

# **Oracle® Content Manager**

Implementation and Administration Guide

Release 11*i*

**Part No. B13535-01**

June 2004

Oracle Content Manager Implementation and Administration Guide, Release 11i

Part No. B13535-01

Copyright © 2003, 2004 Oracle. All rights reserved.

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 Corporation, 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.

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.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

---

---

# Contents

<b>Send Us Your Comments .....</b>	<b>vii</b>
<b>Preface.....</b>	<b>ix</b>
How To Use This Guide .....	x
Documentation Accessibility .....	x
Other Information Sources .....	x
Training and Support.....	xvii
Do Not Use Database Tools to Modify Oracle Applications Data .....	xviii
About Oracle .....	xviii
<b>1 Introduction</b>	
1.1 Oracle Content Manager Overview.....	1-2
1.1.1 Oracle Content Manager Features .....	1-2
1.2 New in this Release .....	1-3
<b>2 Verifying Dependencies</b>	
2.1 Verifying CRM Applications .....	2-2
2.2 Creating an FND User .....	2-2
2.3 Assigning AOL Responsibilities to the User .....	2-3
2.4 Setting Profile Options at User, Application, Responsibility, and Site Levels .....	2-4
2.5 Creating a Workflow With Notifications.....	2-5
<b>3 Implementing Oracle Content Manager</b>	
3.1 Implementation Tasks .....	3-2
3.1.1 Setting Profile Options .....	3-2
3.1.2 Setting Up Users.....	3-3
3.1.3 Launching the OCM Administrator User Interface .....	3-5
3.1.4 Setting Environments for Translations .....	3-5
3.1.5 Assigning Folder Permissions.....	3-5

3.1.6	Setting Session Timeouts.....	3-6
3.1.7	Setting Up Intermedia Index Searches .....	3-7
3.2	Verifying Implementation.....	3-7
3.3	Diagnosing the Errors Using PL/SQL Logging.....	3-9
3.3.1	Setting Up PL/SQL Level Logging.....	3-9

## 4 Administering Oracle Content Manager

4.1	Understanding Content Types .....	4-2
4.2	Creating and Managing Content Types .....	4-3
4.3	Mapping Stylesheets to a Content Type.....	4-7
4.4	Creating and Managing Content Type Groups .....	4-8
4.5	Mapping File Extensions to Content Types.....	4-10
4.6	Data Security Overview.....	4-10
4.6.1	Access Control .....	4-10
4.6.2	Guidelines for Assigning Permissions .....	4-12
4.6.3	Setting Permissions and Exclusions for a Folder.....	4-13
4.6.4	Understanding Access Control with a Scenario .....	4-14
4.7	Categories .....	4-15
4.7.1	Category Access Control.....	4-16
4.7.2	Creating a Category and Setting Access Control.....	4-17

## A Oracle Content Manager Profile Options and Lookups

A.1	Setting Profile Options.....	A-2
A.2	Finding Responsibility ID Values .....	A-3
A.3	Oracle CRM Technology Foundation (JTT) Profile Options .....	A-4
A.4	Oracle Content Manager Profile Options .....	A-5
A.5	Oracle Content Manager Lookups.....	A-6

## B Seeded Content Types and Responsibilities

B.1	Seeded Content Types .....	B-2
B.1.1	IBC_IMAGE.....	B-2
B.1.2	IBC_STYLESHEET .....	B-2
B.1.3	IBC_BASIC_CL .....	B-3
B.1.4	IBC_CONTENT_BLOCK .....	B-3

B.1.5	IBC_COVER_LETTER .....	B-4
B.1.6	IBC_FILE.....	B-4
B.1.7	IBC_IMAGE .....	B-4
B.1.8	IBC_QUERY .....	B-5
B.2	Oracle Content Manager Customer UI Responsibilities .....	B-6
B.2.1	Seeded Responsibilities .....	B-6

## **C Content Item XML Formats**

C.1	XML Format Overview.....	C-2
C.2	Regular Expressions.....	C-2
C.3	An XML Example for Content Item.....	C-3

## **D Concurrent Programs**

D.1	Running Concurrent Programs .....	D-2
D.2	List of Concurrent Programs .....	D-3

## **E Oracle Content Manager API Reference**

E.1	API Overview .....	E-2
E.2	Content Manager API Interface Summary .....	E-2
E.3	Interface ContentItemMeta .....	E-2
E.3.1	Variables for Interface ContentItemMeta .....	E-3
E.3.2	Methods for Interface ContentItemMeta .....	E-3
E.4	Interface ContentItem .....	E-7
E.4.1	Fields Inherited.....	E-8
E.4.2	Methods Inherited by Interface ContentItem .....	E-8
E.4.3	Methods for Interface ContentItem .....	E-9
E.5	Content Manager API Class Summary .....	E-10
E.6	Class Attribute .....	E-11
E.6.1	Constructors for Class Attribute .....	E-11
E.6.2	Methods Inherited.....	E-11
E.6.3	Methods for Class Attribute .....	E-11
E.7	Class ComponentItemAttribute .....	E-12
E.7.1	Constructor for Class ComponentItemAttribute.....	E-13
E.7.2	Methods Inherited.....	E-13

E.7.3	Methods for Class ComponentItemAttribute .....	E-14
E.8	Class ContentCacheManager .....	E-14
E.8.1	Methods Inherited .....	E-14
E.8.2	Methods for Class ContentCacheManager .....	E-15
E.9	Class ContentDeliveryManager .....	E-20
E.9.1	Methods Inherited .....	E-20
E.9.2	Methods for Class ContentDeliveryManager .....	E-21
E.10	Class Rendition .....	E-38
E.10.1	Methods Inherited .....	E-38
E.10.2	Methods for Class Rendition .....	E-39
E.11	Exception Classes .....	E-40
E.11.1	Class ContentDeliveryException .....	E-40
E.11.2	Constructors for Class ContentDeliveryException .....	E-40
E.11.3	Fields Inherited .....	E-41
E.11.4	Methods Inherited .....	E-41
E.12	Data Security Manager .....	E-43

## Index

---

---

# Send Us Your Comments

## **Oracle Content Manager Implementation and Administration Guide, Release 11i**

**Part No. B13535-01**

Oracle welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: [appsdoc\\_us@oracle.com](mailto:appsdoc_us@oracle.com)
- FAX: (650) 506-7200 Attn: Oracle Applications Documentation Manager
- Postal service:  
Oracle Corporation  
Oracle Applications Documentation Manager  
500 Oracle Parkway  
Redwood Shores, CA 94065  
USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.



---

---

# Preface

Welcome to the Oracle Content Manager Implementation and Administration Guide, Release 11*i*.

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.
- Oracle Receivables and Oracle Content Manager.

If you have never used Oracle Receivables or Oracle Content Manager, Oracle suggests you attend one or more of the Oracle Applications training classes available through Oracle University.

- Oracle Self-Service Web Applications.

To learn more about Oracle Self-Service Web Applications, read the *Oracle Self-Service Web Applications Implementation Manual*.

- The Oracle Applications graphical user interface.

To learn more about the Oracle Applications graphical user interface, read the *Oracle Applications User's Guide*.

See [Other Information Sources](#) for more information about Oracle Applications product information.

## How To Use This Guide

The Oracle Content Manager Implementation and Administration Guide contains the information you must understand to use Oracle Content Manager. This guide contains the following chapters:

- [Chapter 1](#) provides a brief introduction to the new features in OCM.
- [Chapter 2](#) lists the tasks to verify if CRM applications are installed correctly.
- [Chapter 3](#) discusses the tasks to implement Oracle Content Manager.
- [Chapter 4](#) describes the tasks that an administrator must perform in Oracle Content Manager.

Apart from the above chapters, this guide includes appendixes that provide profile options and other settings, and some useful reference information.

## Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

### Accessibility of Code Examples in Documentation

JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

## Other Information Sources

You can choose from many sources of information, including documentation, training, and support services, to increase your knowledge and understanding of Oracle Content Manager.

If this guide refers you to other Oracle Applications documentation, use only the Release 11i versions of those guides.

## Online Documentation

All Oracle Applications documentation is available online (HTML or PDF).

- **PDF Documentation**- See the Documentation CD provided with each release for current PDF documentation for your product. This Documentation CD is also available on *OracleMetaLink* and is updated frequently.
- **Online Help** - You can refer to Oracle iHelp for current HTML online help for your product. Oracle provides patchable online help, which you can apply to your system for updated implementation and end user documentation. No system downtime is required to apply online help.
- **11i Release Content Document** - Refer to the Release Content Document for new features listed release. The Release Content Document is available on *OracleMetaLink*.
- **About document** - Refer to the About document for patches that you have installed to learn about new documentation or documentation patches that you can download. The new About document is available on *OracleMetaLink*.

## Related Guides

Oracle Content Manager shares business and setup information with other Oracle Applications products. Therefore, you may want to refer to other guides when you set up and use Oracle Content Manager.

You can read the guides online by choosing Library from the expandable menu on your HTML help window, by reading from the Oracle Applications Document Library CD included in your media pack, or by using a Web browser with a URL that your system administrator provides.

If you require printed guides, you can purchase them from the Oracle Store at <http://oraclestore.oracle.com>.

## Guides Related to All Products

### Oracle Applications User's Guide

This guide explains how to enter data, query, run reports, and navigate using the graphical user interface (GUI). This guide also includes information on setting user profiles, as well as running and reviewing reports and concurrent processes.

You can access this user's guide online by choosing "Getting Started with Oracle Applications" from any Oracle Applications help file.

## **Guides Related to This Product**

### **Oracle iStore Implementation Guide**

Oracle iStore is an E-commerce application that enables companies to build and manage global Internet storefronts, which can be personalized.

Integrating with OCM, the Oracle iStore merchants can access all the features of OCM and also:

- Manage content in global stores by keeping the content separate between content owners and store administrators.
- Preview content from the context of the store.
- Ensure brand consistency across all sites.

### **Oracle Partner Management Implementation Guide**

Oracle Partner Management is a Marketing and Partnering application enabling partner business processes and applications that support partner collaboration.

When vendors create a program, they create a summary listing the details of the program. The summary is an HTML based content, which helps the partners to understand the program requirements, benefits, and so on. The content that is used to create the program is stored and retrieved from OCM.

### **Oracle One-to-One Fulfillment Implementation Guide**

The Oracle One-to-One Fulfillment is a product interaction center that enables Oracle E-Business Suite applications to use a central mechanism for distributing collateral to customers. One-to-One Fulfillment provides the capability to fulfill content across multiple channels, such as e-mail, fax, and print.

Integrating with OCM, Oracle One-to-One Fulfillment enables the following features for its fulfillment users to:

- Support the templates stored in OCM
- Setup different templates for E-mail, fax, and print type requests
- Create and upload the content into OCM.

## **Oracle Marketing Implementation Guide**

Oracle Marketing enables today's marketers to market more effectively. It provides automation and tools for the entire marketing process - from the initial analysis to decide what kind of marketing is required and whom to target, to the actual campaign planning, budget management, list creation and content management, to automated multi-channel execution, and monitoring.

Integrating with OCM, Oracle Marketing empowers the marketer to fully leverage all the features of OCM, thereby bringing in brand consistency, reduction in costs, and better management of marketing assets. In addition, users can:

- Access content across all channels of execution - e-mail, fax, print, and Web.
- Fulfill content from multi-channels.
- Keep track of the content that is used for different campaigns.
- Collaborate content to be shared across the Sales and Marketing users of the enterprise.
- Associate content to Products and use it for recommending content in campaigns.

## **Oracle Proposals Implementation Guide**

Oracle Proposals is a Sales application that enables sales representatives to dynamically generate proposals from templates that contain proposal elements such as cover letters, product descriptions, pricing, and terms and conditions.

Integrating with OCM, Oracle Proposals enables the following features to its users and administrators:

- Organize content in folder hierarchies.
- Store, approve, and version the content.
- Search and select content from OCM repository, which is created by sales representatives, marketers, and other users.
- Maintain content translations.

## **Oracle *interMedia* User's Guide and Reference**

This guide and reference provides information about Oracle *interMedia*. This product enables Oracle<sup>9i</sup> to store, manage, and retrieve geographic location information, images, audio, video, or other heterogeneous media data in an integrated fashion with other enterprise information. Oracle Trading Community

Architecture Data Quality Management uses *interMedia* indexes to facilitate search and matching.

## **Oracle Self–Service Web Applications Implementation Guide**

This manual contains detailed information about the overview and architecture and setup of Oracle Self–Service Web Applications. It also contains an overview of and procedures for using the Web Applications Dictionary.

## **Installation and System Administration**

### **Oracle Applications Concepts**

This guide provides an introduction to the concepts, features, technology stack, architecture, and terminology for Oracle Applications Release 11*i*. It provides a useful first book to read before an installation of Oracle Applications. This guide also introduces the concepts behind Applications-wide features such as Business Intelligence (BIS), languages and character sets, and Self-Service Web Applications.

### **Installing Oracle Applications**

This guide provides instructions for managing the installation of Oracle Applications products. In Release 11*i*, much of the installation process is handled using Oracle Rapid Install, which minimizes the time to install Oracle Applications and the Oracle technology stack by automating many of the required steps. This guide contains instructions for using Oracle Rapid Install and lists the tasks you must perform to finish your installation. You should use this guide in conjunction with individual product user guides and implementation guides.

### **Oracle Applications Implementation Wizard User Guide**

If you are implementing more than one Oracle product, you can use the Oracle Applications Implementation Wizard to coordinate your setup activities. This guide describes how to use the wizard.

### **Upgrading Oracle Applications**

Refer to this guide if you are upgrading your Oracle Applications Release 10.7 or Release 11.0 products to Release 11*i*. This guide describes the upgrade process and lists database and product-specific upgrade tasks. You must be either at Release 10.7 (NCA, SmartClient, or character mode) or Release 11.0, to upgrade to Release 11*i*. You cannot upgrade to Release 11*i* directly from releases prior to 10.7.

## **“About” Document**

For information about implementation and user documentation, instructions for applying patches, new and changed setup steps, and descriptions of software updates, refer to the “About” document for your product. “About” documents are available on Oracle *MetaLink* for most products starting with Release 11.5.8.

## **Maintaining Oracle Applications**

Use this guide to help you run the various AD utilities, such as AutoUpgrade, AutoPatch, AD Administration, AD Controller, AD Relink, License Manager, and others. It contains how-to steps, screenshots, and other information that you must run the AD utilities. This guide also provides information on maintaining the Oracle applications file system and database.

## **Oracle Applications System Administrator’s Guide**

This guide provides planning and reference information for the Oracle Applications System Administrator. It contains information on how to define security, customize menus and online help, and manage concurrent processing.

## **Oracle Alert User’s Guide**

This guide explains how to define periodic and event alerts to monitor the status of your Oracle Applications data.

## **Oracle Applications Developer’s Guide**

This guide contains the coding standards followed by the Oracle Applications development staff and describes the Oracle Application Object Library components that are needed to implement the Oracle Applications user interface described in the *Oracle Applications User Interface Standards for Forms-Based Products*. This manual also provides information to help you build your custom Oracle Forms Developer forms so that the forms integrate with Oracle Applications.

## **Oracle Applications User Interface Standards for Forms-Based Products**

This guide contains the user interface (UI) standards followed by the Oracle Applications development staff. It describes the UI for the Oracle Applications products and how to apply this UI to the design of an application built by using Oracle Forms.

## Other Implementation Documentation

### Oracle Applications Product Update Notes

Use this guide as a reference for upgrading an installation of Oracle Applications. It provides a history of the changes to individual Oracle Applications products between Release 11.0 and Release 11*i*. It includes new features, enhancements, and changes made to database objects, profile options, and seed data for this interval.

### Oracle Workflow Administrator's Guide

This guide explains how to complete the setup steps necessary for any Oracle Applications product that includes workflow-enabled processes, as well as how to monitor the progress of runtime workflow processes.

### Oracle Workflow Developer's Guide

This guide explains how to define new workflow business processes and customize existing Oracle Applications-embedded workflow processes. It also describes how to define and customize business events and event subscriptions.

### Oracle Workflow User's Guide

This guide describes how Oracle Applications users can view and respond to workflow notifications and monitor the progress of their workflow processes.

### Oracle Workflow API Reference

This guide describes the APIs provided for developers and administrators to access Oracle Workflow.

### Oracle Applications Flexfields Guide

This guide provides flexfields planning, setup and reference information for the Oracle Content Manager implementation team, as well as for users responsible for the ongoing maintenance of Oracle Applications product data. This guide also provides information on creating custom reports on flexfields data.

### Oracle eTechnical Reference Manuals

Each eTechnical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for a specific Oracle Applications product. This information helps you convert data from your existing applications, integrate Oracle Applications data with non-Oracle applications, and write custom reports for Oracle Applications products. Oracle eTRM is available on Oracle *Metalink*.

## Oracle Applications Message Manual

This manual describes all Oracle Applications messages. This manual is available in HTML format on the documentation CD-ROM for Release 11*i*.

## Training and Support

### Training

Oracle offers a complete set of training courses to help you and your staff master Oracle Content Manager and reach full productivity quickly. These courses are organized into functional learning paths, so you take only those courses appropriate to your job or area of responsibility.

You have a choice of educational environments. You can attend courses offered by Oracle University at any one of our many education centers, you can arrange for our trainers to teach at your facility, or you can use Oracle Learning Network (OLN), Oracle University's online education utility. In addition, Oracle training professionals can tailor standard courses or develop custom courses to meet your needs. For example, you may want to use your organization structure, terminology, and data as examples in a customized training session delivered at your own facility.

### Support

From on-site support to central support, our team of experienced professionals provides the help and information you need to keep Oracle Content Manager working for you. This team includes your technical representative, account manager, and Oracle's large staff of consultants and support specialists with expertise in your business area, managing an Oracle server, and your hardware and software environment.

### Oracle*MetaLink*

*OracleMetaLink* is your self-service support connection with web, telephone menu, and e-mail alternatives. Oracle supplies these technologies for your convenience, available 24 hours a day, 7 days a week. With *OracleMetaLink*, you can obtain information and advice from technical libraries and forums, download patches, download the latest documentation, look at bug details, and create or update TARs. To use *MetaLink*, register at <http://metalink.oracle.com>.

**Alerts:** You should check *OracleMetaLink* alerts before you begin to install or upgrade any of your Oracle Applications. Navigate to the Alerts page as follows:

Technical Libraries/ERP Applications/Applications Installation and Upgrade/Alerts.

**Self-Service Toolkit:** You may also find information by navigating to the Self-Service Toolkit page as follows: Technical Libraries/ERP Applications/Applications Installation and Upgrade.

## Do Not Use Database Tools to Modify Oracle Applications Data

*Oracle STRONGLY RECOMMENDS that you never use SQL\*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle Applications data unless otherwise instructed.*

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL\*Plus to modify Oracle Applications data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle Applications tables are interrelated, any change you make using Oracle Applications can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

When you use Oracle Applications to modify your data, Oracle Applications automatically checks that your changes are valid. Oracle Applications also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL\*Plus and other database tools do not keep a record of changes.

## About Oracle

Oracle develops and markets an integrated line of software products for database management, applications development, decision support, and office automation, as well as Oracle Applications, an integrated suite of more than 160 software modules for financial management, supply chain management, manufacturing, project systems, human resources and customer relationship management.

Oracle products are available for mainframes, minicomputers, personal computers, network computers and personal digital assistants, allowing organizations to integrate different computers, different operating systems, different networks, and

even different database management systems, into a single, unified computing and information resource.

Oracle is the world's leading supplier of software for information management, and the world's second largest software company. Oracle offers its database, tools, and applications products, along with related consulting, education, and support services, in over 145 countries around the world.



---

---

## Introduction

This chapter discusses the key features of Oracle Content Manager. Topics in this chapter include:

- [Section 1.1, "Oracle Content Manager Overview"](#)
- [Section 1.2, "New in this Release"](#)

## 1.1 Oracle Content Manager Overview

Oracle Content Manager (OCM) is a key component of the Sales, Marketing and E-Commerce applications. It enables content to be stored and managed in a central repository, thereby forming a single source of truth for accurate and consistent content.

In OCM, a content is a structured data that is managed through a lifecycle that includes creating, approving, translating, and publishing. A content can be a text file, an image, an audio, or a video file. For example, a product manager could describe the product's functionality in a text field. To market the product, a marketing person can create an attractive video file using the same product description. To ensure that they both access the same content, it must be available in a central repository.

OCM integrates with Oracle E-Business Suite and enables seamless transition of content to and from Oracle E-Business Suite applications. Integration also facilitates single user sign-on to associate content (product images, white papers, and newsletters) with any E-Business Objects (store sections, inventory products, and marketing campaigns).

### 1.1.1 Oracle Content Manager Features

The key features of OCM include:

- Store data in the central repository
- Create content based on templates
- Control multiple versions of the same content
- Approve content based on the workflow notifications
- Maintain translated versions of the content item
- Maintain renditions of the same content in different formats
- Support content reuse
- Support flexible presentation of content through many channels such as print, e-mail, and fax
- Control users accessing the content by restricting them with permissions
- Preview the content before publishing using stylesheets
- Create many content items at once by uploading a zip file

## 1.2 New in this Release

This guide describes the OCM functionality in the Oracle E-Business Suite 11.5.10 release. If you are implementing this product prior to the release, using product minipacks or family packs, some new functionality may be dependent on integration with other Oracle products. Please consult *OracleMetaLink* for relevant product patches and documentation.

The following new features and enhancements have been added to OCM in this release:

- [Robust Folder Access and Data Security Enhancements](#)
- [Translation Approvals](#)
- [Bulk Import of Files](#)
- [Accessing Multiple Version of Compound Subitemss](#)
- [Grouping of Content Types](#)
- [Custom Attribute LOVs](#)
- [Content Library](#)
- [Explicit Locking of Items](#)
- [Intermedia Text Based Search](#)
- [Flexible URL Encryption](#)
- [Rendition](#)
- [Category](#)
- [New Profiles](#)

### **Robust Folder Access and Data Security Enhancements**

Content creators must organize large amounts of content to easily retrieve and restrict others from accessing it. OCM supports this feature by allowing content creators to store the content in folders, which are secured by access control. You can access and view the contents in a folder only if you have the necessary permissions. The access control can be modified anytime.

### **Translation Approvals**

In the previous release, you could create content items in the base language and approve them; translations did not go through an approval process and only

content items had to be approved. In this release, along with content items, the translations must also be approved.

### **Bulk Import of Files**

OCM supports the ability to import files and images into the repository. File extensions can be mapped to content types in order to upload files into the appropriate content type.

### **Accessing Multiple Version of Compound Subitems**

In the previous release, you could only access the live version of the subitem, while creating a compound item. Now, you can define a compound item to point to a specific version of the approved subitem.

### **Grouping of Content Types**

As more users use OCM, it becomes difficult to manage many content types. OCM supports the ability to group the content types into content type groups. By creating groups, you can restrict the users (while creating a content item) to select a content type only from the subscribed group.

### **Custom Attribute LOVs**

While defining content type attributes, you can define custom LOVs. For example, you can now create custom LOVs for country, salutation, and so on.

### **Content Library**

In the previous release, you would browse the content library based on content types. Now, you can create categories for the published content items and search for content items based on the category.

### **Explicit Locking of Items**

This is a new feature that enables you to lock the content items in order to prevent other users from updating them.

### **Intermedia Text Based Search**

In the previous release, you could only do simple searches based on the name and description of the content items. In this release, you are provided an intermedia search. This is a text based search that allows you to search content items based on the content item attributes. For example, you can search for all white papers containing the text 'technical', where technical is an attribute.

## Flexible URL Encryption

It is important for content items with a URL posted on the web to be encrypted. In the previous release, all the content items were either encrypted or none were encrypted. In this release, the administrator can decide at the content type level, if all content items of the content type are to be encrypted or not.

## Rendition

Rendition is a mechanism by which you can upload the same content in different formats such as pdf, doc, and rtf. In the previous release, you could do this only through the APIs. In this release, you can upload renditions for a content item through the user interface.

## Category

A category is used to logically group a set of content items together. While folders are work areas to organize content items, categories reflect more of a business group. Many organizations define content that can be classified into one or more categories. Categories also have a hierarchy of subcategories for finer classification. You can browse the library based on the categories.

## New Profiles

The following profile options are new in OCM:

- IBC: Content Type Group
- IBC: Enforce Expiration Date
- IBC: Enforce Available Date
- IBC: Debug Output Directory
- IBC: Enable Encryption
- IBC: Enable Debug
- IBC: Default Home Folder
- IBC: Debug Type
- IBC: Display Base Lang
- IBC: Enable Audit Log
- IBC: Use Access Control
- IBC: Customize Approval Workflow

For more information on profiles, see [Section A, "Oracle Content Manager Profile Options and Lookups"](#) in this guide.

---

---

## Verifying Dependencies

This chapter gives instructions on setting up a user, setting up profile options, and setting up workflows. Sections in this chapter include:

- [Section 2.1, "Verifying CRM Applications"](#)
- [Section 2.2, "Creating an FND User"](#)
- [Section 2.3, "Assigning AOL Responsibilities to the User"](#)
- [Section 2.4, "Setting Profile Options at User, Application, Responsibility, and Site Levels"](#)

## 2.1 Verifying CRM Applications

Oracle Content Manager is dependent on the Oracle Applications that provide the underlying technology stack, schema, and structure.

OCM depends on the following modules:

- Oracle CRM Technology Foundation
- 11.5.10 Oracle Application (OA) Framework

You must complete the following tasks successfully to verify that the CRM applications are working properly:

- [Section 2.2, "Creating an FND User"](#)
- [Section 2.3, "Assigning AOL Responsibilities to the User"](#)
- [Section 2.4, "Setting Profile Options at User, Application, Responsibility, and Site Levels"](#)
- [Section 2.5, "Creating a Workflow With Notifications"](#)

## 2.2 Creating an FND User

Use this procedure to create an FND user in the Application Object Library. For more details on creating users, see *Oracle Applications System Administrator's Guide*.

### Prerequisites

None

### Steps

1. Log into Oracle Forms with the System Administrator responsibility and navigate to Security > User > Define.
2. In the User window, enter a unique user name in the User Name field.
3. Enter a password in the Password field and reenter the password for verification.
4. Select a name from the Person LOV.
5. In the Responsibilities subtab, select the CRM HTML Administration responsibility from the drop-down list.
6. Save the new user.

To verify that the user setup is successful, perform the following steps:

7. Log in to the Personal Home page as the newly created user.
8. Enter the password when prompted.

You should now be able to access the Personal Home Page for this user.

---

---

**Note:** For the user to access the HTML applications, you must set additional profile options as detailed in the *Implementing Oracle CRM Application Foundation Guide*.

---

---

## 2.3 Assigning AOL Responsibilities to the User

A responsibility defines an application user's current privileges while working with Oracle Applications. When an application user logs in, the user needs to select a responsibility that grants certain privileges, specifically:

- The functions that the user may access. Functions are determined by the menu options assigned to the responsibility.
- The concurrent programs, such as reports, that the user may run.
- The application database accounts such as forms, concurrent programs, and reports.

You cannot delete a responsibility because this information helps you to create an audit trail. You can deactivate a user's responsibility at any time by setting the End Date to the current date. If you wish to activate the responsibility for the user, then change the End Date to a date after the current date, or clear the value in the End Date.

Use the following procedure to assign AOL responsibilities to the user.

For more information on employee management, see *Managing People Using Oracle HRMS (US)*.

### Prerequisites

An FND user must exist.

### Steps

1. Log into Oracle Forms with the System Administrator responsibility and navigate to Security > User > Define.

2. In the Responsibilities tab, select a responsibility from the List of Values (LOV).
3. Define the Effective Dates for the responsibility to be active.
4. Save your work.

## 2.4 Setting Profile Options at User, Application, Responsibility, and Site Levels

A user profile is a set of changeable options that affect the way the application looks and behaves. As a System Administrator, you can control how Oracle Applications operate by setting user profile options to the required values. You can set user profile options at four different levels: site, application, responsibility, and user.

---

---

**Note:** You can set default values for profiles at Site level for the application. If you do not set a default value for a profile at any of the four levels, then you get errors when you use Oracle Forms, run reports, or run concurrent requests.

---

---

Use the following steps to set profile options.

### Prerequisites

An FND User must exist.

### Steps

1. Log into Oracle Forms with the System Administrator responsibility and navigate to Security > Profile > System.
2. In the Find System Profile Values window, enter the profile option you want to set and click **Find**.

The System Profile Values window appears listing all the profile options that you searched for.

3. Set at least one of the following:
  - a. Set the Site value.

This field displays the current value for all users at the installation site.
  - b. Set the Application value.

This field displays the current value for all users working under responsibilities owned by the application identified in the Find Profile Values region.

- c. Set the Responsibility value.

This field displays the current value for all users working under the responsibility identified in the Find Profile Values region.

- d. Set the User value.

This field displays the current value, if set, for the application user identified in the Find Profile Values region.

4. Save your work.

## 2.5 Creating a Workflow With Notifications

In OCM, the approvers receive a notification when a content item has been submitted for approval, and when a content item is waiting for translation approval. OCM uses Oracle Workflow to send notifications to the designated approvers. Oracle Workflow must be configured and the Workflow Mailer must be set up for the notifications to be sent as e-mails. The users can also log into the Self-Service Applications to view workflow notifications. For more information on configuring Oracle Workflow, see the *Oracle Workflow Administrator's Guide*.

Use this procedure to create and run workflow notifications. For more information on workflows, see *Oracle Workflow User's Guide*.

### Prerequisites

None

### Steps

1. Log into Oracle Forms with the Workflow Administrator responsibility and navigate to Workflow > Launch Processes.
2. In the Launch Processes window, in the Item Type column, click **Document Management**.

If you have renamed the item types, then this option appears in the Internal Name column as WFDM and the Initiate Workflow-WFDM page opens.

3. Enter values in the following fields:

- **Item Key:** Enter your name plus a sequence number (for example, jdoe1001).
- **User Key:** You can copy the value from the Item Key field.
- **Process Name:** Enter Document Review.
- **Process Owner:** Your logged in user name populates automatically.
- **Send Document:** Leave blank.
- **Document Owner:** Select a valid resource name.
- **Document Reviewer:** Select a value from the list of values.
- **Comments:** Enter Workflow Verification.
- **Response Document:** Leave blank.

**4. Click OK.**

The Activities List page opens to show workflow statuses. The status of the workflow you just initiated should be Active.

- 5. If there is an error in the status of the workflow, then click **Exception** in the Result column to see an explanation of the error.**
- 6. Click **View Diagram** to see a graphical representation of the workflow process.**  
Leave the View Diagram window open as you continue to verify the workflow.
- 7. Save your work.**
- 8. To verify that the Workflow notification is set, login to your Personal Home Page.**
  - a. In the list of Self-Service applications, choose the Workflow User Web Application responsibility.**
  - b. In the Navigator, choose Workflow > Find Notifications.**
  - c. In the Type field, enter Document Management.**
  - d. In the To field, enter the document reviewer.**
  - e. Click **Find**.**
  - f. Click **Subject** to open the notification.**  
If you see the notification, then the workflow is set up correctly.
  - g. Click **Approve** to return to the Worklist window.**

For further verification, return to the View Diagram window that you opened earlier. Click **Reload** in the browser window to refresh the contents of the window. After the workflow process is complete, you can see a green line from the Start icon to the End (Approve) icon.



---

---

# Implementing Oracle Content Manager

This chapter discusses the tasks to implement Oracle Content Manager. Sections in this chapter include:

- [Section 3.1, "Implementation Tasks"](#)
- [Section 3.2, "Verifying Implementation"](#)
- [Section 3.3, "Diagnosing the Errors Using PL/SQL Logging"](#)

## 3.1 Implementation Tasks

To implement OCM successfully, Oracle strongly recommends that you execute the following tasks in the order that they are listed below:

1. [Section 3.1.1, "Setting Profile Options"](#)
2. [Section 3.1.2, "Setting Up Users"](#)
3. [Section 3.1.3, "Launching the OCM Administrator User Interface"](#)
4. [Section 3.1.4, "Setting Environments for Translations"](#)
5. [Section 3.1.5, "Assigning Folder Permissions"](#)
6. [Section 3.1.6, "Setting Session Timeouts"](#)
7. [Section 3.1.7, "Setting Up Intermedia Index Searches"](#)

---

---

**Note:** Before implementing Oracle Content Manager (OCM), the functional implementation team in your organization must analyze your business requirements and setup the implementation steps accordingly.

---

---

### 3.1.1 Setting Profile Options

Profiles control the behavior of all the Oracle applications. All Oracle application profile options begin with a two or three letter code. The OCM profiles begin with the prefix IBC.

For more information on setting Oracle application profile options, see *Oracle Applications System Administrator's Guide*.

For more information on JTT profile options, see [Section A.3, "Oracle CRM Technology Foundation \(JTT\) Profile Options"](#).

Use the following procedure to set the OCM profiles.

#### Prerequisites

None

#### Steps

1. To set CRM Technology Foundation profile options, log into Oracle Forms with System Administrator responsibility and set the following profiles at the User level:

2. To set OCM profile options, log into Oracle Forms with the System Administrator responsibility and navigate to Profile > System.
  - a. Select a level to set the profile option.

You can select either Site, Application, Responsibility, or User.
  - b. In the Profile field, enter the profile name or use wild cards to search for OCM profiles. For example, you can enter "IBC%".
  - c. Click **Find**.

The System Profile Values window appears.
  - d. Select a profile and select the required value.
  - e. Save your work.

### 3.1.2 Setting Up Users

You can categorize users in your organization according to the function that they will perform in OCM. The users can be administrators, content item creators, approvers, translators, and so on.

The users can access the application to create, edit, approve, translate, and delete content depending on their responsibilities. In OCM, the administrator is called the Content Super User responsibility. Apart from this, other responsibilities are also seeded in OCM. See [Appendix B.2.1, "Seeded Responsibilities"](#) for more details.

#### Setting Up the Content Super User

You must create an administrator user or an Content Super User who will have all the privileges in the OCM application. The menu assigned to the IBC Super User responsibility is called Oracle Content Manager Root Menu, which is IBC\_ROOT\_MENU\_LVL1.

#### Setting Up Multiple Users for OCM

Multiple users can access OCM having specific responsibilities, depending upon their roles in the organization. Depending on your business requirements, you can create users and assign responsibilities to them. Examples of such roles are Content Creators and Content Users.

With multiple users having access only to their specific functions, management of users becomes more efficient and secure. You must create users and assign them the required responsibilities. See [Appendix B.2.1, "Seeded Responsibilities"](#) for more information.

Apart from assigning appropriate responsibilities, as a Content Super User, you must grant folder permissions to these users. See [Section 3.1.5, "Assigning Folder Permissions"](#).

---

---

**Note:** You must explicitly grant folder permissions even for users with Super User responsibility. All permissions must be explicitly granted on the folders. No privilege is automatically entitled to the super user.

---

---

For information on creating responsibilities and mapping them to tabs, see the *Oracle Applications System Administrator's Guide*.

Use the following procedure to create users and assign responsibilities to them.

### Prerequisites

None

### Steps

1. Log into Oracle Forms with the System Administrator responsibility and navigate to Security > Users > Open.
2. Enter the user's name in the User Name field.
3. In the Password field, enter the user's password and press Enter.
4. Reenter the password and press Enter.
5. In the Responsibilities tab, select an appropriate responsibility from the Responsibility LOV.
6. Additionally, to see workflow notifications in Self-Service applications, select the Workflow Administrator responsibility.
7. Click **Save**.

### 3.1.3 Launching the OCM Administrator User Interface

Typically, users access OCM UI to complete their assigned tasks. To access the UI, users must:

- Verify that the cookies are enabled for the browser.  
For any information related to the browser, see your browser documentation.
- Log into `http://<host>:<port>/OA_HTML/jtfllogin.jsp` using the user name and password that the system administrator has set for you. If you find any error, then try clearing the cookies or see your browser documentation for details.

### 3.1.4 Setting Environments for Translations

The users can use the Update Translations feature to view and update the content items that are marked for translation. To enable this feature in OCM, you must setup the environment for translations.

To be able to translate and view translations, it is recommended that you use the UTF8 character set on all three tiers: database server, middle tier application server, and browser client. Ensure that the character set encoding of the session language is a super set of the base language and the translated language. For instance, the Western European encoding may enable translations between any of the Western European languages such as English and French, but may not work while translating to other languages such as Japanese.

For more information on setting up your environment for translation, see the *Oracle Application System Administrator's Guide*.

### 3.1.5 Assigning Folder Permissions

In OCM, by default, the data security is enabled when the profile IBC: Use Access Control is set to Y. The Content Super User must create and assign the access control for the folders.

---

---

**Note:** Initially, only the Content Super User can access the Root folder. To assign permissions to other users, as an Content Super User, you must add the required permissions to yourself and then grant access permissions to other users. If IBC: Use Control profile is set to N, then there is no access control set for OCM. All the users have all the permissions and anyone can access any data.

---

---

Use the following procedure to assign access control to the folders.

### Prerequisites

None

### Steps

1. Log into Oracle Content Manager as Content Super User and navigate to Content > My Content.
2. To create a folder, click **Create Folders**.
3. Enter a folder name.
4. To override the parent folder's access control, select the Override box.  
The fields in the Permissions and Exclusions regions are enabled.
5. To add rows in the Permissions and Exclusions region, click **Add Another Row**.
6. In the Type LOV, select either User, Group, Responsibility, or Global.
7. In the Name field, enter a user name or click the Search icon.
8. To grant and revoke permissions to the user, select the appropriate box in the Permissions and Exclusions regions.
9. Optionally, to inherit parent folder access control, select the Inherit box.  
The fields in the Permissions and Exclusions regions are disabled.
10. Click **Apply**.

## 3.1.6 Setting Session Timeouts

OCM is built on two frameworks - the OA framework and the JTT framework. The session timeout in the OA framework must always be *greater* than the session timeout in the JTT framework.

### JTT Framework

The JTT session timeout is configured at the JTT Administrative Console. Login into Forms and navigate to Settings > System > Sessions. Enter a value in the Expires if idle for (mins) field. The default value is 30 minutes.

### OA Framework

The OA session timeout is driven by the `session.timeout` directive in the `zone.properties` file of the Apache configuration. The default value is 60 minutes.

Set this value to a number greater than the JTT Framework session timeout value.

## 3.1.7 Setting Up Intermedia Index Searches

Use the following procedure to periodically rebuild intermedia index and optimize the intermedia index.

### Prerequisites

Run the Concurrent Manager.

### Steps

1. Log into Oracle Forms with Oracle Content Manager responsibility and navigate to Requests > Run.
2. Select OCM InterMedia Index Synchronizing for Attribute Bundles from the LOV.
3. Click **Schedules** and select an appropriate frequency. For example, select one hour to run the concurrent program every hour.
4. Click **Submit**.
5. To optimize the intermedia index, select OCM InterMedia Index Optimizing operation for Attribute Bundle from the LOV and follow step 3 and step 4, as listed above.

## 3.2 Verifying Implementation

In this section, a case study is given to describe the tasks to verify if OCM is implemented successfully.

Create the following users and assign them the responsibilities and permissions as listed in [Table 3-1](#) for the /Root/Common folder.

**Table 3-1** *User Name, Responsibility, and Permissions*

User Name	Responsibility	Permissions
Anna	Content Creator	Manage Folders

**Table 3–1 User Name, Responsibility, and Permissions**

User Name	Responsibility	Permissions
John	Content Creator	Manage Items
Joseph	Content Creator	Approve Item
Michael	Content Creator	Translate Item
Nicole	Content Creator	Approve Translation
Terry	Content Creator	Read Item

Table 3–2 lists the tasks that you must perform to verify the implementation.

**Table 3–2 Verifying Implementation Task Sequence**

No.	Task	Reference
1	Log in as Content Super User to create users and assign responsibilities as specified in Table 3–1.	Section 3.1.2, "Setting Up Users".
2	Select /Root/Common folder and set the appropriate permissions to the users as listed in Table 3–1.	Section 4.6.3, "Setting Permissions and Exclusions for a Folder".
3	Log in as John Create a content item, select the Translation Required box, and submit it for approval.	For details, see the <i>Oracle Content Manager User Guide</i> .
4	Joseph receives a notification and approves the content item. As the content item is to be translated, a notification is be sent to Michael to translate the content item.	For details, see the <i>Oracle Content Manager User Guide</i> .
5	Log in as Michael and approve the content item submitted by John.	For details, see the <i>Oracle Content Manager User Guide</i> .
6	Nicole receives a notification. Log in as Nicole. Approve the translation.	For details, see the <i>Oracle Content Manager User Guide</i> .
7	Log in as Terry. You must be able to browse the /Root/Common folder but should not be able to a create content item. You must be only view the content item as you have the Read Item permission.	For details, see the <i>Oracle Content Manager User Guide</i> .

## 3.3 Diagnosing the Errors Using PL/SQL Logging

OCM enables the user to log debug messages. This feature creates separate log files for different users. User can use the log file to diagnose the problems in the application.

For more information on diagnosing, see *Oracle Diagnostics Pack*.

### 3.3.1 Setting Up PL/SQL Level Logging

You must set PL/SQL profiles for the application to log the error messages for different users in different files. The log file is created in the directory as specified in the profile IBC: Debug Log Directory. The syntax of the file name is IBC\_<Username>.log.

Use the following procedure to set up PL/SQL level logging.

#### Prerequisites

PL/SQL profile values must be set.

#### Steps

1. Log into Oracle Forms as the System Administrator and navigate to Profile > System.
2. Select a user.
3. Set the profile option IBC: Enable Debug to Yes at the User level.
4. Set IBC: Debug Log Directory profile at the Site level to a directory that is accessible by the database server.

This is the `utl_file_dir` parameter. To determine this value, `SELECT * from v$parameter where name like 'utl_file_dir'`.

5. Set the `init.ora` `UTL_FILE_DIR` parameter to point to the same directory.
6. Save the settings.



---

---

# Administering Oracle Content Manager

This chapter discusses the tasks that an administrator must perform in Oracle Content Manager. Sections in this chapter include:

- [Section 4.1, "Understanding Content Types"](#)
- [Section 4.2, "Creating and Managing Content Types"](#)
- [Section 4.3, "Mapping Stylesheets to a Content Type"](#)
- [Section 4.4, "Creating and Managing Content Type Groups"](#)
- [Section 4.5, "Mapping File Extensions to Content Types"](#)
- [Section 4.6, "Data Security Overview"](#)
- [Section 4.7, "Categories"](#)

## 4.1 Understanding Content Types

The content types define the structure of a content item through a set of attributes. They are the main tenet of Oracle Content Manager (OCM). Administrators create these content types to ensure that content items are consistent, wherever the same content types are used.

A content type consists of simple attributes, subitems, and compound content types. Examples of content type are: Product Name, Description, Feature Highlights, Overview, Product Information Guide (attachment), and Product Release Date.

### Simple Attributes

Attributes can be of type Extended, Pull-down, and Attachments. For example, an Image Type can have attributes such as image attachment, image height, width, and alternate text.

- The extended attributes are data types such as Boolean, Date, Number, HTML, Text, and URL. While creating a content item of attribute type:
  - Boolean - is displayed as a check box.
  - Date - appears as a date picker.
  - Number - accepts numbers.
  - HTML - is displayed as rich text html editor.
  - Text - accepts characters.
  - Attribute URL - accepts a URL.
- The pull-down attributes appear as values in the drop-down list.
- The attachment attribute enables you to upload attachments for the content item.

### Subitems

Subitems are individual content items based on their corresponding content types. For example, a Sales Kit can consist of White Paper and Datasheet as subitems.

### Compound Content Types

Apart from extended, pull-down list, and attachment attributes, compound content type is another attribute. A content type can have other content types as its components. Such a content type is a compound content type. For example, a Product Brochure content type can have other content types such as Product

Presentation, Product White paper, and Product Demo as its components. This makes Product Brochure a compound content type.

### **Minimum and Maximum Instances**

While creating a compound content type, you can specify the minimum and maximum number of content items that can be added for the compound content item.

If you specify the minimum value to be zero, then the subitem of this type is not mandatory. If you specify a number greater than zero, then while creating the compound content item, it is mandatory to add that many number of content items of this type as subitems.

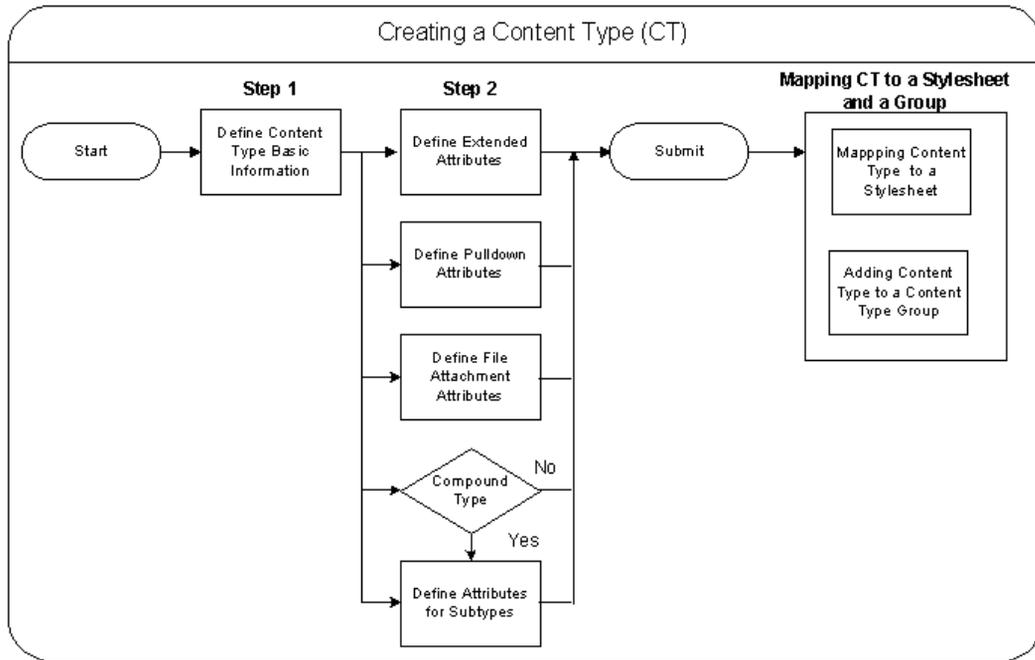
If you specify a number in the maximum field, then that many content items can be added for the compound content item. If you leave it blank, then while creating the compound content item, any number of content items of this type can be added as subitems.

## **4.2 Creating and Managing Content Types**

Only System Administrators and Content Super Users can create and manage content types. [Figure 4-1](#) depicts the process to create a content type.

It is recommended that you understand the requirements of your organization before you define the attributes for the content type. This is because, after a content type is used to create a content item, you cannot update it. If you have to update such a content type, you must first duplicate it and update the duplicate copy.

**Figure 4–1 Creating a Content Type**



Creating content types is a two step process:

1. In the first step, provide the basic details of the content type such as the name, owner, and the default stylesheet.
2. In the second step, define its attributes.

After creating a content type, you can map it to a stylesheet and also add it to a content type group. For details, see sections [Section 4.3, "Mapping Stylesheets to a Content Type"](#) and [Section 4.4, "Creating and Managing Content Type Groups"](#).

Use the following procedure to create, update, and duplicate content types.

### Prerequisites

None

## Steps

1. Log into Oracle Content Manager as the Content Super User and navigate to Content Types.
2. Click **Create Content Type**.
3. Enter the required details such as name, description, owner fields.
4. Select active from the Active Status drop-down list.

Only if the content type is active, it is available for creating content items.

5. Enter an XML tag for the content type in the Content Type XML tag field.  
The XML tag should not contain any spaces and must be unique. OCM stores this data in XML and uses it to present the content item.
6. To select a default stylesheet, click the Search LOV.  
Stylesheets enables you to preview the content item based on the stylesheet definition.
7. Optionally, to encrypt an URL, select the Encryption box.
8. Click **Next**.

In the Create Content Type: *<content type name>*: Attributes page, the content type name and attributes are displayed with their order of display. You cannot modify the order.

9. To define extended attributes for the Content Type, in the Content Type Extended Attributes region, click **Add Another Row**.
  - Enter or select a value in all the required fields.
  - The Field Length field defines the maximum number of characters that can be entered as a value for this attribute while creating content items.
  - The Default Value Updateable? field indicates if the default value can be updated by the user while creating content items. To restrict users from updating the default value for this attribute, select No.
  - The Display Value field indicates the display priority. By default, the fields are displayed in a serial order.
10. To define pull-down attributes for the Content Type, in the Pulldown List Attributes region, click **Add Another Row**.

Enter the values in the field as explained for Content Type Extended Attributes region.

11. To define file attachments, in the File Attachment region, click **Add Another Row**.

Enter the values in the field as explained for Content Type Extended Attributes region.

12. To define a compound content type, in the Compound Content Type region, click **Add Another Row**.
  - Enter the values in the field as explained for Content Type Extended Attributes field.
  - The content type selected becomes the subitem of the compound content item.
  - You can select a default content item from the Search LOV. While creating a content item, this value is displayed as the default subitem.
  - The Default Item Updateable field indicates if the default item can be updated by the user, while creating a content item.

13. Click **Submit**.

The content type is created.

14. To update a content type, select the content type and click **Update**.

15. To duplicate a content type, select the content type and click the **Duplicate** icon.

## 4.3 Mapping Stylesheets to a Content Type

A stylesheet is a content type and tells the rendering engine how to display the content item on the Web. A stylesheet is like any other content type that can be selected to create a content item. You can create a content item using a content type of type stylesheet and store it in the content repository like another content item.

A stylesheet in OCM serves two purposes. Firstly, it defines how the content should be rendered on a Web site or any other device. Secondly, it enables you to preview a content item before it is released. Therefore, it is essential to map a content type to a stylesheet. Stylesheets must conform to the XSL format.

For details on creating content items, see the *Oracle Content Manager User Guide*.

---

---

**Notes:**

- OCM cannot display an item in context with other content items that are used to build Web pages. To display the contents on the Web page, all the contents that are used to build the page must be stored in OCM. This is the responsibility of the application that calls the content item.
  - OCM only supports stylesheets that are of type Extensible Stylesheet Language (XSL).
  - To create your own stylesheets, duplicate the seeded content types.
- 
- 

Use the following procedure to map stylesheets to content types.

**Prerequisites**

You must have content items of type Stylesheet in the repository.

**Steps**

1. Log into Oracle Content Manager as the Content Super User and navigate to Content Type > Stylesheet Mapping.
2. In the Stylesheet Mappings page, select a content type from the Select Content Type drop-down list.
3. Click **Go**.

All the stylesheets that are mapped to the content type are displayed in the Mapped Stylesheets region. The Default Stylesheet column indicates the default stylesheet.

4. To map another stylesheet, click **Add New Row**.
5. Click the Search icon to search for a stylesheet and enter the stylesheet name either partly or fully in the Search By field.
6. Click **Go**.

The search results appear in the Results region. All approved content items of content type Stylesheet are listed.

7. Select the stylesheet to map to the content type and click **Select**.
8. Optionally, to change the default stylesheet setting, select the stylesheet in the Select column.
9. Click **Submit**.

A confirmation message is displayed. The Default Stylesheet column indicates the default stylesheet.

10. Optionally, to delete a stylesheet mapping:
  - a. Select the stylesheet mappings and click the Delete icon from the Delete column.

The stylesheet is removed. But, the mapping is not deleted unless you click the Submit button.
  - b. Click **Submit**.

A confirmation message is displayed, and the mapping is removed.

## 4.4 Creating and Managing Content Type Groups

As more users use OCM, more custom content types are created and it becomes difficult to manage the content types. OCM supports the ability to group the content types into content type groups. By creating groups, you can restrict the users (while creating a content item) to select a content type only from the subscribed group.

Use the following procedure to create and manage Content Type Groups.

## Prerequisites

None

## Steps

1. Log into Oracle Content Manager with the Content Super User and navigate to Content Types > Content Type Group.
2. Click **Create**.
3. Enter the name and description in the Meta Information region.
4. To add a content type to this group, click **Add Another Row**.  
Use the search LOV to select the content types.
5. Click **Submit**.  
The new content type group is created.
6. To update a content type group, select the content type and click **Update**.  
The Update <content type group> page appears. You can add or delete content types in this Content Type Group and click **Submit**.
7. To inactivate a content type group, select the content type group and click **Inactivate**.  
A warning appears asking you to confirm if you want to inactivate the selected content type group. Click **Yes** to inactivate. When a content type group is inactive, all the content types in that group are not available to the users.
8. To duplicate a content type group, select the content type and click **Duplicate**.  
A copy of the selected content type group is created.
9. To activate an inactive content type group, select the Content Type Group and click **Inactive**. Click **Yes** to confirm your action. The Content Type Group is now active.

## 4.5 Mapping File Extensions to Content Types

The bulk upload feature enables you to map file extensions to content types. By using this feature, you can upload files to content types such as Image and File, which are seeded, and user defined file extensions. To enable files to map to user defined file extensions, you must map file extensions to content types.

---

---

**Note:** You can map file extensions only to content types with the file attachments attributes.

---

---

To map file extensions to the content types:

1. Select the content type with file attachment attribute.
2. Click the File Extension icon. The Map File Extension page appears.
3. To add file extension mappings, click the Add Another Row button.
4. Click the Apply button.

## 4.6 Data Security Overview

A folder hierarchy can consist of subfolders and content items. This helps the content creators to organize the content in folders and subfolders.

Folders are associated with a set of meta-data and permissions. For details on different permissions, see [Table 4-1](#).

The IBC: Use Access Control (for data security) profile and the IBC: Default Home Folder (for setting home folder) profile enable security in OCM. They are essential for smooth functioning of OCM. By default, data security is enabled. The Content Super User must assign access control on the folders.

For more information on folders and their properties, see *Oracle Content Manager User Guide*.

### 4.6.1 Access Control

For a folder hierarchy, it is necessary to have robust data security to provide fine grain access control to different users at different folder levels. Folder access control involves assigning a User or a Group the permissions to access the folder. Access control is a combination of permissions and exclusions.

## Permissions

Permissions for a folder are granted to Users, Group, Responsibility, or Global user levels.

The permissions for a folder can be set at the following levels:

- Users - These are users who are defined in the FND APPLICATION as FND USERS.
- Responsibility - The permission can be granted to APPS Responsibilities. When a user logs in, based on the user's login responsibility, the permissions are granted to the user.
- Groups - These are resource groups that have been defined as JTF RESOURCE GROUPS. A user logging into the application must belong to the group to inherit the group permissions.
- Global - Besides Users, Groups and Responsibility, permissions can also be granted at a Global level.

Table 4-1 lists the permissions that you can grant or revoke to a user.

**Table 4-1 Permissions List**

Permission	Description
Manage Folder	Allows you to update the folder properties including the access control.
Create Sub-Folder	Allows you to create subfolders under the given folder.
Manage Item	Allows you to create, update, restore, remove, and delete content items in the given folder.
Approve Item	Allows you to approve content items.
Translate Item	Allows you to translate content items.
Approve Translation	Allows you to approve translations.
Read Item	Allows you to read (view) the content items in the given folder.

## Exclusions

A folder can maintain an exclusion list to exclude certain users or groups by revoking their permissions.

## Inheriting or Overriding Parent Access Control

The permissions of a folder play the role of a template while creating subfolders. The permissions that are granted to a parent folder can be inherited by its subfolder. However, these permissions can be changed anytime.

You can create and manage a folder if you have Create Folder and Manage Folder permissions. In the Folder Properties page, the Access Control region displays the access control (Permissions and Exclusions) for a folder. While creating a folder, if you select the option to:

- Inherit parent access control, then the Permissions and Exclusions regions are disabled. This is the default selection.

The subfolder inherits all the permissions from the parent folder. If you modify the parent folder access control, then the subfolders also reflect the changes.

- Override parent access control, then the Permissions and Exclusions regions are enabled.

You can assign permissions and exclusions for subfolders. If you modify the parent folder access control, then the subfolders do not reflect the changes.

## 4.6.2 Guidelines for Assigning Permissions

In an organization, for users to successfully manage their assigned roles, you may be required to assign other permissions as well, along with the specific permission for the assigned role. See [Table 4-2](#) for details.

**Table 4-2 Recommended Permissions**

Assigned task	Recommended Permissions
Manage Folder	Manage Folder, Create Sub-Folder, Manage Item, and Read Item
Create Sub-Folder	Create Sub-Folder
Manage Item	Manage Item and Read Item
Approve Item	Approve Item, Manage Item, and Read Item
Translate Item	Translate Item and Read Item
Approve Translation	Approve Translation, Translate Item, and Read Item
Read Item	Read Item

### Order of Priority for Access Control

The order of priority for access controls is listed below:

1. User settings override the Responsibility settings, Responsibility settings override Group settings, and Group settings override Global settings.
2. For a user, if the access controls conflict, then permissions override exclusions.

### 4.6.3 Setting Permissions and Exclusions for a Folder

When data security is enabled by setting the IBC: Use Access Control profile to Y, each user must be granted specific privileges that can be performed on each folder. To assign these permissions, use the following procedure.

#### Prerequisites

None

#### Steps

1. Log into Oracle Content Manager as Content Super User and navigate to Content > My Content.
2. Select a folder and click **Folder Properties**.
3. To override parent folder's access control, select the Override radio button.  
The fields in Permissions and Exclusions regions are enabled.
4. To add rows in Permissions and Exclusions region, click **Add Another Row**.
5. In the Type LOV, select either User, Group, Responsibility, or Global.
6. In the Name field, enter a user name or click the Search LOV.
7. To grant and revoke permissions to the user, select the appropriate box in the Permissions and Exclusions regions.
8. Optionally, to inherit parent folder access control, select the Inherit radio button.  
The fields in Permissions and Exclusions regions are disabled.
9. Click **Apply**.  
The folder access controls are set.

## 4.6.4 Understanding Access Control with a Scenario

In this scenario, it is assumed that the profile IBC: User Access Control is set to Y and the access control is with respect to the /Root/Common folder. This scenario also illustrates that User settings overrides the Responsibility settings, Responsibility settings overrides Group settings, and Group settings overrides Global settings. In case of a conflict, permissions override exclusions.

In [Table 4-3](#), "P" indicates the permissions that are granted and "E" indicates the exclusions that are revoked for users with Manage Item permission.

**Table 4-3 Permissions and Exclusions for Manage Item**

Type	Manage Item - Permissions and Exclusions
Global	P
User - John	P
User Harry	P
User - Scott	P
Group - Copyrighters	P
Responsibility - Content Creators	P
User - Nancy	E
Group - Contractors	E
Responsibility - Sales Reps	E

[Table 4-4](#) lists the access controls details that are applicable for individuals, when they navigate to the /Root/Common folder.

**Table 4-4 Access Control Granted to Users**

User Name	Responsibility	Group	Manage Item	Notes
John	Media Manager	-	X	Granted through User Access
Mike	Content Creator	-	X	Granted through Responsibility Access
Joseph	Media Manager	Copyrighters	X	Granted through Group Access
Harry	Media Manager	Contractors	X	Granted as User overrides Group
Scott	Sales Rep.	-	X	Granted as User overrides Responsibility

**Table 4–4 Access Control Granted to Users**

User Name	Responsibility	Group	Manage Item	Notes
David	Content Creator	Contractors	X	Granted as Responsibility overrides Group
Terry	Media Manager	Copyrighters Contractors	X	Granted through group access. Although the permission for Copyrighters is granted and revoked for Contractors, the user still gets the access as Permissions override Exclusions
Jack	External Use	-	X	Granted through Group Access. Although Copyrighters are granted and Contractors are revoked, Jack still gets access as permission overrides exclusions.
Nancy	Media Manager	-	-	Revoked through exclusions. Although Global grant is for all users, Nancy is explicitly excluded.
Henry	Sales Rep.	-	-	Revoked through responsibility exclusions
Nicole	Marketing User	Contractors	-	Revoked through group exclusions.

Table 4–5 explains the data security profiles. You must set these profile options for content items to be created and submitted properly.

**Table 4–5 Data Security Profiles**

Profile Name	Default Value	Comments
IBC: Use Access Control	Y	Now, the data security is enabled. When a user logs in, the application verifies to see if the user has appropriate permissions to perform an action.
IBC: Default Home Folder	/COMMON	Specifies the default home folder for the user.

## 4.7 Categories

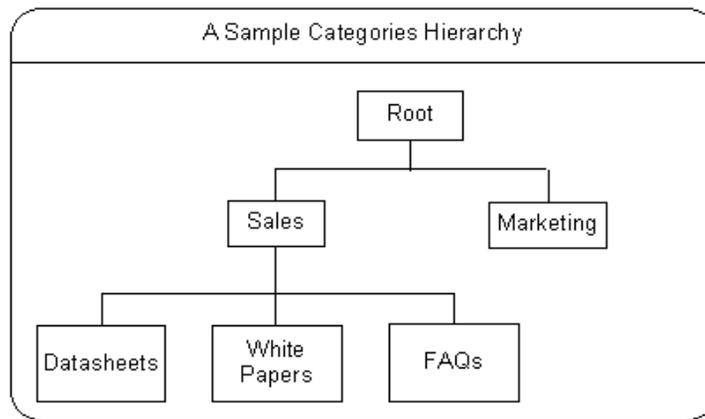
A category is a specifically defined classification, which is used to logically group a set of content items together. While folders are work spaces to organize content items, categories reflect more of a business group. Many organizations define content that can be classified into one or more categories. Categories also have a

hierarchy of subcategories for finer classification. You can browse the library based on the categories.

### Root Category

The Root category is the highest level in the category hierarchy. The Root category is seeded and any seeded category cannot be deleted or renamed. You can create a category under the Root category depending on the permissions you have. See [Figure 4–2](#) for a sample categories hierarchy.

**Figure 4–2 Categories Hierarchy**



In [Figure 4–2](#), Sales and Marketing are two categories under the Root category. Similarly, many such categories can exist under the Root category and subcategories under categories. Datasheets, White Papers, and FAQs are examples of subcategories under Sales.

## 4.7.1 Category Access Control

To maintain a category hierarchy, it is necessary to have robust data security to provide fine grain access control to different users at different category levels. Category access control involves assigning Users or Groups the permissions to access a category. Access control is a combination of permissions and exclusions.

## Permissions

Permissions for a category are granted at Users, Group, Responsibility, or Global levels.

Table 4–6 lists the category permissions.

**Table 4–6 Category Permissions**

Permission	Description
Manage Category	Allows you to create, update, and delete a category.
Publish Content	Allows you to publish content to a category.
View Content	Allows you to view the published content. A library user must have this permission to view and browse the content in the library.

## Exclusions

A category can maintain an exclusion list to exclude certain users or groups by revoking their permissions.

### 4.7.2 Creating a Category and Setting Access Control

The Content Super User or an administrator must update the Root category for users to have Manage Category permission. Without this permission, no user will be able to create a subcategory under the Root category.

As an administrator, you can also assign content items to one or more categories. Only live versions of the content items can be assigned to the categories. Also, you can view, reject, or remove the items from a category.

Use the following procedure to create a category and set access control.

#### Prerequisites

None

#### Steps

1. Log into Oracle Content Manager with the Content Super User permission and navigate to the folder where you would like to create a new folder.
2. Click **Create Categories**.
3. Enter the category name and description.

4. To enable the category, select the Enabled box.
5. In the Access Control region, select the appropriate button to either inherit or override parent access control.
  - To inherit parent access control, select the Inherit radio button. The fields in the Permissions and Exclusions regions are disabled.
  - To customize the access control, select the Override radio button. The fields in the Permissions and Exclusions region are enabled.
6. Click **Apply**.

The category is created.

### Managing Categories

You can perform the following tasks:

- Delete a category with subcategories, but you must have Manage Category permission on the parent category and subcategories. If a category is deleted, then all content item mappings to the category are deleted.
- Disable a category. By disabling a category, all its subcategories are disabled. A disabled category cannot be viewed in the library.
- Enable a category only if the parent category is enabled.

# A

---

---

## Oracle Content Manager Profile Options and Lookups

This chapter describes the process to set profile options and lookups that are required to implement Oracle Content Manager (OCM) successfully. Sections in this chapter include:

- [Appendix A.1, "Setting Profile Options"](#)
- [Appendix A.2, "Finding Responsibility ID Values"](#)
- [Appendix A.3, "Oracle CRM Technology Foundation \(JTT\) Profile Options"](#)
- [Appendix A.4, "Oracle Content Manager Profile Options"](#)
- [Appendix A.5, "Oracle Content Manager Lookups"](#)

## A.1 Setting Profile Options

A profile is a set of changeable options that affect the way the application looks and behaves. As a System Administrator, you can control how Oracle Content Manger operates, by setting the profile options. You can set profile options at four different levels: site, application, responsibility, and user.

Use the following steps to set profile options.

### Prerequisites

Before changing any settings in Oracle Forms, ensure that all Oracle Applications server processes are up and running. In particular, if you have stopped the concurrent managers before applying Oracle Applications patchsets, then restart them now by changing the value to `$COMMON_TOP/admin/scripts`, and executing `adcmctl.sh <APPS username/APPS password> start`.

### Steps

1. Log into Oracle Forms with the System Administrator responsibility and navigate to Security > Profile > System.
2. Check the level(s) at which you want to set the profile option. The available levels are listed below:
  - **Site**
  - **Application**

If you select this level, then choose the application from the Application LOV for which you want to set the profile option.
  - **Responsibility**

If you select this level, then choose the responsibility from the Responsibility LOV for which you want to set the profile option.
  - **User**

If you select this level, then choose the user from the User LOV for whom you want to set the profile option.
3. In the Profile field, enter the profile name, such as `IBC:Use Access Control`, or a wildcard search criterion such as `IBC%`.
4. Click **Find**.

The System Profile Values window opens with the search results.

Verify or set the profile option(s) at the levels that you have selected.

5. Save your work.

## A.2 Finding Responsibility ID Values

Before you set the required Oracle CRM Technology Foundation (JTT) profile options, you must know the values for APPLICATION\_ID and RESPONSIBILITY\_ID for Oracle iContent and IBC Super User responsibilities.

---

---

**Note:** The APPLICATION\_ID value for Oracle iContent is 549 (for Oracle iContent) and the RESPONSIBILITY\_ID value for IBC Super User is 23812.

---

---

Use the following procedure to find the APPLICATION\_ID value and RESPONSIBILITY\_ID value for a responsibility.

### Prerequisites

None

### Steps

1. Log into Oracle Forms with the System Administrator responsibility and navigate to Security > Responsibility > Define.
2. Choose **View > Find**. Search for the responsibility, highlight it, and click **OK** in the search window.  

The Responsibilities window is populated with the record for the responsibility that you chose.
3. With the cursor in any field of the record, choose **Help > Diagnostics > Examine**.  

The Examine Field and Variable Values window appears.
4. In the Examine Field and Variable Values window, choose APPLICATION\_ID from the Field LOV.  

The Value field in the Examine Field and Variable Values window is populated with the value of APPLICATION\_ID.
5. In the Examine Field and Variable Values window, choose RESPONSIBILITY\_ID from the Field LOV.

The Value field in the Examine Field and Variable Values window is populated with the value of RESPONSIBILITY\_ID. Note the required values.

## A.3 Oracle CRM Technology Foundation (JTT) Profile Options

This section summarizes the Oracle CRM Technology Foundation (JTT) profile options that you must set to implementing OCM.

Set the profile options in this section at the Site and Application levels.

---

---

**Note:** Choose Oracle iContent as the application.

---

---

Table A-1 lists the JTT profile options for the OCM UI.

**Table A-1 JTT Profile Options for the OCM UI**

Profile Option Name	Value	Description
JTF_PROFILE_DEFAULT_APPLICATION	549	Default application ID (549=Oracle iContent).
JTF_PROFILE_DEFAULT_CSS	jtfucss.css	Default Oracle CRM Technology Foundation Cascading stylesheet.
JTF_PROFILE_DEFAULT_CURRENCY	USD	Default currency. Enter the currency code in all uppercase letters.
JTF_PROFILE_DEFAULT_RESPONSIBILITY (application level only)	23812	Default responsibility ID (23812=IBC Super User).

### FND Profile Option

The FND: Framework Compatibility Mode profile should be set to 11.5.10 at the application level "Oracle iContent".

## A.4 Oracle Content Manager Profile Options

Table A-2 provides the profile options used by OCM.

**Table A-2 OCM Profile Options**

Profile Name	Description
IBC: Content Type Group	Use this profile to enable users to view only those content types that they are mapped to, in the Content Type Groups.
IBC: Enforce Expiration Date	Each content item created in OCM can be set to have an Expiration Date. If the profile value is set to <i>Yes</i> , then this feature is turned on.
IBC: Enforce Available Date	Each content item created in OCM can be set to have an Available Date. If the profile value is set to <i>Yes</i> , then this feature is turned on.
IBC: Debug Output Directory	Use this profile to set the directory where the debug log files must be created. To find the location, <code>SELECT * from v\$parameter where name like 'utl_file_dir'</code> .
IBC: Enable Encryption	OCM Runtime provides URL access to content item attachments and content items themselves with the capability of encrypting sensitive parameters in the OCM generated URLs. Set this profile value to <i>Y</i> to turn on the encryption.
IBC: Enable Debug	Use this profile to enable debug for Oracle Content Manager Application.
IBC: Default Home Folder	Use this profile to set the Default Home folder for the user.
IBC: Debug Type	Use this profile to set the Content Manager Debug Type.
IBC: Display Base Lang	Use this profile to check whether to display the base language content item version, if the version status is not approved.
IBC: Enable Audit Log	Use this profile to enable Audit Log for Content Manager Application
IBC: Use Access Control	Use this profile to set the access control for OCM. If set to <i>Yes</i> , security is enabled. Security verification is made on the users access control. If set to <i>No</i> , security is disabled. No check is made on the users action.
IBC: Customize Work flow	Use this profile to customize the approval workflow. By default, the value is set to <i>No</i> . This means, if the content creator also has approval privileges on the folder no workflow is launched. But if you have customized the workflow and to launch it, you must set the value to <i>Yes</i> .

## A.5 Oracle Content Manager Lookups

Lookups appear as drop-down values in the user interface. There are three types of lookups:

- **System** - You cannot modify or delete system lookups nor add more values.
- **Extensible** - You can add more values to extensible lookups. But seeded extensible lookups cannot be deleted.
- **User** - You can add or modify the user lookup values. But seeded user lookups cannot be modified or deleted.

---



---

**Note:** You can modify the lookup labels (the text displayed to the user), for all type of lookups.

---



---

Table A-3 lists the OCM Lookup settings.

**Table A-3 OCM Lookup Settings**

Lookup Type	Lookup Values
IBC_ATTRIBUTE_DATA_TYPE	attachment boolean component dateTime decimal html string url
IBC_CITEM_PERMISSIONS	CITEM_READ CITEM_APPROVE CITEM_TRANSLATE CITEM_DELETE CITEM_ARCHIVE CITEM_EDIT CITEM_APPROVE_TRANSLATE

**Table A-3 OCM Lookup Settings**

<b>Lookup Type</b>	<b>Lookup Values</b>
IBC_CITEM_STATUS	APPROVED ARCHIVED PENDING ARCHIVED-CASCADE
IBC_CITEM_VERSION_STATUS	APPROVED ARCHIVED INPROGRESS REJECTED SUBMITTED
IBC_CTYPE_STATUS	ACTIVE INACTIVE
IBC_DIRECTORY_PERMISSIONS	DIR_CITEM_ADD DIR_CREATE DIR_EDIT
IBC_IMAGE_TYPES	GIF JPEG JPG PNG
IBC_CL_ATTRIBUTE_CODES	IBC_CONTENT_BLOCK IBC_IMAGE
IBC_ACTIONS_LIST	COPY CREATE_NEW_VER LATEST_VER LOCK_ITEM PREVIEW_ITEM REMOVE UNLOCK_ITEM UPD_ANOTHER_VER UPD_TRANSLATIONS

**Table A-3 OCM Lookup Settings**

<b>Lookup Type</b>	<b>Lookup Values</b>
IBC_ADV_SEARCH	IS IS_NOT CONTAINS
IBC_AUDIT_ACTIVITIES	MOVE UPDATE ARCHIVE REMOVE REJECT COPY CREATE APPROVE CREATE_NODE UPDATE_NODE UPDATE_NODE_SECURITY
IBC_AUDIT_OBJECTS	ASSOC ABUNDLE COMP CITEM LABEL CIVERSION CTYPE DIRNODE
IBC_CATEGORIES_BROWSE	ALL_CATEGORIES CURRENT_CATEGORY
IBC_CATEGORY_PERMISSIONS	PD_EDIT PD_PUBLISH PD_VIEW

**Table A-3 OCM Lookup Settings**

Lookup Type	Lookup Values
IBC_CITEM_ACTIONS_LIST	COPY CREATE_NEW_VER LATEST_VER LOCK_ITEM PREVIEW_ITEM REMOVE UNLOCK_ITEM UPD_ANOTHER_VER UPD_TRANSLATIONS
IBC_CITEM_ACTION_LIST	IBC_CITEM_VIEW_ASSOCIATIONS IBC_CITEM_VIEW_HISTORY IBC_CITEM_VIEW_USAGE IBC_CITEM_VIEW_VERSIONS
IBC_CITEM_CONTAINS	ANYWHERE DESCRIPTION CONTENT_ITEM
IBC_CITEM_SEARCH_COLUMN	CONTENT_ITEM_NAME DESCRIPTION
IBC_CL_CHANNELS	IBC_EMAIL_CTG IBC_FAX_CTG IBC_PRINT_CTG ALL

**Table A-3 OCM Lookup Settings**

<b>Lookup Type</b>	<b>Lookup Values</b>
IBC_CONTENTITEMS_ACTIONS_LIST	CREATE_NEW_VER LOCK_ITEM LATEST_VER UPD_TRANSLATIONS UNLOCK_ITEM COPY UPD_ANOTHER_VER PREVIEW_ITEM REMOVE
IBC_COPY_ERROR_ACTIONS	CREATE_NEW
IBC_CTG_SEEDED_STATUS	YES NO
IBC_CTG_STATUS	ACTIVE AB INACTIVE
IBC_CTYPE_SUMMARY_STATUS	ACTIVE ALL INACTIVE SEEDED
IBC_DEBUG_TYPES	PIPE FILE
IBC_DIRECTORY_NODE_STATUS	ACTIVE ARCHIVED INACTIVE
IBC_FOLDER_ACTIONS_LIST	COPY REMOVE
IBC_GRANTEE_TYPES	GROUP USER GLOBAL RESPONSIBILITY

**Table A-3 OCM Lookup Settings**

<b>Lookup Type</b>	<b>Lookup Values</b>
IBC_INHERITANCE_STATUS	INHERIT OVERRIDE
IBC_NODE_TYPES	WDIR PDIR
IBC_OBJECT_PERMISSIONS	CITEM_APPROVE CITEM_APPROVE_TRANSLATE CITEM_ARCHIVE CITEM_READ CITEM_DELETE CITEM_EDIT CITEM_TRANSLATE DIR_CITEM_ADD DIR_CREATE DIR_EDIT PD_EDIT PD_VIEW PD_PUBLISH
IBC_PA_FILTER	ALL CONTENT_APPROVAL TRANSLATION_APPROVAL
IBC_PA_NOTIFICATIONS	IBC_CITEM_APPROVE_NOTIFICATION IBC_CITEM_TRANS_APPROVE_NOTIFY
IBC_READONLY_PAGES	ITEM_HISTORY ITEM_USAGE VIEW_VERSIONS

**Table A-3 OCM Lookup Settings**

Lookup Type	Lookup Values
IBC_RENDITIONS	APPLICATION/MSEXCEL APPLICATION/VND.MS-EXCEL APPLICATION/X-JAVASCRIPT APPLICATION/X-MSEXCEL AUDIO/MPEG APPLICATION/POSTSCRIPT APPLICATION/VND.MS-POWERPOINT APPLICATION/X-TAR APPLICATION/VND.MSWORD APPLICATION/X-ZIP-COMPRESSED APPLICATION/MSWORD APPLICATION/RTF APPLICATION/PPT APPLICATION/PDF AUDIO/MIDI VIDEO/X-MSVIDEO VIDEO/X-AVI IMAGE/GIF TEXT/RICHTEXT AUDIO/X-REALAUDIO AUDIO/X-PN-REALAUDIO-PLUGIN AUDIO/X-PN-REALAUDIO TEXT/RTF IMAGE/PJPEG
IBC_SRCH	AFTER BEFORE Y
IBC_USER_GROUPS	GROUP USER

**Table A-3 OCM Lookup Settings**

<b>Lookup Type</b>	<b>Lookup Values</b>
IBC_VERSIONS_CTG	LATEST_VERSION LIVE_VERSION
IBC_VIEW_OPTIONS	HIDE_FOLDERS SHOW_ALL
IBC_YES_NO	N Y
IBC_ZIP_TYPES	ZIPU ZIPL
IBC_RENDITIONS	VIDEO/QUICKTIME TEXT/PLAIN VIDEO/X-SGI-MOVIE VIDEO/MPEG TEXT/XML AUDIO/X-WAV UNKNOWN_MIME_TYPE TEXT/SGML IMAGE/JPEG TEXT/HTML IMAGE/BMP IMAGE/X-MS-BMP IMAGE/X-BMP TEXT/CSS



---

---

## Seeded Content Types and Responsibilities

This appendix chapter details seeded content types, roles, and responsibilities in Oracle Content Manager. Sections in this chapter include:

- [Section B.1, "Seeded Content Types"](#)
- [Section B.2, "Oracle Content Manager Customer UI Responsibilities"](#)

## B.1 Seeded Content Types

The seeded content types in OCM are described below:

### B.1.1 IBC\_IMAGE

[Table B-1](#) gives the seeded attributes for Content Type Image.

**Table B-1 Attributes of IBC\_IMAGE**

Attribute Name	Data Type	Description
Alternate Text	String	When the image binary is not available, or when the browser cannot display the images, this text is displayed.
Image Attachment	Attachment	The image is a binary file.
Description	String	Description of the image.
Height	Decimal	Height of the image.
Link URL	URL	When the image is clicked, indicates to which URL the image is linked
Name	String	Name of the image.
Width	Decimal	Width of the image.

### B.1.2 IBC\_STYLESHEET

[Table B-2](#) gives the seeded attributes for Content Type Stylesheet. Use this information to define stylesheet items which are typically in XSL format that are used to render content items of a particular content type.

**Table B-2 Attributes of IBC\_STYLESHEET**

Attribute Name	Data Type	Description
Attachment	Attachment	Stylesheet file attachment
Delivery Channel	String	Delivery channel this stylesheet is designed for. Values are WEB, FAX, EMAIL, WAP Device, and so on.
Description	String	Description of the stylesheet.
Name	String	Name of the stylesheet.

**Table B-2 Attributes of IBC\_STYLESHEET**

Attribute Name	Data Type	Description
Output Type	String	Indicates the output data format after applying the stylesheet to a content item. Available values are html, wml, text, and so on.

### B.1.3 IBC\_BASIC\_CL

Table B-3 gives the seeded attributes for Content Type Basic Cover Letter structure to be used to fulfillment the content.

**Table B-3 Attributes of IBC\_BASIC\_CL**

Attribute Name	Data Type Code	Description
ATTACHMENT	component	File attachments to be sent with the cover letter.
BODY_HTML	html	HTML content of the cover letter
BODY_TEXT	string	Text content of the cover letter
CBLOCK	component	Static content blocks to be appended to the cover letter content
DESCRIPTION	string	Description of basic cover letter
NAME	string	Name of basic cover letter
QUERY	component	Query associated with the cover letter

### B.1.4 IBC\_CONTENT\_BLOCK

Table B-4 gives the seeded attributes for Content Type Content Block Content Type.

**Table B-4 Attributes of IBC\_CONTENT\_BLOCK**

Attribute Name	Data Type Code	Description
DESCRIPTION	string	Description of content block
HTML	html	HTML Type Data of a Content Block
NAME	string	Name of content block
TEXT	string	Text Type Data of a Content Block

## B.1.5 IBC\_COVER\_LETTER

Table B-5 gives the seeded attributes for Content Type Fulfillment Cover Letter. Use this cover letter structure for fulfilling the content.

**Table B-5 Attributes of IBC\_COVER\_LETTER**

Attribute Name	Data Type Code	Description
ATTACHMENT	string	File attachments to be sent with the cover letter
BODY_HTML	html	HTML Content of the cover letter
BODY_TEXT	string	Text Content of the cover letter
CBLOCK	component	Static content blocks to be appended to the cover letter content
DESCRIPTION	string	Description of cover letter
FOOTER_BLOCK	component	Footer Block of the cover letter
FOOTER_IMAGE	component	Footer Image of the cover letter
HEADER_BLOCK	component	Header Block of the cover letter
HEADER_IMAGE	component	Header Image of the cover letter
NAME	string	Name of cover letter
QUERY	component	Query associated with the cover letter

## B.1.6 IBC\_FILE

Table B-6 gives the seeded attributes for File Content Type.

**Table B-6 Attributes of IBC\_FILE**

Attribute Name	Data Type Code	Description
ATTACHMENT	attachment	Attachment File ID
DESCRIPTION	string	File description
Name	string	File name

## B.1.7 IBC\_IMAGE

Table B-7 gives the seeded attributes for Image Content Type. The generic basic image attributes are defined in the Image content type.

**Table B-7 Attributes of IBC\_IMAGE**

Attribute Name	Data Type Code	Description
ATTACHMENT	attachment	The attached image binary file.
DESCRIPTION	string	Description of image
Name	string	Name of image
ALT_TEXT	string	Displays this text when the binary image is not available, or when the browser cannot display the image
Height	decimal	Height of the image
Link	URL	When the image is clicked, which URL to go to
WIDTH	Width	Width of the image

## B.1.8 IBC\_QUERY

Table B-8 gives the seeded attributes for Query Content Type.

**Table B-8 Attributes of IBC\_QUERY**

Attribute Name	Data Type Code	Description
IS_DATA_QUERY	boolean	Indicates if it is a data query
DESCRIPTION	string	Query description
Name	string	Query name
QTEXT	string	Complete query to be attached
QUERYID	attachment	FND ID of the file, which belongs to the query

## B.2 Oracle Content Manager Customer UI Responsibilities

Oracle Content Manager provides security at two levels - Functional and Data Level.

- **Functional Level:** This controls the availability of tabs and enables certain functions in the UI application. This section details the seeded responsibilities.
- **Data Level:** Enables OCM users to execute different tasks in the application. For more information on the data level security, see [Section A.4, "Oracle Content Manager Profile Options"](#).

### B.2.1 Seeded Responsibilities

[Table B-9](#) lists the seeded responsibilities in OCM along with a brief description of each responsibility.

**Table B-9 Seeded Responsibilities**

Responsibility ID	Responsibility Name	Responsibility Key
23345	Oracle Content Manager	IBC_CONTENT_MANAGER
23810	Content Creator	IBC_CREATOR_KEY
23812	Content Super User	IBC_SUPER_USER_KEY
23814	Content User	IBC_USER_KEY
24782	Content Manager Self-Service Responsibility	IBC_SELF_SERVICE_RESP_KEY

---

---

# Content Item XML Formats

This appendix chapter gives the XML formats for content items. Topics in this section include:

- [Section C.1, "XML Format Overview"](#)
- [Section C.2, "Regular Expressions"](#)
- [Section C.3, "An XML Example for Content Item"](#)

## C.1 XML Format Overview

You must understand the XML formats of OCM content items to create a correct XSL stylesheet.

## C.2 Regular Expressions

The following regular expressions give the formal description of the XML format of OCM content items.

**Content Item XML ::=**

```
{ Content Item Open Tag },  
{ Attachment Attribute }?,  
{ Primitive Attribute }*,  
{ Component Item Attribute Reference }*,  
{ Component Item Attribute }*,  
{ Content Item End Tag }
```

**Content Item Open Tag ::=**

```
<CONTENT_TYPE_CODE datatype="citem" id="content_item_id" version="version_  
number"  
available="available_date" expiration="expiration_date"  
ircode="item_reference_code" ref="f"  
url="ibcGetContentItem.jsp?cItemId=content_item_id">
```

**Content Item End Tag ::=**

```
</CONTENT_TYPE_CODE>
```

**Attachment Attribute ::=**

```
<ATTACHMENT_ATTRIBUTE_TYPE_CODE datatype="attachment" id="attachment_file_id"  
ref="t" file="attachment_file_name" mimeType="attachment_mime_type"  
renditionName="rendition_name" defaultMimeType="t"  
url="ibcGetAttachment.jsp?fileId=attachment_file_id" />
```

**Primitive Attribute ::=**

```
{ Primitive String Attribute } | { Primitive Number Attribute } | { Primitive  
Date Attribute } |  
{ Primitive Boolean Attribute } | { Primitive HTML Attribute } | { Primitive URL  
Attribute }
```

**Primitive String Attribute ::=**

```
<TEXT_ATTRIBUTE_TYPE_CODE datatype="string">  
string value  
</TEXT_ATTRIBUTE_TYPE_CODE>
```

**Primitive Number Attribute ::=**

```
<NUMBER_ATTRIBUTE_TYPE_CODE datatype="decimal">
number value
</NUMBER_ATTRIBUTE_TYPE_CODE>
```

**Primitive Date Attribute ::=**

```
<DATE_ATTRIBUTE_TYPE_CODE datatype="dateTime">
date value
</DATE_ATTRIBUTE_TYPE_CODE>
```

**Primitive Boolean Attribute ::=**

```
<BOOLEAN_ATTRIBUTE_TYPE_CODE datatype="boolean">
boolean value
</BOOLEAN_ATTRIBUTE_TYPE_CODE>
```

**Primitive HTML Attribute ::=**

```
<HTML_ATTRIBUTE_TYPE_CODE datatype="html">
html value
</HTML_ATTRIBUTE_TYPE_CODE>
```

**Primitive URL Attribute ::=**

```
<URL_ATTRIBUTE_TYPE_CODE datatype="url">
url value
</URL_ATTRIBUTE_TYPE_CODE>
```

**Component Item Attribute Reference ::=**

```
<COMPONENT_ATTRIBUTE_TYPE_CODE datatype="component" id="component_item_id"
ref="t" url="ibcGetContentItem.jsp?cItemId=component_item_id" />
```

**Component Item Attribute ::=**

```
{ Component Item Attribute Open Tag},
{ Content Item XML },
{ Component Item Attribute End Tag }
```

**Component Item Attribute Open Tag ::=**

```
<COMPONENT_ATTRIBUTE_TYPE_CODE datatype="component" id="component_item_id"
ref="f" url="ibcGetContentItem.jsp?cItemId=component_item_id" >
```

**Component Item Attribute End Tag ::=**

```
</COMPONENT_ATTRIBUTE_TYPE_CODE>
```

## C.3 An XML Example for Content Item

```
<WHITEPAPER datatype="citem" id="10017" version="2" available="2002-05-01"
```

```
expiration="2003-05-01" ircode="WP123" ref="f"
url="ibcGetContentItem.jsp?cItemId=10017">
  <!-- Attachment Attribute -->
  <WP_ATTACHMENT datatype="attachment" id="24357" ref="t" file="wp123.pdf"
  mimeType="application/pdf" renditionName="PDF" defaultMimeType="t"
  url="ibcGetAttachment.jsp?fileId=24357" />
  <!-- Primitive Attributes -->
  <NAME datatype="string">Oracle 9i</NAME>
  <DESCRIPTION datatype="string">Oracle 9i Description</DESCRIPTION>
  <PUBLISH_DATE datatype="dateTime">27-DEC-01 17:24:18 PST</PUBLISH_DATE>
  <RELEASED datatype="boolean">T</RELEASED>
  <PRODUCT_ID datatype="decimal">1234567</PRODUCT_ID>
  <!-- Component Item Attribute Expanded -->
  <WP_IMAGE datatype="component" id="10003" ref="f"
  url="ibcGetContentItem.jsp?cItemId=10003">
    <IMAGE datatype="citem" id="10003" version="9" available="" expiration=""
    ircode="WPIMAGE123" ref="f" url="ibcGetContentItem.jsp?cItemId=10003">
      <IMAGE_FILE datatype="attachment" id="24358" ref="t"
      file="image123.gif"
      mimeType="image/gif" renditionName="GIF" defaultMimeType="t"
      url="ibcGetAttachment.jsp?cItemId=24358" />
      <NAME datatype="string">Image</NAME>
      <DESCRIPTION datatype="string">Image Description</DESCRIPTION>
      <AUTHOR datatype="string">Test Author</AUTHOR>
    </IMAGE>
  </WP_IMAGE>
</WHITEPAPER>
```

# D

---

---

## Concurrent Programs

This appendix lists all the concurrent programs in Oracle Content Manager. Topics in this section include:

- [Section D.1, "Running Concurrent Programs"](#)
- [Section D.2, "List of Concurrent Programs"](#)

## D.1 Running Concurrent Programs

The procedure for running concurrent programs is the same for all Oracle applications. Use this procedure to run OCM concurrent processes.

### Steps

1. Log in to the Forms application.
2. Select the Oracle Sales Administrator responsibility.
3. Double-click **Concurrent Requests**.
4. Double-click **Run**.
5. Select the Single Request radio button.
6. Click **OK**.
7. Search for the concurrent request that you want to run.
8. Click **Submit**.
9. Select from the search results.
10. Click **OK**.
11. Click **OK** to run the concurrent request.

For a detailed description of the procedures, refer to the *Oracle Applications System Administrator's Guide*.

## D.2 List of Concurrent Programs

Table D-1 lists the concurrent programs in OCM.

**Table D-1 Concurrent Programs**

<b>Concurrent Program Name</b>	<b>Description</b>
OCM InterMedia Index Optimizing operation for Attribute Bundle	Performs an iMT Sync check on all indexes across all IBC iMT-indexed columns.
OCM InterMedia Index Synchronizing for Attribute Bundles	Performs an iMT Optimize check on all indexes across all IBC iMT-indexed columns in a time-distributed fashion.
BulkUploadCP	This is used to bulk upload content items and folders.

---



---

### Notes:

- Oracle recommends you to run the concurrent program at least once in a day or more frequently depending on your business requirements.
  - If you are uploading content items using Bulk Upload, then it is recommended that you run the BulkUpload Concurrent Program immediately. Only after you run this concurrent program, you can perform intermediate searches based on the attributes.
- 
-



---

---

# Oracle Content Manager API Reference

Oracle Content Manager (OCM) provides a set of public Application Programming Interfaces (API) to retrieve contents that are stored in the OCM repository. Sections in this appendix include:

- [Section E.1, "API Overview"](#)
- [Section E.2, "Content Manager API Interface Summary"](#)
- [Section E.3, "Interface ContentItemMeta"](#)
- [Section E.4, "Interface ContentItem"](#)
- [Section E.5, "Content Manager API Class Summary"](#)
- [Section E.6, "Class Attribute"](#)
- [Section E.7, "Class ComponentItemAttribute"](#)
- [Section E.8, "Class ContentCacheManager"](#)
- [Section E.9, "Class ContentDeliveryManager"](#)
- [Section E.10, "Class Rendition"](#)
- [Section E.11, "Exception Classes"](#)

## E.1 API Overview

You can use the APIs to create or customize front-end applications to display contents from the OCM repository.

It is important to note that these public APIs are provided to retrieve contents that are viewed by the end users. Therefore, the contents requested must be public and approved.

The main Java class that you will be using for retrieving contents is `ContentDeliveryManager`. For more details, see [Section E.9, "Class ContentDeliveryManager"](#). OCM internally supports content caching to enhance runtime performance. If you choose to build your own pages to administer the runtime cache, then `ContentCacheManager` provides the APIs to do so. For more details, see [Section E.8, "Class ContentCacheManager"](#).

## E.2 Content Manager API Interface Summary

APIs for the OCM procedures are located in the package `oracle.apps.abc.runtime`. [Table E-1](#) describes the interfaces briefly.

**Table E-1 OCM Interface Summary**

Interface Name	Description
<a href="#">Interface ContentItemMeta</a>	The <code>ContentItemMeta</code> interface provides the APIs to access a content item's meta data.
<a href="#">Interface ContentItem</a>	The <code>ContentItem</code> interface provides the APIs to access the content item's details and attributes.

## E.3 Interface ContentItemMeta

The `ContentItemMeta` interface provides the APIs to access a content item's meta data.

A content item object can be loaded in three different levels: `Meta_Loaded`, `Basic_Loaded`, and `Deep_Loaded`.

- `Meta_Loaded` loads attributes that are required in all items, such as Name, Description, and Reference Code.
- `Basic_Loaded` loads attributes that form a superset of the `Meta_Loaded` attributes. It also contains the user-defined item's attributes. If the item contains component attributes, then they are loaded as references.

- Deep\_Loaded is similar to Basic\_Loaded except that the item's component attributes are fully loaded instead of being references. In other words, the entire component item is loaded.

```
public interface ContentItemMeta
```

### E.3.1 Variables for Interface ContentItemMeta

#### **BASIC\_LOADED**

```
public static final int BASIC_LOADED
```

#### **DEEP\_LOADED**

```
public static final int DEEP_LOADED
```

#### **META\_LOADED**

```
public static final int META_LOADED
```

#### **RCS\_ID**

```
public static final java.lang.String RCS_ID
```

### E.3.2 Methods for Interface ContentItemMeta

Table E-2 is an index of ContentItemMeta interface methods.

**Table E-2 Method Index for Interface ContentItemMeta**

Method	Description
<a href="#">getAttachmentFileId</a>	Returns the attachment file ID if the content item has an attachment. If no attachment, it returns -1.
<a href="#">getAttachmentFileName</a>	Returns the attachment file name if the content item has an attachment. If no attachment, it returns NULL.
<a href="#">getAttachmentUrl</a>	Returns the URL to access the default rendition attachment through HTTP.
<a href="#">getAvailableDate</a>	Returns the available date of the content item.
<a href="#">getContentItemId</a>	Returns the content item ID.
<a href="#">getContentItemName</a>	Returns the name of the content item.
<a href="#">getContentItemTypeCode</a>	Returns the content type code.

**Table E-2 Method Index for Interface ContentItemMeta**

Method	Description
<a href="#">getDefaultRenditionMimeType</a>	Returns the default rendition mime type if the content item has an attachment.
<a href="#">getDefaultRenditionName</a>	Returns the default rendition mime type if the content item has an attachment.
<a href="#">getDescription</a>	Returns the description of the content item.
<a href="#">getExpirationDate</a>	Returns the expiration date of the content item.
<a href="#">getItemReferenceCode</a>	Returns the item reference code.
<a href="#">getLoadLevel</a>	Return the load level of this content item.

**getAttachmentField**

```
public int getAttachmentFileId()
```

**Description**

The `getAttachmentField` method returns the attachment file ID if the content item has an attachment. If no attachment, it returns -1.

**Returns**

The attachment file ID if the content item has an attachment. If no attachment, it returns -1.

**getAttachmentFileName**

```
public java.lang.String getAttachmentFileName()
```

**Description**

The `getAttachmentFileName` method returns the attachment file name if the content item has an attachment. If no attachment, it returns `NULL`.

**Returns**

Returns the attachment file name if the content item has an attachment. If no attachment, it returns `NULL`.

**getAttachmentUrl**

```
public java.lang.String getAttachmentUrl()
```

**Description**

The `getAttachmentUrl` method returns the URL to access the default rendition attachment through HTTP. If no attachment, it returns `NULL`.

**Returns**

Attachment URL

**getAvailableDate**

```
public java.util.Date getAvailableDate()
```

**Description**

Returns the available date of the content item if it was set up. If the date is not set up, it returns `NULL`.

**Returns**

Available Date

**getContentItemId**

```
public int getContentItemId()
```

**Description**

The `getContentItemId` method returns the content item's ID.

**Returns**

Content Item Id

**getContentItemName**

```
public java.lang.String getContentItemName()
```

**Description**

The `getContentItemName` method returns the name of the content item.

**Returns**

Content item name

**getContentTypeCode**

```
public java.lang.String getContentTypeCode()
```

**Description**

The `getContentTypeCode` method returns the type of the content code.

**Returns**

Content type code

**getDefaultRenditionMimeType**

```
public java.lang.String getDefaultRenditionMimeType()
```

**Description**

Returns the default rendition mime type if the content item has an attachment. If no attachment, returns `NULL`.

**Returns**

Default rendition MIME type

**getDefaultRenditionName**

```
public java.lang.String getDefaultRenditionName()
```

**Description**

Returns the default rendition name if the content item has an attachment. If no attachment, returns `NULL`.

**Returns**

Default rendition name

**getDescription**

```
public java.lang.String getDescription()
```

**Description**

The `getDescription` method returns the description of the content item.

**Returns**

Content item description

**getExpirationDate**

```
public java.util.Date getExpirationDate()
```

**Description**

Returns the expiration date of the content item if it was set up. If the content item is not set up, it returns `NULL`.

**Returns**

Expiration Date

**getItemReferenceCode**

```
public java.lang.String getItemReferenceCode()
```

**Description**

The `getItemReferenceCode` method returns the item reference code.

**Returns**

Item reference code

**getLoadLevel**

```
public int getLoadLevel()
```

**Description**

The `getLoadLevel` method returns the level of load in this content item. The return value of this method could be:

- `META_LOADED` - Content item is meta data loaded
- `BASIC_LOADED` - Content item is basic data loaded with references for component items
- `DEEP_LOADED` - Full data loaded with component items fully loaded

**Returns**

Content item load level

## E.4 Interface ContentItem

```
public interface ContentItem  
extends ContentItemMeta
```

The `ContentItem` interface provides the APIs to access the content item's details and attributes. A content item object can be loaded in three different levels: `Meta_Loaded`, `Basic_Loaded`, and `Deep_Loaded`.

If the load level of the content item is `Deep_Loaded`, then the dynamic type of the attribute object will be *ComponentItemAttribute*. The developers can access the full component item through this object type, which is a sub-class of class `Attribute`.

If the load level of the content item is `Basic_Loaded`, then the dynamic type of its component item attribute is *Attribute*.

### E.4.1 Fields Inherited

The following fields are inherited by interface `ContentItem` from interface `oracle.apps.abc.runtime`:

- `BASIC_LOADED`
- `DEEP_LOADED`
- `META_LOADED`

### E.4.2 Methods Inherited by Interface ContentItem

The following methods have been inherited from interface `oracle.apps.abc.runtime`:

- `getAttachmentFileId`
- `getAttachmentFileName`
- `getAttachmentUrl`
- `getAvailableDate`
- `getContentItemId`
- `getContentItemName`
- `getContentTypeCode`
- `getDefaultRenditionMimeType`
- `getDefaultRenditionName`
- `getDescription`
- `getExpirationDate`
- `getItemReferenceCode`
- `getLoadLevel`

## E.4.3 Methods for Interface ContentItem

Table E-3 is an index of ContentItem interface methods.

**Table E-3 Method Index for Interface ContentItem**

Method	Description
<a href="#">getAllAttributes</a>	Returns both primitive and component attributes.
<a href="#">getAllPrimitiveAttributes</a>	Returns all the primitive attributes.
<a href="#">getAllComponentItemAttributes</a>	Returns all the component attributes.
<a href="#">getAttributes</a>	Returns a list of attributes matching the given attribute type code.

### **getAllAttributes**

```
public Attribute[] getAllAttributes()
```

#### **Description**

The `getAllAttribute` method returns both primitive and component attributes.

#### **Returns**

List of attributes

### **getAllPrimitiveAttributes**

```
public Attribute[] getAllPrimitiveAttributes()
```

#### **Description**

The `getAllPrimitiveAttributes` method returns all the primitive attributes. Primitive attributes are attributes whose data types are string, decimal, datetime, and boolean.

#### **Returns**

List of primitive attributes

### **getAllComponentItemAttributes**

```
public Attribute[] getAllComponentItemAttributes()
```

#### **Description**

The `getAllComponentItemAttributes` method returns all the component item attributes.

**Returns**

List of component attributes

**getAttributes**

```
public Attribute[] getAttributes(java.lang.String attributeTypeCode)
```

**Description**

The `getAttributes` method returns a list of attributes matching the given attribute type code. The method returns `NULL` if there is no match.

**Returns**

List of attributes

**Parameters**

`AttributeTypeCode` - This is the type of attribute code.

## E.5 Content Manager API Class Summary

APIs for the Oracle Content Manager procedures are located in the package `oracle.apps.abc.runtime`.

[Table E-4](#) describes the classes briefly.

**Table E-4 OCM Class Summary**

Interface Name	Description
<a href="#">Class Attribute</a>	Class <code>Attribute</code> is the superclass of all classes representing an attribute of a content item.
<a href="#">Class ComponentItemAttribute</a>	<code>ComponentItemAttribute</code> is a sub-class of <code>Attribute</code> . It represents a component item attribute of a content item.
<a href="#">Class ContentCacheManager</a>	The OCM run-time cache enhances performance when delivering content items to the end users.
<a href="#">Class ContentDeliveryManager</a>	The <code>ContentDeliveryManager</code> class provides APIs for applications to retrieve content items stored in the OCM Content Repository that are to be used by public end users.
<a href="#">Class ContentDeliveryException</a>	This class extends the <code>FrameworkException</code> class and indicates that there are problems in calling the OCM Runtime APIs.

## E.6 Class Attribute

```
public class Attribute
    extends java.lang.Object
```

Attribute is the superclass of all classes representing an attribute of a content item.

### E.6.1 Constructors for Class Attribute

#### Attribute

```
public Attribute(java.lang.String attributeTypeCode,
                java.lang.String dataTypeCode,
                java.lang.String attributeValue)
```

### E.6.2 Methods Inherited

The following methods are inherited by class Attribute from class java.lang.Object

- equals
- getClass
- hashCode
- notify
- notifyAll
- toString
- wait

### E.6.3 Methods for Class Attribute

[Table E-5](#) is an index of Attribute class methods.

**Table E-5 Method Index for Class Attribute**

Method	Description
<a href="#">getAttributeTypeCode</a>	Returns the attribute type code of the attribute.
<a href="#">getAttributeValue</a>	Returns the attribute value.
<a href="#">getDataTypeCode</a>	Returns the data type code of this attribute.

### **getAttributeTypeCode**

```
public java.lang.String getAttributeTypeCode()
```

#### **Description**

The `getAttributeTypeCode` method returns the type of attribute code of the attribute.

#### **Returns**

Attribute type code of the attribute.

### **getAttributeValue**

```
public java.lang.String getAttributeValue()
```

#### **Description**

The `getAttributeValue` method returns the attribute value. If the data type code of this attribute is of primitive type (for example, string, decimal, or boolean), then the return value will be the attribute value. If the data type code of this attribute is component, then the return value is be the ID of the component item.

#### **Returns**

Attribute value or Component item ID

### **getDataTypeCode**

```
public java.lang.String getDataTypeCode()
```

#### **Description**

The `getDataTypeCode` method returns the data type code of this attribute.

#### **Returns**

Data type code of this attribute.

## **E.7 Class ComponentItemAttribute**

```
public class ComponentItemAttribute  
extends Attribute
```

`ComponentItemAttribute` is a sub-class of `Attribute`. It represents a component item attribute of a content item.

## E.7.1 Constructor for Class ComponentItemAttribute

### ComponentItemAttribute

```
public ComponentItemAttribute(java.lang.String attributeTypeCode,  
                             java.lang.String dataTypeCode,  
                             java.lang.String attributeValue,  
                             ContentItem item)
```

## E.7.2 Methods Inherited

The following methods are inherited by class `ComponentItemAttribute` from class `oracle.apps.abc.runtime.Attribute`.

- `getAttributeTypeCode`
- `getAttributeValue`
- `getDataTypeCode`

The following methods are inherited by class `ComponentItemAttribute` from class `java.lang.Object`.

- `equals`
- `getClass`
- `hashCode`
- `notify`
- `notifyAll`
- `toString`
- `wait`
- `wait`
- `wait`

## E.7.3 Methods for Class ComponentItemAttribute

Table E-6 is an index of ComponentItemAttribute class methods.

**Table E-6 Method Index for Class ComponentItemAttribute**

Method	Description
<a href="#">getComponentItem</a>	Returns a component item of this Component Item Attribute.

### **getComponentItem**

```
public ContentItem getComponentItem()
```

#### **Description**

The `getComponentItem` method returns a component item of the component item attribute.

#### **Returns**

Component item object.

## E.8 Class ContentCacheManager

```
public class ContentCacheManager  
extends java.lang.Object
```

The OCM run-time cache enhances performance when delivering content items to the end users.

This run-time cache is not used in the OCM Administrative APIs. The `ContentCacheManager` provides APIs for explicitly updating the OCM Runtime Cache. These APIs are primarily intended for applications or modules that do not make use of the OCM Administration module to maintain their content, but instead do the administration of the content themselves. Such modules would require to explicitly update the OCM Runtime Cache. These APIs update only the cache entries, and therefore should always be called after the actual update to the database.

### E.8.1 Methods Inherited

The following methods are inherited by class `ContentCacheManager` from class `java.lang.Object`

- `equals`

- getClass
- hashCode
- notify
- notifyAll
- toString
- wait
- wait
- wait

## E.8.2 Methods for Class ContentCacheManager

Table E-7 is an index of ContentCacheManager class methods.

**Table E-7 Method Index for Class ContentCacheManager**

Method	Description
<a href="#">invalidateAssociation</a>	Invalidates an association between a CRM object and a content item in the OCM cache.
<a href="#">invalidateLabelVersion</a>	Invalidates the specified label version in the OCM cache.
<a href="#">invalidateLiveVersion</a>	Invalidates the live version of the specified content item in the OCM cache.
<a href="#">preloadContentItems</a>	Preloads the requested content items in the versions specified in their labels and in the requested languages into the OCM Cache.
<a href="#">putAssociation</a>	Associates a CRM object and a content item into the OCM Cache.
<a href="#">updateLabelVersion</a>	Updates the OCM cache with a new version number for the specified label version.
<a href="#">updateLiveVersion</a>	Updates the OCM cache with the new live version number for the specified content item.

### invalidateAssociation

```
public static void invalidateAssociation(
    java.lang.String associatedObjectVal1,
    java.lang.String associatedObjectVal2,
    java.lang.String associatedObjectVal3,
```

```
java.lang.String associatedObjectVal4,  
java.lang.String associatedObjectVal5,  
int contentItemId)  
throws oracle.apps.jtf.base.resources.FrameworkException
```

### Description

The `InvalidateAssociation` method invalidates an association between a CRM object and a content item in the OCM cache. This should be used when an association is deleted.

### Parameters

- `associationTypeCode` - the association type
- `associatedObjectVal1` - ID value of the associated object
- `associatedObjectVal2` - ID value of the associated object (pass `NULL` if not applicable)
- `associatedObjectVal3` - ID value of the associated object (pass `NULL` if not applicable)
- `associatedObjectVal4` - ID value of the associated object (pass `NULL` if not applicable)
- `associatedObjectVal5` - ID value of the associated object (pass `NULL` if not applicable)
- `contentItemId` - ID of the content item that is associated with the specified CRM object

### Throws

`oracle.apps.jtf.base.resources.FrameworkException` - Framework Cache Error

### **invalidateLabelVersion**

```
public static void invalidateLabelVersion(  
java.lang.String label,  
int contentItemId)  
throws oracle.apps.jtf.base.resources.FrameworkException
```

### Description

The `InvalidateLabelVersion` method invalidates the specified label version in the OCM cache. This should be used when a version of an item linked to the label has changed.

**Parameters**

- label - label code
- contentItemId - content item ID

**Throws**

oracle.apps.jtf.base.resources.FrameworkException - Framework Cache Error

**invalidateLiveVersion**

```
public static void invalidateLiveVersion(int contentItemId)
throws oracle.apps.jtf.base.resources.FrameworkException
```

**Description**

The `InvalidateLiveVersion` method invalidates the live version of the specified content item in the OCM cache. This should be called when the live version of the given content item has changed.

**Parameters**

contentItemId - content item ID

**Throws**

oracle.apps.jtf.base.resources.FrameworkException - Framework Cache Error

**preloadContentItems**

```
public static void preloadContentItems(int[] contentItemIds,
                                       java.lang.String[] labels,
                                       java.lang.String[][] languageCodes)
throws InvalidInputException,
oracle.apps.jtf.base.resources.FrameworkException
```

**Description**

The `preloadContentItems` method preloads the requested content items in the versions specified in their labels and in the requested languages into the OCM Cache. If any inputs are invalid, then the loading process continues by skipping the invalid ones.

**Parameters**

- contentItemIds - 1D array of content item IDs
- labels - 1D array of labels

- languageCodes - 2D array of language codes (each contentItemId-label mapping returns ties to an array of languages to be loaded)

### Throws

InvalidInputException - Input Error. Input arrays must be parallel (equal in length)

oracle.apps.jtf.base.resources.FrameworkException - Framework Cache Error

### putAssociation

```
public static void putAssociation(java.lang.String associationTypeCode,  
                                java.lang.String associatedObjectVal1,  
                                java.lang.String associatedObjectVal2,  
                                java.lang.String associatedObjectVal3,  
                                java.lang.String associatedObjectVal4,  
                                java.lang.String associatedObjectVal5,  
                                int contentItemId)  
throws oracle.apps.jtf.base.resources.FrameworkException
```

### Description

The putAssociation method associates a CRM object and a content item into the OCM Cache. This should be used when a new association is created.

### Parameters

- associationTypeCode - the association type
- associatedObjectVal1 - ID value of the associated object
- associatedObjectVal2 - ID value of the associated object (pass NULL if not applicable)
- associatedObjectVal3 - ID value of the associated object (pass NULL if not applicable)
- associatedObjectVal4 - ID value of the associated object (pass NULL if not applicable)
- associatedObjectVal5 - ID value of the associated object (pass NULL if not applicable)
- contentItemId - ID of the content item to be associated with the specified CRM object

### Throws

oracle.apps.jtf.base.resources.FrameworkException - Framework Cache Error

### **updateLabelVersion**

```
public static void updateLabelVersion(java.lang.String label,  
                                     int contentItemId,  
                                     int versionNumber)  
throws oracle.apps.jtf.base.resources.FrameworkException
```

#### **Description**

The `UpdateLabelVersion` method updates the OCM cache with a new version number for the specified label version. This should be used when a version of the given item linked to the label has changed.

#### **Parameters**

- `label` - label code
- `contentItemId` - content item ID
- `versionNumber` - the new version number

#### **Throws**

`oracle.apps.jtf.base.resources.FrameworkException` - Framework Cache Error

### **updateLiveVersion**

```
public static void updateLiveVersion(int contentItemId,  
                                     int versionNumber)  
throws oracle.apps.jtf.base.resources.FrameworkException
```

#### **Description**

The `UpdateLiveVersion` method updates the OCM cache with the new live version number for the specified content item. This should be called when the live version of the given content item has changed.

#### **Parameters**

- `contentItemId` - content item ID
- `versionNumber` - the new live version

#### **Throws**

`oracle.apps.jtf.base.resources.FrameworkException` - Framework Cache Error

## E.9 Class ContentDeliveryManager

```
public class ContentDeliveryManager
extends java.lang.Object
```

### Description

The ContentDeliveryManager class provides APIs for applications to retrieve content items stored in the OCM Content Repository that are to be used by public end users.

The ContentDeliveryManager makes use of the OCM Runtime Cache underneath it to enhance the performance of content retrieval. An application that chooses to administer its content on its own, instead of using the OCM administrative component, may need to use ContentCacheManager in parallel.

A content item can be loaded in 3 different levels: META, BASIC and DEEP.

- META will load attributes that are required in all items, such as Name, Description, and Reference Code.
- BASIC will load attributes that form a superset of the META\_LOADED; it will also contain the user-defined item's attributes. If the item contains component attributes, then they will be loaded as references.
- DEEP is similar to BASIC except that the item's component attributes are fully loaded instead of being references (i.e., the entire component item is loaded).

### E.9.1 Methods Inherited

The following methods are inherited by class ContentDeliveryManager from the class java.lang.Object:

- equals
- getClass
- hashCode
- notify
- notifyAll
- toString
- wait
- wait

- wait

## E.9.2 Methods for Class ContentDeliveryManager

Table E-8 is an index of ContentDeliveryManager class methods.

**Table E-8 Method Index for Class ContentDeliveryManager**

Method	Description
<a href="#">getAttachmentUrlByCitemId</a>	Returns the URL to access the IBC binary loader to stream an attachment file through HTTP.
<a href="#">getCitemsMetaByAssoc</a>	Returns a list of content items with their metadata based on association and, optionally, on label.
<a href="#">getCitemsMetaByAssocCtype</a>	Returns a list of content items with their metadata based on association, content type and, optionally, on label.
<a href="#">getContentItemUrlBasic</a>	Returns the URL to access the IBC content item loader to stream a content item (basic load), with the specified stylesheet applied, through HTTP.
<a href="#">getContentItemUrlBasic</a>	Returns the URL to access the IBC content item loader to stream a content item (basic load), with its content type's default stylesheet applied, through HTTP.
<a href="#">getContentItemUrlDeep</a>	Returns the URL to access the IBC content item loader to stream a content item (deep load), with the specified stylesheet applied, through HTTP.
<a href="#">getContentItemUrlDeep</a>	Returns the URL to access the IBC content item loader to stream a content item (deep load), with its content type's default stylesheet applied, through HTTP.
<a href="#">loadContentItemBasic</a>	Returns a content item with its basic data.
<a href="#">loadContentItemBasicWithXSL</a>	Loads a content item (basic load), with the specified stylesheet applied, into an Output stream.
<a href="#">loadContentItemBasicWithXSL</a>	Loads a content item (basic load), with its content type's default stylesheet applied, into an Output stream.
<a href="#">loadContentItemBasicXml</a>	Returns a content item with its basic data in XML.
<a href="#">loadContentItemBasicXmlDom</a>	Returns a content item with its basic data in XML.
<a href="#">loadContentItemDeep</a>	Returns a content item with its full data.

**Table E-8 Method Index for Class ContentDeliveryManager**

Method	Description
<a href="#">loadContentItemDeepWithXSL</a>	Loads a content item (deep load), with the specified stylesheet applied, into an Output stream.
<a href="#">loadContentItemDeepWithXSL</a>	Loads a content item (deep load), with its content type's default stylesheet applied, into an Output stream.
<a href="#">loadContentItemDeepXml</a>	Returns a content item with its full data.
<a href="#">loadContentItemDeepXmlDom</a>	Returns a content item with its full data.
<a href="#">loadContentItemMeta</a>	Returns a content item with its metadata.
<a href="#">loadContentItemsMeta</a>	Returns a list of content items with their metadata based on their IDs.

**getAttachmentUrlByCItemId**

```
public static java.lang.String getAttachmentUrlByCItemId(int cItemId,
                                                         java.lang.String label)
```

**Description**

The `getAttachmentUrlByCItemId` method returns the URL to access the IBC binary loader to stream an attachment file through HTTP.

**Returns**

The URL of the IBC binary loader.

**Parameters**

- `cItemId` - ID of the content item whose attachment has to be streamed back.
- `label` - Label to specify the version of a particular content item (pass NULL if not applicable).

**getAttachmentURLByFileId**

```
public static java.lang.String getAttachmentUrlByFileId(int fileId)
```

**Description**

The `getAttachmentUrlByFileId` method returns the URL to access the IBC binary loader to stream an attachment file through HTTP.

**Returns**

The URL of the IBC binary loader.

### Parameters

fileId - The file ID of the content item attachment that has to be streamed back.

### getAttachmentUrlByItemRefCode

```
public static java.lang.String getAttachmentUrlByItemRefCode
(java.lang.String iRCode,
java.lang.String label)
```

### Description

The `getAttachmentUrlByItemRefCode` method returns the URL to access the IBC binary loader to stream an attachment file through HTTP.

### Returns

The URL of the IBC binary loader.

### Parameters

- `iRCode` - Item reference code of the content item whose attachment has to be streamed back.
- `label` - Label to specify a particular content item's version (pass NULL if not applicable).

### getCItemsMetaByAssoc

```
public static ContentItemMeta[] getCItemsMetaByAssoc(java.lang.String
associationTypeCode,
java.lang.String associatedObjectVal1,
java.lang.String associatedObjectVal2,
java.lang.String associatedObjectVal3,
java.lang.String associatedObjectVal4,
java.lang.String associatedObjectVal5,
java.lang.String label)
throws InputException,
ContentDeliveryException
```

### Description

The `getCItemMetaByAssoc` method returns a list of content items with their metadata based on association and, optionally on label.

### Returns

List of matched content items

### Parameters

- `associationTypeCode` - The association type (required)
- `associatedObjectVal1` - ID value of the associated object (required)
- `associatedObjectVal2` - ID value of the associated object (pass `NULL` if not applicable)
- `associatedObjectVal3` - ID value of the associated object (pass `NULL` if not applicable)
- `associatedObjectVal4` - ID value of the associated object (pass `NULL` if not applicable)
- `associatedObjectVal5` - ID value of the associated object (pass `NULL` if not applicable)
- `label` - The label for retrieving specific approved content item versions that is previously set up (pass `NULL` if not applicable). If a label is passed in, then items that are not linked with the label will not be returned.

### Throws

`InvalidInputException` - Thrown if `associationTypeCode` or `label` is invalid, and if `associationTypeCode` or `associatedObjectVal1` is `NULL`.

`ContentDeliveryException` - Thrown if the content items that match the input criteria should not be returned at runtime for any of these reasons:

- Content item is not approved (status is not `APPROVED`)
- Content item is the home folder
- Content item not being retrieved at a time between its start and end date.

This is also thrown if other system setups are not correct. For example, database connection cannot be established.

### **getCItemsMetaByAssocCtype**

```
public static ContentItemMeta[] getCItemsMetaByAssocCtype
(java.lang.String associationTypeCode,
java.lang.String associatedObjectVal1,
java.lang.String associatedObjectVal2,
java.lang.String associatedObjectVal3,
java.lang.String associatedObjectVal4,
java.lang.String associatedObjectVal5,
java.lang.String contentTypeCode,
java.lang.String label)
```

throws `InvalidInputException`,  
`ContentDeliveryException`

### Description

The `getCitemMetaByAssocCType` method returns a list of content items with their metadata based on association, content type and, optionally on label. Only items that belong to the given association and the given content type will be returned.

### Returns

List of matched content items.

### Parameters

- `associationTypeCode` - the association type (required)
- `associatedObjectVal1` - ID value of the associated object (required)
- `associatedObjectVal2` - ID value of the associated object (pass `NULL` if not applicable)
- `associatedObjectVal3` - ID value of the associated object (pass `NULL` if not applicable)
- `associatedObjectVal4` - ID value of the associated object (pass `NULL` if not applicable)
- `associatedObjectVal5` - ID value of the associated object (pass `NULL` if not applicable)
- `contentTypeCode` - the content type (required)
- `label` - Label for retrieving specific approved content item versions that has been previously set up (pass `NULL` if not applicable). If a label is passed in, then items that are not linked with the label will not be returned.

### Throws

`InvalidInputException` - Thrown if `associationTypeCode` or `label` is invalid, and if `associationTypeCode`, `associatedObjectVal1` or `contentTypeCode` is `NULL`.

`ContentDeliveryException` - Thrown if the content items that match the input criteria should not be returned at runtime for any of these reasons:

- Content item is not approved (status is not `APPROVED`)
- Content item is working directory private

- Content item not being retrieved at a time between its start and end date

Also thrown if other system setups are not correct. For example, database connection cannot be established.

### **getContentItemUrlBasic**

```
public static java.lang.String getContentItemUrlBasic (int contentItemId,  
java.lang.String contentItemLabel,  
int stylesheetId,  
java.lang.String stylesheetLabel)
```

#### **Description**

The `getContentItemUrlBasic` method returns the URL to access the IBC content item loader to stream a content item (basic load), with the specified stylesheet applied, through HTTP.

#### **Returns**

URL of the IBC content item loader.

#### **Parameters**

- `contentItemId` - ID of the content item to be returned.
- `contentItemLabel` - Label to specify a particular content item version (pass `NULL` if not applicable).
- `stylesheetId` - ID of the stylesheet item to be used to apply on the content item.
- `stylesheetLabel` - Label to specify a particular version of the stylesheet (pass `NULL` if not applicable).

### **getContentItemUrlBasic**

```
public static java.lang.String getContentItemUrlBasic  
int contentItemId,  
java.lang.String contentItemLabel,  
java.lang.String stylesheetLabel)
```

#### **Description**

The `getContentItemUrlBasic` method returns the URL to access the IBC content item loader to stream a content item (basic load), with the content type's default stylesheet applied, through HTTP.

#### **Returns**

URL of the IBC content item loader.

#### Parameters

- `contentItemId`- ID of the content item to be returned.
- `contentItemLabel`- Label to specify a particular content item version (pass `NULL` if not applicable).
- `stylesheetLabel`- Label to specify a particular version of the default stylesheet (pass `NULL` if not applicable).

#### **getContentItemUrlDeep**

```
public static java.lang.String getContentItemUrlBasic (int contentItemId,  
java.lang.String contentItemLabel,  
java.lang.String stylesheetLabel)
```

#### Description

The `getContentItemUrlDeep` method returns the URL to access the IBC content item loader to stream a content item (deep load), with the specified stylesheet applied, through HTTP.

#### Returns

URL of the IBC content item loader.

#### Parameters

- `contentItemId` - ID of the content item to be returned.
- `contentItemLabel` - Label to specify a particular content item version (pass `NULL` if not applicable).
- `stylesheetId` - ID of the stylesheet item to be used to apply on the content item.
- `stylesheetLabel` - Label to specify a particular version of the stylesheet (pass `NULL` if not applicable).

#### **getContentItemUrlDeep**

```
public static java.lang.String getContentItemUrlDeep (int contentItemId,  
java.lang.String contentItemLabel,  
java.lang.String stylesheetLabel)
```

#### Description

The `getContentItemUrlDeep` method returns the URL to access the IBC content item loader to stream a content item (deep load), with the content type's default stylesheet applied, through HTTP.

### Returns

URL of the IBC content item loader.

### Parameters

- `contentItemId` - ID of the content item to be returned.
- `contentItemLabel` - Label to specify a particular content item version (pass `NULL` if not applicable).
- `stylesheetLabel` - Label to specify a particular version of the default stylesheet (pass `NULL` if not applicable).

### loadContentItemBasic

```
public static ContentItem loadContentItemBasic (int contentItemId,  
java.lang.String label)  
throws InvalidInputException,  
ContentDeliveryException
```

### Description

The `loadContentItemBasic` method returns the content item with its basic data. Basic data includes the item's metadata, attachment information, all its attributes, and references to any of its component items.

### Returns

Content item with basic data

### Parameters

- `contentItemId` - Content item ID
- `label` - Label for retrieving a specific approved content item version that has been previously set up (pass `NULL` if not applicable). If a label is passed in, then the items that are not linked with the label will not be returned.

### Throws

`InvalidInputException` - Thrown if `contentItemId`, or `label` is invalid.

`ContentDeliveryException` - Thrown if the content item should not be returned at runtime for any of these reasons:

- Content item is not approved (status is not APPROVED)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

Also thrown if other system setups are not correct. For example, database connection cannot be established.

### **loadContentItemBasicWithXSL**

```
public static void loadContentItemBasicWithXSL(int contentItemId,
java.lang.String contentItemLabel,
int stylesheetId,
java.lang.String stylesheetLabel,
java.io.OutputStream out)
throws InvalidInputException,
ContentDeliveryException
```

#### **Description**

The `loadContentItemBasicWithXSL` method loads a content item (basic load), with the content type's default stylesheet applied, into an Output stream.

#### **Parameters**

- `contentItemId` - ID of the content item to be loaded.
- `contentItemLabel` - Label to specify a particular content item version (pass `NULL` if not applicable)
- `stylesheetLabel` - Label to specify a particular version of the default stylesheet (pass `NULL` if not applicable)

#### **Throws**

`InvalidInputException` - Thrown if `contentItemId`, `contentItemLabel`, or `stylesheetLabel` is invalid, and if `out` is `NULL`.

`ContentDeliveryException` - Thrown if the content item or the default stylesheet item should not be retrieved at runtime for any of these reasons:

- Content item is not approved (status is not APPROVED)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

Also thrown if other system setups are not correct. For example, database connection cannot be established.

### **loadContentItemBasicWithXSL**

```
public static void loadContentItemBasicWithXSL
(int contentItemId,
java.lang.String contentItemLabel,
java.lang.String stylesheetLabel,
java.io.OutputStream out)
throws InvalidInputException,
ContentDeliveryException
```

#### **Description**

The `loadContentItemBasicWithXSL` method loads a content item (basic load), with the content type's default stylesheet applied, into an Output stream.

#### **Parameters**

- `contentItemId` - ID of the content item to be loaded.
- `contentItemLabel` - Label to specify a particular content item version (pass `NULL` if not applicable).
- `stylesheetLabel` - Label to specify a particular version of the default stylesheet (pass `NULL` if not applicable).

#### **Throws**

`InvalidInputException` - Thrown if `contentItemId`, `contentItemLabel`, or `stylesheetLabel` is invalid, and if `out` is `NULL`.

`ContentDeliveryException` - Thrown if the content item or the default stylesheet item should not be retrieved at runtime for any of these reasons:

- Content item is not approved (status is not `APPROVED`)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

Also thrown if other system setups are not correct. For example, database connection cannot be established.

### **loadContentItemBasicXml**

```
public static java.lang.String loadContentItemBasicXml
(int contentItemId,
java.lang.String label)
throws InvalidInputException,
ContentDeliveryException
```

### Description

The `loadContentItemBasicXML` method returns a content item with its basic data in XML. Basic data includes the item's metadata, attachment information, all attributes and references to any of its component items.

### Returns

Content item with basic data in XML.

### Parameters

- `contentItemId` - Content item ID
- `Label` - Label for retrieving a specific approved content item version that has been previously set up (pass `NULL` if not applicable). If a label is passed in, items that are not linked with the label will not be returned.

### Throws

`InvalidInputException` - Thrown if `contentItemId`, or `label` is invalid.

`ContentDeliveryException` - Thrown if the content item should not be returned at runtime for any of these reasons:

1. Content item is not approved (status is not `APPROVED`)
2. Content item is working directory private
3. Content item not being retrieved at a time between its start and end date.

Also thrown if other system setups are not correct. For example, database connection cannot be established.

### loadContentItemBasicXmlDom

```
public static org.w3c.dom.Document loadContentItemBasicXmlDom
(int contentItemId,
java.lang.String label)
throws InvalidInputException,
ContentDeliveryException
```

### Description

The `loadContentItemBasicXMLDom` method returns a content item with its basic data in XML. Basic data includes the item's metadata, attachment information, all attributes and references to any of its component items.

### Returns

Content Item with basic data in XML Dom object.

### Parameters

- `contentItemId` - Content item ID
- `label` - Label for retrieving a specific approved content item version that has been previously set up (pass `NULL` if not applicable). If a label is passed in, items that are not linked with the label will not be returned.

### Throws

`InvalidInputException` - Thrown if `contentItemId`, or `label` is invalid.

`ContentDeliveryException` - Thrown if the content item should not be returned at runtime for any of these reasons:

- Content item is not approved (status is not `APPROVED`)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

Also thrown if other system setups are not correct. For example, database connection cannot be established.

### **loadContentItemDeep**

```
public static ContentItem loadContentItemDeep
(int contentItemId,
java.lang.String label)
throws InvalidInputException,
ContentDeliveryException
```

### Description

The `loadContentItemDeep` method returns a content item with its full data. The item's component items are fully loaded instead of being references. If the item's component in turn has some other components, they will also be fully loaded.

### Returns

Content item with full data

### Parameters

- `contentItemId` - Content item ID

- label - Label for retrieving a specific approved content item version that has been previously set up (pass NULL if not applicable). If a label is passed in, items that are not linked with the label will not be returned.

### Throws

InvalidInputException - Thrown if contentItemId, or label is invalid.

ContentDeliveryException - Thrown if the content item should not be returned at runtime for any of these reasons:

- Content item is not approved (status is not APPROVED)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

Also thrown if other system setups are not correct. For example, database connection cannot be established.

### loadContentItemDeepWithXSL

```
public static void loadContentItemDeepWithXSL
(int contentItemId,
java.lang.String contentItemLabel,
int stylesheetId,
java.lang.String stylesheetLabel,
java.io.OutputStream out)
throws InvalidInputException,
ContentDeliveryException
```

### Description

The `loadContentItemDeepWithXSL` method loads a content item (deep load), with the specified stylesheet applied, into an Output stream.

### Parameters

- contentItemId - ID of the content item to be loaded.
- contentItemLabel - Label to specify a particular content item version (pass NULL if not applicable).
- stylesheetId - ID of the stylesheet item to be used to apply on the content item.
- stylesheetLabel - Label to specify a particular version of the stylesheet (pass NULL if not applicable).

### Throws

`InvalidInputException` - Thrown if `contentItemId`, `contentItemLabel`, `stylesheetId`, or `stylesheetLabel` is invalid, and if `out` is `NULL`.

`ContentDeliveryException` - Thrown if the content item or the stylesheet item should not be retrieved at runtime for any of these reasons:

1. Content item is not approved (status is not APPROVED)
2. Content item is working directory private
3. Content item not being retrieved at a time between its start and end date.

Also thrown if other system setup is not correct. For example, database connection cannot be established.

### loadContentItemDeepWithXSL

```
public static void loadContentItemDeepWithXSL
(int contentItemId,
java.lang.String contentItemLabel,
java.lang.String stylesheetLabel,
java.io.OutputStream out)
throws InvalidInputException,
ContentDeliveryException
```

### Description

The `loadContentItemDeepWithXSL` method loads a content item (deep load), with the content type's default stylesheet applied, into an Output stream.

### Parameters

- `contentItemId` - ID of the content item to be loaded.
- `contentItemLabel` - Label to specify a particular content item version (pass `NULL` if not applicable).
- `stylesheetLabel` - Label to specify a particular version of the default stylesheet (pass `NULL` if not applicable).

### Throws

`InvalidInputException` - Thrown if `contentItemId`, `contentItemLabel`, or `stylesheetLabel` is invalid, and if `out` is `NULL`.

`ContentDeliveryException` - Thrown if the content item or the default stylesheet item should not be retrieved at runtime for any of these reasons:

- Content item is not approved (status is not APPROVED)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

Also thrown if other system setups are not correct. For example, database connection cannot be established.

### **loadContentItemDeepXml**

```
public static java.lang.String loadContentItemDeepXml  
(int contentItemId,  
java.lang.String label)  
throws InvalidInputException,  
ContentDeliveryException
```

#### **Description**

The `loadContentItemDeepWithXML` method returns a content item with its full data. The item's component items are fully loaded instead of being references. If the item's component in turn has some other components, they will also be fully loaded.

#### **Returns**

Content item with full data in XML.

#### **Parameters**

- `contentItemId` - Content item ID
- `label` - Label for retrieving a specific approved content item version that has been previously set up (pass `NULL` if not applicable). If a label is passed in, items that are not linked with the label will not be returned.

#### **Throws**

`InvalidInputException` - Thrown if `contentItemId`, or `label` is invalid.

`ContentDeliveryException` - Thrown if the content item should not be returned at runtime for any of these reasons:

- Content item is not approved (status is not APPROVED)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

Also thrown if other system setups are not correct. For example, database connection cannot be established.

### **loadContentItemDeepXmlDom**

```
public static org.w3c.dom.Document loadContentItemDeepXmlDom
(int contentItemId,
java.lang.String label)
throws InvalidInputException,
ContentDeliveryException
```

#### **Description**

The `loadContentItemDeepWithXMLDom` method returns a content item with its full data. The item's component items are fully loaded instead of being references. If the item's component in turn has some other components, they will also be fully loaded.

#### **Returns**

Content item with full data in XML Dom object.

#### **Parameters**

- `contentItemId` - Content item ID
- `label` - Label for retrieving a specific approved content item version that has been previously set up (pass `NULL` if not applicable). If a label is passed in, items that are not linked with the label will not be returned.

#### **Throws**

`InvalidInputException` - Thrown if `contentItemId`, or `label` is invalid.

`ContentDeliveryException` - Thrown if the content item should not be returned at runtime for any of these reasons:

- Content item is not approved (status is not `APPROVED`)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

This is also thrown if other system setups are not correct. For example, database connection cannot be established.

### **loadContentItemMeta**

```
public static ContentItemMeta loadContentItemMeta
(int contentItemId,
```

```
java.lang.String label)  
throws InvalidInputException,  
ContentDeliveryException
```

### Description

The `loadContentItemMeta` method returns a content item with its metadata.

### Returns

Content item with its metadata.

### Parameters

- `contentItemId` - Content item ID
- `label` - Label for retrieving a specific approved content item version that has been previously set up (pass `NULL` if not applicable). If a label is passed in, items that are not linked with the label will not be returned.

### Throws

`InvalidInputException` - Thrown if `contentItemId`, or `label` is invalid.

`ContentDeliveryException` - Thrown if the content item should not be returned at runtime for any of these reasons:

- Content item is not approved (status is not `APPROVED`)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

This is also thrown if other system setups are not correct. For example, database connection cannot be established.

### loadContentItemsMeta

```
public static ContentItemMeta[] loadContentItemsMeta  
(int[] contentItemIds,  
java.lang.String label)  
throws InvalidInputException,  
ContentDeliveryException
```

### Description

The `loadContentItemsMeta` method returns a list of content items with metadata based on their IDs.

### Returns

List of content items.

### Parameters

- `contentItemIds` - List IDs of the content items to be returned
- `label` - Label for retrieving specific approved content item versions that has been previously set up (pass `NULL` if not applicable). If a label is passed in, items that are not linked with the label will not be returned.

### Throws

`InvalidInputException` - Thrown if any of the `contentItemIds`, or `label` is invalid, and if `contentItemIds` is `NULL`.

`ContentDeliveryException` - Thrown if the content items that match the input criteria should not be returned at runtime for any of these reasons:

- Content item is not approved (status is not `APPROVED`)
- Content item is working directory private
- Content item not being retrieved at a time between its start and end date.

This is also thrown if other system setups are not correct. For example, database connection cannot be established.

## E.10 Class Rendition

```
public class Rendition
extends java.lang.Object
```

### Description

Rendition is a specific MIME type of an attachment of a content item.

### E.10.1 Methods Inherited

The following methods are inherited by the Rendition class from the class `java.lang.Object`:

- `equals`
- `getClass`
- `hashCode`
- `notify`

- notifyAll
- toString
- wait
- wait
- wait

## E.10.2 Methods for Class Rendition

Table E-9 is an index of Rendition class methods.

**Table E-9 Method Index for Class Rendition**

Method	Description
<a href="#">getFileId</a>	Returns the file ID of this rendition.
<a href="#">getFileName</a>	Returns the file name of this rendition.
<a href="#">getMimeType</a>	Returns the MIME type of this rendition.
<a href="#">getRenditionName</a>	Returns the translatable name of this rendition MIME type.

### **getFileId**

```
public int getFileId()
```

#### **Description**

The `getFileId` method returns the file ID of this rendition.

#### **Returns**

File ID

### **getFileName**

```
public java.lang.String getFileName()
```

#### **Description**

The `getFileName` method returns the file name of this rendition.

#### **Returns**

File Name

### **getMimeType**

```
public java.lang.String getMimeType()
```

#### **Description**

The `getMimeType` method returns the MIME type of this rendition.

#### **Returns**

MIME type

### **getRenditionName**

```
public java.lang.String getRenditionName()
```

#### **Description**

The `getRenditionName` method returns the translatable name of this rendition MIME type.

#### **Returns**

Rendition name

## **E.11 Exception Classes**

The `ContentDeliveryException` is the exception class for the Oracle Content Manager procedures. The class is located in the package `oracle.apps.ibruntime`.

### **E.11.1 Class ContentDeliveryException**

```
public class ContentDeliveryException  
extends oracle.apps.jtf.base.resources.FrameworkException
```

The `ContentDeliveryException` class extends the `FrameworkException` class and indicates that there are problems in calling the OCM Runtime APIs.

### **E.11.2 Constructors for Class ContentDeliveryException**

#### **ContentDeliveryException**

```
public ContentDeliveryException(java.lang.Exception ex,  
                               java.lang.String errorKey)
```

#### **Description**

Creates a new exception with a base exception and a predefined key.

**Parameters**

- `ex` - Base exception
- `ErrorKey` - `ErrorKey` defined for this exception

**ContentDeliveryException**

```
public ContentDeliveryException(java.lang.String errorKey)
```

**Description**

Creates a new exception with a predefined key.

**Parameters**

`errorKey` - `ErrorKey` defined for this exception

### E.11.3 Fields Inherited

The following fields are inherited by class `ContentDeliveryException` from class `oracle.apps.jtf.base.resources.FrameworkException`.

- `DEBUG`
- `defaultMsgMgr`
- `ERROR`
- `FATAL`
- `INFORMATION`
- `NONE`
- `WARNING`

### E.11.4 Methods Inherited

The following methods are inherited by class `ContentDeliveryException` from class `oracle.apps.jtf.base.resources.FrameworkException`.

- `addException`
- `convertException`
- `getAllInfo`

- `getCurrentMessageManager`
- `getExceptionStack`
- `getExceptionStackRec`
- `getExternException`
- `getKey`
- `getMessage`
- `getMessageManager`
- `getMessageStack`
- `getParameters`
- `getParentExcep`
- `getRootException`
- `getRootExternExcept`
- `getSeverity`
- `getThrowerInfo`
- `getWholeStack`
- `printAllInfo`
- `printAllInfo`
- `printMesg`
- `printMesg`
- `printMessageStack`
- `printMessageStack`
- `printStackTrace`
- `printStackTrace`
- `printThrowerInfo`
- `printThrowerInfo`
- `printWholeStack`
- `printWholeStack`
- `setCurrents`

- `setStackTrace`

The following methods are inherited by class `ContentDeliveryException` from class `java.lang.Throwable`.

- `fillInStackTrace`
- `getLocalizedMessage`
- `printStackTrace`
- `toString`

The following methods are inherited by class `ContentDeliveryException` from class `java.lang.Object`.

- `equals`
- `getClass`
- `hashCode`
- `notify`
- `notifyAll`
- `wait`
- `wait`
- `wait`

## E.12 Data Security Manager

### **establishInheritance**

```
public static void establishInheritance(OracleConnection conn,
                                       String instanceObjectName,
                                       String[] instancePKValues,
                                       String containerObjectName,
                                       String[] containerPKValues,
                                       String inheritanceType,
                                       String commit
                                       )
```

### **Description**

The `establishInheritance` method establishes inheritance tree for security, to define the default permissions for a particular instance. This method should only be called

while creating an instance in Security, or when moving an instance to a different container \* (when an item is moved to a different directory).

### Parameters

- conn: Current JDBC connection
- instanceObjectName: The name defined for the instance object type, check FND\_OBJECTS for values to be passed here.
- instancePKValues: Primary key values for the instance
- containerObjectName: Object Type name for the container of the instance (i.e. in case of an item instance, then supply the object type name for Directory)
- containerPKValues: Primary key values for container
- inheritanceType: Type of inheritance for security (if not sure... pass "FOLDER").
- commit: Flag to commit within PL/SQL procedure

### Throws

```
OracleCallableStatement callableStatement = null;
```

```
String returnStatus;
```

```
int messageCount;
```

```
String messageData;
```

### resetPermissions

```
public static void resetPermissions(OracleConnection conn,  
                                   String instanceObjectName,  
                                   String[] instancePKValues,  
                                   String commit )
```

### Description

The resetPermissions method resets the permissions to inherit from parent and removes all permissions granted directly to the instance.

### Parameters

- conn: Current JDBC connection
- instanceObjectName: Name defined for the instance object type, check FND\_OBJECTS for values to be passed here.
- instancePKValues: Primary key values for the instance

- `commit`: Flag to commit within PL/SQL procedure

### Throws

`OracleCallableStatement callableStatement = null;`

`String returnStatus;`

`int messageCount;`

`String messageData;`

### overridePermissions

```
public static void overridePermissions(OracleConnection conn,  
                                     String instanceObjectName,  
                                     String[] instancePKValues,  
                                     String commit )
```

### Description

The `overridePermissions` method overrides the permissions to break the inheritance with the container. It only applies to the current instances that it inherits.

### Parameters

`conn`: Current JDBC connection

`instanceObjectName`: Name defined for the instance object type, check `FND_OBJECTS` for values to be passed here.

`instancePKValues`: Primary key values for the instance

`commit`: Flag to commit within PL/SQL procedure

### Throws

`OracleCallableStatement callableStatement = null;`

`String returnStatus;`

`int messageCount;`

`String messageData;`

### removeInstance

```
public static void removeInstance(OracleConnection conn,  
                                  String instanceObjectName,
```

```
String[] instancePKValues,  
String commit )
```

**Description**

The `removeInstance` method removes the instance from the Security inheritance tree. This method should be called when you remove the object from the schema (That is, when a folder or a item is permanently deleted.)

**Parameters**

- `conn`: Current JDBC connection
- `instanceObjectName`: Name defined for the instance object type, check `FND_OBJECTS` for values to be passed here.
- `instancePKValues`: Primary key values for the instance
- `commit`: Flag to commit within PL/SQL procedure

**Throws**

```
OracleCallableStatement callableStatement = null;
```

```
String returnStatus;
```

```
int messageCount;
```

```
String messageData;
```

**grantPermission**

```
public static void grantPermission(OracleConnection conn,  
    String instanceObjectName,  
    String[] instancePKValues,  
    String actionCode,  
    String permissionObjectName,  
    String permissionCode,  
    BigDecimal granteeUserId,  
    BigDecimal granteeResourceId,  
    String granteeResourceType,  
    String containerObjectName,  
    String[] containerPKValues,  
    String cascadeFlag,  
    String commit )
```

**Description**

The `grantPermission` method grants a particular permission to an instance.

You can grant a permission at different levels:

**Global:** Permission is granted to everybody at all levels. To grant permissions at the global level, do not provide values for user id, resource type, and resource id.

**User:** Permission is granted to a particular user. Pass the user ID and not the resource ID or the resource type.

**Responsibility:** Permission is granted to all users that have a particular responsibility. You should pass the Resource Type as `RESPONSIBILITY`, resource id must be the responsibility id. Do not pass the user id.

**Group:** Permission is granted to all users belonging to a JTF resource group. You should pass the Resource Type as `RS_GROUP`, and resource id for JTF resource group. Do not pass the user id.

### Parameters

- `conn`: Current JDBC connection
- `instanceObjectTypeName`: The name defined for the instance object type, check `FND_OBJECTS` for values to be passed here.
- `instancePKValues`: Primary key values for the instance.
- `actionCode`: Code for the action of the grant, either "ALLOW" or "RESTRICT"
- `permissionObjectTypeName`: Name of the object associated to the permission granted
- `permissionCode`: Code of the permission being granted. It should be defined in the lookup type for the particular object
- `granteeUserId`: Grantee User Id (If the grant is at user level)
- `granteeResourceId`: Resource Id or Responsibility Id (If the grant is at Group or Responsibility level grant)
- `granteeResourceType`: Either "RS\_GROUP" or "RESPONSIBILITY" depending upon the kind of grant
- `containerObjectTypeName`: Object Type name for the container of the instance (i.e. in case of an item instance, then supply the object type name for Directory)
- `containerPKValues`: Primary key values for container
- `cascadeFlag`: Support cascading of permissions. If not sure send `G_TRUE`.
- `commit`: Flag to commit within PL/SQL procedure

**Throws**

```
OracleCallableStatement callableStatement = null;
String returnStatus;
int messageCount;
String messageData;
int skipAhead;
```

**revokePermission**

```
public static void revokePermission(OracleConnection conn,
                                    String instanceObjectName,
                                    String[] instancePKValues,
                                    String actionCode,
                                    String permissionObjectName,
                                    String permissionCode,
                                    BigDecimal granteeUserId,
                                    BigDecimal granteeResourceId,
                                    String granteeResourceType,
                                    String commit )
```

**Description**

The revokePermission method removes a particular permission from an instance.

**Parameters**

- conn: Current JDBC connection
- instanceObjectName: Name defined for the instance object type, check FND\_OBJECTS for values to be passed here.
- instancePKValues: Primary key values for the instance
- actionCode: Code for the action of the grant, either "ALLOW" or "RESTRICT"
- permissionObjectName: Name of the object associated to the permission granted
- permissionCode: Code of the permission being granted. It should be defined in the lookup type for the particular object
- granteeUserId: Grantee User Id (in the case grant is at user level)
- granteeResourceId: Resource Id or Responsibility Id (in case of Group or Responsibility level grant)

- granteeResourceType: Either "RS\_GROUP" or "RESPONSIBILITY" depending upon the kind of grant
- commit: Flag to commit within PL/SQL procedure

**Throws**

OracleCallableStatement callableStatement = null;

String returnStatus;

int messageCount;

String messageData;

int skipAhead;



---

---

# Index

## A

---

access control  
    inheriting, 4-12  
    overriding, 4-12  
API Reference, E-1  
APPLICATION\_ID, A-3  
assigning  
    AOL responsibilities, 2-3  
    category access control, 4-16  
    folder permission, 4-10  
    folder permissions, 4-12

## C

---

class  
    attribute, E-11  
        constructor, E-11  
        inherited methods, E-11  
        methods, E-11  
    componentitemattribute, E-12  
        constructor, E-13  
        methods, E-14  
        methods inherited, E-13  
    content manager API summary, E-10  
    contentcachemanager, E-14  
        inherited methods, E-14  
        methods, E-15  
    contentdeliveryexception  
        constructor, E-40  
        inherited fields, E-41  
        inherited methods, E-41  
    contentdeliverymanager, E-20  
        inherited methods, E-20

        methods, E-21  
    exception classes, E-40  
        contentdeliveryexception, E-40  
    rendition, E-38  
        inherited methods, E-38  
        methods, E-39  
    concurrent manager, A-2  
    concurrent program  
        BulkUploadCP, D-3  
        OCM InterMedia Index Optimizing operation for  
            Attribute Bundle, D-3  
        OCM InterMedia Index Synchronizing for  
            Attribute Bundles, D-3  
    content type  
        understanding, 4-2  
    creating workflow notifications, 2-5  
    current date, 2-3

## E

---

end date, 2-3  
Examine Field and Variable Values window, A-3

## F

---

finding responsibility ID values, A-3  
FND User, 2-4  
folder security, 4-13

## I

---

IBC\_ROOT\_MENU, 3-3  
implementation task sequence, 3-2  
interface

- content manager API, E-2
- contentitem, E-7
  - inherited fields, E-8
  - inherited methods, E-8
  - methods, E-9
- contentitemmeta, E-2
  - methods, E-3
  - variables, E-3

## J

---

- JTF profiles
  - JTF\_PROFILE\_DEFAULT\_APPLICATION, A-4
  - JTF\_PROFILE\_DEFAULT\_CSS, A-4
  - JTF\_PROFILE\_DEFAULT\_CURRENCY, A-4
  - JTF\_PROFILE\_DEFAULT\_RESPONSIBILITY, A-4

## L

---

- lookups, A-1, A-6

## M

---

- mapping stylesheets, 4-7

## O

---

- OA Framework, 3-7
- OCM Customer UI Responsibilities
  - data level, B-6
  - functional level, B-6
- Oracle Application (OA) Framework, 2-2
- Oracle CRM Technology Foundation, 2-2
- overview
  - API, E-2
  - content type, 4-2
  - oracle content manager, 1-2
  - XML Format, C-2

## P

---

- Personal Home Page, 2-3
- PL/SQL, 3-9
- profile
  - IBC: Content Type Group, A-5

- IBC: Debug Output Directory, A-5
- IBC: Debug Type, A-5
- IBC: Default Home Folder, A-5
- IBC: Display Base Lang, A-5
- IBC: Enable Audit Log, A-5
- IBC: Enable Debug, A-5
- IBC: Enable Encryption, A-5
- IBC: Enforce Available Date, A-5
- IBC: Enforce Expiration Date, A-5
- IBC: Use Access Control, A-2, A-5
- IBC: Use Control, 3-5

- Profile Options
  - Oracle CRM Technology Foundation (JTT), A-4
  - setting, A-2
- profile options
  - Application, A-2
  - Responsibility, A-2
  - Site, A-2
  - User, A-2
- profiles
  - JTT, A-4
  - OCM, A-5
  - setting, A-2

## R

---

- Responsibilities subtab, 2-2
- Responsibilities window, A-3
- responsibility
  - System Administrator, 2-2, 2-4
- responsibility ID finding values, A-3
- running concurrent programs, D-2

## S

---

- seeded content types
  - IBC\_BASIC\_CL, B-3
  - IBC\_CONTENT\_BLOCK, B-3
  - IBC\_COVER\_LETTER, B-4
  - IBC\_FILE, B-4
  - IBC\_IMAGE, B-2, B-4
  - IBC\_QUERY, B-5
  - IBC\_STYLESHEET, B-2
- seeded responsibilities, B-6
- setting JTT profiles, A-4

simple attributes  
  Attachments, 4-2  
  Extended, 4-2  
  Pull-down, 4-2  
stylesheet mapping, 4-7

## **T**

---

types of lookups  
  Extensible, A-6  
  System, A-6  
  user, A-6

## **V**

---

versions, 1-2

## **W**

---

workflow process - graphical representation, 2-6  
Worklist window, 2-6

## **X**

---

XML  
  example for content item, C-3  
  regular expression formats, C-2

