

Paper 390-2009**Integrating SAS[®] Business Intelligence with Microsoft SharePoint**

David R. Henderson and Sean Alexandre,
SAS Institute Inc., Cary, NC

Abstract

Because of the increased interest and deployment of Microsoft SharePoint, the need to integrate SAS[®] Business Intelligence technologies with SharePoint has also risen. This paper describes a variety of approaches to integrating the two technologies, such as using SharePoint document libraries to store content produced by SAS, including SAS output in SharePoint pages and extracting SharePoint list data into SAS for analysis. This paper also introduces the new SAS Web Parts for Microsoft SharePoint, which provide the ability to add SAS[®] Stored Processes and SAS[®] BI Dashboards to a SharePoint page

Introduction

Both SAS and SharePoint have their individual strengths. Although their respective functionalities overlap in a few areas, their true strengths do not. By integrating the two platforms, you can combine the capabilities of each platform to create an even more powerful experience for your users. This paper discusses a variety of approaches to integrating the two technologies and explains the integration uses for each approach. In this paper, we'll cover:

- writing SAS content directly to a SharePoint document library
- using the new SAS Web Parts for Microsoft SharePoint
- extracting SharePoint data into SAS
- adding SAS as a federated search provider in SharePoint

Writing SAS Content to SharePoint Document Libraries

The SAS Output Delivery System (ODS) can produce a variety of types of reports. Whether these reports are PDF documents, HTML pages or Microsoft Excel workbooks, storing these reports in SharePoint provides additional functionality that isn't available if those reports were stored on a shared drive or on a Web server. By adding these reports to SharePoint, you can take advantage of SharePoint RSS feed support, e-mail notifications and workflows. Once this content is in a SharePoint document library, you can take advantage of the permissions in SharePoint to secure the content.

Adding SAS reports to SharePoint works particularly well for reports that do not change frequently, take a long time to produce or need to be archived. Such reports cannot (or should not) be produced at the moment the viewer wants to see them. These reports must be cached somewhere, so a SharePoint document library is a good option. However, because the report is cached, it might not be completely up-to-date. It is only as current as the last time the SAS job was run. While this isn't an approach that produces real-time analytical reports, it fulfills many reporting requirements. The deciding factor on whether to store the report in SharePoint is how dynamic the back-end data is and how current the viewer expects the reports to be.

SharePoint document libraries are exposed as Universal Naming Convention (UNC) paths, so writing content directly into those libraries is as easy as writing to any network location. Writing to a SharePoint document library is simple if SAS is running on a Microsoft Windows operating system, because UNC is

native to that platform. If SAS is running on UNIX or some other operating system, you might need to configure Samba or some similar file access software to allow the UNIX operating system to access the UNC path on the SharePoint server. The following code demonstrates how to write ODS output to a SharePoint document library. You'll probably notice that this code looks no different than if you were writing the file to any other UNC path.

```
data sampledata;
  do n=1 to 1000;
    x=int(ranuni(77777)*7);
    y=int(ranuni(77777)*5);
    output;
  end;
run;

filename out '\\mySharePointServer\Shared Documents\ProcFreqExample.html';

ODS HTML encoding=UTF8 body=out dev=gif;

proc freq data=sampledata;
  tables height*weight / chisq;
run;

ODS HTML close;
```

Because writing to a SharePoint document library is nothing more than saving to a UNC location, it is achievable with SAS 9.1.3 (or even older) technology.

(Note: When writing to a SharePoint document library via a UNC path on Microsoft Windows Server, the Web Client Service must be enabled. This service allows you to access WebDav resources over HTTP. You can enable the Web Client Service via the Windows Services snap-in on the Microsoft Management Console.)

Using SAS BI Web Parts to Include Real-Time Content on a SharePoint Page

Another approach to integration is to pull content from SAS into a SharePoint page. When the site visitor opens a page in their browser, a dynamic report (that typically contains up-to-the-moment data) is generated. Caching possibilities aren't excluded, but caching is performed by SAS technologies rather than SharePoint. In this scenario, security occurs in multiple layers. SharePoint determines who is allowed to open a given site or page. The SAS server requires the viewer's credentials to retrieve the content. A viewer may have permission to open the SharePoint page but may be denied access to the SAS content.

Another advantage of this approach is that the viewer can customize the report content. SAS prompts the viewer for values before the content is produced. Because of this prompting, the viewer can select what content to include in the report. As a result, a sales manager in the East can look at a report of sales data for the East Region. Likewise, a sales rep in the South can look at a report of the sales data for the South region.

Placing SAS content on a SharePoint page can be accomplished by placing a Page Viewer Web Part on the page and configuring it to pull an HTML page from SAS/Intrnet[®]. (A *Web part* is a building block that allows you to place content and control the content's appearance on a SharePoint page.)

In addition to the standard Page Viewer Web Part that is available in SharePoint, SAS provides two custom Web parts, the SAS BI Dashboard Web Part and the SAS Stored Process Web Part that allow

you to pull SAS content into a SharePoint page. Both web parts can be configured to include their corresponding content types in a SharePoint page.

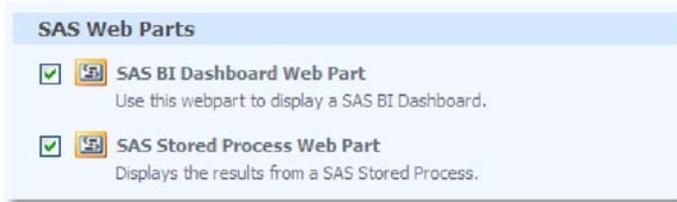


Figure 1. Once installed, the SAS BI Web Parts are listed in the Web Parts Gallery and can be added to any Web parts page.

The SAS BI Web Parts are configured via the SharePoint site's Web.config file which specifies a default connection to the SAS BI Server. Because these Web parts are pulling SAS content in real time, the viewer is seeing up-to-date reports and not reports that were created at some point in the past. This integration approach works well for content that be generated quickly.

The SAS BI Web Parts also handle authentication for the viewer. When a viewer opens a SharePoint page, that viewer is authenticated by SharePoint. (In Internet Explorer, this authentication typically occurs through Windows Integrated Authentication. In other browsers, the viewer is sometimes prompted for a user name and password.) Because the SharePoint site has verified the viewer's identity, the SAS BI Web Parts use that identity to authenticate with the SAS server.

SAS Stored Process Web Part

A SAS Stored Process is a SAS program that is registered in the SAS Metadata Server and can be executed by SAS applications and clients such as SAS Enterprise Guide, SAS Web Report Studio, and the SAS Add-In for Microsoft Office. The SAS Stored Process Web Part allows you to run a stored process and include the results in a Web parts page.

Once you have added the SAS Stored Process Web Part to a Web parts page, open the Tool Pane to configure the Web part. In the Tool Pane, the values in the **SAS Server** and **Port** fields are pulled from the Web.config file.

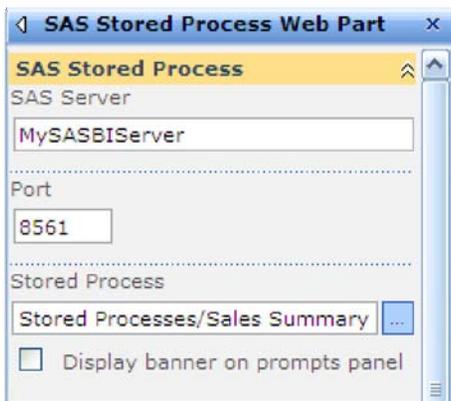


Figure 2. Configure the SAS Stored Process Web Part by opening its Tool Pane.

To configure the Web part, type the full path and name of the stored process in the **Stored Process** field, or click the blue browse button to open a dialog box that allows you to select the stored process from the SAS Folders on the specified SAS server.

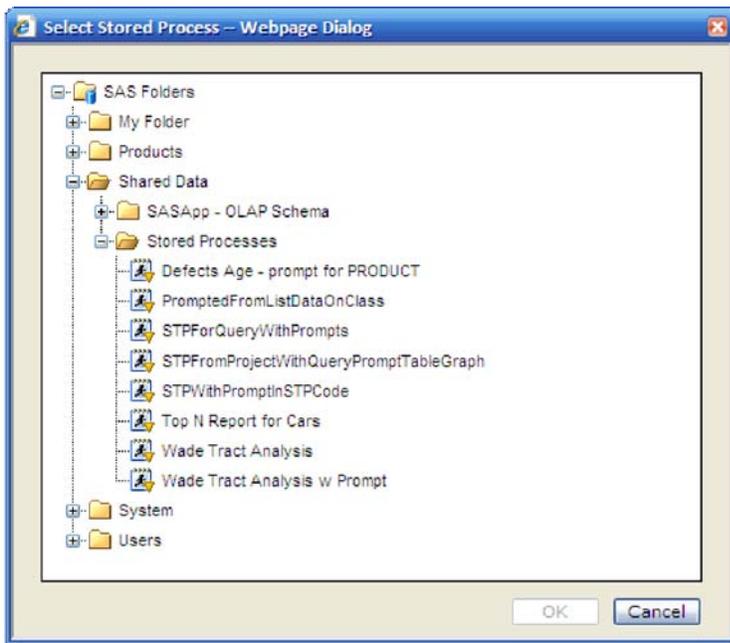


Figure 3. Select a stored process from the SAS Folders.

If the stored process includes prompts, the viewer of the SharePoint page sees those prompts before any content is generated. Once the viewer has provided values for the prompts and clicked **Run**, the results of the stored process are displayed. If the stored process doesn't include prompts, the stored process is run immediately and the viewer sees the results. The Web part is using the SAS Stored Process Web application to prompt the visitor, run the stored process and render the results. Therefore, it is capable of handling any stored process that is supported by the SAS Stored Process Web application. Supported stored processes include those with dynamic and cascading prompts and those that generate tables, images and other content.

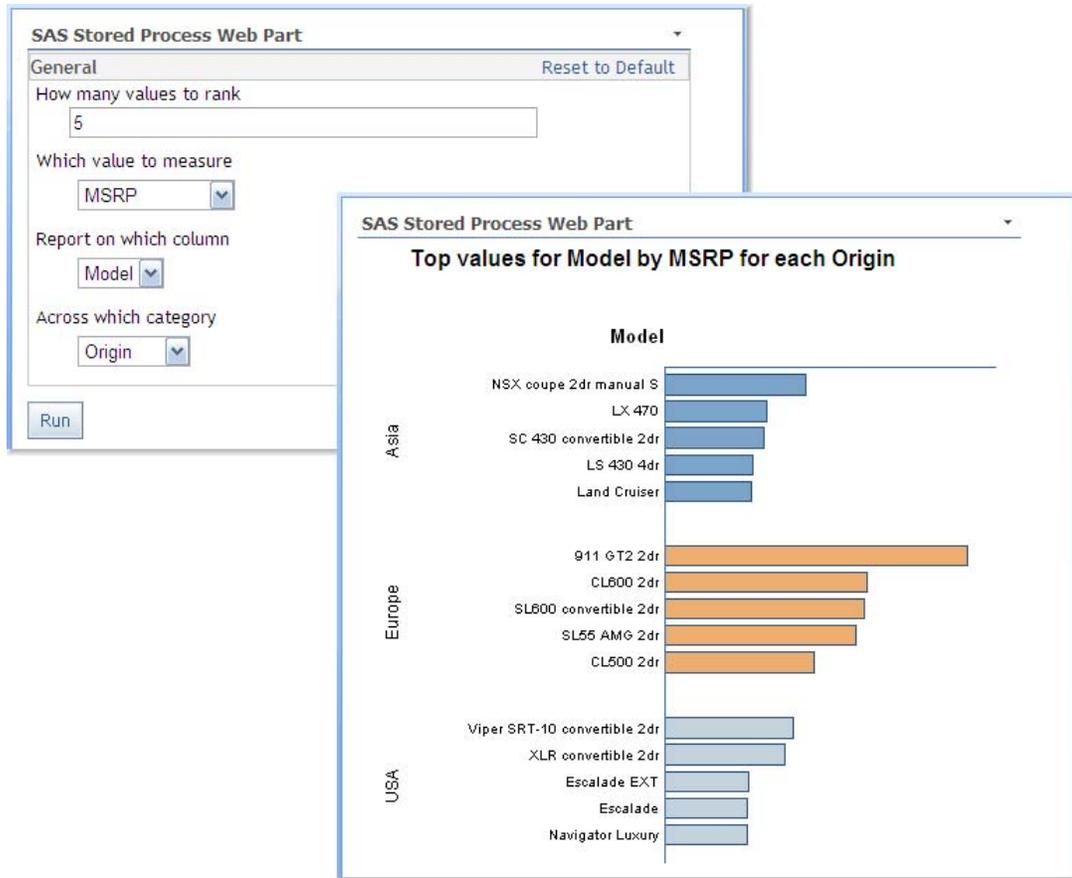


Figure 4. A sample stored process, showing first its prompts and then the results in the SAS Stored Process Web Part.

SAS BI Dashboard Web Part

Included in SAS® Enterprise BI Server is the SAS BI Dashboard Web application, which is a powerful tool to assist you in monitoring the health of your organization. The SAS BI Dashboard Web Part allows you to include dashboards in a SharePoint page. Like the SAS Stored Process Web Part, the values in the **SAS Server** and **Port** fields are pulled from the Web.config file. The **Dashboard** drop-down list on the Tool pane lists all of the dashboards that are available on the specified SAS server.

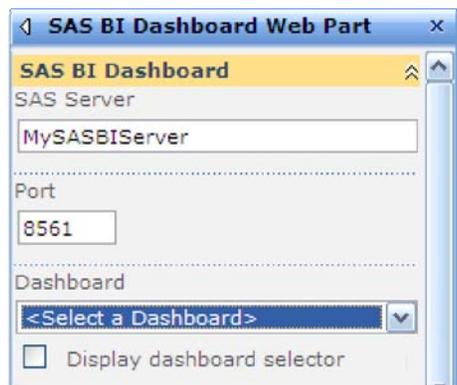


Figure 5. Configure the SAS BI Dashboard Web Part by opening its Tool Panel.

In addition to selecting a dashboard, you also have the option of displaying the dashboard selector. If you select the **Display dashboard selector** check box, a drop-down list of other dashboards that the viewer has access to is available in the Web part. This drop-down list allows the viewer to quickly jump to another dashboard.

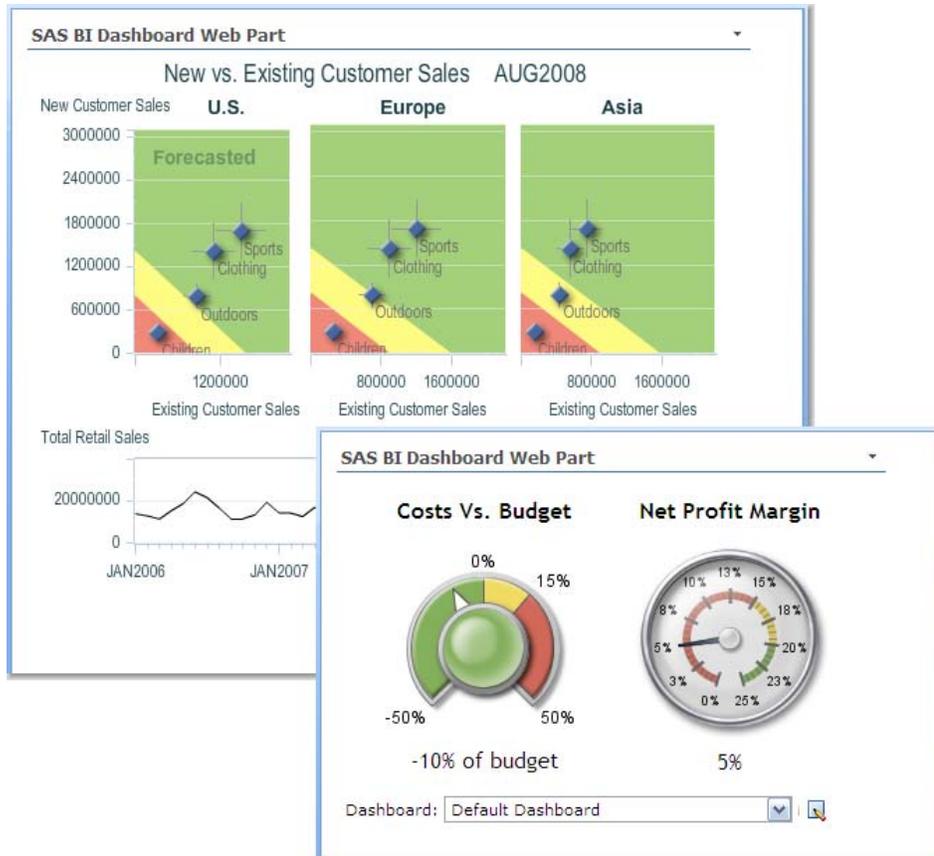


Figure 6. Sample dashboards displayed in the SAS BI Dashboard Web Part

Dashboards displayed in the Web part are fully interactive. If an indicator in the dashboard is linked to other SAS content, clicking that indicator takes the viewer to the appropriate SAS Web application to view that SAS content.

Pulling SharePoint Data into SAS

One primary feature of SharePoint is its ability to gather and store lists of data. SharePoint enables you to create views of data that are sorted and filtered or that reorder and hide columns. However, SharePoint doesn't provide any way to analyze that data. Accessing that SharePoint list data from SAS enables you to use the analytical capabilities of SAS on your list data.

SharePoint allows you to access your list data in the form of an XML document over HTTP by using the RPC Protocol for SharePoint Team Services. (For more information, see "Overview of the RPC Protocol, SharePoint Team Services SDK" by Microsoft Corporation). The XML is easily mapped, which allows SAS to

process the data as if it were any other data source. The URL for the XML document containing the list data is in the following format:

```
http://<servername>/<site>/_vti_bin/owssvr.dll?Cmd=Display&List=<ListID>&XMLDATA=TRUE
```

The value for *ListID* is the Globally Unique Identifier (GUID) for the list. The easiest way to locate that GUID is to copy it from the URL of the list's Settings page. The GUID in the URL is already escaped with the appropriate characters for the open and close braces ({ }) and the dashes.

The first element in the XML is `<s:Schema>`, which provides details about the columns that are included for each row along with its type. Immediately following that section is the `<rs:data>` element that wraps the rows of data.

```
<xml xmlns:s='uuid:BDC6E3F0-6DA3-11d1-A2A3-00AA00C14882'
      xmlns:dt='uuid:C2F41010-65B3-11d1-A29F-00AA00C14882'
      xmlns:rs='urn:schemas-microsoft-com:rowset'
      xmlns:z='#RowsetSchema'>
  <s:Schema id='RowsetSchema'>
    <s:ElementType name='row' content='eltOnly' rs:CommandTimeout='30'>
      <s:AttributeType name='ows_Product_Name' rs:name='Product_Name'
        rs:number='1'>
        <s:datatype dt:type='string' dt:maxLength='512' />
      </s:AttributeType>
      <s:AttributeType name='ows_Unit_Weight_Oz' rs:name='Unit_Weight_Oz'
        rs:number='2'>
        <s:datatype dt:type='float' dt:maxLength='8' />
      </s:AttributeType>
      <s:AttributeType name='ows_Case_Price' rs:name='Case_Price'
        rs:number='3'>
        <s:datatype dt:type='float' dt:maxLength='8' />
      </s:AttributeType>
    </s:ElementType>
  </s:Schema>
  <rs:data>
    <z:row ows_Product_Name='Buttermints Box'
      ows_Unit_Weight_Oz='3.50000000000000' ows_Case_Price='18.0000000000000' />
    <z:row ows_Product_Name='Butterscotch Discs Bag'
      ows_Unit_Weight_Oz='4.50000000000000' ows_Case_Price='14.0000000000000' />
    <z:row ows_Product_Name='Cinnamon Discs Bag'
      ows_Unit_Weight_Oz='4.50000000000000' ows_Case_Price='14.0000000000000' />
    <z:row ows_Product_Name='Dark Choc Bunny'
    ...
```

It's worth noting that the XML document represents the data that is included in the list's default view. If that view is filtered or only includes a portion of the columns, the XML will reflect that. If you are interested in more data or more columns than what's contained in the default view, you have two options. You can change the default view to include all of the data and columns that are of interest in SAS if changing that view is acceptable. Alternatively, you can create a new view that includes all of the items and columns that you'd like to extract. The ID of that view can then be included as an optional parameter in the URL. (Like the list, the easiest way to locate the view's GUID is to pull it from the URL of its Settings page.) With the optional View parameter, the URL follows the following format:

```
http://<servername>/<site>/_vti_bin/owssvr.dll?Cmd=Display&List=<ListID>&View=<ViewID>
&XMLDATA=TRUE
```

Because of the structure of the XML document produced by SharePoint, the SAS XML LIBNAME engine cannot read it directly, but by using SAS[®] XML Mapper, you can easily create an XML map that allows

SAS to read this data. The following XML map was produced by SAS XML Mapper for the previous SharePoint XML document.

```
<?xml version="1.0" encoding="Windows-1252"?>
<!-- ##### -->
<!-- SAS XML Libname Engine Map -->
<!-- Generated by XML Mapper, 902000.3.6.20090116170000_v920 -->
<!-- ##### -->
<SXLEMAP name="SXLEMAP" version="1.2">
  <TABLE name="rs_data">
    <TABLE-PATH syntax="XPath">/xml/rs:data/z:row</TABLE-PATH>
    <COLUMN name="ows_Product_Name">
      <PATH syntax="XPath">/xml/rs:data/z:row/@ows_Product_Name</PATH>
      <TYPE>character</TYPE>
      <DATATYPE>string</DATATYPE>
      <LENGTH>25</LENGTH>
    </COLUMN>
    <COLUMN name="ows_Unit_Weight_Oz">
      <PATH syntax="XPath">/xml/rs:data/z:row/@ows_Unit_Weight_Oz</PATH>
      <TYPE>numeric</TYPE>
      <DATATYPE>double</DATATYPE>
    </COLUMN>
    <COLUMN name="ows_Case_Price">
      <PATH syntax="XPath">/xml/rs:data/z:row/@ows_Case_Price</PATH>
      <TYPE>numeric</TYPE>
      <DATATYPE>double</DATATYPE>
    </COLUMN>
  </TABLE>
</SXLEMAP>
```

With both the URL to the XML document and the XML map file in hand, accessing the SharePoint list data from SAS is now accomplished with a simple fragment of SAS code. The following code maps the data to a library named Candies.

```
/* Environment */
filename Candies URL \
  'http://<server>/<site>/_vti_bin/owssvr.dll?Cmd=Display \
  &List=<ListID>&XMLDATA=TRUE' USER='ringo' PASS='ImASTarr!';
filename CandiesMAP 'c:\public\Candies.map';
libname Candies xml xmlmap= CandiesMAP access=READONLY;

/* Catalog */
proc datasets lib=Candies; run;

/* Contents */
proc contents data=Candies.rs_data varnum; run;

/* Printing */
title 'Table rs_data';
proc print data=Candies.rs_data; run;
title;
```

With the libref defined, you are ready to perform any SAS analytics on the data that you'd like. This approach to accessing SharePoint list data could easily be included in a stored process. Then you can include the stored process in a SharePoint Web parts page by using the SAS Stored Process Web Part. The result would be a SAS report on the SharePoint data. This report is included in a SharePoint page and created in real time.

Federated Search

Another opportunity for integration between SAS and Microsoft SharePoint is in the area of search. SharePoint supports *search federation*—which simply means that other search engines can be registered as providers with SharePoint. When a viewer executes a search, SharePoint calls out to each of the registered providers and then integrates all of the results from the different providers into the final search results, which are then displayed to the viewer. The communication between SharePoint and the search providers is accomplished by using the OpenSearch standard, which is a simple XML data exchange. Included in SAS 9.2 Phase 2 is the Search Interface to SAS, which implements the OpenSearch 1.1 standard. Because the Search Interface to SAS supports the OpenSearch standard, it is recognized by SharePoint.

To register the Search Interface to SAS, open **Shared Services Administration** from the quick launch in SharePoint **Central Administration**. Select **Search settings** below the Search heading. On the Configure Search Settings page, click **Manage Federated Locations** to open the list of locations. Click **New Locations** to add the Search Interface to SAS as a federated location. Alternatively, included in the installation of the Search Interface to SAS is a *federated location definition file* (.FLD). You can use this file to add the federated location for SAS to the list of federated locations in SharePoint.

Shared Services Administration: SharedServices1 > Search Administration > Federated Locations > Add Federated Location

Add Federated Location

Use this page to edit a federated location. On this page, you can enter general information about the location and specify how search results should be obtained and formatted.

* Indicates a required field

OK Cancel

General Information

Location Name
Type a unique name to identify this location within your organization. This name will be visible to service administrators and developers, and it cannot be modified once it has been created.

Location Name: *
SAS92

Display Name
Type a display name for this location. This name will identify the location to service and site administrators. Site administrators can choose to display this name to end-users in federated-enabled Web Parts.

Display Name: *
SAS Results

Description
Type a description for this location. The description will be visible to service

Description: *

Figure 7. Adding a federated search location allows you to include SAS Reports, Stored Processes and SAS Information Maps in search results.

After the federated location for SAS has been added, the search results in SharePoint include hits in the SAS content stored in SAS Metadata, such as reports, stored processes, SAS Information Maps, OLAP cubes and physical tables. The search results also include links to find similar items and a **More Results** link that opens a complete list of search hits. When a viewer clicks on a link to SAS content, the content opens in the appropriate SAS Web application.

sales Advanced Search

Results 1-3 of 3. Your search took 2.08 seconds.

Results by Relevance | View by Modified Date | Alert Me | RSS

SAS Search Results: sales

Total Line Item Sale Amount

Year	Total Line Item Sale Amount
2004	~5,000,000,000
2005	~10,000,000,000
2006	~12,000,000,000
2007	~15,000,000,000

Yearly Sales by Customer Type.srx
Report which shows the Yearly sales by Customer Type in a Bar Chart. Similar Reports
<http://ssdvm01.na.sas.com:8080/SASWebReportStudio/...>

Line Item Sales by Country by Channel.srx
This report shows the net and total line item sales amounts by country and by order channel. Similar Reports
<http://ssdvm01.na.sas.com:8080/SASWebReportStudio/...>

Yearly Sales by Customer Type with Detail Data.srx
This report shows the Yearly Sales by Each Customer with a Graph as well as detail data in Tabular Format. Similar Reports
<http://ssdvm01.na.sas.com:8080/SASWebReportStudio/...>

[More results ...](#)

Appalachian Chocolate: Sales and Marketing
Sales and Marketing ... Sales and Marketing ... Welcome to the Sales and Marketing web site.
<http://punesearch01.apac.sas.com:90/chocolate/docs/salesandmarketing/index.html> - 5KB - 4/12/2007

Chocolate Enterprises Intranet
Sales and Marketing
<http://punesearch01.apac.sas.com:90/chocolate> - 5KB - 4/17/2007
[\[View duplicates\]](#)

Appalachian Chocolate: Human Resources
Sales and Marketing
<http://punesearch01.apac.sas.com:90/chocolate/docs/hr/index.html> - 6KB - 4/13/2007

Figure 8. SharePoint search results including items from SAS.

Federated search is a very nice way of integrating SAS content with SharePoint without actually placing the content into SharePoint. This approach allows you to administer permissions and access to the content through metadata privileges in SAS, while still exposing the content to a wider SharePoint audience.

Conclusion

Because SharePoint and SAS are both such robust and complex platforms, integration between the two can take a variety of forms and require different things in various situations. The topics covered here are by no means a complete list of techniques for integration, but these basic techniques will go a long way toward bringing the two platforms together.

References

- Microsoft Corporation. (2004). Overview of the RPC Protocol, SharePoint Team Services SDK. Available <http://msdn.microsoft.com/en-us/library/ms947752.aspx>

Acknowledgements

The authors would like to thank Tom Booth and Chandra Prakash for their contributions to this paper.

Contact Information

Our integration with Microsoft SharePoint is an ongoing effort at SAS. If you have questions or comments about topics covered in this paper or other areas of integration, we would love to hear from you. Your comments and questions are valued and encouraged. Contact the author at:

David R. Henderson
SAS Institute Inc.
Work Phone: 919-531-4560
E-mail: David.Henderson@sas.com

Sean Alexandre
SAS Institute Inc.
Work Phone: 919-531-0589
E-mail: Sean.Alexandre@sas.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.