

Real-time Data Acquisition in SAP BI



Applies to:

SAP NetWeaver BW.

Summary

Detailed description of Real-time data acquisition along with illustrated step-by-step implementation guide

Authors: Shyam Uthaman & Adlin Sundararaj

Company: Accenture Services Pvt. Ltd.

Created on: 10 March 2011

Author Bio



Shyam Uthaman is working as SAP-BI Consultant in Accenture Services Private He is working simultaneously on multiple projects for different clients in Accenture.



Adlin Sundararaj is working as SAP-BI Consultant in Accenture Services Private Ltd and has experience of working on implementation and maintenance projects.

Table of Contents

1.	Introduction	3
2.	Process Flow	3
3.	Converting an Existing Data Flow to Real-Time Data Acquisition	5
4.	Step-by-Step procedure.....	7
5.	Constraints.....	21
	Related Content.....	23
	Disclaimer and Liability Notice.....	24

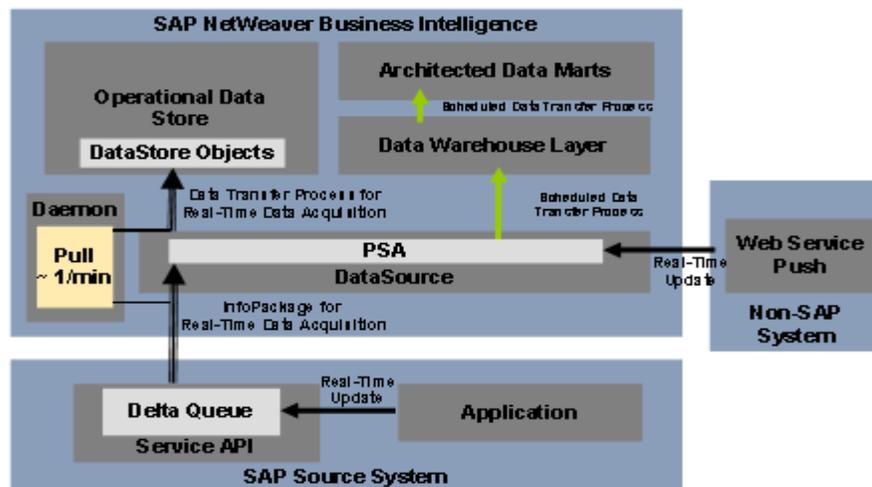
1. Introduction

Real-time data acquisition supports tactical decision-making. It also supports operational reporting by allowing you to send data to the delta queue or PSA table in real-time. You then use a daemon to transfer DataStore objects to the operational DataStore layer at frequent regular intervals. The data is stored persistently in BI.

It is recommended to you use real-time data acquisition if you want to transfer data to BI at shorter intervals (every minute) than scheduled data transfers

2. Process Flow

The following figure illustrates the process flow for real-time data acquisition:



Data is loaded into BI at frequent, regular intervals and is then posted to the DataStore objects that are available for operational reporting.

In BI, special InfoPackages are used for this purpose and data transfer processes for real-time data acquisition are created to further process data from the PSA in the DataStore objects. This is scheduled and executed regularly by a dedicated background process (the daemon).

Data is available for reporting as soon as it has been successfully posted to the DataStore object and activated. Refresh the query display to display the recent data. The query shows the time that the query was last updated by a daemon run, even if no new data was posted.

You can transfer data from the source to the entry layer of BI (the PSA) in two ways:

- **Using a Web service**

You use the Web service to write the data from the source into the PSA. The transfer of data is controlled externally, without a request from BI. Only an InfoPackage (for full upload) is required to determine specific parameters for real-time data acquisition.

- **Using a service API**

Data from an SAP source system can be loaded into the PSA using an InfoPackage created specifically for this purpose. This is triggered when the delta queue in the source system requests data. You have to simulate the initialization of the delta process for the DataSource beforehand.

The following two scenarios are possible:

- The source system application writes the data to the delta queue.

In this case, the daemon retrieves the data without calling the extractor.

- The application does not write data to the delta queue automatically; the extractor writes the data to the delta queue at the request of BI.

For extractors that transfer data synchronously from BI to the service API on request (generic extractors, for example), the daemon calls the extractor, and the extractor writes the data to the delta queue. The data is transferred to BI directly from the delta queue.

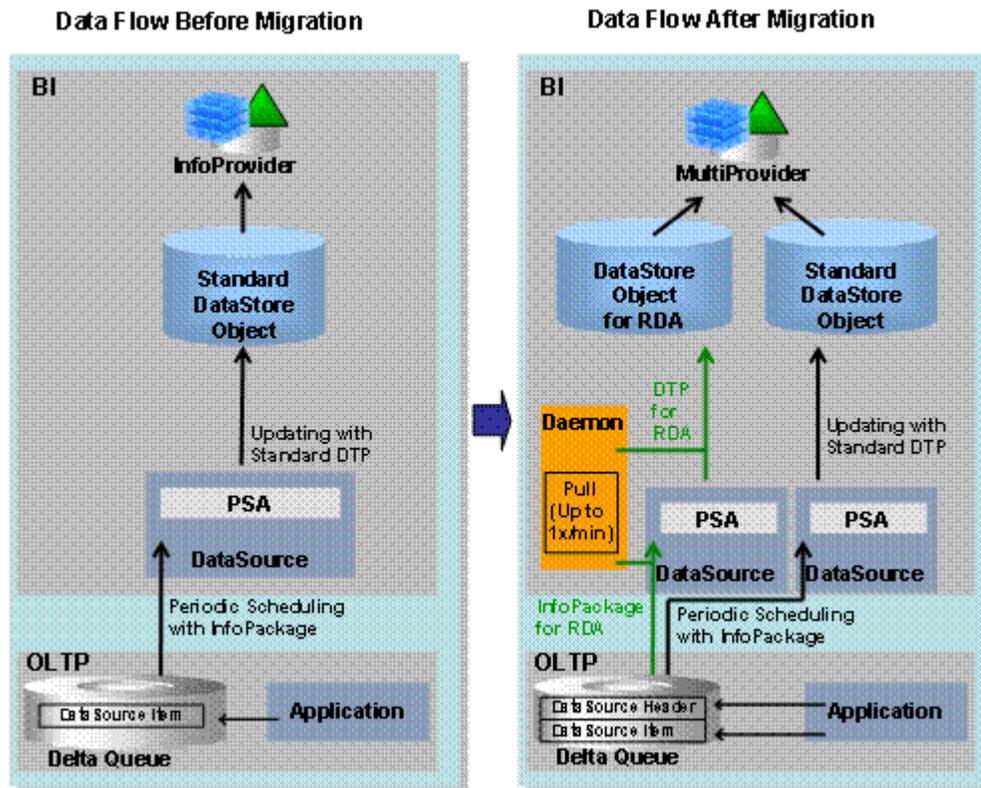
3. Converting an Existing Data Flow to Real-Time Data Acquisition

If you want to integrate the transfer of data with real-time data acquisition into an existing data flow, you have two options:

- **Using two different DataSources**

One DataSource executes the standard data transfer. The other DataSource transfers the data with real-time data acquisition. The data is then combined in a MultiProvider.

The following figure outlines this option:

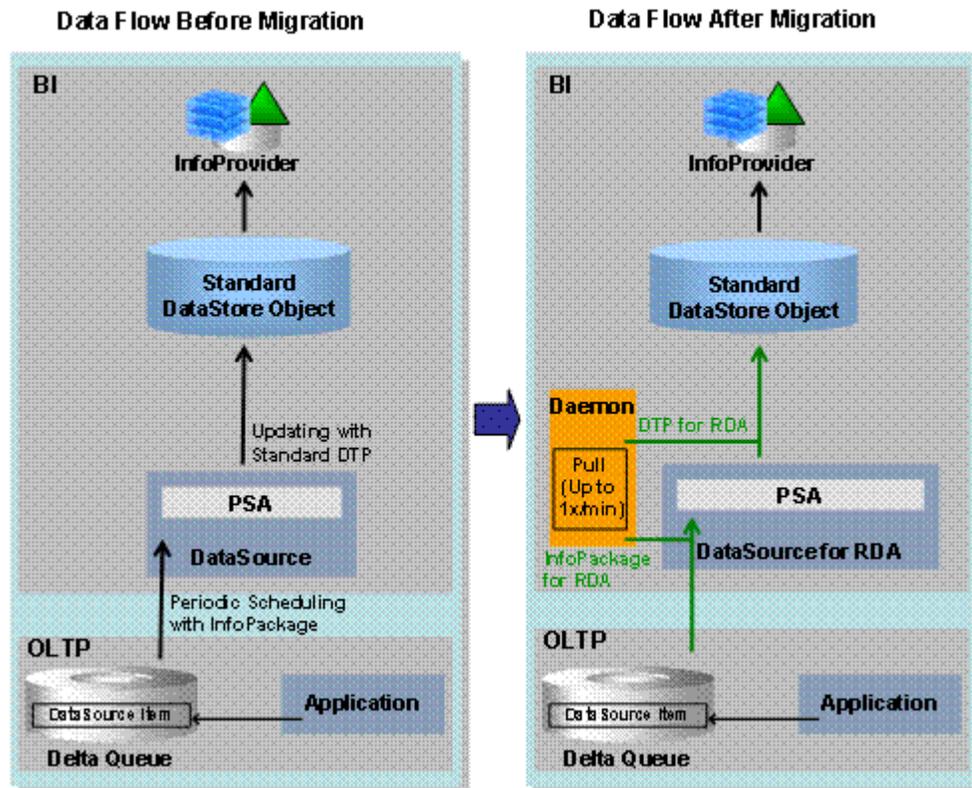


Setting the data flow in this way allows you to continue to use the existing data flow. You can use a separate and possibly smaller DataStore object for the data from the additional DataSource. However, you require a MultiProvider for reporting. To avoid redundant data in the RDA DataStore object, regularly delete the data from the DataStore object once the standard DataStore object is loaded.

- **Using a single DataSource**

You have to replace the standard data transfer completely with a real-time data acquisition scenario.

The following figure outlines this option:



If just one DataSource is to be used, the existing data flow must be fully switched over to real-time data acquisition. This is a result of the fact that a DataSource can have one extraction mechanism only.

If the existing data flow is fully replaced by a new RDA data flow, the data reconciliation for different DataSources does not require any additional administration effort. Switching to this data flow allows you to load larger volumes of data by dividing the load process into several smaller load processes.

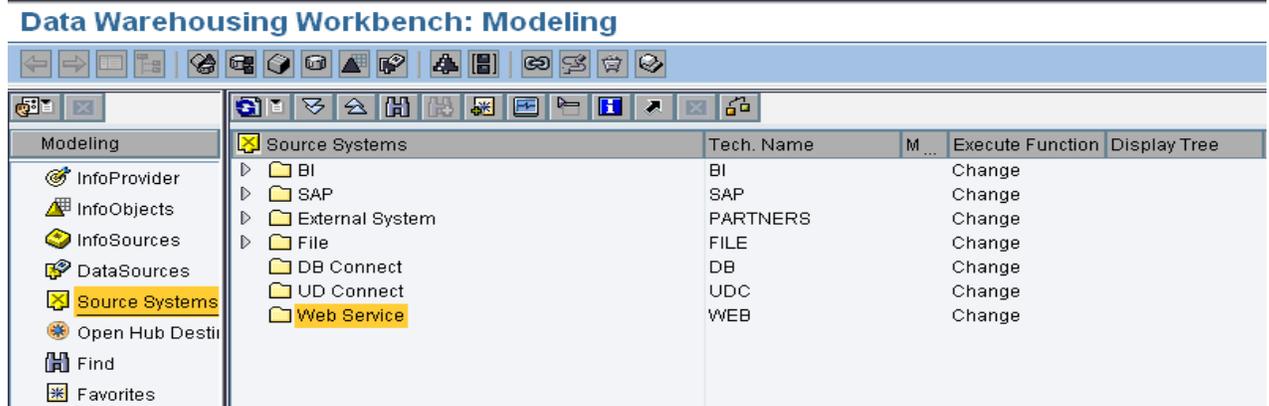
4. Step-by-Step procedure

The source system for RDA could be SAP System or it could be any non-SAP system. SAP is providing most of the Standard Data Sources as real-time enabled. Over here we are implementing this whole process with the help of Web services.

Step 1:

Create Web service Data source in BI system.

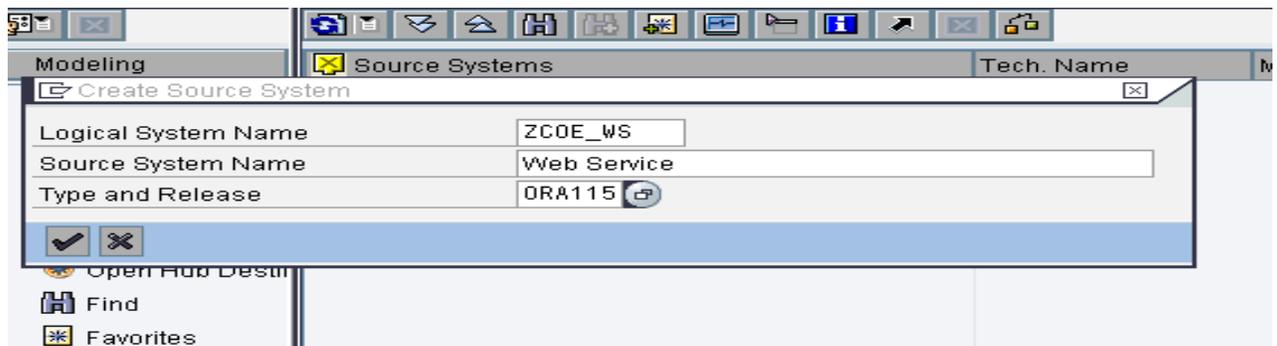
For this go to RSA1 → Source System → web service → Rclick & select create.



Step 2:

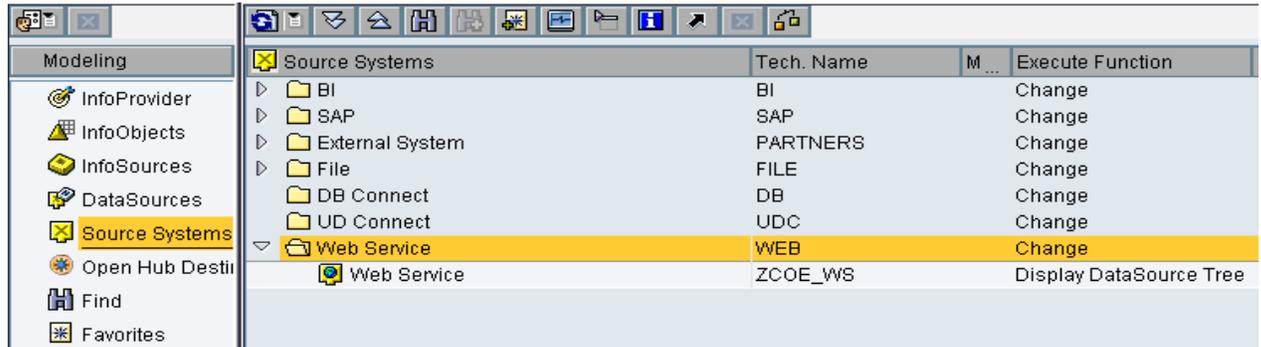
A new window will appear.

Enter the Logical System Name, Source system Name & Type and Release.



Press continue.

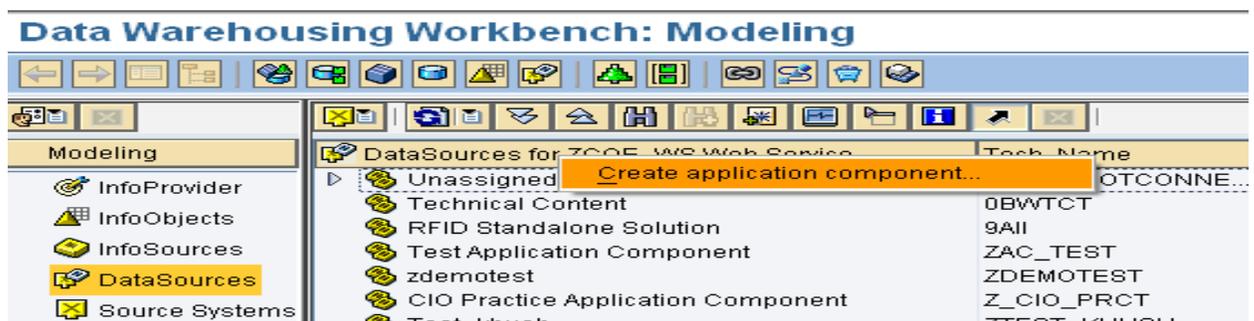
You can see the Source system which we created under the web Service.



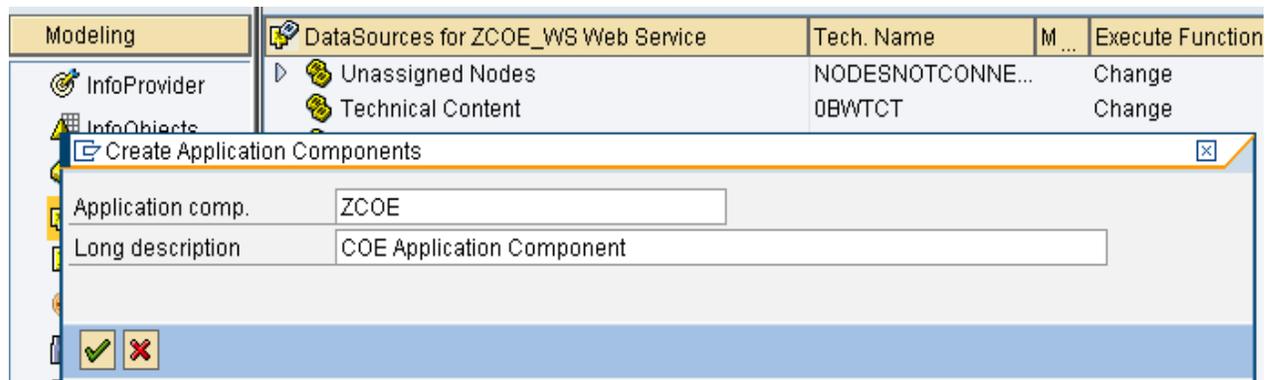
When you Double Click on the source system which we created now, we will be directed to a new window which lists out the data sources for Web Service source systems.

Step 3:

Create Application Component and DataSource for the Web service source system

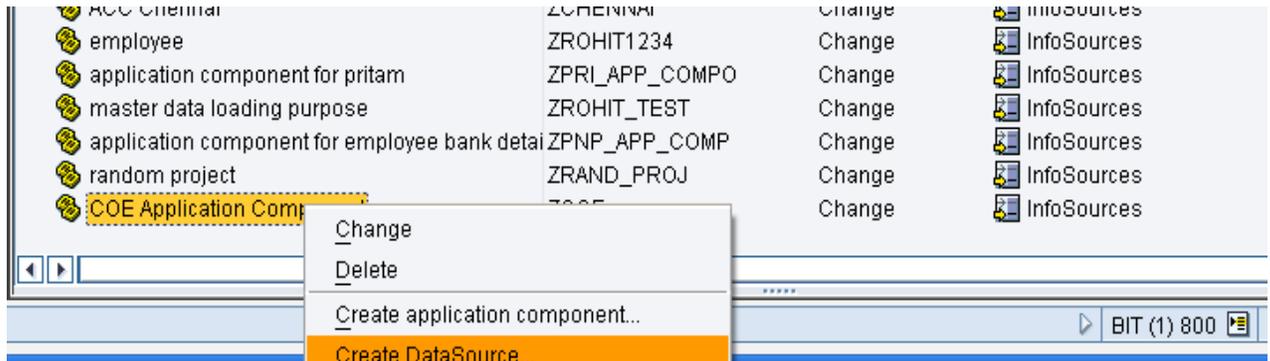


Enter the application Component name and description and Click Continue.

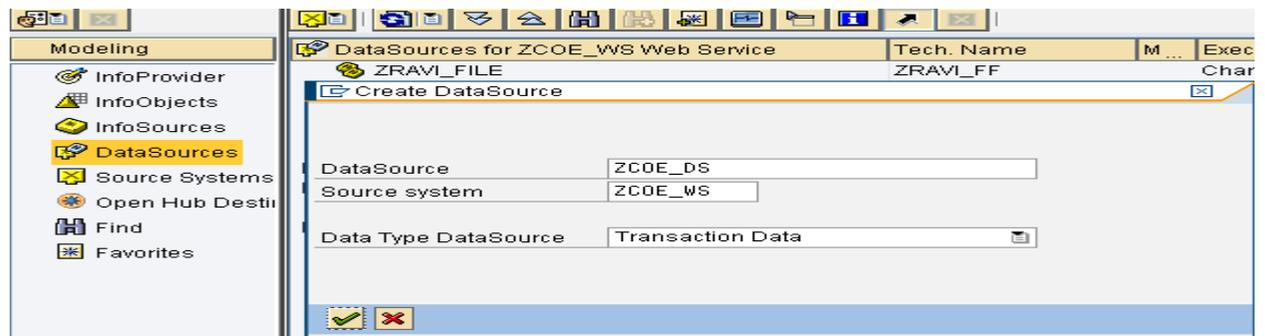


Step 4:

Right click on the Application Component which we created and select Create DataSource

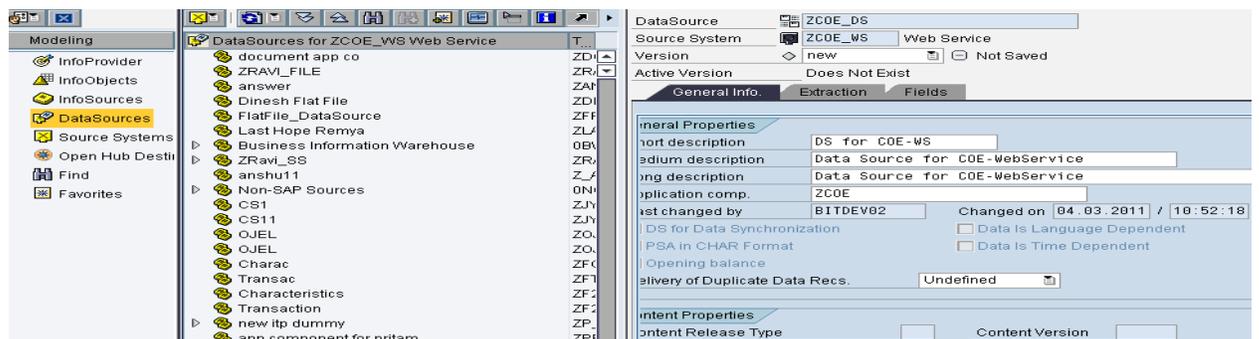


Enter the DataSource Name and Data Type DataSource and press Continue.



E

enter the Description in General Info. Tab of DataSource



Add the fields in the Fields tab of DataSource which you want to display in the DataTarget.

Change DataSource ZCOE_DS(ZCOE_WS)

DataSource: ZCOE_DS DS for COE-WS

Source System: ZCOE_WS Web Service

Version: In Processing Saved Compare with...

Active Version: Executable Edited Version

General Info. Extraction Fields

Pos.	Field	Descript.	D	T	InfoObject	Data type	Length	Decim.	Extern.	L	K	Conv.	Format	SS C.	cur/unit
1	PROD_ID	Product ID	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	@PROD_ID	CHAR	40		40	<input type="checkbox"/>	<input type="checkbox"/>	PRID1	External		
2	PROD_DESCR	Product Desc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	@PROD_DESCR	CHAR	40		40	<input checked="" type="checkbox"/>	<input type="checkbox"/>		Internal		
3	PROD_TYPE	Product Cate	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	@PROD_TYPE	CHAR	2		2	<input type="checkbox"/>	<input type="checkbox"/>		Internal		
4	PRICE	Totl Price, No	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	@PRICE	CURR	17	2	23	<input type="checkbox"/>	<input type="checkbox"/>		Internal		CURRENCY
5	CURRENCY		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	@CURRENCY	CUKY	5		5	<input type="checkbox"/>	<input type="checkbox"/>		Internal		

Save and Activate the DataSource.

After successfully activating the Datasource, go to Extraction tab of DataSource

We can see that the Function Module & Web Service are automatically generated (circled in red below)

Display DataSource ZCOE_DS(ZCOE_WS)

DataSource: ZCOE_DS DS for COE-WS

Source System: ZCOE_WS Web Service

Version: Active Compare with...

Active Version: Executable Edited Version

General Info. Extraction Fields

Delta Process: Full Upload (Delta from InfoPackage Selection Only)

Direct Access: NO DTP Allowed for Direct Access

Real Time: Real-Time Data Acquisition Supported

Adapter: Web Service (Push) Properties

Profile: Basic Auth SOAP Profil

Fixed Service Name:

Fixed DDIC Names:

Function Module: /BIC/CQZCOE_DS00001000

Web Service: /BIC/CQZCOE_DS00001000 Web Service Administration

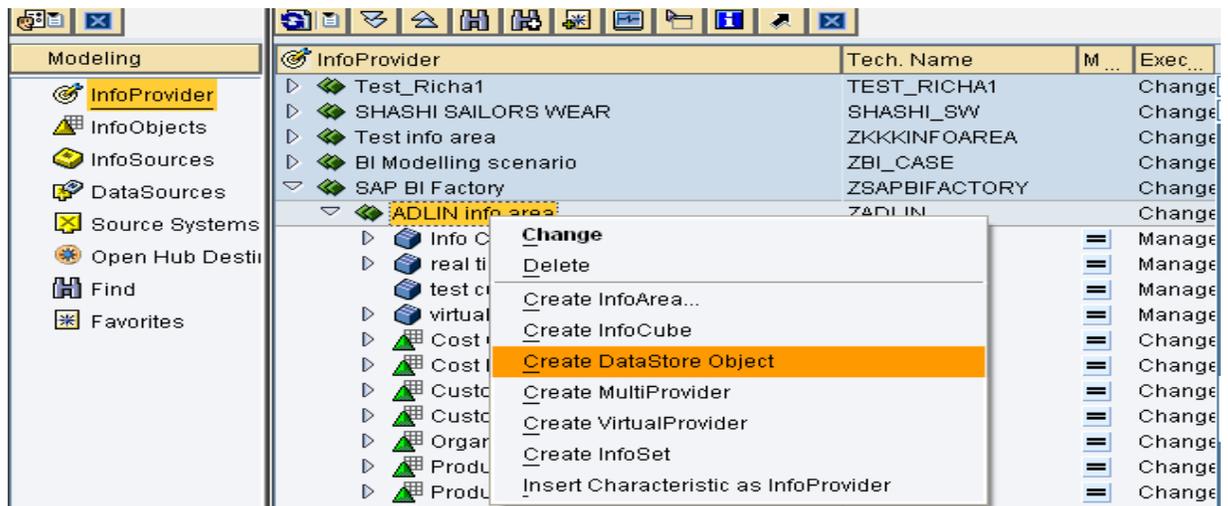
Data Format: Already Binary

Convers. Lang.: User Master Record

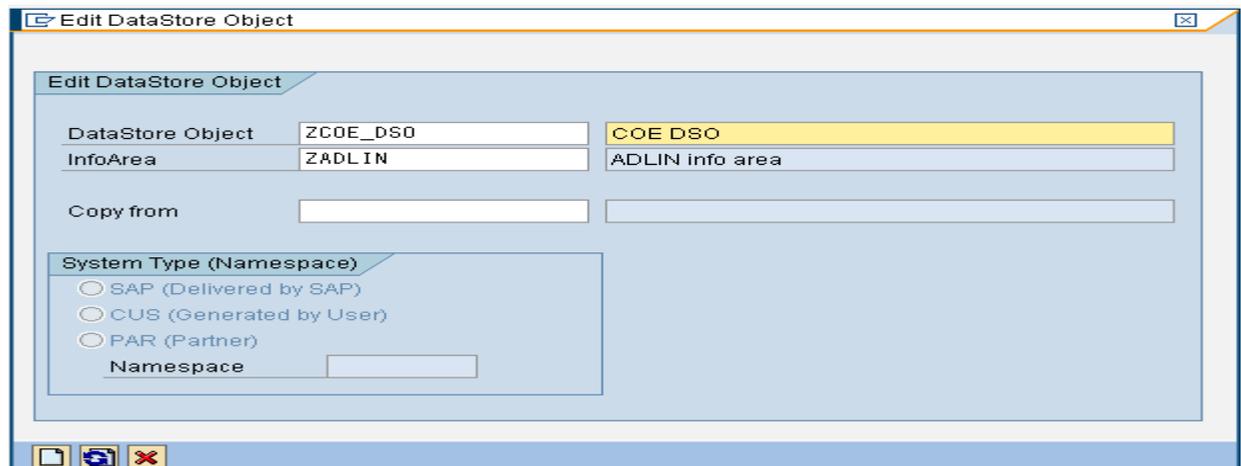
Step 5:

Now Create DataTarget i.e. DSO with the fields which are used in Datasource

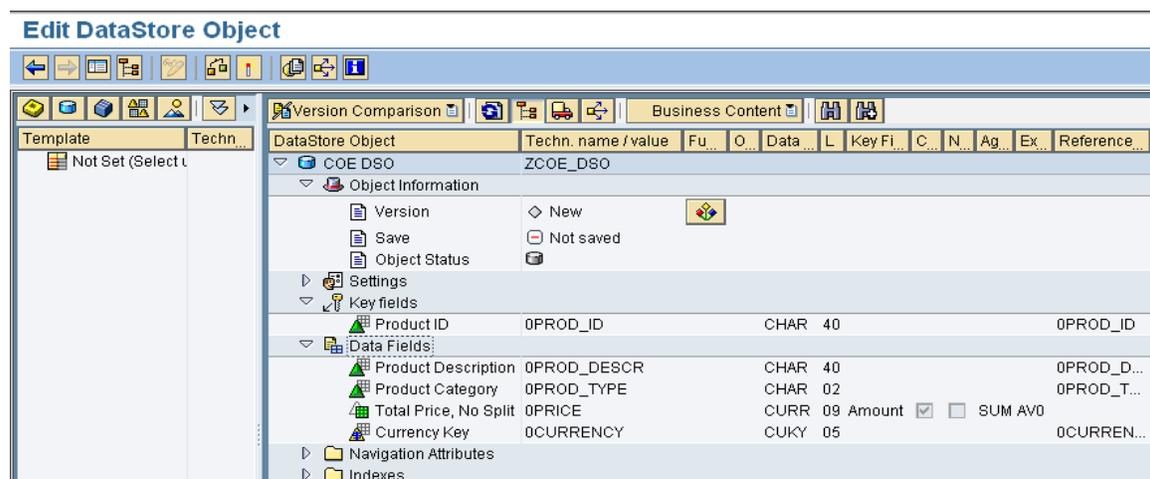
Go to InfoProvider → Right click on InfoArea → Create DSO



Enter the technical Name and description of DSO and click Create.



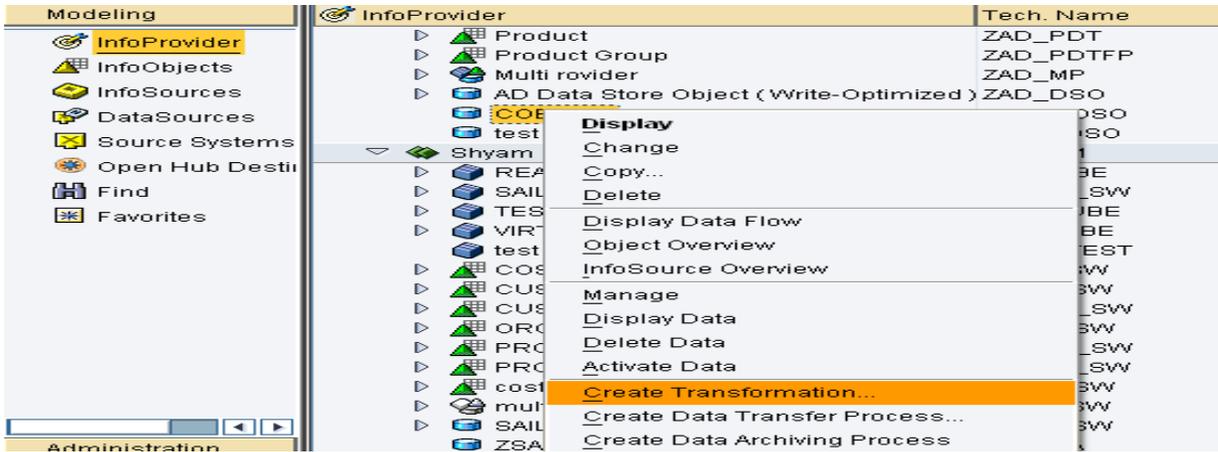
Assign the fields in the DSO as shown below



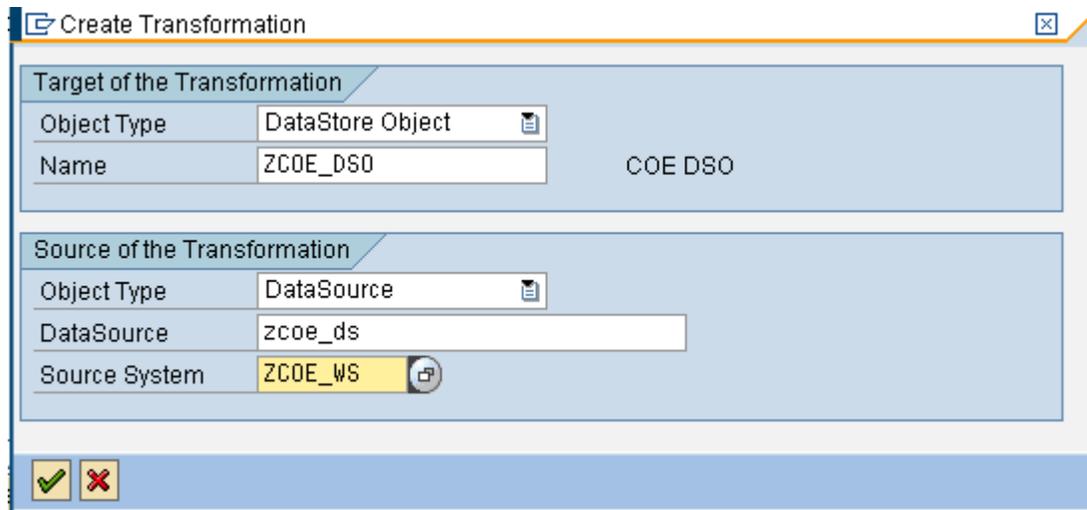
Save and activate the DSO

Step 6:

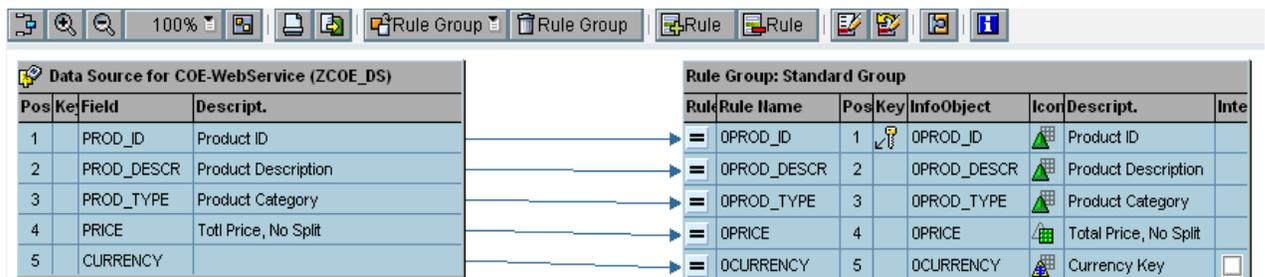
Right click on DSO and create transformation for that DSO.



In the object type give the data source name and the required information.



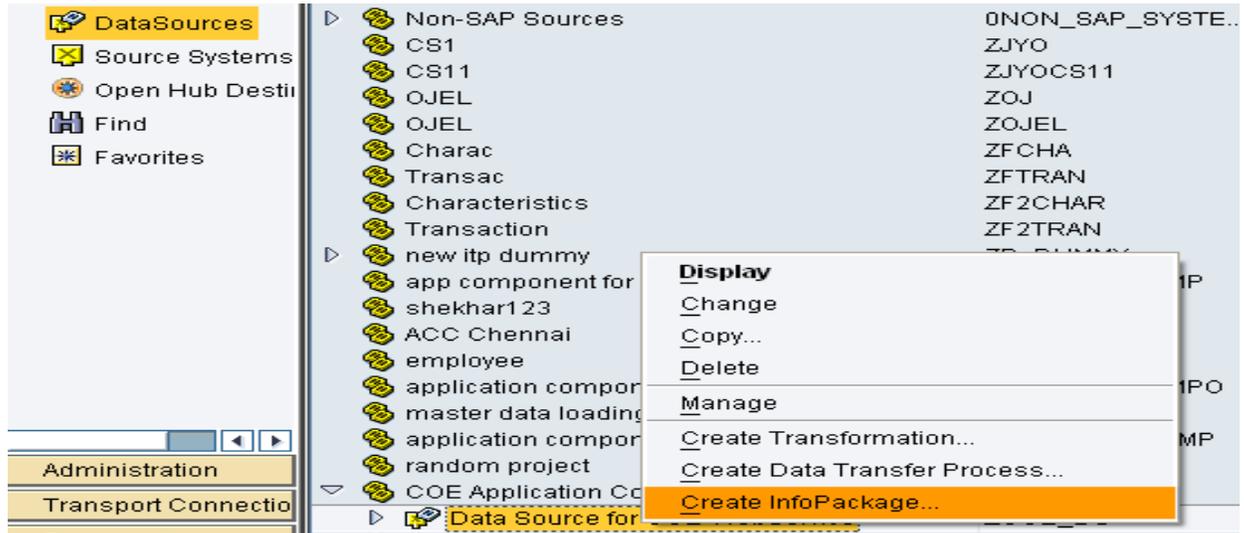
The transformation looks like below.



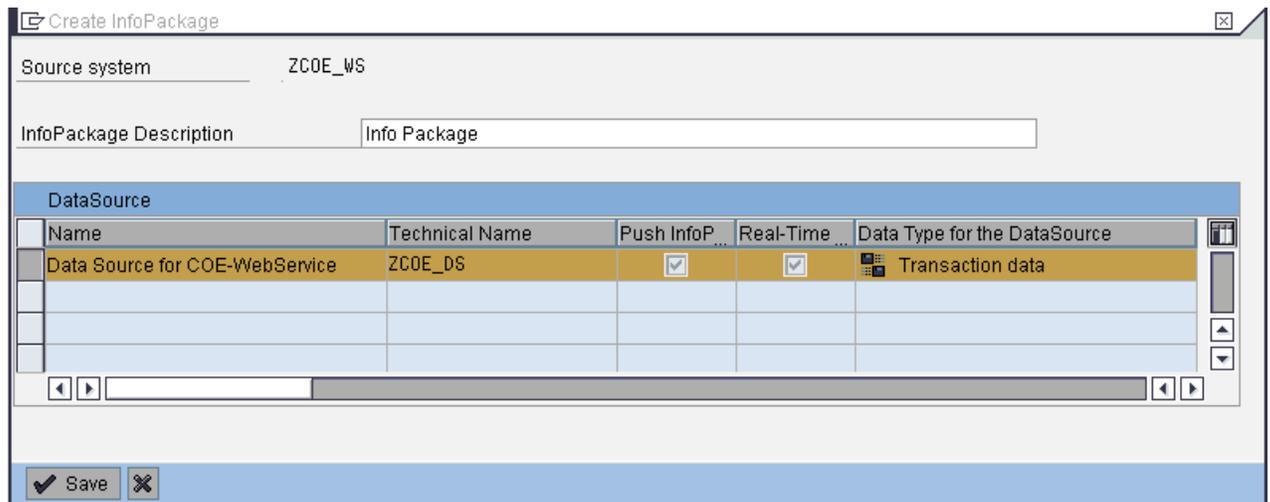
Save and activate the transformation.

Step 7:

Now create InfoPackage for DataSource



Give the description for InfoPackage



In the processing tab, we enter the maximum time for each request to open. Once the limit exceeds, RDA creates new request.

InfoPackage	Info Package(ZPAK_D9ZQJOCOKG6DKXI92Q0ZQZX07)		
DataSource	Data Source for COE-WebService(ZCOE_DS)		
Data Type	Transaction Dat		
Source System	Web Service(ZCOE_WS)		
Last Changed By		Date	Time

Data Selection Extraction **Processing** Schedule

Thresholds for Requests

Automatic Closure of the Request

to Hour(s)

Day(s)

Data Package Size: No. of Rows	10000
Request Size: Number of Rows	100000
Maximum Number of Failed Attempts	100

Step 8:

In the schedule tab we can assign demon. Click on the Assign Demon tab and you will be taken to the RDA monitor tab.

Data Selection Extraction Processing **Schedule**

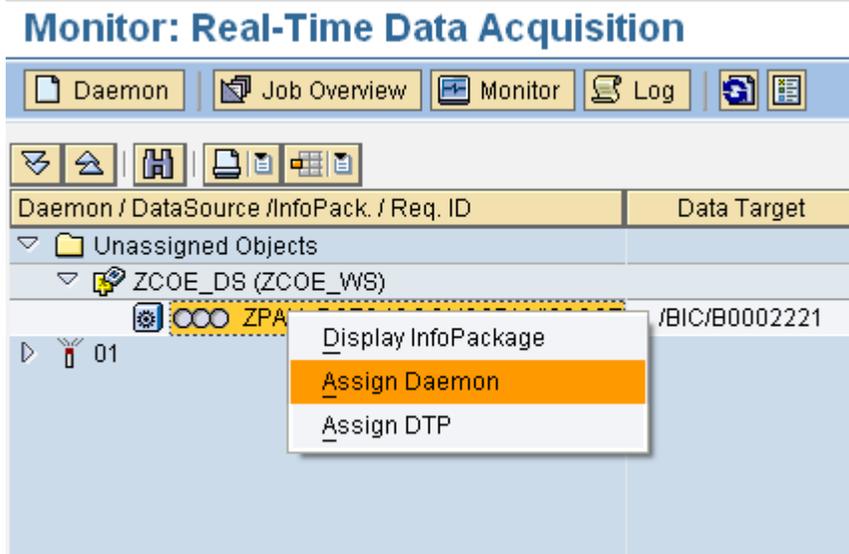
Assign InfoPackage for Real-Time Data Acquisition to Daemon

In this screen you can see that for a particular data source your daemon has been allocated to unassigned nodes automatically.

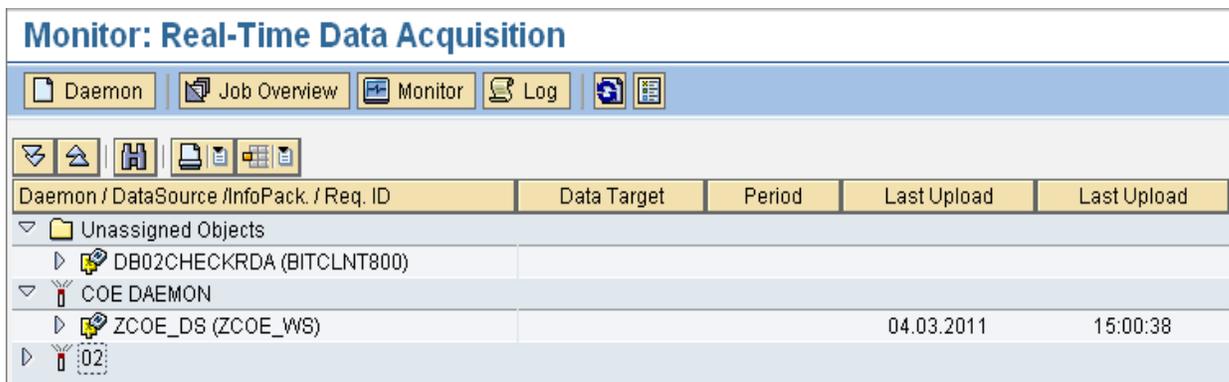
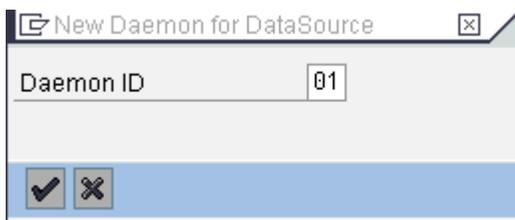
Monitor: Real-Time Data Acquisition

Daemon / DataSource / InfoPack. / Req. ID	Data Target	Period	Last Upload
<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 2px;"> Unassigned Objects </div> <div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 2px;"> ZCOE_DS (ZCOE_WS) </div> <div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 2px;"> OOO ZPAK_D9ZQJOCOKG6DKXI92Q0Z /BIC/B0002221 min </div>			
<div style="border: 1px solid #ccc; padding: 2px;"> 01 </div>			

You need to assign it manually from unassigned node to a new daemon. This is basically required to start the daemon.



Give a daemon id for that. Over here no daemons are already created so I gave it number 1. But you need to number it accordingly to the daemons present in the system.



Assign the DTP for newly created Daemon.

Monitor: Real-Time Data Acquisition

Daemon | Job Overview | Monitor | Log

Unassigned Objects

- 01
 - DB02CHECKRDA (BITCLNT800)
 - ZCOE_DS (ZCOE_WS)
 - ZPAK_D9ZQJOCOKG6DKX192Q0Z /BIC/B0002221 1 min

Context menu for ZCOE_DS (ZCOE_WS):

- Start Upload
- Stop Upload
- DataSource Maintenance
- Assign Daemon
- Delete Assignment
- Assign DTP**
- Job Overview
- Display Log

A new window will open, in that check the DTP to assign.

Assign Data Transfer Processes

RDA Object	InfoProvider	Assign
01		
ZCOE_DS (ZCOE_WS)		
DTP_D9ZR4SEEAV8SMUPFVYGDW2NKN	ZCOE_DSO	<input type="checkbox"/>

After assigning DTP the RDA Monitor screen looks like below

Monitor: Real-Time Data Acquisition

Daemon | Job Overview | Monitor | Log

Unassigned Objects

- 01
 - DB02CHECKRDA (BITCLNT800)
 - ZCOE_DS (ZCOE_WS)
 - ZPAK_D9ZQJOCOKG6DKX192Q0Z /BIC/B0002221 1 min
 - DTP_D9ZR4SEEAV8SMUPFVYGDW2NKN ZCOE_DSO 1 min

Then click on start daemon.

Go to the import tab in the function module

Function Builder: Display /BIC/CQZDS_COE00001000

Function module: /BIC/CQZDS_COE00001000 Active

Attributes Import Export Changing Tables Exceptions Source code

Parameter Name	Type	Associated Type	Default value	Opti	Pas	Short text	Lon...
DATA	TYPE	/BIC/WCQZDS_COE00001000		<input type="checkbox"/>	<input checked="" type="checkbox"/>		

When you double click the associated type , you will reach the below screen

Dictionary: Display Table Type

Table Type: /BIC/WCQZCOE_DS00001000 Active

Short text: DataSource ZCOE_DS (ZCOE_WS)

Attributes Line Type Initialization and Access Key

Line Type: /BIC/CQZCOE_DS00001000

Predefined Type

 Data Type:

 No. of Characters: Decimal Places:

Reference type

Name of Ref. Type:

Reference to Predefined Type

 Data Type:

 Length: Decimal Places:

Then if you double click on the line type you will see the structure of the DSO in the components tab.

Dictionary: Display Structure

Structure: /BIC/CQZCOE_DS00001000 Active

Short Description: DataSource ZCOE_DS (ZCOE_WS)

Attributes Components Entry help/check Currency/quantity fields

Predefined Type 1 / 6

Component	RTy	Component type	Data Type	Length	Decim	Short Description
INCLUDE:	<input type="checkbox"/>	/BIC/CQZCOE_DS00001000		0	0	DataSource ZCOE_DS (ZCOE_WS)
PROD_ID	<input type="checkbox"/>		CHAR	40	0	
PROD_DESCR	<input type="checkbox"/>		CHAR	40	0	
PROD_TYPE	<input type="checkbox"/>		CHAR	2	0	
PRICE	<input type="checkbox"/>		CURR	17	2	
CURRENCY	<input type="checkbox"/>		CUKY	5	0	Totl Price, No Split

Step 10:

Create an RFC in se37 and the code is given below.

Function Builder: Change ZCOE_EST

```

1  FUNCTION ZCOE_EST.
2  *-----
3  *""Local Interface:
4  *-----
5  data:lt_data type table of /BIC/CQZCOE_DS000010000001,
6       l_data type /BIC/CQZCOE_DS000010000001.
7
8  l_data-PROD_ID      = 101.
9  l_data-PROD_DESCR  = 'sony-vaio'.
10 l_data-PROD_TYPE   = 'computers'.
11 l_data-PRICE       = 50000.
12 l_data-CURRENCY    = 'INR'.
13
14 append l_data to lt_data.
15 CALL FUNCTION '/BIC/CQZCOE_DS00001000'
16   EXPORTING
17     DATA          = lt_data.
18 * EXCEPTIONS
19 *   INTERNAL_ERROR = 1
20 *   OTHERS        = 2
21 *
22 IF SY-SUBRC <> 0.
23 * MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
24 *           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
25 ENDIF.
26 ENDFUNCTION.

```

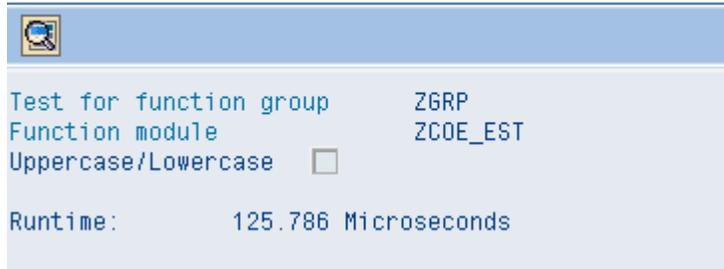
Here we have only used the fields which were used in the DSO and passed the fields some values via hard coding and then collected those values in the internal table lt_data and passed that itab in the function module.

Now execute this RFC

Test Function Module: Initial Screen

Test for function group	ZGRP
Function module	ZCOE_EST
Uppercase/Lowercase	<input type="checkbox"/>

Test Function Module: Result Screen



Test for function group ZGRP
 Function module ZCOE_EST
 Uppercase/Lowercase

Runtime: 125.786 Microseconds

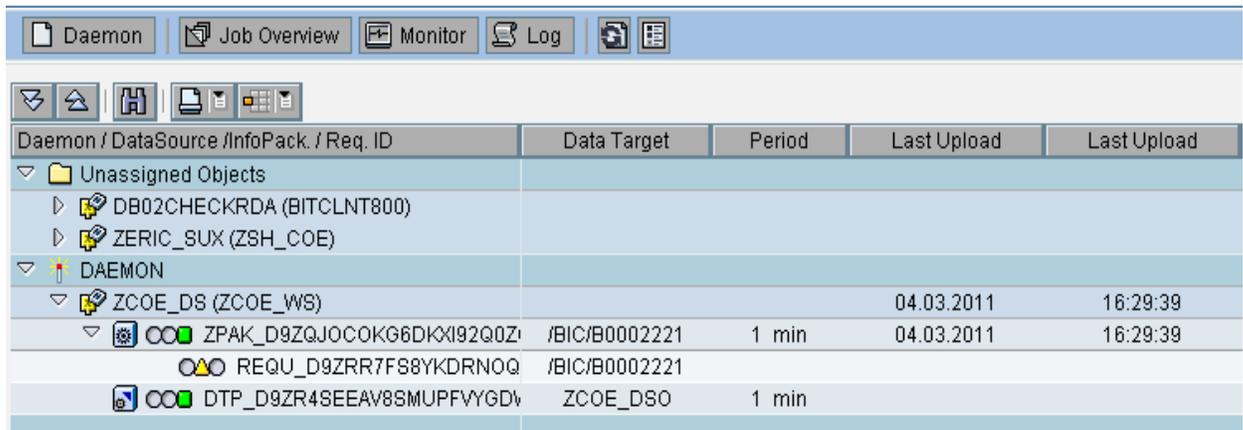
Step 11:

Click on start demon by right clicking on Daemon name.

The record comes to PSA and it will show the open request for DTP.

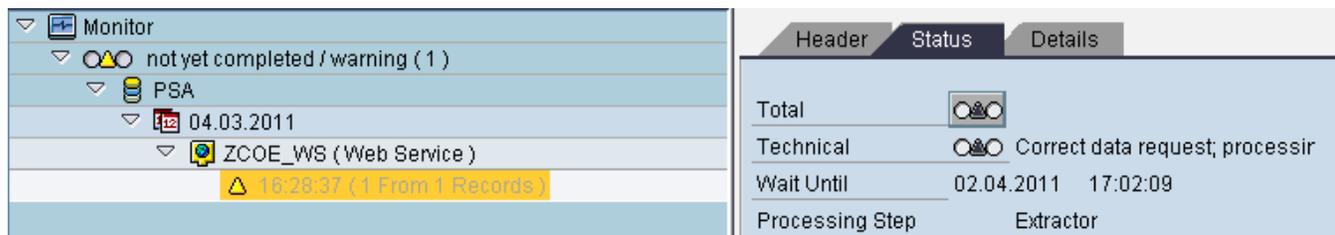
Now when you start its status will become green

Monitor: Real-Time Data Acquisition



Daemon / DataSource / InfoPack / Req. ID	Data Target	Period	Last Upload	Last Upload
Unassigned Objects				
DB02CHECKRDA (BITCLNT800)				
ZERIC_SUX (ZSH_COE)				
DAEMON				
ZCOE_DS (ZCOE_WS)			04.03.2011	16:29:39
ZPAK_D9ZQJOCOKG6DKXI92Q0Z	/BIC/B0002221	1 min	04.03.2011	16:29:39
REQU_D9ZRR7FS8YKDRNOQ	/BIC/B0002221			
DTP_D9ZR48EEAV8SMUPFVYGDV	ZCOE_DSO	1 min		

When you open the monitor, you can find that 1 record is loaded.



Header	Status	Details
Total		
Technical		Correct data request; processir
Wait Until	02.04.2011 17:02:09	
Processing Step	Extractor	

Monitor
 not yet completed / warning (1)
 PSA
 04.03.2011
 ZCOE_WS (Web Service)
 16:28:37 (1 From 1 Records)

Now you can check the data in DSO which is updated in the DSO.

And you can stop the daemon although it will continuously run and when the request comes then it will load the data.

Monitor: Real-Time Data Acquisition

Daemon / DataSource / InfoPack. / Req. ID	Data Target	Period
Unassigned Objects		
DB02CHECKRDA (BITCLNT800)		
ZERIC_SUX (ZSH_COE)		
DAEMON		
ZC	XI92Q0Z /BIC/B0002221	1 min
ZD	KDRNOQ /BIC/B0002221	
ZD	PFVYGDV ZCOE_DSO	1 min

5. Constraints

- You can only use real-time data acquisition to fill DataStore objects. A two-step data transfer is supported; data is first transferred into the PSA and then into the DataStore object. The DataStore object cannot be used as the source for a further real-time data transfer to another DataStore object.
- Master data cannot be transferred to the BI system with real-time data acquisition as otherwise the navigation attributes of the characteristic could no longer be used in aggregates. This is because aggregates cannot react to real-time updates since the change run in the RDA process cannot be triggered automatically for the loaded data.
- DataSources that are used for real-time data acquisition cannot be used in the delta process for standard data transfer (scheduled staging). A data transfer with RDA and a scheduled data transfer cannot be executed simultaneously in the delta process for a DataSource because there may be only one entry in the delta queue for each DataSource and target system.

- If you load data into a DataStore object with real-time data acquisition, you cannot load data into this DataStore object simultaneously with an additional DTP. This is because there can be only one open activation request in a DataStore object. Real-time data acquisition keeps an activation request open parallel to each DTP request. In a DTP request, multiple data packages can be loaded during a given time span. Each data package is activated in the DataStore object immediately after it is transferred. A further data transfer process cannot load into the same DataStore object as long as an activation request is open

Related Content

<https://forums.sdn.sap.com/thread.jspa?threadID=1011993>

<http://forums.sdn.sap.com/thread.jspa?threadID=227411>

<http://forums.sdn.sap.com/thread.jspa?threadID=952755>

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.