Oracle® Fusion Middleware

UCM VCR Adapter Guide for Oracle WebLogic Portal

10*g* Release 3 (10.3.2)

E14664-01

February 2010

Oracle Fusion Middleware UCM VCR Adapter Guide for Oracle WebLogic Portal, 10*g* Release 3 (10.3.2)

E14664-01

Primary Author: William Witman

Contents

List of Examples

List of Figures

List of Tables

Preface

This document explains how to install and configure the Oracle WebLogic Portal UCM VCR Adapter. In addition, the guide describes best practices for modeling content in the UCM content repository and in Site Studio. These best practices help ensure the best exposure of that content through the VCR. Finally, this guide offers several examples that demonstrate techniques for modeling content in UCM to allow the most reliable mapping of content through the adapter

Audience

This document is intended for portal developers, portal administrators, UCM administrators, and UCM content architects.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. 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. Accessibility 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 more information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation

Screen readers 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, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at `http://www.fcc.gov/cgb/consumerfacts/trs.html`, and a list of phone numbers is available at `http://www.fcc.gov/cgb/dro/trsphonebk.html`.

Related Documents

For more information, see the following documents:

- Oracle Fusion Middleware Content Management Guide for Oracle WebLogic Portal
- Oracle Fusion Middleware Content Management SPI Development Guide for Oracle WebLogic Portal

You can find information on UCM, including Content Server and Site Studio here: http://download-west.oracle.com/docs/cd/E10316_01/ouc.htm.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1 Introduction

This chapter includes the following sections:

- Section 1.1, "What Is the UCM VCR Adapter?"
- Section 1.2, "Adapter Architecture"

1.1 What Is the UCM VCR Adapter?

The UCM VCR Adapter provides access through the WebLogic Portal Content Management Virtual Content Repository (VCR) to content stored in the Oracle UCM system. Oracle UCM is a suite of enterprise content management products that include Oracle Content Server and Oracle Site Studio. The adapter allows WebLogic Portal to access UCM-based content for use in portal applications. For example, you could access content stored in UCM with WLP features like Content Presenter and content selectors.

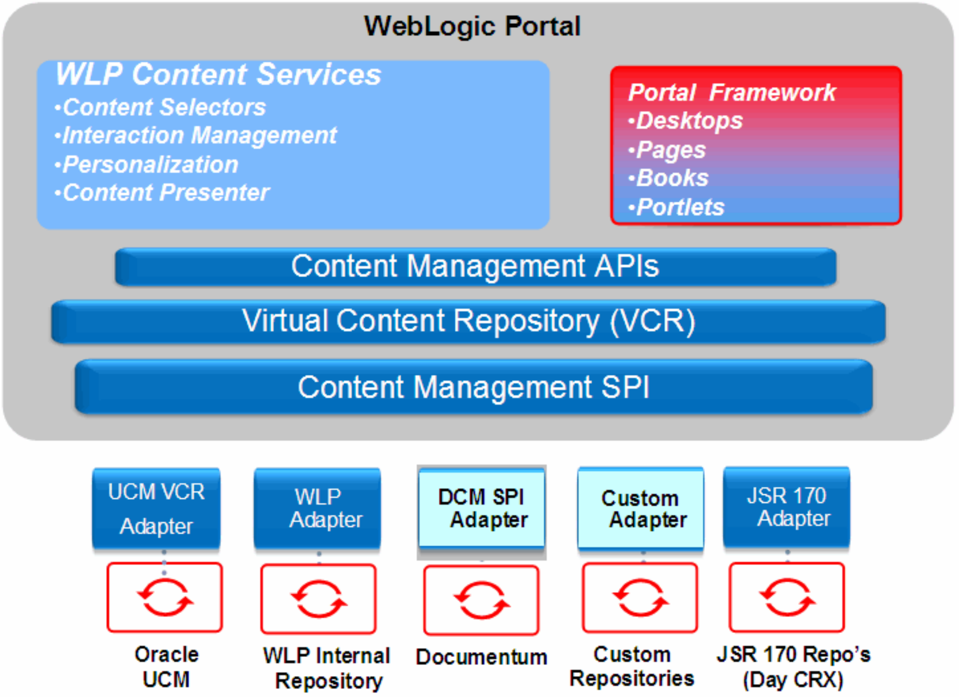
Note: Only Content Server is required to use the adapter. The adapter works whether or not Site Studio is installed on the server. Site Studio is not required; however, extra features are available when Site Studio is installed.

1.2 Adapter Architecture

The UCM VCR Adapter serves as a bridge between the WebLogic Portal Content Management VCR and a UCM repository. The relationships between the various elements are illustrated in Figure 1–1.

Note: Only content in a UCM repository that is specified as State = Released will be accessible through the adapter. Content that is in any other workflow state, or is scheduled for future release, will not be accessible through the VCR.

Figure 1–1 Architecture of the UCM VCR Adapter



2

Installation and Configuration

This chapter includes the following sections:

- Section 2.1, "Installing the UCM VCR Adapter"
- Section 2.2, "Configuring the Adapter"
- Section 2.3, "Configuring UCM"
- Section 2.4, "Security Mapping"
- Section 2.5, "Verifying the Installation"

2.1 Installing the UCM VCR Adapter

The installation process includes these tasks:

- Section 2.1.1, "Ensure the UCM Components are Installed"
- Section 2.1.2, "Add a Reference to the UCM VCR Adapter Shared Library"
- Section 2.1.3, "Modify Cache Settings"

2.1.1 Ensure the UCM Components are Installed

You must have the following UCM components installed to use the UCM VCR Adapter. For details on installing these components, refer to the UCM installation documentation.

2.1.1.1 Server Required Components

1. Oracle Content Server 10gR3 (10.1.3.3.3) with Oracle Text Search (OracleTextSearch), Full Text Search (DATABASE.FULLTEXT), or Metadata-only Search (DATABASE.METADATA) configured.
2. Content Server Update Bundle Component Version 10.1.3.5.0 (CS10gR35UpdateBundle)
3. Folders_g (Included in CS10gR35UpdateBundle; unzip and install the file at component/CS10gR35UpdateBundle/extras/Folders_g.zip)

2.1.1.2 Server Optional Components

1. Site Studio 10gR4 (10.1.4.0.0) component
2. VCR/SPI for Site Studio 10gR4 (component named SiteStudioSPI-Only-10gR4_*.zip)

> **Note:** This component is available on Oracle E-Delivery as one of the components included in part number V16530-01 Oracle Universal Content Management Web Content Management 10gR3 (10.1.3.3.7) for Windows, Linux and Unix.

3. Custom Link Form (ss_docname_link_form.htm) must be checked in. This form is available on OTN.
4. LinkManager8 component (included in CS10gR35UpdateBundle). This component is necessary to resolve SiteStudio document links. For example, this component must be enabled to determine which SiteStudio documents link to a given document.

2.1.1.3 Optional Client Components

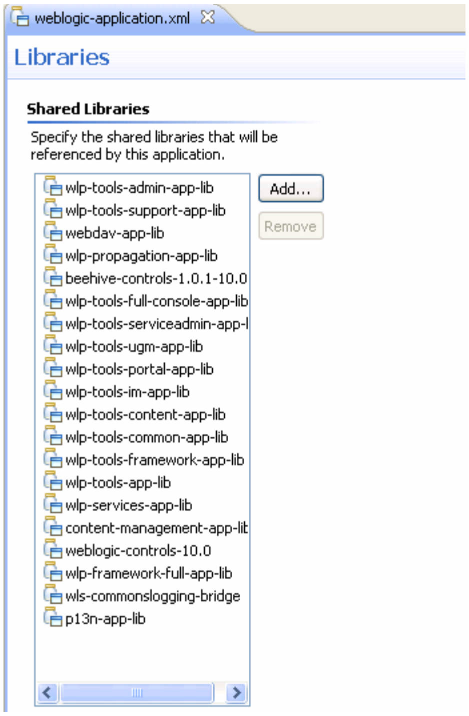
- Site Studio Designer (required for creating and changing region definitions)

2.1.2 Add a Reference to the UCM VCR Adapter Shared Library

Use Oracle Enterprise Pack for Eclipse (OEPE) to add a library reference from your enterprise application to the UCM VCR Adapter library.

1. Start Oracle Enterprise Pack for Eclipse.
2. In the Package Explorer, navigate to the `<EAR_PROJECT>/EarContent/META-INF` folder.
3. Right-click the file `weblogic-application.xml` and select **Open With > WebLogic Deployment Descriptor Editor**.
4. In the editor, select the Libraries tab.
5. Click the Add button, as shown in Figure 2–1.

Figure 2–1 The Libraries Tab



Specify the shared libraries that will be referenced by this application.

- wlp-tools-admin-app-lib
- wlp-tools-support-app-lib
- webdav-app-lib
- wlp-propagation-app-lib
- beehive-controls-1.0.1-10.0
- wlp-tools-full-console-app-lib
- wlp-tools-serviceadmin-app-l
- wlp-tools-ugm-app-lib
- wlp-tools-portal-app-lib
- wlp-tools-im-app-lib
- wlp-tools-content-app-lib
- wlp-tools-common-app-lib
- wlp-tools-framework-app-lib
- wlp-tools-app-lib
- wlp-services-app-lib
- content-management-app-lit
- weblogic-controls-10.0
- wlp-framework-full-app-lib
- wls-commonslogging-bridge
- p13n-app-lib

6. In the Select WebLogic Shared Library dialog, click the **Manage Shared Libraries** link near the bottom of the dialog.
7. In the Preferences dialog, click **Add**.
8. In the Add WebLogic Shared Library dialog, click Browse, and open the following library:

   ```
   <MW_HOME>/wlportal_10.3/content-mgmt/lib/j2ee-modules/
   oracle-ucm-spi-app-lib.ear
   ```

9. In the Add WebLogic Shared Library dialog, click **OK**.
10. Verify that the library was added to the list of Shared Libraries and click **OK**.
11. In the Select WebLogic Shared Library dialog, select the **oracle-ucm-spi-app-lib library**, and click **OK**. The library appears in the Shared Libraries list of `weblogic-application.xml`.
12. Save the file `weblogic-application.xml`.

You have successfully added the UCM VCR Adapter library to your portal project.

2.1.3 Modify Cache Settings

You can configure or adjust UCM repository caches in two ways: in the EAR project file `META-INF/p13n-cache-config.xml` or in the Portal Administration Console (under **Configuration & Monitoring > Service Administration**). Cache settings made in the Portal Administration Console take precedence over the file-based settings. You can also flush the caches from the Portal Administration Console.

Notes: If you already have a `p13n-cache-config.xml` file configured for another repository, simply add the UCM repository cache settings into that file.

Table 2–1 and Table 2–2 describe the available cache settings in `p13n-cache-config.xml`. Table 2–1 describes the VCR caches, which are caches used by the VCR component. These caches are repository specific and can exist for any repository. Table 2–2 describes the SPI caches, which are specific to the UCM SPI Adapter component.

Note: Cache settings described in Table 2–3, " Configuration File Elements" describe the configuration properties for the RIDC security cache, which is used for authorization decisions.

The pattern for the cache entry names is <*elementName*>.<*repositoryName*>, where <repositoryName> is the name of the UCM repository as specified in the `META-INF/content-config.xml` file. For example, `nodeCache.StellentRepository`.

Note that `<time-to-live>` values are specified in milliseconds.

Table 2–1 VCR-Level Cache Entry Descriptions

Cache Entry Name	Description
nodeCache.<*repositoryName*>	Caches node id to node instance for this repository. Defaults: enabled = true; time-to-live = 120000, max-entries = 200.
nodePathCache.<*repositoryName*>	Caches node path to node id for this repository. Defaults: enabled = true; time-to-live = 120000, max-entries = 200.
typeCache.<*repositoryName*>	Caches type id to type instance for this repository Defaults: enabled = true; time-to-live = 600000, max-entries = 200.
typeNameCache.<*repositoryName*>	Caches type name to type id for this repository. Defaults: enabled = true; time-to-live = 600000, max-entries = 200.
binaryCache.<*repositoryName*>	Defaults: enabled = true; time-to-live = 300000, max-entries = 50. **Note:** The maximum binary entry size is specified as the repository property binaryCacheMaxEntrySize which has a default value of 102400 bytes (100 kb).
searchCache.<*repositoryName*>	Caches search results for this repository. Defaults: enabled = true; time-to-live = 300000, max-entries = 500.
nativeAuthCache.<*repositoryName*>	Authorization cache for this repository when using native security. Defaults: enabled = true; time-to-live = 5000, max-entries = 5000.
repositoryConfigCache.<*repositoryName*>	Maximum = 20, time-to-live = (-1) (forever), enabled = true
searchFTSPathCriteriaCache	Maximum = 500, time-to-live = 1 hour
searchExpressionPathCriteriaCache	Maximum = 500, time-to-live = 1 hour

Table 2–2 SPI-Level Cache Entry Descriptions

Cache Entry Name	Description
repo.ucm.typeNameCache.<*repositoryName*>	Caches UCM server type metadata by type name. Defaults: enabled = true; time-to-live = 1800000 (30 minutes), max-entries = 5000
repo.ucm.nodePathToUidCache.<*repositoryName*>	Caches UCM server node ids by node path. Defaults: enabled = true; time-to-live = 1800000 (30 minutes), max-entries = 5000
repo.ucm.nodeUidCache.<*repositoryName*>	Caches UCM server node metadata by node id. Defaults: enabled = true; time-to-live = 1800000 (30 minutes), max-entries = 5000
repo.ucm.securityInfoCache.<*repositoryName*>	Caches UCM server node security information. Defaults: enabled = true; time-to-live = 1800000 (30 minutes), max-entries = 5000
repo.ucm.typeNamesCache.<*repositoryName*>	Caches the list of UCM type names. Defaults: enabled=true, time-to-live=1800000 (30 minutes), max-entries=5000.
repo.ucm.indexedFieldsCache.<*repositoryName*>	Caches information about which UCM fields are indexed. Defaults: enabled=true, time-to-live=1800000 (30 minutes), max-entries=5000.
repo.ucm.securityUserCache.<*repositoryName*>	Stores the association between a WLP user and the associated UCM server userID. Defaults: 5000 entries, time-to-live = 30 minutes
repo.ucm.searchCriteriaCache.<*repositoryName*>	Caches search criteria so repeated searches perform well. Defaults: enabled=true, time-to-live=3600000 (1 hour), max-entries=500.

Note: To configure the UCM SPI RIDC security cache settings described in Table 2–3, you must add the entries as `<repository-property>` elements in `content-config.xml`.

Example 2–1 presents a sample `META-INF/p13n-cache-config.xml` file.

Example 2–1 Sample p13n-cache-config.xml File

```
<?xml version="1.0" encoding="UTF-8"?>
<p13n-cache-config xmlns="http://www.bea.com/ns/p13n/90/p13n-cache-config">
<cache>
    <name>nodeCache.MyUcmRepo</name>
    <enabled>true</enabled>
    <time-to-live>600000</time-to-live>
    <max-entries>5000</max-entries>
</cache>
<cache>
    <name>repo.ucm.nodePathToUidCache.StellentRepository</name>
    <enabled>true</enabled>
    <time-to-live>10000</time-to-live>
    <max-entries>2000</max-entries>
</cache>
</p13n-cache-config>
```

2.2 Configuring the Adapter

After the adapter is added to your WLP project, you are ready to configure it. Configuring means specifying how the adapter will talk to the UCM content repository. For example, the adapter has to know where the UCM server is running, its port number, an admin account name, and so on.

There are two configuration scenarios to consider. If you primarily work with the IDE in a development environment, you can configure the adapter by editing a configuration file. If you are a portal administrator who uses the WLP Administration Console, you can use the Administration Console to configure the adapter.

2.2.1 Configuring the Adapter in a Development Environment

This section explains how to configure the adapter using a configuration file. Developers working primarily in the IDE commonly use this method.

Note: Because of the way in which WLP uses deployment plans, adapter configurations made in the WLP Administration Console take precedence over configurations made in the `content-config.xml` file.

1. Open the `content-config.xml` file. This file is located in the `META-INF` directory of your Portal EAR Project.
2. Edit the required elements in the file using Table 2–3 and Example 2–2 as a guide.
3. Save the configuration file.
4. Redeploy the portal application.

Table 2–3 describes the elements in `content-config.xml` that you must configure to successfully use the adapter.

Table 2–3 Configuration File Elements

Element	Description
class-name	(Required) The class name of the adapter, which is com.oracle.content.spi.ucm.RepositoryImpl.
username	(Optional) If you use the global config user security option (see Section 2.4.1, "WLP-Secured Global Config User" or Section 2.4.2, "Natively-Secured Global Config User"), this element specifies the UCM Server user name that is used to retrieve data from UCM, regardless of the WLP user. In other words, all WLP users in an enterprise application will see the data for this UCM repository instance. **Note:** If you configure the identity propagation option, do not specify this element (see Section 2.4.3, "Identity Propagation with Native Security").

Table 2–3 (Cont.) Configuration File Elements

Element	Description
ContentServerHostname	(Required) The hostname or IP address of the server on which the UCM repository is running.
ContentServerPort	(Required) The Intradoc port number on which the UCM repository is listening. This corresponds to the IntradocServerPort setting in the Content Server configuration file, which defaults to port 4444. You can also find this setting by accessing the UCM server with a browser as an admin, clicking on **Administration** > **Configuration** for xxx, then clicking on the **Server Configurations** link. Configure the UCM SPI to use the 'Server Port'.
ContentServerAdminUser	(Required) The user name of a valid administrative user of the UCM repository. This user must have the admin role.
ContentServerServiceType	(Required) Protocol used to connect with the UCM Server. Valid options are INTRADOC or INTRADOC_SSL. INTRADOC is the default.
ContentServerSSLKeyStoreFile	Required (and only used) if ContentServerServiceType is INTRADOC_SSL. The absolute path to the client SSL keystore file.
ContentServerSSLKeyStorePassword	SSL keystore password. Required (and only used) if ContentServerServiceType is INTRADOC_SSL. The SSL keystore password.
ContentServerSSLKeyStoreAlias	Required (and only used) if ContentServerServiceType is INTRADOC_SSL. The SSL keystore alias name.
ContentServerSSLKeyStoreAliasPassword	Required (and only used) if ContentServerServiceType is INTRADOC_SSL. The SSL keystore alias password.
CacheInvalidationInterval	(Optional) Polling interval in minutes for the UCM Adapter CacheInvalidator IntervalJob (must be equal to or greater than 2 minutes). This cache automatically invalidates cached content in WLP as content is changed on the UCM server. If you set it to 0, automatic cache invalidation will be disabled for this repository.
folder_badge_objectClasses	(Required) Specifies a list of UCM ObjectClasses that are displayed as folders in the WLP Administration Console. For the UCM VCR adapter, set this value to IDC:Folder.
TypeRetrievalShapeUser	(Required) Specifies a UCM user account that is used to retrieve all types (ObjectClasses) from UCM. UCM profiles can take into account the UCM user when determining which fields are on a profile. WLP uses this setting to ensure that the UCM type retrieval is identical regardless of the WLP user. This user account does not need any special roles on the UCM server. **Note:** If you want to create documents of various security groups, the configured `TypeRetrievalShapeUser` must have at least read privilege to those security groups. The default value for this element, `typeShapeUser`, has read privileges for all profiles.

Table 2–3 (Cont.) Configuration File Elements

Element	Description
useNativeSecurity	When this element is enabled, the native UCM security will be used to secure content. When disabled, the VCR federated security (entitlements) will be used to secure content in addition to the native UCM security. Default is false (disabled). For more information, Section 2.4, "Security Mapping." Note: For best performance, use native security (useNativeSecurity = true).
SecurityCacheSize	Used in the default RIDC security cache configuration size. Add this entry as `<repository-property>` element in `content-config.xml`. Default = 500
SecurityUserCacheTtl	Used in the default RIDC security cache configuration. Add this entry as `<repository-property>` element in `content-config.xml`. Default=10*60*1000=10 minutes
SecurityServerCacheTtl	This time to live value is only used if accounts or ACL are used. Add this RIDC security cache entry as `<repository-property>` element in `content-config.xml`. Default=10*60*1000=10 minutes
ContentServerConnectionTimeout	(Optional) The maximum time (in msec) to wait when connecting to the UCM server. Default=30 seconds
ContentServerConnectionPoolSize	(Optional) Optional. The maximum connection pool size. Default=20
ContentServerSocketTimeout	(Optional) The maximum time to wait (in msec) for a UCM server response. Default=70 seconds.
ForceDeleteFolders	(Optional) only used if the Trash folder is enabled on the UCM instance. If set to true, then deleting a folder or document will remove it immediately, bypassing the trash. If set to false, then deleting a folder or document will move it to the trash.

Example 2–2 shows a sample configuration file with the required elements highlighted in bold type.

Example 2–2 Sample Configuration for the UCM VCR Adapter

```
<!-- UCM test repository -->
<content-store>
   <name>UCMRepository</name>
           <class-name>com.oracle.content.spi.ucm.RepositoryImpl
           </class-name>
           <username>wlpApp</username>
       <repository-property>
           <description>Hostname on which the UCM Content Server is running
           </description>
           <name>ContentServerHostname</name>
           <value>10.221.19.72</value>
       </repository-property>
       <repository-property>
           <description>Port on which the UCM Content Server is running (on
    host named above)
    </description>
    <name>ContentServerPort</name>
    <value>4444</value>
</repository-property>
<repository-property>
    <description>Content Server uname in Admin role, for
    security checks
    </description>
    <name>ContentServerAdminUser</name>
    <value>sysadmin</value>
</repository-property>
<repository-property>
    <description>List of folder ObjectClasses</description>
    <name>folder_badge_objectClasses</name>
    <value>IDC:Folder</value>
</repository-property>
<repository-property>
     <description>Single user for all type related interactions
     </description>
      <name>TypeRetrievalShapeUser</name>
      <value>typeShapeUser</value>
</repository-property>
<repository-property>
     <name>useNativeSecurity</name>
     <value>false</value>
</repository-property>
<repository-property>
    <description>Protocol used to connect with the UCM Server.
    Valid options are INTRADOC or INTRADOC_SSL.  INTRADOC is the
    default.
    </description>
    <name>ContentServerServiceType</name>
    <value>INTRADOC</value>
</repository-property>
<repository-property>
    <description>Content Server Username in Admin role, used
    for security checks
    </description>
    <name>ContentServerAdminUser</name>
    <value>sysadmin</value>
</repository-property>
<repository-property>
    <description>Polling interval for the
    CacheInvalidator IntervalJob
    (must be equal to or greater than 2 min)
    </description>
    <name>CacheInvalidationInterval</name>
    <value>2</value>
</repository-property>
<repository-property>
    <description>Location on the filesystem of the
    client SSL keystore
    file. Only used when ContentServerServiceType
    is INTRADOC_SSL
    </description>
    <name>ContentServerSSLKeystoreFile</name>
    <value>c:/client_keystore</value>
</repository-property>
<repository-property>
```

```
                <description>SSL keystore password. Only used when
                ContentServerServiceType is INTRADOC_SSL
                </description>
                <name>ContentServerSSLKeystorePassword</name>
                <value>idcidc</value>
            </repository-property>
            <repository-property>
                <description>SSL keystore alias name.  Only used when
                ContentServerServiceType is INTRADOC_SSL
                </description>
                <name>ContentServerSSLKeystoreAlias</name>
                <value>SecureClient</value>
            </repository-property>
            <repository-property>
                <description>SSL keystore alias password.  Only used when
                ContentServerServiceType is INTRADOC_SSL
                </description>
                <name>ContentServerSSLKeystoreAliasPassword</name>
                <value>idcidc</value>
            </repository-property>
            <read-only>true</read-only>
            <binary-cache-max-entry-size>102400</binary-cache-max-entry-size>
             <!-- metadata search -->
             <search-is-enabled>true</search-is-enabled>
             <!-- full text search -->
             <fulltext-search-is-enabled>true</fulltext-search-is-enabled>
             <search-indexing-is-enabled>false</search-indexing-is-enabled>
    </content-store>
```

2.2.2 Configuring the Adapter in the Administration Console

Users who typically work on staged or production portals use the WebLogic Portal Administration Console to configure content repositories. This section summarizes the basic configuration steps for this scenario. For more detailed instructions, see "Configuring Additional Repositories" in the Oracle Fusion Middleware Content Management Guide for Oracle WebLogic Portal.

Once the UCM repository is deployed and running it must be connected to WebLogic Portal so that it appears as part of its Virtual Content Repository. To add the UCM repository:

1. Open the WebLogic Portal Administration Console.
2. Select the Content tab to enter the Content Management page.
3. Select the **Repositories** link to view the repositories. This link is above the tree view on the left.
4. Click **Virtual Content Repository** at the top of the repository tree.
5. In the editor pane to the right, click the **Add Repository Connection** button.
6. Complete the Add Repository Connection dialog with the following information:

Field	Description
Name	A name you choose for the adapter.
Connection Class	(Required) The class name of the adapter, which is com.oracle.content.spi.ucm.RepositoryImpl.
Datasource JNDI name	Not required.

Field	Description
Username	If using the global config user option (all users of the application use the same credentials to connect to UCM), this field specifies the user name of a valid user of the UCM repository. If not using the global config user option, leave this field blank. For more information on security, see Section 2.4, "Security Mapping."
Password	Not necessary for the UCM VCR Adapter.
Enable Library Services	Set this field to false.

7. Click **Create**. The new repository appears in the repository tree on the left hand side.

> **Note:** If you receive an error after clicking **Create**, click the **Refresh Tree** button. The words “can’t connect” appear next to the new repository name in the Administration Console. To correct the problem, select the new repository and use the Add Property dialog to add the required properties indicated by the error message: ContentServerHostname, ContentServerPort, ContentServerAdminUser.

8. Using the Administration Console, configure the same repository properties listed and described previously in Table 2–3, " Configuration File Elements".

2.3 Configuring UCM

This section explains how to configure UCM to work with the UCM VCR Adapter. Refer to the UCM documentation for details on performing the configurations described in this section.

2.3.1 Enable Connections on the Intradoc (RIDC) Server Port

The Intradoc server port requires two configurations:

- **IntradocServerPort** – The UCM SPI accesses the UCM Content Server through the RIDC libraries. By default, the RIDC libraries communicate with the UCM server over the Intradoc server socket port.
- **SocketHostAddressSecurityFilter** – The security filter setting specifies which client hosts are allowed to contact the UCM server over the Intradoc server socket port. The security filter must be configured to support, at a minimum, the set of hosts that the UCM SPI runs on.

To configure these server port settings:

1. In your UCM server `config.cfg` file, enable the Intradoc server socket port and the security filter, by adding lines like the following:

```
IntradocServerPort=4444
SocketHostAddressSecurityFilter=10.143.121.150 or *.*.*.*
```

2. Restart the UCM server.

> **Note:** See the UCM documentation for more details on these settings.

2.3.2 Define a Profile Trigger Field

By default, a UCM server does not have a profile trigger field configured. This section explains how to select a profile trigger field if one has not already been selected.

> **Note:** The trigger field is an important setting in the UCM server. This field is used to customize the Content Server UI for various types of data. Once the trigger field is selected, it is difficult to change the setting, and doing so can affect the Content Server UI. It's important to carefully consider which trigger field to use. See UCM documentation for more details.

To select a profile trigger field:

1. In a browser, access the content server and log in as an Administrator.
2. Select **Admin Applets** from the **Administration** menu.
3. Open the **Configuration Manager** applet.
4. Select the **Profiles** tab.
5. At the top of the screen, the current Trigger Field is displayed; for example: **Trigger Field: dDocType**.
6. If the Trigger Field is not currently configured, click **Select** and select the desired field.

2.3.3 Optimize the dOriginalName Field

> **Note:** Only perform this task if the Content Server is configured to use the OracleText search engine. You can verify configuration, inspect the `SearchIndexerEngineName` setting in the UCM `config.cfg` file. If it is set to `OracleTextSearch`, then the OracleText search engine is in use.

In the UCM server, the `dOriginalName` field stores the filename of a document. By default, the OracleText search engine performs substring matches when searching on filenames, even if the VCR client requests an exact match.

To configure UCM so exact matches are possible:

1. Select **Text Search Admin** from the **Administration** menu.
2. In the set of non-optimized fields on the left-hand side, in the Special section, select **Original Name** and click the arrow to move it to the right side of the screen.
3. Click **Update**.
4. If necessary, rebuild the search index as follows:
    1. Select **Admin Applets** from the **Administration** menu.
    2. Open the **Repository Manager** applet and select the **Indexer** tab.
    3. In the **Collection Rebuild Cycle** section, click **Start** to begin indexing.

When the State is Finished, the indexing is complete.

2.4 Security Mapping

There are three ways to configure the security and credential mappings between the UCM Adapter and the UCM instance.

- Section 2.4.1, "WLP-Secured Global Config User"
- Section 2.4.2, "Natively-Secured Global Config User"
- Section 2.4.3, "Identity Propagation with Native Security"

Note: The VCR adapter user identity propagation to the UCM content server will not work properly if the UCM Content Server users that are being referenced by WLP exist in an ActiveDirectory and the UCM Content Server has been configured to use a non-LDAP based ActiveDirectory. This configuration is not supported.

A local or global user identity store will work, as will an external LDAP-based store (either ActiveDirectory or other LDAP).

Note: For best performance, use native security (useNativeSecurity = true).

Note: By default, UCM allows users that do not exist in the UCM user store to connect. They are implicitly mapped to the guest role.

For example, if you configure an invalid UCM username in the adapter configuration, WLP will access UCM as a user in the guest role.

If not immediately apparent, this will generally become noticeable when you go to create content or edit content via the WLP Adapter, since users in the guest role generally cannot create or edit content in UCM.

2.4.1 WLP-Secured Global Config User

All users in the application use the same credentials to connect to the UCM instance. The data is secured through both UCM (based on the username) and WebLogic Portal. This is the configuration to use if you plan to use WLP Entitlements and/or Delegated Administration on content. Configure this type of security with these settings in the `META-INF/content-config.xml` file in the EAR Project, as described in the following table.

Field	Value
Username	The user name of a valid user of the UCM repository. The UCM content this user can access may be further reduced by applying entitlements in WLP. If you plan to use WLP Entitlements or Delegated Administration, select a username with the desired level of access to content. WLP users will have 'at most' the privileges of this UCM username.
useNativeSecurity	false

2.4.2 Natively-Secured Global Config User

All users in the application use the same credentials to connect to the UCM instance. Data is secured through UCM. WLP Content Entitlements will not apply. Configure this type of security with these settings in the `META-INF/content-config.xml` file in the EAR Project:

Field	Value
Username	The user name of a valid user of the UCM repository. The UCM content this user can access will be exposed in WLP.
useNativeSecurity	true

2.4.3 Identity Propagation with Native Security

WebLogic Portal and UCM use the same security store, such as a shared LDAP store. Each WLP user sees in the application what they are permitted to see on the UCM side. Entitlements (WLP content security) are disabled.

Field	Value
Username	Do not set a value for this field.
useNativeSecurity	true

2.5 Verifying the Installation

To verify the installation, open the WLP Administration Console, go to the Content Management section, and attempt to browse to the UCM repository and view its contents or types.

3 Using the Adapter

This chapter includes these sections:

- Section 3.1, "Content Design Considerations"
- Section 3.2, "Performing Create, Update, Delete, and Move Operations"
- Section 3.3, "Creating Required Fields"
- Section 3.4, "Working with Unfiled Documents"
- Section 3.5, "Node Property Loading"
- Section 3.6, "Performing Searches"
- Section 3.7, "Scoping Content Selectors and Other Queries to a Repository"
- Section 3.8, "UCM Content Modeling Examples"

3.1 Content Design Considerations

The UCM VCR Adapter provides the conduit through which the WLP Virtual Content Repository (VCR) accesses content in UCM. To function properly, the adapter expects content structures that can be mapped to the structures used in the VCR. For example, UCM uses profiles to manage how metadata is structured and displayed. However, the VCR does not use profiles; therefore, the adapter is required to provide reliable mappings between the two systems.

This section discusses the considerations for modeling content in the UCM to facilitate exposure in the VCR and includes these sections:

- Section 3.1.1, "Content Modeling and Mapping Overview"
- Section 3.1.2, "Best Practices for Modeling Content"
- Section 3.1.3, "Unsupported Features for the UCM VCR Adapter"
- Section 3.1.4, "Unsupported UCM Features"
- Section 3.1.5, "Unsupported Web Content Management (WCM) Features"
- Section 3.1.6, "Property Field Mappings"

Note: If a UCM parameter is changed that might affect search, then the WebLogic Portal server must be restarted.

3.1.1 Content Modeling and Mapping Overview

Content modeled in UCM is modeled using some combination of the following constructs:

- Document – A basic content item construct that includes a full set of metadata fields. Every document contains the same set of fields.
- Content profile – Content profiles allow UCM administrators to express which metadata fields are associated with a given document. Content profiles created by a system administrator can refine the metadata options available to users during check in and searching, as well as what metadata is displayed on a content information page. See also Section 3.8.1, "Creating Profile-Based Content Types with the UCM Administration Console."
- Site Studio region definitions – An add-on container construct created in Site Studio that allows definition of types (region definitions) composed of properties (content elements), in addition to the standard document fields. For example, a region definition could include a combination of text and graphic elements. See also Section 3.8.2, "Creating Region-Based Content Types with Oracle Site Studio."

How these constructs map to WLP VCR content types and nodes are described in the following sections.

3.1.1.1 How Documents are Mapped to VCR Content Types

Documents not associated with a content profile map to a content type named IDC:GlobalProfile in the VCR that includes all of the metadata fields. All of the metadata fields you see in UCM will appear in the VCR type called IDC:GlobalProfile.

To ensure that content profiles are correctly mapped to VCR content types, keep in mind the following mapping behavior:

- Each metadata field in a content profile will correspond to a VCR property definition.
- A single binary property definition named idcPrimaryFile will always relate to the checked-in document. This binary property definition will be the primary property definition for the content type.
- Other metadata fields of type Text, Long Text, Memo, Integer, Decimal, and Date will map to VCR property data types of String, Long, Double, and Calendar.

> **Note:** For information on length constraints for the UCM types Text, Long Text, Memo, Integer, Decimal, and Date, see the Oracle Content Server documentation.

- Text metadata fields using an option list based on the YesNoView or TrueFalseView views will be exposed in WLP as a property of type Boolean.

3.1.1.2 How Content Profiles are Mapped to VCR Content Types

Each content profile in UCM is exposed in the VCR as a content type. The profile's associated fields (defined by rules) are exposed on the VCR type.

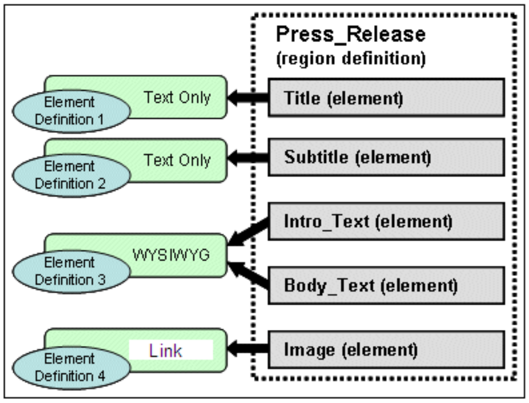
3.1.1.3 How the Combination of Site Studio Region Definitions and Profiles are Mapped to VCR Content Types

Oracle Site Studio is a web development program that offers structured content features. In Site Studio, groups of individual content elements are arranged in region definitions. Each region definition in UCM is exposed in the VCR as a content type. If the region definition is associated with a profile, then the VCR content type will contain property definitions from both the region definition as well as from the profile.

If the region definition is not associated with a profile, then the VCR content type will contain property definitions from both the region definition as well as the standard document metadata fields (the fields from IDC:GlobalProfile).

Figure 3–1, illustrates a region definition called "Press_Release," which consists of the elements Title, Subtitle, Intro_Text, Body_Text, and Image.

Figure 3–1 Elements Assembled into a Region



To ensure that content regions are correctly mapped to VCR content types, keep in mind the following mapping behavior:

- Element definitions map to VCR property definitions. Note that unlike metadata property definitions associated with profiles, property definitions that are defined as part of a region are only searchable by doing a full text search.
- Element definitions map to string, link, or nested data types.
  - Links will be represented when a content custom element form called SS_DOCNAME_LINK_FORM is used that restricts the population of the link field to something Content Server understands as a link.
  - Nested data types will be represented when a Site Studio static list is used. A static list is inherently one level deep and may only contain element definitions of type string (plain text or wysiwyg) and link. Static lists are inherently multi-valued.
  - The static list will be represented as an abstract content type by the VCR. The static list itself acts as a definition for an additional content type comprised of the element definitions in the static list.
  - Both plain text and wysiwyg element types are exposed as String values in the VCR.
- Region definitions, by default, receive all of the property definitions exposed by the global profile type. The content type for the region will extend the IDC:GlobalProfile content type.
- Region definitions may also be associated with a content profile. In this case the set of element definitions from the region, along with the set of metadata fields from the profile, will be represented as a single type by the VCR. The content type for the region will extend the profile content type.

3.1.2 Best Practices for Modeling Content

The VCR uses content types to describe content metadata. Content types are used to define the metadata that you can associate with content. When content contributors add content to a WLP VCR, they can associate the content with a content type. In the same way, when UCM content is surfaced in the VCR, the UCM VCR Adapter must provide some degree of mapping to ensure that the content types are structured correctly. This section describes best practices for modeling data in a WLP Repository or other repository to facilitate migration to the UCM Adapter, and to understand the mapping behavior of the UCM Adapter.

Table 3–1 describes the UCM modeling considerations for VCR content types. For more information on VCR content types, see the *Oracle Fusion Middleware Content Management Guide for Oracle WebLogic Portal*.

Table 3–1 Content Type Modeling Considerations

VCR Content Types	UCM Modeling Consideration	Recommendation
Multiple Binaries	You can have only one binary property per content type.	Use at most one binary per content type. Or use a region-based set of elements that use the custom link type to associate multiple binaries documents with a content type.
Multi-valued Property Definitions	When using profiles, there is limited support for multi-valued properties; only string and SiteStudio static list data types are supported. The data in the UCM is persisted as a single comma-separated value.	Use only multi-valued string property definitions, and avoid using comma values for these properties.
Type Inheritance	Site Studio static list types are exposed as abstract nested types, as are the set of document binary renditions (in a property named idcRenditions).	Understand how region definitions map to content types using type inheritance.
Relationship between Profile Trigger Value and ObjectClass	Each profile in UCM must have a unique trigger value. It is not supported to have multiple profiles with the same trigger value.	Ensure each profile in UCM has a unique trigger value.
Nested Types	Nested types are not supported outside of the single-level nesting support described by the "static list" feature of region definitions.	Understand how region definition static lists and document renditions are exposed as nested types.
Property Definition Names	When using profiles, the maximum length of a metadata field is 29 characters in the UCM, with no embedded spaces.	Limit property definition names to at most 29 characters with no embedded spaces.
Primary Property Definition	In UCM there is no notion of a "primary" property definition. The UCM VCR Adapter will represent the single binary property of a profile-based document as a primary property definition named idcPrimaryFile. Regions will not expose a primary property definition.	Understand how primary properties are represented by the UCM VCR Adapter.
Type Metadata	Content types in UCM do not include the following metadata properties: createdBy, modifiedBy, createdDate, modifiedDate. There is no way to determine who created/modified a type and when it was last created/modified. By contrast, Documents and Folders do support these properties; it is possible to determine who created/modified a doc/folder and when this was done.	Avoid use of these fields on the ObjectClass object in application development.
Property Definitions	The UCM recommends a maximum of 500 custom property definitions for use across all types.	Do not exceed the maximum number of total property definitions in the UCM system.

Table 3–2 describes the UCM modeling considerations for VCR content nodes. In the VCR, the repository is usually represented as a hierarchical collection of nodes. Nodes primarily include folders and content items.

Table 3–2 Content Node Modeling Considerations

VCR Content Nodes	UCM Modeling Considerations	Recommendations
Folders	In UCM, there is a single folder type, named IDC:Folder, that can contain child content, but this type cannot contain folder-specific metadata. The metadata stored on folders is intended to work as default values for the content stored in those folders.	Be aware that folder metadata in UCM refers to child document defaults, rather than folder-specific metadata.
Node Hierarchy	Only folders may have child nodes.	Be aware that nodes cannot be created beneath document nodes.
Node Metadata	The modifiedBy and modifiedDate metadata on nodes have a different meaning than what is exposed by the WLP repository. For UCM documents, these field values are the same as the createdBy/createdDate field values. The values reflect the most recent document checkin user and timestamp. These values do not change for metadata-only updates.	Be aware of this difference in how this data is updated.
String Values	The UCM tools support strings up to 2000 characters in length. The WLP repository supports 4000 character string values.	Keep string values under 2000 characters, or update the UCM schema to support a wider value.
Child Node Ordering	Child node ordering is not supported in UCM.	Do not rely on a given order for child nodes. Alternatively, callers could sort responses from the API or issue queries with sort criteria specified to sort results on a per-call basis.
Integer Values	Integer values default to a zero (0) instead of a null value. The WLP repository stores unset integer values as null.	Do not depend on the presence of a property value for an integer property to detect if that value has been set.

Table 3–3 lists several considerations specific to the search integration.

Table 3–3 Search Considerations and Recommendations

Consideration	Recommendation
If the search is not scoped to documents (cm_isContent = true) or folders (cm_isHierarchy = true), both documents and folders will be searched. Different considerations apply to each, especially around use of cm_path.	When possible, scope the search appropriately by adding `cm_isContent=true` or `cm_isHierarchy=true`. This scope also improves performance.
UCM provides limited support for querying on null or non-null values.	Be aware of the differences in search behavior and do not write search expressions that depend on unsupported criteria.
Recursive search for folders is not supported by UCM, but is supported for documents if configured on the UCM server as described to the right.	Scope the search to only include documents (add a clause like 'cm_isContent = true'. Set the search path on the Search object. Configure the folders_g CollectiveSearchRecursiveContent and related settings like CollectionMaxBranch in the UCM config.cfg file.
The `cm_value` (search across all properties) criteria is not supported.	Do not use `cm_value` in search criteria; it will throw an Exception.
The `likeignorecase` operator is not supported.	The `likeignorecase` operator behaves as the `like` operator.

Table 3–3 (Cont.) Search Considerations and Recommendations

Consideration	Recommendation
Multivalued property operators perform substring matches. This is true for 'containsany' and 'containsall'. In UCM, a field with an option list stores values in a comma-delimited manner. For example, if you have values "A", "B", and "C", these will be represented as "A, B, C". Using a 'containsany' search for 'A, B' will find this item.	Be aware of the differences in search behavior and consider changing the UCM option list delimiter character in the Configuration Manager applet to reduce the potential for finding extra matches.
When searching folders (cm_isHierarchy=true), at most one value can be specified per criteria. Each criteria is logically ANDed with the others to make a more selective query. There is no support for OR or NOT when searching folders.	There is no support for `OR` and `NOT` in UCM folder search.
UCM supports searching for folders and documents, but this is done in two separate service calls. Some properties that exist on a document (for example, `cm_binarySize`) will not exist on folders, so if the search query criteria does not limit the search to documents only, an InvalidPropertyException will be thrown. Fields can be configured as searchable in the UCM Configuration Manager applet.	Add the content type to your search query if you know the property will be only on a document. Example: `cm_objectClass != IDC:Folder` or `cm_isContent = true`
Not all properties are searchable, and if the search encounters a property that is not searchable, it will throw an InvalidPropertyException.	Understand which properties are searchable for a given content type by examining the UCM Configuration Manager information fields section, or by reviewing the ObjectClass definition. Example URLS: `http://myContentServer/idc/idcplg?IdcService=VCR_GET_CONTENT_TYPE&vcrContentType=IDC:Folder&IsSoap=1` then look for the isSearchable field setting.
Not all ObjectClasses are searchable. An attempt to search for a non-searchable ObjectClass will throw an exception. For example, the IDC:FileReference ObjectClass is not searchable.	Be aware that not all ObjectClasses are searchable.
Only String multi-valued properties can be searched.	Don't specify search for multi-valued property types other than String.
`cm_typePath` is not a supported search attribute.	Don't use `cm_typePath` in search criteria.
The `not` operator cannot be used for LONG properties.	Try to restructure the query using supported syntax.
Sorting on non-indexed fields results in an exception. Searching on a non-indexed field throws an exception, with the embedded exception code of, for example, "DRG-10837: section dStatus does not exist".	Understand which fields have been indexed before using them as sort criteria Example: URL: `http://myContentServer/idc/idcplg?IdcService=GET_ADVANCED_SEARCH_OPTIONS&IsSoap=1`, look for "IsSortable"
Null cannot be used to specify a date in date ranges.	Restructure your query using a date range with two valid dates, or use the before or after construction. Example: See Javadoc for com.bea.content.expression.IMetadataQueryParameter

Table 3–3 (Cont.) Search Considerations and Recommendations

Consideration	Recommendation
Wildcard * as repository name in search paths, when specified as a FTS metadata path criteria, will not work for UCM unless the wildcard is at the end of the path specification. For example: a `cm_path` like this: `/MyRepository/Foo/*`	Specify path parameter on Search object directly. Example: `search.setSearchPath("/*/MyFolders/mystuff")`
Empty values are not allowed in a search query.	Don't use a criteria such as `cm_nodeName != ''`.
`Notequals` operator is not supported for non-String properties	Be aware of the differences in search behavior and do not write search expressions that depend on unsupported criteria.
IMetadataQueryParameter buildEquals(String userPropertyField, String[] criteria) is not supported.	Be aware of the differences in search behavior and do not write search expressions that depend on unsupported criteria.
Multiple search paths on the same UCM repository are not supported.	Be aware of the differences in search behavior and do not write search expressions that depend on unsupported criteria.
When searching for folders, recursive search (folder tree search) is not supported, with the exception of searching the entire repository. This is due to UCM limitations. It is not possible, for example, to perform a search of the form, "cm_isHierarchy=true && cm_path like '/MyUCMRepo/foo/bar/*'".	To search the entire repository (all folders), leave out the cm_path criteria. To search the immediate children of a folder, use the cm_parent_uid criteria; for example "cm_isHierarchy=true && cm_parent_uid='xxx'" will find the folders immediately beneath the specified parent folder.
When searching for documents, recursive search (folder tree search) is supported if UCM is configured properly. Also see the Javadoc for com.bea.content.expression.Search regarding the use of search paths.; for example, `"cm_isContent=true && cm_path like '/MyUCMRepo/foo/bar/'"` Only one of cm_path and the Search object's searchPath can be set. If neither is set, then all documents in the repository will be searched (both filed and unfiled).	Configure the content server folders_g CollectionSearchRecursiveContent and related settings like CollectionMaxBranch in the UCM `config.cfg` file. These are described in UCM documentation. To perform a document search scoped to a folder tree, either set the search path, or add a `cm_path` criteria. Also be sure to add a criteria which limits the search to only documents. For example, you could use `"cm_isContent = true && cm_path like '/MyUCMRepo/foo/bar/*'"`
When searching for documents and using the LIKE operator, wildcards (*, ?) are only supported in the last path element.	Be aware of the differences in search behavior and do not write search expressions that depend upon unsupported criteria.
When searching documents (cm_isContent=true), it is not possible to limit the search to more than just a single content type; for example, this is not supported: "cm_isContent=true && (cm_objectClass == 'IDC:MyProfile' || cm_objectClass == 'IDC:AnotherProfile')" because it has multiple explicit content types. This also is not supported: "cm_objectClass = 'IDC:GlobalProfile' || cm_path like '/StellentRepository/*'" because part of the query is limited to an explicit content type and the other part of the query does not limit the content type.	When using cm_objectClass, if you need to limit the search to more than just a single content type, issue multiple queries to achieve the same behavior. If you want to search across all types, remove the cm_objectClass criteria.
For best search performance, scope the search to a single repository.	Either set the search path on the Search object with `search.setSearchPath("/MyUCMRepo")`, or scope the query to the repository by adding a `cm_path` clause like this to the query: `cm_path like '/MyUCMRepo/*'`

Table 3–3 (Cont.) Search Considerations and Recommendations

Consideration	Recommendation
UCM search doesn't support OR when `cm_objectClass` is specified in a query; other parameters can be ANDed with this criteria, but OR is not supported. For example, it is not supported to do this: `'cm_objectClass = 'IDC:GlobalProfile' || myField='bar'`	If this functionality is necessary, issue multiple queries to achieve the same behavior.
`cm_objectClass` criteria only supports `==`, `!=`, and `like`. Use of `!=` is restricted to the case of excluding folders (which behaves the same as adding `cm_isHierarchy=true`).	Be aware of the valid operators when using `cm_objectClass` criteria.
There are several restrictions on `cm_path` operators. The IN operator cannot be used in conjunction with folder search, or with document recursive search, though it can be used with document direct search.	Use a document direct search path instead; for example, `cm_isContent=true && cp_path IN ('/MyUCMRepo/foo', '/MyUCMRepo/bar')`
Use of OR operator in conjunction with document recursive search requires recursive search path to specified on Search object.	When searching documents, in order to use the OR operator in conjunction with document recursive search, specify the search path directly on the Search object. Also be sure to scope the search to only documents (by adding `cm_isContent=true`)
Queries may be case-sensitive depending on the selected UCM search engine.	Be aware that some UCM search engines are case-sensitive. OracleTextSearch is case-insensitive. `Database.FullText` metadata searches are case-sensitive.

3.1.3 Unsupported Features for the UCM VCR Adapter

Table 3–4 lists the content management features that will not be supported by the UCM VCR Adapter.

Table 3–4 Unsupported Features in UCM VCR Adapter

Feature	Consideration	Recommendation
Versioning	The adapter does not support versioning. The latest released document version is available via the SPI.	Do not rely on versioning or access to anything other than the latest released item when using the adapter.
Type Management	The adapter does not include type write capabilities.	Model content types using the UCM User Interface.
Search	You cannot specify multiple object class criteria in a search expression, such as an expression like: `(cm_objectClass == foo && prop1 == value) || (cm_objectClass == bar && prop2 == value2)` The adapter cannot execute this type of search.	It is possible, however, to return multiple nodes representing different content types in a single result set when searching in other ways, such as a prop=value expression.

3.1.4 Unsupported UCM Features

Table 3–5 lists UCM features that are not supported by the adapter. These features will not be exposed in the VCR.

Table 3–5 Considerations for Exposing UCM Content

Feature	Considerations
Renditions	Only the Web-Layout rendition is supported the UCM VCR Adapter. No other rendition types are supported.
Folios	UCM Folio meta-document will not be exposed by the VCR.
HCSF Files	HCSF files (Hypertext Content Server Forms) will not exposed by the VCR.
Dynamic Converter	Dynamic converter will not be exposed by the VCR.
Shortcuts	Shortcuts are not exposed through the UCM VCR Adapter.

3.1.5 Unsupported Web Content Management (WCM) Features

Table 3–6 lists WCM (Site Studio) features that are not supported by the adapter. These features will not be exposed in the VCR.

Table 3–6 Considerations for Exposing WCM Content

Feature	Consideration	Recommendation
Embedded Images in WSYWIG Element Definitions	Embedded images are not supported in a WSYWIG Element Definition that is part of a Region based Content Type. Any content that is created of this type will not be rendered by the UCM VCR adapter. The adapter will return a partial string URI to the image.	Constrain the Use of a WSYWIG Element definition to HTML text only and use a separate element definition with Link(s) to reference images.
Element Definitions of type Image, Dynamic Lists are not supported.	The UCM VCR Adapter upon reading a region-based content type definition that contains one of these element definitions will not know how to interpret the WCM content. What the adapter may return in some cases is a string interpretation of the content.	Do not use these types of Element Definitions as part of a UCM VCR region-based content type definition.
Element Definitions of type Custom have limited support. Custom element values will be represented as strings through the UCM VCR Adapter. The custom form called `SS_DOCNAME_LINK_FORM` is supported by the UCM VCR Adapter as an add-on to support Element definitions that contain link element types.	The UCM VCR adapter upon reading a region-based content type definition that contains custom element definitions will only know how to represent the WCM content as a string value.	Only use custom element forms for string and link values.

3.1.6 Property Field Mappings

This section explains how the VCR system property fields map to UCM content fields when searches are performed. During a search operation that is performed using the WLP search API, the standard VCR system properties listed in Table 3–7 are automatically converted to corresponding UCM content fields by the UCM/VCR Adapter. For example, searches performed through Content Presenter or content selectors perform these conversions automatically.

Note: If you are performing a search, the best practice is to use the system properties as they appear in the WebLogic Portal user interface, such as in the user interface for Content Presenter or content selectors. For example, if you need the binary size, use VaultFileSize (the internal name) rather than cm_binarySize. This practice ensures that even if the system field mapping changes over time, the search behavior remains the same.

Table 3–7 shows how these mappings are made internally by the UCM/VCR Adapter. Note that UCM has two names for every field. One is an internal name, which you can see in UCM when adding a field to a rule, and the other is the field name that appears in the user interface. In Table 3–7, the internal name (for example, dDocName) is listed first, and the UI name (for example, Content ID) is listed below it in parentheses, where applicable.

Caution: The mappings in Table 3–7 are UCM/VCR Adapter-specific and are subject to change.

Table 3–7 Property Field Mappings

WLP VCR System Property	UCM Content Field (When the Object is Document)	UCM Property (When the Object is Folder)
cm_nodeName	dOriginalName (document native filename)	dCollectionName (Virtual Folder Name)
cm_uid	dDocName (Content ID)	IDC:Folder/{dCollectionID}
cm_path	No direct translation. Handled internally based on Folder and document name.	No direct translation. Handled internally.
cm_createdBy	dDocAuthor (Author)	dCollectionCreator (Creator)
cm_createdDate	dCreateDate	dCreateDate (Created)
cm_modifiedBy	dDocAuthor (Author)	dCollectionModifier
cm_modifiedDate	dCreateDate	dLastModifiedDate
cm_objectClass	No direct translation. Handled internally based on the profile (if any) and/or SiteStudio Region Definition (if any).	No direct translation. Folders are mapped to IDC:Folder ObjectClass
cm_contentType	dFormat (from document native file)	N/A
cm_binaryName	dOriginalName (document native filename)	N/A

Table 3–7 (Cont.) Property Field Mappings

WLP VCR System Property	UCM Content Field (When the Object is Document)	UCM Property (When the Object is Folder)
cm_binarySize	VaultFileSize (document native file size)	N/A
cm_isHierarchy	True if the object is a folder; false if object is a document.	N/A
cm_isContent	True if the object is a document; false if object is a folder.	N/A

3.2 Performing Create, Update, Delete, and Move Operations

This section discusses creating, updating, deleting, and moving content and content nodes. Typically, you create and edit content in the Content Management part of the WebLogic Portal Administration Console.

This section includes these topics:

- Section 3.2.1, "Considerations for Creating and Editing Content"
- Section 3.2.2, "Considerations for Creating and Editing Folders"
- Section 3.2.3, "Configuring the Trash Folder"

3.2.1 Considerations for Creating and Editing Content

For creating and editing content, profile-based content, as well as IDC:GlobalProfile and IDC:Folder, is supported. Editing or creating of Oracle Site Studio based content (region definition types) is not supported.

If a content property is not editable in Content Server, it is also not editable through the UCM SPI Adapter. For example, the `dDocName` property is not editable in either Content Server or the WLP Administration Console once a document has been created.

Note that when you create a document, it is not available through the UCM SPI Adapter until it is in released state. The UCM SPI Adapter only exposes documents in released state. After the document is in release state, it is only exposed after it is indexed. When you create a folder, it is available immediately.

Note: The WLP Administration Console cannot always detect when a UCM property is not editable; therefore, in the Administration Console, those properties may appear to be editable even though they are not. An attempt to edit the property will result in an error.

When creating or updating a document node, note that the ObjectClass type specified by the client (the Content Management part of the WebLogic Portal Administration Console) is not used by Content Server. The Content Server's profile configuration determines which ObjectClass is returned for a given document.

When creating or updating a document node with a binary value, note the following rules:

- The node name must match the filename of the binary file. For example, if the binary filename is `future.doc`, your node must be named `future.doc`.

- The server determines the ContentType of a binary value based on the binary filename. You can configure this filename/type mapping with the Content Server Configuration Manager applet using the File Formats menu option.

3.2.2 Considerations for Creating and Editing Folders

This section describes considerations when working with folders.

> **Note:** Creating nodes at the root level is not supported. As a workaround, see the UCM documentation for information on creating system folders.

To create folders, you generally must configure security properly so that you are connecting as a UCM user with the appropriate privileges. If you do not have these privileges, you might see an error message of this form:

```
com.bea.content.RepositoryException:
oracle.stellent.ridc.protocol.ServiceException: Unable to create virtual folder.
Unable to create virtual folder
```

For more information on specifying the UCM user, see Section 2.4, "Security Mapping."

Note that when you create a folder, it is available immediately. When you create a document, it is not available through the UCM SPI Adapter until it is in released state. The UCM SPI Adapter only exposes documents in released state and only after the document is indexed.

3.2.3 Configuring the Trash Folder

The UCM `folders_g` component can be configured to expose "trash" folder. If the trash folder is disabled on the UCM server, then an object you delete from the repository is removed immediately. If the trash folder is enabled on the UCM server, the adapter considers the ForceDeleteFolders `<repository-config>` setting in `content-config.xml`. If this setting is true, a deleted object is deleted immediately from the UCM server. If false (the default), the deleted object is moved to the trash folder.

3.3 Creating Required Fields

If you create a content type (a UCM profile), note that creating required fields for one profile makes those fields required for all profiles. For example, if you create a profile called BookType and create a required field called ISBN, the ISBN field is now required for all other profiles. One option is to make the field required, and define a default value in the rule for profiles which do not need this field. Another option is to make the field optional (not required), and for each profile, make the field required or optional through the profile rule configuration.

3.4 Working with Unfiled Documents

UCM includes the concept of an unfiled document. An unfiled document is a document that does not reside in a hierarchical document structure. Typically, the way to locate unfiled documents is to search for them. In a WebLogic Portal UCM Repository, unfiled documents are exposed as children beneath the root path of the repository: /*repositoryname*. In the WLP client, you can identify an unfiled document by looking at its `xCollectionID` property. You can also search using this property to find only unfiled documents. If this property is 0, the document is unfiled. If it is a nonzero number, the document is filed. The UCM SPI supports reading and updating unfiled documents. The UCM SPI does not support creating unfiled documents.

3.5 Node Property Loading

For performance reasons, certain node properties exposed through the UCM Adapter are not loaded until they are requested. This section lists the node properties that are fully loaded and properties that are not. Access to fully loaded properties will be faster, so they are a good choice for content templates and other tabular structures.

As a template developer, you can optimize performance of your template if you are aware when different property values are loaded. For example, a typical list template will render faster if you refer only to properties that are immediately loaded when the node is first retrieved and avoid properties that are loaded later when needed.

A secondary consideration is dependent on how the node is retrieved: through search versus fetched by ID or as children of a parent. A property that may be loaded immediately on node retrieval for searches (such as "Results of a Query") may be loaded later for other retrieval methods (such as "Contents Under a Folder"). Table 3–8 shows whether or not node properties are loaded immediately upon retrieval, by retrieval mechanism.

Note: Any custom properties added by a profile rule are always immediately loaded on node retrieval.

Table 3–8 Loading of Node Properties by Node Retrieval Mechanism

OCS Global Profile Properties	GET BY PARENT ID ("Contents Under a Folder")	SEARCH ("Results of a Query")	GET BY UUID ("Single Content Item" and "List of Items")
VaultFileSize	N	Y	Y
dCheckoutUser	Y	N	Y
dCreateDate	Y	Y	Y
dDocAccount	Y	Y	Y
dDocAuthor	Y	Y	Y
dDocName	Y	Y	Y
dDocTitle	Y	Y	Y
dDocType	Y	Y	Y
dFormat	Y	Y	Y
dID	Y	Y	Y
dInDate	Y	Y	Y
dIsCheckedOut	Y	N	Y
dOutDate	Y	Y	Y
dReleaseDate	Y	N	Y
dReleaseState	Y	N	Y
dRevClassID	Y	N	Y
dRevLabel	Y	Y	Y

Table 3–8 (Cont.) Loading of Node Properties by Node Retrieval Mechanism

OCS Global Profile Properties	GET BY PARENT ID ("Contents Under a Folder")	SEARCH ("Results of a Query")	GET BY UUID ("Single Content Item" and "List of Items")
dRevRank	Y	N	Y
dRevisionID	Y	Y	Y
dSecurityGroup	Y	Y	Y
dStatus	Y	N	Y
dWebExtension	Y	Y	Y
dWorkflowState	N	N	Y
idcPrimaryFile	Y	Y	Y
idcRenditions	N	N	Y
xCollectionID	Y	Y	Y
xComments	Y	Y	Y
xForceFolderSecurity	Y	Y	Y
xHidden	Y	Y	Y
xInihibitUpdate	Y	Y	Y
xReadOnly	Y	Y	Y

3.6 Performing Searches

When performing searches, note the following:

- Restrict the search path to one repository at a time. This best practice boosts search performance, especially with the UCM Adapter. To do this, either add a clause like this to your query: `"&& cm_path like '/UCMRepo/*"`, or set the search path directly on the Search object, by calling Search.setSearchPath("/MyUCMRepo");
- If you conduct a search which finds more than 200 documents or folders, it is possible under some circumstances to receive duplicate or missing items, if content matching the query is being added or removed while the query is executing.
- Only content in the released state is returned through the UCM Adapter. When documents are created, they may not be queryable until they have been indexed.

3.7 Scoping Content Selectors and Other Queries to a Repository

In order to scope expression-based searches to a repository (for performance as well as query validity reasons), you can restrict the search path to the appropriate repository. Use the following steps to do this:

1. Construct your search with the OR operator as usual in WLP.
2. Edit the Content Selector in the Advanced tab of the Administration Console as follows:
   1. Group the OR criteria with parentheses ( ).
   2. AND that criteria with cm_path to limit the scope of the search. For example:

```
((toProperty('WLP Repository/image', 'description') == 'image') ||
dDocAuthor = 'sysadmin) && cm_path like '/WLP_Repository/*'
```

3.8 UCM Content Modeling Examples

This section presents several basic examples that demonstrate techniques for modeling content in UCM. These techniques are known produce content models that work well with the UCM VCR adapter.

The examples include:

- Section 3.8.1, "Creating Profile-Based Content Types with the UCM Administration Console"
- Section 3.8.2, "Creating Region-Based Content Types with Oracle Site Studio"
- Section 3.8.3, "Creating a New Contributor Data File Based On a Region Definition"

> **Tip:** If you want to try running the examples, be sure the UCM VCR Adapter is installed, as explained in Section 2.1, "Installing the UCM VCR Adapter." Both the Oracle WebLogic Portal server and Oracle Content Server must be running.

3.8.1 Creating Profile-Based Content Types with the UCM Administration Console

This example illustrates how to create a content type and associate a profile with it. The procedure involves creating a content type, adding metadata fields to the type, creating a rule, and creating a profile. The content type created using this technique will be surfaced as a VCR Content Type in WebLogic Portal. See also Section 3.1.2, "Best Practices for Modeling Content."

> **Note:** If you do iterative development where you have already started your WLP server and then you change or add to your UCM content definitions, you will either need to restart your WLP server or flush the P13N caches for the definition changes to take effect. For information on flushing the caches, see Section 2.1.3, "Modify Cache Settings."

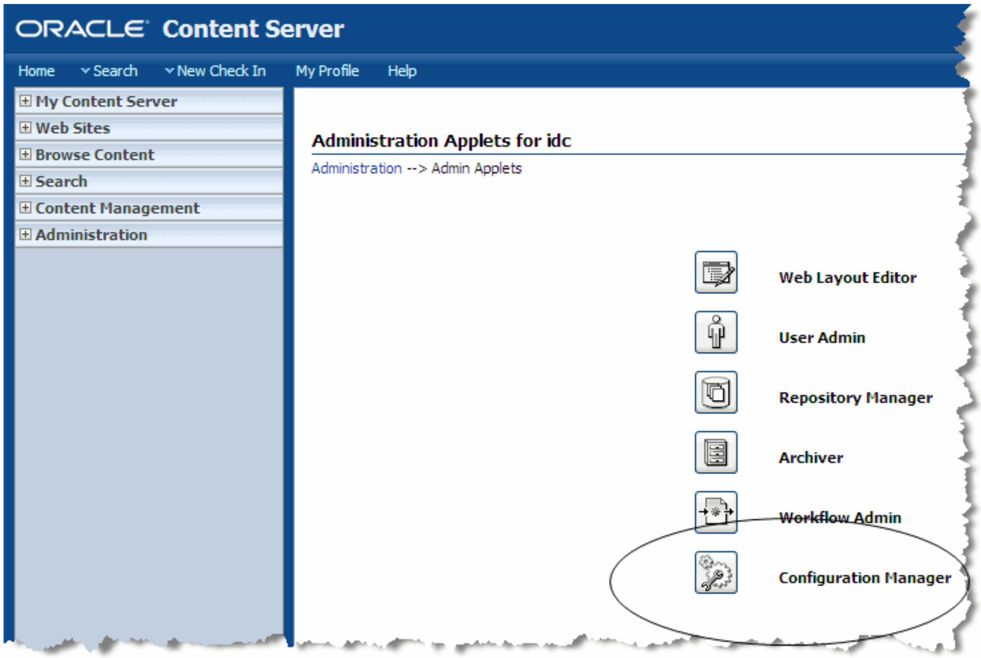
Tasks described in this section include:

- Section 3.8.1.1, "Creating a New Content Type"
- Section 3.8.1.2, "Creating Information Fields"
- Section 3.8.1.3, "Creating a Rule"
- Section 3.8.1.4, "Creating a Profile"

3.8.1.1 Creating a New Content Type

1. Start Oracle Content Server and log in to the user interface.
2. From the Administration tray, select **Admin Applets**.
3. In the Administration page, select **Configuration Manager**, as shown in Figure 3–2. The Configuration Manager window appears.

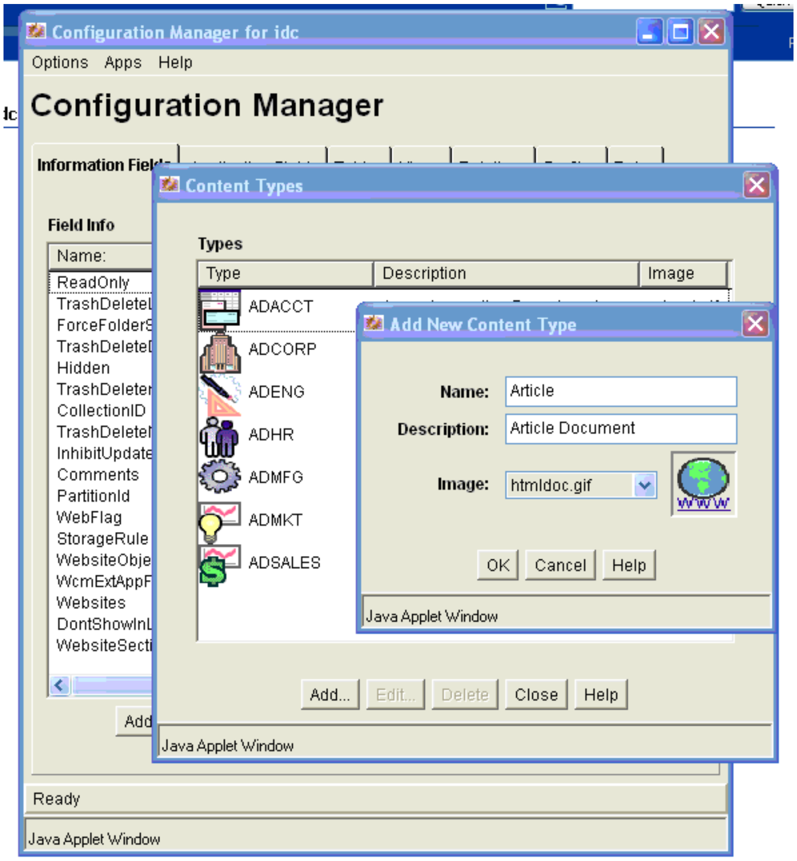
Figure 3–2 Administration Page



4. In the Configuration Manager dialog, Select **Options > Content Types.**

5. In the Content Types dialog, select **Add**. The Add New Content Type dialog appears, as shown in Figure 3–3.

Figure 3–3 Add New Content Type Dialog



Name: Article

Description: Article Document

Image: htmldoc.gif

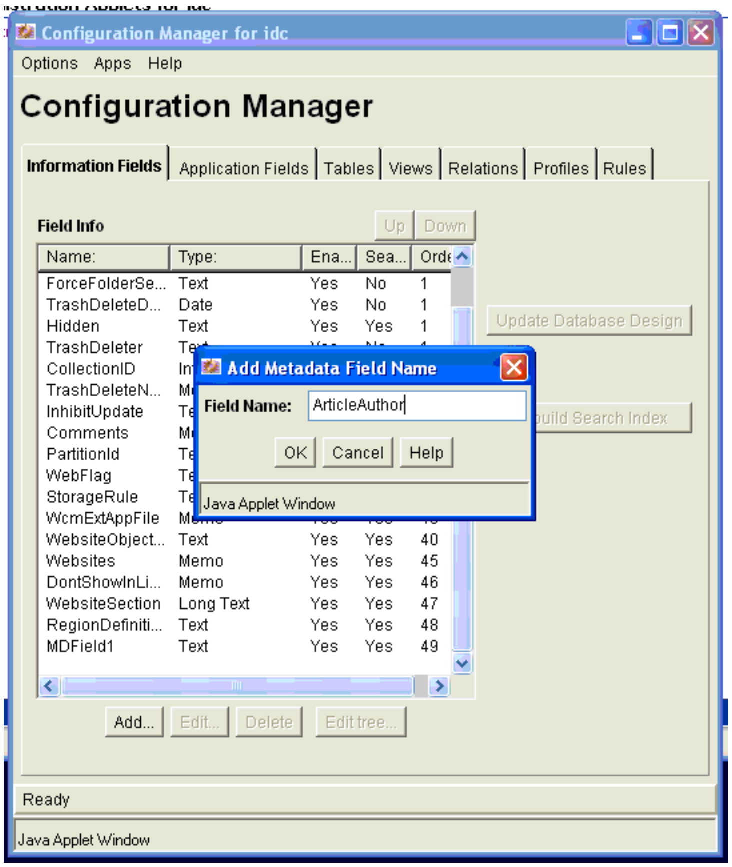
6. Enter a Name and Description for the new content type, and click **OK**. The new type is added to the list of types in the Content Types dialog.

3.8.1.2 Creating Information Fields

This section explains how to create the Information Field(s) (metadata fields) that will be associated with your new VCR Content Type (as content properties).

1. Open the Configuration Manager dialog as explained in Section 3.8.1.1, "Creating a New Content Type."
2. In the Configuration Manager dialog, be sure the Information Fields tab is selected.
3. Click **Add**.
4. In the Add Metadata Field Name dialog (Figure 3–4), enter a Field Name and click **OK**.

Figure 3–4 Add Metadata Field Name Dialog



Name:	Type:	Ena...	Sea...	Orde
ForceFolderSe...	Text	Yes	No	1
TrashDeleteD...	Date	Yes	No	1
Hidden	Text	Yes	Yes	1
TrashDeleter	Te			
CollectionID	In			
TrashDeleteN...	M			
InhibitUpdate	Te			
Comments	M			
PartitionId	Te			
WebFlag	Te			
StorageRule	Te			
WcmExtAppFile	M			
WebsiteObject...	Text	Yes	Yes	40
Websites	Memo	Yes	Yes	45
DontShowInLi...	Memo	Yes	Yes	46
WebsiteSection	Long Text	Yes	Yes	47
RegionDefiniti...	Text	Yes	Yes	48
MDField1	Text	Yes	Yes	49

Field Name: ArticleAuthor

Ready

5. In the next Add Metadata Field dialog, specify the desired field type. Click **OK**. The new field is added to the Field Info list.

6. Click **Update Database Design**.

3.8.1.3 Creating a Rule

This section explains how to create a Rule that will be used with the Profile that defines the Content Type.

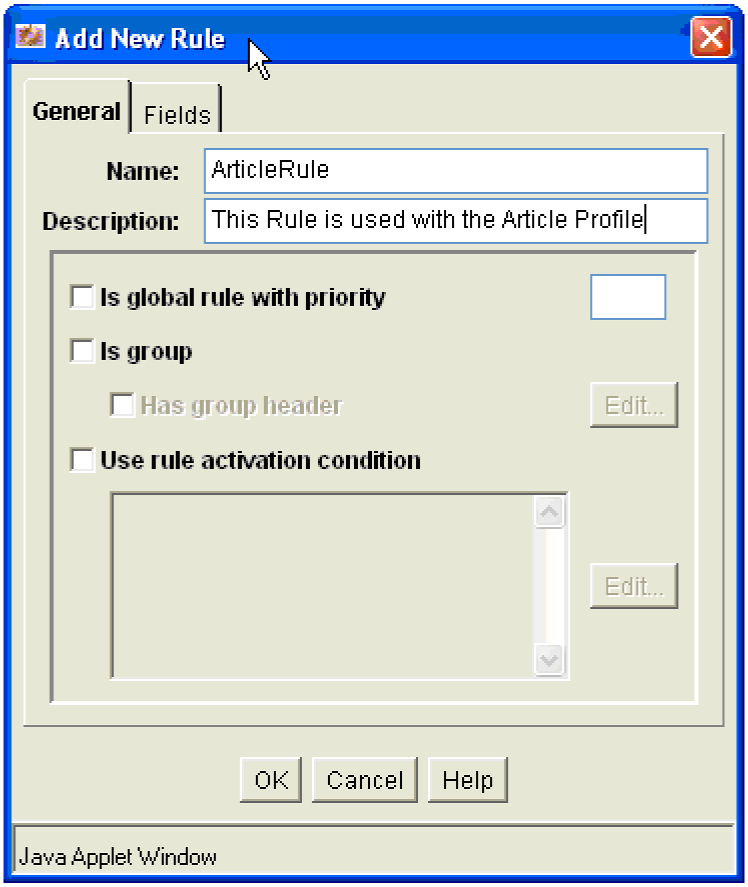
1. Open the Configuration Manager dialog as explained in Section 3.8.1.1, "Creating a New Content Type."

2. In the Configuration Manager dialog, select the Rules tab.

3. Click **Add**.

4. In the Add New Rule dialog, enter the name of the rule, as shown in Figure 3–5.

Figure 3–5 Add New Rule Dialog



Name: ArticleRule

Description: This Rule is used with the Article Profile

□ Is global rule with priority

□ Is group

□ Has group header

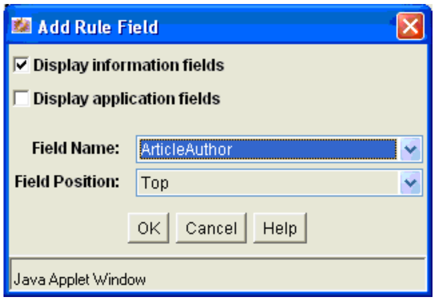
□ Use rule activation condition

5. Select the Fields tab.

6. In the Fields tab, click **Add**.

7. In the Add Rule Field dialog (Figure 3–6), select an Information Field from the Field Name dropdown list. For example, select the Information Field you created in Section 3.8.1.2, "Creating Information Fields." This field will be associated with the Content type that you are creating.

Figure 3–6 Add Rule Field Dialog



☑ Display information fields

□ Display application fields

Field Name: ArticleAuthor

Field Position: Top

8. Click **OK**.

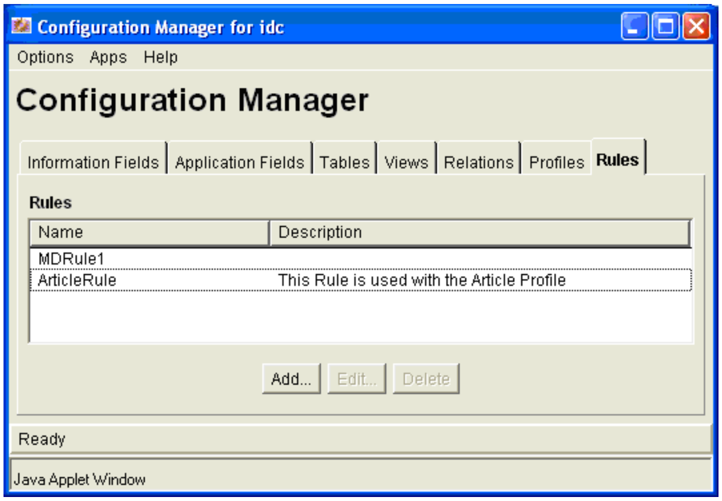
9. In the Add Rule Field dialog, do not change the defaults. Click **OK**.

10. Repeat steps 5 – 9 to add any additional property fields you want associated with this content type. Any required fields should generally be listed in your rule, in

order to support document creation. When your done all of these property fields will appear in the dialog.

11. Click **OK** in the Add New Rule dialog. The new rule appears in the Rules list under the Rules tab of the Configuration Manager dialog, as shown in Figure 3–7.

Figure 3–7 New Rule in the Configuration Manager



Rules

Name	Description
MDRule1	
ArticleRule	This Rule is used with the Article Profile

3.8.1.4 Creating a Profile

This section explains how to create the profile that will define the Content Type you created in Section 3.8.1.1, "Creating a New Content Type."

1. Open the Configuration Manager dialog as explained in Section 3.8.1.1, "Creating a New Content Type."

2. In the Configuration Manager dialog, select the Profiles tab.

3. Click **Select**.

4. In the Edit Trigger Field dialog, select **Type** from the dropdown menu and click **OK**. Note that in the Configuration Manager, the Trigger Field is now set to `dDocType`. This specifies how documents are associated with UCM profiles, in this case through the Type (`dDocType`) field value.

> **Note:** The use of `dDocType` is not required. You can use any other option list metadata field. For details on creating content profiles and using triggers, see “Content Profile Triggers” in *Managing Repository Content*.

5. In the Profiles tab, select **Add**. The Add Profile dialog appears.

6. In the Add Profile dialog, enter a name and click **OK**.

7. In the next Add Profile dialog, enter a label and a description. From the Trigger dropdown menu, Select the Content Type that you created in Section 3.8.1.1, "Creating a New Content Type." See Figure 3–8.

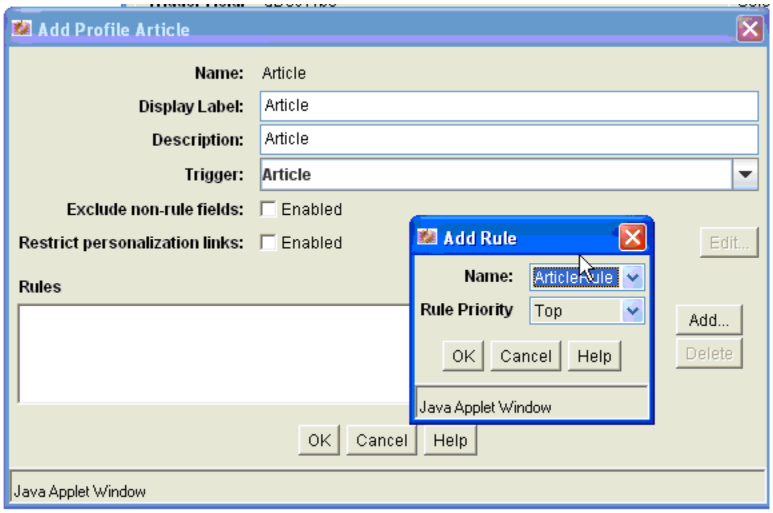
8. Ensure each profile has a unique trigger value. Sharing the same trigger value across multiple profiles is not supported.

9. Check **Exclude non-rule fields**.

10. In the Rules part of the dialog, click **Add**.

11. In the Add Rule dialog, select the Rule you created in Section 3.8.1.3, "Creating a Rule." See Figure 3–8.

Figure 3–8 Add Profile Dialog



Add Profile Article

Name: Article

Display Label: Article

Description: Article

Trigger: Article

Exclude non-rule fields: □ Enabled

Restrict personalization links: □ Enabled

Rules

Add Rule

Name: ArticleRule

Rule Priority: Top

12. Click **OK** in the Add Rule dialog, and click **OK** in the Add Profile dialog.

13. Close the Configuration Manager dialog.

14. Use the UCM Content Management Contribution functions to add content using this new content type.

15. Log on to the WebLogic Portal Administration Console to see this Content Type and any content you create that is associated with this type. Depending on how your UCM Adapter Type caches are configured, you need to flush the caches or restart the server before the updated type is visible in WLP.

3.8.2 Creating Region-Based Content Types with Oracle Site Studio

This example illustrates how to create a Site Studio region definition (which will be mapped to a WLP content type) as well as a contributor data file based on the region definition. In the WLP VCR, the data file will surface as a content node with associated metadata fields. The metadata fields are those associated with the UCM region definition, plus additional metadata that comes from the UCM metadata fields pulled in by a profile. See also Section 3.1.2, "Best Practices for Modeling Content."

> **Note:** If you do iterative development where you have already started your WLP server and then you change or add to your UCM content definitions, you will either need to restart your WLP server or flush the P13N caches for the definition changes to take effect. For information on flushing the caches, see Section 2.1.3, "Modify Cache Settings."

3.8.2.1 Prerequisites

This section explains how to use Oracle Site Studio and the UCM Administration console to create content using region definitions. The goal of this procedure is to create region-based content in UCM and have that content and its properties appear in the WebLogic Portal VCR. The Site Studio region elements will be expressed as property fields in the VCR.

For background information on Site Studio, see Section 3.1.2, "Best Practices for Modeling Content."

Note: The example shown here will use the Article profile definition used in the previous example, Section 3.8.1, "Creating Profile-Based Content Types with the UCM Administration Console."

3.8.2.2 Overview of Steps

The basic steps described in this section are:

1. Create your element definitions. This example demonstrates how to create one element definition for each of the four supported types:

   - WYSIWYG Element
   - Plain Text Element
   - Static List Element
   - Link Element Type (via a custom element using the supplied custom element form)

   Note: Image and Dynamic lists are not supported by the UCM VCR Adapter.

   ---

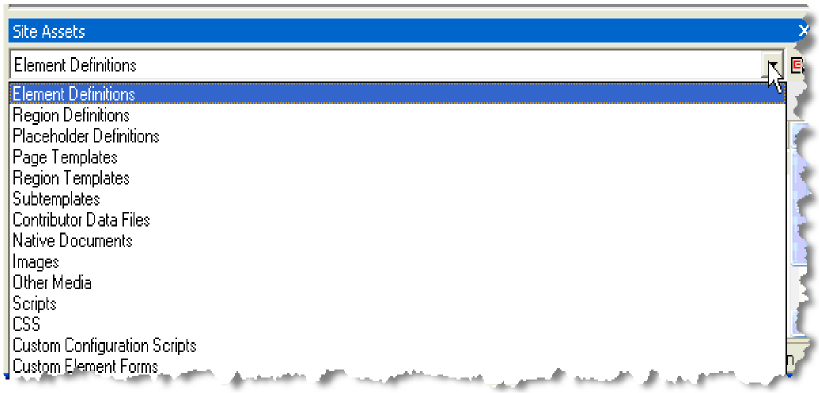
2. Create a Region definition.
3. Add the element definitions to the Region definition.
4. Create a Contributor Data file based on the Region definition.
5. Edit the Contributor Data file.
6. View the Content in the WLP VCR.

3.8.2.3 Creating the Element Definitions

This section explains how to create four element definitions: WYSIWYG, plain text, static list, and link.

1. Start Oracle Site Studio.
2. In the Site Assets dropdown menu in the Site Studio window, select **Element Definitions**, as shown in Figure 3–9.

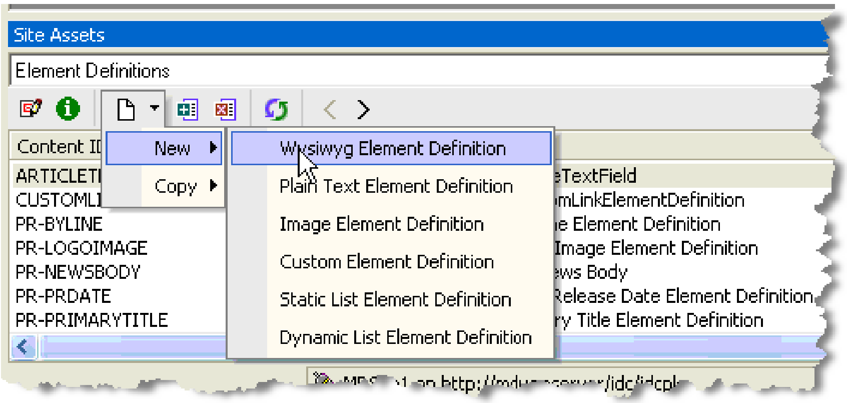
Figure 3–9 Site Assets Menu



3.8.2.3.1 Create the WYSIWYG Element Follow these steps to create a WYSIWYG element:

1. Select the **Page** icon and select **New** > **WYSIWYG Element Definition**, as shown in Figure 3–10. The Content Server Dialog appears.

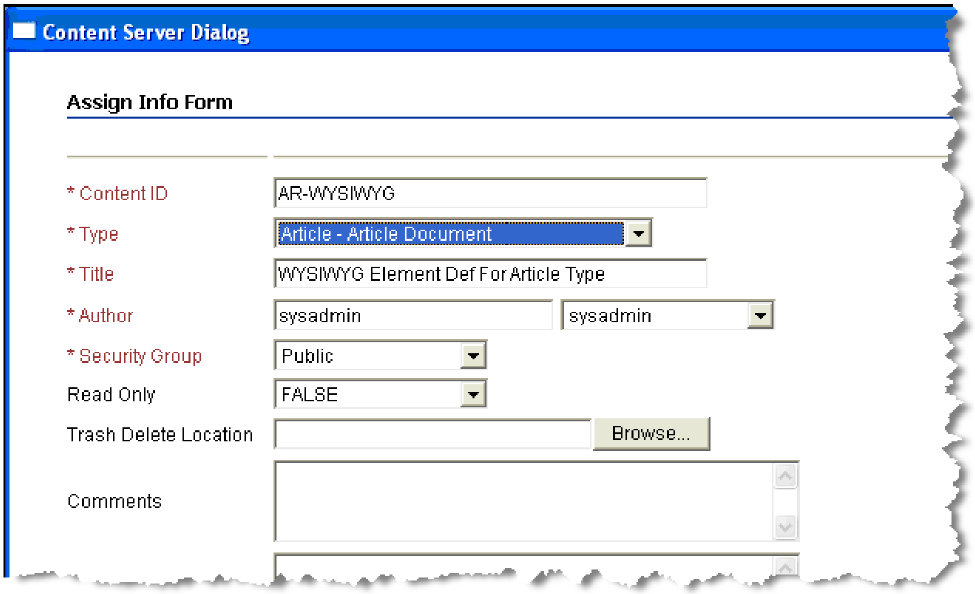
Figure 3–10 Selecting Wysiwyg Element Definition



2. In the Content Server dialog enter a Content ID and a Title, as shown in Figure 3–11.

   Optionally, associate this Element Definition to the Content Type that it will be used with. In this example, the Article content type is used.

Figure 3–11 Assign Info Form for WYSIWYG Content Element



* Content ID	AR-WYSIWYG
* Type	Article - Article Document
* Title	WYSIWYG Element Def For Article Type
* Author	sysadmin sysadmin
* Security Group	Public
Read Only	FALSE
Trash Delete Location	
Comments	

3. Click **Assign Info** at the bottom of the dialog.

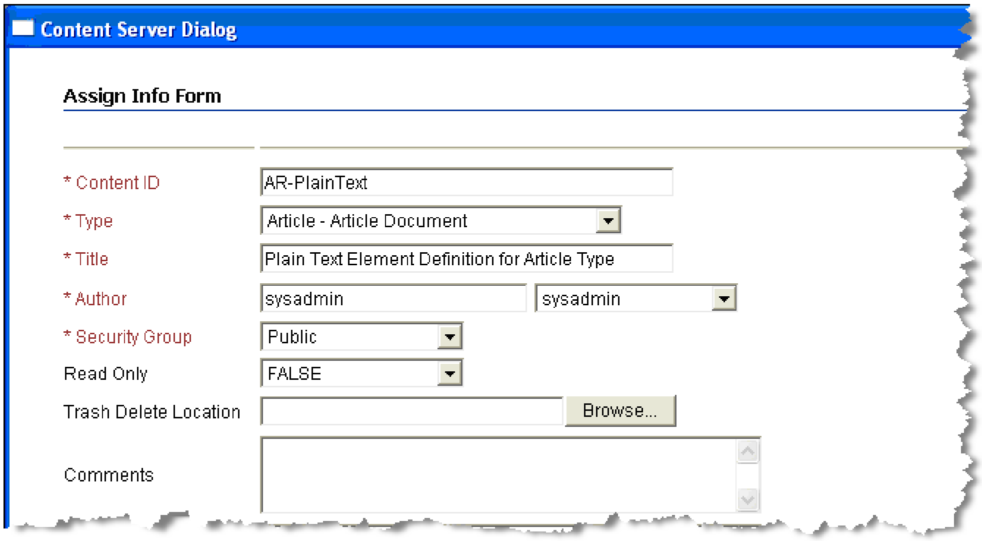
3.8.2.3.2 Create the Plain Text Element

Follow these steps to create a plain text element

1. Select the **Page** icon and select **New** > **Plain Text Element Definition**.
2. In the Content Server Dialog enter a Content ID and a Title, as shown in Figure 3–12.

   Optionally, associate this Element Definition to the Content Type that it will be used with. In this example, the Article content type is used.

Figure 3–12 Assign Info Form for Plain Text Content Element



* Content ID	AR-PlainText
* Type	Article - Article Document
* Title	Plain Text Element Definition for Article Type
* Author	sysadmin sysadmin
* Security Group	Public
Read Only	FALSE
Trash Delete Location	
Comments	

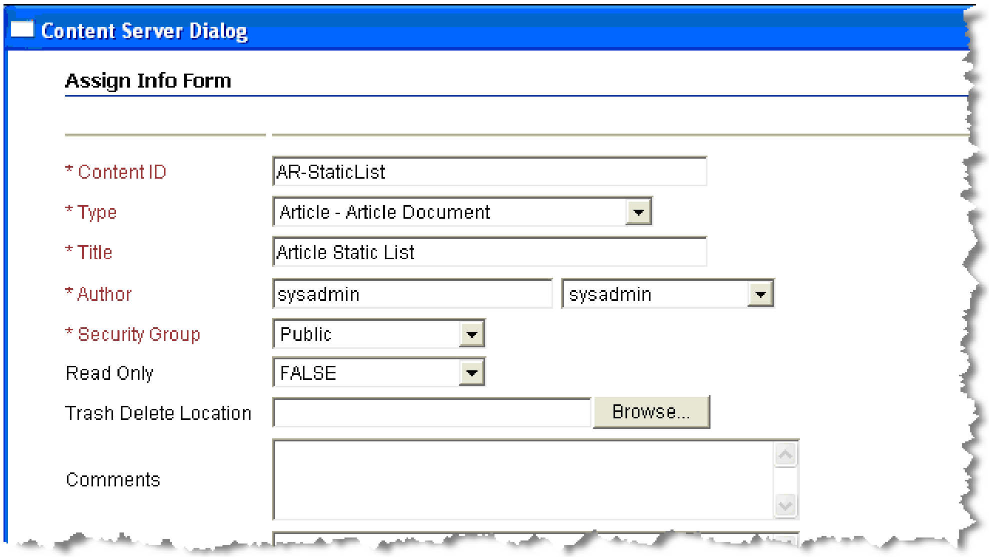
3. Click **Assign Info** at the bottom of the dialog.

3.8.2.3.3 Create the Static List Element

1. Click the **Page** icon and select **New** > **Static List Element Definition**.
2. In the Content Server dialog enter a Content ID and a Title, as shown in Figure 3–13

   Optionally, associate this Element Definition to the Content Type that it will be used with. In this example, the Article content type is used.

Figure 3–13 Assign Info Form for Static List Content Element



* Content ID: AR-StaticList

* Type: Article - Article Document

* Title: Article Static List

* Author: sysadmin | sysadmin

* Security Group: Public

Read Only: FALSE

Trash Delete Location:

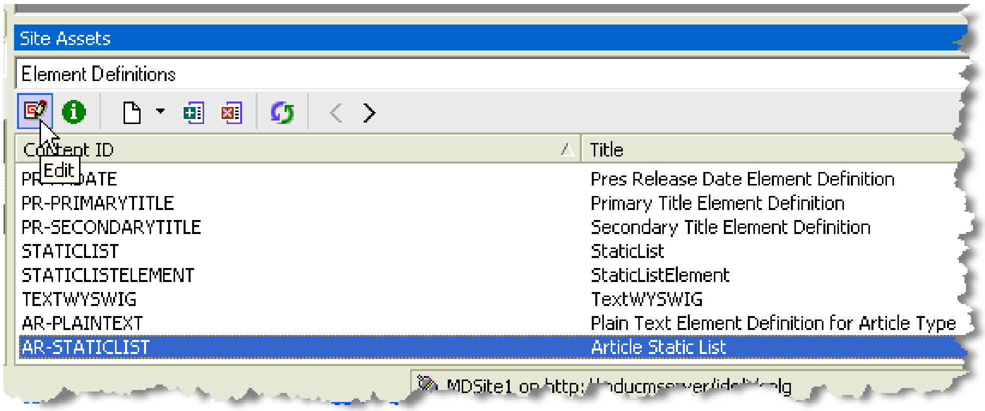
Comments:

3. Click **Assign Info** at the bottom of the dialog.

> **Note:** In the following steps, you further define the sub-elements that will comprise the static list. Keep in mind that when adding elements to the static list you can only add elements that the adapter supports (WYSIWYG, Plain Text, and the Link (custom) element type). Embedding a static list within a static list is not supported. In the example in this section, a static list consisting of two plain text fields is created.

4. Select the static list element definition you just created from the list of elements and select the **Edit** icon, as shown in Figure 3–14. The Elements dialog appears.

Figure 3–14 Selecting the Static List and Clicking the Edit Icon



Site Assets

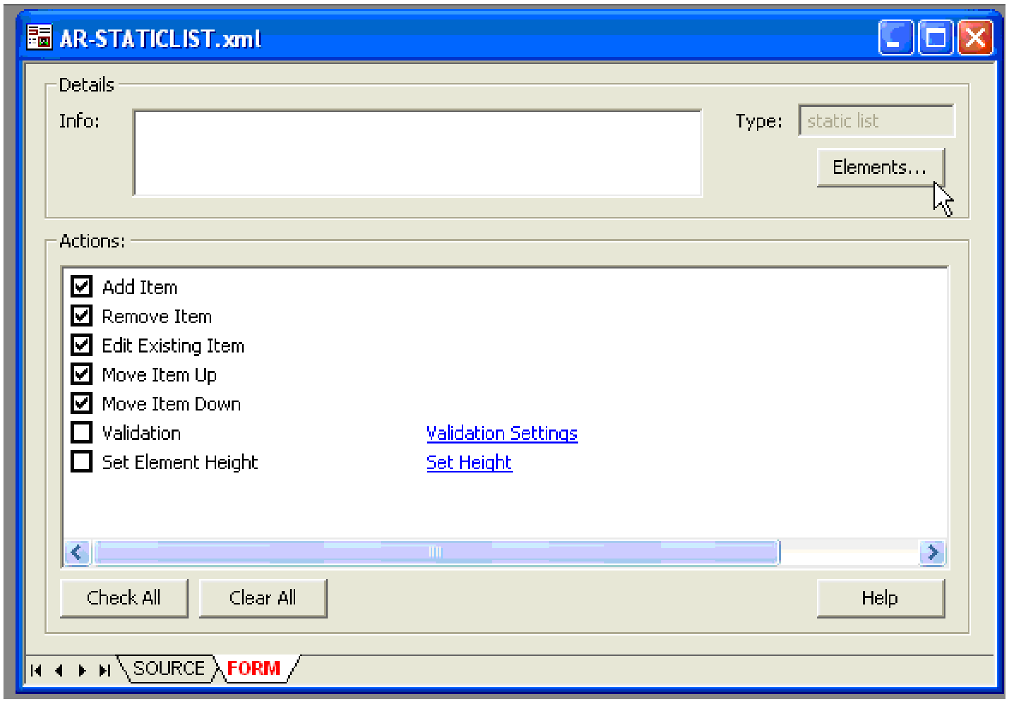
Element Definitions

Content ID	Title
PR DATE	Pres Release Date Element Definition
PR-PRIMARYTITLE	Primary Title Element Definition
PR-SECONDARYTITLE	Secondary Title Element Definition
STATICLIST	StaticList
STATICLISTELEMENT	StaticListElement
TEXTWYSWIG	TextWYSWIG
AR-PLAINTEXT	Plain Text Element Definition for Article Type
AR-STATICLIST	Article Static List

MDSite1 on http://ducmserver/idc lg

5. In the Element Edit dialog, select **Elements**, as shown in Figure 3–15. The Elements dialog appears.

Figure 3–15 Selecting the Elements Button in the Dialog



AR-STATICLIST.xml

Details

Info:

Type: static list

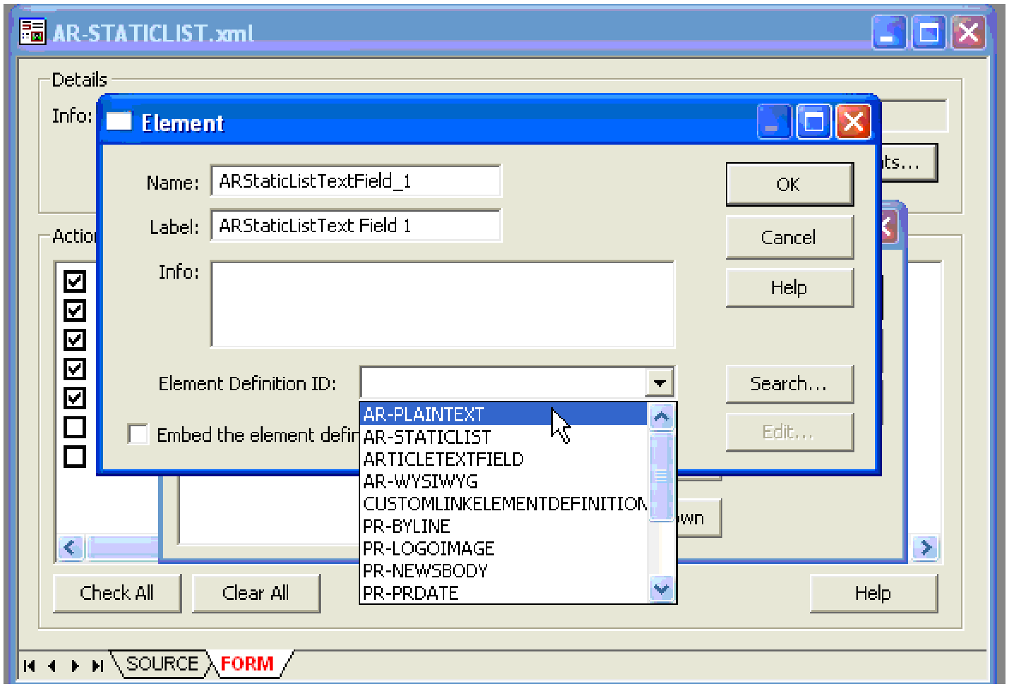
Actions:

☑ Add Item
☑ Remove Item
☑ Edit Existing Item
☑ Move Item Up
☑ Move Item Down
☐ Validation — Validation Settings
☐ Set Element Height — Set Height

6. In the Elements dialog, select **Add**. The Element dialog appears.

7. In the Element dialog, enter the Name and Label for this entry and select the Element Definition to be used for this entry in the list, as shown in Figure 3–16. In this example, the previously created Plain Text element is used. Click **OK**.

Figure 3–16 Completing the Element Dialog



Name: ARStaticListTextField_1

Label: ARStaticListText Field 1

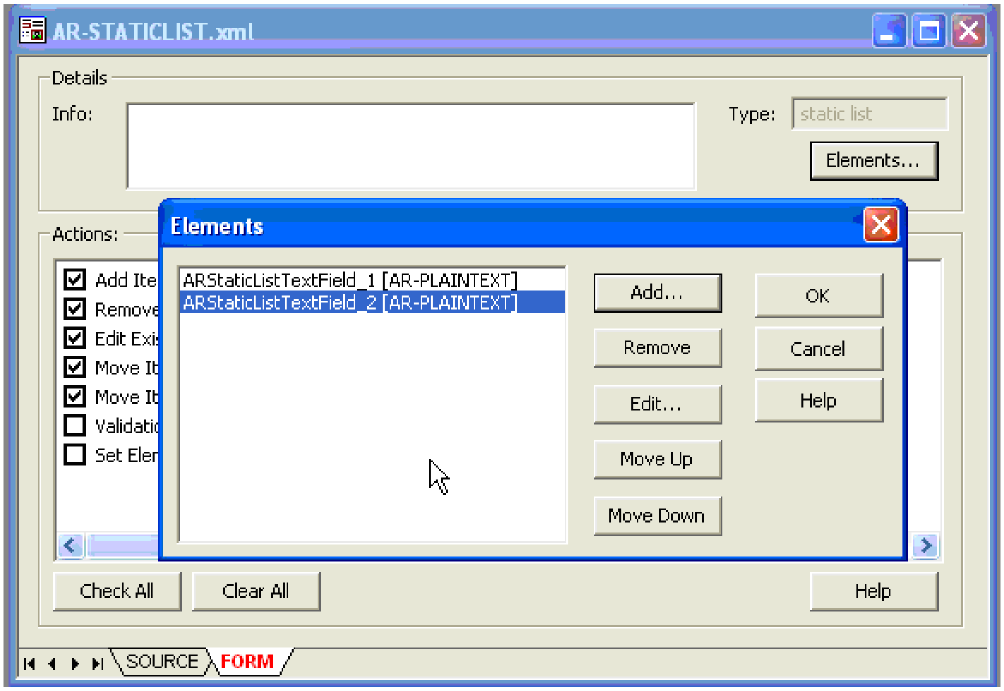
Info:

Element Definition ID:

AR-PLAINTEXT
AR-STATICLIST
ARTICLETEXTFIELD
AR-WYSIWYG
CUSTOMLINKELEMENTDEFINITION
PR-BYLINE
PR-LOGOIMAGE
PR-NEWSBODY
PR-PRDATE

8. After you click OK, the Elements dialog appears again. Click **Add** to repeat the process as needed to add additional elements to the static list. In this example we added one more plain text element, as shown in Figure 3–17 Press the **OK** button to complete the static list elements definition process.

Figure 3–17 The Elements Dialog

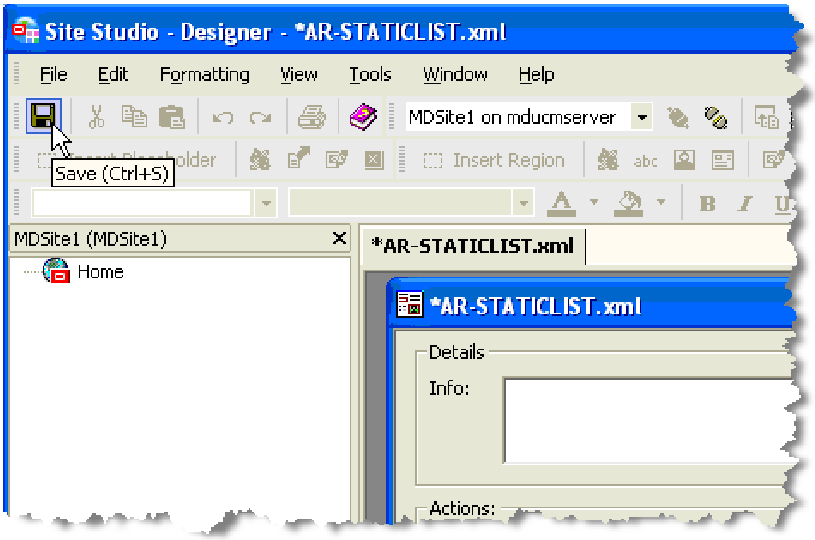


Type: static list

ARStaticListTextField_1 [AR-PLAINTEXT]
ARStaticListTextField_2 [AR-PLAINTEXT]

9. You are now returned to the top-level element definition dialog. *At this point, you must save the element definition*. Click the **Save** icon in the to-left part of the window, as shown in Figure 3–18.

Figure 3–18 Saving the Definitions

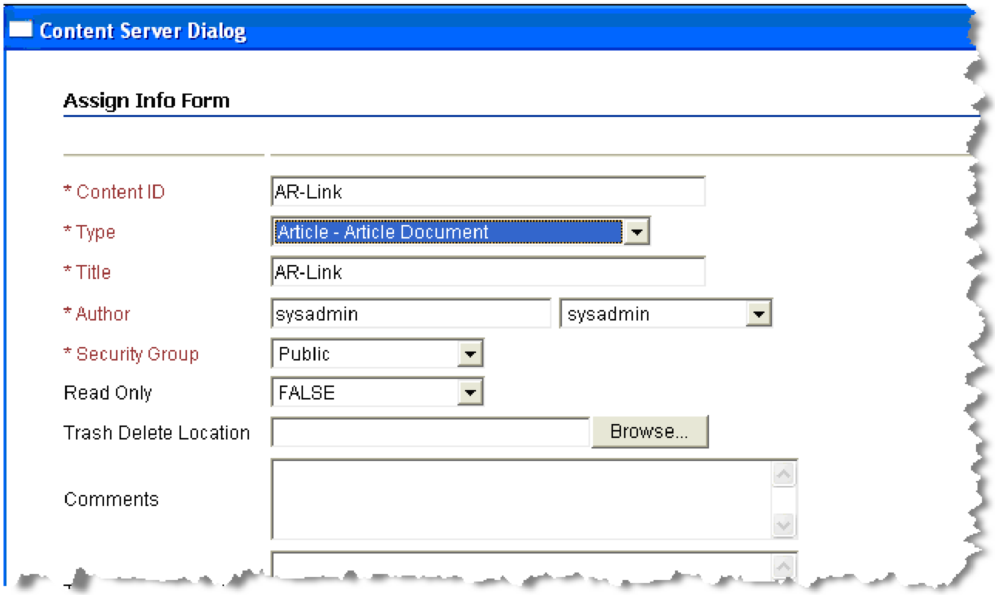


3.8.2.3.4 Create the Link Element A Link element is a custom element that is understood by the UCM VCR Adapter. To be able to create this custom element type you must have loaded the custom element form called `ss_docname_link_form.htm` into your Site Studio Configuration. Refer to the UCM documentation for more information on custom element form.

1. Select the **Page** icon and select **New** > **New Custom Element Definition**.
2. In the Content Server dialog enter a Content ID and a Title, as shown in Figure 3–19.

   Optionally, associate this Element Definition to the Content Type that it will be used with. In this example, the Article content type is used.

Figure 3–19 Assign Info Form for Link Element



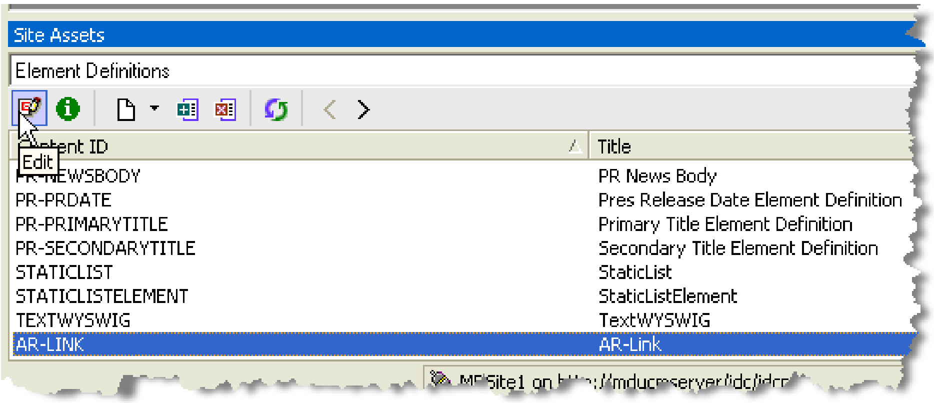
Content Server Dialog

Assign Info Form

* Content ID	AR-Link
* Type	Article - Article Document
* Title	AR-Link
* Author	sysadmin / sysadmin
* Security Group	Public
Read Only	FALSE
Trash Delete Location	
Comments	

3. Select the link element definition that you just created from the list of elements and select the **Edit** icon, as shown in Figure 3–20.

Figure 3–20 Selecting the Edit Icon



Site Assets

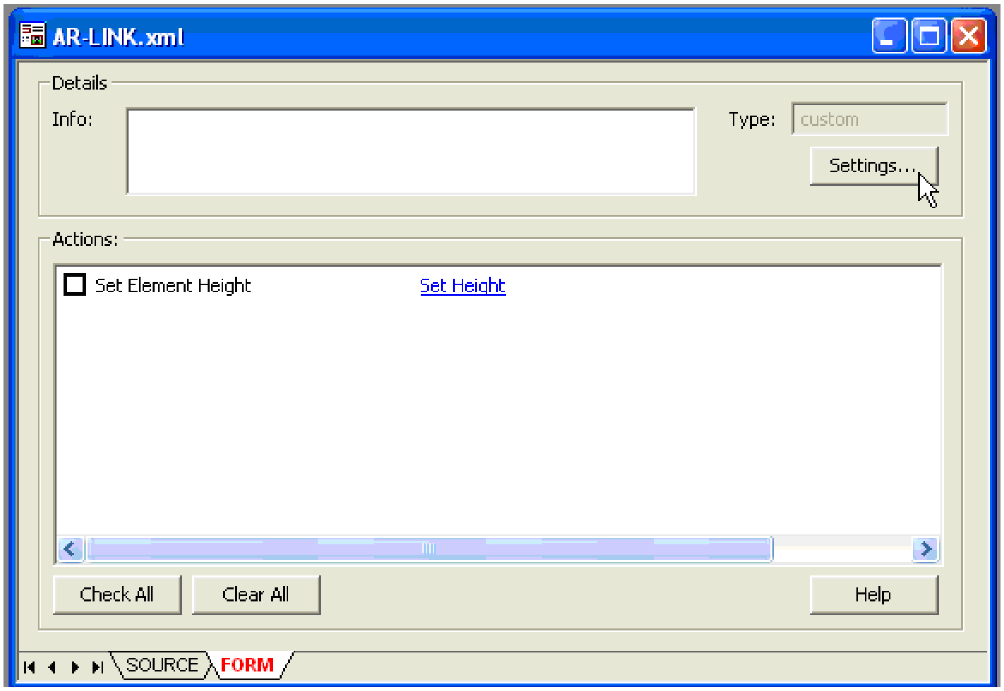
Element Definitions

Content ID	Title
PR-NEWSBODY	PR News Body
PR-PRDATE	Pres Release Date Element Definition
PR-PRIMARYTITLE	Primary Title Element Definition
PR-SECONDARYTITLE	Secondary Title Element Definition
STATICLIST	StaticList
STATICLISTELEMENT	StaticListElement
TEXTWYSWIG	TextWYSWIG
AR-LINK	AR-Link

MPSite1 on http://mducmserver/idc/idc

4. In the edit dialog for the link, click **Settings**, as shown in Figure 3–21. The custom element Settings Dialog appears.

Figure 3–21 The Edit Dialog for the Link Element

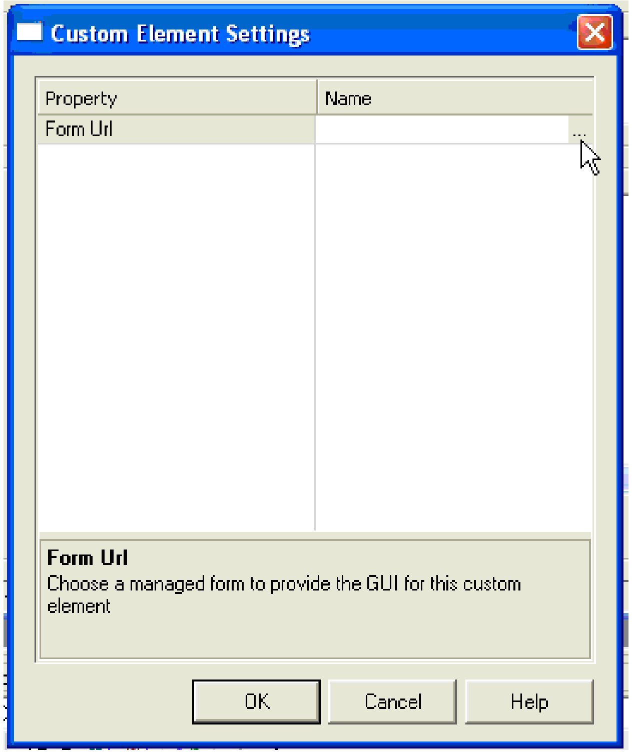


Type: custom

☐ Set Element Height   Set Height

5. In the Settings dialog, select the "**...**" (ellipses) at the end of the name field, as shown in Figure 3–22.

Figure 3–22 The Custom Element Settings Dialog

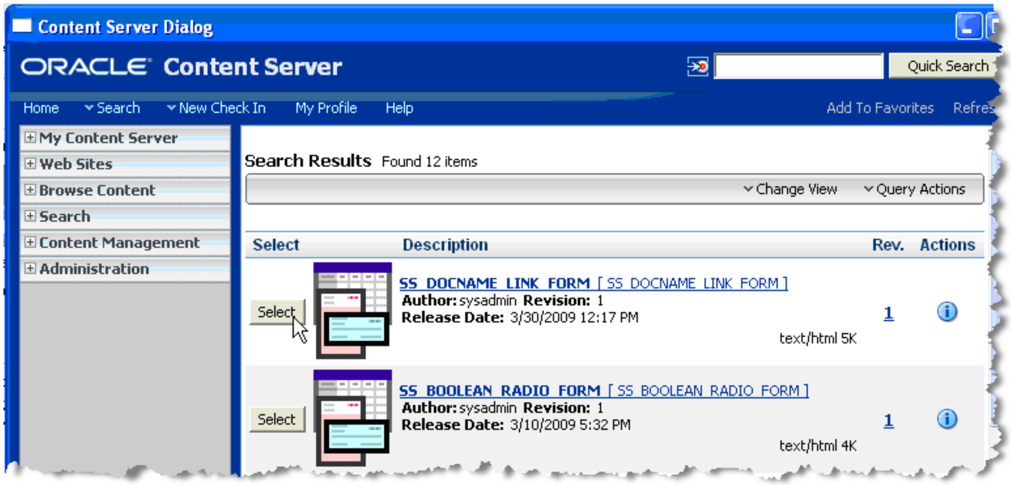


Property	Name
Form Url	...

Form Url
Choose a managed form to provide the GUI for this custom element

6. The Content Server Dialog appears showing all the custom forms available in the system. Using the **Select** button, select the form named **SS_DOCNAME_LINK_FORM [SS_DOCNAME_LINK_FORM]**, as shown in Figure 3–23.

Figure 3–23 Selecting the Form in the Search Results



Search Results Found 12 items

Select	Description	Rev.	Actions
Select	SS_DOCNAME_LINK_FORM [SS_DOCNAME_LINK_FORM] **Author:** sysadmin **Revision:** 1 **Release Date:** 3/30/2009 12:17 PM text/html 5K	1	ⓘ
Select	SS_BOOLEAN_RADIO_FORM [SS_BOOLEAN_RADIO_FORM] **Author:** sysadmin **Revision:** 1 **Release Date:** 3/10/2009 5:32 PM text/html 4K	1	ⓘ

7. The Custom element Settings dialog appears with the Name field populated. Click **OK**.

8. You are now returned to the top-level element definition dialog. *At this point, you must save the element definition*. Click the **Save** icon in the to-left part of the window.

3.8.2.4 Create a Region Definition

The next step is to create a Region definition and add the element definitions to it.

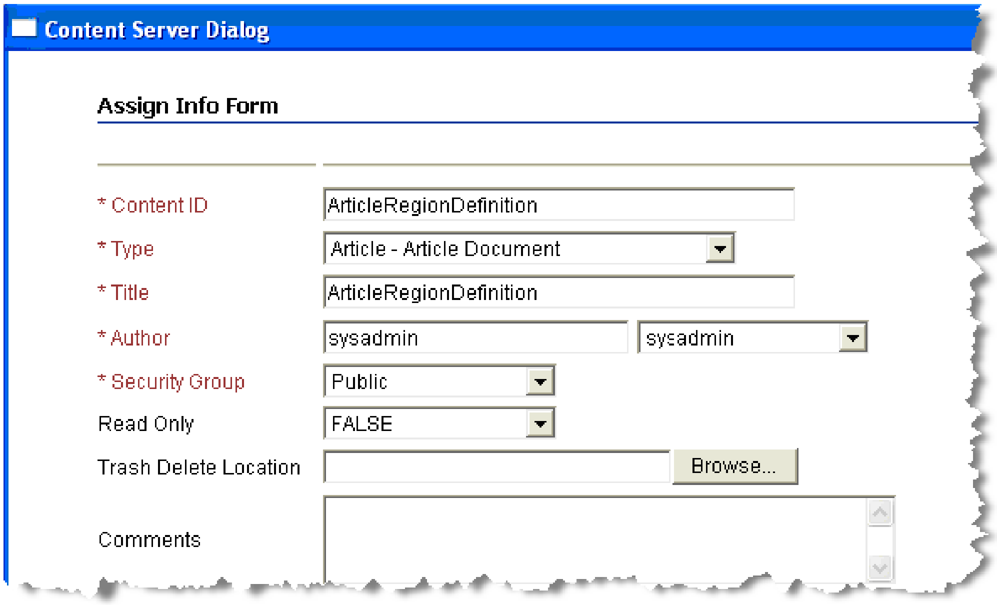
1. In the Site Assets dropdown menu in the Site Studio window, select **Region Definitions**.

2. From the **Page** icon dropdown menu, select **New > Region Definition**.

3. In the Content Server dialog enter a Content ID and a Title, as shown in Figure 3–24.

   Associate this Region Definition to the Content Type that it will be used with. In this example, the Article content type is used.

Figure 3–24 Assign Info Form for Region Definition



Content Server Dialog

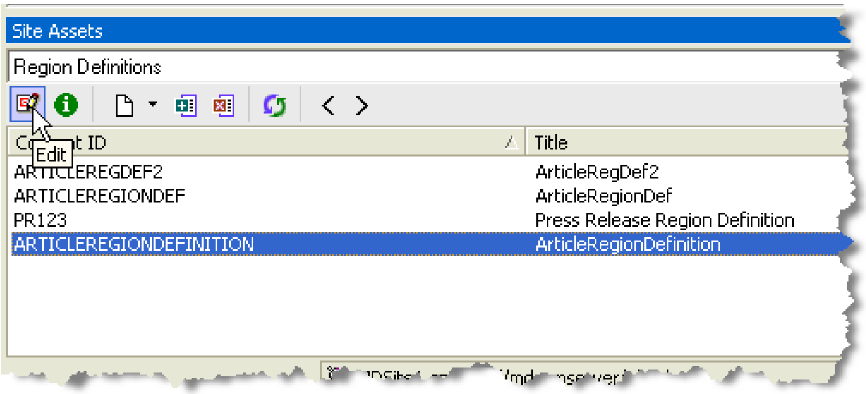
Assign Info Form

* Content ID	ArticleRegionDefinition
* Type	Article - Article Document
* Title	ArticleRegionDefinition
* Author	sysadmin sysadmin
* Security Group	Public
Read Only	FALSE
Trash Delete Location	
Comments	

4. Click **Assign Info** at the bottom of the dialog.

5. Highlight the region definition that was just created and select the **Edit** icon, as shown in Figure 3–25. An Edit Dialog appears.

Figure 3–25 Selecting the Edit Icon



Site Assets

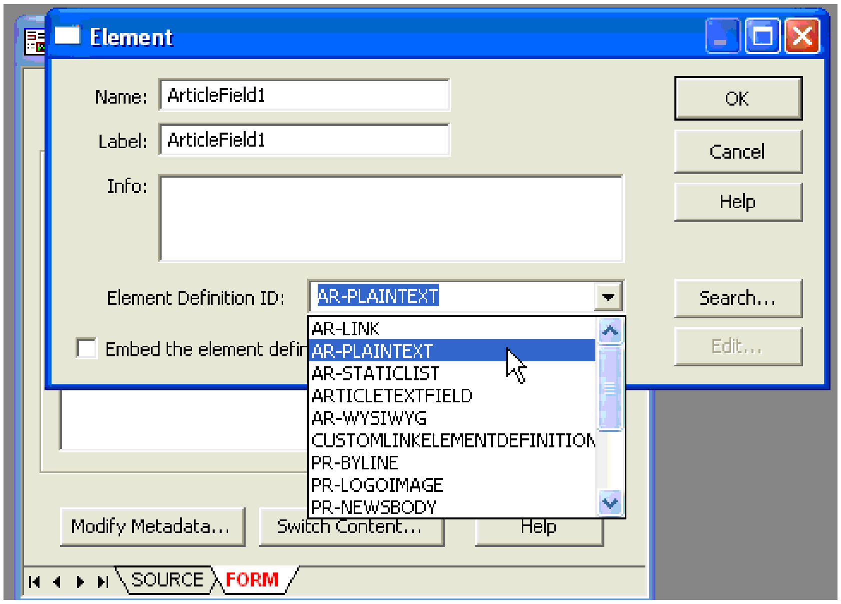
Region Definitions

Content ID	Title
ARTICLEREGDEF2	ArticleRegDef2
ARTICLEREGIONDEF	ArticleRegionDef
PR123	Press Release Region Definition
ARTICLEREGIONDEFINITION	ArticleRegionDefinition

6. In the Edit dialog, click **Add**.

7. In the next dialog, enter the Name and Label fields. The values you enter in these fields will appear in the Property fields in the VCR. Select one of the previously defined element definitions in the Element Definition ID drop down box, as shown in Figure 3–26. Do not check the **Embed the element definition inside the region definition** checkbox. Click **OK** when finished.

Figure 3–26 Selecting a Previously Defined Element Definition



Element

Name: ArticleField1

Label: ArticleField1

Info:

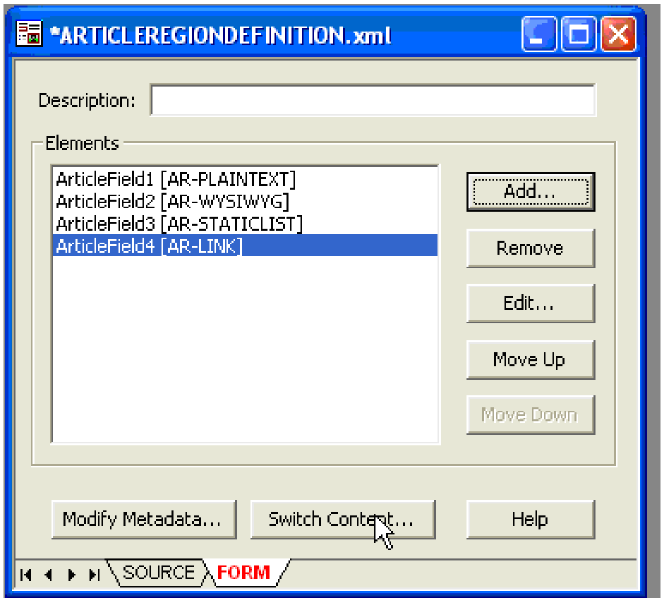
Element Definition ID: AR-PLAINTEXT

AR-LINK
AR-PLAINTEXT
AR-STATICLIST
ARTICLETEXTFIELD
AR-WYSIWYG
CUSTOMLINKELEMENTDEFINITION
PR-BYLINE
PR-LOGOIMAGE
PR-NEWSBODY

□ Embed the element defi

8. Repeat the previous step to add all the other element definitions that you wish to add, and you will see something similar to Figure 3–27.

Figure 3–27 Element Definitions Added to Region Definition List



*ARTICLEREGIONDEFINITION.xml

Description:

Elements

ArticleField1 [AR-PLAINTEXT]
ArticleField2 [AR-WYSIWYG]
ArticleField3 [AR-STATICLIST]
ArticleField4 [AR-LINK]

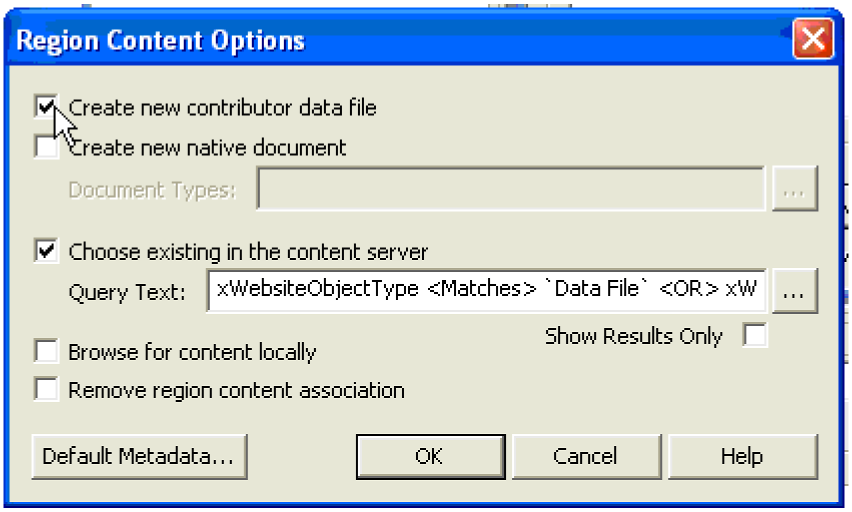
Note: The next step creates a contributor data file based on this region definition.

---

9. Select **Switch Content** at the bottom of the region element list dialog (see Figure 3–27). The Regions Content Options dialog appears.

10. In the Regions Content Options dialog, make sure that the **Create new contributor data file** option box is checked, as shown in Figure 3–28, and click **OK**.

Figure 3–28 Region Content Options Dialog



☑ Create new contributor data file

□ Create new native document

Document Types:

☑ Choose existing in the content server

Query Text: xWebsiteObjectType <Matches> `Data File` <OR> xW

Show Results Only □

□ Browse for content locally

□ Remove region content association

Note: You can optionally associate the region definition with a profile. This technique is useful to expose in the WLP content type a combination of the fields from the profile as well as elements from the region. If no profile association is configured, then the WLP content type will contain a combination of all metadata fields on the content server as well as elements from the region.

To associate a profile with the region definition, select **Modify Metadata** at the bottom of the region edit dialog. The Enable Metadata Modification dialog appears. In this dialog, select a profile from the Profile Trigger Value popup menu in the lower-right hand corner, then click **OK**. This associates the region definition with that profile.

---

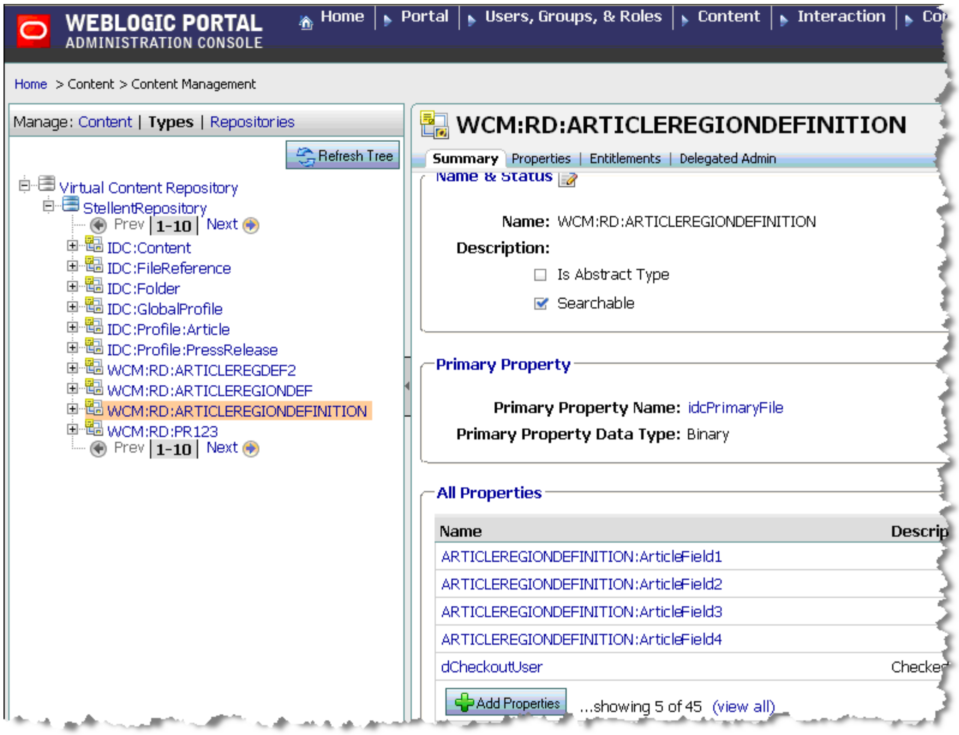
11. You are now returned to the top-level element definition dialog. *At this point, you must save the element definition*. Click the **Save** icon in the to-left part of the window.

The creation of the regions-based content type is now complete.

3.8.2.5 Viewing the Region-Based Content Type from the WLP Repository

Figure 3–29 and Figure 3–30 show what the regions-based content type created in this section looks like when viewed from the WebLogic Portal Administration Console.

Figure 3–29 Region-Based Content Type Surfaced in WebLogic Portal



Name: WCM:RD:ARTICLEREGIONDEFINITION

Description:

□ Is Abstract Type

☑ Searchable

Primary Property Name: idcPrimaryFile

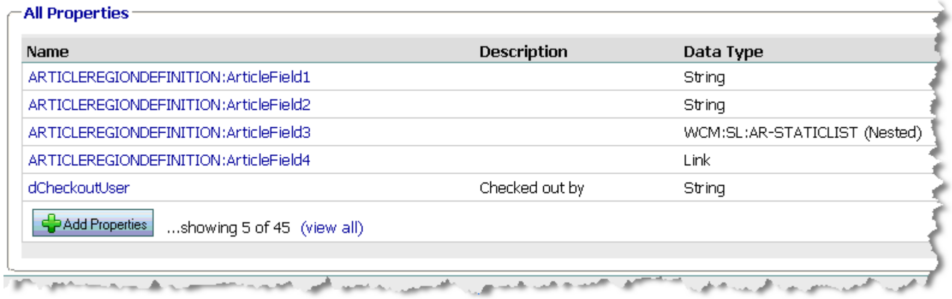
Primary Property Data Type: Binary

Name	Descrip
ARTICLEREGIONDEFINITION:ArticleField1	
ARTICLEREGIONDEFINITION:ArticleField2	
ARTICLEREGIONDEFINITION:ArticleField3	
ARTICLEREGIONDEFINITION:ArticleField4	
dCheckoutUser	Checke

...showing 5 of 45 (view all)

Figure 3–30 shows the property descriptions for the content type.

Figure 3–30 Property Descriptions



Name	Description	Data Type
ARTICLEREGIONDEFINITION:ArticleField1		String
ARTICLEREGIONDEFINITION:ArticleField2		String
ARTICLEREGIONDEFINITION:ArticleField3		WCM:SL:AR-STATICLIST (Nested)
ARTICLEREGIONDEFINITION:ArticleField4		Link
dCheckoutUser	Checked out by	String

...showing 5 of 45 (view all)

3.8.3 Creating a New Contributor Data File Based On a Region Definition

This section explains how to use Oracle Site Studio and the UCM Administration console to create content using region definitions. The goal of this procedure is to create region-based content in UCM and have that content and its properties appear in the WebLogic Portal VCR. The Site Studio region elements will be expressed as property fields in the VCR.

For background information on Site Studio, see Section 3.1.2, "Best Practices for Modeling Content."

3.8.3.1 Prerequisites

As a best practice, it is recommended that you first create a profile definition to associate with a newly created region-based content type definition. If you do not associate a region definition with a profile, then by default the region-based content type will be associated with a global profile and all UCM metadata fields will be exposed when viewing instances of content based on the region. UCM uses profiles to manage how metadata is structured and displayed.

> **Note:** The example in this section uses the Article profile definition discussed in the previous example, Section 3.8.1, "Creating Profile-Based Content Types with the UCM Administration Console."

3.8.3.2 Overview of Steps

The basic steps described in this section are:

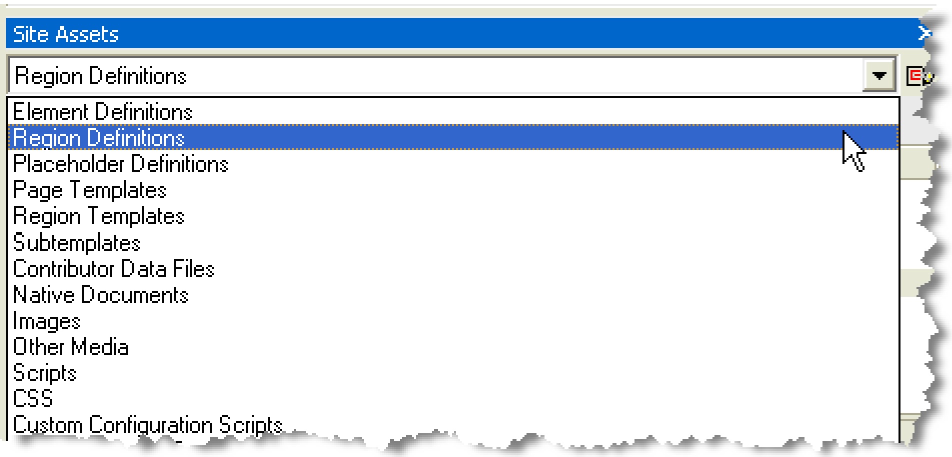
1. Select the region definition from which the new content node will be created.
2. Create an empty contributor data file based on that region definition.
3. Edit the contributor data file to populate the content fields.

3.8.3.3 Selecting the Region Definition

This section describes how to select the region definition in Site Studio. Region definitions can also be selected outside of Site Studio Designer, in the Content Server.

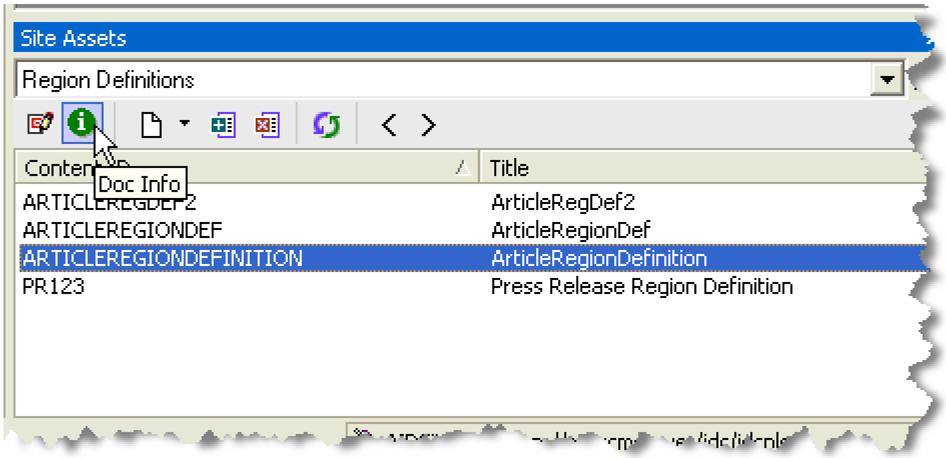
1. Start Oracle Site Studio.
2. In Site Studio, select **Region Definitions** from the Site Assets dropdown menu, as shown in Figure 3–31.

Figure 3–31 Site Assets Dropdown Menu



3. From the resulting list of region definitions, select a region type on which to base the new content node, and select the **Doc Info** button to launch the Oracle Content Server Administration console, as shown in Figure 3–32. It is also possible to create contributor data files outside of Site Studio Designer, on the Content Server. The important part is to first locate the region definition.

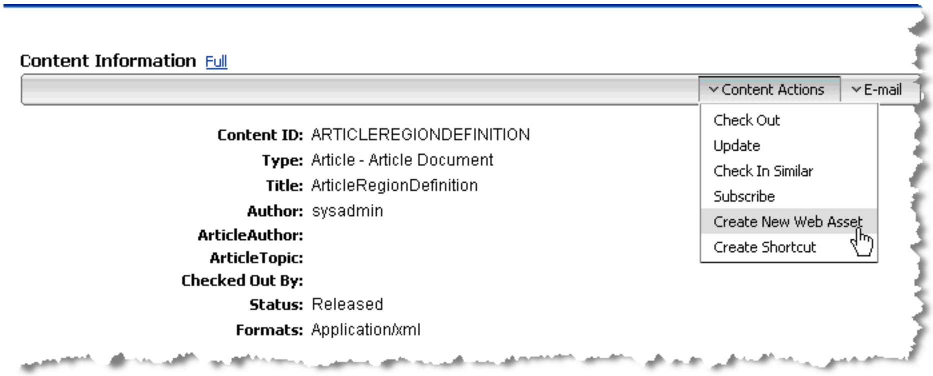
Figure 3–32 Selecting the Doc Info Icon



Content ID	Title
ARTICLEREGDEF2	ArticleRegDef2
ARTICLEREGIONDEF	ArticleRegionDef
ARTICLEREGIONDEFINITION	ArticleRegionDefinition
PR123	Press Release Region Definition

4. From the Content Server Administration console, select **Create New Web Asset** from the Content Actions dropdown menu, as shown in Figure 3–33.

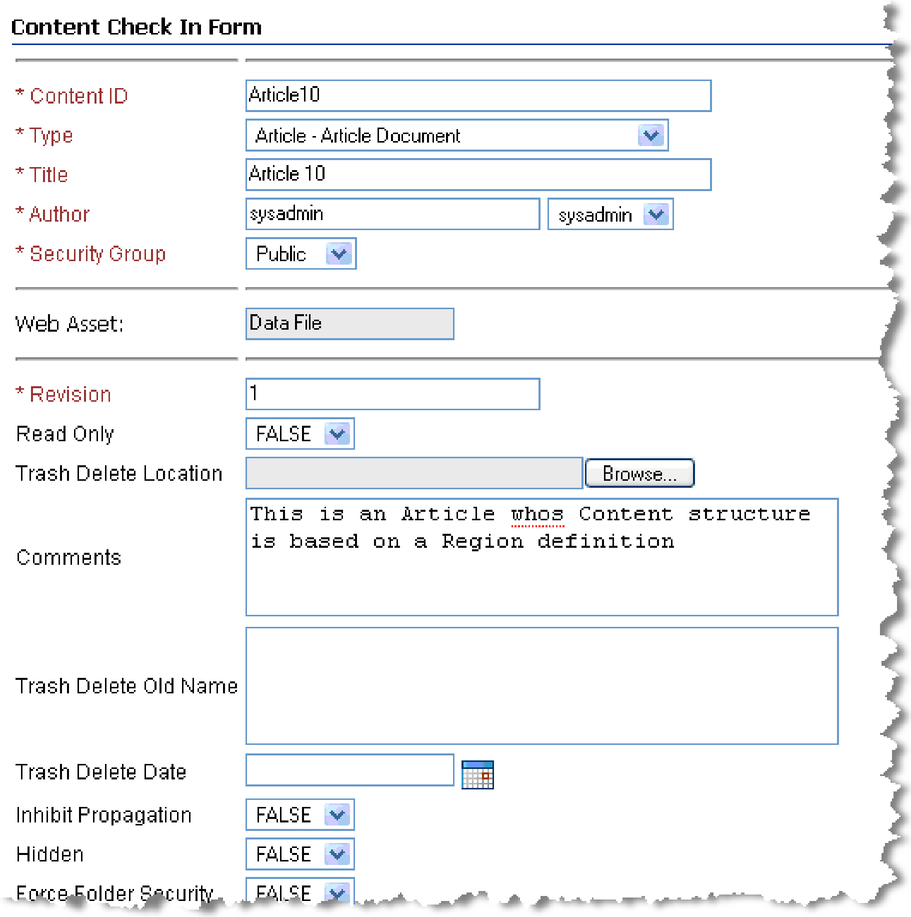
Figure 3–33 Selecting from the Content Actions Menu



Content ID: ARTICLEREGIONDEFINITION
Type: Article - Article Document
Title: ArticleRegionDefinition
Author: sysadmin
ArticleAuthor:
ArticleTopic:
Checked Out By:
Status: Released
Formats: Application/xml

5. The Content Check In form appears for the Contributor data file. Fill out the ContentID and Title fields. Be sure and associate the Type field with the appropriate profile. In this case, select **Article Document**. Set the Author and Security group fields as appropriate. See Figure 3–34.

Figure 3–34 Content Check In Form



Content Check In Form

* Content ID	Article10
* Type	Article - Article Document
* Title	Article 10
* Author	sysadmin sysadmin
* Security Group	Public
Web Asset:	Data File
* Revision	1
Read Only	FALSE
Trash Delete Location	Browse...
Comments	This is an Article whos Content structure is based on a Region definition
Trash Delete Old Name	
Trash Delete Date	
Inhibit Propagation	FALSE
Hidden	FALSE
Force Folder Security	FALSE

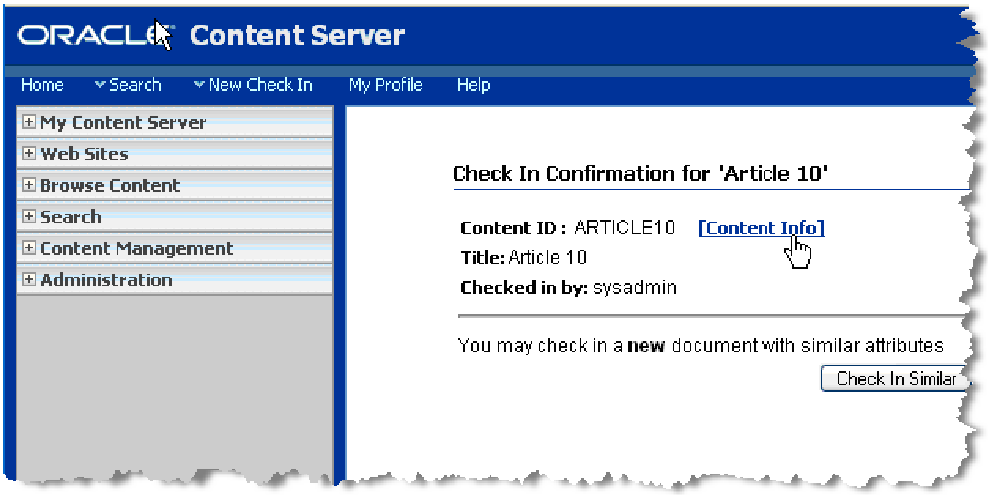
Note: Be sure and set the Folder location to save the data file otherwise it will not be browseable by the Portal Administration Console or Content Presenter, though it can be located via search.

---

6. Select **Check In**. The Check in Confirmation Dialog appears.

7. In the Check In Confirmation Dialog, click the **Content Info** link to navigate to the entry for the contributor data file, as shown in

Figure 3–35 Clicking the Content info Link



Check In Confirmation for 'Article 10'

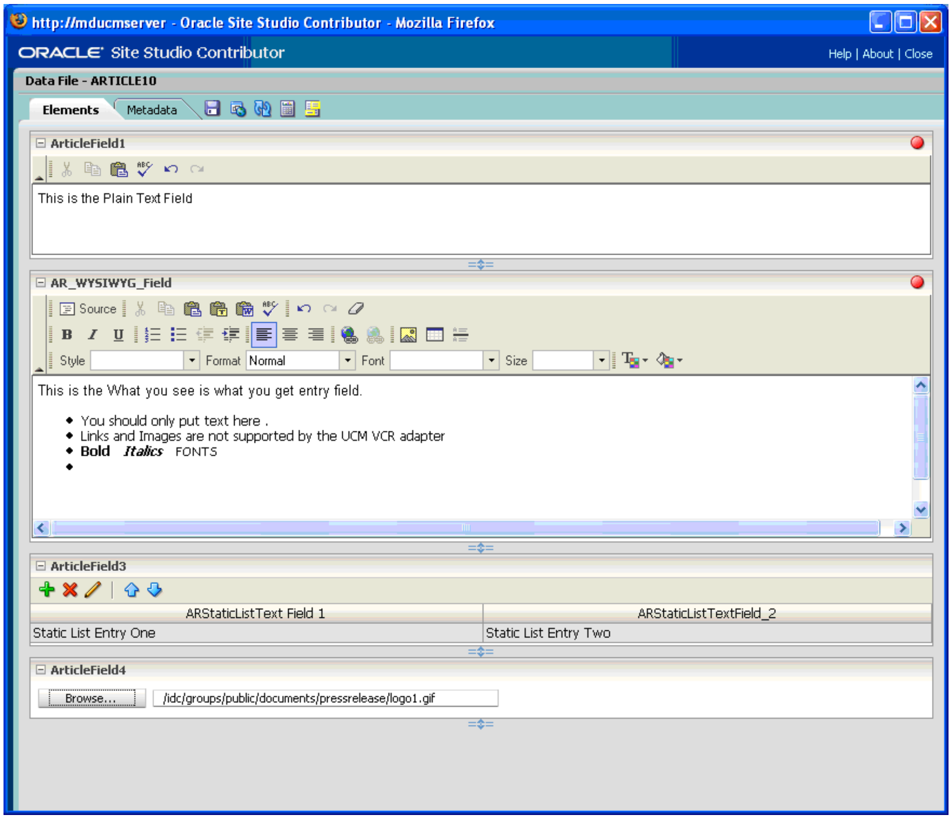
Content ID : ARTICLE10 [Content Info]
Title: Article 10
Checked in by: sysadmin

You may check in a new document with similar attributes

8. From the Content Information screen, select **Content Actions > Edit Data File** from the Content Actions dropdown menu. The Site Studio Contributor Data File Editor appears.

9. Use the editor to enter information in each of the pre-defined region fields (plain text, WYSIWYG, static list, and link. Also, select the Metadata tab and enter metadata for the contributor data file. Note that the available metadata fields are associated with the profile with which this content is associated. See Figure 3–36 and Figure 3–37.

Figure 3–36 Editing the Contributor Data File


http://mducmserver - Oracle Site Studio Contributor - Mozilla Firefox
ORACLE Site Studio Contributor
Help | About | Close
Elements
Metadata
Source
B
I
U
Style
Format
Font
Size
Browse...

Data File - ARTICLE10

ArticleField1

This is the Plain Text Field

AR_WYSIWYG_Field

Format: Normal

This is the What you see is what you get entry field.

- You should only put text here .
- Links and Images are not supported by the UCM VCR adapter
- **Bold** ***Italics*** FONTS
-

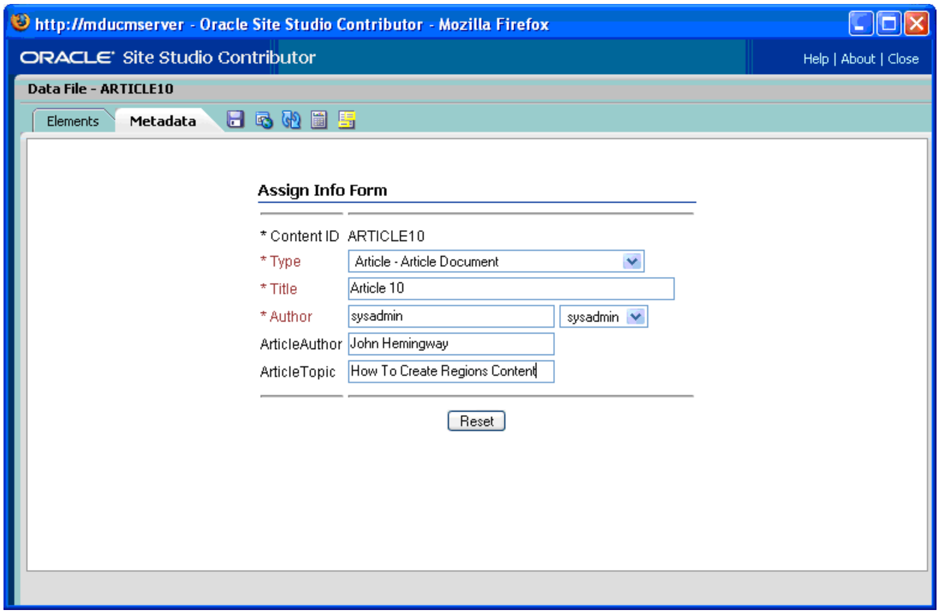
ArticleField3

ARStaticListText Field 1	ARStaticListTextField_2
Static List Entry One	Static List Entry Two

ArticleField4

/idc/groups/public/documents/pressrelease/logo1.gif

Figure 3–37 Editing the File's Metadata

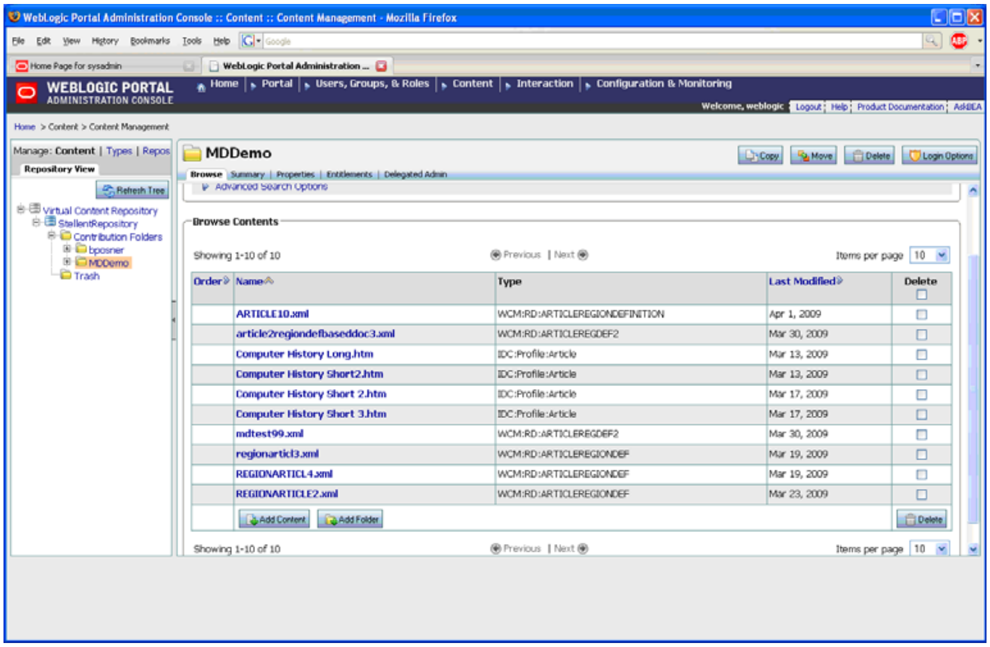

http://mducmserver - Oracle Site Studio Contributor - Mozilla Firefox
ORACLE Site Studio Contributor
Help | About | Close
Data File - ARTICLE10
Elements
Metadata
Assign Info Form
Reset

* Content ID	ARTICLE10
* Type	Article - Article Document
* Title	Article 10
* Author	sysadmin / sysadmin
ArticleAuthor	John Hemingway
ArticleTopic	How To Create Regions Content

10. Save the file.

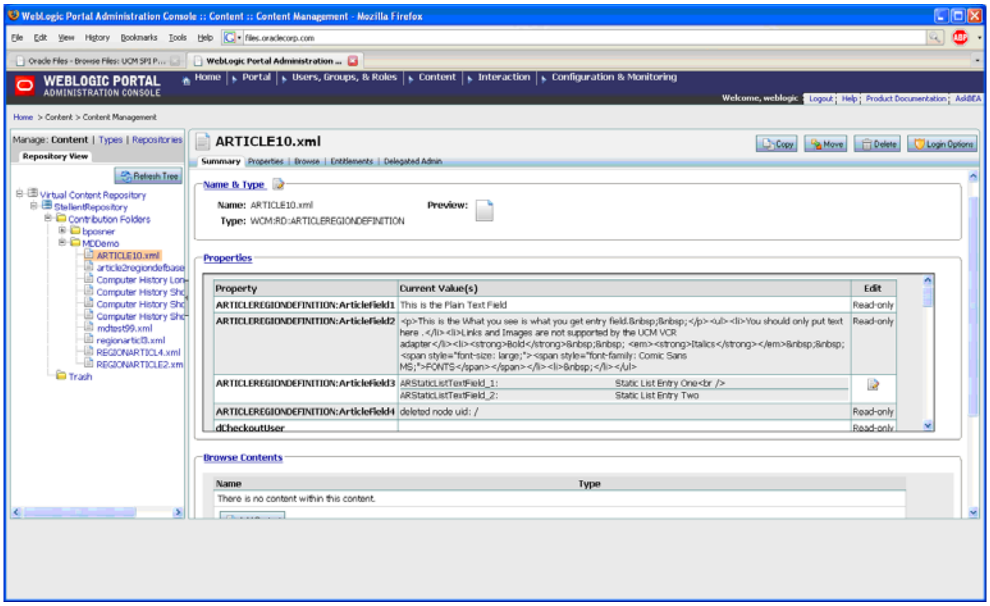
3.8.3.3.1 Viewing the Content in WLP The following figures illustrate how this new content appears in the WebLogic Portal Administration Console. Remember that all Site Studio Region Elements are presented as WLP properties fields. See Figure 3–38, Figure 3–39, Figure 3–40, and Figure 3–41.

Figure 3–38 UCM Content Surfaced in VCR


WebLogic Portal Administration Console :: Content :: Content Management - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Home Page for sysadmin
WebLogic Portal Administration ...
WEBLOGIC PORTAL
ADMINISTRATION CONSOLE
Home | Portal | Users, Groups, & Roles | Content | Interaction | Configuration & Monitoring
Welcome, weblogic | Logout | Help | Product Documentation | AskBEA
Home > Content > Content Management
Manage: Content | Types | Repos
Repository View
Refresh Tree
Virtual Content Repository
StellentRepository
Contribution Folders
bposner
MDDemo
Trash
MDDemo
Copy
Move
Delete
Login Options
Browse | Summary | Properties | Entitlements | Delegated Admin
Advanced Search Options
Browse Contents
Showing 1-10 of 10
Previous | Next
Items per page 10
Add Content
Add Folder
Delete

Order	Name	Type	Last Modified	Delete □
	ARTICLE10.xml	WCM:RD:ARTICLEREGIONDEFINITION	Apr 1, 2009	□
	article2regiondefbaseddoc3.xml	WCM:RD:ARTICLEREGDEF2	Mar 30, 2009	□
	Computer History Long.htm	IDC:Profile:Article	Mar 13, 2009	□
	Computer History Short2.htm	IDC:Profile:Article	Mar 13, 2009	□
	Computer History Short 2.htm	IDC:Profile:Article	Mar 17, 2009	□
	Computer History Short 3.htm	IDC:Profile:Article	Mar 17, 2009	□
	mdtest99.xml	WCM:RD:ARTICLEREGDEF2	Mar 30, 2009	□
	regionarticl3.xml	WCM:RD:ARTICLEREGIONDEF	Mar 19, 2009	□
	REGIONARTICL4.xml	WCM:RD:ARTICLEREGIONDEF	Mar 19, 2009	□
	REGIONARTICLE2.xml	WCM:RD:ARTICLEREGIONDEF	Mar 23, 2009	□

Figure 3–39 UCM Content Fields Surfaced in VCR


WebLogic Portal Administration Console :: Content :: Content Management - Mozilla Firefox
File Edit View History Bookmarks Tools Help
files.oraclecorp.com
WEBLOGIC PORTAL ADMINISTRATION CONSOLE
Home | Portal | Users, Groups, & Roles | Content | Interaction | Configuration & Monitoring
Welcome, weblogic | Logout | Help | Product Documentation | AskBEA
Home > Content > Content Management
Manage: Content | Types | Repositories
Repository View
Refresh Tree
Virtual Content Repository
StellentRepository
Contribution Folders
bposner
MDDemo
ARTICLE10.xml
Trash
Copy
Move
Delete
Login Options
Summary | Properties | Browse | Entitlements | Delegated Admin

ARTICLE10.xml

Name & Type

Name: ARTICLE10.xml

Type: WCM:RD:ARTICLEREGIONDEFINITION

Preview:

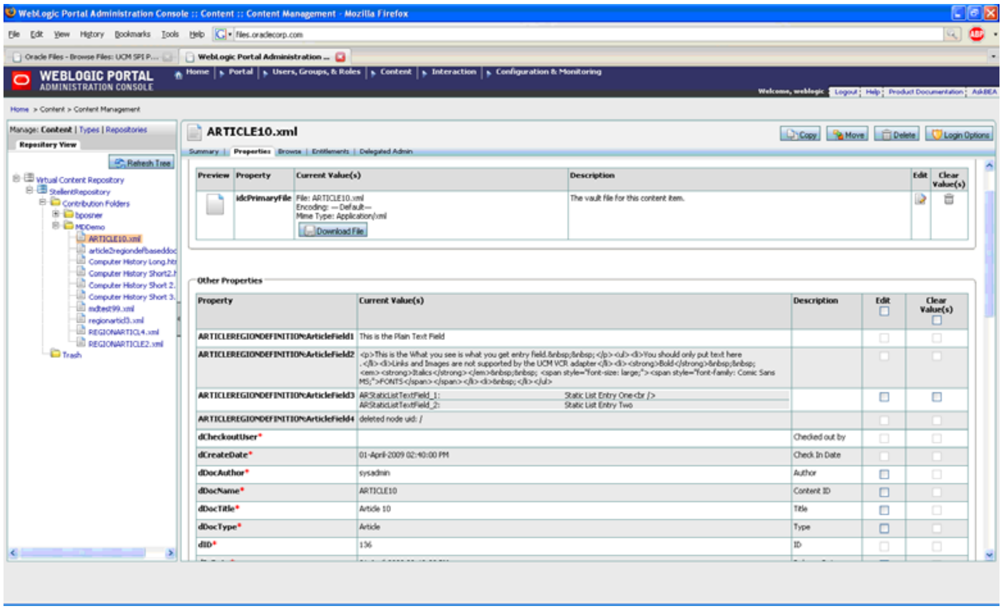
Properties

Property	Current Value(s)		Edit
ARTICLEREGIONDEFINITION:ArticleField1	This is the Plain Text Field		Read-only
ARTICLEREGIONDEFINITION:ArticleField2	<p>This is the What you see is what you get entry field.&nbsp;&nbsp;</p><ul><li>You should only put text here .</li><li>Links and Images are not supported by the UCM VCR adapter</li><li><strong>Bold</strong> <em><strong>Italics</strong></em> <span style="font-size: large;"><span style="font-family: Comic Sans MS;">FONTS</span></span></li><li>&nbsp;</li></ul>		Read-only
ARTICLEREGIONDEFINITION:ArticleField3	ARStaticListTextField_1: ARStaticListTextField_2:	Static List Entry One<br /> Static List Entry Two	
ARTICLEREGIONDEFINITION:ArticleField4	deleted node uid: /		Read-only
dCheckoutUser			Read-only

Browse Contents

Name	Type
There is no content within this content.	

Figure 3–40 UCM Content Properties in the VCR


WebLogic Portal Administration Console :: Content :: Content Management - Mozilla Firefox
File Edit View History Bookmarks Tools Help
files.oraclecorp.com
WEBLOGIC PORTAL ADMINISTRATION CONSOLE
Home | Portal | Users, Groups, & Roles | Content | Interaction | Configuration & Monitoring
Welcome, weblogic | Logout | Help | Product Documentation | AskBEA
Home > Content > Content Management
Manage: Content | Types | Repositories
Repository View
Refresh Tree
Virtual Content Repository
StellentRepository
Contribution Folders
bposner
MDDemo
ARTICLE10.xml
Trash
Copy
Move
Delete
Login Options
Summary | Properties | Browse | Entitlements | Delegated Admin
Download File

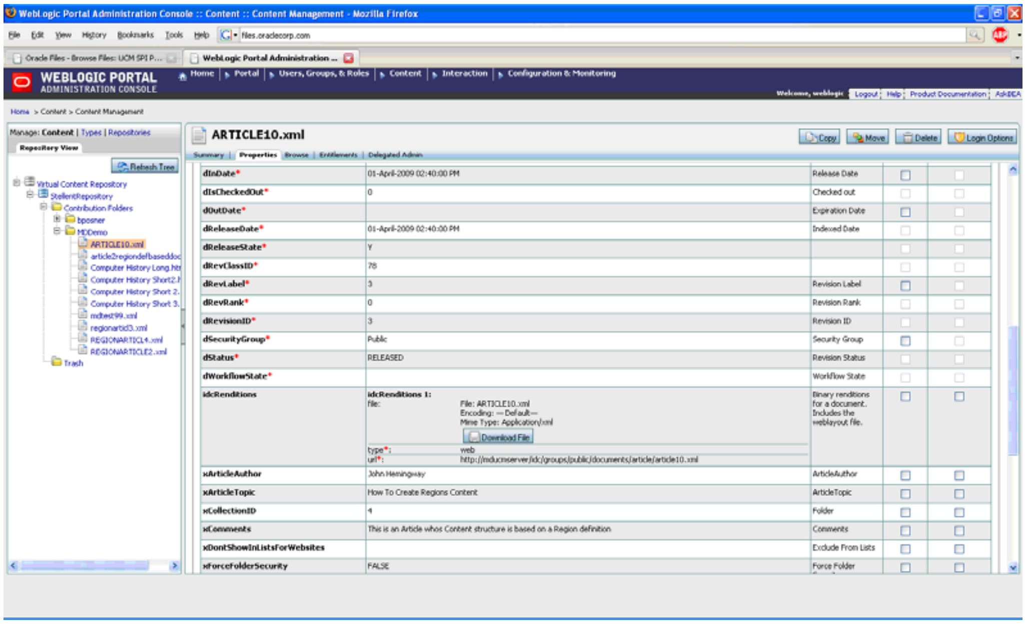
ARTICLE10.xml

Preview	Property	Current Value(s)	Description	Edit	Clear Value(s)
	idcPrimaryFile	File: ARTICLE10.xml Encoding: —Default— Mime Type: Application/xml	The vault file for this content item.		

Other Properties

Property	Current Value(s)		Description	Edit ☐	Clear Value(s) ☐
ARTICLEREGIONDEFINITION%ArticleField1	This is the Plain Text Field			☐	☐
ARTICLEREGIONDEFINITION%ArticleField2	<p>This is the What you see is what you get entry field. </p> <ul> <li>You should only put text here . </li> <li>Links and Images are not supported by the UCM VCR adapter </li> <li> <strong>Bold</strong> <em> <strong>Italics</strong> </em> <span style="font-size: large;"> <span style="font-family: Comic Sans MS;">FONTS</span> </span> </li> <li> </li> </ul>			☐	☐
ARTICLEREGIONDEFINITION%ArticleField3	ARStaticListTextField_1: ARStaticListTextField_2:	Static List Entry One<br /> Static List Entry Two		☐	☐
ARTICLEREGIONDEFINITION%ArticleField4	deleted node uid: /			☐	☐
dCheckoutUser*			Checked out by	☐	☐
dCreateDate*	01-April-2009 02:40:00 PM		Check In Date	☐	☐
dDocAuthor*	sysadmin		Author	☐	☐
dDocName*	ARTICLE10		Content ID	☐	☐
dDocTitle*	Article 10		Title	☐	☐
dDocType*	Article		Type	☐	☐
dID*	136		ID	☐	☐

Figure 3–41 UCM Content Properties Surfaced in the VCR


WebLogic Portal Administration Console :: Content :: Content Management - Mozilla Firefox
File Edit View History Bookmarks Tools Help
files.oracledcorp.com
Oracle Files - Browse Files: UCM SPI P...
WebLogic Portal Administration ...
WEBLOGIC PORTAL
ADMINISTRATION CONSOLE
Home
Portal
Users, Groups, & Roles
Content
Interaction
Configuration & Monitoring
Welcome, weblogic
Logout
Help
Product Documentation
AskBEA
Home > Content > Content Management
Manage: Content | Types | Repositories
Repository View
Refresh Tree
Virtual Content Repository
StellentRepository
Contribution Folders
bposner
MDDemo
ARTICLE10.xml
article2regiondefbaseddoc
Computer History Long.htr
Computer History Short2.h
Computer History Short 2.
Computer History Short 3.
mdtest99.xml
regionartid3.xml
REGIONARTICL4.xml
REGIONARTICLE2.xml
Trash
ARTICLE10.xml
Copy
Move
Delete
Login Options
Summary
Properties
Browse
Entitlements
Delegated Admin
Download File

dInDate*	01-April-2009 02:40:00 PM	Release Date	□	□
dIsCheckedOut*	0	Checked out	□	□
dOutDate*		Expiration Date	□	□
dReleaseDate*	01-April-2009 02:40:00 PM	Indexed Date	□	□
dReleaseState*	Y		□	□
dRevClassID*	78		□	□
dRevLabel*	3	Revision Label	□	□
dRevRank*	0	Revision Rank	□	□
dRevisionID*	3	Revision ID	□	□
dSecurityGroup*	Public	Security Group	□	□
dStatus*	RELEASED	Revision Status	□	□
dWorkflowState*		Workflow State	□	□
idcRenditions	**idcRenditions 1:** file: File: ARTICLE10.xml Encoding: —Default— Mime Type: Application/xml type*: web url*: http://mducmserver/idc/groups/public/documents/article/article10.xml	Binary renditions for a document. Includes the weblayout file.	□	□
xArticleAuthor	John Hemingway	ArticleAuthor	□	□
xArticleTopic	How To Create Regions Content	ArticleTopic	□	□
xCollectionID	4	Folder	□	□
xComments	This is an Article whos Content structure is based on a Region definition	Comments	□	□
xDontShowInListsForWebsites		Exclude From Lists	□	□
xForceFolderSecurity	FALSE	Force Folder	□	□