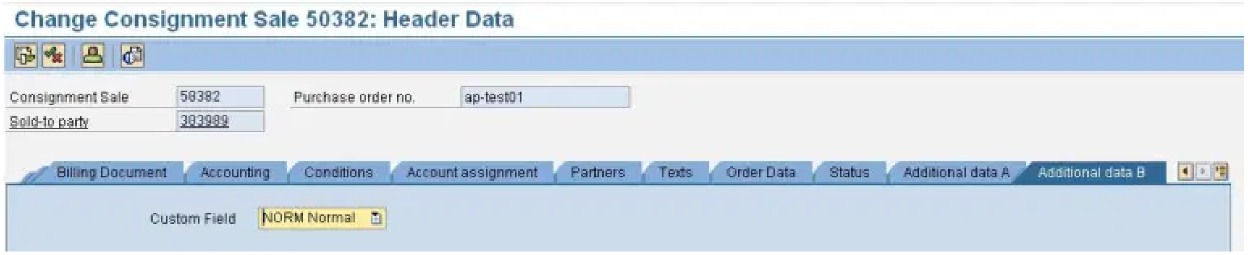
ABAP Journal

Adding Custom Field into Sales Document Maintenance Screen

i
12 Votes

In term of sales document, SAP provides means to add our own data into sales document and so user is able to display and maintain through standard maintenance transaction (e.g. VA01, VA02, and VA03). The fields normally placed under tab 'Additional Data B' at header and item level. Following is one of the ways adding custom fields into the screen.



Change Consignment Sale 50382: Header Data

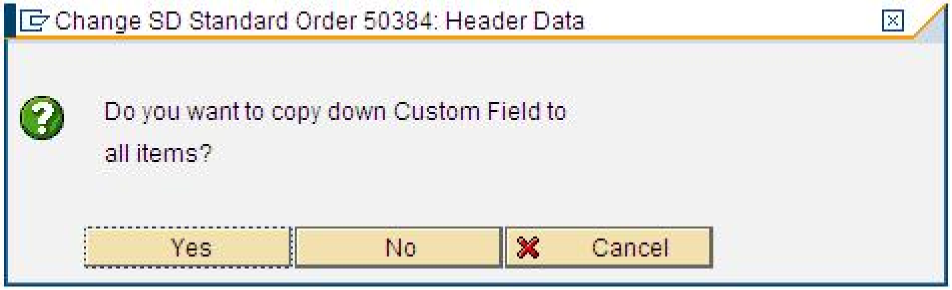
Consignment Sale: 50382
Purchase order no.: ap-test01
Sold-to party: 303989

Custom Field: NORM Normal

(https://abapjournal.files.wordpress.com/2011/09/so-with-custom-field.jpg)

For example there is requirement to add a custom field into header and item level of a standard order, i.e. document type ‘ZOR’. Let’s say the field name is Custom Field and it should be displayed as a list box, with possible values NORM (Normal) and SPEC (Special) and it should default to NORM. At header level, the value is used to ease the user to set all the items and always copy down to the line items. The user, however, is still able to change the value individually in each item.

Later on when user change the field value at header level let’s say from NORM to SPEC, it should copy down the new value to all the line items. The user should be notified before replacing value in each item.



Change SD Standard Order 50384: Header Data

Do you want to copy down Custom Field to all items?

(https://abapjournal.files.wordpress.com/2011/09/popup_to_confirm.jpg)

Append the custom field to database table

We need to prepare containers to store the custom data into the database. We can create new custom transparent tables containing all the custom fields along with key fields to link with sales document tables or the easiest way we can append the field to the sales document tables, i.e. in this example we will use VBAK and VBAP tables. The later approach is easier and simpler because we don’t need to develop our own logic to retrieve and save our custom data from/to database.

We can use Append Structure functionality through transaction SE11 to add our field into the tables. Because our custom field will appear as a list-box, we can create a new domain where the possible values are set and a new data element that will make use of the new domain. The custom field should then make use of the new data element we created as its data type.

In this example we will name the new field as ZZCUSTOM1 so that we will have VBAK-ZZCUSTOM1 and VBAP-ZZCUSTOM1.

Paint the custom fields

Open transaction SE80, program SAPMV45A, screen 8309 (header level screen) and screen 8459 (item level screen).

Go to the screen layout and appends our field there and paint the layout.

If the custom field is slotted in VBAK or VBAP, It is better to name the dynpro-field the same as the field name in database, for example ‘VBAK-ZZCUSTOM1’, because then SAP will be able to determine the corresponding fields of database table VBAK or VBAP where the value should be referring to.

In this example I like to assign function code ‘CS’ and function type ‘S’ so I can add handling logic every time user change the value where in this case the new value of header field will automatically copy down to each item field.

Into each header and item screen in this example, we will create a text named ‘TEXT_ZZCUSTOM1’ and an input field named ‘VBAK-ZZCUSTOM1’ and ‘VBAP-ZZCUSTOM1’ respectively.

Set field behavior

Under the PBO we can adjust the behavior of the field. Let’s say it will appear only for certain document type or it will appear as output only in display transaction, e.g. VA03 and as input field in create or change transaction, e.g. VA01 and VA02. Here we can make use of global variable T180-TRTYP to indicate whether the screen is called for displaying only, changing an existing document, or document creation. T180-TRTYP in this case will contain value of ‘A’, ‘V’, or ‘H’ respectively.

If let’s say we have same custom field at header and item levels, we can name these fields the same like in this example is ‘ZZCUSTOM1’ so that the transaction will automatically copy down the value from header to item level every time new item is created. This action is performed by subroutine VBAP_FUELLEN_VBAK in the standard program, i.e. by following line of code.

```
MOVE-CORRESPONDING VBAK TO VBAP.
```

All custom PBO modules can be placed in include MV45AOZZ.

In this example both header and item screen will share same PBO module. Please note SAP recommends modules name begin with ‘ZZ’.

```
PROCESS BEFORE OUTPUT.
 *                             Verarbeitung vor der Ausgabe

 MODULE ZZPREPARE_CUSTOMFIELD.
```

Where module ZZPREPARE_CUSTOMFIELD implementation will be added into include MV45AOZZ as follow.

```
*&---------------------------------------------------------------------*
 *&      Module  ZZPREPARE_CUSTOMFIELD  OUTPUT
 *&---------------------------------------------------------------------*
 *       Adjust custom field attributes
 *----------------------------------------------------------------------*
 MODULE zzprepare_customfield OUTPUT.
    IF  vbak-auart = 'ZOR'.
 *     Adjust custom field attribute: appear as output-only or input field
       LOOP AT SCREEN.
         IF    screen-name EQ 'VBAP-ZZCUSTOM1'
           OR  screen-name EQ 'VBAK-ZZCUSTOM1'.
           IF t180-trtyp = 'H' OR    "Create
              t180-trtyp = 'V'.      "Change
             screen-input = 1.
           ELSEIF t180-trtyp = 'A'.  "Display
             screen-input = 0.
           ENDIF.
           MODIFY SCREEN.
        ENDIF.
       ENDLOOP.
    ELSE.
*     Hide the custom fields in other document types
       LOOP AT SCREEN.
         IF  screen-name EQ 'TEXT_ZZCUSTOM1' OR
             screen-name EQ 'VBAP-ZZCUSTOM1' OR
             screen-name EQ 'VBAK-ZZCUSTOM1'.
           screen-invisible = 1.
           screen-input = 0.
           MODIFY SCREEN.
         ENDIF.
       ENDLOOP.
     ENDIF.
 ENDMODULE.                 " ZZPREPARE_CUSTOMFIELD  OUTPUT
```

Validate value and handle when value changed

Under the PAI we can put field validation logic and throw error when value is wrong. In case of list box fields or check boxes where function code is assigned to the fields, we can put handling logic here.

All custom PAI modules can be placed in include MV45AIZZ.

In this example we do not need additional PAI modules at item screen however we need on-change-value handler at header screen.

```
PROCESS AFTER INPUT.
 *                             Verarbeitung nach der Eingabe
 FIELD VBAK-ZZCUSTOM1
   MODULE ZZONCHGOF_ZZCUSTOM1 AT CURSOR-SELECTION.
```

The logic will be implemented in include MV45AIZZ as follow:

```
*&--------------------------------------------------------------------*
 *&      Module  ZZONCHGOF_CUSTOM1  INPUT
 *&--------------------------------------------------------------------*
 *       Handle on change of field ZZCUSTOM1
 *---------------------------------------------------------------------*
 MODULE ZZONCHGOF_CUSTOM1 INPUT.


* Is there any item having different value with header’s?
   LOOP AT XVBAP TRANSPORTING NO FIELDS
     WHERE ZZCUSTOM1 NE VBAK-ZZCUSTOM1.
     EXIT.
   ENDLOOP.

 * If yes...
   IF SY-SUBRC = 0.
     PERFORM ZZITEM_CHANGE_ZZCUSTOM1.
```

```
  ENDIF.
ENDMODULE.

*&---------------------------------------------------------------------*
*&      Form  ZZITEM_CHANGE_ZZCUSTOM1
*&---------------------------------------------------------------------*
*       Change item's custom field to the new header's value
*----------------------------------------------------------------------*
form ZZITEM_CHANGE_ZZCUSTOM1 .
  DATA lv_answer TYPE c LENGTH 1.
  FIELD-SYMBOLS <vbap> LIKE LINE OF xvbap.

  CALL FUNCTION 'POPUP_TO_CONFIRM'
    EXPORTING
      text_question               = 'Do you want to copy down Custom Field to all items?'(z01)
    IMPORTING
      ANSWER                      = lv_answer
    EXCEPTIONS
      TEXT_NOT_FOUND              = 1
      OTHERS                      = 2.
  IF sy-subrc <> 0.
    MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
            WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  ENDIF.
  CASE lv_answer.
    WHEN '1'. "Yes, copy down to all itmes.
      LOOP AT xvbap ASSIGNING <vbap> WHERE zzcustom1 <> vbak-zzcustom1.
        <vbap>-zzcustom1 = vbak-zzcustom1.
        IF <vbap>-updkz IS INITIAL.
          <vbap>-updkz = updkz_update.
        ENDIF.
      ENDLOOP.
    WHEN 'A'. "Cancel
      VBAK-ZZCUSTOM1 = *VBAK-ZZCUSTOM1. "Back to old value
    WHEN OTHERS.
  ENDCASE.
endform.                    " ZZITEM_CHANGE_ZZCUSTOM1
```

Please note that we need to set field XVBAP-UPDKZ to ‘U’ to notify the main program that the item has been changed and so that it requires update to database.

Field *VBAK at this point contains the old value before any changes made on the screen.

Populate value from database or set default value

In case of transaction maintaining or displaying existing document we can use user exit USEREXIT_READ_DOCUMENT slotted in include MV45AFZZ to populate custom fields from database for both header and item level. At this point all standard fields have been populated, header data is visible through structure VBAK and item data is visible through internal table XVBAP.

Please note that this is not necessary if the custom fields are appended directly in table VBAK and VBAP table because SAP will automatically populate the values from the tables.

In case of document creation transaction we can set default value of our custom fields at user exit USEREXIT_MOVE_FIELD_TO_VBAK for header level data or USEREXIT_MOVE_FIELD_TO_VBAP for item level data. We can find this user exits in include MV45AFZZ too.

At this point we can check value of T180-TRTYP too and for header level:

SVBAK-TABIX = 0: Create header

SVBAK-TABIX > 0: Change header

Or at item level:

SVBAP-TABIX = 0: Create item

SVBAP-TABIX > 0: Change item

(at this point I notice SVBAP-TABIX value is always 0 or in other words this user exit is triggered only for newly created items)

In this example we just need to set default header value in user exit USEREXIT_MOVE_FIELD_TO_VBAK while the item value will automatically copy from header by standard subroutine VBAP_FUELLEN_VBAK.

```
*---------------------------------------------------------------------*
 *       FORM USEREXIT_MOVE_FIELD_TO_VBAK                              *
 *---------------------------------------------------------------------*
 *       This userexit can be used to move some fields into the sales  *
 *       dokument header workaerea VBAK.                               *
 *                                                                     *
 *       SVBAK-TABIX = 0:  Create header                               *
 *       SVBAK-TABIX > 0:  Change header                               *
 *                                                                     *
 *       This form is called at the end of form VBAK_FUELLEN.          *
 *                                                                     *
 *---------------------------------------------------------------------*
 FORM userexit_move_field_to_vbak.

 * Set default to custom field
   IF vbak-auart = 'ZOR'.
    IF vbak-zzcustom1 IS INITIAL AND svbak-tabix = 0.
       vbak-zzcustom1 = 'NORM'.
     ENDIF.
   ENDIF.

 ENDFORM.                    "USEREXIT_MOVE_FIELD_TO_VBAK
```

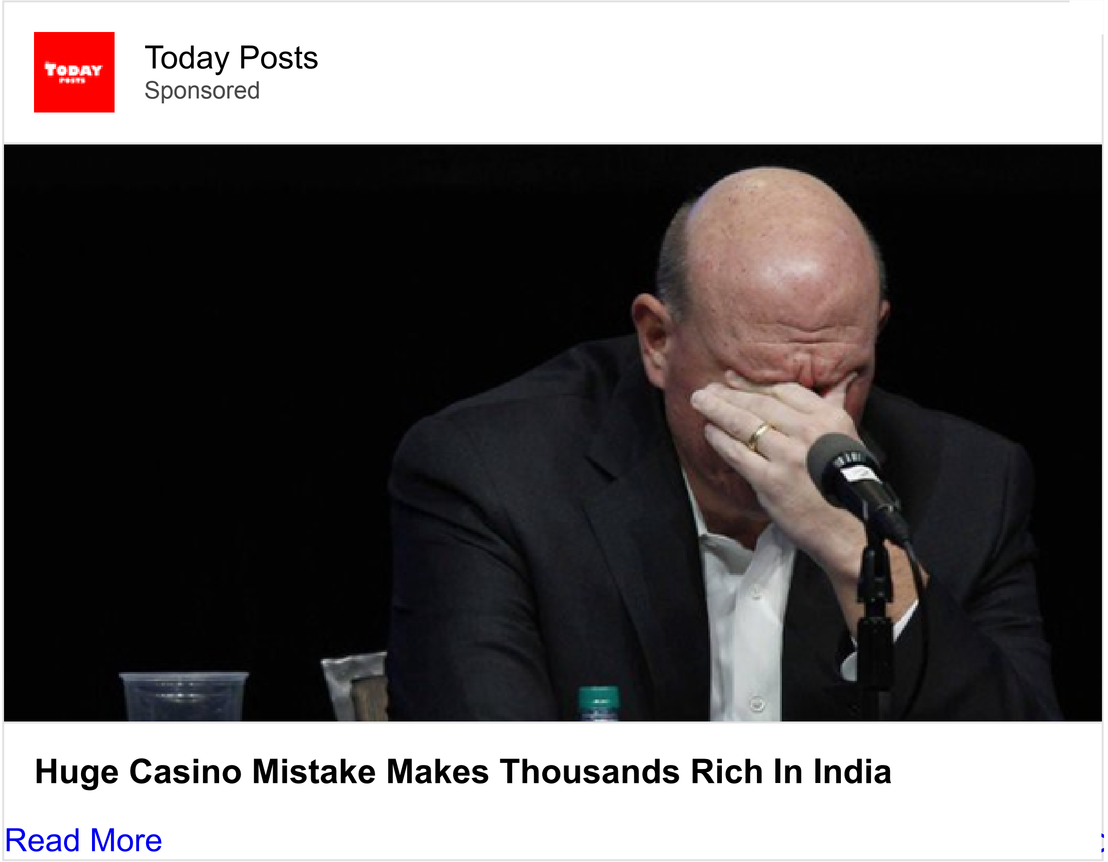
Save to database

In the event custom field is appended directly in table VBAK and VBAP table no additional logic is required as SAP will automatically populate the custom values to database.

We can put saving logic under user exit USEREXIT_SAVE_DOCUMENT in include MV45AFZZ.

At this point VBAK and XVBAP is visible and VBAK-VBELN contains new document number in case of document creation.

For this example we do not need additional logic here.



Powered by wordads.co

Seen ad many times

Not relevant

Offensive

additional data B , contract , custom field , quotation , sales order , SD , User exit

This entry was posted on September 25, 2011, 4:09 am and is filed under Code Template. You can follow any responses to this entry through RSS 2.0. You can leave a response, or trackback from your own site.

COMMENTS (3)   TRACKBACKS (1)

#1 by bangkok seo on May 3, 2013 - 10:57 am

Following all, what acceptable is accepting the absolute artefact or autograph the a good deal of acute commodity if no one sees it?
Web optimization optimise is the very best and a great deal of price-effective way to certain overall body an on the net attendance for your
business.

#2 by anilmakhidi7 on August 6, 2014 - 4:58 pm

Was looking for exactly that. Thank you! I couldn't even find anything on http://scn.sap.com !

#3 by **karl_rockwell@yahoo.com** on September 30, 2014 - 1:30 am

Modifying screens and registering the objects is no longer advisable here. Please retire this document and refer to http://scn.sap.com/community/abap/switching-enhancing-and-adapting-standard-programs/blog/2013/12/12/adding-extra-tabs-in-standard-transactions-for-sales-and-contracts-without-access-key

1. Adding Custom Field into Sales Document Maintenance Screen