

EMC Documentum Repository Services for Microsoft SharePoint

A Detailed Review

Abstract

This white paper reviews the functionality and primary components of EMC[®] Documentum[®] Repository Services for Microsoft SharePoint, including its implementation of the Microsoft API for external BLOB storage. It discusses how Repository Services can address the issue of rapid SharePoint data growth by enabling IT administrators to make more efficient use of SQL Server resources by storing and managing BLOB content in Documentum.

November 2009

Copyright © 2009 EMC Corporation. All rights reserved.

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED “AS IS.” EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date listing of EMC product names, see EMC Corporation Trademarks on EMC.com

All other trademarks used herein are the property of their respective owners.

Part Number h4757

Table of Contents

Executive summary	4
Introduction	4
Audience	4
EMC Documentum Repository Services for Microsoft SharePoint overview.....	5
Functional components	6
Repository Services database	6
External BLOB Storage Handler	7
Performance cache	7
Temporary file cache	7
Services	7
Journal Management Engine	7
Clean-Up Collection Service	8
Data management	8
Object deletion	8
Static folder replication.....	8
Static property mapping	9
Basic mapping	9
Advanced mapping.....	10
Adding new metadata.....	11
Conclusion	12

Executive summary

Microsoft SharePoint is everywhere—and its rate of adoption shows no signs of decline. Virtually all industry research on Microsoft SharePoint notes three things: many large organizations deploy it; they have no plans to stop; and users like it. SharePoint gets good “word-of-mouth.” Yet this popularity comes with a price. Many organizations have thousands of SharePoint sites containing enormous volumes of content.

And there are, of course, SharePoint detractors. And even Microsoft, displaying rare candor, admits the SharePoint platform is not without its weaknesses. But SharePoint is easy to deploy and use, and it adequately fills a critical information management and collaboration niche.

The SharePoint SQL database not only stores metadata but content as well—as binary large objects (BLOBs). Yet as SharePoint content grows, it is a much more efficient use of SQL Server resources to limit their role to metadata storage only. Storing metadata in SQL Server ensures that SharePoint retains ownership of the content, which is especially important if there are workflows or business processes attached to SharePoint content.

To support external data stores, Microsoft released an Application Programming Interface (API) called EBS for external BLOB storage. EBS is a low-level API that intercepts the reads and writes directed at the SQL Content Server and dictates whether the data is stored in the database or is redirected to an external file share. The API is included in the Service Pack 1 release for Microsoft Office SharePoint Server (MOSS) 2007 and Microsoft Windows SharePoint Server (WSS) 3.0.

For EMC® Documentum® Repository Services for Microsoft SharePoint, Documentum created an implementation of the EBS API that includes a custom rule set. Repository Services redirects BLOB content to Documentum where it can be managed with robust content services, while BLOB metadata remains in the SQL database. Moreover, Repository Services operates behind the scenes—completely transparent to the SharePoint user. The SharePoint interface that users find so comfortable remains unchanged. More conventional solutions often tamper with the way SharePoint works, frequently breaking Office integrations, workflow, full-text indexing, custom applications, and so forth. But with Repository Services, custom applications and all SharePoint functions continue to work, which means no acceptance issues for the SharePoint community inside an organization.

Repository Services enables organizations to extend the value of their investment in Microsoft Office SharePoint Server and simultaneously gain the security and scalability of the Documentum platform. Beyond storing SharePoint content outside of SQL servers, Repository Services enables enterprise SharePoint customers to:

- Aggregate SharePoint content in their repositories of record
- Centrally manage that content and apply common security and retention policies
- Leverage advanced enterprise content management (ECM) features such as business process management
- Employ deduplication and hierarchical storage management

Introduction

This white paper provides an overview of EMC Documentum Repository Services and examines its primary functional components.

Audience

This white paper is an overview of Repository Services and its functional capabilities intended for CIOs, developers, IT administrators, and line-of-business managers who would benefit from the integration of Microsoft SharePoint and the EMC Documentum platform.

EMC Documentum Repository Services for Microsoft SharePoint overview

Microsoft discusses the issue of BLOB storage on its developer network website. The company estimates that *“as much as 80 percent of data for an enterprise-scale deployment of Windows SharePoint Server consists of file-based data streams that are stored as BLOB data. However, maintaining large quantities of BLOB data in a Microsoft SQL Server database is a suboptimal use of SQL Server resources. You can achieve equal benefit at lower cost with equivalent efficiency by using an external data store to contain BLOB data.”*¹

As mentioned previously, Microsoft released the EBS API for external BLOB storage. EBS is a low-level API that intercepts the reads and writes directed at the SQL server and dictates whether the data is stored in the database or is redirected to an external file share. The API was included in the Service Pack 1 release for MOSS 2007 and Microsoft WSS 3.0.

For Repository Services, Documentum created an implementation of the EBS API that includes a custom rule set. This rule set permits BLOB content to be redirected to the Documentum repository. As noted, SharePoint still “owns” the content, but Documentum manages it in the background. A single Documentum repository can scale to accommodate billions of objects, accounting for hundreds of terabytes of data.

Note: Not all SharePoint content is stored as BLOB content.

Besides reducing the data management demands on SharePoint SQL servers, Repository Services supports the use of hierarchical storage management (HSM) for the redirected BLOB content. Because a SQL server supports a transactional system, it typically runs on high-performance hardware. From a storage perspective, there is no way of tiering that hardware, but once content is in Documentum it can be pushed to different levels of storage. Likewise content can be deduplicated. In terms of return on investment (ROI), the combination of HSM and deduplication can deliver significant savings. EMC estimates the cost differential between tier one and archive-level storage is in the range of \$50,000 per terabyte per year.

Repository Services also helps organizations address their compliance and information governance requirements more effectively. Many global organizations have chosen Documentum as their repository of record. They store and manage virtually all business-critical content assets with Documentum, whether it’s formal records—physical and electronic—long-term archives, NDA submissions, SOPs, and so forth. With EDRSS, SharePoint content can be subject to the same policies and controls.

Repository Services connects SharePoint to Documentum Content Server using Documentum Foundation Services (DFS). The product is managed in the SharePoint central administration console. Figure 1 illustrates the way Repository Services handles content and metadata in a typical write operation.

¹ "External Storing of Binary Large Objects (BLOBs) in Windows SharePoint Services." 2009. <http://msdn.microsoft.com/en-us/library/bb802976.aspx> (accessed September 25, 2009).

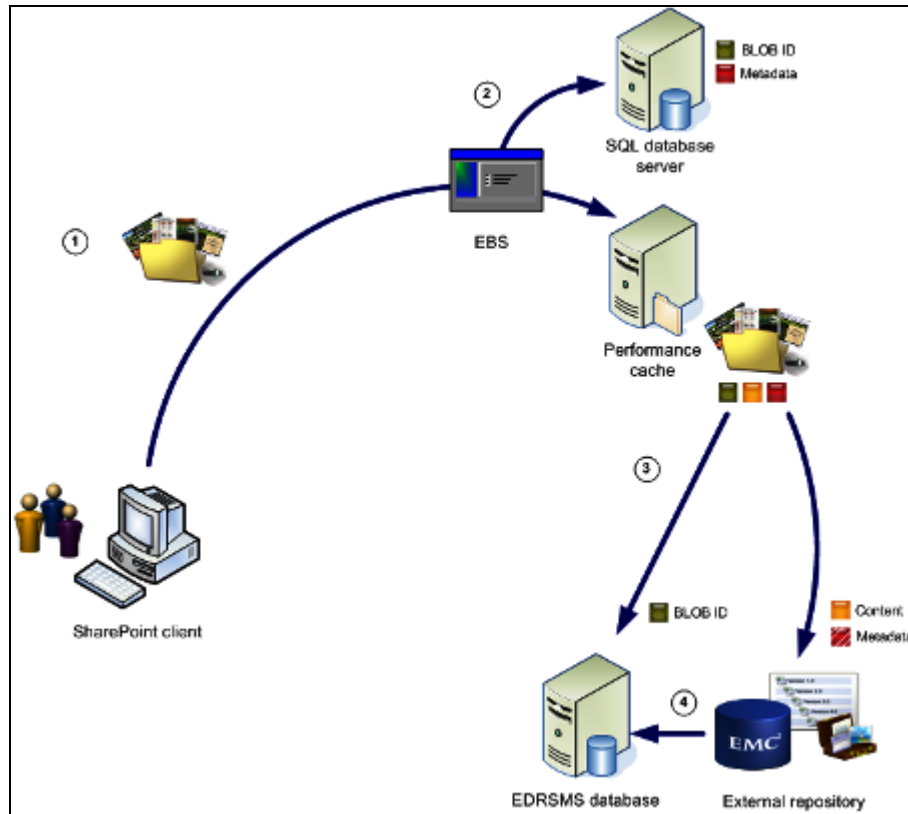


Figure 1. Content flow in the Repository Services write operation

- **Step 1:** The SharePoint user creates BLOB content in SharePoint.
- **Step 2:** EBS intercepts the write operation and redirects the content to the performance cache based on Repository Services rules. The BLOB identifier (ID) is stored in the SQL Server database and the BLOB metadata is stored in SQL Server as usual.
- **Step 3:** The External BLOB Storage Handler (EBSH) writes the routing information into the Repository Services database. Content may be sent from the performance cache to DCTM via DFS. A convenience copy of the metadata in XML is attached as a rendition to the object now in DCTM.
- **Step 4:** The Repository Services routing table is updated with the new content location

Functional components

There are five core components of Repository Services: the Repository Services database, external BLOB Storage Handler (EBSH), performance cache, temporary cache, and a services component, which includes the Journal Management Engine (JME) and the Clean-Up Collection Service.

Repository Services database

The Repository Services database contains all of the BLOB redirection and journaling rules that are implemented via SharePoint Central Administration along with event logs and routing information. The configuration settings define which BLOBs require redirection from the performance cache. The database routing table allows content to be retrieved.

When content is journaled, Repository Services saves the content file to the Documentum repository and attaches a copy of the metadata as a rendition of the object. Repository Services records the SharePoint-generated BLOB ID in a routing table along with a Documentum Object ID.

External BLOB Storage Handler

EBSH is the EMC Documentum implementation of Microsoft's API. EBSH intercepts the reads and writes of content created in SharePoint and moves or retrieves the data according to the redirection rules.

The first stop for all redirected site collection content is the performance cache. Microsoft SharePoint can externalize content based only on site collection and content size within a site collection. So, at this level, Repository Services can dictate that only objects from a particular site collection or above a certain size are externalized and objects below that threshold stay in the SQL Content Server database.

Once in the performance cache, redirection to Documentum is handled by Repository Services where a selective rule set is employed. Repository Services can redirect (journal) all site collection content or portions that exist at lower levels such as at the site and list level. An entire site collection, one or more sites, and one or more lists can be configured for journaling as long as the parent (the site collection) has been configured for redirection.

All of the EBSH actions are not only transparent to the user but to the SharePoint application as well. This application transparency is important because many other approaches to externalizing SharePoint content create shortcuts that may "break" SharePoint Office integrations, workflow, full-text indexing, or some other SharePoint functions, including customizations and third-party add-ons. But with Repository Services and its EBSH implementation, custom applications and all SharePoint functions continue to work, which means no acceptance issues for the SharePoint community inside an organization

Performance cache

The EBSH allows content to be redirected to an external, shared file directory, which EMC terms the performance cache. Redirection to the performance cache is a synchronous operation. Data is queued in the performance cache and moved from there to the Documentum repository. A performance cache can be created for each site collection or all site collections can be redirected to a single performance cache.

There are two configuration options for the performance cache. In the first, once the object has been moved to Documentum, it is deleted from the cache. In the second, the original object is moved to Documentum and a copy remains in the cache. This option uses more disk space but speeds object retrieval when SharePoint end users execute a search.

Temporary file cache

There is only one temporary file cache per instance of Repository Services. The temporary file cache is used for retrieving content from Documentum when there is no copy in the performance cache. Subsequent requests for the same content are read from the temporary cache, which speeds access. Nevertheless, by default, objects are removed from this cache every 30 minutes although this time limit is configurable.

Services

Two services are installed on every Web front-end (WFE) server, which are Microsoft IIS servers, the Journal Management Engine (JME) and the Clean-Up Collection Service (CCS).

Journal Management Engine

Directed by EBSH rules, JME retrieves the content that's in the performance cache and sends it to Documentum. It creates an XML file of the BLOB metadata in SharePoint and directs that to the Documentum repository as well. JME also deletes objects that are out of date or marks them as deleted, depending on the configuration.

Clean-Up Collection Service

CCS flags objects that are out of date. This could be the result of completely deleting an object—a hard delete—which means it’s no longer in any of the recycle bins. It cannot be accessed via the SharePoint user interface so it is considered an “orphaned” object. Orphaned objects can be deleted from the repository by the JME or marked as deleted, with a date that identifies when it was deleted within SharePoint. This is very useful for reporting purposes.

Data management

Object deletion

When a user deletes SharePoint content, it enters a user recycle bin. This is a soft delete. When deleted from the user’s recycle bin, it enters the site administrator’s recycle bin. Once deleted from there—a hard delete—the content is orphaned and no longer referenced by SharePoint. When Repository Services detects a SharePoint hard delete, it can reach into the Documentum repository and delete the object or keep the object but mark it as deleted. A marked object in Documentum is completely owned by Documentum and can be freely manipulated as it is no longer referenced in SharePoint. This is a configuration option in the connector configuration called “Documentum Object Marking.” Many customers use Documentum Object Marking to support long-term archiving.

Static folder replication

Static folder replication re-creates the SharePoint folder structure when content is directed to Documentum. Static folder replication can be enabled or disabled, but if it is disabled there will be no folder structure created in Documentum. There are two static folder replication options:

- Relative creates a “friendly” file path that is not necessarily unique
- Full assures a unique file path

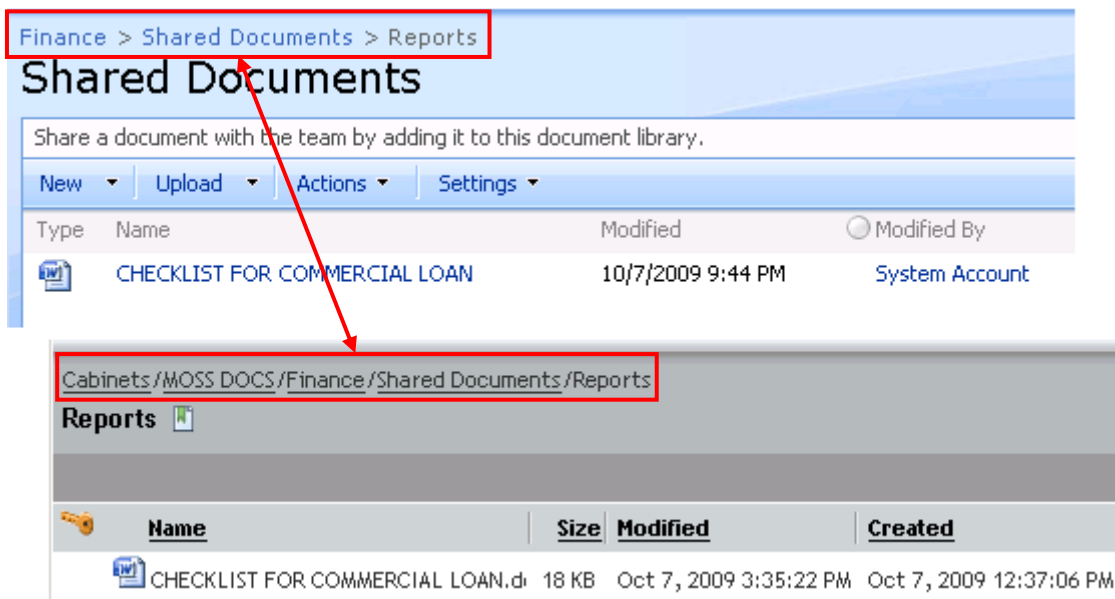


Figure 2. Relative folder replication. At top is SharePoint, the bottom is Documentum

In the folder structure above, Finance, Shared Documents, and Reports are SharePoint folders replicated in Documentum. This folder path is not necessarily unique since it was created at the folder level using the relative option. Relative folder replication can be used to aggregate SharePoint content in a common location in Documentum.

Data can be partitioned separately in Documentum or it can be merged into a common location with or without folder replication. This allows common policy management for aggregated data and allows data to be segregated and placed under separate policy management in one centralized system. Regardless of partitioning in Documentum, the SharePoint user's view of the content remains unchanged.

To keep the content in separate folders in Documentum, full replication creates unique paths by replicating the folder structure from the SharePoint root level. In full replication, each folder has a unique site identifier, which ensures that folders only contain content from one SharePoint site.

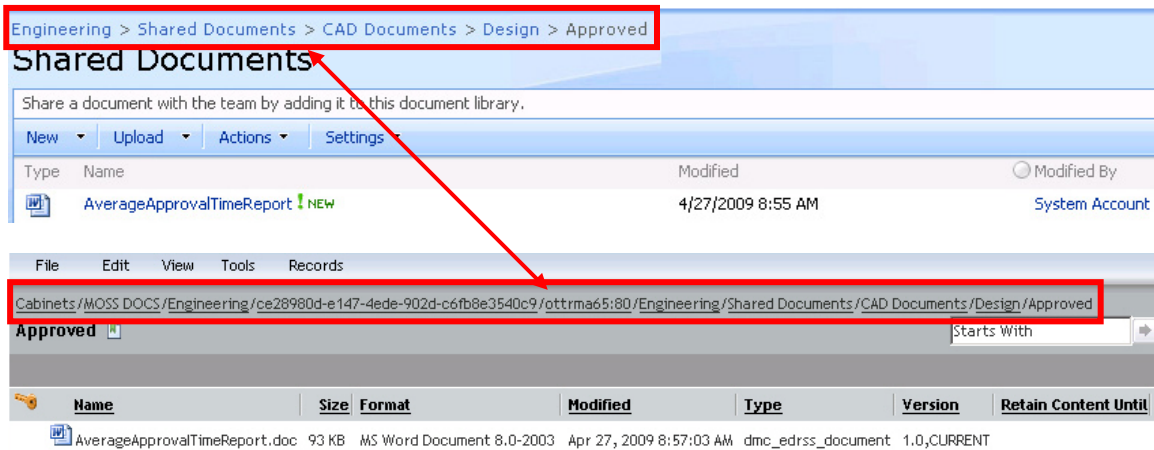


Figure 3. Full folder replication

Static property mapping

Repository Services captures SharePoint metadata so that it can be used in Documentum. It always captures a convenience copy of all of the SharePoint metadata in an XML file that is stored as a rendition of the object created in DCTM. Repository Services can also map values from this file to the properties of the DCTM object. It offers two models for mapping SharePoint metadata to Documentum object properties.

Basic mapping

Basic mapping uses the standard properties for a Documentum object type and maps Name, Title, Subject, and Keywords. Basic mapping uses the “dm_document” type or subtype. Figure 4 shows a basic mapping screen. Note the BLOB ID in the keyword field. If an object is unreferenced in SharePoint because it has been hard deleted and DCTM Object Marking is enabled, the keyword field will also include the date and time of the deletion event.

Name: *

Title:

Subject:

Keywords: [Edit](#) blobid:87ff776f-b35f-11de-9a52-0050568a082e, sp_deleted:10/7/2009 3:35:22 PM

Authors: [Edit](#) DCTM

Full Content Size: 18195

Owner Name: [Edit](#) qaadmin

Version Label: [Edit](#) 1.0, CURRENT

Checkout Date:

Checked Out By:

Lifecycle ID: [Select](#)

Current State:

▶ **Show More**

Show all properties

Figure 4. Basic metadata mapping

Advanced mapping

To map additional attributes to an object, Documentum also provides a custom object type: “dmc_edrss_document.” Advanced mapping adds seven fields to the four basic mapping fields: Created at, Last Modified, Created By, Modified By, Location in MOSS, Comments, and Version. Figure 5 displays the advanced mapping screen.

MOSS BLOB ID: 87ff776f-b35f-11de-9a52-0050568a082e
MOSS Comments:
MOSS Created By: System Account
MOSS Created Date: Oct 7, 2009 4:36:01 PM
MOSS Deleted Status: T
MOSS Deleted Status Date: Oct 7, 2009 3:35:22 PM
MOSS Keywords:
MOSS Last Modified Date: Oct 7, 2009 4:36:19 PM
MOSS Location: /sites/Finance/Shared Documents/Reports/CHECKLIST FOR COMMERCIAL LOAN.docx
MOSS Modified By: SHAREPOINT\system
MOSS Version Number: 1.0
Name: <input type="text" value="CHECKLIST FOR COMMERCIAL LOAN.docx"/> *
Object ID: 090001bc80018d16

Figure 5. Advanced metadata mapping

Adding new metadata

It's easy for users to create new metadata for SharePoint BLOBs. In order to capture metadata that's not automatically mapped as part of the Repository Services object type, Documentum crawls the SharePoint object model and harvests all the metadata and saves this as an XML rendition of the journaled object.

Conclusion

EMC Documentum Repository Services for Microsoft SharePoint redirects content to Documentum where it can be managed with robust content services, while the metadata remains in the SQL Server database. Moreover, Repository Services operates behind the scenes, with no impact to the SharePoint user, who can continue to access and affect content as though it was being stored natively in a SharePoint repository. Moreover, custom applications, existing content workflows, and all SharePoint functions continue to work as usual.

For IT administrators and the organizations they serve, Repository Services extends the value of SharePoint deployments while it contributes to a unified infrastructure that facilitates robust information governance and regulatory compliance. The Documentum platform acts as centralized point of control for SharePoint Teamsites and content. Documents such as contracts, new drug applications, standard operating procedures, and other business-critical content can be stored and managed in Documentum while SharePoint provides the means for universal access, versioning, and the collaborative exchange that these process-intensive content types require.

Beyond storing SharePoint content outside of SQL servers, administrators can also use Repository Services to:

- Aggregate SharePoint content and Teamsites within repositories of record
- Centrally manage content and apply common security and retention policies
- Leverage advanced enterprise content management (ECM) features such as business process management
- Reduce data storage costs through deduplication and tiered storage management

To learn more about EMC solutions for Microsoft SharePoint integration, visit <http://www.EMC.com> or call **800.607.9546** (outside the U.S.: +1.925.600.5802).