

# Elegrity Filtered List Lookup Web Service

Tuesday, December 13, 2010

## Installation:

To install the Elegrity Filtered List Lookup Web Service run the "setup.exe" in the Administrative mode on the machine running Microsoft Office SharePoint Services 2007 (MOSS 2007). The user must have the privileges to modify the system environment in order to successfully install and configure the service. The install might require a reset of Internet Information Services (IIS). Please follow the steps below to verify if the web service was installed and configured successfully.

1. To verify the successful installation and configuration of the service, navigate to any portion of the SharePoint portal site and append the following to the request url: `/_vti_bin/ElegrityListService.asmx?WSDL`.  
The expected result of the request described above is an XML document representation of the Web Service Description Language. If the request described above results in the error, please see the troubleshooting notes section for resolving installation issues.
2. To verify the correct configuration of the web service:
  - i. With Microsoft InfoPath open, create new data connection of type Receive specifying Web Service as the data source for the connection.
  - ii. Input the url from step 1, and click next. If configured correctly, the next screen will show the operations defined for the web service.
  - iii. Select GetListItems operation from the list and click next.
  - iv. Input the fields of the web service parameters as described in the Elegrity Service Usage section.
  - v. Complete the connection setup. The Elegrity Service was installed successfully if no errors were display at the time of the data connection configuration.

NOTE: In order to verify web service output and formatting configuration, one should use either a tool similar to Fiddler for monitoring of HTTP packets or through Microsoft InfoPath controls setup.

The configuration file provided with the installation is a placeholder for actual Lists configuration. To configure the Web Service to suit the needs of your InfoPath form, please read through the Configuration section of this manual.

## Usage:

The Elegrity Filtered List Lookup Web Service provides an InfoPath friendly service wrapper around the standard SharePoint Lists web service GetListItems method. The major drawback of the standard List retrieval functionality of Browser-based forms in Microsoft InfoPath 2007 is inability to specify a filter parameter to return a filtered result set from a SharePoint List. The Web Service wrapper provides a method to be called directly by InfoPath form controls without the need of writing code-behind functions for standard web service calls. In addition to returning a filtered result set, the Web Service can be setup to perform conditional formatting on specified columns including number format operations and string manipulations.

The Web Service uses CAML (Collaborative Application Markup Language) notation for filter based parameters to SharePoint web services. CAML is an xml based markup language created by Microsoft for SharePoint technologies. CAML queries are passed to SharePoint web services in order to accomplish data retrieval based on specific criteria. As InfoPath functionality can only utilize web services which take string parameters, the Web Service enables query based data retrieval directly from InfoPath form by exposing the web service method expecting string parameters in xml representation.

The table below provides description of parameters to the GetListItems method and associated examples of values to be passed from the InfoPath form:

Field Name	Field Description	Example Value
<i>listLocation</i>	Location of the list within SharePoint portal. This should be a path to the SharePoint site (or root URI) containing the list to be queried. The string representation must end with forward slash ("/").	<a href="http://test/site1/">http://test/site1/</a>
<i>listName</i>	Name of the list to be queried. The list name must be exactly the way it is displayed within the SharePoint site.	MyCustomList1

<i>strViewName</i>	Name of the view to query and display. Has the ability to retrieve items directly from a view within the specified list. If this parameter is not specified, the default view is used.	MyView1
<i>strXmlQuery</i>	<p>XML query in CAML notation. List item must match the query in order to be returned to InfoPath form.</p> <p>NOTE: SharePoint follows the internal naming convention of List items when returning items using SharePoint services. The names of columns returned within the data set may be different from the names seen within the SharePoint portal. In order to view all the columns available to specified list, omit this parameter to return a complete dataset from the web service and examine the result to find the matching column name to be used in query.</p>	<pre>&lt;Where&gt;&lt;Eq&gt;&lt;FieldRef Name='ID'/&gt;&lt;Value Type='Number'&gt;5470&lt;/Value&gt;&lt;/Eq&gt;&lt;/Where&gt;</pre>
<i>strXmlViewFields</i>	CAML markup specifying which fields to return from the list query. This parameter is used to limit the number of columns returned per list item, therefore improving the	<pre>&lt;FieldRef Name='ContactID'/&gt;\</pre>

	<p>network traffic to the InfoPath form.</p> <p>NOTE: SharePoint follows the internal naming convention of List items when returning items using SharePoint services. The names of columns returned within the data set may be different from the names seen within the SharePoint portal. In order to view all the columns available to specified list, omit this parameter to return complete dataset from the web service and examine the result to find the matching column name to be used in query.</p>	
<i>rowLimit</i>	Integer value specifying the maximum number of rows to return. If not specified, the web service call defaults to 50 maximum rows.	500
<i>strXmlQueryOptions</i>	Additional query options described in the Microsoft documentation.	<QueryOptions><IncludeMandatoryColumns>FALSE</IncludeMandatoryColumns><DateInUtc>TRUE</DateInUtc></QueryOptions>

**Configuration:**

The Web Service uses the configuration file located within the installation folder. The configuration directs the Web Service how to handle the formatting of specified fields. To provide flexibility in field formatting, the Web Service configuration allows different

settings for each list specified. Such separation also provides the ability to apply different formatting to the fields of same name contained within different SharePoint lists.

The configuration file is broken down into global options (effecting every call to the Web Service) and List-based configuration (Specific to lists being used).

The table below provides a description and example values for the global options within the <options></options> element of the configuration file:

Parameter	Description	Example Value
<i>CacheConfig</i>	Directs the Web Service to either cache List configuration or parse it on every call to the web service. The read and parse operation is an expensive disk operation to be performed on a large file each time the service is called. By directing the Web Service to cache the configuration, the time to read and parse xml configuration is skipped and the copy is taken from the memory cache. The configuration is stored in cache for the period of one day before being automatically refreshed. If an immediate refresh is required, set the value of this property to False to force the Web Service to read the new configuration.	true

The list configuration is stored in the ListConfig element tag. The ListConfig element contains multiple elements of type "attr", each representing a column of the list and formatting options to be used on this column. If the column specified is not found within the result set returned, no actions on that column are taken. An example of a "ListConfig" entry is shown below:

```
<ListConfig Site="http://demo/site1/" ListName="History">  
  <attr Name="ows_Performed_x0020_By" StripSystemInfo="true" FormatRegular="false" FormatRegularString=""  
  FormatExtended="false" FormatExtendedString="{0:U}"/>  
  <attr Name="ows_Editor" StripSystemInfo="true" />
```

```

    <attr Name="ows_ID" StripSystemInfo="false" FormatRegular="true" FormatRegularString="ID: {0:c}" FormatAs="INT" />
    <attr Name="ows_InfoPath_x0020_Form_x0020_ID" FormatRegular="true" FormatRegularString="ID : {0}"
    FormatAs="INT" />
</ListConfig>

```

The table below provides a description and an example values for the List configuration:

Parameter (Attributes)	Description	Example Value
<i>Site (ListConfig)</i>	<p>Specifies the site for which the query must run in order to match the rule set defined within this element. This provides the ability to parse two lists of the same name located within different SharePoint sites.</p> <p>NOTE: The site must include the trailing "/" character.</p>	<a href="http://demo/site1/">http://demo/site1/</a>
<i>ListItemName (ListConfig)</i>	<p>Specifies the name of the List for which this element will configure parameters.</p> <p>NOTE: Configuration elements with identical List name will not result in the exception. The Web Service will use the latest configuration element for the name specified, ignoring the duplicates configured above it.</p>	History
<i>Name (Attr)</i>	<p>Specifies the exact name of the field that is expected to be returned by the web service call. The value specified must exactly match the name of the field to be returned otherwise the attribute is ignored.</p>	ows_Performed_x0020_By

	<p>NOTE: SharePoint returns a dataset in xml format similar to the ListConfig element, where each element within the element of type "Row" represents a column of the list returned. The name specified by the Name attribute of the Web Service configuration is actually the name of the attribute of element Row returned by the SharePoint GetListItems web service.</p>	
<i>StripSystemInfo (Attr)</i>	<p>If set to "true", the Web Service will attempt to strip the system identifier of the field if present. The system identifier is separated from the column data with hash sign (#). A raw value of the field with system information might look like this:</p> <p style="text-align: center;">154;#John Doe</p> <p>The system field will be striped to become a field with actual column value:</p> <p style="text-align: center;">John Doe</p> <p>There is no negative impact to leaving this "true" for all the Attr elements configured within the ListConfig element.</p>	true
<i>FormatRegular (Attr)</i>	<p>If set to "true", the Web Service will attempt to format this field based on the format specified in the FormatRegularString attribute (described below).</p>	true

<i>FormatRegularString (Attr)</i>	Specifies the format to be used if the FormatRegular attribute for this column is set to "true". The format notation follows the String.Format Method notation within the .NET framework, with the exception that there is always a single item (the value itself) to be formatted as opposed to multiple parameters.	{0,5:N1}°F  {0:C4}
<i>FormatAs (Attr)</i>	<p>Defines the type of the value to be used in formatting. At this point there are three formatting values supported:</p> <p>INT, DOUBLE, STRING (default). The format string specified by FormatRegularString must take a valid input specified by FormatAs parameter in order to successfully format the value received from the web service. For example, {0:c} prints the currency in format (\$###). To correctly format the currency FormatAs must be set to type DOUBLE or INTEGER (though integer removes the ability to format the cents values).</p>	INT / DOUBLE / STRING
<i>FormatExtended (Attr)</i>	<p>Turns on the extended format options not found within the standard .NET format functionality.</p> <p>This feature can be used in conjunction with the <i>FormatRegular attribute</i>.</p>	true
<i>FormatExtendedString (Attr)</i>	<p>Currently two formatting options are supported for extended formatting:</p> <p>{0:U} - Transform the string to upper case {0:L} - Transform the string to lower case</p>	{0:U} {0:L}

	<p>This formatting is useful in comparison of strings as InfoPath performs a case sensitive comparison. For example, comparison of Individual == INDIVIDUAL will be false in an InfoPath form. However, if enabled, the extended format feature could translate the field Individual into upper case making a comparison to INDIVIDUAL true, regardless of the case.</p>	
--	--	--

## Troubleshooting

Problem: Error occurs when trying to navigate to the Web Service WSDL.

Solution: Verify the correct files are added to the ISAPI folder of web service extensions. This folder should normally be located under the system driver Common File folder (E.g. C:\Program Files\Common Files\Microsoft Shared\web server extensions\12\ISAPI). The folder should contain the following 3 files:

ElegrityListServicewsdl.aspx  
ElegrityListServicedisco.aspx  
ElegrityListService.asmx

If ISAPI folder does not contain the files above, locate the ISAPI directory for your installation and manually copy the files located in the "ws" folder under the root installation directory of the Elegrity Service to ISAPI folder.

Problem: While configuring the Web Service data connection in Microsoft InfoPath, you receive an error.

Solution: Review the error description as it may contain useful information related to misconfiguration of Elegrity Service configuration file. Verify the Elegrity Service assembly installation within system's assembly cache by navigating to %windir%\assembly folder and finding the ElegrityService assembly in the list of installed assemblies.

Below is a table of common errors that might be received during the configuration of the Web Service within the InfoPath form.

<b>Error</b>	<b>Description</b>
<p>The SOAP response indicates that an error occurred on the server</p> <p>ElegrityService failed with exception: Exception of type 'Microsoft.SharePoint.SoapServer.SoapServerException' was thrown.</p>	<p>Verify the input parameters to the Web Service. Providing an incorrectly formatted URL will result in this error. Refer to the Usage section of this manual for the format of parameters expected by the service.</p> <p>This exception might also indicate the absence of a data source (List, View) that is being requested by the service.</p>
<p>The SOAP response indicates that an error occurred on the server:</p> <p>ElegrityService failed with exception: Could not find file 'PATH_TO_FILE'</p>	<p>As specified with the exception message, the path to the configuration was not found. Verify that the location referred to by PATH_TO_FILE is valid and contains the ElegrityServiceConfig.xml file. This exception might be result of an invalid installation, in which case the path to configuration file must be manually configured using the Environment variable "ELEGRITY_SERVICE_CONFIG". It is also possible to retry the installation to overwrite existing configurations and correct the issue.</p>
<p>The SOAP response indicates that an error occurred on the server:</p> <p>ElegrityService failed with exception: Failed to load configuration file specified by ELEGRITY_SERVICE_CONFIG environment variable.</p>	<p>Verify that environment variable with name ELEGRITY_SERVICE_CONFIG exists and contains a path to ElegrityServiceConfig.xml file.</p>
<p>The SOAP response indicates that an error occurred on the server:</p> <p>ElegrityService failed with exception: Invalid configuration file specified - option with name 'NAME' not found.</p>	<p>This error is due to an incorrect configuration within ElegrityServiceConfig.xml file. Verify the configuration file or replace the file with ElegrityServiceConfig_backup.xml file located in the installation directory.</p>

You might also encounter an error specific to formatting of the configuration attributes. Review the error message and refer to documentation for the .NET String.Format Method for more information and examples on formatting the values of type string, double, and integer.