Enable Trend Reporting - by Manipulating Variable Value

Applies to:

SAP BI

Summary

Scenario is pretty simple, **Trend Analysis** – User enters **"Key Date"** and is interested in analyzing Key figure values under various heads. **For example user enters "Current Date" and is interested in looking at the "Sale on Entered Date", "Sales for Current Month to Date", "Sale for Current Year to Date" and also these values for previous year and many more.** Scenario presented here takes you step by step (including ABAP Code) to enable this kind of reporting.

Word of Caution – General advice is to explore business content delivered variables first & if no relevant variable exists then look forward to create this kind of variables (Customer Exit ones).

Author: **Vikash Agrawal**

Company: **Wipro Technologies**

Created on: 25 June 2006

Author Bio

Vikash Agrawal is a Certified SAP BI Consultant **with Wipro Technologies**. He has extensive cross functional experience and has been with end to end SAP implementations, upgrades, and BI - Portal Integration projects across manufacturing domain. He has also worked in the area of distribution of finished goods as well as remodeling of Distribution Supply Chain.

Table of Contents

Overview

This particular assignment was meant for Senior Sales Manager, who was interested in looking at the "Tree" (almost current status) and "Jungle" (longer horizon) at the same time. Needs were to analyze the following analytics on "Key Date". "Key Date" is decided at the runtime. Few of the analytics covered were"

- Sales on Day – On the "Key Date" decided at the run time (User Entry).

- Month to Date (MTD) – From the 1st of month to "Key Date" - for current year.

- Average Month to Date – MTD divided by the "No of Days" in month till "Key Date" for current year.

- Previous Year Month to Date – MTD for the Previous Year.

- Average Previous Year Month to Date – Average MTD for the Previous Year.

- Year to Date (YTD) – From the 1st to entered "Key Date" for the current year.

- Average Year to Date - YTD divided by the "No of Days" in year till "Key Date" for current year.

- Previous Year to Date – YTD for the Previous Year.

- Average Previous Year to Date – Average YTD for the Previous Year

Approach to Analytics Needs

- Task is very well cut out; need to manipulate "Key Date" values to achieve required analytics.

- As a first step, create a variable on "Sales Date" (or whichever date is used for Sales Date) field (ZPDATE) with user input. It was meant to capture "Key Date" through user entry.

- Create a restricted key figure on Sales & restrict it on "Key Date" value to get first analytics need i.e. "Sales on Day".

- Next step was to manipulate the value of "Key Date" on which I could restrict "Sales" (Key Figure) again and again to get other values.

- Create a new variable (e.g. ZPUTMNTH) and assigned it to be filled up by Customer Exit. (**For simplicity purposes only "Month to Date" value is explained below. Same procedure can be repeated to fulfill other needs. Though in code section, ABAP code is available for almost all the variables**)

Create Variable

- Create one variable (ZPUTMNTH) for "Month to Date" with processing by "Customer Exit". This variable was created on "Date" characteristics.
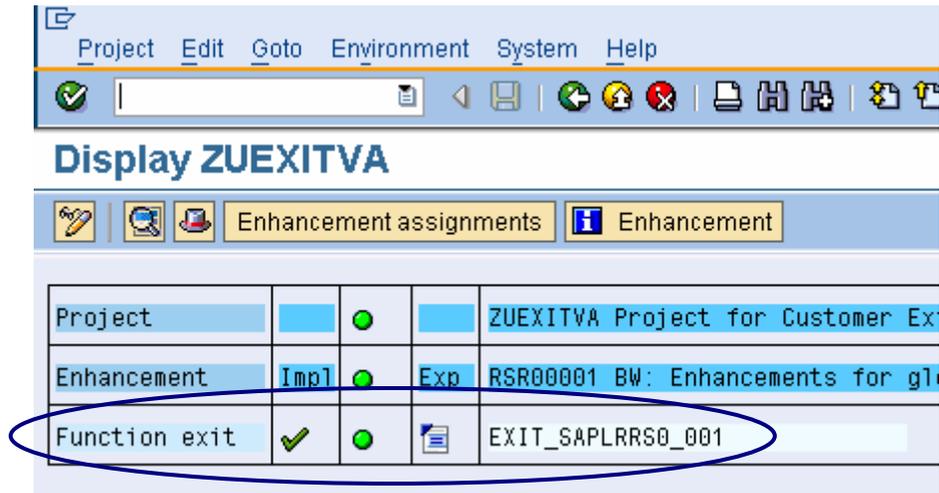
Fill up the Variable

Now after creating the Customer Exit processing type variable (ZPUTMNTH) next task was to manipulate values of it with User Input ("Key Date"). One can follow these steps:

- Go to Tcode "CMOD" and enter "Project Code" to be used and go to"Components".
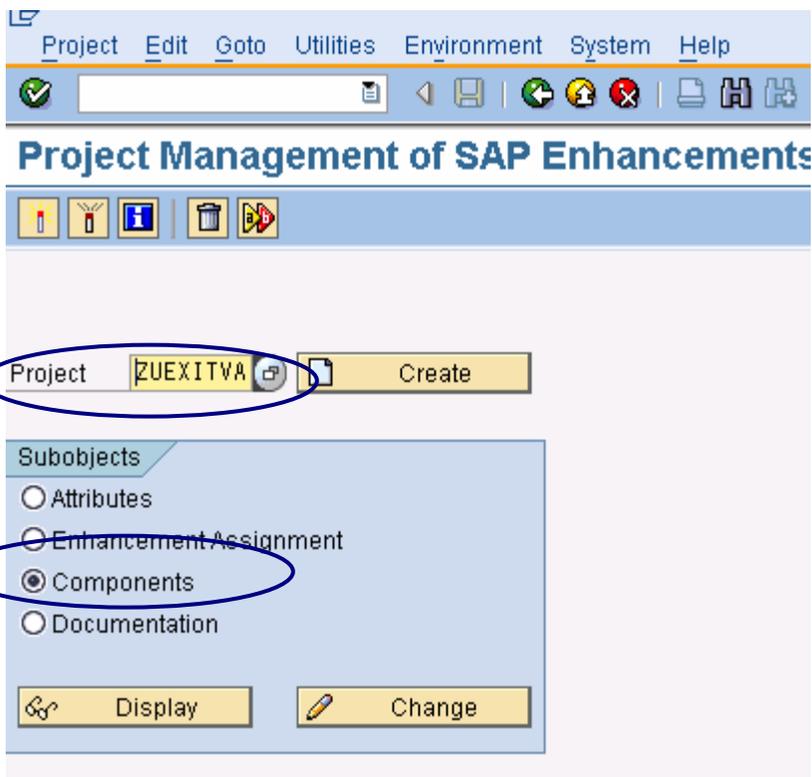
- Double Click on EXIT_SAPLRRS0_001

- Double Click on INCLUDE ZXRSRU01 (if double click on mouse throws some error put cursor on include name and press enter twice).

- It shall take you to ABAP editor, to write ABAP code to fill up value of Customer Exit variable (ZPUTMNTH). Below is Include with Code.

Sample ABAP Code

```
*&---------------------------------------------------------------------*
*&  Include           ZXRSRU01
*&---------------------------------------------------------------------*

DATA: L_S_RANGE TYPE RSR_S_RANGESID.
DATA: LOC_VAR_RANGE LIKE RRRANGEEXIT.
Data: v_loc_date type d.
Data: v_loc_date1 type d.
Data: v_loc(4) type c.

CASE I_VNAM.

*-----------------------------------------------------------
* to calculate "month to date" user input is "Calday" Key Date
*-----------------------------------------------------------
WHEN 'ZPUTMNTH'.
IF I_STEP = 2. "after the popup
```

```
LOOP AT I_T_VAR_RANGE INTO LOC_VAR_RANGE

WHERE VNAM = 'ZPDATE'.
CLEAR L_S_RANGE.
L_S_RANGE-LOW = LOC_VAR_RANGE-LOW(6). "low value, e.g.YYYYMM (200606) part of key date
(20060625)
L_S_RANGE-LOW+6(2) = '01'.              " low value e..g. YYYYMM01 (20060601)
L_S_RANGE-HIGH = LOC_VAR_RANGE-LOW. "high value = input
L_S_RANGE-SIGN = 'I'.
L_S_RANGE-OPT = 'BT'.
APPEND L_S_RANGE TO E_T_RANGE.
EXIT.
ENDLOOP.
ENDIF.


*-------------------------------------------------------------------
* to Calculate no of days up to "month to date" (user input Calday Key date)
*-------------------------------------------------------------------
WHEN 'ZDAYMTH'.
IF I_STEP = 2. "after the popup

LOOP AT I_T_VAR_RANGE INTO LOC_VAR_RANGE

WHERE VNAM = 'ZPDATE'.
CLEAR L_S_RANGE.
L_S_RANGE-LOW = LOC_VAR_RANGE-LOW+6(2). "YYYYMMDD - DD will indicate the no of days
L_S_RANGE-SIGN = 'I'.
L_S_RANGE-OPT = 'EQ'.
APPEND L_S_RANGE TO E_T_RANGE.
EXIT.
ENDLOOP.
ENDIF.
*-------------------------------------------------------------------
* to calculate Previous year "month to date" (User input Calday – Key date)
*-------------------------------------------------------------------

WHEN 'ZDAYPMTH'.
IF I_STEP = 2. "after the popup

LOOP AT I_T_VAR_RANGE INTO LOC_VAR_RANGE

WHERE VNAM = 'ZPDATE'.
CLEAR L_S_RANGE.

L_S_RANGE-LOW = LOC_VAR_RANGE-LOW.
L_S_RANGE-LOW(4) = L_S_RANGE-LOW(4) - 1.  "low value previos year
v_loc_date = L_S_RANGE-LOW.                "take input value in another variable
L_S_RANGE-LOW+6(2) = '01'.
L_S_RANGE-HIGH = v_loc_date.               "high value = input, taken into another
variable
L_S_RANGE-SIGN = 'I'.
L_S_RANGE-OPT = 'BT'.
APPEND L_S_RANGE TO E_T_RANGE.
EXIT.
ENDLOOP.
ENDIF.


*-----------------------------------------------------------
* to calculate "year to date" (User input Calday – key date)
*-----------------------------------------------------------

WHEN 'ZPUTYEAR'.
IF I_STEP = 2. "after the popup
```

```
LOOP AT I_T_VAR_RANGE INTO LOC_VAR_RANGE

WHERE VNAM = 'ZPDATE'.
CLEAR L_S_RANGE.
L_S_RANGE-LOW = LOC_VAR_RANGE-LOW(4)."low value, e.g.200001
L_S_RANGE-LOW+4(4) = '0101'.
L_S_RANGE-HIGH = LOC_VAR_RANGE-LOW. "high value = input
L_S_RANGE-SIGN = 'I'.
L_S_RANGE-OPT = 'BT'.
APPEND L_S_RANGE TO E_T_RANGE.
EXIT.
ENDLOOP.
ENDIF.
*----------------------------------------------------------------
* to calculate "Previous year to date" (User input Calday – key date)
*----------------------------------------------------------------

WHEN 'ZDAYPYR'.
IF I_STEP = 2. "after the popup

LOOP AT I_T_VAR_RANGE INTO LOC_VAR_RANGE

WHERE VNAM = 'ZPDATE'.
CLEAR L_S_RANGE.

L_S_RANGE-LOW = LOC_VAR_RANGE-LOW.
L_S_RANGE-LOW(4) = L_S_RANGE-LOW(4) - 1.  "low value previos year
v_loc_date = L_S_RANGE-LOW.               "take input value in another variable
L_S_RANGE-LOW+4(4) = '0101'.
L_S_RANGE-HIGH = v_loc_date.              "high value = input, taken into another
variable
L_S_RANGE-SIGN = 'I'.
L_S_RANGE-OPT = 'BT'.
APPEND L_S_RANGE TO E_T_RANGE.
EXIT.
ENDLOOP.
ENDIF.

*----------------------------------------------------------------------
* to calculate "Previous year" User input Calday – key date
*----------------------------------------------------------------
WHEN 'ZPYEAR'.
IF I_STEP = 2. "after the popup

LOOP AT I_T_VAR_RANGE INTO LOC_VAR_RANGE

WHERE VNAM = 'ZPDATE'.
CLEAR L_S_RANGE.
v_loc = LOC_VAR_RANGE-LOW(4) - 1.
L_S_RANGE-LOW = v_loc.  "low value previos year
L_S_RANGE-SIGN = 'I'.
L_S_RANGE-OPT = 'EQ'.
APPEND L_S_RANGE TO E_T_RANGE.
EXIT.
ENDLOOP.
ENDIF.

*----------------------------------------------------------
* to calculate No of Days - "year to date"
*----------------------------------------------------------
WHEN 'ZYRDAY'.
IF I_STEP = 2. "after the popup
LOOP AT I_T_VAR_RANGE INTO LOC_VAR_RANGE
WHERE VNAM = 'ZPDATE'.
```

```
CLEAR L_S_RANGE.
v_loc_date1 = LOC_VAR_RANGE-LOW.
v_loc_date = LOC_VAR_RANGE-LOW(4).   "low value, e.g.200001
v_loc_date+4(4) = '0101'.

L_S_RANGE-LOW = v_loc_date1 - v_loc_date.              "get the difference between
current date & first date of year
L_S_RANGE-SIGN = 'I'.
L_S_RANGE-OPT = 'EQ'.
APPEND L_S_RANGE TO E_T_RANGE.
EXIT.
ENDLOOP.
ENDIF.
```

- Save and Activate the Include.

- Go back to Exit and activate again.

- Go back to Project and activate again.

- Variable (ZPUTMNTH) is ready (its values filled up with the manipulation of "Key Date") to be used in query to restrict Key Figure "Sales" on it and get "Month to Date" (MTD) Sales Value.

- Same procedure can be repeated for other Key Figures like YTD etc. ABAP Code for those is also available in ABAP Code section.

Start Using the Variable

Query Output

Key Date – User Entry
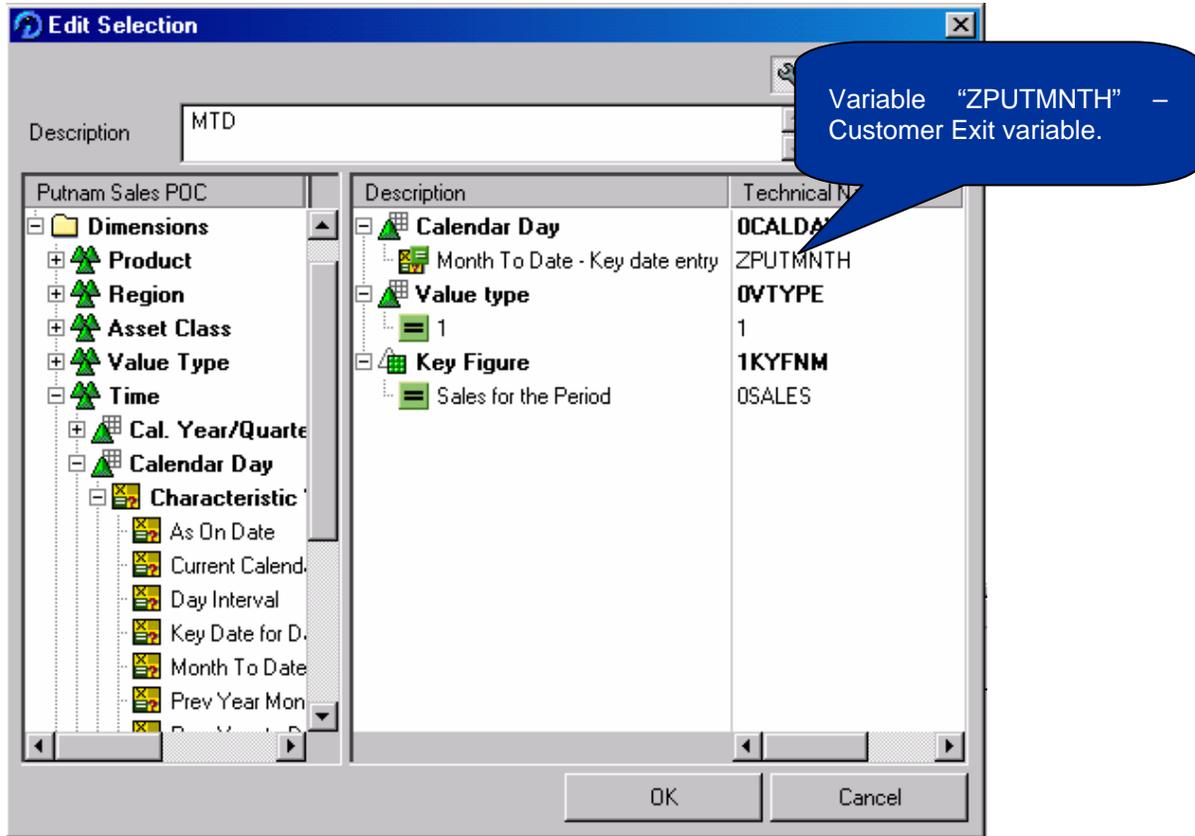
Variables for Ad-Hoc Report

As On Date (*) 19.01.2006   19.01.2006

Execute Check

Exceptions and Conditions | Notes | Export to Excel | Export to CSV

Asset class	Sales on Day	Sales last 5 days	5 days Average	MTD- Daily Average
GROWTH	$ 7.000,00	$ 10.500,00	$ 2.100,00	$ 552,63
VALUE	$ 7.000,00	$ 10.500,00	$ 2.100,00	$ 552,63
BLEND	$ 7.000,00	$ 10.500,00	$ 2.100,00	$ 552,63

Related Content

1. Enterprise Reporting & Query Analysis -

https://www.sdn.sap.com/irj/sdn/docs?rid=/webcontent/uuid/d5db5d85-0a01-0010-bda5-c5834821a1a5

2. Variables in SAP BI – Business Explorer

http://help.sap.com/saphelp_nw04/helpdata/en/f1/0a56a7e09411d2acb90000e829fbfe/frameset.htm

3. Customer Exit Variables

http://help.sap.com/saphelp_nw04/helpdata/en/f1/0a56f5e09411d2acb90000e829fbfe/frameset.htm

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.