

Veränderungen im Bestellwesen von SAP R/3 bis Release 4.5 – Problemstellung und Lösungsansatz

Von Dipl.-Betriebswirt Christoph Wildensee, Hannover¹

1. Einführung

Für einen Revisor ist die Bestell-/Kontraktabwicklung in SAP R/3 eines der zentralen Systeme zum Nachvollzug von Großprojekten. Diese müssen bei Überschreiten bestimmter Wertgrenzen meist Ausschreibungsmodalitäten durchlaufen, die oftmals nicht im Interesse der Fachbereiche liegen. Die Fachbereichsverantwortlichen/Baubeauftragten arbeiten seit je her mit wenigen Unternehmen zusammen und „scheuen oft die nicht erprobte Konkurrenz“. Entsprechend werden Großvorhaben im Rahmen der erlaubten, recht niedrigen Wertgrenze ausgeschrieben und nach Überschreiten dieser Ausschreibungsbeträge über Nachträge/Bestellwerterhöhungen ausgeweitet.

Als Revisor ist diese Praxis mehr als interessant, zumal hier erhebliche wirtschaftliche Nachteile in Form von geringeren Nachlassgewährungen und einer Erschwerung von Vergleich und Nachvollzug von Projekten angenommen werden können. Eine Analyse der Bestellabwicklung ist somit sinnvoll, um Rückschlüsse zu erhalten auf das Ausschreibungsverhalten der Fachbereiche und ggf. sogar bestimmter Mitarbeiter. Ziel ist es u.a.,

- Vorhaben zu Großprojekten zu bündeln
- Projekte umfassend in das SAP- oder angebundene Systeme (z.B. Sysbau) einzupflegen
- erheblich höhere Rabatte auszuhandeln
- und somit insgesamt den Nachvollzug von Projekten maßgeblich zu erhöhen.

SAP R/3 bietet den Funktionsumfang des Nachvollzugs von Bestellungen und Kontrakten. So ist es möglich, zum einen die Bestellentwicklung einzusehen, d.h. die zeitliche Abfolge der kaufmännischen Führung der Bestellung / des Kontraktes mit der Bestellauslösung, den Materialbuchungen bis hin zu den Rechnungen, Gutschriften etc. Zum anderen kann auch die Veränderung einer Bestellung / eines Kontraktes eingesehen werden, d.h., ob sich bestimmte Rahmenbedingungen verändert haben wie z.B. der Bestellwert, ausgeschriebene Mengen usw. Für den Revisor sind beide Sichten von immenser Bedeutung. Zum einen ist die zeitliche Abfolge aufschlussreich, um z.B. den Fertigungszeitraum (Beginn, Ende, Verzögerungen, zeitliche Überschneidungen mit anderen im Umkreis der Ausführung parallel befindlichen Vorhaben etc.) zu analysieren, zum anderen aber auch die Veränderung der Rahmenbedingungen, um zu überprüfen, ob Projekte generell/überwiegend unter Beschränkung oder sogar Ausschluss des Wettbewerbes ausgeschrieben werden und dies somit ggf. zu einem wirtschaftlichen Schaden für das Unternehmen führt.

2. SAP-Funktionalität

Das SAP R/3-System bietet zur Unterstützung dieser Analysen für den Revisor (und selbstverständlich auch für die Fachbereiche und Einkäufer) umfassende Funktionalitäten und Transaktionen an. Beispielhaft seien die folgenden beiden genannt:

2.1 Bestellentwicklung

Die Bestellentwicklung stellt die zeitliche Entwicklung der kaufmännischen Abbildung des Projektes in SAP R/3 dar. Über

1. Logistik -> Materialwirtschaft -> Einkauf
2. Bestellung anzeigen
3. <Bestellnummer> -> <Positionswahl>
4. Position -> Statistik -> Bestellentwicklung

¹ C. Wildensee ist bei der Stadtwerke Hannover AG als IV-Revisor tätig.

oder Transaktion ME23 kann z.B. die Belegentwicklung von Bestellungen eingesehen werden.

Über

1. Logistik -> Materialwirtschaft -> Einkauf
2. Rahmenvertrag -> Kontrakt anzeigen
3. <Kontraktnummer> -> <Positionswahl>
4. Position -> Statistik -> Abrufdokumentation

oder Transaktion ME33K kann z.B. die Belegentwicklung von Kontrakten eingesehen werden.

Für Kontrakte ist leider keine gleichzeitige Übersicht der Bestellentwicklung vorhanden, d.h. eine Übersicht über bereits gelaufene Rechnungen existiert nicht. Bei einem Kontrakt mit einer Vielzahl von sich auf diesen Kontrakt beziehenden Bestellungen ist also der Nachvollzug durch sehr viele Einzelschritte erschwert, da eine Gesamtübersicht über diese sich auf ihn beziehenden Bestellungen und den hierauf wiederum beziehenden Rechnungen nicht existiert. Sinnvoll kann hier die Entwicklung eines Reports sein, der die zum Kontrakt bestehenden Bestellungen einschließlich der hierauf bereits gelaufenen Rechnungsbelege zeigt.

2.2 Veränderungsbelege

Die Darstellung der Veränderungen zu einer Bestellung bzw. zu einem Kontrakt erfolgt wie zuvor beschrieben, jedoch im letzten Schritt mit der Unterscheidung

4. Position -> Statistik -> Änderungen für Bestellungen und Kontrakte.

Die Darstellung dieser Veränderungsbelege liegt nicht in der Form vor, wie dies für eine Weiterverarbeitung sinnvoll ist. Es fehlt die Möglichkeit der

- Mehrfacheingrenzung von Bestellungen
- Eingrenzung von Bestellungen über Auswahl eines Lieferanten innerhalb eines Zeithorizontes mit Gruppenwechsel
- Tabellendarstellung, um die Daten weiter zu verarbeiten, z.B. in Excel.

Transaktion ME23 - Änderungsbeleg zur Bestellung

SachbearbeiterIn	Datum	Uhrzeit	Transaktion	Änderungsbeleg
Belegkopf	<Bestellnummer>			
Position	<Positionsnummer>			
ÄNDERERNAME	15.08.2001	12.48.20	ME22	<int-Änd.nummer>
AEDAT	Änderungsbelegdatum der Einkaufsbelegposition geändert			
alt:	15.04.2000			
neu:	15.08.2001			
BRTWR	Bruttobestellwert in Bestellwährung geändert			
alt:	500.000,00 DEM			
neu:	800.000,00 DEM			
EFFWR	Effektivwert der Position geändert			
alt:	500.000,00 DEM			
neu:	800.000,00 DEM			
MATKL	Warengruppe geändert			
alt:	924104			
neu:	DTEAUS			
MENGE	Bestellmenge geändert			
alt:	500.000,000			
neu:	800.000,000			
NETWR	Nettobestellwert in Bestellwährung geändert			
alt:	500.000,00 DEM			
neu:	800.000,00 DEM			
Einteilung	<Positionsnummer>			
.....				

Abb.1: Änderungsbeleg zur Bestellung

Veränderungsbelege werden in den Tabellen CDHDR und CDPOS gespeichert. Um eine tabellarische Darstellung der relevanten Datensätze zu erhalten, ist es auch möglich, über die Transaktion SE11 – Dictionary: Einstieg – die Tabellen direkt aufzurufen, eine Bestellnummereingrenzung vorzunehmen (-> Tabelleninhalt -> Änderungsbelegobjekt EINKBELEG -> Objektwert = Bestell-/Kontraktnummer, auch Mehrfachselektion möglich; int. Änderungsbelegnummer siehe z.B. Transaktion ME23) und die Inhalte in tabellarischer Form auszugeben.

Dabei tritt ein SAP-Fehler auf, der bewirkt, dass bei der Anzeige der Felder VALUE_NEW und VALUE_OLD die entsprechenden Werte nicht dargestellt werden, sofern sie sich auf Mengen und Beträge beziehen (Currency-Key-Felder [CUKY]).

Tabelle CDPOS anzeigen über SE11 -> Tabelleninhalt

Änderungsbelegobjekt		EINKBELEG
Objektwert		<Bestellnummer>
Belegnummer		<int. Änderungsbelegnummer>
Tabellenname		EKPO
Tab.Schlüssel		<Mandant+Bestell-Nr.+Best.-Pos.>
Feldname		BRTWR
Änderungskz.		U
Textkz.		1
Einheit	(UNIT OLD)	
Einheit	(UNIT NEW)	
CUKY	(CUKY OLD)	DEM
CUKY	(CUKY NEW)	DEM
Wert	(VALUE OLD)	
Wert	(VALUE NEW)	

Hier müssen eigentlich die geänderten Betragswerte stehen !

Abb. 2: Tabelle CDPOS anzeigen

Wie sich gezeigt hat, ist eine Analyse des ABAP-Codes ausgesprochen schwierig. Es wird eine Vielzahl an Strukturen und Funktionsbausteinen genutzt, so dass eine direkte Korrektur der beteiligten Programmteile im eigenen Kundennamensraum nahezu ausgeschlossen scheint. In Release 4.6 ist bereits der List Viewer integriert und die Änderungsbelege werden tabellarisch angezeigt, hier werden auch die Werte vom Typ CURR korrekt dargestellt. Sie können also gleich in Excel weiterverarbeitet werden. Bis Release 4.6 ist dies jedoch nicht ohne weiteres möglich.

3. Lösungsansatz

3.1 Optimierung bei der Bestellentwicklung

Das hier vorliegende Problem ist ein durchgängiges Missverhältnis des SAP-Systems zur Rolle des Revisors. Es steht zwar eine Vielzahl von Reports zur Verfügung, die auch vom Revisor genutzt werden können, jedoch eher selten sind diese Reports auch sinnvoll einsetzbar, wenn es darum geht, über einen größeren Datenbestand (z.B. über einen kompletten Nummernkreis, einer Bestelleingrenzung über einen längeren Zeitraum [von..bis..-Datumseingrenzung] o.ä.) Auswertungen zu fahren. Das SAP-System sieht hier grundsätzlich nur den funktionsbezogenen Sachbearbeiter und nur unzureichend und selten die prüfende Instanz.

Entsprechend kommt es viel zu oft vor, dass die interne Revision eine spezifische Anforderung formuliert, die als Report umgesetzt werden muss. Nach dem Grundsatz der Sparsamkeit unternehmensspezifischer Anpassungen sollte dies ja gerade nicht zu oft vorkommen. Ein Gegensatz, der zu Spannungen führen kann.

Die Darstellung eines oder mehrerer Kontrakte, den zugehörigen Bestellungen und den hierauf gelaufenen Rechnungsstellungen ist eine Möglichkeit, schnell und umfassend auf relevante Informationen zuzugreifen. Der Report muss dabei folgende Anforderungen erfüllen:

- Eingrenzung eines oder mehrerer Kontrakte
- Belegauswahl für Kontrakte, Bestellungen, Lieferpläne und Anfragen, Möglichkeit des Ausblendens von Teilen auf Wunsch
- Ausgabe über strukturierte Liste, möglichst über feste Breite definiert, um im Nachhinein auch eine Excel-Konformität zu erzielen
- Felder-Auswahl (beispielhaft)
 - Einkaufsorganisation
 - Kontrakt + Position
 - Lieferant
 - Bestellung + Position, Bestelldatum, Bestellmenge, Bestellwert
 - Rechnungsnummer, Rechnungswert, Rechnungsdatum
 - Summierung auf Bestell- und Kontraktebene incl. prozentualer Gesamtabdeckung des Kontraktes

In der Anlage ist der Quellcode eines Reports dargestellt, der die hier vorliegenden Kriterien erfüllt.

3.2 Optimierung bei Veränderungsbelegen

Der hier vorliegende Fehler bei der Darstellung von Veränderungsbelegen kann durch eine Anpassung des Basis-Reports RSSCD150 umgangen werden. Die Anpassung des Reports RSSCD150 im eigenen Kundennamensraum kann z.B. wie folgt aussehen:

Die Ausgabeliste soll größer sein als bisher, d.h. die Reportdefinition muss eine Verbreiterung der Liste beinhalten. Zusätzlich erfolgt der Zugriff auch auf die Einkaufsbelege.

```
REPORT ZRSSCD150 MESSAGE-ID CD NO STANDARD PAGE HEADING LINE-SIZE 255.  
TABLES: EKKO, CDHDR.
```

Die Eingrenzungsdefinition sieht dann z.B. – abweichend zum Original – wie folgt aus:

```
PARAMETERS: OBJECT          LIKE  CDHDR-OBJECTCLAS NO-DISPLAY,  
              OBJECTID      LIKE  CDHDR-OBJECTID,  
              AENDERER      LIKE  CDHDR-USERNAME,  
              DATUM         LIKE  CDHDR-UDATE,  
              ZEIT          LIKE  CDHDR-UTIME,  
              NUMMER        LIKE  CDHDR-CHANGENR  DEFAULT ' ',  
              TABNAME       LIKE  CDPOS-TABNAME   DEFAULT ' ',  
              TABKEY        LIKE  CDPOS-TABKEY    DEFAULT ' '.  
  
SELECTION-SCREEN SKIP.  
SELECTION-SCREEN BEGIN OF BLOCK BL1 WITH FRAME TITLE TEXT-999. (Eingrenzung)  
  SELECT-OPTIONS:          H_LIFNR FOR EKKO-LIFNR,  „Kreditorennummer  
                          H_EBELN FOR EKKO-EBELN,  „Belegnr. Eink.beleg  
                          H_BEDAT FOR EKKO-BEDAT.  „Datum d.Eink.belegs  
SELECTION-SCREEN END OF BLOCK BL1.
```

Bei der Möglichkeit der Mehrfachselektion müssen die internen Tabellen, die die Daten aufnehmen, entsprechend dimensioniert sein. Dieses führt zu einer Ausweitung der Tabellen AUSG und ICDHDR mit dem Zusatz DATA: BEGIN OF AUSG OCCURS 1000 und entsprechend DATA: BEGIN OF ICDHDR OCCURS 1000.

Im Bereich INITIALIZATION erfolgt nach dem korrekten AUTHORITY-CHECK-Block die GET/SET-Definition. Da hier die Eingrenzung auf Einkaufsbelege das Ziel ist, ist das feste Hinterlegen von OBJECT = 'EINKBELEG' sinnvoll. Entsprechend wird der folgende Bereich ausgetauscht.

Original:

```
IF SY-SUBRC NE 0.  
  MESSAGE S800.  
  LEAVE.  
ENDIF.
```

```
GET PARAMETER ID 'CDO' FIELD OBJECT.
```

```
IF OBJECT = SPACE.  
  OBJECT = 'BANF'.  
  SET PARAMETER ID 'CDO' FIELD OBJECT.  
  DATUM = SY-DATUM.  
  SET PARAMETER ID 'CDD' FIELD DATUM.  
  ZEIT = '000000'.  
  SET PARAMETER ID 'CDZ' FIELD ZEIT.  
ELSE.
```

Anpassung:

```
IF SY-SUBRC NE 0.  
  MESSAGE S800.  
  LEAVE.  
ENDIF.
```

```
OBJECT = 'EINKBELEG'.  
SET PARAMETER ID 'CDO' FIELD OBJECT.  
GET PARAMETER ID 'CDO' FIELD OBJECT.  
DATUM = SY-DATUM.  
SET PARAMETER ID 'CDD' FIELD DATUM.  
GET PARAMETER ID 'CDD' FIELD DATUM.  
IF OBJECT = SPACE.  
  * OBJECT = 'BANF'.  
  * SET PARAMETER ID 'CDO' FIELD OBJECT.  
  * DATUM = SY-DATUM.  
  * SET PARAMETER ID 'CDD' FIELD DATUM.  
  ZEIT = '000000'.  
  SET PARAMETER ID 'CDZ' FIELD ZEIT.  
ELSE.
```

Der größte Teil des Bereiches START-OF-SELECTION bleibt identisch, jedoch nach dem Funktionsbausteinanruf (CALL FUNCTION 'CHANGEDOCUMENT_READ_HEADERS') erfolgt innerhalb des Auswertens des Returncodes (SY-SUBRC) eine Änderung.

Original:

```
CASE SY-SUBRC.  
WHEN '0000'.  
  IF NUMMER NE SPACE.  
  LOOP AT ICDHDR WHERE  
    CHANGENR NE NUMMER.  
  DELETE ICDHDR.  
  ENDLOOP.  
ENDIF.
```

```
LOOP AT ICDHDR.  
  LIST = SPACE.  
  PERFORM UEBERSCHRIFT_HEADER.  
  LIST = 'H'.  
  PERFORM AUSGABE_HEADER.  
  SKIP.  
  LIST = SPACE.  
  DETAIL.  
  PERFORM LESE_UND_AUSG_POSITIONEN.  
  SKIP.  
  SUMMARY.  
  ULINE.  
  SKIP.  
ENDLOOP.
```

Anpassung:

```
CASE SY-SUBRC.  
WHEN '0000'.  
  IF NUMMER NE SPACE.  
  LOOP AT ICDHDR WHERE  
    CHANGENR NE NUMMER.  
  DELETE ICDHDR.  
  ENDLOOP.  
ENDIF.  
PERFORM AUSGABE_HEADER_EXCEL.  
LOOP AT ICDHDR.  
  SELECT SINGLE * FROM EKKO  
    WHERE EBELN EQ ICDHDR-OBJECTID.  
  IF SY-SUBRC NE 0.  
    CLEAR EKKO.  
  ENDIF.  
  CHECK EKKO-BSTYP EQ 'F' OR  
    EKKO-BSTYP EQ 'K'.  
  CHECK EKKO-EBELN IN H_EBELN.  
  CHECK EKKO-BEDAT IN H_BEDAT.  
  CHECK EKKO-LIFNR IN H_LIFNR.  
  PERFORM LESE_UND_AUSG_POSITIONEN.  
ENDLOOP.
```

Hier sieht man bereits, dass die ursprünglichen Ausgabebereiche zugunsten neuer Unterprogramme nicht mehr benötigt werden, da die neuen Bereiche Excel-konforme Ausgaben erzeugen (Ausgabe_Header_Excel und später Ausg_Positionen_Excel). Zusätzlich erfolgt hier die Überprüfung auf korrekte Belege (z.B. Bestelltyp 'F' und 'K') und die Auswahl innerhalb der Belegdatum- oder Lieferanteneingrenzung.

Original:

```
TOP-OF_PAGE.  
CASE LIST.  
  WHEN 'H'.  
    PERFORM UEBERSCHRIFT_HEADER.  
ENDCASE.
```

Anpassung:

```
TOP-OF_PAGE.  
CASE LIST.  
  WHEN 'H'.  
    * PERFORM UEBERSCHRIFT_HEADER.  
ENDCASE.
```

Hier erfolgt das Herausnehmen der ursprünglichen Header-Überschrift.

Original:

```
FORM LESE_UND_AUSG_POSITIONEN.  
NUMMER = ICDHDR-CHANGENR.  
CALL FUNCTION  
'CHANGEDOCUMENT_READ_POSITIONS'  
...  
CASE SY-SUBRC.  
  WHEN '0000'.  
    PERFORM AUSGABE_POSITIONEN.  
  
  WHEN '0001'.  
    WRITE: 'Keine Positionen gefunden' (101).
```

Anpassung:

```
FORM LESE_UND_AUSG_POSITIONEN.  
NUMMER = ICDHDR-CHANGENR.  
CALL FUNCTION  
'CHANGEDOCUMENT_READ_POSITIONS'  
...  
CASE SY-SUBRC.  
  WHEN '0000'.  
    * PERFORM AUSGABE_POSITIONEN.  
    PERFORM AUSG_POSITIONEN_EXCEL.  
  WHEN '0001'.  
    WRITE: 'Keine Positionen gefunden' (101).
```

Die Ausgabe erfolgt nach dem Lesen der Positionen nicht mehr im Unterprogramm Ausgabe_Positionen, sondern im speziell aufbereiteten Excel-Ausgabe-Bereich. Dieser wird neu aufgebaut.

Anpassung:

```
FORM AUSGABE_HEADER_EXCEL.  
WRITE: / 01  'Änderungsnummer des Belegs' NO-GAP,  ';;' NO-GAP,  
            'Änderungsdatum' NO-GAP,             ';;' NO-GAP,  
            'Objektwert' NO-GAP,                  ';;' NO-GAP,  
            'Art der Änderung: U, I, E, D' NO-GAP, ';;' NO-GAP,  
            'Art der Änderung' NO-GAP,           ';;' NO-GAP,  
            'Feldname' NO-GAP,                   ';;' NO-GAP,  
            'Alter Feldinhalt' NO-GAP,           ';;' NO-GAP,  
            'Neuer Feldinhalt' NO-GAP,           ';;' NO-GAP.  
ENDFORM.
```

```
FORM AUSG_POSITIONEN_EXCEL.  
  SORT AUSG BY TABNAME TABKEY CHNGIND FNAME.  
  LOOP AT AUSG.  
    CASE AUSG-CHNGIND.  
      WHEN 'U'.  
        IF AUSG-TEXT_CASE NE SPACE.  
          INDTEXT = 'Textänderung' (501).  
        ELSE.  
          INDTEXT = 'Änderung' (502).  
        ENDIF.  
      WHEN 'D'.  
        IF AUSG-TEXT_CASE NE SPACE.  
          INDTEXT = 'Textlöschung' (503).  
        ELSE.  
          INDTEXT = 'Löschung' (504).  
        ENDIF.  
      WHEN 'E'.  
        INDTEXT = 'Löschung' (505).  
      WHEN 'I'.  
        IF AUSG-TEXT_CASE NE SPACE.  
          INDTEXT = 'Texterfassung' (506).  
        ELSE.
```

```
                INDTEXT = 'Erfassung' (507).
            ENDIF.
        WHEN 'M'.
            INDTEXT = 'Feld ist nicht in Tabelle' (508).
        ENDCASE.
        WRITE: /01  ICDHDR-CHANGENR NO-GAP,      ';; NO-GAP,
                ICDHDR-UDATE DDMMYY NO-GAP    ';; NO-GAP,
                ICDHDR-OBJECTID (20) NO-GAP    ';; NO-GAP,
                AUSG-CHNGIND NO-GAP            ';; NO-GAP,
                INDTEXT (21) NO-GAP            ';; NO-GAP,
                AUSG-FNAME (10) NO-GAP         ';; NO-GAP,
                AUSG-F_OLD (30) NO-GAP         ';; NO-GAP,
                AUSG-F_NEW (30) NO-GAP         ';; NO-GAP.
    ENDLOOP.
ENDFORM.
--- Ende des Reports ---
```

Dieser letzte Block wertet die Klassifizierung der Veränderungen aus und gibt die Veränderungen in einem Excel-konformen Format aus.

Die Flexibilität dieser vorliegenden Anpassung bewirkt, dass weitere Felder in die Ausgabe-Definitionen einbezogen werden können. Entsprechend können die Änderungen zu Bestellungen je nach Bedarf schnell weiterverarbeitet werden. Die Excel-konforme Ausgabe und der ggf. spätere Datenbankaufbau – sofern vom Volumen her sinnvoll – bedeuten eine Optimierung der Weiterverarbeitungsmöglichkeit. Als Grundvoraussetzung ist hier allerdings der Zugriff auf das Berechtigungsobjekt ‚S_GUI‘ mit Aktivität 61 zu nennen.

4. Abschlussbetrachtung

Die beiden dargestellten Problemfelder können mit nur wenigen Anpassungen des SAP-Systems aufgefangen werden. Solche Modifikationen sind zwar grundsätzlich nicht oder nur selten gewünscht, können jedoch zu einer Optimierung der Prüfhandlungen führen. Sinnvoll kann auch die Separierung der Bestelltexte des SAP-Systems, die Excel-formatige Ausgabe und das Füllen einer Datenbank sein, um nach Schlüsselbegriffen zu suchen und eine Optimierung von Stichproben herbeizuführen. Letztendlich geht es ja bei all den Bemühungen um eine Optimierung der Ergebnisermittlung und –darstellung von Revisionsarbeit.

www.wildensee.de/veroeff.htm

Mit besonderem Dank an Dipl.-Ing. Joachim Bosmann, technischer Revisor bei der Stadtwerke Hannover AG, für die Hinweisgebung.

Anlage:

* Report ZMMKONTR01 - Kontrakt-Liste mit Rechnungswerten zu den Bestellungen *

REPORT ZMMKONTR01 LINE-SIZE 132 LINE-COUNT 65.

TABLES:	T001,	"Buchungskreise
	LFA1,	"Lieferantenstamm (allg. Teil)
	EKKO,	"Einkaufsbelegkopf
	EKPO,	"Einkaufsbelegposition
	EKAB,	"Abrufdokumentation
	EKBE.	"Bestellentwicklung
DATA: BEGIN OF TAB1 OCCURS 0,		"Bestellwerte EKKO,EKPO,EKAB
EKORG	LIKE EKKO-EKORG,	"Einkaufsorganisation
KONNR	LIKE EKAB-KONNR,	"Rahmenvertragsnummer
KTPNR	LIKE EKAB-KTPNR,	"Rahmenvertragsposition
BSART	LIKE EKKO-BSART,	"Vertragsart
MENGE	LIKE EKAB-MENGE,	"abgerufene Positionsmenge
NETWR	LIKE EKAB-NETWR,	"Pos.-wert in HW umgerechnet
BEDAT	LIKE EKAB-BEDAT,	"Belegdatum
EBELN	LIKE EKPO-EBELN,	"Abruf-Belegnummer
EBELP	LIKE EKPO-EBELP,	"Abruf-Belegposition
LIFNR	LIKE EKKO-LIFNR,	"Lieferant
KTMNG	LIKE EKPO-KTMNG,	"Zielmenge
END OF TAB1.		
DATA: BEGIN OF TAB2 OCCURS 0,		"Rechnungswerte EKBE
EKORG	LIKE EKKO-EKORG,	"Einkaufsorganisation
KONNR	LIKE EKAB-KONNR,	"Rahmenvertragsnummer
KTPNR	LIKE EKAB-KTPNR,	"Rahmenvertragsposition
EBELN	LIKE EKPO-EBELN,	"Abruf-Belegnummer
EBELP	LIKE EKPO-EBELP,	"Abruf-Belegposition
BELNR	LIKE EKBE-BELNR,	"Rechnungsnummer
BUDAT	LIKE EKBE-BUDAT,	"Buchungsdatum der Rechnung
DMBTR	LIKE EKBE-DMBTR,	"Betrag immer in Hauswahrung EUR
SHKZG	LIKE EKBE-SHKZG,	"Soll/Haben-Kennz.
END OF TAB2.		
DATA: OMENG	LIKE EKPO-MENGE,	"Offene Menge
OWERT	LIKE EKPO-NETWR,	"Offener Wert
REFE	LIKE EKPO-NETWR,	"Hilfsfeld
NETWRK	LIKE EKPO-NETWR,	"abgerufener Gesamtwert
PROZE(3)	TYPE P DECIMALS 1,	"Erfullung in Prozent
CHAR7(7)	TYPE C,	"Druck-Feld Prozentwert
KTAZAK	TYPE I,	"Anzahl Abrufe Kopf
GESCHRIEBEN(1)	TYPE C VALUE 'N',	"Satz geschrieben
ERSTER(1)	TYPE C VALUE 'J',	"fur Lesen T001
ERST1(1)	TYPE C VALUE 'J',	"fur Lesen TAB2
H_WKURS(8)	TYPE C,	"Hilfsfeld Kurs (CHAR)
DAT1(8)	TYPE C,	"Hilfsfeld Datum
DAT2(8)	TYPE C,	"Hilfsfeld Datum
KONNR_ALT	LIKE EKAB-KONNR,	"fur Gruppenwechsel
KONNR_NEU	LIKE EKAB-KONNR,	"fur Gruppenwechsel
KTPNR_ALT	LIKE TAB1-KTPNR,	"fur Gruppenwechsel
KTPNR_NEU	LIKE TAB1-KTPNR,	"fur Gruppenwechsel

EBELN_ALT	LIKE TAB1-EBELN,	"für Gruppenwechsel
EBELN_NEU	LIKE TAB1-EBELN,	"für Gruppenwechsel
EBELP_ALT	LIKE TAB1-EBELP,	"für Gruppenwechsel
EBELP_NEU	LIKE TAB1-EBELP,	"für Gruppenwechsel
SR_NETWR	LIKE TAB1-NETWR,	"Summe Bestellwert Rechn.-zeilen
SR_DMBTR	LIKE TAB2-DMBTR,	"Summe Rechnungswert Rechn.-zeilen
SK_NETWR	LIKE TAB1-NETWR,	"Summe Bestellwert pro Kontrakt
SK_DMBTR	LIKE TAB2-DMBTR,	"Summe Rechnungswert pro Kontrakt
ZAEHLR(4)	TYPE P VALUE 0.	"Zähler gedruckte Rechnungszeilen

START-OF-SELECTION.

GET EKKO.

CHECK SELECT-OPTIONS.	"Prüfen Selektionsbedingungen
CHECK EKKO-BSTYP EQ 'K'.	"nur Kontrakte
CHECK EKKO-LOEKZ EQ SPACE.	"keine Gelöschten
GESCHRIEBEN = 'N'.	

* umrechnen Abrufbetrag EKAB-NETWR in Hauswährung

IF ERSTER = 'J'.	" Hauswährung nur einmal ermitteln
ERSTER = 'N'.	
SELECT SINGLE * FROM T001	
WHERE BUKRS = EKKO-BUKRS.	
ENDIF.	

GET EKPO.

CHECK SELECT-OPTIONS.	"Prüfen Selektionsbedingungen
CHECK EKPO-LOEKZ EQ SPACE.	"keine Gelöschten

SELECT * FROM EKAB
WHERE KONNR EQ EKPO-EBELN
AND KTPNR EQ EKPO-EBELP
AND LOEKZ EQ SPACE.

GESCHRIEBEN = 'J'.

CLEAR TAB1.

MOVE EKKO-EKORG	TO TAB1-EKORG.
MOVE EKKO-EBELN	TO TAB1-KONNR.
MOVE EKKO-BSART	TO TAB1-BSART.
MOVE EKKO-LIFNR	TO TAB1-LIFNR.
MOVE EKPO-EBELP	TO TAB1-KTPNR.
MOVE EKAB-EBELN	TO TAB1-EBELN.
MOVE EKAB-EBELP	TO TAB1-EBELP.
MOVE EKAB-MENGE	TO TAB1-MENGE.

IF EKAB-WAERS = 'DEM'

AND EKAB-WKURS = 0.

"Kurs fehlt
"DEM in EUR umrechnen

MOVE '51.12919'	TO H_WKURS.
MOVE H_WKURS	TO EKAB-WKURS.

ENDIF.

* umrechnen in Hauswährung

IF EKAB-WAERS NE T001-WAERS	" T001-WAERS ist Hauswährung
AND EKAB-NETWR NE 0.	

PERFORM UMRECHNEN_FREMDW_IN_HAUSW.

ENDIF.

MOVE EKAB-NETWR	TO TAB1-NETWR.
MOVE EKAB-BEDAT	TO TAB1-BEDAT.
APPEND TAB1.	

```
SELECT * FROM EKBE                                "zugehörige Rechnungen
  WHERE EBELN EQ EKAB-EBELN
     AND EBELP EQ EKAB-EBELP
     AND ( BEWTP EQ 'N'                            "NB-L
          OR BEWTP EQ 'P'                            "NB-B
          OR BEWTP EQ 'Q'                            "Re-L
          OR BEWTP EQ 'R'                            "Re
          OR BEWTP EQ 'S'                            "SPED
          OR BEWTP EQ 'X' ).                        "NB

CLEAR TAB2.
MOVE EKKO-EKORG      TO TAB2-EKORG.
MOVE EKKO-EBELN     TO TAB2-KONNR.
MOVE EKPO-EBELP     TO TAB2-KTPNR.
MOVE EKAB-EBELN     TO TAB2-EBELN.
MOVE EKAB-EBELP     TO TAB2-EBELP.
MOVE EKBE-BELNR     TO TAB2-BELNR.
MOVE EKBE-BUDAT     TO TAB2-BUDAT.

IF EKBE-SHKZG EQ 'H'.
  EKBE-DMBTR = EKBE-DMBTR * -1.
ENDIF.
MOVE EKBE-DMBTR     TO TAB2-DMBTR.
MOVE EKBE-SHKZG     TO TAB2-SHKZG.
APPEND TAB2.
ENDSELECT.
ENDSELECT.

GET EKPO LATE.
IF GESCHRIEBEN = 'N'.                                "nur Pos. ohne Abrufe
  GESCHRIEBEN = 'J'.
  CLEAR TAB1.
  MOVE EKKO-EKORG      TO TAB1-EKORG.
  MOVE EKKO-EBELN     TO TAB1-KONNR.
  MOVE EKKO-BSART     TO TAB1-BSART.
  MOVE EKKO-LIFNR     TO TAB1-LIFNR.
  MOVE EKPO-EBELP     TO TAB1-KTPNR.
  APPEND TAB1.
ENDIF.

GET EKKO LATE.

IF GESCHRIEBEN = 'N'.                                "nur Kopfinformationen vorh.
  GESCHRIEBEN = 'J'.
  CLEAR TAB1.
  MOVE EKKO-EKORG      TO TAB1-EKORG.
  MOVE EKKO-EBELN     TO TAB1-KONNR.
  MOVE EKKO-BSART     TO TAB1-BSART.
  APPEND TAB1.
ENDIF.

END-OF-SELECTION.
```

```
LOOP AT TAB1.
  MOVE TAB1-KONNR      TO KONNR_NEU.
  MOVE TAB1-KTPNR     TO KTPNR_NEU.
  MOVE TAB1-EBELN     TO EBELN_NEU.
  MOVE TAB1-EBELP     TO EBELP_NEU.

  IF KONNR_ALT = SPACE.                    " 1. Satz
    MOVE KONNR_NEU    TO KONNR_ALT.
    MOVE KTPNR_NEU    TO KTPNR_ALT.
    MOVE EBELN_NEU    TO EBELN_ALT.
    MOVE EBELP_NEU    TO EBELP_ALT.
  ENDIF.

  IF KONNR_NEU = KONNR_ALT                  "Bestell-/Rechnungssumme
  AND KTPNR_NEU = KTPNR_ALT
  AND EBELN_NEU = EBELN_ALT
  AND EBELP_NEU = EBELP_ALT.
  ELSE.
    IF ZAEHLR GT 1.
      WRITE: /83 '-----', 111 '-----'.
      WRITE: /45 'Summe Bestellpos.', 78 SR_NETWR, '*',
            106 SR_DMBTR, '*'.
    ENDIF.
    ZAEHLR = 0.
    SR_NETWR = 0.
    SR_DMBTR = 0.
  ENDIF.

  IF KONNR_NEU NE KONNR_ALT.                "Kontraktsumme
  IF SK_NETWR NE 0
  OR SK_DMBTR NE 0.
*   Erfüllungsgrad in % errechnen (Bestellwert zu Rechnungswert)
*   Druck in CHAR-Feld
  MOVE SPACE TO CHAR7.
  IF SK_NETWR NE 0.
    REFE = SK_DMBTR * 100 / SK_NETWR.
  ELSE.
    MOVE 0 TO REFE.
  ENDIF.
  MOVE REFE TO PROZE.
  MOVE PROZE TO CHAR7.
  SHIFT CHAR7 LEFT.
  REPLACE ' ' WITH ',' INTO CHAR7.
  MOVE '%' TO CHAR7+6(1).
  WRITE: /45 'Summe Kontrakt', 78 SK_NETWR, '**',
        106 SK_DMBTR, '**', CHAR7.
  ENDIF.
  SK_NETWR = 0.
  SK_DMBTR = 0.
  ULINE.
  ENDIF.

  ERST1 = 'J'.
  MOVE TAB1-BEDAT+6(2) TO DAT1+0(2).  "TT
  MOVE ' ' TO DAT1+2(1).
  MOVE TAB1-BEDAT+4(2) TO DAT1+3(2).  "MM
  MOVE ' ' TO DAT1+5(1).
```

```
MOVE TAB1-BEDAT+2(2) TO DAT1+6(2).  "JJ
WRITE: / TAB1-EKORG, TAB1-KONNR, TAB1-KTPNR, TAB1-BSART(2),
      TAB1-LIFNR+3(7), TAB1-EBELN, TAB1-EBELP, DAT1,
      TAB1-MENGE, TAB1-NETWR.
SR_NETWR = SR_NETWR + TAB1-NETWR.
SK_NETWR = SK_NETWR + TAB1-NETWR.
```

```
LOOP AT TAB2.                                "Rechnungswerte zu Abrufen
  IF TAB1-EKORG = TAB2-EKORG
  AND TAB1-KONNR = TAB2-KONNR
  AND TAB1-KTPNR = TAB2-KTPNR
  AND TAB1-EBELN = TAB2-EBELN
  AND TAB1-EBELP = TAB2-EBELP.
  MOVE TAB2-BUDAT+6(2) TO DAT2+0(2).  "TT
  MOVE '.' TO DAT2+2(1).
  MOVE TAB2-BUDAT+4(2) TO DAT2+3(2).  "MM
  MOVE '.' TO DAT2+5(1).
  MOVE TAB2-BUDAT+2(2) TO DAT2+6(2).  "JJ
  IF ERST1 = 'J'.
    ERST1 = 'N'.
    WRITE: TAB2-BELNR, TAB2-DMBTR, DAT2.  "gleiche Zeile
    ZAEHLR = ZAEHLR + 1.                "Zähler gedruckte Rechnungszeilen
    SR_DMBTR = SR_DMBTR + TAB2-DMBTR.
    SK_DMBTR = SK_DMBTR + TAB2-DMBTR.
  ELSE.
    WRITE: /95 TAB2-BELNR, TAB2-DMBTR, DAT2.  "Zähler gedruckte Rechnungszeilen
    ZAEHLR = ZAEHLR + 1.
    SR_DMBTR = SR_DMBTR + TAB2-DMBTR.
    SK_DMBTR = SK_DMBTR + TAB2-DMBTR.
  ENDIF.
ENDIF.
ENDLOOP.
MOVE KONNR_NEU TO KONNR_ALT.
MOVE KTPNR_NEU TO KTPNR_ALT.
MOVE EBELN_NEU TO EBELN_ALT.
MOVE EBELP_NEU TO EBELP_ALT.
ENDLOOP.
```

```
IF SK_NETWR NE 0
OR SK_DMBTR NE 0.
* Erfüllungsgrad in % errechnen (Bestellwert zu Rechnungswert)
* Druck in CHAR-Feld
MOVE SPACE TO CHAR7.
IF SK_NETWR NE 0.
  REFE = SK_DMBTR * 100 / SK_NETWR.
ELSE.
  MOVE 0 TO REFE.
ENDIF.
MOVE REFE TO PROZE.
MOVE PROZE TO CHAR7.
SHIFT CHAR7 LEFT.
REPLACE '.' WITH ',' INTO CHAR7.
MOVE '%' TO CHAR7+6(1).
WRITE: /45 'Summe Kontrakt', 78 SK_NETWR, '***',
      106 SK_DMBTR, '***', CHAR7.
ENDIF.
SK_NETWR = 0.
```

SK_DMBTR = 0.

ULINE.

" nach letztem Satz

```
*-----*
* FORM UMRECHNEN_FREMDW_IN_HAUSW *
* der Abrufwert wird in Hauswahrung umgerechnet *
*-----*
```

FORM UMRECHNEN_FREMDW_IN_HAUSW.

CALL FUNCTION 'CONVERT_TO_LOCAL_CURRENCY'

EXPORTING

```
* CLIENT = SY-MANDT
  DATE = EKKO-BEDAT
  FOREIGN_AMOUNT = EKAB-NETWR
  FOREIGN_CURRENCY = EKAB-WAERS
  LOCAL_CURRENCY = T001-WAERS
  RATE = EKAB-WKURS
  TYPE_OF_RATE = 'M'
  READ_TCURR = 'X'
```

" Betrag in Fremdwahrung

IMPORTING

```
* EXCHANGE_RATE =
* FOREIGN_FACTOR =
  LOCAL_AMOUNT = EKAB-NETWR
* LOCAL_FACTOR =
* EXCHANGE_RATEX =
* FIXED_RATE =
* DERIVED_RATE_TYPE =
```

" Betrag in HW umrechnen

EXCEPTIONS

```
NO_RATE_FOUND = 1
OVERFLOW = 2
NO_FACTORS_FOUND = 3
NO_SPREAD_FOUND = 4
DERIVED_2_TIMES = 5
OTHERS = 6
```

IF SY-SUBRC <> 0.

WRITE: / 'Fehler bei Wahrungsumrechnung, SY-SUBRC', SY-SUBRC,
'Bestell-Nr./Pos.', EKAB-EBELN, EKAB-EBELP.

ENDIF.

ENDFORM.

--- Ende des Reports ---