

Realtime
publishers

The Shortcut Guide[™] To



Centralized SharePoint Administration

sponsored by

 **AvePoint[®]**
Unleashing the Power of SharePoint[™]

Wendy Henry

Chapter 4: SharePoint Archiving and Reporting Best Practices	61
Establishing Realistic Archiving and Reporting Goals	62
Best Practices for Archiving SharePoint Data	63
Best Practices for User Activity Reporting in SharePoint	64
Considerations for Auditing and Reporting	66
Limitations of MOSS Records Center and MOSS Report Center	67
MOSS Records Center	67
MOSS Report Center	70
Complexity of SQL Server Reporting Services Design and SharePoint	73
Why NOT to Directly Query SharePoint Databases	75
SharePoint Database Schemas are NOT Intuitive	75
Microsoft Best Practices Encourage Programmatic Access to SharePoint Databases... ..	77
You Break It, You Bought It	78
Archiving and Reporting: The Strengths Are in Third-Party Solutions	78
Archiving Advantages	78
Reporting Advantages	79
Supplementing Native Tools	79
Summary	80

Copyright Statement

© 2009 Realtime Publishers. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtime Publishers (the “Materials”) and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtime Publishers its web site sponsors. In no event shall Realtime Publishers or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtime Publishers and the Realtime Publishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtime Publishers, please contact us via e-mail at info@realtimepublishers.com.

[**Editor's Note:** This eBook was downloaded from Realtime Nexus—The Digital Library for IT Professionals. All leading technology eBooks and guides from Realtime Publishers can be found at <http://nexus.realtimepublishers.com>.]

Chapter 4: SharePoint Archiving and Reporting Best Practices

The previous chapters of this book have explored the hurdles of distributed administration and dispersed localizations in large SharePoint enterprises. One of the remaining challenges that all SharePoint Administrators face over time is data life cycle management. With the passage of hours, data that has been entered into SharePoint will inevitably be accessed and likely more than once. Some data may have a life span or “shelf life” after which it is not required to be readily available if available at all. Other data may require an audit trail of some sort to identify who has been accessing it, when, and for what purpose. Deciding how long to hold onto information in SharePoint and determining what exactly needs to be reported about that information during its residence can keep an experienced administrator awake at night.

Microsoft assists in such endeavors by offering a series of best practices articles for Microsoft Office SharePoint Server 2007 that are helpful when implementing native MOSS 2007 solutions for archiving and reporting. Alas, no such solutions are available in Windows SharePoint Services 3.0 environments without custom programming or third-party utilities, so there are no best practice suggestions on archiving and reporting for the free implementation of SharePoint. However, SharePoint uses Microsoft SQL Server 2005 or 2008 as its repository, so there are a myriad of archiving and reporting strategies that could be executed directly against the SharePoint content databases without involving SharePoint per se, opening up an entirely separate can of best practice worms for SQL Server.

Resource

For more information about SharePoint best practices, visit the new Best Practices Resource Center for SharePoint Server 2007 at Microsoft TechNet at <http://technet.microsoft.com/office/sharepointserver/bb736746.aspx>.

This chapter will explore the art of determining reasonable data retention and reporting goals as well as Microsoft suggestions for implementing native solutions using MOSS 2007. We will then take a look at the limitations of the native MOSS 2007 archiving and reporting tools as well as the difficulty of defining SQL Server Reporting Services to report on SharePoint data. The chapter will also outline reasons for fighting the urge to directly query SharePoint databases in SQL Server as a means of archiving or reporting. Lastly, we will discuss the advantages of Microsoft Partner and third-party solutions that have already tamed the SharePoint databases and have done most of the archive and report design work for you.

Establishing Realistic Archiving and Reporting Goals

Most successful archiving and reporting implementations are based on a foundation of *realistic* business goals. Not all data needs to remain available into the distant future. In fact, the art of determining exactly which data needs to be kept and for how long can make the difference between an attainable retention policy and unmanageable data bloat. Furthermore, data that requires archiving should be examined for summarization potential to make efficient use of system resources. Some of the factors that influence data life spans are outlined in Table 4.1.

Factor	Influence
Regulatory Compliance	Taxonomy-classed data must be retained per regulatory requirement
Industry Standard Compliance	Specific data retention for no less than the industry standard to remain competitive
Performance Analysis	Historical analysis of company performance
Market Analysis	Trend analysis of company's market demographic
Legal	Retain what is known or should be known to be relevant to legal proceedings (required by law)
Renewable Documents	Contracts or other property/legal documents that are revised or renewed at specific timeframes

Table 4.1: Possible factors that affect retention policy planning.

Similarly, every conceivable facet of every piece of data in SharePoint does not need to be captured in a report. Realistic reporting goals should focus on the purpose of the reported information and what value it brings to the business decision process. Reporting simply for the sake of revealing is unnecessary utilization of system resources. For example, say a Sales Manager must make the difficult decision to discontinue a product. Pulling a report that reveals the popularity of product X in different market segments would prove much more helpful in his decision than a report that details the monthly sales figures of each of his salespeople.

In the end, organizations must develop archiving and reporting policies that are:

- Practical and designed for their particular endeavors
- Flexible enough to accommodate special circumstances such as legal proceedings or stockholder inquiries
- Enforceable and easily implementable by trained users
- Reviewed at periodic intervals to be updated as needed

Resource

There are entire Web sites and consulting firms that assist companies with establishing archiving and reporting policies. The process of writing such policies fall beyond the scope of this book. However, Microsoft has a team dedicated to Records Management and several helpful tips can be gained from their blog at <http://blogs.msdn.com/recman>.

Best Practices for Archiving SharePoint Data

Archiving data, any data, should be conducted in a manner that best fits a company's overall business goals. Because every company is different, there is no *single set* of best practice guidelines for an archive strategy. However, there are some fundamentals that should be honored in any archiving endeavor regardless of platform; these basics apply to SharePoint data as well. First and foremost, a quick vocabulary lesson on the components of a well thought-out archive plan:

- **Archive Policy**—A documented policy that outlines retention and access rules and the business policies that justify them, identifies responsible parties and required systems, and prescribes archive validation and delivery mechanisms
- **Retention Rules**—Includes a retention schedule for archiving data based on criteria (SharePoint calls this an expiration policy), routing rules for archiving expired data to alternative location or storage, and any approval or notification rules desired
- **Access Rules**—Includes archive permission structure, modification allowances, metadata management, and search exposure
- **Validation and Delivery**—Systems designed to validate integrity of archived data and deliver archives upon request

One of the initial best practices for archiving SharePoint data is to develop an enforceable archive policy that accommodates the business rules for SharePoint information. For instance, say the Finance team has a document library on their team site in SharePoint. And say out of the 100 document items found in that library, 20 of the documents have not been modified in more than 2 years. Now imagine that one of the business rules about Finance data states any Finance document that has not been modified in at least 2 years should be moved to an archive library. Poof! Those 20 documents should be marked for archive and become candidates for expiration policy processing according to an archive policy that honors the business rule.

But that example is a bit broad in that the expiration policy was designed around an entire library in Finance. Depending on the data taxonomy classifications used to construct your logical SharePoint design, enforcing the same expiration policy across an entire library may not be realistic. Table 4.2 provides additional best practices for archiving SharePoint data.

Practices	Advantages
Separate expiration policies	Different data can be archived at different intervals
Assign retention rules to content types instead of document libraries	Allows different archiving criteria for different items within the same library (if a single list supports multiple content types); enforces a common archive policy for all items of a specific content type across multiple libraries
Plan separate archiving system for static Web pages, list items, and static Web parts	MOSS 2007 Records Center only accepts documents, so alternative archiving solutions must be sought for other business information stored in SharePoint such as wiki pages, blogs, discussion threads, list items, and static Web parts and pages
Automate archiving when possible	Manual archiving burdens users and is vulnerable to human errors and omissions
Standardize access rules	Defining standard archive access permissions and methods reduces unauthorized access

Table 4.2: Advantages of SharePoint archiving best practices.

Lastly, one of the final best practices for any SharePoint archive policy is to establish a formal review procedure. Like any organic system, SharePoint content and its use will grow and change over time. The archive policy must reflect current business requirements and provide appropriate retention for the manner in which users are accessing the data today, not months or years ago. By defining a date schedule (perhaps quarterly or semi-annually) for reviewing the archive policy, the SharePoint administrators can be given ample time to adjust the environment and implement new archiving needs.

Best Practices for User Activity Reporting in SharePoint

There are many types of reporting that can be exercised in a SharePoint environment. Some reports pull numbers from Excel spreadsheets to describe a business process or direction. Other reports retrieve metadata about specific items in SharePoint to classify content or find errant items. But of all the types of reporting possible in a SharePoint enterprise, the most commonly demanded reports are those exposing end user activity. Perhaps to satisfy audit demands, perhaps to proactively plan burgeoning system resource requirements, whatever the reason, user activity reporting in SharePoint is a hot topic.

Both Windows SharePoint Services 3.0 and Microsoft Office SharePoint Server 2007 have a native usage analysis process that can write end user activity to a temporary file daily then process that temporary file to a more permanent and reader-friendly log file. These log files can be located in a directory of the administrator's choice and will record up to 31-days worth of activity per Web application for up to 31 months. In fact, MOSS 2007 offers additional usage analysis configuration in its Shared Service Provider to allow SSP administrators to include Search behavior in the audit. But the processing of usage analysis is global in nature, on or off (see Figure 4.1), and therefore can be very resource intensive on the Web front-end servers in your SharePoint farm. Before engaging usage analysis to report on user activity natively within SharePoint, you should investigate the systems cost.

Resource

For more information about setting up Usage Analysis in SharePoint, see the Microsoft Support article at <http://support.microsoft.com/kb/825541>.

Figure 4.1: Usage Analysis Processing setup page from Central Administration—Operations.

Best practices concerning end user activity reporting include:

- Design a plan—Document the standard configuration settings for usage analysis on the farm and identify parties responsible for reading usage reports (see Figure 4.2) at both the individual site and site collection levels
- Archive reports (only when necessary)—The logs begin overwriting the oldest log upon reaching the “Number of log files to create” setting, so an archive policy will be needed to maintain logs more than a month old
- Do not duplicate audit effort—If you’re using Internet Information Services to record site visit information, do not duplicate efforts in SharePoint unnecessarily

- Schedule Usage Analysis Processing—Limit processing to begin and end within non-peak hours for SharePoint and SQL Server resources
- Monitor the SharePoint Timer Service—This service is responsible for executing the usage analysis processing job on schedule

Site usage reports	Site collection usage reports
<ul style="list-style-type: none"> ▪ Site usage summary ▪ Requests ▪ Users ▪ Referrers ▪ Destination pages ▪ Home page 	<ul style="list-style-type: none"> ▪ Site collection usage summary ▪ Requests ▪ Users ▪ Referrers ▪ Destination pages ▪ Home page ▪ Search queries ▪ Search results

Figure 4.2: Native usage reports in MOSS 2007.

Considerations for Auditing and Reporting

One of the first considerations for SharePoint auditing and reporting is an IT industry rule of thumb: Too much of a good thing isn't good. The purpose of SharePoint auditing should not be omnipotence over user behavior but rather the satisfaction of specific proof of use requests, such as those demanded by regulatory compliance, litigation, or unauthorized use prevention. And the purpose of SharePoint reporting should not be to micro-measure every facet of SharePoint usage but rather to highlight areas for improvement. Be reasonable when identifying audit and report requirements.

A second consideration for reporting on SharePoint data is whether to employ detailed or summarized reports. Summarizing information requires less storage space and can alleviate hardware burdens. On the flip side, summarizing report data prevents *drill-down* opportunities for users seeking more details on why the summarized values are what they are. If summarization is not an option, consider compression tools that can squeeze detailed audit data onto less storage space or tools that can transfer audit and report data onto alternative near-online or offline facilities.

Note

Neither WSSv3 nor MOSS 2007 offers near-online or offline storage options for archive or audit data. A third-party utility is required to engage in performance optimizing retention and reporting strategies.

Unfortunately, the native tools in SharePoint for archiving and reporting on data are quite limited in configuration and storage options. The WSSv3 product alone offers no archive solution and the report capabilities are limited to SharePoint Designer 2007 on-board reports. MOSS 2007 offers both an archive and a reporting solution but the products are inflexible and based on inherent SQL Server products requiring additional administration and maintenance. If you are keeping mission-critical business information in a SharePoint environment, you should look into Microsoft Partner and third-party solutions for archiving and reporting in SharePoint.

Limitations of MOSS Records Center and MOSS Report Center

When it comes to archiving SharePoint information, Microsoft Office SharePoint Server 2007 includes a records center site definition designed to import document library documents and encapsulate them into sealed records bearing unique record metadata according to defined routing rules. In actuality, the Records Center is a fully functioning Web service incorporated into MOSS 2007 to help administrators adhere to document retention policies without overloading the production SharePoint sites. Another site definition included in MOSS 2007 called Report Center is designed to offer report and Key Performance Indicator (KPI) data viewing in a centralized display of summarized decision-making information. But while *freely included* is always delightful, the limitations of these two native products in MOSS 2007 could knock them out of the running as true enterprise-class solutions.

MOSS Records Center

On the upside, the MOSS Records Center is relatively easy to set up and quite friendly to end users who want to manually archive a chosen document in SharePoint. Automatic archiving can be accomplished through the use of workflows associated with either an entire document library or a specific content type and the Web service itself requires only a little configuration, which can be accomplished in the Central Administration GUI. As out of the box archiving solutions go, it isn't bad. The challenges come in defining routing rules that can be restricted to only a particular subset of qualifying content, not all, or in defining rules beyond the canned column expiration criteria allowed on the configuration pages. Another downside to the MOSS Records Center is that it archives documents only and does so dependent upon the SharePoint index, which is often managed according to navigation needs rather than archiving needs. For example, confidential columns or columns that are not usually searched for by users will tend to be excluded from the index to keep it lean, which also precludes routing rules from using those columns as expiration criteria.

Because the MOSS Records Center is supported by a separate Web service, a SharePoint farm that wants to submit documents to the Records Center must be configured to connect to the target Records Center service by URL (namely the URL of the MOSS site created using the Records Center site template). This setting can be made via the External Service Connections links on the Application Management page of Central Administration (see Figure 4.3). Unfortunately, the single value URL field limits a SharePoint farm to using only a single target Records Center for all document archiving from all Web applications in the farm. There is no opportunity to workload balance document records across multiple Records Centers. And unless the target Records Center allows anonymous access, each Web application's application pool identity will need to be authenticated by the external Records Center service, which requires domain user accounts as the application pool security context if the Records Center is on a MOSS server in a different ADS domain or forest. Though these are concerns of the SharePoint administrator that are mostly transparent to business users, they ultimately affect the environment in which the users archive their data.

Central Administration > Application Management > Configure Connection to Records Center

Configure Connection to Records Center

Records Center Connection

To connect to a Records Center, enter the URL and a display name for a Records Center server. Unless the Records Center is configured to allow records to be anonymously submitted, you must configure each Web application to use a domain user account.

Do not connect to a Records Center

Connect to a Records Center

URL:

Example:

http://server/portal/_vti_bin/officialfile.aspx

Display name:

OK Cancel

Figure 4.3: Configure external service connection to Records Center from Central Administration.

The MOSS 2007 Records Center automatically builds the lists and libraries in Table 4.3 to support the archiving of documents from submitting SharePoint sites. Each of the lists and libraries serve a specific purpose in the archive process, but additional lists and libraries can be added as well—the Records Center is a MOSS site after all. Alas, though you could build additional record routing lists using the custom list template, you cannot duplicate route associations to inbound documents that have previously been defined. Thus, there is no inherent way to define more than one route for a given document to achieve conditional archiving without the overhead of authoring complex workflows.

Libraries & Lists	Purpose
Hold Reports	Library of hold reports (one per hold) that list all of the records involved
Missing Properties	Library of record-specific items that define missing metadata values filled in by users during document submission
Records Pending Submission	Temporary location for documents submitted that were missing required metadata
Unclassified Records	Catch-all library that stores records that did not get handled by any of the defined routes because they didn't meet any of the routing rules
Holds	List of Hold definitions that can later be associated with a given record to suspend routing
Links	Default links list providing custom hyperlinks
Record Routing	List of routing definitions that define criteria and route to destination library for any records meeting that criteria
Records Center	Default announcements list that has been renamed for the site
Submitted E-mail Records	Temporary storage of submitted email records
Tasks	Task list used by routing workflows by default

Table 4.3: Default libraries and lists generated in the MOSS Records Center site definition.

Finally, perhaps the most limiting attribute of the MOSS Records Center is the simplistic routing instructions used by the Record Routing list items (see Figure 4.4). Creating a new route involves only specifying a destination location for the record and filling in a multi-value list of “aliases” that, if found among any metadata value or body text of the submitted document, will cause this particular routing item to catch the document and place it into the configured location. There is no opportunity for advanced metadata tagging or conditional routing! If you need to archive more than just documents (such as Web part content and static pages, list items, and custom asp.net content) and you need to do so by routing each item according to complex criteria, look to a Microsoft Partner third-party solution.

Corporate Portal > Records > Record Routing > New Item

Record Routing: New Item

OK Cancel

Attach File | Spelling... * indicates a required field

Title *	<input type="text"/>
Description	<input type="text"/>
Location *	<input type="text"/> The title of the library where records matching this record routing item should be stored. Libraries used to store submitted records cannot be deleted.
Aliases	<input type="text"/> A '/' delimited list of alternative names that represent this record routing entry.
Default	<input type="checkbox"/> If checked, this routing item will be used for submitted records that do not match the title or aliases of any other record routing item.

OK Cancel

Figure 4.4: MOSS Records Center Record Route creation page.

MOSS Report Center

One of the out of the box enterprise features in MOSS 2007 Enterprise Edition is the Report Center. The Report Center site is designed to be a central location for report viewing and business intelligence. In fact, the Collaboration Portal site template in MOSS automatically builds a *Reports* first-tier child site as part of the portal template when it is used to construct the top-level site of a new site collection. The Report Center site definition constructs various lists and libraries and includes a sample business intelligence (BI) dashboard to illustrate the very *gooey* (GUI) nature of the site's capabilities. Information workers and managers tend to appreciate the one-stop centralized nature of the MOSS Report Center and the colorful graphic display of summarized data that can be filtered and expanded to show additional details (see Figure 4.5).

But the challenges of creating truly useful dashboards lie in the complexity of the supporting sub-structure. Namely, the KPI and KPI Details lists, Excel workbooks, and the Excel Calculation Services (ECS) that ship only with MOSS 2007 Enterprise Edition—none of which are designed to query the SharePoint content databases to report on SharePoint itself but rather are intended to report on data values stored in Excel spreadsheets, specific SharePoint list items, SQL Server 2005 or 2008 Analysis Services cubes, or values manually entered into a KPI list.

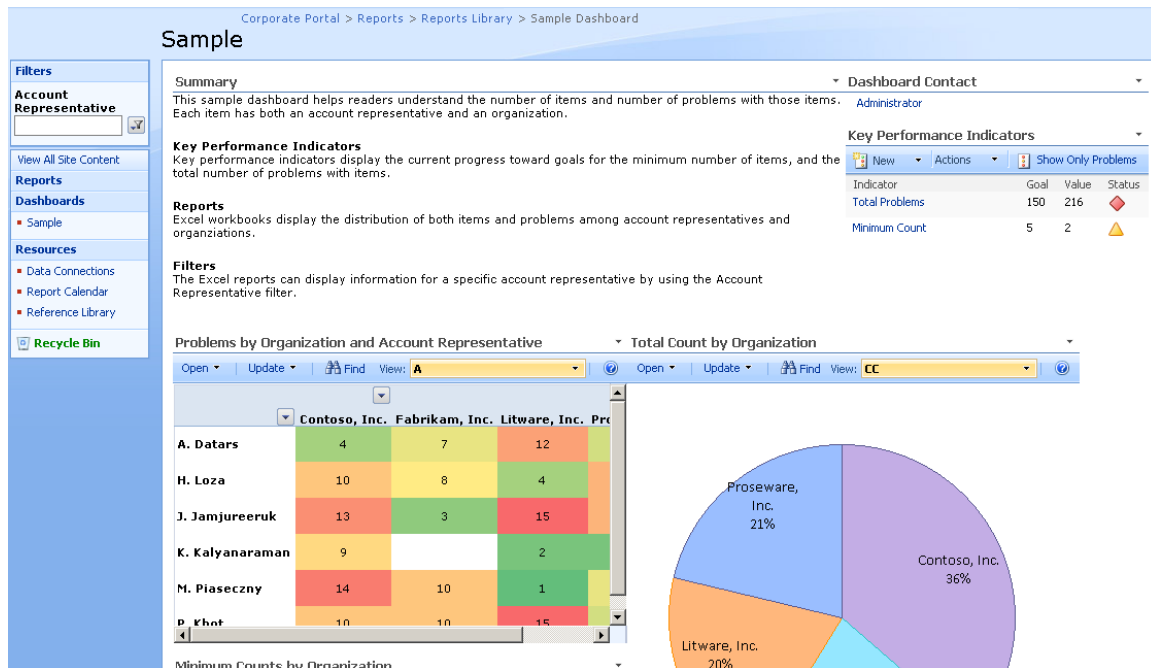


Figure 4.5: Sample dashboard in MOSS Report Center.

Well, what's so bad about that? Wait, things get more complicated than just page design. Additional concerns include:

- **Site Template**—The Report Center is provided by the Enterprise Feature in MOSS 2007 Enterprise Edition (not available in MOSS 2007 Standard Edition) and is built from a site template that, much like the Publishing Site template, houses all site pages in a Pages library that has both Require Check Out and Document Version History (Major and Minor) enabled. This means every time a page is checked out, even if no edits take place, it generates a new minor (draft) version of the page requiring major version publishing before any users other than editors can see it thanks to the Draft Item Security setting on the library. Therefore, expect content approval and versioning overhead.
- **ECS**—If one of the data sources for the reports is going to be Excel workbooks, ECS is required to extract cell values, perform functions, and display graphics (such as charts and graphs) that have been embedded into the Excel workbook. ECS ships only with MOSS 2007 Enterprise Edition and delivers through a Shared Service Provider (remember, no SSP is created by default during MOSS installation). Only a Tier-2 Shared Services Administrator can define ECS, and before a workbook's data can be used in a report, the workbook must be uploaded to one of the named Trusted File Locations (SharePoint, File System, or HTTP) in the ECS configuration (see Figure 4.6). Each Trusted File Location is configured with unique session timeouts, size limits, cache times, and refresh intervals, which thwarts centralized administration of all locations and can confuse report readers when different report content refreshes at different rates.

Excel Services Settings

- ▣ Edit Excel Services settings
- ▣ Trusted file locations
- ▣ Trusted data connection libraries
- ▣ Trusted data providers
- ▣ User-defined function assemblies

Excel Services
Add Trusted File Location

OK Cancel

Location
A Windows SharePoint Services location, network file share, or Web folder from which a server running Excel Services is permitted to access workbooks.

Address
The full Windows SharePoint Services location, network file share or Web folder address of this trusted location.

Location Type
Storage type of this trusted location:

Windows SharePoint Services
 UNC
 HTTP

Trust Children
Trust child libraries or directories.
 Children trusted

Description
The optional description of the purpose of this trusted location.

▣ Session Management
 ▣ Workbook Properties
 ▣ Calculation Behavior
 ▣ External Data
 ▣ User-Defined Functions

Figure 4.6: ECS administration in Shared Service Provider Administration site.

- SQL Server Reporting Services (SSRS)—Delivering SSRS reports through SharePoint requires add-on Web parts and SSRS expertise. More on this in the next section.
- No inherent reports—All reports in the Report Center have to be designed, there are no predefined reports for content or SharePoint statistics.
- Multi-Lingual Support—Using SharePoint Language Packs to alter the language of the site page does not transcribe the exposed report content as well. Delivering report data itself (Excel workbook values, graph keys, and so on) in an alternative language requires building the reports from data stored in the alternative language.

Resource

For more information about how to create and manage all things SharePoint, visit Microsoft's SharePoint 2007 Solution Center at <http://support.microsoft.com/ph/11373>.

Provided that you are still sold on using MOSS 2007 Report Center despite the administration overhead, there is one more thing to consider. In a MOSS 2007 Enterprise Edition installation with the Enterprise Feature enabled, the Web parts and KPI lists that support report views are available on any page in any site, not just the Report Center site. In a highly distributed administration model where many information workers are granted design and contribute privileges across many sites, dashboards and reporting Web parts may get added to various sites in the site collections. There is no centralized method of managing report Web parts across multiple sites in MOSS 2007, so adherence to corporate standards and acceptable use will need to be enforced.

Complexity of SQL Server Reporting Services Design and SharePoint

One last option for displaying BI information in SharePoint is to expose formal reports created in SQL Server Reporting Services. Though this book is not meant to be a primer for the SQL Server product, it would be remiss not to discuss SSRS given Microsoft's intent to bring SSRS and SharePoint closer (case in point: SQL Server Reporting Services' SharePoint-Integrated installation mode). The art of creating SSRS reports could and has filled entire technical volumes, so we won't cover the steps here. Rather, we'll simply point out some of the challenges you may face when displaying SSRS reports in SharePoint.

First, let's take a look at the limitations of exposing reports in SharePoint when SSRS has been installed in Native Mode. This means SSRS is running independently on a SQL Server, not in SharePoint-integrated mode. In native mode, SQL Server 2005 and SQL Server 2008 Reporting Services maintains a Web site called Report Manager for viewing and conducting lightweight administration of reports. There is no inherent synchronization between SSRS/Report Manager and SharePoint. In this environment, you have four options for exposing external SSRS reports on your SharePoint pages:

- Provide a link to the URL of SSRS Report Manager or to the individual report (no parameter values allowed in report URL) by adding an item to a Links list or Summary Links Web part on the SharePoint page
- Engage the MOSS Page Viewer Web part and provide the URL of SSRS Report Manager or to the individual report (again, no parameter values allowed in the URL)
- Retrieve the 32-bit SSRS native Web parts installed on the SQL Server Reporting Services server during SSRS installation (located by default in the 32-bit Program Files directory at %Program Files%\Microsoft SQL Server\100\Tools\Reporting Services\SharePoint\RSWebParts.cab) and perform an STSADM operation to add the Web part package to SharePoint (the operation is addwppack). Then the native SSRS report viewer Web part can be added to SharePoint pages.
- Write custom Web parts to expose SSRS reports and add functionality such as parameter value input, drill-through reporting, multi-page report pagination, and more. Requires .NET programming expertise.

Unfortunately, none of the aforementioned options creates any umbilical relationship between SharePoint and SSRS. In fact, the strategies are essentially glorified redirectors, which means duplicate administration efforts in both the SharePoint and the SSRS systems to keep reports updated and the information fresh.

Now let's peruse SSRS installed in SharePoint-Integrated mode. This installation option was first introduced in SP2 for SQL Server 2005 Reporting Services and continues to be available in the 2008 version of the product. SharePoint-Integrated installation mode results in using SharePoint to manage and deliver SSRS reports instead of a separate Report Manager Web site in IIS. In fact, installing SSRS in this mode is required to use the Microsoft SQL Server Reporting Services Add-In for SharePoint Technologies from Microsoft. If you download the add-in, here's what you get:

- Report Viewer Web Part (similar in appearance to the native report viewer Web part that installs with SSRS in native mode but built differently) exposes SSRS reports in their entirety within a Web part on a SharePoint Web page
- Report Server content types, management pages, delivery mechanisms to SharePoint libraries, and an extension to Reporting Services Configuration Manager to set up SharePoint-Integrated mode for SSRS

Resource

To download the Microsoft SSRS Add-In for SharePoint Technologies, visit <http://www.microsoft.com/downloads/details.aspx?FamilyID=1e53f882-0c16-4847-b331-132274ae8c84>.

Some considerations before incorporating SSRS reports into SharePoint include:

- ADS SPN—If SSRS is running on a separate server from SharePoint (non-integrated mode) in a Kerberos realm, an ADS engineer will need to set a SPN to facilitate delegation
- No non-SSRS Report support—Both the native report viewer Web part in SSRS and the SSRS Add-In Report Viewer Web part are designed to display only reports created in SQL Server Reporting Services and not reports created using an alternative reporting software such as Crystal Reports by SAP-Business Objects
- Loss of inherent SSRS functionality—For example, it is possible to create linked reports in SSRS that spawn a second report from a hyperlink in the first report; it is also possible for the second report in such a scenario to expect input parameters and for those parameter values to be forwarded to the second report by the first report upon spawning; however, in the MOSS SSRS report viewer Web part, the passing of parameter values functionality is lost when multiple connected Web parts are used to display the second report in a second Web part

Though SSRS is able to gather data from a myriad of possible data sources, including SQL Server databases, it can be a difficult solution for reporting values from SharePoint databases due to their complex schema. If your company is following a strategy of housing all mission-critical data itself in SharePoint lists and libraries, then designing reports from SharePoint data in SSRS can prove to be a challenge—at least more of a challenge than designing reports that pull from clearer, more structured data sources.

Why NOT to Directly Query SharePoint Databases

One of the drawbacks of the native BI tools in MOSS 2007 Enterprise Edition such as KPI lists and Web parts is that while they are useful for designing dashboards that expose data values stored both externally and inside SharePoint, they cannot satisfy additional administrative reporting needs because they do not include any features for analyzing the SharePoint environment itself (such as usage or modification auditing). Microsoft touts the integration of SSRS as a solution, but the downside to using SSRS for retrieving metadata about the SharePoint configuration and content databases is the complexity of the databases themselves.

SharePoint Database Schemas are NOT Intuitive

SharePoint databases serve separate purposes in supporting the SharePoint environment (see Table 4.4). Therefore, querying SQL Server to retrieve, say, audit information about new Web applications including their content would require querying multiple tables in multiple databases. Also, configuration settings, metadata, and data values are spread across multiple tables within a given SharePoint database, requiring complex join statements to produce a single relative result set.

Database	Default Name	Stores
Farm Configuration	SharePoint_Config (1 per farm)	Farm/server settings for sites, databases, servers, and Web applications
Central Administration content database	SharePoint_AdminContent_{GUID} (1 per farm)	Central Admin sites, pages, and Web parts configurations (data values input via Central Admin Web pages are stored in the SharePoint_Config database)
Shared Services Provider Configuration	SharedServices_DB (1 per SSP)	SSP settings including user profiles, audiences, files, site usage, BDC, Excel and Forms services

Database	Default Name	Stores
Shared Services Provider Content	SSP_Content (1 per SSP)	SSP Administration sites, pages, and Web parts configurations (data values input are stored in the SharedServices_DB database)
Shared Services Provider Search	SharedServices_Search_DB (1 per SSP)	Metadata of crawled content, history and search logs, crawl statistics, and search configuration settings
WSS Search	WSS_Search (1 per farm)	Metadata of Web application content from content databases; used for search in WSS-only environment and used for Help searches in MOSS environment
Web Application Content	WSS_Content({GUID}) (1 per web application by default, can scale to multiple per web application)	Data for site collections, structure details, content, files, and security details (no custom schema allowed)

Table 4.4: MOSS databases.

Resource

For a complete reference on the SharePoint databases, check out the MSDN article at <http://msdn.microsoft.com/library/bb447514.aspx>.

Furthermore, all the SharePoint databases are significantly denormalized. For example, the SharePoint_Config database housing farm configuration values consists of various user-defined tables. However, there is not a 1:1 intuitive association between an object we humans manage (such as one particular server in the farm) and a given table. For example, say you wanted to retrieve information about a specific server in your SharePoint farm such as its name, what services its running, and the last time it communicated with another server in the farm. Although you might think that simply querying the dbo.Servers table in SharePoint_Config would be enough, it would only give you the server's name and its object GUID in SharePoint. To determine services details for that server, you would then need to query dbo.Services (with a join based on the GUID of the server object) and include other queries to dbo.Dependencies and dbo.TimerRunningJobs and other tables to round out your result set.

Similarly, the content database(s) of any given Web application in SharePoint is also comprised of distributed tables bearing cryptic names. For example, imagine you need to retrieve all document files created by a specific user. One would think a query to `dbo.AllDocs` using a WHERE clause to filter the results by creator would do the trick. Not so, as the document files are actually located in the `dbo.AllDocStreams` table in an image data-type column. Also, the author information about files is not maintained in a single column bearing a descriptive name such as *author* but rather by the GUID number of the people object that was added to SharePoint's security structure for the site in question to represent the given user who uploaded the document to SharePoint.

Resource

For a breakdown of Configuration and Content database tables, visit [http://msdn.microsoft.com/library/dd587562\(office.11\).aspx](http://msdn.microsoft.com/library/dd587562(office.11).aspx).

As you can see, using Transact-SQL to directly query or manipulate the SharePoint databases would require extensive investigation into the table design and database schema of each database prior to writing your T-SQL statement. Not that there is anything preventing you from doing so, but Microsoft recommends a better way: writing custom .NET code using the SharePoint object model in Visual Studio.

Microsoft Best Practices Encourage Programmatic Access to SharePoint Databases

Microsoft discourages directly querying or manipulating the SharePoint databases via SQL Server tools such as SQL Server Management Studio or the SQLCMD Command Line Interface (CLI) utility. Instead, programmatically accessing the SharePoint databases hides the complex database schema underneath a logically laid out SharePoint object model, making development more efficient than administrative queries. Also, Visual Studio offers more error handling, source control, and packaging options than a direct T-SQL query utility.

Resources

Visual Studio, programming, and the C#.NET language fall far beyond the scope of this book. However, if you would like more information about writing or troubleshooting programs for SharePoint, the following MSDN articles are a great place to start: "Programming in Windows SharePoint Services" at <http://msdn.microsoft.com/library/ms430674.aspx> and "Best Practices: Common Coding Issues When Using the SharePoint Object Model" at <http://msdn.microsoft.com/library/bb687949.aspx>.

Of course, there is a certain skill set required to write .NET code and the experience of using the Visual Studio application itself to consider. Expertise in the Visual C# and Visual Basic programming languages as well as familiarity with the SharePoint object model are mandatory. The SharePoint object model is vast (its diagram could fill an entire wall poster), so navigation experience in it would be helpful as well. Microsoft offers the Visual Studio application for program authoring, which includes a sophisticated GUI that may take some getting used to. Let's just say not every SharePoint administrator is going to be able to write custom SharePoint programs without the help of a skilled developer.

You Break It, You Bought It

This guide would be remiss if we didn't state the obvious: If you programmatically customize or directly alter the schemas of the SharePoint databases, don't expect a warm reception from Microsoft Product Support when your customizations don't work. If you break SharePoint, a likely support call may end in Microsoft recommending re-installation of SharePoint, thereby wiping the slate clean of all your hard work without satisfying the business need that initiated the custom design in the first place. Even worse, you may inadvertently violate your company's service agreement with Microsoft and endanger future customer support. If you stop to consider the millions (or more) of ways SharePoint could be altered, it's easy to see that it would be unreasonable to expect Microsoft to anticipate every conceivable change and have their support team trained for every inevitability. So beware and proceed with caution when directly or programmatically accessing the SharePoint databases.

Archiving and Reporting: The Strengths Are in Third-Party Solutions

In recent years, Microsoft has silently been opening a door for the software industry by shipping products to market leaving "room for improvement." The practice suits Microsoft's commitment to their aggressive release schedules and it creates a market gap for Microsoft Partner companies to fill with feature add-ins and third-party utilities. One could even say Microsoft is philanthropically supporting entrepreneurs by *not* including every conceivable function in the release of their products. SharePoint is no exception, and although MOSS ships with archiving and reporting components, we have already explored their many limitations. So let's take a look at some of the goodies to be found in third-party tools available for SharePoint.

Archiving Advantages

When it comes to archiving needs, there is virtually no limit to the demands that can be made by information workers according to business rules. And although you cannot expect any one single third-party tool to satisfy all possible archiving needs, common functionality unmet by SharePoint to look for in a solution candidate include:

- Flexible Destinations—Additional Web service destinations beyond a SharePoint Records Center serviced by an .asmx defined Web service
- Flexible Storage—The ability to store archived content somewhere other than SharePoint databases in SQL Server, such as near-online or offline file systems
- Multiple Content Types Support—Archiving more content types than just documents
- Advanced Metadata Management—Custom tags beyond SharePoint library columns that offer custom record metadata security, indexing and individual record tag metadata value management including locks, read-only attributes, update rules, and auditing

Flexible destinations and storage options allow you to stretch your hardware investment and avoid using costly SQL Server storage for infrequently accessed data. And as not all mission-critical data in SharePoint comes in the form of document files, employing a solution that can archive list items and Web part content will round out your record strategy. Advanced metadata management capabilities in a SharePoint archiving solution give you the freedom to define and populate custom tags beyond your SharePoint columns at record creation time. Thus, you could potentially automate both record and custom tag creation and drive them from the original content's metadata values, user actions, system events, or a combination of events.

Reporting Advantages

As SharePoint becomes more ingrained in your corporate network, the need to monitor and report on the system will become increasingly important. There are so few friendly and reliable usage analysis tools in SharePoint itself, so you may look to purchase a Microsoft Partner software solution to ensure regulatory compliance of structure and security as well as govern SharePoint's logical architecture. When weighing the choice of native versus externally purchased tools, consider the advantages many third-party reporting applications bring to SharePoint

- No need to write complex SSRS reports drawing directly from confusing SharePoint database schemas
- Reliable Timer Service Job maintenance
- Pre-written common usage and system reports
- Centralized SharePoint infrastructure reports
- Realtime platform and usage analysis details

The simplicity and ease with which a third-party utility installs and displays your reporting solution may quickly justify the cost when compared with writing your own. Many Microsoft Partners offer centralized SharePoint administration application suites that include reporting and monitoring tools for large enterprises or small farms. If your SharePoint environment has been up and running for awhile, look for a product that includes a discovery utility that can automatically analyze your SharePoint enterprise and report back on all physical servers and logical architecture (especially if there is no change control process or documentation requirement in place).

Supplementing Native Tools

Both Windows SharePoint Services v3.0 and MOSS 2007 ship out of the box with a myriad of administration tools. However, because these collaboration platforms can be implemented according to any company's business model and whim, it would be impossible for Microsoft to include every fathomable administration need in their utility design. The flexible nature of SharePoint is both its beauty and its curse.

It is true that SharePoint provides the ability to customize archiving, reporting, and administration through the use of custom content types, custom columns, workflows and programmatic access via the SharePoint object model. However, designing these solutions can prove time consuming and difficult. Why get overwhelmed when there are Microsoft Partners in the market who have already taken the guesswork out of SharePoint by offering reliable, pluggable solutions?

When shopping for SharePoint third-party solutions, be sure to find a reputable Microsoft Partner with expertise in your version of SharePoint. Not all programmers are SharePoint programmers and thus not all SharePoint third-party solutions are created equal. Be careful not to reinvent the wheel by purchasing a solution that is already in your SharePoint environment but rather opt for additional features that are either missing from the product or that would be costly to produce in-house. And when it comes to any Microsoft add-in, caveat emptor!

Summary

This chapter outlined processes for determining reasonable data retention and reporting goals and examined Microsoft suggestions for implementing native solutions using MOSS 2007. We exposed the limitations of the native MOSS 2007 archiving and reporting tools and the challenges of using SSRS to report on SharePoint data. We examined the pitfalls of directly querying SharePoint databases in SQL Server, and the advantages of Microsoft Partner and third-party solutions for archiving and reporting.

I hope that you have enjoyed the centralized administration of SharePoint topics introduced and detailed in this guide and can now proceed forward with a workable administration plan for your environment.

Download Additional eBooks from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this eBook to be informative, we encourage you to download more of our industry-leading technology eBooks and video guides at Realtime Nexus. Please visit <http://nexus.realtimepublishers.com>.