



# 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

© Copyright 2009 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice.

These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

These materials are provided "as is" without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall not be liable for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials.

SAP does not warrant the accuracy or completeness of the information, text, graphics, links or other items contained within these materials. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third party web pages nor provide any warranty whatsoever relating to third party web pages.

SAP NetWeaver "How-to" Guides are intended to simplify the product implementation. While specific product features and procedures typically are explained in a practical business context, it is not implied that those features and procedures are the only approach in solving a specific business problem using SAP NetWeaver. Should you wish to receive additional information, clarification or support, please refer to SAP Consulting.

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.

#### Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressly prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.

## Document History

<b>Document Version</b>	<b>Description</b>
2.00	Small correction in VBA coding
1.00	First official release of this guide

## Typographic Conventions

Type Style	Description
<i>Example Text</i>	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
<b>Example text</b>	Emphasized words or phrases in body text, graphic titles, and table titles
Example text	File and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
<b>Example text</b>	User entry texts. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example text>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
EXAMPLE TEXT	Keys on the keyboard, for example, F2 or ENTER.

## Icons

Icon	Description
	Caution
	Note or Important
	Example
	Recommendation or Tip

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.

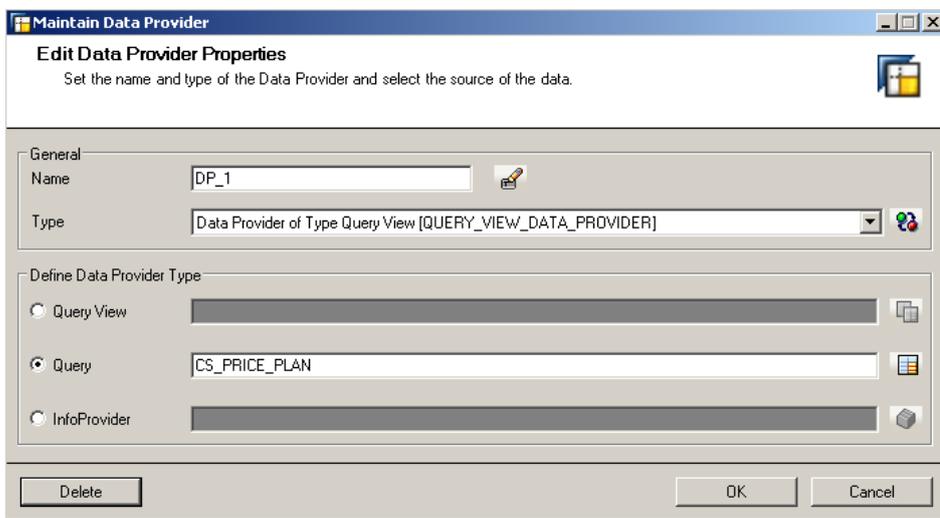
The OSS Note 1128508 describes the introduced concept as well.

## 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

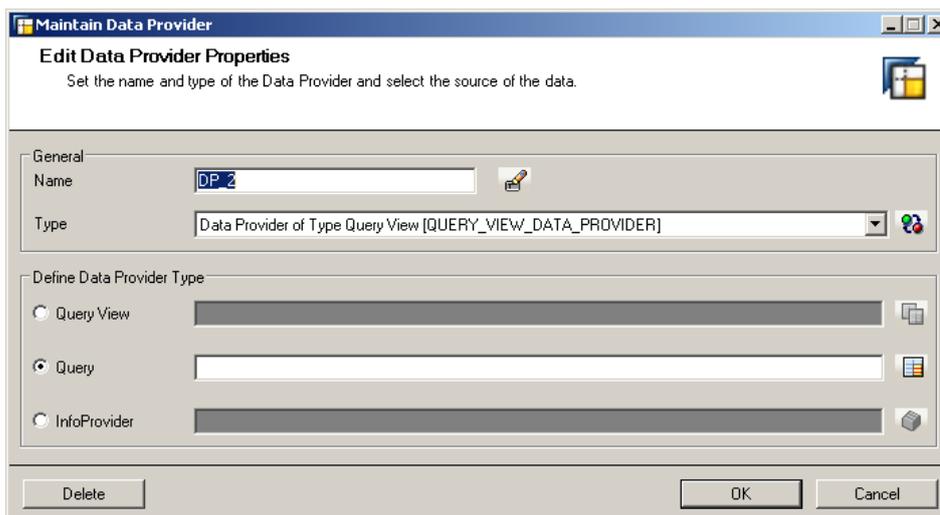


The screenshot shows the 'Maintain Data Provider' dialog box with the following configuration:

- General:**
  - Name: DP\_1
  - Type: Data Provider of Type Query View [QUERY\_VIEW\_DATA\_PROVIDER]
- Define Data Provider Type:**
  - Query View
  - Query: CS\_PRICE\_PLAN
  - InfoProvider

Buttons: Delete, OK, Cancel

4. Create an additional Data Provider without specifying a query



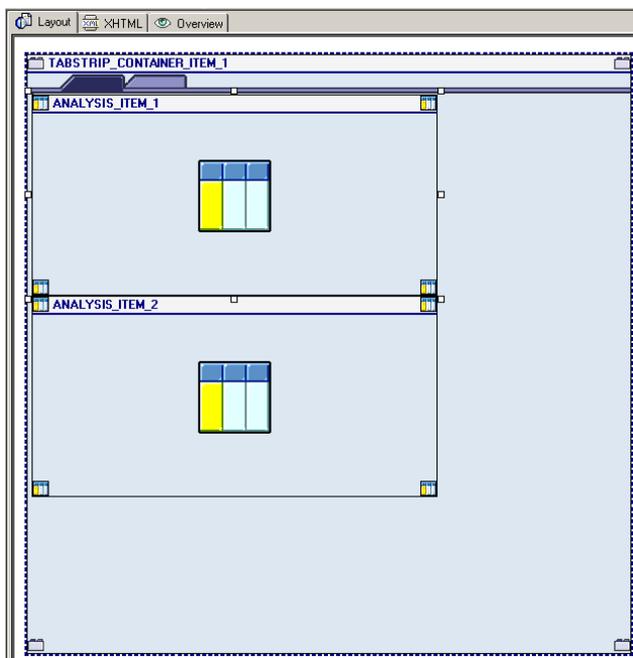
The screenshot shows the 'Maintain Data Provider' dialog box with the following configuration:

- General:**
  - Name: DP\_2
  - Type: Data Provider of Type Query View [QUERY\_VIEW\_DATA\_PROVIDER]
- Define Data Provider Type:**
  - Query View
  - Query: (empty field)
  - InfoProvider

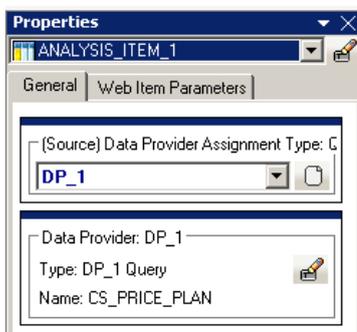
Buttons: Delete, OK, Cancel

## 4.1.2 Create a Tab Strip Container and the Analysis Items

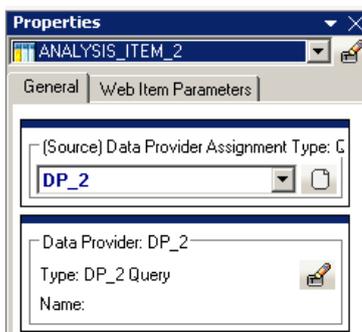
1. Add a new Tab Pages Item
2. Add two Analysis Items



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)

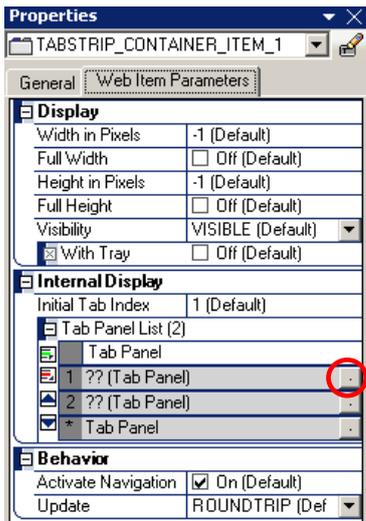


### 4.1.3 Set the Command on Activation

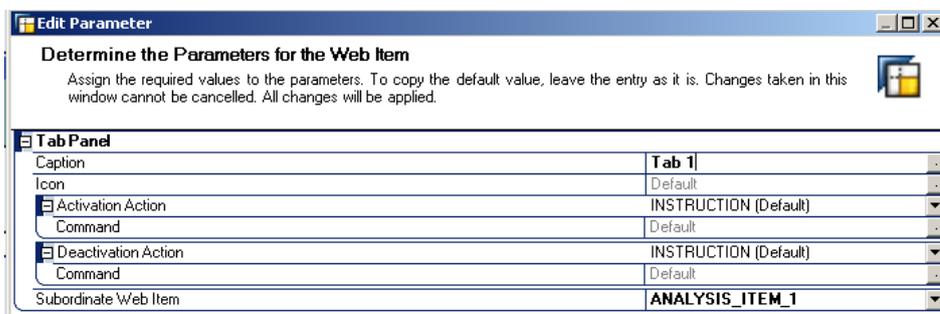
1. There is the possibility to add commands during the activation of a tab strip. This possibility is used to set the dynamically the Data Provider on the second tab strip.

Mark the TABSTRIP\_CONTAINER\_ITEM\_1 and go the Properties section and choose Web Item Parameters

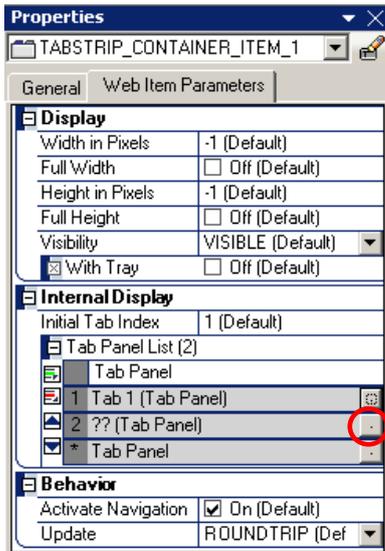
2. Edit the parameters of the first tab



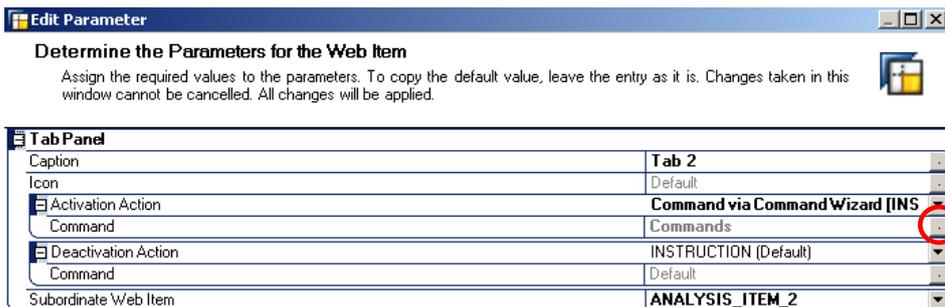
3. Add a caption in the appearing screen and the subordinate Web Item is the ANALYSIS\_ITEM\_1 and press OK.



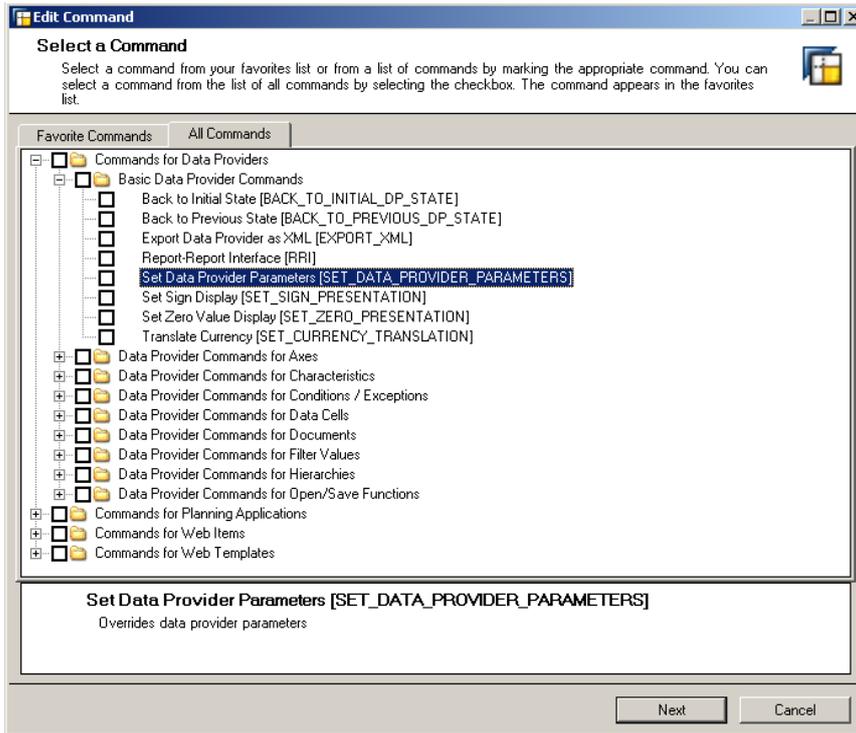
4. Edit the parameters of the second tab



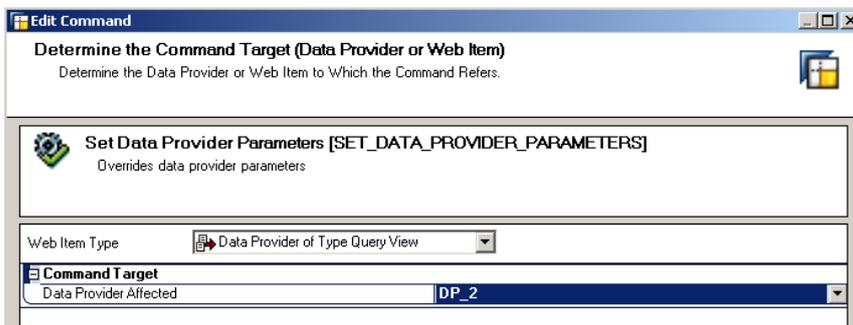
5. Add a caption in the appearing screen and assign the subordinate Web Item ANALYSIS\_ITEM\_2
6. Add a command for the Activation Action



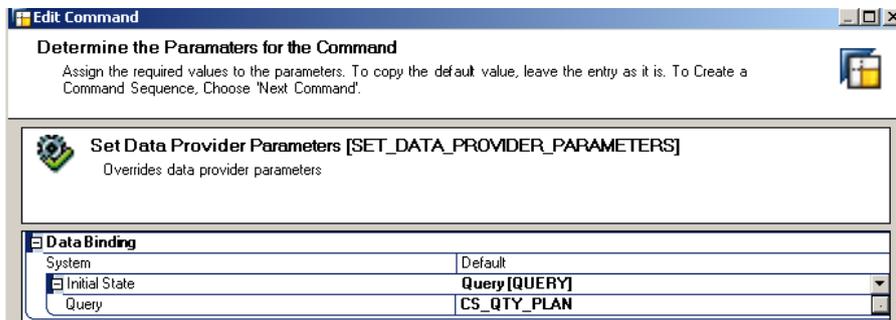
- Choose the command SET\_DATA\_PROVIDER\_PARAMETERS and press next



- Choose for the Data Provider Affected DP\_2 and press next



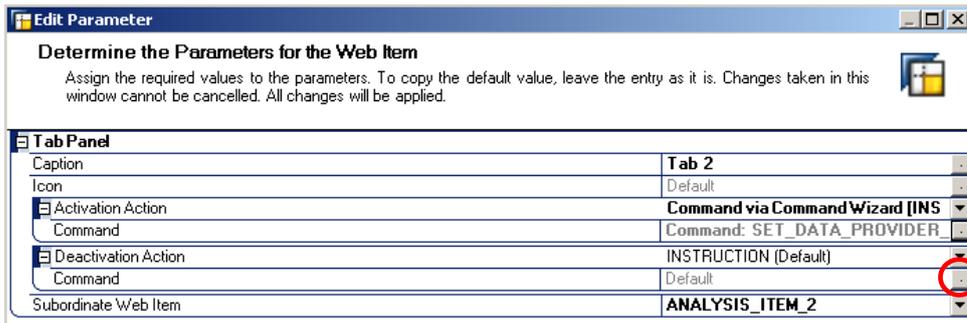
- Choose the data provider which should be display at the second tab and press OK



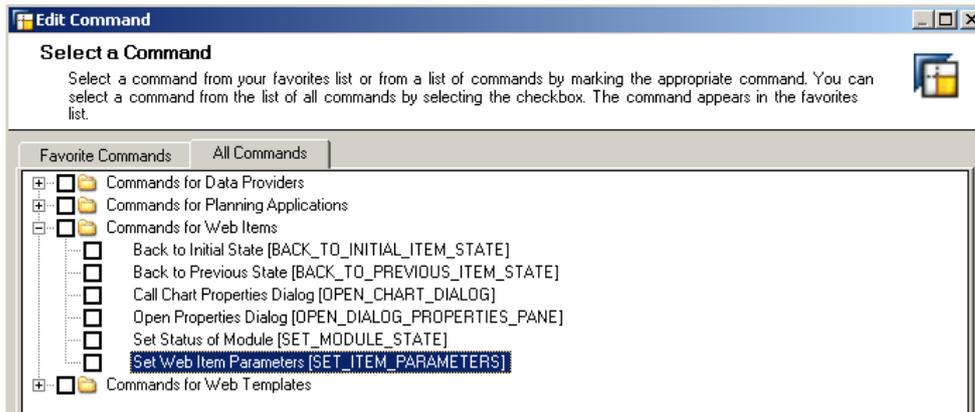
### 4.1.4 Set the Deactivation Action

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.

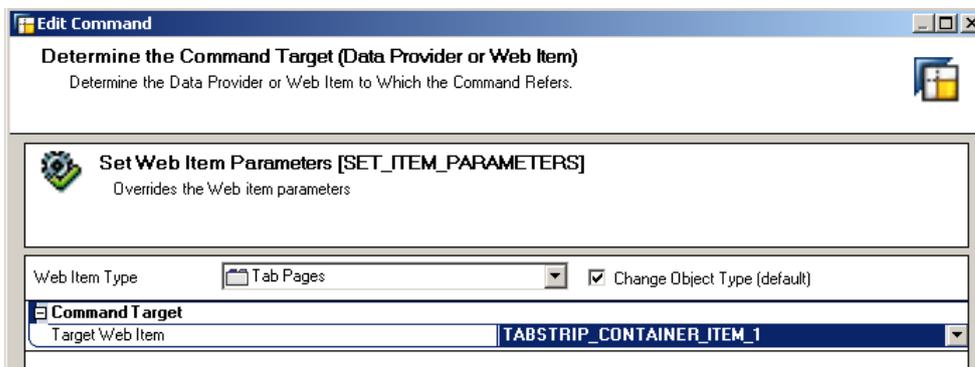
In Edit Parameter for the second tab strip choose Deactivation Action



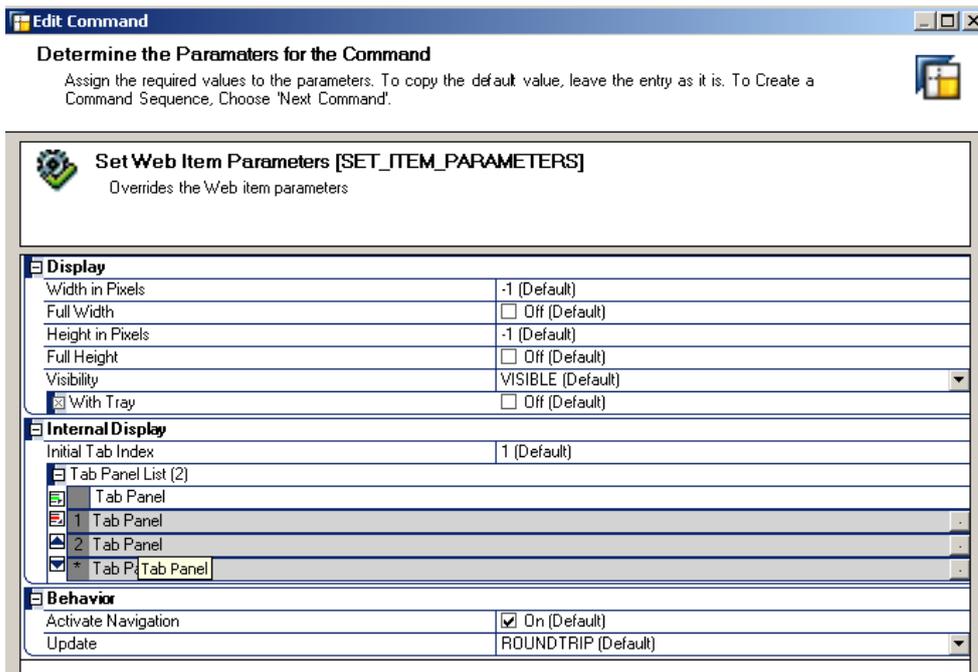
2. Choose the command SET\_ITEM\_PARAMETERS



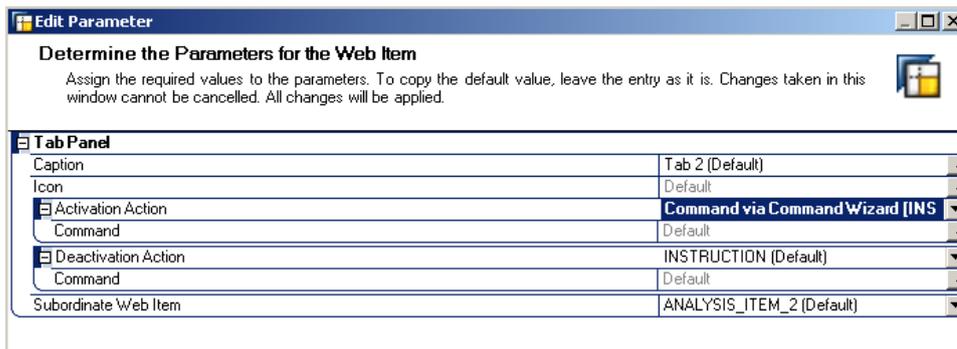
3. Web Item Type Tab Pages and Target Web Item TABSTRIP\_CONTAINER\_ITEM\_1



4. Choose the second Tab strip



5. Make no changes because the default parameters are sufficient



You can now save and test you planning application in the web

BEx Web - Test of Dynamic data Provider set - Windows Internet Explorer

http://pwwdf2836.wdf.sap.corp:50000/iri/servlet/prt/portal/prtroot/pcd13aportal\_content12fcom.sap.pct12platform\_add\_ons12fcom.sap.ip.bi12fiViews12fcom.sap.ip.bi.bex?TEMPLATE=TEST\_CS\_DP\_SET&DUN

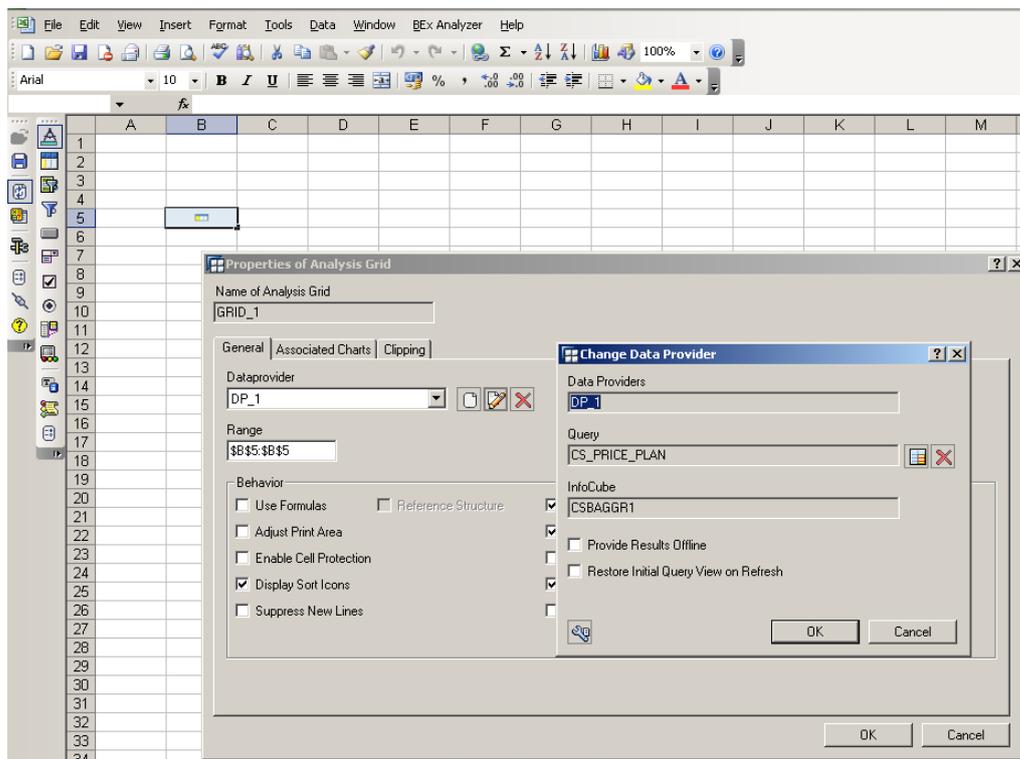
SAP Support Portal - SAP-Hi... BEx Web - Test of Dyna... x

Sales Price		Sales Quantity												Overall Result
Fiscal year/period	JAN 2007	FEB 2007	MAR 2007	APR 2007	MAY 2007	JUN 2007	JUL 2007	AUG 2007	SEP 2007	OCT 2007	NOV 2007	DEC 2007	Overall Result	
	Sales Price	Sales Price	Sales Price	Sales Price	Sales Price	Sales Price	Sales Price	Sales Price	Sales Price	Sales Price	Sales Price	Sales Price	Sales Price	
Material	EUR	EUR	EUR	EUR	EUR								EUR	
1 Material 1	0,00	0,00	0,00	0,00	0,00								0,00	
2 Material 2														
Overall Result	0,00	0,00	0,00	0,00	0,00								0,00	

## 4.2 Dynamic Data Providers in BEx Analyzer

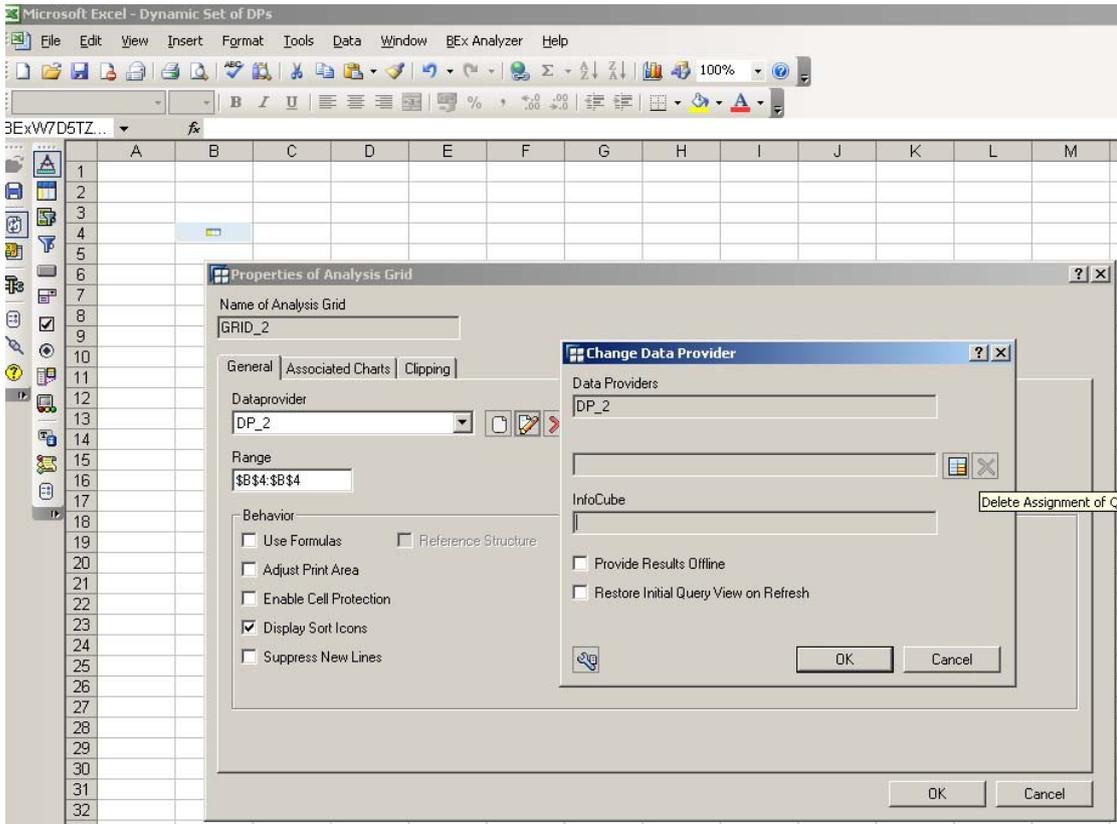
### 4.2.1 Create Worksheets and Add Queries

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

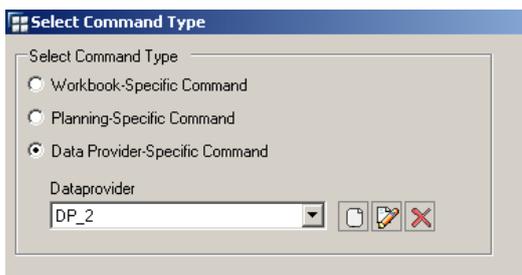


### 4.2.2 Create a Button Setting the Data Provider

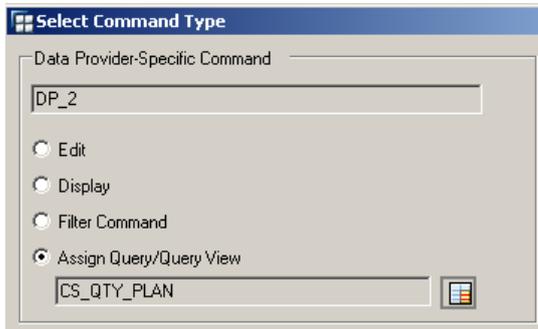
In the web application we have the possibility to add directly a command in the tab strip item itself. In the BEx Analyzer this features do not exists. We will have to combine Excel techniques with BEx Analyzer functionality: we will use a VBA macro that is executed on Worksheet change and then triggers 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.

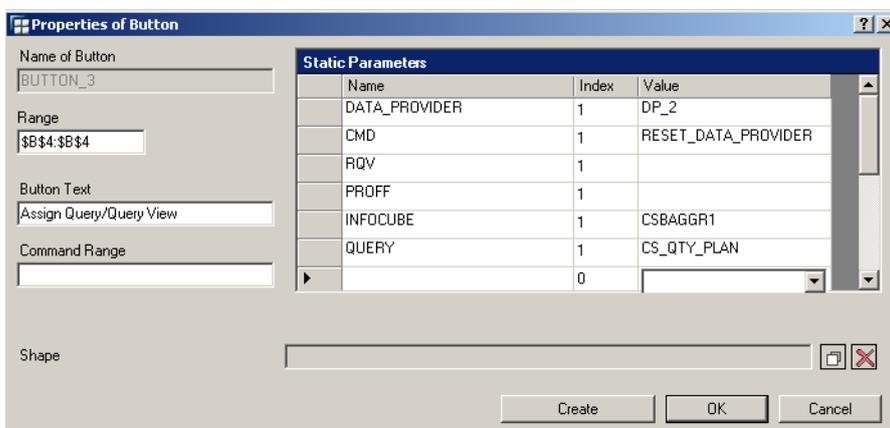
1. Add Button in third Worksheet right mouse click "Properties"
2. Select Command Type "Data Provider Specific Command" and choose Data Provider "DP\_2" and press "Next"



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)

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

4. Go to ThisWorkbook and add following code

```
Dim lDp2Set As String

Sub Workbook_SheetActivate(ByVal Sh As Object)
```

```
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](http://www.sdn.sap.com/irj/sdn/howtoguides)