How To... Dynamically Set Data Providers in SAP NetWeaver BW Planning Applications

Applicable Releases:
SAP NetWeaver 7.0

IT Practice:
Business Information Management

IT Scenario:
Business Planning and Analytical Services

Version 2.0
June 2009
## Document History

<table>
<thead>
<tr>
<th>Document Version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.00</td>
<td>Small correction in VBA coding</td>
</tr>
<tr>
<td>1.00</td>
<td>First official release of this guide</td>
</tr>
</tbody>
</table>
### Typographic Conventions

<table>
<thead>
<tr>
<th>Type Style</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example Text</td>
<td>Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation</td>
</tr>
<tr>
<td>Example Text</td>
<td>Emphasized words or phrases in body text, graphic titles, and table titles</td>
</tr>
<tr>
<td>Example text</td>
<td>File and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.</td>
</tr>
<tr>
<td>Example text</td>
<td>User entry texts. These are words or characters that you enter in the system exactly as they appear in the documentation.</td>
</tr>
<tr>
<td>&lt;Example text&gt;</td>
<td>Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.</td>
</tr>
<tr>
<td>EXAMPLE TEXT</td>
<td>Keys on the keyboard, for example, F2 or ENTER.</td>
</tr>
</tbody>
</table>

### Icons

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🔴</td>
<td>Caution</td>
</tr>
<tr>
<td>🔥</td>
<td>Note or Important</td>
</tr>
<tr>
<td>⬤</td>
<td>Example</td>
</tr>
<tr>
<td>🛠️</td>
<td>Recommendation or Tip</td>
</tr>
</tbody>
</table>
Table of Contents

1. Business Scenario............................................................................................................... 1
2. Background Information..................................................................................................... 1
3. Prerequisites ........................................................................................................................ 1
4. Step-by-Step Procedure...................................................................................................... 2
   4.1 Dynamic Data Providers in Web Applications .............................................................. 2
      4.1.1 Create the Data Providers ............................................................................... 2
      4.1.2 Create a Tab Strip Container and the Analysis Items ..................................... 3
      4.1.3 Set the Command on Activation ...................................................................... 4
      4.1.4 Set the Deactivation Action.............................................................................. 7
   4.2 Dynamic Data Providers in BEx Analyzer .................................................................... 9
      4.2.1 Create Worksheets and Add Queries .............................................................. 9
      4.2.2 Create a Button Setting the Data Provider .................................................... 10
      4.2.3 Trigger the Button after a Worksheet Change............................................... 11
1. Business Scenario

It is very common that within a planning application several plan queries are used. For usability and performance reasons these queries are usually placed on different tab strips (BEx Web) or Worksheets (BEx Analyzer).

An easy example could be a revenue planning where the end user can enter the prices, the sold volumes, and can view the calculated revenue. The end user starts with entering the prices on the first tab strip and moves then on to the second tab strip with the sales volumes and the calculated revenue.

In this How to Paper we will show how the performance of such planning applications can be improved.

2. Background Information

When a planning application with several data providers is opened, all data providers are initiated. The underlying queries are read (not the data!) and - if necessary – variables are processed. In the BEx Analyzer also the data for all Data Providers is retrieved from the BI Server no matter if the data is displayed on the initial Worksheet or other Worksheets.

In this How to Paper we will show, how Data Providers can be dynamically filled only when they are needed. This improves the performance when the planning application is started as only the data providers visible on the first screen are instantiated. Once the user switches to the next tab strip or worksheet the necessary Data Providers will be loaded on request. [Note that a tab strip change in a planning application using this technique will be slower than in an application loading all Data Providers when starting.]

This technique is especially helpful in planning applications with many tab strips/Worksheets where not every one of them will be used in each planning session (e.g. sheets showing comparison reports, adjustment layouts etc).

Alternatively to the concept introduced in this paper, the techniques can also be used to create a similar solution where different layouts can be toggled by pushing a button.

3. Prerequisites

There are no special prerequisites for this How to Paper.

The OSS Note 1128508 describes the introduced concept as well.
4. **Step-by-Step Procedure**

4.1 **Dynamic Data Providers in Web Applications**

4.1.1 **Create the Data Providers**

1. Open your BEx Web Application Designer
2. Create a new Web Application
3. Create a new Data Provider and enter the query which should be visible at the first tab strip

   ![Create the Data Providers](image1)

4. Create an additional Data Provider without specifying a query

   ![Create an additional Data Provider](image2)
4.1.2 Create a Tab Strip Container and the Analysis Items

1. Add a new Tab Pages Item

2. Add two Analysis Items

3. Mark the Analysis_Item_1 and assign in the Properties section the Data Provider 1 (DP_1)

4. Mark the Analysis_Item_2 and assign in the Properties section the Data Provider 1 (DP_2)
4.1.3 Set the Command on Activation

1. There is the possibility to add commands during the activation of a tab strip. This possibility is used to set the dynamically the Data Provider on the second tab strip.

   Mark the TABSTRIP_CONTAINER_ITEM_1 and go the Properties section and choose Web Item Parameters

2. Edit the parameters of the first tab

   ![Web Item Parameters](image)

3. Add a caption in the appearing screen and the subordinate Web Item is the ANALYSIS_ITEM_1 and press OK.

   ![Edit Parameter](image)
4. Edit the parameters of the second tab

![Properties](image)

5. Add a caption in the appearing screen and assign the subordinate Web Item ANALYSIS_ITEM_2

6. Add a command for the Activation Action

![Edit Parameter](image)
7. Choose the command SET_DATA_PROVIDER_PARAMETERS and press next.


9. Choose the data provider which should be displayed at the second tab and press OK.
4.1.4 Set the Deactivation Action

1. If you use the web application as it is the command on activation is executed any time the tab strip is changed. Thus the Data Provider is instantiated any time a tab strip is activated. This means that all navigation and filter steps are lost. To avoid the situation a deactivation action is used in order to change the activation action and thus make sure that each Data Provider is only instantiated once.

   In Edit Parameter for the second tab strip choose Deactivation Action

2. Choose the command SET_ITEM_PARAMETERS

3. Web Item Type Tab Pages and Target Web Item TABSTRIP_CONTAINER_ITEM_1

   Determine the Command Target (Data Provider or Web Item)
   
   Set Web Item Parameters [SET_ITEM_PARAMETERS]}

   Command Target
   
   Target Web Item TABSTRIP_CONTAINER_ITEM_1
How To... Dynamically Set Data Providers in SAP NetWeaver BW Planning Applications

4. Choose the second Tab strip

5. Make no changes because the default parameters are sufficient

You can now save and test your planning application in the web
4.2 Dynamic Data Providers in BEx Analyzer

4.2.1 Create Worksheets and Add Queries

1. Open your BEx Analyzer and create a new Workbook. Make sure the Workbook contains several Worksheets.
2. Open an Analysis View
3. Insert an Analysis grid in first worksheet
4. Assign a Data Provider
5. Insert Table in second Worksheet
6. Do not assign a Data Provider

![Image of Excel interface showing Data Provider settings]

4.2.2 Create a Button Setting the Data Provider

In the web application we have the possibility to add directly a command in the tab strip item itself. In the BEx Analyzer this features do not exists. We will have to combine Excel techniques with BEx Analyzer functionality: we will use a VBA macro that is executed on Worksheet change and then triggers a macro, that sets a query in the Data Provider. As we want to use the Command Wizard we create a button setting the Data Provider and will then simulated a button click by the VBA macro.

Before the Workbook is saved make sure that the relevant worksheet is chosen and the Data Providers which are not at the active Worksheet are not assigned to a query.

1. Add Button in third Worksheet right mouse click “Properties”
2. Select Command Type “Data Provider Specific Command” and choose Data Provider “DP_2” and press “Next”

![Image of VBA macro settings]
3. Select in the “Assign Query/Query View” choose the query which you want to see on the second Worksheet and press next.

Your screen should afterwards look like this with your InfoCube and Query

4.2.3 Trigger the Button after a Worksheet Change

1. Save your Workbook. The BEx Analyzer will automatically create a macro that is executed once the button you have just created is called.

2. Go to Visual Basic Editor (Tools → Macro → Visual Basic Editor)

3. Go to Sheet 4 (Sheet 3) and check that the macro for the button has been created. It should look like the following (the name of the button might be different)

```vba
Public Sub BUTTON_3_Click()
    Dim BEx1 As Object
    Set BEx1 = Application.Run("BExAnalyzer.xla!GetBEx")
    Call BEx1.RaiseButtonClick(Parent.Name, "BUTTON_3")
End Sub
```

4. Go to ThisWorkbook and add following code

```vba
Dim lDp2Set As String

Sub Workbook_SheetActivate(ByVal Sh As Object)
```

June 2009
If Sh.Name = "Sheet2" Then
    If lDp2Set <> "X" Then
        lDp2Set = "X"
        Call Sheet4.BUTTON_3_Click
    End If
End If
End Sub

Save your Workbook. You can now test your application. Remember: when saving again to make sure that the Data Providers on the Worksheets that are not visible when starting the application are still empty.
www.sdn.sap.com/irj/sdn/howtoguides