Step by Step Guide to Enhance a Data Source

Applies to:
SAP BI 7.0. For more information, visit the Business Intelligence homepage

Summary
This article provides a step by step guide to enhance a Standard Datasource as well as an LO Extractor Datasource with code.

Author:     Arminder Singh
Company:    Accenture
Created on: 26 November 2009

Author Bio
Arminder Singh is working as SAP BI Consultant in Accenture Services Private Ltd having extensive experience in implementation of SAP BI projects. His expertise includes SAP BI, ABAP and Business Objects.
# Table of Contents

Overview.........................................................................................................................3  
For LO Extractor Datasource: ..........................................................................................3  
For Standard Data Source:.............................................................................................8  
Related Content .............................................................................................................15  
Disclaimer and Liability Notice ......................................................................................16
Overview

In almost every project we get into a situation in which some extra fields are required in addition to what's already present in a standard data-source. In case of an LO extractor datasource we have a pool of reserved fields which can be added to the extractor at any time but for other datasources we have to append the structure and then write an enhancement code for populating the appended fields. We will explain each process with screen-shots for a better understanding.

For LO Extractor Datasource:

In the present example we are enhancing the data-source 2LIS_13_VDITM with an additional field RFBSK (Posting Status) from the communication structure MVCBRK which is present as a reserve field for this datasource.

One important thing to note here is that the enhancement and the load of setup tables should be done at a posting free time i.e. no new documents are created or processed at that time so the best way is to lock out the users while filling up the setup tables so that no loss of data takes place. Also batch job schedule need to be put on hold.

1. The first step here is to log-on to the source system and goto transaction LBWG to delete the setup tables for the application specific to this datasource and in the present case being 13 – SD Billing BW.

   ![Deletion of Setup Data](image)

   Execute the transaction by selecting application 13 and you will get a prompt for your choice, click on yes.

2. Then we check for any Outbound Queue for the specific application component in transaction LBWQ and if some entries are present for that then we have to delete them.
Select the queue and click on delete.

3. We should also check in transaction RSA7 for our particular data-source and if some records have been posted then we should also delete them.

4. After completing these steps goto transaction LBWE which is for customizing the data-sources like enhancements or choosing a different Update Mode depending on the requirement.
Click on the specific application component, and click on maintenance for our data-source.

The next screen shows all the reserved fields available for this data-source and we can just choose our field from the right side and move it to the left.
After this step our datasource will become inactive and we will see a red light indicator, so just click on the icon for datasource generation and unhide the newly enhanced field and Generate the datasource.
After this step the indicator for datasource will turn yellow and now you have to just click on the Inactive icon to activate this datasource.

5. We are done with the enhancement of the datasource and now we have to reload the setup tables using transaction SBIW or we can directly use the transaction OLI9BW for our application component **13 – SD Billing BW**.

<table>
<thead>
<tr>
<th>Statistical Setup from Old Documents: Billing Docs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document data restriction</td>
</tr>
<tr>
<td>Archiving session</td>
</tr>
<tr>
<td>Sales Organization</td>
</tr>
<tr>
<td>Company code</td>
</tr>
<tr>
<td>SD document</td>
</tr>
<tr>
<td>Time data restriction</td>
</tr>
<tr>
<td>Archiving time</td>
</tr>
<tr>
<td>Sales Org. time</td>
</tr>
<tr>
<td>Company code time</td>
</tr>
<tr>
<td>SD doc. time</td>
</tr>
</tbody>
</table>

Just enter the Name of run and enter the termination date & time as per your needs so that all the documents are processed. Click on execute to start the document processing.

6. After this step we are done with the enhancement on the Source System side and can simply check in RSA3 whether we are getting the proper data, now just logon to the BI system, replicate the data-source and activate it for the changes to appear in BI. And thereafter we can do the necessary mappings and modeling.

If the fields that need to be enhanced are not available in the communication structures of the datasource in LBWE then we need to follow the below mentioned approach for them as well.
For Standard Data Source:

In the present example we will enhance a master data standard datasource 0VEN_COMPC_ATTR with three new fields WITHT, WT_WITHCD and WT_SUBJCT all coming from the same table LFBW.

1. Logon to the Source System and goto transaction RSA6 which is used for post-processing datasources after they have been activated from the Business Content using transaction RSA5. Choose the datasource 0VEN_COMPC_ATTR (in the present case) and double click on it.

   The following screen with all the present fields will appear.

   **DataSource: Customer version Display**

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Shorttext</th>
<th>Selection</th>
<th>Hide field</th>
<th>Inversion</th>
<th>Field only</th>
</tr>
</thead>
<tbody>
<tr>
<td>MONT</td>
<td>Reconciliation Account In General Ledger</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALTKN</td>
<td>Previous Master Record Number</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BE6RU</td>
<td>Authorization Group</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLNKZ</td>
<td>Subsidy Indicator for Determining the Risk</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. Open the extract structure and then click on append structure.
Step by Step Guide to Enhance a Data Source

If the enhancement is being done for the first time then you'll have to create a new append structure by specifying a technical name.

Then you add the new fields with a prefix of ZZ and specify the component type. Afterwards check, save and activate the append structure and this will appear in the extract structure of the datasource.

One important point to remember is that if you are enhancing with a key-figure then do not forget to add its respective unit/currency field in the append structure.

Now refresh the system and again goto transaction RSA6 and edit the datasource 0VEN_COMPC_ATTR. Unhide the newly enhanced fields which will be present at the end.
5. After completing these steps we need to write an enhancement code for populating these new fields from their respective table. For this, go to transaction CMOD and choose the project for Enhancements ZBW00001 (in present case), if it’s not present then you have to create a project using the enhancement RSAP0001. SAP provides enhancement RSAP0001 that can be used to populate the extract structure.

Check the radio button for Components and click on change.
It has four components that are specific to each of the four types of R/3 DataSources:

- **Transaction data** - `EXIT_SAPLRSAP_001`
- **Master data attributes** - `EXIT_SAPLRSAP_002`
- **Master data texts** - `EXIT_SAPLRSAP_003`
- **Master data hierarchies** - `EXIT_SAPLRSAP_004`

6. Go to enhancement of Master data attributes datasources and click on include ZXRSAU02. All the codes will be written inside this include.

Here we are not using the old technique of writing a Case statement and then a When condition for each datasource as it makes the code quite messy and if several developers/teams are modifying the code then creates challenges for transport management.

So after using the below code in the Include we can simply create a new program starting with ZBWIN_XXXX for each data-source which will be independent from any other enhancement so won’t create any problems.
Step by Step Guide to Enhance a Data Source

ABAP Editor: Display Include ZXRSAU02

7. Now we'll create a new executable program in transaction SE38 for our datasource.

And then activate the include program.
And write the following code:

REPORT ZBWIN_0VEN_COMPC_ATTR.

TYPE-pools: sbiwa.

CONSTANTS : gc_vbeln TYPE vbeln VALUE '2'.

*---------------------------------------------------------------------
* Form EXECUTE_USER_EXIT
*---------------------------------------------------------------------
*   text
*---------------------------------------------------------------------
*  -->P_I_T_SELECT   text
*  -->P_I_T_FIELDS   text
*  -->P_C_T_DATA    text
*  -->P_C_T_MESSAGES text
*---------------------------------------------------------------------
FORM execute_user_exit
  TABLES i_t_select TYPE sbiwa_t_select
        i_t_fields TYPE sbiwa_t_fields
        i_t_data STRUCTURE BIW_LFB1_S   "Datasource structure"
        c_t_messages STRUCTURE balmi.

  TYPES: BEGIN OF T_LFBW,
    LIFNR TYPE LFBW-LIFNR,
    BUKRS TYPE LFBW-BUKRS,
    WITHT TYPE LFBW-WITHT,
    WT_SUBJCT TYPE LFBW-WT_SUBJCT,
    WT_WITHCD TYPE LFBW-WT_WITHCD,
  END OF T_LFBW.

  DATA: i_LFBW TYPE STANDARD TABLE OF T_LFBW,
       I_VEN_COMPC_ATTR TYPE STANDARD TABLE OF BIW_LFB1_S.
  DATA: l_tabix TYPE SY-TABIX.

  REFRESH: I_LFBW,
           I_VEN_COMPC_ATTR.
  CLEAR:  WA_LFBW,
          WA_VEN_COMPC_ATTR.

  I_VEN_COMPC_ATTR = i_t_data[].

  SELECT LIFNR
          BUKRS
          WITHT
          WT_SUBJCT
          WT_WITHCD
  FROM   LFBW
  INTO TABLE I_LFBW
  FOR ALL ENTRIES IN I_VEN_COMPC_ATTR
  WHERE  LIFNR = I_VEN_COMPC_ATTR-LIFNR AND
BUKRS = I_VEN_COMPC_ATTR-BUKRS.

IF SY-SUBRC = 0.
  SORT I_LFBW BY LIFNR.
ENDIF.

LOOP AT I_T_DATA INTO WA_VEN_COMPC_ATTR.
  l_tabix = sy-tabix.
  
  READ TABLE I_LFBW INTO WA_LFBW
  WITH KEY LIFNR = WA_VEN_COMPC_ATTR-LIFNR
  BUKRS = WA_VEN_COMPC_ATTR-BUKRS BINARY SEARCH.
  
  IF SY-SUBRC = 0.
    WA_VEN_COMPC_ATTR-ZZWITHT = WA_LFBW-WITHT.
    WA_VEN_COMPC_ATTR-ZZWIT_Subject = WA_LFBW-WT_SUBJECT.
    WA_VEN_COMPC_ATTR-ZZWIT_WITHCD = WA_LFBW-WT_WITHCD.
  ENDIF.
  MODIFY i_t_data FROM WA_VEN_COMPC_ATTR INDEX l_tabix.
ENDLOOP.

ENDFORM. " EXECUTE_USER_EXIT

Check and activate the program.

8. Finally check the data using transaction RSA3. And with this we finish with the enhancement of the data-source in the source system. Now just logon to the BI system, replicate the data-source and activate it for the changes to appear in BI. And thereafter we can do the necessary mappings and modeling.

All above changes will ask for a transport request. Assign appropriate development class/Package and assign all these objects into a transport request.
Related Content
For more information, visit the Business Intelligence homepage.
Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.