurity` configuration variable can be used to enhance your security settings. See Altering Security with EnableForcedACLSecurity (page 3-6) for details.

Chapter 2

Collaboration Manager Overview

Overview

Most of the administration functions in Collaboration Manager are based on Content Server technology. However, some features of the Collaboration Manager are significantly different from the Content Server, and therefore require different administration. This section provides an overview of the Collaboration Manager administration features that are different from those in Content Server.

Note: This guide is intended to supersede information in the administration guides provided with Content Server only when features are different than standard Content Server features. For most administrative functions, see the administration guides.

This section includes these topics:

- Batch Loading (page 2-6)

Product Overview

Collaboration Manager enables project groups to create a content development environment. Users can easily create and remove ad-hoc project teams to create content in a secure fashion.

For an overview of Collaboration Manager features and information on the user interface, see the *Collaboration Manager User Guide*.

Software Requirements

Collaboration Manager can run on the same operating systems and is compatible with the same third-party software products as Content Server 10*g*R3, with the following exceptions:

- Collaboration Manager does not support the Classic view. Either the Trays or Top Menus view must be used in the Content Server client user profile.
- The current version of the Folders component is required for Collaboration Manager 10*g*R3. The current version of the Folders component is shipped as an extra with Content Server 10*g*R3.
- This version of Collaboration Manager does not support Records Manager or Governance, Risk, and Compliance Manager.
- The Folders and Threaded Discussions components must be installed before the Collaboration Manager. If the Collaboration Manager is installed first, project notifications and the Active Workflows page may not function correctly.
- Collaboration Manager 10*g*R3 supports the following add-on products:
    - Inbound Refinery
    - PDF Converter
    - Dynamic Converter
    - Desktop Outlook Integration
    - Desktop WebDAV Client

  Other add-on products require Consulting Services to customize the user interface.

TERMINOLOGY

The following table lists terminology usage in Collaboration Manager and Content Server:

Content Server	Collaboration Manager
alias	group
permission	access or permission
workflow	route or workflow
reviewer/contributor	editor or contributor
consumer	consumer

ADMINISTRATOR RESPONSIBILITIES

In addition to the standard Content Server administration tasks, the system administrator works with the user who is designated a collaboration project leader, helping that person set up security permissions for elements of the collaboration. That project leader is responsible for the day-to-day activities of a collaboration project.

The system administrator is responsible for maintaining the following Collaboration Manager features:

- **System-level security—**The system administrator creates roles, creates groups (aliases), and assigns roles and accounts to users. Contributors can then assign permissions to projects, folders, and content. See Chapter 3 (*Setting Collaboration Manager Security)* for more information.
- **Workflows—**The system administrator creates Collaboration workflows and workflow tokens. Contributors then assign users to workflows for specific content items. See Chapter 4 (*Using a Collaboration Workflow)* for more information.
- **Folders Component—**Collaboration Manager uses the Content Server Folders component to manage content in a hierarchical structure. For more information on administration of this component, see the *Folders Component* section of the *Folders/WebDAV Component Guide*.
- **System Properties Settings—**When you install the Folders component, you must enable the "Allow get copy for user with read privilege" setting. **It is important that**

you do not disable this setting. If this setting is disabled, users who have only Read permission will not be able to see content items in Collaboration Manager folders.

❖ **Library Web Site**—Because Collaboration Manager provides a different means of creating and accessing content from a hierarchical structure, you do not need to create pages for the Library, and therefore you will not use the Web Layout Editor.

Security

By using a special security model, Collaboration Manager provides additional levels of security so permissions can be specified independently for projects, folders, and individual content items. See Chapter 3 (*Setting Collaboration Manager Security)* for more information.

Workflow Wizard

Collaboration Manager enables you to create Collaboration workflows, which are template-based, user-initiated workflows. See Chapter 4 (*Using a Collaboration Workflow)* for more information.

Note: You can also set up traditional Criteria workflows in Collaboration Manager but Collaboration Manager does not support Basic workflows. See the *Content Server Workflow Implementation Guide* for information on Criteria and Basic workflows.

Collaboration Projects Administration

The Collaboration Manager includes a Configuration page, a Project Roots page and a Project Listing page that are used to view collaboration project settings, defined project roots, and current projects. These pages are accessed from the Collaboration Projects folder on the Administration tray.

See Chapter 4 (*Using a Collaboration Workflow)* for more information.

USER INFORMATION FIELDS

Collaboration Manager defines options for the User Type field (defined using the User Admin applet) and provides two additional user information fields:

User Information Field	**Description**
User Type	The option list includes two options: **Internal**—The user is an employee of your company or a member of your organization. **External**—The user has access to your Collaboration Manager system but is not an employee of your company or member of your organization. **Important:** One of these User Types must be specified for a user to appear in the lists that are used to select members for projects, folders, and content items. **Note:** See Groups (page 3-4) for information on creating internal and external groups.
Company	The user's company or organization.
Phone	The user's phone number.

SUBSCRIPTIONS

Collaboration Manager uses Criteria subscriptions to enable users to subscribe to projects and folders. You can also create custom Criteria subscriptions in Collaboration Manager using the same procedure as for Content Server. See the *Managing Repository Content Guide* for details.

Batch Loading

You can batch load content into Collaboration Manager in the same way you would batch load content into Content Server. However, you need to include the following metadata so that the content is placed in the correct project and folder:

Metadata Field	**Description**
dSecurityGroup	Must specify the **Projects** security group.
dDocAccount	Specify the project where the content item is located (for example, *prj/PRJ0000024*). The project number can be determined by looking at the Project ID column on the Project Listing Page (page 5-6). The dDocAccount value is derived by adding the prefix *prj/* onto the listed Project ID value. Or, the project number can also be found in the *dDocAccount* column of the *Collections* table in the Content Server database.
xCollectionID	Specify the Collection ID of the folder where the content item is located. If the content item is located at the project level, this is the Collection ID of the project. The Collection ID can be determined by looking at the Folder ID column on the Project Listing Page (page 5-6). The xCollectionID value is the same as the listed Folder ID value. Or, the Collection ID can also be found in the *xCollectionID* column of the *Collections* table in the Content Server database.
xClbraUserList	Specify users and their permissions for the content item. • Each user name is preceded by an ampersand (&). • Each user's permissions follow the user name in parentheses. • User names are separated by commas. • The user specified as the Author in the *dDocAuthor* field should have RWDA permission. (Authors given less than RWDA permission will not be able to perform all content item functions.) For example: `xClbraUserList=&sysadmin(RWDA),&user1(RW),&guest(R)`

Metadata Field	Description
xClbraAliasList	Specify groups and their permissions for the content item. • Each group name is preceded by an @ symbol. • Each group's permissions follow the group name in parentheses. • Group names are separated by commas. For example: `xClbraAliasList=@Mktg(RWDA),@Mktg_ext(RW)`

The following example shows a typical batch load record for Collaboration Manager:

```
Action=insert
dDocName= 32465_prj_schedule
dDocType=ADACCT
dDocTitle=Project Schedule for 32465 Project
dDocAuthor=sysadmin
dDocAccount=prj/PRJ24
dSecurityGroup=Projects
xCollectionID=11
xParentCollectionID=10
xClbraUserList=&sysadmin(RWDA),&rgarcia(RWDA),&pkelly(RW), &dmarkov(R)
xClbraAliasList=@Mktg(RWDA),@Mktg_ext(RW)
primaryFile=C:\batchtestlisa2.txt
dInDate=8/03/02
<<EOD>>
```

See the *Managing System Settings and Processes Guide* for more information about batch loading.

Chapter

3

SETTING COLLABORATION MANAGER SECURITY

OVERVIEW

Collaboration Manager relies on a combination of security methods to control access to project content. These methods include the standard Content Server security groups and accounts, as well as access lists that can modify access to specific content items and folders. This section describes the Collaboration Manager security model. The following topics are included:

- Overview of Security Issues (page 3-1)
- Altering Security with EnableForcedACLSecurity (page 3-6)
- Setting Up Collaboration Manager Users (page 3-7)

Note: This section supersedes information in the *Managing Security and User Access Guide* only when Collaboration Manager security is different than standard Content Server security.

OVERVIEW OF SECURITY ISSUES

This section covers these topics:

- Administrator Responsibilities (page 3-2)
- Security Levels (page 3-2)
- Permissions (page 3-3)

- Projects Security Group (page 3-3)
- prj Account (page 3-4)
- Groups (page 3-4)

Administrator Responsibilities

Collaboration Manager's security model reduces the amount of time and effort needed by a system administrator to maintain user permissions and security structures.

Typically, after you create user logins and aliases (called *groups* in Collaboration Manager) and assign system-level permissions, projects are created and all further permissions are assigned or re-assigned by workgroup members and collaboration project leaders.

Important: Do not change the default permissions for the *admin* role or the *sysadmin* user that are predefined in Content Server. The default permissions give administrators access to all projects, folders, and content items which is helpful if an administrator needs to troubleshoot security issues. If certain users need to have a limited set of administrative permissions, you should create a new role with the appropriate subadministrator permissions. See the *Managing Security and User Access Guide* for more information.

Security Levels

Collaboration Manager applies security at four levels:

- **System**—The system administrator assigns system-level permissions and rights to each user through the Projects Security Group (page 3-3) and prj Account (page 3-4). A user's permissions at this security level determine if they are able to create projects, change member access to content, and perform administrative duties, such as adding new users or creating workflow templates.
- **Project**—Each project has a list of users and groups (similar to Content Server aliases) that have specific permissions for that project. Each project has a *project lead*, who has full administrative permission for the project.
- **Folder**—Each folder has a list of users and groups that have specific permissions for that folder. Members who have access to the parent project can be restricted from the folder, and members who do not have access to the parent project can be given permission for an individual folder. Each folder has a *folder owner*, who has full administrative permission for the folder.

- **Content Item**—Each content item and discussion has a list of users and groups that have specific permissions for that item. Members who have access to the parent project and folder can be restricted from the content item or discussion, and members who do not have access to the parent project or folder can be given permission for an individual content item or discussion. Each content item and discussion has an *Author*, who has full administrative permission for the content item or discussion.

Note: A user with the default *admin* role has additional permissions at the project, folder, and content levels. See Security by Function (page A-2) for details.

See Also

– *Permissions (page 3-3)*

– *Determining Access (page 3-5)*

– *Security by Function (page A-2)*

– *Setting Up Collaboration Manager Users (page 3-7)*

Permissions

Permissions determine what access a user has to work with projects, folders, and content items. The following permissions can be assigned to users at all four security levels:

- Read (R)
- Write (W)
- Delete (D)
- Admin (A)

See Security by Function (page A-2) for more information.

Projects Security Group

In addition to the Public and Secure security groups that are pre-defined in Content Server, Collaboration Manager creates a *Projects* security group. Users **must** have permission for the *Projects* security group to be able to access any Collaboration Manager content or perform any Collaboration Manager functions.

prj Account

In Collaboration Manager, accounts are enabled and an account named *prj* is created as the top-level project account. This is similar to accounts in large document management sites (for example, a *Human Resources* account with sub-groups such as Internal, External Recruiting, Policies, and so on). Another account can be used as a top-level account, if necessary.

- Each project created in the Collaboration Manager becomes a sub-account of the *prj* account.
- Project accounts are numbered in the order in which they were created. For example, the first project created in a new instance of Collaboration Manager will be *prj/PRJ0000001*.
- It is possible to create a non-project account for a user, but such accounts are not recognized by the Collaboration Manager.
- Anyone who has been given permission for the *prj* account will have that permission for the projects to which they are given permission.

Groups

The *group* feature of Collaboration Manager is the same as the *alias* feature in Content Server. When you define an alias in the User Admin tool, it appears as a group in the Collaboration Manager user interface.

Tech Tip: We recommend that you use groups as much as possible to help users work more efficiently and to enhance Collaboration Manager performance.

Creating Groups

Collaboration Manager differentiates between *internal* and *external* groups, similar to the way it differentiates between internal and external users. However, while you define the User Type (internal or external) in the User Admin tool, the group type is defined by how you name the group.

- **External groups** have an **_ext** suffix on the alias name (for example, *Mktg_ext*).
- **Internal groups** are defined by any other alias name that does not end in **_ext** (for example, *Mktg*).

Determining Access

Caution: Be careful not to include external users in internal groups and vice versa. This may result in users having more permission or less permission than you intended.

The permission a user has to a particular project, folder, or content item is a combination of their system-level permissions and the permission they have for that particular item. The following diagrams show examples of how security group, account, and item permissions would intersect.

Figure 3-1 Item Permissions

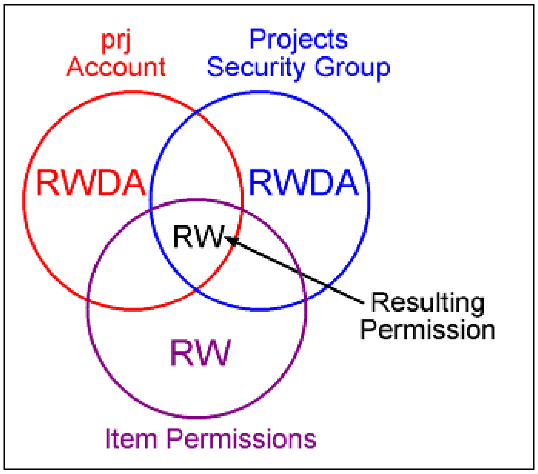
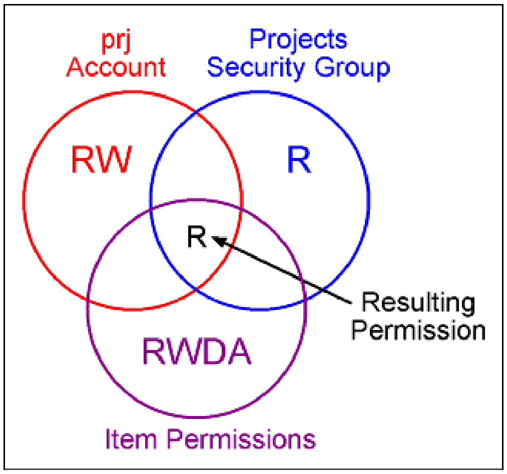


Figure 3-2 Item Permissions (more restrictive)



When different permissions intersect, the resulting permission is the ‘lowest’ common permission (that is, it is the permission with the most restrictive access).

Note: There are some exceptions to the intersection rule:

- To be able to update a content item’s member list, a user with RWDA permission for the content item needs only RW permission for the Projects security group and *prj* account.
- To be able to view a content item, a user with RWDA permission for the Projects security group and R permission for the *prj* account does not need any permission for the content item.

Altering Security with EnableForcedACLSecurity

The `EnableForcedACLSecurity` configuration variable can be used to alter the behavior of collaboration project security. This variable allows administrators to further delineate what information users can access. Without this ability, users with standard Content Server Admin permissions/rights could access portions of a project for which they did not have project rights. For example, a user with standard Content Server Admin permissions on the *Projects* security group could view a folder even if they did not have permissions to that folder established in the project. By using the `EnableForcedACLSecurity` configuration variable, an administrator can set security to always evaluate the access lists for specific content items and folders. For example:

❖ If `EnableForcedACLSecurity=TRUE` (the default)

- the access list of each content item or folder augments standard Content Server security models for each user that is not assigned the Content Server ADMIN role. This means that if a user is not assigned the Content Server ADMIN role, then access to project content is dependent on the intersection of the R, W, D, or A permissions established for that content, and the standard Content Server security group and accounts permissions. See Determining Access (page 3-5).
- if a user is assigned the Content Server ADMIN *role* when EnableForcedACLSecurity=TRUE, then the entity access lists are bypassed, and standard Content Server security models determine access to the content.

❖ If `EnableForcedACLSecurity= FALSE`

- entity access lists are bypassed, and standard Content Server security models determine access to the content.

See Appendix A (*Security Details*) for further details about the effect of the `EnableForcedACLSecurity` configuration variable.

Setting Up Collaboration Manager Users

Before setting up your Collaboration Manager users, determine how your site prefers to allow access to its data. There are two types of access models (and various permutations of each):

- A conservative model: most users are given the minimum permissions possible and collaboration project administrators can extend permissions as needed to individual content, folders, or projects.
- A liberal model: users are given wide permissions and the project administrators can restrict permissions to individual parts of the project.

Regardless of which type of model you follow, use these two steps to set up your Collaboration Manager security model:

1. Create user roles with specific permissions to the *Projects* security group.
2. Assign each user a role and specific permissions to the *prj* account.

Wide Permissions (Best Practice)

To allow for the most flexibility and the least system administrator involvement, you should give users broad system-level access to the Collaboration Manager, and then allow workgroup members to assign or re-assign permissions at the project, folder, and document levels. With this in mind, most users can be set up with the following system-level permissions:

User	**Permission to *Projects* Security Group**	**Permission to *prj* Account**
Subadministrator who can create projects	RWDA	RWDA
Contributor who can create projects	RWD	RWDA
Contributor who cannot create projects	RWD	RWD

To accomplish this security model, you could create three new roles, such as *subadmin*, *project_creator*, and *contributor*, and assign the appropriate *Projects* security group permission to each role. You would then assign each user the appropriate role and assign them the *prj* account with the appropriate permissions.

Minimum Permissions

If you need to restrict some users to less permission than the “wide permission” security model allows, you can set any combination of permissions you wish. See Appendix A (*Security Details*) for a description of the **minimum** permissions required to perform various Collaboration Manager functions.

Chapter

4

USING A COLLABORATION WORKFLOW

OVERVIEW

This section describes the following collaboration workflow concepts and administrative tasks:

- ❖ Understanding Collaboration Workflows (page 4-1)
- ❖ Setting Up Workflow Tokens (page 4-4)
- ❖ Setting Up a Collaboration Workflow (page 4-7)

Note: This section is intended to supersede the information about Collaboration workflows in the *Workflow Implementation Guide.*For an overview of workflows, information on Criteria workflows, and details on jumps and templates, see the *Workflow Implementation Guide*.

UNDERSTANDING COLLABORATION WORKFLOWS

This section covers these topics:

- ❖ About Collaboration Workflows (page 4-2)
- ❖ Collaboration Workflow Samples (page 4-2)
- ❖ Collaboration Workflow Tips (page 4-3)

About Collaboration Workflows

A Collaboration workflow is similar to a Criteria workflow, except that a workflow template and reviewers are specified by a Collaboration Manager user during file checkin rather than being defined by an administrator in the Workflow Admin tool.

❖ *Collaboration workflows* appear on the first page of the Workflow Wizard where users select a "workflow template" to route a particular content item. To access the list of available workflows, the Workflow Wizard must be launched by clicking the Configure button (associated with the Selected Workflow metadata field on the checkin window in the Collaboration Manager). Refer to *Working with Project Links* in the *Collaboration Manager User Guide*.

The remaining pages in the Workflow Wizard are based on the number of steps that contain tokens. There is always a summary page in the Wizard regardless of which workflow is chosen. For more details see *Using the Workflow Wizard* in the *Collaboration Manager User Guide*.

For users to be able to manage their own Collaboration workflows, the system administrator must first create *workflow tokens*, create *workflow information fields* that correspond to the tokens, and then add the token names to a *configuration file*.

❖ After the tokens are set up, the project administrator can create *Collaboration workflows*. A Collaboration workflow is simply a Criteria workflow with the following specific settings:

- **Security Group** = projects
- **Criteria** = Selected Workflow matches *workflow name*
- **Is Collaboration Project** check box selected
- Custom entry script for the first step

See Also

– *Setting Up a Collaboration Workflow (page 4-7)*

Collaboration Workflow Samples

Sample workflow tokens and sample Collaboration workflow templates are predefined in the Workflow Admin tool.

❖ We recommend that you use the templates as a starting point to create your own Collaboration workflows, but leave the original templates intact.

- You can use the sample tokens as is in your Collaboration workflows, you can modify them for your needs, and/or you can create your own tokens.

Collaboration Workflow Tips

- All users assigned to a Collaboration workflow must have Read permission for the content item. Editors (contributors) must have Write permission for the content item to be able to check the revision in and out.
- You cannot add or delete steps while a Collaboration workflow is enabled.
- Disabling a Collaboration workflow releases any revisions still in the workflow process to the system.
- A Collaboration workflow can use jumps to sub-workflows and other Criteria workflows in the *Projects* security group, and can jump to other steps in the same workflow.
- Users can select multiple users and groups as reviewers for each workflow step, even if you have only defined one workflow token for the step.
- You may want to assign as many workflow tokens as approvals that are required for a particular workflow step. Each token appears as a separate user field on the Choose Workflow page of the checkin window. For example, if a step requires two approvals, assign two workflow tokens so that the user knows to assign at least two reviewers to the workflow.
- **Include detailed descriptions for workflows, steps, and tokens.**
  - The workflow description appears on the Choose Workflow page of the Workflow Wizard. Contributors use it to choose a workflow. For example, “Two edit steps, one review step” is a better description than “Edit and review”.
  - The workflow step descriptions appear on the Add Reviewers page of the Workflow Wizard. Contributors use it to choose reviewers. For example, “First edit cycle, 2 approvals required” is a better description than “Edit”.
  - Token descriptions appear on the Add Reviewers page of the Workflow Wizard. Contributors use it to choose reviewers. For example, “Assign users for copy edit” is a better description than “Editor.”

SETTING UP WORKFLOW TOKENS

This section covers these topics:

- About Workflow Tokens (page 4-4)
- Creating a Workflow Token (page 4-4)
- Setting Up Workflow Information Fields (page 4-5)
- Adding Tokens to the Configuration File (page 4-7)

About Workflow Tokens

A workflow token is a piece of Idoc Script that defines variable users in a workflow.

- In a Collaboration workflow, tokens serve as placeholders for the reviewers, which are not defined until a contributor checks in the content item.
- Tokens can be used by multiple workflows.
- To set up a workflow token, you must:
  1. Create the token. See Creating a Workflow Token (page 4-4).
  2. Create a metadata field with the same name as the token. See Setting Up Workflow Information Fields (page 4-5).
  3. Add the token to the configuration file. See Adding Tokens to the Configuration File (page 4-7).

Note: See the *Content Server Workflow Implementation Guide* for detailed information on workflow tokens.

Creating a Workflow Token

To create a workflow token, perform the following tasks:

1. Launch the Workflow Admin applet.
2. Select **Options—Tokens**.

   The Workflow Tokens screen is displayed.
3. Click **Add**.

   The Add Token screen is displayed.
4. Enter the token name in the **Token Name** field.

Important: The **Token Name** must match the name of the corresponding workflow information field. See Setting Up Workflow Information Fields (page 4-5).

5. Enter a detailed description in the **Description** field. This description will appear below the user field on the Choose Workflow page of the checkin window.
6. In the Users field, enter the following text:

   ```
   <$wfAddUser(wfGet("token_name"), "user")$>
   ```

   where *token_name* is the same name you entered in the Token Name field.
7. Click **OK**.
8. Click **Close**.

See Also

– *About Workflow Tokens (page 4-4)*

– *Setting Up Workflow Information Fields (page 4-5)*

– *Adding Tokens to the Configuration File (page 4-7)*

Setting Up Workflow Information Fields

This section covers these topics:

❖ About Workflow Information Fields (page 4-5)

❖ Creating a Workflow Information Field (page 4-6)

About Workflow Information Fields

Custom metadata fields are required to hold the routing information for a Collaboration workflow. Each information field corresponds to a workflow token. You create workflow information fields as you would create any other custom metadata fields in the Configuration Manager tool.

See Also

– *Creating a Workflow Information Field (page 4-6)*

– *About Workflow Tokens (page 4-4)*

Creating a Workflow Information Field

To create a workflow information field that corresponds to a new workflow token:

1. Launch the Configuration Manager applet.
2. Click the **Information Fields** tab.
3. Click **Add**.

   The Add Custom Info screen is displayed.

4. Enter the name of the token you created.

Important: The **Field Name** must match the name of the corresponding workflow token. See Creating a Workflow Token (page 4-4).

5. Click **OK**.

   The Add Custom Info Field screen is displayed.

6. Set the properties for the field as follows:

Field	**Value**
Field Caption	*field_name*
Field Type	Memo
Require Value check box	Clear
Enable on User Interface check box	Selected
Enable for Search Index check box	Selected
Enable Option List check box	Clear

7. Click **OK**.
8. Click **Update Database Design**. (It is not necessary to rebuild the index.)

For more information, see the *Managing Repository Content Guide*.

See Also

– *About Workflow Information Fields (page 4-5)*

– *About Workflow Tokens (page 4-4)*

Adding Tokens to the Configuration File

This section covers these topics:

- About Configuration File Entries (page 4-7)
- Adding a Token to the Configuration File (page 4-7)

About Configuration File Entries

When you create a new workflow token, you need to add the token name to the *prjWizardTokens* entry in the following file:

<install_dir>/*<instance>*/data/components/CollaborationManager/config.cfg

Adding a Token to the Configuration File

To add a workflow token to the configuration file:

1. In a text editor, open the following file:

   <install_dir>/*<instance>*/data/components/CollaborationManager/config.cfg

2. Add the new token name to the end of the *prjWizardTokens* entry. For example, if your new token name is *ReviewStep3*:

   ```
   prjWizardTokens=ReviewStep1,ReviewStep2,ReviewStep3
   ```

3. Save and close the file.

4. Restart the Content Server.

See Also

– *About Workflow Tokens (page 4-4)*

SETTING UP A COLLABORATION WORKFLOW

To create a Collaboration workflow, create a Criteria workflow in the Workflow Admin tool as follows:

Note: See the *Content Server Workflow Implementation Guide* for detailed information on Criteria workflows.

1. Launch the Workflow Admin applet.

2. Click the **Criteria** tab.
3. Click **Add**.

   The New Criteria Workflow screen is displayed.
4. Enter a name in the **Workflow Name** field.
5. Enter a detailed description in the **Description** field. This description appears in the Choose Workflow list from the Configure Workflow Wizard on the Content Checkin page.
6. Select the **Projects** Security Group.
7. If templates are available, the **Use Template** checkbox appears. Select the check box and the name of the template you wish to use.
8. Select the **Has Criteria Definition** check box.
9. Select **Selected Workflow** from the **Field** list.
10. Enter the workflow name in the **Value** field.

Important: The text in the **Value** field must match the **Workflow Name** field.

11. Select the **Is Collaboration Project** check box (click **Enable**).
12. Click **OK**.
13. If a template was not used to create steps, or if you want to add other steps, click **Add** in the right pane.

    The Add New Step screen is displayed.
14. Enter an appropriate **Name** and **Description** for the step. The step description appears on the Configure Workflow Wizard Step page.
15. Select a **Type** of review step:
    - Users can review the current revision.
    - Users can review and edit (replace) the current revision.
    - Users can review the current revision or create new revisions.
16. Click **Add Token**.

    The Add Token to Step screen is displayed.
17. Select the reviewer tokens for the step.

- To select a range of tokens, click one token and then hold down the Shift key while clicking another token.
- To select tokens individually, hold down the Ctrl key while clicking each token.

18. Click **OK**.
19. Click the **Exit Conditions** tab.
20. Specify how many reviewers must approve the revision before it passes to the next step.
    - To require approval by all reviewers, select **All reviewers**.
    - To specify a minimum number of reviewers who must approve the revision, select **At least this many reviewers** and enter the number.

Note: You can enter zero (0) in the **At least this many reviewers** field to notify reviewers that the revision has reached the step, but reviewers will not be able to approve, reject, or edit the revision at that step. The workflow will pass to the next step automatically.

21. If the step requires additional exit conditions to be able to pass to the next step:
    a. Select the **Use Additional Exit Condition** check box.
    b. Click **Edit**.
    c. The Edit Additional Exit Condition screen is displayed.
    d. Select a metadata field from the **Field** choice list.
    e. Select an operator from the **Operator** choice list.
    f. Select a value from the **Value** choice list.
    g. Click **Add** to add the conditional statement to the Condition Clause.
    h. Repeat for as many conditions as required.
    i. To edit the condition expression, select the **Custom Condition Expression** check box and edit the script.
    j. Click **OK**.

Note: Exit conditions are useful when metadata could be changed by an external process during the workflow step.

22. Add the required custom script to the entry event:

Important: This custom script is required for the first step in every Collaboration workflow. Do not include this script in any other workflow steps.

    a. Click the **Events** tab.

b. Click the **Edit** button next to the Entry field.

c. The Edit Script screen is displayed. Click the **Custom** tab.

d. Select the **Custom Script Expression** check box.

e. Enter the following script:

```
<$exec inc("prj_token_init_script")$>
```

f. Click **OK**.

Note: If you will be creating several Collaboration workflows, we recommend that you create a script template for this entry event script. See the *Content Server Workflow Implementation Guide* for more information.

23. If the workflow requires additional conditional steps or special processing, add the appropriate scripts on the Events tab.

24. Click **OK**.

25. Add, edit, and delete steps as necessary to complete the workflow.

    - To add another step to the workflow, repeat steps 13 through 24.
    - To edit an existing step, select the step and click **Edit**.
    - To delete an existing step, select the step and click **Delete**.

26. Make sure that the Collaboration workflow is selected in the left pane, and click **Enable**.

    A confirmation screen is displayed.

27. Click **Yes** to activate the selected workflow.

Content items that match your specified criteria will automatically enter the workflow.

See Also

– *Understanding Collaboration Workflows (page 4-1)*

– *Setting Up Workflow Tokens (page 4-4)*

Chapter

5

PROJECTS ADMINISTRATION

OVERVIEW

This section describes the following Projects Administration topics:

- About Projects Administration (page 5-1)
- Administering Projects (page 5-7)

ABOUT PROJECTS ADMINISTRATION

Collaboration Manager provides administration pages that are used to view collaboration project settings, review the defined project roots, and check the current list of projects.

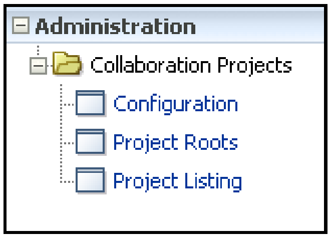
Each page can be accessed by selecting the specific page from:

- The Collaboration Projects folder on the Administration tray (refer to Accessing the Administration Pages (page 5-2).
- From the Actions pull-down menus on each page (refer to Actions Pull-Down Menus (page 5-2).

The administration pages include the following:

- Configuration Page (page 5-3)
- Project Roots Page (page 5-5)
- Project Listing Page (page 5-6)

Accessing the Administration Pages

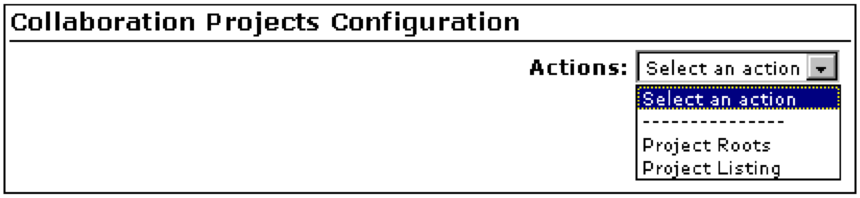


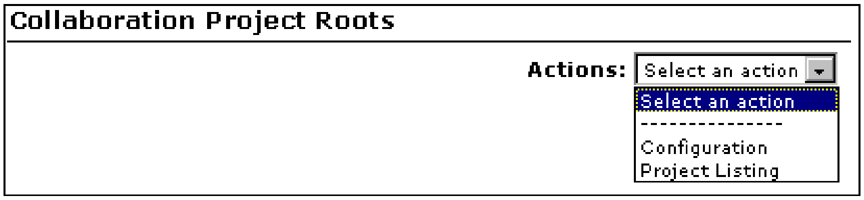
To access the Collaboration Projects administration pages:

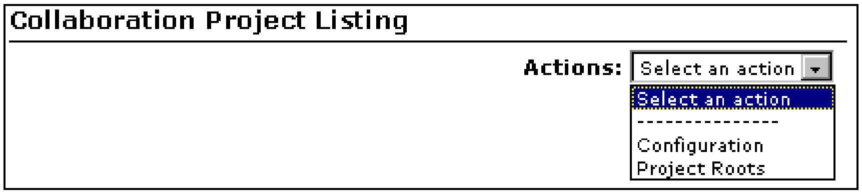
1. Click the **Administration** tray in the navigation pane.
2. Open the **Collaboration Projects** folder.
3. Select the specific page link.

   The applicable page displays.

Actions Pull-Down Menus







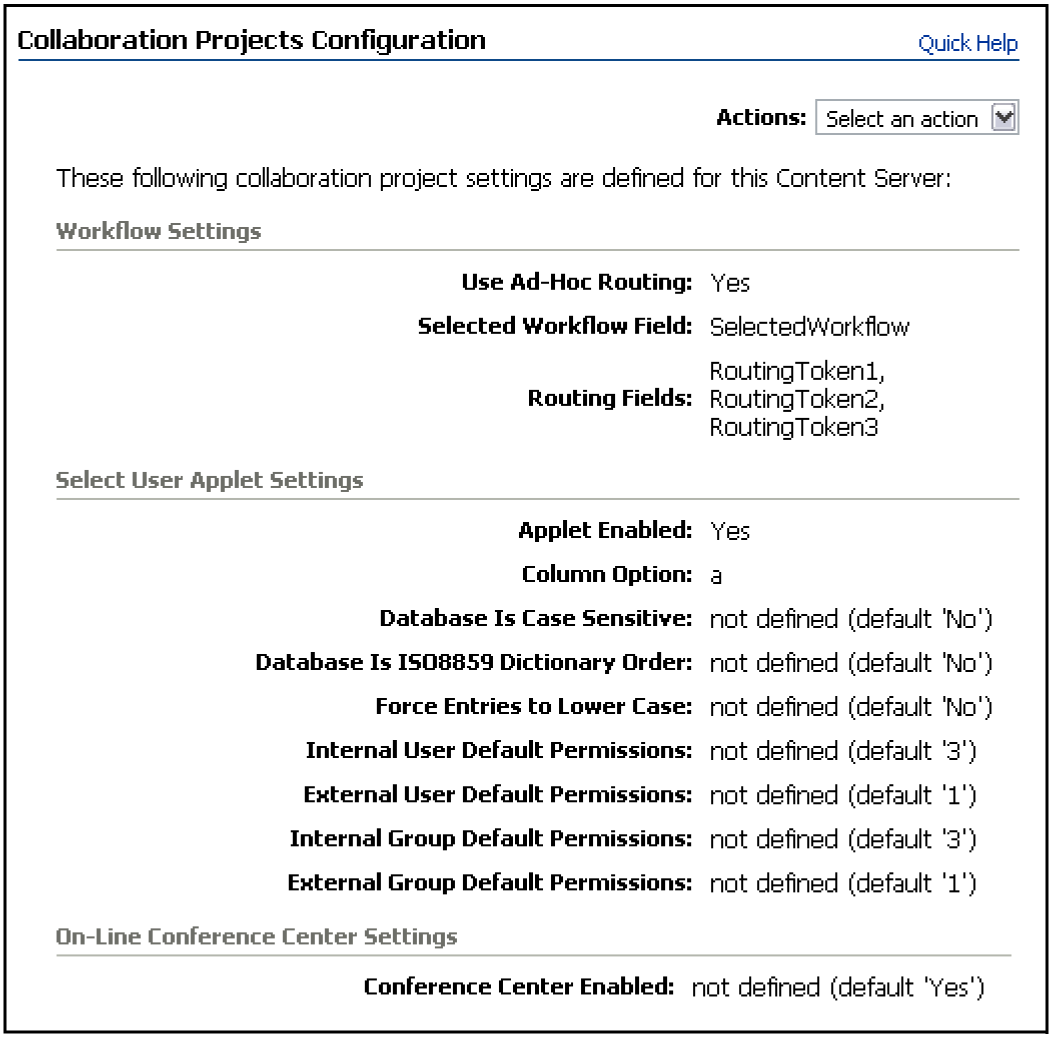
The Actions pull-down lists at the top right of each Collaboration Projects administration page enables you to access the other Collaboration Projects administration pages.

To access additional Collaboration Projects administration pages:

1. Click the arrow in the Actions pull-down list box.
2. Select the specific option.

The selected administration page displays.

Configuration Page



Collaboration Projects Configuration

Actions: Select an action

These following collaboration project settings are defined for this Content Server:

Workflow Settings

Use Ad-Hoc Routing:	Yes
Selected Workflow Field:	SelectedWorkflow
Routing Fields:	RoutingToken1, RoutingToken2, RoutingToken3

Select User Applet Settings

Applet Enabled:	Yes
Column Option:	a
Database Is Case Sensitive:	not defined (default 'No')
Database Is ISO8859 Dictionary Order:	not defined (default 'No')
Force Entries to Lower Case:	not defined (default 'No')
Internal User Default Permissions:	not defined (default '3')
External User Default Permissions:	not defined (default '1')
Internal Group Default Permissions:	not defined (default '3')
External Group Default Permissions:	not defined (default '1')

On-Line Conference Center Settings

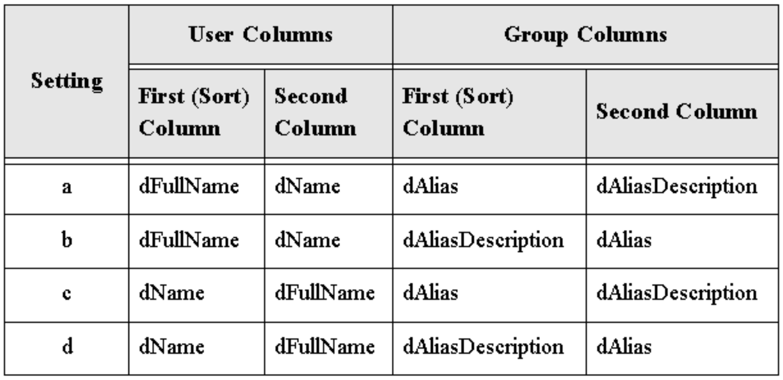
Conference Center Enabled:	not defined (default 'Yes')

The Configuration page is used to view Collaboration Manager settings. See Viewing Collaboration Manager Settings (page 5-7).

The *Configuration Setting* column in the following table refers to the configuration variables that are set in the following file:

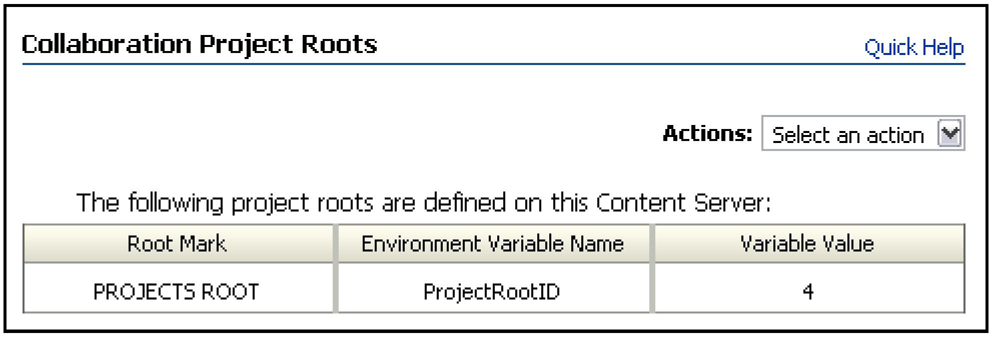
<install_dir>/*<instance>*/data/components/CollaborationManager/config.cfg

If a setting is not specifically defined, the default value is shown. See the *Collaboration Manager Installation Guide* for more information on these configuration settings.

<table>
<tr><th>Feature</th><th>Configuration Setting</th><th>Description</th></tr>
<tr><td colspan="3"><b>Workflow Settings</b></td></tr>
<tr><td>Use Ad-Hoc Routing</td><td>prjUseCustomRouting</td><td>Shows whether user-defined content item routing is enabled.</td></tr>
<tr><td>Selected Workflow Field</td><td>prjWizardWorkflowField</td><td>Shows the name of the metadata field that Collaboration workflows use as the criteria field.</td></tr>
<tr><td>Routing Fields</td><td>prjWizardTokens</td><td>Shows the name of the metadata fields that are defined as workflow tokens.</td></tr>
<tr><td colspan="3"><b>Select User Applet Settings</b></td></tr>
<tr><td>Applet Enabled</td><td>UseUserSelectApplet</td><td>Shows if the Select Member applet is enabled. Note that if you are using Apple Macintosh systems, the Select User applet is not displayed even if this option is enabled. In its place, a popup application is used which has similar functionality.</td></tr>
<tr><td>Column Option</td><td>UserSelectAppletColumnOption</td><td>Shows the display and sort order for the user and group columns in the Select Member applet. The combinations for each setting are shown below.</td></tr>
<tr><td colspan="3">
<table>
<tr><th rowspan="2">Setting</th><th colspan="2">User Columns</th><th colspan="2">Group Columns</th></tr>
<tr><th>First (Sort) Column</th><th>Second Column</th><th>First (Sort) Column</th><th>Second Column</th></tr>
<tr><td>a</td><td>dFullName</td><td>dName</td><td>dAlias</td><td>dAliasDescription</td></tr>
<tr><td>b</td><td>dFullName</td><td>dName</td><td>dAliasDescription</td><td>dAlias</td></tr>
<tr><td>c</td><td>dName</td><td>dFullName</td><td>dAlias</td><td>dAliasDescription</td></tr>
<tr><td>d</td><td>dName</td><td>dFullName</td><td>dAliasDescription</td><td>dAlias</td></tr>
</table></td></tr>
<tr><td>Database is Case Sensitive</td><td>UserSelectAppletDatabaseCaseSensitive</td><td>Shows whether the configuration setting has been enabled for a case-sensitive database.</td></tr>
</table>

Feature	Configuration Setting	Description
Database is ISO8859 Dictionary Order	UserSelectAppletDatabaseIso8859DictionaryOrder	Shows whether the configuration setting has been enabled for a SQL Server database.
Force Entries to Lower Case	UserSelectAppletForceEntriesToLowerCase	Shows whether the configuration setting has been enabled for an altered database query.
Internal User Default Permissions	UserSelectAppletInternalUserDefaultPermissions	Show the codes for the permissions that are selected by default in the Select Member applet. The following values are possible: **ValuePermissions** 1Read 3Read, Write 7Read, Write, Delete 15Read, Write, Delete, Admin
External User Default Permissions	UserSelectAppletExternalUserDefaultPermissions	
Internal Group Default Permissions	UserSelectAppletInternalAliasDefaultPermissions	
External Group Default Permissions	UserSelectAppletExternalAliasDefaultPermissions	
On-Line Conference Center Settings		
Conference Center Enabled	DisableMeetingCenter	Shows whether the On-line Conferencing feature is enabled.

Project Roots Page



Collaboration Project Roots

Actions: Select an action

The following project roots are defined on this Content Server:

Root Mark	Environment Variable Name	Variable Value
PROJECTS ROOT	ProjectRootID	4

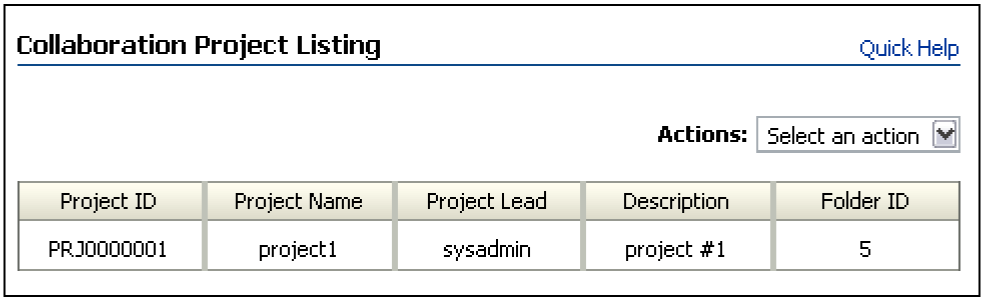
Project roots define top-level folders in the Collaboration Manager system. Each collaboration project is created within a project root folder. They provide a convenient way to organize project-related folder structures. The primary purpose of this page is to allow roots to be set up if they have not been set up through the install program. This applies primarily to custom roots because PROJECTS ROOT is set up by the installer.

The Project Roots page is used to:

- View project roots. See Viewing Project Roots (page 5-8).
- Set up custom project roots. See Configuring Project Roots (page 5-8).

Feature	Description
Root Mark column	Shows the *dCollectionMark* value (a property that identifies special folders) for the root folder. The default Root Mark for Collaboration Manager is *PROJECTS ROOT.*
Environment Variable Name column	Shows the variable name where the collection ID (*dCollectionID*) value corresponding to the *dCollectionMark* value will be stored. The default variable for Collaboration Manager is *ProjectRootID*.
Variable Value column	Shows the collection ID (*dCollectionID*) value for the root folder. This is an internal number assigned by the Folders component. If a folder name has not been defined for a project root, clicking the link displays a prompt dialog box, which enables you to name the folder. (This folder name appears at the root level in the Folders component.)

Project Listing Page



Project ID	Project Name	Project Lead	Description	Folder ID
PRJ0000001	project1	sysadmin	project #1	5

The Project Listing page is used to view a list of all projects in the Collaboration Manager. The following information appears on this screen:

Feature	Description
Project ID column	Shows the project ID for each project in the Collaboration Manager. The list is sorted in ascending Project ID order.
Project Name column	Shows the name of each project.
Project Lead column	Shows the project lead for each project.
Description column	Shows the description for each project.
Folder ID column	Shows the identification number assigned to each project.

ADMINISTERING PROJECTS

Several tasks are associated with administering collaboration projects. This section discusses these topics:

- Viewing Collaboration Manager Settings (page 5-7)
- Viewing Project Roots (page 5-8)
- Configuring Project Roots (page 5-8)
- Viewing the Collaboration Manager Project List (page 5-9)

Viewing Collaboration Manager Settings

To view the configuration settings for Collaboration Manager:

1. Log in to Collaboration Manager as the system administrator.
2. Click the **Administration** tray in the navigation pane.
3. Open the **Collaboration Projects** folder.
4. Select the **Configuration** page link.

   The Configuration Page (page 5-3) displays the Collaboration Manager settings.

Viewing Project Roots

To view the project roots for Collaboration Manager:

1. Log in to Collaboration Manager as the system administrator.
2. Click the **Administration** tray in the navigation pane.
3. Open the **Collaboration Projects** folder.
4. Select the **Project Roots** page link.

   The Project Roots Page (page 5-5) displays the Collaboration Manager settings.

   If any root folders have not been named, a **setup** link appears in the Variable Value column. See Configuring Project Roots (page 5-8) for more information.

Configuring Project Roots

Note: Configuring custom project roots may require additional component customization to the Content Server in order to assign projects to the custom root folder. It is strongly recommended that this activity be performed as part of customization by Oracle Consulting Services.

To configure a project root for Collaboration Manager:

1. In a text editor, open the following file:

   <*install_dir*>/<*instance*>/data/components/CollaborationManager/config.cfg

2. Add a line to define the new project root as follows:

   ```
   projectRoots=dCollectionMark,variable
   [; dCollectionMark,variable ...
   [; dCollectionMark,variable]]
   ```

   where

   - *dCollectionMark* is a unique name that identifies the project root folder.
   - *variable* is the variable name in which to store the *dCollectionID* that corresponds to the project root folder.

   For example:

   ```
   projectRoots=PROJ ROOT,ProjRootID;OTHER ROOT,OtherRootID
   ```

3. Save and close the file.
4. Restart the Content Server.

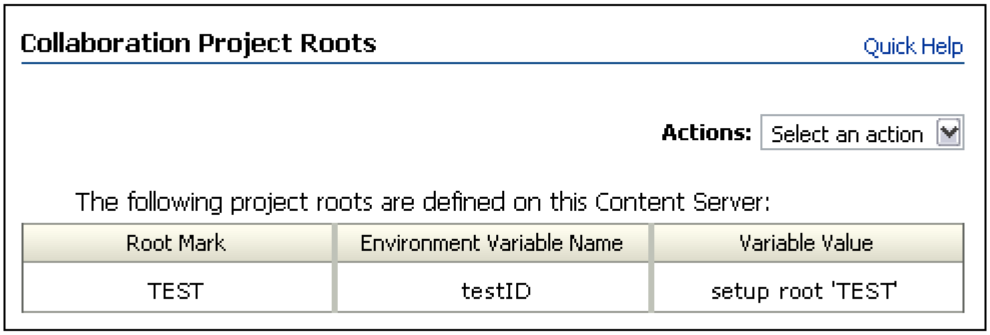
5. Click the **Administration** tray in the navigation pane.
6. Open the **Collaboration Projects** folder.
7. Select the **Project Roots** page link.

   The Project Roots Page (page 5-5) displays and includes a link in the Variable Value column to set up the project root.

Note: If a project root is defined but a folder does not exist on the system for it, then the Variable Value column contains a link that launches the appropriate folder service to set up the root.

For example, by adding the following line ‘`projectRoots=TEST,testID`’ to the config.cfg file and restarting the server, the Variable Value field on the Project Roots page contains the ‘setup root TEST’ link.

Figure 5-3 Collaboration Project Roots screen



The following project roots are defined on this Content Server:

Root Mark	Environment Variable Name	Variable Value
TEST	testID	setup root 'TEST'

8. Click the **setup root <*root mark*>** link in the Variable Value column for the new project root.

   You are prompted to enter a name for the root folder.

9. Enter a name for the root folder. This folder name appears at the root level in the Folders component.
10. Click **OK**.

    The *dCollectionID* value (Folder ID) for the new root folder is displayed in the Variable Value column.

Viewing the Collaboration Manager Project List

To view a list of all projects in Collaboration Manager:

1. Log in to Collaboration Manager as the system administrator.

2. Click the **Administration** tray in the navigation pane.
3. Open the **Collaboration Projects** folder.
4. Select the **Project Listing** page link.

Appendix

A

SECURITY DETAILS

OVERVIEW

This appendix describes the **minimum** permissions required to perform various Collaboration Manager functions. This information is included to help administrators create special-case security configurations and to assist in troubleshooting.

Note: To allow for the most flexibility and the least system administrator involvement, you should give users broad system-level access to the Collaboration Manager, and then allow workgroup members to assign permissions at the project, folder, and document levels. See Wide Permissions (Best Practice) (page 3-7) for more information.

- This chapter contains the following topics:
- About Maximum Permissions (page A-2)
- About Minimum Permissions (page A-2)
- Security by Function (page A-2)
- Security Example (page A-12)

As described in Altering Security with EnableForcedACLSecurity (page 3-6), the `EnableForcedACLSecurity` configuration variable can be used to alter the behavior of collaboration project security. By using the `EnableForcedACLSecurity` configuration variable, an administrator can set security to rely exclusively on the specific permissions which are set for aspects of a project such as folders, projects, and content.

The effect of this variable is discussed in the individual sections that describe security settings.

About Maximum Permissions

As a “best practice,” administrators should provide users with maximum system-level access to the Collaboration Manager and then allow collaboration project leaders to assign permissions to the project, folder, and content items. Using this model, the following permissions would be assigned:

- To the subadministrator who creates projects: RWDA permission to the *Projects* security group and RWDA permission to the *prj* account.
- To the contributor who can create projects: RWD permission to the *Projects* security group and RWDA permission to the *prj* account.
- To the individual contributor who cannot create projects but who participates in projects: RWD permission to the *Projects* security group and RWD permission to the *prj* account.
- See the *Collaboration Manager System Administration Guide* for details about broad permissions.

About Minimum Permissions

In Collaboration Manager, many functions can be performed from more than one location in the user interface. In some cases, this results in different permissions being required to perform the same function, depending on how the item is accessed.

For example, if a user has permission to view a content item but does not have permission to view the project or folder where the content item is located, they cannot drill down to the content item on the Browse Content tray. However, they can access the content item from a number of locations in the user interface, including the search results page, workflow notifications, subscription notifications, and so forth.

The Security by Function (page A-2) section indicates where minimum permissions are different when a task is performed from different interface locations.

Security by Function

There are three main areas where security can be assigned:

- Project Security (page A-3)
- Folder Security for Projects (page A-5)

- Content Item Security (page A-8)

Project Security

A user's permission to perform project functions is determined by their security group, account permissions, **and** their permissions for the particular project.

- The system administrator assigns system-level security—roles and accounts—to all users.
- Project permissions are assigned by the project lead or by any user who has Admin permission for the project.

Project Minimum Permissions

This section describes the **minimum** permissions required to perform project functions.

- A user with the *admin* role does not need specific permission for the project to perform project functions. Generally, an *admin* user's permission for the *prj* account determines which functions they can perform.
- By default, the project lead has RWDA permission for the project. The project lead's project permissions cannot be changed.
- By default, the user who creates a project has RWDA permission for the project. This user cannot change their own permission for the project, but another user with Admin permission for the project can change their permissions for them.

This table shows only the effect of the Admin role on security permissions. The `EnableForcedACLSecurity` configuration variable does not affect Admin role permissions.

A useful way to interpret this table is in the form of a sentence:

> *In order to <**project function**>, on <**where function is performed**>, if I have <**do not have**> the Admin role, I must have **this** permission.*

For example:

> In order to **view a project** on the **My Projects page**, if I **have** the Admin role, I **do not need permissions** to the security group, *prj* account, project account or project.
>
> In order to **view a project** on the **My Projects page** if I **do not have** the Admin role, I must **Read** permission to the Projects security group, the prj account and the project.

Project Function	Where Function is Performed	Has *admin* Role	Permission to *Projects* Security Group	Permission to *prj* Account or Specific Project Account	Permission to Specific Project
View Project	My Projects page, Find Projects Results page, Trash	yes	None	None	None
		no	R	R	R
	Exploring “name” page	yes	None	R	None
		no	R	R	R
Check Content Into and Out of Project	Exploring “name” page	yes	None	RW	None
		no	RW	RW	RW
Update Project Metadata	Edit Collaboration Project page, Project Information page	yes	None	RW	None
		no	RW	RW	RWDA
Update Project Members	Members page, Edit Project	yes	None	RW	None
		no	RW	RW	RWDA
Delete Project	My Projects page, Find Projects results, Trash	yes	None	RWD	None
		no	RWD	RWD	RWDA
Add Project	Create New Project link	yes	None	None	N/A
		no	RW	RWDA	N/A
Restore Project	Trash	yes	None	RW	RW for original project
		no	RW	RW	RW for original project

A user does not need permission to the Projects security group to add a project, but R permission is required to view the content, and RW permission is required to contribute to the project. Generally, broader permission to the Projects security group (RW) is desirable for those who need to add projects.

Sending a project to the Trash is considered a move and requires fewer permissions than those needed to delete a project.

See Also

- *Setting Up Collaboration Manager Users (page 3-7)*
- *Project Creation Example (page A-12)*
- *Project Functions Example (page A-13)*

Folder Security for Projects

A user's permission to perform folder functions is determined by their security group and account permissions and their permissions for the particular folder.

- The system administrator assigns system-level security—roles and accounts—to all users.
- Folder permissions are assigned by the folder owner or by any user who has Admin permission for the folder.

Folder Minimum Permissions

This section describes the **minimum** permissions required to perform folder functions.

- A user with the default *admin* role (RWDA for the *Projects* security group) can perform all functions for any folder, regardless of their specific project and folder permissions.
- Being assigned the admin role does not automatically grant permissions to the *prj* account. That must be assigned separately.
- By default, the folder owner has RWDA permission for the folder. The folder owner's permissions cannot be changed.
- By default, the user who creates a folder has RWDA permission for the folder. This user cannot change their own permission for the folder, but another user with Admin permission for the folder can change their permissions for them.

If `EnableForcedACLSecurity=FALSE`, the following table describes the behavior in effect.

If `EnableForcedACLSecurity=TRUE` ***and*** the user **is** assigned the Admin *role*, the following table describes the behavior in effect.

If `EnableForcedACLSecurity=TRUE` and the user **is not** assigned the Admin *role*, the Admin permissions (the **A** in RWDA permissions to the security group) are no longer sufficient to grant various folder operation privileges. Only users with the Admin role can bypass folder and content item permissions. Therefore, specific permission will be needed for the project or the folder. Those entries marked with RWD**A** indicate where these specific permissions will be needed because the Admin role is not in use.

For example, when this is enabled even if a user has RWDA permission to the Projects security group, the folder and content item permissions now apply for that user. Therefore, they would need RW permission to a specific folder in order to update that folder's metadata.

A useful way to interpret this table is in the form of a sentence:

> *In order to <**folder function**>, on <**where function is performed**>, I must have **this** permission.*

For example:

> In order to **restore a folder** from the **Trash page**, I must have RW permission to the Projects security group, RW permission to the *prj* account, no necessary permission for the project, and RW permission for the original parent folder and RW for the folder being restored.

Folder Function	**Where Function is Performed**	**Permission to *Projects* Security Group**	**Permission to *prj* Account or Specific Project Account**	**Permission to Project**	**Permission to Specific Folder**
View Folder	Trash	R	R	None	R
		RWD**A**	R	None	None
	Exploring “name” page	R	R	R	R
		RWD**A**	R	R	None
Check Content In/Out of Folder	Exploring “name” page	RW	RW	R (see **Security Note**, below)	RW
Update Folder Metadata	Hierarchy Folder Config page	RW	RW	R	RW
		RWD**A**	RW	R	None
Update Folder Members (see **Permission Note** below)	Hierarchy Folder Config page	RW	RW	R	RWDA
		RWD**A**	RW	R	None
Delete Folder	Exploring “name” page, Trash	RWD	RWD	R	RWD
		RWD**A**	RWD	R	None
Add Folder	Exploring “name” page	RW	RW	R	RW for parent folder/project
		RWD**A**	RW	R	None
Restore Folder	Trash	RW	RW	None	RW for original parent folder and RW for folder being restored
Move Folder	Exploring “name” page	RW	RW	R	R for parent folder, RW for target folder, and RW for folder being moved
Create Folder Shortcut	Exploring “name” page	RW	RW	R	R for original folder, RW for target folder

Folder Function	Where Function is Performed	Permission to *Projects* Security Group	Permission to *prj* Account or Specific Project Account	Permission to Project	Permission to Specific Folder
Delete Folder Shortcut	Exploring “name” page	RW	R	R	R to folder containing the shortcut

Security note: If the `CollaborationUseLegacySecurity` configuration variable is set to TRUE, security will be as it was for version 7.0 in which a user must have RW permission on a project to contribute content to sub-folders that grant RW (as opposed to R).

Permission note: When `EnableForcedACLSEcurity=true`, permissions needed to the folder are not RWDA but RW.

See Also

– *Folder Creation Example (page A-14)*

– *Folder Functions Example (page A-15)*

– *Setting Up Collaboration Manager Users (page 3-7)*

Content Item Security

Note: Content item security also applies to discussion items. Posting to a discussion is the same function as checking in a content item.

A user’s permission to perform content item functions is determined by their security group, account permissions, **and** their permissions for the content item.

- The system administrator assigns system-level security—roles and accounts—to all users.
- Content item permissions are assigned by the Author or by any user who has Admin permission for the content item.

Content Item Minimum Permissions

This section describes the **minimum** permissions required to perform content item functions.

- A user with the admin role can perform all functions for any content item, regardless of their specific project, folder, and content item permissions. The only exception to this is that Read permission for the parent folder is required to view a content item on the Exploring “name” page.

- By default, the Author of a content item has RWDA permission for the item. The Author’s permissions cannot be changed.
- By default, the user who checks in a content item has RWDA permission for the item. This user cannot change their own permission for the content item, but another user with Admin permission for the content item can change their permissions for them.

If `EnableForcedACLSecurity=FALSE`, the following table describes the behavior in effect.

If `EnableForcedACLSecurity=TRUE` ***and*** the user **is** assigned the Admin *role*, the following table describes the behavior in effect.

If `EnableForcedACLSecurity=TRUE` and the user **is not** assigned the Admin *role*, then the entity access list permissions for each content item augment the standard Content Server security model of security groups and accounts. Only users with the Admin role can bypass folder and content item permissions as established with the entity access list. Therefore, specific permission will be needed for the project or the folder. Those entries marked with RWD**A** indicate where these specific permissions will be needed because the Admin role is not in use.

For example, when this is enabled even if a user has RWDA permission to the Projects security group, the folder and content item permissions now apply for that user. Therefore, they would need RW permission to a specific folder in order to update that folder’s metadata.

A useful way to interpret this table is in the form of a sentence:

> *In order to <**content item function**>, on <**where function is performed**>, I must have **this** permission.*

For example:

> In order to **view an item** on the **Search Results page**, if I have **Read** permission to *Projects*, the *prj* account, and the Content Item, I do not need permissions to the parent folder or project.

Content Item Function	**Where Function is Performed**	**Permission to *Projects* Security Group**	**Permission to *prj* Account or Specific Project Account**	**Permission to Parent Folder or Project**	**Permission to Content Item**
View Content Item	Search results page, notification e-mails, Trash	R	R	None	R
		RWDA	R	None	None
	Exploring “name” page	R	R	R	R
		RWDA	R	R	None
Check Out Content Item	Exploring “name” page	RW	RW	R	RW
		RWDA	RW	R for project; none for subfolder	None
	Dashboard, Content Information page, Workflow Content page	RW	RW	R for project; none for subfolder	RW
		RWDA	RW	R	None
Check In New Content Item	Exploring “name” page	RW	RW	RW	RWDA automatically assigned to author
		RWDA	RW	R	RWDA automatically assigned to author
Check In Content Item revision	Exploring “name” page	RW	RW	RW	RW
		RWDA	RW	R	None
Update Content Item Members	Info Update Form	RW	RW	RW	RWDA
		RWDA	RWDA	None	None

Content Item Function	Where Function is Performed	Permission to *Projects* Security Group	Permission to *prj* Account or Specific Project Account	Permission to Parent Folder or Project	Permission to Content Item
Delete Content Item	Exploring “name” page	RWD	RWD	R	RWD
		RWD**A**	RWD	R	None
	Content Information page, Trash	RWD	RWD	None	RWD
		RWD**A**	RWD	None	None
Check in for other users	Exploring “name” page	RWD**A**	RWDA	R	None
Restore Content Item	Trash	RW	RW	RW for original parent folder	RW
Move Content Item	Exploring “name” page	RW	RW	R	RW for target folder, RW for content item
Create Content Item Shortcut	Exploring “name” page	RW	RW	R	R for content item, RW for target folder
		RWD**A**	RWDA	R	R for content item, None for target folder
Delete Content Item Shortcut	Exploring “name” page	RWD	RWD	R	RWD for content
		RWD**A**	R	None	None

If Trash is enabled, deleting a content item is considered a move and requires fewer permissions.

See Also

– *Content Checkin Example (page A-16)*

– *Content Functions Example (page A-17)*

– *Setting Up Collaboration Manager Users (page 3-7)*

SECURITY EXAMPLE

In this example, several internal users are set up as contributors and subadministrators with the following profiles in Collaboration Manager:

User Name	Roles	*Projects* Security Group Permission	*prj* Account Permission	Admin Rights
sysadmin	admin	RWDA	RWDA	All
hchang	subadmin	RWDA	RWDA	User Admin, Workflow Admin
pkelly	project_creator	RWD	RWDA	None
rgarcia	contributor	RWD	RWD	None
sjones	contributor	RWD	RWD	None
dmarkov	guest	R	R	None

The example is developed further in the following topics:

- Project Creation Example (page A-12)
- Project Functions Example (page A-13)
- Folder Creation Example (page A-14)
- Content Checkin Example (page A-16)
- Content Functions Example (page A-17)
- Discussions Example (page A-19)
- Workflow Example (page A-20)

Project Creation Example

With this scenario, three users can create projects: *sysadmin*, *hchang*, and *pkelly*. The other three users cannot create projects because they do not have Admin permission for the *prj* account.

In our example, *hchang* sets up a project (using **Browse Content—Manage Project—New Project**) named **2004 Annual Report**. This project will contain all of the content and communications related to the publication of the company's annual report. The following project permissions are specified:

- *pkelly* is the project lead (RWDA permission)
- *rgarcia* has RWD permission
- *sjones* has RW permission
- *dmarkov* has RW permission

See Also

– *Project Security (page A-3)*

– *Project Functions Example (page A-13)*

Project Functions Example

After assigning the appropriate permission, users can perform the following functions for the 2004 Annual Report project:

User Name	Project Functions	Explanation
sysadmin	All	The *sysadmin* user has the *admin* role and RWDA for the *prj* account, so they can perform all project functions.
hchang	All	The user who created the project has RWDA permission by default.
pkelly	All	The project lead has RWDA permission by default.
rgarcia	View, Check In Content	This user has RWD permission for the project, but you need Admin permission to be able to update or delete a project.
sjones	View, Check In Content	RW permission for the project allows this user to perform basic project tasks, but they cannot update or delete the project.

User Name	Project Functions	Explanation
dmarkov	View	Although this user has RW permission for the project, they only have Read permission for the *Projects* security group and *prj* account.

See Also

– *Project Security (page A-3)*

– *Project Creation Example (page A-12)*

Folder Creation Example

Within the 2004 Annual Report project, *pkelly* creates two new folders (using the Action pull-down menu from the Exploring Project page), naming the folders **Design** and **Content**. The Design folder is for graphics and layout files, and the Content folder will contain text and spreadsheet documents. Note that the content types for graphics and content have been pre-defined.

The following folder permissions are specified:

User Name	Design Folder Permission	Content Folder Permission
pkelly	folder owner (RWDA)	folder creator (RWDA)
rgarcia	RW	folder owner (RWDA)
sjones	RWDA	RWD
dmarkov	none	RW

After these folders are created and work on the project begins, a large number of content items are checked in to the Content folder.

See Also

– *Folder Security for Projects (page A-5)*

– *Folder Functions Example (page A-15)*

Folder Functions Example

Users can perform the following functions for the **Design** folder:

User Name	Folder Functions	Explanation
sysadmin	All	The *sysadmin* user has the *admin* role, so they can perform all folder functions by default.
hchang	All	This user has RWDA rights to the *Projects* security group and the *prj* account.
pkelly	All	The folder owner has RWDA permission by default.
rgarcia	All except Delete Folder	RW permission for the **Design** folder (along with RWD to the *Projects* security group, *prj* account, and the project) allows this user to perform most folder tasks.
sjones	All	RWDA permission for the **Design** folder (along with RWD for the *Projects* Security Group and *prj* Account, and RW for the project) allows this user to perform all folder functions.
dmarkov	None	This user was excluded from the member list, so they cannot view the **Design** folder.

Users can perform the following functions for the **Content** folder:

User Name	Folder Functions	Explanation
sysadmin	All	The *sysadmin* user has the *admin* role, so they can perform all folder functions by default.

User Name	Folder Functions	Explanation
hchang	All	This user has RWDA rights to the *Projects* security group and the *prj* account.
pkelly	All	The user who created the folder has RWDA permission by default.
rgarcia	All	The folder owner has RWDA permission by default.
sjones	All except Update Members	RWD permission for the **Content** folder (along with RWD for the *Projects* Security Group and *prj* Account, and RW for the project) allows most tasks.
dmarkov	View	Although this user has RW permission for the **Content** folder, they only have Read permission for the *Projects* security group and *prj* account.

See Also

– *Folder Security for Projects (page A-5)*

– *Folder Creation Example (page A-14)*

Content Checkin Example

Now that a project and folders are set up, the users can begin checking content items in to the folders.

- All users except *dmarkov* can check content in at the project level.
- *pkelly*, *rgarcia*, and *sjones* can check content in to the Design folder and the Content folder. In addition, *hchang* and the sysadmin can because they have Admin permission.

The following content items are checked in to the 2004 Annual Report project:

Content Item	Folder	User Checking In File	Author	Access List
Project schedule	2004 Annual Report	pkelly	pkelly	pkelly(RWDA), hchang(R), rgarcia(RW), sjones(R), dmarkov(R)
Quark design file	Design	pkelly	pkelly	pkelly(RWDA), sjones (RWDA), rgarcia(R)
Zip of graphics	Design	sjones	sjones	pkelly(RW), sjones(RWDA), rgarcia(R)
Financials spreadsheet	Content	sjones	sjones	pkelly(RW), sjones(RWDA), rgarcia(RWD),dmarkov(RW)
Annual report text	Content	rgarcia	rgarcia	pkelly(RW), rgarcia(RWDA), dmarkov(RW)

See Also

– *Content Functions Example (page A-17)*

Content Functions Example

Users have the following access to the content items in the 2004 Annual Report project:

User Name	Content Item	Functions	Explanation
sysadmin	All	All	The *sysadmin* user can perform all content item functions by default.

User Name	Content Item	Functions	Explanation
hchang	Project schedule	All	This user has RWDA rights to the *Projects* security group and the *prj* account.
	Quark design file	All	This user has RWDA rights to the *Projects* security group and the *prj* account.
	Zip of graphics		
	Financials spreadsheet		
	Annual report text		
pkelly	Project schedule	All except Check In for Other Users	RWDA permission to the *Projects* security group is required to check in for other users.
	Quark design file		
	Zip of graphics	All except Update Members, Delete Content Item, and Check In for Other Users	RW permission for the content item.
	Financials spreadsheet		
	Annual report text		
rgarcia	Project schedule	All except Update Members, Delete Content Item, and Check In for Other Users	RW permission for the content item.
	Quark design file	View and Create Shortcut	R permission for the content item.
	Zip of graphics		
	Financials spreadsheet	All except Update Members and Check In for Other Users	RWD permission for the content item.
	Annual report text	All	RWDA permission for the content item.

User Name	Content Item	Functions	Explanation
sjones	Project schedule	View and Create Shortcut	R permission for the content item.
	Quark design file	All	RWDA permission for the content item.
	Zip of graphics		
	Financials spreadsheet		
	Annual report text	None	This user was excluded from the member list, so they cannot view the content item.
dmarkov	Project schedule	View	R permission for the content item.
	Quark design file	None	This user was excluded from the member lists, so they cannot view these content items.
	Zip of graphics		
	Financials spreadsheet	View Content	Although this user has RW permission for these content items, they only have Read permission for the *Projects* security group and *prj* account. To delete a content item shortcut, this user must have Read permission for the parent folders of both the shortcut and the original content item.
	Annual report text		

See Also

– *Content Checkin Example (page A-16)*

Discussions Example

As design of the annual report is taking shape, *sjones* wants to discuss some of the layout details. She submits a post to the discussion associated with the **Quark design file** content item. She then realizes that *rgarcia* has only Read permission because the discussion inherits permissions from the associated content item. Since *sjones* has Admin permission for the discussion, she can update the member list for the discussion to give *rgarcia* RW permission; the content item permissions remain unchanged.

Workflow Example

When the annual report is ready to be published, it needs to go through a final review. *pkelly* checks the annual report file into the **2004 Annual Report** project, assigns RW permission to *rgarcia* and *sjones*, and specifies that it needs to enter a three-step workflow.

- At the first step, *sjones* and *rgarcia* are editors. Their RW permission for the content item enables them to check out the content item and edit it before approval.
- At the second step, *pkelly* is the sole editor. Because he is the author, he has RWDA permission, and can edit the content item one last time. He also realizes that he has not given *dmarkov*, the final reviewer, access to the content item; before approval, he changes the member list to give *dmarkov* Read permission for the content item.

At the last step, *dmarkov* has final approval of the annual report. Read permission for the content item allows him to approve or reject the item. However, if this was an editor step, *dmarkov* would need RW permission to be able to check out the content item.

Appendix

B

Third Party Licenses

Overview

This appendix includes a description of the Third Party Licenses for all the third party products included with this product.

Apache Software License

```
* Copyright 1999-2004 The Apache Software Foundation.

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*      http://www.apache.org/licenses/LICENSE-2.0

*
```

```
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
```

W3C® Software Notice and License

```
* Copyright © 1994-2000 World Wide Web Consortium,
* (Massachusetts Institute of Technology, Institut National de
* Recherche en Informatique et en Automatique, Keio University).
* All Rights Reserved.  http://www.w3.org/Consortium/Legal/
*
* This W3C work (including software, documents, or other related items) is
* being provided by the copyright holders under the following license. By
* obtaining, using and/or copying this work, you (the licensee) agree that
* you have read, understood, and will comply with the following terms and
* conditions:
*
* Permission to use, copy, modify, and distribute this software and its
* documentation, with or without modification, for any purpose and without
* fee or royalty is hereby granted, provided that you include the following
* on ALL copies of the software and documentation or portions thereof,
* including modifications, that you make:
*
*   1. The full text of this NOTICE in a location viewable to users of the
*      redistributed or derivative work.
*
*   2. Any pre-existing intellectual property disclaimers, notices, or terms
*      and conditions. If none exist, a short notice of the following form
*      (hypertext is preferred, text is permitted) should be used within the
*      body of any redistributed or derivative code: "Copyright ©
*      [$date-of-software] World Wide Web Consortium, (Massachusetts
*      Institute of Technology, Institut National de Recherche en
*      Informatique et en Automatique, Keio University). All Rights
*      Reserved. http://www.w3.org/Consortium/Legal/"
*
*   3. Notice of any changes or modifications to the W3C files, including the
*      date changes were made. (We recommend you provide URIs to the location
*      from which the code is derived.)
*
* THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS
* MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT
* NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR
* PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE
* ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.
*
* COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR
* CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR
* DOCUMENTATION.
*
* The name and trademarks of copyright holders may NOT be used in advertising
* or publicity pertaining to the software without specific, written prior
* permission. Title to copyright in this software and any associated
* documentation will at all times remain with copyright holders.
*
```

ZLIB LICENSE

* zlib.h -- interface of the 'zlib' general purpose compression library
version 1.2.3, July 18th, 2005

Copyright (C) 1995-2005 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages

arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly jloup@gzip.org
Mark Adler madler@alumni.caltech.edu

GENERAL BSD LICENSE

Copyright (c) 1998, Regents of the University of California

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

"Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

"Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

"Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GENERAL MIT LICENSE

Copyright (c) 1998, Regents of the Massachusetts Institute of Technology

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

UNICODE LICENSE

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories http://www.unicode.org/Public/, http://www.unicode.org/reports/, and http://www.unicode.org/cldr/data/ . Unicode Software includes any source code published in the Unicode Standard or under the directories http://www.unicode.org/Public/, http://www.unicode.org/reports/, and http://www.unicode.org/cldr/data/.

NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright © 1991-2006 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

____Unicode and the Unicode logo are trademarks of Unicode, Inc., and may be registered in some jurisdictions. All other trademarks and registered trademarks mentioned herein are the property of their respective owners

MISCELLANEOUS ATTRIBUTIONS

Adobe, Acrobat, and the Acrobat Logo are registered trademarks of Adobe Systems Incorporated.

FAST Instream is a trademark of Fast Search and Transfer ASA.

HP-UX is a registered trademark of Hewlett-Packard Company.

IBM, Informix, and DB2 are registered trademarks of IBM Corporation.

Jaws PDF Library is a registered trademark of Global Graphics Software Ltd.

Kofax is a registered trademark, and Ascent and Ascent Capture are trademarks of Kofax Image Products.

Linux is a registered trademark of Linus Torvalds.

Mac is a registered trademark, and Safari is a trademark of Apple Computer, Inc.

Microsoft, Windows, and Internet Explorer are registered trademarks of Microsoft Corporation.

MrSID is property of LizardTech, Inc. It is protected by U.S. Patent No. 5,710,835. Foreign Patents Pending.

Oracle is a registered trademark of Oracle Corporation.

Portions Copyright © 1994-1997 LEAD Technologies, Inc. All rights reserved.

Portions Copyright © 1990-1998 Handmade Software, Inc. All rights reserved.

Portions Copyright © 1988, 1997 Aladdin Enterprises. All rights reserved.

Portions Copyright © 1997 Soft Horizons. All rights reserved.

Portions Copyright © 1995-1999 LizardTech, Inc. All rights reserved.

Red Hat is a registered trademark of Red Hat, Inc.

Sun is a registered trademark, and Sun ONE, Solaris, iPlanet and Java are trademarks of Sun Microsystems, Inc.

Sybase is a registered trademark of Sybase, Inc.

UNIX is a registered trademark of The Open Group.

Verity is a registered trademark of Autonomy Corporation plc

Index