

CDA

ZCL\_CRM\_BOR\_TASK

Release 740

System

---

Class: ZCL\_CRM\_BOR\_TASK  
Status: Active  
Inh. from: ZCL\_CRM\_AUDIT\_TASK

Attributes

Description: BOR task  
Instantiation: Public  
Final  
Not released  
Fixed pt.arithmetic  
Category: General Object Type  
Package: ZCRM  
Original lang.: EN  
Created by: P000520  
Created on: 15.12.2014

Attributes

Public attributes

Attrib.	Cat	Description	Ref. Type	Init. value
C_TYPE	Const	Object Type	TYPE SWO_OBJTYP	'BUS2000125'

Protected attribute

Attrib.	Cat	Description	Ref. Type
Init. value			

---

AO_CASE	Inst	BOR case	TYPE REF TO ZCL_CRM_BOR_CASE
---------	------	----------	------------------------------

---

Methods

Public methods

CONSTRUCTOR  
Description: CONSTRUCTOR  
Instance mthd

Importing parameter

VALUE(P\_KEY) TYPE SWO\_TYPEID OPTIONAL (Object key)

```

METHOD constructor.
  CALL METHOD super->constructor
  EXPORTING
    p_key = p_key.
  me->av_type = c_type.
ENDMETHOD.

```

#### GET\_LINKED\_CASE\_ID

Description: Returns the ID of the linked Case if it exists.

Instance mthd

Returning parameter

VALUE(R\_EXT\_KEY) TYPE SCMG\_EXT\_KEY (Case ID)

---

1	08.03.2015	20:54:25	
CDA	ZCL_CRM_BOR_TASK	Release 740	System

---

#### Exceptions

Type "E") CX\_BAPI\_ERROR (BAPI Error Exception: Migrated Error Messages

```

METHOD get_linked_case_id.
  DATA lx_error TYPE REF TO cx_bapi_error.
  IF me->av_object IS INITIAL.
    TRY .
      me->set_attributes( ).
      CATCH cx_bapi_error INTO lx_error.
        RAISE EXCEPTION lx_error.
    ENDTRY.
  ENDIF.
  IF me->ao_case IS NOT INITIAL.
    TRY .
      r_ext_key = me->ao_case->get_plain_attribute( 'EXT_KEY'
    ).
    CATCH cx_bapi_error INTO lx_error.
      RAISE EXCEPTION lx_error.
    ENDTRY.
  ENDIF.
ENDMETHOD.

```

#### SET\_OPPORTUNITY\_REFERENCE

Description: Creates a task with reference to an opportunity.

Instance mthd

Importing parameter

P\_RESP TYPE BU\_PARTNER (Business Partner Number)

PO\_PTASK TYPE REF TO ZCL\_CRM\_BOR\_TASK (Predecessor task)

```

        PO_OPPORTUNITY TYPE REF TO ZCL_CRM_BOR_OPPORTUNITY   (CRM
Oportunity)
        PO_LOG TYPE REF TO ZCL_CRM_BAL_LOG   OPTIONAL   (BAL Log)

Exceptions
        CX_BAPI_ERROR   (BAPI Error Exception: Migrated Error Messages
Type "E")
        CX_UUID_ERROR   (Error Class for UUID Processing Errors)
        CX_STATIC_CHECK   (Exceptions with Static and Dynamic Check of
RAISING Clause)
        METHOD set_opportunity_reference.
            DATA: lt_args          TYPE          swconttab,
                  lt_retr          TYPE          swconttab,
                  lt_header        TYPE          crmt_bus2000110_header_t,
                  lw_header        TYPE
bapibus2000110_header_ins,
                  lt_headerx       TYPE
crmt_bus2000110_headerx_t,
                  lw_headerx       TYPE
bapibus2000110_header_insx,
                  lt_partner       TYPE          crmt_partner_t,
                  lw_partner       TYPE          bapibus20001_partner_ins,
                  lt_partnerx      TYPE          crmt_partnerx_t,
                  lw_partnerx      TYPE
bapibus20001_partner_insx,
                  lt_organisation  TYPE          crmt_orgman_t,
                  lt_organisationx TYPE          crmt_orgmanx_t,
                  lt_date          TYPE          crmt_date_t,
                  lw_date          TYPE
bapibus20001_appointment_ins,
                  lt_datex        TYPE          crmt_datex_t,
                  lw_datex        TYPE
bapibus20001_appointment_insx,
                  lt_text         TYPE          crmt_text_t,
                  lt_textx        TYPE          crmt_textx_t,
                  lt_outcome       TYPE          crmt_outcome_t,

```

---

```

08.03.2015          20:54:25
2

```

---

```

ZCL_CRM_BOR_TASK          Release 740          System
CDA

```

---

```

            lt_outcomex      TYPE          crmt_outcomex_t,
            lt_reason        TYPE          crmt_reason_t,
            lt_reasonx       TYPE          crmt_reasonx_t,
            lt_status        TYPE          crmt_status_t,
            lt_statusx       TYPE          crmt_statusx_t,
            lt_location       TYPE          crmt_location_t,

```

```

                lt_locationx      TYPE          crmt_locationx_t,
                lt_inputfields    TYPE
crmt_bapi_inputfields_tab,
                lt_documentflow   TYPE
cmst_bapibus20001_docflowins_t,
                lw_documentflow   TYPE
bapibus20001_doc_flow_ins,
                lt_journal        TYPE          bapibus2000110_journal_t,
                lt_journalx       TYPE
bapibus2000110_journalx_t,
                lt_material       TYPE
bapibus2000110_material_t,
                lt_materialx      TYPE
bapibus2000110_materialx_t,
                lt_extensionin    TYPE          crmt_bapiparex,
                lt_createdprocess TYPE
cmst_bapibus20001_header_ins_t,
                lt_return         TYPE          bapiret2_t,
                lw_return         TYPE          bapiret2,
                lo_api            TYPE REF TO lcl_api,
                lx_error          TYPE REF TO cx_bapi_error,
                lx_general_error  TYPE REF TO cx_static_check,
                lt_apartners      TYPE          ztt_crm_bus_partner,
                lw_apartner       TYPE          zst_crm_bus_partner,
                lv_guid_16        TYPE          crmt_object_guid.
    lo_api ?= get_api( ).
*-----*
*   Set activity header
*
*-----*
TRY.
    lv_guid_16 = lo_api->get_new_guid( ).
    lw_header-guid = lv_guid_16.
    lw_header-process_type =
        po_ptask->get_child_type( ).
    lw_header-description =
        po_opportunity->get_plain_attribute(
'DESCRIPTION' ).
    lw_header-posting_date = sy-datlo.
    lw_header-mode          = 'A'.
    lw_header-category      = '101'.
    lw_header-priority      =
        po_opportunity->get_plain_attribute(
'PRIORITY' ).
    APPEND lw_header TO lt_header.
* // set activity header change fields
    lw_headerx-guid        = abap_true.
    lw_headerx-process_type = abap_true.
    lw_headerx-description = abap_true.
    lw_headerx-posting_date = abap_true.
    lw_headerx-mode        = abap_true.
    lw_headerx-category    = abap_true.
    lw_headerx-priority    = abap_true.
    APPEND lw_headerx TO lt_headerx.
*-----*
*   Set date and time
*
```

```

*-----*
lw_date-ref_guid      = lw_header-guid.

-----*

08.03.2015           20:54:25

3

CDA      ZCL_CRM_BOR_TASK      Release 740      System

-----*

lw_date-ref_kind      = 'A'.
lw_date-date_from     = sy-datlo.
lw_date-time_from     = sy-timlo.
lw_date-timezone_from = sy-zonlo.
lw_date-appt_type     = 'ORDERACTUAL'.
APPEND lw_date TO lt_date.
* // set date and time change fields
lw_datex-ref_guid     = abap_true.
lw_datex-ref_kind     = abap_true.
lw_datex-date_from    = abap_true.
lw_datex-time_from    = abap_true.
lw_datex-timezone_from = abap_true.
lw_datex-appt_type    = abap_true.
APPEND lw_datex TO lt_datex.

*-----*
* Set responsible partner
*
*-----*

lw_partner-ref_guid   = lw_header-guid.
lw_partner-ref_kind   = 'A'.
lw_partner-ref_partner_handle = '1'.
lw_partner-partner_no = p_resp.
* lw_partner-partner_fct = '00000014'.
lw_partner-no_type    = 'BP'.
lw_partner-display_type = 'BP'.
APPEND lw_partner TO lt_partner.
* // set partner change fields
lw_partnerx-ref_guid   = abap_true.
lw_partnerx-ref_kind   = abap_true.
lw_partnerx-ref_partner_handle = abap_true.
lw_partnerx-partner_no = abap_true.
lw_partnerx-partner_fct = abap_true.
lw_partnerx-no_type    = abap_true.
lw_partnerx-display_type = abap_true.
APPEND lw_partnerx TO lt_partnerx.

*-----*

```

```

*      * Set responsible partner
*
*-----*
      lw_partner-ref_guid          = lw_header-guid.
      lw_partner-ref_kind          = 'A'.
      lw_partner-ref_partner_handle = '2'.
      lw_partner-partner_no        =
        po_ptask->get_object_plain_attribute(
          EXPORTING
            p_object = 'EmployeeResponsible'
            p_attrib = 'BusinessPartner'
          ).
      lw_partner-partner_fct        = 'SOLICITA'.
      lw_partner-no_type           = 'BP'.
      lw_partner-display_type       = 'BP'.
      APPEND lw_partner TO lt_partner.
* // set partner change fields
      lw_partnerx-ref_guid          = abap_true.
      lw_partnerx-ref_kind          = abap_true.
      lw_partnerx-ref_partner_handle = abap_true.
      lw_partnerx-partner_no        = abap_true.

```

---

	08.03.2015	20:54:25	
--	------------	----------	--

---

4			
---	--	--	--

---

	ZCL_CRM_BOR_TASK	Release 740	System
--	------------------	-------------	--------

---

CDA			
-----	--	--	--

---

```

      lw_partnerx-partner_fct        = abap_true.
      lw_partnerx-no_type           = abap_true.
      lw_partnerx-display_type       = abap_true.
      APPEND lw_partnerx TO lt_partnerx.
*-----*
*      * Set audited partners
*
*-----*
      TRY .
        lt_apartners = po_ptask->get_partners( ).
        CATCH cx_bapi_error INTO lx_error.
          RAISE EXCEPTION lx_error.
      ENDTRY.
      m_add_audited_partner: 'ZFILAUDI',
                             'ZFUNCAUD'.
*-----*
*      * Set binary relationships
*

```

```

*-----*
-----*
        lw_documentflow-ref_guid  = lw_header-guid.
        lw_documentflow-ref_kind  = 'A'.
        lw_documentflow-brel_mode = 'A'.
        lw_documentflow-objkey_a  = po_opportunity->av_objid.
        lw_documentflow-objtype_a = po_opportunity->av_type.
        lw_documentflow-logsys_a  = lo_api-
>get_own_logical_system( ).
        lw_documentflow-vona_kind = 'A'.
        lw_documentflow-reltype   = 'VONA'.
        lw_documentflow-objkey_b  = lw_header-guid.
        lw_documentflow-brel_kind = 'A'.
        APPEND lw_documentflow TO lt_documentflow.
*-----*
-----*
*   Creating the task                               *
*-----*
-----*
        swc_set_table: lt_args 'Header'           lt_header,
                        lt_args 'HeaderX'          lt_headerx,
                        lt_args 'Partner'          lt_partner,
                        lt_args 'PartnerX'         lt_partnerx,
                        lt_args 'Organisation'      lt_organisation,
                        lt_args 'OrganisationX'     lt_organisationx,
                        lt_args 'Date'             lt_date,
                        lt_args 'DateX'            lt_datex,
                        lt_args 'Text'             lt_text,
                        lt_args 'TextX'            lt_textx,
                        lt_args 'Outcome'          lt_outcome,
                        lt_args 'OutcomeX'         lt_outcomex,
                        lt_args 'Reason'           lt_reason,
                        lt_args 'ReasonX'          lt_reasonx,
                        lt_args 'Status'           lt_status,
                        lt_args 'StatusX'          lt_statusx,
                        lt_args 'Location'         lt_location,
                        lt_args 'LocationX'        lt_locationx,
                        lt_args 'InputFields'      lt_inputfields,
                        lt_args 'DocumentFlow'     lt_documentflow,
                        lt_args 'Journal'          lt_journal,
                        lt_args 'JournalX'         lt_journalx,
                        lt_args 'Material'         lt_material,

```

08.03.2015

20:54:25

5

ZCL\_CRM\_BOR\_TASK  
CDA

Release 740

System

```

                                lt_args 'MaterialX'      lt_materialx,
                                lt_args 'ExtensionIn'     lt_extensionin.
lo_api->prepare_task_creation( ).
me->call_static_method(
    EXPORTING
        p_name      = 'CreateMultiple'
        pt_container = lt_args
    RECEIVING
        rt_container = lt_retr ).
CLEAR lt_header.
swc_get_element lt_retr 'CreatedProcess' lt_header.
swc_get_table lt_retr 'Return' lt_return.
READ TABLE lt_header INTO lw_header INDEX 1.
IF lw_header-object_id IS INITIAL.
    RAISE EXCEPTION TYPE cx_bapi_error
    EXPORTING
        status = lt_return.
ELSE.
    lo_api->commit( ).
    me->av_objid = lw_header-guid.
    TRY .
        lo_api->initialize_task( lv_guid_16 ).
        CATCH cx_bapi_error INTO lx_error.
        po_log->put_messages( lx_error->status ).
    ENDTRY.
    IF lt_return IS NOT INITIAL AND po_log IS SUPPLIED.
        po_log->put_messages( lt_return ).
    ENDIF.
ENDIF.
CATCH cx_static_check INTO lx_general_error.
FREE lo_api.
RAISE EXCEPTION lx_general_error.
ENDTRY.
FREE lo_api.
ENDMETHOD.

```

GET\_PARTNERS  
 Description: Returns a table data and a references to related  
 BUS1006005.  
 Instance mthd

Returning parameter  
 VALUE(RT\_BPDATA) TYPE ZTT\_CRM\_BUS\_PARTNER (Table of business  
 partners with BOR reference)  
 Exceptions  
 CX\_BAPI\_ERROR (BAPI Error Exception: Migrated Error Messages  
 Type "E")

```

METHOD get_partners.
    CONSTANTS: lc_bus1006005 TYPE swo_objtyp VALUE 'BUS1006005',
               lc_user      TYPE swo_objtyp VALUE 'USER'.
    DATA: lv_guid TYPE crmt_object_guid,
           lt_partners TYPE crmt_partner_external_wrkt,
           lv_objky TYPE SWO_TYPEID,
           lw_partner TYPE zst_crm_bus_partner,
           lo_api TYPE REF TO lcl_api,
           lx_error TYPE REF TO cx_bapi_error.

```



6

---

08.03.2015

20:54:25

CDA

ZCL CRM BOR TASK

Release 740

System

```

FIELD-SYMBOLS <lw_partner> TYPE crmt_partner_external_wrk.
IF me->av_object IS INITIAL AND me->av_objid IS NOT INITIAL.
  TRY .
    me->set_attributes( ).
    CATCH cx_bapi_error INTO lx_error.
    RAISE EXCEPTION lx_error.
  ENDTRY.
ENDIF.
lo_api ?= me->get_api( ).
lv_guid = me->av_objid.
TRY .
  lt_partners = lo_api->get_partners( lv_guid ).
  CATCH cx_bapi_error INTO lx_error.
  RAISE EXCEPTION lx_error.
ENDTRY.
LOOP AT lt_partners ASSIGNING <lw_partner>.
  lv_objky = <lw_partner>-ref_partner_no.
  CASE <lw_partner>-ref_no_type.
    WHEN 'BP'.
      lo_api->get_bor_reference(
        EXPORTING
          p_type = lc_bus1006005
          p_key  = lv_objky
        IMPORTING
          r_obref = lw_partner-object ).
    WHEN 'US'.
      lo_api->get_bor_reference(
        EXPORTING
          p_type = lc_user
          p_key  = lv_objky
        IMPORTING
          r_obref = lw_partner-object ).
    WHEN OTHERS.
      CONTINUE.
  ENDCASE.
  lw_partner-kind      = <lw_partner>-ref_kind.
  lw_partner-type      = <lw_partner>-ref_no_type.
  lw_partner-function  = <lw_partner>-ref_partner_fct.
  lw_partner-partner_no = <lw_partner>-ref_partner_no.
  APPEND lw_partner TO rt_bpdata.
  CLEAR lw_partner.
ENDLOOP.
ENDMETHOD.

```

## Private methods

## GET\_DB

Description: Returns a database mediator instance.

Instance mthd

Returning parameter

VALUE(RO\_DB) TYPE REF TO OBJECT (Database mediator)

METHOD get\_db.

DATA lo\_db TYPE REF TO lcl\_db.

---

	08.03.2015	20:54:25	
--	------------	----------	--

---

7			
---	--	--	--

---

	ZCL_CRM_BOR_TASK	Release 740	System
--	------------------	-------------	--------

---

CDA			
-----	--	--	--

---

```

lo_db = NEW lcl_db( ).
IF lo_db IS BOUND.
    ro_db = lo_db.
ENDIF.
FREE lo_db.
ENDMETHOD.

```

## GET\_API

Description: Returns a API calls mediator instance.

Instance mthd

Returning parameter

VALUE(RO\_API) TYPE REF TO OBJECT (API calls mediator)

method GET\_API.

DATA lo\_api TYPE REF TO lcl\_api.

lo\_api = NEW lcl\_api( ).

IF lo\_api IS BOUND.

ro\_api = lo\_api.

ENDIF.

FREE lo\_api.

endmethod.

## Redefined Methods

## SET\_ATTRIBUTES

METHOD set\_attributes.

DATA: lx\_error TYPE REF TO cx\_bapi\_error,

lo\_api TYPE REF TO lcl\_api,

lv\_cguid TYPE swo\_typeid.

TRY.

```

        CALL METHOD super->set_attributes.
        CATCH cx_bapi_error INTO lx_error.
        RAISE EXCEPTION lx_error.
    ENDTRY.
    lo_api ?= me->get_api( ).
    TRY .
        lv_cguid = lo_api->get_case( me->av_objid ).
        CATCH cx_bapi_error INTO lx_error.
        RAISE EXCEPTION lx_error.
    ENDTRY.
    IF lv_cguid IS NOT INITIAL.
        me->ao_case = NEW zcl_crm_bor_case( lv_cguid ).
    ENDIF.
ENDMETHOD.

SET_INSTANCE_BY_ID
METHOD set_instance_by_id.
    DATA: lw_return    TYPE      bapiret2,
           lt_return    TYPE      bapiret2_t,
           lvc_dummy(1) TYPE      c,
           lv_guid      TYPE      crmt_object_guid,
           lv_id        TYPE      crmt_object_id_db,

```

---

	08.03.2015	20:54:25	
--	------------	----------	--

---

8			
---	--	--	--

---

	ZCL_CRM_BOR_TASK	Release 740	System
--	------------------	-------------	--------

---

CDA			
-----	--	--	--

---

```

        lo_db      TYPE REF TO lcl_db.
    lo_db ?= get_db( ).
    lv_id = p_id.
    SHIFT lv_id RIGHT DELETING TRAILING space.
    OVERLAY lv_id WITH '0000000000'.
    TRY .
        lo_db->get_guid( EXPORTING p_type = me->av_type
                        p_id      = lv_id
                        RECEIVING r_guid = lv_guid ).
    CATCH cx_sy_sql_error.
        FREE lo_db.
        MESSAGE ID '00'
                TYPE 'E'
                NUMBER 398
                WITH 'Task'(m01) p_id 'not found!'(m02) space
                INTO lvc_dummy.
        lw_return-type      = sy-msgty.
        lw_return-id        = sy-msgid.
        lw_return-number    = sy-msgno.

```

```

        lw_return-message_v1 = sy-msgv1.
        lw_return-message_v2 = sy-msgv2.
        lw_return-message_v3 = sy-msgv3.
        lw_return-message_v4 = sy-msgv4.
        APPEND lw_return TO lt_return.
        RAISE EXCEPTION TYPE cx_bapi_error
            EXPORTING
                status = lt_return.
    ENDTRY.
    me->av_objid = lv_guid.
    FREE lo_db.
ENDMETHOD.

```

#### Local Types

```

*** use this source file for any type of declarations (class
*** definitions, interfaces or type declarations) you need for
*** components in the private section
Local class definitions
*** use this source file for the definition and implementation of
*** local helper classes, interface definitions and type
*** declarations
INCLUDE <cntn03>.
*-----*
----*
* Class LCL_DB
* OVERVIEW: Objects of this class mediates database queries. It
aims to
*          isolate this development from external dependencies,
as
*          might be desirable in tests.
*          The instances of this object is provided by the
private
*          method GET_DB of the main class (ZCL_CRM_BOR_TASK).
By
*          redefining the method GET_DB one can provide
appropriated
*          stubs, which must redefine the desired methods of
LCL_DB,
*          instead of accessing the actual database tables.

```

9

08.03.2015

20:54:25

CDA      ZCL\_CRM\_BOR\_TASK

Release 740

System

```

*-----*
----*
CLASS lcl_db DEFINITION.
  PUBLIC SECTION.
    METHODS get_guid IMPORTING p_type          TYPE
crmt_subobject_category_db
                                p_id          TYPE
crmt_object_id_db
                                RETURNING VALUE(r_guid) TYPE
crmt_object_guid
                                RAISING   cx_sy_sql_error.
*-----*
----*
* Method GET_GUID
*-----*
----*
* REQUIRES: The order type and id must be provided.
* EFFECTS.: If no related record is found in table
CRMD_ORDERADM_H
*
returns      returns exception type CX_SY_SQL_ERROR, otherwise
*
the related GUID.
*-----*
----*
* ->   P_TYPE      Business Trans. Cat.
* ->   P_ID        Transaction ID.
* <-   R_GUID      UUID in character form.
*-----*
----*
ENDCLASS.
*-----*
----*
* Class LCL_API
* OVERVIEW: Objects of this class mediates API calls. It aims to
*           isolate this development from external dependencies,
as
*           might be desirable in tests.
*           The instances of this object is provided by the
private
*           method GET_API of the main class (ZCL_CRM_BOR_TASK).
*           By redefining the method GET_API one can provide
*           appropriated stubs, which must redefine the desired
*           methods of LCL_API, instead of accessing the real
APIs.
*-----*
----*
CLASS lcl_api DEFINITION.
  PUBLIC SECTION.
    METHODS: get_case IMPORTING p_guid          TYPE swo_typeid
                                RETURNING VALUE(r_guid) TYPE swo_typeid
                                RAISING   cx_bapi_error,
*-----*
----*
* Method GET_CASE
*-----*
----*
* EFFECTS.: If the BOL api returns an error, this method returns
*           exception type CX_BAPI_ERROR, otherwise it returns
the GUID
*
of the linked Case.

```

```

* REFERENCE: FM CRM_GRM_GET_LINKED_CASE.
*-----*
----*
* ->    P_GUID      Object key.
* <-    R_GUID      Linked Case GUID.
*-----*
----*
        get_new_guid RETURNING VALUE(r_guid) TYPE guid_16
                RAISING  cx_uuid_error ,
*-----*
----*
* Method GET_NEW_GUID
*-----*
----*
* EFFECTS.: If the API class CL_SYSTEM_UUID returns an exception,
this
*           method passes away exception type CX_UUID_ERROR,
otherwise

```

---

```

10      08.03.2015      20:54:25

```

---

```

CDA      ZCL_CRM_BOR_TASK      Release 740      System

```

---

```

*           it returns a brand new 16 GUID.
*-----*
----*
* <-    R_GUID      GUID in 'CHAR' Format in Uppercase.
*-----*
----*
        get_own_logical_system RETURNING VALUE(r_lgsys) TYPE logsys
                RAISING  cx_bapi_error,
*-----*
----*
* Method GET_OWN_LOGICAL_SYSTEM
*-----*
----*
* EFFECTS.: If the API OWN_LOGICAL_SYSTEM_GET returns an error,
this
*           method returns exception type CX_BAPI_ERROR,
otherwise it
*           returns the Name of own logical system.
*-----*
----*
* <-    R_LGSYS      Name of own logical system.
*-----*
----*
        commit,

```

```

-----*
* -----*
* Method COMMIT
* -----*
-----*
* EFFECTS.: If the API BAPI_TRANSACTION_COMMIT returns an error,
this
*           method returns exception type CX_BAPI_ERROR.
* MODIFIES: Commits database changes.
* -----*
-----*
*           prepare_task_creation RAISING cx_bapi_error,
* -----*
-----*
* Method PREPARE_TASK_CREATION
* -----*
-----*
* EFFECTS.: If the API BAPI_BUSPROCESSND_PROCSETTINGS returns an
error,
*           this method returns exception type CX_BAPI_ERROR,
otherwise
*           sets up the environment to the call of
*           BAPI_ACTIVITYCRM_CREATEMULTI.
* MODIFIES: Prepares the processing environment for the calling
of
*           BAPI_ACTIVITYCRM_CREATEMULTI.
* -----*
-----*
*           get_partners IMPORTING p_guid          TYPE
crmt_object_guid
*                           RETURNING VALUE(rt_partners) TYPE
crmt_partner_external_wrkt
*                           RAISING   cx_bapi_error,
* -----*
-----*
* Method GET_PARTNERS
* -----*
-----*
* EFFECTS.: If the API CRM_ORDER_READ returns an error, this
method
*           returns exception type CX_BAPI_ERROR, otherwise
returns
*           the partners table supplied by the API.
* -----*
-----*
*           get_bor_reference IMPORTING p_type  TYPE swo_objtyp
*                               p_key   TYPE swo_typeid
*                               EXPORTING r_obref TYPE obj_record,
* -----*
-----*
* Method GET_BOR_REFERENCE
* -----*
-----*
* EFFECTS.: Returns the BOR reference provided by API SWO_CREATE.
* -----*
-----*
*           initialize_task IMPORTING p_guid TYPE crmt_object_guid
*                           RAISING   cx_bapi_error.

```

---

11                      08.03.2015                                      20:54:25

---

CDA                      ZCL\_CRM\_BOR\_TASK                                      Release 740                                      System

---

```

-----*
*-----*
* Method INITIALIZE_TASK
*-----*
-----*
* EFFECTS.: If the API CRM_ORDER_INITIALIZE returns an error,
this
*           method returns exception type CX_BAPI_ERROR,
otherwise
*           initializes a task.
* MODIFIES: Releases a lock on a task.
*-----*
-----*

ENDCLASS.
CLASS lcl_db IMPLEMENTATION.
METHOD get_guid.
    SELECT guid
      FROM crmd_orderadm_h
     UP TO 1 ROWS
    INTO r_guid
   WHERE object_id = p_id
     AND object_type = p_type.
ENDSELECT.
IF sy-subrc <> 0.
    RAISE EXCEPTION TYPE cx_sy_sql_error.
ENDIF.
ENDMETHOD.
ENDCLASS.
CLASS lcl_api IMPLEMENTATION.
METHOD get_case.
    TYPES:
        BEGIN OF ty_case_key,
            obj_guid      TYPE char32,
            obj_parent_guid TYPE char32,
            case_guid      TYPE char32,
        END OF ty_case_key.
    DATA: lv_tguid TYPE char32,
           lv_cguid TYPE char32,
           lw_return TYPE bapiret2,
           lt_return TYPE bapiret2_t,
           ls_message TYPE string,
           lx_root TYPE REF TO cx_root.
    DATA: lo_core TYPE REF TO cl_crm_bol_core,

```



```

        lo_case_query TYPE REF TO cl_crm_bol_query_service,
        lo_case       TYPE REF TO cl_crm_bol_entity,
        lo_iterator   TYPE REF TO if_bol_entity_col_iterator,
        lo_result     TYPE REF TO if_bol_entity_col,
        lv_name       TYPE          name_komp.
* Get query service, set properties and conduct query
  lo_core = cl_crm_bol_core=>get_instance( ).
  TRY.
    lo_core->start_up( 'CRM_CMG' ).
    CATCH cx_crm_genil_general_error INTO lx_root.
      m_raise_exception_from_root.
  ENDTRY.
  lo_case_query =
    cl_crm_bol_query_service=>get_instance( 'CmgCaseStandard'
).
    lv_name = if_crm_cmg_constants=>gc_fieldname_object_guid.
    lv_tguid = p_guid.

```

---

08.03.2015	20:54:25
------------	----------

---

ZCL_CRM_BOR_TASK	Release 740	System
------------------	-------------	--------

---

```

lo_case_query->set_property( iv_attr_name = lv_name
                           iv_value      = lv_tguid ).
lo_result = lo_case_query->get_query_result( ).
lo_iterator = lo_result->get_iterator( ).
"assumption: only link to the first case of the predecessor
lo_case = lo_iterator->get_first( ).
IF lo_case IS BOUND.
  lv_name = if_scmg_case_read=>case_guid.
  lv_cguid = lo_case->get_property_as_string( lv_name ).
ENDIF.
r_guid = lv_cguid.
FREE: lo_case, lo_iterator, lo_result, lo_case_query,
lo_core.
ENDMETHOD.
METHOD get_new_guid.
  DATA lx_error TYPE REF TO cx_uuid_error.
  TRY .
    r_guid = cl_system_uuid=>create_uuid_x16_static( ).
    CATCH cx_uuid_error INTO lx_error.
      RAISE EXCEPTION lx_error.
  ENDTRY.
ENDMETHOD.
METHOD get_own_logical_system.
  DATA: lw_return TYPE bapiret2,
        lt_return  TYPE bapiret2_t,

```

```

        ls_message TYPE string.
CALL FUNCTION 'OWN_LOGICAL_SYSTEM_GET'
  IMPORTING
    own_logical_system      = r_lgsys
  EXCEPTIONS
    own_logical_system_not_defined = 1
    OTHERS                      = 2.
IF sy-subrc <> 0.
  m_raise_exception.
ENDIF.
ENDMETHOD.
METHOD commit.
  CALL FUNCTION 'BAPI_TRANSACTION_COMMIT'
    EXPORTING
      wait = 'X'.
ENDMETHOD.
METHOD prepare_task_creation.
  DATA: lw_process_settings TYPE bapibus20001_procsettings,
        lw_return           TYPE bapiret2,
        lt_return           TYPE bapiret2_t,
        ls_message          TYPE string.
  lw_process_settings-msglevel = '6'. "professional
  lw_process_settings-save     = abap_true.
  CALL FUNCTION 'BAPI_BUSPROCESSND_PROCSETTINGS'
    EXPORTING
      process_settings_in = lw_process_settings
  TABLES
    return                = lt_return.
  READ TABLE lt_return WITH KEY type = 'E' INTO lw_return.
  IF sy-subrc = 0.

```

13

08.03.2015

20:54:25

CDA

ZCL\_CRM\_BOR\_TASK

Release 740

System

```

        m_raise_message.
        m_raise_exception.
      ENDIF.
    ENDMETHOD.
  METHOD get_partners.
    DATA: lt_guids TYPE crmt_object_guid_tab,
          lt_return TYPE bapiret2_t,
          lw_return TYPE bapiret2,
          ls_message TYPE string.
    APPEND p_guid TO lt_guids.
    CALL FUNCTION 'CRM_ORDER_READ'
      EXPORTING

```

```

        it_header_guid      = lt_guids
IMPORTING
        et_partner          = rt_partners
EXCEPTIONS
        document_not_found  = 1
        error_occurred      = 2
        document_locked     = 3
        no_change_authority = 4
        no_display_authority = 5
        no_change_allowed   = 6
        OTHERS               = 7.
IF sy-subrc <> 0.
    m_raise_exception.
ENDIF.
ENDMETHOD.
METHOD get_bor_reference.
    swc_create_object r_obref p_type p_key.
ENDMETHOD.
METHOD initialize_task.
    DATA: lt_guids TYPE crmt_object_guid_tab,
           lt_return TYPE bapiret2_t,
           lw_return TYPE bapiret2,
           ls_message TYPE string.
    APPEND p_guid TO lt_guids.
    CALL FUNCTION 'CRM_ORDER_INITIALIZE'
        EXPORTING
            it_guids_to_init = lt_guids
    EXCEPTIONS
        error_occurred      = 1
        OTHERS               = 2.
IF sy-subrc <> 0.
    m_raise_exception.
ENDIF.
ENDMETHOD.
ENDCLASS.
Macros
*** use this source file for any macro definitions you need
*** in the implementation part of the class
INCLUDE <cntn02>.
DEFINE m_raise_exception.
    message id sy-msgid

```

14

08.03.2015

20:54:25

CDA

ZCL\_CRM\_BOR\_TASK

Release 740

System

```

        type sy-msgty
        number sy-msgno
        with sy-msgv1 sy-msgv2
            sy-msgv3 sy-msgv4
        into ls_message.
    lw_return-type      = sy-msgty.
    lw_return-id        = sy-msgid.
    lw_return-number    = sy-msgno.
    lw_return-message_v1 = sy-msgv1.
    lw_return-message_v2 = sy-msgv2.
    lw_return-message_v3 = sy-msgv3.
    lw_return-message_v4 = sy-msgv4.
    lw_return-message    = ls_message.
    append lw_return to lt_return.
    raise exception type cx_bapi_error
        exporting
            status = lt_return.
END-OF-DEFINITION.
DEFINE m_raise_exception_from_root.
    lw_return-type      = 'E'.
    lw_return-message    = lx_root->get_text( ).
    append lw_return to lt_return.
    raise exception type cx_bapi_error
        exporting
            status = lt_return.
END-OF-DEFINITION.
DEFINE m_raise_message.
    message id lw_return-id
        type lw_return-type
        number lw_return-number
        with lw_return-message_v1 lw_return-message_v2
            lw_return-message_v3 lw_return-message_v4
        into ls_message.
END-OF-DEFINITION.
DEFINE m_add_audited_partner.
    READ TABLE lt_apartners INTO lw_apartner
        WITH KEY function = &1.
    IF sy-subrc = 0.
        lw_partner-ref_guid      = lw_header-guid.
        lw_partner-ref_kind      = 'A'.
        lw_partner-ref_partner_handle = '2'.
        lw_partner-partner_no     = lw_apartner-partner_no.
        lw_partner-partner_fct    = lw_apartner-function.
        lw_partner-no_type       = lw_apartner-type.
        lw_partner-display_type   = lw_apartner-type.
        APPEND lw_partner TO lt_partner.
    * // set partner change fields
        lw_partnerx-ref_guid      = abap_true.
        lw_partnerx-ref_kind      = abap_true.
        lw_partnerx-ref_partner_handle = abap_true.
        lw_partnerx-partner_no     = abap_true.
        lw_partnerx-partner_fct    = abap_true.
        lw_partnerx-no_type       = abap_true.
        lw_partnerx-display_type   = abap_true.

```

CDA	ZCL_CRM_BOR_TASK	Release 740	System
-----	------------------	-------------	--------

---

---

```
        APPEND lw_partnerx TO lt_partnerx.  
    ENDIF.  
END-OF-DEFINITION.
```

<hr/>			
16	08.03.2015	20:54:25	
CDA	ZCL_CRM_BOR_TASK	Release 740	System
<hr/>			
<hr/>			

	Overview	
	Attributes	
1	Attributes	
1	Attributes	
	Public attributes	
1	Protected attribute	
1		
	Methods	
1	Public methods	
1	CONSTRUCTOR	
1	GET_LINKED_CASE_ID	
1	SET OPPORTUNITY_REFERENCE	
2	GET PARTNERS	
6	Private methods	
7	GET_DB	
7	GET_API	
8	Redefined Methods	
8		
8		

8		
	Local Types	
9		
	Local class definitions	
9		
	Macros	14

---

	08.03.2015	20:54:25
17		