

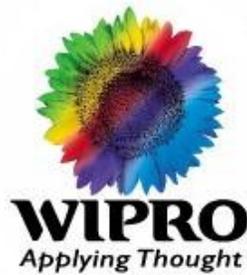
**Best-Practices & Learning's**

**ON**

**SAP ABAP**

**By**

**Prammendran**



Prammenthiran.rajendran@wipro.com

Jan 5, 2010

Version- 1.0 - 5<sup>th</sup> Jan 2010 - Initial Document

## Introduction:

This document has been prepared based on the lessons learnt from the Bacardi implementation project (V1). This can also be used as Best Practice in SAP ABAP programming. It covers the common mistakes while developing an ABAP object (From Junior to senior level).

## Scope

This document is intended for ABAP developers from Junior to senior level.

## Best-Practices/Learning

- Danger of using "CHECK" statement in the user exits which has multiple enhancements:-

CHECK statement will terminate the control from the current subroutine. So, the control will not go to the next enhancement of the same user exit.

For example, We have multiple enhancements in the user exit subroutine "USEREXIT\_MOVE\_FIELD\_TO\_VBAK" Of the program "MV45AFZZ" as below,

```
FORM USEREXIT_MOVE_FIELD_TO_VBAK.
*****
*$$$-Start: (2)-----
ENHANCEMENT 1 ZOBILLING_BLOCK.      "active ver

    include ZONO_BILL_BLOCK.
*    perform f_bill_block(ZONO_BILL_BLOCK).

ENDENHANCEMENT.
ENHANCEMENT 1 ZO_SALES_ORDER_BLOCK_ITEM_DEL.
```

```
ENHANCEMENT 1 ZI_CARRIER_FORWARDING.      "active version
*-----*
* Description          : Carrier and forwarding agent needs to be automatically
*                      populated in sales order.
*
```

```
SELECT SINGLE zcarr
FROM zomaster_cntrl
INTO ws_check
WHERE zauart = xvbak-auart
AND zvkorg = xvbak-vkorg
AND zcarr = c_y.

**CHECK sy-subrc = 0. "Prem
IF sy-subrc = 0.     "Zmastercontrol table
```

CHECK statement has been replaced with IF statement to proceed further.

```
ENDENHANCEMENT.  
*$$$-End: (2)-----  
  
* vbak-zzfield = xxxx-zzfield2.  
  
*****  
*$$$-Start: (3)-----  
ENHANCEMENT 1 ZEI_FISCALREP_PARTNER. "active version
```

Earlier, the CHECK statement of the above screen shot has stopped the program control and it came out from the sub-routine without processing the next enhancement of the same user exit. So, it has been replaced with IF.....ENDIF.

➤ Putting much logic in Smart forms can be avoided:-

Smart form has the very good feature of writing the logic / ABAP code directly rather than in Driver program. But, this feature will become a disadvantage when it comes for rework / support team. Because it is bit difficult to understand the process flow logic of the whole smart form and change the logic (even if it is a very small correction). So, it is always good to write the whole logic or as much as possible logic in driver program itself and pass the final result to Smart form.

**SAP Form Builder: Display Program Lines DISCOUNT\_DETAILS**

The screenshot displays the SAP Form Builder interface for the program 'DISCOUNT\_DETAILS'. The left pane shows a hierarchical tree of form elements, with 'DISCOUNT\_DETAILS' selected under 'LINE\_ITEM\_DETAILS'. The right pane shows the 'Program Lines' editor with the following ABAP code:

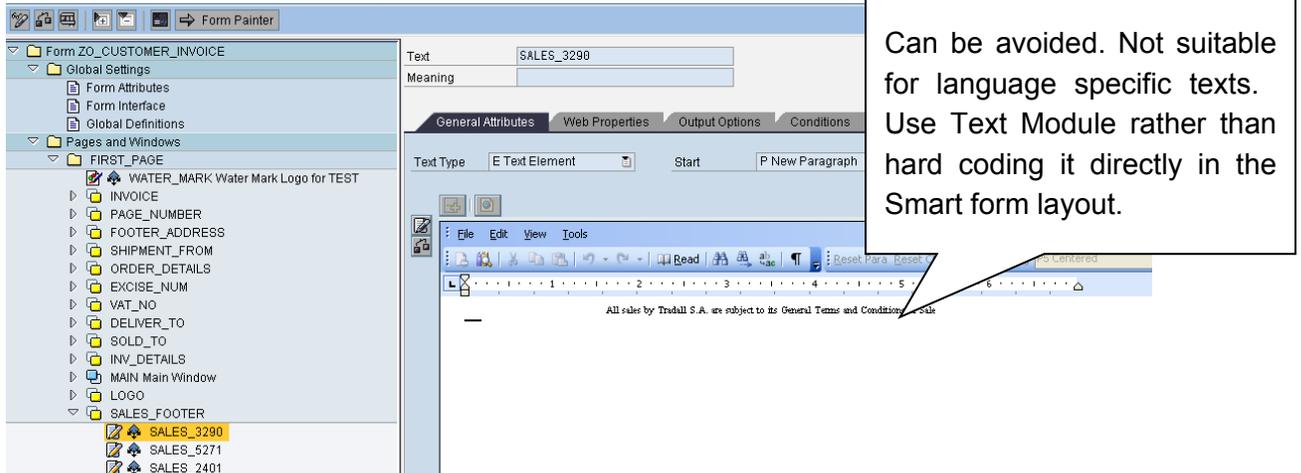
```
1 data: ws_cu_unitp type konv-kwer  
2 WS_C_DISC1 = WA_ITEM-DISC.  
3  
4 SHIFT WS_C_DISC1 LEFT DELETING LEADING ' '.  
5  
6  
7 ws_cu_unitp = WS_C_DISC1.  
8 WS_P_UNITP = ws_cu_unitp.  
9 ws_p_unitp = ws_p_unitp * -1.  
10 write ws_p_unitp to WS_C_DISC1.  
11 SHIFT WS_C_DISC1 LEFT DELETING LEADING ' '.  
12 CONCATENATE '-' WS_C_DISC1 INTO WS_C_DISC1.  
13  
14  
15  
16  
17  
18  
19  
20
```

A callout box on the right contains the text: "This can be avoided. Write the logic in Driver program itself for better readability and flexibility to change / rework whenever required."

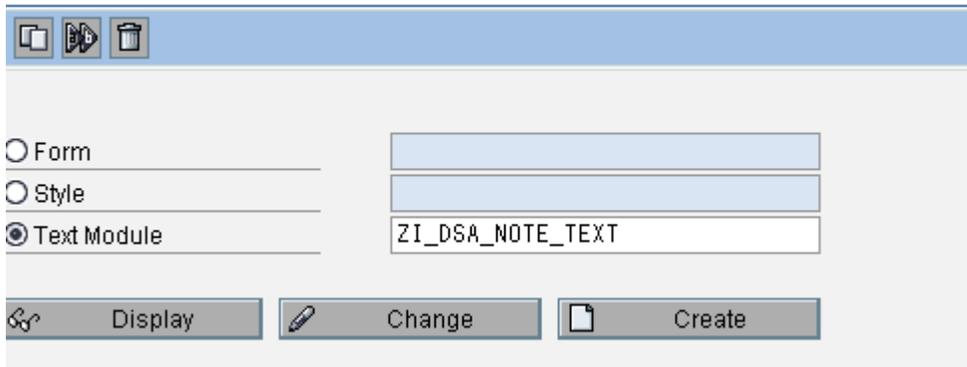
➤ Avoid hard coding the text in smart forms / ABAP programs :-

Hard coding the specified text directly in the program or Smart forms will lead to language specific issue. For example, the header / footer text may change if the logon language is Germany or French. See the below screen shot for better understanding.

SAP Form Builder: Display Text SALES\_3290



## SAP Smart Forms: Initial Screen



Text module can be uniquely assigned to a language using the logon language. When including a text module, we can also access translations of the text module, for example, to include an English text module in a German form.

- Danger of using "USER\_SETTING = 'X'" while calling the Smart forms function module from the driver program :-

Using this command is fine in foreground processing. But this will put a DUMP in Background processing. We can not expect a default local printer for all the user ids. This can be avoided by writing an extra logic as below,

```
FORM f0000_entry USING w_retcode LIKE sy-subrc
                    w_screen TYPE c .

ws_screen = w_screen.
```

```
IF ws_screen IS INITIAL.
  ws_usettings = space.
  wa_composer_param-tddest      = nast-ldest.
  wa_control_param-device      = 'PRINTER'.
  wa_composer_param-tdprinter  = 'SWIN'.
  wa_composer_param-tdimmed    = nast-dimme.
  wa_composer_param-tddelete   = nast-delet
```

```
ELSE.
  ws_usettings = 'X'.
ENDIF.
```

```
IF nast-nacha = '5'.
  wa_control_param-getotf = 'X'.
  wa_control_param-no_dialog = 'X'.
  wa_composer_param-tdimmed = space.
ENDIF.
```

```
CALL FUNCTION ws_fm_name
  EXPORTING
    archive_index      = toa_dara
    archive_parameters = arc_params
    control_parameters = wa_control_param
    mail_recipient     = wa_recipient
    mail_sender        = wa_sender
    output_options     = wa_composer_param
    user_settings      = ws_usettings
```

Extra logic for background processing. In this case, program will take the network printer which has been configured through condition table.

For Foreground processing. Program will take the default printer as per the user setting (SU01 ).

➤ Avoid Hard coding the Language key in the WHERE conditions :-

Generally, developers (especially junior developers) used to hard code the language key with 'E' (English). This will give a wrong text if the user has logged in with a different language like German or French.

```
SELECT land1
       landx
FROM t005t
INTO CORRESPONDING FIELDS OF TABLE t005t
WHERE spras = sy-langu.
```

Good practice: Log on language key has been taken.

```
SELECT spras
       land1
       landx
FROM t005t
INTO CORRESPONDING FIELDS OF TABLE t005t
WHERE spras = c_e-AND
       land1 = wa_adrc-country.
```

Bad practice: Language has been hard coded through constant (English).

➤ Be aware of Domain Conversion Rule :-

The domain Conversion rule of a particular field will affect the hardcoded / passed value in WHERE condition. This will not give you any record. For example, PALEDGER = '10' will be converted to '02'. Be aware of that. Better to declare a variable / constant with the same table/field type and pass the value, rather than direct hard coding (check)

Check the domain for a particular field as below.

Domain: LEDBO Active  
Short Description: Ledger for profitability segments (CO-PA)

Format  
Data Type: CHAR Character String  
No. Characters: 2  
Decimal Places: 0

Output Characteristics  
Output Length: 2  
Convers. Routine: LEDBO

“Conversion routine” for the field “PALEDGER”

### Data Browser: Table CE1BC01: Selection Screen

Number of Entries: 4 E

Restrictions

CTy	Name
10	Company code currency
12	Comp. Code crcy, PrCtr valuation
B0	Operating concern currency
B2	OpConcern crcy, PrCtr valuation

Search help shows as “10”. But we need to pass it as “02” in our custom program. Then only it can fetch the required data.

```
SELECT vbund rec_waers ww002 ww001 vtweg artnr kindnr
werks vv001 vv005 vv220 vv101 vv106 vv102 vv107
vv104 vv105 vv109 vv105 vv110 vv111 vv112 budat
INTO CORRESPONDING FIELDS OF TABLE it_ce1bc01
WHERE paledger = '02' AND vrgar = 'X' AND bukrs = p_bukrs AND
gjahr = p_gjahr AND perde >= p_perst AND perde <= p_perend.
```

➤ SAP Standard Internal Tables has to be handled carefully in user exits

We have to update the SAP standard internal tables very carefully in user exits, if there is a requirement to change / add data. For example, we had a requirement to update the Partner Functions of the Sales order through SD module User exit (USEREXIT\_MOVE\_FIELD\_TO\_VBAK). We faced several “Update Termination Errors” while saving the Sales order because of this user exit. We analyzed the user exit code with standard functionality and updated the (SAP standard) internal tables XVBPA and YVBPA with “Update” flag (UPDKZ) as below. This has resolved the issue of getting several dumps in SM13 / ST22.

```
Loop at xvbpa where updkz = 'I'. "Only for inserted records through user exit.
  read table i_cvbpa into wa_cvbpa with key vbeln = xvbpa-vbeln
  posnr = xvbpa-posnr parvw = xvbpa-parvw.
  if sy-subrc = 0. "already found in original table vbpa
    xvbpa-updkz = 'U'. "So, we should not insert, but can be updated
    modify xvbpa.
  endif.
Endloop.

"Check the old entries in VBPA, compare with YVBPA and update YVBPA (Required for updating the existing record)
Loop at i_cvbpa into wa_cvbpa. "Compare the newly re-determined VBPA with old VBPA entries
  read table xvbpa with key vbeln = wa_cvbpa-vbeln
  posnr = wa_cvbpa-posnr parvw = wa_cvbpa-parvw.
  if sy-subrc <> 0. "Not found xvbpa - removed through determination, then should be in yvbpa
    read table yvbpa with key vbeln = wa_cvbpa-vbeln
    posnr = wa_cvbpa-posnr parvw = wa_cvbpa-parvw.
    if sy-subrc <> 0.
      MOVE-CORRESPONDING wa_cvbpa to yvbpa.
      yvbpa-updkz = 'E'. "We should insert and mark as Delete to update the original VBPA entry
      append yvbpa.
    endif.
  endif.
```

Update Flag UPDKZ has to be updated properly for both XVBPA & YVBPA.

## Conclusions

It is always recommended to include all the Lessons Learnt / Best Practices in the “Code Review Check list” to avoid such mistakes in future deliverables. These points are applicable to both Implementation and Support Projects.

Thank You,  
Prammendran.