Extraction of CRM DataSource: using BADI – Step by Step Implementation

Applies to:
SAP Net Weaver BW. For more information, visit the EDW homepage

Summary
Detailed description on how to extract a CRM Data source - using BADI

Author: Adlin Sundararaj
Company: Accenture Services Pvt. Ltd.
Created on: 20 June, 2011

Author Bio
Adin S is working as SAP-BI Consultant in Accenture Services Private Ltd and has experience of working on implementation and maintenance projects.
Table of Contents

Introduction .................................................................................................................................................. 3

Business Scenario ........................................................................................................................................ 3

Step By Step Implementation ..................................................................................................................... 3

Step 1: Checking the Source System (CRM in our case) .................................................................................. 3

Step 2: Add the SDM field in the Source System .......................................................................................... 4

Step 2.1: Create an append structure for the 0CRM_SALES_ORDER_I ......................................................... 4

Step 2.2: Check if the business content is active for the SAP CRM sales order DataSource ......................... 5

Step 2.3: Check the extraction structure and the BDoc mapping .................................................................. 5

Step 2.4: Create a new BAdI implementation to fill the enhanced field in the append structure .................. 6

Step 2.5: Activate the new BAdI implementation ......................................................................................... 7

Step 2.6: Test the new BAdI implementation in the extractor checker ....................................................... 7

Conclusion ..................................................................................................................................................... 8

Related Content .......................................................................................................................................... 9

Disclaimer and Liability Notice .................................................................................................................. 10
**Introduction**

SAP CRM uses BW Adapter to extract data from SAP CRM and send it to SAP Business Information Warehouse and SAP NetWeaver Business Intelligence.

We can use Adapter when we have

- Synchronization Business Documents (sBDocs) &
- Messaging Business Documents (mBDocs).

For sBDocs, BW Adapter extracts objects relevant to mobile clients.

For mBDocs, BW Adapter extracts objects for CRM business transactions and CRM billing.

To populate the enhanced fields in real time, which means that the enhanced fields are written to the delta at the time we save the transaction.

In this case, we can use a Business Document (BDoc) Business Add-In (BAdI) to enhance the CRM DataSource. This is a new process — many CRM-BW implementations still use the traditional BAdI or user exit processes.

**Business Scenario**

In our example, we are going to extract a custom field from CRM system to BW using BADI.

Our CRM DataSource → 0CRM_SALES_CONTR_I

Custom field → Service Delivery Manager (SDM)

Our standard DS does not have all the fields which are required by customer, so we need to manually add the required fields and do the extraction.

**Step By Step Implementation**

**Step 1: Checking the Source System (CRM in our case)**

Go to the source system and check the Sales order data.

As we can see below, the Service Delivery Manager (SDM) field is present in CRM system

Service Delivery Manager (SDM) – is a partner function which is maintained in CRM system.

If we check RSA6 in Source System (CRM), in this screen you can see the list of all of the activated Data Sources in the source system.
Double Click on the DataSource 0CRM_SALES_ORDER_I this field is not available in the Extractor.

We need to extract this field Service Delivery Manager (SDM) to BW.

**Step 2: Add the SDM field in the Source System**

Follow the below six steps to add the new Custom field (SDM) to the SAP CRM sales order DataSource

**Step 2.1: Create an append structure for the 0CRM_SALES_ORDER_I**

For this Goto RSA6 → Locate the Datasource 0CRM_SALES_ORDER_I and click on **Enhance Extraction Structure** which is present on the top of the screen.

New window will open for append structure name

![Create Append Structure for](image)

**Append Name**: ZAPPEND_SALES_ORDER

Give any technical name and press **✓** to continue.

Now add the field Service Delivery Manager (SDM) in the append structure.

Create a field SDM in BU_PARTNER domain since it is a business partner function.

**Dictionary: Maintain Data Element**

<table>
<thead>
<tr>
<th>Data element</th>
<th>ZZSDM</th>
<th>New(Revised)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short Description</td>
<td>SDM</td>
<td></td>
</tr>
</tbody>
</table>

And maintain the field label as well.

Now save and activate.

Now if you can see, new field will be created in the extraction structure of 0CRM_SALES_ORDER_I Datasource.

Save and activate the Append structure & extraction Structure.
Step 2.2: Check if the business content is active for the SAP CRM sales order DataSource

Follow IMG menu path SBIW → Data Transfer to the SAP BIW → Settings for Application-Specific Data Sources (CRM).

In this example, I have a Messaging Business Document (mBDocs) because I am not dealing with mobile clients. For mBDocs, BW Adapter extracts objects for CRM business transactions and CRM billing use the BAdI CRM_BWA_MFLOW. BW Adapter calls the mBDocs BAdI CRM_BWA_MFLOW to enhance the partner function Service Delivery Manager (SDM).

In SAP CRM 2005, use transaction code BWA5 to check if the business content is active for your CRM DataSource — in my example, this is 0CRM_SALES_ORDER_I.

### BW Adapter: Version Management: Metadata

<table>
<thead>
<tr>
<th>DataSource</th>
<th>Name</th>
<th>Active</th>
</tr>
</thead>
<tbody>
<tr>
<td>0CRM SALES ACT 1</td>
<td>Activities</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SALES ACT 1</td>
<td>Activity Item</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SALES CONTR 1</td>
<td>CRM Sales Contracts: Item</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SALES ORDER 1</td>
<td>CRM Sales Order Item</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SRY CODES</td>
<td>CRM Service Data for Service Codes (DataSource)</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SRY CONFIRM H</td>
<td>CRM Service Confirmation Header Data (DataSource)</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SRY CONEIRM I</td>
<td>CRM Service Confirmation Item Data (DataSource)</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SRY CONTRACT H</td>
<td>CRM Contract Data (DataSource)</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SRY JBASE ATTR</td>
<td>CRM JBase Master Data (DataSource)</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SRY JBASE TRAN</td>
<td>CRM JBase Transaction Data (DataSource)</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SRY PROCESS H</td>
<td>CRM Service Orders Header Data (DataSource)</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SRY PROCESS I</td>
<td>CRM Service Orders: Item Data (DataSource)</td>
<td>✔️</td>
</tr>
<tr>
<td>0CRM SRY PRM PLAN</td>
<td>CRM Service Data from Planned Orders (DataSource)</td>
<td>✔️</td>
</tr>
</tbody>
</table>

Step 2.3: Check the extraction structure and the BDoc mapping

Use transaction code BWA1 (creates or enhances new CRM-specific Data Sources using the BW Adapter). Enter the CRM DataSource you want to check — in our example, this is 0CRM_SALES_ORDER_I. Click on the Display button to see the extract structure and confirm that it matches our CRM DataSource.
The above figure shows that the BW Adapter knows about the additional field, but it does not yet know how to fill it. In the following steps you show BW Adapter how to fill these fields. The Mapping tab contains the mapping of BDoc fields to the extract structure fields, but you don’t need to take any action here.

**Step 2.4: Create a new BAdI implementation to fill the enhanced field in the append structure**

Object-Oriented (OO) ABAP code in a BAdI is based on the concept of classes, methods, and inheritance, but you do not have to understand this fully to implement a BAdI. Simply think of methods as a function module with the same import and export parameters and follow the simple instructions below to implement the BDoc BAdI.

In transaction SE19 (BAdI implementations), in the Create Implementation section, select Classic BAdI and enter CRM_BWA_MFLOW as the BAdI Name (Figure 8). Click on the Create Impl. button and name the new implementation ZCRM_BWA_MFLOW_IMPL.
After you create ZCRM_BWA_MFLOW_IMPL, click on the Interface tab to see the BAdI implementation with the Interface Name IF_EX_CRM_BWA_MFLOW (Figure 9). The system assigns the method ENHANCE_DATA_SOURCE, which contains the logic to populate the enhanced field.

### Business Add-In Builder: Display Implementation ZCRM_BWA_MFLOW_IMPL

<table>
<thead>
<tr>
<th>Implementation Name</th>
<th>ZCRM_BWA_MFLOW_IMPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Implementation Short Text</td>
<td>BAdI Implementation for BAdI Def CRM_BWA_MFLOW</td>
</tr>
<tr>
<td>Definition name</td>
<td>CRP_BWA_MFLOW</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENHANCE_DATA_SOURCE</td>
<td>ADAP Code Customer-Specific Enhancements for</td>
</tr>
</tbody>
</table>

### Step 2.5: Activate the new BAdI implementation

In transaction SE19,

Click on Enhancement Implementation → Activate to activate ZCRM_BWA_MFLOW_IMPL.

After you implement this BAdI, you can also display your implementation by using transaction code SBIW and following IMG menu path Data Transfer to the SAP Business Information Warehouse → Settings for Application-Specific Data Sources (CRM) → Settings for BW Adapter → BAdI BW Adapter: Enhancement of DataSource in the Messaging Flow.

### Step 2.6: Test the new BAdI implementation in the extractor checker

After you add the new implementation ZCRM_BWA_MFLOW_IMPL, run transaction RSA3. You’ll see the below screen in (without the Debug Mode check box selected).
Execute the extractor checker. If you filter the value to this value SDM,

<table>
<thead>
<tr>
<th>(Number of Recs)</th>
<th>1 (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indx</td>
<td>Cnt</td>
</tr>
<tr>
<td>OTH</td>
<td>100</td>
</tr>
<tr>
<td>OTH</td>
<td>100</td>
</tr>
<tr>
<td>OTH</td>
<td>100</td>
</tr>
<tr>
<td>OTH</td>
<td>100</td>
</tr>
<tr>
<td>OTH</td>
<td>100</td>
</tr>
</tbody>
</table>

If the value is not what you expected — in this example 910001 — select Debug Mode and run the extractor checker again. Set the BAdI breakpoint for debugging at the first stop when you run transaction RSA3.

Now that you have the enhanced field in the extract structure and have populated the value with the BDoc BAdI, it is ready for the BW team to extract to SAP Net Weaver BI. Follow the same process that you would for any other enhancements:

- Replicate the enhanced Data Source in SAP BW
- Add the enhanced Info Object in the data container: Data Store Object or Info Cube
- Map the enhanced fields in the transformation or Info Source
- Load the data

**Conclusion**

Like this we can write all our enhancements in separate methods in the same class. By this we improve performance and flexibility to work on respective enhancements without disturbing other enhancements.
Related Content


For more information, visit the EDW 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.