Integrating with SharePoint

A software vendor's perspective

In my last article on SharePoint published in IDM in April 2008 (SharePoint – A Primer: For EDRM and ECM managers) I promised a future article on integrating with SharePoint. Well, days have a way of filling up and time has a way of flying by so it has taken a little longer than expected.

The core problem isn't SharePoint (though creating applications in SharePoint isn't for the faint hearted); it is anyone's understanding of the word "integration".

For example:

- Do you want me to take information from SharePoint and add it to my application?
- Do you want me to take information from my application and add it to SharePoint?
- Do you want my application to be able to search information in SharePoint?
- Do you want SharePoint to be able to search for information in my application?

There are literally hundreds of such questions one has to ask before you can define what the customer means by "integration".

In order to simplify this paper, I am going to assume that in most cases the customer actually wants SharePoint to be its corporate portal; the common, shared 'gateway' into most information within the enterprise. This is OK because this is what SharePoint is really good at as long as you keep it simple.

In order to illustrate how this integration process works I am going to show you how we integrate our product RecFind 6 to SharePoint. Note however that what you see is a 'generic' solution only; every customer wants it to look and work differently; this is not a standard product.

For our sample integration we built a simple SharePoint portal and added the functionality to search the RecFind 6 database (and other data sources) and to run a few simple RecFind functions. This is what our sample portal looks like:
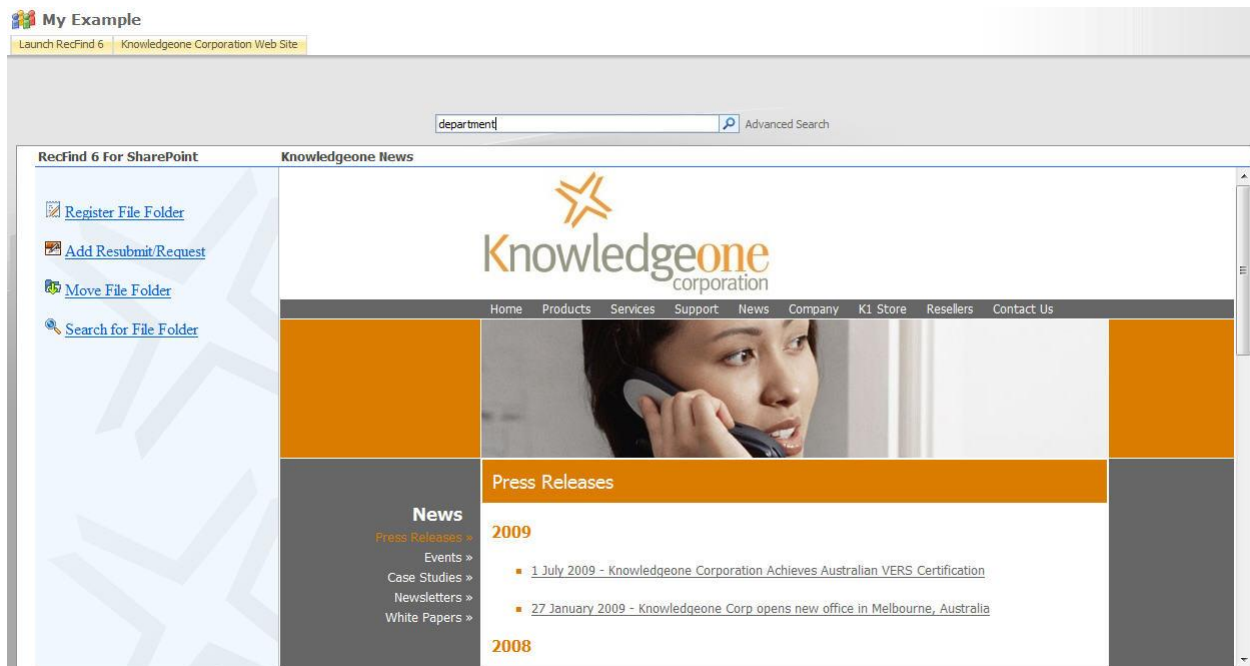
The search we chose to implement is called a Federated search and it is a standard feature of SharePoint. You may be told it requires the use of another server called Search Server 2008 but it actually doesn't if you use the right components and the right version of SharePoint. For the technically minded:

- In order to perform Federated Searches and other advance searching features in MOSS 2007, You need to update to MOSS 2007 SP1 and install the Infrastructure Update for Microsoft Office Servers
- SharePoint pages use WebParts (Microsoft version of web widgets). WebParts can be stand alone components or can connect to other WebParts on the same page. i.e., they can be consumers or producers in SharePoint parlance.
- Federated searches are based on OpenSearch 1.0/1.1 and implemented via a Federated Connector that returns its search results in XML format as an RSS feed.

Our task is made easier because we built our full text search engine in RecFind 6 using SQL Server's standard text search functionality (using iFilters). We automatically index all Metadata and electronic documents (stored as Blobs in our database). This means we are 'ready' for a SharePoint Federated search.

In our example our Federated search searches the RecFind database, the Adventure Works database and the Internet. It is an example of a true knowledge management search, connecting to all sources of knowledge.

However, then we have to present the information 'found' by the search in a way that makes sense to the user. We do this in two stages. The first is to present the results as URLs (hyperlinks) just as Bing does within the portal. See example below:
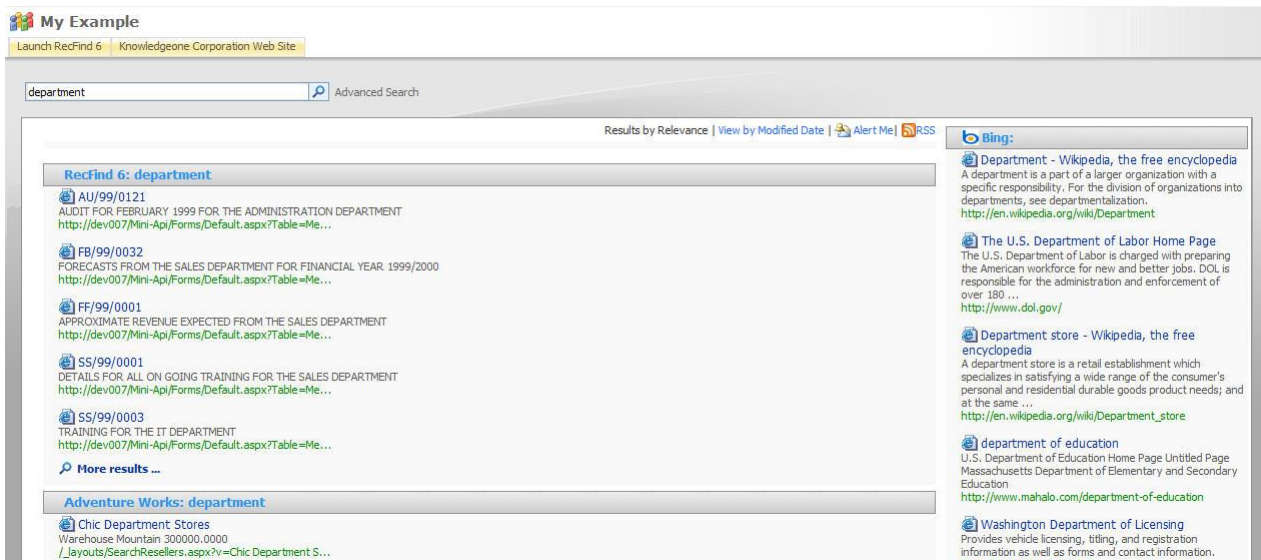
Now if a user wants to see the details/full document he/she clicks on the hyperlink.
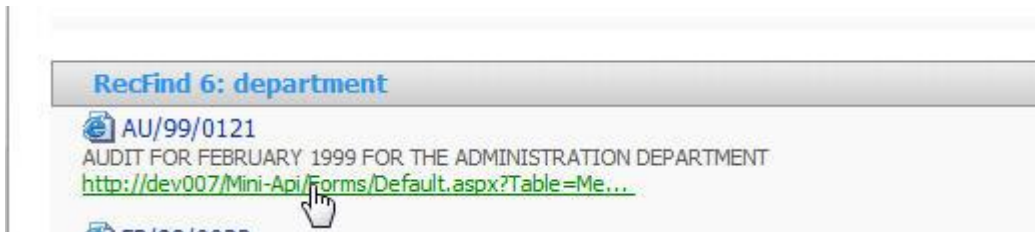
It is at this stage we have to write some code because we need to be able to display the information held in RecFind 6 in a consistent and sensible manner and you can't display all the different types of information we store in RecFind 6's database in Internet Explorer.

Maintain Document Security

We also need to make sure that all requests go through our security system. We have to enforce the same access rights that apply to RecFind 6 users. The requirement for document security doesn't go away just because we are using a SharePoint portal. Luckily we had already developed a suitable product for this called the RecFind 6 MINI API. It is the 'middleman' between RecFind 6 and any external request for information and it enforces RecFind6's security regime. When the user clicks on the URL, this is what our MINI API display screen looks like when displaying Metadata from RecFind 6:

Type	🔍 STANDARD - File Folder
File Folder Number	AU/99/0121
PartNumber	1
Barcode#	F0000013
SecurityCode	🔍 STD_EVERYONE
CurrentLocation	Location Registry - since 03 Feb 1999
Title1	🔍 ASSET MANAGEMENT
Title2	🔍 AUDIT
Title3	🔍 ACCRUAL
Title4	
Title5	

We handle electronic documents a little differently, using the authoring package (or in this case ADOBE) to display the electronic document in its 'natural' form:
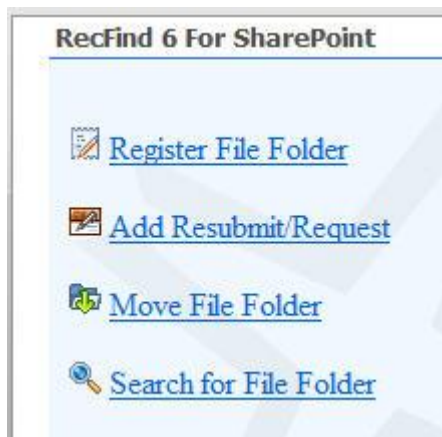
RecFind 6: sales

🌐 PersonList.pdf
S.DIAS@EXERCISES.COM
http://dev007/Mini-Api/Forms/Default.aspx?Table=ED...

In this example we also provide access to some basic RecFind functions in addition to a full text search. We chose some basic records management functions:

RecFind 6 For SharePoint

📝 Register File Folder

📝 Add Resubmit/Request

📁 Move File Folder

🔍 Search for File Folder

This is harder than doing a full text search because now you are forced to write code to replicate some of the functionality of RecFind 6.

This is where I would like to make a very strong point. If you have a sophisticated application like RecFind 6 with powerful text, Metadata and BOOLEAN searches and a host of other functionality you would be a fool to try to replicate it in SharePoint because that is exactly what SharePoint is not good at. If you need to do the hard things then just use the native application.

For the Technically minded reader

This last section is again for the more technically minded. Explaining in detail how we built our portal is beyond the scope or length of this article so this is a very brief summary for our IT literate readers.

Once you have installed MOSS 2007, SP1, and then Infrastructure Update for Microsoft Office Servers, you then have the ability to add federated locations to your SharePoint searches.

First you have to add your desired federated location through the SharePoint Central Administration page. Once you navigate through a few menus, you get to a section called Manage Federated Locations. Here we create a new one called RecFind 6
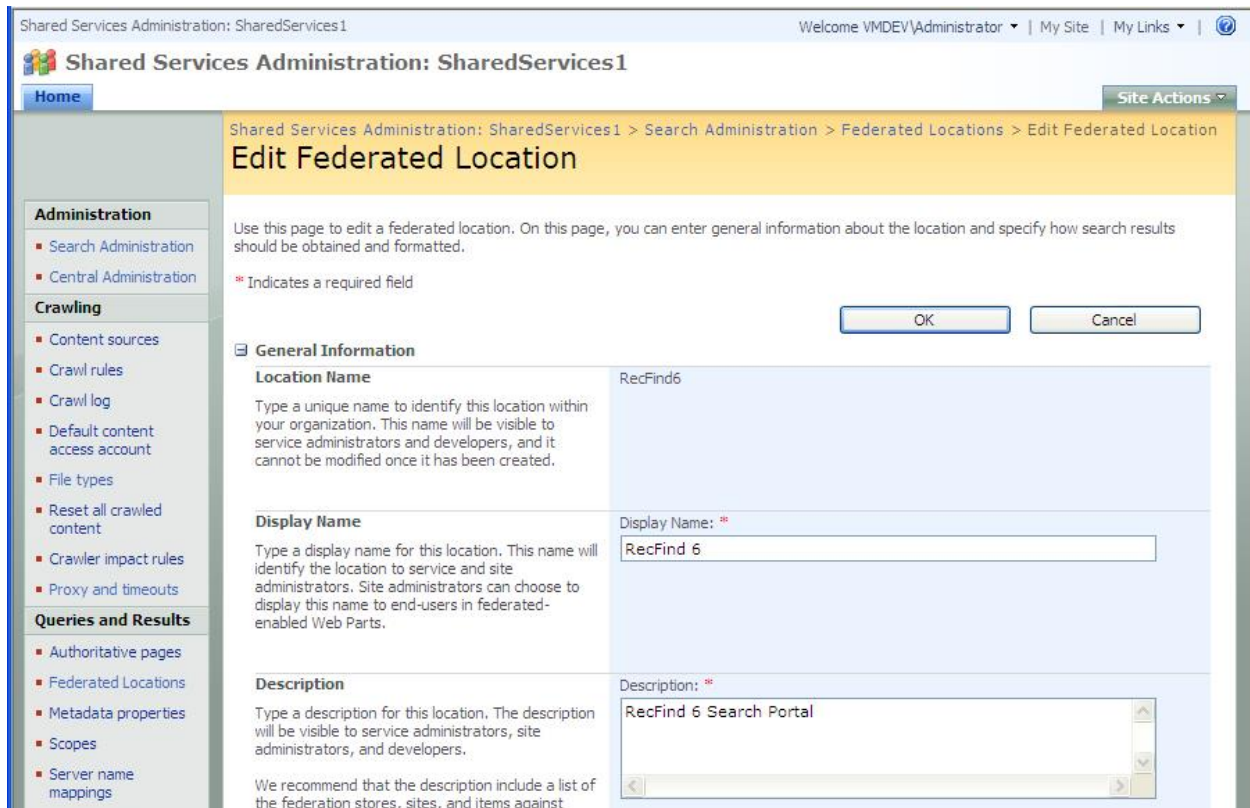
You need to select that you are using OpenSearch 1.0/1.1 and you need to specify the URL where SharePoint will send its query term and get back results as XML.

OpenSearch:

> Select OpenSearch 1.0/1.1 to display results from another search engine that can receive a query by using a URL and return results as structured XML. Use this protocol to search Web sites that support RSS or Atom, or to search remote Search indexes on other farms.

Federated connector (Query Template) you use:

> Query Template
> Specify the template for passing queries to the OpenSearch location's URL. In the template, the case-sensitive parameter "{searchTerms}" represents the keywords

entered into the Search box by end-users. If you specified a prefix trigger, "{searchTerms}" represents the keywords with the prefix removed. Instead of "{searchTerms}", you can also use capture groups defined by your trigger pattern (see Trigger Pattern above).

The query template is equivalent to the URL template in OpenSearch. The URL specified should return structured XML (typically RSS or Atom results). It should not consist of a URL that shows an HTML-based search results page.

Example A: The following URL would query a remote search server site:
http://server/SearchCenter/_layouts/srchrss.aspx?k={searchTerms}

Example B: This query would display medical documents with a specific ID number:
http://www.example.com/search-rss.aspx?q=docid:{searchTerms}%20topic=medical

Example C: This query would search only the contents of the "<city>" capture group (instead of the entire user query):
http://server/SearchCenter/_layouts/srchrss.aspx?k={city}

Basically it is saying it passes the search term to your federated connector. You can also use prefixes and patterns. It is just a means of refining the search, so that a user could specify a particular table only.

You will also have to provide a means of authentication and there are many types to choose from. A federated connector can use the authentication method to apply security to the results it returns to SharePoint.

Once you have set up your federated location, you just need to add it as a "Federated Results" Web Part on the results page of the desired SharePoint Site. This means it will be searched along with any other sources you have set up in your results page of SharePoint.

When results are returned from our federated connector, each record contains a link to that particular record in the Mini-API (our standard interface tool with RecFind 6).

Summary

SharePoint is a great tool for corporate portals and it has all the functionality required to integrate it to any application or data source. However, building integration requires a clear idea of what is required and it is a very technical job; even though all the tools are there it is definitely not an easy task.

My best advice is keep it simple, do not try to get SharePoint to do what it is not good at.