



# INTRODUCTION TO THE BUSINESS DATA CATALOG (BDC) IN MICROSOFT OFFICE SHAREPOINT SERVER 2007 (MOSS)

BY BRETT LONSDALE, MCSD.NET , MCT

COMBINED KNOWLEDGE

[WWW.COMBINED-KNOWLEDGE.COM](http://WWW.COMBINED-KNOWLEDGE.COM)

[BRETT@COMBINED-KNOWLEDGE.COM](mailto:BRETT@COMBINED-KNOWLEDGE.COM)

## INTRODUCTION

The intention of this White Paper is to introduce the benefits of using Business Data Catalog (BDC), Microsoft Office SharePoint Server 2007 Enterprise Edition (MOSS) within your Organisation.

BDC was introduced with MOSS 2007 Enterprise and provides Information Workers and Managers with the ability to view Line of Business Data on a SharePoint Web Part Page. Many senior managers don't have the time on their hands to learn how to query the data and produce reports on how their business is running, BDC can help by providing Easy to Use Web Parts that can filter the data, display the columns of information that they want to see, and through the use of SharePoint Designer provide a rich conditional formatted view of the data in their back end systems. BDC data

can also be used within our Lists and Libraries allowing us to store BDC LOB data as metadata against our Documents and List Items.

MOSS can also index BDC data sources providing Information Workers with the ability to search Line of Business data using the same search tool that they use in MOSS to search their Documents, Lists, Sites & Workspaces, File Shares, and Web Sites.

Within this White Paper we will explore from start to finish how to connect to our Line of Business Data, how to configure the BDC Web Parts, how to configure searching of BDC Data, and how to use BDC data in our Lists and Libraries.

The setup used to work provide this White Paper is:

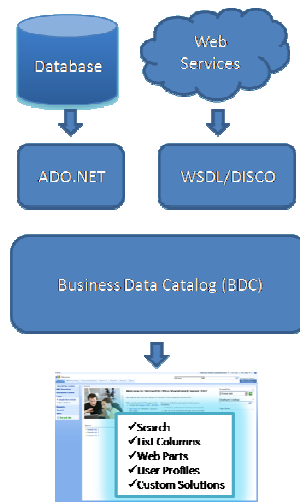
- Microsoft Vista Ultimate Host Operating System
  - Microsoft Virtual PC 2007
- Microsoft Windows Server 2003 Service Pack 2 Stand Alone
  - Microsoft SQL Express Edition
  - ASP.NET
  - IIS 6.0
  - Microsoft Office SharePoint Server 2007 Enterprise Edition
  - Microsoft SQL Server Management Studio Express
  - Adventure Works Database
  - Microsoft Office Ultimate 2007
  - Microsoft Visual Studio 2005 Professional Edition

## BUSINESS DATA CATALOG – THE ARCHITECTURE

In order to access LOB data in MOSS you need to firstly create an Application Definition file that is imported into SharePoint via the Shared Services Provider ensuring that the data you import is available across Web Applications in your server farm. The Application Definition file provides connection information so that the data can be access by SharePoint. Filters, Parameters, and Actions are also configured in the Application Definition file allowing you to filter the data

retrieved, provide parameters to Stored Procedures, and perform Actions such as linking to custom functions or Web Sites that do something further with the data.

The Application Definition file can be configured to fetch data from lots of different data sources including SQL, Oracle, Siebel, SAP, and any data that is accessible via a Web Service. ADO.NET allows connections over OLEDB, and ODBC providers.



Once the Application Definition has been imported, The Entity/s (Table/Stored Procedure) can be accessed via several different Web Parts on any Web Part page in SharePoint. The Web Parts can be further configured to allow more filtering, hiding of columns, and connecting to other Web Parts that may also contain data from BDC even from a different Data Source. The Web Parts are covered in detail later in this White Paper.

When building custom lists or Libraries, BDC data can be used to populate a column within the List or Library. E.g. If you were to create a Proposal Document for a Customer, you could select the Customer as metadata for the document using the Customer ID column in the library which in turn displays Customers from a column in your database table. The column can also be populated via the Information Panel available in Microsoft Office 2007 applications.

MOSS also provides an extensive search allowing you to create a Search Centre for your BDC data. During the creation of the Application Definition File, an ID Enumerator can be created usually on a key field in the Table allowing the iteration of the data either through the Object Model or the MOSS Indexer. This means that not only can you search on Data stored throughout SharePoint, but you can also search your back end databases through MOSS.

User Profiles play an important role in MOSS with the much improved My Site, and People Searches etc. Data can be imported via BDC data sources into your User Profile information.

Finally, you can create custom applications using the BDC Object Model, which is beyond the scope of this white paper but will be covered in a future White Paper. For an example of a custom

application, you could try out the Key Performance Indicator Web Part from Lightning Tools which displays KPI's on BDC data.

## THE APPLICATION DEFINITION FILE

The Application Definition file is an XML file based on the BDCMetadata.XSD schema file that can be found in the (*installdirectory*:\Program Files\ Microsoft Office Servers\12.0\Bin) folder.

As you can see in the below example you configure the Application Definition file for each application you wish to create allowing you to define associations between Entities (Relationships), define the connection string to the database, configure the filters, Id Enumerators etc.

The image shows an XML code snippet for an application definition file, annotated with callout boxes explaining various parts:

- Provide a Name for your BDC Application:** Points to the `Name="AdventureWorksLOBSystem"` attribute in the `<LobSystem>` element.
- Reference to the BDC MetaData Schema File:** Points to the `xmlns="http://schemas.microsoft.com/office/2006/03/BusinessDataCatalog BDCMetadata.XSD"` attribute.
- Properties for the Data Connection:** Points to the `<Properties>` block within the `<LobSystemInstance>` element.
- Provide an Instance Name:** Points to the `Name="AdventureWorksInstance"` attribute in the `<LobSystemInstance>` element.
- The Entity (Table|SP) you are connecting to:** Points to the `Name="RdbConnection Integrated Security"` property within the `<Properties>` block.
- ID Field:** Points to the `Name="ProductID"` attribute in the `<Identifier>` element.
- Methods:** Points to the `<Method Name="GetProduction.[Product]">` element.
- Columns to Return:** Points to the `<Property Name="RdbCommandText">` element.
- Filters:** Points to the `<FilterDescriptor Type="Comparison" Name="price" />` element.
- Parameters:** Points to the `<Parameter Direction="In" Name="@price">` element.

```

<?xml version="1.0" standalone="yes" ?>
- <LobSystem xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://schemas.microsoft.com/office/2006/03/BusinessDataCatalog BDCMetadata.XSD"
  xmlns="http://schemas.microsoft.com/office/2006/03/BusinessDataCatalog" Type="Database" Version="1.0.0"
  Name="AdventureWorksLOBSystem">
- <Properties>
  <Property Name="WildcardCharacter" Type="System.String">%</Property>
</Properties>
- <LobSystemInstance Name="AdventureWorksInstance">
  <Properties>
    <Property Name="DatabaseAccessProvider" Type="System.String">SqlServer</Property>
    <Property Name="AuthenticationMode" Type="System.String">Passworh</Property>
    <Property Name="RdbConnection Data Source" Type="System.String">WINZK3\OFFICESERVERS</Property>
    <Property Name="RdbConnection Initial Catalog" Type="System.String">AdventureWorks</Property>
    <Property Name="RdbConnection Integrated Security" Type="System.String">SSPI</Property>
    <Property Name="RdbConnection Pooling" Type="System.String">true</Property>
  </Properties>
</LobSystemInstance>
</LobSystemInstances>
- <Entities>
- <Entity EstimatedInstanceCount="0" Name="Production.Product">
  <Identifiers>
    <Identifier Name="ProductID" TypeName="System.Int32" />
  </Identifiers>
  <Methods>
- <Method Name="GetProduction.[Product]">
  <Properties>
    <Property Name="RdbCommandText" Type="System.String">Select [ProductID],[Name],[ProductNumber],[MakeFlag],
      [FinishedGoodsFlag],[Color],[SafetyStockLevel],[ReorderPoint],[StandardCost],[Size],[SizeUnitMeasureCode],
      [WeightUnitMeasureCode],[Weight],[DaysToManufacture],[ProductLine],[Class],[Style],[ProductSubcategoryID],
      [ProductModelID],[SellStartDate],[SellEndDate],[DiscontinuedDate],[rowguid],[ModifiedDate],[ListPrice] From Production.
      [Product] where([ReorderPoint] < @price)</Property>
    <Property Name="RdbCommandType" Type="System.Data.CommandType">Text</Property>
  </Properties>
  <FilterDescriptors>
    <FilterDescriptor Type="Comparison" Name="price" />
  </FilterDescriptors>
  <Parameters>
- <Parameter Direction="In" Name="@price">
    <TypeDescriptor TypeName="System.Int16" AssociatedFilter="price" Name="ReorderPoint">
      <DefaultValues>
        <DefaultValue MethodInstanceName="Production.[Product]Finder" Type="System.Int16">10</DefaultValue>
      </DefaultValues>
    </TypeDescriptor>
  </Parameters>
  </Method>
</Entity>
  </Entities>
</LobSystem>

```

```

PublicKeyToken="077a5c561934e089" Name="Production.[Product]DataRecord" >
  <TypeDescriptors>
    <TypeDescriptor TypeName="System.Int32" IdentifierName="[ProductID]" Name="ProductID" />
    <TypeDescriptor TypeName="System.String" Name="Name" />
    <TypeDescriptor TypeName="System.String" Name="ProductNumber" />
    <TypeDescriptor TypeName="System.Boolean" Name="MakeFlag" />
    <TypeDescriptor TypeName="System.Boolean" Name="FinishedGoodsFlag" />
    <TypeDescriptor TypeName="System.String" Name="Color" />
    <TypeDescriptor TypeName="System.Int16" Name="SafetyStockLevel" />
    <TypeDescriptor TypeName="System.Int16" Name="ReorderPoint" />
    <TypeDescriptor TypeName="System.Decimal" Name="StandardCost" />
    <TypeDescriptor TypeName="System.String" Name="Size" />
    <TypeDescriptor TypeName="System.String" Name="SizeUnitMeasureCode" />
    <TypeDescriptor TypeName="System.String" Name="WeightUnitMeasureCode" />
    <TypeDescriptor TypeName="System.Decimal" Name="Weight" />
    <TypeDescriptor TypeName="System.Int32" Name="DaysToManufacture" />
    <TypeDescriptor TypeName="System.String" Name="ProductLine" />
    <TypeDescriptor TypeName="System.String" Name="Class" />
    <TypeDescriptor TypeName="System.String" Name="Style" />
    <TypeDescriptor TypeName="System.Int32" Name="ProductSubcategoryID" />
    <TypeDescriptor TypeName="System.Int32" Name="ProductModelID" />
    <TypeDescriptor TypeName="System.DateTime" Name="SellStartDate" />
    <TypeDescriptor TypeName="System.DateTime" Name="SellEndDate" />
    <TypeDescriptor TypeName="System.DateTime" Name="DiscontinuedDate" />
    <TypeDescriptor TypeName="System.Guid" Name="rowguid" />
    <TypeDescriptor TypeName="System.DateTime" Name="ModifiedDate" />
    <TypeDescriptor TypeName="System.Decimal" Name="ListPrice" />
  </TypeDescriptors>
</TypeDescriptor>
</TypeDescriptors>
</TypeDescriptor>
</Parameters>
</Parameter>
</MethodInstances>
<MethodInstance Name="Production.[Product]Finder" Type="Finder" ReturnParameterName="Product"
  ReturnPropertyDescriptorName="Production.[Product]DataReader" ReturnPropertyDescriptorLevel="0" />
</MethodInstances>
</Method>
<Method Name="Production.[Product]SpecificFinder">
  <Properties>
    <Property Name="RdbCommandText" Type="System.String">Select [ProductID],[Name],[ProductNumber],[MakeFlag],
    [FinishedGoodsFlag],[Color],[SafetyStockLevel],[ReorderPoint],[StandardCost],[ListPrice],[Size],[SizeUnitMeasureCode],
    [WeightUnitMeasureCode],[Weight],[DaysToManufacture],[ProductLine],[Class],[Style],[ProductSubcategoryID],
    [ProductModelID],[SellStartDate],[SellEndDate],[DiscontinuedDate],[rowguid],[ModifiedDate] From Production.[Product]
    Where ([ProductID]=@ProductID)</Property>
  </Properties>

```

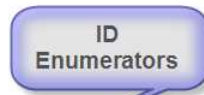
Type Descriptors set the Type of Data to be returned for each column

Finder Methods allow searching

```

- <TypeDescriptor TypeName="System.Data.IDataReader, System.Data, Version=2.0.3600.0, Culture=neutral,
  PublicKeyToken=b77a5c561934e089" Name="Production.ProductSubcategoryDataReader" IsCollection="true">
- <TypeDescriptors>
- <TypeDescriptor TypeName="System.Data.IDataRecord, System.Data, Version=2.0.3600.0, Culture=neutral,
  PublicKeyToken=b77a5c561934e089" Name="Production.ProductSubcategoryDataRecord">
- <TypeDescriptors>
  <TypeDescriptor TypeName="System.Int32" IdentifierEntityName="Production.ProductSubcategory"
    IdentifierName="[ProductSubcategoryID]" Name="ProductSubcategoryID" />
  <TypeDescriptor TypeName="System.Int32" Name="ProductCategoryID" />
  <TypeDescriptor TypeName="System.String" Name="Name" />
  <TypeDescriptor TypeName="System.Guid" Name="rowguid" />
  <TypeDescriptor TypeName="System.DateTime" Name="ModifiedDate" />
</TypeDescriptors>
</TypeDescriptor>
</TypeDescriptors>
</TypeDescriptor>
</Parameter>
</Parameters>
</Method>
- <Method Name="Production.[ProductCategory]IDEnumerator">
- <Properties>
  <Property Name="RdbCommandType" Type="System.String">Text</Property>
  <Property Name="RdbCommandText" Type="System.String">Select [ProductCategoryID] from Production.[ProductCategory]
  </Property>
</Properties>
- <Parameters>
- <Parameter Direction="Return" Name="[ProductCategoryID]s">
- <TypeDescriptor TypeName="System.Data.IDataReader, System.Data, Version=2.0.3600.0, Culture=neutral,
  PublicKeyToken=b77a5c561934e089" Name="Production.[ProductCategory]s" IsCollection="true">
- <TypeDescriptors>
- <TypeDescriptor TypeName="System.Data.IDataRecord, System.Data, Version=2.0.3600.0, Culture=neutral,
  PublicKeyToken=b77a5c561934e089" Name="Production.[ProductCategory]">
- <TypeDescriptors>
  <TypeDescriptor TypeName="System.Int32" IdentifierName="[ProductCategoryID]" Name="ProductCategoryID" />
  <TypeDescriptor TypeName="System.Guid" Name="rowguid" />
</TypeDescriptors>
</TypeDescriptor>
</TypeDescriptors>
</TypeDescriptor>
</Parameter>
</Parameters>
- <MethodInstances>
  <MethodInstance Name="[ProductCategoryID]EnumeratorInstance" Type="IdEnumerator"
    ReturnParameterName="[ProductCategoryID]s" />

```




---

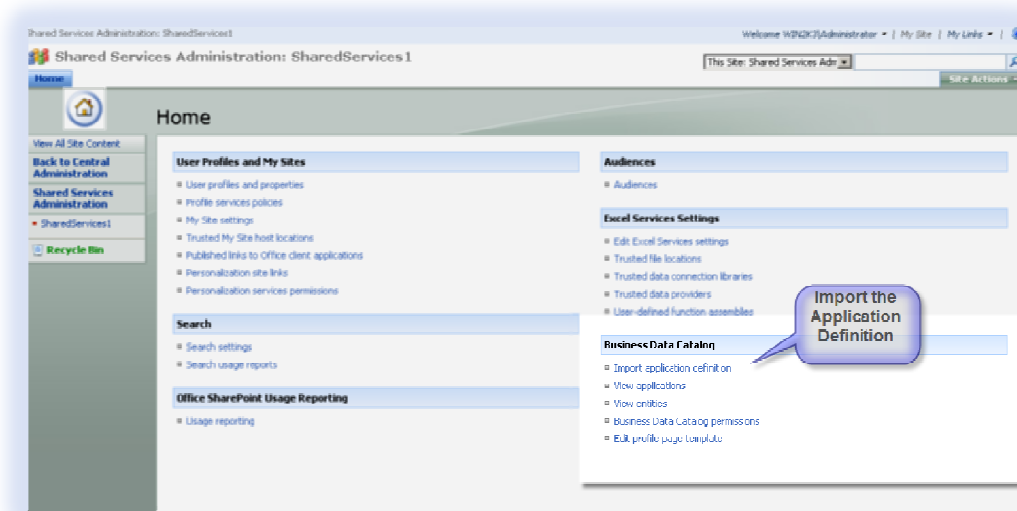
*NOTE: Creating the Application Definition file can take some time even for an experienced Developer. BDC Metaman can create the Application Definition file for you within minutes, allowing you to configure all of the above. There are two products: Developer, and Professional. The Developer version is free and allows connections to SQL databases.*

---

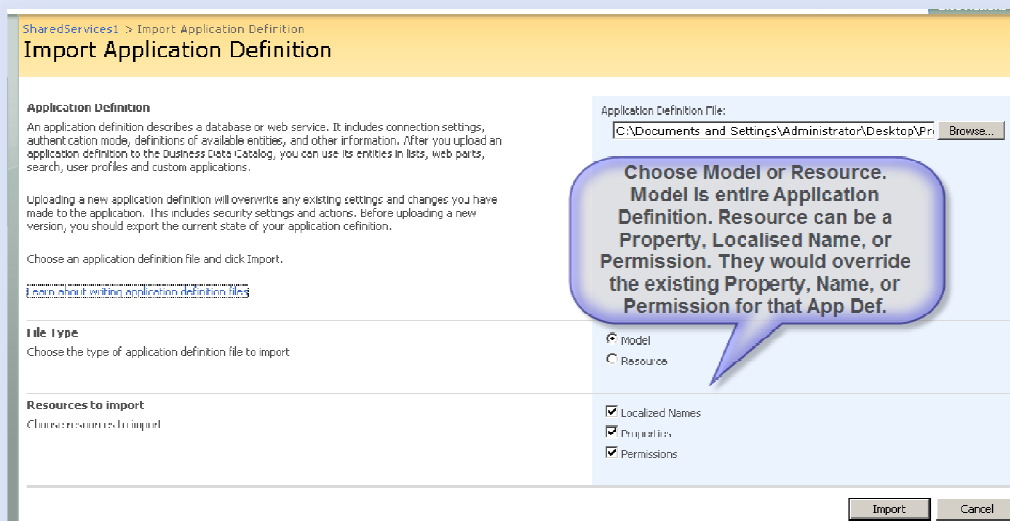
## IMPORTING THE APPLICATION DEFINITION FILE

Once you have successfully created the Application Definition file, you need to import it into your Shared Services Provider. To do this:

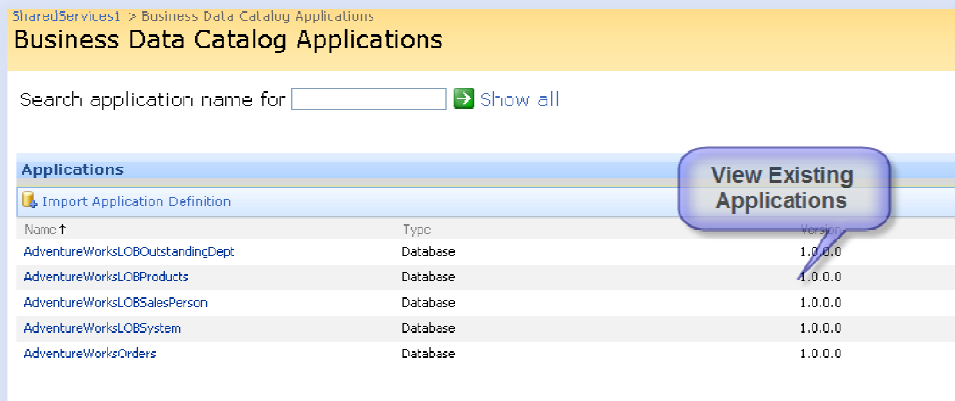
1. Navigate to SharePoint 3.0 Central Administration
2. Choose your Shared Services Provider from the Quick Launch Menu.
3. Choose 'Import Application Definition' from the Business Data Catalog Section.



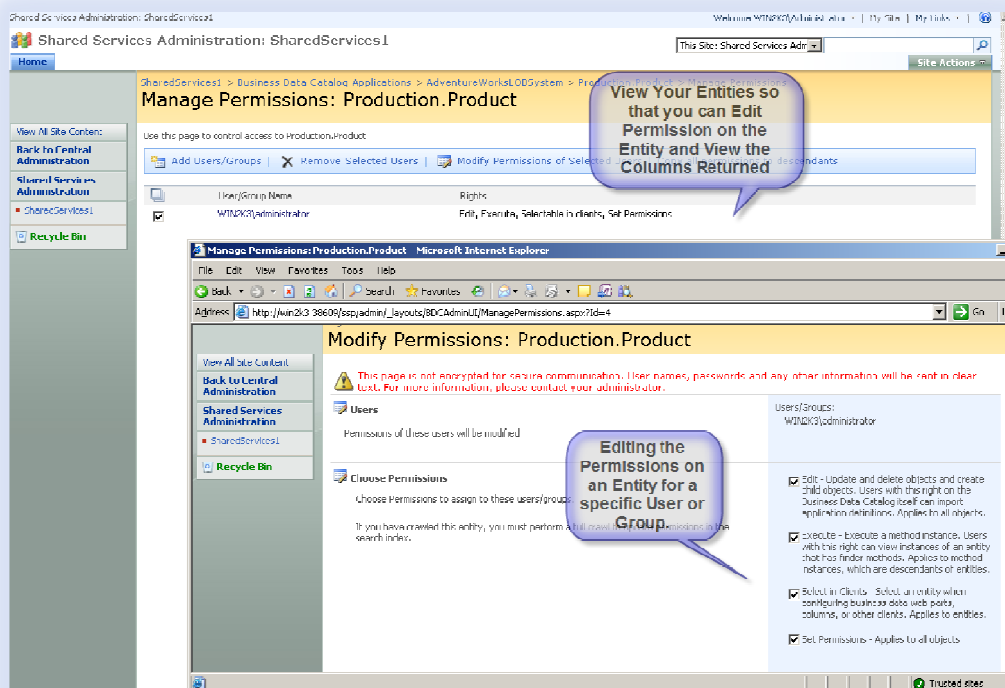
You can then import the Application Definition file entirely, or just part of it by choosing resource such as Localised Name (for multi languages), Properties, or Permissions overriding any existing Properties, Permissions or Localized Names with the same name.



Once you have imported the Application Definition file, you can view your existing Applications. This page also allows the Viewing, Editing, Exporting and Deleting of your Applications. Exporting the Application is a good thing to do as it will allow you to backup your Application.



The Entities can also be viewed via the Business Data Catalog section in 'Shared Services' menu. You can then View the Columns available in the Entity, or Edit permissions that users or groups will receive on the Entity. By Editing the Permissions you can specify who can Edit, Update and Delete Objects in the Entity, Who can Execute methods in the Entity, who can select the Entity in Web Parts, and Finally who can set permissions on the Entity.

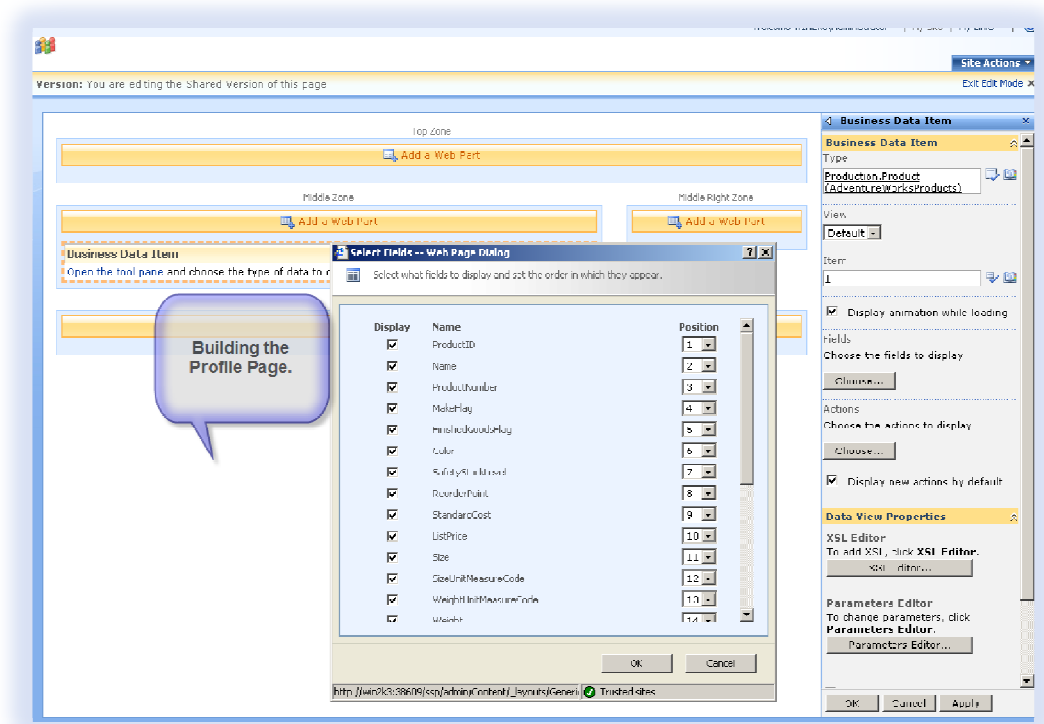




The last option in the Business Data Catalog section on the Shared Services menu is to Edit the Business Data Profile Template.

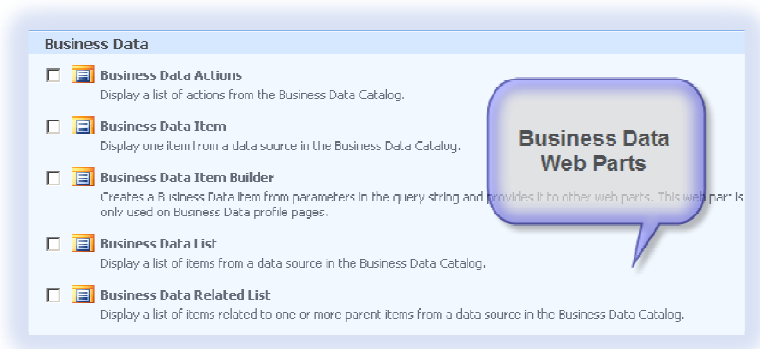
Business data profile pages are Web Part pages that display details about a business data item defined in the BDC Entity. Business data profiles may, for example, display a product name along with more detailed information on the product.

Business data profiles are created automatically for each entity using the Business Data Item Builder Web Part. This Web Part, used only for business data profiles, builds a business data item from identifiers in the URL and sends it to connected Web Parts. A View Profile link appears in the Business Data Web Parts allowing you to view the Profile Page.



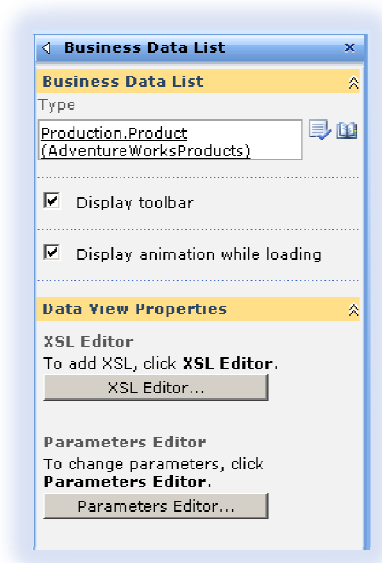
## USING THE BDC WEB PARTS

There are several BDC Web Parts available allowing you to display the data from BDC on a Web Part page, and connect the Web Parts together using Web Part Connections.



We are going to concentrate on the Business Data Item Web Part and the Business Data List. The Business Data Item displays one row from your entity at a time. The Business Data List display all the items you like that adhere to the filter you setup either in the Web Part itself or in the Application Definition file.

To configure the Business Data List Web Part, Add the Web Part to your page, and modify it Shared Properties. Using the Browse button, look for the Entity that you wish to add. In our case we are adding the Production.Product Entity.



I hope you found this White Paper useful, I would be grateful for any comments you may have and whether it was useful or not. The Web Part initially displays all of the columns making the Web Part to wide. We can trim it down by choosing which columns we want to display. Click Edit View on the top right hand side of the Web Part to remove some columns. Whilst editing the view you can setup Pagination, Sorting, Filtering, and the Order of the Columns.

Production.Product List

ProductID	Name	FinishedGoodsFlag	Color	ListPrice
1	Adjustable Race	False		0.0000
2	Bearing Ball	False		0.0000
3	OD Ball Bearing	False		0.0000
4	Leadset Ball Bearings	False		0.0000
316	Blade	False		0.0000
317	LL Crankarm	False	Black	0.0000
318	HL Crankarm	False	Black	0.0000
319	HL Crankarm	False	Black	0.0000
320	Chaining Bolts	False	Silver	0.0000
321	Chaining Nut	False	Silver	0.0000
322	Chaining	False	Black	0.0000
323	Crown Race	False		0.0000
324	Chain Stays	False		0.0000
325	Down Tube	False		0.0000
326	Decal 1	False		0.0000
327	Decal 2	False		0.0000
328	Mountain End Caps	False		0.0000
329	Road End Caps	False		0.0000
330	Touring End Caps	False		0.0000
331	Front End	False		0.0000

Completed Business Data List Web Part

The next Web Part that we will configure is the Business Data Item Web Part that just display's one item at a time. We will display one item at a time using Web Part connections. Add a Business Data Item Web Part onto the same page. Configure it's properties so that the Type is set to Production.Product again as with the Business Data List Web Part. Set the Item property to an item otherwise the connections won't work. To add the connection, click the Business Data List Web Part Menu and choose Connections to configure the connection to the Business Data Item Web Part.

The screenshot shows a SharePoint page titled 'Pages - BDC Web Parts' in Microsoft Internet Explorer. The page contains two web parts: 'Production.Product List' and 'Production.Product'. The 'Production.Product List' web part is displaying a table of product data. A context menu is open over the 'Production.Product List' web part, showing options like 'Minimize', 'Close', 'Drift', 'Modify Shared Web Part', 'Connections', and 'Send Selected Item to'. The 'Send Selected Item to' option is selected, and a sub-menu is open showing 'Production.Product' as the target.

**Production.Product List** edit x

Actions 1 - 20 Edit View

Providing Selected Item to Production.Product

ProductID	Name	FinishedGoodsFlag	Color	ListPrice
1	Adjustable Race	False		0.0000
2	Adjustable Race	False		0.0000
3	Adjustable Race	False		0.0000
4	Adjustable Race	False		0.0000
316	Adjustable Race	False		0.0000
317	Adjustable Race	False	Black	0.0000
318	ML Crankarm	False	Black	0.0000
319	ML Crankarm	False	Black	0.0000
320	Chaining Bolts	False	Silver	0.0000
321	Chaining Nut	False	Silver	0.0000
322	Chaining	False	Black	0.0000
323	Crown Race	False		0.0000

Once the Web Parts are connected, Fire the connection by clicking the Radio Button. This will display more detail about the product in the consuming Web Part.

**Production.Product** edit x

View Profile

Receiving Item from Production.Product List

ProductID: 1  
 Name: Adjustable Race  
 ProductNumber: AR-5381  
 MakeFlag: False  
 FinishedGoodsFlag: False  
 Color:  
 SafetyStockLevel: 1000  
 ReorderPoint: 750  
 StandardCost: 0.0000  
 ListPrice: 0.0000  
 Size:  
 SizeUnitMeasureCode:  
 WeightUnitMeasureCode:  
 Weight:  
 DaysToManufacture: 0  
 ProductLine:  
 Class:

## USING BDC DATA IN COLUMNS

You can lookup and Store BDC Data in List & Library columns so that it can be used to describe documents or list items. You can then filter your documents or list items on the BDC data, Create Views, and Search the Documents based on metadata.

Additional Column Settings

Specify detailed options for the type of information you selected.

Choose Business Data as a Column Type

Choose The Entity & Column to be stored

Number (1, 2, 3, 4, 5)  
 Currency (\$, €, #)  
 Date and Time  
 Lookup (information already on this site)  
 Yes/No (check box)  
 Person or Group  
 Hyperlink or Picture  
 Calculated (calculation based on other columns)  
 XPS Name Near  
 Business data  
 Todd World Address

Uppercase: [ ]

Require that this column contains information:  
 Yes  No

Type: Sales.SalesPerson (AdventureWorks.SalesPerson)

Display the field of the selected type:  
 SalesQuota

Display the actions menu  
 Link this column to the profile page

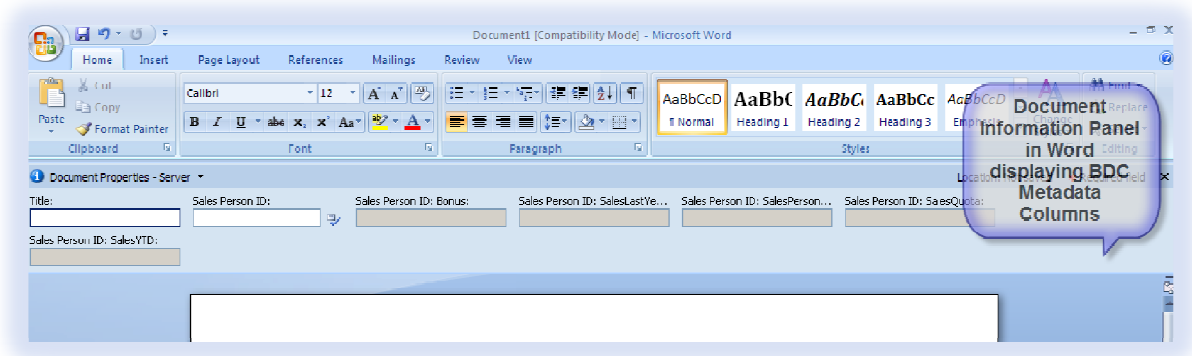
Add a column to show each of these additional fields:

Bonus  
 CommissionPct  
 ModifiedDate  
 Rowguid  
 SalesLastYear  
 SalesPersonID  
 SalesQuota  
 SalesYTD  
 TerritoryID

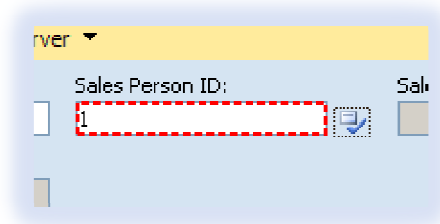
Add Additional Columns

OK Cancel

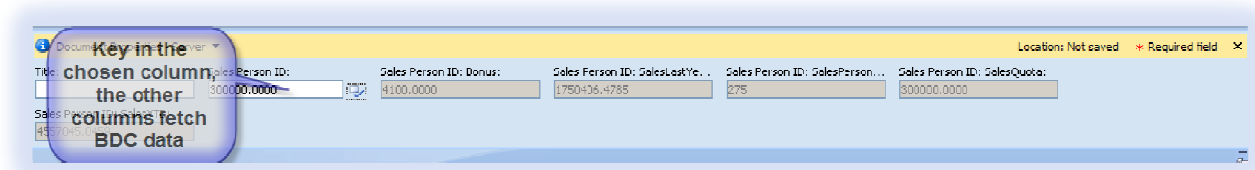
The Document Information Panel in word can then be used to populate the metadata when creating Microsoft Office documents. The Document Information Panel can be further customized using InfoPath either for the entire document library or for a particular content type.



As you key in data via the document information panel, validation is performed on the database to ensure the data you entered exists.



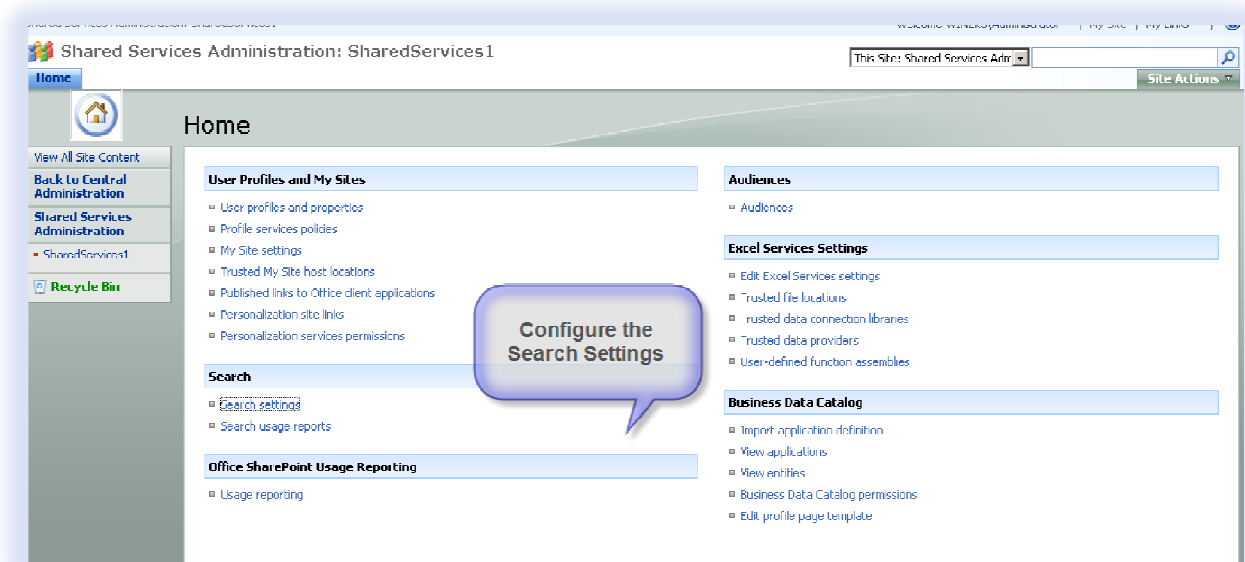
Once you have chosen valid data, the additional columns you added fetch data from the database and displays/stores them as meta data against the document.



## SEARCHING BDC DATA

Using the MOSS search Site, you can search for People (based on a profile), Documents, List Items, Web Pages etc. You can also configure the Search so that it includes our BDC data. We can configure the Index Crawling Schedule, the Properties we want to search, and also configure the Search Tab and the Search Results page.

First of all, we need to Add the Content Source telling SharePoint what to index. To do this, Navigate to SharePoint Central Administration, Navigate to your Shared Services Provider and Click Search Settings.



Provide a name for your content source such as Products, specify that we are searching for BDC data, choose the application that you want to crawl, and then setup a crawling Schedule. Choose to carry out an initial Full Crawl and wait for the Crawl to complete.

Shared Services Administration: SharedServices1 > Search Settings > Content Sources > Add Content Source

## Add Content Source

Use this page to add a content source.

\* Indicates a required field.

**Name**  
Type a name to describe this content source.

**Content Source Type**  
Select what type of content will be crawled.  
Note: This cannot be changed after this content source is created since it is not supported to edit it.

**Applications**  
Select whether to crawl all applications in the Business Data Catalog, or include only selected applications in this content source.  
An application may be included in only one Business Data content source.

**Crawl Schedules**  
Select the crawl schedules for this content source.

**Name:** \*  
Products

Select the type of content to be crawled:

- SharePoint Sites
- Web Sites
- File Shares
- Exchange Public Folders
- Business Data

Select the Business Data Catalog applications to be crawled:

- Crawl entire Business Data Catalog
- Crawl selected applications:
  - AdventureWorksInstance
  - AdventureWorksInstanceOrders
  - AdventureWorksOutstandingDepts
  - AdventureWorksProducts
  - AdventureWorksSalesPerson

Select the schedule that this should be a part of:

Full crawl  
None

Incremental crawl  
None

Start full crawl of this content source

Start Full Crawl

Shared Services Administration: SharedServices1 > Search Settings > Content Sources

## Manage Content Sources

Use this page to add, edit, or delete content sources, and to manage crawls.

[New Content Source](#) | [Refresh](#)

Type	Name	Status	Next Full Crawl	Next Incremental Crawl
	Local Office SharePoint Server sites	Idle	None	7/13/2007 10:21:00 PM
	Products	Crawling Full	7/13/2007 10:21:00 PM	7/13/2007 10:21:00 PM

Once the data has finished Crawling, you can map your properties that you want to search. You need to ensure that you have created an ID Enumerator in the application definition file for the search to be successful.

Shared Services Administration: SharedServices1 > Search Settings

## Configure Search Settings

**Crawl Settings**

Indexing status:	Idle
Items in index:	400
Errors in log:	1
Content sources:	2 defined (Local Office SharePoint Server sites, Products)
Crawl rules:	0 defined
Default content access account:	NT AUTHORITY\LOCAL SERVICE
Managed properties:	127 defined
Search alerts status:	Active
Propagation status:	Propagation not required

- Content sources and crawl schedules
- Crawl rules
- File types
- Crawl logs
- Default content access account
- Metadata property mappings**
- Server name mappings
- Search-based alerts
- Search result removal
- Reset all crawled content

**Scopes**

Scopes:	2 defined (People, All Sites)
Update status:	Idle

**Metadata  
Property  
Mappings**

Once we have mapped our properties that we want to search, we can add rules to our search scope. We add the Products search scope in the rule and specify to include it in the search.



Shared Services Administration: SharedServices1 > Search Settings > Scopes

## View Scopes

Use this page to view and manage search scopes. The order in which the search scopes appear in this list is the order in which they will appear in the search scope list next to the Search box.

[New Scope](#) | [Refresh](#) | 1 -

Title	Update Status	Items
<b>Shared (3)</b>		
<a href="#">People</a>	Ready	1
<a href="#">All Sites</a>	Ready	221
<a href="#">Products</a>	Empty - Add rules	empty
<a href="#">http://win2k3:36609/ssp/admin (0)</a>		
<a href="#">http://win2k3/ (0)</a>		

**Create a new Search Scope**

Shared Services Administration: SharedServices1 > Search Settings > Scopes > Scope Properties and Rules > Add Scope Rule

## Add Scope Rule

**Scope Rule Type**  
Scope rules define what is in or not in a scope. Use different types of rules to match items in various ways.

**Content Source**

**Behavior**  
Decide how this rule should be applied to the overall scope. The scope-wide filter is used when combining the items matching all rules to determine what is in the scopes overall.

Web Address (http://server/site)  
 Property Query (Author = John Doe)  
 Content Source  
 All Content

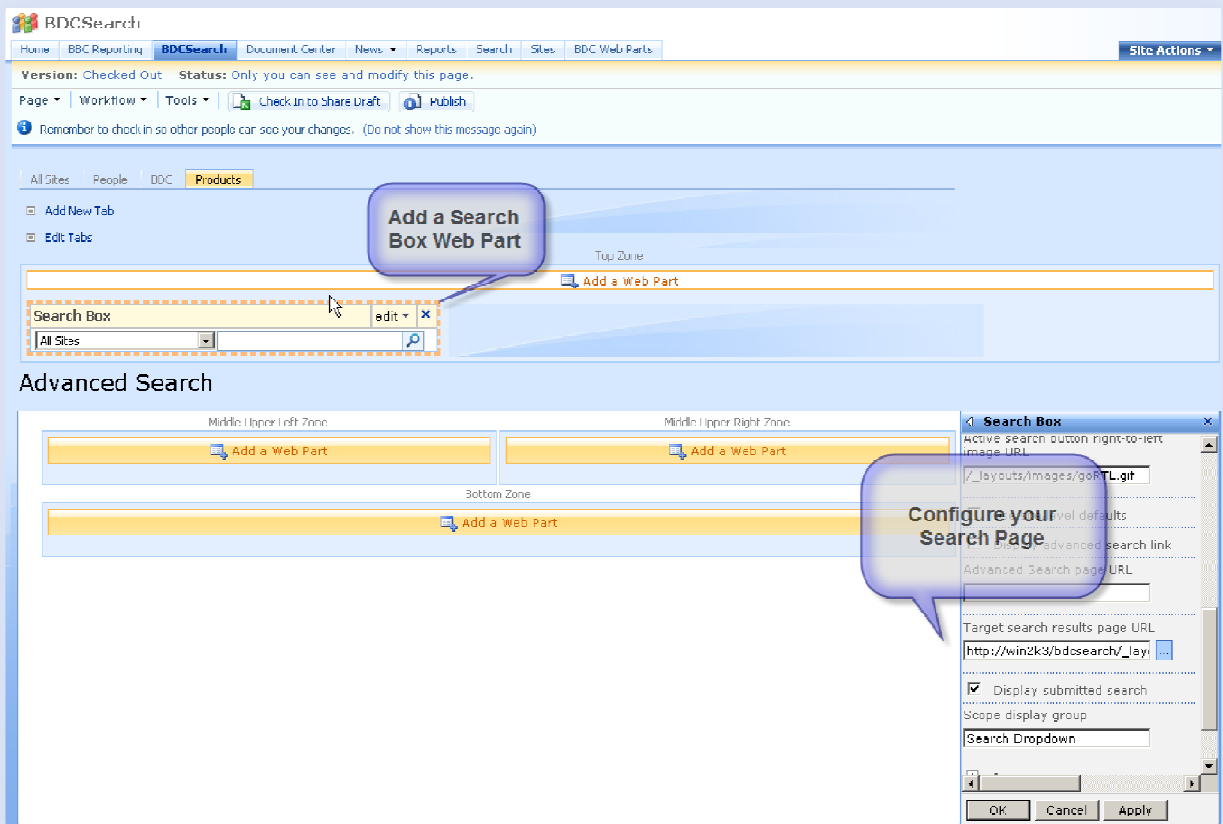
Products

Include - Any item that matches this rule will be included, unless the item is excluded by another rule.  
 Require - Every item in the scope must match this rule.  
 Exclude - Items matching this rule will be excluded from the scope.

OK Cancel

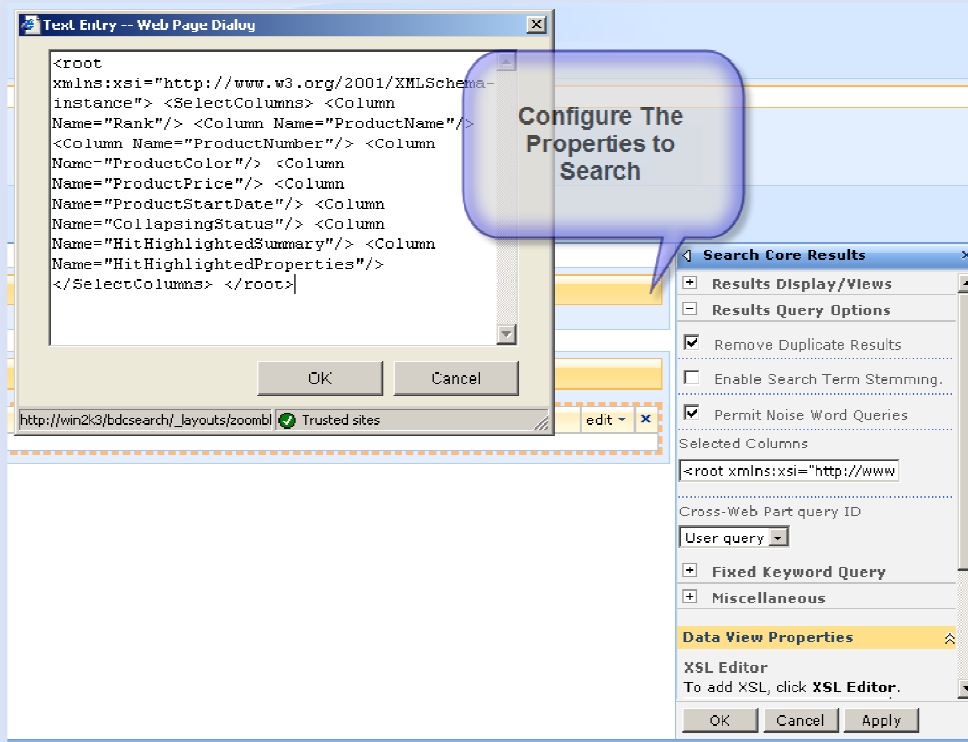
**Add the Search Scope Rule**

We are then ready to build our Search Site. We could just use the existing Search Site, however, that will include Documents, People, and List Items in the results. We can add our own Tab to the search site, our own search results page, and our own web parts. This will ensure that we only see our Product information in the search results pane.

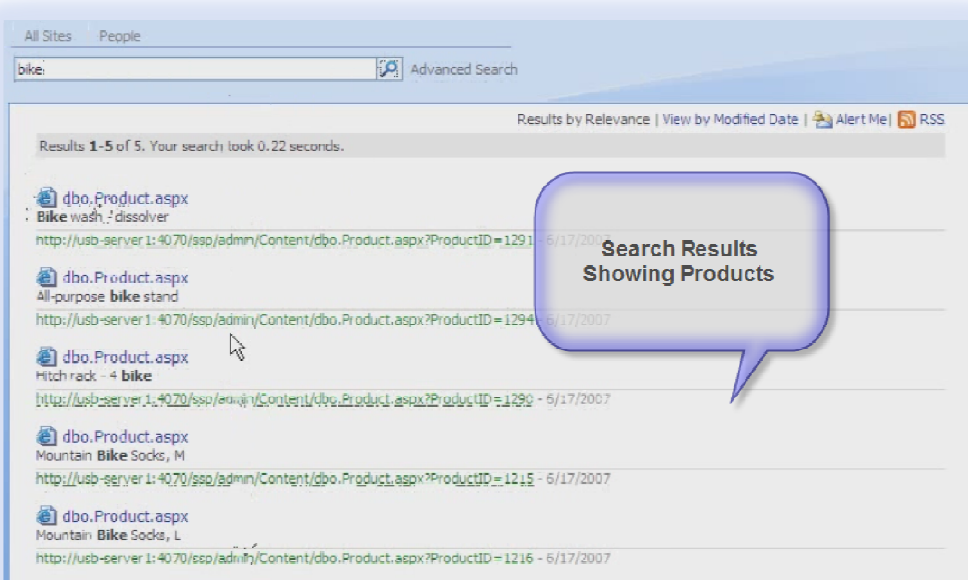


Create a new Search Page, and Add a new Tab to the Search Site pointing to the new page. You can also configure the web part to point to the Search Result page.

Then add another Web Part. We need to add a Search Core Results Web Part. We need to configure that to search the properties that we mapped earlier.



Once that is complete, you are ready to search you BDC Data.



Brett Lonsdale

[brett@combined-knowledge.com](mailto:brett@combined-knowledge.com)