Email Custom Body through Action Configuration

Applies to:
SAP CRM 7.0. For more information, visit the Customer Relationship Management homepage.

Summary
This document helps in implementing the custom email body in CRM 7.0 through action configuration and BADI implementation.

Author: Sree Harsha Naga
Company: Prospecta Software Australia Pty Ltd
Created on: 19 May 2011

Author Bio
Sree Harsha Naga, Senior SAP Technical Consultant having 8 years of experience.
Table of Contents
Creating Action ........................................................................................................................................3
Action Definition ....................................................................................................................................8
Business Partner Changes ..................................................................................................................10
Implementation of the BADI ..............................................................................................................10
Creating Z Function Module ..............................................................................................................19
Executing the Action ..........................................................................................................................23
Result ..................................................................................................................................................24
Related Content ....................................................................................................................................25
Disclaimer and Liability Notice ...........................................................................................................26
Creating Action

Creating the Action Definition

Go to -> SPRO -> IMG -> Customer Relationship Management -> basic Functions -> Actions -> Actions In transaction -> Create Actions with Wizard

Enter the action definition & Description

Set the action when the needs to perform, when saving the document or Immediate or Using Selection report.
Setting the partner determination, this needs to done when the action should be depend on the business partner setting.
Select the action processing whether External Communication or Method call or Smartform Mail etc…

Now set the send processing details Form name, Form type (please select PDF based forms), Format (BADI Implementation which we will be creating later) & personalization type.
Enter the action description

If you need any schedule condition you can define it here
If you need any start condition you can define it here

Press the complete button
**Action Definition**

Now go to the transaction CRMC_ACTION_DEF to see the details. Select the action profile.

Double click on the action definition on the tree menu and select the action you have created.

Double click on the processing type on the tree. You can see the details created.
Now go to the transaction CRMC_ACTION_CONF to see the details. Select the action profile and double-click, select the action on the right-hand side. You can see the details as below screen.

Change the communication method to print and save.
Business Partner Changes

Enter the transaction BP.

Change the business partner details which you are going to use in the creation of the application. Change the BP Role to sold-to-party and maintain the communication details (email & standardComm. Method as email)

Implementation of the BADI

DOC_PERSONALIZE_BCS

Enter the transaction SE18. Enter the BADI name. Goto->enhancement Implementation->create
Enter the Implementation Name

```
**BAI Builder: Initial Screen for Definitions**

- Enhancement spot
- Implementation Name: DOC_PERSONALIZE_BCS

Select the filter. Goto the interface

```

```
**Business Add-In Builder: Change Implementation ZGM_MOVE_PDF**

- Implementation Name: ZGM_MOVE_PDF
- Implementation Short Text: email
- Definition name: DOC_PERSONALIZE_BCS

Attributes: Interface

- General Data
  - Package: SMARTC1
  - Language: English
  - Last changed by: 
  - Last change: 00:00:00

- Type
  - ZGM_MOVE_PDF

- Filter-depend.
  - Filter type: BCOITOTAL

Defined filters:
- ZGM_MOVE_PDF_TO_CASE

Write your code in the method PERSONALIZE_PDF_DOC_PDFIF or copy the code give below and enhance according to your requirement.
* Data declarations concerning receiver processing

```
DATA: LO_PD_PARTNER TYPE REF TO CL_PARTNER_PPF,
     LV_PPF_PARTNER TYPE PPFDPARTN0,
     LV_RECEIVER_ID TYPE BU_PARTNER,
     LV_RECEIVER_ID_U TYPE SYUNAME,
     FACADE TYPE REF TO CL_BCS,
     L_PERSONALIZE_DATA2 TYPE REF TO CL_PERSONALIZE_DATA_BCS,
     SEND_REQ TYPE REF TO CL_SEND_REQUEST_BCS,
     RECIPIENT_NEW TYPE REF TO CL_CAM_ADDRESS_BCS,
     L_BAS_ADDR TYPE AD_ADDRNUM,
     L_BAS_PERS TYPE AD_PERSNUM,
     L_BAS_COMMTYPE TYPE SO_COMTYPE,
     LV_COMMTYPE TYPE BCSDCOMTYP,
     L_CAM TYPE REF TO CL_CAM_ADDRESS_BCS,
     LO_SENDER TYPE REF TO IF_SENDER_BCS,
     BCS_EXCEPTION TYPE REF TO CX_BCS,              '#EC NEEDED
     LV_CONTACT_ADDR_ID TYPE AD_ADDRNUM.
```

* Data declarations concerning personalization

```
DATA: DUMMY(254) TYPE C,                  '#EC NEEDED
     LS_ARCHIVE TYPE TOA_DARA,
     L_GUID TYPE OS_GUID,
     LO_ORDER TYPE REF TO CL_DOC_CRM_ORDER,
     LT_HEADER_GUID TYPE CRMT_OBJECT_GUID_TAB,
     LV_OBJECT_GUID TYPE CRMT_OBJECT_GUID,
```
LS_DOC_DATA   TYPE BCSS_FAXCV,
LS_DOC        TYPE CRMT_CHM_DOC.

* Data declarations concerning PDF form processing
DATA:    LV_LANGUAGE  TYPE LANGU,
         LC_REPLANGU  TYPE LANGU  VALUE 'E',
         LS_PARTNER_FT TYPE CRM_PARTNER_FT,
         BAL_MSG     TYPE BAL_S_MSG,
         BSP_MESSAGE TYPE CRM_BSP_APPLOG,
         LS_EXCEPTION TYPE BAL_S_EXC,
         LV_PARTNER_GUID TYPE BU_PARTNER_GUID,
         LV_RELATION_PARTNER TYPE BU_PARTNER,
         LV_COUNTRY   TYPE AD_NAMCTRY.

* RMPS specific declaration
DATA:    LV_CASE_EXISTS    TYPE CRM_BOOLEAN,
         LV_OBJECT_TYPE   TYPE SWO_OBJTYP,
         LV_DOC_NAME      TYPE RMSP_TITLE,
         LV_SUBRC         TYPE SY-SUBRC.

* Receiver address determination for Channel Partner related contact persons, i.e. Partner Manager and Partner Employee:
* The contact relationship address is taken instead of the standard address.

* Get receiver
LO_PD_PARTNER = IO_PERSONALIZE_DATA->GET_PARTNER( ).
IF LO_PD_PARTNER IS BOUND.
   LV_PPF_PARTNER = LO_PD_PARTNER->GET_PARTNNO( ).
ENDIF.

* Is receiver business partner or user?
LV_RECEIVER_ID = LV_PPF_PARTNER.
CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
   EXPORTING
      INPUT = LV_RECEIVER_ID
   IMPORTING
      OUTPUT = LV_RECEIVER_ID.

* Is receiver business partner?
CALL FUNCTION 'BUPA_NUMBERS_READ'
   EXPORTING
      IV_PARTNER = LV_RECEIVER_ID
   IMPORTING
      EV_PARTNER_GUID = LV_PARTNER_GUID
   EXCEPTIONS
      NO_PARTNER_SPECIFIED   = 1
      NO_VALID_RECORD_FOUND  = 2
      INCONSISTENT_PARAMETERS = 3
      OTHERS                 = 4.
IF SY-SUBRC <> 0.
* Is receiver user?
LV_RECEIVER_ID_U = LV_PPF_PARTNER.
CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
   EXPORTING
      INPUT = LV_RECEIVER_ID_U
   IMPORTING
      OUTPUT = LV_RECEIVER_ID_U.

SAP COMMUNITY NETWORK  SDN - sdn.sap.com  |  BPX - bpx.sap.com  |  BOC - boc.sap.com  |  UAC - uac.sap.com
© 2011 SAP AG
OUTPUT = LV_RECEIVER_ID_U.

CALL FUNCTION 'CRM_CENTRALPERSON_GET'
  EXPORTING
    IV_USERNAME = LV_RECEIVER_ID_U
  IMPORTING
    EV_BU_PARTNER_GUID = LV_PARTNER_GUID
  EXCEPTIONS
    NO_CENTRAL_PERSON = 1
    NO_BUSINESS_PARTNER = 2
    NO_ID = 3
    OTHERS = 4.
IF SY-SUBRC <> 0.
  CALL METHOD CL_LOG_PPF=>ADD_MESSAGE
    EXPORTING
      IP_PROBLEMCLASS = '2'
      IP_HANDLE = IP_APPLICATION_LOG.
ENDIF.

CALL FUNCTION 'BUPA_NUMBERS_READ'
  EXPORTING
    IV_PARTNER_GUID = LV_PARTNER_GUID
  IMPORTING
    EV_PARTNER = LV_RECEIVER_ID
  EXCEPTIONS
    NO_PARTNER_SPECIFIED = 1
    NO_VALID_RECORD_FOUND = 2
    INCONSISTENT_PARAMETERS = 3
    OTHERS = 4.
IF SY-SUBRC <> 0.
  CALL METHOD CL_LOG_PPF=>ADD_MESSAGE
    EXPORTING
      IP_PROBLEMCLASS = '2'
      IP_HANDLE = IP_APPLICATION_LOG.
  RETURN.
ENDIF.

******************************************************************************
* Get receiver and sender related data
******************************************************************************
TRY.
* Extract data
  LS_DOC_DATA-RECEIVER = IO_PERSONALIZE_DATA->GET_RECIPIENT_DATA( ).
  LS_DOC_DATA-SENDER = IO_PERSONALIZE_DATA->GET_SENDER_DATA( ).
  LS_DOC_DATA-SR_DATA-SUBJECT = 'email from output letters..'.
  CATCH CX_BCS INTO BCS_EXCEPTION.

* fill exception structure
  LS_EXCEPTION-MSGTY = 'E'.
  LS_EXCEPTION-EXCEPTION = BCS_EXCEPTION.
  LS_EXCEPTION-PROBCLASS = '2'.
  LS_EXCEPTION-DETLEVEL = '1'.
  CLEAR LS_EXCEPTION-ALSORT.
  GET TIME STAMP FIELD LS_EXCEPTION-TIME_STMP.
  CLEAR LS_EXCEPTION-MSG_COUNT.
CALL FUNCTION 'BAL_LOG_EXCEPTION_ADD' 
  EXPORTING 
    I_LOG_HANDLE = IP_APPLICATION_LOG 
    I_S_EXC = LS_EXCEPTION. 
ENDTRY.

'* Get application specific data
**************************************************************************
* LO_ORDER ?= IO_APPL_OBJECT. 
* LV_OBJECT_GUID = LO_ORDER->GET_CRM_OBJ_GUID( ).
**************************************************************************
* CALL FUNCTION 'CRM_CHM_BP_LANG_CNTRY_GET' 
  EXPORTING 
    IV_PARTNER = LV_RECEIVER_ID 
    IV_REL_PARTNER = LV_RELATION_PARTNER 
    * IV_FLAG_DFT_LANG = ' ' 
  IMPORTING 
    EV_LANGUAGE = LV_LANGUAGE 
    EV_COUNTRY = LV_COUNTRY 

  CS_DOCPARAMS-LANGU = LV_LANGUAGE. 
  CS_DOCPARAMS-COUNTRY = LV_COUNTRY. 
  CS_DOCPARAMS-REPLANGU1 = LC_REPLANGU. 
  LV_DOC_NAME = IP_FORM_NAME.
**************************************************************************
* PDF Form processing preparation
**************************************************************************
*-----------fill archive parameters for archive link ------------------
* IF IS_OUTPUTPARAMS-ARCMODE = '2' OR 
  IS_OUTPUTPARAMS-ARCMODE = '3'.
  
  READ TABLE CS_DOCPARAMS-DARATAB INTO LS_ARCHIVE INDEX 1. 
* Just fill the id of your actual BOR object 
* the id is the same as the instance guid
* CALL METHOD CL_FACTORY_PPF=>GET_GUID_BY_REF 
  EXPORTING 
    IP_CLASSNAME = 'CL_DOC_CRM_ORDER' 
    IO_REF = LO_ORDER 
  RECEIVING 
    RP_GUID = L_GUID. 

  LS_ARCHIVE-OBJECT_ID = L_GUID. 
  MODIFY CS_DOCPARAMS-DARATAB FROM LS_ARCHIVE INDEX 1. 
ENDIF.
* Set the the getpdf parameter to get the pdf file 
* IS_OUTPUTPARAMS-GETPDF = IF_CRM_BOOLEAN=>TRUE. 
* Set output parameters and open spool job 
* IS_OUTPUTPARAMS-NOPRINT = IF_CRM_BOOLEAN=>FALSE. 
* Open spool job
* CALL FUNCTION 'FP_JOB_OPEN' 
  CHANGING 
    IE_OUTPUTPARAMS = IS_OUTPUTPARAMS 
  EXCEPTIONS 
    CANCEL = 1 
    USAGE_ERROR = 2
SYSTEM_ERROR  =  3
INTERNAL_ERROR =  4
OTHERS        =  5.

IF SY-SUBRC <> 0.
  MESSAGE E003(BCS_MEDIUM)
     WITH 'FP_JOB_OPEN' SY-SUBRC
     INTO DUMMY.
  CALL METHOD CL_LOG_PPF=>ADD_MESSAGE
     EXPORTING
     IP_PROBLEMCLASS = '1'
     IP_HANDLE       = IP_APPLICATION_LOG.
  EXIT.
ENDIF.

CALL FUNCTION IP_FUNCTION_NAME
  EXPORTING
     /1BCDWB/DOCPARAMS = CS_DOCPARAMS
     OBJECT_GUID      = LV_OBJECT_GUID
     IS_DOC_DATA      = LS_DOC_DATA
  IMPORTING
     /1BCDWB/FORMOUTPUT = ES_FORMOUTPUT
  EXCEPTIONS
     USAGE_ERROR      = 1
     SYSTEM_ERROR     = 2
     INTERNAL_ERROR   = 3
     OTHERS           = 4.

IF SY-SUBRC <> 0.
  CASE SY-SUBRC.
    WHEN 1.
      MESSAGE E012(BCS_MEDIUM)
         WITH IP_FUNCTION_NAME INTO DUMMY.
    WHEN 2.
      MESSAGE E013(BCS_MEDIUM)
         WITH IP_FUNCTION_NAME INTO DUMMY.
    WHEN OTHERS.
      MESSAGE E014(BCS_MEDIUM)
         WITH IP_FUNCTION_NAME INTO DUMMY.
  ENDCASE.
  CALL METHOD CL_LOG_PPF=>ADD_MESSAGE
     EXPORTING
     IP_PROBLEMCLASS = '1'
     IP_HANDLE       = IP_APPLICATION_LOG.
  EXIT.
ENDIF.

* Close spool job
CALL FUNCTION 'FP_JOB_CLOSE'
  IMPORTING
    E_RESULT   = ES_JOBOUTPUT
  EXCEPTIONS
    USAGE_ERROR = 1
    SYSTEM_ERROR = 2
    INTERNAL_ERROR = 3
    OTHERS      = 4.
IF SY-SUBRC <> 0.
CASE SY-SUBRC.
  WHEN 1.
    MESSAGE E012(BCS_MEDIUM)
      WITH IP_FUNCTION_NAME INTO DUMMY.
  WHEN 2.
    MESSAGE E013(BCS_MEDIUM)
      WITH IP_FUNCTION_NAME INTO DUMMY.
  WHEN OTHERS.
    MESSAGE E014(BCS_MEDIUM)
      WITH IP_FUNCTION_NAME INTO DUMMY.
ENDCASE.
CALL METHOD CL_LOG_PPF=>ADD_MESSAGE
  EXPORTING
    IP_PROBLEMCLASS = '1'
    IP_HANDLE = IP_APPLICATION_LOG.
EXIT.
ENDIF.

*------------------------------------------------------------------------*
* Change the email subject with the description of the form
DATA LS_NAMES TYPE ZGM_FORM_NAMES.
SELECT SINGLE * FROM ZGM_FORM_NAMES INTO LS_NAMES WHERE NAME = LV_DOC_NAME.
CP_DOCUMENT_TITLE = LS_NAMES-FDESC.
*------------------------------------------------------------------------*

DATA:
  LV_SEND_TO_ALL TYPE OS_BOOLEAN,
  LV_ADDR TYPE ZSDPRT_ADDR_S,
  LV_OUTP TYPE ZIV_DEVICE,
  LV_SUBJ TYPE SO_OBJ_DES,
  LV_COM_TYP(3) TYPE C.
DATA: L_EMAILADR TYPE AD_SMTPADR.
      CLEAR: L_EMAILADR.
DATA: LT_LINES TYPE STANDARD TABLE OF TLINE,
      LS_LINES TYPE TLINE.
REFRESH : LT_LINES.
CLEAR : LV_COM_TYP.
LV_COM_TYP = LS_DOC_DATA-RECEIVER-DATA-DEFLT_COMM.
*if the business partner communication type is email
IF LV_COM_TYP = 'INT'.
* maintain the standard text for the body of the email.
CALL FUNCTION 'READ_STDTEXT'
  EXPORTING
    ID = 'ST'
    LANGUAGE = SY-LANGU
    NAME = 'ZGM_EMAIL_BODY'
TABLES
    LINES = LT_LINES
EXCEPTIONS
  ID = 1
  LANGUAGE = 2
  NAME = 3
  NOT_FOUND = 4
  REFERENCE_CHECK = 5
  OTHERS = 6.
IF SY-SUBRC <> 0.
  ENDIF.
  * Prepare the email content
DATA: L_LINE TYPE SO_TEXT255.
    LT_TEXT TYPE TABLE OF SOLI.
CLEAR: L_LINE.

REFRESH: LT_TEXT.
LOOP AT LT_LINES INTO LS_LINES.
    L_LINE = LS_LINES-TDLINE.
    APPEND L_LINE TO LT_TEXT.
ENDLOOP.

* Get the email address of the receiver
DATA :
    LV_PARTNER TYPE BU_PARTNER,
    LT_ADSMTP TYPE STANDARD TABLE OF BAPIADSMTP,
    LS_ADSMTP TYPE BAPIADSMTP.
CLEAR : LV_PARTNER.
    LV_PARTNER = LV_PPF_PARTNER.
CALL FUNCTION 'BUPA_ADDRESS_GET_DETAIL'
    EXPORTING
        IV_PARTNER = LV_PARTNER
        TABLES
            ET_ADSMTP = LT_ADSMTP.
READ TABLE LT_ADSMTP INTO LS_ADSMTP WITH KEY STD_NO = 'X'.
    L_EMAILADR = LS_ADSMTP-E_MAIL.

CLEAR: LV_SEND_TO_ALL, LV_ADDR, LV_OUTP, LV_SUBJ.
MOVE L_EMAILADR TO LV_ADDR-RECIPI_EMAIL_ADDR.
MOVE 'test@yahoo.com' TO LV_ADDR-SENDER_EMAIL_ADDR.
MOVE 'E' TO LV_OUTP.
MOVE CP_DOCUMENT_TITLE TO LV_SUBJ.
* Function module to send the email with the PDF attachment
CALL FUNCTION 'ZGM_PDF_SEND_DATA'
    EXPORTING
        IV_DEVICE = LV_OUTP
        IV_EMAIL_SUBJECT = LV_SUBJ
        IT_EMAIL_TEXT = LT_TEXT
        IS_MAIN_DATA = ES_FORMOUTPUT
        IV_LANGUAGE = SY_LANGU
        IS_ADDRESS = LV_ADDR
    IMPORTING
        EV_SEND_TO_ALL = LV_SEND_TO_ALL
    EXCEPTIONS
        EXC_DOCUMENT = 1
        EXC_SEND_REQUEST = 2
        EXC_ADDRESS = 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.
ENDIF.
Creating Z Function Module

The function module will need to create as it not available in CRM system make a Z Function module.

Code is as follows:

```FUNCTION ZGM_PDF_SEND_DATA.
** Local Interface:
* IMPPORTING
  ** REFERENCE(IV_DEVICE) TYPE ZIV_DEVICE
  ** REFERENCE(IV_EMAIL_SUBJECT) TYPE SO_OBJ_DES
  ** REFERENCE(IT_EMAIL_TEXT) TYPE BCSY_TEXT
  ** REFERENCE(IS_MAIN_DATA) TYPE FPFORMOUTPUT
  ** REFERENCE(IV_LANGUAGE) TYPE SYLANGU
  ** REFERENCE(IS_ADDRESS) TYPE ZSDPRT_ADDR_S
* EXPORTING
  ** REFERENCE(EV_SEND_TO_ALL) TYPE OS_BOOLEAN
* EXCEPTIONS
  ** EXC_DOCUMENT```
** EXC_SEND_REQUEST
** EXC_ADDRESS

DATA:
  LO_BCS_DOCUMENT  TYPE REF TO CL_DOCUMENT_BCS,
  LO_SEND_REQUEST  TYPE REF TO CL_BCS,
  LO_RECIPIENT    TYPE REF TO IF_RECIPIENT_BCS,
  LO_SENDER      TYPE REF TO IF_SENDER_BCS,
  LO_EXC_DOC_BCS TYPE REF TO CX_DOCUMENT_BCS,
  LO_EXC_SEND_REQ_BCS TYPE REF TO CX_SEND_REQ_BCS,
  LO_EXC_ADDRESS_BCS TYPE REF TO CX_ADDRESS_BCS,
  LV_PDF_SIZE     TYPE SO_OBJ_LEN,
  LT_BCS_PDF      TYPE SOLIX_TAB.

*--- Call the business communication service
TRY.
  LO_SEND_REQUEST = CL_BCS=>CREATE_PERSISTENT( ).
  LV_PDF_SIZE = XSTRLEN( IS_MAIN_DATA-PDF ).
  CALL FUNCTION 'SCMS_XSTRING_TO_BINARY'
  EXPORTING
    BUFFER = IS_MAIN_DATA-PDF
    TABLES
      BINARY_TAB = LT_BCS_PDF.
  CASE IV_DEVICE.
    WHEN GC_DEVICE-EMAIL.
      CALL METHOD CL_DOCUMENT_BCS=>CREATE_DOCUMENT
      EXPORTING
        I_TYPE = 'RAW'
        I_SUBJECT = IV_EMAIL_SUBJECT
        I_LANGUAGE = IV_LANGUAGE
        I_TEXT = IT_EMAIL_TEXT
      RECEIVING
        RESULT = LO_BCS_DOCUMENT.
  *--- Add the document as an attachment
    CALL METHOD LO_BCS_DOCUMENT->ADD_ATTACHMENT
    EXPORTING
      I_ATTACHMENT_TYPE = 'PDF'
      I_ATTACHMENT_SIZE = LV_PDF_SIZE
      I_ATTACHMENT_SUBJECT = IV_EMAIL_SUBJECT
      I_ATTACHMENT_LANGUAGE = IV_LANGUAGE
      I_ATT_CONTENT_HEX = LT_BCS_PDF.
  *--- Set the e-mail address of the recipient
    CALL METHOD CL_CAM_ADDRESS_BCS=>CREATE_INTERNET_ADDRESS
    EXPORTING
      I_ADDRESS_STRING = IS_ADDRESS-RECIP_EMAIL_ADDR
    RECEIVING
      RESULT = LO_RECIPIENT.
  *--- Set the e-mail address of the sender
    IF NOT IS_ADDRESS-SENDER_EMAIL_ADDR IS INITIAL.
      CALL METHOD CL_CAM_ADDRESS_BCS=>CREATE_INTERNET_ADDRESS
      EXPORTING
        I_ADDRESS_STRING = IS_ADDRESS-SENDER_EMAIL_ADDR
      RECEIVING
        RESULT = LO_SENDER.
    ELSE.

LO_SENDER = CL_SAPUSER_BCS=>CREATE( SY-UNAME ).
ENDIF.

WHEN GC_DEVICE=FAX.
CALL METHOD CL_DOCUMENT_BCS=>CREATE_DOCUMENT
EXPORTING
  I_TYPE       = 'PDF'
  I_SUBJECT    = IV_EMAIL_SUBJECT
  I_LENGTH     = LV_PDF_SIZE
  I_LANGUAGE   = IV_LANGUAGE
  I_HEX        = LT_BCS_PDF
RECEIVING
  RESULT      = LO_BCS_DOCUMENT.
  *--- Set the fax address of the recipient
CALL METHOD CL_CAM_ADDRESS_BCS=>CREATE_FAX_ADDRESS
EXPORTING
  I_COUNTRY   = IS_ADDRESS-RECIP_FAX_COUNTRY
  I_NUMBER    = IS_ADDRESS-RECIP_FAX_NUMBER
RECEIVING
  RESULT      = LO_RECIPIENT.
  *--- Set the fax address of the sender
IF NOT IS_ADDRESS-SENDER_FAX_NUMBER IS INITIAL.
CALL METHOD CL_CAM_ADDRESS_BCS=>CREATE_FAX_ADDRESS
EXPORTING
  I_COUNTRY   = IS_ADDRESS-SENDER_FAX_COUNTRY
  I_NUMBER    = IS_ADDRESS-SENDER_FAX_NUMBER
RECEIVING
  RESULT      = LO_SENDER.
ELSE.
  LO_SENDER = CL_SAPUSER_BCS=>CREATE( SY-UNAME ).
ENDIF.
ENDCASE.
  *--- Assign document to the send request
CALL METHOD LO_SEND_REQUEST->SET_DOCUMENT( LO_BCS_DOCUMENT ).
  *--- Add the sender
CALL METHOD LO_SEND_REQUEST->SET_SENDER
EXPORTING
  I_SENDER = LO_SENDER.
  *--- Add the recipient
CALL METHOD LO_SEND_REQUEST->ADD_RECIPIENT
EXPORTING
  I_RECIPIENT = LO_RECIPIENT.
  *--- Send the document
CALL METHOD LO_SEND_REQUEST->SEND
EXPORTING
  I_WITH_ERROR_SCREEN = GC_TRUE
RECEIVING
  RESULT      = EV_SEND_TO_ALL.
  *--- Exception handling
CATCH CX_DOCUMENT_BCS INTO LO_EXC_DOC_BCS.
  IF NOT LO_EXC_DOC_BCS->MSGID IS INITIAL
  AND NOT LO_EXC_DOC_BCS->MSGTY IS INITIAL
  AND NOT LO_EXC_DOC_BCS->MSGNO IS INITIAL.
  MESSAGE ID LO_EXC_DOC_BCS->MSGID
    TYPE LO_EXC_DOC_BCS->MSGTY
    NUMBER LO_EXC_DOC_BCS->MSGNO
WITH LO_EXC_DOC_BCS->MSGV1
    LO_EXC_DOC_BCS->MSGV2
    LO_EXC_DOC_BCS->MSGV3
    LO_EXC_DOC_BCS->MSGV4
RAISING EXC_DOCUMENT.
ELSE.
    RAISE EXC_DOCUMENT.
ENDIF.

CATCH CX_SEND_REQ_BCS INTO LO_EXC_SEND_REQ_BCS.
    IF NOT LO_EXC_SEND_REQ_BCS->MSGID IS INITIAL
    AND NOT LO_EXC_SEND_REQ_BCS->MSGTY IS INITIAL
    AND NOT LO_EXC_SEND_REQ_BCS->MSGNO IS INITIAL.
        MESSAGE ID LO_EXC_SEND_REQ_BCS->MSGID
            TYPE LO_EXC_SEND_REQ_BCS->MSGTY
            NUMBER LO_EXC_SEND_REQ_BCS->MSGNO
        WITH LO_EXC_SEND_REQ_BCS->MSGV1
            LO_EXC_SEND_REQ_BCS->MSGV2
            LO_EXC_SEND_REQ_BCS->MSGV3
            LO_EXC_SEND_REQ_BCS->MSGV4
        RAISING EXC_SEND_REQUEST.
    ELSE.
        RAISE EXC_SEND_REQUEST.
    ENDIF.
CATCH CX_ADDRESS_BCS INTO LO_EXC_ADDRESS_BCS.
    IF NOT LO_EXC_ADDRESS_BCS->MSGID IS INITIAL
    AND NOT LO_EXC_ADDRESS_BCS->MSGTY IS INITIAL
    AND NOT LO_EXC_ADDRESS_BCS->MSGNO IS INITIAL.
        MESSAGE ID LO_EXC_ADDRESS_BCS->MSGID
            TYPE LO_EXC_ADDRESS_BCS->MSGTY
            NUMBER LO_EXC_ADDRESS_BCS->MSGNO
        WITH LO_EXC_ADDRESS_BCS->MSGV1
            LO_EXC_ADDRESS_BCS->MSGV2
            LO_EXC_ADDRESS_BCS->MSGV3
            LO_EXC_ADDRESS_BCS->MSGV4
        RAISING EXC_ADDRESS.
    ELSE.
        RAISE EXC_ADDRESS.
    ENDIF.
ENDTRY.
ENDFUNCTION.
Structure Used in the Function module

<table>
<thead>
<tr>
<th>Data Element Used in the Function module</th>
</tr>
</thead>
</table>

Executing the Action

Go to the web UI and select the application, schedule the new action which we have created and save the application.
Result

Go to SOST transaction and you can see the latest action executed if you have given as immediate processing in action definition.

Cc:  
S: CONDITIONS OF GRANT

Attachment: CONDITIONS OF GRANT.PDF (18 KB)

$0.10

Please find attached correspondence from regarding your grant application.

If you have any queries please contact a member of our Grants Team on or

With kind regards,
Related Content


ABAP - Sending email with class CL CRM EMAIL DATA

AP CRM 7.0 – E2C Setup CRM via Email Toolset

For more information, visit the Customer Relationship Management homepage
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.