Accelerator – External Data Form (v7.1)

User Guide

Contents

1.0   Introduction

The Front Office External Data Form accelerator can be configured to generate multiple External List Boxes from any Microsoft SQL Server data source.

This version supports External List Boxes for version 7.1 upwards. They are a more powerful version of External List Boxes found in previous versions that offer searching, filters and paging functionality.

2.0   Installation

The application is a Microsoft .NET Web application (Version 4), requiring IIS 5.1+, .Net 4 can be downloaded here if required: http://www.microsoft.com/download/en/details.aspx?id=17718

Note: If installing on IIS7 ensure that Role Services for 'IIS 6 Management Compatibility' is checked in Server Manager.

To install:

- log in on the target installation server with administrator privileges
- Extract the files from the zip file into a temporary folder
- Using Windows explorer open the temporary folder, then run setup.exe
- Click the next button
- Set the virtual directory (default AcceleratorExternalDataForm71)
- Set the target application pool (ASP.NET v4 on IIS7). Create a new app pool on server 2003/XP.
- Click the next button, then again to start the installation
  - Grant the NETWORK SERVICE account (or account that the selected application pool runs under) "Full Control" rights to: \inetpub\wwwroot\<selected virtual directory>\logs

3.0 Configuration

3.1 Adapter configuration

On the server where the adapter has been installed, edit the adapters web.config (for default installations this will be in inetpub\wwwroot\AcceleratorExternalDataForm71) and update the following settings:

- DapiURL – Should be set to the url of your Front Office installations Public Web Service (this can be found on the Admin/Configuration Check window)
- DapiUser & DapiPassword – set these if your Front Office Api has been secured with basic authentication.

If required the appSettings section can be encrypted by following the instructions here:

http://msdn.microsoft.com/en-us/library/vstudio/zhhddkxy(v=vs.100).aspx

3.2 Edit the xml configuration file

Using an XML editor open up the configuration file found in \inetpub\wwwroot\<selected virtual directory>\xml\elbconfig.xml.

Repeat (and edit) the following block of xml for each external list box required:

```xml
<ExternalListBoxConfig>
        <Code>USERS</Code>
        <SQLStatement> Select UserId,UserName from users where useraccountstatus='A' and
userName like '%[SEARCH]%' and AccessProfileID like '%[AccessProfile]%'</SQLStatement>
        <SQLConnectionString>Data Source=BIOMNIVDEMO;Initial
Catalog=DIRECTA;Integrated Security=False;User
Id=webdirecta;Password=directaweb</SQLConnectionString>
        <PagingEnabled>true</PagingEnabled>
        <SearchingEnabled>true</SearchingEnabled>
        <UserFilteringEnabled>true</UserFilteringEnabled>
        <UserFilters>
                <Filter id="AccessProfile" label="Access Profile" defaultValue="DEF"
sqlStatement="select AccessProfileName as DisplayName, AccessProfileId as Value from
AccessProfile"></Filter>
        </UserFilters>
        <KeyColumnName>UserId</KeyColumnName>
        <Columns>
                <Column>
                        <Name>UserId</Name>
                        <Heading>User ID</Heading>
                        <Width>100</Width>
                        <Type>String</Type>
                        <ColumnDataSource>UserId</ColumnDataSource>
                </Column>
                <Column>
                        <Name>UserName</Name>
                        <Heading>User Name</Heading>
                        <Width>300</Width>
                        <Type>String</Type>
                        <ColumnDataSource>UserName</ColumnDataSource>
                </Column>
        </Columns>
</ExternalListBoxConfig>
```

Element	Description
<Code>	A unique code for the external list box
<InsertSQLStatement>	Optional SQL insert statement
<SQLStatement>	The SQL statement to run for the list box (see below)
<SQLConnectionString>	The SQL connection string (.net format)
<PagingEnabled>	Set to true if paging required, false if not
<SearchingEnabled>	Set to true if search box is required, false if not

`<UserFilteringEnabled>`	Set to true if filtering is required, false if not
`<UserFilters>`	Include this element if filtering is enabled
`<Filter id="AccessProfile" label="Access Profile" defaultValue="%">`	For manual configuration of filters repeat for each filter required. Set the id and display name (label) and default value of the filter. The **id** of the filter should be used in the SQL statement (contained within square brackets)
`<Filter id="AccessProfile" label="Access Profile" defaultValue="DEF" sqlStatement="select AccessProfileName as DisplayName, AccessProfileId as Value from AccessProfile"></Filter>`	For configuration of a filter from a SQL statement repeat for each filter required. Set the id and display name (label) and default value of the filter. The SQL statement should return 2 columns of data (DisplayName and Value) and the default value should always be present in the result set
`<Option displayName="All">`	Repeat for each filter option required, set the display name and the actual value of the filter that will be passed to the SQL statement.
`<KeyColumnName>`	The name of the column to use as the ID
`<Columns>`	Container for multiple column elements
`<Column>`	Container for column elements
`<Name>`	The name of the column
`<Heading>`	The heading of the column
`<Width>`	The width of the column (in pixels)
`<Type>`	The type of the column (String or Image)
`<ColumnDataSource>`	The data source of the column (must match a field name returned from the SQL statement)

Depending on the functionality configured the SQL statement needs to contain the search string and filters, e.g:

Select UserId,UserName from users where useraccountstatus='A' and userName like '%[SEARCH]%' and AccessProfileID like '%[AccessProfile]%'

[SEARCH] will be replaced with the search string passed from Front Office

[AccessProfile] will be replaced with the value of the filter with id of AccessProfile

The SQL statement (and insert SQL statement) can also contain xpath statements that run over the xml that is passed to the list box by Front Office (xsd schema is here: \inetpub\wwwroot\<selected virtual directory>\schemas\Flexifield.xsd), for example:

Select UserId, UserName from Users where userid like
'%{n:bXML/n:Parameters/n:Request/n:RequestHeader/n:UserFields/n:UserField[n:FieldCode='USERFILT
ER']/n:SingleValueType/n:Value}%'

This will return users that have a user id containing the value of the USERFILTER form field. Each XML
element must use a namespace prefix of "n:".

The xpath statement needs to be contained within curly brackets { }. Multiple xpath statements per SQL
statement are supported.

There are 2 replacement parameters that can be used in the SQL / XPATH statements:

[SEARCH] – This will be replaced by the external list boxes search string passed from Front Office

[CULTURE] – This will be replaced by the requesting users culture code

3.3 Front Office Configuration

3.3.1 Create External System Definition / Field(s)

Create a new external system definition as below (adjusting server / virtual directory as per install):

For each external list box configured create a new request field:

Important: Set the Parameters field to the code specified in the configuration xml

3.3.2 Assign Request Field(s) To Request Type(s)

Build the new request fields into the desired request types. In use the field will look something like this:

When the request is viewed, fields will be displayed like this:

3.3.3 Fulfilment Data Format

When the request is sent for fulfilment the xml for the user field(s) will be constructed like this:

```
<UserField>
  <FieldCode>PICKUSER</FieldCode>
  <FieldLabel>Users</FieldLabel>
  <FieldValue>
    &lt;Rows XsdVersion="5" KeyColumnName="UserId" DisplayColumnName="UserId"&gt;
    &lt;Row&gt;
    &lt;UserId&gt;ADMIN&lt;/UserId&gt;
    &lt;UserName&gt;System Administrator&lt;/UserName&gt;
    &lt;/Row&gt;
    &lt;Row&gt;
    &lt;UserId&gt;amanager&lt;/UserId&gt;
    &lt;UserName&gt;Andrew Manager&lt;/UserName&gt;
    &lt;/Row&gt;
    &lt;/Rows&gt;
  </FieldValue>
</UserField>
```