Oracle® Fusion Middleware

Configuring the XDS Connector for Oracle WebCenter Content

11*g* Release 1 (11.1.1)

E35898-01

July 2013

This document describes how to configure and enable Cross Enterprise Document Sharing (XDS) to use WebCenter Content as an XDS repository.

The following topics are discussed:

About XDS Usage

Cross Enterprise Document Sharing (XDS) is a standard used by the *Integrating The Healthcare Enterprise* (IHD) initiative to help medical devices that create documents, such as MRI machines, to better communicate with content repositories. For more information about the initiative, including details about the XDS specification, see `http://wiki.ihe.net`.

XDS is a data exchange specification composed of the following elements which communicate with each other using standard SOAP-based web services:

- **Patient Identity Source**: identity management used to determine the user.
- **Document Registry**: used to store metadata for a medical document and used for searching. The location of the registry is set during the configuration of the WebCenter Content XDS Connector.
- **Document Consumer**: a web interface or other type of viewer used to view the medical content.
- **Document Source**: the creator of the medical content. This is typically an X-ray or MRI machine that creates medical images. The unique ID for the document is supplied by the Document Source. It is not generated by WebCenter Content.
- **Document Repository**: the content repository used to store the medical content.

Activity must be logged to an Audit Trail and Node Authentication (ATNA) compatible server in order to be compliant.

The WebCenter Content XDS Connector is used to allow Oracle WebCenter Content and the content server to act as a repository in an XDS exchange.

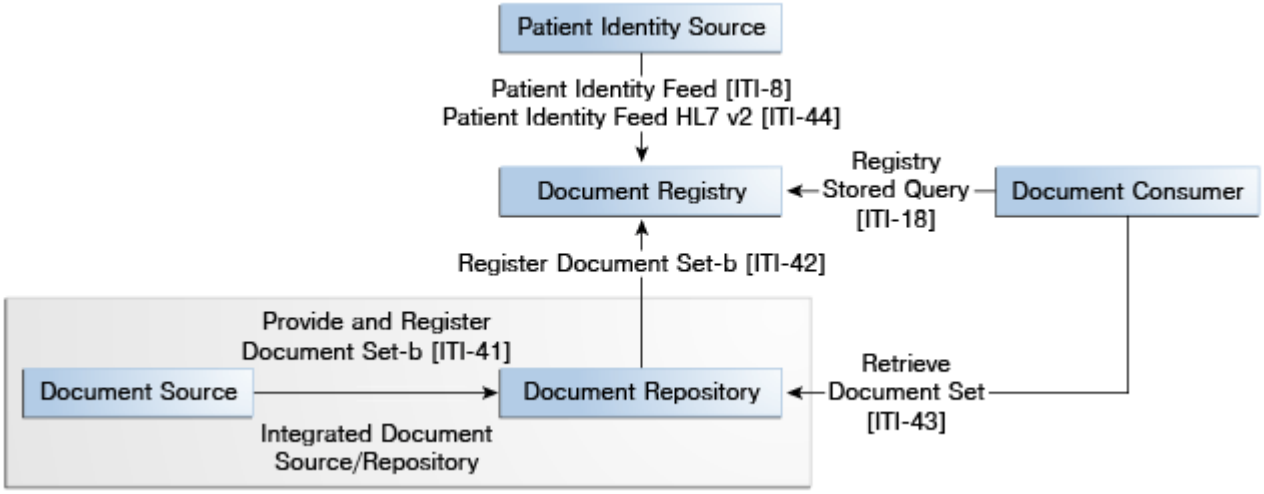
With the WebCenter Content XDS Connector enabled, documents are downloaded to the repository in WebCenter Content. The downloading is based on a unique XDS-specific document ID supplied by the Document Source.

The WebCenter Content XDS Connector is an XDS repository that must be integrated with a separate XDS registry. Any calls that are only for a registry are not implemented. Any call that requires both a registry and a repository, such as the Provide and Register Document Set-b, use WebCenter Content as the XDS repository and a remote XDS registry for the metadata.

The WebCenter Content XDS Connector listens for requests from Document Sources and performs one of two actions:

- store a document with its metadata: the document is stored and the metadata is forwarded to a central Document Registry.
- store a document: the document and its unique ID are stored.

The following figure shows the general flow of information in an XDS data exchange system.



Design of the WebCenter Content XDS Connector

The XDS repository is specified in two Web Service Description Language (WSDL) documents:

- XDS.b_DocumentRepository.wsdl
- XDS-I.b_ImagingDocumentSource.wsdl

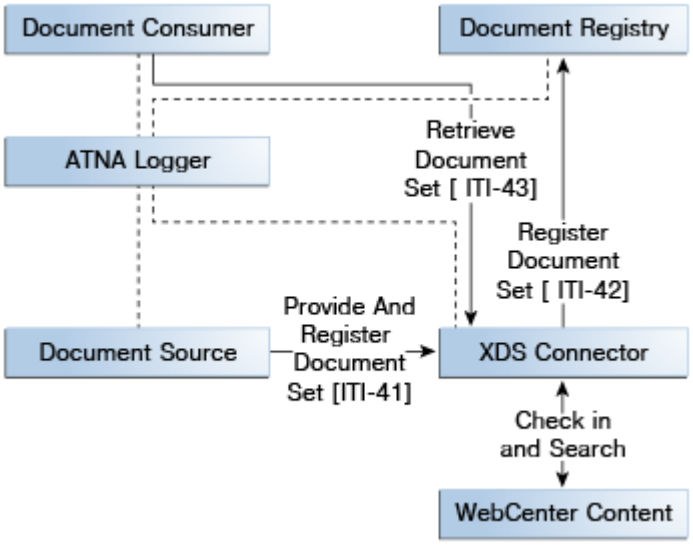
These WSDLs provide implementation requirements for the following XDS-compatible web services:

- XDS.b Provide and Register Document Set-b [ITI-41]
- XDS.b Retrieve Document Set [ITI-43]
- XDS-I.b Provide and Register Image Document Set [RAD-68]
- XDS-I.b Retrieve Imaging Document Set [RAD-69]

The XDS.b Register Document Set-b [ITI-42] registry service is included but is not implemented. Even though this implementation does not support the XDS Registry service calls, it stores the metadata from a Provide and Register request, bundling the data into an XML file which is checked into WebCenter Content with the image document. If you should lose your XDS registry, the entire history can thus be restored using the XDS repository in WebCenter Content.

Note that the WSDL definition for the Provide and Register transaction sent by the Imaging Document Source [RAD-68] is no different than the transaction sent by the XDS.b Document Source in [ITI-41].

The following figure demonstrates how the WebCenter Content XDS Connector works with the WebCenter Content repository to store information.



Configuring WebCenter Content

In order to use the WebCenter Content XDS Connector with WebCenter Content, you must create a profile as well as create a local user, as described in the following steps. Example names are given in the following text. You can use other names if needed.

You should create the custom metadata field `xXdsDocUid` before creating the profile. This metadata field is used to store the unique ID of an XDS document.

For details about creating custom metadata fields and information about creating profiles, see *Managing Oracle WebCenter Content*. For details about adding users, see *Administering Oracle WebCenter Content*.

1. Create the following metadata fields in WebCenter Content and set a default derived value for these fields:
   - The `xXdsDocUid` memo field. This is the unique ID from the XDS source that is passed in a ProvideAndRegister service request.
   - The `xXdsDocGuid` memo field. This is the auto-generated and globally unique ID for this document.
   - The `xXdsDocMimeType` Long Text metadata field. This is the mime type of the document.
   - The `xXdsDocHash` Long Text metadata field. This is the hash of the document, a HEX encoded SHA1 digest of the file.
2. Create a trigger named `xdsDocument` in the `xIdcProfileTrigger` list of triggers.
3. Create a profile rule named `xdsDefaults`.
4. Create a content profile named `xdsDocument`. Add an `xdsDefaults` rule to the profile. In this rule, set the default metadata values for all XDS documents checked in to the system. For example, you will need to set a derived value for security group (such as `public`) and a derived value for content type (such as `document`).
5. Create a new local user named `xdsUser`.

6. Grant the xdsUser sufficient rights to contribute and download XDS documents. For example, grant the user read/write access to the Public security group.
7. Set the `IntradocServerPort` variable in WebCenter Content Server's config.cfg file to enable direct access through the Admin Server.

   ```
   IntradocServerPort=4444
   ```

   The config.cfg file is typically located in a location similar to the following:

   ```
   FMW_HOME\user_projects\domains\base_domain\ucm\cs\config
   ```

Configuring the WebCenter Content XDS Connector WAR File

The information about how to connect to WebCenter Content is bundled into this WAR file in the web.xml file.

For a production deployment, make changes to the WAR file with a Deployment Plan. For a development installation, changes can be made manually.

Locate the Web Application Archive (WAR) file in the ucm/Distribution/XDS directory.

Follow these steps to configure the file:

1. Unzip the ucmxds.war file.
2. Navigate to the WEB-INF directory and open the web.xml file for editing. Edit the following information:
   - Modify the `UcmUrl` value to be the RIDC connection string to WebCenter Content. This must be the IdcURL that can be used by RIDC to determine the connection information to the server. It takes the form:

     ```
     idc://servername:port
     ```

   - Make sure to have a globally unique XDS ID. Modify the `XdsRepositoryUid` value to be unique ID for this XDS repository, for example:

     ```
     1.3.6.1.4.1.21367.13.40.92
     ```

   - Modify the `XdsRegistryEndpoint` value to point to your XDS registry's web service endpoint. If an XDS registry is not set up, use the following public registry for testing:

     ```
     http://ihexds.nist.gov:12080/tf6/services/xdsregistryb
     ```

   - Set the URL to your ATNA logger. Modify the `AtnaRepositoryURI` to point to the ATNA logging endpoint, for example:

     ```
     syslog://localhost:514
     ```

   - Set the user in the ATNA server where issues will be logged. Modify the `AtnaSystemUserId` to be the ID of the system that is doing the logging, for example:

     ```
     ucmxds@example.com
     ```

If you configured WebCenter Content with different names for the XDS user, the XDS profile, or the profile trigger as described in Configuring WebCenter Content, then you must change the following variables in the web.xml file:

- `UcmUser`: the user ID that XDS uses to connect to the content server. In the previous section, this was designated as `xdsUser`.
- `UcmXdsProfileValue`: the profile value used when XDS content is checked in to the content server. For example, `xdsDoc`.

- `UcmXdsProfileField`: the core metadata field used to store the profile trigger. For example, `xIdcProfile` or `xProfileTrigger`.
- `UcmCgiURL`: the HttpCgiURL parameter for the UCM server. For example, `http://my.server.name:16200/cs/example`.
- `TracingEnabled`: set to true if you want verbose tracing sent to UCM. This is useful when testing the deployment, but should be set to false during production. See Monitoring the Component for details.

Deploying the Application

After configuring the WAR file, deploy the application. This documentation assumes that you are familiar with the process for deploying applications on Oracle WebLogic Server. For complete documentation about application deployment, see *Deploying Applications to Oracle WebLogic Server*.

1. Log in as an administrator and start the WebLogic Admin Server.
2. Click **Deployments** in the left navigation area.
3. If **ucmxds** is already installed, select it and click **Delete**.
4. Click **Install**.
5. Click **Upload Your Files**.
6. When prompted for the Deployment Archive, browse to the ucmxds.war file and click **Next**.
7. Click **Next** again.
8. Click **Install This Deployment as an application** and click **Next**.
9. Select the WebCenter Content server to deploy the application to and click **Next**.
10. Accept the defaults and click **Finish**.
11. Verify that the application is running using the method used at your site for verification.

Enabling and Using the WebCenter Content XDS Connector

The WebCenter Content XDS Connector has no user interface per se. It is a SOAP web service endpoint for the services that are used, and a Java Server Page (JSP) that provides the URLs to the WSDLs. An introductory overview web page is available which lists information about the current configuration and provides a summary of setup instructions.

To integrate with the WebCenter Content XDS Connector, you need an XDS-compatible agent. It must point to the SOAP endpoints in the WebCenter Content XDS Connector. There is one endpoint for an XDS.b compatible repository and one for an XDS-I.b compatible repository, as in this example:

```
http://example.com:13200/ucmxds/DocumentRepositoryB
http://example.com:13200/ucmxds/ImagingDocumentSource
```

These locations will vary depending on the how the WAR file was deployed. After deployment, the root home page has links to these endpoints.

Monitoring the Component

To enable tracing:

1. Log in to the content server as an administrator.
2. Choose **Administration**, then **System Audit Information**.
3. Select **xds** and choose the type of tracing, either standard or verbose.

Tracing will only occur if the **TracingEnabled** flag is set to true when you deploy the UCMXDS application. See Configuring the WebCenter Content XDS Connector WAR File for details.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Configuring the XDS Connector for Oracle WebCenter Content, 11*g* Release 1 (11.1.1)
E35898-01