

SAP How-to Guide

SAP Mobility

SAP CRM Sales 2.0 Mobile Application

Checking Customer Fact Sheets on your device every morning

A Branded Service provided by SAP Customer Solution Adoption

Applicable Releases:

SAP CRM Sales 2.0 Mobile Application

Version 1.0

Feb 2012



The Best-Run Businesses Run SAP™

© Copyright 2012 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP BusinessObjects Explorer, StreamWork, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects Software Ltd. Business Objects is an SAP company.

Sybase and Adaptive Server, iAnywhere, Sybase 365, SQL Anywhere, and other Sybase products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Sybase, Inc. Sybase is an SAP company.

All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

The information in this document is proprietary to SAP. No part of this document may be reproduced, copied, or transmitted in any form or for any purpose without the express prior written permission of SAP AG.

This document is a preliminary version and not subject to your license agreement or any other agreement with SAP. This document contains only intended strategies, developments, and functionalities of the SAP® product and is not intended to be binding upon SAP to any particular course of business, product strategy, and/or development. Please note that this document is subject to change and may be changed by SAP at any time without notice.

SAP assumes no responsibility for errors or omissions in this document. SAP does not warrant the accuracy or completeness of the information, text, graphics, links, or other items contained within this material. This document is provided without a warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

SAP shall have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials. This limitation shall not apply in cases of intent or gross negligence.

The statutory liability for personal injury and defective products is not affected. SAP has no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third-party Web pages nor provide any warranty whatsoever relating to third-party Web pages.

SAP "How-to" Guides are intended to simplify the product implementation. While specific product features and procedures typically are explained in a practical business context, it is not implied that those features and procedures are the only approach in solving a specific business problem using SAP NetWeaver. Should you wish to receive additional information, clarification or support, please refer to SAP Consulting.

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, except if such damages were caused by SAP intentionally or grossly negligent.

Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressly prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.

Document History

Document Version	Description
1.00	First official release of this guide. Feb 2012

Typographic Conventions

Type Style	Description
<i>Example Text</i>	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Cross-references to other documentation
Example text	Emphasized words or phrases in body text, graphic titles, and table titles
Example text	File and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
Example text	User entry texts. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example text>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
EXAMPLE TEXT	Keys on the keyboard, for example, F2 or ENTER.

Icons





Icon	Description
	Caution
	Note or Important
	Example
	Recommendation or Tip

Table of Contents

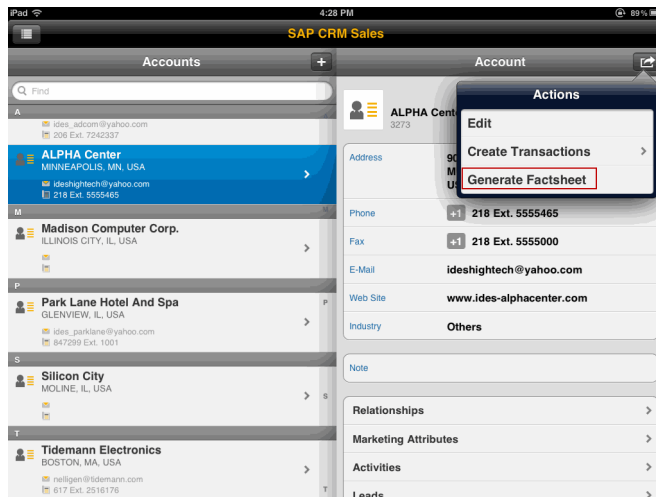
- 1. Overview.....1
 - 1.1 Scenario1
 - 1.2 A: Generate Fact Sheet by scheduled background job.1
 - 1.3 B: Create an activity with Fact Sheet attachment. 2
- 2. Sample solutions4
 - 2.1 A. Generate Fact Sheet by scheduled background job 4
 - 2.2 B. Create an Activity with Fact Sheet attachment..... 4
- 3. Troubleshooting..... 5
- 4. Sample programs6
 - 4.1 A. Generate Fact Sheet by scheduled background job 6
 - 4.2 B. Create an Activity with Fact Sheet attachment..... 7

1. Overview

1.1 Scenario

You would like to check the Fact Sheet of your accounts every morning.

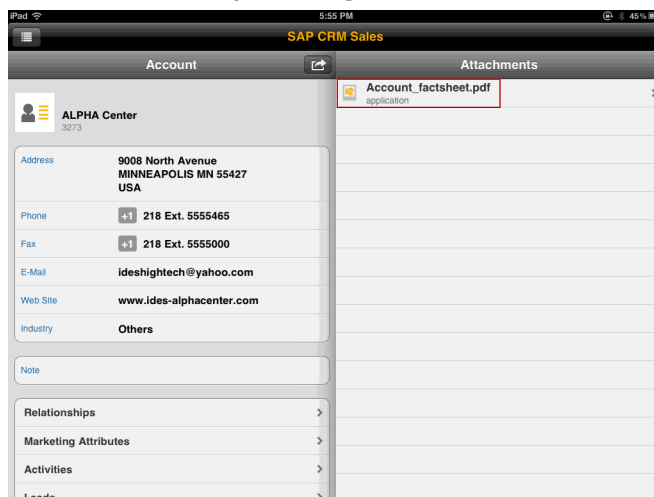
You can generate Fact Sheet from your device but it takes a few minutes to generate Fact Sheet.



The sample programs of this document give workaround solutions.

1.2 A: Generate Fact Sheet by scheduled background job.

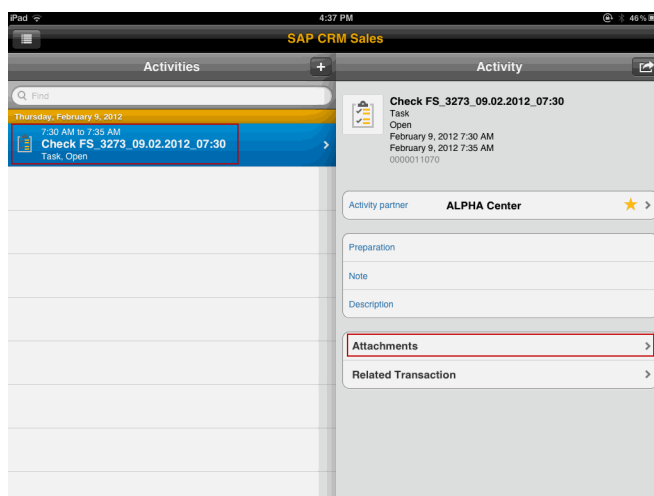
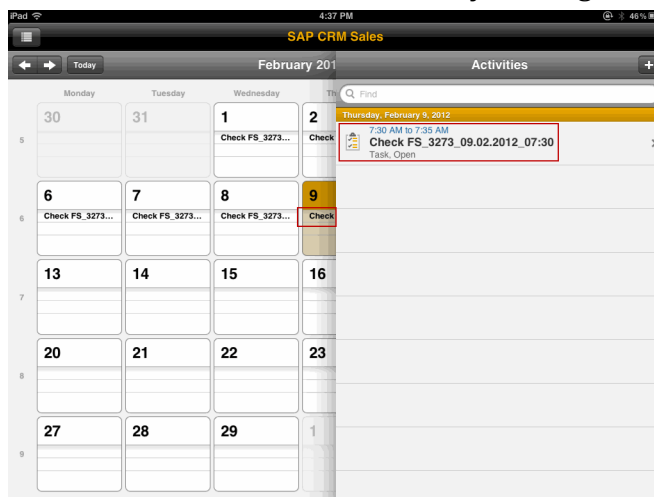
You can check the Fact Sheet from *Account* -> *Attachment*. The Fact Sheet is generated in CRM backend every morning.



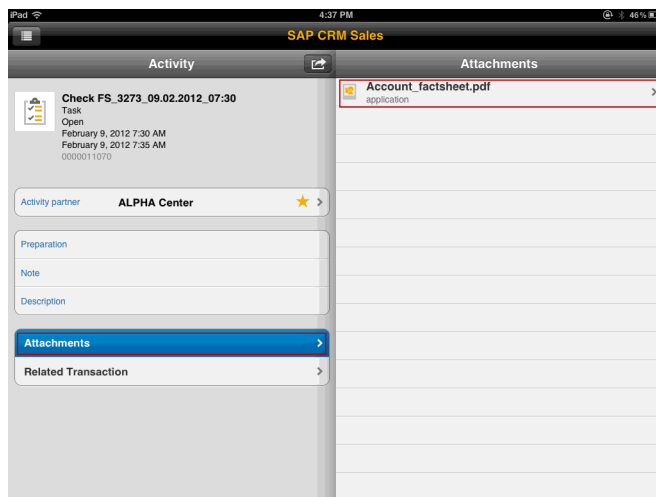
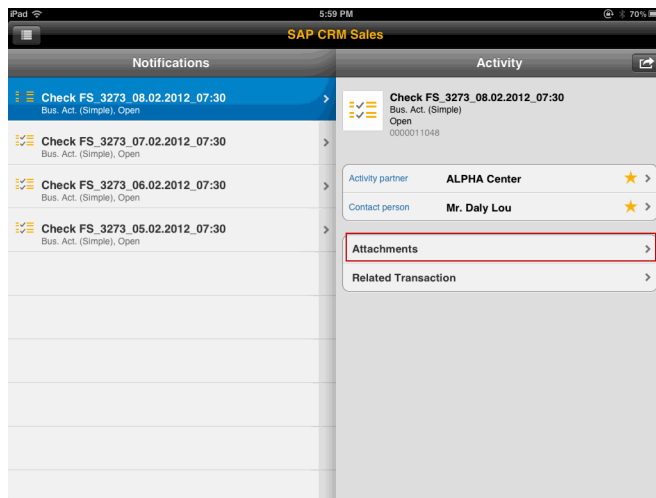
Company ALPHA Center PO Box 1030 MINNEAPOLIS MN 55427 USA				
Account Overview:				
Contact person for account 3273				
Name	Telephone	Extensi on	Function	Company department
Mr. Lou Daly	763	592990 0		
1 Total contact persons.				
Relationships for account 3273				
Relationship	Partner ID	Name	Valid from	Valid to
Has the Employee Responsible	401000	Perry Johnson	01/01/0001	12/31/9999
Is End Customer Of	300864	ED Systems	01/01/0001	12/31/9999
2 Total relationships.				
Data for sales area:				
US Sales & Service Service				
Delivery priority	Shipping conditions	Customer group	Terms of payment	Exchange rate type
00	01		0001	

1.3 B: Create an activity with Fact Sheet attachment.

You can check the Fact Sheet from *Calendar* -> *Activity*. System generates an activity task which has Fact Sheet attachment every morning.



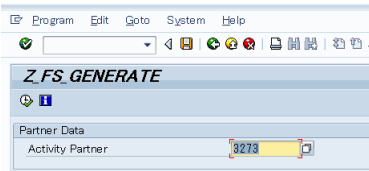
Checking Fact Sheet on your device every morning.



2. Sample solutions

2.1 A. Generate Fact Sheet by scheduled background job

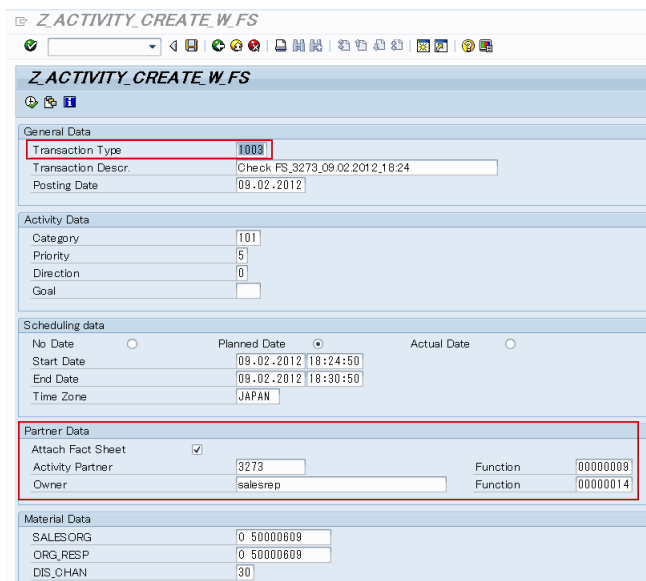
1. Copy the program CRM_CALL_ACTIVITY_INTERFACE to Z_FS_GENERATE in SE38 transaction
2. Replace the source code by sample program A.
3. Test the program Z_FS_GENERATE. Enter the Business Partner number.



4. If there is no error, create a variant.
5. Schedule the program as periodic job in SM36 transaction

2.2 B. Create an Activity with Fact Sheet attachment

1. Copy the program CRM_CALL_ACTIVITY_INTERFACE to Z_ACTIVITY_CREATE_W_FS in SE38 transaction
2. Replace the source code by sample program B.
3. Create the text element 200 "Attach Fact Sheet".
4. Test the program Z_ACTIVITY_CREATE_W_FS.
 - Select *Transaction Type*. This example uses <1003> Task
 - Check the *Attach Fact Sheet* flag
 - Enter Account ID in *Activity Partner* field
 - Check the user in *Owner* field



5. If test is OK, create a variant.

Note: When creating a variant, *Posting Date*, *Start Date* and *End Date* should be blank.

6. Schedule the program as periodic job in SM36 transaction.

3. Troubleshooting

After executing the program, you can check the generated BDoc in the SMW01 transaction.

If BDoc is not generated, issue is in z program. Check your input parameters.

If data is not transferred to device. In this case, create an activity from CRM WebUI and test.

Transaction: SMW01

Display BDoc Messages

BDoc Message Instance

BDoc Type (Generation Name)

BDoc Message ID

User (Creator)

BDoc State

Errors Non Final Intermediate State

Rejected Processed

BDoc State

Send Date and Time

No Limitation Yesterday Today 1 Hour 5 Minutes

Other Selection

Generated BDoc after Z_FS_GENERATE program is executed.

Selection Result: 4 records found.

Refresh

State	BDoc State	Description	BDoc Type	Flow Context	Description
☑	Confirmed (fully processed)		CRM_DOC_MBDIOC	mBDoc	Notification
☑	Confirmed (fully processed)		ATTACHMNT_WR	sBDoc	Notification
☑	Confirmed (fully processed)		CRM_DOC_MBDIOC	mBDoc	Notification
☑	Confirmed (fully processed)		ATTACHMNT_WR	sBDoc	Notification

Generated BDoc after Z_ACTIVITY_CREATE_W_FS program is executed.

Selection Result: 4 records found.

Refresh

State	BDoc State	Description	BDoc Type	Flow Context	Description
☑	Confirmed (fully processed)		ACTIVITY_OBJECT	sBDoc	Notification
☑	Confirmed (fully processed)		ATTACHMNT_WR	sBDoc	Notification
☑	Confirmed (fully processed)		CRM_DOC_MBDIOC	mBDoc	Notification
☑	Confirmed (fully processed)		BUS_TRANS_MSG	mBDoc	Notification

4. Sample programs

4.1 A. Generate Fact Sheet by scheduled background job

```

*&-----*
*& Report  Z_FS_GENERATE                               *
*&-----*
*&-----*
*&-----*
*&-----*
*&-----*

report  z_fs_generate line-size 150.

data: lt_return type table of bapiret2,
      ls_return type bapiret2.

*..... Partner selection screen ..... *
selection-screen begin of block partner with frame title text-104.
selection-screen begin of line.
selection-screen comment 1(25) text-057 for field cont_no.
selection-screen position pos_low.
parameters cont_no type bu_partner memory id bpa.
selection-screen end   of line.
selection-screen end of block partner.

start-of-selection.
*-----*
*  Generate Fact Sheet
*-----*
  CALL FUNCTION '/MSA/MD_BUPA_GEN_FACTSHEET'
    EXPORTING
      PARTNER      = cont_no
*   OBJECT_GUID =
*   BUS_CAT      =
    TABLES
      RETURN      = lt_return.

loop at lt_return into ls_return.
write: ls_return-message(80), ls_return-id, ls_return-number.
endloop.
write: 'end'.

```

4.2 B. Create an Activity with Fact Sheet attachment

```

*&-----*
*& Report  Z_ACTIVITY_CREATE_W_FS                               *
*&-----*
*&-----*
*&-----*
*&-----*
*&-----*

report  z_activity_create_w_fs line-size 150.

include: lcrm_busprocessnd_bapicon, com_partner.
data: lr_structdescr_bapi_te      type ref to cl_abap_structdescr,
      lr_structdescr_add_fields type ref to cl_abap_structdescr.

data: lt_return2      type table of bapiret2,
      ls_return2      type bapiret2,
      lt_return       type table of bapiret2,
      lv_msgtext(80)  type c.
data: ls_activity_bapi type bapibus2000110_header_ins,
      ls_activity_bapix type bapibus2000110_header_insx,
      ls_date_bapi     type bapibus20001_appointment_ins,
      ls_date_bapix   type bapibus20001_appointment_insx,
      ls_partner_bapi  type bapibus20001_partner_ins,
      ls_partner_bapix type bapibus20001_partner_insx,
      ls_organization_bapi type BAPIBUS20001_ORGMAN_INS,
      ls_organization_bapix type BAPIBUS20001_ORGMAN_INSX,
      ls_te_activity_i type bapi_te_crmd_activity_i,
      ls_te_activity_ix type bapi_te_crmd_activity_ix,
      ls_activity_i_add_fields type crmt_activity_i_add_fields.
* // BAPI Tables
data: lt_activity_bapi type table of bapibus2000110_header_ins,
      lt_activity_bapix type table of bapibus2000110_header_insx,
      lt_date_bapi     type table of bapibus20001_appointment_ins,
      lt_date_bapix   type table of bapibus20001_appointment_insx,
      lt_partner_bapi  type table of bapibus20001_partner_ins,
      lt_partner_bapix type table of bapibus20001_partner_insx,
      lt_organization_bapi type table of BAPIBUS20001_ORGMAN_INS,
      lt_organization_bapix type table of BAPIBUS20001_ORGMAN_INSX.

*..... Order header selection screen ..... *
selection-screen begin of block orderadm_h with frame title text-101.
parameters:
  proctype type crmt_process_type          default '1003' obligatory, "
Task
  descript type crmt_process_description,
  pdate    type crmt_posting_date          default sy-datum.
selection-screen end of block orderadm_h.

*..... Activity header selection screen ..... *

```

```
selection-screen begin of block activity_h with frame title text-102.
parameters:
  category type bapibus2000110_header_ins-category default '101', "Task
  priority type bapibus2000110_header_ins-priority default '5',
  direct type crmt_direction default '0',
  object type bapibus2000110_header_ins-objective.
selection-screen end of block activity_h.
```

**..... Appointments selection screen **

```
selection-screen begin of block date with frame title text-103.
```

```
selection-screen begin of line.
```

```
selection-screen comment 1(15) text-053 for field appt_no.
```

```
parameters:
```

```
  appt_no type crmt_boolean radiobutton group appt.
```

```
selection-screen comment 30(15) text-054 for field appt_pl.
```

```
parameters:
```

```
  appt_pl type crmt_boolean radiobutton group appt default 'X'.
```

```
selection-screen comment 60(15) text-055 for field appt_ac.
```

```
parameters:
```

```
  appt_ac type crmt_boolean radiobutton group appt.
```

```
selection-screen end of line.
```

```
selection-screen begin of line.
```

```
selection-screen comment 1(25) text-051 for field date_fro.
```

```
selection-screen position pos_low.
```

```
parameters:
```

```
  date_fro type bapibus20001_appointment_ins-date_from,
```

```
  time_fro type bapibus20001_appointment_ins-time_from.
```

```
selection-screen end of line.
```

```
selection-screen begin of line.
```

```
selection-screen comment 1(25) text-052 for field date_to.
```

```
selection-screen position pos_low.
```

```
parameters:
```

```
  date_to type bapibus20001_appointment_ins-date_to,
```

```
  time_to type bapibus20001_appointment_ins-time_to.
```

```
selection-screen end of line.
```

```
parameters tzone type timezone value check.
```

```
selection-screen end of block date.
```

**..... Partner selection screen **

```
selection-screen begin of block partner with frame title text-104.
```

```
selection-screen begin of line.
```

```
selection-screen comment 1(25) text-200 for field fs_gen.
```

```
parameters fs_gen type crmt_boolean default 'X'.
```

```
selection-screen end of line.
```

```
selection-screen begin of line.
```

```
selection-screen comment 1(25) text-057 for field cont_no.
```

```
selection-screen position pos_low.
```

```
parameters cont_no type bu_partner memory id bpa default '3273'.
```

```
selection-screen comment 70(15) text-059 for field cont_fct.
```

```
parameters cont_fct type bapibus20001_partner_ins-
```

```
partner_fct default '00000009'.
```

```

selection-screen end    of line.
selection-screen begin of line.
selection-screen comment 1(25) text-058 for field resp_no.
selection-screen position pos_low.
parameters resp_no type bapibus20001_partner_ins-
partner_no default 'salesrep'. "sy-uname
selection-screen comment 70(15) text-059 for field resp_fct.
parameters resp_fct type bapibus20001_partner_ins-
partner_fct default '00000014'.
selection-screen end    of line.
selection-screen end of block partner.

*..... Organization selection screen .....
.. *
selection-screen begin of block organization with frame title text-105.
parameters:
  salesorg type BAPIBUS20001_ORGMAN_INS-sales_org default 'O 50000609',
  org_resp type BAPIBUS20001_ORGMAN_INS-
sales_org_resp default 'O 50000609',
  dis_chan type BAPIBUS20001_ORGMAN_INS-dis_channel default '30'.
selection-screen end of block organization.

*-----*
*  INITIALIZATION                                *
*-----*
initialization.
  data: lv_datum(10) type c,
        lv_uzeit(6)  type c..
  write: sy-datum to lv_datum dd/mm/yyyy,
         sy-uzeit to lv_uzeit using edit mask '__:_'..
  concatenate 'Check FS' cont_no lv_datum lv_uzeit into descript separat
ed by '_'.
  date_fro = date_to = sy-datum.
  time_fro = sy-uzeit.
  time_to  = sy-uzeit + 360.
  CALL FUNCTION 'GET_SYSTEM_TIMEZONE'
    IMPORTING
      timezone          = tzone
    EXCEPTIONS
      customizing_missing = 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.

  lr_structdescr_bapi_te    ?= cl_abap_typedescr=>describe_by_data( ls_t
e_activity_i ).
  lr_structdescr_add_fields ?= cl_abap_typedescr=>describe_by_data( ls_a
ctivity_i_add_fields ).

```

```

*-----*
*  AT SELECTION-SCREEN OUTPUT                                *
*-----*
at selection-screen output.
  loop at screen.
    case screen-name.
      when 'RESULT'.
        read table lr_structdescr_bapi_te-
>components with key name = 'KV_RESULT'
          transporting no fields.
        if sy-subrc ne 0.
*          // Current parameter is not a part of the BAPI structure
          screen-input = false.
          modify screen.
        endif.
      when others. /// generic checks
        read table lr_structdescr_add_fields-
>components with key name = screen-name
          transporting no fields.
        if sy-subrc eq 0.
*          // Current parameter is part of the structure CRMT_ACTIVITY_I_
ADD_FIELDS
          read table lr_structdescr_bapi_te-
>components with key name = screen-name
          transporting no fields.
        if sy-subrc ne 0.
*          // Current parameter is not a part of the BAPI structure
          screen-input = false.
          modify screen.
        endif.
      endif.
    endcase.
  endloop.

*-----*
*  START-OF-SELECTION                                    *
*-----*
start-of-selection.

  format color col_normal.
  CALL FUNCTION 'GUID_CREATE'
    IMPORTING
      ev_guid_32 = ls_activity_bapi-guid.

*-----*
*  Set activity header                                    *
*-----*
  if descript is initial.
    write: sy-datum to lv_datum dd/mm/yyyy,
           sy-zeit to lv_zeit using edit mask '__:__'..
    concatenate 'Check FS' cont_no lv_datum lv_zeit into descript separ

```

```

ated by ' '.
endif.
if pdate is initial.
  pdate = sy-datum.
endif.
ls_activity_bapi-process_type = proctype.
ls_activity_bapi-description = descript.
ls_activity_bapi-posting_date = pdate.
ls_activity_bapi-mode = gc_mode-create.
ls_activity_bapi-category = category.
ls_activity_bapi-priority = priority.
ls_activity_bapi-direction = direct.
ls_activity_bapi-objective = object.
* // set activity header change fields
ls_activity_bapix-guid = true.
ls_activity_bapix-process_type = true.
ls_activity_bapix-description = true.
ls_activity_bapix-posting_date = true.
ls_activity_bapix-mode = true.
ls_activity_bapix-category = true.
ls_activity_bapix-priority = true.
ls_activity_bapix-direction = true.
ls_activity_bapix-objective = true.
append ls_activity_bapi to lt_activity_bapi.
append ls_activity_bapix to lt_activity_bapix.

*-----*
* Set date and time *
*-----*

if date_fro is initial.
  date_fro = date_to = sy-datum.
  time_fro = sy-uzeit.
  time_to = sy-uzeit + 300.
endif.

ls_date_bapi-ref_guid = ls_activity_bapi-guid.
ls_date_bapi-ref_kind = gc_object_ref_kind-orderadm_h.
ls_date_bapi-date_from = date_fro.
ls_date_bapi-time_from = time_fro.
ls_date_bapi-timezone_from = tzone.
ls_date_bapi-date_to = date_to.
ls_date_bapi-time_to = time_to.
ls_date_bapi-timezone_to = tzone.
* // set date and time change fields
ls_date_bapix-ref_guid = true.
ls_date_bapix-ref_kind = true.
ls_date_bapix-date_from = true.
ls_date_bapix-time_from = true.
ls_date_bapix-timezone_from = true.
ls_date_bapix-date_to = true.
ls_date_bapix-time_to = true.

```



```

ls_date_bapix-timezone_to = true.
if not appt_pl is initial.
  ls_date_bapi-appt_type = gc_appointment_type-planned_date.
  ls_date_bapix-appt_type = true.
  append ls_date_bapi to lt_date_bapi.
  append ls_date_bapix to lt_date_bapix.
elseif not appt_ac is initial.
  ls_date_bapi-appt_type = gc_appointment_type-actual_date.
  ls_date_bapix-appt_type = true.
  append ls_date_bapi to lt_date_bapi.
  append ls_date_bapix to lt_date_bapix.
endif.

*-----*
* Set activity partner and contact partner *
*-----*

ls_partner_bapi-ref_guid = ls_activity_bapi-guid.
ls_partner_bapi-ref_kind = gc_object_ref_kind-orderadm_h.
ls_partner_bapix-ref_guid = true.
ls_partner_bapix-ref_kind = true.
if not cont_no is initial.
  ls_partner_bapi-ref_partner_handle = 1.
  ls_partner_bapi-partner_no = cont_no.
  ls_partner_bapi-partner_fct = cont_fct.
  ls_partner_bapi-no_type = gc_partner_no_type-
business_partner_no.
  ls_partner_bapi-display_type = gc_partner_no_type-
business_partner_no.
  ls_partner_bapix-ref_partner_handle = true.
  ls_partner_bapix-partner_no = true.
  ls_partner_bapix-partner_fct = true.
  ls_partner_bapix-no_type = true.
  ls_partner_bapix-display_type = true.
  append ls_partner_bapi to lt_partner_bapi.
  append ls_partner_bapix to lt_partner_bapix.
endif.
if not resp_no is initial.
  ls_partner_bapi-ref_partner_handle = 2.
  ls_partner_bapi-partner_no = resp_no.
  ls_partner_bapi-partner_fct = resp_fct.
  ls_partner_bapi-no_type = gc_partner_no_type-user.
  ls_partner_bapi-display_type = gc_partner_no_type-user.
  ls_partner_bapix-ref_partner_handle = true.
  ls_partner_bapix-partner_no = true.
  ls_partner_bapix-partner_fct = true.
  ls_partner_bapix-no_type = true.
  ls_partner_bapix-display_type = true.
  append ls_partner_bapi to lt_partner_bapi.
  append ls_partner_bapix to lt_partner_bapix.
endif.

```

```

*-----
*  Set Organisation
*-----
ls_organization_bapi-ref_guid = ls_activity_bapi-guid.
ls_organization_bapi-ref_kind = 'A'.
ls_organization_bapi-mode = 'A'.
ls_organization_bapi-sales_org = salesorg.
ls_organization_bapi-sales_org_resp = org_resp.
ls_organization_bapi-dis_channel = dis_chan.

ls_organization_bapix-ref_guid = true.
ls_organization_bapix-ref_kind = true.
ls_organization_bapix-mode = true.
ls_organization_bapix-sales_org = true.
ls_organization_bapix-sales_org_resp = true.
ls_organization_bapix-dis_channel = true.
append ls_organization_bapi to lt_organization_bapi.
append ls_organization_bapix to lt_organization_bapix.

*-----*
*  Run the BAPI
*-----*
*  /* do the bapi call */
data: ls_process_settings type bapibus20001_procsettings,
      lt_created_header  type table of bapibus20001_header_ins,
      ls_created_header  type bapibus20001_header_ins.
ls_process_settings-msglevel = gc_msglevel-administrator.
ls_process_settings-save     = true.
CALL FUNCTION 'BAPI_BUSPROCESSND_PROCSETTINGS'
  EXPORTING
    process_settings_in = ls_process_settings
  TABLES
    return              = lt_return2.
CALL FUNCTION 'BAPI_ACTIVITYCRM_CREATEMULTI'
  TABLES
    header              = lt_activity_bapi
    headerx             = lt_activity_bapix
    partner             = lt_partner_bapi
    partnerx           = lt_partner_bapix
    ORGANISATION       = lt_organization_bapi
    ORGANISATIONX      = lt_organization_bapix
    date               = lt_date_bapi
    datex              = lt_date_bapix
*  TEXT                =
*  TEXTX              =
*  REASON             =
*  REASONX           =
*  OUTCOME            =
*  OUTCOMEX          =
*  STATUS             =
*  STATUSX           =

```

```

*      LOCATION          =
*      LOCATIONX        =
*      document_flow    = lt_docflow_bapi
*      journal           = lt_journal_bapi
*      journalx          = lt_journal_bapix
*      material          = lt_material_bapi
*      materialx         = lt_material_bapix
*      extensionin      = lt_extensionin
*      created_process   = lt_created_header
*      return            = lt_return2.
read table lt_created_header into ls_created_header index 1.
message s015(crm_userface) with ls_created_header-object_id
      into lv_msgtext.

*-----*
*  Generate Fact Sheet
*
*-----*

if fs_gen = true.
  CALL FUNCTION '/MSA/MD_BUPA_GEN_FACTSHEET'
    EXPORTING
*      PARTNER          = cont_no
      OBJECT_GUID      = ls_activity_bapi-guid
      BUS_CAT          = 'ACTV'
    TABLES
      RETURN           = lt_return.
endif.

format color col_positive.
write: / gc_msgtype-success, lv_msgtext,
      ls_created_header-logical_system,
      ls_created_header-guid hotspot.
format color off.
* // write messages to standard device
loop at lt_return2 into ls_return2.
  case ls_return2-type.
    when gc_msgtype-abort.
      write: / ls_return2-type color col_negative.
    when gc_msgtype-error.
      write: / ls_return2-type color col_total.
    when others.
      write: / ls_return2-type color off.
  endcase.
  write: ls_return2-message(80), ls_return2-id, ls_return2-number,
        ls_return2-log_no, ls_return2-log_msg_no.
endloop.

CALL FUNCTION 'BAPI_TRANSACTION_COMMIT'
  EXPORTING
    wait = true
  IMPORTING

```

```
    return = ls_return2.  
if not ls_return2 is initial.  
  case ls_return2-type.  
    when gc_msgtype-abort.  
      write: / ls_return2-type color col_negative.  
    when gc_msgtype-error.  
      write: / ls_return2-type color col_total.  
    when others.  
      write: / ls_return2-type color off.  
  endcase.  
  write: ls_return2-message(80), ls_return2-id, ls_return2-number.  
endif.
```



www.sap.com/contactsap

www.sdn.sap.com/irj/sdn/howtoguides



The Best-Run Businesses Run SAP™