How-To Write to Secured Dimension Members in Default Logic

Applicable Releases:

SAP BusinessObjects Planning and Consolidation 7.5, version for SAP NetWeaver

IT Practice / Topic Area:

Development

Version 1.0.0

July 2010
## Document History

<table>
<thead>
<tr>
<th>Document Version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0.0</td>
<td>Initial release</td>
</tr>
</tbody>
</table>
## Typographic Conventions

<table>
<thead>
<tr>
<th>Type Style</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example Text</strong></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><strong>Example text</strong></td>
<td>Emphasized words or phrases in body text, graphic titles, and table titles</td>
</tr>
<tr>
<td><strong>Example text</strong></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><strong>Example text</strong></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><strong>&lt;Example text&gt;</strong></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><strong>EXAMPLE TEXT</strong></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. Example Configuration ....................................................................................................... 2
   4.1 Create a copy of ApShell .............................................................................................. 2
   4.2 Create a User for Secured writeback Activities ............................................................ 3
   4.3 Create Member Access Profiles ................................................................................... 5
      4.3.1 All Access Member Access Profile .................................................................. 5
      4.3.2 Limited Access Member Access Profile ........................................................... 7
   4.4 Configure Users ............................................................................................................ 9
      4.4.1 Configuring the BPC Client User ..................................................................... 9
      4.4.2 Configuring the Impersonation Account ......................................................... 11
5. Development Tasks ........................................................................................................... 13
   5.1 Import the ZUJR_Impersonate Class ......................................................................... 13
      5.1.1 Copy R/K files to NetWeaver System ............................................................ 13
      5.1.2 Import the Transport of Copies ...................................................................... 13
   5.2 Create the Sample BADI Implementation .................................................................. 16
      5.2.1 Creating the BADI Implementation ................................................................. 16
      5.2.2 Set the Filter Value ........................................................................................ 19
      5.2.3 Implement IF_UJ_CUSTOM_LOGIC~EXECUTE ......................................... 21
      5.2.4 Update Default Logic to call the Sample BADI .............................................. 25
6. Example Execution ............................................................................................................ 26
   6.1 Use Case 1 ................................................................................................................. 26
   6.2 Use Case 2 .................................................................................................................. 31
1. Business Scenario
SAP BusinessObjects Planning and Consolidation, version for the Microsoft platform (hereafter referred to as BPC_MS) allows default logic to write data to secured members that the calling user does not have access to. This functionality could be used to ensure that data written to secured dimension members was the result of a calculation, and not manual data entry (i.e. Input Schedules). SAP BusinessObjects Planning and Consolidation, version for SAP NetWeaver (hereafter referred to as BPC_NW) performs the standard security checks with data written from default logic, thus this method no longer works by default.

⚠️ CAUTION
Using this functionality could potentially cause SOX compliance issues! Ensure you investigate the ramifications of bypassing standard BPC_NW security mechanisms with your project team before implementing.

2. Background Information
This How-To guide provides a BPC_NW based solution to this feature difference between BPC_MS and BPC_NW. It includes a class that can be called within BADI’s (or other custom code) that allows you to impersonate a user with the necessary privileges during writeback. It also includes details for an example BADI implementation based on the delivered Apshell application set.

3. Prerequisites
- SAP GUI / Developer Access to SAP NetWeaver Business Warehouse (SAP Netweaver BW)
- SAP BusinessObjects Planning and Consolidation, version for SAP NetWeaver
  - Version 7.5
4. Example Configuration
This section outlines the configuration of an example application set for use with the sample BADI.

4.1 Create a copy of ApShell
The sample BADI is based on the Planning application delivered in ApShell. The first step of configuring the example is to create a copy of ApShell. To do this:

1. Log into the BPC Administration Client

2. Select “Add New Application Set” from the Action Pane

   Manage Application Sets

   Application Set: Task
   - Add a new application set
   - Set application set status
   - Set template version
   - Refresh client-side dimension files
   - Delete an application set
   - User Activity

3. Enter the Application Set name, a description and select “APSHELL” as the application to duplicate. Select “Go to next step”.

   Manage Application Sets

Add a New Application Set - Step 1 of 2

What is the name of the new application set?
APSHELL_BPS

What is the description of the new application set?
Test application set

Which application set would you like to duplicate?
APSHELL

Go to Next Step 2 of 2
4. Leave all options selected. Click “Add new Application Set”.

4.2 Create a User for Secured writeback Activities

In this section we will create a user that will have access to write to all secured dimensions. This user will be used in the sample BADI for impersonation.

In addition, this user will only exist in SAP NETWEAVER BW. This limits the risk of someone connecting directly to BPC_NW with this user.

To create the user:

1. Log into SAPGUI

2. Go to t-code SE16.

3. Enter UJE_USER as the table name and click the Table Contents button (F7)
4. Enter the name of the application set you created in step 4.1 and click Execute (F8).

5. Select the “Table Entry -> Create” menu option (F5).

6. Enter the name of the application set created in step 4.1 as the APPSET ID, and a new user id for the USER ID. Click “Save”
Note
The user id does not need to exist in Active Directory in fact, it is better if it doesn’t. That way it cannot be used to log in via the BPC client(s) and can only be used programmatically.

4.3 Create Member Access Profiles

In this step we will create two member access profiles. The first one will have read and write access to all secured dimension members, while the second will be limited to the “ACTUAL” category in the Planning application.

4.3.1 All Access Member Access Profile

In this step, we will create a member access profile with full access (read and write) to the secured dimensions.

1. Log into the application set created in step 4.1 via BPC Administration.

2. Expand Security in the left hand pane, and then select Member Access Profiles.

4. Enter MAP_FULL as the Member Access Profile name and optionally a description. Click Next.

5. Enter the following for the Planning Application in the Access section:
   - Access: Read & Write / Dimension: Category / Member: [ALL]
   - Access: Read & Write / Dimension: P_CC / Member: [ALL]
6. Enter the following for the RATE Application in the Access Section:
   • Access: Read & Write / Dimension: Category / Member: [ALL]
   • Access: Read & Write / Dimension: R_ENTITY / Member: [ALL]

7. Click Next.

8. Do not assign to any users or teams at this point. Click Next.

9. Click Apply

4.3.2 Limited Access Member Access Profile

In this step, we will create a member access profile with limited access to the Category secured dimension in the Planning application.

1. Log into the application set created in step 4.1 via BPC Administration.

2. Expand Security in the left hand pane, and then select Member Access Profiles.


4. Manage Member Access Profiles

Member Access Profile Tasks
   Add new member access profile
5. Enter MAP_LIMITED as the Member Access Profile name and optionally a description. Click Next.

6. Enter the following for the Planning application in the Access section:

- Access: Read & Write / Dimension: Category / Member: ACTUAL
- Access: Read Only / Dimension: Category / Member: [ALL]
- Access: Read & Write / Dimension: P_CC / Member: [ALL]
7. Enter the following for the Rate application in the Access section:
   - Access: Read & Write / Dimension: Category / Member: [ALL]
   - Access: Read & Write / Dimension: R_ENTITY / Member: [ALL]

<table>
<thead>
<tr>
<th>PLANNING Application</th>
<th>RATE Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define access rights for the secured dimensions in the following application: RATE</td>
<td></td>
</tr>
<tr>
<td>Access</td>
<td>Dimension</td>
</tr>
<tr>
<td>Read &amp; Write</td>
<td>CATEGORY</td>
</tr>
<tr>
<td>Read &amp; Write</td>
<td>R_ENTITY</td>
</tr>
</tbody>
</table>

8. Click Next.

9. Do not assign to any users or teams at this point. Click Next.

10. Click Apply

4.4 Configure Users

In this section we will configure two user accounts. The first can be any user account that exists in Active Directory. The second is the user account that we created in section 4.2.

4.4.1 Configuring the BPC Client User

The username used in this step will vary for each installation. Any valid Active Directory account can be used. In this example, the user id is “SAP_ALL\1814959”.

1. Log into the application set created in step 4.1 via BPC Administration.

2. Expand Security -> Users -> Domain (where domain is your domain)
3. Select the user account you will use with this step and select “Modify User” in the Action Pane

4. Click “Next” in the User Setup section.

5. Click “Next” in the User Detail section.

6. Do not assign the user to a team.

7. Select the “Assign to Task Profiles” tab. Assign the user to the FULL_TSK task profile.

8. Select the “Assign to Member Access profiles” tab and assign the user to MAP_LIMITED.
9. Click Next

<table>
<thead>
<tr>
<th>Summary</th>
<th>Review the summary, then click Apply to save your settings and process security</th>
</tr>
</thead>
</table>
Here is a summary of your new user selections:

<table>
<thead>
<tr>
<th>User ID</th>
<th>Teams</th>
<th>Task Profiles</th>
<th>Member Access Profiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP_ALLV014959</td>
<td>&lt;unassigned&gt;</td>
<td>FULL_T3K</td>
<td>MAP_LIMITED</td>
</tr>
</tbody>
</table>

10. Click Apply

### 4.4.2 Configuring the Impersonation Account

The username used in this step is the same user id created in step 4.2.

1. Log into the application set created in step 4.1 via BPC Administration.

2. Expand Security -> Users -> Domain (where domain is your domain)

   - [AP SHELL_DPS]
   - Dimension Library
   - Application
   - Work Status
   - Security
   - [Users]
     - [SAP_ALL]
     - [I014959]
     - [I014959_5]

3. Select the user account you will use in this step and select “Modify User” in the Action Pane.

   **Manage Users**

   **User Tasks**
   - Add new user

   **I014959 Tasks**
   - Modify user
   - Delete user

4. Click “Next” in the User Setup section.

5. Click “Next” in the User Detail section.
6. Do not assign the user to a team.

7. Select the “Assign to Task Profiles” tab. Assign the user to the FULL_TSK task profile.

   Task profile:
   FULL_TSK

   Selected users:
   SAP_ALLUV14959
   SAP_ALLUV14959_S

8. Select the “Assign to Member Access profiles” tab and assign the user to MAP_LIMITED.

   Member access profile:
   MAP_FULL

   Selected users:
   SAP_ALLUV14959_S

9. Click Next

   **Summary**

   Review the summary, then click **Apply** to save your settings and process security

   Here is a summary of your new user selections:

<table>
<thead>
<tr>
<th>User ID</th>
<th>Teams</th>
<th>Task Profiles</th>
<th>Member Access Profiles</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP_ALLUV14959_S</td>
<td>&lt;unassigned&gt;</td>
<td>FULL_TSK</td>
<td>MAP_FULL</td>
</tr>
</tbody>
</table>

10. Click Apply
5. Development Tasks

This section outlines the development tasks involved in importing the ZUJR_IMPERSONATE class, as well as setting up and configuring the sample BADI for use case testing in section 6.

5.1 Import the ZUJR_Impersonate Class

This section describes the process of importing the ZUJR_IMPERSONATE class from a “transport of copies” into your NetWeaver system.

The ZUJR_IMPERSONATE class exposes one public method, write_data_as_user. This method takes the following import parameters:

- P_USER_ID – the user id to impersonate during writeback
- P_APPSET_ID – the application set id where the writeback operation will occur.
- P_APP_ID – the application id where the writeback operation will occur.
- PT_DATA – the data to writeback
- P_REVERSE_SIGN – a true/false value that determines whether signs are reversed for INC/LEQ accounts during writeback.

It exports the following parameters:

- PT_MESSAGE – a table of messages returned by the writeback engine
- PT_STATUS – a structure detailing the number of submitted/accepted and rejected records as returned by the writeback engine.
- PT_ERROR_RECORDS – a table including the details for any rejected records that the writeback engine encountered.

This method provides extensive logging, which can be found in t-code UJFS under the following path:

root/webfolders/$application_set/$application/privatepublications/$domain/$user/$datetime/default.log

where:

- $application_set = your application set
- $application = your application (PLANNING in this example)
- $domain = Active Directory domain of submitting user
- $user = Active Directory user id of submitting user
- $datetime = Date / Timestamp of default logic execution

5.1.1 Copy R/K files to NetWeaver System

1. Copy K900318.EPM to the usr\sap\trans\cofiles directory.
2. Copy R900318.EPM to the usr\sap\trans\data.

5.1.2 Import the Transport of Copies

1. Log into SAPGUI for the SAP Netweaver BW system.
2. Go to the transport management system (t-code STMS)
3. Select the import overview (F5)

4. Double click the appropriate transport queue.

5. Select Extras -> Other Requests -> Add

6. Enter EPMK900318 as the Transport Request and click continue.

7. Select “Yes” in the Add Transport Request dialog.

8. Select EPMK900318 in the Request list and select Import Request (F11 – small truck)
9. Enter the appropriate “Target Client” and select continue.

10. Select “Yes” to start the import.
11. You should see a line similar to the following in the request list once the import has completed successfully.

Tip
You can take a look at the code for class ZUJR_IMPERSONATE in t-code SE80 once the import is complete.

### 5.2 Create the Sample BADI Implementation

This section describes the process of creating a sample BADI that utilizes the ZUJR_IMPERSONATE class that was imported in section 5.1.

#### 5.2.1 Creating the BADI Implementation

To create the BADI Implementation:

1. Log into SAPGUI for SAP Netweaver BW
2. Go to the BADI Builder (t-code SE18)
3. Enter UJ_CUSTOM_LOGIC as the Enhancement Spot and click the Display button.

4. Click the Create BADI Implementation Button

5. Click the “Create Enhancement Implementation” button (F8).

6. Enter “Z_EI_IMPERSONATE” as the Enhancement Implementation as well as a descriptive short text, click continue.
7. Select “Local Object” in the “Create Object Directory Entry” dialog.

8. Select the Z_IE_IMPERSONATE Enhancement Implementation and click continue.

9. Enter the following in the “Create BADI Implementation” dialog, then click continue:
   - BADI Implementation: Z_BADI_IMPERSONATE
   - Description: Impersonation BADI Sample
   - Implementing Class: Z_CL_IMPERSONATE

5.2.2 Set the Filter Value
This section continues from where section 5.2.1 leaves off.

1. Expand the BADI implementation Z_BADI_IMPERSONATE

2. Select Filter Val.

3. Click the Display/Change button.
4. Click the “Create Filter Combination” button.

5. Double click on the question marks “???” in the Value1 column.

6. Enter “IMPERSONATE” in the Value 1 textbox and click continue

7. Click the Activate button ( ).
8. Activate all objects related to the impersonation BADI.

5.2.3 Implement IF_UJ_CUSTOM_LOGIC~EXECUTE

In this section we will implement the Execute method for our sample BADI. This sample BADI changes the category for records being submitted to FORECAST, then calls the zujr_impersonate-
>write_data_as_user method to write the data back as our impersonated user. This should allow users without access to FORECAST to be able to update records in the FORECAST category through this BADI, but not from an input schedule.

The instructions continue from where section 5.2.2 left off.

1. Expand the BADI implementation Z_BADI_IMPERSONATE

2. Select Implementing Class.

3. Double click on method IF_UJ_CUSTOM_LOGIC~EXECUTE
4. Select “Yes” in the Create Method Implementation dialog box.

![Create Method Implementation dialog box]

5. Click the Display ↔ Change button.

6. Copy and paste the following code into the IF_UJ_CUSTOM_LOGIC~EXECUTE method body:
method IF_UJ_CUSTOM_LOGIC-EXECUTE.

constants:

  c_category           type string value 'CATEGORY'. " used to assign the SIGNEDDATA component

  " used to assign the SIGNEDDATA component

data:

  l_ref                type ref to data, " used to properly assign the attributes to field more complex field symbols

  lr_error             type ref to data,

  lo_impersonate       type ref to ZUJR_IMPERSONATE,

  lo_exception         type ref to cx_static_check,

  lt_message           type uj0_t_message,

  l_status             type ujr_s_status_records, " used in writ

  eback call

  l_log                type string,

  l_measure            type uj_dim_member.

FIELD-SYMBOLS:

  <l_category>           type any, " used to hold the category in our example below

  <ls_line>              type any, " represents a line of e_ct_data

  a

  <ls_new_line>          type any, " represents a line of e_ct_data

  a

  <lt_new_data>          type standard table, " Holds the new records

  a

  <lt_error_records>     type standard table.

create data l_ref like line of ct_data.
assign l_ref->* to <ls_line>.
assign l_ref->* to <ls_new_line>.

create data l_ref like ct_data.
assign l_ref->* to <lt_new_data>.

create data l_ref like ct_data.
assign l_ref->* to <lt_error_records>.

* change CATEGORY to BUDGET (secured) at runtime

  loop at ct_data ASSIGNING <ls_line>.
  move-
  corresponding <ls_line> to <ls_new_line>.
  " create a new line based on the last line
  assign component c_category of STRUCTURE <ls_new_line> to <l_categ
  ory>.
  " get category
  <l_category> = 'FORECAST'.
  collect <ls_new_line> into <lt_new_data>.
endloop.

create object lo_impersonate.

try.
  lo_impersonate->write_data_as_user(
    exporting
      p_user_id        = 'Domain\User'
      p_appset_id      = i_appset_id
      p_app_id         = i_appl_id
      pt_data          = lt_new_data
      p_reverse_sign   = abap_true
    importing
      pt_message       = lt_message
      pt_status        = l_status
      pt_error_records = lt_error_records
  )
  catch cx_static_check into lo_exception.
    l_log = 'Exception encountered calling lo_impersonate->write_data_as_user'.
    cl_ujk_logger=>log( i_object = l_log ).
  endtry.

  refresh ct_data.

endmethod.
7. Change the text 'Domain\User' on line 47 to the username you created in section 4.2 and configured in section 4.4.2.

```java
try {
    lo_impersonate->write_data_as_user(
        exporting
        p_user_id = 'Domain\User',
        p_appset_id = i_appset_id
    )
}
```

**Example**

In my case I changed the p_user_id to ‘SAP_ALL\814959_S’

8. Click the Save button ( )

9. Click the Activate button ( ) and activate the updated method.

```
<table>
<thead>
<tr>
<th>D</th>
<th>Object</th>
<th>Obj. name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>METH Z_CL_IMPERSONATE</td>
<td>IF_UJ_CUSTOM_LOGIC~EXECUTE</td>
</tr>
</tbody>
</table>
```

### 5.2.4 Update Default Logic to call the Sample BADI

In this section, we will update default logic to call the sample BADI.

1. Log into the application set created in section 4.1 via the BPC_NW Administration client.

2. Expand Application -> Planning -> Script Logic and select DEFAULT.LGF.

```
- APSHELL_DPS
- Dimension Library
- Application
  - PLANNING
    - Work Status Settings
    - Dimensions
    - Drill Throughs
    - Business Rules
    - Journals
- Script Logic
  - ALLOCATION.LGF
  - DEFAULT.LGF
  - FKTRANS.LGF
```
3. Copy and paste the following script logic into DEFAULT.LGF.
   // Call sample impersonation BADI
   *START_BADI IMPERSONATE
   QUERY = ON
   WRITE = ON
   *END_BADI

4. Click Validate and Save in the Action Pane.

   **Applications**

   **Manage Script Logic**

   **Script Logic Task**
   Create New Logic

   **DEFAULT.LGF Task**
   Validate Only
   Validate and Save
   Delete Logic
   Save Logic

---

6. **Example Execution**

   In this section, we will review two use cases that demonstrate the behavior of the ZUJR_IMPERSONATE class, as well as our sample BADI.

   **6.1 Use Case 1**

   This use-case demonstrates the intended use of this solution by writing data to the ACTUAL category (which the user has access to) via an input schedule, and writing the same records to the FORECAST category (which the user does not have access to).

   To execute this use case:

   1. Log into BPC for Excel as the user configured in section 4.4.1.
2. Set your current view to the following:

![Current View](image)

3. Click “Data Input” in the Action Pane.

**Getting Started**

**Available Task Categories**

- Reporting & Analysis
- Data Input
- Journals
- Manage Data
- Open System Reports

4. Click “Open a Blank Workbook” in the Action Pane.

**Data Input Options**

- **Build New**
  - Open a blank workbook
- **Build a schedule using drag & drop**
- **Build a schedule using a dynamic template**
- **Open Existing**
  - Open an existing schedule
  - Open an existing schedule from My Schedules folder
5. Type "=evdre()" in cell A1 and click the Refresh button ( ).

6. Click "Yes" when asked if you want to clear existing input data.

7. Add TIME to columns and P_ACCT to rows. Set member selection to “BAS” and click OK.

8. Enter a value in each Account/Time Period combination.
9. Click “Send Data” in the Action Pane.

   **Schedule Tasks**
   - Send data
   - Retrieve data
   - Edit schedule using drag & drop
   - Set workbook options
   - Worksheet dimension lock options
   - Park N Go
   - Save to My Schedules folder

10. Click “Send Active Worksheet” in the Action Pane.

   **Send Data Options**
   - Send active worksheet
   - Send active workbook
   - Submit supporting document
   - Set workflow (approval status)

11. Click “Yes” when asked if you want to send the data records.

12. You should receive a success message indicating the records were submitted successfully.

   ![Refresh schedule results](image)

   **Note**
   The results will not include details on the success or failure of default logic. You can consult the logs for more detailed information (see section 5.1 for more information).
13. Change the current view Category from ACTUAL to FORECAST.

14. Click the Expand All ( ) button.

15. You can verify that all records written to the ACTUAL category were also written to FORECAST.
6.2 Use Case 2

This use-case demonstrates what happens if a user tries to enter data directly to the FORECAST category.

Note

This use case continues where use case 1 left off.

1. Ensure you are in the same input schedule created in use case 1.

2. Ensure that Category is set to FORECAST in the current view.

3. Change the data in the input schedule for all account/time combinations.
4. Click “Send Data” in the Action Pane.

5. Click “Send Active Worksheet” in the Action Pane.

6. Click “Yes” when asked if you want to send the data records.

7. You will receive error messages stating that you cannot write records to the FORECAST member in the Category dimension. All records will be rejected.