



External CardEntry in SAP ECC

Implementation Manual

Revision 1.5.1

This document contains information proprietary to, and is the property of, Delego Software Inc. ("Delego"). The recipient of this document, by its retention and use, agrees to hold this document and its contents in confidence. This document shall not be transferred or communicated to any third party, without the prior written consent of Delego, in whole or part, by any means. This document or any other applicable documents provided by Delego shall be returned to Delego upon request. This Proprietary Information Notice is an integral part of this document and shall not be removed or altered.

©2014 Delego Software Inc.

931 COMMISSIONERS ROAD EAST SUITE 200 LONDON, ONTARIO, CANADA N5Z 3H9
P: 1-855-407-1198 | **E:** INFO@DELEGO.CA | **W:** WWW.DELEGOSOFTWARE.COM



Table of Contents

1	General	3
1.1	Preliminary	3
1.2	About This Manual	3
1.3	How External Credit Card Entry Works	3
2	Implementation	6
2.1	Transports	6
2.2	Fundamental Program Elements	6
2.3	Screen Independent Elements	8
2.4	Sales Order Overview Screens	23
2.5	Payment Card Tab	32
2.6	Customer Master	36
	Notices	40

“We believe transactions should be simple, technology should be seamless and security should not be stressful.”



1 General

1.1 Preliminary

Some merchants do not wish to enter credit card data into the SAP ECC systems. This manual describes the changes needed to implement a separate button on the sales order entry screen that will call a web page for the entry of the credit card number.

1.2 About This Manual

The information in this manual is intended to provide the necessary information to implement the ability for users to enter credit card details on a separate web page and have this information integrate with the SAP ECC system. This function is only available for order entry in ECC 6.0. Earlier SAP ERP versions are not supported nor is the CRM system.

Delego Software, Inc. assumes that the user of this manual is adept with configuring the ECC product. The changes necessary to implement this functionality are, in part, delivered with transports that are imported to the system. Other parts must be added manually as described.

1.3 How External Credit Card Entry Works

Delego Software has developed External Credit Card Entry to allow credit card data to be securely entered on a data entry form that is external and separate from SAP.

External Card Entry is a lightweight web application. The web application is accessed via a web browser. Installation and management of the Card Entry web application is described in the guide *Delego Card Entry*.

The web page called by this application allows the user to enter the following details card details:

- a) Credit card number
- b) Expiry Date
- c) Cardholder Name
- d) CVV value
- e) Card Type
- f) Billing Address

The web page can also be configured to integrate with keyboard wedges to be used for card swipe.

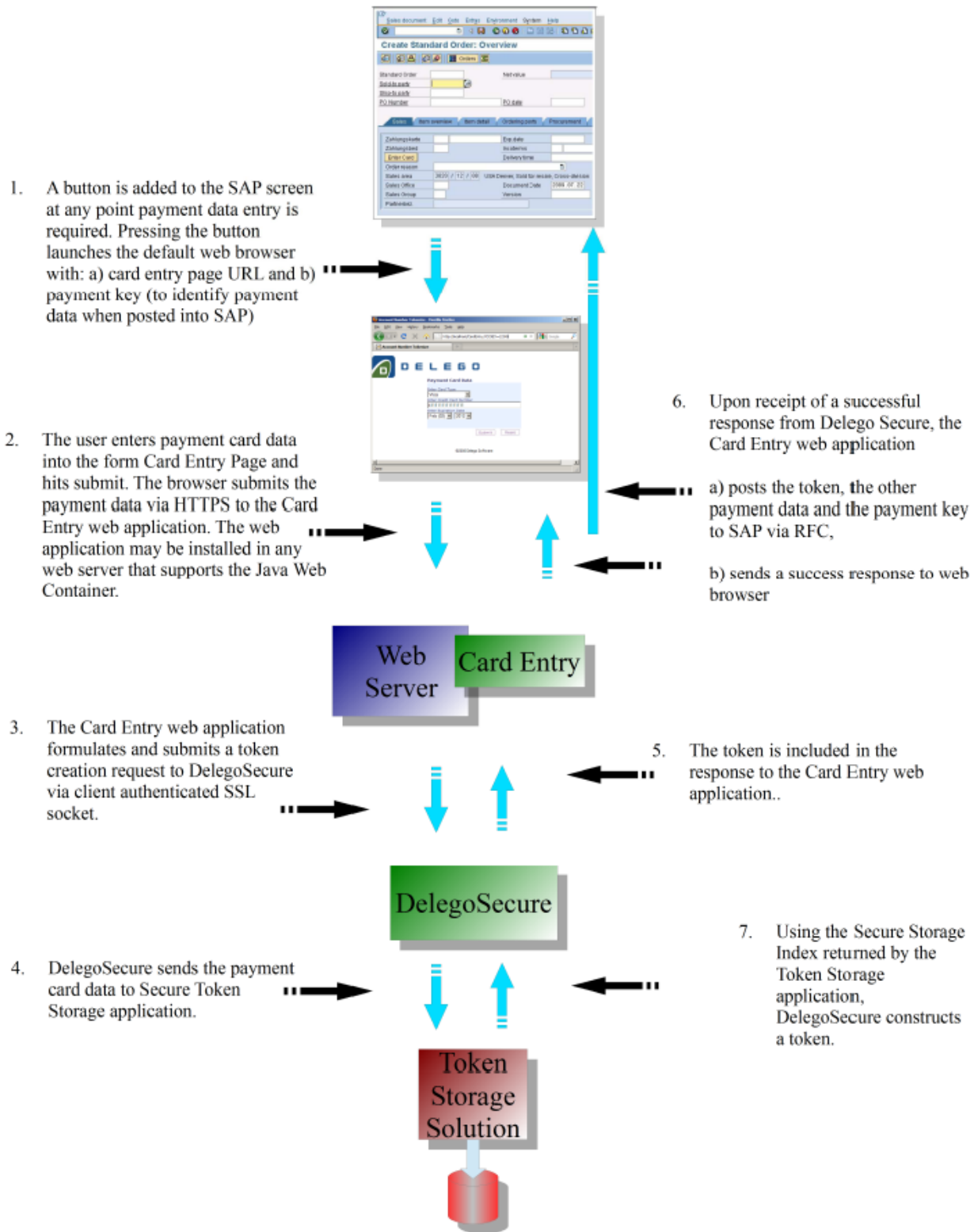
The credit card primary account number is secured via DelegoSecure as either an encrypted string or a token. For more information about DelegoSecure, see the guide *DelegoSecure*.



The secured card number along with the other card data is posted to SAP and is then used to populate the payment data entry fields on the SAP screen.

External Card Entry, along with other Delego components like External Card Validation, provides the means to effectively and efficiently remove sensitive card holder data from SAP.

The annotated diagram on the next page presents a step by step description of the External Card Entry process from SAP screen to DelegoSecure and back to SAP screen.





2 Implementation

There are two approaches to implementing external card entry in your SAP system. You can use the transports provided by Delego Software to directly import the various structure extensions, function modules and screen changes. If you have made local changes to any of the effected screens or would like more control over the changes follow the instructions described in this manual.

2.1 Transports

Delego Software has created transports to assist in implementing external card entry in SAP ECC systems. The transports deliver all of the code outlined in this manual except the changes in the sections with the warning: entitled:

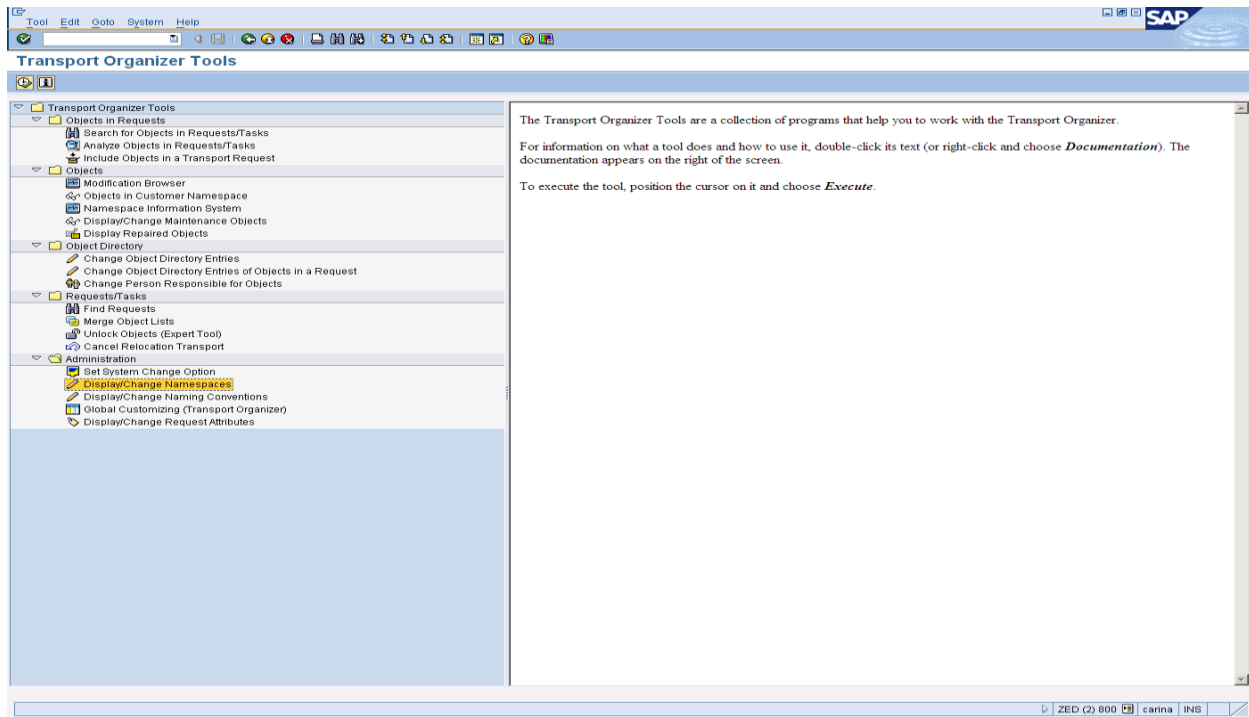
Manual Operation – Not Delivered with Transports

See the Delego *Transports* guide for more information on importing transports from Delego.

2.2 Fundamental Program Elements

2.2.1 /DELEGO/ Namespace

If the /DELEGO/ namespace has not already been created, use transaction SE03 to create the namespace /DELEGO/.



2.2.2 Package /DELEGO/EXTERNAL



Use the Package Builder to create the package /DELEGO/EXTERNAL.

Package	/DELEGO/EXTERNAL
Short Description	External Data Entry
Appl. component	ca
Software Component	home
Transport Layer	ZR47
Package Type	Not a Main Package

2.2.3 Messages

Add two messages (49 and 50) to the message class /DELEGO/DELEGO.

For message 50, enter the short text "Error entering credit card details on website. No card has been entered."

For message 49, enter the short text "Waiting for credit entry on website. Hit enter when done." And enter further help text, remove the Self-explanatory mark and enter these details.

&CAUSE&

After the card entry button is pushed, a website is called. The system waits until the credit card number has been entered on the website screen.

&SYSTEM_RESPONSE&

Waits until further entry.

&WHAT_TO_DO&

Hit the enter key when the credit card details have been saved.

&SYS_ADMIN&

2.2.4 Function Group /DELEGO/EXTERNAL

Using SE80 define a new function group /DELEGO/EXTERNAL as a member of the package /DELEGO/EXTERNAL.

Function group	/delego/external
Short text	Integrate external data entry
Person Responsible	DELEGO



2.3 Screen Independent Elements

2.3.1 Data Table /DELEGO/CCEXT

Using transaction SE11, create a user table /DELEGO/CCEXT. Assign it to the package /DELEGO/EXTERNAL. The table should have these settings

Short Text: Delego: External Credit Cards

Delivery Class: L

Data Browser/Table View Maint.: Display/Maintenance Allowed

The structure of the table is:

Field	Key	Initial Values	Data Element
MANDT	X	X	MANDT
CCKEY	X	X	CHAR20
CCINS			CCINS
CCNUM			CCNUM
CCNAME			CCNAME
CCFOL			CCFOL
CVVAL			CVVAL
CVVST			CVVST
DATBI			DATBI
DATAB			DATAB
TRACK1			CHAR80
TRACK2			CHAR80
.INCLUDE			CCADDRESS
CDATE			CRDATE
CTIME			CRTIME



2.3.2 Data Table /DELEGO/CE_URL

Using transaction SE11, create a user table /DELEGO/CE_URL. Assign it to the package /DELEGO/EXTERNAL. The table should have these settings

Short Text: DelegoCardEntry: URL master data table

Delivery Class: A

Data Browser/Table View Maint.: Display/Maintenance Allowed

The structure of the table is:

Field	Key	Initial Values	Data Element
MANDT	X	X	MANDT
SYSID	X	X	SYSID
URL1			URL
URL2			URL
URL3			URL

2.3.3 Populate Table /DELEGO/CE_URL

Manual Operation – Not Delivered with Transports

Using transaction SM31, populate the table for each of the instances of SAP that will use DelegoCardEntry (aka. DeleGoCE). The System id, SYSID, is the instance system id. Place in the field URL1 the URL of the DeleGoCE system. If the URL1 field has not sufficient space, then continue the URL in field URL2 and URL3. The fields URL1, URL2 and URL3 are concatenated when the final URL is built.

2.3.4 Function Module /DELEGO/CC_GET_EXT

Using transaction SE80, create a function module called /DELEGO/CC_GET_EXT in the function group /DELEGO/EXTERNAL within the package /DELEGO/EXTERNAL

Imports

None

Exports

Field	Type Spec.	Associated Type	Pass Value
X_CCEXT	TYPE	/DELEGO/CCEXT	X

**Changing**

None

Tables

None

Exceptions

NUMBER_ERROR Number Error

ERROR_REGISTRATION Registration Error

Source Code

```

FUNCTION /DELEGO/CC_GET_CCEXT.
*-----
***"Local Interface:
* EXPORTING
*   VALUE(X_CCEXT) TYPE /DELEGO/CCEXT
* EXCEPTIONS
*   NUMBER_ERROR
*   ERROR_REGISTRATION
*-----

constants: gc_refid_object      TYPE nrobj          VALUE '/DELEGO/CC'.
data: x_ckey type /delego/ccext-ckey,
      ls_o2pagename type o2pagename,
      ce_url        TYPE /delego/ce_url,
      lv_url_bsp     TYPE string,
      lv_url_bsp_c   TYPE text2048,
      xutc           TYPE ttzdata-timestamp.

ENHANCEMENT-SECTION      /DELEGO/ES_EXTERNAL_ENTRY_DATA SPOTS /DELEGO/EXTERNAL STATIC .

      data: gc_url type text2048 value 'https://carina.delego.ca:8080/CardEntryZED/'.
      data: gc_skey type char64 value
'OAKCDHVIAICUFUEWO77629756KDNVH8UEHGJAUGTGI8ERI63YRHFJU326IJYTEDT'.

END-ENHANCEMENT-SECTION.

ENHANCEMENT-SECTION      /DELEGO/ES_EXTERNAL_ENTRY SPOTS /DELEGO/EXTERNAL.

* create an implementation of this enhancement to change the value of the URL as
* needed.
      if sy-sysid = 'ZED'.
          gc_url = 'https://carina.delego.ca:8080/CardEntryZED/'.
      elseif sy-sysid = 'ZEQ'.
          gc_url = 'http://hercules.delego.ca/CardEntry/'.
      *
          gc_url = 'http://10.1.1.101:8080/CardEntryZEQ/'.
      elseif sy-sysid = 'ZET'.
          gc_url = 'http://draco.delego.ca/CardEntry/'.
      endif.

END-ENHANCEMENT-SECTION.
*$$$-Start: /DELEGO/ES_EXTERNAL_ENTRY-----
---$$$*
ENHANCEMENT 2 /DELEGO/IM_CHANGE_ADDRESS_DATA.      "active version

* create an implementation of this enhancement to change the value of the URL as

```



```
* needed.
  if sy-sysid = 'ZED'.
*   gc_url = 'https://carina.delego.ca:8443/CardEntryZED/'.
   gc_url = 'http://carina.delego.ca:8080/CardEntryZED/'.
  elseif sy-sysid = 'ZEQ'.
   gc_url = 'http://10.1.1.101/CardEntryZEQ/'.
  elseif sy-sysid = 'ZET'.
   gc_url = 'http://draco.delego.ca/CardEntry/'.
  endif.

ENDENHANCEMENT.
*$$-End:   /DELEGO/ES_EXTERNAL_ENTRY-----
---$*$*

* fetch the reference number used to communicate with the web page
CALL FUNCTION 'NUMBER_GET_NEXT'
  EXPORTING
    nr_range_nr          = '01'
    object               = gc_refid_object
    quantity             = '1'
  IMPORTING
    number               = x_ckey
  EXCEPTIONS
    interval_not_found   = 1
    number_range_not_intern = 2
    object_not_found     = 3
    quantity_is_0        = 4
    quantity_is_not_1    = 5
    interval_overflow    = 6
    buffer_overflow      = 7
    OTHERS               = 8.

IF sy-subrc NE 0.
  RAISE number_error.
ENDIF.

* Get a UTC timestamp
CALL FUNCTION 'TZ_UTC_SYSTEMCLOCK'
  IMPORTING
*   UTC_DATE          =
*   UTC_TIME          =
  UTC_TIMESTAMP      = xutc.

* Create the hash of the secret key and the utc timestamp
data: keystack type string,
      xskey type hash160.
concatenate
  gc_skey
  xutc
into keystack.

CALL FUNCTION 'CALCULATE_HASH_FOR_CHAR'
  EXPORTING
*   ALG              = 'SHA1'
   DATA            = keystack
*   LENGTH           = 0
  IMPORTING
   HASH             = xskey
*   HASHLEN          =
*   HASHX            =
```



```
*      HASHXLEN          =
      EXCEPTIONS
      UNKNOWN_ALG        = 1
      PARAM_ERROR        = 2
      INTERNAL_ERROR      = 3
      OTHERS              = 4.

      IF SY-SUBRC <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*       WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
      ENDIF.

      clear lv_url_bsp_c.

* use the database entry before the coded entry
      select single * from /delego/ce_url into ce_url where sysid = sy-sysid.
      if sy-subrc = 0.
        concatenate ce_url-url1
                     ce_url-url2
                     ce_url-url3
                     into lv_url_bsp_c.
      else.
        MOVE gc_url TO lv_url_bsp_c.
      endif.

      CONCATENATE
        lv_url_bsp_c
        '?MANDT='
        sy-mandt
        '&CCKEY='
        x_ckey
        '&sec1='
        xutc
        '&sec2='
        xskey
        INTO lv_url_bsp_c.

      CALL FUNCTION 'CALL_BROWSER'
      EXPORTING
        URL              = lv_url_bsp_c
        WINDOW_NAME      = 'Delego External Card Entry'
        NEW_WINDOW       = 'X'
*       BROWSER_TYPE    =
*       CONTEXTSTRING   =
      EXCEPTIONS
        FRONTEND_NOT_SUPPORTED = 1
        FRONTEND_ERROR        = 2
        PROG_NOT_FOUND        = 3
        NO_BATCH              = 4
        UNSPECIFIED_ERROR     = 5
        OTHERS                = 6.

      IF sy-subrc NE 0.
        RAISE error_registration.
      ENDIF.

      MESSAGE I049(/DELEGO/MESSAGES).

* retrieve the data from the data tables
```



```

select single * from /delego/ccext into x_ccext where cckey = x_cckey.
if sy-subrc <> 0.
    message w050 (/DELEGO/MESSAGES) .
else.
    delete from /delego/ccext where cckey = x_cckey.
endif.

```

ENDFUNCTION.

2.3.5 Changing the Reference to URL (Superseded by table /DELEGO/CE_URL)

Manual Operation – Not Delivered with Transports

Manually change the URL to match the location of the call to the web page for the data entry. To do so, create an implementation of the enhancement section /DELEGO/CC_EXTERNAL_ENTRY_DATA and change the value of "gc_url" to the appropriate value.

2.3.6 Changing the Security Key

Manual Operation – Not Delivered with Transports

Manually change the value of the security key, gc_skey. To do so, create an implementation of the enhancement spot /DELEGO/ES_EXTERNAL_ENTRY_DATA and change the value of "gc_skey". The value must be an alphanumeric string that is 64 character long. No special characters or blanks are permitted. This key must match the key in the configuration file for DeleGoCardEntry.

2.3.7 Include /DELEGO/CC_EXTERNAL_ENTRY

Using transaction SE80, create an include called /DELEGO/CC_EXTERNAL_ENTRY as part of the package /DELEGO/EXTERNAL. Insert the following code:

```

*&-----*
*& Include /DELEGO/CC_EXTERNAL_ENTRY
*&-----*
*& Call at SAPMV45A via screen sequence control
*&-----*

*&-----*
*& Form /DELEGO/CC_EXTERNAL_GET
*&-----*
* Credit card entry via external web page
*-----*
FORM /DELEGO/CC_EXTERNAL_GET.
tables: /delego/ccext.
constants: gc_refid_object TYPE nrobj VALUE '/DELEGO/CC'.
data: x_ccext type /delego/ccext,
      x_cckey type /delego/ccext-cckey,
      ls_o2pagename type o2pagename,
      lv_url_bsp TYPE string,

```



```
lv_url_bsp_c TYPE text2048.
```

```
ENHANCEMENT-SECTION /DELEGO/ES_EXTERNAL_ENTRY_DATA SPOTS /DELEGO/ES_EXTERNAL_ENTRY STATIC .
```

```
data: gc_url type text2048 value 'http://carina.delego.ca:8080/CardEntry/'.
```

```
END-ENHANCEMENT-SECTION.
```

```
ENHANCEMENT-SECTION /DELEGO/ES_EXTERNAL_ENTRY SPOTS /DELEGO/ES_EXTERNAL_ENTRY.
```

```
* create an implementation of this enhancement to change the value of the URL as  
* needed.
```

```
END-ENHANCEMENT-SECTION.
```

```
* fetch the reference number used to communicate with the web page
```

```
CALL FUNCTION 'NUMBER_GET_NEXT'
```

```
EXPORTING
```

```
nr_range_nr = '01'
```

```
object = gc_refid_object
```

```
quantity = '1'
```

```
IMPORTING
```

```
number = x_ckey
```

```
EXCEPTIONS
```

```
interval_not_found = 1
```

```
number_range_not_intern = 2
```

```
object_not_found = 3
```

```
quantity_is_0 = 4
```

```
quantity_is_not_1 = 5
```

```
interval_overflow = 6
```

```
buffer_overflow = 7
```

```
OTHERS = 8.
```

```
IF sy-subrc NE 0.
```

```
RAISE number_error.
```

```
ENDIF.
```

```
MOVE gc_url TO lv_url_bsp_c.
```

```
CONCATENATE
```

```
lv_url_bsp_c
```

```
'?MANDT='
```

```
sy-mandt
```

```
'&CCKEY='
```

```
x_ckey
```

```
INTO lv_url_bsp_c.
```

```
CALL FUNCTION 'CALL_BROWSER'
```

```
EXPORTING
```

```
URL = lv_url_bsp_c
```

```
WINDOW_NAME = 'Delego External Card Entry'
```

```
NEW_WINDOW = 'X'
```

```
* BROWSER_TYPE =
```

```
* CONTEXTSTRING =
```

```
EXCEPTIONS
```

```
FRONTEND_NOT_SUPPORTED = 1
```

```
FRONTEND_ERROR = 2
```



```
PROG_NOT_FOUND = 3
NO_BATCH = 4
UNSPECIFIED_ERROR = 5
OTHERS = 6.
```

```
IF sy-subrc NE 0.
  RAISE error_registration.
ENDIF.
```

```
MESSAGE I049(/DELEGO/MESSAGES).
```

```
* retrieve the data from the data tables
select single * from /delego/ccext into x_ccext where cckey = x_cckey.
if sy-subrc <> 0.
  message w050(/DELEGO/MESSAGES).
```

```
clear ccdata-ccnum.
else.
  move-corresponding x_ccext to ccdata.
  delete from /delego/ccext where cckey = x_cckey.
```

```
ENHANCEMENT-SECTION /DELEGO/ES_EXTERNAL_ENTRY1 SPOTS /DELEGO/ES_EXTERNAL_ENTRY.
```

```
* add any additional fields that may be customer specific
* move: x_ccext-track1 to ccdata-zztrack1,
* x_ccext-track2 to ccdata-zztrack2.
```

```
END-ENHANCEMENT-SECTION.
```

```
$$$-Start: /DELEGO/ES_EXTERNAL_ENTRY1-----
---$$$
ENHANCEMENT 5 /DELEGO/MV45AF0Z. "active version
```

```
* add any additional fields that may be customer specific
move-corresponding x_ccext to ccard_cvv.
move: x_ccext-track1 to ccdata-zztrack1,
x_ccext-track2 to ccdata-zztrack2.
```

```
ENDENHANCEMENT.
```

```
$$$-End: /DELEGO/ES_EXTERNAL_ENTRY1-----
-$$$
```

```
endif.
```

```
* Kartendaten sichern
PERFORM zahlungskarten_bearbeiten.
```

```
* Absicherungsform wird gesetzt
PERFORM vbak_bearbeiten IN PROGRAM sapfv45k.
```

```
ENDFORM. "/DELEGO/CC_EXTERNAL_GET
```



2.3.8 Include /DELEGO/CC_FPLTC_ADD

Using transaction SE80, create the include /DELEGO/CC_FPLTC_ADD in the package /DELEGO/EXTERNAL.

```
*&-----*
*& Include      /DELEGO/CC_FPLTC_ADD
*&-----*

FORM /DELEGO/CC_FPLTC_ADD.

    data: x_ccext type /delego/ccext.

    clear xfplt.
    clear fpltc.

    clear fplt-fpltr.

    * call the web page to collect the credit card details
    CALL FUNCTION '/DELEGO/CC_GET_CCEXT'
        IMPORTING
            X_CCEXT          = x_ccext
        EXCEPTIONS
            NUMBER_ERROR     = 1
            ERROR_REGISTRATION = 2
            OTHERS            = 3.

    * tell the user something failed if it did.
    if sy-subrc <> 0.
        message w050(/DELEGO/MESSAGES).
        exit.
    endif.

    move-corresponding x_ccext to fpltc.

    * export the address data to memory to be retrieved by the calling program
    export x_ccext to memory id '/DELEGO/CCEXT'.

ENHANCEMENT-SECTION      /DELEGO/ES_EXTERNAL_ENTRY1 SPOTS /DELEGO/ES_SAPLV60F.

    * add any additional fields that may be customer specific
    *   move: x_ccext-track1 to fpltc-zztrack1,
    *         x_ccext-track2 to fpltc-zztrack2.

END-ENHANCEMENT-SECTION.

    * insert the card information into the table and flag it as such
    FLG_XFPLT = 'I'.

    perform fplt_fuellen.
    perform CC_FPLT_FAKWR_PRUEFEN.
    perform fplt_bearbeiten.
    PERFORM CC_UNBEG.
    PERFORM CC_FPLT_TOTAL.

ENDFORM.
```

2.3.9 Include MV45AFZZ

*Manual Operation – Not Delivered with Transports*

Add a call to the include /DELEGO/CC_EXTERNAL_ENTRY to the bottom of the code section. The include has the necessary code to define the operation when the card entry button is pushed.

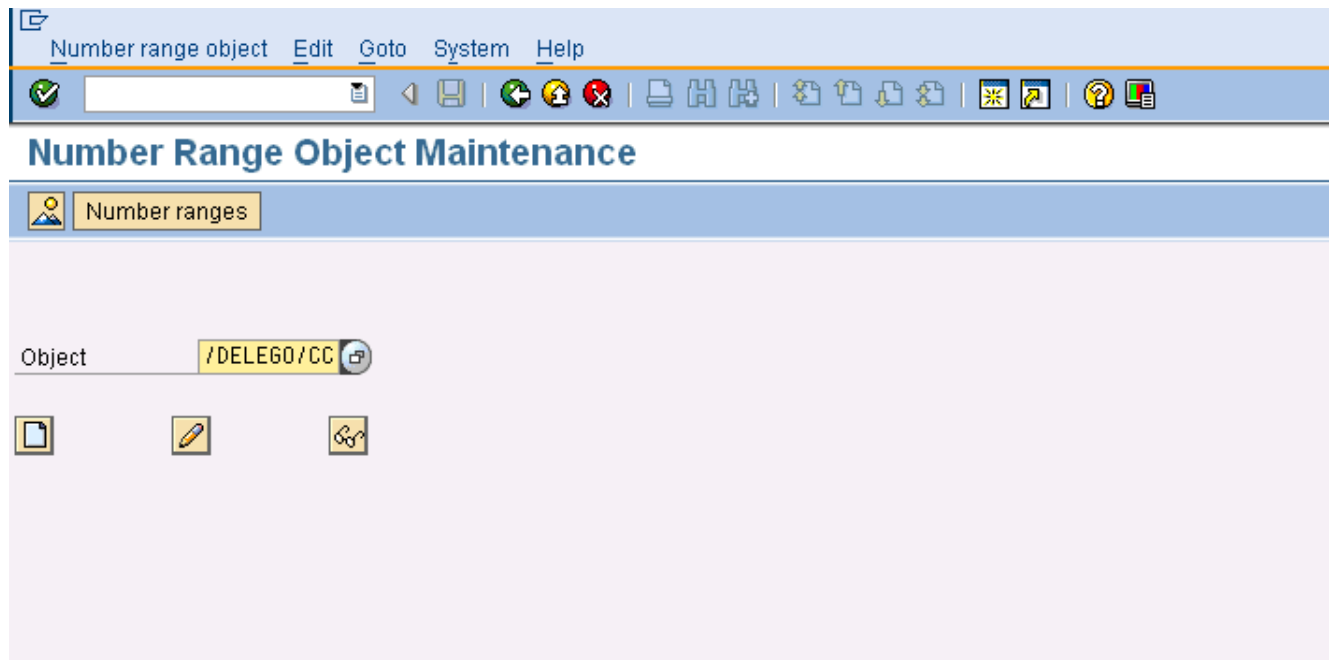
```
*eject
*-----*
* FORM USEREXIT_SAVE_DOCUMENT_PREPARE *
*-----*
* This userexit can be used for changes or checks, before a *
* document is saved. *
* *
* If field T180-TRTYP contents 'H', the document will be *
* created, else it will be changed. *
* *
* This form is called at the beginning of form BELEG_SICHERN *
* *
*-----*
FORM userexit_save_document_prepare.

ENDFORM. "USEREXIT_SAVE_DOCUMENT_PREPARE
*eject

*{ INSERT
*
include /DELEGO/CC_EXTERNAL_ENTRY.
*} INSERT
```

2.3.10 Create a new Number Range /DELEGO/CC

- 1) Use transaction SNRO to create the number range object /DELEGO/CC used to create credit card reference numbers.





- 2) Define the number range as follows:

Short Text: Delego Number Range

Long Text: External Credit Card Number Range

Number Length Domain: Char16

Warning %: 5

Main Memory Buffering: X

No. of Numbers in Buffer: 0

Number Range Object Edit Goto System Help

Number Range Object: Create

Object: /DELEGO/CC Number range object has no intervals

Short text: Delego Number Range

Long text: External Credit Card Number Retrieval

Interval characteristics

Subobject data element:

To-year flag: ☐

Number length domain: char16

No interval rolling: ☐

Customizing specifications

Number range transaction:

Warning %: 5

Main memory buffering: ☒ No. of numbers in buffer: 0

Group specification

Group table: Maintain text

Subobject field in group table:

Fld. NoRangeElement: Delete group ref.

Fields int./ext. no.range no.:

Fld. NoRangeNo:

Display element text: ☐

- 3) Save and assign this number range to the package /DELEGO/EXTERNAL.



The 'Create Object Directory Entry' dialog box is shown. The 'Object' field contains 'R3TR NROB /DELEGO/CC'. The 'Attributes' section includes: 'Package' set to '/DELEGO/EXTERNAL', 'Person Responsible' set to 'RMCCAMMON', 'Original System' set to 'ZED', and 'Original language' set to an empty field. At the bottom, there are buttons for 'Local Object', 'Lock Overview', and a close button.

2.3.11 Define the Number Range

Manual Operation – Not Delivered with Transports

- 1) Using transaction SNRO enter /DELEGO/CC for the number range object and select the NUMBER RANGES button. Number range intervals should not be transported and, therefore, must be entered in every client of every ECC instance.

The screenshot shows the 'SNRO' transaction screen for 'External Credit Card Number Retrieval'. The menu bar includes 'Number range object', 'Edit', 'Goto', 'Interval', 'System', and 'Help'. The toolbar contains various icons for navigation and editing. The main area has a header 'External Credit Card Number Retrieval' and a sub-header 'Intervals'. Below this, there are buttons for 'Intervals' (with a magnifying glass icon) and 'Status' (with a pencil icon). A third 'Intervals' button (with a pencil icon) is located at the bottom left.

- 2) Change the intervals:



Number range object Edit Goto Interval System Help

Maintain Number Range Intervals

Interval

NR Object Delego Number Range

No.	From number	To number	Current number	Ext
-----	-------------	-----------	----------------	-----

Entry 0 / 0

- 3) Add the interval 01 with the range 1000000000000000 to 9999999999999999

Insert Interval

New Interval				
No	From number	To number	Current number	Ext
01	1000000000000000	9999999999999999	0	☐

Existing Number Ranges				
No	From number	To number	Number Range Status	Ext



2.3.12 Function Module /DELEGO/CC_INSERT_EXT_NUMBER

Using transaction SE80, create a new function module called /DELEGO/CC_INSERT_EXT_NUMBER as a member of the Function Group /DELEGO/EXTERNAL. Enter the short text, "Insert external number to table".

Match the data on the screen shots to the data use to create the function module.

Attributes:

Remote-Enabled Module: Selected

Function Builder: Display /DELEGO/CC_INSERT_EXT_NUMBER

Function module: /DELEGO/CC_INSERT_EXT_NUMBER Active

Attributes | Import | Export | Changing | Tables | Exceptions | Source code

Classification

Function Group: /DELEGO/EXTERNAL Integrate external data entry

Short Text: Insert external number to table

Processing Type

☐ Normal Function Module

☒ Remote-Enabled Module

☐ Update Module

☒ Start Immed.

☐ Immediate Start, No Restart

☐ Start Delayed

☐ Coll.run

General Data

Person Responsible: RMCCAMMON

Last Changed By: RMCCAMMON

Changed on: 2009.06.04

Package: /DELEGO/EXTERNAL

Program Name: /DELEGO/SAPLEXTENAL

INCLUDE Name: /DELEGO/LEXTENALU01

Original Language: EN

Not released

☐ Edit Lock

☐ Global

Import

Parameter Name	Associated Type	Optional	Pass Value
I_CCEXT	/DELEGO/CCEXT		X

Export

No export parameters.

Changing

No changing parameters.

Tables

No table parameters.

Exceptions



<i>Exception</i>	<i>Short Text</i>
INSERT_FAILED	Insertion to data table failed
DUPLICATE_KEY	Key exists in data table

Source Code

```
FUNCTION /DELEGO/CC_INSERT_EXT_NUMBER.  
*-----  
*""Local Interface:  
* IMPORTING  
* VALUE(I_CCEXT) TYPE /DELEGO/CCEXT  
* EXCEPTIONS  
* INSERT_FAILED  
* DUPLICATE_KEY  
*-----  
tables: /delego/ccext.  
  
* check for existing entry  
select single * from /delego/ccext client specified  
where mandt = i_ccext-mandt  
and cckey = i_ccext-cckey.  
if sy-subrc = 0.  
raise duplicate_key.  
endif.  
  
* save the creation date and time.  
move: sy-datum to i_ccext-cdate,  
sy-uzeit to i_ccext-ctime.  
  
* insert new entry  
insert /delego/ccext client specified from i_ccext.  
if sy-subrc <> 0.  
raise insert_failed.  
endif.  
  
ENDFUNCTION.
```

2.4 Sales Order Overview Screens

In the event that the customer does not wish to use the screens provided by Delego Software, then changes can be made to the ECC screens directly. Changes to the SAP-provided screens represent a core modification that are the responsibility of the customer. The customer can follow the same instructions provided above and make the necessary adjustments for the screen control.

2.4.1 Standard Order Overview Sales Tab

Follow these steps to create a screen which will call the screen with the credit card entry button.



- 1) Copy screen SAPMV45A-4400 to create screen SAPMV45A-9400. Be sure to copy all elements of the screen.
- 2) Change the screen flow for the PBO of screen 9400 as directed below.

* Tabstrip Karte 4.0: Subscreen: Übersicht Verkauf Default

* PBO *****
PROCESS BEFORE OUTPUT.

* Aufruf der Funktionsbausteine zur Feltauswahl
MODULE FELDAUSWAHL.

CALL SUBSCREEN SUBSCREEN_TC
INCLUDING 'SAPMV45A' '4900'.

```
*{ REPLACE 1
*\ CALL SUBSCREEN HEADER_FRAME
*\ INCLUDING 'SAPMV45A' '4440'.
CALL SUBSCREEN HEADER_FRAME
INCLUDING 'SAPMV45A' '9440'.
*} REPLACE
```

* PAI *****
PROCESS AFTER INPUT.

2.4.2 General Header Data Section

Follow these steps to create a new screen with the credit card entry button.

- 1) Copy screen SAPMV45A-4440 to create screen SAPMV45A-9440. Be sure to copy all elements of the screen.
- 2) Create a new push button. Locate the button immediately below the card number text field.



The screenshot shows the 'Screen Painter' dialog box for a 'Pushbutton' element. The 'Name' field is 'ZZDLGOCCPB' and the 'Text' field is 'Enter_Card'. The 'Line' is 14, 'Column' is 2, 'Def.Length' is 10, and 'Vis.Length' is 10. The 'Height' is 1. The 'Groups' field shows '037'. The 'FctCode' is 'ZZDLGO' and 'FctType' is blank. The 'Context menu form' is 'ON_CTMENU_'. The 'Attributes' section shows 'Format' as 'CHAR' and 'From dict.' as unchecked. The 'Conv. Exit' field is blank. The 'Search help' field is blank. The 'Ref. field' field is blank. The 'Parameter ID' field is blank. The 'SET Parameter', 'GET Parameter', 'Foreign key check', and 'Upper/Lower Case' checkboxes are all unchecked.

Name: ZZDLGOCCPB
Type: Pushbutton
Text: Enter_Card
Visible length: 10
Height: 1
Line: 14
Column: 2
Defined Length: 10
FctCode: ZZDLGOCCPB
FctType: blank
Format: CHAR
Group1: blank
Group2: blank
Group3: 037
Group4: blank

2.4.3 Other Order Types

The user may also wish to have the card entry button on other screens as well. Listed below are possible screens that can be enhanced in the same way as was done for SAPMV45A-4440. Follow the same steps outlined in Section 5 for each of the desired screens listed in section 16.

2.4.4 Order Header Payment Card Tab



Screen Painter: Attributes

El. type: Pushbutton

Name: ZZDELPBCC

Text: Enter_Card

Icon Name: [Icon]

Tooltip: [Empty]

Line: 30 Def.Length: 10

Column: 104 Vis.Length: 10

Height: 1

Groups: [Empty]

Switch: [Checked] Reaction: D

FctCode: /DELEGO FctType: [Empty]

Context Menu Form: ON_CTMENU_ [Empty]

Attributes:

Dict Program Display

Format: CHAR

☐ From dict. Modify: [Empty]

Conv. Exit: [Empty]

Search Help: [Empty]

Ref. Field: [Empty]

Parameter ID: [Empty]

☐ SET Parameter

☐ GET Parameter

☐ Foreign Key Check

☐ Upper/Lower Case

1. Login to the ECC system using the German language (DE). The language translations are copy better this way.
2. Using transaction SE80, call function group VF60F.
3. Copy screen 4204 to 9204.
4. Log off.
5. Log in to the ECC system.
6. Using transaction SE80, call function group VF60F.
7. Edit screen 9204. Be sure to edit it in the original German language. Translations are once again easier.
8. Add a button at the bottom of the screen with the following details.

Name: ZZDELPBCC
 Type: Pushbutton
 Text: Enter_Card
 Visible length: 10
 Height: 1
 Line: 30
 Column: 104
 Defined Length: 10
 FctCode: /DELEGO//PBCC
 FctType: blank
 Format: CHAR
 Group1: blank
 Group2: blank
 Group3: blank
 Group4: blank

9. Change the translations for screen 9204 according to the following table

DTXT	German	English
1	Zahlungskarten	Credit Card
2	Status der letzten Autorisierung	Authorization Status



4	Manuelle Autorisierung	Manual Authorization
	Abgerechnet in Fakturen	Settled in Billing Docs
	Enter Card	Enter Card



2.4.5 Include RV60FUST

Manual Operation – Not Delivered with Transports

Add an include called /DELEGO/CC_FPLTC_ADD to this user modifiable include.

```
***INCLUDE RV60FUST .

*****
* *
* This include is reserved for user modifications *
* *
* Additional data definitions *
* *
* The name of additional data definitions should begin with 'ZZ'. *
* *
*****

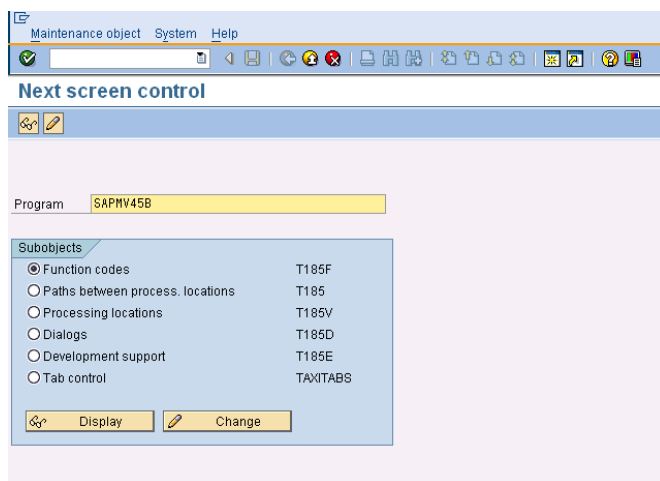
* DATA: ZZFIELD(8) TYPE C.
*
include /DELEGO/CC_FPLTC_ADD.
```

2.4.6 Screen Flow Control for Sales Order Header

Manual Operation – Not Delivered with Transports

To get the SAP ECC system to call the appropriate screens, the flow control must be changed from the standard processing for the program SAPMV45B. Follow these steps to make the necessary change.

- 1) Use transaction VFBS to add push button functionality to the screen. On the initial screen select "Function Codes".



- 2) Add the function codes ZZDLGOCCPB for both "Add (H)" and "Change (V)" modes.



Program	TType	FCode	Description	AcCat	ScrGr	Panel	HdGr	WGr	Type	Dialog
SAPMV45B	H	ZZDLGOCCPB		H	☒					
SAPMV45B	V	ZZDLGOCCPB		Y	☐					

- 3) Add the paths between processing locations by selecting this option on the initials screen of VFBS.

Maintenance object System Help

Next screen control

Program: SAPMV45B

Subobjects:

- ☐ Function codes T185F
- ☒ Paths between process. locations T185
- ☐ Processing locations T185V
- ☐ Dialogs T185D
- ☐ Development support T185E
- ☐ Tab control TAXITABS

Display Change

- 4) Create a new entry for the function code ZZDLGOCCPB according to the details in the following screenshot.



Table View Edit Goto Selection Utilities(M) System Help

Display View "Next screen process. T185: Path between processing locat

Program: SAPMV45B
Scr.seq.group: *
It.gr.scr.seq.: *

Processing location	Next location
Screen group: U0	Subsequent scr.group:
Panel: *	Subsequent panel:
Function Code: ZZDL60CCPB	Function Code: ENT1

Routines to be called

Screen change module	/DELEGO/CC_EXTERNAL_GET
Program	SAPMV45A
Return (F3) module	
Return program	

Flow control

Parameter source: A	Rep.call using CALL: ☐
Stack operation: 0	Background: ☐

- 5) Use transaction VFBS to change the screen sequence control for transactions using credit card in sales orders. Enter in the program name SAPMV45B and select the option for 'Processing Locations (T185V)'. *Change.*

Maintenance object System Help

Next screen control

Program: SAPMV45B

Subobjects

☐ Function codes	T185F
☐ Paths between process. locations	T185
☒ Processing locations	T185V
☐ Dialogs	T185D
☐ Development support	T185E
☐ Tab control	TAXITABS

Display Change

- 6) Position the data at program SAPMV45B, Screen group U0, and panel ERF1.



Another entry

Program SAPMV45B

Screen group U0

Panel ERF1

✓ ✗

- 7) Select this line and either double-click on it or use the detail command to access the complete details. In the Tab Control section, change the value of Screen module 2 from 4400 to 9400.

Table view Edit Goto Selection criteria Utilities System Help

Change View "Next screen processing T185V: Processing locations": Over

New entries

Program	Grp.	Panel	Screen	Status	Title	TCode	Program	Location description
SAPMV45B	U0	ERF1	4001	U	U01			Overview: Entry A. Orde
SAPMV45B	U0	ERF2	4001	U	U01			Overview: Entry B. Orde
SAPMV45B	U0	ERF3	4001	U	U01			Overview: Entry A. Sche
SAPMV45B	U0	ERF4	4001	U	U01			Overview: Entry B. M co
SAPMV45B	U0	ERF6	4001	U	U01			Overview: Entry B. Quo
SAPMV45B	U0	ERF7	4001	U	U01			Overview: Entry A. Cred
SAPMV45B	U0	ERF8	4001	U	U01			Overview: Entry B. Cred
SAPMV45B	U0	ERFA	4001	U	U01			Overview: Entry A. Retu
SAPMV45B	U0	ERFB	4001	U	U01			Overview: Entry A. Qual
SAPMV45B	U0	ERFD	4001	U	U01			Overview: Orderer Cred
SAPMV45B	U0	ERFE	421	UBED	U00			Independent Requirem
SAPMV45B	U0	ERFF	4001	U	U01			Overview: Entry A. Supp
SAPMV45B	U0	ERFG	4001	U	U01			Overview: Entry A. Cont
SAPMV45B	U0	ERFH	4001	U	U01			Overview: Entry B. Com
SAPMV45B	U0	ERFI	4001	U	U01			Overview: Entry B. Supp
SAPMV45B	U0	ERFT	4008	U	U01			Overview: Entry A with t
SAPMV45B	U0	ERGK	4005	U	U01			Overview: Entry A. Mast
SAPMV45B	U0	ERM1	4001	U	U01			Overview: Entry A. MAIS
SAPMV45B	U0	ERM2	4001	U	U01			Overview: Entry B. MAIS
SAPMV45B	U0	F2V1	4001	U	U01			Overview: Entry B. Valu
SAPMV45B	U0	F2V2	4001	U	U01			Overview: Entry B. Valu
SAPMV45B	U0	FEAZ	700	FE	F01			Display Error Log
SAPMV45B	U0	FRF2	4001	U	U01			Overview: Entry B. Orde
SAPMV45B	U0	FRF4	4001	U	U01			Overview: Entry B. M co
SAPMV45B	U0	FRF6	4001	U	U01			Overview: Entry B. Quo

Position... Entry 219 of 283



Change View "Next screen processing T185V: Processing locations": Deta

Program: SAPMV45B
Screen group: U0
Panel: ERF1
Location descr.: Overview: Entry A Order, Quotation (tab)

Menu Control

Status	U	Title para 1	VAKT-TEXT
Title	U01	Title para 2	
		Title para 3	
		Title para 4	

Call

Transaction code:
Program name:

DialogFct.mod

CALL module:
CALL program:

Screen

Screen no.: 4001
Fmodule screen:
Column/line Fr.: To: ☐ Return

Tab control

Screen module 1: 4021 (Header-Subscreen)
Screen module 2: 9400 (Tab-Subscreen)

- 8) Repeat steps 5 through 7 for any additional screens that the user wishes the card entry button to appear on according the following table.

Program	Screen Group	Panel	Original Screen Module 2	Revised Screen Module 2
SAPMV45B	U0	E1V1	4430	9430
SAPMV45B	U0	E1V2	4432	9432
SAPMV45B	U0	ERF1	4400	9400
SAPMV45B	U0	ERF3	4406	9406
SAPMV45B	U0	ERF7	4414	9414
SAPMV45B	U0	ERFA	4400	9400
SAPMV45B	U0	ERFB	4406	9406
SAPMV45B	U0	ERFF	4425	9425
SAPMV45B	U0	ERFG	4426	9426
SAPMV45B	U0	ERFT	4402	9402

2.5 Payment Card Tab



2.5.1 Screen Flow Control SAPLV60F

Manual Operation – Not Delivered with Transports

To get the SAP ECC system to call the appropriate screens, the flow control must be changed from the standard processing for the program SAPLV60F. Follow these steps to make the necessary change.

- 1) Use transaction VFBS to add push button functionality to the screen. On the initial screen select "Function Codes".

Maintenance object System Help

Next screen control

Program

Subobjects

☒ Function codes	T185F
☐ Paths between process. locations	T185
☐ Processing locations	T185V
☐ Dialogs	T185D
☐ Development support	T185E
☐ Tab control	TAXITABS

- 2) Add the function codes /DELEGO/PBCC for both "Add (H)" and "Change (V)" modes.



Table View Edit Goto Selection Utilities(M) System Help

Display View "Next screen process. T185: Path between processing locat

Program: SAPLV60F
 Scr.seq.group: *
 It.gr.scr.seq: *

Processing location	Next location
Screen group: *	Subsequent scr.group:
Panel: *	Subsequent panel:
Function Code: /DELE60/PBCC	Function Code: ENT1

Routines to be called

Screen change module: /DELEGO/CC_PARTNER_ADDR_CHANGE
 Program: SAPMV45A
 Return (F3) module:
 Return program:

Flow control

Parameter source: A Rep.call using CALL: ☐
 Stack operation: 0 Background: ☐

2.5.2 Payment Card Tab Screen Change (LV60FF0C)

Manual Operation – Not Delivered with Transports

The management of screens for the payment card tab on the sales order header is not controlled completely by the configuration in VFBS. Instead, it is hard-coded. Therefore, an enhancement to standard functionality is required.

- 1) Using transaction SE80, call up function group V60F.
- 2) Open include LV60FF0C.
- 3) Change to Enhancement mode (SHIFT-F4).
- 4) Show the implicit enhancements (Edit > Enhancement Operations > Show Implicit Enhancement Options)
- 5) Create an implementation of the implicit enhancement for form CUA_SETZEN, start.
- 6) Add the following code to the enhancement /DELEGO/LV60F. Create this object if needed.

```
* -----*
* FORM CUA_SETZEN *
* -----*
* .....*
* -----*
form cua_setzen.
*****"$"\SE:(1 ) Form CUA_SETZEN, Start S
*$-$-Start: (1 )-----
*$*$
ENHANCEMENT 2 /DELEGO/LV60F. "active version
```



```

* Delego Software, Inc
* copyright, 2010
* change the subscreen of the sales order header payment card tab from 4204 to 9204 so that
* the card entry button can appear
con_subscreen_zahlkarte = 9204.

```

```

ENDENHANCEMENT.

```

```

*$$$-End: (1 )-----*$$$
refresh cua_exclude.

```

2.6 Customer Master

2.6.1 Enhance Function Module SD_CCARD_MAINTAIN_DIALOG

Create an enhancement implementation, /DELEGO/SD_CCARD_MAINTAIN_DLG, of the implicit start enhancement for the function module, SD_CCARD_MAINTAIN_DIALOG. Insert the following code in the enhancement. Assign this enhancement to the package, /DELEGO/EXTERNAL.

```

ENHANCEMENT 3 /DELEGO/SD_CCARD_MAINTAIN_DLG. "active version
*
* "-----
* "Lokale Schnittstelle:
* " IMPORTING
* "     VALUE(I_KNA1) LIKE KNA1 STRUCTURE KNA1 OPTIONAL
* "     VALUE(I_DYNCL) LIKE T020-DYNCL
* "     VALUE(I_AKTYP) LIKE T020-AKTYP
* "     VALUE(I_TITLE) LIKE SY-MSGV1
* "     VALUE(I_STAND_ALONE) TYPE C OPTIONAL
* " CHANGING
* "     VALUE(E_DATALOSS) LIKE RV00F-DATALOSS OPTIONAL
* "-----

* Authority-Check
CALL FUNCTION 'SD_CCARD_KNA1_AUTHORITY_CHECK'
  EXPORTING
    I_DYNCL      = I_DYNCL
    I_AKTYP      = I_AKTYP
  EXCEPTIONS
    NO_AUTHORITY = 4.
IF SY-SUBRC NE 0.
  MESSAGE ID SY-MSGID TYPE 'I' NUMBER SY-MSGNO
    WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  EXIT.
ENDIF.

* Globale setzen
KNA1 = I_KNA1.
AKTYP_GLOBAL = I_AKTYP.
TITLE_GLOBAL = I_TITLE.
STAND_ALONE = I_STAND_ALONE.

IF STAND_ALONE = YES OR KUNNR_GLOBAL NE KNA1-KUNNR.
* Daten laden
CALL FUNCTION 'SD_CCARD_READ'
  EXPORTING

```



```

        I_KUNNR = I_KNA1-KUNNR.
    ENDIF.

* Das Pflegedynpro aufrufen
    MOVE NO TO DATEN_GESPERRT.

    CALL SCREEN 9100.

    MOVE G_DATALOSS TO E_DATALOSS.

    check 1 = 2.

ENDENHANCEMENT.

```

2.6.2 Enhance Include LV20AI0F

Using transaction SE38, create an implementation called /DELEGO/LV20AI0F of the END implicit enhancement for the include LV20AI0F. The enhancement implementation should be placed in the /DELEGO/EXTERNAL package. Place the following code into the enhancement implementation.

```

ENHANCEMENT 1  /DELEGO/LV20AI0F.      "active version
*
*&-----*
*&      Module  /DELEGO/CARD_ENTRY  INPUT
*&-----*
*      text
*-----*
MODULE /DELEGO/CARD_ENTRY INPUT.
    data: x_ccext type /delego/ccext.
    data: x_address type addr1_data.

    IF FCODE = 'PBCE'.
        CALL FUNCTION '/DELEGO/CC_GET_CCEXT'
            IMPORTING
                X_CCEXT          = x_ccext
            EXCEPTIONS
                NUMBER_ERROR      = 1
                ERROR_REGISTRATION = 2
                OTHERS             = 3.

        if sy-subrc <> 0.
            message w050 (/DELEGO/MESSAGES).
            exit.
        endif.

        move-corresponding x_ccext to vnum.
        AKTUELLE_ZEILE = ANZ_EINTRAEGE + 1.
        perform vnum_abhandeln.
        perform vckun_abhandeln.

    ENDIF.

```



```
ENDMODULE.                                " /DELEGO/CARD_ENTRY  INPUT
ENDENHANCEMENT.
```

2.6.3 Create Screen SAPLV20A/9100

Using transaction SE80 and the program definition for SAPLV20A,

- 1) Copy screen 0100 to screen 9100.
- 2) Add a pushbutton to screen 9100.

The screenshot shows the SAP Screen Painter window titled "Screen Painter: A...". The "El. type" is set to "Pushbutton". The "Name" field contains "ZZDELPBCC". The "Text" field contains "Enter_Card". The "Icon Name" field is empty. The "Tooltip" field is empty. The "Line" is 1, "Def. Length" is 10, "Column" is 74, "Vis. Length" is 10, and "Height" is 1. The "Groups" field contains "001". The "Switch" is set to "D". The "Reaction" is set to "D". The "FctCode" is "PBCE" and the "FctType" is set to "D". The "Context Menu Form" is "ON_CTMENU_". The "Attributes" section is expanded, showing the "Dict" tab. The "Format" is "CHAR". The "From dict." checkbox is unchecked. The "Modify" dropdown is set to "D". The "Conv. Exit" field is empty. The "Search Help" field is empty. The "Ref. Field" field is empty. The "Parameter ID" field is empty. The "SET Parameter", "GET Parameter", "Foreign Key Check", and "Upper/Lower Case" checkboxes are all unchecked. The bottom status bar shows the time "11:47".

- 3) Modify the flow logic for the screen to insert the module /DELEGO/CARD_ENTRY.



```
PROCESS AFTER INPUT.
  MODULE FCODE_BEARBEITEN AT EXIT-COMMAND.
  LOOP WITH CONTROL TCTRL_ZAHLUNGSKARTEN.
  CHAIN.
    FIELD: LV20A-SELKZ,
           VCNUM-CCINS,
           VCNUM-CCNUM,
           VCNUM-CCNAME,
           CCDATE-EXDATAB,
           CCDATE-EXDATBI,
           VCNUM-CCTYP,
           VCNUM-CCLOCK,
           VCKUN-CCDEF,
           VCNUM-CCACCNAME.
    MODULE CCDATE_EXDATE_POS_I ON CHAIN-REQUEST.
    MODULE DATENZEILE_BEARBEITEN ON CHAIN-REQUEST.
  ENDCHAIN.
ENDLOOP.

* xvcnum_index setzen
  MODULE TCTRL_ZAHLUNGSKARTEN_BLAETTERN.

* Start of Insert
  MODULE /DELEGO/CARD_ENTRY.
* End of Insert

MODULE FCODE_BEARBEITEN.
```



Notices

This document contains information proprietary to, and is the property of, Delego Software Inc. (“Delego”). The recipient of this document, by its retention and use, agrees to hold this document and its contents in confidence. This document shall not be transferred or communicated to any third party, without the prior written consent of Delego, in whole or part, by any means. This document or any other applicable documents provided by Delego shall be returned to Delego upon request. This Proprietary Information Notice is an integral part of this document and shall not be removed or altered. Every effort has been made to ensure the accuracy of this manual. Delego Software Inc. makes no representations or warranties with respect to any of the information contained in this manual and specifically disclaims any express or implied warranties of merchant-ability or fitness for a particular purpose with respect to such information. Delego Software Inc. shall not be liable for any errors or for incidental, indirect or consequential damages in connecting with the furnishing, performance or use of this manual.

© 2014 Delego Software Inc.