

Z\_CRM\_UPLOAD Example.txt

REPORT z\_crm\_upload  
 LINE-SIZE 100  
 LINE-COUNT 65  
 NO STANDARD PAGE HEADING  
 MESSAGE-ID 38.

\* Types

\*

TYPES:

```
BEGIN OF ty_bron_klant,
  regid(5)   TYPE c,      "Regel ID
  regel(6)   TYPE n,      "Regelnummer
  bron(3)    TYPE c,      "Bronstelsysteem
  relnr(60)  TYPE c,      "Relatienummer
  init(10)   TYPE c,      "Initialen
  tussv(20)  TYPE c,      "Tussenvoegsels
  naam(40)   TYPE c,      "Achternaam
  adres(60)  TYPE c,      "Straatnaam of postbus
  huisnr(10) TYPE c,      "Huisnummer
  hntoev(10) TYPE c,      "Huisnummer toevoeging
  postcd(10) TYPE c,      "Postcode
  woonpl(40) TYPE c,      "Plaats
  gebd(8)    TYPE c,      "Geboortedatum jjjjmdd
  land(2)    TYPE c,      "Landcode
  gesl(1)    TYPE c,      "Geslacht
  taal(2)    TYPE c,      "Taalcode
  tel1(30)   TYPE c,      "Telefoonnummer 1
  tel2(30)   TYPE c,      "Telefoonnummer 1
END OF ty_bron_klant,
```

```
BEGIN OF ty_bron_inter,
  regid(5)   TYPE c,      "Regel ID
  regel(6)   TYPE n,      "Regelnummer
  bron(3)    TYPE c,      "Bronstelsysteem
  agsnr(10)  TYPE c,      "Agentschapnummer
  naam(40)   TYPE c,      "Achternaam
  telnr(30)  TYPE c,      "Telefoonnummer
  land(2)    TYPE c,      "Landcode
  taal(2)    TYPE c,      "Taalcode
END OF ty_bron_inter,
```

```
BEGIN OF ty_bron_polis,
  regid(5)   TYPE c,      "Regel ID
  regel(6)   TYPE n,      "Regelnummer
  bron(3)    TYPE c,      "Bronstelsysteem
  relnr(60)  TYPE c,      "Relatienummer
  agsnr(6)   TYPE c,      "Agentschapnummer
  polnr(16)  TYPE c,      "Polisnummer
  codemy(3)  TYPE c,      "Maatschappijcode
  label(35)  TYPE c,      "Label polis
  prod(40)   TYPE c,      "Productomschrijving
  idat(8)    TYPE c,      "Ingangsdatum polis
  edat(8)    TYPE c,      "Einddatum polis
  stat(4)    TYPE c,      "Status
END OF ty_bron_polis,
```

```
BEGIN OF ty_best_klant,
  idnumber   TYPE but0id-idnumber,
  partner    TYPE but0id-partner,
  institute   TYPE but0id-institute,
END OF ty_best_klant,
```

```
BEGIN OF ty_best_inter,
  partner    TYPE but0id-partner,
  idnumber   TYPE but0id-idnumber,
END OF ty_best_inter,
```

Z\_CRM\_UPLOAD Example.txt

```

BEGIN OF ty_best_polis,
  object_id          TYPE crmd_orderadm_h-object_id,
  po_number_uc       TYPE crmd_sales-po_number_uc,
  guid               TYPE crmd_orderadm_h-guid,
END OF ty_best_polis,

BEGIN OF ty_tj30t,
  txt04              TYPE tj30t-txt04,
  estat              TYPE tj30t-estat,
END OF ty_tj30t.
*
* Constanten: Voor lijstopmaak
*
CONSTANTS:

  cc_split            TYPE c                VALUE ';' ,
  cc_kl_part_gr       TYPE bu_group         VALUE 'Y8' ,
  "Nummerreeks voor verzekeringsnemer
  cc_in_part_gr       TYPE bu_group         VALUE 'Y2' ,
  "Nummerreeks voor intermediar
  cc_identf_cat_klant TYPE bu_id_category   VALUE 'ZPOLIS' ,
  cc_identf_cat_inter TYPE bu_id_category   VALUE 'ZAGENT' ,
  cc_part_rol_k        TYPE bu_partnerrole   VALUE 'ZVERZN' ,
  cc_part_rol_i        TYPE bu_partnerrole   VALUE 'ZINTER' ,
  cc_procc_type        TYPE crmt_process_type VALUE 'ZASP' ,
  cc_partnfunc_v       TYPE crmt_partner_fct  VALUE 'ZVERZN' , "Partner
  functie voor verzekeringsnemer
  cc_partnfunc_i       TYPE crmt_partner_fct  VALUE 'ZINTER' , "Partner
  functie voor intermediar
  cc_profile_zasprol   TYPE tj30t-stsma      VALUE 'ZASPROL' .

INCLUDE: crm_object_names_con.

DATA:

  pr_rc               TYPE syst-subrc,
  st_partn_gr         TYPE bapibus1006_head,
  st_centr_dt         TYPE bapibus1006_central,
  st_centr_dt_x       TYPE bapibus1006_central_x,
  st_centr_ps         TYPE bapibus1006_central_person,
  st_centr_ps_x       TYPE bapibus1006_central_person_x,

  st_adres_dt         TYPE bapibus1006_address,
  st_adres_dt_x       TYPE bapibus1006_address_x,
  st_telef_dt         TYPE bapiadtel,
  st_telef_dt_x       TYPE bapiadtelx,
  st_centr_or         TYPE bapibus1006_central_organ,
  st_centr_or_x       TYPE bapibus1006_central_organ_x,
  st_bp_roles         TYPE bapibus1006_bproles,
  st_return           TYPE bapiRET2,
  st_ident_fk         TYPE bapibus1006_identification_key,
  st_ident_fc         TYPE bapibus1006_identification.
*
* Interne tabellen t.b.v. de BAPI verwerking
*
DATA:

  tb_bron              TYPE STANDARD TABLE OF string,
  tb_bron_klant        TYPE STANDARD TABLE OF ty_bron_klant,
  tb_bron_inter        TYPE STANDARD TABLE OF ty_bron_inter,
  tb_bron_polis        TYPE STANDARD TABLE OF ty_bron_polis,
  tb_verslag           TYPE STANDARD TABLE OF z_upl_verslag
                        WITH HEADER LINE,
  ls_verslag_text      TYPE z_upl_verslag-text,

*  tb_fout_klant       TYPE STANDARD TABLE OF ty_fout_klant,
  tb_header_guid       TYPE crmt_object_guid_tab,

```

Z\_CRM\_UPLOAD Example.txt

```

tb_telef_data      TYPE STANDARD TABLE OF bapiadtel,
tb_telef_data_x    TYPE STANDARD TABLE OF bapiadtelx,
tb_return          TYPE STANDARD TABLE OF bapiret2,
tb_header_ins      TYPE STANDARD TABLE OF bapibus20001_header_ins,
tb_header_upd      TYPE STANDARD TABLE OF bapibus20001_header_upd,
tb_sales           TYPE STANDARD TABLE OF bapibus20001_sales,
tb_partner         TYPE STANDARD TABLE OF bapibus20001_partner_ins,
tb_appointm        TYPE STANDARD TABLE OF
bapibus20001_appointment_ins,
tb_status          TYPE STANDARD TABLE OF bapibus20001_status_ins,
tb_inp_fields      TYPE STANDARD TABLE OF
bapibus20001_input_fields,
tb_created_process TYPE STANDARD TABLE OF bapibus20001_header_ins,
tb_saved_process   TYPE STANDARD TABLE OF bapibus20001_object_id
                    WITH HEADER LINE,
tb_best_klant      TYPE STANDARD TABLE OF ty_best_klant,
tb_best_inter      TYPE STANDARD TABLE OF ty_best_inter,
tb_best_polis      TYPE STANDARD TABLE OF ty_best_polis,

tb_tj30t           TYPE SORTED TABLE OF ty_tj30t
                    WITH UNIQUE KEY txt04
                    WITH HEADER LINE,

tb_tsad4           TYPE SORTED TABLE OF tsad4
                    WITH UNIQUE KEY prefix_txt
                    WITH HEADER LINE.

```

```

DATA:
  lt_sls_input_fields      TYPE cmst_bapibus20001_inputfield_t,
  ls_sls_input_fields      TYPE bapibus20001_input_fields,
  lv_tstamp                TYPE crmt_date_timestamp_from.

```

DATA:

```

tp_rege1(6)  TYPE n,
tp_zp_nr     TYPE bu_partner,
tp_im_nr     TYPE bu_partner,
tp_agşnr(10) TYPE n,
tp_guid      TYPE crmt_object_guid.

```

DATA:

```

wa_return      TYPE bapiret2,
wa_bron        TYPE string,
wa_bron_klant  TYPE ty_bron_klant,
wa_bron_inter  TYPE ty_bron_inter,
wa_bron_polis  TYPE ty_bron_polis,
wa_filenaam    TYPE string,
wa_header_ins  TYPE bapibus20001_header_ins,
wa_header_upd  TYPE bapibus20001_header_upd,
wa_polis_ext_reference
               TYPE bapibus20001_sales-po_number_sold,
wa_sales       TYPE bapibus20001_sales,
wa_partner     TYPE bapibus20001_partner_ins,
wa_appointm    TYPE bapibus20001_appointment_ins,
wa_status      TYPE bapibus20001_status_ins,
wa_inp_fields  TYPE bapibus20001_input_fields,
wa_best_klant  TYPE ty_best_klant,
wa_best_inter  TYPE ty_best_inter,
wa_best_polis  TYPE ty_best_polis,

wa_desktop     TYPE string.

```

\*

\* Integers

\*

DATA:

```

vi_aant_kl_vw TYPE i,
vi_aant_kl_sk TYPE i,
vi_aant_kl_gd TYPE i,
vi_aant_kl_ft TYPE i,
vi_aant_im_vw TYPE i,
vi_aant_im_sk TYPE i,
vi_aant_im_gd TYPE i,
vi_aant_im_ft TYPE i,
vi_aant_pl_vw TYPE i,
vi_aant_pl_sk TYPE i,
vi_aant_pl_gd TYPE i,
vi_aant_pl_ft TYPE i,
vi_lenght    TYPE i,
vi_lines     TYPE i.

```

DATA:

```

BEGIN OF imp_srtfd,
  id(21)                                TYPE c VALUE 'UPL_DATA_BRON_V1',
END   OF imp_srtfd,

```

```

BEGIN OF imp_log,
  file                                TYPE file_name,
  upl_datum                            TYPE d,
  upl_tijd                             TYPE t,
  nr_of_recs                          TYPE i,
  proc_datum                          TYPE d,
  proc_time                            TYPE t,
END   OF imp_log,

```

```

BEGIN OF proc_srtfd,
  id(12)                                TYPE c VALUE 'PROC_DATA_',
  cat(5)                                TYPE c,
  rest(3)                               TYPE c VALUE '_V1',
END   OF proc_srtfd.

```

```

*****
*** SELECTION-SCREEN
*****

```

SELECTION-SCREEN BEGIN OF BLOCK b1 WITH FRAME TITLE text-se1.

\*PARAMETERS:

\*

\* pa\_bron TYPE filename OBLIGATORY.

SELECTION-SCREEN SKIP 1.

PARAMETERS:

```

pa_klnt RADIOBUTTON GROUP ent1,
pa_intr RADIOBUTTON GROUP ent1,
pa_pols RADIOBUTTON GROUP ent1.

```

SELECTION-SCREEN SKIP 1.

PARAMETERS:

```

pa_init RADIOBUTTON GROUP bst1,
pa_delt RADIOBUTTON GROUP bst1.

```

SELECTION-SCREEN SKIP 1.

SELECT-OPTIONS:

so\_rege1 FOR tp\_rege1 NO-EXTENSION OBLIGATORY.

Z\_CRM\_UPLOAD Example.txt

SELECTION-SCREEN END OF BLOCK b1.

```
*****  
***  INITIALIZATION  
*****
```

INITIALIZATION.

```
so_regel-low  = 1.  
so_regel-high = 999999.
```

```
APPEND so_regel.
```

```
*****  
***  START-OF-SELECTION  
*****
```

START-OF-SELECTION.

\*

\* Lees bron gegevens

```
PERFORM import_data_from_db.
```

\*

\* Splitsen bestand

\*

```
PERFORM splitsen_bestand.
```

```
IF pr_rc IS NOT INITIAL.
```

```
  CLEAR tb_verslag.
```

```
  tb_verslag-soort = 'ERR'.
```

```
  tb_verslag-cat   = 'ABORT'.
```

```
  tb_verslag-text = 'Bronbestand bevat onbekende gegevens'.
```

```
  APPEND tb_verslag.
```

```
  tb_verslag-text = 'Verwerking afgebroken'.
```

```
  APPEND tb_verslag.
```

```
ENDIF.
```

```
FREE tb_bron.
```

```
*****  
***  END-OF-SELECTION  
*****
```

END-OF-SELECTION.

```
IF pr_rc = 0.
```

\*

\* Verwerking (INTER altijd eerst voor de koppeling)

\*

```
IF pa_klant = 'X'.
```

```
  PERFORM verwerk_klant.
```

```
  FREE tb_bron_klant.
```

```
ENDIF.
```

```
IF pa_intr = 'X'.
```

```
  PERFORM verwerk_inter.
```

```
  FREE tb_bron_inter.
```

```
ENDIF.
```

```
IF pa_polis = 'X'.
```

```
  PERFORM verwerk_polis.
```

```
  FREE tb_bron_polis.
```

```
ENDIF.
```

```
PERFORM verslag.
```

```
ENDIF.
```

```
PERFORM export_verslag.
```

Z\_CRM\_UPLOAD Example.txt

```

*&-----*
*&      Form  import_data_from_db
*&-----*
*      text
*-----*
FORM import_data_from_db.
  IMPORT log = imp_log
         file = tb_bron
         FROM DATABASE indx(zu)
         ID imp_srtfd.
  CLEAR tb_verslag.
  tb_verslag-soort = 'ALG'.
  tb_verslag-cat   = 'FILE'.
  CONCATENATE 'Bronbestand:' imp_log-file+0(60) INTO tb_verslag-text.
  APPEND tb_verslag.
  IF imp_log-file+60(60) IS NOT INITIAL.
    CONCATENATE ' ' imp_log-file+60(60) INTO tb_verslag-text.
    APPEND tb_verslag.
  ENDIF.
  IF imp_log-file+120(60) IS NOT INITIAL.
    CONCATENATE ' ' imp_log-file+120(60) INTO tb_verslag-text.
    APPEND tb_verslag.
  ENDIF.
ENDFORM.                "import_data_from_db

*&-----*
*&      Form  SPLITSEN_BESTAND
*&-----*
FORM splitsen_bestand.

  DATA:
    lo_regel                TYPE i.

  CLEAR tb_verslag.
  tb_verslag-soort = 'ALG'.
  tb_verslag-cat   = 'REGEL'.
  CONCATENATE 'Selectie:' so_regel-low '-' so_regel-high INTO tb_verslag-text.
  APPEND tb_verslag.

  LOOP AT tb_bron INTO wa_bron.
    lo_regel = syst-tabix.
    IF sy-tabix GE so_regel-low AND
       sy-tabix LE so_regel-high.

      CASE wa_bron(5).
        WHEN 'KLANT'.
          IF pa_klant = 'X'.
            PERFORM split_klant USING lo_regel.
          ENDIF.
        WHEN 'INTER'.
          IF pa_intr = 'X'.
            PERFORM split_inter USING lo_regel..
          ENDIF.
        WHEN 'POLIS'.
          IF pa_polis = 'X'.
            PERFORM split_polis USING lo_regel..
          ENDIF.
        WHEN OTHERS.
          IF wa_bron IS NOT INITIAL.
            CLEAR tb_verslag.
            tb_verslag-soort = 'ERR'.
            tb_verslag-regelnr = lo_regel.
            tb_verslag-cat   = 'ERROR'.
            tb_verslag-text = wa_bron.
            APPEND tb_verslag.
            pr_rc = 16.
          ENDIF.
      ENDCASE.
    ENDIF.
  ENDLOOP.

```

Z\_CRM\_UPLOAD Example.txt

```

        ENDIF.
    ENDLOOP.
ENDFORM.                " SPLITSEN_BESTAND

*&-----*
*&      Form  VERWERK_KLANT
*&-----*
FORM verwerk_klant.

CHECK tb_bron_klant[] IS NOT INITIAL.
* Tussenvoegsels
SELECT * FROM tsad4 INTO tb_tsad4.
    TRANSLATE tb_tsad4-prefix_txt TO UPPER CASE.
    INSERT tb_tsad4 INTO TABLE tb_tsad4.
ENDSELECT.

LOOP AT tb_bron_klant INTO wa_bron_klant.
    vi_aant_kl_vw = vi_aant_kl_vw + 1.

    PERFORM lees_klant
        USING wa_bron_klant-relnr.
    CASE pr_rc.
        WHEN 0.
            * Klant bestaat
            IF pa_delt = 'X'.
                * - alleen bij wijzigingen
                PERFORM wijzigen_klant.
            ELSE.
                ADD 1 TO vi_aant_kl_sk.
            ENDIF.
        WHEN OTHERS.
            * Klant bestaat niet
            PERFORM opvoeren_klant.
    ENDCASE.
ENDLOOP.

ENDFORM.                " VERWERK_KLANT

*&-----*
*&      Form  VERWERK_INTER
*&-----*
FORM verwerk_inter.

CHECK tb_bron_inter[] IS NOT INITIAL.

LOOP AT tb_bron_inter INTO wa_bron_inter.
    vi_aant_im_vw = vi_aant_im_vw + 1.
    PERFORM lees_inter
        USING wa_bron_inter-agsnr.
    CASE pr_rc.
        WHEN 0.
            * Intermediar bestaat
            IF pa_delt = 'X'.
                * alleen bij wijzigen
                PERFORM wijzigen_inter.
            ELSE.
                ADD 1 TO vi_aant_im_sk.
            ENDIF.
        WHEN OTHERS.
            * Intermediar bestaat niet
            PERFORM opvoeren_inter.
    ENDCASE.
ENDLOOP.

FREE tb_best_inter.
ENDFORM.                " VERWERK_INTER

*&-----*

```

```
*& Form VERWERK_POLIS
*-----*
```

```
FORM verwerk_polis.
```

```
DATA:
  lo_codemy_reference(5) TYPE c.
```

```
CHECK tb_bron_polis[] IS NOT INITIAL.
* Status tabel
SELECT txt04 estat FROM tj30t
  INTO TABLE tb_tj30t
  WHERE stsma = cc_profile_zaspro1
  AND spras = 'N'.
```

```
LOOP AT tb_bron_polis INTO wa_bron_polis.
  vi_aant_pl_vw = vi_aant_pl_vw + 1.
  IF wa_bron_polis-polnr+15(1) IS NOT INITIAL.
    CONCATENATE 'Polisnummer max. 15 lang:' wa_bron_polis-polnr
      INTO ls_verslag_text SEPARATED BY space.
    PERFORM fout_polis.
    CONTINUE.
  ENDIF.
```

```
IF wa_bron_polis-codemy IS NOT INITIAL.
  CONCATENATE '(' wa_bron_polis-codemy ')'
    INTO lo_codemy_reference.
```

```
ELSE.
  CLEAR lo_codemy_reference.
```

```
ENDIF.
CONCATENATE wa_bron_polis-bron
  wa_bron_polis-polnr
  lo_codemy_reference
  INTO wa_polis_ext_reference SEPARATED BY space.
```

```
PERFORM lees_polis.
```

```
CASE pr_rc.
```

```
  WHEN 0.
```

```
* Polis bestaat
  IF pa_delt = 'X'.
*   alleen bij wijzigen
    PERFORM wijzigen_polis.
  ELSE.
    ADD 1 TO vi_aant_pl_sk.
  ENDIF.
```

```
  WHEN OTHERS.
```

```
* Polis bestaat niet
  PERFORM opvoeren_polis.
```

```
ENDCASE.
```

```
ENDLOOP.
```

```
ENDFORM. " VERWERK_POLIS
```

```
*-----*
```

```
*& Form SPLIT_KLANT
```

```
*-----*
```

```
FORM split_klant
  USING imp_regel.
```

```
SPLIT wa_bron AT cc_split INTO wa_bron_klant-regid
  wa_bron_klant-bron
  wa_bron_klant-relnr
  wa_bron_klant-init
  wa_bron_klant-tussv
  wa_bron_klant-naam
  wa_bron_klant-adres
  wa_bron_klant-huisnr
  wa_bron_klant-hntoev
  wa_bron_klant-postcd
  wa_bron_klant-woonpl
  wa_bron_klant-gebdt
  wa_bron_klant-land
```



```

                                Z_CRM_UPLOAD Example.txt
                                wa_bron_klant-gesl
                                wa_bron_klant-taal
                                wa_bron_klant-tel1
                                wa_bron_klant-tel2.
wa_bron_klant-rege1 = imp_rege1.
APPEND wa_bron_klant TO tb_bron_klant.

ENDFORM.                                " SPLIT_KLANT

*&-----*
*&      Form  SPLIT_INTER
*&-----*
FORM split_inter
  USING imp_rege1.

  SPLIT wa_bron AT cc_split INTO wa_bron_inter-regid
                                wa_bron_inter-bron
                                wa_bron_inter-agsnr
                                wa_bron_inter-naam
                                wa_bron_inter-telnr
                                wa_bron_inter-land
                                wa_bron_inter-taal.

wa_bron_inter-rege1 = imp_rege1.
APPEND wa_bron_inter TO tb_bron_inter.

ENDFORM.                                " SPLIT_INTER

*&-----*
*&      Form  SPLIT_POLIS
*&-----*
FORM split_polis
  USING imp_rege1.

  SPLIT wa_bron AT cc_split INTO wa_bron_polis-regid
                                wa_bron_polis-bron
                                wa_bron_polis-relnr
                                wa_bron_polis-agsnr
                                wa_bron_polis-polnr
                                wa_bron_polis-codemy
                                wa_bron_polis-label
                                wa_bron_polis-prod
                                wa_bron_polis-idat
                                wa_bron_polis-edat
                                wa_bron_polis-stat.

wa_bron_polis-rege1 = imp_rege1.
APPEND wa_bron_polis TO tb_bron_polis.

ENDFORM.                                " SPLIT_POLIS

*&-----*
*&      Form  lees_klant
*&-----*
*      text
*-----*
*      -->IP_RELNR  text
*-----*
FORM lees_klant
  USING ip_relnr.

DATA:
  lo_idnumber                                TYPE but0id-idnumber.

IF ip_relnr(1) CO '0123456789'.
  lo_idnumber = ip_relnr.
ELSE.
  lo_idnumber = ip_relnr+1.
ENDIF.

```

Z\_CRM\_UPLOAD Example.txt

```

SELECT idnumber
       partner
       institute
FROM but0id
INTO wa_best_klant
WHERE idnumber = lo_idnumber
      AND type   = cc_identf_cat_klant.
      IF wa_best_klant-institute = ip_relnr.
      EXIT.
      ENDIF.
ENDSELECT.
pr_rc = sy-subrc.
IF pr_rc <> 0.
  CLEAR wa_best_klant.
ENDIF.
ENDFORM.                "lees_klant

*&-----*
*&      Form  message_text
*&-----*
*      text
*-----*
FORM message_text.
  MESSAGE ID wa_return-id TYPE wa_return-type NUMBER wa_return-number
  INTO ls_verslag_text
  WITH wa_return-message_v1
        wa_return-message_v2
        wa_return-message_v3
        wa_return-message_v4.

ENDFORM.                "message_text

*&-----*
*&      Form  FOUT_KLANT
*&-----*
FORM fout_klant.
  CLEAR tb_verslag.
  tb_verslag-soort   = 'ERR'.
  tb_verslag-regelnr = wa_bron_klant-regelnr.
  tb_verslag-cat     = 'KLANT'.
  tb_verslag-bron    = wa_bron_klant-bron.
  tb_verslag-id      = wa_bron_klant-relnr.
  tb_verslag-text    = ls_verslag_text.
  APPEND tb_verslag.

  vi_aant_kl_ft = vi_aant_kl_ft + 1.

ENDFORM.                " FOUT_KLANT

*&-----*
*&      Form  OPVOEREN_KLANT
*&-----*
FORM opvoeren_klant.

* Aanleggen klant

REFRESH: tb_telef_data,
         tb_return.

CLEAR: st_centra_dt,
       st_centra_ps,
       st_adres_dt,
       st_telef_dt,
       st_partn_gr,
       st_bp_roles,

```

Z\_CRM\_UPLOAD Example.txt

```

st_ident_fk,
st_ident_fc.

st_partn_gr-partn_cat      = '1'.
st_partn_gr-partn_grp     = cc_kl_part_gr.
st_centr_dt-partnerlanguageiso = wa_bron_klant-taal.
IF syst-subrc <> 0.
  PERFORM fout_klant.
  EXIT.
ENDIF.
st_centr_ps-initials      = wa_bron_klant-init.

IF wa_bron_klant-tussv IS NOT INITIAL.
  PERFORM get_prefix1
    USING wa_bron_klant-tussv
          st_centr_ps-prefix1.
  IF st_centr_ps-prefix1 IS INITIAL.
    CONCATENATE 'Tussenvoegsel niet toegestaan:' wa_bron_klant-tussv
      INTO ls_verslag_text SEPARATED BY space.
    PERFORM fout_klant.
    EXIT.
  ENDIF.
ENDIF.

st_centr_ps-lastname      = wa_bron_klant-naam.
st_centr_ps-birthdate    = wa_bron_klant-gebd.

CASE wa_bron_klant-gesl.
  WHEN '1'.
    st_centr_ps-sex        = wa_bron_klant-gesl.
    st_centr_dt-title_key = '0001'.      "Mevrouw
  WHEN '2'.
    st_centr_ps-sex        = wa_bron_klant-gesl.
    st_centr_dt-title_key = '0002'.      "De heer
  WHEN OTHERS.
    st_centr_dt-title_key = '0005'.
ENDCASE.
st_adres_dt-street        = wa_bron_klant-adres.
st_adres_dt-house_no     = wa_bron_klant-huisnr.
st_adres_dt-house_no2    = wa_bron_klant-hntoev.
st_adres_dt-postl_cod1   = wa_bron_klant-postcd.
st_adres_dt-city         = wa_bron_klant-woonpl.
st_adres_dt-country      = wa_bron_klant-land.

IF NOT wa_bron_klant-tel1 IS INITIAL.
  ADD 1 TO st_telef_dt-consnumber.
  st_telef_dt-telephone = wa_bron_klant-tel1.
  APPEND st_telef_dt TO tb_telef_data.
ENDIF.

IF NOT wa_bron_klant-tel2 IS INITIAL.
  ADD 1 TO st_telef_dt-consnumber.
  st_telef_dt-r_3_user = 3.
  st_telef_dt-telephone = wa_bron_klant-tel2.
  APPEND st_telef_dt TO tb_telef_data.
ENDIF.

CALL FUNCTION 'BAPI_BUPA_CREATE_FROM_DATA'
  EXPORTING
    partnercategory      = st_partn_gr-partn_cat
    partnergroup         = st_partn_gr-partn_grp
    centraldata          = st_centr_dt
    centraldataperson    = st_centr_ps
    addressdata          = st_adres_dt
  IMPORTING
    businesspartner      = tp_zp_nr
  TABLES
    telefodata           = tb_telef_data

```

```

                                Z_CRM_UPLOAD Example.txt
return                                = tb_return.

LOOP AT tb_return INTO wa_return WHERE type = 'E'.
  PERFORM message_text.
  PERFORM fout_klant.
ENDLOOP.
IF sy-subrc <> 0.                                "GEEN Fout gevonden
  st_ident_fk-identificationcategory = cc_identf_cat_klant.

  IF wa_bron_klant-relnr(1) CO '0123456789'.
    st_ident_fk-identificationnumber = wa_bron_klant-relnr.
  ELSE.
    st_ident_fk-identificationnumber = wa_bron_klant-relnr+1.
  ENDIF.

  st_ident_fc-idinstitute = wa_bron_klant-relnr.

CALL FUNCTION 'BUPA_IDENTIFICATION_ADD'
  EXPORTING
    iv_partner                = tp_zp_nr
    iv_identificationcategory = st_ident_fk-identificationcategory
    iv_identificationnumber   = st_ident_fk-identificationnumber
    is_identification         = st_ident_fc
  TABLES
    et_return                 = tb_return.

LOOP AT tb_return INTO wa_return WHERE type = 'E'.
  PERFORM message_text.
  PERFORM fout_klant.
ENDLOOP.

IF sy-subrc <> 0.                                "GEEN Fout gevonden
  st_partn_gr-bpartner        = tp_zp_nr.
  st_bp_roles-partnerrole     = cc_part_rol_k.

CALL FUNCTION 'BAPI_BUPA_ROLE_ADD_2'
  EXPORTING
    businesspartner          = st_partn_gr-bpartner
    businesspartnerrole      = st_bp_roles-partnerrole
  TABLES
    return                   = tb_return.

LOOP AT tb_return INTO wa_return WHERE type = 'E'.
  PERFORM message_text.
  PERFORM fout_klant.
ENDLOOP.

IF sy-subrc <> 0.                                "GEEN Fout gevonden
  CONCATENATE 'Nummer:' tp_zp_nr
              INTO ls_verslag_text SEPARATED BY space.
  PERFORM goed_klant USING 'OPV'.
  CALL FUNCTION 'BAPI_TRANSACTION_COMMIT'.
ENDIF.

ENDIF.

ENDIF.

ENDFORM.                                " OPVOEREN_KLANT

*&-----*
*& Form get_prefix1
*&-----*
* text
*-----*
* -->IMP_TUSSV text
* -->IMP_PREFIX1 text
*-----*

```

```

FORM get_prefix1
      USING imp_tussv
      imp_prefix1.
DATA:
  lo_tussv      TYPE tsad4-prefix_txt.

lo_tussv = imp_tussv.
TRANSLATE lo_tussv TO UPPER CASE.
READ TABLE tb_tsad4 WITH KEY prefix_txt = lo_tussv.
IF syst-subrc = 0.
  imp_prefix1 = tb_tsad4-prefix_key.
ELSE.
  CLEAR imp_prefix1.
ENDIF.
ENDFORM.                  "get_prefix1

```

```

*&-----*
*&      Form  GOED_KLANT
*&-----*

```

```

FORM goed_klant
      USING imp_org.
CLEAR tb_verslag.
tb_verslag-soort   = 'OK '.
tb_verslag-regelnr = wa_bron_klant-regel.
tb_verslag-cat     = 'KLANT'.
tb_verslag-origin  = imp_org.
tb_verslag-bron    = wa_bron_klant-bron.
tb_verslag-id      = wa_bron_klant-relnr.
tb_verslag-text    = ls_verslag_text.
APPEND tb_verslag.

vi_aant_kl_gd = vi_aant_kl_gd + 1.
ENDFORM.                  " GOED_KLANT

```

```

*&-----*
*&      Form  WIJZIGEN_KLANT
*&-----*

```

```

FORM wijzigen_klant.
*
*  Wijzigen klant
*
DATA:
  lo_commit      TYPE c.

CLEAR lo_commit.

REFRESH: tb_telef_data,
         tb_telef_data_x,
         tb_return.

CLEAR: st_centr_dt,
       st_centr_dt_x,
       st_centr_ps,
       st_centr_ps_x,
       st_adres_dt,
       st_adres_dt_x,
       st_telef_dt,
       st_telef_dt_x.

* Gegevens gewijzigd ??
PERFORM prepare_klant_wijz.

IF st_centr_dt_x  IS NOT INITIAL OR
   st_centr_ps_x  IS NOT INITIAL.
  CALL FUNCTION 'BAPI_BUPA_CENTRAL_CHANGE'
    EXPORTING
      businesspartner      = wa_best_klant-partner

```

```

                                Z_CRM_UPLOAD Example.txt
centraldata                    = st_centr_dt
centraldataperson              = st_centr_ps
centraldata_x                  = st_centr_dt_x
centraldataperson_x            = st_centr_ps_x
TABLES
  return                        = tb_return.

LOOP AT tb_return INTO wa_return WHERE type = 'E'.
  PERFORM message_text.
  PERFORM fout_klant.
ENDLOOP.
IF syst-subrc <> 0.
  lo_commit = 'X'.
  CONCATENATE 'Nummer:' wa_best_klant-partner '(Central)'
    INTO ls_verslag_text SEPARATED BY space.
  PERFORM goed_klant USING 'WIJZ'.
ENDIF.
ENDIF.
IF st_adres_dt_x IS NOT INITIAL OR
  tb_telef_data_x[] IS NOT INITIAL.
CALL FUNCTION 'BAPI_BUPA_ADDRESS_CHANGE'
  EXPORTING
    businesspartner = wa_best_klant-partner
    addressdata     = st_adres_dt
    addressdata_x   = st_adres_dt_x
  TABLES
    bapiadtel       = tb_telef_data
    bapiadtel_x     = tb_telef_data_x
    return          = tb_return.
LOOP AT tb_return INTO wa_return WHERE type = 'E'.
  PERFORM message_text.
  PERFORM fout_klant.
ENDLOOP.
IF syst-subrc <> 0.
  lo_commit = 'X'.
  CONCATENATE 'Nummer:' wa_best_klant-partner '(Adres)'
    INTO ls_verslag_text SEPARATED BY space.
  PERFORM goed_klant USING 'WIJZ'.
ENDIF.
ENDIF.
IF lo_commit = 'X'.
  CALL FUNCTION 'BAPI_TRANSACTION_COMMIT'.
ENDIF.
ENDFORM.                                " WIJZIGEN_KLANT

*&-----*
*&      Form  prepare_klant_wijz
*&-----*
*      text
*-----*
*      -->IMP_RC      text
*-----*
FORM prepare_klant_wijz.
CALL FUNCTION 'BAPI_BUPA_CENTRAL_GETDETAIL'
  EXPORTING
    businesspartner      = wa_best_klant-partner
*  VALID_DATE            = SY-DATLO
  IMPORTING
    centraldata          = st_centr_dt
    centraldataperson    = st_centr_ps
  TABLES
    telefondatanonaddress = tb_telef_data
    return                = tb_return.

LOOP AT tb_return INTO wa_return WHERE type = 'E'.
  PERFORM message_text.
  PERFORM fout_klant.

```

```

ENDLOOP.
IF syst-subrc = 0.
  EXIT.
ENDIF.
CALL FUNCTION 'BAPI_BUPA_ADDRESS_GETDETAIL'
  EXPORTING
    businesspartner      = wa_best_klant-partner
    ADDRESSGUID          =
    VALID_DATE           = SY-DATLO
    RESET_BUFFER         =
  IMPORTING
    addressdata          = st_adres_dt
  TABLES
    return               = tb_return.
.
* Tussenvoegsel
DATA:
  lo_prefix_key          TYPE tsad4-prefix_key.

IF wa_bron_klant-tussv IS NOT INITIAL.
  PERFORM get_prefix1
    USING wa_bron_klant-tussv
          lo_prefix_key .
  IF lo_prefix_key IS INITIAL.
    CONCATENATE 'Tussenvoegsel niet toegestaan:' wa_bron_klant-tussv
      INTO ls_verslag_text SEPARATED BY space.
    PERFORM fout_klant.
    EXIT.
  ENDIF.
ENDIF.
IF st_centr_ps-prefix1 NE lo_prefix_key.
  st_centr_ps-prefix1 = lo_prefix_key.
  st_centr_ps_x-prefix1 = 'X'.
ENDIF.
* Initialen
IF st_centr_ps-initials NE wa_bron_klant-init.
  st_centr_ps-initials = wa_bron_klant-init.
  st_centr_ps_x-initials = 'X'.
ENDIF.
* Naam
IF st_centr_ps-lastname NE wa_bron_klant-naam.
  st_centr_ps-lastname = wa_bron_klant-naam.
  st_centr_ps_x-lastname = 'X'.
ENDIF.
* Geboortedatum
IF st_centr_ps-birthdate NE wa_bron_klant-gebd.
  st_centr_ps-birthdate = wa_bron_klant-gebd.
  st_centr_ps_x-birthdate = 'X'.
ENDIF.
* Geslacht
IF st_centr_ps-sex NE wa_bron_klant-gesl.
  st_centr_ps_x-sex = 'X'.
  st_centr_dt_x-title_key = 'X'.
  CASE wa_bron_klant-gesl.
    WHEN '1'.
      st_centr_ps-sex = wa_bron_klant-gesl.
      st_centr_dt-title_key = '0001'. "Mevrouw
    WHEN '2'.
      st_centr_ps-sex = wa_bron_klant-gesl.
      st_centr_dt-title_key = '0002'. "De heer
    WHEN OTHERS.
      st_centr_dt-title_key = '0005'.
  ENDCASE.
ENDIF.
* Taal
IF st_centr_dt-partnerlanguageiso NE wa_bron_klant-taal.
  st_centr_dt-partnerlanguageiso = wa_bron_klant-taal.

```

Z\_CRM\_UPLOAD Example.txt

```

st_centr_dt_x-partnerlanguageiso = 'X'.
ENDIF.
* Straat
IF st_adres_dt-street NE wa_bron_klant-adres.
  st_adres_dt-street = wa_bron_klant-adres.
  st_adres_dt_x-street = 'X'.
ENDIF.
* Huisnummer
IF st_adres_dt-house_no NE wa_bron_klant-huisnr.
  st_adres_dt-house_no = wa_bron_klant-huisnr.
  st_adres_dt_x-house_no = 'X'.
ENDIF.
* Huisnummer toev
IF st_adres_dt-house_no2 NE wa_bron_klant-hntoev.
  st_adres_dt-house_no2 = wa_bron_klant-hntoev.
  st_adres_dt_x-house_no2 = 'X'.
ENDIF.
* Postcode
IF st_adres_dt-postl_cod1 NE wa_bron_klant-postcd.
  st_adres_dt-postl_cod1 = wa_bron_klant-postcd.
  st_adres_dt_x-postl_cod1 = 'X'.
ENDIF.
* Plaats
IF st_adres_dt-city NE wa_bron_klant-woonpl.
  st_adres_dt-city = wa_bron_klant-woonpl.
  st_adres_dt_x-city = 'X'.
ENDIF.
* Land
IF st_adres_dt-country NE wa_bron_klant-land.
  st_adres_dt-country = wa_bron_klant-land.
  st_adres_dt_x-country = 'X'.
ENDIF.
* Telefoongegevens
IF tb_telef_data[] IS INITIAL.
  IF NOT wa_bron_klant-tel1 IS INITIAL.
    ADD 1 TO st_telef_dt-consnumber.
    st_telef_dt-telephone = wa_bron_klant-tel1.
    APPEND st_telef_dt TO tb_telef_data.
    st_telef_dt_x-telephone = 'X'.
    st_telef_dt_x-updateflag = 'I'.
    APPEND st_telef_dt_x TO tb_telef_data_x.

  ENDIF.
  IF NOT wa_bron_klant-tel2 IS INITIAL.
    ADD 1 TO st_telef_dt-consnumber.
    st_telef_dt-r_3_user = 3.
    st_telef_dt-telephone = wa_bron_klant-tel2.
    APPEND st_telef_dt TO tb_telef_data.
    st_telef_dt_x-telephone = 'X'.
    st_telef_dt_x-updateflag = 'I'.
    APPEND st_telef_dt_x TO tb_telef_data_x.
  ENDIF.
ELSE.
  REFRESH tb_telef_data.
ENDIF.
ENDFORM.                                "prepare_klant_wijz

```

```

*&-----*
*&      Form lees_inter
*&-----*
*      text
*-----*
*      -->IP_AGSNR  text
*-----*
FORM lees_inter
  USING ip_agsnr.

```



```
DATA:
  lo_idnumber.
```

```
SELECT SINGLE
  partner
  idnumber
  FROM but0id
  INTO wa_best_inter
  WHERE idnumber = ip_agssnr
  AND type = cc_identf_cat_inter.
pr_rc = sy-subrc.
IF pr_rc <> 0.
  CLEAR wa_best_inter.
ENDIF.
```

```
ENDFORM.                "lees_inter
```

```
*&-----*
*&      Form  FOUT_INTER
*&-----*
```

```
FORM fout_inter.
  CLEAR tb_verslag.
  tb_verslag-soort = 'ERR'.
  tb_verslag-regelnr = wa_bron_inter-regel.
  tb_verslag-cat = 'INTER'.
  tb_verslag-bron = wa_bron_inter-bron.
  tb_verslag-id = wa_bron_inter-agssnr.
  tb_verslag-text = ls_verslag_text.
  APPEND tb_verslag.
```

```
vi_aant_im_ft = vi_aant_im_ft + 1.
```

```
ENDFORM.                " FOUT_INTER
```

```
*&-----*
*&      Form  OPVOEREN_INTER
*&-----*
```

```
FORM opvoeren_inter.
```

```
*
* Aanleggen intermediair
```

```
REFRESH: tb_telef_data,
          tb_return.
```

```
CLEAR: st_centr_dt,
        st_adres_dt,
        st_centr_or,
        st_telef_dt,
        st_partn_gr,
        st_ident_fk,
        st_ident_fc,
        st_bp_roles.
```

```
st_partn_gr-partn_cat = '2'.
st_partn_gr-partn_grp = cc_in_part_gr.
st_centr_dt-partnerlanguageiso = wa_bron_inter-taal.
st_centr_dt-partnerexternal = wa_bron_inter-agssnr.
st_centr_or-name1 = wa_bron_inter-naam.
st_adres_dt-country = wa_bron_inter-land.
IF st_adres_dt-country IS INITIAL.
  st_adres_dt-country = 'NL'.
ENDIF.
IF NOT wa_bron_inter-telnr IS INITIAL.
  st_telef_dt-telephone = wa_bron_inter-telnr.
  APPEND st_telef_dt TO tb_telef_data.
ENDIF.
```

Z\_CRM\_UPLOAD Example.txt

```

CALL FUNCTION 'BAPI_BUPA_CREATE_FROM_DATA'
  EXPORTING
    businesspartnerextern = st_partn_gr-bpartner
    partnercategory       = st_partn_gr-partn_cat
    partnergroup          = st_partn_gr-partn_grp
    centraldata           = st_cent_r_dt
    centraldataorganization = st_cent_r_or
    addressdata           = st_adres_dt
  IMPORTING
    businesspartner       = tp_im_nr
  TABLES
    telefodata            = tb_telef_data
    return                = tb_return.

LOOP AT tb_return INTO wa_return WHERE type = 'E'.
  PERFORM message_text.
  PERFORM fout_inter.
ENDLOOP.

IF sy-subrc <> 0.                                "GEEN Fout gevonden

  st_ident_fk-identificationcategory = cc_identf_cat_inter.
  st_ident_fk-identificationnumber   = wa_bron_inter-agsnr.
*  st_ident_fc-idinstitute            = wa_bron_inter-agsnr.

  CALL FUNCTION 'BUPA_IDENTIFICATION_ADD'
    EXPORTING
      iv_partner                = tp_im_nr
      iv_identificationcategory = st_ident_fk-identificationcategory
      iv_identificationnumber   = st_ident_fk-identificationnumber
      is_identification         = st_ident_fc
    TABLES
      et_return                 = tb_return.

  LOOP AT tb_return INTO wa_return WHERE type = 'E'.
    PERFORM message_text.
    PERFORM fout_inter.
  ENDLOOP.

  IF sy-subrc <> 0.                                "Fout gevonden
    st_partn_gr-bpartner = tp_im_nr.
    st_bp_roles-partnerrole = cc_part_rol_i.

    CALL FUNCTION 'BAPI_BUPA_ROLE_ADD_2'
      EXPORTING
        businesspartner = st_partn_gr-bpartner
        businesspartnerrole = st_bp_roles-partnerrole
      TABLES
        return = tb_return.

    LOOP AT tb_return INTO wa_return WHERE type = 'E'.
      PERFORM message_text.
      PERFORM fout_inter.
    ENDLOOP.

    IF sy-subrc <> 0.                                "GEEN Fout gevonden
      CONCATENATE 'Nummer:' tp_im_nr INTO ls_verslag_text.
      PERFORM goed_inter USING 'OPV'.
      CALL FUNCTION 'BAPI_TRANSACTION_COMMIT'.
    ENDIF.
  ENDIF.
ENDIF.
ENDFORM.                                " OPVOEREN_INTER

*&-----*
*&      Form      GOED_INTER
*&-----*

```

```

FORM goed_inter
  USING imp_org.
  CLEAR tb_verslag.
  tb_verslag-soort = 'OK '.
  tb_verslag-regelnr = wa_bron_inter-regel.
  tb_verslag-cat = 'INTER'.
  tb_verslag-origin = imp_org.
  tb_verslag-bron = wa_bron_inter-bron.
  tb_verslag-id = wa_bron_inter-agsnr.
  tb_verslag-text = ls_verslag_text.
  APPEND tb_verslag.

  vi_aant_im_gd = vi_aant_im_gd + 1.

ENDFORM.          " GOED_INTER

*&-----*
*&      Form WIJZIGEN_INTER
*&-----*
FORM wijzigen_inter.
*
* Wijzigen intermediair
*
  DATA:
    lo_commit          TYPE c.

  CLEAR lo_commit.

  REFRESH: tb_telef_data,
           tb_telef_data_x,
           tb_return.

  CLEAR: st_centr_dt,
         st_centr_dt_x,
         st_centr_or,
         st_centr_or_x,
         st_centr_ps,
         st_centr_ps_x,
         st_adres_dt,
         st_adres_dt_x,
         st_telef_dt,
         st_telef_dt_x.

* Gegevens gewijzigd ??
  PERFORM prepare_inter_wijz.

  IF st_centr_dt_x IS NOT INITIAL OR
     st_centr_or_x IS NOT INITIAL.
    CALL FUNCTION 'BAPI_BUPA_CENTRAL_CHANGE'
      EXPORTING
        businesspartner = wa_best_inter-partner
        centraldata      = st_centr_dt
        centraldataorganization = st_centr_or
        centraldata_x    = st_centr_dt_x
        centraldataorganization_x = st_centr_or_x
      TABLES
        return          = tb_return.

    LOOP AT tb_return INTO wa_return WHERE type = 'E'.
      PERFORM message_text.
      PERFORM fout_inter.
    ENDLOOP.
  IF syst-subrc <> 0.
    lo_commit = 'X'.
    CONCATENATE 'Nummer:' wa_best_inter-partner '(Central)'
      INTO ls_verslag_text SEPARATED BY space.
    PERFORM goed_inter USING 'WIJZ'.
  ENDIF.

```

Z\_CRM\_UPLOAD Example.txt

```

ENDIF.
IF st_adres_dt_x      IS NOT INITIAL OR
   tb_telef_data_x[] IS NOT INITIAL.
  CALL FUNCTION 'BAPI_BUPA_ADDRESS_CHANGE'
    EXPORTING
      businesspartner = wa_best_inter-partner
      addressdata      = st_adres_dt
      addressdata_x    = st_adres_dt_x
    TABLES
      bapiadtel        = tb_telef_data
      bapiadtel_x      = tb_telef_data_x
      return           = tb_return.
  LOOP AT tb_return INTO wa_return WHERE type = 'E'.
    PERFORM message_text.
    PERFORM fout_inter.
  ENDLOOP.
  IF syst-subrc <> 0.
    lo_commit = 'X'.
    CONCATENATE 'Nummer:' wa_best_inter-partner '(Adres)'
      INTO ls_verslag_text SEPARATED BY space.
    PERFORM goed_inter USING 'WIJZ'.
  ENDIF.
ENDIF.
IF lo_commit = 'X'.
  CALL FUNCTION 'BAPI_TRANSACTION_COMMIT'.
ENDIF.

ENDFORM.                " WIJZIGEN_INTER
*&-----*
*&      Form prepare_inter_wijz
*&-----*
*      text
*-----*
*      -->IMP_RC      text
*-----*
FORM prepare_inter_wijz.
  CALL FUNCTION 'BAPI_BUPA_CENTRAL_GETDETAIL'
    EXPORTING
      businesspartner      = wa_best_inter-partner
*   VALID_DATE             = SY-DATLO
    IMPORTING
      centraldata          = st_cent_rdt
      centraldataorganization = st_cent_ror
    TABLES
      telefondatanonaddress = tb_telef_data
      return                = tb_return.

  LOOP AT tb_return INTO wa_return WHERE type = 'E'.
    PERFORM message_text.
    PERFORM fout_klant.
  ENDLOOP.
  IF syst-subrc = 0.
    EXIT.
  ENDIF.
  CALL FUNCTION 'BAPI_BUPA_ADDRESS_GETDETAIL'
    EXPORTING
      businesspartner      = wa_best_inter-partner
*   ADDRESSGUID            =
*   VALID_DATE             = SY-DATLO
*   RESET_BUFFER           =
    IMPORTING
      addressdata          = st_adres_dt
    TABLES
      return                = tb_return.

* Naam
IF st_cent_ror-name1 NE wa_bron_inter-naam.
  st_cent_ror-name1 = wa_bron_inter-naam.

```

```

                                Z_CRM_UPLOAD Example.txt
    st_centr_or_x-name1 = 'X'.
ENDIF.
* Taal
IF st_centr_dt-partnerlanguageiso NE wa_bron_inter-taal.
    st_centr_dt-partnerlanguageiso = wa_bron_inter-taal.
    st_centr_dt_x-partnerlanguageiso = 'X'.
ENDIF.
* Land
IF wa_bron_inter-land IS INITIAL.
    wa_bron_inter-land = 'NL'.
ENDIF.
IF st_adres_dt-country NE wa_bron_inter-land.
    st_adres_dt-country = wa_bron_inter-land.
    st_adres_dt_x-country = 'X'.
ENDIF.
* Telefoongegevens
IF tb_telef_data[] IS INITIAL.
    IF NOT wa_bron_inter-telnr IS INITIAL.
        st_telef_dt-telephone = wa_bron_inter-telnr.
        APPEND st_telef_dt TO tb_telef_data.
        st_telef_dt_x-telephone = 'X'.
        st_telef_dt_x-updateflag = 'I'.
        APPEND st_telef_dt_x TO tb_telef_data_x.
    ENDIF.
ELSE.
    READ TABLE tb_telef_data INDEX 1 INTO st_telef_dt.
    st_telef_dt-telephone = wa_bron_inter-telnr.
    MODIFY tb_telef_data FROM st_telef_dt INDEX 1 .
    st_telef_dt_x-telephone = 'X'.
    st_telef_dt_x-updateflag = 'U'.
    APPEND st_telef_dt_x TO tb_telef_data_x.
ENDIF.
ENDFORM.                                "prepare_inter_wijz

*&-----*
*&      Form lees_polis
*&-----*
*      text
*-----*
FORM lees_polis.

SELECT SINGLE
    object_id
    po_number_uc
    guid
    INTO wa_best_polis
    FROM crmv_purchase_no
    WHERE po_number_uc = wa_polis_ext_reference.
pr_rc = syst-subrc.
IF pr_rc <> 0.
    CLEAR wa_best_polis.
ENDIF.

ENDFORM.                                "lees_polis

*&-----*
*&      Form FOUT_POLIS
*&-----*
FORM fout_polis.

CLEAR tb_verslag.
tb_verslag-soort = 'ERR'.
tb_verslag-regelnr = wa_bron_polis-regel.
tb_verslag-cat = 'POLIS'.
tb_verslag-bron = wa_bron_polis-bron.
tb_verslag-id = wa_bron_polis-polnr.
tb_verslag-text = ls_verslag_text.
APPEND tb_verslag.

```

Z\_CRM\_UPLOAD Example.txt

vi\_aant\_pl\_ft = vi\_aant\_pl\_ft + 1.

ENDFORM.

" FOUT\_POLIS

```

*&-----*
*&      Form  OPVOEREN_POLIS
*&-----*
FORM opvoeren_polis.
*
* Aanleggen polis
*
PERFORM lees_klant
      USING wa_bron_polis-relnr.
IF pr_rc <> 0.
  CONCATENATE 'Klant niet gevonden - relatienummer:' wa_bron_polis-relnr
    INTO ls_verslag_text SEPARATED BY space.
  PERFORM fout_polis.
ELSE.
  PERFORM lees_inter
    USING wa_bron_polis-agsnr.
ENDIF.

IF pr_rc <> 0.
  CONCATENATE 'Intermediair niet gevonden -nummer:' wa_bron_polis-agsnr
    INTO ls_verslag_text SEPARATED BY space.
  PERFORM fout_polis.
ELSE.

  REFRESH: tb_header_ins,
            tb_return,
            tb_sales,
            tb_partner,
            tb_appointm,
            tb_status,
            tb_created_process,
            tb_saved_process,
            tb_inp_fields,
            lt_sls_input_fields.

  CLEAR: wa_header_ins,
          wa_sales,
          wa_partner,
          wa_appointm,
          wa_status,
          wa_inp_fields,
          ls_sls_input_fields.

* Vullen polis header gegevens
  CALL FUNCTION 'CRM_GUID_CREATE'
    IMPORTING
      ev_guid = tp_guid.

  wa_header_ins-guid          = tp_guid.
  wa_header_ins-process_type = cc_procc_type.
  wa_header_ins-description  = wa_bron_polis-prod.

  APPEND wa_header_ins TO tb_header_ins.

* Vullen input fields voor header gegevens
  CLEAR ls_sls_input_fields.
  ls_sls_input_fields-ref_guid = tp_guid.
  ls_sls_input_fields-ref_kind = 'A'.
  ls_sls_input_fields-objectname = 'ORDERADM_H'.

  ls_sls_input_fields-fieldname = 'PROCESS_TYPE'.
  INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
  ls_sls_input_fields-fieldname = 'DESCRIPTION'.

```

```
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```

```
* Vullen polis sales gegevens
```

```
wa_sales-ref_guid      = tp_guid.
wa_sales-ref_kind      = 'A'.
```

```
wa_sales-po_number_sold = wa_polis_ext_reference.
```

```
wa_sales-po_number_ship = wa_bron_polis-label.
```

```
APPEND wa_sales TO tb_sales.
```

```
* Vullen input fields voor sales gegevens
```

```
CLEAR ls_sls_input_fields.
```

```
ls_sls_input_fields-ref_guid = tp_guid.
```

```
ls_sls_input_fields-ref_kind = 'A'.
```

```
ls_sls_input_fields-objectname = 'SALES'.
```

```
ls_sls_input_fields-fieldname = 'PO_NUMBER_SOLD'.
```

```
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```

```
ls_sls_input_fields-fieldname = 'PO_NUMBER_SHIP'.
```

```
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```

```
* Vullen polis partner gegevens
```

```
* Eerst verzekeringsnemer toevoegen
```

```
wa_partner-ref_guid      = tp_guid.
```

```
* wa_partner-ref_handle    = '0000000001'.
```

```
wa_partner-ref_kind      = 'A'.
```

```
wa_partner-ref_partner_handle = '0001'.
```

```
* wa_partner-ref_partner_fct = cc_partnfunc_v.
```

```
* wa_partner-ref_partner_no  = wa_goed_klant-klantnr.
```

```
* wa_partner-ref_no_type     = 'BP'.
```

```
* wa_partner-ref_display_type = 'BP'.
```

```
wa_partner-partner_fct    = cc_partnfunc_v.
```

```
wa_partner-partner_no     = wa_best_klant-partner.
```

```
wa_partner-no_type       = 'BP'.
```

```
wa_partner-display_type  = 'BP'.
```

```
* wa_partner-mainpartner    = 'X'.
```

```
wa_partner-kind_of_entry = 'C'.
```

```
APPEND wa_partner TO tb_partner.
```

```
* Daarna intermediar toevoegen
```

```
wa_partner-ref_partner_handle = '0002'.
```

```
wa_partner-partner_fct      = cc_partnfunc_i.
```

```
wa_partner-partner_no      = wa_best_inter-partner.
```

```
APPEND wa_partner TO tb_partner.
```

```
* Vullen input fields voor partner gegevens
```

```
CLEAR ls_sls_input_fields.
```

```
ls_sls_input_fields-ref_guid = tp_guid.
```

```
ls_sls_input_fields-ref_kind = 'A'.
```

```
ls_sls_input_fields-objectname = 'PARTNER'.
```

```
ls_sls_input_fields-logical_key = '0001'.
```

```
ls_sls_input_fields-fieldname = 'PARTNER_FCT'.
```

```
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```

```
ls_sls_input_fields-fieldname = 'PARTNER_NO'.
```

```
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```

```
ls_sls_input_fields-fieldname = 'NO_TYPE'.
```

```
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```

```
ls_sls_input_fields-fieldname = 'DISPLAY_TYPE'.
```

```
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```

```
* ls_sls_input_fields-fieldname = 'MAINPARTNER'.
```

Z\_CRM\_UPLOAD Example.txt

```
* INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.  
ls_sls_input_fields-fieldname = 'KIND_OF_ENTRY'.  
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.  
  
ls_sls_input_fields-logical_key = '0002'.  
ls_sls_input_fields-fieldname = 'PARTNER_FCT'.  
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.  
ls_sls_input_fields-fieldname = 'PARTNER_NO'.  
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.  
ls_sls_input_fields-fieldname = 'NO_TYPE'.  
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.  
ls_sls_input_fields-fieldname = 'DISPLAY_TYPE'.  
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.  
* ls_sls_input_fields-fieldname = 'MAINPARTNER'.  
* INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.  
ls_sls_input_fields-fieldname = 'KIND_OF_ENTRY'.  
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```

\* Vullen polis geldigheids duur gegeven

```
CONVERT DATE wa_bron_polis-idat  
INTO TIME STAMP lv_tstamp TIME ZONE sy-zonlo.
```

```
wa_appointm-ref_guid      = tp_guid.  
wa_appointm-ref_kind     = 'A'.  
wa_appointm-appt_type    = 'VALIDFROM'.  
wa_appointm-timestamp_from = lv_tstamp.  
wa_appointm-timezone_from = sy-zonlo.  
wa_appointm-timezone_to  = sy-zonlo.  
* wa_appointm-rule_name   = 'TODAY'.                "JB20110503"  
wa_appointm-duration     = 0.  
wa_appointm-mode        = 'A'.  
wa_appointm-timestamp_to = lv_tstamp.  
wa_appointm-date_from   = wa_bron_polis-idat.
```

```
APPEND wa_appointm TO tb_appointm.
```

```
CONVERT DATE wa_bron_polis-edat  
INTO TIME STAMP lv_tstamp TIME ZONE sy-zonlo.
```

```
wa_appointm-ref_guid      = tp_guid.  
wa_appointm-ref_kind     = 'A'.  
wa_appointm-appt_type    = 'VALIDTO'.  
wa_appointm-timezone_from = sy-zonlo.  
wa_appointm-timezone_to  = sy-zonlo.  
wa_appointm-timestamp_from = lv_tstamp.  
wa_appointm-date_from    = wa_bron_polis-edat.  
wa_appointm-duration     = 0.  
wa_appointm-mode        = 'A'.
```

```
APPEND wa_appointm TO tb_appointm.
```

\* Vullen input fields voor geldigheids duur

```
ls_sls_input_fields-ref_guid = tp_guid.  
ls_sls_input_fields-ref_kind = 'A'.  
ls_sls_input_fields-objectname = 'APPOINTMENT'.  
ls_sls_input_fields-logical_key = 'VALIDFROM'.  
"JB20110305"
```

```
ls_sls_input_fields-fieldname = 'DOMINANT'.  
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.  
ls_sls_input_fields-fieldname = 'DURATION'.  
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.  
ls_sls_input_fields-fieldname = 'RULE_NAME'.  
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```



Z\_CRM\_UPLOAD Example.txt

```
* ls_sls_input_fields-fieldname = 'SHOWLOCAL'.
* INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
ls_sls_input_fields-fieldname = 'TIMESTAMP_FROM'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
ls_sls_input_fields-fieldname = 'TIMEZONE_FROM'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
ls_sls_input_fields-fieldname = 'TIME_UNIT'.
ls_sls_input_fields-ref_guid = tp_guid.
ls_sls_input_fields-ref_kind = 'A'.
ls_sls_input_fields-objectname = 'APPOINTMENT'.
ls_sls_input_fields-logical_key = 'VALIDTO'.
"JB20110305"
```

```
ls_sls_input_fields-fieldname = 'DOMINANT'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
ls_sls_input_fields-fieldname = 'DURATION'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
ls_sls_input_fields-fieldname = 'RULE_NAME'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
* ls_sls_input_fields-fieldname = 'SHOWLOCAL'.
* INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.

ls_sls_input_fields-fieldname = 'TIMESTAMP_FROM'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
ls_sls_input_fields-fieldname = 'TIMEZONE_FROM'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
ls_sls_input_fields-fieldname = 'TIME_UNIT'.
```

\* Vullen polis status gegevens

```
wa_status-ref_guid = tp_guid.
READ TABLE tb_tj30t
  WITH KEY txt04 = wa_bron_polis-stat.

wa_status-ref_kind = 'A'.
wa_status-status = tb_tj30t-estat.
wa_status-user_stat_proc = cc_profile_zasprol.
wa_status-activate = 'X'.

APPEND wa_status TO tb_status.
```

\* Vullen input fields voor status

```
CLEAR ls_sls_input_fields.
ls_sls_input_fields-ref_guid = tp_guid.
ls_sls_input_fields-ref_kind = 'A'.
ls_sls_input_fields-objectname = 'STATUS'.

ls_sls_input_fields-fieldname = 'STATUS'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
ls_sls_input_fields-fieldname = 'USER_STAT_PROC'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
ls_sls_input_fields-fieldname = 'ACTIVATE'.
INSERT ls_sls_input_fields INTO TABLE lt_sls_input_fields.
```

CALL FUNCTION 'BAPI\_SLSTRANSACT\_CREATEMULTI'

```
TABLES
  header          = tb_header_ins
*   ITEM          =
  return         = tb_return
  sales          = tb_sales
  partner        = tb_partner
  appointment    = tb_appointm
  status         = tb_status
  input_fields   = lt_sls_input_fields
  created_process = tb_created_process
  saved_process  = tb_saved_process.
```

Z\_CRM\_UPLOAD Example.txt

```

LOOP AT tb_return INTO wa_return WHERE type = 'E'.
  PERFORM message_text.
  PERFORM fout_polis.
ENDLOOP.

```

```

IF sy-subrc <> 0.                                "GEEN Fout gevonden
  READ TABLE tb_saved_process INDEX 1.
  CONCATENATE 'Nummer:' tb_saved_process-object_id
              INTO ls_verslag_text SEPARATED BY space.
  PERFORM goed_polis USING 'OPV'.
  CALL FUNCTION 'BAPI_TRANSACTION_COMMIT'.
ENDIF.
ENDIF.

```

```

ENDFORM.                                " OPVOEREN_POLIS

```

```

*&-----*
*&      Form  GOED_POLIS
*&-----*

```

```

FORM goed_polis
  USING imp_org.
  CLEAR tb_verslag.
  tb_verslag-soort      = 'OK '.
  tb_verslag-regelnr   = wa_bron_polis-regel.
  tb_verslag-cat       = 'POLIS'.
  tb_verslag-origin   = imp_org.
  tb_verslag-bron      = wa_bron_polis-bron.
  tb_verslag-id        = wa_bron_polis-polnr.
  tb_verslag-text      = ls_verslag_text.
  APPEND tb_verslag.

```

```

vi_aant_pl_gd = vi_aant_pl_gd + 1.

```

```

ENDFORM.                                " GOED_POLIS

```

```

*&-----*
*&      Form  WIJZIGEN_POLIS
*&-----*

```

```

FORM wijzigen_polis.

```

```

* Wijzigen polis
* Lezen polisgegevens

```

```

DATA:
  lt_req_obj          TYPE crmt_object_name_tab,
  lt_appointm        TYPE crmt_appointment_wrkt,
  ls_appointm        LIKE LINE OF lt_appointm,
  lt_status           TYPE crmt_status_wrkt,
  ls_status          LIKE LINE OF lt_status,

  lt_appointm_com    TYPE crmt_appointment_comt,
  ls_appointm_com    LIKE LINE OF lt_appointm_com,
  lt_status_com       TYPE crmt_status_comt,
  ls_status_com      LIKE LINE OF lt_status_com,

  lt_input_fields     TYPE crmt_input_field_tab,
  ls_input_fields     LIKE LINE OF lt_input_fields,
  ls_fieldnames       LIKE LINE OF ls_input_fields-field_names,
  lt_object_guid      TYPE crmt_object_guid_tab.

```

```

REFRESH tb_header_guid.

```

```

APPEND wa_best_polis-guid TO tb_header_guid.
APPEND wa_best_polis-guid TO lt_object_guid.

```

```

INSERT gc_object_name-appointment INTO TABLE lt_req_obj.

```

```

                                Z_CRM_UPLOAD Example.txt
INSERT gc_object_name-status      INTO TABLE lt_req_obj.

CALL FUNCTION 'CRM_ORDER_READ'
  EXPORTING
    it_header_guid      = tb_header_guid
    it_requested_objects = lt_req_obj
  IMPORTING
    et_appointment     = lt_appointm
    et_status          = lt_status
  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.
  MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
    WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4
    INTO ls_verslag_text.
  PERFORM fout_polis.
  EXIT.
ENDIF.
* Check datum
LOOP AT lt_appointm INTO ls_appointm.
  CASE ls_appointm-appt_type.
    WHEN 'VALIDFROM'.
      IF ls_appointm-date_from <> wa_bron_polis-idat.
        MOVE-CORRESPONDING ls_appointm TO ls_appointm_com.
        CONVERT DATE wa_bron_polis-idat
          INTO TIME STAMP lv_tstamp TIME ZONE sy-zonlo.
        ls_appointm_com-timestamp_from = lv_tstamp.
        INSERT ls_appointm_com INTO TABLE lt_appointm_com.

        ls_input_fields-ref_guid = wa_best_polis-guid.
        ls_input_fields-ref_kind = 'A'.
        ls_input_fields-objectname = 'APPOINTMENT'.
        ls_input_fields-logical_key = 'VALIDFROM'.
        ls_fieldnames-fieldname = 'TIMESTAMP_FROM'.
        INSERT ls_fieldnames INTO TABLE ls_input_fields-field_names.
        ls_fieldnames-fieldname = 'TIMEZONE_FROM'.
        INSERT ls_fieldnames INTO TABLE ls_input_fields-field_names.
        INSERT ls_input_fields INTO TABLE lt_input_fields.

      ENDIF.
    WHEN 'VALIDTO'.
      IF ls_appointm-date_to <> wa_bron_polis-edat.
        CLEAR ls_appointm_com.
        CLEAR ls_input_fields.
        MOVE-CORRESPONDING ls_appointm TO ls_appointm_com.
        CONVERT DATE wa_bron_polis-edat
          INTO TIME STAMP lv_tstamp TIME ZONE sy-zonlo.
        ls_appointm_com-timestamp_from = lv_tstamp.
        INSERT ls_appointm_com INTO TABLE lt_appointm_com.

        ls_input_fields-ref_guid = wa_best_polis-guid.
        ls_input_fields-ref_kind = 'A'.
        ls_input_fields-objectname = 'APPOINTMENT'.
        ls_input_fields-logical_key = 'VALIDTO'.
        ls_fieldnames-fieldname = 'TIMESTAMP_FROM'.
        INSERT ls_fieldnames INTO TABLE ls_input_fields-field_names.
        ls_fieldnames-fieldname = 'TIMEZONE_FROM'.
        INSERT ls_fieldnames INTO TABLE ls_input_fields-field_names.
        INSERT ls_input_fields INTO TABLE lt_input_fields.

      ENDIF.
    WHEN OTHERS.

```

```

        ENDCASE.
    ENDLOOP.

* Check Status
READ TABLE tb_tj30t
    WITH KEY txt04 = wa_bron_polis-stat.

LOOP AT lt_status INTO ls_status
    WHERE user_stat_proc = cc_profile_zaspro1.
    IF ls_status-status <> tb_tj30t-estat.
        MOVE-CORRESPONDING ls_status TO ls_status_com.
        ls_status_com-ref_guid = ls_status-guid.
        ls_status_com-ref_kind = 'A'.
        ls_status_com-activate = 'X'.
        ls_status_com-status = tb_tj30t-estat.
        INSERT ls_status_com INTO TABLE lt_status_com.

        CLEAR ls_input_fields.
        ls_input_fields-ref_guid = wa_best_polis-guid.
        ls_input_fields-ref_kind = 'A'.
        ls_input_fields-objectname = 'STATUS'.
        CONCATENATE ls_status_com-status
            ls_status_com-user_stat_proc
            INTO ls_input_fields-logical_key.
        ls_fieldnames-fieldname = 'ACTIVATE'.
        INSERT ls_fieldnames INTO TABLE ls_input_fields-field_names.
        INSERT ls_input_fields INTO TABLE lt_input_fields.
    ENDIF.
ENDLOOP.

CHECK lt_input_fields[] IS NOT INITIAL.

CALL FUNCTION 'CRM_ORDER_MAINTAIN'
    EXPORTING
        it_appointment      = lt_appointm_com
        it_status            = lt_status_com
    CHANGING
        ct_input_fields      = lt_input_fields
    EXCEPTIONS
        error_occurred      = 1
        document_locked     = 2
        no_change_allowed   = 3
        no_authority        = 4
        OTHERS               = 5.
IF sy-subrc <> 0.
    MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
        WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4
        INTO ls_verslag_text.
    PERFORM fout_polis.
    EXIT.
ENDIF.

CALL FUNCTION 'CRM_ORDER_SAVE'
    EXPORTING
        it_objects_to_save = lt_object_guid
    EXCEPTIONS
        document_not_saved = 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
        INTO ls_verslag_text.
    PERFORM fout_polis.
    EXIT.
ENDIF.
CONCATENATE 'Nummer:' wa_best_polis-object_id '- gewijzigd'
    INTO ls_verslag_text SEPARATED BY space.
PERFORM goed_polis USING 'WIJZ'.

```

```

                                Z_CRM_UPLOAD Example.txt
CALL FUNCTION 'BAPI_TRANSACTION_COMMIT'.

ENDFORM.                            " WIJZIGEN_POLIS

*&-----*
*&      Form  VERSLAG
*&-----*
*-----*
FORM verslag.

  ULINE.
*
* Datum/tijd
*
WRITE: /7 'Datum:', sy-datum, ' Tijd:', sy-uzeit, AT 50 'Interface CRM'.

SKIP 1.
*
* Selectie params
*
WRITE: /7 'Bronbestand', imp_log-file(50).

IF pa_init = 'X'.
  WRITE /7 'Initi el'.
ELSE.
  WRITE /7 'wijzigingen'.
ENDIF.

WRITE: /7 'Regel van', so_regel-low, 't/m', so_regel-high.

WRITE /7 'Productierun'.

SKIP 1.
*
* Tellers afdrukken
*
WRITE: /7 'Klanten'.
WRITE: /14 'Totaal verwerkt :', vi_aant_kl_vw,
      /14 'Correct verwerkt:', vi_aant_kl_gd,
      /14 'Foutief          :', vi_aant_kl_ft.

SKIP 1.

WRITE: /7 'Intermediairs'.
WRITE: /14 'Totaal verwerkt :', vi_aant_im_vw,
      /14 'Correct verwerkt:', vi_aant_im_gd,
      /14 'Foutief          :', vi_aant_im_ft.

SKIP 1.

WRITE: /7 'Polissen'.
WRITE: /14 'Totaal verwerkt :', vi_aant_pl_vw,
      /14 'Correct verwerkt:', vi_aant_pl_gd,
      /14 'Foutief          :', vi_aant_pl_ft.

  ULINE.
ENDFORM.                            " VERSLAG

*&-----*
*&      Form  export_verslag
*&-----*
*      text
*-----*
FORM export_verslag.
CASE 'X'.
  WHEN pa_klnt. proc_srtfd-cat = 'KLANT'.

```

```
Z_CRM_UPLOAD Example.txt
  WHEN pa_intr. proc_srtfd-cat = 'INTER'.
  WHEN pa_pols. proc_srtfd-cat = 'POLIS'.
ENDCASE.

EXPORT verslag = tb_verslag
  TO DATABASE indx(zu)
  ID proc_srtfd.
ENDFORM.                                "export_verslag
```