opentext™

OpenText™ Documentum™ Content Management

XML Store Administration Guide

Store and process XML data in the repository.

EDCCFMTXML250400-AGD-EN-01

OpenText™ Documentum™ Content Management
XML Store Administration Guide
EDCCFMTXML250400-AGD-EN-01
Rev.: 2025-Nov-18

This documentation has been created for OpenText™ Documentum™ Content Management CE 25.4.
It is also valid for subsequent software releases unless OpenText has made newer documentation available with the product, on an OpenText website, or by any other means.

Open Text Corporation

275 Frank Tompa Drive, Waterloo, Ontario, Canada, N2L 0A1

Tel: +1-519-888-7111
Toll Free Canada/USA: 1-800-499-6544 International: +800-4996-5440
Fax: +1-519-888-0677
Support: https://support.opentext.com
For more information, visit https://www.opentext.com

Table of Contents

Chapter 1

Overview

OpenText Documentum Content Management (CM) XML Store is a module of OpenText™ Documentum™ Content Management Server that gives OpenText Documentum Content Management (CM) Server extended capabilities to store and process XML data in the repository by integrating it with BaseX, a highly scalable, native XML database.

XML Store adds standards-based XQuery to the XML capabilities of Documentum CM Server. XML Store enables richer searches and reusing and composing of content in a more flexible manner.

XML Store offers fine-grained access to any content fragment without requiring that content to be chunked or burst into individual content objects. This functionality helps users to conduct searches throughout the enterprise and achieve more flexible reuse and content composition.

XML Store preserves XML content as is, without mapping XML to RDBMS table rows and columns. The XML structure is preserved allowing users to efficiently and accurately query content at any level of detail. For example, individual elements, attributes, content objects, or metadata attributes, even on large information sets. As a native XML repository, XML Store provides performance advantages over relational databases and file systems through specialized XML indexing methods, caching, and architecture optimized for XML.

XML Store optimizes performance for XML content files and handles access to XML content using XQuery. The XML Store works with all Documentum CM Server features such as versioning, security, and lifecycles, including XML applications.

XML Store provides the following features:

- XML-specific storage type for storing XML content.
- XQuery interface for searching and retrieving XML content and Documentum attributes through XQuery syntax.
- XML Store administrator interface to improve search performance by creating XML indexes and controlling the generation of segments and file or segment mappings.

1.1 XML Store-enabled repository

XML Store adds a native XML database server to the Documentum CM Server repository. When you deploy XML Store for a Documentum CM Server, a BaseX instance is deployed, which is a container of one or more BaseX databases. A Documentum CM Server instance is associated with the BaseX database that maps to a repository in the Documentum CM Server. When you enable XML Store for a Documentum CM Server repository, a corresponding BaseX database is created.

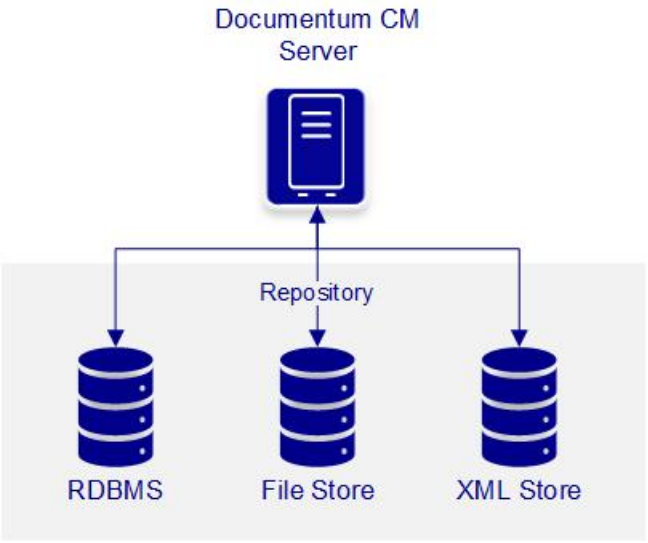

Figure 1-1: XML Store-enabled repository

1.2 Requirements

To administer XML Store on Documentum CM Server, you need the following:

- Administrative privileges on the computer where XML Store is deployed
- Working knowledge of Microsoft Windows or Linux
- Working knowledge of Documentum CM Server administration and configuration, including high-availability (HA) configurations

Chapter 2

Administering XML Store

2.1 Storing XML content in XML Store

XML documents are not saved in XML Store by default but must be configured with an assignment policy. Assignment policies are available with Content Storage Services enabled on the repository.

You can create and manage assignment policies using Documentum Administrator.

To configure assignment policies:

1. Open Documentum Administrator and sign in to the repository.
2. Click **Administration > Storage Management > Assignment Policies**.
3. Create new policies or modify existing policies as described in *Storage Management* chapter of the *OpenText Documentum Content Management - Server Administration and Configuration Guide (EDCCS-AGD)*.

2.2 Creating additional XML storage areas

After an initial XML Store is installed with Documentum CM Server, additional XML storage areas can be configured using Documentum Administrator.

To add an XML storage area:

1. Open Documentum Administrator and connect to the repository where you want to create a new XML storage area.
2. Go to **Administration > Storage Management > Storage**.
3. Click **File > New > External Free Store**.
4. Provide the following information on the **New External Free Store - Info** page:

File store property	Description
Name	The name of the new XML storage area.
Execute Plug-In	• Specify whether the plug-in is executed on the server or client. • Choose **xmlstore_win32shrlib_externalstorage** for your platform.
Storage Class	Click **XML** to create an XML storage area.

File store property	Description
Application server URL	Enter the URL of the xDB application server. For example: `http://myhost:9080/XMLStoreService/servlet/XMLStoreServiceServlet/`
Store Location	Enter `C:\Program Files (x86)\BaseX\data\` (if BaseX in local machine) or `\\<hostname or IP>\C$\Program Files (x86)\BaseX\data\` (it is a shared path, if BaseX is installed in a remote machine) as the file path of the new XML storage area in the BaseX installed host machine.

5. Click **OK**.

2.3 BaseX administration tool

XML Store deployment also includes BaseX Graphical User Interface (GUI) and command line client, an administration tool that lets developers and administrators manage BaseX databases through a graphical user interface.

For more information about GUI and command line client, see *BaseX documentation*.

2.4 Migrating from an xDB (embedded or external) to an external BaseX XML Store

If you have already deployed XML Store using an xDB, you can still migrate content to an external BaseX XML Store using `MIGRATE_CONTENT`. This is used only for repositories that are already XML Store enabled, upgraded to Documentum CM Server 7.0 or later, and accessible to multiple Documentum CM Server nodes.

To migrate XML Store from an xDB to an external BaseX:

1. Make sure that the external BaseX is properly installed and configured.
2. Make sure that Documentum CM Server and repositories are upgraded to version 7.0 or later and XML Store is enabled. For information about upgrading Documentum CM Server and repositories, see *OpenText Documentum Content Management - On-Premises Upgrade and Migration Guide (EDCCS-UMD)*.
3. Encrypt the xDB superuser password using the following command:

```
dm_encrypt_password -file <path to empty file where password needs to be stored>
-encrypt <xDB superuser password> -keyname <aek key for repository>
```

For example:

```
dm_encypt_password -file C:\documetum\xhive_storage\
<name of repository>_xdbpassword.txt -encrypt admin -keyname CSaek
```

4. Reinitiate the xDB plug-in, update the xhive store object, and run the content migration as follows:

   a. Reinitiate the xDB plug-in using the following command format.

   ```
   retrieve,c,dm_plugin where object_name='xhive_win32shrlib_externalstorage'
   checkout,c,l
   setfile,c,l,C:\Documentum\product\<product_release_version>\bin
   \xstoreplugin.dll,win32shrlib
   checkin,c,l
   ```

   The output of the command obtains the new r_object_id of the plug-in object.

   b. Update the xhive store object with both the plug-in ID and password file location. For example:

   ```
   retrieve,c,dm_extern_free where name='testenv_xhivestore'
   set,c,l,a_plugin_id[0]
   <new plug-in r_object_id>
   set,c,l,a_storage_param_name[2]
   password_location
   set,c,l,a_storage_param_value[2]
   <path to the newly created xhive password file>
   save,c,l
   ```

   c. Make sure that the BaseX server is up and running and both xHiveconnector and XMLStoreService applications are deployed in Java Method Server (JMS) successfully.

   Note: Make sure that both the servlets are up and running.

   d. Run the content migration using MIGRATE_CONTENT. For example:

   ```
   apply,c,NULL,MIGRATE_CONTENT,SOURCE_STORE,S,testenv_xhivestore,TARGET_STORE,S,
   testenv_xmlstore,LOG_FILE,S,C:\XB_migr_log.log,REMOVE_ORIGINAL,B,F
   ```

   For more information, see *OpenText Documentum Content Management - Server DQL Reference Guide (EDCCS-DRD)*.

5. After you run MIGRATE_CONTENT, the content in xDB is successfully migrated to BaseX and XML Store is configured to use the external BaseX. Then, restart JMS.

2.5 Managing BaseX passwords

The BaseX security model is based on two levels: user and administrator.

2.5.1 Enabling Vault for a repository

You must provide the BaseX administrator or the superuser password in the Vault server before you configure the repository. Provide the Vault secret password for the XML Store administrator in the format, `XML_STORE_PASSWORD/<repository_name>`. For example, `XML_Store_PASSWORD/testrepo`.

2.5.2 BaseX administrator or superuser account and password

A BaseX has one administrator or superuser account with `admin` as user name. The administrator or superuser account is used for the initial and default BaseX installation process and enables initial database configuration. The administrator or superuser can create and delete databases and perform administrative operations such as changing settings and all other operations on databases. The superuser can access regular BaseX data.

When you enable XML Store for the first Documentum CM Server repository, you must set a BaseX administrator or superuser password. For an external BaseX, you must set the administrator or superuser password when installing the BaseX instance and provide this information when configuring Documentum CM Server.

The BaseX administrator or superuser password is used to create the BaseX database for the initial repository, and for any other Documentum CM Server repository for which you enable XML Store. After the initial XML Store deployment, you will have one BaseX database for each repository. After you set the BaseX administrator or superuser password for the first time, the Documentum CM Server configuration program does not ask for the password again when you enable XML Store for subsequent repositories of the Documentum CM Server.

The *BaseX documentation* provides information about managing users and the `ALTER PASSWORD` syntax. For example, to change password of the `admin` user to *<actual password>*, use the following command:

```
> ALTER PASSWORD admin <actual password>
```

Note: Local permissions overwrite global permissions. A `test` user is allowed only to access (read and write) database starting with the letters `unit`. If no local permissions are set, the global rights are inherited.

2.6 Configuring Tomcat HTTP connector for large XML file imports and exports

If you want to import and export large XML files in and out of XML Store, you must configure the Tomcat HTTP connector to allow for more HTTP requests in an open connection. Otherwise, large file imports and exports may fail because Tomcat may close the connections before the files get transferred completely.

To configure Tomcat HTTP Connector for large XML file imports and exports, perform one of the following steps:

- Open the `web.xml` file located at `$DM_JMS_HOME/webapps/manager/WEB-INF/` in an XML editor and then increase the maximum upload file size and the maximum request size as follows:

```
<multipart-config>
  <!-- 1000MB max -->
  <max-file-size>1048576000</max-file-size>
  <max-request-size>1048576000</max-request-size>
  <file-size-threshold>0</file-size-threshold>
</multipart-config>
```

- Go to `$TOMCAT_HOME/conf/`, open the `server.xml` file, and modify your connector as follows:

```
<Connector port="9080" protocol="HTTP/1.1" connectionTimeout="20000"
redirectPort="8443" maxSwallowSize = "-1"/>
```

2.7 Additional BaseX administrative tasks

You can perform additional administrative tasks on the BaseX Server and databases independent of Documentum CM Server administration.

The *BaseX documentation* provides instructions on the following administration tasks:

- Managing Database Server
- Managing Index Module
- Working with XQueries

2.8 Limitations

When you upload, retrieve, delete, or perform any other operations using BaseX, it is not possible to open the `basexgui.bat` file or any other `.bat` file of a client simultaneously.

2.9 Sample xmlstore.properties file

The `xmlstore.properties` file in a JMS setup is for managing important information related to the XML database and external BaseX deployments. This file is located in the `XMLStoreProperties.ear` file.

Example 2-1: xmlstore.properties for an external BaseX configuration

The Documentum Installer updates the *`bootstrapFileName`* parameter value to `REMOTE` instructing the *`XMLStoreService`* to use the *`xmldbhostaddress`* and *`xmldbhostport`* parameters to connect to the external BaseX. After the utility runs, the `xmlstore.properties` file appears as follows:

```
extServiceClassName=com.xmlstore.service.BaseXXmlStoreService
xmldbhostaddress=<host name of BaseX>
xmldbhostport=1984
securityHintSize=200
processBatchSize=100
prettyPrintFormat=true
bootstrapFileName=REMOTE
Content-Server-Version=<release_version>
externalStoreServletClassName=com.xmlstore.controller.ExternalStoreServlet
url-pattern=/servlet/XhiveConnectorServlet
```