



Integrating SharePoint Sites within WebSphere Portal

November 2007

Contents

Executive Summary	2
Proliferation of SharePoint Sites	2
Silos of Information	2
Security and Compliance	3
Overview: Mainsoft SharePoint Integrator for WebSphere Portal	3
Enterprise Scenario	3
Mainsoft Integrator Portlets	5
The List Viewer Portlet	5
Microsoft Office Integration Using the Item Viewer Portlet	6
Delivering Single Sign-On in Mixed Environments	7
Centralized, Enterprise-Wide Access to Departmental SharePoint Sites	7
Single Sign-On Access to SharePoint in a Heterogeneous Organization	7
Trusted Identity in a Microsoft-Centric Organization	8
Mainsoft SharePoint Integrator Software Development Kit	9
Mainsoft SharePoint Integrator Architecture	9
SharePoint ASP.NET Web Control Toolkit	9
Creating Composite Applications	11
Conclusion	11



Executive Summary

Companies with strategic investments in IBM WebSphere® Portal are looking for ways to integrate Microsoft® SharePoint® data with .NET and Java™ applications in order to improve enterprise IT performance. Because of its simplicity and ease of use, many workgroups use SharePoint for departmental collaboration portals while enterprise IT relies on WebSphere for its enterprise portal infrastructure and front-end to its Service Oriented Architecture. Now, customers are looking for solutions that integrate the end-user experience across software platforms and bring SharePoint sites into compliance with corporate security standards and branding.

Mainsoft® [SharePoint Integrator for WebSphere Portal](#) expands WebSphere Portal's ability to serve as an Über-portal that accesses and aggregates all enterprise assets, including SharePoint Services, Microsoft Office document libraries, and .NET and Java applications, within a scalable, secure enterprise portal. IT organizations can respond quickly to new IT requirements, delivering composite and workflow applications across SharePoint data and .NET and Java applications in a single sign-on, role-based environment.

This whitepaper explores the capabilities and architecture of Mainsoft SharePoint Integrator for WebSphere Portal and illustrates how a superior integration across .NET and Java can deliver a seamless end-user experience and improve employee productivity. Single sign-on and authentication schemes in a mixed .NET and Java environment, and composite applications across disparate platforms, are examined in detail.

Proliferation of SharePoint Sites

Many organizations have turned to the Microsoft SharePoint family of products for day-to-day departmental collaboration sites. Windows® SharePoint Services comes bundled with the Windows Server at no additional charge, and Microsoft Office makes it possible for end users to set up collaborative work spaces at the click of a button. As a result, SharePoint sites tend to multiply rapidly across the organization, with little or no input from the central IT organization.

As SharePoint contents become strategic to the enterprise, companies need to deliver departmental resources to broader audiences within and outside the organization, without compromising corporate security standards or workgroups' autonomy.

Silos of Information

Multiple, separately-administered SharePoint installations and content sprawl create silos of information across the enterprise that are difficult to integrate with enterprise applications and legacy systems. While valuable to individual workgroups and departments, the lack of coordination and integration among those installations prevents enterprises from getting the full value out of that information. Customer data may, for example, be stored in a SharePoint custom list or in a document library, but there is no easy way to integrate this data into the Java-based enterprise CRM system. Interoperability issues between .NET and Java EE can limit the effectiveness of IT, requiring end users to reenter data into different systems.



Security and Compliance

Exposing SharePoint sites to enterprise or external audiences invariably requires complying with strict corporate security, governance, and branding requirements. This is not a trivial task. Providing corporate-wide access to data from multiple sites, which are running on multiple servers with different user directories, dramatically increases the complexity of identity management and the risk of identity theft. The central IT department also needs to simplify workflows and eliminate the need for users to reenter their credentials in order to access content stored on the SharePoint sites. Finally, the enterprise needs to find an effective way to unify branding and navigational models, without having to re-implement these attributes on each SharePoint site.

In order to fully utilize all enterprise assets and break down silos between individual workgroups, integration of SharePoint sites with enterprise portals is a necessity. However, until now, the semantics gap between the .NET and Java platforms has made all but the most superficial integrations difficult to achieve.

Mainsoft SharePoint Integrator bridges this divide and supports deeper integration and workflows between workgroup sites and enterprise portals.

Overview: Mainsoft SharePoint Integrator for WebSphere Portal

Mainsoft SharePoint Integrator provides the federation of content from SharePoint directly into WebSphere Portal, enabling a seamless enterprise user experience across SharePoint contents and .NET and Java applications. SharePoint lists and Office document libraries from multiple SharePoint sites can be aggregated into a secure, role-based view to the enterprise from a single point of access. The data stored on these sites can also fully participate in inter-portlet communications and composite applications.

The SharePoint Integrator includes:

- **Ready-to-use SharePoint Integrator Portlets**, which provide secured access to SharePoint lists and documents libraries within the WebSphere Portal environment.
- **A Visual Studio®-based Software Development Kit (SDK) for SharePoint**, which enables .NET developers to build composite applications that include SharePoint data and .NET and Java portlets. .NET developers can also customize the Federation portlets, supplied in source code form.

Enterprise Scenario

To demonstrate the superior integration that can be achieved between SharePoint sites and WebSphere Portal, let's take a look at a sample order fulfillment and processing portal that gives employees single sign-on, integrated access to their ASP.NET applications, SharePoint data and Java portlets within WebSphere Portal (Figure 1). Until now, these processes were handled manually, with employees reentering order details into separate applications.

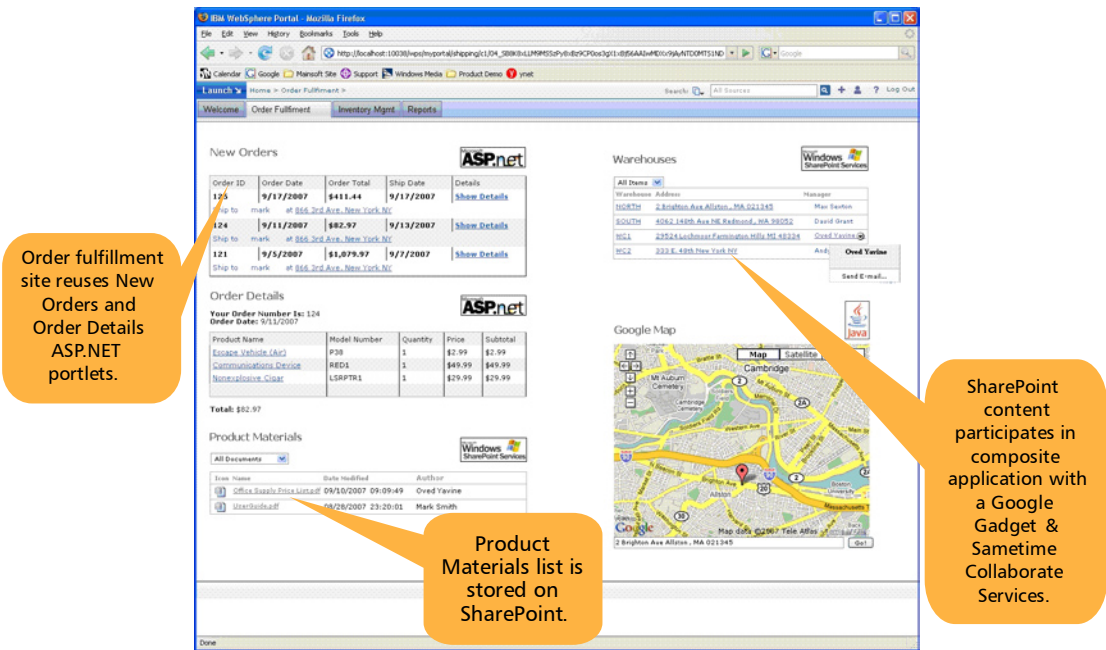


Figure 1. SharePoint data available within WebSphere Portal.

In this example, an order fulfillment clerk logs onto WebSphere Portal using her corporate user credentials. She accesses the Order Fulfillment page, which displays five portlets comprising the Order Fulfillment application. She can click on a new order, select an item from the Order Details portlet, such as the "Communication Device," and then obtain the list of documents that will be shipped with the order from the Product Materials list, which are stored on a SharePoint site. The Product Materials portlet is an instance of a SharePoint Integrator List Viewer portlet (Figure 2) wired with the Order Details portlet. It gets the product ID from the Order Details portlet and then retrieves the related documents from the Product Materials document libraries stored on SharePoint. Notice the fulfillment clerk doesn't need to re-enter her SharePoint credentials in order to access the documents and print them from Microsoft Office.

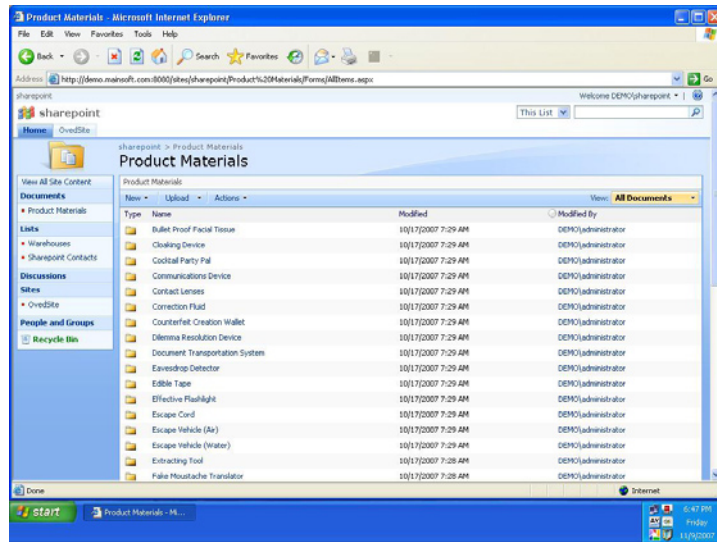


Figure 2: The SharePoint document library remains on the SharePoint site.



Enterprise users like the fulfillment clerk can have secured access to the SharePoint document libraries using the credentials and permissions stored on WebSphere Portal. Meanwhile, the product documentation team can continue to maintain the document library on SharePoint using their departmental user directory.

Returning to the example, the fulfillment clerk next selects the warehouse that will ship the order. Another instance of the SharePoint list viewer integrates a separate SharePoint list that contains the warehouse name, address, manager, and inventory. The Warehouse portlet is wired to the Google Map Java portlet. When the fulfillment clerk selects the warehouse address, Google Maps will plot it on the map. If she has a question, she can select the warehouse manager's name, and IBM Same Time collaboration services will pop up so she can talk to the warehouse manager using instant messaging, e-mail or voice over IP using the People Awareness tag.

Business users can work more efficiently with secured access to Office document libraries and other SharePoint data participating in composite applications and enterprise workflows. Once they are logged into WebSphere Portal, they do not need to re-enter information into applications that are running on disparate platforms, and there's no need to follow separate login procedures to access data stored on SharePoint sites.

Mainsoft Integrator Portlets

The Mainsoft Integrator Portlets includes the List Viewer portlet, which provides secured access to SharePoint document libraries and to SharePoint list such as contacts, announcements, tasks, events, and custom lists. The Item Viewer portlet displays the contents of a single item in the SharePoint List. When accessing SharePoint document libraries containing Microsoft Office documents, the Integrator portlet provides direct access to the SharePoint document from the Microsoft Office application.

The List Viewer Portlet

Configuration. Administrators connect to the SharePoint site by providing the site URL (Figure 3).

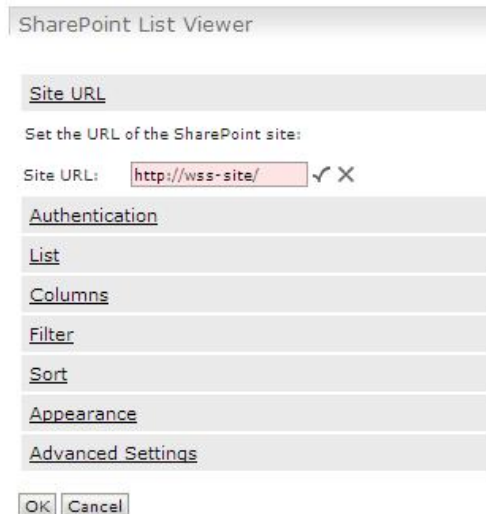


Figure 3: Selecting the SharePoint site.

The List Viewer portlet discovers the available lists so administrators can select the lists to integrate and the columns they want to display. They can also add criteria to filter the list items (Figure 4).

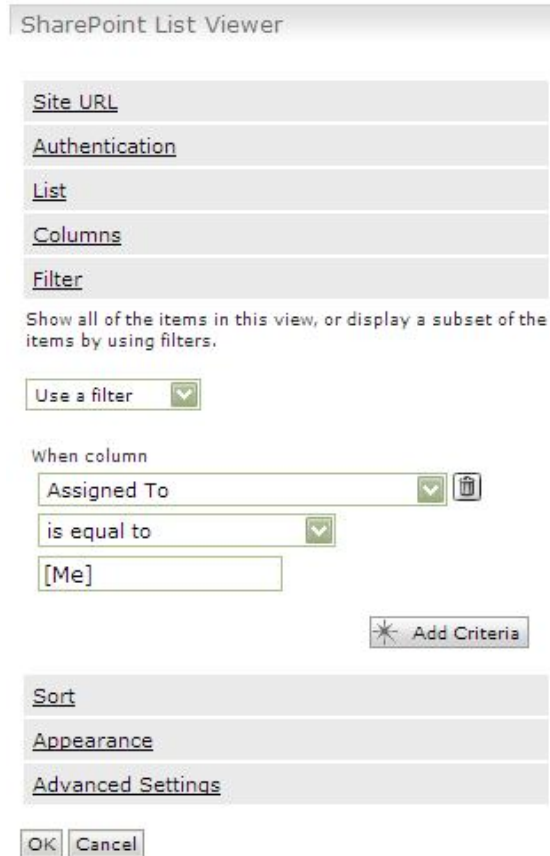


Figure 4: Adding a filter criteria.

Administrators can also filter the items, define sorting, and customize the portlet's appearance.

Search capacities. Portlet users can search their SharePoint list content and review ranked results to easily find specific documents and data.

Caching and paging. The SharePoint Integrator portlets optimize the user experience by caching SharePoint data and paging long lists for fast retrieval.

Microsoft Office Integration Using the Item Viewer Portlet

The SharePoint Integrator Item Viewer portlet provides:

Microsoft Office integration. The Item Viewer portlet can link directly to documents on the SharePoint site, enabling the same access to Office documents from WebSphere Portal as accessing documents directly from SharePoint. The portlets support check-in, check-out, rights-management, meta-data, document action and document information panels.

Full access to SharePoint documents. End users can check-out, check-in, and update documents stored in SharePoint sites.

Delivering Single Sign-On in Mixed Environments

The SharePoint Integrator relies on WebSphere Portal security and supports multiple enterprise authentication schemes for portal-to-portal interoperability.

Centralized, Enterprise-Wide Access to Departmental SharePoint Sites

The SharePoint Integrator Portlets enable central IT organizations to give enterprise users access to multiple SharePoint sites, each using a different user directory. In this scenario, the central IT organization controls enterprise users' access to SharePoint sites using WebSphere Portal authentication, while the line of business department maintains its ability to update SharePoint site contents directly using SharePoint credentials (Figure 5).

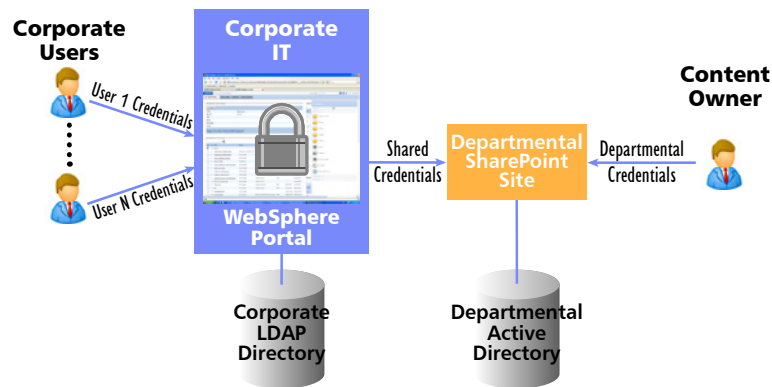


Figure 5: Centralized access control to departmental sites.

Enterprise users log in to WebSphere Portal using their portal credentials and access SharePoint document libraries and SharePoint lists based on roles and group memberships stored on WebSphere Portal. The sales department, for example, can access to the SharePoint content related to their activities only. They cannot access the SharePoint document libraries restricted to employees in other departments. Teams maintaining their respective SharePoint sites can continue to access them using SharePoint credentials to add, remove, or update documents.

To implement centralized access to departmental SharePoint sites, portal administrators define role-based security rules to access the various SharePoint List Viewer portlet instances, each one configured to display a specific SharePoint list. Then, they configure each portlet to access the SharePoint site using shared system credentials. This way, role-based access control is centralized at the corporate portal, and authorization levels for enterprise users to access the SharePoint site are defined in SharePoint.

Single Sign-On Access to SharePoint in a Heterogeneous Organization

The SharePoint Integrator also supports single sign-on in organizations that have multiple user registries (see Figure 6).

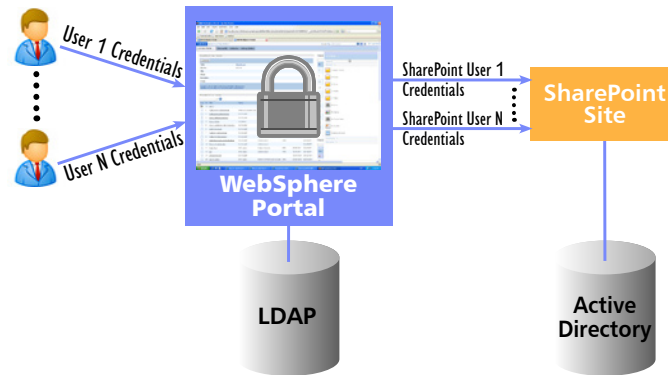


Figure 6: Single sign-on access across multiple user directories.

In this scenario, enterprise users who want to access SharePoint data will login to WebSphere Portal using their portal credentials, and the Integrator portlets will retrieve their SharePoint credentials stored in WebSphere Credential Vault to authenticate them on the SharePoint site. The portal administrator needs to configure the portlet to store the SharePoint user credentials, and end users will use the portlet personalization form to input their SharePoint credentials the first time they access the SharePoint site.

Trusted Identity in a Microsoft-Centric Organization

Finally, the SharePoint Integrator enables enterprises using Microsoft Active Directory as their corporate user directory to access SharePoint sites from WebSphere Portal. Single sign-on access to the Windows domain is based on Kerberos (Figure 7).

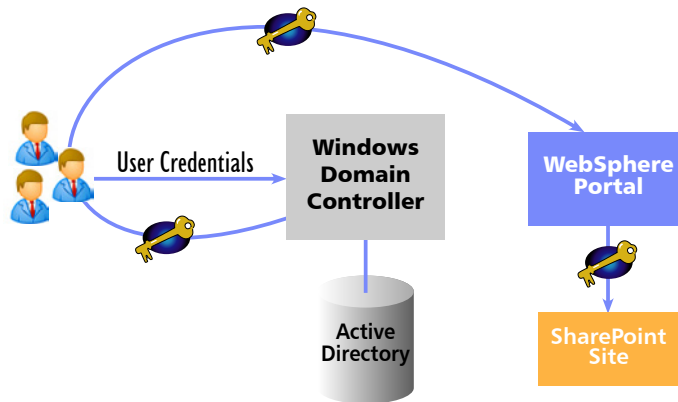


Figure 7: Single sign-on based on Kerberos.

Both WebSphere Portal and SharePoint sites are connected to the corporate Active Directory for authentication. WebSphere Portal is defined as a trusted entity within the Windows domain, so it does not require additional credentials, and it delegates the Windows domain identity to the SharePoint server. With this authentication scheme, the Mainsoft SharePoint Integrator enables IT organizations to use their trusted single sign-on environment to provide direct access to SharePoint sites from WebSphere Portal.



Mainsoft SharePoint Integrator Software Development Kit

The Mainsoft SharePoint Integrator SDK enables .NET developers to build custom portlets and composite applications that include SharePoint data and to customize the Integrator Portlets, supplied with the SDK in source code form. SharePoint data, previously accessible using SharePoint Web pages exclusively, can be integrated with enterprise applications using composite applications and inter-portlet communications. The SharePoint SDK introduces SharePoint ASP.NET data controls and provides ASP.NET support on WebSphere Portal.

Mainsoft SharePoint Integrator Architecture

Mainsoft SharePoint Integrator is a Visual Studio-based SDK that enables .NET developers to create JSR 168-compliant portlets using ASP.NET 2.0, the .NET 2.0 Framework, C#, and Visual Basic® 2.0 languages. The .NET portlets run natively on WebSphere Portal in the JSR 168 container.

Mainsoft's patented cross compiler compiles .NET source code into Java bytecode. The generated class files are packed in a standard JSR 168 WAR file, together with a robust Java runtime implementation of ASP.NET, ADO.NET, and the .NET Framework. Because the .NET source code remains unchanged, .NET developers can continue to maintain their applications using the Visual Studio development environment (see Figure 8).

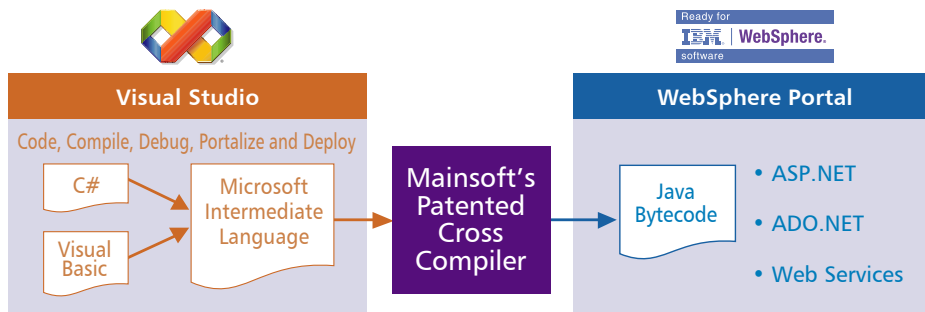


Figure 8: SharePoint Integrator SDK components.

Mainsoft SharePoint Integrator SDK enables Visual Studio developers to take full advantage of the WebSphere Portal platform, invoking WebSphere Portal services such as credential vault, content APIs, and [cooperative portlets](#) as well as standard JSR 168 APIs directly using C# or Visual Basic.

SharePoint ASP.NET Web Control Toolkit

The SharePoint DataSource Control

The SharePointDataSource control is a major component of the SharePoint Integrator SDK. It represents a SharePoint data store that complies with the ASP.NET 2.0 declarative model for data access. It can be used out-of-the-box with any ASP.NET data-bound UI control such as GridView, DataList, or TreeControl to display SharePoint data onto the portal, without having to write a single line of code.

The SharePointDataSource control provides a rich designer wizard, which takes the developer through the configuration steps, including connecting to the SharePoint site, selecting lists selection, selecting fields, conducting queries, and sorting. With just a few clicks, the ASP.NET developers can present the SharePoint data in a JSR 168 portlet. (See Figures 9 and 10.)

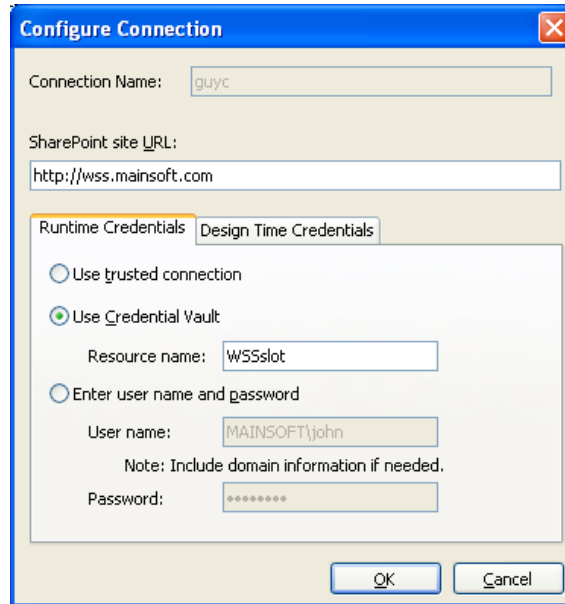


Figure 9: Configuring the SharePoint connection at design time.

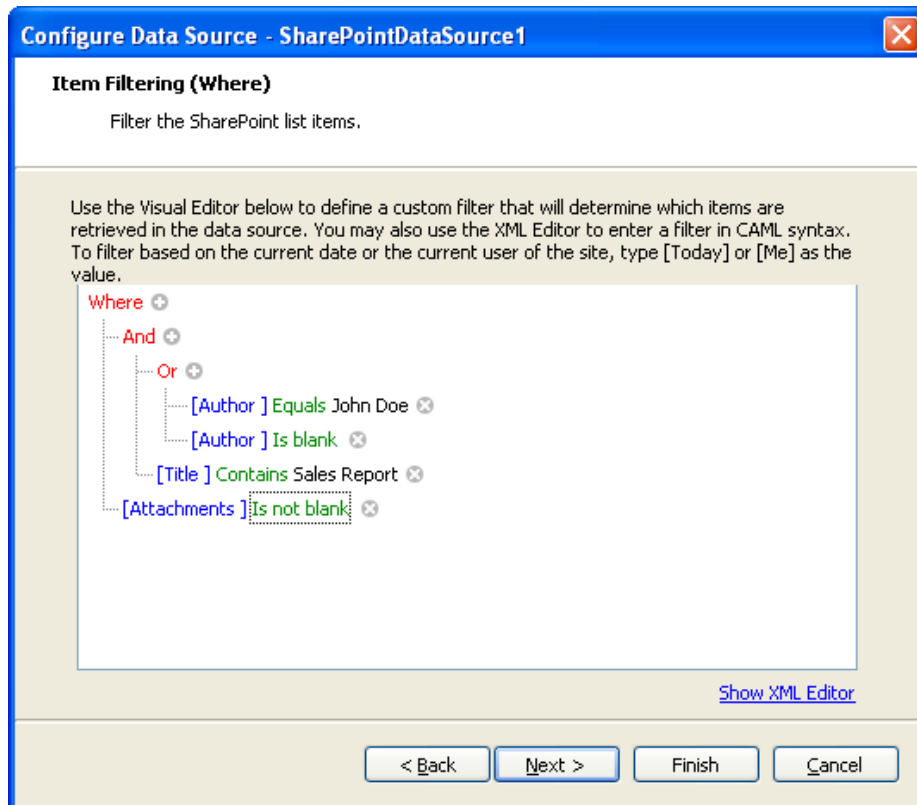


Figure 10: SharePoint graphic query builder.



The SaveCredentials Control

The SharePoint Integrator SDK provides also the SaveCredentials control, which enables ASP.NET developers to build portal Web Forms that capture SharePoint credentials and store them in the WebSphere Portal Credential Vault slot. Using the SaveCredentials control you can provide a single sign-on experience for portlet users that are viewing SharePoint content.

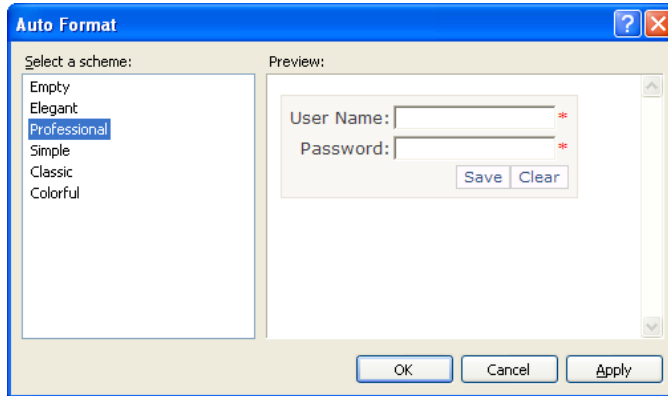


Figure 11: The SaveCredentials control.

Creating Composite Applications

Returning again to the enterprise order fulfillment scenario, let's see how an enterprise .NET developer created the SharePoint portlets that comprise the order fulfillment composite application.

Using Mainsoft SharePoint Integrator SDK, a .NET developer created a new C# for Java EE project, selecting the ASP.NET Portal Application template. The developer then created the "ProductMaterials" portlet main grid using the SharePoint DataSource control bounded to an ASP.NET 2.0 GridView. Then, to create the workflow from the "Order Details," he added a WSDL file in the Visual Studio project, through which the portlet subscribes to an inter-portlet communication event. The portlet receives the product ID as an input parameter from the "Order Details" portlet and uses this product ID as a query parameter to select documents from the SharePoint Product site. In a similar way, the developer created the "Warehouse" portlet, which uses the same product ID to query the Warehouses SharePoint lists and display the warehouses that have the product in stock.

To create an enterprise mash-up with the Java-based Google Map portlet, the developer defined an event in the "Product Materials" portlet to produce the address in the exact format expected by the Google Map portlet. Once a warehouse is selected in the "Product Materials" portlet, it fires the event and triggers the Google map portlet to display the selected warehouse map. He also used the Person control from the SharePoint Integrator control toolbox to display the warehouse manager's name. The Person control dropdown menu launches the IBM SameTimes IM portlet.

Using Mainsoft's Visual Studio-based SDK, the enterprise developer integrated SharePoint document libraries and SharePoint data into the fulfillment process, increasing the fulfillment agent's productivity and delivering the flexibility expected in a Service Oriented Architecture.



Conclusion

WebSphere Portal gives enterprises a long-term basis for a flexible SOA, with role-based composite applications and full support for on-the-glass integration. But a portal can only fulfill the SOA promise when all enterprise data and services are equally accessed through the enterprise portal, regardless of their technology foundation, and when all developers can continue to apply their existing skills and code. Mainsoft SharePoint Integrator enables WebSphere Portal to aggregate SharePoint sites into an open standards-based architecture that integrates SharePoint Services and Office document libraries into composite and workflow applications.

To begin your evaluation, download a 30-day trial of the SharePoint Integrator portlets, available at <http://dev.mainsoft.com/Default.aspx?tabid=296>.