Anhang F

ABAP/4-Beispiel-Programme

62 SAP Schulungsprogramme umfassen die ABAP/4-Beispiel-Auswahl. Der jeweilige Programmtext der Beispiel-Programme ist im folgenden abgedruckt.

In den Untersuchungen dieser Diplomarbeit sind die Programme nur mit ihrem intern vergebenen Namen aufgeführt. Das Listung der Beispiel-Auswahl enthält sowie den internen Namen und den SAP-Reportnamen.

In bezug auf das Orginal-Programm wurden einige Beispiele verändert. Diese kennzeichen sich durch einen Stern (*) im Namen aus.

Programm S-UM-001 (RFSCHU04)

```
REPORT RFSCHU04.

SEGMENTS: KA, KB, KC.

GET KA.
  WRITE: KTNRA, NAME1, PLORT, LAND1.

GET KB.
  WRITE: 10 BUKRB, AKONT, BEGRU.

GET KC.
  WRITE: 13 JAHRC, KUMUM.
```

Programm S-UM-002 (RFSCHU07)

```
REPORT RFSCHU07.

SEGMENTS: KA.

START-OF-SELECTION.
  WRITE TEXT-001.

TOP-OF-PAGE.
  WRITE TEXT-002.

GET KA.
  DETAIL.
  WRITE: KTNRA, NAME1, PLORT, LAND1.

END-OF-PAGE.
  WRITE TEXT-003.

END-OF-SELECTION.
  WRITE TEXT-004.
```

Programm S-UM-003 (RFSCHU30*)

```
REPORT RFSCHU30.
SEGMENTS: KA, KB.

DATA: Y1 LIKE SY-CUROW,
      Y2 LIKE SY-CUROW.

GET KB.
  FORMAT INTENSIFIED OFF.
  WRITE: KA-KTNRA USING NO EDIT MASK,
         KB-BUKRB,
         KA-NAME1,
         KA-ORT01,
         KA-LAND1.
  HIDE:  KA-KTNRA,
         KB-KTNRB,
         KB-BUKRB.

END-OF-SELECTION.
  SET PF-STATUS 'BEG'.

AT PF14.
  Y1 = SY-CUROW.
  Y2 = Y1 + 7.
  WINDOW STARTING AT  1 Y1
           ENDING AT 79 Y2.
  READ SEGMENT KA.
  ULINE.
  WRITE: KA-KTNRA USING NO EDIT MASK,
         KB-BUKRB,
         KA-NAME1,
         KA-ORT01,
         KA-LAND1.
  ULINE.
  WRITE: / 'Bankkonto:      '(001), 18 KA-BANKN,
         / 'Bankleitzahl:   '(002), 18 KA-BANKL,
         / 'Bankanschrift:  '(003), 18 KA-BANKA,
         / 'Postscheckkonto:'(004), 18 KA-PSKTO.
  ULINE.
```

Fortsetzung Programm S-UM-003 (RFSCHU30*)

```
AT PF15.
  WINDOW STARTING AT 46 10
           ENDING AT 77 15
             WITH FRAME.
  READ SEGMENT: KA, KB.
  WRITE:    KA-KTNRA USING NO EDIT MASK,
         10 KA-NAME1.
  ULINE.
  WRITE: / 'Bestellwert'(011),
           SY-DATUM+2(2),
           15 ':',
           18 KB-BJWRT,
         / '...  im Vorjahr:'(012),18 KB-BVWRT,
         / 'Einkaufsgruppe:'(013),18 KB-EKGRP,
         / 'Gesamtumsatz:  '(014),18 KB-UMSAZ.
AT PF16.
  Y1 = SY-CUROW.
  Y2 = Y1 + 6.
  WINDOW STARTING AT 20 Y1
           ENDING AT 61 Y2
             WITH FRAME
            TITLE 'Konditionen'(026).
  READ SEGMENT: KA, KB.
  WRITE: KA-KTNRA USING NO EDIT MASK,
         KA-NAME1.
  ULINE.
  WRITE: / 'Zahlungskoditinen: '(021), 22 KB-ZTERM,
         / 'Wechselkonditionen:'(022), 22 KB-WKOND,
         / 'Zahlungsarten:     '(023), 22 KB-ZAHLA,
         / 'Zahlungsschluessel:'(024), 22 KB-ZLSCH,
         / 'Vermerk:           '(025), 22 KB-KVERM.
AT LINE-SELECTION.
  WINDOW STARTING AT 01 16
           ENDING AT 79 23
             WITH FRAME.
  READ SEGMENT: KA.
  WRITE: / KTNRA USING NO EDIT MASK,
           KA-NAME1,
         / KA-NAME2 UNDER NAME1,
         / KA-STRAS UNDER NAME1,
         / KA-ORT01 UNDER NAME1.
```

Programm S-UM-004 (RMSCHU11)

```
REPORT RMSCHU11.

DATA: ERGEBNIS TYPE P,
      P1 TYPE P VALUE 1,
      P2 TYPE P VALUE 3,
      P3 TYPE P VALUE 7,
      P4 TYPE P VALUE 20.

PARAMETERS:
  ERSDA LIKE SY-DATUM.

INITIALIZATION.
  ERSDA = SY-DATUM - 90.

START-OF-SELECTION.
  DETAIL.

  ERGEBNIS = P1 * P2 * P3 * P4.
  WRITE: / '1 * 3 * 7 * 20 =',
           30(6) ERGEBNIS.

  ERGEBNIS = SY-DATUM - ERSDA.
  WRITE: / SY-DATUM DD/MM/YY,
           '-',
           ERSDA DD/MM/YY,
           '=',
           30(6) ERGEBNIS.

  WRITE: / (6) ERGEBNIS,
           '* 113 / 7 ='.
           ERGEBNIS = ERGEBNIS * 113 / 7.
  WRITE: 30(6) ERGEBNIS.

  WRITE: / '(',
           (6) ERGEBNIS,
           '- 80 ) * 17 ='.
           ERGEBNIS = ( ERGEBNIS - 80 ) * 17.
  WRITE: 30(6) ERGEBNIS.

  SUMMARY. ULINE.
```

Programm S-UM-005 (RMSCHU21)

```
REPORT RMSCHU21.

DATA: C1(1) TYPE C,
      C8(8) TYPE C,
      C6(6) TYPE C,
      P8(8) TYPE P,
      X1(1) TYPE X,
      X4(4) TYPE X,
      D1 TYPE 1,
      D2 TYPE 2.

START-OF-SELECTION.
  C1 = ’A’.
  MOVE C1 TO C8.
  WRITE: /1 ’C’, 5 ’1 ’, 10 C1, 41 ’C’, 45 ’8 ’, 60 C8.

  C8 = ’ABCDEFGH’.
  MOVE C8 TO C1.
  WRITE: /1 ’C’, 5 ’8 ’, 10 C8, 41 ’C’, 45 ’1 ’, 60 C1.

  C8 = ’-5380’.
  MOVE C8 TO P8.
  WRITE: /1 ’C’, 5 ’8 ’, 10 C8, 41 ’P’, 45 ’8 ’, 60 P8.

  C8 = ’15’.
  MOVE C8 TO X1.
  WRITE: /1 ’C’, 5 ’8 ’, 10 C8, 41 ’X’, 45 ’1 ’, 60 X1.

  C8 = ’-5380’.
  MOVE C8 TO X4.
  WRITE: /1 ’C’, 5 ’8 ’, 10 C8, 41 ’X’, 45 ’4 ’, 60 X4.

  C6 = ’121084’.
  MOVE C6 TO D1.
  WRITE: /1 ’C’, 5 ’6 ’, 10 C6, 41 ’1’, 45 ’6 ’, 60 D1.

  C6 = ’121084’.
  MOVE C6 TO D2.
  WRITE: /1 ’C’, 5 ’6 ’, 10 C6, 41 ’2’, 45 ’6 ’, 60 D2.
```

Fortsetzung Programm S-UM-005 (RMSCHU21)

```
P8 = 30250.
MOVE P8 TO C8.
WRITE: /1 'P', 5 '8 ', 10 P8, 41 'C', 45 '8 ', 60 C8.

P8 = 30250.
MOVE P8 TO X4.
WRITE: /1 'P', 5 '8 ', 10 P8, 41 'X', 45 '4 ', 60 X4.

P8 = 30250.
MOVE P8 TO X1.
WRITE: /1 'P', 5 '8 ', 10 P8, 41 'X', 45 '1 ', 60 X1.

P8 = 30250.
MOVE P8 TO D1.
WRITE: /1 'P', 5 '8 ', 10 P8, 41 '1', 45 '6 ', 60 D1.

X1 = 15.
MOVE X1 TO X4.
WRITE: /1 'X', 5 '1 ', 10 X1, 41 'X', 45 '4 ', 60 X4.

P8 = 30250.
MOVE P8 TO D2.
WRITE: /1 'P', 5 '8 ', 10 P8, 41 '2', 45 '6 ', 60 D2.

D1 = '121084'.
MOVE D1 TO P8.
WRITE: /1 '1', 5 '6 ', 10 D1, 41 'P', 45 '8 ', 60 P8.

D1 = '121084'.
MOVE D1 TO X4.
WRITE: /1 '1', 5 '6 ', 10 D1, 41 'X', 45 '4 ', 60 X4.

D1 = '121084'.
MOVE D1 TO D2.
WRITE: /1 '1', 5 '6 ', 10 D1, 41 '2', 45 '6 ', 60 D2.
```

Programm S-UM-006 (RFSCHU06)

```
REPORT RFSCHU06.

SEGMENTS: KA, KB, KC.

GET KA.
  WRITE: SPACE INPUT,
         KTNRA BLINKING USING NO EDIT MASK.
  FORMAT INTENSIFIED OFF.
  WRITE: NAME1 UNDERSCORED, PLORT INVERSE.

GET KC.
  DETAIL.
  WRITE: 10 KB-BUKRB, JAHRC, KUMUM INTENSIFIED.

GET KA LATE.
  ULINE.
```

Programm S-UM-007 (RFSCHU11)

```
REPORT RFSCHU11.

DATA: P1 TYPE P VALUE 2,
      P2 TYPE P VALUE 3,
      D1 TYPE P DECIMALS 2 VALUE '1.11',
      D2 TYPE P DECIMALS 2 VALUE '2.22',
      XTYPE P,
      YTYPE P DECIMALS 2,
      DAT TYPE 1 VALUE '240491'.

WRITE: 50 'P1 = 2D1 = 1,11',
       /50 'P2 = 3D2 = 2,11'.

X = P1 + P2.
WRITE: / 'X = P1 + P2', 30(6) X.

X = P1 - P2.
WRITE: / 'X = P1 - P2', 30(6) X.

X = P1 * P2.
WRITE: / 'X = P1 * P2', 30(6) X.
```

Fortsetzung Programm S-UM-007 (RFSCHU11)

```
X = P1 / P2.
WRITE: / 'X = P1 / P2', 30(6) X.
WRITE: / 'Feld X ist o h n e
           Nachkommastellen definiert!'.
ULINE.

Y = D1 + D2.
WRITE: / 'Y = D1 + D2', 30(6) Y.

Y = D1 - D2.
WRITE: / 'Y = D1 - D2', 30(6) Y.

Y = D1 * D2.
WRITE: / 'Y = D1 * D2', 30(6) Y.

Y = P1 / P2.
WRITE: / 'Y = P1 / P2', 30(6) Y.
WRITE: / 'Feld Y ist m i t (2)
           Nachkommastellen definiert!',
       / '( bei Attributen ist
           Festpunktarithmetik angekreuzt )'.
ULINE.

Y = P1 + P2 / 5.
WRITE: / 'P1 + P2 / 5', 30(6) Y.

Y = ( P1 + P2 ) / 5.
WRITE: / '( P1 + P2 ) / 5',30(6) Y.
ULINE.

WRITE: 50 'SY-DATUM = 100591',
       /50 'DAT= 240491'.

X = SY-DATUM - DAT.
WRITE: / 'X = SY-DATUM - DAT', 30(6) X.
```

Programm S-UM-008 (RFSCHU17)

```
REPORT RFSCHU17.

SEGMENTS: KB.

GET KB.
NEW-PAGE.
WRITE: / 'SY-MANDT', SY-MANDT.
WRITE: / 'SY-UNAME', SY-UNAME.
WRITE: / 'SY-LANGU', SY-LANGU.
WRITE: / 'SY-DATUM', SY-DATUM.
WRITE: / 'SY-UZEIT', SY-UZEIT.
WRITE: / 'SY-BATCH', SY-BATCH.
WRITE: / 'SY-DBNAM', SY-DBNAM.
WRITE: / 'SY-SUBRC', SY-SUBRC.
WRITE: / 'SY-PAGNO', SY-PAGNO.
WRITE: / 'SY-CPAGE', SY-CPAGE.
WRITE: / 'SY-TITLE', SY-TITLE.
SKIP.
WRITE: / 'SY-WAERS', SY-WAERS.
```

Programm S-UM-009 (RFSCHU34)

```
REPORT RFSCHU34.

SEGMENTS:   KA.
PARAMETERS: NAME(12) DEFAULT SY-UNAME.
DYNPROS:    D00001, D02041, D02060.

START-OF-SELECTION.
  MOVE: SY-MANDT TO D00001-MANDT,
        NAME     TO D00001-BNAME,
        SY-UZEIT TO D00001-BNAME+6,
        'PASS'   TO D00001-BCODE,
        SY-LANGU TO D00001-SPRAS.
  TRANSFER-DYNPRO D00001.

GET KA.
  MOVE: 'S'       TO D02041-SP,
        'TS02'    TO D02041-TC,
        KA-KTNRA  TO D02041-KTNRA,
        '01'      TO D02041-VCODE.
  TRANSFER-DYNPRO D02041.

  MOVE: '01'    TO D02060-REGIO.
  TRANSFER-DYNPRO D02060.

  WRITE: KA-KTNRA USING NO EDIT MASK,
         KA-NAME1, KA-PLORT, KA-LAND1.
```

Programm S-UM-010 (RFSCHU18*)

```
REPORT RFSCHU18.
SEGMENTS: KA, KB, KC.
DATA: PSALDO TYPE P,
      MSALDO TYPE P.
TABLES T001.

GET KB.
  SELECT SINGLE * FROM T001
                  WHERE BUKRS EQ KB-BUKRB.

GET KC.
  NEW-PAGE.
  WRITE: KA-KTNRA, KA-NAME1, 48 T001-TEXT, KC-JAHRC,
         /10 KA-NAME2, /10 KA-ORT01.
  ULINE. DETAIL.
  PSALDO = KC-UMSAV.

  MSALDO = KC-UM01S - KC-UM01H.
  PSALDO = PSALDO + MSALDO.
  WRITE:   /01 '01',
            08 KC-UM01S CURRENCY LOCAL,
            27 KC-UM01H CURRENCY LOCAL,
            45 MSALDO CURRENCY LOCAL,
            64 PSALDO CURRENCY LOCAL.

  MSALDO = KC-UM02S - KC-UM02H.
  PSALDO = PSALDO + MSALDO.
  WRITE:   /01 '02',
            08 KC-UM02S CURRENCY LOCAL,
            27 KC-UM02H CURRENCY LOCAL,
            45 MSALDO CURRENCY LOCAL,
            64 PSALDO CURRENCY LOCAL.

  MSALDO = KC-UM03S - KC-UM03H.
  PSALDO = PSALDO + MSALDO.
  WRITE:   /01 '03',
            08 KC-UM03S CURRENCY LOCAL,
            27 KC-UM03H CURRENCY LOCAL,
            45 MSALDO CURRENCY LOCAL,
            64 PSALDO CURRENCY LOCAL.

  SUMMARY. ULINE.
```

Programm S-UM-011 (RMSCHU18*)

```
REPORT RMSCHU18.

SEGMENTS: KA, KB, KC.

GET KC.
  NEW-PAGE.
  KUMUM = KUMUM * -1.
  WRITE:         BUKRB,
                 KTNRA,
          16     JAHRC,
          27     NAME1,
        / 27     PLORT,
          59(15) KUMUM.
  ULINE.
  WRITE: /    TEXT-001,
           40 TEXT-002,
           61 TEXT-003.
  ULINE.

  WRITE: /       '01',
           37(15) UM01S,
           59(15) UM01H.

  WRITE: /       '02',
           37(15) UM02S,
           59(15) UM02H.

  WRITE: /       '03',
           37(15) UM03S,
           59(15) UM03H.

  WRITE: /       '04',
           37(15) UM04S,
           59(15) UM04H.

  WRITE: /       '05',
           37(15) UM05S,
           59(15) UM05H.

  WRITE: /       '06',
           37(15) UM06S,
           59(15) UM06H.
```

Programm S-M-001 (RFSCHU18*)

```
REPORT RFSCHU18.
SEGMENTS: KA, KB, KC.
DATA: PSALDO TYPE P,
TABLES T001.

GET KB.
  SELECT SINGLE * FROM T001
                  WHERE BUKRS EQ KB-BUKRB.

GET KC.
  NEW-PAGE.
  WRITE: KA-KTNRA, KA-NAME1, 48 T001-TEXT, KC-JAHRC,
         /10 KA-NAME2, /10 KA-ORT01.
  ULINE. DETAIL.
  PSALDO = KC-UMSAV.

  PERFORM UMSAETZE USING: KC-UM01S KC-UM01H '01',
                          KC-UM02S KC-UM02H '02',
                          KC-UM03S KC-UM03H '03'.

  SUMMARY. ULINE.

* ----- FORM UMSAETZE ---------------------------- *
FORM UMSAETZE USING UMS UMH TXT.
  DATA: MSALDO TYPE P.

  MSALDO = UMS - UMH. PSALDO = PSALDO + MSALDO.
  WRITE:   /01 TXT,
            08 UMS CURRENCY LOCAL,
            27 UMH CURRENCY LOCAL,
            45 MSALDO CURRENCY LOCAL,
            64 PSALDO CURRENCY LOCAL.
ENDFORM.
```

Programm S-M-002 (RFSCHU18*)

```
REPORT RFSCHU18.
SEGMENTS: KA, KB, KC.
DATA: PSALDO TYPE P,
TABLES T001.

GET KB.
  SELECT SINGLE * FROM T001
                  WHERE BUKRS EQ KB-BUKRB.

GET KC.
  NEW-PAGE.
  WRITE: KA-KTNRA, KA-NAME1, 48 T001-TEXT, KC-JAHRC,
         /10 KA-NAME2, /10 KA-ORT01.
  ULINE. DETAIL.
  PSALDO = KC-UMSAV.

  CALL FUNCTION 'UMSAETZE'
              EXPORTING UMS  = KC-UM01S
                        UMH  = KC-UM01H
                        TXT  = '01'
                        PSAE = PSALDO
              IMPORTING PSAI = PSALDO.

  CALL FUNCTION 'UMSAETZE'
              EXPORTING UMS  = KC-UM02S
                        UMH  = KC-UM02H
                        TXT  = '02'
                        PSAE = PSALDO
              IMPORTING PSAI = PSALDO.

  CALL FUNCTION 'UMSAETZE'
              EXPORTING UMS  = KC-UM03S
                        UMH  = KC-UM03H
                        TXT  = '03'
                        PSAE = PSALDO
              IMPORTING PSAI = PSALDO.

  SUMMARY. ULINE.
```

Programm S-M-003 (RMSCHU18*)

```
REPORT RMSCHU18.

SEGMENTS: KA, KB, KC.

GET KC.
  NEW-PAGE.
  KUMUM = KUMUM * -1.
  WRITE:         BUKRB,
                 KTNRA,
          16     JAHRC,
          27     NAME1,
        / 27     PLORT,
          59(15) KUMUM.
  ULINE.
  WRITE: /    TEXT-001,
           40 TEXT-002,
           61 TEXT-003.
  ULINE.

  PERFORM POSTENZEILE USING: UM01S UM01H '01',
                             UM02S UM02H '02',
                             UM03S UM03H '03',
                             UM04S UM04H '04',
                             UM05S UM05H '05',
                             UM06S UM06H '06',

* ----- FORM POSTENZEILE----------------------- *
FORM POSTENZEILE USING SOLL HABEN MONAT.
  WRITE:
         /        MONAT,
           37(15) SOLL,
           59(15) HABEN.
ENDFORM.
```

Programm N-UM-001 (RFSCHU13*)

```
REPORT RFSCHU13.
SEGMENTS: KA.

GET KA.
  IF PSTLZ = SPACE.
    FORMAT INTENSIFIED.
  ELSE.
    FORMAT INTENSIFIED OFF.
  ENDIF.

  WRITE: KTNRA, PSTLZ, ORT01, STRAS.
```

Programm N-UM-001-1 (RFSCHU13)

```
REPORT RFSCHU13.
SEGMENTS: KA.

GET KA.
  IF PSTLZ = SPACE OR ORT01 = SPACE OR STRAS = SPACE.
    FORMAT INTENSIFIED.
  ELSE.
    FORMAT INTENSIFIED OFF.
  ENDIF.

  WRITE: KTNRA, PSTLZ, ORT01, STRAS.
```

Programm N-UM-001-2 (RFSCHU13*)

```
REPORT RFSCHU13.
SEGMENTS: KA.

GET KA.
  IF PSTLZ = SPACE AND ORT01 = SPACE AND STRAS = SPACE.
    FORMAT INTENSIFIED.
  ELSE.
    FORMAT INTENSIFIED OFF.
  ENDIF.

  WRITE: KTNRA, PSTLZ, ORT01, STRAS.
```

Programm N-UM-001-3 (RFSCHU13*)

```
REPORT RFSCHU13.
SEGMENTS: KA.

GET KA.
  IF PSTLZ = SPACE.
    IF ORT01 = SPACE.
      IF STRAS = SPACE.
        FORMAT INTENSIFIED.
      ENDIF.
    ENDIF.
  ELSE.
    FORMAT INTENSIFIED OFF.
  ENDIF.

  WRITE: KTNRA, PSTLZ, ORT01, STRAS.
```

Programm N-UM-003 (RFSCHU22)

```
REPORT RFSCHU22.
SEGMENTS: BK, BP, *SA, *DA, *KA.
DATA: BEZEICHNUNG(40).

START-OF-SELECTION.
  WRITE 'Selektionsstart'.

GET BK.
  WRITE: BUKRS, BELNR,
         BUDAT DD/MM/YY, BLDAT DD/MM/YY, WAERS.

GET BP.
  CASE KOART.
    WHEN 'D'.
      MOVE KUNNR TO *DA-KTNRA.
      READ SEGMENT *DA.
      MOVE *DA-NAME1 TO BEZEICHNUNG.
    WHEN 'S'.
      MOVE HKONT TO *SA-KTNRA.
      READ SEGMENT *SA.
      MOVE *SA-SKBEZ TO BEZEICHNUNG.
    WHEN OTHERS.
      MOVE SPACE TO BEZEICHNUNG.
  ENDCASE.

  DETAIL.
  WRITE: (3) BUZEI, BSCHL, UMSKZ, GSBER,
         BEZEICHNUNG, DMSHB CURRENCY BK-WAERS.
```

Programm N-UM-004 (RFSCHU15)

```
REPORT RFSCHU15.
SEGMENTS: KA, KB, KC.
DATA: BEGIN OF UTAB OCCURS 100,
        LAND1,
        KUMUM,
      END OF UTAB.

GET KA.
  FILL UTAB.
GET KC.
  FILL UTAB.
  COLLECT UTAB.
END-OF-SELECTION.
  SORT UTAB.
  LOOP AT UTAB.
    WRITE: / UTAB-LAND1, UTAB-KUMUM.
  ENDLOOP.
  ULINE.
```

Programm N-UM-005 (RFSCHU19)

```
REPORT RFSCHU19.
SEGMENTS: KA, KB, KC.
DATA: BEGIN OF RANGLISTE OCCURS 50,
        KTNRA,
        UMSATZ TYPE P,
      END OF RANGLISTE.

GET KA.
  RANGLISTE-KTNRA = KA-KTNRA.
GET KC.
  RANGLISTE-UMSATZ = KC-KUMUM / 100000 * -1.
  APPEND RANGLISTE SORTED BY UMSATZ.
END-OF-SELECTION.
  LOOP AT RANGLISTE.
    MOVE RANGLISTE-KTNRA TO KA-KTNRA.
    READ SEGMENT KA.
    WRITE: (4) SY-TABIX NO-SIGN, KA-NAME1,
                KA-ORT01, RANGLISTE-UMSATZ.
  ENDLOOP.
```

Programm N-UM-006 (RFSCHU26)

```
REPORT RFSCHU26.

SEGMENTS: KA, KB, KC.
TABLES: T001.
DATA: SKZ LIKE KC-SKZ01, SBL LIKE KC-SBL01.

GET KC.
  NEW-PAGE.
  WRITE: KA-KTNRA USING NO EDIT MASK, KA-NAME1.
  SELECT SINGLE * FROM T001
         WHERE BUKRS = KB-BUKRB.
  WRITE: 50 T001-TEXT, 76 KC-JAHRC,
        /10 KA-NAME2,
        /10 KA-ORT01.
  ULINE.
  FORMAT INTENSIFIED OFF.
  WHILE SKZ NE SPACE
        VARY SKZ FROM SKZ01 NEXT SKZ02
        VARY SBL FROM SBL01 NEXT SBL02.
    WRITE: / SKZ, SBL.
  ENDWHILE.
  FORMAT INTENSIFIED ON.
  ULINE.
```

Programm N-UM-007 (RFSCHU33*)

```
REPORT RFSCHU33 LINE-SIZE 120.

TABLES: TRDIR.

DATA: LINENR TYPE P, TITEL(70).
DATA: NAME1 LIKE TRDIR-NAME,
      NAME2 LIKE TRDIR-NAME.
DATA: BEGIN OF TEXT/TAB OCCURS 50,
        ID, NR(8), TEXT(132),
      END OF TEXT/TAB.

PARAMETERS: REPORT LIKE TRDIR-NAME
                   DEFAULT 'RFSCHU'.
```

Fortsetzung Programm N-UM-007 (RFSCHU33*)

```
MOVE REPORT TO NAME1.
TRANSLATE NAME1 USING '* '.
MOVE NAME1 TO NAME2.
TRANSLATE NAME2 USING ' 9'.

SELECT * FROM TRDIR
              WHERE NAME BETWEEN NAME1 AND NAME2.
  REFRESH TEXT/TAB.
  READ TEXTPOOL TRDIR-NAME LANGUAGE SY-LANGU
                           INTO TEXT/TAB.
  TITEL = SPACE.
  CLEAR TEXT/TAB.
  READ TABLE TEXT/TAB WITH KEY 'R'.
  TITEL = TEXT/TAB-TEXT.
  WRITE:    TRDIR-NAME, (50) TITEL, TRDIR-SUBC,
            TRDIR-RSTAT,TRDIR-DBNA, TRDIR-CLAS,
            TRDIR-APPL,
         83 TRDIR-CNAM, TRDIR-UNAM,TRDIR-NAME.
ENDSELECT.
SET PF-STATUS '0001'.

AT PF13.
  SET PF-STATUS '0002'.
  WINDOW STARTING AT 54 3
           ENDING AT 79 22.
  WRITE: 'erstellt geaendert'(001).
  ULINE.
  DO 18 TIMES.
    LINENR = SY-INDEX + 4.
    READ LINE LINENR OF CURRENT PAGE.
    WRITE: / SY-LISEL+82(12), SY-LISEL+95(12).
  ENDDO.

AT PF14.
  SET PF-STATUS '0002'.
  SY-SUBRC = 0.
  WHILE SY-SUBRC = 0.
    READ LINE SY-INDEX INDEX 0.
    WRITE: / SY-LISEL(53), SY-LISEL+82(12),
             SY-LISEL+95(12).
  ENDWHILE.
```

Programm N-UM-008 (RFSCHU25)

```
REPORT RFSCHU25.
SEGMENTS: KA, KB, KC.
TABLES: T001.

DATA: MSALDO LIKE KC-UMSAV,
      PSALDO LIKE KC-UMSAV,
      UMS LIKE KC-UM01S,
      UMH LIKE KC-UM01H.

GET KC.
  NEW-PAGE.
  WRITE: KA-KTNRA USING NO EDIT MASK,
         KA-NAME1.

  SELECT SINGLE * FROM T001 WHERE BUKRS = KB-BUKRB.

  WRITE: 48 T001-TEXT, KC-JAHRC,
         /10 KA-NAME2,
         /10 KA-ORT01.
  ULINE.
  PSALDO = UMSAV.
  FORMAT INTENSIFIED OFF.

  DO 12 TIMES
        VARYING UMS FROM UM01S NEXT UM02S
        VARYING UMH FROM UM01H NEXT UM02H.
    MSALDO = UMS - UMH.
    PSALDO = PSALDO + MSALDO.
    WRITE: /01(3) SY-INDEX,
            08    UMS CURRENCY LOCAL,
            27    UMH CURRENCY LOCAL,
            45    MSALDO CURRENCY LOCAL,
            64    PSALDO CURRENCY LOCAL.
  ENDDO.

  FORMAT INTENSIFIED ON.
  ULINE.
```

Programm N-UM-009 (RFSCHU27*)

```
REPORT RFSCHU27.

SEGMENTS: KA, KB, KC.

DATA: BEGIN OF COMMON PART,
        UMSATZ TYPE P,
      END OF COMMON PART.

GET KB LATE.
  WRITE: KA-KTNRA USING NO EDIT MASK,
         KB-BUKRB,
         (25) KA-NAME1,
         (20) KA-ORT01,
         UMSATZ CURRENCY KA-WAERS.
  HIDE: KA-KTNRA,
        KB-BUKRB,
        UMSATZ.
  UMSATZ = 0.

GET KC.
  UMSATZ = UMSATZ + KC-KUMUM.

AT LINE-SELECTION.
  READ SEGMENT KA.
  HF-KOART = 'K'.
  HF-LIFNR = KA-KTNRA.
  HF-KBUKR = KB-BUKRB.
  FORMAT INTENSIFIED OFF.
  FOR ALL HF.
    WRITE:    HF-ZUONR,
           15 HF-BELNR,
           25 HF-GSBER,
           30 HF-BLART,
           35 HF-BUDAT,
           45 HF-BUZEI,
           55 HF-WAERS,
           65 HF-DMSHB.
  ENDFOR.
```

Programm N-UM-010 (RFSCHU37)

```
REPORT RFSCHU37.

DATA: BEGIN OF SATZ,
        TELF1 LIKE KA-TELF1, SPACE,
        KTNRA LIKE KA-KTNRA, SPACE,
        NAME1 LIKE KA-NAME1,
      END OF SATZ.

PARAMETERS: MAPPE(12) DEFAULT SY-UNAME.

DYNPROS: D00001, D02041, D02060.

MOVE: SY-MANDT TO D00001-MANDT,
      MAPPE    TO D00001-BNAME,
      SY-UZEIT TO D00001-BNAME+6,
      'PASS'   TO D00001-BCODE,
      SY-LANGU TO D00001-SPRAS.
TRANSFER-DYNPRO D00001.

UPLOAD SATZ.
  MOVE: 'S'        TO D02041-SP,
        'TS02'     TO D02041-TC,
        SATZ-KTNRA TO D02041-KTNRA,
        '01'       TO D02041-VCODE.
  TRANSFER-DYNPRO D02041.
  MOVE SATZ-TELF1 TO D02060-TELF1.
  TRANSFER-DYNPRO D02060.
  WRITE: / SATZ.
ENDUPLOAD.
```

Programm N-UM-011 (RMSCHU41)

```
REPORT RMSCHU41.

SEGMENTS: MA,
          MB,
          BV.

DATA:
      BEGIN OF EINZEL-SATZ,
        LENGTH(4),
        MATNR,
        BUKRS,
        WERKS,
        BANFN,
        LFDAT,
        PSTYP,
        BAMNG,
      END OF EINZEL-SATZ,
      BEGIN OF SUMMEN-SATZ OCCURS 1000,
        LENGTH(4),
        MATNR,
        LFDAT,
        BAMNG,
        END OF SUMMEN-SATZ.

START-OF-SELECTION.
  UPLOAD EINZEL-SATZ FROM FILE 1.
    WRITE:
           /(18) EINZEL-SATZ-MATNR,
            EINZEL-SATZ-BANFN,
            EINZEL-SATZ-LFDAT DD/MM/YY,
            EINZEL-SATZ-PSTYP,
            EINZEL-SATZ-BAMNG.
  ENDUPLOAD.
  UPLOAD SUMMEN-SATZ FROM FILE 2.
    WRITE:
           /(18) SUMMEN-SATZ-MATNR,
            SUMMEN-SATZ-LFDAT DD/MM/YY UNDER LFDAT,
            SUMMEN-SATZ-BAMNG UNDER BAMNG.
  ENDUPLOAD.
```

Programm N-UM-012 (RFSCHU20*)

```
REPORT RFSCHU20.
SEGMENTS: KA, KB, KC.
DATA: BEGIN OF UTAB OCCURS 200,
            LAND1,
            KUMUM,
      END OF UTAB,
      DM/KUMUM LIKE KC-KUMUM,
      UMSATZ TYPE P,
      PROZENT TYPE F.

GET KA.
  FILL UTAB.

GET KC.
  CURRENCY-CONVERSION KUMUM SY-WAERS TO DM/KUMUM 'DM'.
  UTAB-KUMUM = DM/KUMUM.
  COLLECT UTAB.

END-OF-SELECTION.
  SORT UTAB.
  LOOP AT UTAB.
    AT FIRST.
      SUM.
      UMSATZ = UTAB-KUMUM.
      PROZENT = UTAB-KUMUM * 100 / UMSATZ.
      WRITE: /(10) UTAB-LAND1,
              (20) UTAB-KUMUM, PROZENT DECIMALS 5.
    ENDAT.
    AT NEW LAND1.
      SUMMARY.
      SUM.
      PROZENT = UTAB-KUMUM * 100 / UMSATZ.
       WRITE: /(10) UTAB-LAND1,
               (20) UTAB-KUMUM, PROZENT DECIMALS 5.
    ENDAT.
    PROZENT = UTAB-KUMUM * 100 / UMSATZ.
    WRITE: /(10) UTAB-LAND1,
            (20) UTAB-KUMUM, PROZENT DECIMALS 5.
    AT END OF LAND1.  ULINE. ENDAT.
  ENDLOOP.
```

Programm N-UM-013 (RFSCHU23)

```
REPORT RFSCHU23.
SEGMENTS: SA, SB, SC.
TABLES: T074, T074T, T001.
DATA: BEGIN OF TAB OCCURS 100,
          MAFID(1), UMSKZ(1), SKONT(8),
          SALDO TYPE P DECIMALS 2,
      END OF TAB,
      SOLL TYPE P, HABEN TYPE P.

PARAMETERS: BUKREIS  LIKE SB-BUKRB DEFAULT '01',
            GJAHR    LIKE SC-JAHRC DEFAULT '1990',
            GSBER    LIKE SC-GSBER DEFAULT '00'.

SELECT SINGLE * FROM T001 WHERE BUKRS = BUKREIS.

SELECT * FROM T074 WHERE BUMOD = T001-MODIF.
  MOVE-CORRESPONDING T074 TO TAB.
  APPEND TAB.
ENDSELECT.

SORT TAB.
MOVE SY-LANGU TO T074T-SPRAS.

LOOP AT TAB.
  MOVE: TAB-SKONT TO SC-KTNRC,
        BUKREIS   TO SC-BUKRC,
        GJAHR     TO SC-JAHRC,
        GSBER     TO SC-GSBER.
  READ SEGMENT SC.
  IF SY-SUBRC EQ 0.
    ADD SC-UM01S THEN SC-UM02S
                 UNTIL SC-UM16S GIVING SOLL.
    ADD SC-UM01H THEN SC-UM02H
                 UNTIL SC-UM16H GIVING HABEN.
    TAB-SALDO = SC-UMSAV + SOLL - HABEN.
    MODIFY TAB.
  ENDIF.
ENDLOOP.
```

Fortsetzung Programm N-UM-013 (RFSCHU23)

```
LOOP AT TAB.
  AT NEW UMSKZ.
    SUM.
    SUMMARY.
    SKIP.
    MOVE TAB-MAFID TO T074T-MAFID.
    MOVE TAB-UMSKZ TO T074T-UMSKZ.
    READ TABLE T074T.
    WRITE: /01 T074T-LTEXT, 60 TAB-SALDO, '*'.
  ENDAT.
  DETAIL.
  MOVE TAB-SKONT TO SA-KTNRA.
  READ SEGMENT SA.
  WRITE: / SA-KTNRA, SA-SKBEZ, 60 TAB-SALDO.
ENDLOOP.
```

Programm N-UM-014 (RFSCHU65*)

```
REPORT RFSCHU65.
TABLES: TRDIR.

DATA: TITLE LIKE SY-TITLE.
      BEGIN OF TEXT/TAB OCCURS 50.
               ID, NR(8), TEXT(132).
      END OF TEXT/TAB.

PARAMETERS: SORTIER(6).
FIELD-GROUPS: HEADER, ENTRY.

WHILE SORTIER NE SPACE.
  CASE SORTIER(1).
     WHEN '1'.  INSERT TRDIR-APPL  INTO HEADER.
     WHEN '2'.  INSERT TRDIR-SUBC  INTO HEADER.
     WHEN '3'.  INSERT TRDIR-RSTAT INTO HEADER.
     WHEN '4'.  INSERT TRDIR-DBNA  INTO HEADER.
     WHEN '5'.  INSERT TRDIR-CLAS  INTO HEADER.
     WHEN '6'.  INSERT TRDIR-CNAM  INTO HEADER.
  ENDCASE.
  SHIFT SORTIER.
ENDWHILE.

INSERT TRDIR-NAME  INTO HEADER.
INSERT TRDIR       INTO ENTRY.

SELECT * FROM TRDIR.  EXTRACT ENTRY.
ENDSELECT.

END-OF-SELECTION.
  SORT.
  LOOP.
    REFRESH TEXT/TAB.
    READ TEXTPOOL TRDIR-NAME LANGUAGE SY-LANGU
         INTO TEXT/TAB.
    MOVE SPACE TO TITLE.  CLEAR TEXT/TAB.
    READ TABLE TEXT/TAB WITH KEY 'R'.
    TITLE = TEXT/TAB-TEXT.
    WRITE: TRDIR-NAME, (40) TITLE, TRDIR-APPL,
           TRDIR-CLAS, TRDIR-RSTAT, TRDIR-SUBC,
           TRDIR-DBNA, TRDIR-CNAM.
  ENDLOOP.
```

Programm N-UM-015 (RFSCHU42*)

```
REPORT RFSCHU42.
SEGMENTS: LH.

DATA: BEGIN OF SATZ,
         RDW(4), TABLE(4), MODID(1), DATUM TYPE D,
         USER LIKE LH-BNAME, NEW(80), OLD(80),
      END OF SATZ.

DATA: TABNAME(4),
      BEGIN OF STAT OCCURS 500,
        TABLE(4), ATTACH(3) TYPE P, INSERT(3) TYPE P,
        UPDATE(3) TYPE P, DELETE(3) TYPE P,
      END OF STAT.

PARAMETERS: NAME(6) DEFAULT 'SAPV01'.

DO.
  READ DATASET NAME INTO SATZ.
*  IF SY-SUBRC NE 0.  EXIT. ENDIF.
  MOVE SATZ-TABLE TO STAT-TABLE.
  STAT-INSERT = STAT-UPDATE = STAT-DELETE = 0.
  MOVE 1 TO STAT-ATTACH.

  CASE SATZ-MODID.
    WHEN 'I'. MOVE 1 TO STAT-INSERT.
    WHEN 'U'. MOVE 1 TO STAT-UPDATE.
    WHEN 'D'. MOVE 1 TO STAT-DELETE.
  ENDCASE.
  COLLECT STAT.

  IF TABNAME NE SATZ-TABLE.
    SKIP 2.
    MOVE SATZ-TABLE TO TABNAME.
    RESERVE 5 LINES.
    WRITE: 'Tabelle'(001), SATZ-TABLE.
    SKIP.
  ENDIF.
```

Fortsetzung Programm N-UM-015 (RFSCHU42*)

```
  IF SATZ-MODID = 'D'.
    MOVE SATZ-OLD TO SATZ-NEW.
    MOVE SPACE TO SATZ-OLD.
  ENDIF.

  WRITE:  SATZ-DATUM DD/MM/YY,
          SATZ-USER,
          SATZ-MODID,
        / SATZ-NEW,
        / SATZ-OLD.
ENDDO.

TABNAME = SPACE.
NEW-PAGE NO-HEADING.
WRITE 'Zugriffsstatistik'(002).
WRITE 'Tabelle  Zugriffe(gesamt)    Insert'
      'Update      Delete'(003).

SORT STAT.

LOOP AT STAT.
  NEW-LINE.
  WRITE:    STAT-TABLE,
         13 STAT-ATTACH,
         30 STAT-INSERT,
         42 STAT-UPDATE,
         52 STAT-DELETE.
ENDLOOP.

TOP-OF-PAGE.
  IF TABNAME NE SPACE.
    WRITE: 'Tabelle'(001), SATZ-TABLE.
  ENDIF.
```

Programm N-UM-016 (RFSCHU72*)

```
REPORT RFSCHU72.
SEGMENTS: KA.

TABLES: INDX.

DATA: BEGIN OF RANG OCCURS 10,
        KTNRA, NAME1, UMSATZ TYPE P DECIMALS 2,
      END OF RANG,

      BEGIN OF KEY,
        DBNAME(4),INDEXNAME(18),
      END OF KEY.

MOVE 'LIFA' TO KEY-DBNAME.
MOVE 'RANGLISTE' TO KEY-INDEXNAME.

IMPORT RANG
       FROM DATABASE INDX(IX) ID KEY.

IF SY-SUBRC NE 0.
  WRITE: 'Falscher Index'.
ENDIF.

LOOP AT RANG.
  WRITE: /(4) SY-TABIX NO-SIGN.
  WRITE: RANG-KTNRA, RANG-NAME1, RANG-UMSATZ.
ENDLOOP.
SKIP.
ULINE.
WRITE: 'Erstellungsdatum:'(001),
       INDX-AEDAT DD/MM/YY,
     / 'Ersteller:'(002),
       INDX-USERA UNDER INDX-AEDAT.
```

Programm N-UM-017 (RFSCHU38)

```
REPORT RFSCHU38.

DATA: BEGIN OF T OCCURS 200,
         FORMAT, TEXT(60),
      END OF T.
DATA: BEGIN OF O OCCURS 200,
         FORMAT, TEXT(60),
      END OF O.
DATA: BEGIN OF DDNAME,
         'SD', 'ABAP', KEYW(30),
      END OF DDNAME.
DATA: I TYPE P, MAX TYPE P.
DATA: KEYW LIKE DDNAME-KEYW.

SKIP.
WRITE: 13 'Schluesselwort:'(001), KEYW INPUT.

AT LINE-SELECTION.
  READ LINE 5 FIELD VALUE KEYW.
  TRANSLATE KEYW TO UPPER CASE.
  DDNAME-KEYW = KEYW.
  REFRESH: T, O.
  READ DOCUMENTATION DDNAME LANGUAGE SY-LANGU INTO O.
  READ DOCUMENTATION DDNAME LANGUAGE SY-LANGU INTO T.
  EDITOR-CALL FOR T.
  I = MAX = 0.
  LOOP AT T.
    I = I + 1.
    IF MAX = 0.
      READ TABLE O INDEX I.
      IF O-TEXT NE T-TEXT.
        MAX = I.
      ENDIF.
    ENDIF.
    WRITE / T-TEXT.
  ENDLOOP.
  IF MAX = 0.
    WRITE / 'Tabelle unveraendert'(002).
  ELSE.
    WRITE: / 'Tabelle ab Eintrag'(003),
             MAX, 'geaendert'(004).
  ENDIF.
```

Programm N-UM-018 (RFSCHU63)

```
REPORT RFSCHU63.

SEGMENTS: KA, KB, HF.
TABLES: T001.

FIELD-GROUPS: HEADER,
              ANSCHRIFT,
              POSTEN.

INSERT: KB-BUKRB KA-KTNRA HF-GSBER HF-BUDAT HF-BELNR
        INTO HEADER,
        KA-NAME1 KA-NAME2 KA-PLORT KA-STRAS
        INTO ANSCHRIFT,
        HF-ZUONR HF-DMSHB HF-BLART HF-ZALDT
        HF-BSCHL HF-UMSKZ
        INTO POSTEN.

GET KB.
  EXTRACT ANSCHRIFT.

GET HF.
  EXTRACT POSTEN.

END-OF-SELECTION.
  SORT.
  LOOP.
    AT NEW BUKRB.
      NEW-PAGE.
    ENDAT.

    AT ANSCHRIFT WITH POSTEN.
      FORMAT INTENSIFIED ON.
      SKIP.
      RESERVE 4 LINES.
      WRITE: / KA-KTNRA USING NO EDIT MASK, KA-NAME1,
             / KA-NAME2 UNDER KA-NAME1,
             / KA-STRAS UNDER KA-NAME1,
             / KA-PLORT UNDER KA-NAME1.
    ENDAT.
```

Fortsetzung Programm N-UM-018 (RFSCHU63)

```
    AT POSTEN.
      DETAIL.
      WRITE: /13 HF-GSBER, HF-BUDAT DD/MM/YY, HF-BELNR,
                 HF-ZUONR, HF-BLART, HF-ZALDT DD/MM/YY,
                 HF-BSCHL, HF-UMSKZ, HF-DMSHB.
    ENDAT.

    AT END OF HF-GSBER.
      IF CNT(HF-BELNR) > 1.
        FORMAT INTENSIFIED ON.
        WRITE: / 'Summe Geschaeftsbereich'(001),
                 HF-GSBER,
                 SUM(HF-DMSHB) UNDER HF-DMSHB, '*'.
        SKIP.
      ENDIF.
    ENDAT.

    AT END OF KA-KTNRA.
      IF CNT(HF-GSBER) > 0.
        FORMAT INTENSIFIED ON.
        WRITE: / 'Summe'(002), KA-KTNRA, KA-NAME1,
                  SUM(HF-DMSHB) UNDER HF-DMSHB, '**'.
      ENDIF.
    ENDAT.

    AT END OF KB-BUKRB.
      IF CNT(KA-KTNRA) > 0.
        FORMAT INTENSIFIED OFF.
        SELECT SINGLE * FROM T001
                            WHERE BUKRS = KB-BUKRB.
        WRITE: / 'Summe'(002), T001-TEXT,
                  SUM(HF-DMSHB) UNDER HF-DMSHB, '***'.
      ENDIF.
    ENDAT.

    AT LAST.
      FORMAT INTENSIFIED ON.
      SKIP.
      WRITE: / 'Insgesamt:'(003),
                SUM(HF-DMSHB) UNDER HF-DMSHB, '<=='.
    ENDAT.
  ENDLOOP.
```

Programm N-UM-019 (RFSCHU71)

```
REPORT RFSCHU71.
SEGMENTS: KA, KC.
TABLES: INDX.
DATA: BEGIN OF RANG OCCURS 10,
        KTNRA, NAME1, UMSATZ TYPE P DECIMALS 2,
      END OF RANG,
      BEGIN OF KEY,
        DBNAME(4),INDEXNAME(18),
      END OF KEY.

START-OF-SELECTION.
  MOVE: SY-DATUM TO INDX-AEDAT,
        SY-UNAME TO INDX-USERA,
        SY-REPID TO INDX-PGMID.

GET KA.
  FILL RANG.
  RANG-UMSATZ = 0.

GET KC.
  KUMUM = KUMUM / 10000 * -1.
  RANG-UMSATZ = RANG-UMSATZ + KUMUM.

GET KA LATE.
  APPEND RANG SORTED BY UMSATZ.

END-OF-SELECTION.
  LOOP AT RANG.
    FORMAT INTENSIFIED ON.
    WRITE: /(4) SY-TABIX NO-SIGN.
    FORMAT INTENSIFIED OFF.
    WRITE: RANG-KTNRA, RANG-NAME1, RANG-UMSATZ.
  ENDLOOP.

  WRITE:   'Erstellungsdatum:'(001),
           INDX-AEDAT DD/MM/YY,
         / 'Ersteller:'(002),
           INDX-USERA UNDER INDX-AEDAT.
  MOVE 'LIFA' TO KEY-DBNAME.
  MOVE 'RANGLISTE' TO KEY-INDEXNAME.
  EXPORT RANG TO DATABASE INDX(IX) ID KEY.
```

Programm N-UM-020 (RFSCHU60*)

```
REPORT RFSCHU60 NO STANDARD PAGE HEADING.
SEGMENTS: KA, KB, HF.
TABLES: T001.
FIELD-GROUPS: HEADER, ANSCHRIFT, POSTEN.

INSERT KA-KTNRA KB-BUKRB HF-BELNR HF-BUZEI
    HF-DMSHB HF-WAERS INTO HEADER.
INSERT KA-NAME1 KA-NAME2 KA-ORT01 KA-LAND1
    INTO ANSCHRIFT.
INSERT HF-ZUONR HF-BLART HF-WRSHB HF-BUDAT
    INTO POSTEN.

GET KB.
  EXTRACT ANSCHRIFT.

GET HF.
  EXTRACT POSTEN.
  WRITE:     HF-ZUONR, HF-BLART, HF-BELNR,
         (3) HF-BUZEI,
             HF-BUDAT, HF-DMSHB,
             HF-WRSHB NO-ZERO CURRENCY HF-WAERS,
             HF-WAERS.

END-OF-SELECTION.
  SET PF-STATUS '0001'.

AT PF13.
  SY-LSIND = 1.
  SORT BY KA-KTNRA KB-BUKRB HF-BELNR HF-BUZEI.
  LOOP.
    AT ANSCHRIFT.
      NEW-PAGE.
    ENDAT.
    AT POSTEN.
      WRITE: HF-ZUONR, HF-BLART, HF-BELNR,
             (3) HF-BUZEI,
             HF-BUDAT, HF-DMSHB,
             HF-WRSHB NO-ZERO CURRENCY HF-WAERS,
             HF-WAERS.
    ENDAT.
  ENDLOOP.
```

Fortsetzung Programm N-UM-020 (RFSCHU60*)

```
AT PF15.
  SY-LSIND = 1.
  SORT BY KA-KTNRA KB-BUKRB HF-WAERS.
  LOOP.
    AT ANSCHRIFT.
      NEW-PAGE.
    ENDAT.
    AT POSTEN.
      WRITE: HF-ZUONR, HF-BLART, HF-BELNR,
             (3) HF-BUZEI,
             HF-BUDAT, HF-DMSHB,
             HF-WRSHB NO-ZERO CURRENCY HF-WAERS,
             HF-WAERS.
    ENDAT.
  ENDLOOP.
TOP-OF-PAGE.
  FORMAT INTENSIFIED ON.
  SELECT SINGLE * FROM T001 WHERE BUKRS = KB-BUKRB.
  WRITE: KA-KTNRA USING NO EDIT MASK,
         KA-NAME1, 41 T001-TEXT,
         68 'Seite:'(001), (4) SY-PAGNO NO-SIGN,
       / KA-NAME2 UNDER KA-NAME1,
         41 'Buchungskreis'(002), KB-BUKRB,
       / KA-ORT01 UNDER KA-NAME1.
  SKIP.
  WRITE: ' Zuordnung BA BelegNr BZ Budat'(003),
      45 'Betrag FWBetrag Waehrung'(004).
  ULINE /(68).   SKIP.
TOP-OF-PAGE DURING LINE-SELECTION.
  FORMAT INTENSIFIED ON.
  SELECT SINGLE * FROM T001 WHERE BUKRS = KB-BUKRB.
  WRITE: KA-KTNRA USING NO EDIT MASK,
         KA-NAME1, 41 T001-TEXT,
         68 'Seite:'(001), (4) SY-PAGNO NO-SIGN,
       / KA-NAME2 UNDER KA-NAME1,
         41 'Buchungskreis'(002), KB-BUKRB,
       / KA-ORT01 UNDER KA-NAME1.
  SKIP.
  WRITE: ' Zuordnung BA BelegNr BZ Budat'(003),
      45 'Betrag FWBetrag Waehrung'(004).
  ULINE /(68).   SKIP.
```

Programm N-UM-21 (RFSCHU32*)

```
REPORT RFSCHU32 MESSAGE-ID 39.

SEGMENTS: KA.
DATA: Y1 LIKE SY-CUROW, Y2 LIKE SY-CUROW.
DATA: MARKFIELD VALUE '_', FLAG.
DATA: FIELDNAME(8).
DATA: BEGIN OF INFO,
        TEXT(10),
        KTNRA,
        SPACE,
        NAME1,
        END OF INFO.

GET KA.
  FORMAT INTENSIFIED OFF.
  WRITE: MARKFIELD INPUT, FLAG, KA-KTNRA USING NO EDIT MASK,
         KA-NAME1, KA-BANKN, KA-BANKL.
  HIDE:  KA-KTNRA,KA-NAME1,
         KA-BANKN, KA-BANKL, KA-BANKA, KA-BANKS, KA-PSKTO.

END-OF-SELECTION.
  SET PF-STATUS '0001'.

AT PF2.
  SET PF-STATUS '0002'.
  READ SEGMENT KA.
  GET CURSOR FIELD FIELDNAME.
  CASE FIELDNAME.
    WHEN 'KA-NAME1'.
      MOVE-CORRESPONDING KA TO INFO.
      INFO-TEXT = 'Lieferant'(011).
      Y1 = SY-CUROW.
      IF Y1 LT 14.
        Y1 = 15.
      ELSE.
        Y1 = 04.
      ENDIF.
      Y2 = Y1 + 07.
      WINDOW STARTING AT 01 Y1
               ENDING AT 79 Y2
                 WITH FRAME
                TITLE INFO.
```

Fortsetzung Programm N-UM-21 (RFSCHU32*)

```
    WRITE:
           / 'Anrede..........'(013) INTENSIFIED OFF,
             KA-ANRED INTENSIFIED,
           / 'Name 1..........'(015) INTENSIFIED OFF,
             KA-NAME1 INTENSIFIED,
           / 'Name 2..........'(016) INTENSIFIED OFF,
             KA-NAME2 INTENSIFIED,
           / 'Strasse.........'(017) INTENSIFIED OFF,
             KA-STRAS INTENSIFIED,
           / 'Postfach........'(018) INTENSIFIED OFF,
             KA-PFACH INTENSIFIED,
           / 'Ort.............'(019) INTENSIFIED OFF,
             KA-ORT01 INTENSIFIED,
           / 'Land............'(020) INTENSIFIED OFF,
             KA-LAND1 INTENSIFIED.
  WHEN 'KA-BANKN'.
    MOVE-CORRESPONDING KA TO INFO.
    INFO-TEXT = 'Lieferant'(011).
    Y1 = SY-CUROW.
    IF Y1 LT 14.
      Y1 = 15.
    ELSE.
      Y1 = 04.
    ENDIF.
    Y2 = Y1 + 07.
    WINDOW STARTING AT 01 Y1
             ENDING AT 79 Y2
               WITH FRAME
              TITLE INFO.
    FORMAT INTENSIFIED OFF.
    WRITE:
          / 'Bankkonto.......'(042), KA-BANKN,
          / 'Bankleitzahl....'(044), KA-BANKL,
          / 'Bankanschrift...'(046), KA-BANKA,
          / 'Bankland........'(047), KA-BANKS,
          / 'Postscheckkonto.'(048), KA-PSKTO.
  WHEN 'KA-BANKL'.
    MOVE-CORRESPONDING KA TO INFO.
    INFO-TEXT = 'Lieferant'(011).
    Y1 = SY-CUROW.
    IF Y1 LT 14.
      Y1 = 15.
```

Fortsetzung Programm N-UM-21 (RFSCHU32*)

```
      ELSE.
        Y1 = 04.
      ENDIF.
      Y2 = Y1 + 07.
      WINDOW STARTING AT 01 Y1
               ENDING AT 79 Y2
                 WITH FRAME
                TITLE INFO.
      FORMAT INTENSIFIED OFF.
      WRITE:
            / 'Bankkonto.......'(042), KA-BANKN,
            / 'Bankleitzahl....'(044), KA-BANKL,
            / 'Bankanschrift...'(046), KA-BANKA,
            / 'Bankland........'(047), KA-BANKS,
            / 'Postscheckkonto.'(048), KA-PSKTO.
    WHEN OTHERS.
      MESSAGE I850.
ENDCASE.

AT PF14.
  SET PF-STATUS '0003'.
  SY-SUBRC = 0.
  WHILE SY-SUBRC = 0.
    CLEAR MARKFIELD.
    READ LINE SY-INDEX FIELD VALUE MARKFIELD.
    IF MARKFIELD NE SPACE AND MARKFIELD NE '_'.
      FORMAT INTENSIFIED.
      WRITE: / 'Lieferant'(011), KA-KTNRA, KA-NAME1.
      SKIP.
      FORMAT INTENSIFIED OFF.
      WRITE:
            / 'Bankkonto.......'(042), KA-BANKN,
            / 'Bankleitzahl....'(044), KA-BANKL,
            / 'Bankanschrift...'(046), KA-BANKA,
            / 'Bankland........'(047), KA-BANKS,
            / 'Postscheckkonto.'(048), KA-PSKTO.
      ULINE.
```

Fortsetzung Programm N-UM-21 (RFSCHU32*)

```
      MODIFY CURRENT LINE
             LINE FORMAT  INTENSIFIED
             FIELD VALUE  MARKFIELD FROM SPACE
                          FLAG FROM '*'
             FIELD FORMAT KA-NAME1 UNDERSCORED.
    ENDIF.
  ENDWHILE.

TOP-OF-PAGE DURING LINE-SELECTION.
  IF SY-PFKEY = '0003'.
    FORMAT INTENSIFIED ON.
    WRITE: /25 'Bankdaten der Lieferanten'(004).
    ULINE.
    FORMAT INTENSIFIED OFF.
  ENDIF.
```

Programm N-M-001 (RFSCHU20)

```
REPORT RFSCHU20.
SEGMENTS: KA, KB, KC.
DATA: BEGIN OF UTAB OCCURS 200,
            LAND1,
            KUMUM,
      END OF UTAB,
      DM/KUMUM LIKE KC-KUMUM,
      UMSATZ TYPE P.   PROZENT TYPE F.

GET KA.
  FILL UTAB.

GET KC.
  CURRENCY-CONVERSION KUMUM SY-WAERS TO DM/KUMUM 'DM'.
  UTAB-KUMUM = DM/KUMUM.
  COLLECT UTAB.

END-OF-SELECTION.
  SORT UTAB.
  LOOP AT UTAB.
    AT FIRST.
      SUM.
      UMSATZ = UTAB-KUMUM.
      PERFORM LINE.
    ENDAT.

    AT NEW LAND1.
      SUMMARY.  SUM.
      PERFORM LINE.
    ENDAT.

    PERFORM LINE.

    AT END OF LAND1.  ULINE. ENDAT.
  ENDLOOP.

FORM LINE.
  PROZENT = UTAB-KUMUM * 100 / UMSATZ.
  WRITE: /(10) UTAB-LAND1,
          (20) UTAB-KUMUM, PROZENT DECIMALS 5.
ENDFORM.
```

Programm N-M-002 (RFSCHU60)

```
REPORT RFSCHU60 NO STANDARD PAGE HEADING.
SEGMENTS: KA, KB, HF.
TABLES: T001.
FIELD-GROUPS: HEADER, ANSCHRIFT, POSTEN.

INSERT KA-KTNRA KB-BUKRB HF-BELNR HF-BUZEI
    HF-DMSHB HF-WAERS INTO HEADER.
INSERT KA-NAME1 KA-NAME2 KA-ORT01 KA-LAND1
    INTO ANSCHRIFT.
INSERT HF-ZUONR HF-BLART HF-WRSHB HF-BUDAT
    INTO POSTEN.

GET KB.
  EXTRACT ANSCHRIFT.

GET HF.
  EXTRACT POSTEN.
  PERFORM AUSGABE-POSTEN.

END-OF-SELECTION.
  SET PF-STATUS '0001'.

AT PF13.
  SY-LSIND = 1.
  SORT BY KA-KTNRA KB-BUKRB HF-BELNR HF-BUZEI.
  PERFORM AUSGABE.

AT PF15.
  SY-LSIND = 1.
  SORT BY KA-KTNRA KB-BUKRB HF-WAERS.
  PERFORM AUSGABE.

TOP-OF-PAGE.
  PERFORM AUSGABE-ANSCHRIFT.

TOP-OF-PAGE DURING LINE-SELECTION.
  PERFORM AUSGABE-ANSCHRIFT.
```

Fortsetzung Programm N-M-002 (RFSCHU60)

```
*---------------------------------------------------*
*       FORM AUSGABE                                *
*---------------------------------------------------*
FORM AUSGABE.
  LOOP.
    AT ANSCHRIFT.
      NEW-PAGE.
    ENDAT.
    AT POSTEN.
      PERFORM AUSGABE-POSTEN.
    ENDAT.
  ENDLOOP.
ENDFORM.
*---------------------------------------------------*
*       FORM AUSGABE-ANSCHRIFT                      *
*---------------------------------------------------*
FORM AUSGABE-ANSCHRIFT.
  FORMAT INTENSIFIED ON.
  SELECT SINGLE * FROM T001 WHERE BUKRS = KB-BUKRB.
  WRITE:  KA-KTNRA USING NO EDIT MASK,
           KA-NAME1, 41 T001-TEXT,
           68 'Seite:'(001), (4) SY-PAGNO NO-SIGN,
         / KA-NAME2 UNDER KA-NAME1,
           41 'Buchungskreis'(002), KB-BUKRB,
         / KA-ORT01 UNDER KA-NAME1.
  SKIP.
  WRITE:    ' Zuordnung BA BelegNr BZ Budat'(003),
         45 'Betrag FWBetrag Waehrung'(004).
  ULINE /(68).    SKIP.
ENDFORM.
*--------------------------------------------------*
*       FORM AUSGABE-POSTEN                        *
*--------------------------------------------------*
FORM AUSGABE-POSTEN.
  WRITE:     HF-ZUONR, HF-BLART, HF-BELNR,
         (3) HF-BUZEI,
             HF-BUDAT, HF-DMSHB,
             HF-WRSHB NO-ZERO CURRENCY HF-WAERS,
             HF-WAERS.
ENDFORM.
```

Programm N-M-003 (RFSCHU32)

```
REPORT RFSCHU32 MESSAGE-ID 39.

SEGMENTS: KA.
DATA: Y1 LIKE SY-CUROW, Y2 LIKE SY-CUROW.
DATA: MARKFIELD VALUE '_', FLAG.
DATA: FIELDNAME(8).

GET KA.
  FORMAT INTENSIFIED OFF.
  WRITE: MARKFIELD INPUT, FLAG, KA-KTNRA USING NO EDIT MASK,
         KA-NAME1, KA-BANKN, KA-BANKL.
  HIDE:  KA-KTNRA,KA-NAME1,
         KA-BANKN, KA-BANKL, KA-BANKA, KA-BANKS, KA-PSKTO.

END-OF-SELECTION.
  SET PF-STATUS '0001'.

AT PF2.
  SET PF-STATUS '0002'.
  READ SEGMENT KA.
  GET CURSOR FIELD FIELDNAME.
  CASE FIELDNAME.
    WHEN 'KA-NAME1'.
      PERFORM SPLIT-SCREEN USING 'N'.
    WHEN 'KA-BANKN'.
      PERFORM SPLIT-SCREEN USING 'B'.
    WHEN 'KA-BANKL'.
      PERFORM SPLIT-SCREEN USING 'B'.
    WHEN OTHERS.
      MESSAGE I850.
  ENDCASE.

AT PF14.
  SET PF-STATUS '0003'.
  SY-SUBRC = 0.
  WHILE SY-SUBRC = 0.
    CLEAR MARKFIELD.
    READ LINE SY-INDEX FIELD VALUE MARKFIELD.
```

Fortsetzung Programm N-M-003 (RFSCHU32)

```
    IF MARKFIELD NE SPACE AND MARKFIELD NE '_'.
      FORMAT INTENSIFIED.
      WRITE: / 'Lieferant'(011), KA-KTNRA, KA-NAME1.
      SKIP.
      PERFORM BANK.
      ULINE.
      MODIFY CURRENT LINE
             LINE FORMAT  INTENSIFIED
             FIELD VALUE  MARKFIELD FROM SPACE
                          FLAG   FROM '*'
             FIELD FORMAT KA-NAME1 UNDERSCORED.
    ENDIF.
  ENDWHILE.

*-------- F o r mS p l i t - S c r e e n ---------*
FORM SPLIT-SCREEN USING FLAG.
  DATA: BEGIN OF INFO,
          TEXT(10),
          KTNRA,
          SPACE,
          NAME1,
        END OF INFO.

  MOVE-CORRESPONDING KA TO INFO.
  INFO-TEXT = 'Lieferant'(011).
  Y1 = SY-CUROW.
  IF Y1 LT 14.
    Y1 = 15.
  ELSE.
    Y1 = 04.
  ENDIF.
  Y2 = Y1 + 07.

  WINDOW  STARTING AT 01 Y1
          ENDING   AT 79 Y2
          WITH FRAME
          TITLE INFO.
  CASE FLAG.
    WHEN 'N'.
      PERFORM ADRESSE.
    WHEN 'B'.
```

Fortsetzung Programm N-M-003 (RFSCHU32)

```
          PERFORM BANK.
    ENDCASE.
ENDFORM.

TOP-OF-PAGE DURING LINE-SELECTION.
    IF SY-PFKEY = '0003'.
      FORMAT INTENSIFIED ON.
      WRITE: /25 'Bankdaten der Lieferanten'(004).
      ULINE.
      FORMAT INTENSIFIED OFF.
    ENDIF.

*------- F o r m A d r e s s e -------------*
FORM ADRESSE.
    WRITE:
        / 'Anrede..........'(013) INTENSIFIED OFF,
          KA-ANRED INTENSIFIED,
        / 'Name 1..........'(015) INTENSIFIED OFF,
          KA-NAME1 INTENSIFIED,
        / 'Name 2..........'(016) INTENSIFIED OFF,
          KA-NAME2 INTENSIFIED,
        / 'Strasse.........'(017) INTENSIFIED OFF,
          KA-STRAS INTENSIFIED,
        / 'Postfach........'(018) INTENSIFIED OFF,
          KA-PFACH INTENSIFIED,
        / 'Ort.............'(019) INTENSIFIED OFF,
          KA-ORT01 INTENSIFIED,
        / 'Land............'(020) INTENSIFIED OFF,
          KA-LAND1 INTENSIFIED.
ENDFORM.

*-------- F o r m B a n k --------------*
FORM BANK.
    FORMAT INTENSIFIED OFF.
    WRITE:
        / 'Bankkonto.......'(042), KA-BANKN,
        / 'Bankleitzahl....'(044), KA-BANKL,
        / 'Bankanschrift...'(046), KA-BANKA,
        / 'Bankland........'(047), KA-BANKS,
        / 'Postscheckkonto.'(048), KA-PSKTO.
ENDFORM.
```

Programm U-UM-001 (RFSCHU10)

```
REPORT RFSCHU10.
SEGMENTS: KA, KB, KC.

DATA: TDM-UMSATZ TYPE P.

SELECT-OPTIONS: TDM FOR TDM-UMSATZ.

GET KC.
  TDM-UMSATZ = KUMUM / 1000 / 100 * -1.
  CHECK TDM.
  FORMAT INTENSIFIED OFF.
  WRITE:     KA-KTNRA,
             KA-NAME1,
             KB-BUKRB,
             KC-JAHRC,
          60 TDM-UMSATZ.
```

Programm U-UM-002 (RFSCHU09)

```
REPORT RFSCHU09.
SEGMENTS: KA, KB, KC.

GET KC.
  CHECK: JAHRC = '1990', KUMUM < -100000.
  DETAIL.
  WRITE: KA-KTNRA, KA-NAME1, KB-BUKRB,
         JAHRC, 60 KUMUM.
```

Programm U-UM-003 (RFSCHU16)

```
REPORT RFSCHU16.
SEGMENTS: KA, BK, BP, GP, SA.

GET BK.
  WRITE: BK-BELNR, BK-BUDAT DD/MM/YY, BK-BLART.

GET BP.
  WRITE: /10 BP-LIFNR USING NO EDIT MASK,
             (40) KA-NAME1, BP-DMSHB.

GET GP.
  CHECK KOART = 'S'.
  MOVE HKONT TO SA-KTNRA.
  READ SEGMENT SA.
  DETAIL.
  WRITE: /10 GP-HKONT, (40) SA-SKBEZ, GP-DMSHB.
```

Programm U-UM-004 (RFSCHU06)

```
REPORT RMSCHU06.
SEGMENTS: MA, MC, MR.

GET MA.
  NEW-PAGE.

GET MR.
  DETAIL.
  WRITE:
             BUKRS,
             WERKS,
             RESNR,
             RSDAT DD/MM/YYYY,
             STATU,
             WKDAT DD/MM/YYYY,
             MENGE,
             ENMNG,
             MEINS.
  SKIP 1.
```

Fortsetzung Programm U-UM-004 (RFSCHU06)

```
GET MC LATE.
  CHECK RESME NE 0.
  NEW-PAGE NO-HEADING.
  WRITE:
          TEXT-009,
          WERKC,
        / TEXT-010,
          RESME,
          MEINS.
  NEW-PAGE WITH-HEADING.

TOP-OF-PAGE.
  WRITE:
         TEXT-008,
         (18) IDNRC.
  ULINE.

END-OF-PAGE.
  WRITE 'U E B E R L A U F Z E I L E'.
END-OF-SELECTION.
  NEW-PAGE NO-HEADING.
  WRITE:
           TEXT-001,
         / TEXT-002,
           MD/IDNR1,
         / TEXT-003,
           MD/IDNR2,
         / TEXT-004,
           MD/BUKR1,
         / TEXT-005,
           MD/BUKR2,
         / TEXT-006,
           MD/WERK1,
         / TEXT-007,
           MD/WERK2.
  NEW-PAGE.
```

Programm U-UM-005 (RMSCHU39)

```
REPORT RMSCHU39.

SEGMENTS: MA, MB, BV.

DATA:
      BEGIN OF EINZEL-SATZ,
        LENGTH(4),
        MATNR,
        BUKRS,
        WERKS,
        LFDAT,
        PSTYP,
        BAMNG,
      END OF EINZEL-SATZ,
      BEGIN OF SUMMEN-SATZ OCCURS 1000,
        LENGTH(4),
        MATNR,
        LFDAT,
        BAMNG,
      END OF SUMMEN-SATZ,
      SPOOLNR1 LIKE SY-SPONR,
      SPOOLNR2 LIKE SY-SPONR.

SELECT-OPTIONS:
  KONTI FOR PSTYP.

PARAMETERS:
  EXTRAKT1(6) TYPE C DEFAULT 'E-SATZ',
  EXTRAKT2(6) TYPE C DEFAULT 'S-SATZ'.

START-OF-SELECTION.
  SET PF-STATUS '0001'.

GET MA.
  CHECK SELECT-OPTIONS.

GET MB.
  CHECK SELECT-OPTIONS.
```

Fortsetzung Programm U-UM-005 (RMSCHU39)

```
GET BV.
  CHECK SELECT-OPTIONS.
  FILL EINZEL-SATZ.
  TRANSFER EINZEL-SATZ TO EXTRAKT1.
  FILL SUMMEN-SATZ.
  COLLECT SUMMEN-SATZ.
  WRITE:
        (18) MATNR,
             BANFN,
             LFDAT DD/MM/YY,
             PSTYP,
             BAMNG.

END-OF-SELECTION.
  MOVE SY-SPONR TO SPOOLNR1.
  NEW-PAGE.
  LOOP AT SUMMEN-SATZ.
    AT NEW LFDAT.
      TRANSFER SUMMEN-SATZ TO EXTRAKT2.
    ENDAT.
    WRITE:
          /(18) SUMMEN-SATZ-MATNR,
                SUMMEN-SATZ-LFDAT DD/MM/YY UNDER LFDAT,
                SUMMEN-SATZ-BAMNG UNDER BAMNG.
  ENDLOOP.
  MOVE SY-SPONR TO SPOOLNR2.

AT PF15.
  SET PF-STATUS '0002'.
  WRITE 'Extrakt-Protokoll Einzelstze'(001).
  DO.
    READ DATASET SPOOLNR1 INTO EINZEL-SATZ.
    IF SY-SUBRC NE 0.
      EXIT.
    ENDIF.
    WRITE:
          / SY-INDEX,
            EINZEL-SATZ.
  ENDDO.
  NEW-PAGE.
  WRITE 'Extrakt-Protokoll Summenstze'(002).
```

Fortsetzung Programm U-UM-005 (RMSCHU39)

```
DO.
  READ DATASET SPOOLNR2 INTO SUMMEN-SATZ.
  IF SY-SUBRC NE 0.
    EXIT.
  ENDIF.
  WRITE:
        / SY-INDEX,
          SUMMEN-SATZ.
ENDDO.
```

Programm U-UM-006 (RFSCHU35)

```
REPORT RFSCHU35.

TABLES: TRDIR.
DATA: BEGIN OF TEXT_TAB OCCURS 50,
        ID, NR(8), TEXT(132),
      END OF TEXT_TAB.
DATA: BEGIN OF CODE_TAB OCCURS 1000,
        TEXT(132),
      END OF CODE_TAB.
DATA: TITEL(70).
DATA: NAME1 LIKE TRDIR-NAME, NAME2 LIKE TRDIR-NAME.

PARAMETERS: REPORT LIKE TRDIR-NAME DEFAULT 'RFSCHU'.

* -- display list ------------------------------ *
MOVE REPORT TO NAME1.
TRANSLATE NAME1 USING '* '.
MOVE NAME1 TO NAME2.
TRANSLATE NAME2 USING ' 9'.

SELECT * FROM TRDIR
         WHERE NAME BETWEEN NAME1 AND NAME2.
  REFRESH TEXT_TAB.
  READ TEXTPOOL TRDIR-NAME LANGUAGE SY-LANGU
       INTO TEXT_TAB.
  MOVE SPACE TO TITEL.
  CLEAR TEXT_TAB.
  READ TABLE TEXT_TAB WITH KEY 'R'.
  MOVE TEXT_TAB-TEXT TO TITEL.
  WRITE: TRDIR-NAME, (68) TITEL.
ENDSELECT.

MOVE SPACE TO TRDIR-NAME.

AT LINE-SELECTION.
  CHECK TRDIR-NAME NE SPACE.
  READ REPORT TRDIR-NAME INTO CODE_TAB.
  NEW-PAGE PRINT ON LINE-COUNT 65.
```

Fortsetzung Programm U-UM-006 (RFSCHU35)

```
    LOOP AT CODE_TAB.
      WRITE _ CODE_TAB-TEXT.
    ENDLOOP.

    NEW-PAGE PRINT OFF.
    WINDOW STARTING AT 1 23
             ENDING AT79 23.
    WRITE: 'Report'(001), TRDIR-NAME, 'gedruckt.'(002).
    ADD 1 TO SY-CUROW.
    IF SY-CUROW >= 23.
      SY-CUROW = 24.
      SY-CUCOL = 5.
    ENDIF.
    SET CURSOR SY-CUCOL SY-CUROW.
    MOVE SPACE TO TRDIR-NAME.
```

Programm U-UM-007 (RFSCHU39)

```
REPORT RFSCHU39.
SEGMENTS: KA.
DATA:     BEGIN OF TAB OCCURS 5,
            DYNPRO(2000),
          END OF TAB,
          RETURN/CODE LIKE SY-SUBRC.
DYNPROS:  D02041, D02060.
TABLES:   T100.

GET KA.
  CHECK KA-LAND1 = 'D' AND KA-REGIO = SPACE.

  MOVE: KA-KTNRA  TO D02041-KTNRA,
        '01'      TO D02041-VCODE.
  MOVE D02041 TO TAB-DYNPRO.
  APPEND TAB.

  IF KA-PSTLZ(1) BETWEEN '1' AND '5'.
    MOVE: '01'    TO D02060-REGIO.
  ELSE.
    MOVE: '02'    TO D02060-REGIO.
  ENDIF.
```

Fortsetzung Programm U-UM-007 (RFSCHU39)

```
MOVE D02060 TO TAB-DYNPRO.
APPEND TAB.

WRITE: / KA-KTNRA USING NO EDIT MASK,
         KA-NAME1, KA-PLORT, KA-LAND1.

CALL TRANSACTION 'TS02' USING TAB MODE 'S'.
RETURN/CODE = SY-SUBRC.

MOVE SY-LANGU TO T100-SPRSL.
MOVE SY-MSGID TO T100-ARBGB.
MOVE SY-MSGNO TO T100-MSGNR.
READ TABLE T100.
REPLACE '$' WITH SY-MSGV1 INTO T100-TEXT.

CASE RETURN/CODE.
  WHEN 0.
    WRITE: / 'O.K', SY-MSGNO, (60) T100-TEXT.
  WHEN 4.
    WRITE: / 'Error', SY-MSGNO, (60) T100-TEXT.
  WHEN 8.
    WRITE: / 'Error in update task'.
ENDCASE.

REFRESH TAB.
```

Programm U-UM-008 (RFSCHU42)

```
REPORT RFSCHU42.
SEGMENTS: LH.

DATA: BEGIN OF SATZ,
         RDW(4), TABLE(4), MODID(1), DATUM TYPE D,
         USER LIKE LH-BNAME, NEW(80), OLD(80),
      END OF SATZ.

DATA: TABNAME(4),
      BEGIN OF STAT OCCURS 500,
        TABLE(4), ATTACH(3) TYPE P, INSERT(3) TYPE P,
        UPDATE(3) TYPE P, DELETE(3) TYPE P,
      END OF STAT.

PARAMETERS: NAME(6) DEFAULT 'SAPV01'.

DO.
  READ DATASET NAME INTO SATZ.
  IF SY-SUBRC NE 0.  EXIT. ENDIF.
  CHECK SATZ-TABLE NE SPACE.
  MOVE SATZ-TABLE TO STAT-TABLE.
  STAT-INSERT = STAT-UPDATE = STAT-DELETE = 0.
  MOVE 1 TO STAT-ATTACH.

  CASE SATZ-MODID.
    WHEN 'I'. MOVE 1 TO STAT-INSERT.
    WHEN 'U'. MOVE 1 TO STAT-UPDATE.
    WHEN 'D'. MOVE 1 TO STAT-DELETE.
  ENDCASE.
  COLLECT STAT.

  IF TABNAME NE SATZ-TABLE.
    SKIP 2.
    MOVE SATZ-TABLE TO TABNAME.
    RESERVE 5 LINES.
    WRITE: 'Tabelle'(001), SATZ-TABLE.
    SKIP.
  ENDIF.
```

Fortsetzung Programm U-UM-008 (RFSCHU42)

```
  IF SATZ-MODID = 'D'.
    MOVE SATZ-OLD TO SATZ-NEW.
    MOVE SPACE TO SATZ-OLD.
  ENDIF.

  WRITE:  SATZ-DATUM DD/MM/YY,
          SATZ-USER,
          SATZ-MODID,
        / SATZ-NEW,
        / SATZ-OLD.
ENDDO.

TABNAME = SPACE.
NEW-PAGE NO-HEADING.
WRITE 'Zugriffsstatistik'(002).
WRITE 'Tabelle  Zugriffe(gesamt)    Insert'
      Update      Delete'(003).

SORT STAT.

LOOP AT STAT.
  NEW-LINE.
  WRITE:    STAT-TABLE,
         13 STAT-ATTACH,
         30 STAT-INSERT,
         42 STAT-UPDATE,
         52 STAT-DELETE.
ENDLOOP.

TOP-OF-PAGE.
  IF TABNAME NE SPACE.
    WRITE: 'Tabelle'(001), SATZ-TABLE.
  ENDIF.
```

Programm U-UM-009 (RFSCHU72)

```
REPORT RFSCHU72.
SEGMENTS: KA.

TABLES: INDX.

DATA: BEGIN OF RANG OCCURS 10,
        KTNRA, NAME1, UMSATZ TYPE P DECIMALS 2,
      END OF RANG,

      BEGIN OF KEY,
        DBNAME(4),INDEXNAME(18),
      END OF KEY.

MOVE 'LIFA' TO KEY-DBNAME.
MOVE 'RANGLISTE' TO KEY-INDEXNAME.

IMPORT RANG
       FROM DATABASE INDX(IX) ID KEY.

IF SY-SUBRC NE 0.
  WRITE: 'Falscher Index'.
  STOP.
ENDIF.

LOOP AT RANG.
  WRITE: /(4) SY-TABIX NO-SIGN.
  WRITE: RANG-KTNRA, RANG-NAME1, RANG-UMSATZ.
ENDLOOP.
SKIP.
ULINE.
WRITE: 'Erstellungsdatum:'(001),
       INDX-AEDAT DD/MM/YY,
     / 'Ersteller:'(002),
       INDX-USERA UNDER INDX-AEDAT.
```

Programm U-UM-010 (RMSCHU25)

```
REPORT RMSCHU25.

SEGMENTS: KA, KB, KC.

DATA: UMSATZSOLL   LIKE UM01S,
      UMSATZHABEN  LIKE UM01H,
      UMSATZ       LIKE KUMUM,
      PNULL        TYPE P,
      MONAT(2)     TYPE P,
      JAHRL(2)     TYPE P.

SELECT-OPTIONS:
  LAND FOR KA-LAND1.

PARAMETERS:
  MONATE(2) DEFAULT '12'.

START-OF-SELECTION.
  IF MONATE GT '12'.
    NEW-PAGE NO-HEADING.
    SKIP TO LINE 14.
    POSITION 15.
    WRITE:
      'Parameter Monatswert" muss kleiner/
                     gleich 12 sein'(001).
      POSITION 15.
    WRITE:
          / 'eingegebener Wert:'(002),
            MONATE.
    STOP.
  ENDIF.

GET KA.
  CHECK SELECT-OPTIONS.

GET KC.
  NEW-PAGE.
  CHECK SELECT-OPTIONS.
  MOVE SY-DATUM+4(2) TO MONAT.
  MOVE SY-DATUM+2(2) TO JAHRL.
```

Fortsetzung Programm U-UM-010 (RMSCHU25)

```
WRITE:
          BUKRC,
          JAHRC,
       12 KTNRC,
          NAME1,
      /21 PLORT.
ULINE.
DETAIL.
DO MONATE TIMES VARYING UMSATZSOLL  FROM UM01S NEXT UM02S
                VARYING UMSATZHABEN FROM UM01H NEXT UM02H.
  MONAT = MONAT - 1.
  IF MONAT = PNULL.
    MONAT = 12.
    JAHRL = JAHRL - 1.
  ENDIF.
  UMSATZ = UMSATZHABEN - UMSATZSOLL.
  WRITE:
        /      'Monat'(003),
            (3) MONAT,
          15(3) JAHRL,
          30    UMSATZSOLL,
                UMSATZHABEN,
                UMSATZ.
ENDDO.
```

Programm U-UM-011 (RFSCHU19)

```
REPORT RMSCHU19.

SEGMENTS: MA, IE.

DATA: BEGIN OF RANGLISTE OCCURS 50,
        BSTNR,
        MENGE,
      END OF RANGLISTE,
      FLAG,
      RANG TYPE P.

GET IE.
  CHECK LVKNZ NE 'L'.
  CHECK MENGE GT 0.
  FILL RANGLISTE.
  COLLECT RANGLISTE SORTED BY MENGE.
  MOVE 'J' TO FLAG.

GET MA LATE.
  CHECK FLAG EQ 'J'.
  MOVE SPACE TO FLAG.
  NEW-PAGE.
  WRITE:   'Materialnummer:  '(004),
           (18) IDNRA,
         / 'Materialbez.:  '(005),
           17 KTX01.
  ULINE.
  WRITE: / TEXT-001,
           TEXT-002,
           27 TEXT-003.
  ULINE.
  DETAIL.
  LOOP AT RANGLISTE.
    RANG = RANG + 1.
    WRITE:
           /(2) RANG NO-SIGN,
            RANGLISTE-BSTNR,
            30(16) RANGLISTE-MENGE.
  ENDLOOP.
  REFRESH RANGLISTE.
  RANG = 0.
```

Programm U-UM-012 (RFSCHU24)

```
REPORT RMSCHU24.

SEGMENTS: MR.

DATA: KONTIERUNG(24).

SELECT-OPTIONS: RES-KZ FOR RSVKZ,
                RES-STAT FOR STATU.

GET MR.
  CHECK SELECT-OPTIONS.
  CASE RSVKZ.
    WHEN 'A'. MOVE MR-KDBUK TO KONTIERUNG.
              MOVE '/'      TO KONTIERUNG+2(1).
              MOVE MR-KDANR TO KONTIERUNG+3.
              MOVE '/'      TO KONTIERUNG+11(1).
              MOVE MR-KDHPS TO KONTIERUNG+12(4).
              MOVE '/'      TO KONTIERUNG+16(1).
              MOVE MR-KDUPS TO KONTIERUNG+17(4).
    WHEN 'D'. MOVE MR-KUNNR TO KONTIERUNG.
    WHEN 'F'. MOVE MR-AUFNR TO KONTIERUNG.
              MOVE '/'      TO KONTIERUNG+8(1).
              MOVE MR-AUFPS TO KONTIERUNG+9(2).
    WHEN 'I'. MOVE MR-ANLN1 TO KONTIERUNG.
              MOVE '/'      TO KONTIERUNG+8(1).
              MOVE MR-ANLN2 TO KONTIERUNG+9.
    WHEN 'K'. MOVE MR-KOSTL TO KONTIERUNG.
    WHEN 'O'. MOVE MR-TPANR TO KONTIERUNG .
    WHEN 'P'. MOVE MR-PROJN TO KONTIERUNG.
    WHEN 'T'. MOVE MR-TPANR TO KONTIERUNG.
    WHEN 'U'. MOVE MR-WRKUM TO KONTIERUNG.
              MOVE '/'      TO KONTIERUNG+2(1).
              MOVE MR-LGOUM TO KONTIERUNG+3.
    WHEN 'W'. MOVE SPACE TO KONTIERUNG.
    WHEN OTHERS. MOVE '??????'  TO KONTIERUNG.
  ENDCASE.
  WRITE:
          (18) ARTNR,
        / RESNR,
          RSDAT DD/MM/YY,
          WKDAT DD/MM/YY,
          STATU,
```

Programm U-UM-012 (RFSCHU24)

```
          MENGE,
          ENMNG,
*(DEL)    WAMNG,
          57 RSVKZ,
          64 KONTIERUNG.
  SKIP.
END-OF-SELECTION.
NEW-PAGE NO-HEADING.
  WRITE:
         / 'Datenbanskspezifische Abgrenzungen'(011),
         / TEXT-001,
           (18) MD/IDNR1,
         / TEXT-002,
           (18) MD/IDNR2,
         / TEXT-003,
           MD/BUKR1,
         / TEXT-004,
           MD/BUKR2,
         / TEXT-005,
           MD/WERK1,
         / TEXT-006,
           MD/WERK2,
         / TEXT-007,
           MD/LORT1,
         / TEXT-008,
           MD/LORT2,
         / TEXT-009,
           MD/MACO1,
         / TEXT-010,
           MD/MACO2.
  ULINE.
  WRITE:
         / 'Reportspezifische Abgrenzungen'(012),
         / 'Kontierungskennzeichen'(013),
           RES-KZ-LOW,
           RES-KZ-HIGH,
         / 'Reservierungsstatus'(014),
           RES-STAT-LOW,
           RES-STAT-HIGH.
```

Programm U-UM-013 (RFSCHU27)

```
REPORT RMSCHU27.

SEGMENTS: MA,
          MB,
          BV,
          *IE.

DATA: BEGIN OF COMMON PART,
        BUKREIS LIKE BUKRB,
        MATERIAL LIKE IDNRA,
      END OF COMMON PART.
  DATA: FLAG TYPE C VALUE 'N'.

SELECT-OPTIONS:
  MAT-ARTFOR MTART,
  MATKLASFOR MATKL,
  EINK-GRP FOR EKGRP.

GET MA.
  CHECK SELECT-OPTIONS.

GET MB.
  CHECK SELECT-OPTIONS.
  NEW-PAGE.
  WRITE:
              BUKRB,
         (18) IDNRB,
              EKGRP,
              MTART,
              MATKL,
         (37) KTX01.
  HIDE: BUKRB, IDNRB.
  ULINE.

GET BV.
  CHECK SELECT-OPTIONS.
  CHECK BSTAT EQ 'A' OR BSTAT EQ 'N'.
  DETAIL.
```

Fortsetzung Programm U-UM-013 (RFSCHU27)

```
  WRITE: BUKRS,
         WERKS,
         BANFN,
         BADAT DD/MM/YY,
         BAMNG,
         MEINS,
         SBNAM.
  HIDE: BUKRB, IDNRB.

AT LINE-SELECTION.
  MOVE BUKRB TO BUKREIS.
  MOVE IDNRB TO MATERIAL.

  MOVE: BUKREIS  TO *IE-BUKRS,
        MATERIAL TO *IE-IDTNR,
        'L'      TO *IE-INFKZ.
  FOR ALL *IE.
    MOVE 'J' TO FLAG.
    WRITE:
           / *IE-LIFNR.
    IF *IE-BNEPR NE PNULL.
      WRITE:
             *IE-BNEPR.
    ELSE.
      WRITE:
             *IE-NEPRS.
    ENDIF.
    WRITE:
           *IE-WAERS,
           *IE-BSTNR,
           *IE-LBDAT DD/MM/YY,
           *IE-MENGE.
  ENDFOR.
  IF FLAG EQ 'N'.
    WRITE: 'zu Material'(001),
           MATERIAL,
           'in Buchungskreis'(002),
           BUKREIS,
           'keine Info-Saetze vorhanden'(003).
  ENDIF.
```

Programm U-UM-014 (RFSCHU28)

```
REPORT RMSCHU28.

SEGMENTS: MA,
          MB,
          *IE.

DATA: BEGIN OF COMMON PART,
        BUKREIS LIKE BUKRB,
        MATERIAL LIKE IDNRA,
      END OF COMMON PART.
DATA: PNULL TYPE P.
  DATA: FLAG TYPE C VALUE 'N'.

SELECT-OPTIONS:
  EINK-GRP FOR EKGRP.

GET MB.
  CHECK SELECT-OPTIONS.
  WRITE:
              BUKRB,
         (18) IDNRB,
              EKGRP,
              MTART,
              MATKL,
         (37) KTX01.
  HIDE : BUKRB, IDNRB.

AT LINE-SELECTION.
  MOVE BUKRB TO BUKREIS.
  MOVE IDNRB TO MATERIAL.
  MOVE: BUKREIS  TO *IE-BUKRS,
        MATERIAL TO *IE-IDTNR,
        'L'      TO *IE-INFKZ.
  FOR ALL *IE.
    MOVE 'J' TO FLAG.
    WRITE:
           / *IE-LIFNR.
```

Fortsetzung Programm U-UM-014 (RFSCHU28)

```
    IF *IE-BNEPR NE PNULL.
     WRITE:
             *IE-BNEPR.
    ELSE.
      WRITE:
             *IE-NEPRS.
    ENDIF.
    WRITE:
          *IE-WAERS,
          *IE-BSTNR,
          *IE-LBDAT DD/MM/YY,
          *IE-MENGE.
  ENDFOR.
  IF FLAG EQ 'N'.
    WRITE: 'zu Material'(001),
           MATERIAL,
           'in Buchungskreis'(002),
           BUKREIS,
           'keine Info-Saetze vorhanden'(003).
  ENDIF.

TOP-OF-PAGE DURING LINE-SELECTION.
  WRITE:
         TEXT-004.
  ULINE.
```

Programm U-UM-015 (RFSCHU64)

```
REPORT RFSCHU64.
SEGMENTS: KA, HF.
DATA: OBLIGO TYPE P,
      TDM-OBLIGO TYPE P,
      BEGIN OF P OCCURS 400,
        BELNR, BUZEI, BUDAT, ZUONR, DMSHB, SHKZG,
      END OF P.

SELECT-OPTIONS T-OBLIGO FOR TDM-OBLIGO.

FIELD-GROUPS: HEADER,
              LIEFERANT,
              POSTEN.

INSERT OBLIGO KA-KTNRA HF-DMSHB   INTO HEADER.
INSERT KA-NAME1 KA-PLORT KA-STRAS INTO LIEFERANT.
INSERT HF-BELNR HF-BUZEI HF-BUDAT
       HF-ZUONR                   INTO POSTEN.

GET KA.
  CLEAR OBLIGO.
  REFRESH P.

GET HF.
  CHECK: AUGLZ = 0.
  ADD DMSHB TO OBLIGO.
  FILL P.
  APPEND P.

GET KA LATE.
  TDM-OBLIGO = OBLIGO / 100000 * -1.
  CHECK T-OBLIGO.
  EXTRACT LIEFERANT.
  LOOP AT P.
    MOVE: P-BELNR TO HF-BELNR,
          P-BUZEI TO HF-BUZEI,
          P-SHKZG TO HF-SHKZG,
          P-BUDAT TO HF-BUDAT,
          P-ZUONR TO HF-ZUONR,
          P-DMSHB TO HF-DMSHB.
    EXTRACT POSTEN.
  ENDLOOP.
```

Fortsetzung Programm U-UM-015 (RFSCHU64)

```
END-OF-SELECTION.
  SORT.
  LOOP.
    AT LIEFERANT.
      NEW-PAGE.
      SUMMARY.
      WRITE:    KA-KTNRA, KA-NAME1,
             40 OBLIGO CURRENCY LOCAL,
            /   KA-STRAS UNDER KA-NAME1,
            /   KA-PLORT UNDER KA-NAME1.
    ENDAT.
    AT POSTEN.
      DETAIL.
      WRITE: HF-BELNR UNDER KA-NAME1,
             HF-BUZEI, HF-BUDAT, HF-ZUONR,
             HF-DMSHB.
    ENDAT.
    AT LAST.
      SUMMARY.
      SKIP.
      WRITE: 'Insgesamt:'(001),
             SUM(HF-DMSHB) CURRENCY LOCAL UNDER HF-DMSHB.
    ENDAT.
  ENDLOOP.
```

Programm U-M-001 (RFSCHU27)

```
REPORT RMSCHU27.

SEGMENTS: MA,
          MB,
          BV,
          *IE.

DATA: BEGIN OF COMMON PART,
        BUKREIS LIKE BUKRB,
        MATERIAL LIKE IDNRA,
      END OF COMMON PART.

SELECT-OPTIONS:
  MAT-ARTFOR MTART,
  MATKLASFOR MATKL,
  EINK-GRP FOR EKGRP.

GET MA.
  CHECK SELECT-OPTIONS.

GET MB.
  CHECK SELECT-OPTIONS.
  NEW-PAGE.
  PERFORM MATZEILE.
  HIDE: BUKRB, IDNRB.
  ULINE.

GET BV.
  CHECK SELECT-OPTIONS.
  CHECK BSTAT EQ 'A' OR BSTAT EQ 'N'.
  DETAIL.
  WRITE: BUKRS,
         WERKS,
         BANFN,
         BADAT DD/MM/YY,
         BAMNG,
         MEINS,
         SBNAM.
  HIDE: BUKRB, IDNRB.
```

Fortsetzung Programm U-M-001 (RFSCHU27)

```
AT LINE-SELECTION.
  MOVE BUKRB TO BUKREIS.
  MOVE IDNRB TO MATERIAL.
  PERFORM ANGEBOT.
FORM ANGEBOT.
  DATA: FLAG TYPE C VALUE 'N'.

  MOVE: BUKREIS  TO *IE-BUKRS,
        MATERIAL TO *IE-IDTNR,
        'L'      TO *IE-INFKZ.
  FOR ALL *IE.
    MOVE 'J' TO FLAG.
    WRITE: / *IE-LIFNR.
    IF *IE-BNEPR NE PNULL.
      WRITE: *IE-BNEPR.
    ELSE.
      WRITE: *IE-NEPRS.
    ENDIF.
    WRITE: *IE-WAERS,
           *IE-BSTNR,
           *IE-LBDAT DD/MM/YY,
           *IE-MENGE.
  ENDFOR.
  IF FLAG EQ 'N'.
    WRITE: 'zu Material'(001),
           MATERIAL,
           'in Buchungskreis'(002),
           BUKREIS,
           'keine Info-Saetze vorhanden'(003).
  ENDIF.
ENDFORM.

FORM MATZEILE.
  WRITE: BUKRB,
         (18) IDNRB,
         EKGRP,
         MTART,
         MATKL,
         (37) KTX01.
  HIDE : BUKRB, IDNRB.
ENDFORM.
```

Programm U-M-002 (RFSCHU28)

```
REPORT RMSCHU28.

SEGMENTS: MA,
          MB,
          *IE.

DATA: BEGIN OF COMMON PART,
        BUKREIS LIKE BUKRB,
        MATERIAL LIKE IDNRA,
      END OF COMMON PART.
DATA: PNULL TYPE P.

SELECT-OPTIONS:
  EINK-GRP FOR EKGRP.

GET MB.
  CHECK SELECT-OPTIONS.
  PERFORM MATZEILE.

AT LINE-SELECTION.
  MOVE BUKRB TO BUKREIS.
  MOVE IDNRB TO MATERIAL.
  PERFORM ANGEBOT.

FORM MATZEILE.
  WRITE:
              BUKRB,
         (18) IDNRB,
              EKGRP,
              MTART,
              MATKL,
         (37) KTX01.
  HIDE : BUKRB, IDNRB.
ENDFORM.

FORM ANGEBOT.
  DATA: FLAG TYPE C VALUE 'N'.

  MOVE: BUKREIS  TO *IE-BUKRS,
        MATERIAL TO *IE-IDTNR,
        'L'      TO *IE-INFKZ.
```

Fortsetzung Programm U-M-002 (RFSCHU28)

```
  FOR ALL *IE.
    MOVE 'J' TO FLAG.
    WRITE:
           / *IE-LIFNR.
    IF *IE-BNEPR NE PNULL.
      WRITE:
             *IE-BNEPR.
    ELSE.
      WRITE:
             *IE-NEPRS.
    ENDIF.
    WRITE:
          *IE-WAERS,
          *IE-BSTNR,
          *IE-LBDAT DD/MM/YY,
          *IE-MENGE.
  ENDFOR.
  IF FLAG EQ 'N'.
    WRITE: 'zu Material'(001),
           MATERIAL,
           'in Buchungskreis'(002),
           BUKREIS,
           'keine Info-Saetze vorhanden'(003).
  ENDIF.
ENDFORM.

TOP-OF-PAGE DURING LINE-SELECTION.
  WRITE:
        TEXT-004.
  ULINE.
```

Programm U-M-003 (RMSCHU30)

```
REPORT RMSCHU30.

SEGMENTS: MA,
          MB,
          BV,
          *MC,
          *IE.

DATA:
      Y1(2) TYPE P,
      Y2(2) TYPE P,
      X1(2) TYPE P,
      X2(2) TYPE P,
      LOOPIND(2) TYPE P,
      NAME(8) TYPE C VALUE 'RMSCHU60',

      BEGIN OF LIFTAB OCCURS 20,
        LIFNR LIKE *IE-LIFNR,
        NEPRS LIKE *IE-NEPRS,
        BNEPR LIKE *IE-BNEPR,
        WAERS LIKE *IE-WAERS,
        PEINH LIKE *IE-PEINH,
        MEINS LIKE *IE-MEINS,
      END OF LIFTAB,

      PF23COUNT(2) TYPE P VALUE 1,
      MAXIND(2) TYPE P VALUE 1,
      STARTIND(2) TYPE P.

SELECT-OPTIONS:
                EINK-GRP FOR EKGRP,
                MAT-ART FOR MTART,
                MAT-KLAS FOR MATKL.

TOP-OF-PAGE.
  SKIP TO LINE 23.
  WRITE 34 'PF8 => Menue'(001).
  SKIP TO LINE 6.
```

Fortsetzung Programm U-M-003 (RMSCHU30)

```
GET MA.
  CHECK SELECT-OPTIONS.

GET MB.
  CHECK SELECT-OPTIONS.

GET BV.
 CHECK SELECT-OPTIONS.
  CHECK BSTAT EQ 'A' OR BSTAT EQ 'N'.
  DETAIL.
  WRITE:
              BUKRS,
              WERKS,
         (18) MATNR,
              BANFN,
              BSTAT,
              BADAT DD/MM/YY,
              BAMNG,
              MEINS,
              SBNAM.
  HIDE:  BUKRS,
         WERKS,
         MATNR,
         MTART,
         EKGRP.

AT PF8.
  WINDOW STARTING AT 25 17
         ENDING AT 55 23.
WRITE: / '+----------------------------+',
       / ':    PF9 :  Bestandsanzeige   :'(002),
       / ':    PF15:  Bezugsquellen     :'(003),
       / ':    PF16:  Submit RMSCHU60   :'(004),
       / ':    PF17:  Anstoss TE21      :'(005),
       / '+----------------------------+'.
```

Fortsetzung Programm U-M-003 (RMSCHU30)

```
AT PF9.
  Y1 = SY-CUROW + 1.
  IF Y1 GT 16.
    Y1 = 16.
  ENDIF.
  Y2 = Y1 + 7.
  WINDOW STARTING AT 01 Y1
         ENDING AT 79 Y2.
  MOVE: MATNR TO *MC-IDNRC,
        BUKRS TO *MC-BUKRC,
        WERKS TO *MC-WERKC.
  READ SEGMENT *MC.
  WRITE:  '+----------------------------------------+',
        / ' Material:             '(006),
   23(18) *MC-IDNRC,
        / ' Lagerbestand         '(007),
       23 *MC-LABST,
        / ' Verfuegbarer Bestand '(008),
       23 *MC-VERME,
        / ' Umlagerungsbestand   '(009),
       23 *MC-UMLME,
        / ' Reservierter Bestand '(010),
       23 *MC-RESME,
        / '+-------------------------------------+'.

AT PF15.
  REFRESH LIFTAB.
  MAXIND = 1.
  PF23COUNT = 1.
  MOVE: BUKRS TO *IE-BUKRS,
        MATNR TO *IE-IDTNR,
        'L' TO *IE-INFKZ.
  FOR ALL *IE.
    CHECK *LVKNZ NE 'L'.
    FILL LIFTAB FROM *IE.
    APPEND LIFTAB.
  ENDFOR.
  PERFORM BEZUGSQUELLEN.
```

Fortsetzung Programm U-M-003 (RMSCHU30)

```
AT PF16.
  SUBMIT (NAME) AND RETURN
       WITH MA/IDNRA EQ MATNR
       WITH MA/BUKRB EQ BUKRS
       WITH MA/WERKC EQ WERKS
       WITH EINKGRP IN EINK-GRP
       WITH MAT-ART IN MAT-ART.

AT PF17.
  CALL TRANSACTION 'TE21'.

AT PF22.
  PF23COUNT = 1.
  PERFORM BEZUGSQUELLEN.

AT PF23.
  PF23COUNT = PF23COUNT + 1.
  STARTIND = MAXIND - ( 5 * PF23COUNT ).
  IF STARTIND LE 0.
    STARTIND = 1.
  ENDIF.
  MAXIND = STARTIND.
  PERFORM BEZUGSQUELLEN.

FORM BEZUGSQUELLEN.
  WINDOW STARTING AT 31 07
         ENDING AT 79 15.
  WRITE: / TEXT-012,
         / ':  Liefnr.  Ang.prs.  Best.prs.',
              'Waehr pro Me:'(013),
         / ':  :',
        / TEXT-014.
  SKIP TO LINE 2.
```

Fortsetzung Programm U-M-003 (RMSCHU30)

```
  DO 5 TIMES.
    READ TABLE LIFTAB INDEX MAXIND.
    CHECK SY-SUBRC EQ '0000'.
    WRITE:
           /3 LIFTAB-LIFNR,
         (10) LIFTAB-NEPRS CURRENCY LIFTAB-WAERS,
         (10) LIFTAB-BNEPR CURRENCY LIFTAB-WAERS,
              LIFTAB-WAERS,
              LIFTAB-PEINH,
              LIFTAB-MEINS.
              MAXIND = MAXIND + 1.
  ENDDO.
ENDFORM.
```

Programm U-M-004 (RMSCHU42)

```
REPORT RMSCHU42.

SEGMENTS: MA,
          IE.

TABLES: INDX.

DATA:
     BEGIN OF EXPORT-KEY,
       ID-1(4),
       RECID(3) TYPE P,
     END OF EXPORT-KEY,
     BEGIN OF ITAB OCCURS 50,
       LIFNR,
       BSTNR,
       MENGE,
     END OF ITAB,
     SUM-MENGE(9) TYPE P DECIMALS 2,
     PROZME TYPE P DECIMALS 2,
     HPROZME LIKE PROZME,
     KUMPROZ LIKE PROZME,
     ENTRYCOUNT(2)TYPE P,
     MAXENTRY(2)  TYPE P,
     LINE-NO(2)   TYPE P,
     BALKEN(40)   TYPE C,
     STERN(40)    TYPE C VALUE
                  '****************************************',
     STRICHPOS(2) TYPE P,
     XSCALE(2)    TYPE P,
     SEL-LINE(2)  TYPE P,
     PNULL(9)     TYPE P,
     MERK         VALUE 'N',
     ZEILENKZ.

PARAMETERS:
  EXPORTID(4)   TYPE C DEFAULT 'XXXX',
  PRINT(1)      TYPE C DEFAULT 'N',
  BASE(1)       TYPE C DEFAULT '5',
  Y-ACHSE(2)    TYPE C DEFAULT '16',
  X-ACHSE(2)    TYPE C DEFAULT '60'.
```

Fortsetzung Programm U-M-004 (RMSCHU42)

```
START-OF-SELECTION.
  SET PF-STATUS '0001'.
  MOVE EXPORTID TO EXPORT-KEY-ID-1.

GET MA.
  MOVE 'N' TO MERK.

GET IE.
  CHECK LVKNZ NE 'L'.
  CHECK MENGE NE 0.
  FILL ITAB.
  APPEND ITAB SORTED BY MENGE.
  MOVE 'J' TO MERK.

GET MA LATE.
  CHECK MERK EQ 'J'.
  NEW-PAGE.
  WRITE:
          'Materialnummer:  '(006),
          IDNRA,
        / 'Materialbez.:  '(007),
          KTX01.
  MOVE 'M' TO ZEILENKZ.
  HIDE:
       ZEILENKZ,
       IDNRA,
       KTX01.
  CLEAR: ZEILENKZ.
  ULINE.
  WRITE:
        / TEXT-001,
          TEXT-002,
          TEXT-003.
  ULINE.
  DETAIL.
  LOOP AT ITAB.
    MAXENTRY = MAXENTRY + 1.
  ENDLOOP.
```

Fortsetzung Programm U-M-004 (RMSCHU42)

```
  LOOP AT ITAB.
    AT FIRST.
      SUM.
      SUM-MENGE = ITAB-MENGE.
      PERFORM ZEILE.
    ENDAT.
    PERFORM ZEILE.
    PERFORM GRAFIK1.
  ENDLOOP.

  REFRESH ITAB.

FORM ZEILE.
  PROZME = ITAB-MENGE * 100 * 100 / SUM-MENGE.
  MOVE 'L' TO ZEILENKZ.
  WRITE:
        /        ITAB-LIFNR,
          11     ITAB-BSTNR,
          28(16) ITAB-MENGE,
            (16) PROZME.
  MOVE PROZME TO HPROZME.
  HIDE:
       ZEILENKZ,
       ITAB-LIFNR,
       ITAB-BSTNR,
       ITAB-MENGE,
       HPROZME.
  CLEAR: ZEILENKZ.
ENDFORM.

FORM GRAFIK1.
  PROZME = PROZME / 100.
  KUMPROZ = KUMPROZ + PROZME.
  ENTRYCOUNT = ENTRYCOUNT + 1.
  HIDE: PROZME, KUMPROZ, ENTRYCOUNT, MAXENTRY.
ENDFORM.
```

Fortsetzung Programm U-M-004 (RMSCHU42)

```
AT PF16.
  SET PF-STATUS '0002'.
  IF PRINT = 'J'.
    NEW-PAGE PRINT ON.
  ENDIF.
  LINE-NO = BASE - 1.
  SKIP TO LINE LINE-NO.
  STRICHPOS = 5.
  POSITION STRICHPOS.
  WRITE: '% Bestellmenge'(008).
  LINE-NO = LINE-NO + 1.
  DO Y-ACHSE TIMES.
    SKIP TO LINE LINE-NO.
    LINE-NO = LINE-NO + 1.
    POSITION STRICHPOS.
    WRITE '!'.
  ENDDO.
  STRICHPOS = 6.
  DO X-ACHSE TIMES.
    POSITION STRICHPOS.
    WRITE: '-'.
    STRICHPOS = STRICHPOS + 1.
  ENDDO.
  DO 10 TIMES.
    XSCALE = X-ACHSE / 10.
    XSCALE = XSCALE * SY-INDEX.
    XSCALE = XSCALE + 5.
    POSITION XSCALE.
    WRITE '+'.
  ENDDO.
  POSITION STRICHPOS.
  WRITE: '% Lieferanten'(009).
  SEL-LINE = 9.
  DO 14 TIMES.
    READ LINE SEL-LINE.
    LINE-NO   = ( BASE + Y-ACHSE ) -
                ( Y-ACHSE * KUMPROZ / 100 ).
    STRICHPOS = 5 + X-ACHSE * ENTRYCOUNT / MAXENTRY.
    SKIP TO LINE LINE-NO.
    POSITION STRICHPOS.
```

Fortsetzung Programm U-M-004 (RMSCHU42)

```
      WRITE '*'.
      SEL-LINE = SEL-LINE + 1.
    ENDDO.

    IF PRINT = 'J'.
      NEW-PAGE PRINT OFF.
    ENDIF.

  AT PF17.
    SET PF-STATUS '0002'.
    IF PRINT = 'J'.
      NEW-PAGE PRINT ON.
    ENDIF.
    SEL-LINE = 9.
    DO 14 TIMES.
      READ LINE SEL-LINE.
      CHECK SY-LISEL(8) NE SPACE.
      MOVE STERN TO BALKEN PERCENTAGE PROZME.
      WRITE:
            / SY-LISEL(8),
              BALKEN.
      SEL-LINE = SEL-LINE + 1.
    ENDDO.
    IF PRINT = 'J'.
      NEW-PAGE PRINT OFF.
    ENDIF.

  AT PF18.
    SET PF-STATUS '0003'.
    DO.
      CLEAR ZEILENKZ.
      READ LINE SY-INDEX OF CURRENT PAGE.
      IF SY-SUBRC NE 0.
        EXIT.
      ENDIF.
```

Fortsetzung Programm U-M-004 (RMSCHU42)

```
    CASE ZEILENKZ.
      WHEN 'M'. EXPORT-KEY-RECID = EXPORT-KEY-RECID + 1.
                MOVE: SY-DATUM TO INDX-AEDAT,
                      SY-UNAME TO INDX-USERA,
                      SY-REPID TO INDX-PGMID.
                EXPORT ZEILENKZ
                       IDNRA
                       KTX01 TO DATABASE INDX(EK)
                       ID EXPORT-KEY.
      WHEN 'L'. EXPORT-KEY-RECID = EXPORT-KEY-RECID + 1.
                MOVE: SY-DATUM TO INDX-AEDAT,
                      SY-UNAME TO INDX-USERA,
                      SY-REPID TO INDX-PGMID.
                EXPORT ZEILENKZ
                       ITAB-LIFNR
                       ITAB-BSTNR
                       ITAB-MENGE
                       PROZME FROM HPROZME
                       TO DATABASE INDX(EK) ID EXPORT-KEY.
    ENDCASE.
  ENDDO.
  WRITE:
         'Anzahl exportierter Stze:'(012),
         EXPORT-KEY-RECID,
         'Export-ID auf INDX:      '(013),
         EXPORT-KEY-ID-1.

TOP-OF-PAGE DURING LINE-SELECTION.
  CASE SY-PFKEY.
    WHEN '0002'.
      WRITE:
        15 'Graphische Darstellung der
            Umsatzkonzentration'(011).
      ULINE /15(46).
    WHEN '0003'.
      WRITE:
        24 'Exportstatistik auf Datei INDX'(014).
      ULINE /24(30).
  ENDCASE.
  SKIP.
```

Programm U-M-005 (RFSCHU31)

```
REPORT RFSCHU31 NO STANDARD PAGE HEADING.

SEGMENTS: KA, KB, *KA, *KC, *HF.
TABLES: BKPF, BSEG.

DATA: BEGIN OF INFO,
        TEXT(6),
        SPACE,
        COUNTER(2),
      END OF INFO,
      BEGIN OF IDENT,
        BUKRB LIKE KB-BUKRB,
        BELNR LIKE HF-BELNR,
      END OF IDENT.
DATA: BEGIN OF TAB OCCURS 20.
        INCLUDE STRUCTURE BSEG.
      END OF TAB.
DATA: FLAG VALUE 'X',
      COUNTER TYPE P,
      SAVE/KTNRA LIKE KA-KTNRA,
      SAVE/BUKRB LIKE KB-BUKRB,
      FIRST/LINE TYPE P VALUE 3,
      START/WINDOW TYPE P VALUE 13,
      FIRST/WINDOW/LINE TYPE P,
      LINES TYPE P,
      LAST/LINE TYPE P VALUE 22,
      LINE1 TYPE P,
      LINE2 TYPE P,
      LIST0/CUROW LIKE SY-CUROW,
      LIST1/CUROW LIKE SY-CUROW.
```

Fortsetzung Programm U-M-005 (RFSCHU31)

```
GET KB.
  FORMAT INTENSIFIED OFF.
  WRITE: KA-KTNRA USING NO EDIT MASK,
         KB-BUKRB, KA-NAME1,
         KA-ORT01.
  HIDE: KA-KTNRA, KB-BUKRB, SY-WAERS, FLAG.

END-OF-SELECTION.
  SET PF-STATUS '0001'.
  INFO-TEXT = 'Info'(002).
  FLAG = SPACE.
  LINES = START/WINDOW - FIRST/LINE.
  FIRST/WINDOW/LINE = START/WINDOW + 1.

TOP-OF-PAGE.
  WRITE: 'Konto-Nr Bk NameOrt'(001),
         76 SY-PAGNO NO-SIGN.
  ULINE.

AT PF13.
  CASE SY-PFKEY.
    WHEN '0001'.
      IF SY-CUROW = 24 OR FLAG = SPACE.
        LIST0/CUROW = FIRST/LINE.
      ELSE.
        LIST0/CUROW = SY-CUROW.
      ENDIF.
      COUNTER = 1.
    WHEN OTHERS.
      CHECK SY-CUROW GE FIRST/LINE
            AND SY-CUROW LT START/WINDOW
            AND FLAG NE SPACE.
      IF LIST1/CUROW EQ SY-CUROW.
        LIST0/CUROW = LIST0/CUROW + 1.
        IF LIST0/CUROW GT LAST/LINE.
          SCROLL LIST INDEX 0 FORWARD.
          LIST0/CUROW = FIRST/LINE.
        ENDIF.
      ELSE.
        IF LIST1/CUROW = LIST0/CUROW.
           LIST0/CUROW = SY-CUROW.
        ELSE.
```

Fortsetzung Programm U-M-005 (RFSCHU31)

```
        LIST0/CUROW = SY-CUROW + LINES.
      ENDIF.
    ENDIF.
ENDCASE.
SET PF-STATUS '0002'.
FLAG = SPACE.
READ LINE LIST0/CUROW OF CURRENT PAGE INDEX 0.
CHECK FLAG NE SPACE.
KB-KTNRB = KA-KTNRA.
READ SEGMENT: KA,KB.
IF LIST0/CUROW < START/WINDOW.
  LINE1 = FIRST/LINE.
  LINE2 = START/WINDOW - 1.
ELSE.
  LINE1 = START/WINDOW.
  LINE2 = LAST/LINE.
ENDIF.
SY-LSIND = 1.
READ LINE 1 OF CURRENT PAGE INDEX 0.
WRITE: / SY-LISEL.
READ LINE 2 OF CURRENT PAGE INDEX 0.
IF LIST0/CUROW GE START/WINDOW.
  MOVE 'untere Haelfte'(009) TO SY-LISEL+60.
ENDIF.
WRITE: / SY-LISEL.
SAVE/KTNRA = KA-KTNRA.
SAVE/BUKRB = KB-BUKRB.
WHILE LINE1 LE LINE2.
  FLAG = SPACE.
  READ LINE LINE1 OF CURRENT PAGE INDEX 0.
  LINE1 = LINE1 + 1.
  CHECK FLAG NE SPACE.
  WRITE: / SY-LISEL.
  HIDE: FLAG.
ENDWHILE.
KA-KTNRA = SAVE/KTNRA.
KB-BUKRB = SAVE/BUKRB.
LIST1/CUROW = LIST0/CUROW.
IF LIST1/CUROW GE START/WINDOW.
  LIST1/CUROW = LIST1/CUROW - LINES.
ENDIF.
```

Fortsetzung Programm U-M-005 (RFSCHU31)

```
  SY-CUROW = LIST1/CUROW.
  SET CURSOR 1 LIST1/CUROW.
  PERFORM SPLIT/SCREEN.
  FLAG = SPACE.

AT PF14.
  SET PF-STATUS '0002'.
  CHECK SY-CUROW GE FIRST/LINE AND SY-CUROW LT START/WINDOW
  AND FLAG NE SPACE.
  IF LIST1/CUROW = SY-CUROW.
    COUNTER = COUNTER + 1.
    IF COUNTER = 6.
      COUNTER = 1.
    ENDIF.
  ELSE.
    IF LIST1/CUROW = LIST0/CUROW.
      LIST0/CUROW = SY-CUROW.
    ELSE.
      LIST0/CUROW = SY-CUROW + LINES.
    ENDIF.
    LIST1/CUROW = SY-CUROW.
    FLAG = SPACE.
    READ LINE LIST0/CUROW OF CURRENT PAGE INDEX 0.
    CHECK FLAG NE SPACE.
    KB-KTNRB = KA-KTNRA.
    READ SEGMENT: KA,KB.
  ENDIF.
  PERFORM SPLIT/SCREEN.
  FLAG = SPACE.

AT PF3.
  SY-LSIND = 1.

AT PF2.
  SET PF-STATUS '0004'.
  MOVE *HF-KBUKR TO IDENT-BUKRB.
  MOVE *HF-BELNR TO IDENT-BELNR.
  IMPORT BKPF BSEG TO TAB
  FROM DATABASE ABEZ ID IDENT.
```

Fortsetzung Programm U-M-005 (RFSCHU31)

```
  IF SY-SUBRC NE 0.
    WRITE: / 'Beleg nicht gefunden'(021),
                 IDENT-BUKRB, IDENT-BELNR.
  ELSE.
    WRITE: / BKPF-BUKRS, BKPF-BELNR,
           15 BKPF-BLART, 20 BKPF-BLDAT DD/MM/YYYY,
           35 BKPF-WAERS.
    SKIP.

    LOOP AT TAB.
      WRITE:/ TAB-BUZEI, TAB-DMBTR CURRENCY BKPF-WAERS,
      TAB-KOSTL, TAB-HKONT.
    ENDLOOP.
  ENDIF.

TOP-OF-PAGE DURING LINE-SELECTION.
  CASE SY-PFKEY.
    WHEN '0003'.
      IF COUNTER EQ 5.
        WRITE: /50 'Beleganzeige Seite'(004),
                    (3) SY-PAGNO,
                  / 'Beleg-Nr Bz BuchDatum'(005).
        WRITE:  26 'Zuordnung BA BS U Betrag'(006).
      ENDIF.
    WHEN '0004'.
      WRITE: / 'BK BelNr.  BABelDatum Waehrung'(007).
      WRITE: / 'ZeiBetrag Kost.St.  Konto'(008).
      ULINE.
  ENDCASE.

FORM SPLIT/SCREEN.
  INFO-COUNTER = COUNTER.
  WINDOW LIST STARTING AT 1 FIRST/WINDOW/LINE
         ENDINGAT 79 LAST/LINE
         WITH FRAME
         TITLE INFO.
```

Fortsetzung Programm U-M-005 (RFSCHU31)

```
  CASE COUNTER.
    WHEN 1.
      PERFORM INFO-1.
    WHEN 2.
      PERFORM INFO-2.
    WHEN 3.
      PERFORM INFO-3.
    WHEN 4.
      PERFORM INFO-4.
    WHEN 5.
      SET PF-STATUS '0003'.
      PERFORM INFO-5.
  ENDCASE.
ENDFORM.

FORM INFO-1.
  FORMAT INTENSIFIED ON.
  WRITE: / 'Lieferant'(011), 50 'Zentrale'(012),
           KA-LNRZE NO-ZERO.
  IF KA-LNRZE EQ '00000000'.
    WRITE:
          / 'Anrede..........'(013),
            ANRED INTENSIFIED,
         50 'Keine Zentrale angegeben'(014),
          / 'Name 1..........'(015),
            NAME1 ,
          / 'Name 2..........'(016) ,
            NAME2 INTENSIFIED,
          / 'Strasse.........'(017) ,
            STRAS ,
          / 'Postfach........'(018) ,
            PFACH ,
          / 'Ort.............'(019) ,
            ORT01 ,
          / 'Land............'(020) ,
            LAND1 .
  ELSE.
    MOVE KA-LNRZE TO *KA-KTNRA.
    READ SEGMENT *KA.
    WRITE:
          / 'Anrede..........'(013) ,
```

Fortsetzung Programm U-M-005 (RFSCHU31)

```
            ANRED , 50 *ANRED,
          / 'Name 1..........'(015) ,
            NAME1 , 50 *NAME1,
          / 'Name 2..........'(016) ,
            NAME2 , 50 *NAME2,
          / 'Strasse.........'(017) ,
            STRAS , 50 *STRAS,
          / 'Postfach........'(018) ,
            PFACH , 50 *PFACH,
          / 'Ort.............'(019) ,
            ORT01 , 50 *ORT01,
          / 'Land............'(020) ,
            LAND1 , 50 *LAND1.
  ENDIF.
ENDFORM.

FORM INFO-2.
  FORMAT INTENSIFIED ON.
  WRITE: 'Bankverbindung und Zahlungsdaten'(041).
  FORMAT INTENSIFIED OFF.
  WRITE:
        / 'Bankkonto.......'(042), KA-BANKN,
       40 'Zahlungskonditionen.'(043), KB-ZTERM,
        / 'Bankleitzahl....'(044), KA-BANKL,
       40 'Wechselkonditionen..'(045), KB-WKOND,
        / 'Bankanschrift...'(046), KA-BANKA,
       40 'Zahlungsarten.......'(047), KB-ZAHLA,
        / 'Postscheckkonto.'(048), KA-PSKTO,
       40 'Zahlungsschluessel..'(049), KB-ZLSCH.
  FORMAT INTENSIFIED ON.
  WRITE: 'Telefon, Telex, Telefax'(050).
  FORMAT INTENSIFIED OFF.
  WRITE:
        / 'Telefon.........'(051), KA-TELF1,
       40 'Telex '(052),KA-TELX1,
        / 'Telefax.........'(053), KA-TELFX.
ENDFORM.
```

Fortsetzung Programm U-M-005 (RFSCHU31)

```
FORM INFO-3.
  SKIP TO LINE 2.
  FORMAT INTENSIFIED ON.
  WRITE: 10 'Geschaeftsjahr Jahresumsatz '(054).
  FORMAT INTENSIFIED OFF.
  SKIP.
  MOVE: KA-KTNRA TO *KC-KTNRC, KB-BUKRB TO *KC-BUKRC.
        *KC-JAHRC = SY-DATUM(4).
  WRITE: 16 *KC-JAHRC.
  READ SEGMENT *KC.
  IF SY-SUBRC = 0.
    WRITE: 31 *KC-KUMUM.
  ELSE.
    WRITE: 33 'Nicht bebucht'(033) INTENSIFIED OFF.
  ENDIF.
  SKIP.
  *KC-JAHRC = SY-DATUM(4) - 1.
  WRITE: 16 *KC-JAHRC.
  READ SEGMENT *KC.
  IF SY-SUBRC = 0.
    WRITE: 31 *KC-KUMUM.
  ELSE.
    WRITE: 33 'Nicht bebucht'(033) INTENSIFIED OFF.
  ENDIF.
ENDFORM.

FORM INFO-4.
  MOVE: KA-KTNRA TO *KC-KTNRC, KB-BUKRB TO *KC-BUKRC.
  *KC-JAHRC = SY-DATUM(4).
  FORMAT INTENSIFIED ON.
  WRITE: / 'Verkehrszahlen im Geschaeftsjahr'(031),
           *JAHRC.
  FORMAT INTENSIFIED OFF.
  READ SEGMENT *KC.
  IF SY-SUBRC = 0.
  WRITE:
        / 'Periode', ' Soll Haben'(032),
       40 'Periode', ' Soll Haben'(032).
  DETAIL. INPUT.
```

Fortsetzung Programm U-M-005 (RFSCHU31)

```
  WRITE:
    / '01', *UM01S CURRENCY LOCAL, *UM01H CURRENCY LOCAL,
   40 '07', *UM07S CURRENCY LOCAL, *UM07H CURRENCY LOCAL,
    / '02', *UM02S CURRENCY LOCAL, *UM02H CURRENCY LOCAL,
   40 '08', *UM08S CURRENCY LOCAL, *UM08H CURRENCY LOCAL,
    / '03', *UM03S CURRENCY LOCAL, *UM03H CURRENCY LOCAL,
   40 '09', *UM09S CURRENCY LOCAL, *UM09H CURRENCY LOCAL,
    / '04', *UM04S CURRENCY LOCAL, *UM04H CURRENCY LOCAL,
   40 '10', *UM10S CURRENCY LOCAL, *UM10H CURRENCY LOCAL,
    / '05', *UM05S CURRENCY LOCAL, *UM05H CURRENCY LOCAL,
   40 '11', *UM11S CURRENCY LOCAL, *UM11H CURRENCY LOCAL,
    / '06', *UM06S CURRENCY LOCAL, *UM06H CURRENCY LOCAL,
   40 '12', *UM12S CURRENCY LOCAL, *UM12H CURRENCY LOCAL.
  ELSE.
    SKIP. DETAIL.
    WRITE 'Nicht bebucht'(033).
  ENDIF.
ENDFORM.

FORM INFO-5.
  MOVE: 'K' TO *HF-KOART,
        KA-KTNRA TO *HF-LIFNR,
        KB-BUKRB TO *HF-KBUKR.

  FOR ALL *HF.
    WRITE: / *HF-BELNR, *HF-BUZEI, *HF-BUDAT DD/MM/YYYY,
    *HF-ZUONR, *HF-BLART, *HF-BSCHL,
    *HF-UMSKZ, *HF-DMSHB CURRENCY *HF-WAERS.
    HIDE: *HF-KBUKR, *HF-BELNR.
  ENDFOR.
  WRITE: /20 '*** Ende der Beleganzeige ***'(003).
 ENDFORM.
```

Programm U-M-006 (RMSCHU34)

```
REPORT RMSCHU34
                NO STANDARD PAGE HEADING.

SEGMENTS: MA,
          MB,
          MC.

DATA:
     REPID      LIKE SY-REPID,
     BNAME      LIKE TSDC-ODC/NAME,
     PASSW      LIKE TSDC-ODC/PASSW,
     ODCKZ(1)   TYPE C,
     POSCOUNT(2) TYPE P,
     POSANZ(2)   TYPE C,
     PNULL       TYPE P,
     YBSTA(1)    TYPE X VALUE '02',

     BEGIN OF DISPLAY-TAB OCCURS 23,
       ZEILE(79),
     END OF DISPLAY-TAB.

TABLES:MBWT.

DYNPROS: D00001,
         D00002,
         D08431,
         D08432,
         D08433.

SELECT-OPTIONS:
               MAT-ART FOR MTART,
               MATKLAS FOR MATKL,
               AENDDAT FOR ZKDAT.

PARAMETERS:
            ST-PREIS(1) TYPE C DEFAULT ' ',
            POS-ANZ(2)  TYPE C DEFAULT '50',
            SIMULATE(1) TYPE C DEFAULT 'X',
            ODC-KZ(1)   TYPE C.
```

Fortsetzung Programm U-M-006 (RMSCHU34)

```
INITIALIZATION.
  MOVE 'I'        TO AENDDAT-SIGN.
  MOVE 'BT'       TO AENDDAT-OPTION.
  MOVE SY-DATUM   TO AENDDAT-LOW.
  MOVE '99991231' TO AENDDAT-HIGH.
  APPEND AENDDAT.

START-OF-SELECTION.
  REPID = SY-REPID.
  REFRESH DYNPRO: D00001,
                  D08431,
                  D08432,
                  D08433.
  IF ODC-KZ = SPACE.
    ODCKZ = SPACE.
  ELSE.
    ODCKZ = 'X'.
  ENDIF.
  CALL FUNCTION 'LESEN/TAB/TSDC'
            EXPORTING REPORT = REPID
                      ODC/KENNZ  = ODCKZ
            IMPORTING BENUTZER   = BNAME
                      PASSWORT   = PASSW.

  MOVE: SY-MANDT     TOD00001-MANDT,
        BNAME        TOD00001-BNAME(4),
        'TR21'       TOD00001-BNAME+4(4),
        SY-UZEIT     TOD00001-BNAME+8(4),
        PASSW        TOD00001-BCODE,
        'TR21'       TOD00001-TCODE,
        SY-LANGU     TOD00001-SPRAS.
  IF SIMULATE EQ ' '.
    IF ODC-KZ = SPACE.
      TRANSFER-DYNPRO D00001.
    ELSE.
      TRANSFER-DYNPRO D00001
      GENERATING ONLINE COMMUNICATION.
    ENDIF.
  ENDIF.
```

Fortsetzung Programm U-M-006 (RMSCHU34)

```
GET MA.
  CHECK SELECT-OPTIONS.

GET MB.
  CHECK SELECT-OPTIONS.

GET MC.
  CHECK STATC O YBSTA.
  CLEAR MBWT.
  IF BWTTY NE ' '.
    SELECT * FROM MBWT WHERE
             ARTNR EQ MA-IDNRA AND
             BUKRS EQ MB-BUKRB AND
             WERKS EQ MC-WERKC AND
             MMBWT EQ 'B'.
      CHECK MBWT-ZKDAT GE AENDDAT-LOW.
      IF AENDDAT-HIGH NE '00000000'.
        CHECK MBWT-ZKDAT LE AENDDAT-LOW.
      ENDIF.
      IF ST-PREIS EQ 'J'.
        CHECK MBWT-BWPRS NE PNULL.
      ELSE.
        CHECK MBWT-ZKPRS NE PNULL.
      ENDIF.
      IF POSCOUNT = 0.
        PERFORM KOPF.
      ENDIF.
      IF POSCOUNT <> 0 AND POSCOUNT LE POS-ANZ.
        PERFORM POSITION.
      ENDIF.
      IF POSCOUNT > POS-ANZ.
        PERFORM BUCHEN.
        CLEAR POSCOUNT.
      ENDIF.
      POSCOUNT = POSCOUNT + 1.
    ENDSELECT.
  ELSE.
```

Fortsetzung Programm U-M-006 (RMSCHU34)

```
      CHECK SELECT-OPTIONS.
      IF ST-PREIS EQ 'J'.
        CHECK BWPRS NE PNULL.
      ELSE.
        CHECK ZKPRS NE PNULL.
      ENDIF.
      IF POSCOUNT = 0.
        PERFORM KOPF.
    ENDIF.
    IF POSCOUNT <> 0 AND POSCOUNT LE POS-ANZ.
      PERFORM POSITION.
    ENDIF.
    IF POSCOUNT > POS-ANZ.
      PERFORM BUCHEN.
      CLEAR POSCOUNT.
    ENDIF.
    POSCOUNT = POSCOUNT + 1.
  ENDIF.

END-OF-SELECTION.
  PERFORM BUCHEN.
  IF ODC-KZ = 'X'.
    REFRESH DYNPRO  D00002.
    MOVE 'EIRC'  TO D00002-TC.
    MOVE 'S'     TO D00002-SP.
    TRANSFER-DYNPRO D00002.
    SKIP 5.
    WRITE: / 'Der Mappenname lautet:'(001),D00001-BNAME.
  ELSE.
    SKIP 5.
    WRITE: / 'Der Mappenname lautet:'(001),D00001-BNAME.
  ENDIF.
```

Fortsetzung Programm U-M-006 (RMSCHU34)

```
FORM KOPF.
  MOVE: 'S'TO D08431-SP,
        'TR21' TO D08431-TC,
        BUKRCTO D08431-BUKRS.
  IF SIMULATE EQ ' '.
    TRANSFER-DYNPRO D08431.
    REFRESH DYNPRO D08431.
  ELSE.
    MOVE-DYNPRO D08431 TO DISPLAY-TAB.
    PERFORM DISPLAY.
  ENDIF.
  IF BWTTY NE ' '.
    MOVE: MBWT-ARTNRTO D08432-ARTNR,
    MBWT-WERKSTO D08432-WERKS,
    MBWT-BWTARTO D08432-BWTAR.
  ELSE.
    MOVE: IDNRCTO D08432-ARTNR,
    WERKCTO D08432-WERKS.
  ENDIF.
  IF SIMULATE EQ ' '.
    TRANSFER-DYNPRO D08432.
    REFRESH DYNPRO D08432.
  ELSE.
    MOVE-DYNPRO D08432 TO DISPLAY-TAB.
    PERFORM DISPLAY.
  ENDIF.
  IF BWTTY NE ' '.
    IF ST-PREIS EQ 'X'.
      MOVE MBWT-BWPRS TO D08433-NPREI.
    ELSE.
      MOVE MBWT-ZKPRS TO D08433-NPREI.
    ENDIF.
  ELSE.
    IF ST-PREIS EQ 'X'.
      MOVE BWPRS TO D08433-NPREI.
    ELSE.
      MOVE ZKPRS TO D08433-NPREI.
    ENDIF.
  ENDIF.
ENDFORM.
```

Fortsetzung Programm U-M-006 (RMSCHU34)

```
FORM POSITION.
  IF BWTTY NE ' '.
    MOVE: MBWT-ARTNR TO D08433-ARTNR,
    MBWT-WERKS TO D08433-WERKS,
    MBWT-BWTAR TO D08433-BWTAR.
  ELSE.
    MOVE: IDNRCTO D08433-ARTNR,
          WERKCTO D08433-WERKS.
  ENDIF.
  IF SIMULATE EQ ' '.
    TRANSFER-DYNPRO D08433.
    REFRESH DYNPRO: D08433.
  ELSE.
    MOVE-DYNPRO D08433 TO DISPLAY-TAB.
    PERFORM DISPLAY.
  ENDIF.
  IF BWTTY NE ' '.
    IF ST-PREIS EQ 'X'.
      MOVE MBWT-BWPRS TO D08433-NPREI.
    ELSE.
      MOVE MBWT-ZKPRS TO D08433-NPREI.
    ENDIF.
  ELSE.
    IF ST-PREIS EQ 'X'.
      MOVE BWPRS TO D08433-NPREI.
    ELSE.
      MOVE ZKPRS TO D08433-NPREI.
    ENDIF.
  ENDIF.
ENDFORM.
```

Fortsetzung Programm U-M-006 (RMSCHU34)

```
FORM BUCHEN.
  MOVE: 'BU' TO D08433-PRFCO.
  IF SIMULATE EQ ' '.
    TRANSFER-DYNPRO D08433.
    REFRESH DYNPRO: D08433.
  ELSE.
    MOVE-DYNPRO D08433 TO DISPLAY-TAB.
    PERFORM DISPLAY.
  ENDIF.
ENDFORM.

FORM DISPLAY.
  NEW-PAGE.
  LOOP AT DISPLAY-TAB.
    WRITE DISPLAY-TAB-ZEILE.
  ENDLOOP.
  REFRESH DISPLAY-TAB.
ENDFORM.
```

Programm U-M-007 (RFSCHU80)

```
REPORT RFSCHU80 NO STANDARD PAGE HEADING.

PARAMETERS: PROGNAME(8) DEFAULT 'RFSCHU80'.

* Programmsource
DATA: BEGIN OF T OCCURS 100, LINE(72), END OF T.
DATA: BEGIN OF SPECIALS OCCURS 100,
        WORD(30),
      END OF SPECIALS.

* Tabellen einlesen
READ REPORT PROGNAME INTO T.
PERFORM FILL-SPECIALS.
* ...  Status setzen
SET PF-STATUS 'CODE'.
* ...  und Source ausgeben
LOOP AT T.
  NEW-LINE.
  IF T-LINE = SPACE.
    SKIP.
  ELSEIF T-LINE(1) = '*'.
    WRITE T-LINE INTENSIFIED ON.
  ELSE.
    PERFORM T-LINE-OUT.
  ENDIF.
ENDLOOP.

*-------------------------------------*
* Doku zu Schkuesselwort anzeigen     *
*-------------------------------------*
AT PF2.
* Keine Blinde Zeilenauswahl
  CHECK SY-WINRO = 0 AND SY-WINCO = 0.

* Zeige Doku zu ausgewaehltem Schluesselwort an
  PERFORM DISPLAY-DOCU.

* Letztes Zeichen des aktuellen/vorangehenden Worts
  DATA: LC(1), OLC(1) VALUE '.'.
```

Fortsetzung Programm U-M-007 (RFSCHU80)

```
*--------------------------------------*
* Ausgabe einer einzelnen Zeile        *
*--------------------------------------*
FORM T-LINE-OUT.
  DATA: OFS TYPE P.
  DATA: LEN TYPE P.
  DATA: POS TYPE P.
  DATA: WRD(30).
  FIELD-SYMBOLS: <W>.
  FIELD-SYMBOLS: <B>.
  ASSIGN T-LINE(1) TO <B>.
  DO 72 TIMES.
    IF <B> NE SPACE.
      ADD 1 TO LEN.
      LC = <B>.
    ENDIF.
    IF <B> = SPACE OR SY-INDEX = 72.
      IF LEN NE 0.
        OFS = SY-INDEX - LEN - 1.
        ASSIGN T-LINE+OFS(LEN) TO <W>.
        POS = OFS + 1.
        POSITION POS.
        FORMAT INTENSIFIED OFF.
        IF OLC = '.'  OR OLC = ','.
          WRD = <W>.
          REPLACE '.' WITH SPACE INTO WRD.
          READ TABLE SPECIALS WITH KEY WRD BINARY SEARCH.
          IF SY-SUBRC = 0.
            FORMAT INTENSIFIED ON.
          ENDIF.
        ENDIF.
        WRITE <W>.
        OLC = LC.
        LEN = 0.
      ENDIF.
    ENDIF.
    IF SY-INDEX LT 72.
      ASSIGN <B>+1 TO <B>.
    ENDIF.
  ENDDO.
ENDFORM.
```

Fortsetzung Programm U-M-007 (RFSCHU80)

```
*--------------------------------------------------*
* Tabelle mit Struktur-Schluesselwoertern fuellen *
*--------------------------------------------------*
FORM FILL-SPECIALS.
  PERFORM APPEND-SPECIAL USING:
          'PROGRAM', 'REPORT', 'IF', 'ELSE',
          'ENDIF', 'CASE', 'WHEN', 'ENDCASE',
          'DO', 'ENDDO', 'WHILE', 'ENDWHILE',
          'LOOP', 'ENDLOOP', 'UPLOAD', 'ENDUPLOAD',
          'ENDAT', 'FOR', 'ENDFOR', 'EXIT', 'STOP',
          'FORM', 'ENDFORM', 'PERFORM', 'CALL',
          'MODULE', 'ENDMODULE',
          'FUNCTION', 'ENDFUNCTION',
          'AT', 'GET', 'INITIALIZATION'
          'START-OF-SELECTION', 'END-OF-SELECTION',
          'TOP-OF-PAGE', 'END-OF-PAGE', .
  SORT SPECIALS BY WORD.
ENDFORM.

*------------------------------------------*
* Serviceroutine fuer FILL-SPECIALS:      *
*   Eintragen eines einzelnen Worts       *
*------------------------------------------*
FORM APPEND-SPECIAL USING WORD.
  SPECIALS-WORD = WORD.
  APPEND SPECIALS.
ENDFORM.

*----------------------------------------------*
* Zeigt Doku an (PF2).                        *
*----------------------------------------------*
FORM DISPLAY-DOCU.
  DATA WORD(72).
  DATA SUBRC LIKE SY-SUBRC.
  PERFORM GET-CONTENT USING WORD SUBRC.
  IF SUBRC > 0.  EXIT. ENDIF.
  PERFORM SHOW-WORD-DOCU USING WORD SUBRC.
  IF SUBRC > 0.  EXIT. ENDIF.
ENDFORM.
```

Fortsetzung Programm U-M-007 (RFSCHU80)

```
*-----------------------------------------------*
* Besorgt ausgewaehltes Wort                    *
*-----------------------------------------------*
FORM GET-CONTENT USING WORD SUBRC.
  DATA NAME(30).
  DATA OFFSET TYPE P.
  DATA FDPOS LIKE SY-FDPOS VALUE 99.
* Inhalt des durch Cursorpositionierung
* ausgewaehlten Worts nach WORD
  GET CURSOR FIELD NAME OFFSET OFFSET VALUE WORD.
  IF WORD <= SPACE.
    PERFORM PICK-ERROR.
    SUBRC = 4.
    EXIT.
  ENDIF.
  IF WORD CS '.'.
    FDPOS = SY-FDPOS.
  ENDIF.
  IF WORD CS ':' AND SY-FDPOS < FDPOS.
    FDPOS = SY-FDPOS.
  ENDIF.
  IF FDPOS < 99.
    PERFORM SPLIT-WORD USING WORD OFFSET FDPOS.
  ENDIF.
ENDFORM.
```

Fortsetzung Programm U-M-007 (RFSCHU80)

```
*-------------------------------------------------*
* Ruft Doku fuer ausgewaehltes Schluesselwort auf *
*-------------------------------------------------*
FORM SHOW-WORD-DOCU USING WORD SUBRC.
* Window oben oder unten
  DATA Y1 LIKE SY-WINY1.
  DATA Y2 LIKE SY-WINY2.
* Tabelle fuer Doku
  DATA: BEGIN OF DTAB OCCURS 100,
          FORMAT(1),
          LINE(61),
        END OF DTAB.
* Schluessel fuer READ DOCUMENTATION
  DATA: BEGIN OF DDNAME,
          'SD',
          'ABAP',
          WORD(30),
        END OF DDNAME.
* Titel fuer Window
  DATA: BEGIN OF TITEL,
          TEXT(30),
          WORD(30),
        END OF TITEL.
* Schluessel fuellen
  DDNAME-WORD= WORD.
* ...  und Doku lesen
  READ DOCUMENTATION DDNAME LANGUAGE SY-LANGU INTO DTAB.
* Keine Doku da ==> Fehlermeldung
  IF SY-SUBRC NE 0.
    PERFORM PICK-ERROR.
    SUBRC = 4.
    EXIT.
  ENDIF.
```

Fortsetzung Programm U-M-007 (RFSCHU80)

```
* Fuelle Titel
  TITEL-TEXT = 'Schluesselwortdokumentation'(003).
  TITEL-WORD = WORD.
  CONDENSE TITEL.
* Window an die richtige Stelle
  IF SY-CUROW < 12.
    Y1 = 13.
    Y2 = 22.
  ELSE.
    Y1 = 2.
    Y2 = 10.
  ENDIF.
  WINDOW LIST STARTING AT 8Y1
         ENDING AT 72 Y2
         WITH FRAME
         TITLE TITEL.
* Gib Doku ins Window aus
  FORMAT INTENSIFIED.
  SET BLANK LINES ON.
  LOOP AT DTAB.
* .PAG creates NEW-PAGE.
    IF DTAB-LINE(4) = '.PAG'.
      NEW-PAGE.
    ELSE.
      IF DTAB-LINE(1) = '.'  AND DTAB-LINE+1 = SPACE.
        SKIP 1.
      ELSE.
        IF DTAB-LINE(1) = '*'.
          FORMAT INTENSIFIED ON COLOR 5.
        ENDIF.
        WRITE / DTAB-LINE.
        FORMAT INTENSIFIED OFF COLOR 1.
      ENDIF.
    ENDIF.
  ENDLOOP.
ENDFORM.
```

Fortsetzung Programm U-M-007 (RFSCHU80)

```
*-------------------------------------*
* Bringt Fehlermeldung in Window      *
*-------------------------------------*
FORM PICK-ERROR.
* Window oben oder unten
  DATA Y1 LIKE SY-WINY1.
  DATA Y2 LIKE SY-WINY2.
  IF SY-CUROW < 14.
    Y1 = 18.
    Y2 = 19.
  ELSE.
    Y1 = 8.
    Y2 = 9.
  ENDIF.
  WINDOW STARTING AT 20 Y1
         ENDING AT 60 Y2
         WITH FRAME.
  WRITE: 'Bitte stellen Sie den Cursor'(001).
  WRITE: 'auf ein ABAP-Schluesselwort.'(002).
  ENDFORM.
```

Fortsetzung Programm U-M-007 (RFSCHU80)

```
*-----------------------------------------*
* Serviceroutine fuer GET-CONTENT:        *
*   findet richtiges Stueck des Worts     *
*-----------------------------------------*
FORM SPLIT-WORD USING WORD VALUE(OFFSET) FDPOS.
  DATA LENGTH TYPE P.
  FIELD-SYMBOLS <TAIL>.
  DO.
    IF FDPOS < OFFSET.
      SHIFT WORD BY FDPOS PLACES.
      SHIFT WORD.
      OFFSET = OFFSET - FDPOS - 1.
      FDPOS = 99.
      IF WORD CS '.'.
        FDPOS = SY-FDPOS.
      ENDIF.
      IF WORD CS ':' AND SY-FDPOS < FDPOS.
        FDPOS = SY-FDPOS.
      ENDIF.
      IF FDPOS = 99.
        EXIT.
      ENDIF.
    ELSE.
      LENGTH = 72 - FDPOS.
      ASSIGN WORD+FDPOS(LENGTH) TO <TAIL>.
      CLEAR <TAIL>.
      EXIT.
    ENDIF.
  ENDDO.
ENDFORM.

TOP-OF-PAGE.
SY-TVAR0 = PROGNAME.
```