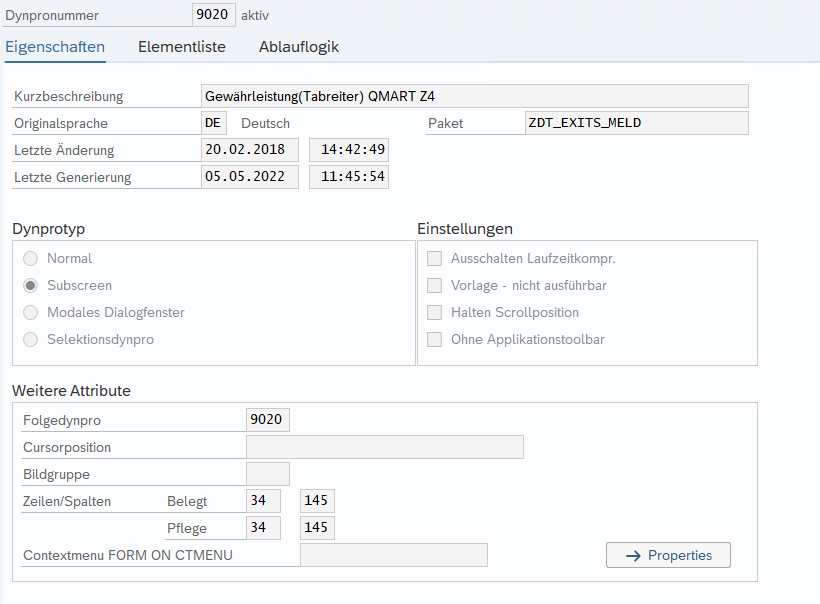
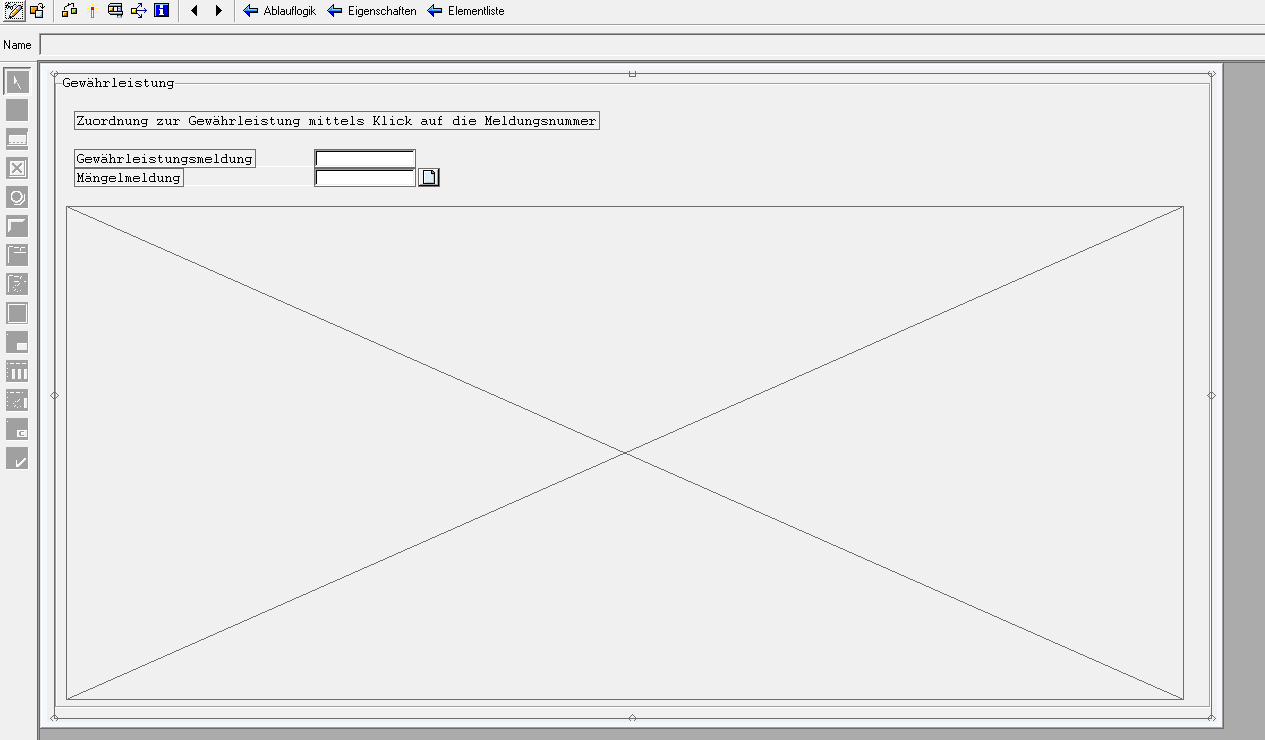
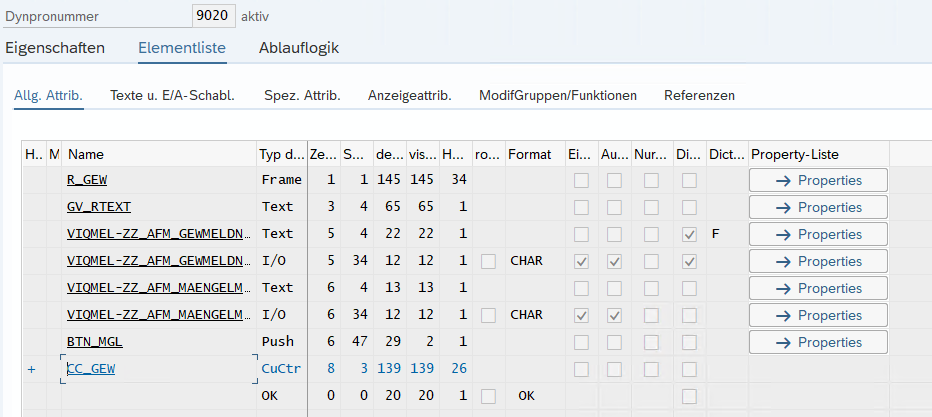
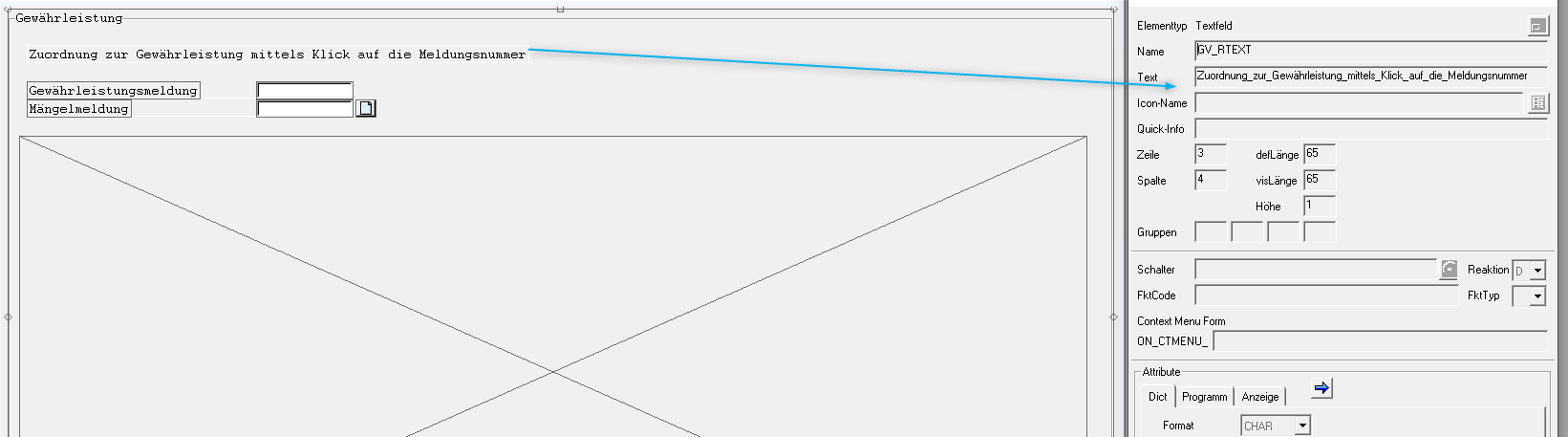
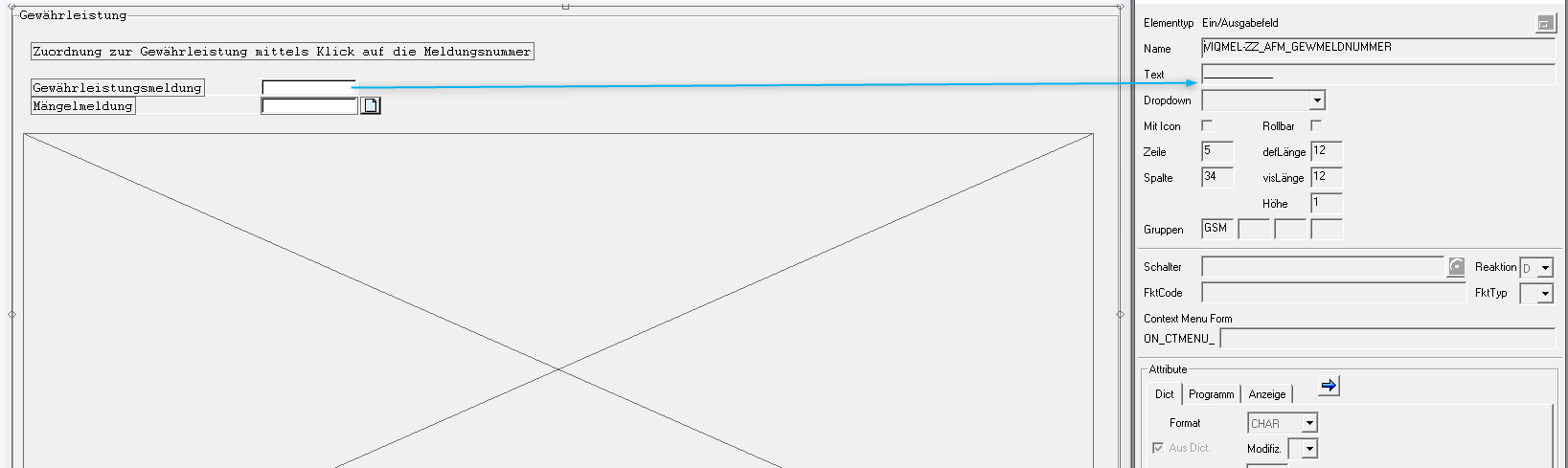
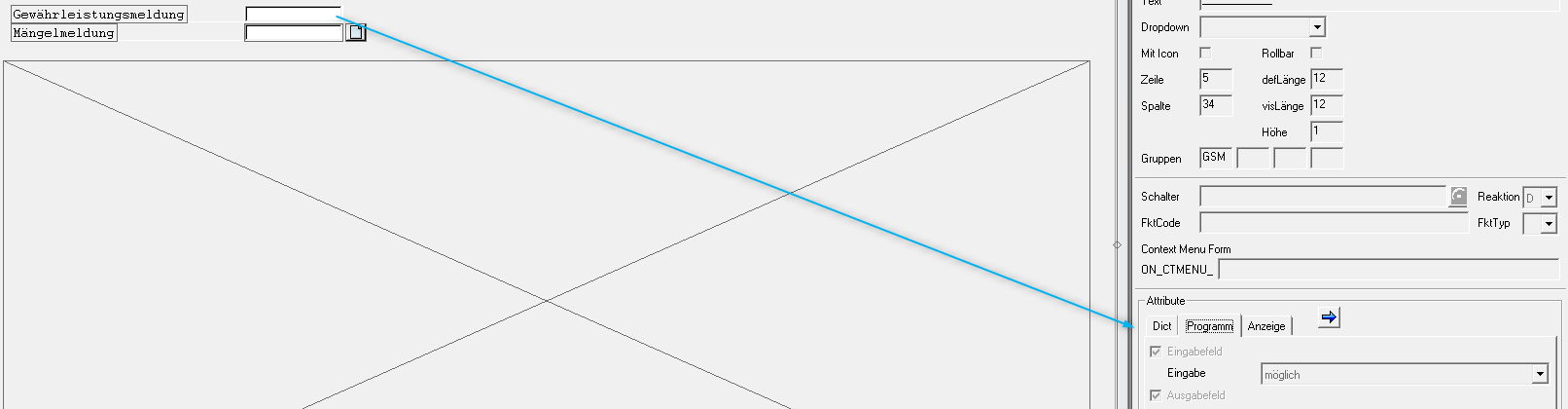
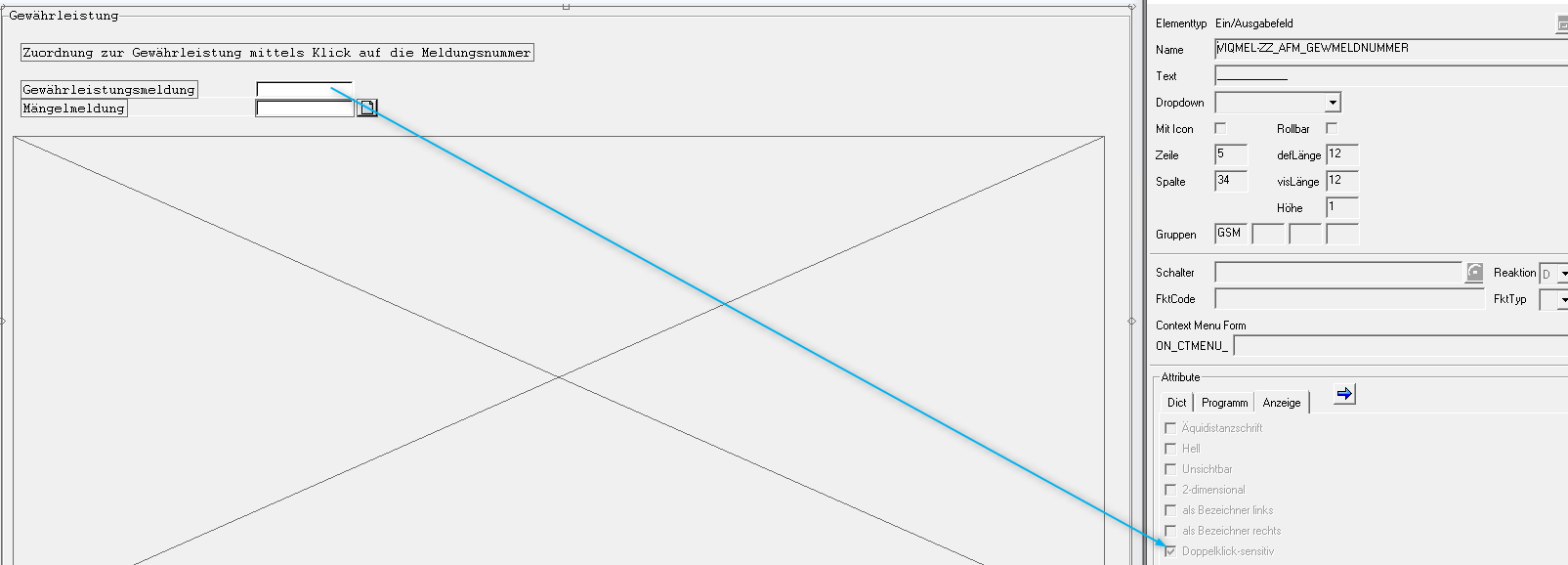
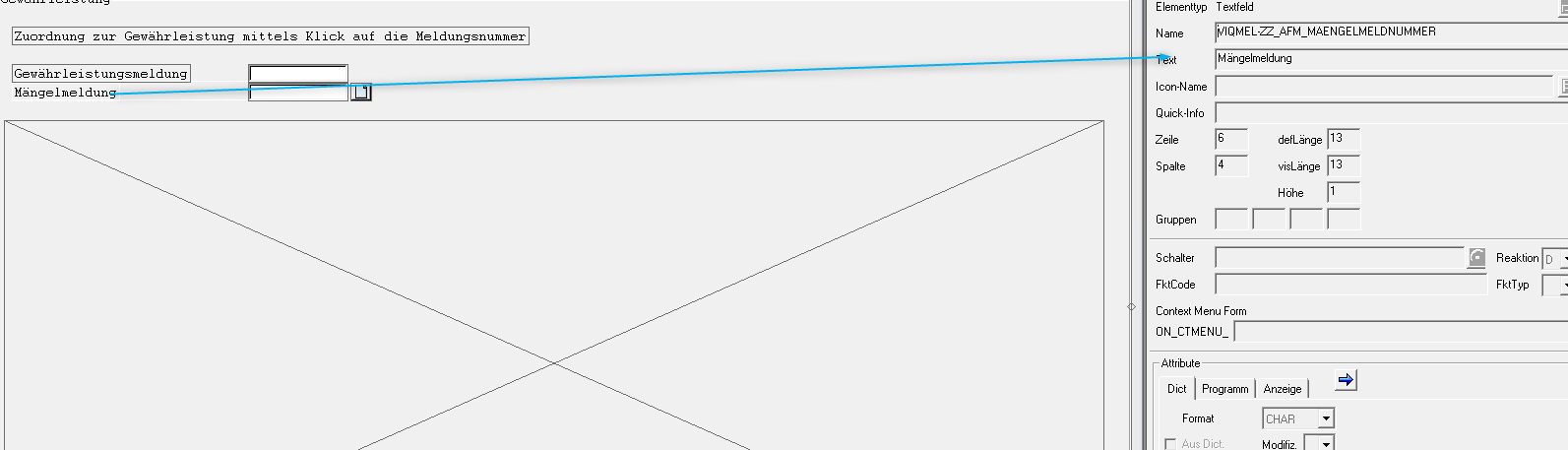
Bildschirm 🡪 9020  
  
  


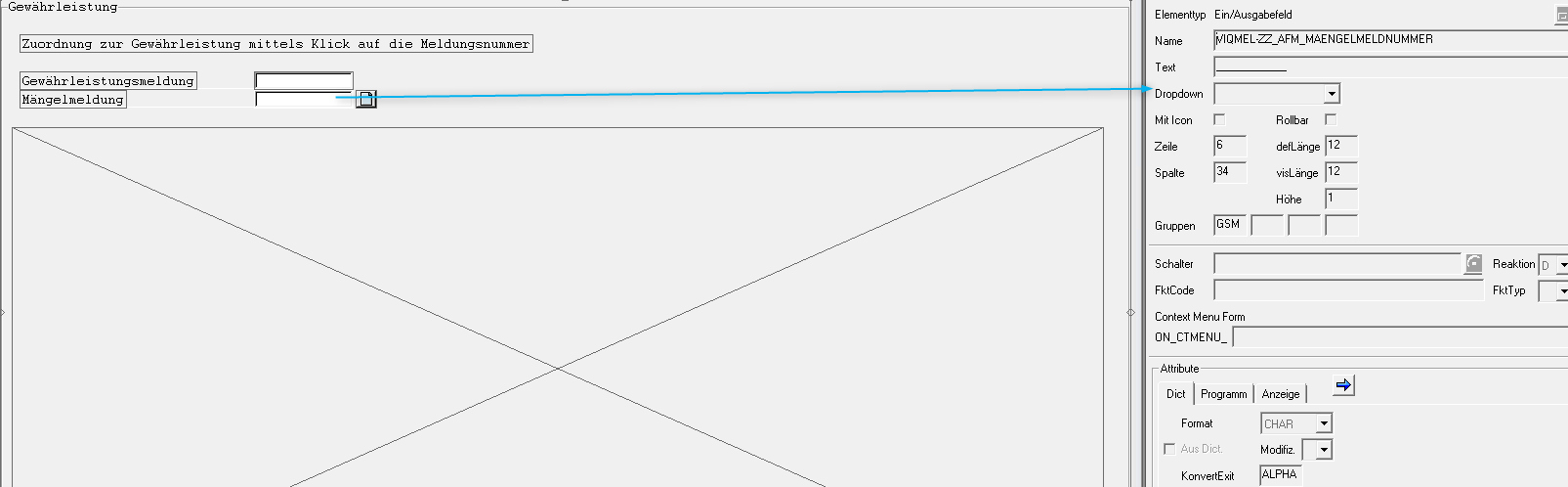


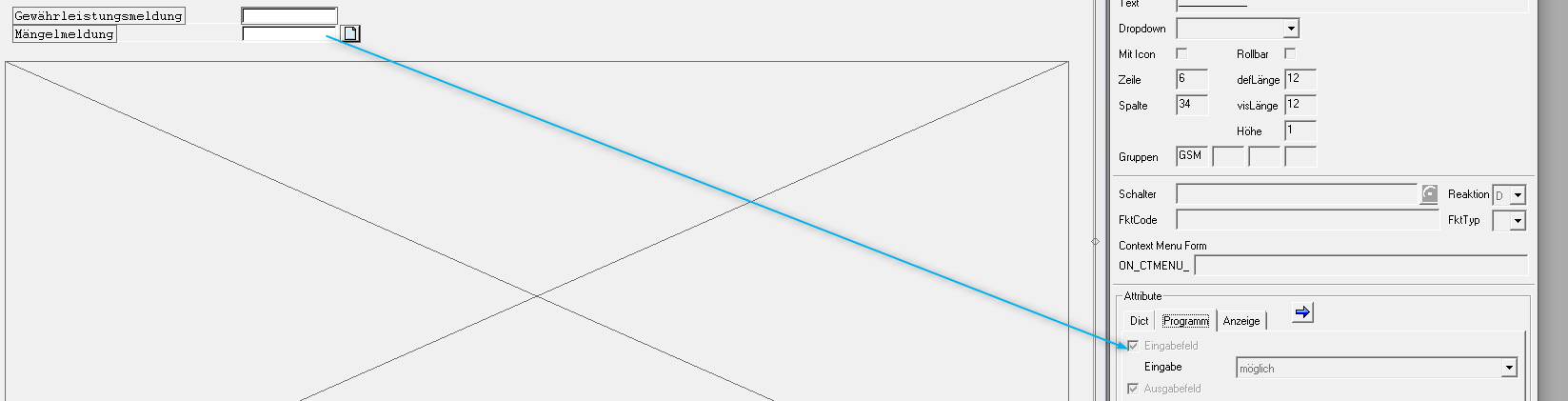
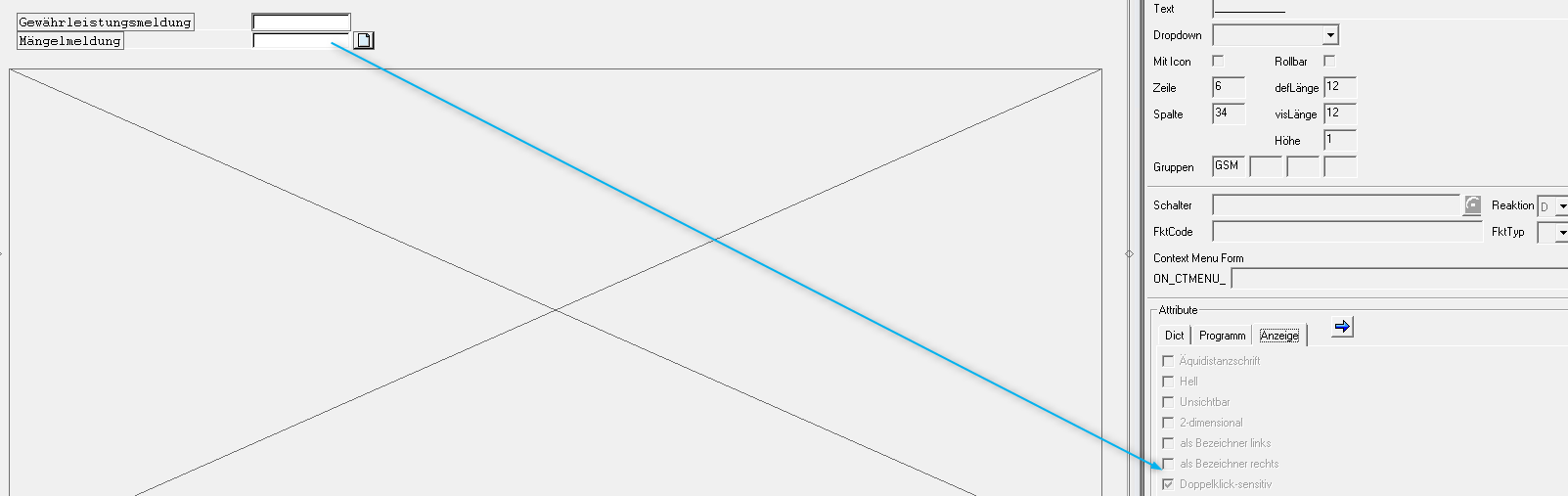


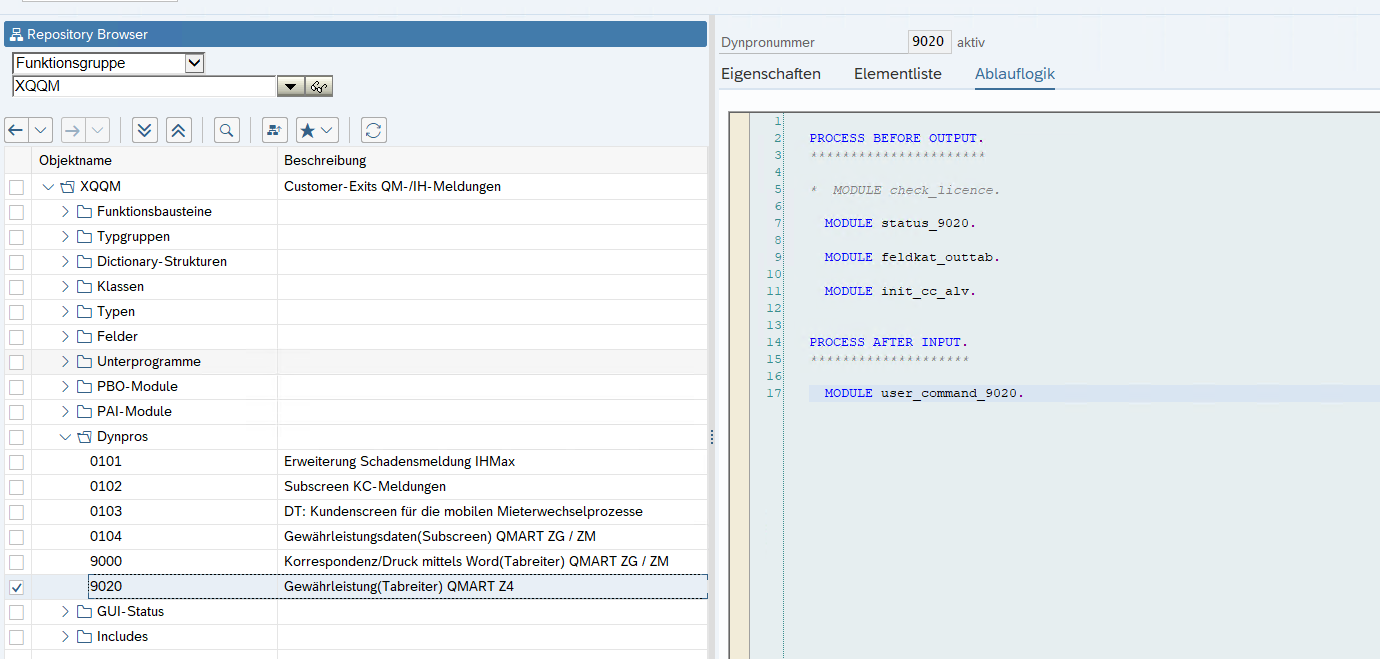










PROCESS BEFORE OUTPUT.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
*\* MODULE check\_licence.*  
  
 MODULE status\_9020.  
  
 MODULE feldkat\_outtab.  
  
 MODULE init\_cc\_alv.  
  
  
PROCESS AFTER INPUT.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
 MODULE user\_command\_9020.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMO20 .*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module STATUS\_9020 OUTPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE status\_9020 OUTPUT.  
  
*\* SET PF-STATUS 'xxxxxxxx'.*  
*\* SET TITLEBAR 'xxx'.*  
  
 LOOP AT SCREEN.  
 IF screen-group1 EQ 'GSM'.  
 screen-input = 0.  
 ENDIF.  
 MODIFY SCREEN.  
 ENDLOOP.  
  
*\* Drucktaste im Anzeigemodus ausgeschaltet werden*  
 IF g\_aktyp = 'A' OR viqmel-zz\_afm\_gewmeldnummer IS INITIAL.  
 LOOP AT SCREEN.  
 IF screen-name EQ 'BTN\_MGL'.  
 screen-active = 0.  
 ENDIF.  
 MODIFY SCREEN.  
 ENDLOOP.  
 ENDIF.  
  
*\* Import die Mängelmeldungsnummer von ABAP Memory*  
 IF flag\_mglm = '1'.  
 GET PARAMETER ID 'IQM' FIELD gv\_newmel.  
 IF gv\_newmel NE gv\_oldmel.  
 IF gv\_newmel NE viqmel-zz\_afm\_gewmeldnummer.  
*\* viqmel-z\_smmeldnummer = gv\_newmel.*  
 viqmel-zz\_afm\_maengelmeldnummer = gv\_newmel.  
 ENDIF.  
 ENDIF.  
 CLEAR flag\_mglm.  
 ENDIF.  
  
ENDMODULE. *" STATUS\_9020 OUTPUT*

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMO21 .*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module FELDKAT\_OUTTAB OUTPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE feldkat\_outtab OUTPUT.  
  
 DATA lv\_gw\_qmart TYPE qmel-qmart.  
  
*\*Feldkatalog ist gebaut.*  
 PERFORM build\_gt\_gw\_fcat.  
  
*\*Füllt die GW interne tabelle mit Dateien*  
 PERFORM fill\_gt\_gw\_fields.  
  
ENDMODULE. *" FELDKAT\_OUTTAB OUTPUT*

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF02.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form BUILD\_GT\_GW\_FCAT*  
*\*&---------------------------------------------------------------------\**  
FORM build\_gt\_gw\_fcat .  
  
 IF gt\_gw\_fcat IS INITIAL.  
*\* The following function module generates a fieldcatalog according*  
*\* to a given structure.*  
 CALL FUNCTION 'LVC\_FIELDCATALOG\_MERGE'  
 EXPORTING  
 i\_structure\_name = 'ZST\_AFM\_ZGEW\_ALV'  
 CHANGING  
 ct\_fieldcat = gt\_gw\_fcat.  
 DATA: ls\_fieldcat TYPE lvc\_s\_fcat.  
  
 CLEAR ls\_fieldcat.  
  
 ls\_fieldcat-fieldname = 'ICON'.  
 APPEND ls\_fieldcat TO gt\_gw\_fcat.  
  
*\* Now change the fieldcatalog to hide fields and to determine*  
*\* some initial calculations for chosen fields.*  
 LOOP AT gt\_gw\_fcat INTO gs\_gw\_fcat.  
 CASE gs\_gw\_fcat-fieldname.  
*\* hide columns which are already displayed in our tree*  
 WHEN 'OBJNR' OR 'QMART' OR 'ZZ\_AFM\_HIERARCHIE' .  
 gs\_gw\_fcat-no\_out = 'X'.  
 WHEN 'PARNR'.  
 gs\_gw\_fcat-col\_pos = 3.  
 WHEN 'EQUNR'.  
 gs\_gw\_fcat-col\_pos = 7.  
 gs\_gw\_fcat-outputlen = '10'.  
 WHEN 'TPLNR'.  
 gs\_gw\_fcat-outputlen = '20'.  
 WHEN 'ZZ\_AFM\_GEWBEGINN' OR 'ZZ\_AFM\_GEWENDE'.  
 gs\_gw\_fcat-outputlen = '20'.  
 WHEN 'ICON'.  
*\* gs\_gw\_fcat-no\_out = 'X'.*  
 gs\_gw\_fcat-col\_pos = 4.  
 gs\_gw\_fcat-icon = 'X'.  
 gs\_gw\_fcat-outputlen = '4'.  
 WHEN 'QMNUM'.  
 gs\_gw\_fcat-hotspot = 'X'.  
 ENDCASE.  
 MODIFY gt\_gw\_fcat FROM gs\_gw\_fcat.  
 ENDLOOP.  
 ENDIF.  
  
ENDFORM.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF07.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form FILL\_GT\_GW\_FIELDS*  
*\*&---------------------------------------------------------------------\**  
FORM fill\_gt\_gw\_fields .  
  
 TYPES: BEGIN OF ty\_tplnr,  
 tplnr TYPE tplnr,  
 kat TYPE i,  
 END OF ty\_tplnr.  
  
 DATA lt\_tplnr TYPE TABLE OF ty\_tplnr.  
 DATA ls\_tplnr TYPE ty\_tplnr.  
 DATA lt\_tp TYPE TABLE OF iflo.  
 DATA ls\_tp TYPE iflo.  
 DATA lt\_htp TYPE TABLE OF rihiflo.  
 DATA ls\_htp TYPE rihiflo.  
 DATA lt\_equi TYPE TABLE OF rihequi.  
 DATA ls\_equi TYPE rihequi.  
 DATA lv\_qmart TYPE qmart.  
 DATA ls\_outab LIKE LINE OF gt\_gw\_fields.  
  
*\*Entkopplung von den Kunden Meldungsarten Bezeichnungen*  
 SELECT SINGLE zkunde\_qmart FROM /afm/kundeqmart INTO lv\_qmart  
 WHERE /afm/kundeqmart~qmart = 'SG'.  
  
*\* Die Hierarchie des ausgewählten Tech.Platz in eigenen ITABs besorgt ist*  
 CALL FUNCTION 'PM\_PREDECESSOR\_IFLO'  
 EXPORTING  
*\* PLTXT = ' '*  
 selmod = 'D'  
*\* TPLMA = ' '*  
 tplnr = viqmel-tplnr  
 TABLES  
 iflo\_tab = lt\_tp  
 EXCEPTIONS  
 no\_predecessor = 1  
 no\_valid\_tplnr = 2  
 OTHERS = 3.  
 IF sy-subrc <> 0.  
*\* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO*  
*\* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.*  
 ENDIF.  
  
 CALL FUNCTION 'PM\_HIERARCHY\_CALL'  
 EXPORTING  
*\* DATUM = SY-DATUM*  
*\* EQUNR = ' '*  
*\* GRAFICS = ' '*  
*\* LEVDO = '01'*  
*\* LEVUP = '00'*  
*\* MATNR = ' '*  
*\* READ\_NEW = ' '*  
*\* SANIN = 'X'*  
 select\_equi = 'X'  
*\* SELECT\_IFLO = 'X'*  
*\* SELECT\_STPO = ' '*  
 selmod = 'D'  
*\* STKKZ = ' '*  
*\* TECHV = ' '*  
 tplnr = viqmel-tplnr  
*\* WERKS = ' '*  
 with\_equi = 'X'  
*\* WITH\_EQUI\_HIER = 'X'*  
 with\_iflo\_hier = 'X'  
*\* WITH\_BTYP = 'X'*  
*\* WITH\_MARA = ' '*  
*\* WITH\_IBASE\_HIER = ' '*  
*\* CAPID = ' '*  
*\* EMENG = 0*  
*\* ARBPL\_INT = ' '*  
*\* AUTHORITY\_CHECK = 'X'*  
*\* I\_GRBOM =*  
 TABLES  
 equi\_tab = lt\_equi  
 iflo\_tab = lt\_htp  
*\* STPO\_TAB =*  
 EXCEPTIONS  
 no\_hierarchy = 1  
 no\_object\_defined = 2  
 no\_selection = 3  
 no\_valid\_equnr = 4  
 no\_valid\_matnr = 5  
 no\_valid\_selmod = 6  
 no\_valid\_tplnr = 7  
 no\_valid\_grbom\_werks = 8  
 OTHERS = 9.  
 IF sy-subrc <> 0.  
*\* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO*  
*\* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.*  
 ENDIF.  
  
*\*die Über und Unten-geordneten tech.Platz Tabellen sind in einem ITAB zusammengepackt und ein Zeichen ist definiert.*  
*\*Das Zeichen identifiziert in die neue Tabelle welche TPLNR sind untergeordnete und welche sind übergeordnete*  
 LOOP AT lt\_tp INTO ls\_tp.  
 ls\_tplnr-tplnr = ls\_tp-tplnr.  
 ls\_tplnr-kat = '1'.  
 APPEND ls\_tplnr TO lt\_tplnr.  
 ENDLOOP.  
  
 LOOP AT lt\_htp INTO ls\_htp.  
 ls\_tplnr-tplnr = ls\_htp-tplnr.  
 ls\_tplnr-kat = '2'.  
 APPEND ls\_tplnr TO lt\_tplnr.  
 ENDLOOP.  
  
*\*Dopple Einträge sind gelöscht*  
 SORT lt\_tplnr BY tplnr.  
 DELETE ADJACENT DUPLICATES FROM lt\_tplnr.  
  
 LOOP AT lt\_tplnr INTO ls\_tplnr WHERE tplnr = viqmel-tplnr.  
 ls\_tplnr-kat = 0.  
 MODIFY lt\_tplnr FROM ls\_tplnr TRANSPORTING kat.  
 ENDLOOP.  
  
 IF viqmel-equnr IS NOT INITIAL.  
 READ TABLE lt\_equi INTO ls\_equi WITH KEY tplnr = viqmel-tplnr equnr = viqmel-equnr TRANSPORTING NO FIELDS.  
 IF sy-subrc = 0 .  
 LOOP AT lt\_tplnr INTO ls\_tplnr.  
 IF ls\_tplnr-kat = '2'.  
 DELETE lt\_tplnr WHERE tplnr = ls\_tplnr-tplnr.  
 ENDIF.  
 ENDLOOP.  
 ENDIF.  
 ENDIF.  
  
*\*Die outtab Tabelle für den ALV ist gefüllt*  
  
 SELECT qmnum  
 viqmelst~qmart  
 viqmelst~qmtxt  
 viqmelst~objnr  
*\* ihpa~parnr*  
 zz\_afm\_gewbeginn  
 zz\_afm\_gewende  
 zz\_afm\_geweurobetrag  
 viqmelst~tplnr  
 viqmelst~equnr  
 pltxt FROM viqmelst  
 INNER JOIN iflotx ON iflotx~tplnr = viqmelst~tplnr  
*\* Info Meldenderdaten (Partner) entfällt hier !*  
*\* INNER JOIN ihpa ON ihpa~objnr = viqmelst~objnr*  
 INTO CORRESPONDING FIELDS OF TABLE gt\_gw\_fields FOR ALL ENTRIES IN lt\_tplnr  
 WHERE viqmelst~tplnr = lt\_tplnr-tplnr  
*\* Vorbereitung für den Fall die Equipments eingeplant sind*  
*\* WHERE ( viqmelst~tplnr = viqmel-tplnr ) OR ( viqmelst~equnr = viqmel-equnr )*  
 AND viqmelst~qmart = lv\_qmart.  
  
 LOOP AT gt\_gw\_fields INTO ls\_outab.  
 LOOP AT lt\_tplnr INTO ls\_tplnr.  
 IF ls\_tplnr-tplnr = ls\_outab-tplnr.  
 ls\_outab-zz\_afm\_hierarchie = ls\_tplnr-kat.  
 ENDIF.  
 ENDLOOP.  
 MODIFY gt\_gw\_fields FROM ls\_outab TRANSPORTING zz\_afm\_hierarchie.  
 ENDLOOP.  
  
*\*Die Ikonen Spalte ist mit entsprechenden Ikonen ausgefüllt.*  
 CLEAR ls\_outab.  
 LOOP AT gt\_gw\_fields INTO ls\_outab.  
 CASE ls\_outab-zz\_afm\_hierarchie.  
 WHEN 0.  
 ls\_outab-icon = '@0E@'.  
 WHEN 1.  
 ls\_outab-icon = '@0I@'.  
 WHEN 2.  
 ls\_outab-icon = '@0H@'.  
 ENDCASE.  
 MODIFY gt\_gw\_fields FROM ls\_outab TRANSPORTING icon.  
 ENDLOOP.  
  
 CLEAR ls\_outab.  
 LOOP AT gt\_gw\_fields INTO ls\_outab WHERE equnr IS NOT INITIAL.  
 IF ls\_outab-equnr NE viqmel-equnr.  
 DELETE gt\_gw\_fields WHERE tplnr = viqmel-tplnr AND equnr = ls\_outab-equnr.  
 ENDIF.  
 ENDLOOP.  
  
*\*Dopple Einträge sind gelöscht*  
 SORT gt\_gw\_fields BY qmnum.  
 DELETE ADJACENT DUPLICATES FROM gt\_gw\_fields COMPARING qmnum.  
  
*\* Die outtab Tabelle ist über den Tech.Platz Nummer sortiert.*  
 SORT gt\_gw\_fields BY tplnr.  
  
ENDFORM.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMO22 .*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module INIT\_CC\_ALV OUTPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE init\_cc\_alv OUTPUT.  
  
 IF go\_gw\_cont\_alv IS INITIAL.  
 CREATE OBJECT go\_gw\_cont\_alv  
 EXPORTING  
*\* PARENT =*  
 container\_name = 'CC\_GEW'  
*\* STYLE =*  
*\* LIFETIME = lifetime\_default*  
*\* REPID =*  
*\* DYNNR =*  
*\* NO\_AUTODEF\_PROGID\_DYNNR =*  
 EXCEPTIONS  
 cntl\_error = 1  
 cntl\_system\_error = 2  
 create\_error = 3  
 lifetime\_error = 4  
 lifetime\_dynpro\_dynpro\_link = 5  
 OTHERS = 6  
 .  
 IF sy-subrc <> 0.  
 MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
 WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
 ENDIF.  
 ENDIF.  
  
 IF go\_gw\_alv IS INITIAL.  
 CREATE OBJECT go\_gw\_alv  
 EXPORTING  
*\* I\_SHELLSTYLE = 0*  
*\* I\_LIFETIME =*  
 i\_parent = go\_gw\_cont\_alv  
*\* I\_APPL\_EVENTS = space*  
*\* I\_PARENTDBG =*  
*\* I\_APPLOGPARENT =*  
*\* I\_GRAPHICSPARENT =*  
*\* I\_NAME =*  
*\* I\_FCAT\_COMPLETE = SPACE*  
 EXCEPTIONS  
 error\_cntl\_create = 1  
 error\_cntl\_init = 2  
 error\_cntl\_link = 3  
 error\_dp\_create = 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.  
 ENDIF.  
  
 PERFORM exc\_functions CHANGING gt\_toolbar.  
 PERFORM set\_layout CHANGING gt\_layo.  
  
*\*Ereignisse registrieren*  
 SET HANDLER lcl\_gui\_alv\_grid=>on\_toolbar FOR go\_gw\_alv.  
 SET HANDLER lcl\_gui\_alv\_grid=>on\_doubleclick FOR go\_gw\_alv.  
 SET HANDLER lcl\_gui\_alv\_grid=>on\_usercommand FOR go\_gw\_alv.  
 SET HANDLER lcl\_gui\_alv\_grid=>on\_click\_row FOR go\_gw\_alv.  
  
 CALL METHOD go\_gw\_alv->set\_table\_for\_first\_display  
 EXPORTING  
*\* I\_BUFFER\_ACTIVE =*  
*\* I\_BYPASSING\_BUFFER =*  
*\* I\_CONSISTENCY\_CHECK =*  
*\* I\_STRUCTURE\_NAME =*  
*\* IS\_VARIANT =*  
*\* I\_SAVE =*  
*\* I\_DEFAULT = 'X'*  
 is\_layout = gt\_layo  
*\* IS\_PRINT =*  
*\* IT\_SPECIAL\_GROUPS =*  
 it\_toolbar\_excluding = gt\_toolbar  
*\* IT\_HYPERLINK =*  
*\* IT\_ALV\_GRAPHICS =*  
*\* IT\_EXCEPT\_QINFO =*  
*\* IR\_SALV\_ADAPTER =*  
 CHANGING  
 it\_outtab = gt\_gw\_fields  
 it\_fieldcatalog = gt\_gw\_fcat  
*\* IT\_SORT =*  
*\* IT\_FILTER =*  
*\* EXCEPTIONS*  
*\* INVALID\_PARAMETER\_COMBINATION = 1*  
*\* PROGRAM\_ERROR = 2*  
*\* TOO\_MANY\_LINES = 3*  
*\* others = 4*  
 .  
 IF sy-subrc <> 0.  
*\* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO*  
*\* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.*  
 ENDIF.  
  
 ELSE.  
  
 DATA ls\_stable TYPE lvc\_s\_stbl.  
  
 ls\_stable-row = 'X'.  
 ls\_stable-col = 'X'.  
 CALL METHOD go\_gw\_alv->refresh\_table\_display  
 EXPORTING  
 is\_stable = ls\_stable  
*\* I\_SOFT\_REFRESH =*  
*\* EXCEPTIONS*  
*\* FINISHED = 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.  
  
 CALL METHOD go\_gw\_alv->set\_selected\_rows  
 EXPORTING  
 it\_index\_rows = gt\_rowid  
*\* IT\_ROW\_NO =*  
*\* IS\_KEEP\_OTHER\_SELECTIONS =*  
 .  
 ENDIF.  
  
ENDMODULE. *" INIT\_CC\_ALV OUTPUT*

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF05.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form EXC\_FUNCTIONS*  
*\*&---------------------------------------------------------------------\**  
*\* <--P\_GT\_TOOLBAR text*  
*\*----------------------------------------------------------------------\**  
FORM exc\_functions CHANGING p\_gt\_toolbar TYPE ui\_functions.  
  
 DATA ls\_exclude TYPE ui\_func.  
  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_fc\_maximum .  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_fc\_minimum .  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_fc\_subtot .  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_fc\_sum .  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_fc\_average .  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_mb\_sum .  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_mb\_subtot .  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_fc\_print.  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_fc\_graph.  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_fc\_info.  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
 ls\_exclude = cl\_gui\_alv\_grid=>mc\_fc\_views.  
 APPEND ls\_exclude TO p\_gt\_toolbar.  
  
ENDFORM.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF06.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form SET\_LAYOUT*  
*\*&---------------------------------------------------------------------\**  
*\* <--P\_GT\_LAYO text*  
*\*----------------------------------------------------------------------\**  
FORM set\_layout CHANGING p\_gt\_layo TYPE lvc\_s\_layo.  
  
 p\_gt\_layo-col\_opt = 'X'.  
 p\_gt\_layo-zebra = 'X'.  
 p\_gt\_layo-sel\_mode = 'A'.  
 p\_gt\_layo-no\_rowmark = ' '.  
  
ENDFORM.

Include🡪 ZXQQMI02

*\*&---------------------------------------------------------------------\**  
*\*& Module USER\_COMMAND\_9020 INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE user\_command\_9020 INPUT.  
  
 CASE sy-ucomm.  
  
 WHEN 'ZUR'.  
 IF g\_aktyp <> 'A'.  
 CLEAR viqmel-zz\_afm\_gewmeldnummer.  
 ENDIF.  
  
 WHEN 'ARCH1'.  
 IF g\_aktyp <> 'A'.  
 fl\_rueck = '1'.  
 EXPORT fl\_rueck TO MEMORY ID 'SCHLIESSEN'.  
 PERFORM go\_to\_maengelmeld.  
 PERFORM clear\_fields.  
 ENDIF.  
  
 WHEN 'GEW'.  
 IF g\_aktyp <> 'A'.  
 viqmel-zz\_afm\_gewmeldnummer = gw\_meldnumm.  
 ENDIF.  
  
 WHEN 'MGL'.  
 PERFORM create\_maengel\_meldung.  
  
 WHEN 'FWD'.  
 PERFORM forward\_to\_meld.  
  
 ENDCASE.  
  
 IF sy-ucomm = 'ZUR' OR sy-ucomm = 'GEW' OR sy-ucomm = 'MGL' OR sy-ucomm = 'ARCH1' OR sy-ucomm = 'FWD'.  
 PERFORM supress\_f31(sapliqs0).  
 ENDIF.  
  
 PERFORM go\_to\_meld USING 'VIQMEL-ZZ\_AFM\_GEWMELDNUMMER' viqmel-zz\_afm\_gewmeldnummer.  
*\* PERFORM go\_to\_meld USING 'VIQMEL-Z\_SMMELDNUMMER' viqmel-z\_smmeldnummer.*  
 PERFORM go\_to\_meld USING 'VIQMEL-ZZ\_AFM\_MAENGELMELDNUMMER' viqmel-zz\_afm\_maengelmeldnummer.  
  
ENDMODULE. *" USER\_COMMAND\_9020 INPUT*

*Include🡪* *ZXQQMF09*

*\*&---------------------------------------------------------------------\**  
*\*& Form GO\_TO\_MAENGELMELD*  
*\*&---------------------------------------------------------------------\**  
FORM go\_to\_maengelmeld .  
  
 DATA wa\_bdcdata TYPE bdcdata.  
 DATA lt\_bdcdata TYPE TABLE OF bdcdata.  
  
 CLEAR: wa\_bdcdata,  
 lt\_bdcdata.  
 wa\_bdcdata-program = 'SAPLIQS0'.  
 wa\_bdcdata-dynpro = '0100'.  
 wa\_bdcdata-dynbegin = 'X'.  
 APPEND wa\_bdcdata TO lt\_bdcdata.  
  
 CLEAR wa\_bdcdata.  
 wa\_bdcdata-fnam = 'RIWO00-QMNUM'.  
*\* wa\_bdcdata-fval = viqmel-z\_smmeldnummer.*  
 wa\_bdcdata-fval = viqmel-zz\_afm\_maengelmeldnummer.  
 APPEND wa\_bdcdata TO lt\_bdcdata.  
  
 CLEAR wa\_bdcdata.  
 wa\_bdcdata-fnam = 'BDC\_OKCODE'.  
 wa\_bdcdata-fval = 'MELD'.  
 APPEND wa\_bdcdata TO lt\_bdcdata.  
  
 CALL TRANSACTION 'IW22' USING lt\_bdcdata MODE 'E'.  
  
ENDFORM.

*Include🡪* *ZXQQMF03*

*\*&---------------------------------------------------------------------\**  
*\*& Form CLEAR\_FIELDS*  
*\*&---------------------------------------------------------------------\**  
FORM clear\_fields .  
  
 DATA lv\_qmdab TYPE qmel-qmdab.  
  
 SELECT SINGLE qmdab FROM qmel INTO lv\_qmdab  
 WHERE qmnum = viqmel-zz\_afm\_maengelmeldnummer.  
  
 IF lv\_qmdab IS NOT INITIAL.  
*\* CLEAR viqmel-z\_smmeldnummer.*  
 CLEAR viqmel-zz\_afm\_maengelmeldnummer.  
 CLEAR viqmel-zz\_afm\_gewmeldnummer.  
 ENDIF.  
  
ENDFORM.

Include🡪 ZXQQMF04

*\*&---------------------------------------------------------------------\**  
*\*& Form CREATE\_MAENGEL\_MELDUNG*  
*\*&---------------------------------------------------------------------\**  
FORM create\_maengel\_meldung .  
  
 TYPES:BEGIN OF ty\_ls\_gew.  
 INCLUDE TYPE viqmel.  
 TYPES parnr TYPE i\_parnr.  
 TYPES END OF ty\_ls\_gew.  
  
 DATA ls\_viqmel TYPE ty\_ls\_gew.  
  
 DATA wa\_bdcdata TYPE bdcdata.  
 DATA lt\_bdcdata TYPE TABLE OF bdcdata.  
  
 DATA lv\_qmart TYPE qmart.  
 DATA flag TYPE c LENGTH 1.  
 DATA ls\_params TYPE ctu\_params.  
  
*\*&---------------------------------------------------------------------\**  
  
 SELECT SINGLE zz\_afm\_ebeln  
 zz\_afm\_ebelp  
 zz\_afm\_gewbestellsum  
 zz\_afm\_gewprozent  
 zz\_afm\_geweurobetrag  
 zz\_afm\_gewbeginn  
 zz\_afm\_gewende  
*\* ihpa~parnr*  
 FROM viqmel  
*\* INNER JOIN ihpa ON ihpa~objnr = viqmel~objnr*  
 INTO CORRESPONDING FIELDS OF ls\_viqmel  
 WHERE qmnum = viqmel-zz\_afm\_gewmeldnummer.  
  
 IF sy-subrc = 0.  
*\* Entkopplung von den Kunden Meldungsarten Bezeichnungen*  
 SELECT SINGLE zkunde\_qmart FROM /afm/kundeqmart INTO lv\_qmart  
 WHERE /afm/kundeqmart~qmart = 'SM'.  
 ls\_viqmel-qmtxt = viqmel-qmtxt.  
 ls\_viqmel-tplnr = viqmel-tplnr.  
 ls\_viqmel-equnr = viqmel-equnr.  
 ls\_viqmel-zz\_afm\_gewmeldnummer = viqmel-zz\_afm\_gewmeldnummer.  
  
*\* Übernahme Störmeldungsnummer*  
 ls\_viqmel-zz\_afm\_stoermeldnummer = viqmel-qmnum.  
  
*\* Übernahme Prio aus Z4(Störmeldungsnummer) in ZM*  
 ls\_viqmel-priok = viqmel-priok.  
  
*\* Zeichen um den Mängelmeldungs Feld ist gefüllt nur wenn die Drucktatste Anlegen ist betätigt*  
*\* und eine neue Mängelmeldung ist angelegt*  
 flag\_mglm = '1'.  
 GET PARAMETER ID 'IQM' FIELD gv\_oldmel.  
*\* Speichert die Daten von der Gewährleistungsmeldung in ABAP Memory*  
*\* um sie in der neue angelegte Mängelmeldung zu kopieren*  
 EXPORT ls\_viqmel TO MEMORY ID 'NEUE\_MÄNGELMELDUNG'.  
  
 wa\_bdcdata-program = 'SAPLIQS0'.  
 wa\_bdcdata-dynpro = '0100'.  
 wa\_bdcdata-dynbegin = 'X'.  
 APPEND wa\_bdcdata TO lt\_bdcdata. CLEAR wa\_bdcdata.  
  
 wa\_bdcdata-fnam = 'RIWO00-QMART'.  
 wa\_bdcdata-fval = lv\_qmart.  
 APPEND wa\_bdcdata TO lt\_bdcdata. CLEAR wa\_bdcdata.  
 SET PARAMETER ID 'QMR' FIELD lv\_qmart.  
  
 wa\_bdcdata-fnam = 'BDC\_OKCODE'.  
 wa\_bdcdata-fval = 'MELD'.  
 APPEND wa\_bdcdata TO lt\_bdcdata. CLEAR wa\_bdcdata.  
  
 ls\_params-nobinpt = 'X'.  
 ls\_params-dismode = 'E'.  
 ls\_params-updmode = 'A'.  
  
 CALL TRANSACTION 'IW21' USING lt\_bdcdata  
 OPTIONS FROM ls\_params.  
  
 ENDIF.  
  
ENDFORM.

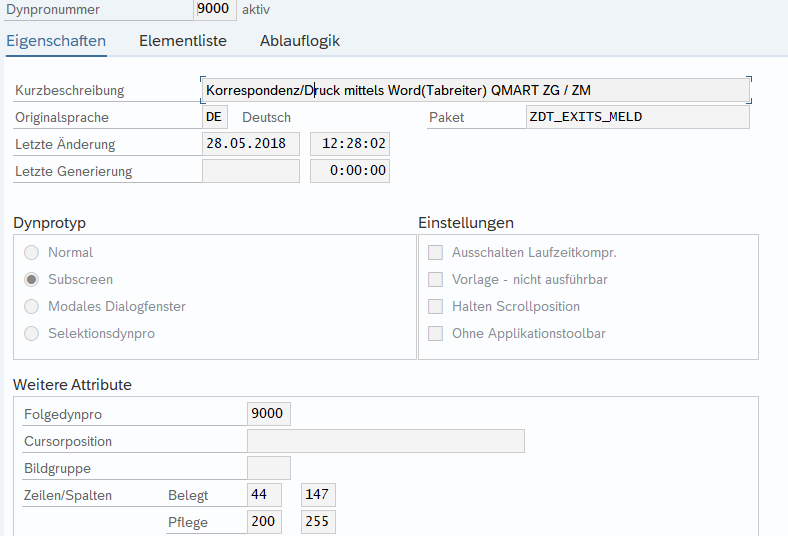
*Include🡪* *ZXQQMF08*

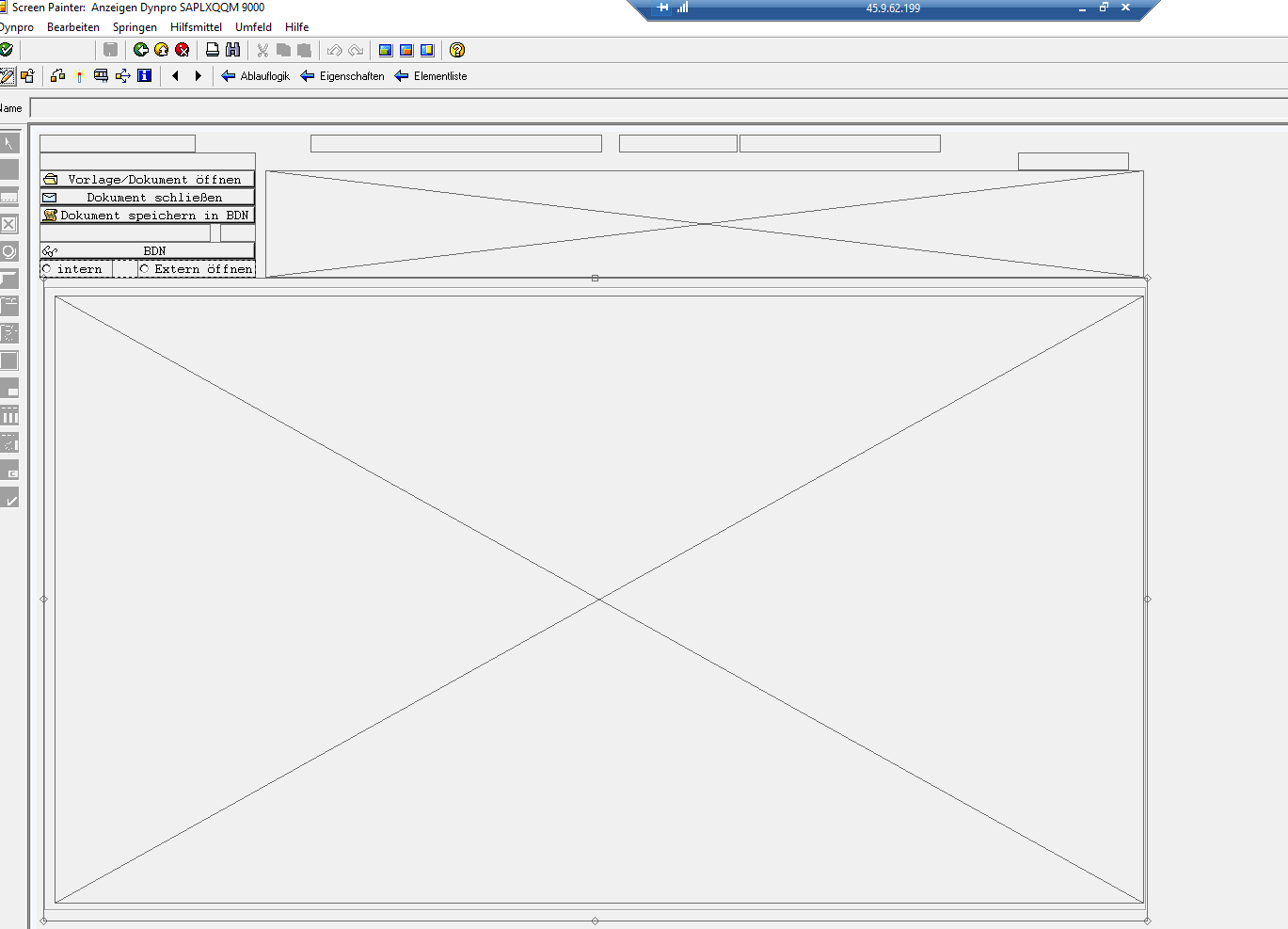
*\*&---------------------------------------------------------------------\**  
*\*& Form FORWARD\_TO\_MELD*  
*\*&---------------------------------------------------------------------\**  
FORM forward\_to\_meld .  
  
 DATA wa\_bdcdata TYPE bdcdata.  
 DATA lt\_bdcdata TYPE TABLE OF bdcdata.  
  
*\*&---------------------------------------------------------------------\**  
  
 CLEAR: wa\_bdcdata, lt\_bdcdata.  
 wa\_bdcdata-program = 'SAPLIQS0'.  
 wa\_bdcdata-dynpro = '0100'.  
 wa\_bdcdata-dynbegin = 'X'.  
 APPEND wa\_bdcdata TO lt\_bdcdata.  
  
 CLEAR wa\_bdcdata.  
 wa\_bdcdata-fnam = 'RIWO00-QMNUM'.  
 wa\_bdcdata-fval = gw\_meldnumm.  
 APPEND wa\_bdcdata TO lt\_bdcdata.  
  
 CLEAR wa\_bdcdata.  
 wa\_bdcdata-fnam = 'BDC\_OKCODE'.  
 wa\_bdcdata-fval = 'MELD'.  
 APPEND wa\_bdcdata TO lt\_bdcdata.  
  
 CALL TRANSACTION 'IW23' USING lt\_bdcdata MODE 'E'.  
  
ENDFORM.

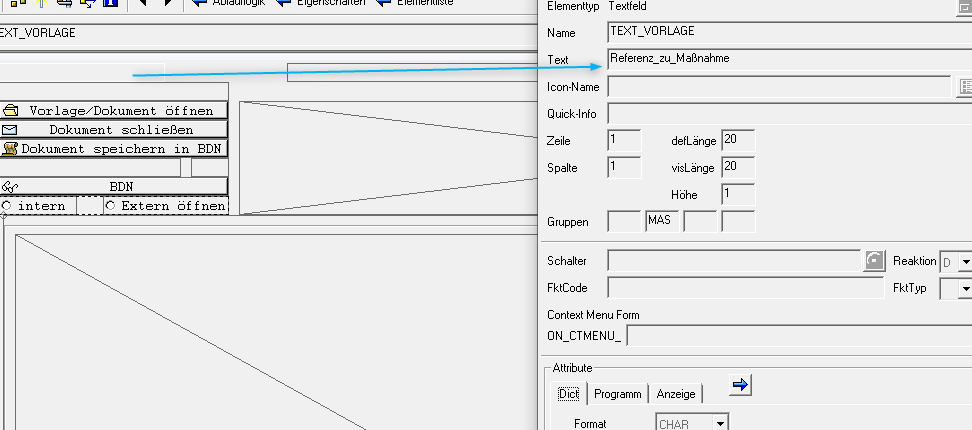
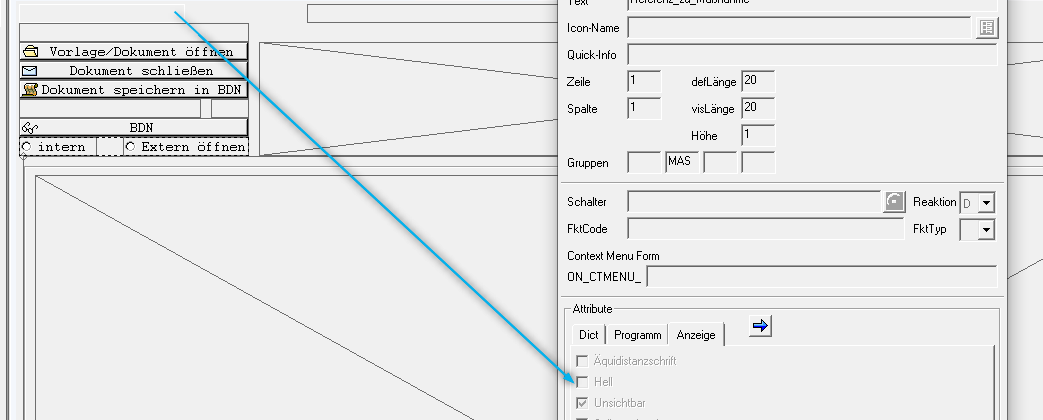
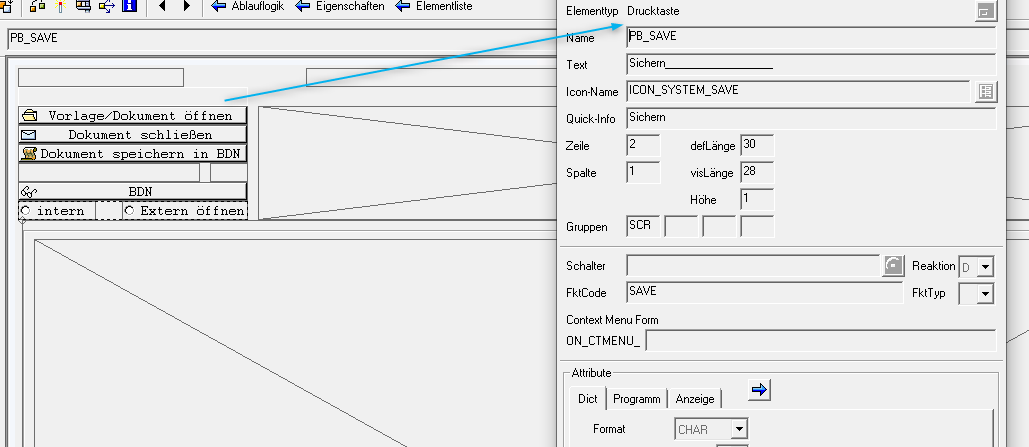
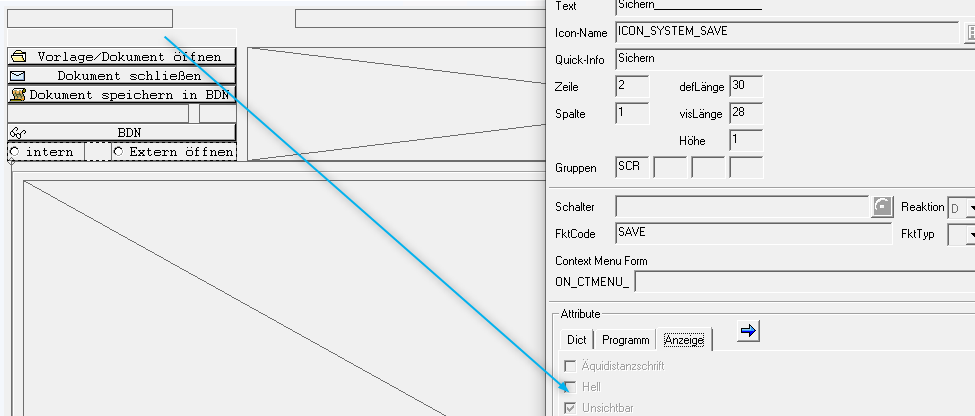
*Include🡪* *ZXQQMF10*

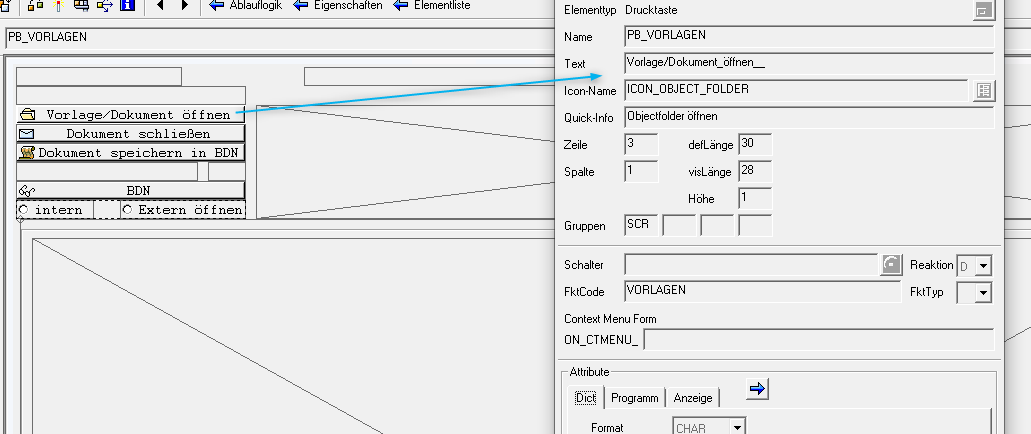
*\*&---------------------------------------------------------------------\**  
*\*& Form GO\_TO\_MELD*  
*\*&---------------------------------------------------------------------\**  
*\* -->P\_0080 text*  
*\* -->P\_VIQMEL\_ZZ\_AFM\_GEWMELDNUMMER text*  
*\*----------------------------------------------------------------------\**  
FORM go\_to\_meld USING p\_qmart TYPE string  
 p\_qmnum TYPE qmnum.  
  
 DATA lcurso TYPE string. *"LIKE riwo1-curso.*  
 DATA ls\_qmel LIKE qmel.  
 DATA lt\_bdcdata TYPE TABLE OF bdcdata.  
 DATA wa\_bdcdata TYPE bdcdata.  
  
*\*----------------------------------------------------------------------\**  
  
 GET CURSOR FIELD lcurso.  
  
 IF lcurso = p\_qmart.  
  
 IF p\_qmnum IS NOT INITIAL.  
 SELECT SINGLE qmnum FROM qmel INTO CORRESPONDING FIELDS OF ls\_qmel  
 WHERE qmnum = p\_qmnum.  
  
 IF sy-dbcnt <> 0 AND ls\_qmel-qmnum IS NOT INITIAL.  
  
 CLEAR: wa\_bdcdata,  
 lt\_bdcdata.  
 wa\_bdcdata-program = 'SAPLIQS0'.  
 wa\_bdcdata-dynpro = '0100'.  
 wa\_bdcdata-dynbegin = 'X'.  
 APPEND wa\_bdcdata TO lt\_bdcdata.  
  
 CLEAR wa\_bdcdata.  
 wa\_bdcdata-fnam = 'RIWO00-QMNUM'.  
 wa\_bdcdata-fval = p\_qmnum.  
 APPEND wa\_bdcdata TO lt\_bdcdata.  
  
 CLEAR wa\_bdcdata.  
 wa\_bdcdata-fnam = 'BDC\_OKCODE'.  
 wa\_bdcdata-fval = 'MELD'.  
 APPEND wa\_bdcdata TO lt\_bdcdata.  
  
 CALL TRANSACTION 'IW22' USING lt\_bdcdata MODE 'E'.  
  
 ENDIF.  
 ENDIF.  
 ENDIF.  
  
ENDFORM.

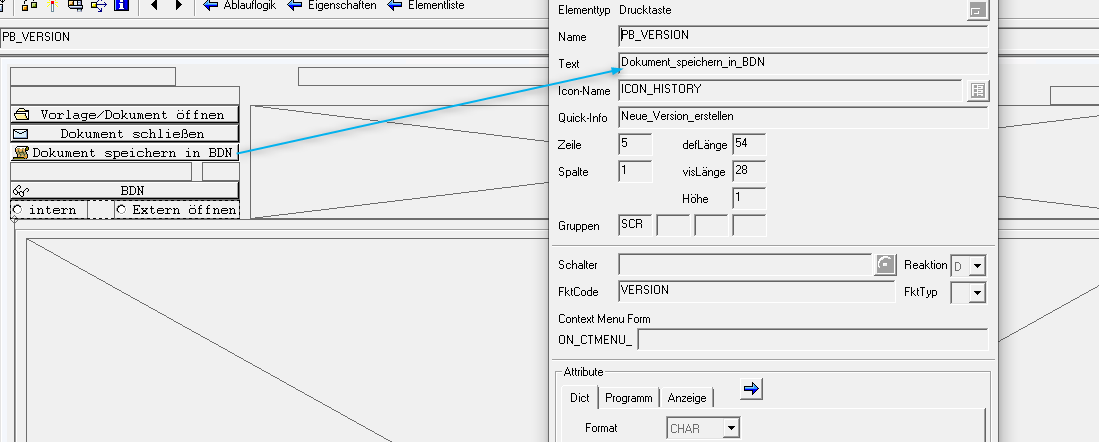
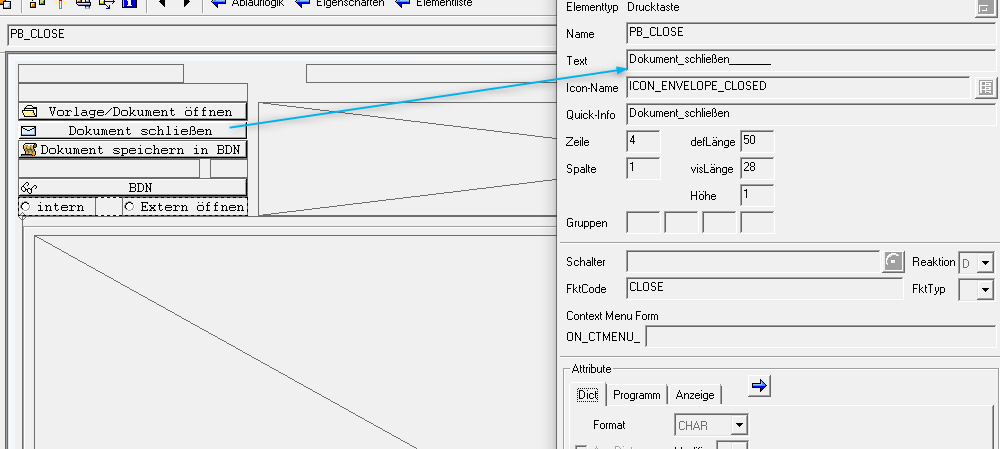
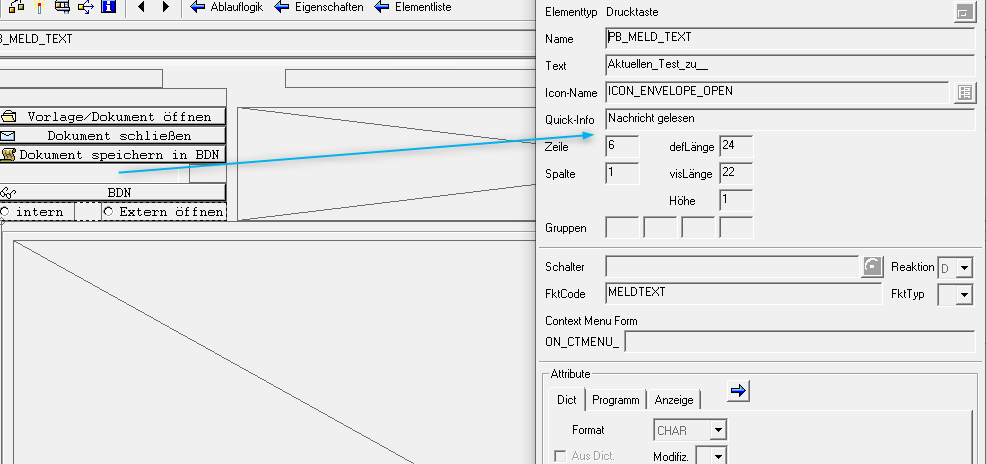
*Class and Methods and Data declarations are in TOP include ZXQQMTOP  
\*&---------------------------------------------------------------------\**  
*\*& Include ZXQQMTOP*  
*\*&---------------------------------------------------------------------\**  
  
TYPE-POOLS: slis.  
CONSTANTS:  
 gc\_anwendung TYPE /datrain/al\_de\_anwendung VALUE 'KC',  
 gc\_netto(1) VALUE 'N',  
 gc\_brutto(1) VALUE 'B'.  
  
DATA lt\_sel\_rows TYPE lvc\_t\_row.  
DATA ls\_sel\_rows LIKE LINE OF lt\_sel\_rows.  
  
DATA lt\_rows TYPE lvc\_t\_row.  
DATA ls\_selected\_line TYPE lvc\_s\_row.  
DATA lv\_row\_index TYPE lvc\_index.  
  
DATA: gv\_fehler TYPE c,  
 gv\_qmnum TYPE qmnum.  
  
DATA: gv\_aktyp TYPE akttyp,  
 gv\_appid TYPE /datrain/al\_de\_appid,  
 gs\_scr\_prop TYPE /datrain/al\_st\_scr\_prop,  
 gs\_infl\_data TYPE zdt\_st\_infl\_data\_xqqm,  
 gv\_fld\_name(30), *" Feldname für get cursor field*  
 gv\_hwp\_ekorg TYPE ekorg,  
 gv\_lif TYPE parvw,  
*\* Versicherungsfall*  
 gv\_ver TYPE ila\_ver,  
  
 gv\_gar TYPE ila\_gar, *" Garantiefall*  
 gv\_mbl TYPE ila,  
 gv\_lv\_flag TYPE flag, *" LV eingebunden*  
 gv\_fobl\_flag TYPE flag, *" Folgebelege bereits generiert*  
 gv\_init TYPE flag,  
 gv\_hwp\_flag TYPE flag, *" Handwerker ist portalintegriert*  
 gv\_set\_flag TYPE boolean,  
 gv\_portal\_lifnr TYPE boolean,  
 gv\_old\_ila TYPE ila,  
 gv\_amt\_gross\_old TYPE dmbrt, *" Zwischenspeicherung des vorherigen Bruttowertes*  
 gv\_vera\_flag TYPE flag, *" VERA-Position zugeordnet*  
 gv\_info TYPE text50,  
 gv\_reinfo TYPE text100,  
 gv\_recninfo TYPE recntxt,  
 gv\_recninfo\_mbl TYPE recntxt,  
 gv\_snunr TYPE snunr,  
 gv\_rexmbez TYPE xmbez,  
 gv\_chr\_part TYPE text50,  
 gv\_wgbez TYPE wgbez,  
 gv\_chr\_ilatx TYPE ilatx,  
 gv\_smart TYPE qmart,  
 gv\_sinstbez TYPE sinstbez,  
 gv\_fb\_psp\_name TYPE funcname,  
 gv\_parvw\_rg TYPE parvw,  
 gv\_parvw\_re TYPE parvw,  
 gv\_dh\_role\_type TYPE rebprole,  
 gv\_lifnr\_old TYPE lifnr,  
 gv\_tplnr\_old TYPE tplnr,  
 gv\_manum TYPE /datrain/vl\_de\_manum,  
 gv\_identkey TYPE recaidentkey,  
 gv\_lelim TYPE wrbtr,  
  
 gs\_xihpa TYPE ihpavb,  
 gs\_redata TYPE /datrain/fl\_st\_redata,  
 gs\_liflst TYPE /datrain/kc\_st\_lifdata,  
 gs\_mat TYPE /datrain/kc\_st\_mat,  
 gs\_stlvpos TYPE /datrain/sc\_st\_stlvlist,  
 gs\_return TYPE bapiret2,  
 gs\_comm\_data TYPE /datrain/pa\_st\_comm\_data,  
 gs\_po\_buff TYPE /datrain/po\_buff,  
  
 gt\_liflst TYPE TABLE OF /datrain/kc\_st\_lifdata,  
 gt\_mat TYPE TABLE OF /datrain/kc\_st\_mat,  
 gt\_stlvpos TYPE TABLE OF /datrain/sc\_st\_stlvlist,  
 gt\_return TYPE bapiret2\_t,  
  
*\* Versicherungsparameter*  
 gt\_vers TYPE TABLE OF /datrain/vl\_st\_f4,  
  
  
  
 gx\_relation TYPE REF TO /datrain/fl\_cl\_ba\_relation,  
 gx\_log TYPE REF TO /datrain/al\_cl\_log,  
 gx\_partner TYPE REF TO /datrain/pa\_cl\_partner.  
  
*\* BEKO-Bestellungen*  
DATA: gs\_beko TYPE zdt\_beko,  
 gt\_su TYPE /datrain/bb\_tt\_su,  
 gv\_txt\_sempsl1 TYPE rescxsu,  
 gv\_txt\_sempsl2 TYPE rescxsu.  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
*\* Danilo Bock, Datatrain GmbH 07.01.16*  
*\* Erweiterungen für mobile Mieterwechselprozesse*  
*\* Meldungsart MW*  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
DATA: gv\_mv\_intreno TYPE recaintreno,  
 gv\_mv\_partner TYPE bu\_partner,  
 gs\_vicncn TYPE vicncn,  
 gs\_contract TYPE bapi\_re\_contract\_int,  
 gs\_contract\_object TYPE bapi\_re\_object\_rel\_int,  
 gt\_contract\_object TYPE bapi\_re\_t\_object\_rel\_int,  
 gv\_ntreason TYPE recnxntreason,  
 gv\_visit\_date TYPE qmdat,  
 gv\_visit\_date\_txt TYPE char40,  
 gv\_visit\_time TYPE mzeit,  
 gv\_visit\_time\_txt TYPE char40,  
 gv\_datetext TYPE char40,  
 gt\_partner TYPE TABLE OF bapi\_re\_partner,  
 gs\_tenant\_comm\_data TYPE zdt\_wa\_st\_tenant\_comm\_data.  
  
CONSTANTS:  
 gc\_anwendung\_wa TYPE /datrain/al\_de\_anwendung VALUE 'WA',  
 gc\_appid\_wa TYPE /datrain/al\_de\_appid VALUE 'DK'.  
  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
*\* O.Meyer(Begis) 02.18*  
*\* /AFM/Gewährleistung - Meldungsarten Z4; ZG; ZM*  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\**  
*\* Dynpro 0104*  
*\*\*\*\*\*\*\*\*\*\*\*\*\**  
DATA z\_qmtxt LIKE qmel-qmtxt.  
DATA z\_txz01 LIKE ekpo-txz01.  
  
*\*Variables für die Berechnung des Gewährleistungsbetrags.*  
DATA gv\_effrwert TYPE ekpo-effwr.  
  
DATA wa\_ekpo TYPE bapiekpo.  
DATA it\_ekpo TYPE TABLE OF bapiekpo.  
  
DATA flag\_gew TYPE c LENGTH 1.  
DATA fl\_chkba TYPE c LENGTH 1.  
DATA fl\_chkgw TYPE c LENGTH 1.  
  
DATA gv\_parnr TYPE i\_parnr.  
  
*\* DATA z\_act LIKE t365-aktyp.*  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\**  
*\* Dynpro 9000*  
*\*\*\*\*\*\*\*\*\*\*\*\*\**  
TYPE-POOLS soi.  
TYPE-POOLS vrm.  
  
INCLUDE <icon>.  
  
*\*CONSTANTS c\_class TYPE bds\_clsnam VALUE 'ZGWL\_MAENGELMELDUNG'.*  
*\*CONSTANTS c\_cltyp TYPE bds\_clstyp VALUE 'OT'.*  
*\*CONSTANTS c\_objkey TYPE bds\_typeid VALUE 'KORRESPONDENZVORLAGEN'.*  
  
  
CONSTANTS c\_class\_mangel1 TYPE bds\_clsnam VALUE 'ZKORR\_MAENGEL\_1'.  
CONSTANTS c\_cltyp\_mangel1 TYPE bds\_clstyp VALUE 'OT'.  
CONSTANTS c\_objkey\_mangel1 TYPE bds\_typeid VALUE 'ZKORR\_MAENGEL\_1'.  
CONSTANTS c\_class\_mangel2 TYPE bds\_clsnam VALUE 'ZKORR\_MAENGEL\_2'.  
CONSTANTS c\_cltyp\_mangel2 TYPE bds\_clstyp VALUE 'OT'.  
CONSTANTS c\_objkey\_mangel2 TYPE bds\_typeid VALUE 'ZKORR\_MAENGEL\_2'.  
CONSTANTS c\_class\_mangel\_info\_1 TYPE bds\_clsnam VALUE 'ZKORR\_MAENGEL\_INFO\_1'.  
CONSTANTS c\_cltyp\_mangel\_info\_1 TYPE bds\_clstyp VALUE 'OT'.  
CONSTANTS c\_objkey\_mangel\_info\_1 TYPE bds\_typeid VALUE 'ZKORR\_MAENGEL\_INFO\_1'.  
CONSTANTS c\_class\_mangel\_info\_2 TYPE bds\_clsnam VALUE 'ZKORR\_MAENGEL\_INFO\_2'.  
CONSTANTS c\_cltyp\_mangel\_info\_2 TYPE bds\_clstyp VALUE 'OT'.  
CONSTANTS c\_objkey\_mangel\_info\_2 TYPE bds\_typeid VALUE 'ZKORR\_MAENGEL\_INFO\_2'.  
  
CONSTANTS c\_classmass TYPE bds\_clsnam VALUE 'ZMASSNAHME\_KORRESPONDENZ'.  
CONSTANTS c\_cltypmass TYPE bds\_clstyp VALUE 'OT'.  
CONSTANTS c\_objkeymass TYPE bds\_typeid VALUE 'MASSNAHME'.  
CONSTANTS c\_zweitemass TYPE bds\_typeid VALUE 'MASSNAHME\_ZWEITE\_FRIST'.  
  
*\*Dokument ID für Mängelmeldung Maßnahme Vorlagen*  
CONSTANTS c\_sm1frmass TYPE bds\_typeid VALUE '1FR'.  
CONSTANTS c\_sm2frmass TYPE bds\_typeid VALUE '2FR'.  
CONSTANTS c\_smersmass TYPE bds\_typeid VALUE 'ERS'.  
CONSTANTS c\_smuermass TYPE bds\_typeid VALUE 'UER'.  
  
  
DATA gv\_newnam TYPE bds\_typeid. *"Nimmt den neuen Objektschlüssel auf, wenn geänderte Vorlage*  
*"zur Meldung gespeichert werden soll*  
  
  
*\* Globale Instanzvariablen Office*  
DATA go\_container TYPE REF TO cl\_gui\_custom\_container.  
DATA go\_control TYPE REF TO i\_oi\_container\_control.  
DATA go\_document TYPE REF TO i\_oi\_document\_proxy.  
*\*DATA go\_spreadsheet TYPE REF TO i\_oi\_spreadsheet.*  
DATA go\_worddoc TYPE REF TO i\_oi\_word\_processor\_document. *"Word-Interface*  
DATA go\_formular TYPE REF TO i\_oi\_form. *"Formular-Interface*  
  
*\* Globale Instanzvariablen ALV-Grid Vorlagenauswahl*  
DATA go\_cont\_alv TYPE REF TO cl\_gui\_custom\_container.  
DATA go\_alv TYPE REF TO cl\_gui\_alv\_grid.  
  
*\* Struktur der im Grid anzuzeigenden Felder bzgl. Vorlagendateien*  
DATA: BEGIN OF gs\_vorl,  
 doc\_id TYPE bds\_docid,  
 prop\_name TYPE bds\_propna,  
 prop\_value TYPE bds\_propva,  
 comp\_id TYPE bds\_compid,  
 uri TYPE bds\_uri,  
 datum(10) TYPE c,  
 type TYPE bds\_clsnam,  
 doc(1) TYPE c,  
 END OF gs\_vorl.  
DATA gt\_vorl LIKE TABLE OF gs\_vorl.  
DATA ls\_vorl LIKE LINE OF gt\_vorl.  
  
*\* Daten bzgl. Feldkatalog/Layout/Variante*  
DATA gs\_fcat TYPE lvc\_s\_fcat.  
DATA gt\_fcat TYPE lvc\_t\_fcat.  
DATA gs\_outtab LIKE gs\_vorl.  
DATA gt\_outtab LIKE gt\_vorl.  
DATA gs\_layout TYPE lvc\_s\_layo.  
DATA gs\_variant TYPE disvariant.  
  
DATA gs\_signature TYPE bapisignat.  
DATA gt\_signatures TYPE STANDARD TABLE OF bapisignat.  
DATA gt\_sign\_vorl TYPE STANDARD TABLE OF bapisignat.  
DATA gs\_sign\_vorl TYPE bapisignat.  
DATA gs\_component TYPE bapicompon.  
DATA gt\_components TYPE STANDARD TABLE OF bapicompon.  
DATA gt\_components\_upd TYPE STANDARD TABLE OF bapicompon.  
DATA gs\_uri TYPE bapiuri.  
DATA gt\_uris TYPE STANDARD TABLE OF bapiuri.  
DATA gt\_uris\_vorl TYPE STANDARD TABLE OF bapiuri.  
DATA gs\_properties TYPE sbdst\_properties.  
DATA gt\_properties TYPE STANDARD TABLE OF sbdst\_properties.  
  
DATA lt\_uris2 TYPE STANDARD TABLE OF bapiuri.  
*\* DATA ls\_uri TYPE bapiuri.*  
DATA lv\_object\_key2 TYPE bapibds01-objkey.  
*\* DATA lt\_data2 TYPE STANDARD TABLE OF bapiconten.*  
*\* DATA lv\_doc\_size2 TYPE i.*  
DATA lt\_signature2 TYPE STANDARD TABLE OF bapisignat.  
DATA ls\_signature2 TYPE bapisignat.  
*\* DATA lt\_sign\_copy2 TYPE STANDARD TABLE OF bapisignat.*  
*\* DATA ls\_sign\_copy TYPE bapisignat.*  
DATA lt\_sign\_sort\_r2 TYPE STANDARD TABLE OF bapisignat WITH HEADER LINE.  
*\* DATA lt\_sign\_sort\_v TYPE STANDARD TABLE OF bapisignat WITH HEADER LINE.*  
DATA lt\_components2 TYPE STANDARD TABLE OF bapicompon.  
DATA ls\_component2 TYPE bapicompon.  
  
DATA gv\_url TYPE bds\_uri.  
DATA gv\_objektschl TYPE bapibds01-objkey. *"Objektschlüssel*  
  
DATA gv\_doc\_open. *"Kennzeichen Dok. geöffnet*  
DATA gv\_is\_open TYPE i.  
DATA gv\_changed TYPE i. *"Dokument geändert?*  
DATA gv\_selected. *"Kennzeichen Dokument wurde ausgewählt*  
DATA gv\_get\_vorl. *"Kennzeichen Vorlagen eingelesen*  
DATA gv\_inplace VALUE '1'. *"Kennzeichen intern oder im externen Modus öffnen*  
DATA rb\_inplace VALUE 'X'.  
DATA rb\_outplace.  
DATA gv\_exist.  
DATA gv\_version(10) TYPE c.  
DATA gv\_docnr(1) TYPE n. *"Zähler der Dokumentvorlagen: Rüge1 , Rüge2 etc.*  
DATA gv\_ruege TYPE fieldname. *"Listbox Rüge-Nr.-Auswahl*  
DATA gv\_sel\_vorl TYPE bds\_dcount. *"doc\_count der ausgewählten Vorlage*  
DATA gv\_vorl. *"Kennzeichen: eine Vorlage wurde geöffnet*  
DATA gv\_saved TYPE c LENGTH 1.  
  
DATA gt\_fields TYPE STANDARD TABLE OF soi\_form\_item.  
  
DATA gv\_string\_new(50) TYPE c.  
DATA gv\_string\_old(50) TYPE c.  
  
TYPES: BEGIN OF t\_repl,  
 string\_new(150) TYPE c,  
 string\_old(150) TYPE c,  
 END OF t\_repl.  
  
DATA gt\_replaces TYPE TABLE OF t\_repl WITH HEADER LINE.  
DATA gs\_repl TYPE t\_repl.  
  
DATA gt\_param TYPE TABLE OF /afm/formpar.  
DATA gs\_param TYPE /afm/formpar.  
  
DATA gt\_mgpar TYPE TABLE OF /afm/formpar.  
DATA gs\_mgpar TYPE /afm/formpar.  
  
DATA gt\_gmpar TYPE TABLE OF /afm/formpar.  
DATA gs\_gmpar TYPE /afm/formpar.  
  
*\*----globale Variable Typ QMART um die QMART Bezeichnung zwischen Code und*  
*\*----Kunden zu entkopplen*  
DATA gv\_qmart TYPE qmart.  
  
*\*Flags für die Maßnahme Details dynamische Anzeige*  
DATA fl\_massn TYPE i.  
DATA fl\_txtma TYPE i.  
*\*Globale Variablen für die Maßnahme Details*  
DATA gv\_txtma TYPE wqmsm-matxt.  
DATA gv\_txcdsm TYPE qtxtcdgrsm.  
DATA gv\_mncod TYPE wqmsm-mncod.  
  
TABLES qpct.  
  
DATA gt\_oaer TYPE TABLE OF /afm/oaer.  
DATA gs\_oaer TYPE /afm/oaer.  
  
DATA: BEGIN OF bdcdata OCCURS 0.  
 INCLUDE STRUCTURE bdcdata.  
DATA: END OF bdcdata.  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\**  
*\* Dynpro 9020*  
*\*\*\*\*\*\*\*\*\*\*\*\*\**  
DATA: g\_aktyp TYPE t365-aktyp. *"Aufrufmodus der Transaktion für Q-Meldungen*  
DATA flag\_mglm TYPE c LENGTH 1.  
  
*\* Globale Instanzvariablen ALV-Grid BA Meldung in Gewähleistung Tabreiter*  
DATA go\_gw\_cont\_alv TYPE REF TO cl\_gui\_custom\_container.  
DATA go\_gw\_alv TYPE REF TO cl\_gui\_alv\_grid.  
DATA go\_gw\_toolbar TYPE REF TO cl\_gui\_toolbar.  
DATA gt\_toolbar TYPE ui\_functions.  
DATA gt\_layo TYPE lvc\_s\_layo.  
  
DATA gs\_gw\_fcat TYPE lvc\_s\_fcat.  
DATA gt\_gw\_fcat TYPE lvc\_t\_fcat.  
  
*\*Itab und wa für die ALV Grid BA Meldung in Gewähleistung Tabreiter*  
TYPES: BEGIN OF ty\_zgew\_alv.  
 INCLUDE TYPE zst\_afm\_zgew\_alv.  
TYPES: icon TYPE icon-id,  
 END OF ty\_zgew\_alv.  
  
DATA gt\_gw\_fields TYPE TABLE OF ty\_zgew\_alv.  
DATA gs\_gw\_fields LIKE LINE OF gt\_gw\_fields.  
  
DATA gw\_meldnumm TYPE qmnum.  
  
DATA gt\_rowid TYPE lvc\_t\_row.  
DATA gs\_rowid TYPE lvc\_s\_row.  
  
*\*Wenn den Anwender die neue Mängel Meldung nicht speichert, das entsprechende*  
*\*Feld in Gewährleistung Tabreiter (BA Meldung) sollte leer sein.*  
DATA gv\_oldmel TYPE qmnum.  
DATA gv\_newmel TYPE qmnum.  
*\*Flag für die Zurücksetzen Funktionalität, um die Mängemeldung zu schließen*  
DATA fl\_rueck TYPE c LENGTH 1.  
*\*Work Area um die Daten von die Mängelmeldung zu speichern*  
DATA gs\_maengl TYPE viqmel.  
  
*\*Hilfestrukturen um die Maßnahmentabelle und die markierte Maßnahme*  
*\*von Mängelmeldung - Fristsetzung Tabreiter zu holen*  
DATA: BEGIN OF inxsm OCCURS 10.  
 INCLUDE STRUCTURE inxx.  
DATA: END OF inxsm.  
  
DATA: BEGIN OF iviqmsm OCCURS 10.  
 INCLUDE STRUCTURE wqmsm.  
DATA: END OF iviqmsm.  
  
*\*----------------------------------------------------------------------\**  
*\* CLASS lcl\_gui\_alv\_grid - D E F I N I T I O N*  
*\*----------------------------------------------------------------------\**  
  
CLASS lcl\_gui\_alv\_grid DEFINITION.  
  
 PUBLIC SECTION.  
  
 CLASS-METHODS on\_toolbar FOR EVENT toolbar OF cl\_gui\_alv\_grid  
 IMPORTING e\_object.  
  
 CLASS-METHODS on\_doubleclick FOR EVENT double\_click OF cl\_gui\_alv\_grid  
 IMPORTING es\_row\_no.  
 CLASS-METHODS on\_click\_row FOR EVENT hotspot\_click OF cl\_gui\_alv\_grid  
 IMPORTING es\_row\_no.  
 CLASS-METHODS on\_usercommand FOR EVENT user\_command OF cl\_gui\_alv\_grid  
 IMPORTING e\_ucomm.  
  
ENDCLASS. *"lcl\_gui\_alv\_grid DEFINITION*  
  
*\*----------------------------------------------------------------------\**  
*\* CLASS lcl\_gui\_alv\_grid - I M P L E M E N T A T I O N*  
*\*----------------------------------------------------------------------\**  
CLASS lcl\_gui\_alv\_grid IMPLEMENTATION.  
  
 METHOD on\_toolbar.  
  
 DATA l\_wa\_button TYPE stb\_button.  
  
 l\_wa\_button-function = 'ZUR'.  
 l\_wa\_button-icon = icon\_refresh.  
 l\_wa\_button-butn\_type = cntb\_btype\_button.  
 l\_wa\_button-text = 'Zurücksetzen'.  
  
 INSERT l\_wa\_button INTO TABLE e\_object->mt\_toolbar.  
  
 ENDMETHOD. *"on\_toolbar*  
  
 METHOD on\_doubleclick.  
 DATA ls\_gw\_fields TYPE zst\_afm\_zgew\_alv.  
 READ TABLE gt\_gw\_fields INTO ls\_gw\_fields INDEX es\_row\_no-row\_id.  
  
 IF sy-subrc NE 0.  
 EXIT.  
 ENDIF.  
  
 gw\_meldnumm = ls\_gw\_fields-qmnum.  
*\**  
 CALL METHOD cl\_gui\_cfw=>set\_new\_ok\_code  
 EXPORTING  
 new\_code = 'FWD'.  
  
 ENDMETHOD. *"on\_doubleclick*  
  
 METHOD on\_click\_row.  
  
 DATA ls\_gw\_fields1 TYPE zst\_afm\_zgew\_alv.  
 READ TABLE gt\_gw\_fields INTO ls\_gw\_fields1 INDEX es\_row\_no-row\_id.  
  
 IF sy-subrc NE 0.  
 EXIT.  
 ENDIF.  
 CLEAR gs\_rowid.  
 CLEAR gt\_rowid.  
 gs\_rowid-index = es\_row\_no-row\_id.  
 APPEND gs\_rowid TO gt\_rowid.  
 gw\_meldnumm = ls\_gw\_fields1-qmnum.  
*\**  
 CALL METHOD cl\_gui\_cfw=>set\_new\_ok\_code  
 EXPORTING  
 new\_code = 'GEW'.  
  
 ENDMETHOD. *"on\_click\_row*  
  
 METHOD on\_usercommand.  
  
*\* IF viqmel-z\_smmeldnummer IS NOT INITIAL.*  
 IF viqmel-zz\_afm\_maengelmeldnummer IS NOT INITIAL.  
 CALL METHOD cl\_gui\_cfw=>set\_new\_ok\_code  
 EXPORTING  
 new\_code = 'ARCH1'.  
 ELSE.  
 CALL METHOD cl\_gui\_cfw=>set\_new\_ok\_code  
 EXPORTING  
 new\_code = 'ZUR'.  
 ENDIF.  
 ENDMETHOD.  
 *"on\_usercommand*  
ENDCLASS. *"lcl\_gui\_alv\_grid IMPLEMENTATION*

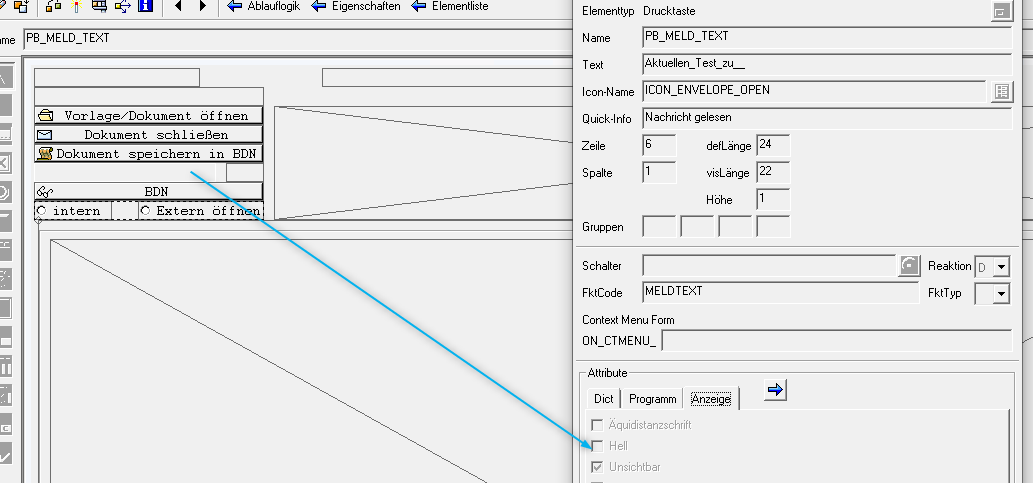
*Bildschirm🡪9000*

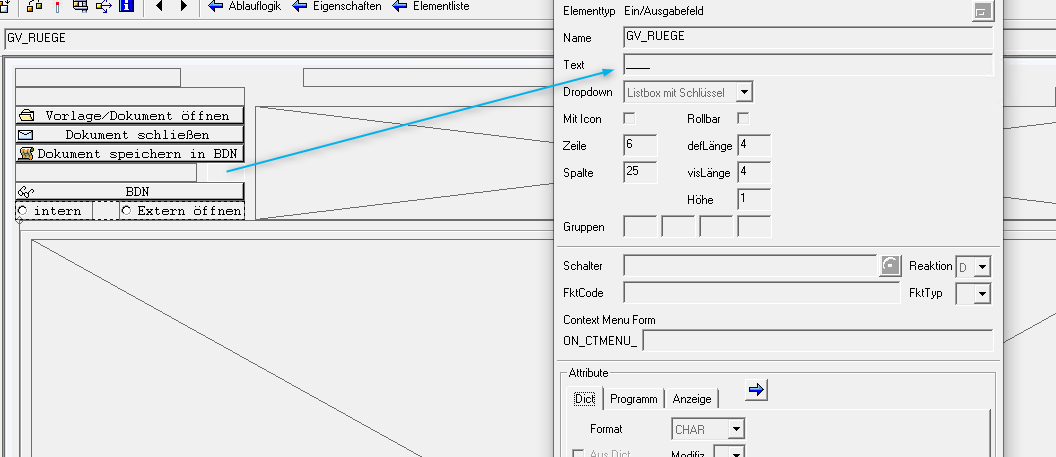
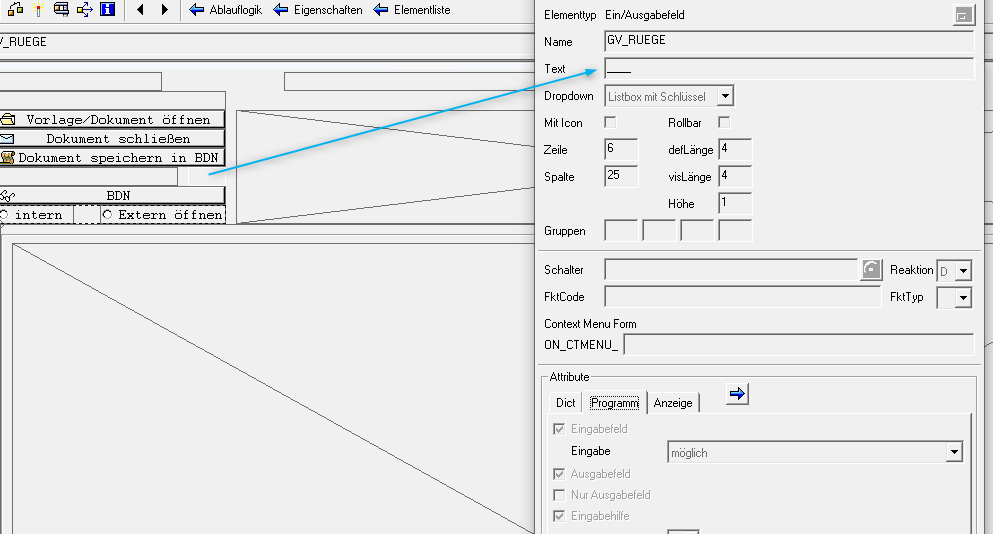
**

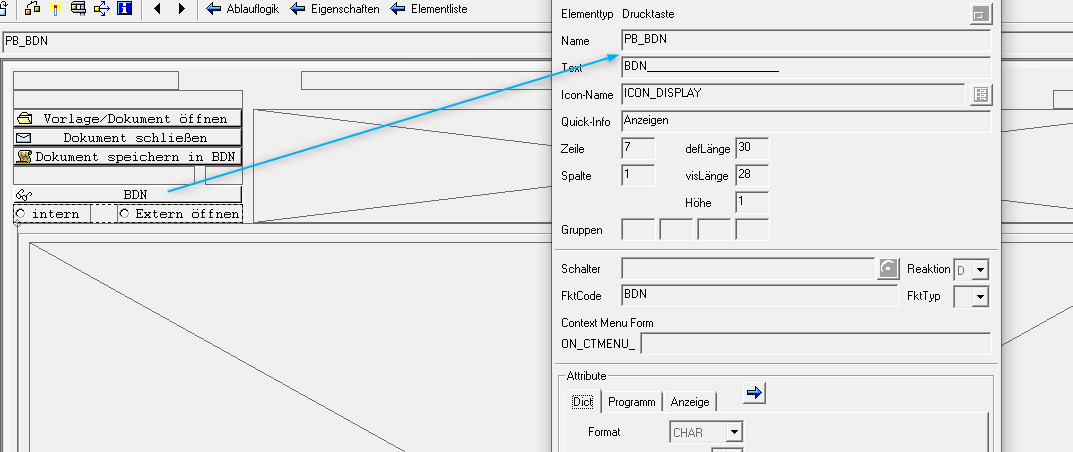
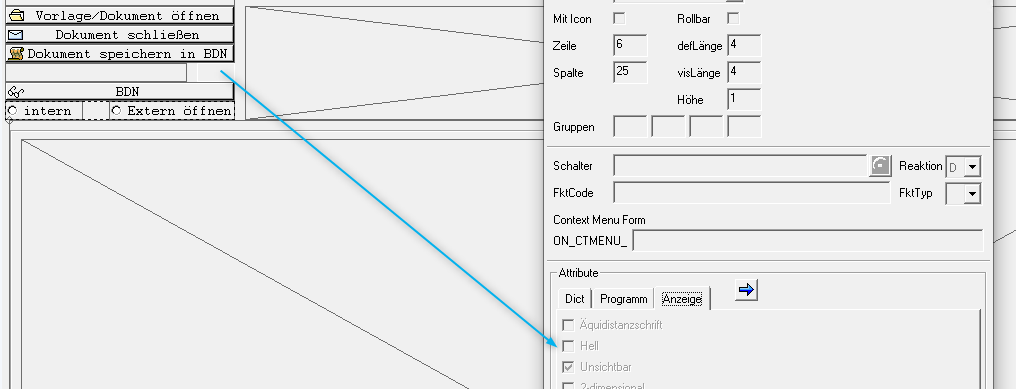
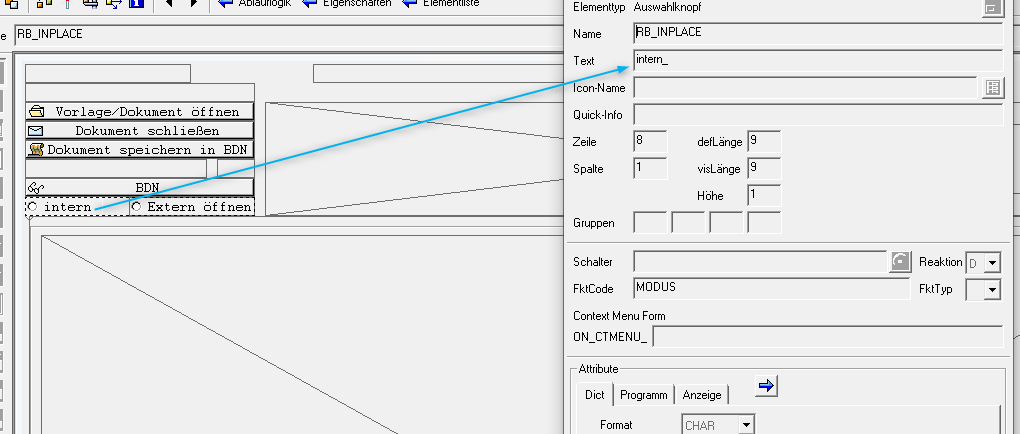
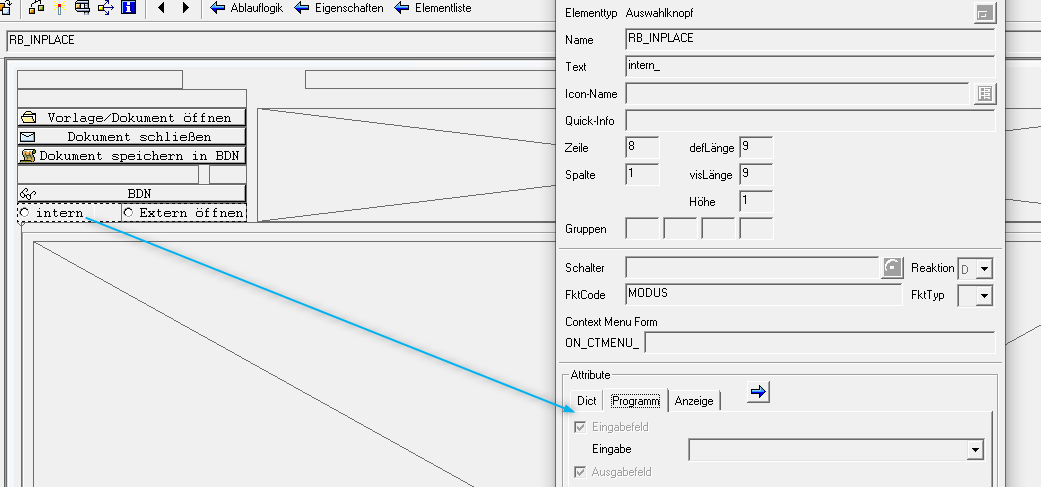
**   

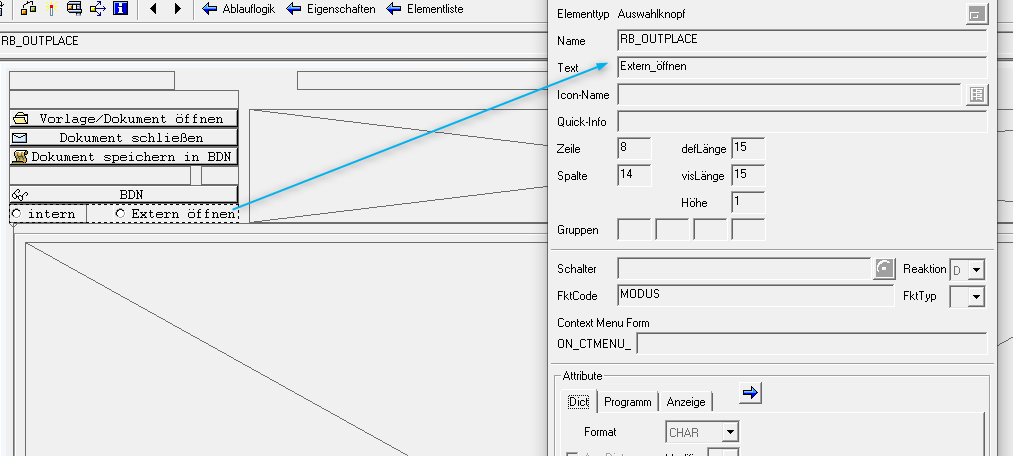
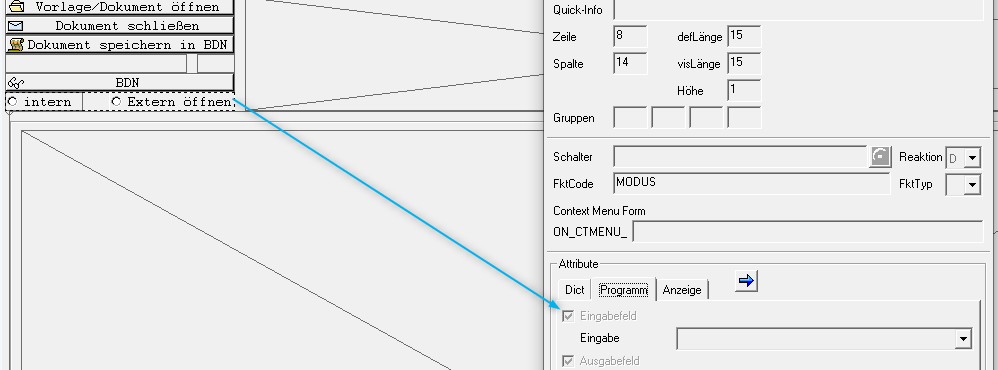
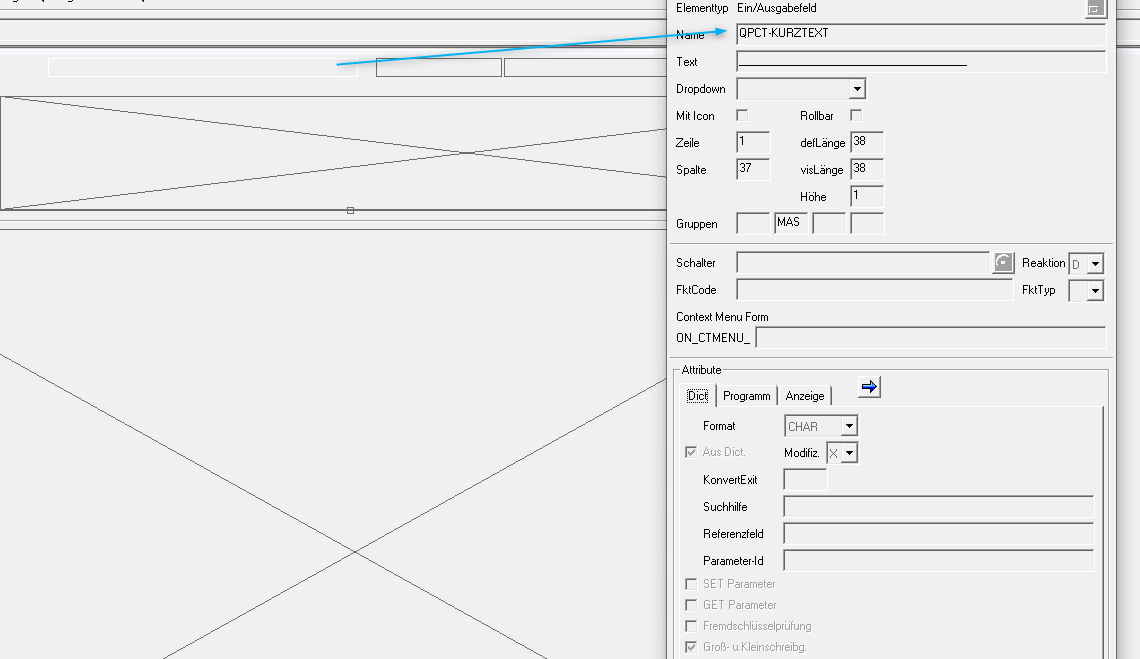


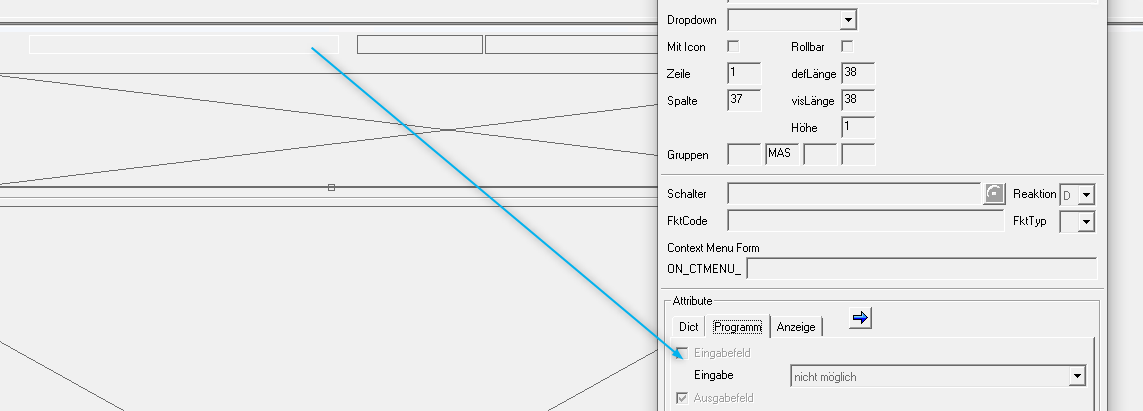
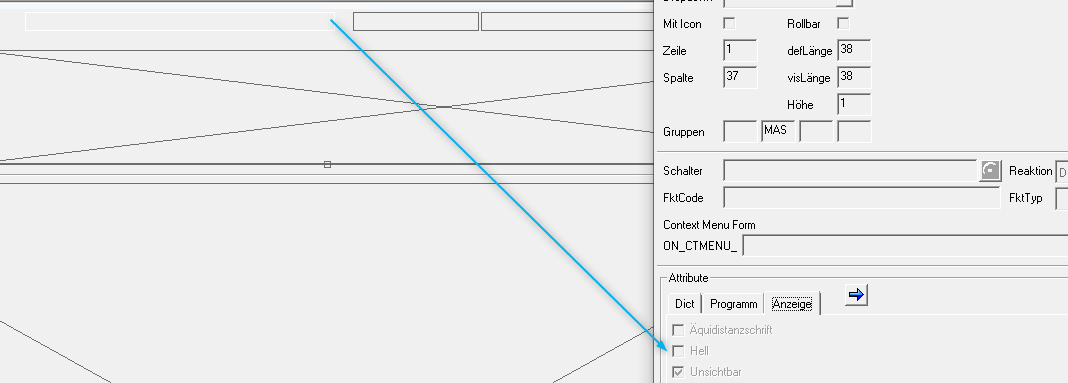
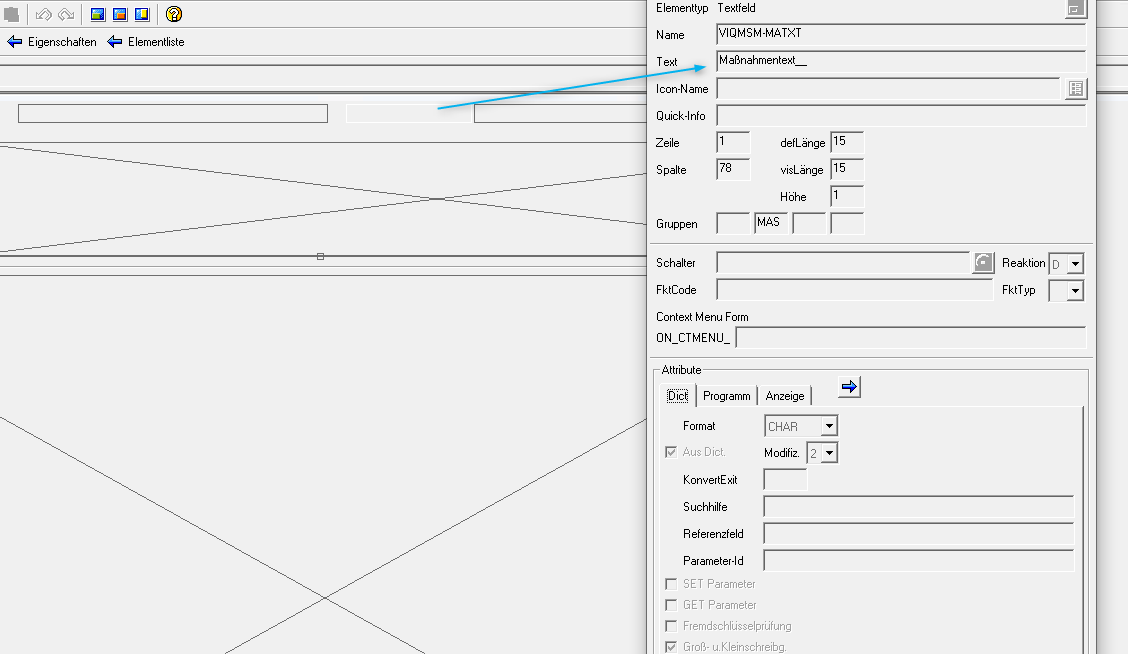
 

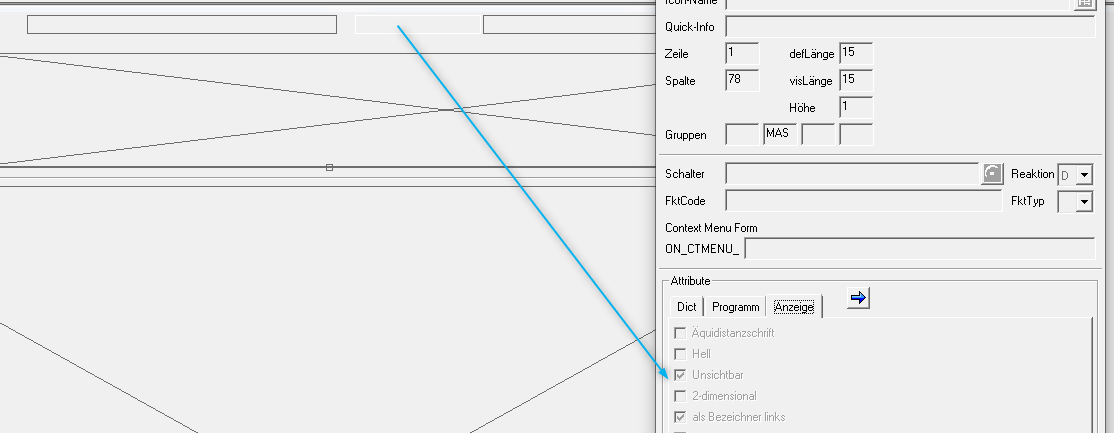
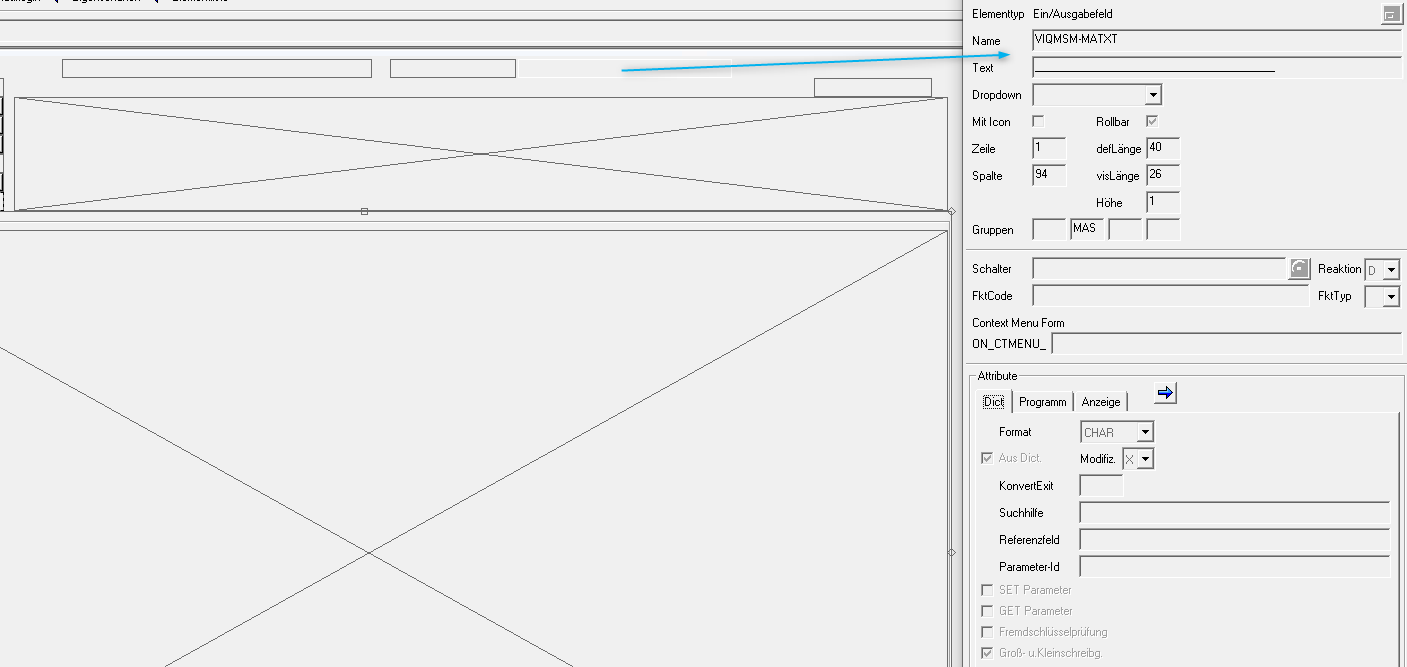
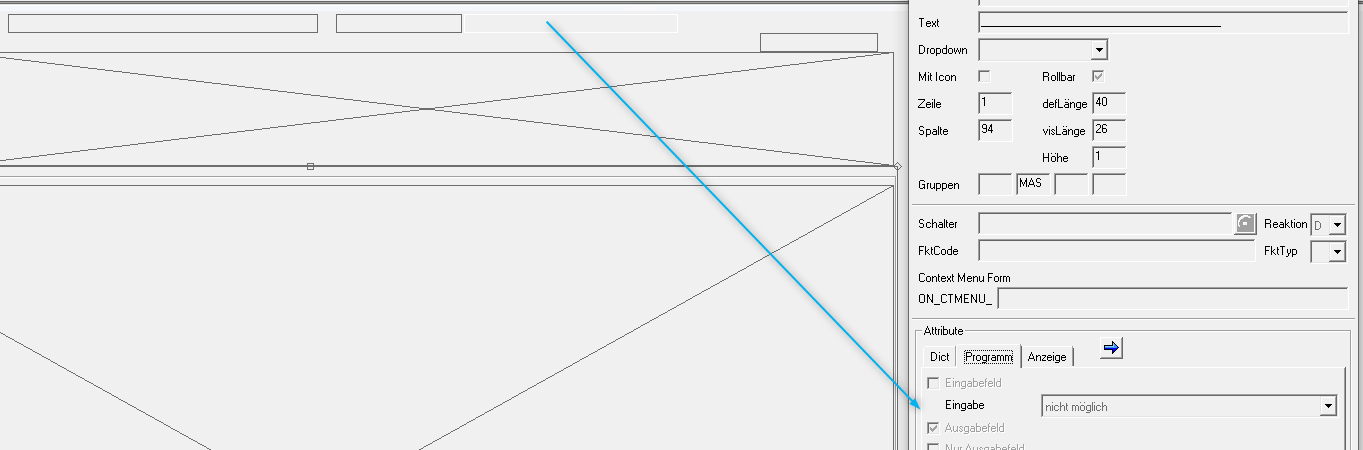
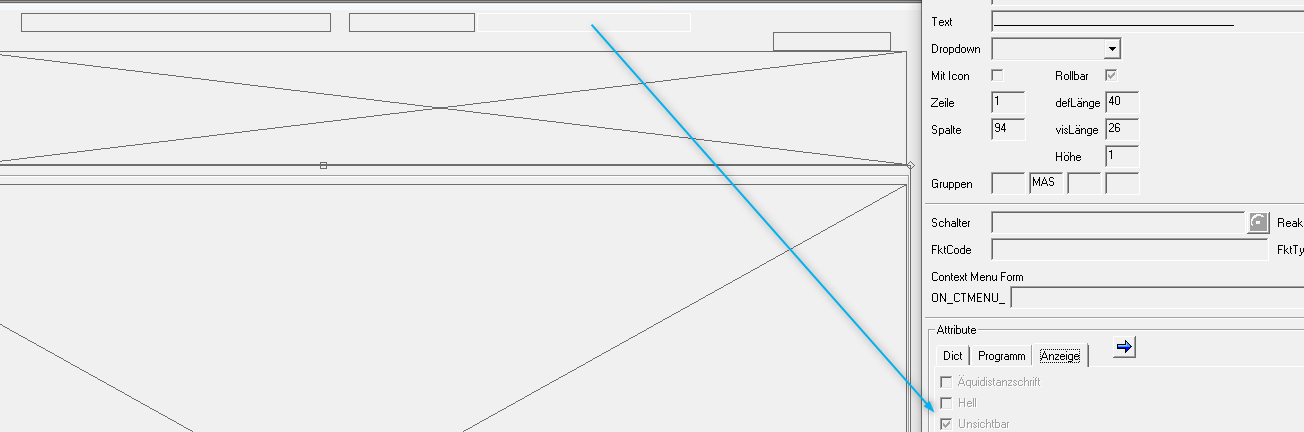


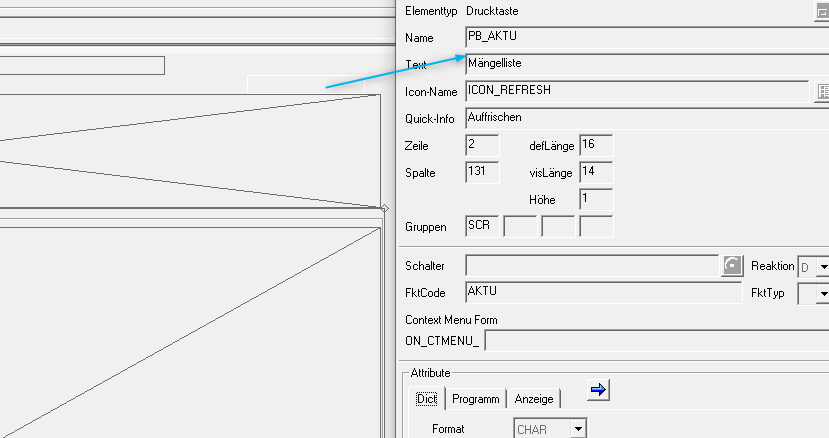
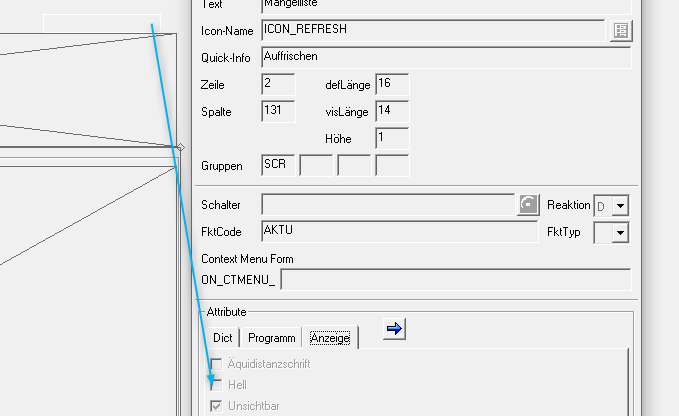
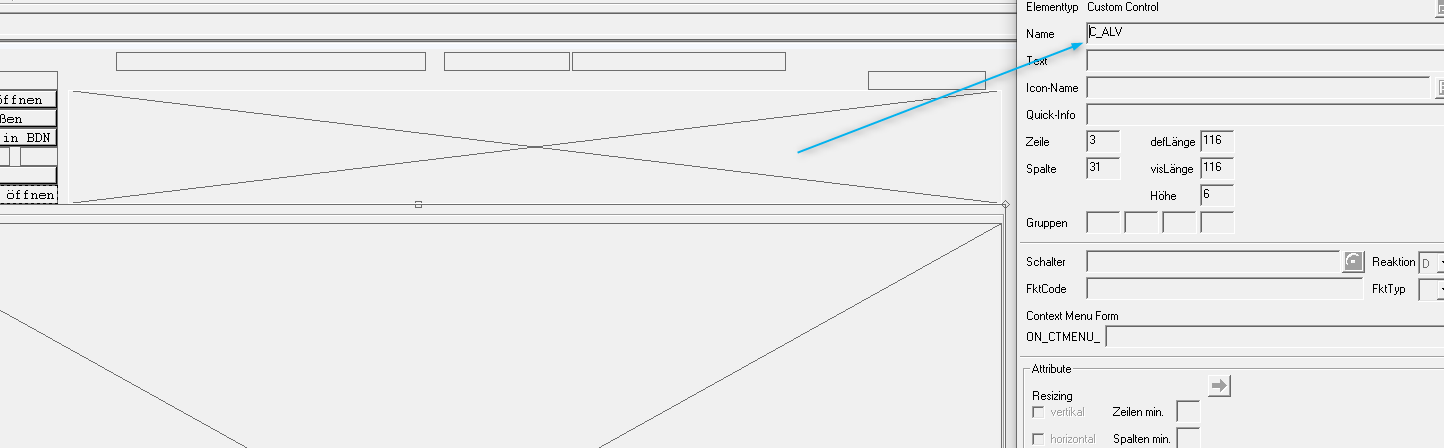
 

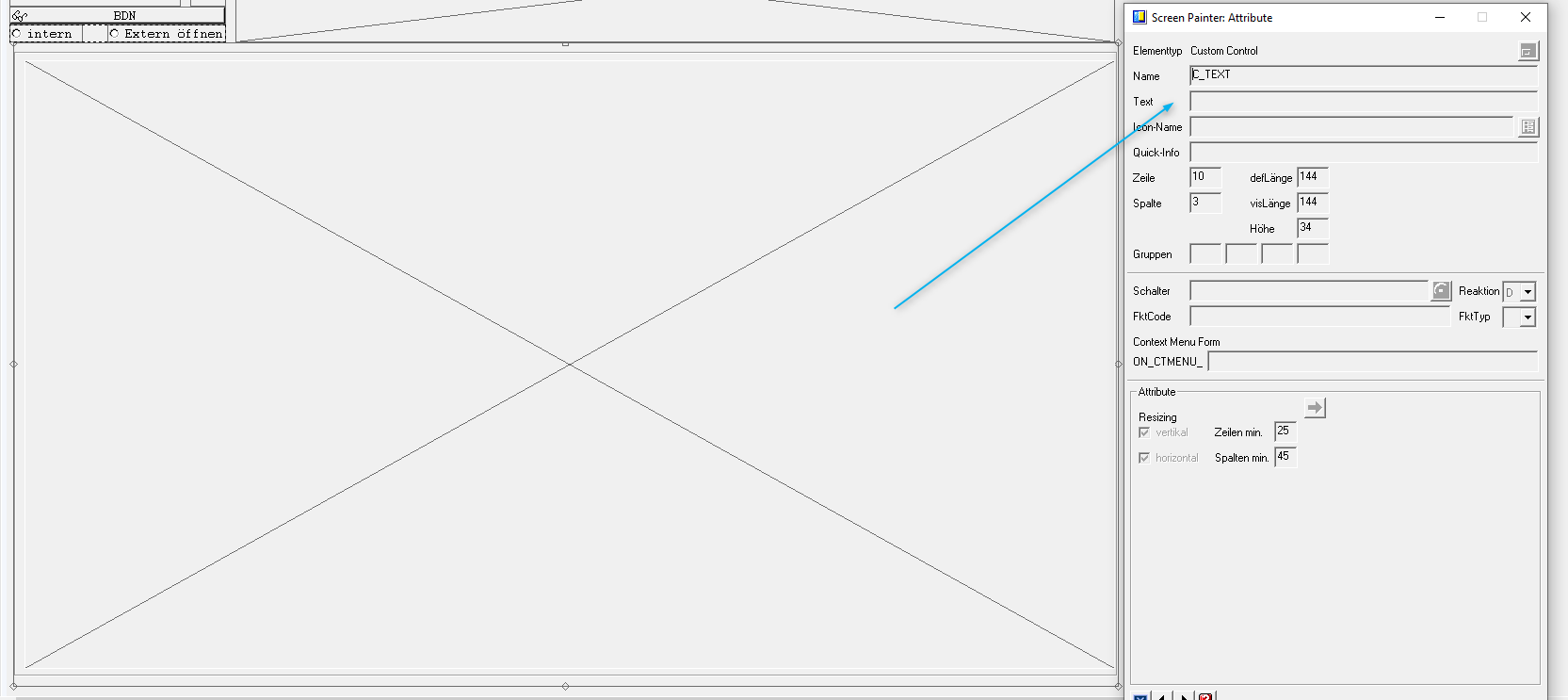
  

   
PROCESS BEFORE OUTPUT.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
 MODULE status\_9000. *" <-- LOAD\_DOCUMENT*  
  
 MODULE alv.  
  
 MODULE fieldcat.  
  
 MODULE display.  
  
  
PROCESS AFTER INPUT.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
 MODULE selekt\_vorlage.  
  
 MODULE user\_command\_9000.  
  
  
PROCESS ON VALUE-REQUEST.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
 FIELD gv\_ruege MODULE create\_auswahl.

Include🡪 ZXQQMO08

*\*&---------------------------------------------------------------------\**  
*\*& Module STATUS\_9000 OUTPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE status\_9000 OUTPUT.  
  
*\* SET PF-STATUS 'xxxxxxxx'.*  
*\* SET TITLEBAR 'xxx'.*  
  
*\*&---------------------------------------------------------------------\**  
  
 DATA lv\_error TYPE c.  
 IMPORT lv\_error FROM MEMORY ID 'ZZ\_ERROR'.  
 IF sy-subrc = 0.  
*\* MESSAGE i000(/afm/bau) WITH 'Achtung: ' 'Bitte die Mängelmeldung ' viqmel-qmnum ' erst speichern!'.*  
 DATA lv\_txt1 TYPE sta\_text.  
 CLEAR lv\_txt1.  
 gv\_qmnum = viqmel-qmnum.  
 SHIFT gv\_qmnum LEFT DELETING LEADING '0'.  
 CONCATENATE  
 'Bitte die Mängelmeldung:' gv\_qmnum 'erst speichern!' INTO lv\_txt1 SEPARATED BY space .  
 CALL FUNCTION 'POPUP\_TO\_INFORM'  
 EXPORTING  
 titel = 'Warnung: '  
 txt1 = lv\_txt1  
 txt2 = ''  
*\* TXT3 = ' '*  
*\* TXT4 = ' '*  
 .  
 gv\_fehler = 'X'.  
 EXIT.  
 ENDIF.  
  
*\*&---------------------------------------------------------------------\**  
  
*\*Entkopplung von den Kunden Meldungsarten Bezeichnungen*  
 SELECT SINGLE qmart FROM /afm/kundeqmart INTO gv\_qmart  
 WHERE /afm/kundeqmart~zkunde\_qmart = viqmel-qmart.  
  
 IF gv\_qmart = 'SG' OR gv\_qmart = 'SM'. *"Gewährleistung-/ Mängelmeldung*  
*\* Felder im Anzeigemodus auf dunkel schalten*  
 IF sy-tcode EQ 'IW53' OR sy-tcode EQ 'IW23'.  
 LOOP AT SCREEN.  
 IF screen-group1 EQ 'SCR'.  
 screen-input = 0.  
 ENDIF.  
 MODIFY SCREEN.  
 ENDLOOP.  
 ELSE.  
 ENDIF.  
  
*\*Holt von SAP Memory die Maßnahmen Tabelle und die ausgewählte Maßnahme*  
*\*um weiter dem entsprechenden Korrespondez Anwendungs Objekt von BDN zu zeigen*  
 CLEAR inxsm.  
 CLEAR iviqmsm.  
 CLEAR fl\_txtma.  
  
 IMPORT inxsm FROM MEMORY ID 'INXSM'.  
 IMPORT iviqmsm FROM MEMORY ID 'IVIQMSM'.  
  
*\*Übernimmt die Maßnahme Details von den entprechenden Tabellen*  
 PERFORM get\_massnahme\_details.  
  
 IF ( gv\_qmart = 'SM' OR gv\_qmart = 'SG' ) AND fl\_txtma = 1.  
 LOOP AT SCREEN.  
 IF screen-group2 EQ 'MAS'.  
 screen-invisible = 0.  
 screen-active = 1.  
 ENDIF.  
 MODIFY SCREEN.  
 ENDLOOP.  
 qpct-kurztext = gv\_txcdsm.  
 viqmsm-matxt = gv\_txtma.  
 ELSE.  
 LOOP AT SCREEN.  
 IF screen-group2 EQ 'MAS'.  
 screen-invisible = 1.  
 screen-active = 0.  
 ENDIF.  
 MODIFY SCREEN.  
 ENDLOOP.  
 qpct-kurztext = '' .  
 viqmsm-matxt = '' .  
 ENDIF.  
  
*\* Vorlagen holen*  
 IF gv\_get\_vorl EQ space.  
 PERFORM get\_template\_url CHANGING gv\_url.  
 ELSEIF ( gv\_get\_vorl NE space AND gv\_qmart = 'SM' ) OR ( gv\_get\_vorl NE space AND gv\_qmart = 'SG' ).  
 PERFORM get\_template\_url CHANGING gv\_url.  
 ENDIF.  
  
 IF gv\_init EQ space. *"nur beim ersten Aufruf*  
*\* Existiert bereits ein Dokument zur Meldungsnummer, soll diese direkt geladen werden:*  
 DATA lt\_uris TYPE STANDARD TABLE OF bapiuri.  
 DATA ls\_uri TYPE bapiuri.  
 DATA lv\_object\_key TYPE bapibds01-objkey.  
*\* DATA lt\_data TYPE STANDARD TABLE OF bapiconten.*  
 DATA lv\_doc\_size TYPE i.  
 DATA lt\_signature TYPE STANDARD TABLE OF bapisignat.  
 DATA ls\_signature TYPE bapisignat.  
 DATA lt\_sign\_copy TYPE STANDARD TABLE OF bapisignat.  
 DATA ls\_sign\_copy TYPE bapisignat.  
 DATA lt\_sign\_sort\_r TYPE STANDARD TABLE OF bapisignat WITH HEADER LINE.  
 DATA lt\_sign\_sort\_v TYPE STANDARD TABLE OF bapisignat WITH HEADER LINE.  
 DATA lt\_components TYPE STANDARD TABLE OF bapicompon.  
 DATA ls\_component TYPE bapicompon.  
  
 IF gv\_doc\_open NE 'X'.  
 CLEAR gt\_components\_upd.  
 REFRESH gt\_components\_upd.  
 CLEAR gt\_signatures.  
 REFRESH gt\_signatures.  
  
 lv\_object\_key = viqmel-qmnum.  
  
 CASE gv\_qmart.  
  
 WHEN 'SG'.  
 IF sy-tcode EQ 'IW22' OR sy-tcode EQ 'IW23'.  
 IF fl\_txtma = 1.  
 CLEAR lv\_object\_key.  
 CONCATENATE viqmel-qmnum gv\_mncod INTO lv\_object\_key.  
  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'  
 EXPORTING  
 classname = c\_classmass  
 classtype = c\_cltypmass  
*\* OBJECT\_KEY = gv\_objektschl*  
 object\_key = lv\_object\_key  
 TABLES  
 uris = lt\_uris  
 signature = lt\_signature  
 components = lt\_components  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 ELSE.  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'  
 EXPORTING  
 classname = c\_class\_mangel1  
 classtype = c\_cltyp\_mangel1  
*\* OBJECT\_KEY = gv\_objektschl*  
 object\_key = lv\_object\_key  
 TABLES  
 uris = lt\_uris  
 signature = lt\_signature  
 components = lt\_components  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 ENDIF.  
 ELSE.  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'  
 EXPORTING  
 classname = c\_class\_mangel1  
 classtype = c\_cltyp\_mangel1  
*\* OBJECT\_KEY = gv\_objektschl*  
 object\_key = lv\_object\_key  
 TABLES  
 uris = lt\_uris  
 signature = lt\_signature  
 components = lt\_components  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 ENDIF.  
  
 WHEN 'SM'.  
 IF sy-tcode EQ 'IW22' OR sy-tcode EQ 'IW23'.  
 IF fl\_txtma = 1.  
 CLEAR lv\_object\_key.  
 CONCATENATE viqmel-qmnum gv\_mncod INTO lv\_object\_key.  
  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'  
 EXPORTING  
 classname = c\_classmass  
 classtype = c\_cltypmass  
*\* OBJECT\_KEY = gv\_objektschl*  
 object\_key = lv\_object\_key  
 TABLES  
 uris = lt\_uris  
 signature = lt\_signature  
 components = lt\_components  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 ELSE.  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'  
 EXPORTING  
 classname = c\_class\_mangel1  
 classtype = c\_cltyp\_mangel1  
*\* OBJECT\_KEY = gv\_objektschl*  
 object\_key = lv\_object\_key  
 TABLES  
 uris = lt\_uris  
 signature = lt\_signature  
 components = lt\_components  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 ENDIF.  
 ELSE.  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'  
 EXPORTING  
 classname = c\_class\_mangel1  
 classtype = c\_cltyp\_mangel1  
*\* OBJECT\_KEY = gv\_objektschl*  
 object\_key = lv\_object\_key  
 TABLES  
 uris = lt\_uris  
 signature = lt\_signature  
 components = lt\_components  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 ENDIF.  
  
 WHEN OTHERS.  
*\* CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'*  
*\* EXPORTING*  
*\* classname = c\_class*  
*\* classtype = c\_cltyp*  
*\* object\_key = lv\_object\_key*  
*\* TABLES*  
*\* uris = lt\_uris*  
*\* signature = lt\_signature*  
*\* components = lt\_components*  
*\* EXCEPTIONS*  
*\* nothing\_found = 1*  
*\* parameter\_error = 2*  
*\* not\_allowed = 3*  
*\* error\_kpro = 4*  
*\* internal\_error = 5*  
*\* not\_authorized = 6*  
*\* OTHERS = 7.*  
  
 ENDCASE.  
  
 CASE sy-subrc.  
 WHEN 0.  
*\* beim Start soll zunächst die aktuellste Version des "höchsten" Rüge-Schreibens angezeigt werden*  
*\* Attribut BDS\_KEYWORD enthält die Nr. der Rüge mit vorangestelltem "document"*  
*\* Attribut BDS\_STATE enthält die "Versions-Nr."*  
  
 lt\_sign\_copy = lt\_signature.  
*\* Aktuellste Rüge-Nr. ziehen <=> höchster Eintrag in BDS\_KEYWORD*  
 LOOP AT lt\_signature INTO ls\_signature  
 WHERE prop\_name EQ 'BDS\_KEYWORD'.  
 IF ls\_signature-prop\_value CP 'document\*'.  
 ls\_signature-prop\_value = ls\_signature-prop\_value.  
 APPEND ls\_signature TO lt\_sign\_sort\_r.  
 ENDIF.  
 ENDLOOP.  
 SORT lt\_sign\_sort\_r DESCENDING BY prop\_value.  
  
 READ TABLE lt\_sign\_sort\_r INDEX 1. *" => höchste Rüge-Nr.*  
  
*\* höchste Versions-Nr. zur Rüge mit höchster Nr. holen:*  
*\* erst alle Einträge zur Rügen-Nr. in die lt\_sign\_sort\_v einlesen*  
 CLEAR ls\_signature.  
 CLEAR ls\_sign\_copy.  
  
 LOOP AT lt\_signature INTO ls\_signature  
 WHERE prop\_name = 'BDS\_KEYWORD'  
 AND prop\_value = lt\_sign\_sort\_r-prop\_value.  
*\* => jeweilige doc\_count zum Dkument mit höchster Rüge-Nr. bekannt*  
 READ TABLE lt\_sign\_copy INTO ls\_sign\_copy  
 WITH KEY doc\_count = ls\_signature-doc\_count  
 prop\_name = 'BDS\_STATE'.  
 IF sy-subrc = 0.  
 APPEND ls\_sign\_copy TO lt\_sign\_sort\_v.  
 ENDIF.  
 ENDLOOP.  
  
 SORT lt\_sign\_sort\_v DESCENDING BY prop\_value.  
 READ TABLE lt\_sign\_sort\_v INDEX 1. *" => höchste Versions-Nr.*  
  
 CLEAR ls\_signature.  
 LOOP AT lt\_signature INTO ls\_signature  
 WHERE doc\_id EQ lt\_sign\_sort\_v-doc\_id.  
 APPEND ls\_signature TO gt\_signatures.  
  
 READ TABLE lt\_uris INTO ls\_uri WITH KEY doc\_count = ls\_signature-doc\_count.  
 gv\_url = ls\_uri-uri.  
 gs\_signature = ls\_signature.  
  
 ENDLOOP.  
  
 READ TABLE lt\_components INTO ls\_component WITH KEY doc\_count = ls\_signature-doc\_count.  
 ls\_component-doc\_count = ls\_signature-doc\_count.  
 APPEND ls\_component TO gt\_components\_upd.  
  
 PERFORM load\_document USING gv\_url gs\_vorl-doc.  
 gv\_init = 'X'.  
 gv\_exist = 'X'.  
 gv\_objektschl = lv\_object\_key.  
  
 WHEN OTHERS.  
*\* noch kein Dokument zur Meldung vorhanden*  
 ENDCASE.  
 ENDIF.  
  
 ELSE.  
  
 IF sy-ucomm = 'VORLAGEN'.  
 READ TABLE gt\_vorl INTO gs\_vorl INDEX ls\_selected\_line-index.  
 IF gs\_vorl-comp\_id+0(12) = viqmel-qmnum. *" gespeichertes Dokument ist geöffnet!*  
 LOOP AT SCREEN.  
 IF screen-name EQ 'PB\_VERSION'. *" Button " Neue Version anlegen" deaktivieren!*  
 screen-intensified = 1.  
 screen-input = 0.  
 ENDIF.  
 MODIFY SCREEN.  
 ENDLOOP.  
 ENDIF.  
 ENDIF.  
  
 ENDIF. *" gv\_init EQ space. "nur beim ersten Aufruf*  
  
 ENDIF.  
  
 IF sy-ucomm = 'BAUVERTRAG'.  
 PERFORM supress\_f31(sapliqs0).  
 ENDIF.  
  
ENDMODULE.

Include🡪 ZXQQMF15

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF15.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form GET\_MASSNAHME\_DETAILS*  
*\*&---------------------------------------------------------------------\**  
FORM get\_massnahme\_details .  
  
 DATA lv\_index TYPE sy-index.  
 DATA ls\_viqmsm TYPE wqmsm.  
 DATA lv\_txcdsm TYPE qtxtcdgrsm. *" Kurztext Maßnahmecode*  
  
 LOOP AT inxsm.  
 IF inxsm-mark = 'X'.  
 fl\_txtma = fl\_txtma + 1.  
 lv\_index = inxsm-tabpt.  
 ENDIF.  
 ENDLOOP.  
  
 READ TABLE iviqmsm INTO ls\_viqmsm INDEX lv\_index.  
  
 SELECT SINGLE kurztext FROM qpct INTO lv\_txcdsm  
 WHERE qpct~code = ls\_viqmsm-mncod  
 AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 gv\_mncod = ls\_viqmsm-mncod.  
 gv\_txtma = ls\_viqmsm-matxt.  
 gv\_txcdsm = lv\_txcdsm.  
 ELSE.  
 CLEAR: gv\_mncod, gv\_txtma, gv\_txcdsm.  
 ENDIF.  
  
ENDFORM.

Include🡪 ZXQQMF16

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF16.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form GET\_TEMPLATE\_URL*  
*\*&---------------------------------------------------------------------\**  
*\* <--P\_GV\_URL text*  
*\*----------------------------------------------------------------------\**  
FORM get\_template\_url CHANGING cv\_url.  
  
 DATA lt\_uris TYPE STANDARD TABLE OF bapiuri.  
 DATA ls\_uri TYPE bapiuri.  
 DATA lv\_object\_key TYPE bapibds01-objkey.  
 DATA lt\_data TYPE STANDARD TABLE OF bapiconten.  
 DATA lv\_doc\_size TYPE i.  
 DATA lt\_signature TYPE STANDARD TABLE OF bapisignat.  
 DATA ls\_signature TYPE bapisignat.  
 DATA lt\_sign\_copy TYPE STANDARD TABLE OF bapisignat.  
 DATA ls\_sign\_copy TYPE bapisignat.  
 DATA lt\_sign\_sort\_r TYPE STANDARD TABLE OF bapisignat WITH HEADER LINE.  
 DATA lt\_sign\_sort\_v TYPE STANDARD TABLE OF bapisignat WITH HEADER LINE.  
  
 DATA lv\_classname TYPE bds\_clsnam.  
 DATA lv\_classtype TYPE bds\_clstyp.  
*\* DATA: begin of lt\_sign\_sort\_v occurs 0. "alle Signaturtabs müssen bei Versionen > 9 noch mit weiterem*  
*\* include structure bapisignat. "Zähler versehen werden (prop\_value2), damit danach RICHTI sortiert werden kann.*  
*\* data prop\_value2(3) type n.*  
*\* DATA: end of lt\_sign\_sort\_v.*  
 DATA lt\_components TYPE STANDARD TABLE OF bapicompon.  
 DATA ls\_component TYPE bapicompon.  
 DATA lv\_lines TYPE n.  
  
 DATA gt\_vorlagen TYPE TABLE OF /afm/oaer.  
 DATA ls\_vorlagen TYPE /afm/oaer.  
  
*\*----------------------------------------------------------------------\**  
  
 CLEAR gt\_components\_upd.  
 REFRESH gt\_components\_upd.  
 REFRESH gt\_signatures.  
  
 CLEAR fl\_massn.  
  
 CASE sy-ucomm.  
*\* ----------------------------------------------*  
 WHEN '10'. *"Vorlagendaten holen*  
*\* ----------------------------------------------*  
 IF gv\_qmart = 'SG' OR gv\_qmart = 'SM'.  
 LOOP AT inxsm.  
 IF inxsm-mark = 'X'.  
 fl\_massn = fl\_massn + 1.  
 ENDIF.  
 ENDLOOP.  
  
 DATA lv\_vorlage TYPE /afm/oaer-vorlage.  
 CLEAR lv\_vorlage.  
 DATA gt\_vorlagen\_save TYPE TABLE OF /afm/oaer.  
 CLEAR gt\_vorlagen\_save.  
 FREE: gt\_vorlagen, gt\_vorl.  
  
*\* Laden der abhänigen Dokumente (mit und ohne Vorlagen)*  
 IF fl\_massn > 0.  
*\* Vorlage zu Maßnahmen und Meldungsart laden*  
 SELECT \* FROM /afm/oaer INTO TABLE gt\_vorlagen WHERE qmart = gv\_qmart AND mncod = gv\_mncod.  
 gt\_vorlagen\_save = gt\_vorlagen.  
*\* + Dokumente zur Meldungsart laden ohne Maßnahmenbezug - mit Vorlagen!*  
 LOOP AT gt\_vorlagen INTO ls\_vorlagen.  
 CONCATENATE viqmel-qmnum ls\_vorlagen-mncod INTO lv\_vorlage.  
 ls\_vorlagen-vorlage = lv\_vorlage.  
 APPEND ls\_vorlagen TO gt\_vorlagen\_save.  
 CLEAR lv\_vorlage.  
 ENDLOOP.  
 gt\_vorlagen = gt\_vorlagen\_save.  
 ELSE.  
*\* Dokumente zur Meldungsart laden ohne Maßnahmenbezug - ohne Vorlagen!*  
 SELECT \* FROM /afm/oaer INTO TABLE gt\_vorlagen WHERE qmart = gv\_qmart.  
 LOOP AT gt\_vorlagen INTO ls\_vorlagen.  
 CONCATENATE viqmel-qmnum ls\_vorlagen-mncod INTO lv\_vorlage.  
 ls\_vorlagen-vorlage = lv\_vorlage.  
 MODIFY gt\_vorlagen FROM ls\_vorlagen INDEX sy-tabix.  
 CLEAR lv\_vorlage.  
 ENDLOOP.  
 ENDIF.  
  
*\* Über bestehende Vorlagen iterieren*  
 LOOP AT gt\_vorlagen INTO ls\_vorlagen.  
*\* Objektschlüssel ist bei Vorlage immer aus Customizing zu ziehen*  
 lv\_object\_key = ls\_vorlagen-vorlage. *" <-- Z.B. ZKORR\_MAENGEL\_1*  
  
*\* prüfen ob es eine Vorlage oder ein bestehendes Document ist*  
 If lv\_object\_key cs viqmel-qmnum.  
 gs\_vorl-doc = 'X'.  
 else.  
 gs\_vorl-doc = ''.  
 endif.  
*\* Mehr als eine Maßnahme selektiert?!?*  
 IF fl\_massn > 1.  
 CALL FUNCTION 'POPUP\_TO\_INFORM'  
 EXPORTING  
 titel = 'Maßname Korrespondenz'  
 txt1 = 'Bitte nur eine Maßnahme markieren !'(103)  
 txt2 = ''  
*\* TXT3 = ' '*  
*\* TXT4 = ' '*  
 .  
 EXIT.  
 ENDIF.  
  
*\* Tabellen leeren*  
 FREE: lt\_uris, lt\_signature, lt\_components.  
  
*\* Dokument abhänig zu selektierten Vorlagen laden*  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'  
 EXPORTING  
*\* LOGICAL\_SYSTEM =*  
 classname = ls\_vorlagen-cls\_nam  
 classtype = ls\_vorlagen-clstyp  
*\* CLIENT = SY-MANDT*  
 object\_key = lv\_object\_key  
*\* URL\_LIFETIME =*  
*\* STANDARD\_URL\_ONLY =*  
*\* DATA\_PROVIDER\_URL\_ONLY =*  
*\* WEB\_APPLIC\_SERVER\_URL\_ONLY =*  
*\* URL\_USED\_AT =*  
 TABLES  
 uris = lt\_uris  
 signature = lt\_signature  
 components = lt\_components  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
  
 CASE sy-subrc.  
 WHEN 0.  
*\* Nur die Vorlagen zur Systemsprache einlesen.*  
 LOOP AT lt\_uris INTO ls\_uri.  
 READ TABLE lt\_signature INTO ls\_signature  
 WITH KEY doc\_count = ls\_uri-doc\_count  
 prop\_name = 'LANGUAGE'  
 prop\_value = sy-langu.  
 IF sy-subrc NE 0.  
 DELETE lt\_uris.  
 READ TABLE lt\_components INTO ls\_component  
 WITH KEY doc\_count = ls\_uri-doc\_count.  
 IF sy-subrc = 0.  
 DELETE lt\_components INDEX sy-tabix.  
 ENDIF.  
 LOOP AT lt\_signature INTO ls\_signature WHERE doc\_count = ls\_uri-doc\_count.  
 DELETE lt\_signature INDEX sy-tabix.  
 ENDLOOP.  
 ENDIF.  
 ENDLOOP.  
 DESCRIBE TABLE lt\_uris LINES lv\_lines.  
 IF lv\_lines GT 0.  
 gv\_get\_vorl = 'X'.  
 ENDIF.  
 WHEN OTHERS.  
*\* MESSAGE w000(38) WITH 'Keinen Text gefunden'(m09).*  
 ENDCASE.  
  
*\* Sichern der eingelesenen Daten*  
 MOVE: lt\_uris[] TO gt\_uris[],  
 lt\_uris[] TO gt\_uris\_vorl[],  
 lt\_signature[] TO gt\_sign\_vorl[].  
  
*\* Daten fürs ALV bereitstellen*  
 DATA lv\_datum TYPE dats.  
 CLEAR lv\_datum.  
 CLEAR gs\_vorl.  
 CLEAR ls\_uri.  
 CLEAR ls\_signature.  
 CLEAR ls\_component.  
  
*\* Daten aus Vorlage übernehmen und sichern*  
 LOOP AT lt\_uris INTO ls\_uri.  
  
 MOVE-CORRESPONDING ls\_uri TO gs\_vorl.  
  
 READ TABLE lt\_components INTO ls\_component INDEX sy-tabix.  
 MOVE-CORRESPONDING ls\_component TO gs\_vorl.  
  
 READ TABLE lt\_signature INTO ls\_signature WITH KEY doc\_count = ls\_uri-doc\_count  
 prop\_name = 'DESCRIPTION'.  
 MOVE-CORRESPONDING ls\_signature TO gs\_vorl.  
  
 READ TABLE lt\_signature INTO ls\_signature WITH KEY doc\_count = ls\_uri-doc\_count  
 prop\_name = 'CREATED\_AT'.  
 lv\_datum = ls\_signature-prop\_value+0(8).  
 CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
 EXPORTING  
 date\_internal = lv\_datum  
 IMPORTING  
 date\_external = gs\_vorl-datum  
 EXCEPTIONS  
 date\_internal\_is\_invalid = 1  
 OTHERS = 2.  
 IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
 ENDIF.  
  
*\* Übernahme der Dokumenteklasse (ZKORR\_MAENGEL etc...)*  
 gs\_vorl-type = ls\_vorlagen-CLS\_NAM.  
  
 APPEND gs\_vorl TO gt\_vorl.  
 ENDLOOP.  
 ENDLOOP.  
 ENDIF. *" gv\_qmart = 'SG' OR gv\_qmart = 'SM'*  
  
*\* --------------------------------------*  
 WHEN 'MELDTEXT'.  
*\* --------------------------------------*  
 DATA ls\_vorl LIKE LINE OF gt\_vorl.  
*\* Wenn Text geladen wird*  
 lv\_object\_key = viqmel-qmnum.  
  
 CASE gv\_qmart.  
  
 WHEN 'SM'.  
  
*\* Wie kann erkannt werden welche Vorlage ausgewählt wurde?!?*  
 FREE lt\_sel\_rows.  
 CALL METHOD go\_alv->get\_selected\_rows  
 IMPORTING  
 et\_index\_rows = lt\_sel\_rows.  
 LOOP AT lt\_sel\_rows INTO ls\_sel\_rows.  
 READ TABLE gt\_vorl INTO ls\_vorl INDEX ls\_sel\_rows-index.  
 ENDLOOP.  
  
 lv\_object\_key = c\_class\_mangel1.  
  
*\* IF gv\_qmart = 'SM' AND fl\_txtma = 1 .*  
*\* CLEAR lv\_object\_key.*  
*\* CONCATENATE viqmel-qmnum gv\_mncod INTO lv\_object\_key.*  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'  
 EXPORTING  
*\* LOGICAL\_SYSTEM =*  
 classname = c\_class\_mangel1  
 classtype = c\_cltyp\_mangel1  
*\* CLIENT = SY-MANDT*  
 object\_key = lv\_object\_key  
*\* URL\_LIFETIME =*  
*\* STANDARD\_URL\_ONLY =*  
*\* DATA\_PROVIDER\_URL\_ONLY =*  
*\* WEB\_APPLIC\_SERVER\_URL\_ONLY =*  
*\* URL\_USED\_AT =*  
 TABLES  
 uris = lt\_uris  
 signature = lt\_signature  
 components = lt\_components  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
*\* ELSE.*  
*\* CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'*  
*\* EXPORTING*  
*\*\* LOGICAL\_SYSTEM =*  
*\* classname = c\_class500*  
*\* classtype = c\_cltyp500*  
*\*\* CLIENT = SY-MANDT*  
*\* object\_key = lv\_object\_key*  
*\*\* URL\_LIFETIME =*  
*\*\* STANDARD\_URL\_ONLY =*  
*\*\* DATA\_PROVIDER\_URL\_ONLY =*  
*\*\* WEB\_APPLIC\_SERVER\_URL\_ONLY =*  
*\*\* URL\_USED\_AT =*  
*\* TABLES*  
*\* uris = lt\_uris*  
*\* signature = lt\_signature*  
*\* components = lt\_components*  
*\* EXCEPTIONS*  
*\* nothing\_found = 1*  
*\* parameter\_error = 2*  
*\* not\_allowed = 3*  
*\* error\_kpro = 4*  
*\* internal\_error = 5*  
*\* not\_authorized = 6*  
*\* OTHERS = 7.*  
*\* ENDIF.*  
*\**  
*\* WHEN 'SG'.*  
*\**  
*\* IF gv\_qmart = 'SG' AND fl\_txtma = 1 .*  
*\* CLEAR lv\_object\_key.*  
*\* CONCATENATE viqmel-qmnum gv\_mncod INTO lv\_object\_key.*  
*\* CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'*  
*\* EXPORTING*  
*\*\* LOGICAL\_SYSTEM =*  
*\* classname = c\_classmass*  
*\* classtype = c\_cltypmass*  
*\*\* CLIENT = SY-MANDT*  
*\* object\_key = lv\_object\_key*  
*\*\* URL\_LIFETIME =*  
*\*\* STANDARD\_URL\_ONLY =*  
*\*\* DATA\_PROVIDER\_URL\_ONLY =*  
*\*\* WEB\_APPLIC\_SERVER\_URL\_ONLY =*  
*\*\* URL\_USED\_AT =*  
*\* TABLES*  
*\* uris = lt\_uris*  
*\* signature = lt\_signature*  
*\* components = lt\_components*  
*\* EXCEPTIONS*  
*\* nothing\_found = 1*  
*\* parameter\_error = 2*  
*\* not\_allowed = 3*  
*\* error\_kpro = 4*  
*\* internal\_error = 5*  
*\* not\_authorized = 6*  
*\* OTHERS = 7.*  
*\* ELSE.*  
*\* CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'*  
*\* EXPORTING*  
*\*\* LOGICAL\_SYSTEM =*  
*\* classname = c\_class500*  
*\* classtype = c\_cltyp500*  
*\*\* CLIENT = SY-MANDT*  
*\* object\_key = lv\_object\_key*  
*\*\* URL\_LIFETIME =*  
*\*\* STANDARD\_URL\_ONLY =*  
*\*\* DATA\_PROVIDER\_URL\_ONLY =*  
*\*\* WEB\_APPLIC\_SERVER\_URL\_ONLY =*  
*\*\* URL\_USED\_AT =*  
*\* TABLES*  
*\* uris = lt\_uris*  
*\* signature = lt\_signature*  
*\* components = lt\_components*  
*\* EXCEPTIONS*  
*\* nothing\_found = 1*  
*\* parameter\_error = 2*  
*\* not\_allowed = 3*  
*\* error\_kpro = 4*  
*\* internal\_error = 5*  
*\* not\_authorized = 6*  
*\* OTHERS = 7.*  
*\* ENDIF.*  
*\**  
*\* WHEN OTHERS.*  
*\* CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'*  
*\* EXPORTING*  
*\*\* LOGICAL\_SYSTEM =*  
*\* classname = c\_class*  
*\* classtype = c\_cltyp*  
*\*\* CLIENT = SY-MANDT*  
*\* object\_key = lv\_object\_key*  
*\*\* URL\_LIFETIME =*  
*\*\* STANDARD\_URL\_ONLY =*  
*\*\* DATA\_PROVIDER\_URL\_ONLY =*  
*\*\* WEB\_APPLIC\_SERVER\_URL\_ONLY =*  
*\*\* URL\_USED\_AT =*  
*\* TABLES*  
*\* uris = lt\_uris*  
*\* signature = lt\_signature*  
*\* components = lt\_components*  
*\* EXCEPTIONS*  
*\* nothing\_found = 1*  
*\* parameter\_error = 2*  
*\* not\_allowed = 3*  
*\* error\_kpro = 4*  
*\* internal\_error = 5*  
*\* not\_authorized = 6*  
*\* OTHERS = 7.*  
  
 ENDCASE.  
  
 CASE sy-subrc.  
 WHEN 0.  
 lt\_sign\_copy = lt\_signature.  
  
*\* --------------------------------------------------------------------------------------------------------------------------------*  
 IF gv\_ruege IS INITIAL. *"wurde keine Nr.zum Rüge-Schreiben angegeben, den aktuellsten Text (Version + Rüge) ziehen*  
*\* --------------------------------------------------------------------------------------------------------------------------------*  
  
*\* Aktuellste Rüge-Nr. ziehen <=> höchster Eintrag in BDS\_KEYWORD*  
 LOOP AT lt\_signature INTO ls\_signature  
 WHERE prop\_name EQ 'BDS\_KEYWORD'.  
  
 IF ls\_signature-prop\_value CP 'document\*'.  
 ls\_signature-prop\_value = ls\_signature-prop\_value.  
 APPEND ls\_signature TO lt\_sign\_sort\_r.  
 ENDIF.  
 ENDLOOP.  
 SORT lt\_sign\_sort\_r DESCENDING BY prop\_value.  
  
 READ TABLE lt\_sign\_sort\_r INDEX 1. *" => höchste Rüge-Nr.*  
*\* höchste Versions-Nr. zur Rüge mit höchster Nr. holen:*  
*\* erst alle Einträge zur Rügen-Nr. in die lt\_sign\_sort\_v einlesen*  
 CLEAR ls\_signature.  
 CLEAR ls\_sign\_copy.  
  
 LOOP AT lt\_signature INTO ls\_signature  
 WHERE prop\_name = 'BDS\_KEYWORD'  
 AND prop\_value = lt\_sign\_sort\_r-prop\_value.  
*\* => jeweilige doc\_count zum Dkument mit höchster Rüge-Nr. bekannt*  
 READ TABLE lt\_sign\_copy INTO ls\_sign\_copy  
 WITH KEY doc\_count = ls\_signature-doc\_count  
 prop\_name = 'BDS\_STATE'.  
 IF sy-subrc = 0.  
*\* MOVE ls\_sign\_copy-prop\_value TO*  
 APPEND ls\_sign\_copy TO lt\_sign\_sort\_v.  
 ENDIF.  
 ENDLOOP.  
  
 SORT lt\_sign\_sort\_v DESCENDING BY prop\_value.  
 READ TABLE lt\_sign\_sort\_v INDEX 1. *" => höchste Versions-Nr.*  
  
 CLEAR ls\_signature.  
 LOOP AT lt\_signature INTO ls\_signature  
 WHERE doc\_id EQ lt\_sign\_sort\_v-doc\_id.  
 APPEND ls\_signature TO gt\_signatures.  
 READ TABLE lt\_uris INTO ls\_uri WITH KEY doc\_count = ls\_signature-doc\_count.  
 gv\_url = ls\_uri-uri.  
 gs\_signature = ls\_signature.  
 ENDLOOP.  
  
 READ TABLE lt\_components INTO ls\_component WITH KEY doc\_count = ls\_signature-doc\_count.  
 APPEND ls\_component TO gt\_components\_upd.  
  
*\* --------------------------------------------------------------------------------------------------------------------------------*  
 ELSE. *" Rüge-Nr. wurde aus der Listbox ausgewählt*  
*\* --------------------------------------------------------------------------------------------------------------------------------*  
  
 CLEAR ls\_signature.  
 CLEAR ls\_sign\_copy.  
  
 DATA lv\_doc(12) TYPE c.  
 CONCATENATE 'document' gv\_ruege INTO lv\_doc.  
  
 LOOP AT lt\_signature INTO ls\_signature  
 WHERE prop\_name = 'BDS\_KEYWORD'  
 AND prop\_value = lv\_doc.  
*\* => jeweilige doc\_count zum Dkument mit höchster Rüge-Nr. bekannt*  
 READ TABLE lt\_sign\_copy INTO ls\_sign\_copy  
 WITH KEY doc\_count = ls\_signature-doc\_count  
 prop\_name = 'BDS\_STATE'.  
 IF sy-subrc = 0.  
 APPEND ls\_sign\_copy TO lt\_sign\_sort\_v.  
 ENDIF.  
 ENDLOOP.  
  
 SORT lt\_sign\_sort\_v DESCENDING BY prop\_value.  
*\* SORT lt\_sign\_sort\_v DESCENDING BY prop\_value2.*  
 READ TABLE lt\_sign\_sort\_v INDEX 1. *" => höchste Versions-Nr.*  
  
 CLEAR ls\_signature.  
 LOOP AT lt\_signature INTO ls\_signature  
 WHERE doc\_id EQ lt\_sign\_sort\_v-doc\_id.  
 APPEND ls\_signature TO gt\_signatures.  
 READ TABLE lt\_uris INTO ls\_uri WITH KEY doc\_count = ls\_signature-doc\_count.  
 gv\_url = ls\_uri-uri.  
 gs\_signature = ls\_signature.  
 ENDLOOP.  
  
 READ TABLE lt\_components INTO ls\_component WITH KEY doc\_count = ls\_signature-doc\_count.  
 APPEND ls\_component TO gt\_components\_upd.  
  
*\* --------------------------------------------------------------------------------------------------------------------------------*  
 ENDIF.  
*\* --------------------------------------------------------------------------------------------------------------------------------*  
  
 WHEN OTHERS.  
*\* MESSAGE w000(38) WITH 'Keinen Text gefunden'.*  
 ENDCASE.  
  
*\* Sichern der Daten des ausgewählten Dokumentes*  
 MOVE: ls\_uri TO gs\_uri,  
 ls\_signature TO gs\_signature,  
 ls\_component TO gs\_component,  
 lv\_object\_key TO gv\_objektschl,  
 lt\_uris[] TO gt\_uris[].  
  
 ENDCASE.  
  
ENDFORM.

Include🡪 ZXQQMO09

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMO09.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module ALV OUTPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE alv OUTPUT.  
  
 IF go\_cont\_alv IS INITIAL.  
  
 CREATE OBJECT go\_cont\_alv  
 EXPORTING  
*\* parent =*  
 container\_name = 'C\_ALV'  
*\* style =*  
*\* lifetime = lifetime\_default*  
*\* repid =*  
*\* dynnr =*  
*\* no\_autodef\_progid\_dynnr =*  
 EXCEPTIONS  
 cntl\_error = 1  
 cntl\_system\_error = 2  
 create\_error = 3  
 lifetime\_error = 4  
 lifetime\_dynpro\_dynpro\_link = 5  
 OTHERS = 6.  
 IF sy-subrc <> 0.  
 MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
 WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
 ENDIF.  
  
 CHECK NOT go\_cont\_alv IS INITIAL.  
  
 IF go\_alv IS INITIAL.  
 CREATE OBJECT go\_alv  
 EXPORTING  
*\* i\_shellstyle = 0*  
*\* i\_lifetime =*  
 i\_parent = go\_cont\_alv  
*\* i\_appl\_events = space*  
*\* i\_parentdbg =*  
*\* i\_applogparent =*  
*\* i\_graphicsparent =*  
*\* i\_name =*  
*\* i\_fcat\_complete = space*  
 EXCEPTIONS  
 error\_cntl\_create = 1  
 error\_cntl\_init = 2  
 error\_cntl\_link = 3  
 error\_dp\_create = 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.  
 ENDIF.  
 ENDIF. *"go\_alv IS INITIAL*  
  
 ENDIF. *"go\_cont\_alv IS INITIAL*  
  
ENDMODULE.

Include🡪 ZXQQMO10

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMO10.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module FIELDCAT OUTPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE fieldcat OUTPUT.  
  
 CLEAR: gt\_fcat, gt\_fcat[],  
 gs\_fcat.  
  
 CALL FUNCTION 'LVC\_FIELDCATALOG\_MERGE'  
 EXPORTING  
*\* I\_BUFFER\_ACTIVE =*  
 i\_structure\_name = 'ZST\_AFM\_MELD\_TEXT'  
*\* I\_CLIENT\_NEVER\_DISPLAY = 'X'*  
*\* I\_BYPASSING\_BUFFER =*  
*\* I\_INTERNAL\_TABNAME =*  
 CHANGING  
 ct\_fieldcat = gt\_fcat  
 EXCEPTIONS  
 inconsistent\_interface = 1  
 program\_error = 2  
 OTHERS = 3.  
 IF sy-subrc <> 0.  
 MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
 WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
 ENDIF.  
  
 LOOP AT gt\_fcat INTO gs\_fcat.  
 IF gs\_fcat-fieldname EQ 'DOC\_ID' OR gs\_fcat-fieldname EQ 'PROP\_NAME'  
 OR gs\_fcat-fieldname EQ 'URI'.  
 gs\_fcat-no\_out = 'X'.  
 ELSEIF gs\_fcat-fieldname EQ 'PROP\_VALUE'.  
 MOVE 1 TO gs\_fcat-col\_pos.  
 gs\_fcat-outputlen = '50'.  
 gs\_fcat-col\_opt = 'X'.  
 ELSEIF gs\_fcat-fieldname EQ 'COMP\_ID'.  
 MOVE 3 TO gs\_fcat-col\_pos.  
 gs\_fcat-outputlen = '60'.  
 gs\_fcat-col\_opt = 'X'.  
 ENDIF.  
 MODIFY gt\_fcat FROM gs\_fcat INDEX sy-tabix.  
 CLEAR gs\_fcat.  
 ENDLOOP.  
  
*\* ab hier Zusatzfelder*  
 MOVE 2 TO gs\_fcat-col\_pos.  
 gs\_fcat-col\_opt = 'X'.  
 MOVE 'DATUM' TO gs\_fcat-fieldname.  
 MOVE 'D' TO gs\_fcat-inttype.  
 MOVE '10' TO gs\_fcat-outputlen.  
 MOVE 'Erstellungsdatum' TO gs\_fcat-seltext.  
 MOVE 'Erstellungsdatum' TO gs\_fcat-scrtext\_l.  
 MOVE 'Erst.Datum' TO gs\_fcat-scrtext\_m.  
 MOVE 'Datum' TO gs\_fcat-scrtext\_s.  
 APPEND gs\_fcat TO gt\_fcat.  
 CLEAR gs\_fcat.  
  
ENDMODULE.

Include🡪 ZXQQMO11

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMO11.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module DISPLAY OUTPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE display OUTPUT.  
  
 gs\_layout-sel\_mode = 'A'.  
  
 IF NOT go\_alv IS INITIAL.  
  
 CALL METHOD go\_alv->set\_table\_for\_first\_display  
 EXPORTING  
*\* i\_buffer\_active =*  
*\* i\_bypassing\_buffer =*  
*\* i\_consistency\_check =*  
 i\_structure\_name = 'ZST\_AFM\_MELD\_TEXT'  
*\* is\_variant =*  
 i\_save = 'A'  
*\* i\_default = 'X'*  
 is\_layout = gs\_layout  
*\* is\_print =*  
*\* it\_special\_groups =*  
*\* it\_toolbar\_excluding =*  
*\* it\_hyperlink =*  
*\* it\_alv\_graphics =*  
*\* it\_except\_qinfo =*  
*\* ir\_salv\_adapter =*  
 CHANGING  
 it\_outtab = gt\_vorl  
 it\_fieldcatalog = gt\_fcat  
*\* it\_sort =*  
*\* it\_filter =*  
 EXCEPTIONS  
 invalid\_parameter\_combination = 1  
 program\_error = 2  
 too\_many\_lines = 3  
 OTHERS = 4.  
 IF sy-subrc <> 0.  
 MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
 WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
 ENDIF.  
  
*\*&---------------------------------------------------------------------\**  
  
*\*\*\* markierte Zeile weiterhin anzeigen*  
  
 DATA: lt\_index\_rows TYPE lvc\_t\_row,  
 ls\_index\_rows TYPE lvc\_s\_row.  
  
 CLEAR: lt\_index\_rows, ls\_index\_rows.  
 IF ls\_sel\_rows-index IS NOT INITIAL.  
 ls\_index\_rows-index = ls\_sel\_rows-index.  
 APPEND ls\_index\_rows TO lt\_index\_rows.  
  
 CALL METHOD go\_alv->set\_selected\_rows  
 EXPORTING  
 it\_index\_rows = lt\_index\_rows  
*\* it\_row\_no =*  
*\* is\_keep\_other\_selections =*  
 .  
  
 CALL METHOD go\_alv->refresh\_table\_display  
*\* EXPORTING*  
*\* is\_stable =*  
*\* i\_soft\_refresh =*  
 EXCEPTIONS  
 finished = 1  
 OTHERS = 2.  
 IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
 ENDIF.  
  
 ENDIF.  
  
  
 ENDIF. *" NOT go\_alv IS INITIAL*  
  
ENDMODULE.

Include🡪 ZXQQMI12

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI12.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module SELEKT\_VORLAGE INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE selekt\_vorlage INPUT.  
  
 CALL METHOD go\_alv->get\_selected\_rows  
 IMPORTING  
 et\_index\_rows = lt\_rows.  
  
 IF NOT lt\_rows[] IS INITIAL.  
*\* GRID lediglich mit einzelner Zeilenauswahl konzipiert*  
 READ TABLE lt\_rows INDEX 1 INTO ls\_selected\_line.  
 CALL METHOD cl\_gui\_cfw=>flush.  
 IF sy-subrc = 0.  
 READ TABLE gt\_uris\_vorl INDEX ls\_selected\_line-index INTO gs\_uri.  
 IF sy-subrc = 0.  
 MOVE gs\_uri-uri TO gv\_url.  
 MOVE gs\_uri-doc\_count TO gv\_sel\_vorl.  
 gv\_selected = 'X'.  
 ENDIF.  
 ENDIF.  
*\* CLEAR ls\_selected\_line-index.*  
 ENDIF.  
  
ENDMODULE.

Include🡪 ZXQQMI15

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI15.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module USER\_COMMAND\_9000 INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE user\_command\_9000 INPUT.  
  
 IF gv\_qmart = 'SG' OR gv\_qmart = 'SM'. *"Mängelmeldung*  
  
 DATA: lv\_ucomm TYPE sy-ucomm.  
 lv\_ucomm = sy-ucomm. *"O.Meyer(Begis) 02.04.15*  
  
 CASE sy-ucomm.  
  
 WHEN 'VORLAGEN'.  
  
 IF gv\_doc\_open NE 'X'.  
 FREE lt\_sel\_rows.  
 CALL METHOD go\_alv->get\_selected\_rows  
 IMPORTING  
 et\_index\_rows = lt\_sel\_rows.  
*\* ET\_ROW\_NO =*  
 .  
 LOOP AT lt\_sel\_rows INTO ls\_sel\_rows.  
 READ TABLE gt\_vorl INTO ls\_vorl INDEX ls\_sel\_rows-index.  
 ENDLOOP.  
 gv\_url = ls\_vorl-uri.  
 IF gv\_url IS INITIAL.  
 MESSAGE i000(38) WITH TEXT-007. *" Bitte erst eine Vorlage auswählen!*  
 ELSE.  
 PERFORM load\_document USING gv\_url ls\_vorl-doc.  
 ENDIF.  
 ELSE.  
 MESSAGE e000(38) WITH TEXT-008. *" Bitte erst die geöffnete Datei schliessen*  
 ENDIF.  
  
 CASE gv\_qmart.  
 WHEN 'SM'.  
 IF fl\_txtma = 1.  
*\* BDN Adresse Details für die Vorlage ermitteln*  
 SELECT SINGLE cls\_nam clstyp typeid FROM /afm/oaer INTO CORRESPONDING FIELDS OF gs\_oaer  
 WHERE mncod = gv\_mncod  
 AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 gv\_objektschl = gs\_oaer-typeid.  
 ENDIF.  
 ELSE.  
*\* BDN Adresse Details für die Vorlage ermitteln*  
 SELECT SINGLE cls\_nam clstyp typeid FROM /afm/oaer INTO CORRESPONDING FIELDS OF gs\_oaer  
 WHERE mncod = 'ALLG'  
 AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 gv\_objektschl = gs\_oaer-typeid.  
 ENDIF.  
 gv\_objektschl = 'MAENGELSBESEITIGUNG'.  
 ENDIF.  
 WHEN 'SG'.  
 IF fl\_txtma = 1.  
*\* BDN Adresse Details für die Vorlage ermitteln*  
 SELECT SINGLE cls\_nam clstyp typeid FROM /afm/oaer INTO CORRESPONDING FIELDS OF gs\_oaer  
 WHERE mncod = gv\_mncod  
 AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 gv\_objektschl = gs\_oaer-typeid.  
 ENDIF.  
 gv\_objektschl = 'MASSNAHME'.  
 ELSE.  
*\* BDN Adresse Details für die Vorlage ermitteln*  
 SELECT SINGLE cls\_nam clstyp typeid FROM /afm/oaer INTO CORRESPONDING FIELDS OF gs\_oaer  
 WHERE mncod = 'ALLG'  
 AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 gv\_objektschl = gs\_oaer-typeid.  
 ENDIF.  
 gv\_objektschl = 'MAENGELSBESEITIGUNG'.  
 ENDIF.  
 ENDCASE.  
  
 WHEN 'MELDTEXT'.  
  
 PERFORM get\_template\_url CHANGING gv\_url.  
 IF gv\_doc\_open NE 'X'.  
 PERFORM load\_document USING gv\_url gs\_vorl-doc.  
 ELSE.  
 MESSAGE i000(38) WITH TEXT-009. *" Bitte erst die geöffnete Datei schliessen*  
 ENDIF.  
  
 WHEN 'VERSION'.  
  
 IF gv\_doc\_open = 'X'.  
 FREE lt\_sel\_rows.  
 CALL METHOD go\_alv->get\_selected\_rows  
 IMPORTING  
 et\_index\_rows = lt\_sel\_rows.  
  
 LOOP AT lt\_sel\_rows INTO ls\_sel\_rows.  
 READ TABLE gt\_vorl INTO ls\_vorl INDEX ls\_sel\_rows-index.  
 ENDLOOP.  
 gv\_url = ls\_vorl-uri.  
 IF gv\_url IS INITIAL.  
*\* MESSAGE i000(38) WITH TEXT-013. " Bitte erst eine Vorlage auswählen!*  
 ELSE.  
 PERFORM save\_as\_new\_version USING ls\_sel\_rows-index.  
 UPDATE qmel SET zz\_afm\_1frerstelldat = viqmel-zz\_afm\_1frerstelldat *" 1.Frist am*  
 WHERE qmnum = viqmel-qmnum. *" ZM Meldung!*  
 CLEAR gv\_vorl.  
 PERFORM cleanup.  
 ENDIF.  
 ELSE.  
 MESSAGE e000(38) WITH TEXT-013.  
 ENDIF.  
  
 WHEN 'SAVE'.  
  
 PERFORM document\_changed.  
*\* PERFORM save\_document\_as.*  
 CLEAR gv\_changed.  
 CLEAR gv\_vorl.  
*\* gv\_saved = 'X'.*  
 PERFORM cleanup.  
  
 WHEN 'CLOSE'.  
  
 PERFORM document\_changed.  
 CLEAR gv\_changed.  
 CLEAR gv\_vorl.  
 PERFORM cleanup.  
  
 WHEN 'MODUS'.  
  
 CASE rb\_inplace.  
 WHEN 'X'.  
 gv\_inplace = '1'.  
 WHEN OTHERS.  
 CLEAR gv\_inplace.  
 ENDCASE.  
  
 WHEN 'BDN'. *"Absprung in den BDN / OAER*  
  
 FREE lt\_sel\_rows.  
 CALL METHOD go\_alv->get\_selected\_rows  
 IMPORTING  
 et\_index\_rows = lt\_sel\_rows.  
 LOOP AT lt\_sel\_rows INTO ls\_sel\_rows.  
 READ TABLE gt\_vorl INTO ls\_vorl INDEX ls\_sel\_rows-index.  
 ENDLOOP.  
 gv\_url = ls\_vorl-uri.  
 IF gv\_url IS INITIAL.  
 MESSAGE i000(38) WITH TEXT-007. *" Bitte erst eine Vorlage auswählen!*  
 ELSE.  
 PERFORM show\_navigator USING ls\_sel\_rows-index.  
 ENDIF.  
  
 WHEN 'AKTU'.  
  
 PERFORM sachverhalt\_aktual.  
  
 WHEN OTHERS.  
  
*\* BACK/TAB-Wechsel etc.*  
 PERFORM document\_changed.  
 CLEAR gv\_changed.  
  
 ENDCASE.  
  
*\* ---------------------------------------------------------------------------*  
  
 sy-ucomm = lv\_ucomm. *" evt. durch YES überschrieben - POPUP wollen sie speichern!*  
  
 CASE sy-ucomm.  
 WHEN 'MELDTEXT' OR 'VERSION' OR 'SAVE' OR 'CLOSE' OR  
 'VORLAGEN' OR 'MODUS' OR  
*\* 'YES' OR " wollen sie speichern!*  
 'BDN' OR 'OPT1' OR 'AKTU' OR 'CANC' OR 'BAUVERTRAG'. *"OR 'OPT2'*  
 PERFORM supress\_f31(sapliqs0).  
 WHEN OTHERS.  
 ENDCASE.  
  
 ENDIF.  
  
ENDMODULE.

Include🡪 ZXQQMF17

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF17.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form LOAD\_DOCUMENT*  
*\*&---------------------------------------------------------------------\**  
*\* Sofern eine Datei ausgewählt worden ist, hier die nötigen*  
*\* Instanzen erzeugen und die jeweiligen Methoden zur Anzeige*  
*\* aufrufen*  
*\*----------------------------------------------------------------------\**  
FORM load\_document USING iv\_url doc.  
  
 TYPES: tyt\_errors TYPE STANDARD TABLE OF REF TO i\_oi\_error  
 WITH NON-UNIQUE DEFAULT KEY.  
  
 TABLES: tiv3g, tiv3h.  
  
 DATA: lo\_error TYPE REF TO i\_oi\_error,  
 lt\_errors TYPE tyt\_errors,  
 ls\_ole\_sheet TYPE ole2\_object,  
 ls\_ole\_appl TYPE ole2\_object,  
 ls\_ole\_cell TYPE ole2\_object,  
 ls\_handle TYPE cntl\_handle.  
  
 DATA lt\_class TYPE TABLE OF sclass.  
 DATA ls\_class TYPE sclass.  
 DATA lt\_objectdata TYPE TABLE OF clobjdat.  
 DATA ls\_objectdata TYPE clobjdat.  
 DATA lv\_objnr TYPE objnum. *"C50*  
 DATA lv\_betreuer TYPE atwrt.  
 DATA lv\_betr\_tel TYPE atwrt.  
  
 DATA lv\_gwl\_meld TYPE qmnum.  
 DATA lv\_ebeln TYPE ebeln.  
 DATA lv\_lifnr TYPE elifn.  
 DATA lv\_ekgrp TYPE ekgrp.  
 DATA lv\_ihrez TYPE ihrez.  
 DATA lv\_unsez TYPE unsez.  
 DATA lv\_txz01 TYPE txz01.  
 DATA lv\_erdat TYPE erdat.  
 DATA lv\_pos1 TYPE ebelp.  
 DATA ls\_lfa1 TYPE lfa1.  
 DATA ls\_t024 TYPE t024.  
 DATA ls\_adrc TYPE adrc.  
  
*\* Zusätzliche Lokale Variablen für Mängelmeldung Korrespondenz Vorlagen*  
 DATA lv\_mandn TYPE name1\_bas. *" Mandant Name*  
 DATA lv\_mands TYPE stras\_gp. *" Mandant Straße und Nummer*  
 DATA lv\_mandp TYPE pstlz\_bas. *" Mandant PLZ*  
 DATA lv\_mando TYPE pfort\_adrs. *" Mandant Standort*  
 DATA lv\_txcdsm TYPE qtxtcdgrsm. *" Kurztext Maßnahmecode*  
 DATA lv\_peter TYPE peter . *" Geplanter Endtermin der Maßnahme*  
 DATA lv\_gwende TYPE viqmel-zz\_afm\_gewende. *" Gewährleistung Endetermin*  
 DATA lv\_tplnr TYPE tplnr. *" Techn. Platz*  
 DATA ls\_ltxt TYPE string.  
 DATA lv\_datum TYPE c LENGTH 10.  
 DATA lv\_datum\_org LIKE sy-datum.  
  
*\* Daten RE-FX Partnerdaten*  
 DATA lv\_bukrs TYPE bukrs.  
 DATA lv\_we TYPE rebdbeno.  
 DATA lv\_mo TYPE rebdrono.  
  
*\* Daten RE-FX Gebäudedaten*  
 DATA lv\_compcode\_tp TYPE bapi\_re\_building\_key-comp\_code.  
 DATA lv\_businessentitynumber\_tp TYPE bapi\_re\_building\_key-business\_entity.  
 DATA lv\_buildingnumber\_tp TYPE bapi\_re\_building\_key-building.  
  
 DATA ls\_object\_info TYPE bapi\_re\_rental\_object. *" <-- Mietobjekt*  
  
 DATA lv\_sb\_name TYPE string.  
 DATA lv\_sb\_tel TYPE string.  
 DATA lv\_sb\_email TYPE string.  
 DATA ls\_partner\_head TYPE bus000.  
  
 DATA ls\_address TYPE bus000\_ext.  
  
 DATA gt\_partner\_tel TYPE TABLE OF adtel.  
 DATA ls\_partner\_tel TYPE adtel.  
 DATA gt\_partner\_smtp TYPE TABLE OF adsmtp.  
 DATA ls\_partner\_smtp TYPE adsmtp.  
 DATA lv\_partner1\_anrede TYPE string.  
 DATA lv\_partner2\_anrede TYPE string.  
 DATA lv\_partner1\_name TYPE string.  
 DATA lv\_partner1\_vorname TYPE string.  
 DATA lv\_partner2\_name TYPE string.  
 DATA lv\_partner2\_vorname TYPE string.  
 DATA: lv\_street\_k TYPE string,  
 lv\_house\_num1\_k TYPE string,  
 lv\_post\_code1\_k TYPE string,  
 lv\_city1\_k TYPE string.  
 DATA lv\_partner\_tel\_1 TYPE string.  
 DATA lv\_partner\_tel\_2 TYPE string.  
 DATA lv\_partner\_smtp TYPE string.  
 DATA lv\_value TYPE string.  
 DATA gt\_partner\_mv TYPE TABLE OF bapi\_re\_partner.  
 DATA ls\_partner\_mv TYPE bapi\_re\_partner.  
 DATA lv\_partner\_status TYPE string.  
 DATA ls\_tplnr\_data TYPE bapi\_itob.  
  
*\* Partnerdaten RE-FX*  
 DATA gt\_partner TYPE TABLE OF bapi\_re\_partner.  
 DATA gs\_partner TYPE bapi\_re\_partner.  
  
 DATA ls\_thead TYPE thead.  
 DATA lt\_lines TYPE TABLE OF tline.  
 DATA ls\_lines TYPE tline.  
 DATA lt\_result\_tab TYPE match\_result\_tab.  
  
 DATA lv\_index TYPE sy-index.  
 DATA ls\_viqmsm TYPE wqmsm.  
  
 lv\_objnr = viqmel-tplnr.  
  
 DATA wa\_kopf TYPE bapi2080\_nothdre. *"SM Meldung Kopfdaten*  
 DATA wa\_kopf\_stoer TYPE bapi2080\_nothdre. *"SM Meldung Kopfdaten*  
 DATA wa\_bakf TYPE bapi2080\_nothdre. *"BA Meldung Kopfdaten*  
 DATA lt\_masst TYPE TABLE OF bapi2080\_nottaske.  
 DATA ls\_masst TYPE bapi2080\_nottaske.  
 DATA lt\_masst\_stoer TYPE TABLE OF bapi2080\_nottaske.  
 DATA ls\_masst\_stoer TYPE bapi2080\_nottaske.  
 DATA lt\_pardt TYPE TABLE OF bapi2080\_notpartnre.  
 DATA ls\_pardt TYPE bapi2080\_notpartnre.  
 DATA lt\_pardt\_stoer TYPE TABLE OF bapi2080\_notpartnre.  
 DATA ls\_pardt\_stoer TYPE bapi2080\_notpartnre.  
 DATA lv\_paran TYPE ad\_titletx. *" Partner Anrede*  
 DATA lv\_parnm TYPE ad\_name1. *" Partner Name*  
 DATA lv\_parst TYPE ad\_street. *" Partner Straße*  
 DATA lv\_parnu TYPE ad\_hsnm1. *" Partner Hausnummer*  
 DATA lv\_parpl TYPE ad\_pstcd1. *" Partner PLZ*  
 DATA lv\_paror TYPE ad\_city1. *" Partner Ort*  
 DATA lv\_parnr TYPE i\_parnr. *" Partner Nummer*  
 DATA lv\_objnn TYPE j\_objnr. *" Objekt Nummer*  
 DATA lv\_parvw TYPE parvw. *"*  
 DATA wa\_diadrp TYPE diadrp. *" Adresse Struktur*  
 DATA lv\_parad TYPE ad\_addrnum.  
 DATA lv\_1frer TYPE peter.  
 DATA lv\_2frer TYPE peter.  
 DATA ls\_qmel TYPE qmel.  
 DATA lv\_2frdt TYPE erdat.  
 DATA lv\_ersdt TYPE erdat.  
 DATA lt\_tp TYPE TABLE OF iflo.  
 DATA ls\_tp TYPE iflo.  
 DATA lv\_gebau TYPE pltxt.  
 DATA lv\_obtp TYPE pltxt.  
 DATA ls\_addr TYPE ad\_addrnum.  
 DATA lt\_adrc TYPE TABLE OF adrc.  
 DATA lv\_adna TYPE string.  
 DATA wa\_behe TYPE bapimepoheader.  
 DATA lv\_berda TYPE erdat.  
 DATA lt\_ekpo TYPE TABLE OF bapiekpo.  
 DATA wa\_ekpo TYPE bapiekpo.  
 DATA lv\_baktx TYPE qmtxt.  
 DATA lv\_bamld TYPE qmnum.  
 DATA lv\_gmmld TYPE qmtxt.  
 DATA lv\_pltxt TYPE pltxt.  
  
 DATA ls\_iflo TYPE iflo.  
 DATA ls\_adrc\_tp TYPE adrc.  
 DATA lv\_tplnplz TYPE ad\_city1.  
 DATA lv\_tplnort TYPE ad\_citynum.  
 DATA lv\_tplntxt TYPE pltxt.  
 DATA lv\_text TYPE c LENGTH 40.  
  
 TYPES:BEGIN OF ty\_ls\_gew.  
 INCLUDE TYPE viqmel.  
 TYPES parnr TYPE i\_parnr.  
 TYPES END OF ty\_ls\_gew.  
  
 DATA ls\_viqmel TYPE ty\_ls\_gew.  
  
*\*----------------------------------------------------------------------\**  
  
 IF go\_container IS INITIAL.  
 CREATE OBJECT go\_container  
 EXPORTING  
*\* parent =*  
 container\_name = 'C\_TEXT' *" <-- auf Dynpro 9000*  
*\* style =*  
*\* lifetime = lifetime\_default*  
*\* repid =*  
*\* dynnr =*  
*\* no\_autodef\_progid\_dynnr =*  
 EXCEPTIONS  
 cntl\_error = 1  
 cntl\_system\_error = 2  
 create\_error = 3  
 lifetime\_error = 4  
 lifetime\_dynpro\_dynpro\_link = 5  
 OTHERS = 6.  
 IF sy-subrc <> 0.  
 MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
 WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
 ENDIF.  
 ENDIF.  
  
*\* Instanz der OI erzeugen*  
 CALL METHOD c\_oi\_container\_control\_creator=>get\_container\_control  
 IMPORTING  
 control = go\_control  
 error = lo\_error.  
*\* retcode =*  
 APPEND lo\_error TO lt\_errors.  
  
*\* Control initialisieren*  
 CALL METHOD go\_control->init\_control  
 EXPORTING  
*\* dynpro\_nr = SY-DYNNR*  
*\* gui\_container = ' '*  
 inplace\_enabled = gv\_inplace *"' ' or 1*  
*\* inplace\_mode = 0*  
*\* inplace\_resize\_documents = ' '*  
*\* inplace\_scroll\_documents = ' '*  
*\* inplace\_show\_toolbars = 'X'*  
 no\_flush = 'x'  
*\* parent\_id = cl\_gui\_cfw=>dynpro\_0*  
 r3\_application\_name = 'Meldung'  
*\* register\_on\_close\_event = ' '*  
*\* register\_on\_custom\_event = ' '*  
*\* rep\_id = SY-REPID*  
*\* shell\_style = 1384185856*  
 parent = go\_container  
*\* name =*  
*\* autoalign = 'x'*  
 IMPORTING  
 error = lo\_error  
*\* retcode =*  
 EXCEPTIONS  
 javabeannotsupported = 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.  
 APPEND lo\_error TO lt\_errors.  
  
*\* Get document proxy*  
 CALL METHOD go\_control->get\_document\_proxy  
 EXPORTING  
*\* document\_format = 'NATIVE'*  
 document\_type = 'Word.Document'  
 no\_flush = 'X'  
*\* register\_container = ' '*  
 IMPORTING  
 document\_proxy = go\_document  
 error = lo\_error.  
*\* retcode =*  
 APPEND lo\_error TO lt\_errors.  
  
*\* Dokument laden*  
 CALL METHOD go\_document->open\_document  
 EXPORTING  
 document\_title = ' '  
 document\_url = iv\_url  
 no\_flush = 'X'  
 open\_inplace = gv\_inplace *"' ' or 1*  
*\* open\_readonly = ' '*  
*\* protect\_document = ' '*  
*\* onsave\_macro = ' '*  
*\* startup\_macro = ''*  
*\* user\_info =*  
 IMPORTING  
 error = lo\_error.  
*\* retcode =*  
 APPEND lo\_error TO lt\_errors.  
  
 IF lo\_error->has\_succeeded = 'X'.  
 gv\_doc\_open = 'X'.  
 ENDIF.  
  
 CALL METHOD go\_document->is\_open  
 EXPORTING  
 no\_flush = ' '  
 IMPORTING  
 error = lo\_error  
*\* RETCODE =*  
 ret\_value = gv\_is\_open.  
 APPEND lo\_error TO lt\_errors.  
  
 CASE gv\_is\_open.  
 WHEN 1.  
 gv\_init = 'X'.  
 IF sy-ucomm = 'VORLAGEN'.  
 gv\_vorl = 'X'.  
 ENDIF.  
*\* Generierung Word-Interface*  
 CALL METHOD go\_document->get\_wordprocessor\_interface  
 EXPORTING  
 no\_flush = ' '  
 IMPORTING  
 error = lo\_error  
*\* RETCODE =*  
 wp\_interface = go\_worddoc.  
 WHEN OTHERS.  
 ENDCASE.  
 APPEND lo\_error TO lt\_errors.  
  
*\* Abfrage nach dem Automation-Handle der Dokumentes*  
 CALL METHOD go\_document->get\_document\_handle  
 EXPORTING  
 no\_flush = 'X'  
 IMPORTING  
 error = lo\_error  
 handle = ls\_handle.  
*\* retcode =*  
 APPEND lo\_error TO lt\_errors.  
  
*\*----------------------------------------------------------------------\**  
  
 DATA lv\_stoernummer TYPE /afm/stoermeldnummer.  
 DATA ls\_viqmel\_stoer TYPE viqmel.  
 DATA lv\_pspnr TYPE ps\_posid.  
 DATA ls\_po\_header TYPE bapiekkol.  
 DATA lv\_days TYPE i.  
 DATA lv\_zuordnung TYPE c LENGTH 4.  
 DATA lv\_frist LIKE sy-datum.  
 DATA lv\_post1 TYPE ps\_post1.  
 DATA lv\_xwetext TYPE rebdxbe.  
  
 DATA: mystring TYPE string.  
  
 TABLES: tsad3t,  
 ekkn,  
 iflot.  
  
*\* Störmeldung ermitteln*  
 SELECT SINGLE zz\_afm\_stoermeldnummer FROM viqmel INTO lv\_stoernummer WHERE qmnum = viqmel-qmnum.  
  
*\* Infos zur Störmeldung lesen*  
 SELECT SINGLE \* FROM viqmel INTO ls\_viqmel\_stoer WHERE qmnum = lv\_stoernummer. *" QMART = Z4*  
  
*\* Infos zum TPLNR lesen*  
 SELECT SINGLE fltyp FROM iflot INTO iflot-fltyp  
 WHERE tplnr = viqmel-tplnr.  
  
*\* Projekt aus der Bestellung ziehen. Erste Position stellt den Hauptauftrag dar*  
 SELECT SINGLE MIN( ebelp ) FROM ekpo INTO lv\_pos1  
 WHERE ebeln EQ viqmel-zz\_afm\_ebeln  
 AND loekz EQ space.  
  
*\*----------------------------------------------------------------------\**  
  
 DATA: lv\_compcode TYPE bapi\_re\_bus\_entity\_key-comp\_code,  
 lv\_businessentitynumber TYPE bapi\_re\_bus\_entity\_key-business\_entity.  
 DATA: ls\_object\_address TYPE bapi\_re\_obj\_address.  
 DATA: ls\_building TYPE bapi\_re\_building.  
  
  
*\*Prüfen ob eine Vorlage oder ein bestehendes Document geöffnet wurde*  
 IF doc IS INITIAL.  
  
 IF sy-ucomm = 'VORLAGEN'. *"Replace nur beim öffnen von Vorlagen, nicht mehr bei Texten zur Meldung*  
  
 IF gv\_qmart = 'SM' OR gv\_qmart = 'SG'.  
 IF gv\_qmart = 'SM' AND fl\_txtma = '1'.  
  
*\*\*\* Einlesen der Ersetzungsparameter und Füllen der Replace Tabelle*  
  
 SELECT \* FROM /afm/formpar INTO TABLE gt\_gmpar WHERE sprache = sy-langu.  
  
*\*\*\* Selektierte Maßnahme lesen*  
  
 LOOP AT inxsm.  
 IF inxsm-mark = 'X'.  
 lv\_index = inxsm-tabpt.  
 ENDIF.  
 ENDLOOP.  
  
 READ TABLE iviqmsm INTO ls\_viqmsm INDEX lv\_index.  
  
*\*\*\* Störmeldung lesen*  
  
 CALL FUNCTION 'BAPI\_ALM\_NOTIF\_GET\_DETAIL'  
 EXPORTING  
 number = lv\_stoernummer  
 IMPORTING  
 notifheader\_export = wa\_kopf\_stoer  
 TABLES  
 notiftask = lt\_masst\_stoer  
 notifpartnr = lt\_pardt\_stoer.  
  
*\*\*\* Mängelmeldung lesen*  
  
 CALL FUNCTION 'BAPI\_ALM\_NOTIF\_GET\_DETAIL'  
 EXPORTING  
 number = viqmel-qmnum  
 IMPORTING  
 notifheader\_export = wa\_kopf  
 TABLES  
 notiftask = lt\_masst  
 notifpartnr = lt\_pardt.  
*\* notlongtxt = lt\_notlongtxt.*  
  
  
*\*\*\* Bestellung lesen*  
  
 lv\_ebeln = viqmel-zz\_afm\_ebeln.  
 CALL FUNCTION 'BAPI\_PO\_GETDETAIL'  
 EXPORTING  
 purchaseorder = lv\_ebeln  
 IMPORTING  
 po\_header = ls\_po\_header  
 TABLES  
 po\_items = lt\_ekpo.  
  
*\*\*\* Lieferantendaten*  
  
*\* Holen in lokalen Variablen die Lieferanten Stammdaten*  
 SELECT SINGLE lifnr ekgrp aedat ihrez unsez FROM ekko INTO (lv\_lifnr, lv\_ekgrp, lv\_erdat, lv\_ihrez, lv\_unsez)  
 WHERE ebeln EQ lv\_ebeln.  
 IF sy-subrc = 0.  
 SELECT SINGLE \* FROM lfa1 INTO ls\_lfa1  
 WHERE lifnr EQ lv\_lifnr  
 AND spras EQ sy-langu.  
 ENDIF.  
  
*\*\*\* PSP Informationen aus Bestellung Ebene1 lesen - PSP Hauptprojekt definieren ps\_psp\_pnr*  
  
 DATA: psp LIKE prps-pspnr. *"Projekt PSP int.Nr*  
 DATA: wf\_posid LIKE prps-posid. *"PSP externe Nr.*  
  
 DATA lv\_tabix.  
  
*\*\* SELECT SINGLE psphi FROM prps INTO lv\_posid WHERE posid = viqmel-zz\_dt\_pspel. " nicht aus Z4 !*  
 SELECT SINGLE ps\_psp\_pnr FROM ekkn INTO ekkn-ps\_psp\_pnr  
 WHERE ebeln = viqmel-zz\_afm\_ebeln  
 AND ebelp = lv\_pos1.  
 SELECT SINGLE psphi FROM prps INTO psp  
 WHERE pspnr EQ ekkn-ps\_psp\_pnr.  
 CALL FUNCTION 'CONVERSION\_EXIT\_KONPD\_OUTPUT'  
 EXPORTING  
 input = psp *"NUMC 8*  
 IMPORTING  
 output = wf\_posid. *"CHAR 24*  
 SELECT SINGLE post1 FROM proj INTO lv\_post1  
 WHERE pspnr EQ psp.  
  
*\*\*\* Ermittlung 1. + 2. Frist - Priorität aus Mängelmeldung(ZM) statt Störmeldung(Z4) + x Tage*  
*\* 1 = hoch ( 7 Tage)*  
*\* 2 = mittel (14 Tage)*  
*\* 3 = niedrig (21 Tage)*  
*\* n = Notfall ( 1 Tag)*  
 lv\_datum\_org = sy-datum.  
 CASE viqmel-priok.  
 WHEN '1'.  
 lv\_frist = lv\_datum\_org + 7.  
 WHEN '2'.  
 lv\_frist = lv\_datum\_org + 14.  
 WHEN '3'.  
 lv\_frist = lv\_datum\_org + 21.  
 WHEN 'N'.  
 lv\_frist = lv\_datum\_org + 1.  
 ENDCASE.  
  
 IF lv\_classname = 'ZKORR\_MAENGEL\_1' OR lv\_classname = 'ZKORR\_MAENGEL\_INFO\_1'.  
 IF ls\_vorl-comp\_id+0(12) <> viqmel-qmnum. *" <-- Dokument geöffnet - Datum nicht NEU setzen!*  
 viqmel-zz\_afm\_1frerstelldat = lv\_frist.  
 ENDIF.  
 ENDIF.  
  
*\*\*\* Detailsinfos RE-FX Mietvertrag- wenn es eine Mieteinheit ist - FLTYP=6*  
  
 CLEAR: gt\_partner\_mv, gt\_partner\_mv[],  
 ls\_partner\_mv.  
  
 CALL FUNCTION 'BAPI\_RE\_CN\_GET\_DETAIL'  
 EXPORTING  
 compcode = ls\_viqmel\_stoer-bukrs  
 contractnumber = ls\_viqmel\_stoer-zz\_dt\_kc\_recnnr  
 TABLES  
 partner = gt\_partner\_mv.  
  
 CLEAR: lv\_partner\_tel\_1, lv\_partner\_tel\_2,  
 ls\_partner\_tel.  
  
 CLEAR lv\_tabix.  
 LOOP AT gt\_partner\_mv INTO ls\_partner\_mv.  
 lv\_tabix = sy-tabix.  
*\* Status des Partners definieren*  
 IF ls\_partner\_mv-role\_type = 'TR0600'.  
 lv\_partner\_status = 'Mieter'.  
 ELSEIF ls\_partner\_mv-role\_type = 'TR0800'.  
 lv\_partner\_status = 'Eigentümer'.  
 ENDIF.  
*\* Detaildaten Partner ermitteln*  
 CLEAR ls\_partner\_head.  
 FREE: gt\_partner\_tel, gt\_partner\_smtp.  
 CALL FUNCTION 'BUP\_PARTNER\_GET'  
 EXPORTING  
 i\_partner = ls\_partner\_mv-partner  
 i\_valdt\_sel = sy-datlo  
 IMPORTING  
 e\_but000 = ls\_partner\_head  
 e\_but000\_address = ls\_address  
 TABLES  
 t\_adtel = gt\_partner\_tel  
 t\_adsmtp = gt\_partner\_smtp  
 EXCEPTIONS  
 partner\_not\_found = 1  
 wrong\_parameters = 2  
 internal\_error = 3  
 blocked\_partner = 4  
 OTHERS = 5.  
 IF sy-subrc <> 0.  
 CLEAR: ls\_partner\_head, ls\_partner\_head,  
 gt\_partner\_tel, gt\_partner\_tel[], gt\_partner\_smtp, gt\_partner\_smtp[].  
 ENDIF.  
*\* zum Mieter / Eigentümer*  
 IF lv\_tabix = 1.  
 lv\_partner1\_anrede = ls\_partner\_head-title.  
 lv\_partner1\_name = ls\_partner\_head-name\_last.  
 lv\_partner1\_vorname = ls\_partner\_head-name\_first.  
 ELSEIF lv\_tabix = 2.  
 lv\_partner2\_anrede = ls\_partner\_head-title.  
 lv\_partner2\_name = ls\_partner\_head-name\_last.  
 lv\_partner2\_vorname = ls\_partner\_head-name\_first.  
 ENDIF.  
*\* Addresse - Strasse Hausnr. PLZ Ort*  
 lv\_street\_k = ls\_address-street.  
 lv\_house\_num1\_k = ls\_address-house\_num1.  
 lv\_post\_code1\_k = ls\_address-post\_code1.  
 lv\_city1\_k = ls\_address-city1.  
*\* Telefon 1 bzw. HANDY*  
 CLEAR ls\_partner\_tel.  
 IF lv\_tabix = 1.  
 READ TABLE gt\_partner\_tel INTO ls\_partner\_tel INDEX 1.  
 IF sy-subrc = 0.  
 lv\_partner\_tel\_1 = ls\_partner\_tel-tel\_number.  
 ENDIF.  
*\* Telefon 2 bzw. HANDY*  
 READ TABLE gt\_partner\_tel INTO ls\_partner\_tel INDEX 2.  
 IF sy-subrc = 0.  
 lv\_partner\_tel\_2 = ls\_partner\_tel-tel\_number.  
 ENDIF.  
 ENDIF.  
*\* E-Mail*  
 CLEAR ls\_partner\_smtp.  
 READ TABLE gt\_partner\_smtp INTO ls\_partner\_smtp INDEX 1.  
 IF lv\_partner\_smtp IS INITIAL.  
 lv\_partner\_smtp = ls\_partner\_smtp-smtp\_addr.  
 ENDIF.  
 ENDLOOP.  
  
*\*\*\* Detailsinfos wenn es ein Gebäude ist - FLTYP=5*  
  
 IF gt\_partner\_mv IS INITIAL.  
 IF iflot-fltyp = '5'.  
 lv\_partner\_status = 'Eigentümer'.  
 ENDIF.  
 ENDIF.  
  
*\*\*\* Sachbearbeiter(USER) aus VIQMEL-ARBPL(ZM) ermitteln*  
  
 DATA: lv\_uktext LIKE riwo00-ktext.  
 DATA: lv\_name\_first TYPE ad\_namefir,  
 lv\_name\_last TYPE ad\_namelas.  
 TABLES: adrp, adr6, adcp, adrc.  
  
 DATA lv\_arbpl TYPE lgwid.  
  
*\* ARBPL - wird im UE: ZXWOCU21 zur Mängelmeldung(ZM) ermitelt!*  
  
 CLEAR lv\_arbpl.  
 lv\_arbpl = viqmel-arbpl.  
*\* zum ARBPL für Korrespondenzdaten (Name, Telefon, mail etc.) ermittelt*  
 CLEAR lv\_uktext.  
 CALL FUNCTION 'CR\_WORKSTATION\_READ'  
 EXPORTING  
 id = lv\_arbpl  
 msgty = 'S'  
 IMPORTING  
*\* arbpl =*  
*\* werks =*  
 ktext = lv\_uktext.  
 SPLIT lv\_uktext AT ',' INTO lv\_name\_last lv\_name\_first.  
 SHIFT lv\_name\_first LEFT DELETING LEADING space.  
  
 CLEAR: adrp-persnumber, adrp-name\_text.  
 SELECT SINGLE persnumber name\_text FROM adrp INTO ( adrp-persnumber, adrp-name\_text )  
 WHERE name\_first = lv\_name\_first  
 AND name\_last = lv\_name\_last  
 AND pers\_group = 'BC01'. *" SAP-Benutzer*  
  
*\* E-Mail - Mitarbeiter*  
 CLEAR: adr6-addrnumber, adr6-smtp\_addr.  
 IF adrp-persnumber IS NOT INITIAL.  
 SELECT SINGLE addrnumber smtp\_addr FROM adr6 INTO ( adr6-addrnumber, adr6-smtp\_addr )  
 WHERE persnumber = adrp-persnumber.  
 ENDIF.  
  
*\* Telefon mit Leerzeichen - Mitarbeiter*  
 CLEAR: adrc-tel\_number.  
 IF adr6-addrnumber IS NOT INITIAL.  
 SELECT SINGLE tel\_number FROM adrc INTO adrc-tel\_number  
 WHERE addrnumber = adr6-addrnumber.  
 ENDIF.  
*\* Durchwahl - Mitarbeiter*  
 CLEAR: adcp-tel\_extens.  
 IF adrp-persnumber IS NOT INITIAL.  
 SELECT SINGLE tel\_extens FROM adcp INTO adcp-tel\_extens  
 WHERE persnumber = adrp-persnumber.  
 IF adcp-tel\_extens IS INITIAL.  
 adcp-tel\_extens = '0'.  
 ENDIF.  
 ENDIF.  
  
*\*\*\* Bez.WE Wirtschaftseinheit ermitteln - aus Bestellung*  
  
*\* CLEAR: lv\_imkey, lv\_xwetext.*  
*\* SELECT SINGLE imkey FROM ekkn INTO lv\_imkey*  
*\* WHERE ebeln = viqmel-zz\_afm\_ebeln*  
*\* AND ebelp = lv\_pos1.*  
*\* IF sy-subrc = 0.*  
*\* SELECT SINGLE xwetext FROM vibdbe INTO lv\_xwetext*  
*\* WHERE imkey = lv\_imkey.*  
*\* ENDIF.*  
  
*\*\*\* Wirtschaftseinheit-Daten zurückliefern - aus Mietobjekt*  
  
 CLEAR: ls\_object\_address,  
 lv\_xwetext.  
 lv\_compcode = ls\_viqmel\_stoer-bukrs.  
 lv\_businessentitynumber = ls\_viqmel\_stoer-zz\_dt\_kc\_recnnr+0(4).  
 CALL FUNCTION 'BAPI\_RE\_BE\_GET\_DETAIL'  
 EXPORTING  
 compcode = lv\_compcode *" BUKRS*  
 businessentitynumber = lv\_businessentitynumber *" WE*  
 detail\_data\_from = sy-datum *" reca0\_date-min*  
 detail\_data\_to = reca0\_date-max *" '99991231'*  
 IMPORTING  
 object\_address = ls\_object\_address.  
 lv\_xwetext = ls\_object\_address-object\_id.  
  
 ENDIF. *" gv\_qmart = 'SM'*  
  
 ELSE.  
 ENDIF. *" gv\_qmart = 'SM' OR gv\_qmart = 'SG'.*  
  
*\*----------------------------------------------------------------------\**  
  
*\*\*\* Datum der Erstellung des Dokuments ermitteln*  
  
 DATA lv\_datum\_frist1 LIKE sy-datum.  
 DATA gt\_signature TYPE TABLE OF bapisignat.  
 DATA ls\_signature TYPE bapisignat.  
 DATA lv\_objkey TYPE bds\_typeid.  
 DATA lv\_clsnam TYPE bds\_clsnam.  
 DATA lv\_clstyp TYPE bds\_clstyp.  
  
 SELECT SINGLE \* FROM /afm/oaer INTO gs\_oaer WHERE mncod = 'F1'.  
 lv\_clsnam = gs\_oaer-cls\_nam.  
 lv\_clstyp = gs\_oaer-clstyp.  
 CONCATENATE viqmel-qmnum 'F1' INTO lv\_objkey.  
  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_INFO'  
 EXPORTING  
*\* LOGICAL\_SYSTEM =*  
 classname = lv\_clsnam  
 classtype = lv\_clstyp  
 client = sy-mandt  
 object\_key = lv\_objkey  
 all = 'X'  
*\* CHECK\_STATE = ' '*  
 TABLES  
*\* COMPONENTS =*  
 signature = gt\_signature  
*\* CONNECTIONS =*  
*\* EXTENDED\_COMPONENTS =*  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
 ENDIF.  
  
 LOOP AT gt\_signature INTO ls\_signature.  
 IF ls\_signature-prop\_name = 'CREATED\_AT'.  
 lv\_datum\_frist1 = ls\_signature-prop\_value+0(8).  
 ENDIF.  
 ENDLOOP.  
  
*\*\*\* Lage der Wohnung MO aus TPLNR / eventuell handelt es sich auch nur um das GEBÄUDE! 1001-4562-0002*  
  
*\*\* Infos technischer Platz*  
*\* CALL FUNCTION 'BAPI\_FUNCLOC\_GETDETAIL'*  
*\* EXPORTING*  
*\* functlocation = ls\_viqmel\_stoer-tplnr*  
*\* IMPORTING*  
*\* data\_general\_exp = ls\_tplnr\_data.*  
  
 CLEAR: ls\_object\_address,  
 ls\_object\_info.  
  
 IF iflot-fltyp = '5'.  
*\* Partnerinfo zum Gebäude lesen*  
 lv\_compcode\_tp = ls\_viqmel\_stoer-bukrs.  
 lv\_businessentitynumber\_tp = ls\_viqmel\_stoer-tplnr+5(4).  
 lv\_buildingnumber\_tp = ls\_viqmel\_stoer-tplnr+10(4).  
 CALL FUNCTION 'BAPI\_RE\_BU\_GET\_DETAIL' *" siehe auch --> CALL FUNCTION 'BAPI\_RE\_BE\_GET\_DETAIL'*  
 EXPORTING  
 compcode = lv\_compcode\_tp  
 businessentitynumber = lv\_businessentitynumber\_tp  
 buildingnumber = lv\_buildingnumber\_tp  
 IMPORTING  
 object\_address = ls\_object\_address.  
 CONCATENATE  
 ls\_object\_address-postl\_cod1 ` `  
 ls\_object\_address-city `, `  
 ls\_object\_address-street\_lng  
 ls\_object\_address-house\_no `, ` INTO ls\_object\_info-rental\_object\_text.  
 ELSE.  
*\* Partnerinfo zu Mietobjekt lesen*  
 lv\_bukrs = ls\_viqmel\_stoer-bukrs.  
 lv\_we = ls\_viqmel\_stoer-tplnr+5(4).  
 lv\_mo = ls\_viqmel\_stoer-tplnr+15(4).  
 CALL FUNCTION 'BAPI\_RE\_RO\_GET\_DETAIL'  
 EXPORTING  
 compcode = lv\_bukrs  
 businessentitynumber = lv\_we  
 rentalobjectnumber = lv\_mo  
 IMPORTING  
 rental\_object = ls\_object\_info  
 object\_address = ls\_object\_address.  
 CONCATENATE  
 ls\_object\_address-postl\_cod1 ` `  
 ls\_object\_address-city `, `  
 ls\_object\_address-street\_lng  
 ls\_object\_address-house\_no `, ` INTO ls\_object\_info-rental\_object\_text.  
*\* Etage - 2. OG*  
 CLEAR tiv3g-xstockk.  
 SELECT SINGLE xstockk FROM tiv3g INTO tiv3g-xstockk  
 WHERE spras = sy-langu  
 AND sstockw = ls\_object\_info-floor.  
*\* Lage - links*  
 CLEAR tiv3h-xklgesch.  
 SELECT SINGLE xklgesch FROM tiv3h INTO tiv3h-xklgesch  
 WHERE spras = sy-langu  
 AND rlgesch = ls\_object\_info-location\_on\_floor.  
 ENDIF.  
  
*\* Details Technischer Platz lesen*  
 CALL FUNCTION 'FUNC\_LOCATION\_READ'  
 EXPORTING  
*\* AUTH\_CHECK = ' '*  
 auth\_tcode = 'IL03'  
*\* BUFFER\_BYPASS = ' '*  
*\* DYFIELD = ' '*  
*\* NO\_OTHER\_LANGUAGE = ' '*  
 spras = sy-langu  
 tplnr = viqmel-tplnr  
 IMPORTING  
 iflo\_wa = ls\_iflo  
*\* PLTXT =*  
*\* EXCEPTIONS*  
*\* IFLOT\_NOT\_FOUND = 1*  
*\* ILOA\_NOT\_FOUND = 2*  
*\* NO\_AUTHORITY = 3*  
*\* OTHERS = 4*  
 .  
 IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
 ENDIF.  
  
*\* Adressdaten TPLNR sammeln*  
 CLEAR ls\_adrc\_tp.  
 SELECT SINGLE \* FROM adrc INTO ls\_adrc\_tp WHERE addrnumber = ls\_iflo-adrnr.  
  
 CLEAR: lv\_tplntxt, lv\_tplnplz, lv\_tplnort.  
  
*\* Bezeichnung Technischer Platz*  
 lv\_tplntxt = ls\_iflo-pltxt.  
 lv\_tplnplz = ls\_adrc\_tp-city1.  
 lv\_tplnort = ls\_adrc\_tp-city\_code.  
  
*\*----------------------------------------------------------------------\**  
  
*\* Die Platzhalterersetzungen wurden bewusst für die jeweiligen Länder separat*  
*\* erstellt und hier abgefragt, um evtl. Änderungen je Land flexibel zu halten*  
*\**  
*\* ACHTUNG: Aber wegen der Funktion der Aktualisierung des Sachverhaltes (Langtextes)*  
*\* im WORD-Dokument ist für den Langtext IMMER &13& in Tabelle /AFM/FORMPAR*  
*\* zu verwenden. Es wird im FORM SACHVERHALT\_AKTUAL nicht zwischen den Ländern unterscheiden,*  
*\* sondern lediglich nach dem Parameter &13& abgefragt.*  
  
 CLEAR gt\_replaces.  
 REFRESH gt\_replaces.  
  
 CASE sy-langu.  
 WHEN 'D'.  
 IF gv\_qmart = 'SM' OR gv\_qmart = 'SG'.  
 IF gv\_qmart = 'SM' AND fl\_txtma = '1'.  
  
 LOOP AT gt\_gmpar INTO gs\_gmpar.  
 CASE gs\_gmpar-param.  
  
*\*\*\* Adresse - Lieferant*  
 WHEN '&01&'.*" OR '&01\_1&'. "Lieferant Name*  
 gt\_replaces-string\_new = ls\_lfa1-name1.  
 gt\_replaces-string\_old = '&01&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&01\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 WHEN '&02&'.*" OR '&02\_1&'. "Lieferant Strasse*  
 gt\_replaces-string\_new = ls\_lfa1-stras.  
 gt\_replaces-string\_old = '&02&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&02\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 WHEN '&03&'.*" OR '&03\_1&'. "Lieferant PLZ Ort*  
 CONCATENATE ls\_lfa1-pstlz ls\_lfa1-ort01 INTO gt\_replaces-string\_new SEPARATED BY space.  
 gt\_replaces-string\_old = '&03&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&03\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
*\*\*\* Adresse - Mieter*  
 WHEN '&56&'.  
 CONCATENATE lv\_street\_k lv\_house\_num1\_k INTO gt\_replaces-string\_new SEPARATED BY space.  
 gt\_replaces-string\_old = '&56&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 WHEN '&57&'.  
 CONCATENATE lv\_post\_code1\_k lv\_city1\_k INTO gt\_replaces-string\_new SEPARATED BY space.  
 gt\_replaces-string\_old = '&57&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&04&'.*" OR '&04\_1&'. "Meldungsnummer*  
 gt\_replaces-string\_new = viqmel-qmnum.  
 gt\_replaces-string\_old = '&04&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&04\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&05&'.*" OR '&05\_1&'. "Bezeichung Projekt PSP aus Bestellung Ebene 1*  
 gt\_replaces-string\_new = lv\_post1.  
 gt\_replaces-string\_old = '&05&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&05\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&06&'.*" OR '&06\_1&'. "Verwaltungseinheit - Bez. der Wirtschaftseinheit*  
 gt\_replaces-string\_new = lv\_xwetext. *"lv\_txz01.*  
 gt\_replaces-string\_old = '&06&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&06\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&60&'. *" Technischer Platz PLZ & ORT*  
 CLEAR lv\_text.  
 CONCATENATE lv\_tplnplz lv\_tplnort INTO lv\_text SEPARATED BY space.  
 gt\_replaces-string\_new = lv\_text.  
 gt\_replaces-string\_old = '&60&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&60\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&61&'. *" Technischer Platz Beschreibung*  
 gt\_replaces-string\_new = lv\_tplntxt.  
 gt\_replaces-string\_old = '&61&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&61\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&07&'.*" OR '&07\_1&'. "Adresse PLZ + Ort - Mietobjekt / Gebäude*  
*\* IF iflot-fltyp = '6'.*  
 gt\_replaces-string\_new = ls\_object\_info-rental\_object\_text.  
 gt\_replaces-string\_old = '&07&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&07\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
*\* ELSEIF iflot-fltyp = '5'.*  
*\* ENDIF.*  
  
 WHEN '&08&'. *"Lage der Wohnung 1. OG rechts*  
 IF tiv3g-xstockk IS NOT INITIAL OR tiv3h-xklgesch IS NOT INITIAL.  
 CONCATENATE tiv3g-xstockk ` ` tiv3h-xklgesch INTO lv\_value.  
 ELSE.  
 CLEAR lv\_value.  
 ENDIF.  
 gt\_replaces-string\_new = lv\_value.  
 gt\_replaces-string\_old = '&08&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&08\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&09&'.  
*\* gt\_replaces-string\_new = .*  
*\* gt\_replaces-string\_old = '&09&'.*  
*\* APPEND gt\_replaces. CLEAR gt\_replaces.*  
  
 WHEN '&10&'.  
*\* gt\_replaces-string\_new = .*  
*\* gt\_replaces-string\_old = '&10&'.*  
*\* APPEND gt\_replaces. CLEAR gt\_replaces.*  
  
 WHEN '&11&' OR '&11\_1&'. *"Mieter / Eigentümer (Ermittlung über RE-FX GP)*  
 gt\_replaces-string\_new = lv\_partner\_status.  
 gt\_replaces-string\_old = '&11&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&11\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&13&'.*" OR '&13\_1&'. "Vertragsnummer 1*  
 gt\_replaces-string\_new = ls\_viqmel\_stoer-zz\_dt\_kc\_recnnr.  
 gt\_replaces-string\_old = '&13&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&13\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&12&'.*" OR '&12\_1&'. " Mieter: Name, Vorname 1*  
 CLEAR lv\_value.  
 CONCATENATE lv\_partner1\_name `, ` lv\_partner1\_vorname INTO lv\_value.  
 gt\_replaces-string\_new = lv\_value.  
 gt\_replaces-string\_old = '&12&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&12\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 WHEN '&14&'.*" OR '&14\_1&'. " Mieter: Name, Vorname 2*  
 CLEAR lv\_value.  
 CONCATENATE lv\_partner2\_name `, ` lv\_partner2\_vorname INTO lv\_value.  
 IF lv\_value+0(1) = ','.  
 CLEAR lv\_value.  
 ENDIF.  
 gt\_replaces-string\_new = lv\_value.  
 gt\_replaces-string\_old = '&14&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&14\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 WHEN '&15&'.*" OR '&15\_1&'. " Mieter: Festnetz 1 / HANDY*  
 gt\_replaces-string\_new = lv\_partner\_tel\_1.  
 gt\_replaces-string\_old = '&15&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&15\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
*\* WHEN '&58&'." OR '&58\_1&'. " Mieter: Festnetz 2*  
*\* gt\_replaces-string\_new = lv\_partner\_tel\_2.*  
*\* gt\_replaces-string\_old = '&58&'.*  
*\* APPEND gt\_replaces.*  
*\* gt\_replaces-string\_old = '&58\_1&'.*  
*\* APPEND gt\_replaces.*  
*\* CLEAR gt\_replaces.*  
 WHEN '&16&'.*" OR '&16\_1&'. " Mieter: Festnetz 2 / HANDY*  
 IF lv\_partner\_tel\_2 IS NOT INITIAL.  
 CONCATENATE `, ` lv\_partner\_tel\_2 INTO lv\_value.  
 ELSE.  
 CLEAR lv\_value.  
 ENDIF.  
 gt\_replaces-string\_new = lv\_value.  
 gt\_replaces-string\_old = '&16&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&16\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 WHEN '&17&'.*" OR '&17\_1&'. " Mieter: E-Mail*  
 IF lv\_partner\_smtp IS NOT INITIAL.  
 MOVE lv\_partner\_smtp TO lv\_value.  
 ELSE.  
 CLEAR lv\_value.  
 ENDIF.  
 gt\_replaces-string\_new = lv\_value.  
 gt\_replaces-string\_old = '&17&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&17\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&18&'.*" OR '&18\_1&'. "Bestelldatum*  
 CLEAR lv\_datum.  
 CALL FUNCTION 'FORMAT\_DATE\_4\_OUTPUT'  
 EXPORTING  
 datin = ls\_po\_header-doc\_date  
 format = 'DD.MM.YYYY'  
 IMPORTING  
 datex = lv\_datum.  
 gt\_replaces-string\_new = lv\_datum.  
 gt\_replaces-string\_old = '&18&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&18\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&19&'.*" OR '&19\_1&'. "Bestellnummer*  
 gt\_replaces-string\_new = ls\_po\_header-po\_number.  
 gt\_replaces-string\_old = '&19&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&19\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&20&'.*" OR '&20\_1&'. "Abnahmebegin (Datum aus ZG Meldung)*  
 CLEAR lv\_datum.  
 CALL FUNCTION 'FORMAT\_DATE\_4\_OUTPUT'  
 EXPORTING  
 datin = viqmel-zz\_afm\_gewbeginn  
 format = 'DD.MM.YYYY'  
 IMPORTING  
 datex = lv\_datum.  
 gt\_replaces-string\_new = lv\_datum.  
 gt\_replaces-string\_old = '&20&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&20\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&21&'.*" OR '&21\_1&'.*  
 CALL FUNCTION 'FORMAT\_DATE\_4\_OUTPUT'  
 EXPORTING  
 datin = lv\_frist  
 format = 'DD.MM.YYYY'  
 IMPORTING  
 datex = lv\_datum.  
 gt\_replaces-string\_new = lv\_datum.  
 gt\_replaces-string\_old = '&21&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&21\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces. CLEAR lv\_datum.  
  
 WHEN '&22&'.*" OR '&22\_1&'.*  
 CALL FUNCTION 'FORMAT\_DATE\_4\_OUTPUT'  
 EXPORTING  
 datin = sy-datum  
 format = 'DD.MM.YYYY'  
 IMPORTING  
 datex = lv\_datum.  
 gt\_replaces-string\_new = lv\_datum.  
 gt\_replaces-string\_old = '&22&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&22\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces. CLEAR lv\_datum.  
  
 WHEN '&23&'. *"Mitarbeiter zum ARBPL aus VIQMEL*  
 gt\_replaces-string\_new = adrp-name\_text.  
 gt\_replaces-string\_old = '&23&'.  
 APPEND gt\_replaces. CLEAR gt\_replaces.  
  
 WHEN '&24&'. *"Mangelbeschreibung aus Langtext ZM(Z4)*  
 PERFORM ltext TABLES lt\_lines  
 CHANGING ls\_thead.  
 CLEAR mystring.  
 LOOP AT lt\_lines INTO ls\_lines.  
 FIND 'CET' IN ls\_lines.  
 IF sy-subrc = 0.  
 CONTINUE.  
 ENDIF.  
 SHIFT ls\_lines-tdline LEFT DELETING LEADING '\*'.  
 CONCATENATE mystring ls\_lines-tdline INTO mystring.  
 SHIFT mystring LEFT DELETING LEADING space .  
 ENDLOOP.  
 gt\_replaces-string\_new = mystring.  
 gt\_replaces-string\_old = '&24&'.  
 APPEND gt\_replaces. CLEAR gt\_replaces.  
  
 WHEN '&25&'. *"Mitarbeiter(Ansprechpartner) zum ARBPL aus VIQMEL*  
 gt\_replaces-string\_new = adrp-name\_text.  
 gt\_replaces-string\_old = '&25&'.  
 APPEND gt\_replaces. CLEAR gt\_replaces.  
  
 WHEN '&26&'. *"Telefon+Durchwahl zum ARBPL aus VIQMEL*  
 CONCATENATE adrc-tel\_number ` ` '-' ` ` adcp-tel\_extens INTO gt\_replaces-string\_new.  
 gt\_replaces-string\_old = '&26&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&26\_1&'.  
 APPEND gt\_replaces.  
 gt\_replaces-string\_old = '&26\_2&'.  
 APPEND gt\_replaces. CLEAR gt\_replaces.  
  
 WHEN '&27&'. *"E-Mail zum Mitarbeiter aus VIQMEL*  
 gt\_replaces-string\_new = adr6-smtp\_addr.  
 gt\_replaces-string\_old = '&27&'.  
 APPEND gt\_replaces. CLEAR gt\_replaces.  
*\* Partner 1*  
 WHEN '&28&'. *"Anrede*  
 CLEAR tsad3t-title\_medi.  
 IF lv\_partner1\_anrede IS NOT INITIAL.  
 IF lv\_partner1\_anrede = '0002'.  
 tsad3t-title\_medi = 'Herr'.  
 ELSE.  
 SELECT SINGLE title\_medi FROM tsad3t INTO tsad3t-title\_medi  
 WHERE title = lv\_partner1\_anrede.  
 ENDIF.  
 ENDIF.  
 gt\_replaces-string\_new = tsad3t-title\_medi.  
 gt\_replaces-string\_old = '&28&'.  
 APPEND gt\_replaces. CLEAR gt\_replaces.  
 WHEN '&29&'.*" OR '&29\_1&'. "Vorname*  
 gt\_replaces-string\_new = lv\_partner1\_vorname.  
 gt\_replaces-string\_old = '&29&'. *" Kopf*  
 APPEND gt\_replaces.  
*\* gt\_replaces-string\_old = '&29\_1&'.*  
*\* APPEND gt\_replaces.*  
 CLEAR gt\_replaces.  
 WHEN '&30&'.*" OR '&30\_1&'. "Nachname*  
 gt\_replaces-string\_new = lv\_partner1\_name.  
 gt\_replaces-string\_old = '&30&'. *" Kopf*  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 CONCATENATE lv\_partner1\_name ',' INTO gt\_replaces-string\_new.  
 gt\_replaces-string\_old = '&30\_1&'. *" Anrede*  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
*\* Partner 2*  
 WHEN '&34&'. *"Anrede*  
 CLEAR tsad3t-title\_medi.  
 IF lv\_partner2\_anrede IS NOT INITIAL.  
 IF lv\_partner2\_anrede IS NOT INITIAL.  
 IF lv\_partner2\_anrede = '0002'.  
 tsad3t-title\_medi = 'Herr'.  
 ELSE.  
 SELECT SINGLE title\_medi FROM tsad3t INTO tsad3t-title\_medi  
 WHERE title = lv\_partner2\_anrede.  
 ENDIF.  
 ENDIF.  
 CONCATENATE 'guten Tag' ` ` tsad3t-title\_medi INTO gt\_replaces-string\_new.  
 ELSE.  
 gt\_replaces-string\_new = tsad3t-title\_medi.  
 ENDIF.  
 gt\_replaces-string\_old = '&34&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 WHEN '&31&'.*" OR '&31\_1&'. "Vorname*  
 gt\_replaces-string\_new = lv\_partner2\_vorname.  
 gt\_replaces-string\_old = '&31&'. *" Kopf*  
 APPEND gt\_replaces.  
*\* gt\_replaces-string\_old = '&31\_1&'.*  
*\* APPEND gt\_replaces.*  
 CLEAR gt\_replaces.  
 WHEN '&32&'.*" OR '&32\_1&'. "Nachname*  
 gt\_replaces-string\_new = lv\_partner2\_name.  
 gt\_replaces-string\_old = '&32&'. *" Kopf*  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 CONCATENATE lv\_partner2\_name ',' INTO gt\_replaces-string\_new.  
 gt\_replaces-string\_old = '&32\_1&'. *" Anrede*  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&33&'. *"Datum Störmeldung Z4*  
 CLEAR lv\_datum.  
 CALL FUNCTION 'FORMAT\_DATE\_4\_OUTPUT'  
 EXPORTING  
 datin = viqmel-qmdat  
 format = 'DD.MM.YYYY'  
 IMPORTING  
 datex = lv\_datum.  
 gt\_replaces-string\_new = lv\_datum.  
 gt\_replaces-string\_old = '&33&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
  
 WHEN '&40&'.*" OR '&40\_1&' OR '&40\_2&'.*  
 CLEAR lv\_datum.  
 CALL FUNCTION 'FORMAT\_DATE\_4\_OUTPUT'  
 EXPORTING  
 datin = lv\_datum\_frist1  
 format = 'DD.MM.YYYY'  
 IMPORTING  
 datex = lv\_datum.  
 gt\_replaces-string\_new = lv\_datum.  
 gt\_replaces-string\_old = '&40&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 gt\_replaces-string\_new = lv\_datum.  
 gt\_replaces-string\_old = '&40\_1&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 gt\_replaces-string\_new = lv\_datum.  
 gt\_replaces-string\_old = '&40\_2&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 CLEAR lv\_datum.  
  
 WHEN '&41&'. *" Korrerspondenz 2.Frist !*  
 CLEAR lv\_datum.  
 CALL FUNCTION 'FORMAT\_DATE\_4\_OUTPUT'  
 EXPORTING  
 datin = viqmel-zz\_afm\_1frerstelldat  
 format = 'DD.MM.YYYY'  
 IMPORTING  
 datex = lv\_datum.  
 gt\_replaces-string\_new = lv\_datum.  
 gt\_replaces-string\_old = '&41&'.  
 APPEND gt\_replaces.  
 CLEAR gt\_replaces.  
 CLEAR lv\_datum.  
  
 ENDCASE.  
 ENDLOOP.  
 ELSE.  
 ENDIF.  
 ELSE.  
 ENDIF. *" gv\_qmart = 'SM' OR gv\_qmart = 'SG'*  
  
 ENDCASE. *" sy-langu*  
  
*\*----------------------------------------------------------------------\**  
  
 PERFORM replace\_t TABLES gt\_replaces  
 CHANGING lo\_error.  
 APPEND lo\_error TO lt\_errors.  
  
*\* Bereinigen des Langtextes:*  
 LOOP AT lt\_lines INTO ls\_lines.  
 SHIFT ls\_lines-tdline LEFT DELETING LEADING '\*'.  
 MODIFY lt\_lines FROM ls\_lines.  
 ENDLOOP.  
  
 IF lt\_lines IS NOT INITIAL.  
 PERFORM insert\_ltext TABLES lt\_lines  
 CHANGING lo\_error.  
 APPEND lo\_error TO lt\_errors.  
 ENDIF.  
  
 ENDIF. *"sy-ucomm = 'VORLAGEN'.*  
 ENDIF.  
  
*\*----------------------------------------------------------------------\**  
  
*\*\*\* Fehler anzeigen*  
*\* """ OBjekt besitzt nicht die angeforderte COM-Schnittstelle Meldung nach go\_document->get\_spreadsheet\_interface*  
*\* LOOP AT lt\_errors INTO lo\_error.*  
*\* CALL METHOD lo\_error->raise\_message*  
*\* EXPORTING*  
*\* type = 'W'*  
*\* EXCEPTIONS*  
*\* message\_raised = 1*  
*\* flush\_failed = 2*  
*\* OTHERS = 3.*  
*\**  
*\* IF sy-subrc <> 0.*  
*\* MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno*  
*\* WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.*  
*\* ENDIF.*  
*\* ENDLOOP.*  
  
ENDFORM. *" LOAD\_DOCUMENT*

*Include🡪 ZXQQMF20*

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF20.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form REPLACE\_T*  
*\*&---------------------------------------------------------------------\**  
*\* text*  
*\*----------------------------------------------------------------------\**  
*\* -->P\_GT\_REPLACES text*  
*\* <--P\_LO\_ERROR text*  
*\*----------------------------------------------------------------------\**  
FORM replace\_t TABLES p\_gt\_replaces STRUCTURE gs\_repl  
 CHANGING p\_lo\_error.  
  
 IF NOT go\_worddoc IS INITIAL.  
 LOOP AT p\_gt\_replaces.  
 CALL METHOD go\_worddoc->replace  
 EXPORTING  
 flag = ' '  
*\* NO\_FLUSH = ' '*  
 pos = 'MAINDOC'  
 replace\_string = p\_gt\_replaces-string\_new  
 search\_string = p\_gt\_replaces-string\_old  
 IMPORTING  
 error = p\_lo\_error.  
 ENDLOOP.  
 ENDIF.  
  
ENDFORM.

Include🡪 *ZXQQMF18*

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF18.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form INSERT\_LTEXT*  
*\*&---------------------------------------------------------------------\**  
*\* text*  
*\*----------------------------------------------------------------------\**  
*\* -->P\_LT\_LINES text*  
*\* <--P\_LO\_ERROR text*  
*\*----------------------------------------------------------------------\**  
FORM insert\_ltext TABLES p\_lt\_lines STRUCTURE tline  
 CHANGING p\_lo\_error.  
  
 DATA lt\_info\_tab TYPE soi\_cols\_table.  
 DATA ls\_info\_tab TYPE soi\_cols.  
 DATA lv\_doc\_nr TYPE i.  
  
 MOVE '2' TO ls\_info\_tab-colindex.  
 APPEND ls\_info\_tab TO lt\_info\_tab.  
  
 CASE sy-langu.  
 WHEN 'D' OR 'Q'. *"deutsch*  
 lv\_doc\_nr = 3.  
*\* WHEN 'C' OR 'L' OR 'Q'. "polnisch, tschechisch o. slowakisch: Empfängerbereich ist ebenfalls als Tabelle eingerichtet*  
*\* lv\_doc\_nr = 2.*  
*\* WHEN OTHERS.*  
*\* lv\_doc\_nr = 1. "rumänisch*  
 ENDCASE.  
  
 IF NOT go\_worddoc IS INITIAL.  
  
 CALL METHOD go\_worddoc->insert\_table  
 EXPORTING  
 data\_table = p\_lt\_lines[]  
 info\_table = lt\_info\_tab  
 lowerbound = 0  
 upperbound = 0  
 doctable\_number = lv\_doc\_nr  
 clearoption = 1  
 startrow = 1  
 varsize = 'X'  
*\* no\_flush = ' '*  
*\* wholetable = ' '*  
 IMPORTING  
 error = p\_lo\_error.  
*\* retcode =*  
  
 ENDIF.  
  
ENDFORM.

Include🡪 ZXQQMF25

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF25.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form SAVE\_AS\_NEW\_VERSION*  
*\*&---------------------------------------------------------------------\**  
FORM save\_as\_new\_version USING ind.  
  
 DATA lt\_uris TYPE STANDARD TABLE OF bapiuri.  
 DATA ls\_uri TYPE bapiuri.  
 DATA lv\_object\_key TYPE bapibds01-objkey.  
 DATA lt\_data TYPE STANDARD TABLE OF bapiconten.  
 DATA lv\_doc\_size TYPE i.  
 DATA lt\_signature TYPE STANDARD TABLE OF bapisignat.  
 DATA ls\_signature TYPE bapisignat.  
 DATA lt\_components TYPE STANDARD TABLE OF bapicompon.  
 DATA ls\_component TYPE bapicompon.  
 DATA ls\_properties TYPE bapiproper.  
*\* DATA ls\_properties TYPE sbdst\_properties.*  
 DATA lt\_properties TYPE sbdst\_properties. *"STANDARD TABLE OF*  
 DATA lv\_new\_vers TYPE sbdst\_doc\_ver\_no.  
 DATA lv\_ktext TYPE qmtxt.  
 DATA lv\_classname TYPE bds\_clsnam.  
 DATA lv\_classtype TYPE bds\_clstyp.  
 DATA gt\_info\_sign TYPE STANDARD TABLE OF bapisignat.  
 DATA gs\_info\_sign TYPE bapisignat.  
 DATA lv\_infotext TYPE string.  
 DATA lv\_txt\_frist(7) TYPE c.  
 DATA lv\_txt\_typ(6) TYPE c.  
  
  
*\*&---------------------------------------------------------------------\**  
  
*\* => aktuelle "eigene Version" sowie Rü+ge-Nr. holen*  
 gt\_info\_sign = gt\_signatures.  
  
 LOOP AT gt\_info\_sign INTO gs\_info\_sign WHERE ( prop\_name EQ 'BDS\_STATE' OR prop\_name EQ 'BDS\_KEYWORD' ).  
 IF gs\_info\_sign-prop\_name EQ 'BDS\_STATE'.  
 MOVE gs\_info\_sign-prop\_value TO gv\_version.  
 ELSEIF gs\_info\_sign-prop\_name EQ 'BDS\_KEYWORD'.  
 IF gs\_info\_sign-prop\_value CP 'document\*'.  
 MOVE gs\_info\_sign-prop\_value+8(1) TO gv\_docnr.  
 ENDIF.  
 ENDIF.  
 ENDLOOP.  
  
*\*Wenn ist Mängelmeldung, eine Maßnahme ausgewählt ist und keine Versionen vorhanden*  
*\*die Versionnummer sollte von 1 beginnen*  
 IF ( gv\_qmart = 'SM' OR gv\_qmart = 'SG' ) AND fl\_txtma = 1 AND gt\_info\_sign IS INITIAL.  
 gv\_version = 0.  
*\* gv\_docnr = gv\_sel\_vorl.*  
 READ TABLE gt\_sign\_vorl INTO gs\_sign\_vorl WITH KEY doc\_count = gv\_sel\_vorl prop\_name = 'BDS\_KEYWORD'.  
 IF sy-subrc = 0.  
 MOVE gs\_sign\_vorl-prop\_value+8(1) TO gv\_docnr.  
 ENDIF.  
 ENDIF.  
  
*\*Docnr hochzählen*  
 gv\_docnr = gv\_docnr + 1.  
  
*\* Eigene Versionsnummer hochzählen*  
 gv\_version = gv\_version + 1.  
 SHIFT gv\_version LEFT DELETING LEADING space.  
  
 IF NOT go\_document IS INITIAL.  
 CALL METHOD go\_document->save\_document\_to\_table  
 CHANGING  
 document\_size = lv\_doc\_size  
 document\_table = lt\_data.  
 ELSE.  
 MESSAGE i000(38) WITH TEXT-013. *" Es ist kein Dokument geöffnet.*  
 EXIT.  
 ENDIF.  
  
 MOVE viqmel-qmnum TO gv\_objektschl.  
  
 ls\_signature-prop\_name = 'BDS\_DOCUMENTTYPE'.  
 ls\_signature-prop\_value = 'BDS\_TEXT'.  
 ls\_signature-doc\_count = '00000001'.  
 ls\_signature-doc\_ver\_no = '00000001'.  
 ls\_signature-doc\_var\_id = '00000001'.  
 ls\_signature-comp\_count = '00000001'.  
 APPEND ls\_signature TO lt\_signature.  
  
 ls\_signature-prop\_name = 'BDS\_DOCUMENTCLASS'.  
 ls\_signature-prop\_value = 'DOC'.  
 APPEND ls\_signature TO lt\_signature.  
  
*\*&---------------------------------------------------------------------\**  
*\* Vorlagedaten lesen*  
 SELECT SINGLE cls\_nam clstyp typeid FROM /afm/oaer INTO CORRESPONDING FIELDS OF gs\_oaer  
 WHERE mncod = gv\_mncod AND cls\_nam = GS\_VORL-TYPE AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 lv\_classname = gs\_oaer-cls\_nam.  
 lv\_classtype = gs\_oaer-clstyp.  
 ENDIF.  
  
 CLEAR lv\_txt\_frist.  
 IF lv\_classname = 'ZKORR\_MAENGEL\_1' OR lv\_classname = 'ZKORR\_MAENGEL\_INFO\_1' OR lv\_classname = 'ZKORR\_MAENGEL\_3'.  
 lv\_txt\_frist = '1.Frist'.  
 ENDIF.  
  
 IF lv\_classname = 'ZKORR\_MAENGEL\_2' OR lv\_classname = 'ZKORR\_MAENGEL\_INFO\_2' or lv\_classname = 'ZKORR\_MAENGEL\_4'.  
 lv\_txt\_frist = '2.Frist'.  
 ENDIF.  
  
clear lv\_txt\_typ.  
 IF lv\_classname = 'ZKORR\_MAENGEL\_INFO\_1' OR lv\_classname = 'ZKORR\_MAENGEL\_INFO\_2'.  
 lv\_txt\_typ = 'Info'.  
 ENDIF.  
  
 IF lv\_classname = 'ZKORR\_MAENGEL\_1' OR lv\_classname = 'ZKORR\_MAENGEL\_3' OR lv\_classname = 'ZKORR\_MAENGEL\_2'  
 OR lv\_classname = 'ZKORR\_MAENGEL\_4'.  
 lv\_txt\_typ = 'Mangel'.  
 ENDIF.  
  
 ls\_signature-prop\_name = 'DESCRIPTION'.  
 CONCATENATE  
 sy-datum+0(4) '-' sy-datum+4(2) '-' sy-datum+6(2)` `  
 lv\_txt\_frist ` `  
 lv\_txt\_typ ` `  
 viqmel-qmtxt INTO ls\_signature-prop\_value.  
 lv\_infotext = ls\_signature-prop\_value.  
 APPEND ls\_signature TO lt\_signature.  
  
*\*&---------------------------------------------------------------------\**  
  
 ls\_signature-prop\_name = 'LANGUAGE'.  
 ls\_signature-prop\_value = sy-langu.  
 APPEND ls\_signature TO lt\_signature.  
  
 ls\_signature-prop\_name = 'BDS\_STATE'.  
 ls\_signature-prop\_value = gv\_version.  
 APPEND ls\_signature TO lt\_signature.  
  
 ls\_signature-prop\_name = 'BDS\_KEYWORD'.  
 CONCATENATE 'document' gv\_docnr INTO ls\_signature-prop\_value.  
 APPEND ls\_signature TO lt\_signature.  
  
 CONCATENATE viqmel-qmnum '\_' gv\_docnr '\_V' gv\_version '.doc' INTO ls\_component-comp\_id.  
 ls\_component-mimetype = 'application/vnd.ms-word'.  
 ls\_component-comp\_size = lv\_doc\_size.  
 ls\_component-doc\_count = '00000001'.  
 ls\_component-comp\_count = '00000001'.  
 APPEND ls\_component TO lt\_components.  
  
 MOVE viqmel-qmnum TO gv\_newnam.  
  
*\*&---------------------------------------------------------------------\**  
  
 CASE gv\_qmart.  
  
 WHEN 'SM'.  
 IF fl\_txtma = 1.  
*\* BDN Adresse Details für die Vorlage ermitteln*  
 REFRESH gt\_oaer.  
 SELECT \* FROM /afm/oaer INTO TABLE gt\_oaer  
 WHERE mncod = gv\_mncod  
 AND bsart = ''  
 AND vorlage = gs\_vorl-type  
 AND sprache = sy-langu.  
  
 IF sy-subrc = 0.  
 READ TABLE gt\_oaer INTO gs\_oaer INDEX 1.  
 lv\_classname = gs\_oaer-cls\_nam.  
 lv\_classtype = gs\_oaer-clstyp.  
 ENDIF.  
  
 CLEAR gv\_newnam.  
 CONCATENATE viqmel-qmnum gv\_mncod INTO gv\_newnam.  
  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_CREA\_TAB'  
 EXPORTING  
*\* LOGICAL\_SYSTEM =*  
*\* classname = c\_classmass*  
*\* classtype = c\_cltypmass*  
 classname = lv\_classname  
 classtype = lv\_classtype  
 client = sy-mandt  
 object\_key = gv\_newnam  
*\* BINARY\_FLAG = 'X'*  
*\* IMPORTING*  
*\* OBJECT\_KEY =*  
 TABLES  
 signature = lt\_signature  
 components = lt\_components  
 content = lt\_data  
*\* ASCII\_CONTENT =*  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 ELSE.  
 ENDIF.  
  
 WHEN 'SG'.  
 IF fl\_txtma = 1.  
*\* BDN Adresse Details für die Vorlage ermitteln*  
 SELECT SINGLE cls\_nam clstyp typeid FROM /afm/oaer INTO CORRESPONDING FIELDS OF gs\_oaer  
 WHERE mncod = gv\_mncod  
 AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 lv\_classname = gs\_oaer-cls\_nam.  
 lv\_classtype = gs\_oaer-clstyp.  
 ENDIF.  
  
 CLEAR gv\_newnam.  
 CONCATENATE viqmel-qmnum gv\_mncod INTO gv\_newnam.  
  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_CREA\_TAB'  
 EXPORTING  
*\* LOGICAL\_SYSTEM =*  
*\* classname = c\_classmass*  
*\* classtype = c\_cltypmass*  
 classname = lv\_classname  
 classtype = lv\_classtype  
 client = sy-mandt  
 object\_key = gv\_newnam  
*\* BINARY\_FLAG = 'X'*  
*\* IMPORTING*  
*\* OBJECT\_KEY =*  
 TABLES  
 signature = lt\_signature  
 components = lt\_components  
 content = lt\_data  
*\* ASCII\_CONTENT =*  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 ELSE.  
*\* BDN Adresse Details für die Vorlage ermitteln*  
 SELECT SINGLE cls\_nam clstyp typeid FROM /afm/oaer INTO CORRESPONDING FIELDS OF gs\_oaer  
 WHERE mncod = 'ALLG'  
 AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 lv\_classname = gs\_oaer-cls\_nam.  
 lv\_classtype = gs\_oaer-clstyp.  
 ENDIF.  
  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_CREA\_TAB'  
 EXPORTING  
*\* LOGICAL\_SYSTEM =*  
*\* classname = c\_class500*  
*\* classtype = c\_cltyp500*  
 classname = lv\_classname  
 classtype = lv\_classtype  
 client = sy-mandt  
 object\_key = gv\_newnam  
*\* BINARY\_FLAG = 'X'*  
*\* IMPORTING*  
*\* OBJECT\_KEY =*  
 TABLES  
 signature = lt\_signature  
 components = lt\_components  
 content = lt\_data  
*\* ASCII\_CONTENT =*  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 ENDIF.  
 ENDCASE.  
  
*\*&---------------------------------------------------------------------\**  
  
 IF sy-subrc <> 0.  
 MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
 WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
 ELSE.  
 gv\_objektschl = gv\_newnam.  
 gt\_signatures = lt\_signature.  
 gt\_components\_upd = lt\_components.  
*\* Neue Version erhält auch eine neue doc\_id, sodass diese hier neu in die gs\_signature gesetzt werden muss für den Fall,*  
*\* dass direkt weitere Änderungen im Dokument gemacht werden*  
 READ TABLE gt\_signatures INDEX 1 INTO gs\_signature.  
*\* Info-Meldung über die erfolgreiche Anlage einer neuen Version - per Baustein*  
  
 DATA lv\_version(4) TYPE c.  
  
 CONCATENATE 'V' gv\_version INTO lv\_version.  
  
 CALL FUNCTION 'POPUP\_TO\_INFORM'  
 EXPORTING  
 titel = 'Neue Version des Dokumentes angelegt:'  
 txt1 = lv\_infotext  
 txt2 = 'wurde erfolgreich gespeichert!'.  
 ENDIF.  
  
ENDFORM.

Include🡪 ZXQQMF21

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF21.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form CLEANUP*  
*\*&---------------------------------------------------------------------\**  
FORM cleanup .  
  
 IF NOT go\_document IS INITIAL.  
 CALL METHOD go\_document->close\_document.  
 ENDIF.  
  
 CLEAR gv\_doc\_open.  
 CLEAR gv\_url.  
 CLEAR gv\_vorl.  
 CLEAR gv\_selected.  
  
 IF NOT go\_control IS INITIAL.  
 CALL METHOD go\_control->destroy\_control  
 EXPORTING  
 no\_flush = ' '.  
 ENDIF.  
  
 FREE go\_worddoc.  
 CLEAR go\_worddoc.  
  
 IF NOT go\_document IS INITIAL.  
 CALL METHOD go\_document->release\_document.  
 ENDIF.  
  
 FREE: go\_document,  
 go\_control.  
  
 CLEAR: go\_document,  
 go\_control.  
  
ENDFORM.

Include🡪 ZXQQMF22

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF22.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form DOCUMENT\_CHANGED*  
*\*&---------------------------------------------------------------------\**  
FORM document\_changed .  
 TYPES: tyt\_errors TYPE STANDARD TABLE OF REF TO i\_oi\_error  
 WITH NON-UNIQUE DEFAULT KEY.  
  
 DATA: lo\_error TYPE REF TO i\_oi\_error,  
 lt\_errors TYPE tyt\_errors.  
 DATA: lv\_answer TYPE c.  
  
*\*&---------------------------------------------------------------------\**  
  
 CLEAR: gv\_changed, lv\_answer.  
  
 IF NOT go\_document IS INITIAL.  
 CALL METHOD go\_document->has\_changed  
*\* EXPORTING*  
*\* no\_flush = ' '*  
 IMPORTING  
 error = lo\_error  
*\* retcode =*  
 ret\_value = gv\_changed.  
 IF NOT ( lo\_error->has\_succeeded = 'X' AND  
 lo\_error->has\_failed = ' ' ).  
 MESSAGE i000(38) WITH TEXT-014 lo\_error->error\_code TEXT-015. *" beim Abfragen des Dokumentstatus*  
 ENDIF.  
 ENDIF.  
  
 IF NOT gv\_changed IS INITIAL.  
 IF gv\_vorl NE 'X'. *" keine Vorlage geöffnete!*  
 CALL FUNCTION 'POPUP\_TO\_DECIDE\_INFO'  
 EXPORTING  
*\* DEFAULTOPTION = 'Y'*  
 titel = 'Das Dokument wurde geändert!'  
 textline1 = 'wollen Sie das Dokument sichern?'  
 textline2 = '- ansonsten Abbrechen mit ''X'''  
 start\_column = 25  
 start\_row = 6  
 IMPORTING  
 answer = lv\_answer.  
 ELSE.  
 lv\_answer = 'J'.  
 ENDIF.  
*\* IF lv\_answer EQ 'J'.*  
*\* PERFORM save\_document\_as.*  
*\* gv\_saved = 'X'.*  
*\* ENDIF.*  
 ENDIF.  
  
ENDFORM.

Include🡪 ZXQQMF27

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF27.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form SHOW\_NAVIGATOR*  
*\*&---------------------------------------------------------------------\**  
FORM show\_navigator USING ind.  
  
 DATA lv\_newnam TYPE bds\_typeid.  
 DATA lv\_classname TYPE bds\_clsnam.  
 DATA lv\_classtype TYPE bds\_clstyp.  
  
*\*&---------------------------------------------------------------------\**  
  
 IF gv\_qmart = 'SG' OR gv\_qmart = 'SM'.  
  
 CLEAR bdcdata[].  
  
 REFRESH gt\_oaer.  
  
 IF ls\_vorl-comp\_id CS viqmel-qmnum. *"Wenn ja bereits generiertes Formular, keine Vorlage!*  
  
 CLEAR: lv\_classname, lv\_classtype, lv\_newnam.  
  
*\* Auslesen der Korrespondenzdaten,prüfen welche Frist / Maßnahme*  
 FIND '1.Frist' IN ls\_vorl-prop\_value.  
 IF sy-subrc = 0.  
 gv\_mncod = 'F1'.  
 ELSE.  
 gv\_mncod = 'F2'.  
 ENDIF.  
  
*\* Daten ermitteln mit Maßnahmencode*  
 SELECT SINGLE \* FROM /afm/oaer INTO gs\_oaer  
 WHERE mncod = gv\_mncod AND bsart = '' AND cls\_nam = ls\_vorl-type  
 AND sprache = sy-langu.  
  
 lv\_classname = gs\_oaer-cls\_nam. *" BDN Adresse Details für markierte Dokumente ermitteln*  
 lv\_classtype = gs\_oaer-clstyp.  
 CONCATENATE viqmel-qmnum gv\_mncod INTO lv\_newnam. *"Dateinamen aus Meldungsnummer und Maßnahmencode bilden*  
  
 ELSE. *" Vorlage*  
  
 IF gv\_mncod IS NOT INITIAL.  
*\* Entsprechende Daten zur selektierten Korrespondez lesen*  
 SELECT SINGLE \* FROM /afm/oaer INTO gs\_oaer WHERE mncod = gv\_mncod AND bsart = '' AND cls\_nam = ls\_vorl-type AND sprache = sy-langu.  
 ELSE.  
*\* Entsprechende Daten zur selektierten Korrespondez lesen*  
 SELECT SINGLE \* FROM /afm/oaer INTO gs\_oaer WHERE bsart = '' AND cls\_nam = ls\_vorl-type AND sprache = sy-langu.  
 ENDIF.  
 IF sy-subrc = 0.  
 lv\_classname = gs\_oaer-cls\_nam. *" BDN Adresse Details für die Vorlage ermitteln*  
 lv\_classtype = gs\_oaer-clstyp.  
 lv\_newnam = gs\_oaer-vorlage.  
 ENDIF.  
  
 ENDIF.  
  
 PERFORM bdc\_dynpro USING 'BDSFIND\_1' '1001'.  
 PERFORM bdc\_field USING 'BDC\_OKCODE'  
 '=ONLI'.  
 PERFORM bdc\_field USING 'BDC\_CURSOR'  
 'OBJKEY'.  
 PERFORM bdc\_field USING 'CLASSN'  
 lv\_classname.  
 PERFORM bdc\_field USING 'CLASSTYP'  
 lv\_classtype.  
 PERFORM bdc\_field USING 'OBJKEY'  
 lv\_newnam.  
  
*\* BDN mit Parametern starten*  
 CALL TRANSACTION 'OAER' USING bdcdata  
 MODE 'E'  
 UPDATE 'S'.  
  
 ENDIF.  
  
ENDFORM.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF28.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form BDC\_DYNPRO*  
*\*&---------------------------------------------------------------------\**  
*\* -->P\_0053 text*  
*\* -->P\_0054 text*  
*\*----------------------------------------------------------------------\**  
FORM bdc\_dynpro USING program dynpro.  
  
 CLEAR bdcdata.  
 bdcdata-program = program.  
 bdcdata-dynpro = dynpro.  
 bdcdata-dynbegin = 'X'.  
 APPEND bdcdata.  
  
ENDFORM.

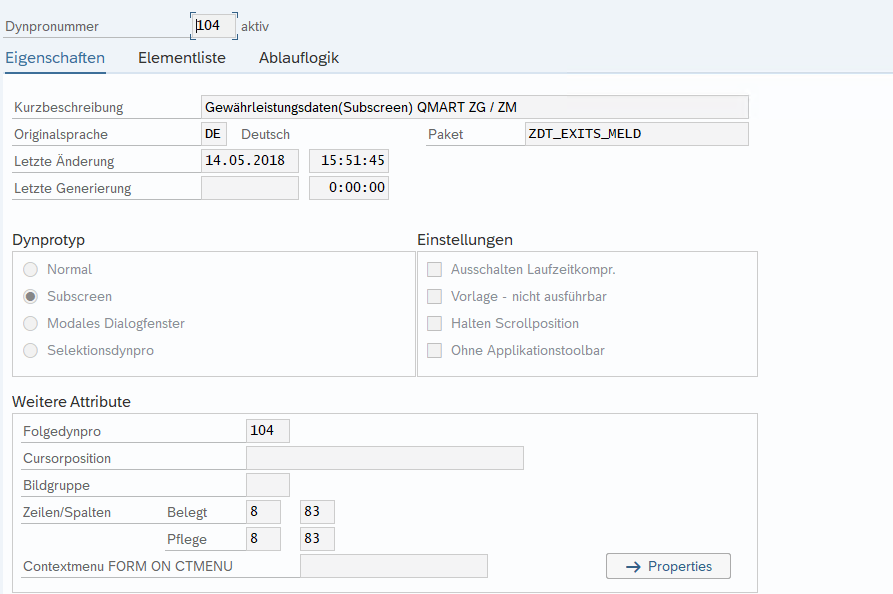
*\*&---------------------------------------------------------------------\**  
*\*& Form BDC\_FIELD*  
*\*&---------------------------------------------------------------------\**  
*\* -->P\_0058 text*  
*\* -->P\_0059 text*  
*\*----------------------------------------------------------------------\**  
FORM bdc\_field USING fnam fval.  
  
 CLEAR bdcdata.  
 bdcdata-fnam = fnam.  
 bdcdata-fval = fval.  
 APPEND bdcdata.  
  
ENDFORM.

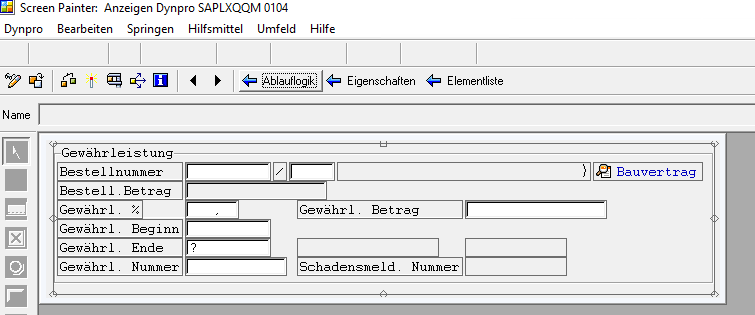
*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF23.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form SACHVERHALT\_AKTUAL*  
*\*&---------------------------------------------------------------------\**  
FORM sachverhalt\_aktual .  
  
 TYPES: tyt\_errors TYPE STANDARD TABLE OF REF TO i\_oi\_error  
 WITH NON-UNIQUE DEFAULT KEY.  
 DATA ls\_thead TYPE thead.  
 DATA lt\_lines TYPE TABLE OF tline.  
 DATA ls\_line TYPE tline.  
 DATA lt\_result\_tab TYPE match\_result\_tab.  
 DATA lo\_error TYPE REF TO i\_oi\_error.  
 DATA lt\_errors TYPE tyt\_errors.  
  
*\*&---------------------------------------------------------------------\**  
  
*\* Wird etwas im Langtext (Sachverhalt) verändert und in den Reiter Formulardruck gesprungen,*  
*\* wird mit READ\_TEXT der aktuelle text eingelesen. Es muss nicht extra vorher gesichert werden.*  
  
 CLEAR gt\_param.  
 REFRESH gt\_param.  
  
 IF gv\_qmart = 'SM' OR gv\_qmart = 'SG'.  
 CLEAR gt\_mgpar.  
 REFRESH gt\_mgpar.  
  
*\* Einlesen der Ersetzungsparameter und Füllen der Replace-Tabelle*  
 SELECT \* FROM /afm/formpar INTO TABLE gt\_mgpar  
 WHERE sprache EQ sy-langu.  
  
 LOOP AT gt\_mgpar INTO gs\_mgpar.  
 CASE gs\_mgpar-param.  
 WHEN '&13&'. *"Mängelliste: Langtext Sachverhalt aus der Meldung*  
 PERFORM ltext TABLES lt\_lines  
 CHANGING ls\_thead.  
 ENDCASE.  
 ENDLOOP.  
  
 ELSE.  
*\* Einlesen der Ersetzungsparameter und Füllen der Replace-Tabelle*  
 SELECT \* FROM /afm/formpar INTO TABLE gt\_param  
 WHERE sprache EQ sy-langu.  
  
 LOOP AT gt\_param INTO gs\_param.  
 CASE gs\_param-param.  
 WHEN '&13&'. *"Mängelliste: Langtext Sachverhalt aus der Meldung*  
 PERFORM ltext TABLES lt\_lines  
 CHANGING ls\_thead.  
 ENDCASE.  
 ENDLOOP.  
  
 ENDIF.  
  
*\* Bereinigen des Langtextes:*  
 LOOP AT lt\_lines INTO ls\_line.  
 SHIFT ls\_line-tdline LEFT DELETING LEADING '\*'.  
 MODIFY lt\_lines FROM ls\_line.  
 ENDLOOP.  
  
 PERFORM actualize\_ltext TABLES lt\_lines  
 CHANGING lo\_error.  
 APPEND lo\_error TO lt\_errors.  
  
*\*----------------------------------------------------------------------\**  
  
*\*\*\* Fehler anzeigen*  
*\* """ OBjekt besitzt nicht die angeforderte COM-Schnittstelle Meldung nach go\_document->get\_spreadsheet\_interface*  
*\* LOOP AT lt\_errors INTO lo\_error.*  
*\* CALL METHOD lo\_error->raise\_message*  
*\* EXPORTING*  
*\* type = 'W'*  
*\* EXCEPTIONS*  
*\* message\_raised = 1*  
*\* flush\_failed = 2*  
*\* OTHERS = 3.*  
*\**  
*\* IF sy-subrc <> 0.*  
*\* MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno*  
*\* WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.*  
*\* ENDIF.*  
*\* ENDLOOP.*  
  
ENDFORM.

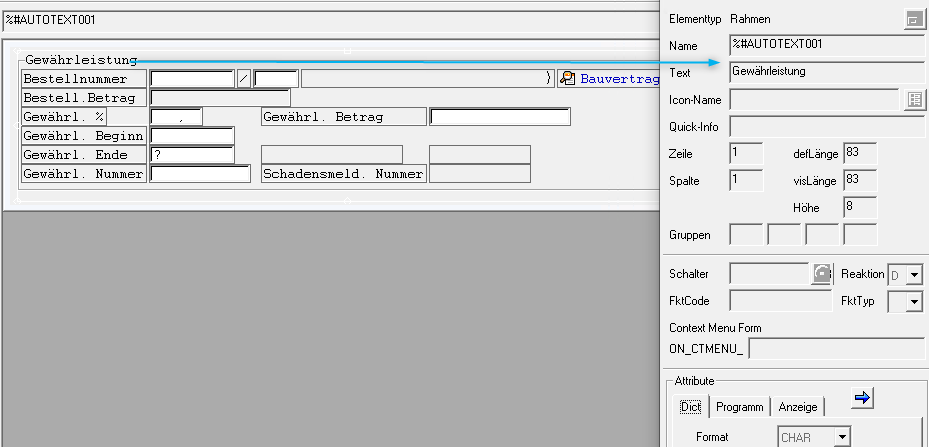
*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF19.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form LTEXT*  
*\*&---------------------------------------------------------------------\**  
*\* -->P\_LT\_LINES text*  
*\* <--P\_LS\_THEAD text*  
*\*----------------------------------------------------------------------\**  
FORM ltext TABLES p\_lt\_lines STRUCTURE tline  
 CHANGING p\_ls\_thead STRUCTURE thead.  
  
 p\_ls\_thead-tdid = 'LTXT'.  
 p\_ls\_thead-tdspras = sy-langu.  
 p\_ls\_thead-tdname = viqmel-qmnum.  
 p\_ls\_thead-tdobject = 'QMEL'.  
  
 CALL FUNCTION 'READ\_TEXT'  
 EXPORTING  
*\* CLIENT = SY-MANDT*  
 id = p\_ls\_thead-tdid  
 language = p\_ls\_thead-tdspras  
 name = p\_ls\_thead-tdname  
 object = p\_ls\_thead-tdobject  
*\* ARCHIVE\_HANDLE = 0*  
*\* LOCAL\_CAT = ' '*  
 IMPORTING  
 header = p\_ls\_thead  
 TABLES  
 lines = p\_lt\_lines[]  
 EXCEPTIONS  
 id = 1  
 language = 2  
 name = 3  
 not\_found = 4  
 object = 5  
 reference\_check = 6  
 wrong\_access\_to\_archive = 7  
 OTHERS = 8.  
 IF sy-subrc <> 0.  
*\* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO*  
*\* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.*  
 ENDIF.  
  
ENDFORM.

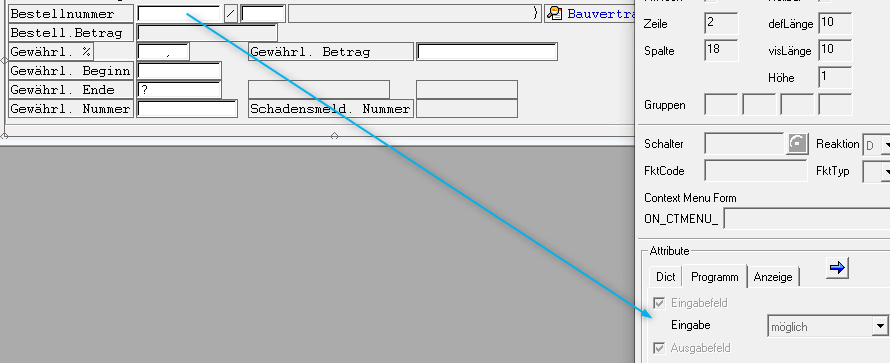
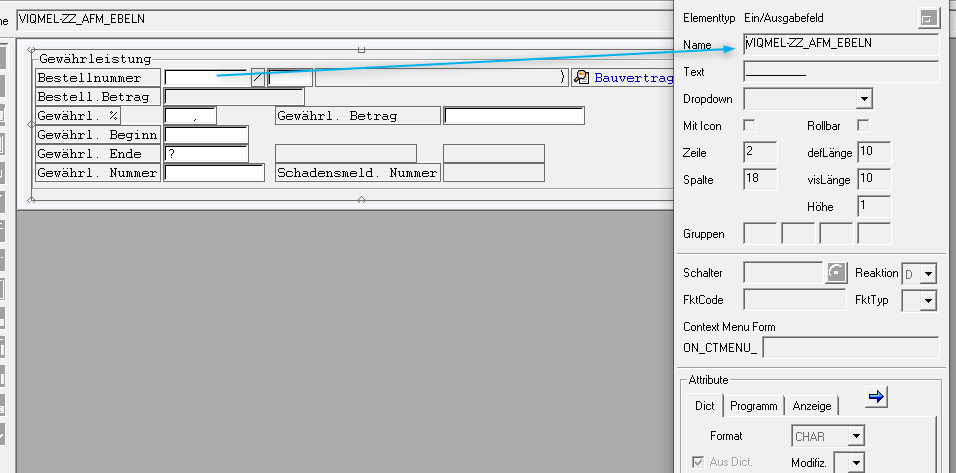
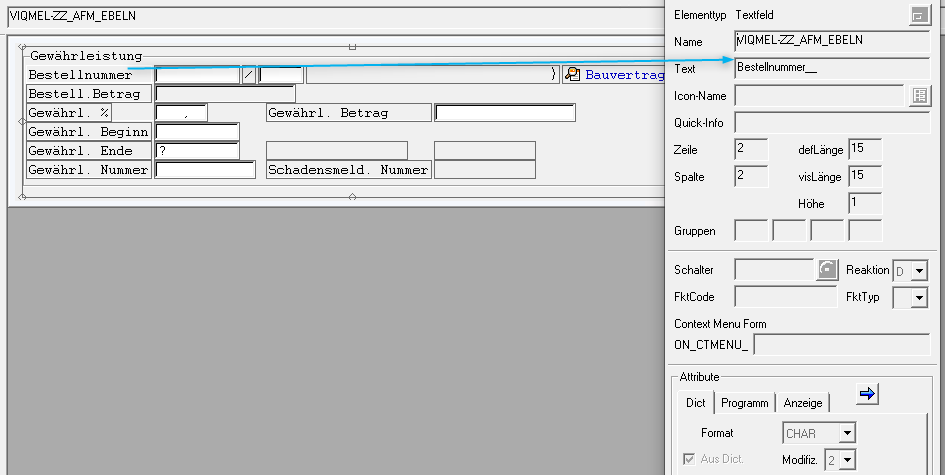
*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF24.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form ACTUALIZE\_LTEXT*  
*\*&---------------------------------------------------------------------\**  
*\* -->P\_LT\_LINES text*  
*\* <--P\_LO\_ERROR text*  
*\*----------------------------------------------------------------------\**  
FORM actualize\_ltext TABLES p\_lt\_lines STRUCTURE tline  
 CHANGING p\_lo\_error.  
  
*\* per Index*  
 DATA lt\_info\_tab TYPE soi\_cols\_table.  
 DATA ls\_info\_tab TYPE soi\_cols.  
*\*DATA lt\_data\_tab TYPE table.*  
 DATA lv\_doc\_nr TYPE i.  
  
*\*----------------------------------------------------------------------\**  
  
 MOVE '2' TO ls\_info\_tab-colindex.  
 APPEND ls\_info\_tab TO lt\_info\_tab.  
  
 CASE sy-langu.  
 WHEN 'D'. *"deutsch*  
 lv\_doc\_nr = 3.  
 WHEN OTHERS.  
 lv\_doc\_nr = 1.  
 ENDCASE.  
  
 IF NOT go\_worddoc IS INITIAL.  
  
 CALL METHOD go\_worddoc->clear\_table  
 EXPORTING  
 doctable\_number = lv\_doc\_nr  
 IMPORTING  
 error = p\_lo\_error.  
  
 CALL METHOD go\_worddoc->insert\_table  
 EXPORTING  
 data\_table = p\_lt\_lines[]  
 info\_table = lt\_info\_tab  
 lowerbound = 0  
 upperbound = 0  
 doctable\_number = lv\_doc\_nr  
 clearoption = 1  
 startrow = 1  
 varsize = 'X'  
*\* no\_flush = ' '*  
*\* wholetable = ' '*  
 IMPORTING  
 error = p\_lo\_error.  
*\* retcode =*  
  
 ENDIF.  
  
ENDFORM.

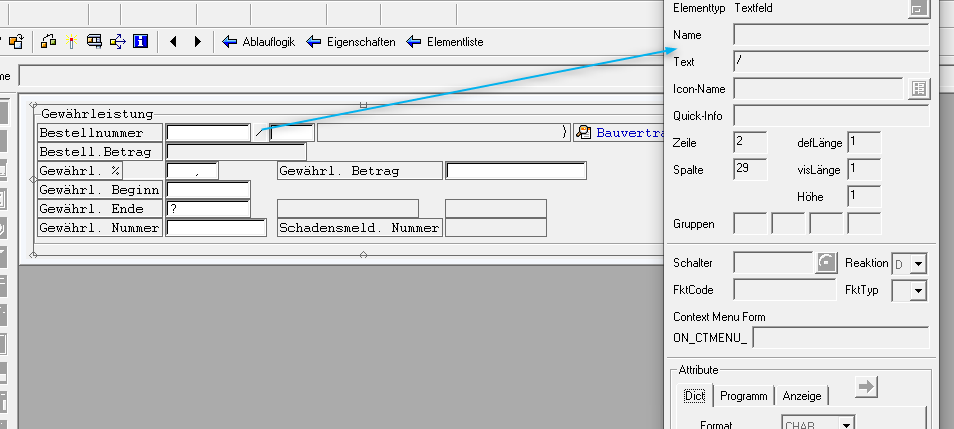
*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI13.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module CREATE\_AUSWAHL INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE create\_auswahl INPUT.  
  
 DATA lt\_values TYPE vrm\_values WITH HEADER LINE.  
 DATA lv\_id TYPE vrm\_id.  
 DATA lv\_classname TYPE bds\_clsnam.  
 DATA lv\_classtype TYPE bds\_clstyp.  
  
*\*&---------------------------------------------------------------------\**  
  
 CLEAR: lt\_uris2,  
 lt\_signature2,  
 lt\_components2,  
 lt\_sign\_sort\_r2,  
 lt\_sign\_sort\_r2[],  
 lt\_values,  
 lt\_values[].  
  
 lv\_id = 'GV\_RUEGE'.  
 lv\_object\_key2 = viqmel-qmnum.  
  
 IF gv\_qmart = 'SG' OR gv\_qmart = 'SM'.  
*\* IF gv\_qmart = 'SM' AND fl\_txtma = 1.*  
 IF ( fl\_txtma = 1 AND gv\_qmart = 'SM' ) OR ( fl\_txtma = 1 AND gv\_qmart = 'SG' ).  
 CLEAR lv\_object\_key2.  
 CLEAR lt\_uris2[].  
 CLEAR lt\_signature2[].  
 CLEAR lt\_components2[].  
*\* BDN Adresse Details für die Vorlage ermitteln*  
 SELECT SINGLE cls\_nam clstyp typeid FROM /afm/oaer INTO CORRESPONDING FIELDS OF gs\_oaer  
 WHERE mncod = gv\_mncod  
 AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 lv\_classname = gs\_oaer-cls\_nam.  
 lv\_classtype = gs\_oaer-clstyp.  
 ENDIF.  
 CONCATENATE viqmel-qmnum gv\_mncod INTO lv\_object\_key2.  
 ELSE.  
*\* BDN Adresse Details für die Vorlage ermitteln*  
 SELECT SINGLE cls\_nam clstyp typeid FROM /afm/oaer INTO CORRESPONDING FIELDS OF gs\_oaer  
 WHERE mncod = 'ALLG'  
 AND sprache = sy-langu.  
 IF sy-subrc = 0.  
 lv\_classname = gs\_oaer-cls\_nam.  
 lv\_classtype = gs\_oaer-clstyp.  
 ENDIF.  
 ENDIF. *" ( fl\_txtma = 1 AND gv\_qmart = 'SM' ) OR ( fl\_txtma = 1 AND gv\_qmart = 'SG' )*  
 ENDIF. *" gv\_qmart = 'SG' OR gv\_qmart = 'SM'.*  
  
 CALL FUNCTION 'BDS\_BUSINESSDOCUMENT\_GET\_URL'  
 EXPORTING  
*\* LOGICAL\_SYSTEM =*  
 classname = lv\_classname  
 classtype = lv\_classtype  
*\* CLIENT = SY-MANDT*  
 object\_key = lv\_object\_key2  
*\* URL\_LIFETIME =*  
*\* STANDARD\_URL\_ONLY =*  
*\* DATA\_PROVIDER\_URL\_ONLY =*  
*\* WEB\_APPLIC\_SERVER\_URL\_ONLY =*  
*\* URL\_USED\_AT =*  
 TABLES  
 uris = lt\_uris2  
 signature = lt\_signature2  
 components = lt\_components2  
 EXCEPTIONS  
 nothing\_found = 1  
 parameter\_error = 2  
 not\_allowed = 3  
 error\_kpro = 4  
 internal\_error = 5  
 not\_authorized = 6  
 OTHERS = 7.  
 CASE sy-subrc.  
 WHEN 0.  
*\* lt\_sign\_copy = lt\_signature.*  
*\* Aktuellste Rüge-Nr. ziehen <=> höchster Eintrag in BDS\_KEYWORD*  
 LOOP AT lt\_signature2 INTO ls\_signature2  
 WHERE prop\_name EQ 'BDS\_KEYWORD'.  
 IF ls\_signature2-prop\_value CP 'document\*'.  
 ls\_signature2-prop\_value = ls\_signature2-prop\_value.  
 APPEND ls\_signature2 TO lt\_sign\_sort\_r2.  
 ENDIF.  
 ENDLOOP.  
 SORT lt\_sign\_sort\_r2 DESCENDING BY prop\_value.  
 LOOP AT lt\_sign\_sort\_r2.  
 MOVE lt\_sign\_sort\_r2-prop\_value+8(1) TO lt\_values-key.  
 CONCATENATE 'Mängelrüge'(m05) lt\_sign\_sort\_r2-prop\_value+8(1) INTO lt\_values-text SEPARATED BY space.  
 APPEND lt\_values. CLEAR lt\_values.  
 ENDLOOP.  
 DELETE ADJACENT DUPLICATES FROM lt\_values.  
 ENDCASE.  
  
 CALL FUNCTION 'VRM\_SET\_VALUES'  
 EXPORTING  
 id = lv\_id  
 values = lt\_values[].  
  
ENDMODULE.

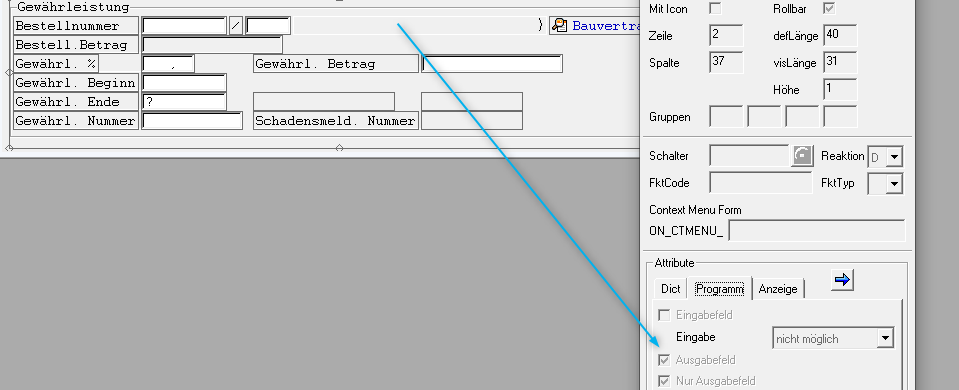
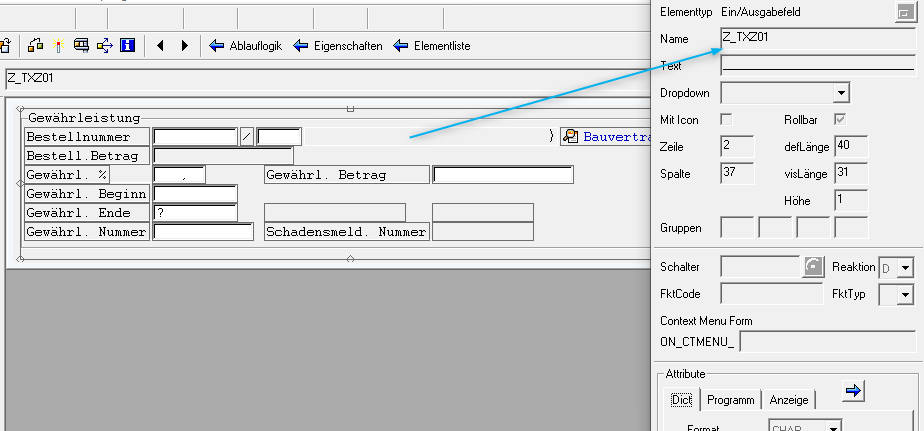
Bildschirm🡪0104

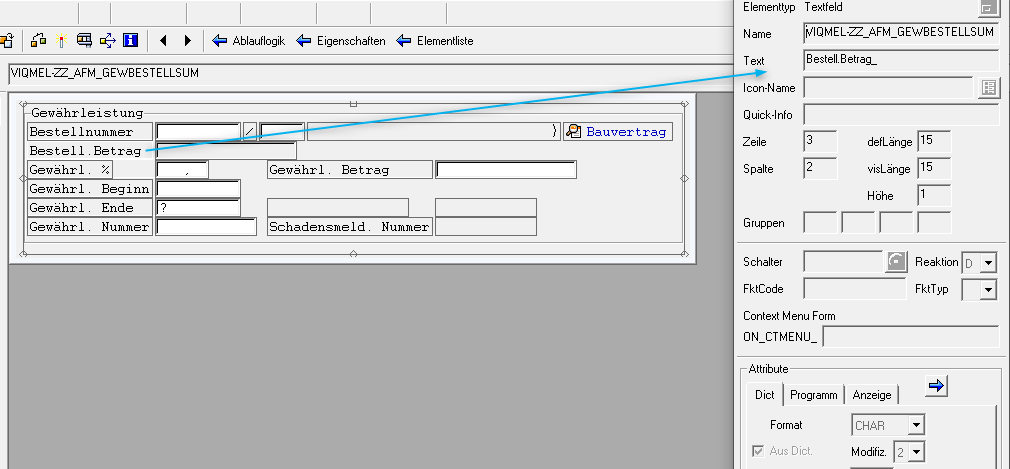
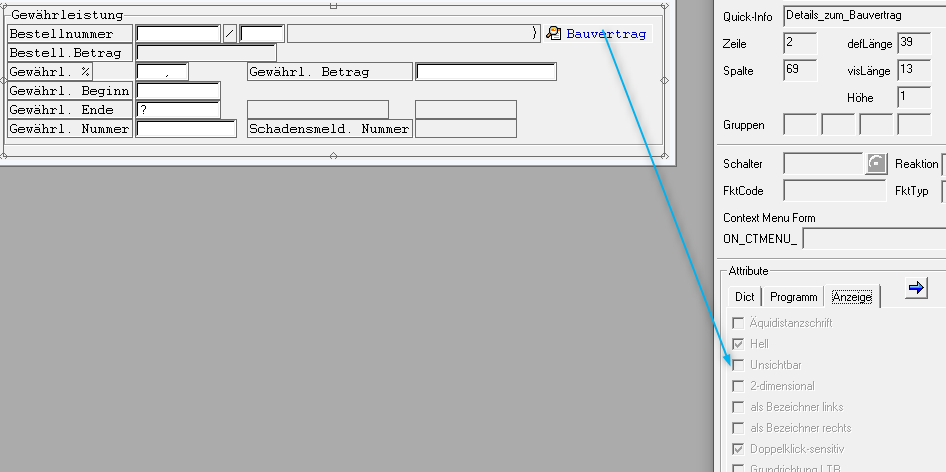
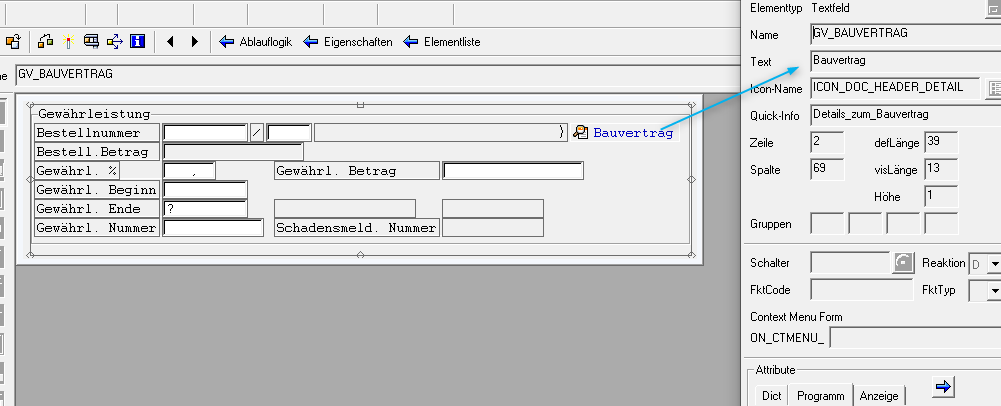


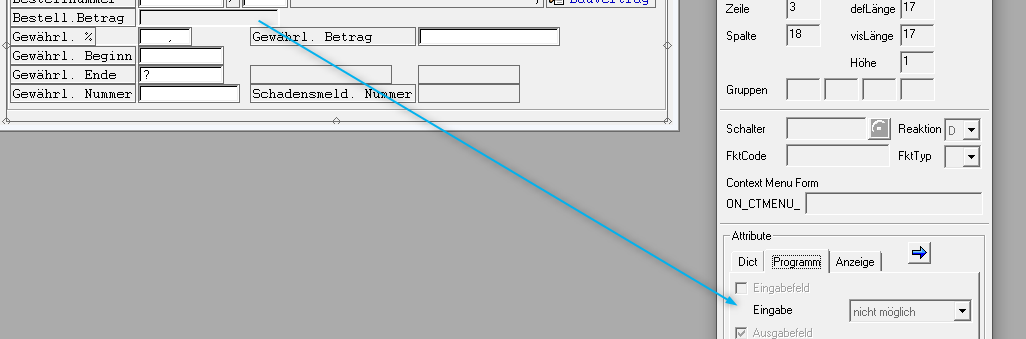
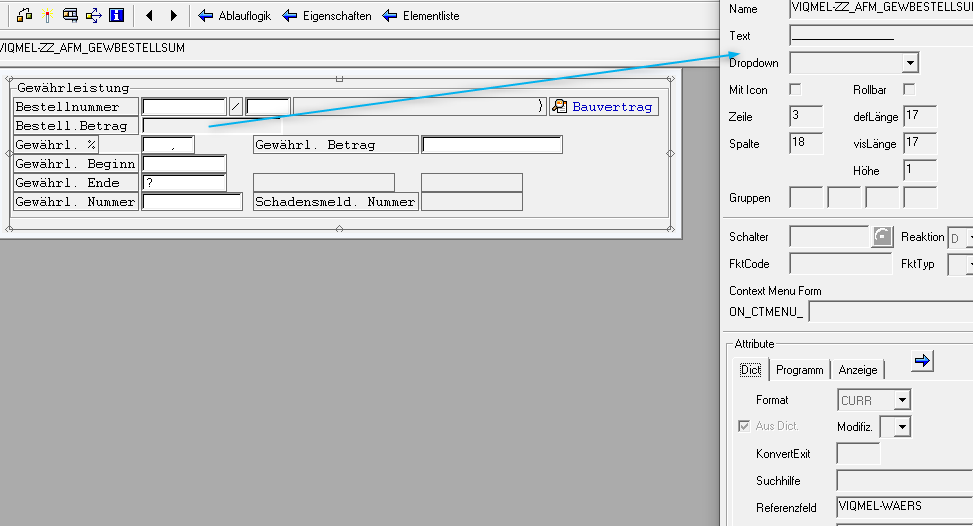
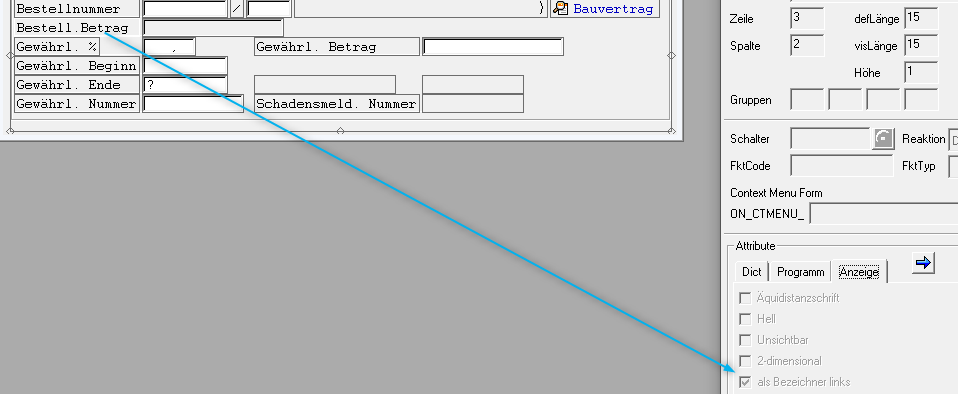


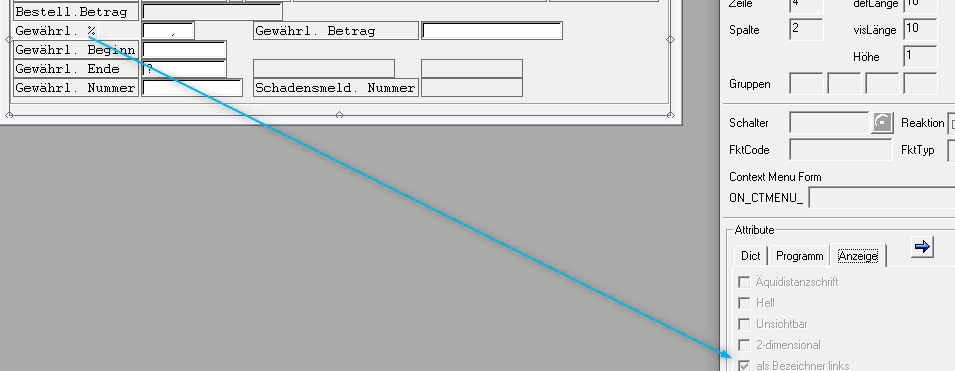
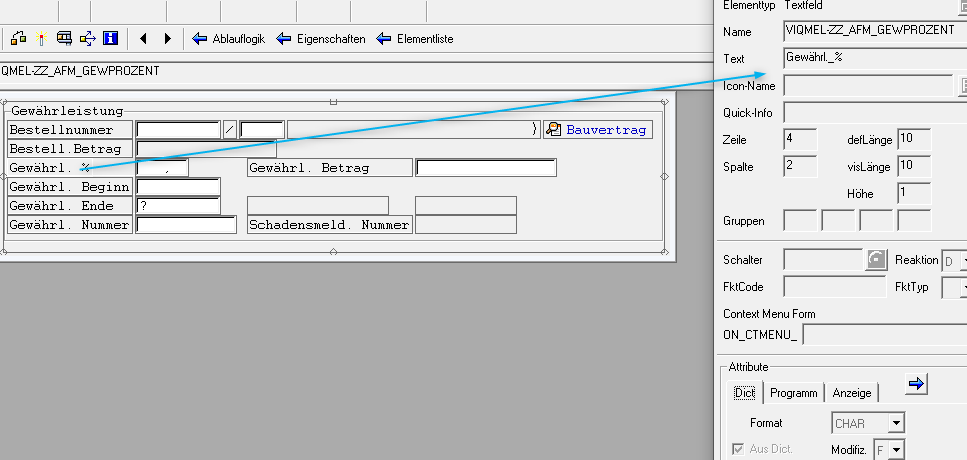


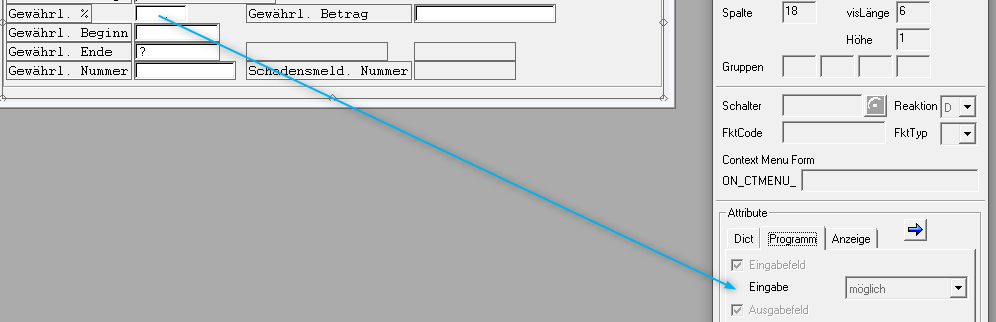
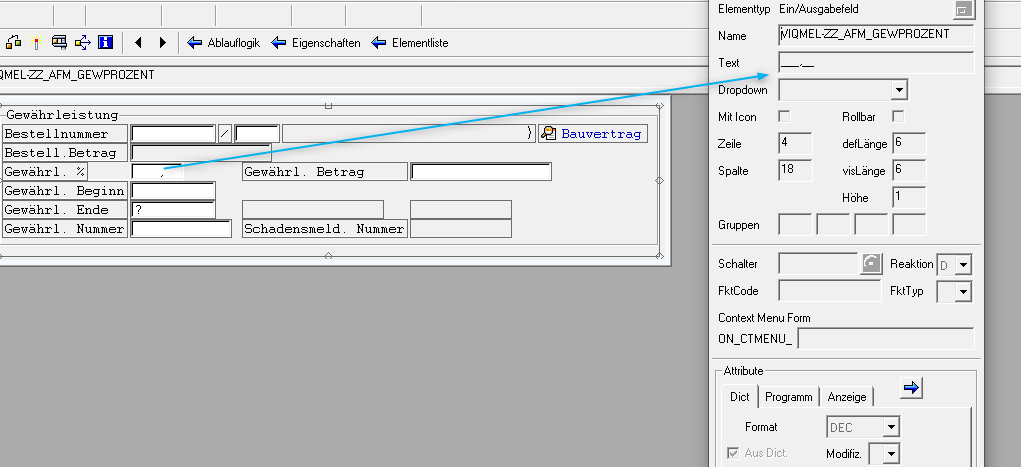


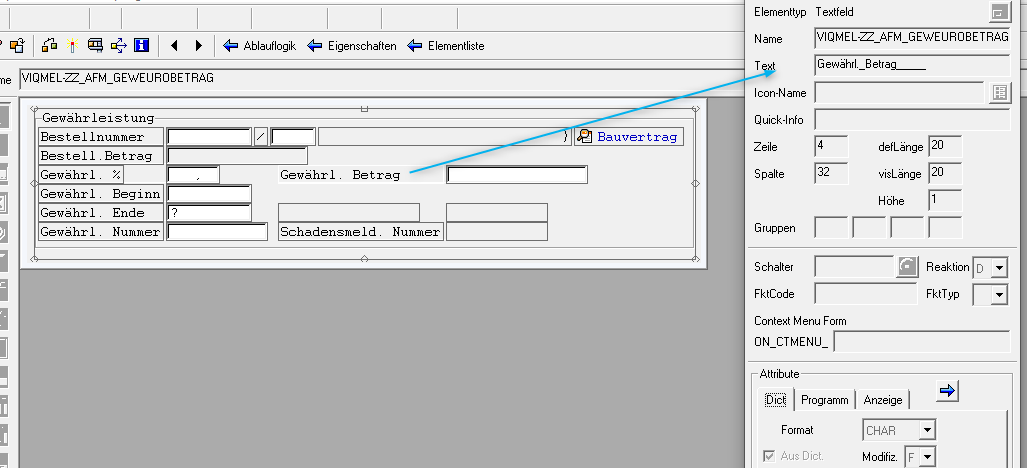


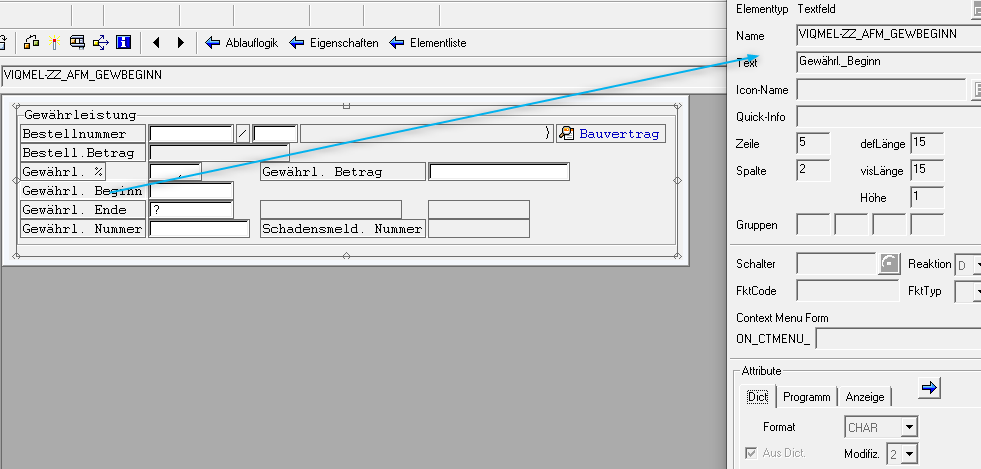
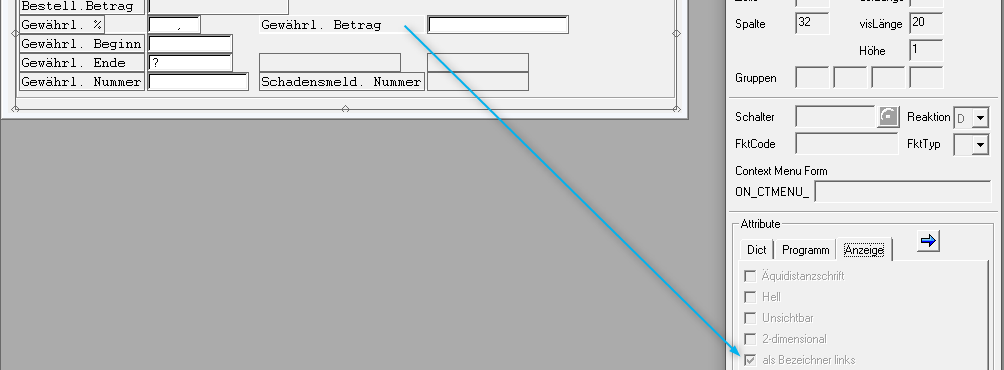


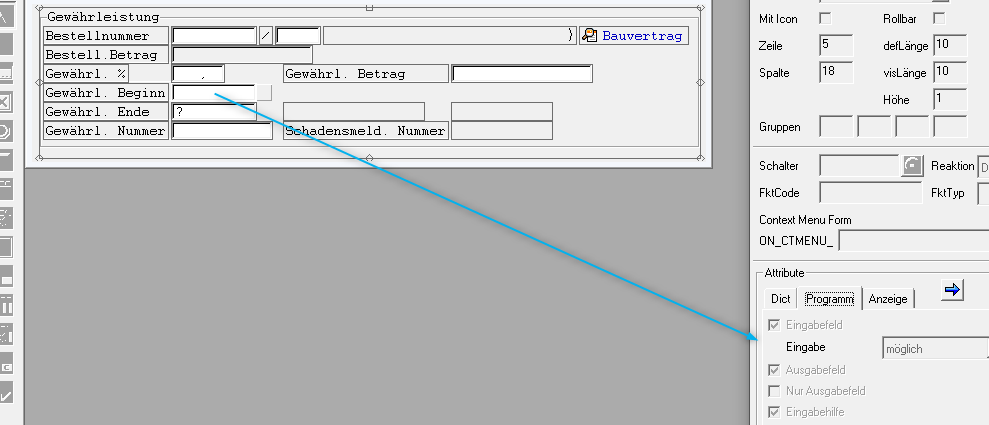
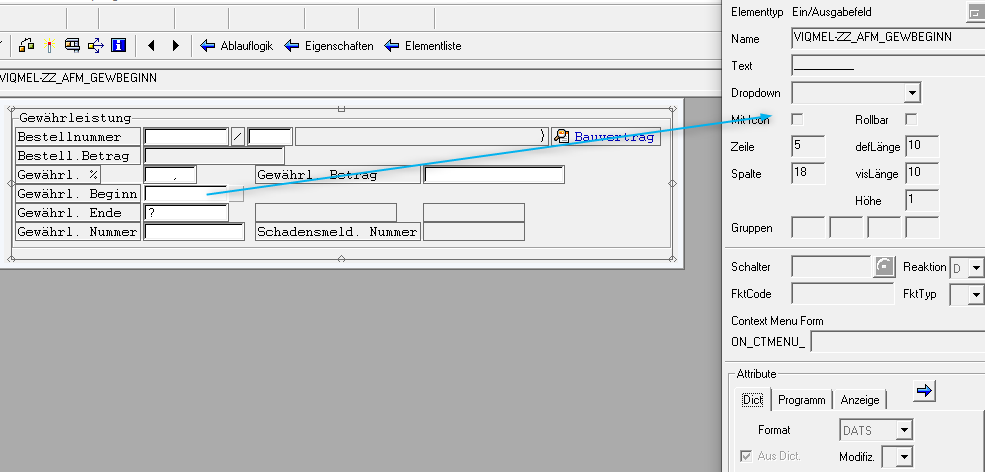
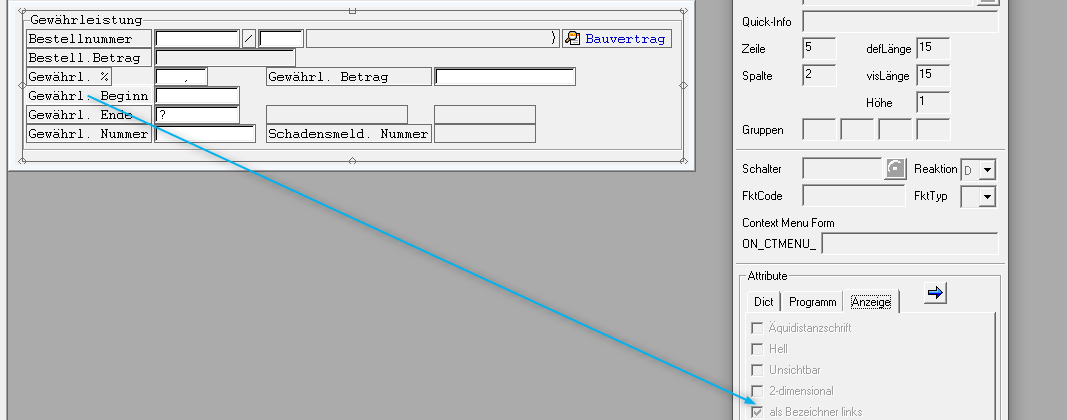


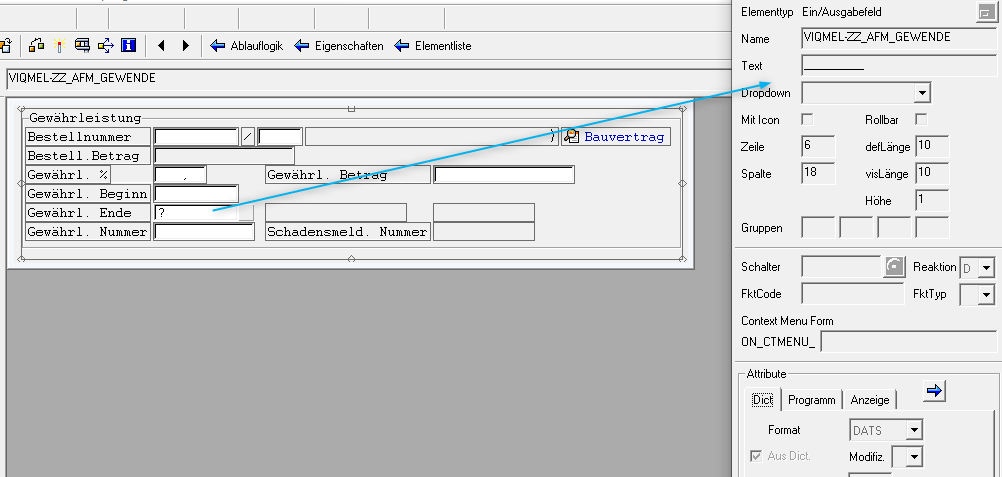
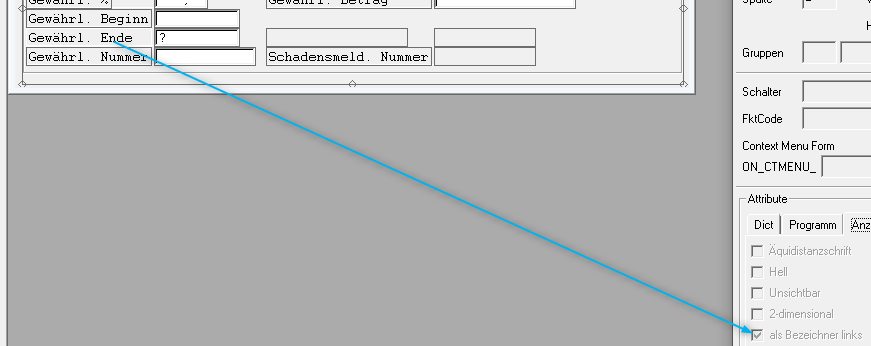
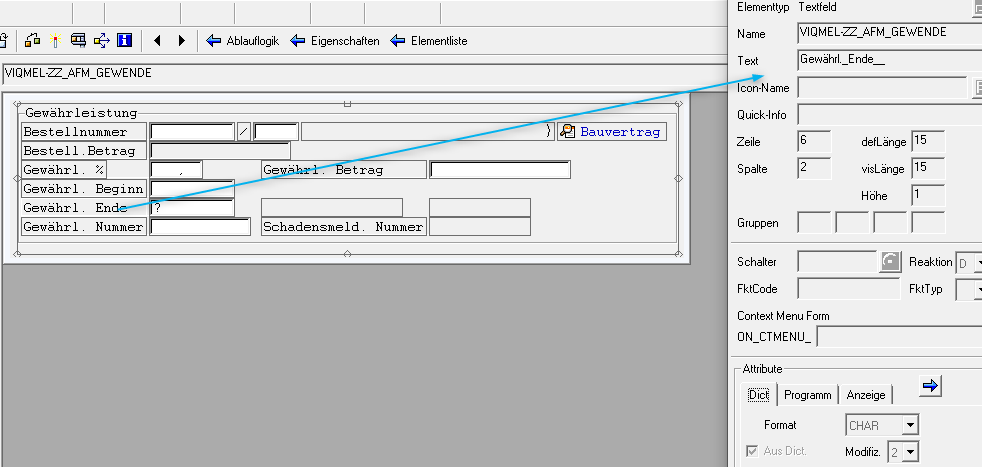


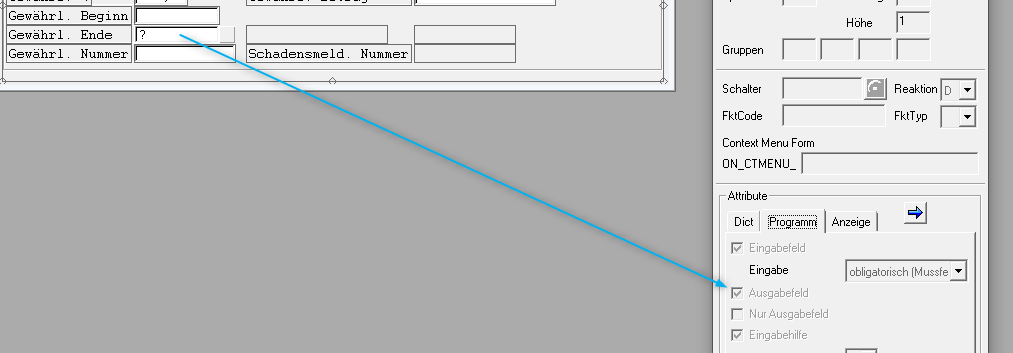


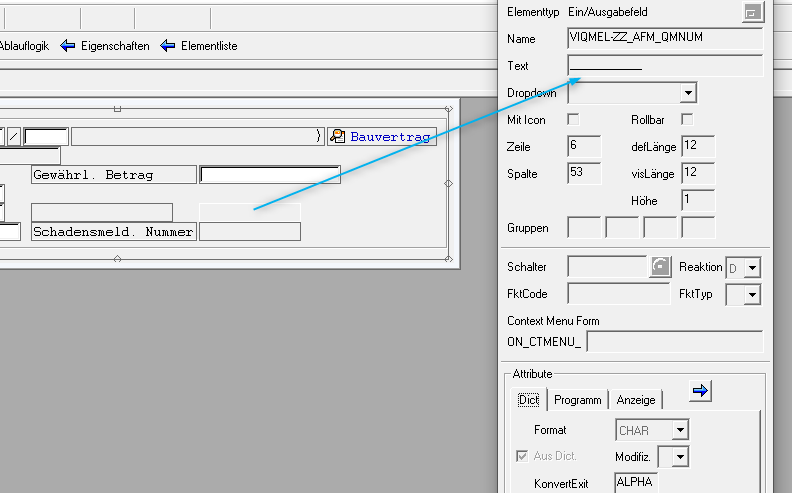
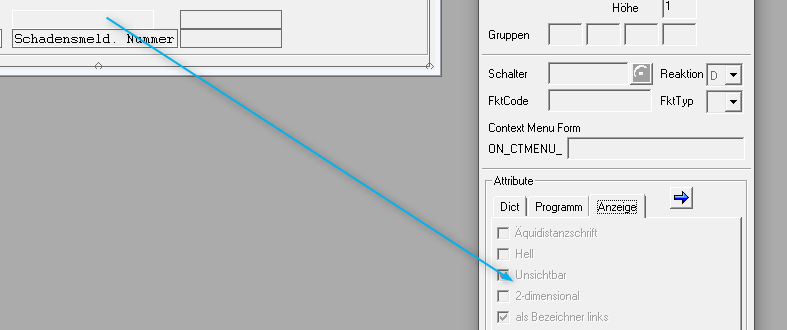
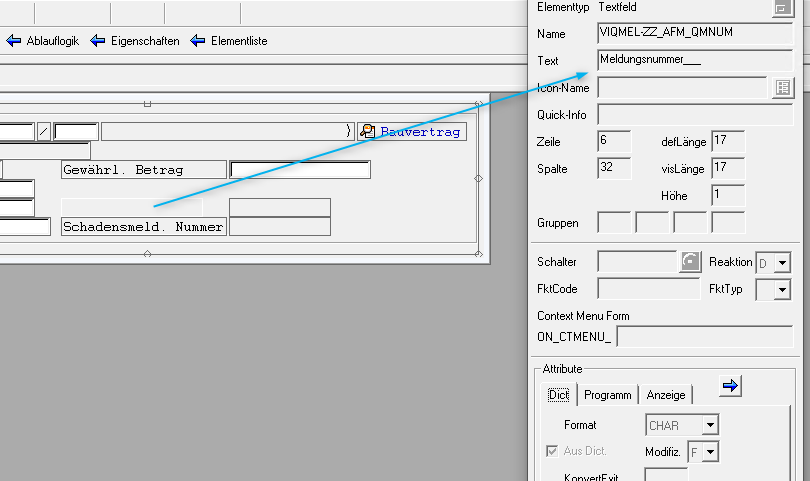


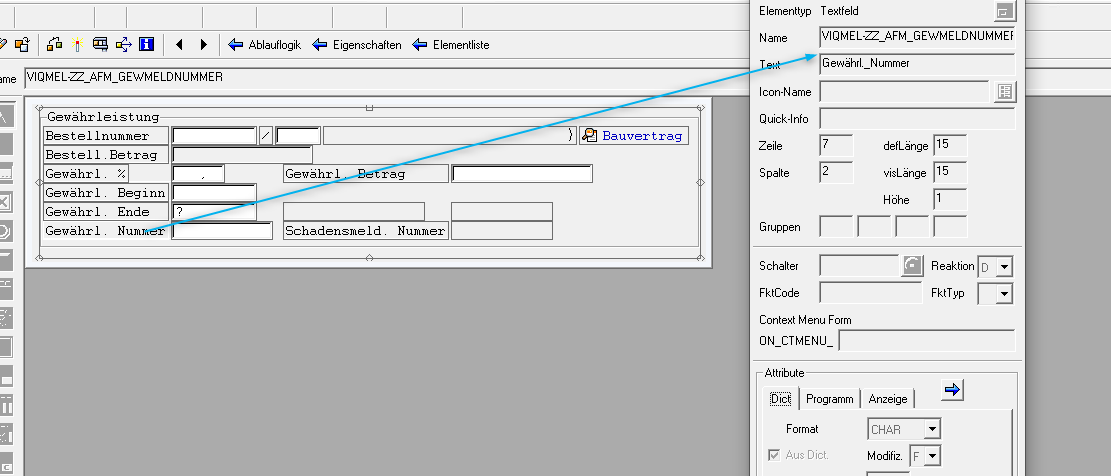
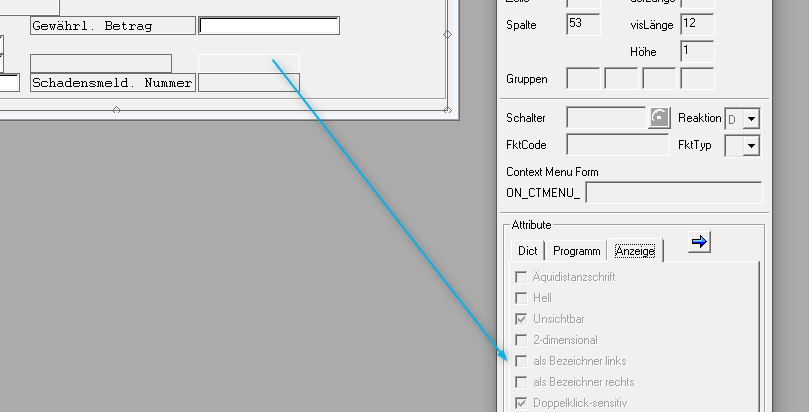
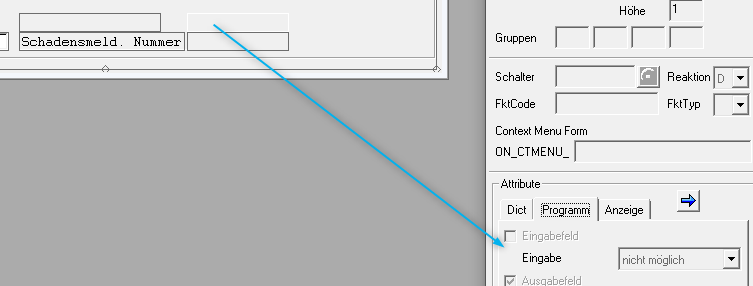


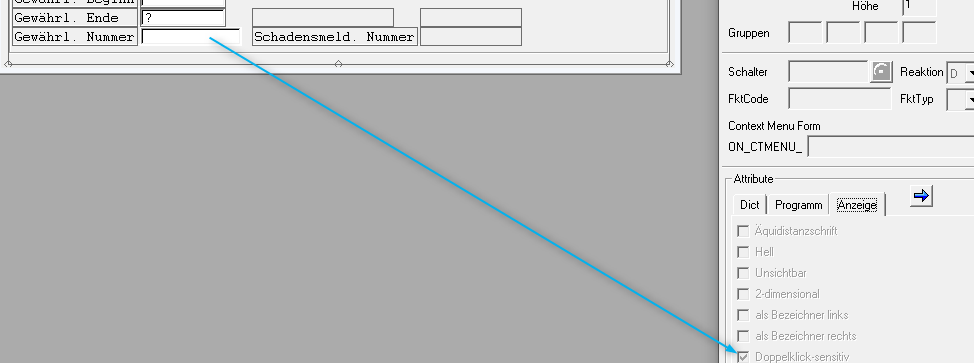
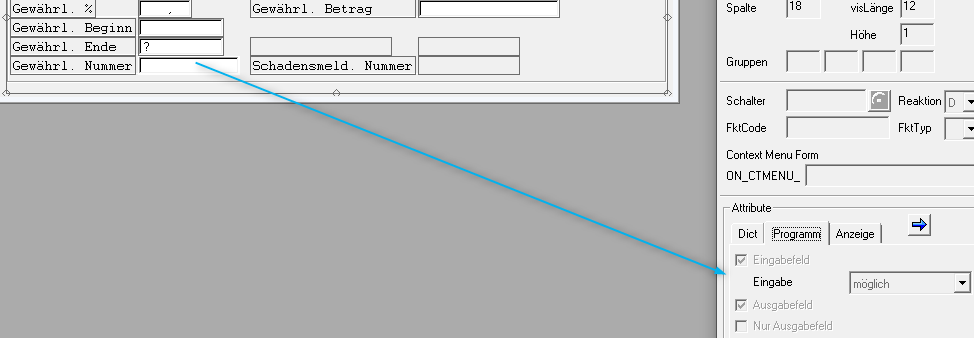
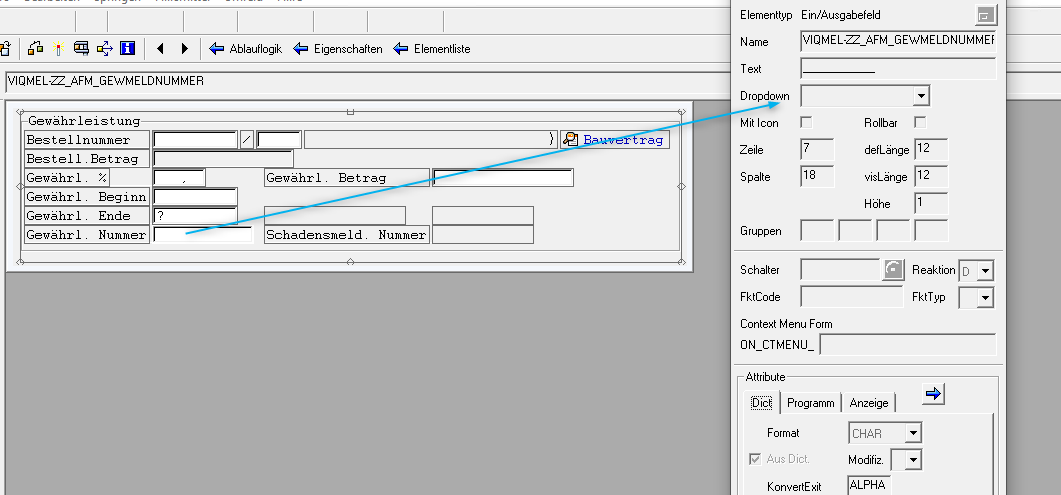
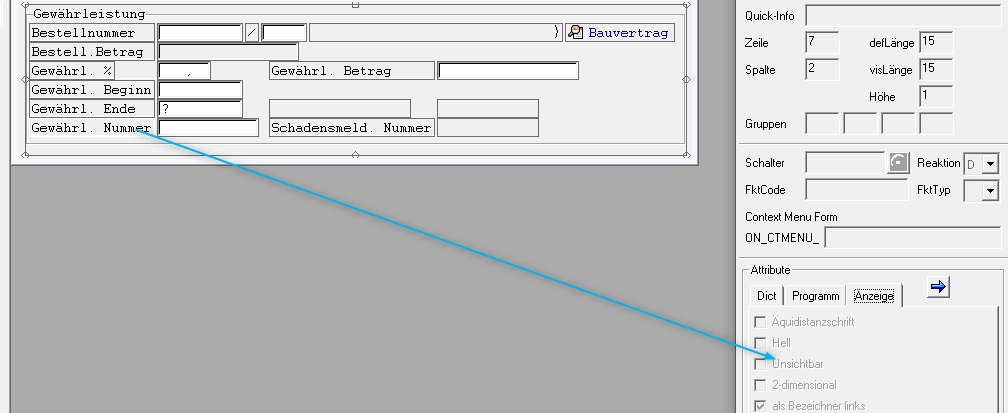


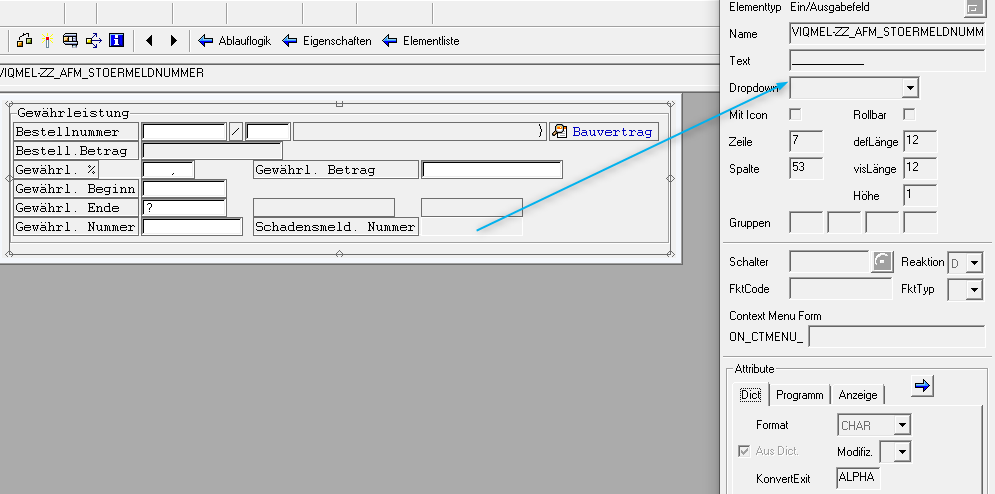
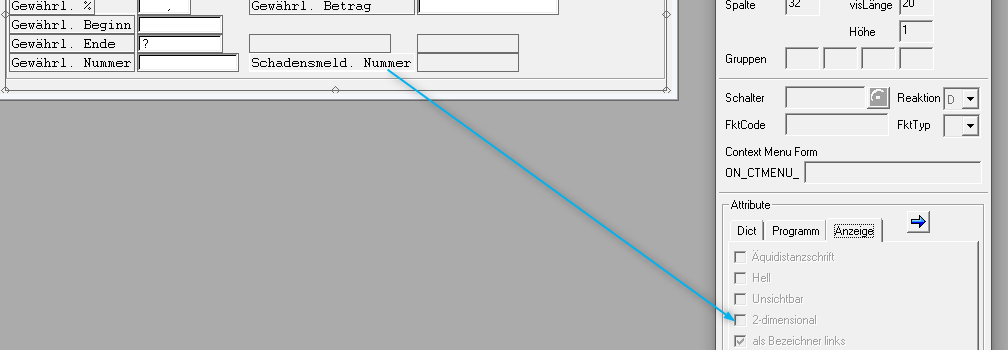
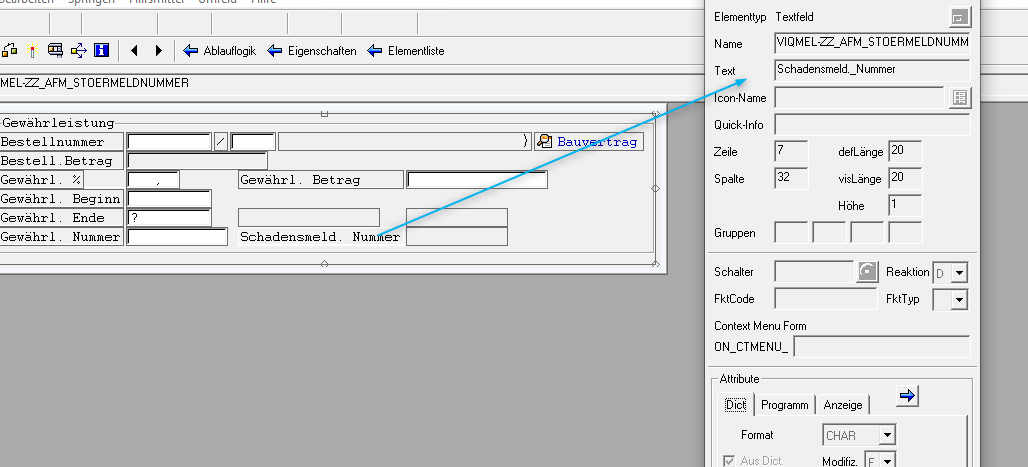


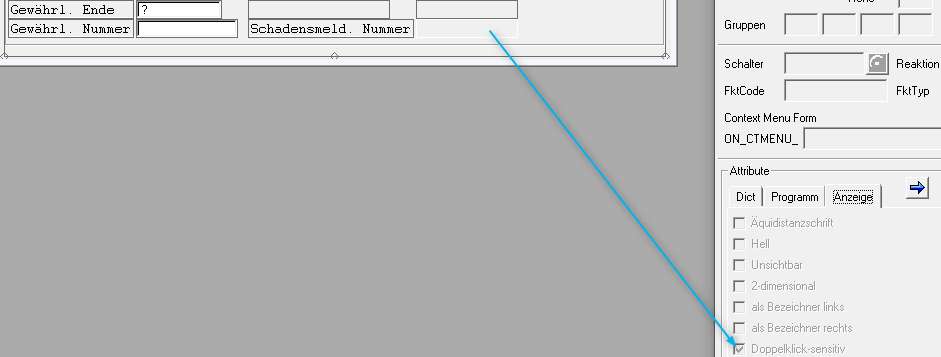
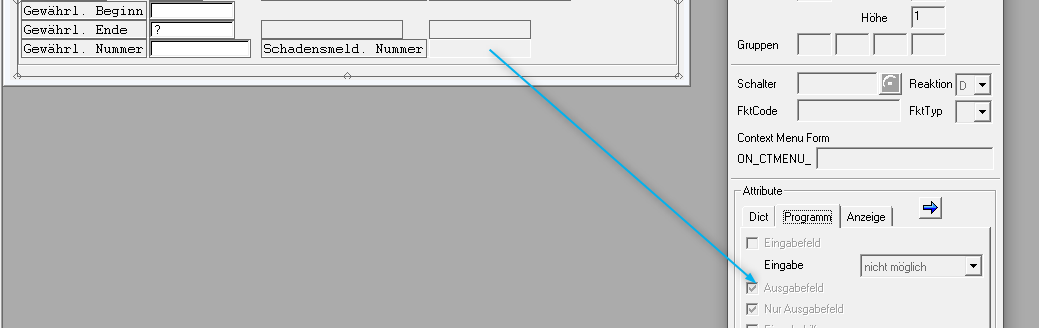












PROCESS BEFORE OUTPUT.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
 MODULE status\_0104.  
  
  
PROCESS AFTER INPUT.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
 FIELD: viqmel-zz\_afm\_ebeln,  
 viqmel-zz\_afm\_ebelp,  
 viqmel-zz\_afm\_qmnum,  
*\* viqmel-zz\_afm\_stoermeldnummer,*  
 viqmel-zz\_afm\_gewmeldnummer.  
  
 MODULE user\_command\_0104.  
  
*\* prüft ob gewünschter Endetermin liegt vor dem Meldungsdatum.*  
 FIELD viqmel-zz\_afm\_gewende  
 MODULE check\_enddate\_gg\_melddatum.  
  
 FIELD viqmel-zz\_afm\_gewende  
 MODULE check\_enddate.  
 FIELD viqmel-zz\_afm\_gewbeginn  
 MODULE check\_begindate ON INPUT.  
  
*\* Berechnung des Gewährleistungsbetrag*  
 FIELD viqmel-zz\_afm\_gewprozent  
 MODULE calc\_gewbetrag ON REQUEST.  
  
 FIELD viqmel-zz\_afm\_geweurobetrag  
 MODULE calc\_gewprozent ON REQUEST.  
  
*\*Prüft ob die Eingaben gültig sind!*  
 FIELD viqmel-zz\_afm\_gewmeldnummer  
 MODULE check\_gew\_meldung ON INPUT .  
  
 FIELD viqmel-zz\_afm\_gewprozent  
 MODULE calc\_gewahrleistung ON REQUEST.  
 FIELD viqmel-zz\_afm\_geweurobetrag  
 MODULE calc\_gewahrleistung ON REQUEST.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMO07.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module STATUS\_0104 OUTPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE status\_0104 OUTPUT.  
  
*\* SET PF-STATUS 'xxxxxxxx'.*  
*\* SET TITLEBAR 'xxx'.*  
  
 PERFORM status\_0104.  
  
 LOOP AT SCREEN.  
 IF screen-group1 EQ 'BAU'.  
 IF NOT viqmel-zz\_afm\_ebeln IS INITIAL.  
 MOVE '1' TO screen-input.  
 ELSE.  
 MOVE '0' TO screen-input.  
 ENDIF.  
 MODIFY SCREEN.  
 ENDIF.  
 ENDLOOP.  
  
ENDMODULE.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI03.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module USER\_COMMAND\_0104 INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE user\_command\_0104 INPUT.  
  
 PERFORM user\_command\_0104.  
  
ENDMODULE.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMF13.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form USER\_COMMAND\_0104*  
*\*&---------------------------------------------------------------------\**  
FORM user\_command\_0104 .  
  
*\* Kurztext zur Bestellnummer lesen*  
 CONSTANTS: p\_parvw TYPE parvw VALUE 'LF'.  
 DATA: lcurso LIKE riwo1-curso,  
 lqmart LIKE qmel-qmart,  
 lqmtyp LIKE tq80-qmtyp,  
 p\_lifnr TYPE lifnr,  
 p\_parnr TYPE i\_parnr.  
  
*\*&---------------------------------------------------------------------\**  
  
 SET PARAMETER ID 'ZEBELN' FIELD viqmel-zz\_afm\_ebeln.  
 IF NOT viqmel-zz\_afm\_ebeln IS INITIAL AND NOT viqmel-zz\_afm\_ebelp IS INITIAL.  
 SELECT SINGLE txz01 INTO z\_txz01 FROM ekpo  
 WHERE ebeln = viqmel-zz\_afm\_ebeln AND ebelp = viqmel-zz\_afm\_ebelp.  
 IF sy-subrc > 0.  
 MESSAGE 'Bestellung wurde nicht gefunden' TYPE 'S'.  
 CLEAR z\_txz01.  
 ENDIF.  
 ENDIF.  
  
 IF NOT viqmel-zz\_afm\_qmnum IS INITIAL.  
 SELECT SINGLE qmtxt INTO z\_qmtxt FROM qmel  
 WHERE qmnum = viqmel-zz\_afm\_qmnum.  
 IF sy-subrc > 0.  
 MESSAGE 'Garantiemeldung nicht gefunden' TYPE 'S'.  
 CLEAR z\_qmtxt.  
 ENDIF.  
 ENDIF.  
  
 IF NOT viqmel-zz\_afm\_ebeln IS INITIAL.  
 SELECT SINGLE lifnr FROM ekko INTO p\_lifnr  
 WHERE ebeln EQ viqmel-zz\_afm\_ebeln.  
 IF sy-subrc EQ 0.  
 MOVE: p\_lifnr TO p\_parnr.  
 gv\_parnr = p\_parnr.  
 CALL FUNCTION 'PM\_PARTNER\_MAINTAIN'  
 EXPORTING  
*\* OBJNR = ' '*  
 parnr = p\_parnr  
 parvw = 'LF'  
*\* ADRNR =*  
*\* ADR\_HANDLE =*  
*\* TABIX = 0*  
*\* TABIX\_OBJNR = 0*  
*\* MSGTY = 'E'*  
*\* CPD\_POP\_UP = 'X'*  
*\* CHECK\_DEBITOR = 'X'*  
*\* IMPORTING*  
*\* COUNTER =*  
 EXCEPTIONS  
 invalid\_parnr = 1  
 cpd\_customer = 2  
 invalid\_parvw = 3  
 parvw\_unique = 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.*  
 ENDIF.  
 ENDIF.  
 ELSE.  
 CLEAR gv\_parnr.  
 ENDIF.  
  
*\*Berechnet die Basissumme für die Berechnung des Gewährleistungsbetrag*  
 CLEAR gv\_effrwert.  
 IF NOT viqmel-zz\_afm\_ebeln IS INITIAL.  
 CALL FUNCTION 'ZFB\_AFM\_GET\_BESTELLBETRAG'  
 EXPORTING  
 purchaseorder = viqmel-zz\_afm\_ebeln  
 TABLES  
 po\_items1 = it\_ekpo.  
 IF sy-dbcnt > 0.  
 IF NOT viqmel-zz\_afm\_ebelp IS INITIAL.  
 READ TABLE it\_ekpo WITH KEY po\_item = viqmel-zz\_afm\_ebelp INTO wa\_ekpo TRANSPORTING po\_number po\_item eff\_value .  
 IF sy-subrc = 0.  
 gv\_effrwert = wa\_ekpo-eff\_value.  
 ENDIF.  
 ELSE.  
 LOOP AT it\_ekpo INTO wa\_ekpo.  
 gv\_effrwert = gv\_effrwert + wa\_ekpo-eff\_value.  
 ENDLOOP.  
 ENDIF.  
 ENDIF.  
 ENDIF.  
 viqmel-zz\_afm\_gewbestellsum = gv\_effrwert.  
  
*\*&---------------------------------------------------------------------\**  
  
 CASE sy-ucomm.  
  
 WHEN 'AZOB'. *"Doppelclick auf Bestellnummer oder ?*  
  
 GET CURSOR FIELD lcurso.  
 IF lcurso = 'VIQMEL-ZZ\_AFM\_EBELN' OR lcurso = 'VIQMEL-ZZ\_AFM\_EBELP'.  
 SET PARAMETER ID 'BES' FIELD viqmel-zz\_afm\_ebeln.  
 CALL TRANSACTION 'ME23' AND SKIP FIRST SCREEN.  
  
 ELSEIF lcurso = 'VIQMEL-ZZ\_AFM\_QMNUM'.  
 SET PARAMETER ID 'IQM' FIELD viqmel-zz\_afm\_qmnum.  
 CLEAR: lqmart, lqmtyp.  
 SELECT SINGLE qmart INTO lqmart FROM qmel  
 WHERE qmnum = viqmel-zz\_afm\_qmnum.  
 SELECT SINGLE qmtyp INTO lqmtyp FROM tq80  
 WHERE qmart = lqmart.  
 IF lqmtyp = '03'.  
 CALL TRANSACTION 'IW53' AND SKIP FIRST SCREEN.  
 ELSE.  
 CALL TRANSACTION 'IW23' AND SKIP FIRST SCREEN.  
 ENDIF.  
  
 ELSEIF lcurso = 'GV\_BAUVERTRAG'.  
 IF NOT viqmel-zz\_afm\_ebeln IS INITIAL.  
 SET PARAMETER ID 'BES' FIELD viqmel-zz\_afm\_ebeln.  
 CALL TRANSACTION '/AFM/BAUMOD' AND SKIP FIRST SCREEN.  
 ENDIF.  
 ELSE.  
  
 CASE lcurso.  
 WHEN 'VIQMEL-ZZ\_AFM\_GEWMELDNUMMER'.  
 IF fl\_chkgw IS INITIAL.  
 SET PARAMETER ID 'IQM' FIELD viqmel-zz\_afm\_gewmeldnummer.  
 CALL TRANSACTION 'IW22' AND SKIP FIRST SCREEN.  
 ENDIF.  
 WHEN 'VIQMEL-ZZ\_AFM\_STOERMELDNUMMER'.  
 IF fl\_chkgw IS INITIAL.  
 DATA: lt\_enq TYPE TABLE OF seqg3,  
 ls\_enq TYPE seqg3.  
 CLEAR: lt\_enq, lt\_enq[].  
*\* Sperreinträge aus der Sperrtabelle lesen*  
 CALL FUNCTION 'ENQUEUE\_READ'  
 EXPORTING  
 gclient = sy-mandt  
 gname = 'QMEL'  
*\* GARG = ' '*  
 guname = sy-uname  
*\* LOCAL = ' '*  
*\* FAST = ' '*  
*\* GARGNOWC = ' '*  
*\* IMPORTING*  
*\* NUMBER =*  
*\* SUBRC =*  
 TABLES  
 enq = lt\_enq  
 EXCEPTIONS  
 communication\_failure = 1  
 system\_failure = 2  
 OTHERS = 3.  
 IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
 ENDIF.  
 READ TABLE lt\_enq INTO ls\_enq WITH KEY garg+3 = viqmel-zz\_afm\_stoermeldnummer.  
 IF sy-subrc = 0.  
 SET PARAMETER ID 'IQM' FIELD viqmel-zz\_afm\_stoermeldnummer.  
 CALL TRANSACTION 'IW23' AND SKIP FIRST SCREEN. *" <-- lege ZM an aus Z4 oder bearbeitet ZM aus Z4 - nur ANZEIGE möglich !*  
 ELSE.  
 SET PARAMETER ID 'IQM' FIELD viqmel-zz\_afm\_stoermeldnummer.  
 CALL TRANSACTION 'IW22' AND SKIP FIRST SCREEN.  
 ENDIF.  
 ENDIF.  
 WHEN OTHERS.  
 ENDCASE.  
 ENDIF.  
  
 ENDCASE.  
  
ENDFORM.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI04.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module CHECK\_ENDDATE\_GG\_MELDDATUM INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE check\_enddate\_gg\_melddatum INPUT.  
  
 IF viqmel-zz\_afm\_gewende < viqmel-qmdat.  
 IF viqmel-zz\_afm\_gewende IS NOT INITIAL.  
 MESSAGE e474(im) WITH viqmel-zz\_afm\_gewende '' viqmel-qmdat . *" Gewünschtes Ende &1 &2 liegt vor dem Meldungsdatum &3 &4*  
 ENDIF.  
 ENDIF.  
  
ENDMODULE.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI05.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module CHECK\_ENDDATE INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE check\_enddate INPUT.  
  
 IF viqmel-qmart = 'ZG'. *"<-- Gewährleistungsmeldung*  
 IF viqmel-zz\_afm\_gewende IS INITIAL.  
 MESSAGE e483(im). *" <-- Bitte ein Bezugsdatum eingeben*  
 ENDIF.  
 ENDIF.  
  
ENDMODULE.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI06.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module CHECK\_BEGINDATE INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE check\_begindate INPUT.  
  
 IF viqmel-zz\_afm\_gewende < viqmel-zz\_afm\_gewbeginn .  
 IF viqmel-zz\_afm\_gewende IS NOT INITIAL.  
 MESSAGE e470(im) WITH viqmel-zz\_afm\_gewende '' viqmel-zz\_afm\_gewbeginn. *" Endtermin liegt &1 &2 vor Starttermin &3 &4*  
 ENDIF.  
 ENDIF.  
  
ENDMODULE.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI07.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module CALC\_GEWBETRAG INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE calc\_gewbetrag INPUT.  
  
 flag\_gew = '1'.  
  
ENDMODULE.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI08.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module CALC\_GEWPROZENT INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE calc\_gewprozent INPUT.  
  
 flag\_gew = '2'.  
  
ENDMODULE.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI09.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module CHECK\_GEW\_MELDUNG INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE check\_gew\_meldung INPUT.  
  
 PERFORM check\_id USING 'ZG' 'ZG' viqmel-zz\_afm\_gewmeldnummer.  
  
ENDMODULE.

*\*----------------------------------------------------------------------\**  
*\*\*\*INCLUDE ZXQQMI11.*  
*\*----------------------------------------------------------------------\**  
  
*\*&---------------------------------------------------------------------\**  
*\*& Module CALC\_GEWAHRLEISTUNG INPUT*  
*\*&---------------------------------------------------------------------\**  
MODULE calc\_gewahrleistung INPUT.  
  
 IF viqmel-zz\_afm\_gewprozent > 100.  
 MESSAGE e530(im). *" <-- Eingaben unzulässig, bitte überprüfen Sie die Eingaben*  
 ELSE.  
 IF flag\_gew = '1'.  
 viqmel-zz\_afm\_geweurobetrag = ( gv\_effrwert \* viqmel-zz\_afm\_gewprozent ) / 100.  
 ENDIF.  
 ENDIF.  
  
 IF viqmel-zz\_afm\_geweurobetrag > gv\_effrwert.  
 MESSAGE e530(im). *" <-- Eingaben unzulässig, bitte überprüfen Sie die Eingaben*  
 ELSE.  
 IF flag\_gew = '2'.  
 viqmel-zz\_afm\_gewprozent = ( viqmel-zz\_afm\_geweurobetrag \* 100 ) / gv\_effrwert.  
 ENDIF.  
 ENDIF.  
  
ENDMODULE.

Bildschirm🡪0103