

---

# 1 Einleitung

Dieses Handbuch beschreibt den Funktionsumfang und die Arbeitsweise des Programms PERCON.

## 1.1 Kurzbeschreibung des Produkts

PERCON (PERipheral CONverter) ist ein Programm zum Übertragen und Umsetzen von Daten auf beliebige Datenträger.

PERCON V2.6A ist ablauffähig ab BS2000 Version 10.0A.

PERCON bietet folgende Funktionen für den Dialog- und Stapelbetrieb an:

- Kopieren von Dateien; Ändern von Dateimerkmale
- Editieren von Bandinhalten nach SYSLST, nach SYSOUT oder in katalogisierte Dateien
- Duplizieren von Bändern bzw. Magnetbandkassetten
- Auswählen von Sätzen bzw. Blöcken
- Umstrukturieren von Sätzen
- Aufbereiten von Dateien oder Bandinhalten
- Bilden von Gruppen
- Aufrufen als Unterprogramm
- Anschließen von Benutzerrouinen zur Satz- oder Kennsatzbearbeitung oder zur Fehlerbehandlung

## 1.2 Zielsetzung und Zielgruppen des Handbuchs

Dieses Handbuch wendet sich an BS2000-Anwender, die ihre Daten mit Hilfe von PERCON konvertieren möchten.

Sie sollten über BS2000-Kenntnisse verfügen und mit den wichtigsten Kommandos vertraut sein. Nähere Informationen finden Sie in den Handbüchern „Kommandos“ Band 1 - 4 [11].

Zudem sollten Sie mit der BS2000-Kommandosprache SDF (System Dialog Facility) vertraut sein, da die Benutzeroberfläche, Bildschirmgestaltung und Benutzerführung von PERCON durch SDF bestimmt werden (siehe Handbuch „Einführung in die Dialogschnittstelle SDF“ [8]).

Ergänzend zu dem hier vorliegenden Handbuch gibt es das Tabellenheft „PERCON“ [5]. Es enthält eine Zusammenstellung aller PERCON-Anweisungen einschließlich ihrer Syntax. Das Tabellenheft ist als Nachschlagewerk für den mit PERCON bereits vertrauten Benutzer gedacht.

## 1.3 Konzept des Handbuchs

Das Handbuch besteht aus folgenden Kapiteln:

- **Einleitung**

Dieses Kapitel enthält eine Übersicht, welche Funktionen PERCON bietet.

- **Funktionen**

Sie finden eine Beschreibung der Funktionen und mit welchen Anweisungen sie sich realisieren lassen.

- **Arbeiten mit PERCON**

Es wird beschrieben, wie PERCON gestartet und beendet wird, wie die Anweisungen einzugeben sind und Dateien zugewiesen werden.

- **Anweisungen**

Die Anweisungen sind in alphabetischer Reihenfolge beschrieben.

- **PERCON als Unterprogramm**

Es enthält die Informationen, die Sie benötigen, um PERCON aus Benutzerprogrammen aufrufen zu können.

- **Benutzeranschlüsse**

Es wird beschrieben, wie die Steuerung des PERCON-Laufs an Benutzermodule übergehen kann.

- **Anwendungsbeispiele**

Sie finden ausgewählte Beispiele für häufige Anwendungsfälle von PERCON.

- **Meldungen**

Es sind alle Meldungen von PERCON mit ihrer Bedeutung und den notwendigen Maßnahmen aufgeführt.

- **Stichwörter**

Das Stichwortverzeichnis am Ende des Handbuchs dient dem schnelleren Auffinden von Problem- und Begriffserläuterungen.

## **Readme-Datei**

Funktionelle Änderungen und Nachträge der aktuellen Produktversion zu diesem Handbuch entnehmen Sie bitte ggf. der produktspezifischen Readme-Datei. Sie finden die Readme-Datei auf Ihrem BS2000-Rechner unter dem Dateinamen *SYSRME.produkt.version.sprache*. Die Benutzerkennung, unter der sich die Readme-Datei befindet, erfragen Sie bitte bei Ihrer zuständigen Systembetreuung. Die Readme-Datei können Sie mit dem Kommando `/SHOW-FILE` oder mit einem Editor ansehen oder auf einem Standarddrucker mit folgendem Kommando ausdrucken:

```
/PRINT-DOCUMENT dateiname, LINE-SPACING=*BY-EBCDIC-CONTROL
```

bei SPOOL-Versionen kleiner 3.0A:

```
/PRINT-FILE FILE-NAME=dateiname, LAYOUT-CONTROL=  
PARAMETERS(CONTROL-CHARACTERS=EBCDIC)
```

## 1.4 Änderungen gegenüber der vorigen Ausgabe

Die PERCON-Version V2.6A enthält gegenüber der Vorgängerversion folgende wesentlichen Änderungen:

- Eliminierung fest vorgegebener Dateinamen und Kennungen  
Dadurch ist PERCON als Subsystem mit der Möglichkeit der Koexistenz von Produktversionen einsetzbar.
- Ausgeben von Meldungen über S-Variablen
- Ausgeben nicht selektierter Sätzen in sogenannte Restdateien (Anweisung SELECT-INPUT-RECORDS)
- Dezimal-Aufbereitung von Feldern (Anweisung SET-RECORD-MAPPING)
- Behandeln von Steuerzeichen bei der Aufbereitung von Ausgabesätzen (Operand REPLACE-CONTROL-CHAR der Anweisung SET-PAGE-LAYOUT)
- Vorbelegen des Ausgabebereichs mit Sätzen einer bereits existierenden Ausgabedatei (Operand FILLER=\*OUTPUT der Anweisung SET-RECORD-MAPPING)  
Einzelne Felder in den Sätzen einer bereits existierenden Ausgabedatei können damit aktualisiert werden.
- Festlegen einer Mindest-Satzlänge beim Aufbauen von Ausgabesätzen (Operand MIN-RECORD-LENGTH der Anweisung SET-RECORD-MAPPING)
- Aufbauen von Sekundärschlüsseln für NK-ISAM-Ausgabedateien entsprechend einer angegebenen Eingabedatei (Operand ALTERNATE-INDEX der Anweisung ASSIGN-OUTPUT-FILE)
- Aufbereiten von PERCON-Kennwörtern ab Ausgabelänge 1 (in den Anweisungen SET-RECORD-MAPPING und SET-GROUP-ATTRIBUTES)
- Setzen eines Kommando-Returncodes

## Inkompatibilitäten

### OPENC-Ausgang

In der PERCON-Version V2.5A wurde generell im OPENC-Ausgang mit dem Makro VERIF ein Wiederherstellen der betroffenen Datei versucht. Das führte u.U. bei ISAM-Dateien zu Veränderungen des Datei-Inhalts.

Das Verhalten im OPENC-Ausgang wurde inkompatibel geändert. Bei ISAM-Dateien erfolgt die Wiederherstellung nur noch nach Zustimmung durch den Benutzer (siehe Abschnitt „Maßnahmen im OPENC-Ausgang“ auf Seite 32).

### Rückkehr-Informationen bei UP-Aufruf

Beim Unterprogramm-Aufruf von PERCON kann der Benutzer ein Feld für Rückkehr-Informationen bereitstellen, in dem u.a. in Form einer Bitleiste alle aufgetretenen PERCON-Meldungen vermerkt werden sollen. Das Feld ist so dimensioniert, daß nur die Meldungen von PER0000 bis PER0095 dort eingetragen werden können.

Diese Schnittstelle wird nicht weiterentwickelt. Damit sind die Meldungen ab PER0096 über diese Bitleiste nicht abfragbar. Als neue Lösung werden für die Meldungen die S-Variablen angeboten, die allerdings in der BS2000-Version V10.0A noch nicht zur Verfügung stehen (siehe Abschnitt „Ausgabe von Meldungen in S-Variablen“ auf Seite 247).

### Aktualisieren von ISAM-Dateien

Wird eine bereits existierende ISAM-Datei als Ausgabedatei verwendet, so kann mit dem Operanden OVERWRITE (Anweisung ASSIGN-OUTPUT-FILE) festgelegt werden, ob Sätze mit gleichem Schlüssel überschrieben werden dürfen oder nicht.

Bei OVERWRITE=\*NO können, unabhängig von der DUPKEY-Angabe, nur Sätze mit noch nicht vorhandenem Schlüssel eingefügt werden. Wurde vom Benutzer dennoch versucht, einen Satz mit bereits vorhandenem Schlüssel einzutragen, so meldete PERCON bisher mit der Meldung PER0036 den DVS-Fehler 0AAF und brach den Konvertierungsschritt ab. Da in diesem Fall aber kein DVS-, sondern ein Benutzerfehler vorliegt, wurde die Meldung PER0036 durch die neue Meldung PER0098 ersetzt.

### Eintragen eines CCSN in den Katalog-Eintrag

Am Ende eines Konvertierungsschrittes wird von PERCON versucht, für Ausgabedateien den CCS-Namen in den Katalog einzutragen (nur falls erforderlich). Tritt dabei ein Fehler auf, so wird in der PERCON-Version V2.5A der Konvertierungsschritt mit der Meldung PER0023 abgebrochen, obwohl die Datei-Erstellung selbst ordentlich beendet wurde.

Dieses Verhalten ist jetzt entschärft. Anstelle der Meldung PER0023 wird die Warnung PER0109 ausgegeben. Der Konvertierungsschritt wird nun aus diesem Grunde nicht mehr abgebrochen.

### **EXECUTE-SYSTEM-COMMAND im Dialog verboten**

Ab BS2000/OSD-BC V2.0 ist mit PERCON V2.6A im Dialog die PERCON-eigene Anweisung EXECUTE-SYSTEM-COMMAND nicht mehr erlaubt. In Prozeduren und im Stapelbetrieb wird sie aber weiter unterstützt (Kompatibilität).

Als Ersatz ist die SDF-Standard-Anweisung EXECUTE-SYSTEM-CMD zu verwenden. Zu beachten ist, daß sich die beiden Anweisungen in ihrer Syntax unterscheiden.

### **Lochkartengeräte**

Die Ein- und Ausgabe über Lochkarten wird im geführten Dialog nicht mehr angeboten (Operand FILE=CARD in der Anweisung ASSIGN-INPUT-FILE und Operand FILE=PUNCH in der Anweisung ASSIGN-OUTPUT-FILE).

Diese Operanden werden weiterhin unterstützt in Prozeduren, im Stapelbetrieb und im nicht geführten Dialog.

## **1.5 Verwendete Metasprache**

- Die Metasyntax der PERCON-Anweisungen ist am Anfang des Kapitels „PERCON-Anweisungen“ ab Seite 55 beschrieben.
- Literaturhinweise werden im Text in Kurztiteln angegeben, die in Anführungszeichen stehen. Die vollständigen Titel finden Sie im Literaturverzeichnis. Daran anschließend finden Sie Hinweise zur Bestellung von Handbüchern.
- Benutzereingaben in den Ablaufbeispielen sind in halbfetter Schrift wiedergegeben.

---

## 2 Funktionen

Bei allen PERCON-Funktionen werden Daten übertragen. Man unterscheidet drei Grundfunktionen und fünf Nebenfunktionen.

Die drei **Grundfunktionen** sind:

- Datei kopieren
- Band editieren
- Band duplizieren

Sie unterscheiden sich durch ihre Zugriffsart auf die Ein- und Ausgabedaten.

Grundfunktion	Zugriff auf	
	Eingabedaten	Ausgabedaten
Datei kopieren	logisch	logisch
Band editieren	physikalisch	logisch
Band duplizieren	physikalisch	physikalisch

Beim **logischen Zugriff** greift man über den Dateinamen auf Dateien zu. Die zu bearbeitenden Einheiten sind Sätze, die in ihrer logischen Reihenfolge bearbeitet werden.

Beim **physikalischen Zugriff** greift man über die Archivnummer auf Bänder/Magnetbandkassetten zu. Die zu bearbeitenden Einheiten sind Blöcke, die in ihrer physikalischen Reihenfolge bearbeitet werden.

Treffen Aussagen sowohl auf Bänder als auch auf Magnetbandkassetten zu, wird im folgenden nur noch von Bändern gesprochen. Sollten Unterschiede bestehen, wird an der betreffenden Stelle darauf hingewiesen.

Zu den Grundfunktionen können zusätzlich fünf Nebenfunktionen ausgewählt werden.

Die fünf **Nebenfunktionen** sind:

- Auswählen von Sätzen
- Aufbauen von Sätzen
- Aufbereiten von Sätzen oder Blöcken
- Gruppieren von Sätzen
- Positionieren von Bändern

Folgende Kombinationsmöglichkeiten von Grund- und Nebenfunktionen sind erlaubt.

<b>Nebenfunktionen</b>	<b>Grundfunktionen</b>		
	<b>Datei kopieren</b>	<b>Band editieren</b>	<b>Band duplizieren</b>
Auswählen	+	+	-
Aufbauen	+	-	-
Aufbereiten	+	+	-
Gruppieren	+	-	-
Positionieren	-	+	-

- + zulässige Kombination
- unzulässige Kombination



## 2.1 Grundfunktionen

### 2.1.1 Kopieren von Dateien

Beim Kopieren von Dateien werden eine oder mehrere Eingabedateien in eine oder mehrere Ausgabedateien übertragen. Mehrere Eingabedateien werden nacheinander in der Reihenfolge ihrer Zuweisung bearbeitet; in mehrere Ausgabedateien werden die Sätze gleichzeitig übertragen. Als **Eingabedateien** können Dateien auf Platte oder Band (katalogisierte Dateien), auf Disketten, auf Lochkarten oder SYSDTA verwendet werden.

Als **Ausgabedateien** können Dateien auf Platte oder Band (katalogisierte Dateien), auf Disketten, auf Lochkarten, SYSOUT oder SYSLST verwendet werden.

Die Dateien werden über die Anweisungen ASSIGN-INPUT-FILE oder ASSIGN-OUTPUT-FILE zugewiesen. Soll nur eine einzige Ein- oder Ausgabedatei zugewiesen werden, kann dies über ein SET-FILE-LINK-Kommando mit dem Dateikettungsnamen PCIN oder PCOUT erfolgen.

### 2.1.2 Editieren von Bandinhalten

Beim Editieren eines Bandes wird ein Eingabeband blockweise gelesen und pro Block ein Satz in eine oder mehrere Ausgabedateien übertragen. Als Ausgabedateien sind auch die Systemdateien SYSOUT und SYSLST zulässig.

Das Eingabeband muß mit der ASSIGN-INPUT-TAPE-Anweisung zugewiesen werden. Die Ausgabedatei wird über eine ASSIGN-OUTPUT-FILE-Anweisung und/oder ein SET-FILE-LINK-Kommando zugewiesen. Für mehrere Ausgabedateien bestehen ebenfalls beide Möglichkeiten der Zuweisung.

Vor der Ausgabe kann das Eingabeband über eine CHANGE-INPUT-TAPEPOSITION-Anweisung positioniert werden. Die Übertragung der Datenblöcke wird mit der START-TAPE-PROCESSING-Anweisung gestartet. Positionierung und Übertragung können beliebig oft wiederholt werden.

Das Editieren wird durch die Anweisung START-CONVERSION oder END abgeschlossen.

Mit dieser Funktion können auch nicht BS2000-konforme oder beschädigte Bänder editiert werden.

### 2.1.3 Duplizieren von Bändern

Beim Duplizieren von Bändern wird ein Eingabeband blockweise übertragen. Es können auch Multifile/Multivolume-Sets (MF/MV-Sets, siehe Handbuch „Einführung in das DVS“ [1]) übertragen werden.

Die Bänder müssen mit den Anweisungen ASSIGN-INPUT-TAPE und ASSIGN-OUTPUT-TAPE über die Archivnummern zugewiesen werden.

Ein Band wird bis zu der das Band abschließenden doppelten Bandmarke gelesen.

Werden beim Duplizieren für die Ausgabedaten zuwenig Ausgabebänder angegeben, gibt PERCON die Meldung PER0084 aus und fordert vom Operator ein weiteres Ausgabeband an. Es werden solange Bänder nachgefordert, bis der Dupliziervorgang abgeschlossen ist. Lehnt der Operator eine dieser Aufforderungen ab, gibt PERCON die Meldung PER0069 aus und bricht den Konvertierungsschritt mit einer unvollständigen Kopie ab.

Es können Bänder oder MF/MV-Sets mit Standard- oder Nichtstandardkennsätzen oder ohne Kennsätze bearbeitet werden.

Für MF/MV-Sets mit Standardkennsätzen gilt:

Unterscheiden sich Ein- und Ausgabebänder in ihrer Größe oder in ihrer Schreibdichte, werden Bandmarken und Kennsätze, die nur zum Fortsetzen der Daten auf dem nächsten Band gedient haben, nicht in die Ausgabe übernommen (unterschiedliche Anzahl von Labels bei der Ein- und Ausgabe). Haben dagegen Ein- und Ausgabebänder gleiche Größe und Schreibdichte, werden alle Standardkennsätze (auch solche, die für die Datenfortsetzung notwendig sind) in das MF/MV-Ausgabeset übertragen. Es entsteht ein MF/MV-Set mit Standardkennsätzen.

Für MF/MV-Sets ohne Standardkennsätze gilt:

PERCON überträgt den Inhalt jedes Eingabebandes ohne Umstrukturierung der Daten auf jeweils ein Ausgabeband. Es entsteht wieder ein MF/MV-Set ohne Standardkennsätze.

#### *Hinweise*

- FDDRL-Sicherungsbandsets können mit PERCON nur dann kopiert werden, wenn der von FDDRL angebotene Benutzeranschluß von PERCON verwendet wird. Einzelne FDDRL-Bänder können aber ohne Benutzeranschluß kopiert werden, siehe Beispiel auf Seite 234.
- Für das Kopieren von ARCHIVE-Sicherungsbandsets muß statt PERCON die von ARCHIVE angebotene Kopierfunktion verwendet werden, siehe Handbuch „ARCHIVE“ [12].
- Beim physikalischen Zugriff auf Bänder, d.h. beim Editieren und Duplizieren, werden auch Kennsätze wie Datenblöcke gelesen und geschrieben. Voraussetzung ist, daß der Benutzer bestimmte Berechtigungen besitzt (siehe TPIGNORE / TAPE-ACCESS im JOIN-Eintrag).

## 2.2 Nebenfunktionen

### 2.2.1 Auswählen von Sätzen oder Blöcken

Beim Kopieren einer Datei können Sätze, beim Editieren eines Bandes Blöcke für die weitere Bearbeitung ausgewählt werden. Die Auswahl erfolgt durch die Angabe eines logischen Ausdrucks in der SELECT-INPUT-RECORDS-Anweisung und zwar durch:

- Vergleich einzelner Felder eines Satzes/Blocks mit einem angegebenen Literal (Konstante)
- Vergleich eines von PERCON intern mitgeführten Block-, Byte- oder Satzzählers (siehe Seite 73) mit einem angegebenen Literal
- Prüfung eines Feldes auf Datenformat (numerisch, alphabetisch)
- Prüfung auf ab- oder aufsteigende Sortierung
- Prüfung eines Feldes auf zulässige Zeichen anhand einer Zeichentabelle

Als Vergleichsoperatoren sind möglich

- gleich
- kleiner
- kleiner gleich
- größer
- größer gleich
- ungleich

Aus mehreren solcher Vergleiche kann mit den logischen Operatoren

- AND
- OR

ein logischer Ausdruck gebildet werden. Der Ausdruck wird von links nach rechts abgearbeitet, wobei der Operator „AND“ Vorrang vor dem Operator „OR“ hat. Ergibt die Auswertung den Wert „wahr“, wird der Eingabesatz/-block in die über OUTPUT-LINK-NAME angegebene Ausgabedatei übernommen, sonst wird er nicht übernommen.

Zusätzlich kann durch Klammerung die Reihenfolge der Auswertung der Ausdrücke festgelegt werden.

Sätze bzw. Blöcke, für die keines der Auswahlkriterien zutrifft, können in sogenannte „Restdateien“ übernommen werden.

## 2.2.2 Aufbauen von Sätzen

Beim Kopieren von Dateien können die Ausgabesätze nach den Wünschen des Benutzers aufgebaut werden. Der Aufbau erfolgt durch Angaben in der SET-RECORD-MAPPING-Anweisung aus:

- Feldern der Eingabesätze
- Literalen (Konstanten)
- Kennwörtern, die PERCON zur Verfügung stellt

Zur Vorbelegung des Ausgabebereichs kann entweder der Eingabesatz oder ein Füllzeichen verwendet werden. Bei bereits existierenden ISAM-Ausgabedateien können auch Sätze dieser Dateien zur Vorbelegung verwendet werden.

Im Ausgabesatz kann eine Zeichenkette alphanumerisch, sedezimal oder binär, eine Zahl gepackt, gezont oder aufbereitet dargestellt werden.

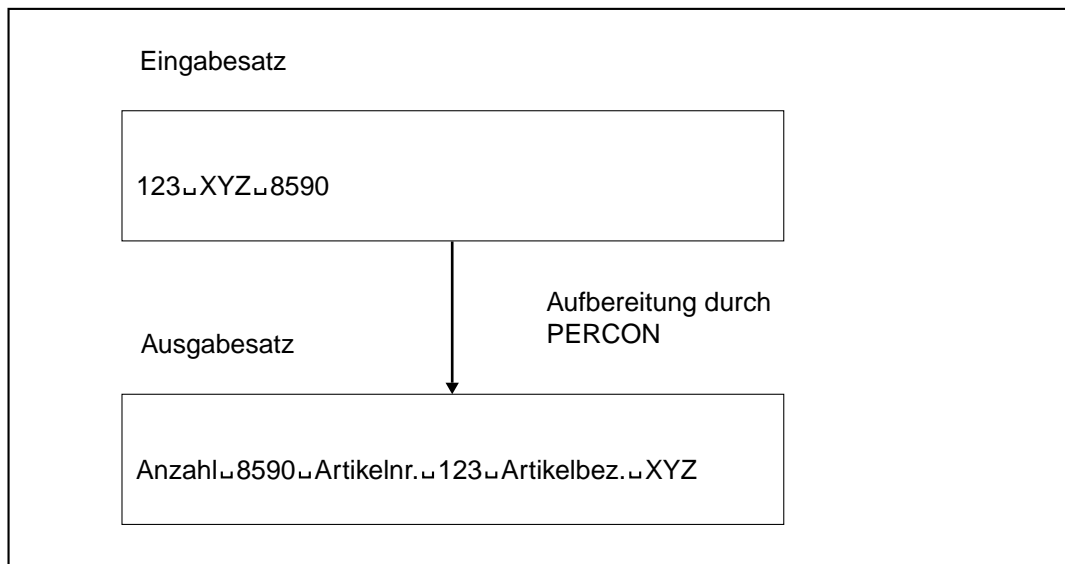


Bild 1: Aufbau eines Ausgabesatzes

Beim Aufbau eines Ausgabesatzes kann es erforderlich sein, die Dateimerkmale zu verändern, z.B. muß beim Einfügen von Literalen in einen Satz fester Länge die Satzlänge verändert werden.

Wird der aufgebaute Satz noch aufbereitet, so ist beim Festlegen der Ausgabe position darauf zu achten, ob der unaufbereitete Satz ein zusätzliches Satzlängenfeld im Datenteil erhalten soll (siehe Abschnitt „Aufbereiten von Sätzen oder Blöcken“ auf der nächsten Seite).

Haben die aufzubauenden Sätze keine feste Länge, kann eine einzuhaltende Minimallänge vorgeschrieben werden.

### 2.2.3 Aufbereiten von Sätzen oder Blöcken

Beim Kopieren einer Datei und beim Editieren eines Bandes können die auszugebenden Sätze seitengerecht über die SET-PAGE-LAYOUT-Anweisung aufbereitet werden. Die Ausgabe erfolgt nach SYSOUT, nach SYSLST oder in eine katalogisierte Datei. Die aufbereiteten Ausgabesätze haben variables Satzformat in einer SAM-Datei.

Als Grundlage für die Aufbereitung dienen die unaufbereiteten Ausgabesätze, die standardmäßig folgende Eigenschaften haben:

- maximale Länge: 32768 Byte
- undefiniertes Satzformat

Die Standardeinstellung der unaufbereiteten Sätze kann durch folgende Parameter in der SET-FILE-LINK-Anweisung geändert werden:

- Mit RECORD-SIZE und/oder BUFFER-LENGTH wird die Satzlänge festgelegt. Sind beide Operanden angegeben, legt der kleinere Wert von beiden die maximale Satzlänge fest.
- Mit RECORD-FORMAT wird das Format des unaufbereiteten Ausgabesatzes beschrieben. Ist RECORD-FORMAT=VARIABLE, dann erhält der unaufbereitete Satz zusätzlich ein Satzlängenfeld, das als Datenteil in die Aufbereitung einbezogen wird.

#### *Hinweis*

Wurde die Ausgabedatei mit FILE-ATTRIBUTES = INPUT-FILE vereinbart und weist die erste Eingabedatei das Satzformat VARIABLE auf, dann erhält der unaufbereitete Satz ebenfalls das zusätzliche Satzlängenfeld im Datenteil.

Soll die Ausgabe in eine katalogisierte Datei erfolgen, haben die Angaben im TFT-Eintrag Vorrang vor den Angaben in der ASSIGN-OUTPUT-FILE-Anweisung. Angaben im Katalogeintrag werden nicht ausgewertet.

Der Ausgabesatzzähler RECORD-COUNTER zählt die unaufbereiteten Sätze.

Beim Aufbereiten einer leeren Datei wird eine Seite mit der Kopfzeile erzeugt. Wird keine Kopfzeile gefordert, entsteht ein leeres Blatt.

## Gestalten der Ausgabe durch Seiten- und Zeilenaufbereitung

Mit der SET-PAGE-LAYOUT-Anweisung kann das Druckbild zeilen- und seitenweise festgelegt werden:

- Die Überschriftzeile kann gestaltet werden (Operand HEADER-LINE)
- Zeilenbereiche, in die die Ausgabesätze geschrieben werden sollen, können festgelegt werden (Operand OUTPUT-AREA)

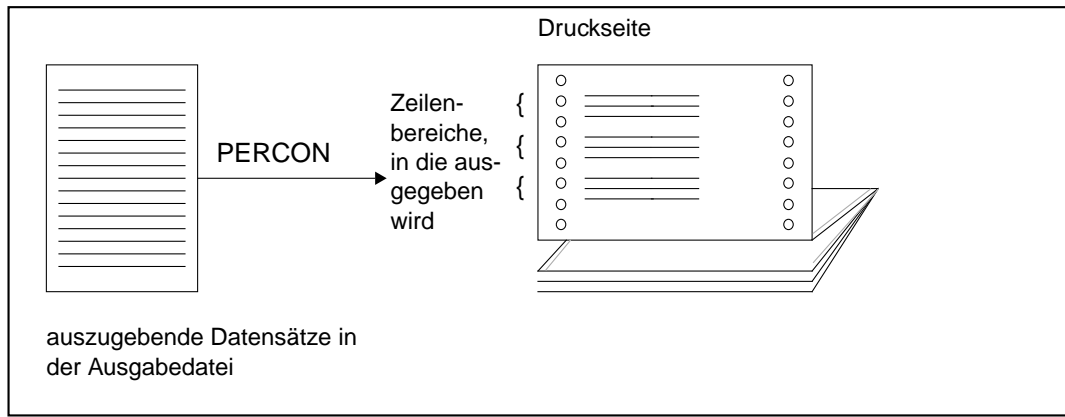


Bild 2: Gestalten der Zeilenbereiche

- Spaltenbereiche, in die pro Zeile geschrieben werden soll, können festgelegt werden. Diese Bereiche werden jeweils durch ein Leerzeichen getrennt (Operand COLUMN-SIZE).

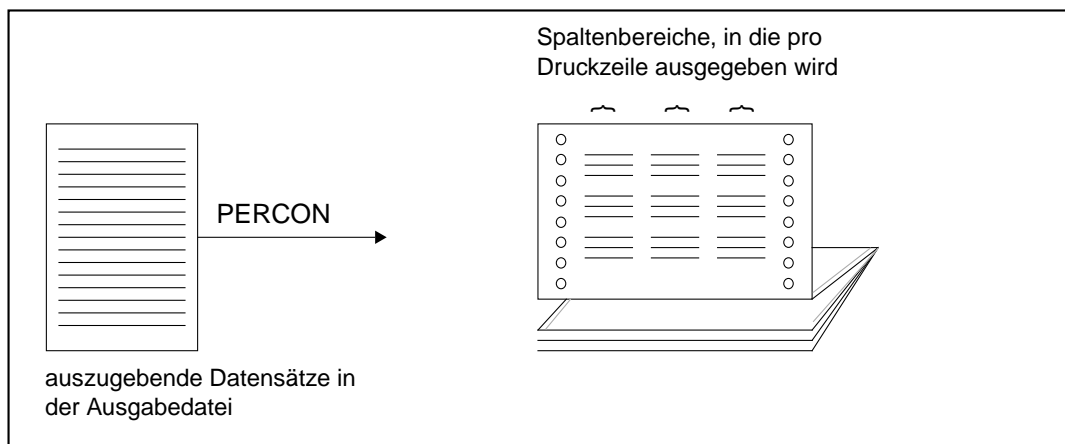


Bild 3: Gestalten der Spaltenbereiche

- Die Anzahl der Zeichen des unaufbereiteten Ausgabesatzes pro Ausgabezeile kann definiert werden (Operand LINE-SIZE).
- Die Ausgabe von Zeilen mit gleichem Inhalt kann unterdrückt werden (Operand SUPPRESS-EQUAL-LINES).
- Pro Seite können Zeilen festgelegt werden, in die ein vom Benutzer vorgegebener Text eingefügt werden soll (Benutzerzeilen). In diese Zeilen werden keine Ausgabesätze geschrieben (Operand USER-LINE).
- Daten können im Sedezimalformat, im Zeichenformat oder in beidem dargestellt werden (Operand OUTPUT-FORMAT).
- Die Anzahl der Leerzeilen nach einer Ausgabezeile kann definiert werden (Operand SPACING).

Die maximale Länge eines aufbereiteten Ausgabesatzes ergibt sich aus dem Zeilenvorspann und den Operanden COLUMN-SIZE, LINE-SIZE und OUTPUT-FORMAT. Sie darf 204 Byte nicht überschreiten.

## Standard-Seitengestaltung

Bei der Ausgabe nach SYSLST oder nach SYSOUT sowie bei der Angabe einer SET-GROUP-ATTRIBUTES-Anweisung erfolgt eine Seitenaufbereitung, auch wenn keine SET-PAGE-LAYOUT-Anweisung angegeben wurde.

Die Standard-Seitengestaltung hat folgende Form:

### Beim Kopieren einer Datei

Seitengestaltung	Ausgabemedium	
	SYSLST oder katalogisierte Datei	SYSOUT
Seitenüberschrift (HEADER-LINE)	Seitenzähler	keine
Datenbereich (OUTPUT-AREA)	Zeile 5-66	alle Zeilen
Datenformat (OUTPUT-FORMAT)	Zeichen	Zeichen
Zeilenlänge (LINE-SIZE)	136 Zeichen	64 Zeichen
Gruppenlänge (COLUMN-SIZE)	136 Zeichen	16 Zeichen
Unterdrücken gleicher Folgezeilen (SUPPRESS-EQUAL-LINES)	ja	ja
Zusatzinformationen (als Vorspann in der Zeile)	keine	9 Zeichen Satzposition (BYTCNT)

### Beim Editieren eines Bandes

Seitengestaltung	Ausgabemedium	
	SYSLST	SYSOUT
Seitenüberschrift (HEADER-LINE)	Datum, Uhrzeit, Seitenzähler	keine
Datenbereich (OUTPUT-AREA)	Zeile 5-66	alle Zeilen
Datenformat (OUTPUT-FORMAT)	Zeichen und sedezimal	Zeichen und sedezimal
Zeilenlänge (LINE-SIZE)	100 Zeichen	64 Zeichen
Gruppenlänge (COLUMN-SIZE)	20 Zeichen	16 Zeichen
Unterdrücken gleicher Folgezeilen (SUPPRESS-EQUAL-LINES)	ja	ja
Zusatzinformationen (als Vorspann in der Zeile)	21 Zeichen Bandmarkenzähler (TMCNT) Blockzähler (BLKCNT) Satzposition (BYTCNT)	9 Zeichen Satzposition (BYTCNT)



## 2.2.4 Gruppieren von Sätzen

Beim Bilden von Gruppen werden die Sätze einer Datei nach einem gemeinsamen Kriterium, dem Gruppenbegriff, zu Gruppen zusammengefaßt.

Dazu müssen in der SET-GROUP-ATTRIBUTES-Anweisung die Felder des Eingabesatzes angegeben werden, die für eine Gruppe den Gruppenbegriff darstellen (im Beispiel BC, EF und IK). Definierte Gruppen können zusammen mit Daten dieser Gruppe und vom Benutzer angegebenen Informationen ausgegeben werden.

Bei der Gruppenbildung wird implizit eine Aufbereitung der Sätze vorgenommen (SET-PAGE-LAYOUT).

### Ablauf der Gruppenbildung mit Gruppenwechsel

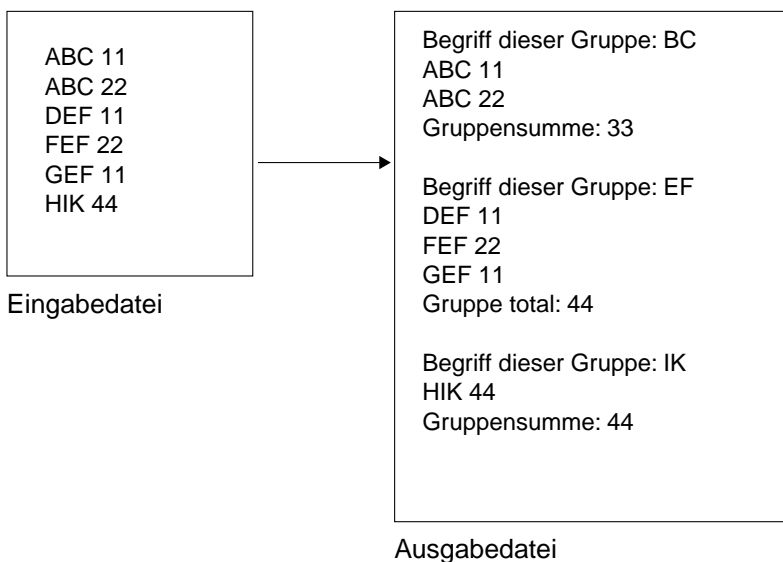
Ein oder auch mehrere Felder des Eingabesatzes werden als Gruppenbegriff definiert. Aufeinanderfolgende Sätze mit gleichem Gruppenbegriff bilden eine Gruppe. Sollen alle Sätze mit gleichem Gruppenbegriff als Gruppe erkannt werden, muß die Datei nach dem Gruppenbegriff sortiert werden (z.B. mit dem Softwareprodukt SORT, siehe Handbuch „SORT“ [4]).

Ein Gruppenwechsel liegt dann vor, wenn beim Lesen der Datei zwei aufeinanderfolgende Sätze verschiedene Gruppenbegriffe haben.

PERCON stellt Routinen zur Verfügung, die bei einem Gruppenwechsel durchlaufen werden. Diese Routinen 'Gruppenvorlauf' und 'Gruppennachlauf' werden vor bzw. nach einem Gruppenwechsel durchlaufen.

Folgende Aktionen können beim Gruppenwechsel ausgeführt werden:

- Aufbereitung von Gruppenwechselzeilen aus Feldern des Eingabesatzes, aus Kennwörtern und Literalen
- Aufbau von Gruppenwechselzeilen durch Benutzermodule
- Vorschub um 1 bis 15 Zeilen
- Vorschub um eine Seite
- Ausgabe von Summen vorgegebener Satzfelder der durchlaufenen Gruppe

**Beispiel**

Gruppendefinitionen dürfen geschachtelt werden, d.h. für eine Datei können mehrere Gruppenbegriffe erklärt werden, die sich überlappen dürfen. Man spricht dann von Gruppenstufen. Bis zu 8 Gruppenstufen sind möglich. Durch das Numerieren der Stufen von 1 bis 8 wird eine Wertung vorgenommen. Stufe 1 ist die höchste Stufe. Erfolgt bei einer höherwertigen Stufe ein Gruppenwechsel, findet auch bei allen niederen Stufen ein Gruppenwechsel statt, auch wenn für diese Gruppen das Kriterium für den Gruppenwechsel nicht gegeben ist. Ein Beispiel zum Gruppenwechsel finden Sie auf Seite 215.

Zu jeder Gruppenstufe einer Datei wird ein Zähler, der GROUP-COUNTER, geführt (siehe Seite 75).

## 2.2.5 Positionieren von Bändern

Beim Editieren von Bändern kann vor jeder Übertragung von Datenblöcken das Eingabeband positioniert werden. Über die Anweisung CHANGE-INPUT-TAPEPOSITION kann

- eine bestimmte Anzahl von Blöcken
- eine bestimmte Anzahl von Bandmarken
- auf den Bandanfang
- auf das Bandende
- hinter das Bandende (nur vom Systemverwalter)
- auf die nächste doppelte Bandmarke

vor- oder zurückpositioniert werden. Die Funktion wird sofort ausgeführt.

## 3 Arbeiten mit PERCON

In diesem Kapitel wird das Starten, Beenden und Unterbrechen des PERCON-Laufs beschrieben. Außerdem wird gezeigt, wie PERCON in Konvertierungsschritten arbeitet, in welcher Reihenfolge die PERCON-Anweisungen eingegeben werden und wie die Ein- und Ausgabe zugewiesen wird. Weiterhin wird beschrieben, wie katalogisierte Dateien bearbeitet werden, wie sich PERCON in XS-Umgebung verhält und wie erweiterte Zeichensätze in PERCON verwendet werden.

### 3.1 Starten des PERCON-Laufs

Mit dem Kommando /START-PERCON oder /PERCON wird PERCON als autonomes Programm aufgerufen.

Anwendungsbereiche: **FILE, UTILITIES**

START-PERCON / PERCON
-----------------------

<b>CPU-LIMIT</b> = <b>*JOB-REST</b> / <integer 1..32767>
--

, <b>MONJV</b> = <b>*NONE</b> / <full-filename 1..54 without-gen-vers>
--

, <b>PROGRAM-MODE</b> = <b>24</b> / <b>*ANY</b>
---

, <b>VERSION</b> = <b>*STD</b> / <product-version> / <product-version without-corr> / <product-version without-corr without-man>
---

**CPU-LIMIT = \*JOB-REST / <integer 1..32767>**

CPU-Zeit, die PERCON beim Ablauf verbrauchen darf. Wird diese Zeit überschritten, wird der Benutzer im Dialogbetrieb vom System benachrichtigt; im Stapelbetrieb wird der PERCON-Lauf beendet.

**CPU-LIMIT = \*JOB-REST**

Ist im SET-LOGON-PARAMETERS-Kommando der Operand CPU-LIMIT=\*STD angegeben worden, gibt es keine Zeitbeschränkung für das Programm.

Ist im SET-LOGON-PARAMETERS-Kommando der Operand CPU-LIMIT=t angegeben worden, wird als Zeitbeschränkung für den PERCON-Lauf der bei der Systemgenerierung festgelegte Wert verwendet.

**MONJV = \*NONE / <full-filename 1..54 without-gen-vers>**

Name der Jobvariablen, die den PERCON-Lauf überwachen soll. Die Jobvariable muß bereits katalogisiert sein (steht nur Benutzern mit dem Software-Produkt JV zur Verfügung, siehe Handbuch „Jobvariablen“ [9]).

Während des Ablaufs von PERCON setzt das System die Jobvariablen auf folgende Werte:

Wert	Bedeutung/Anlaß der Wertzuweisung
\$R	PERCON läuft.
\$T	PERCON wurde normal beendet.
\$A	PERCON wurde abnormal beendet.

**MONJV = \*NONE**

Es wird keine Jobvariable vereinbart.

**PROGRAM-MODE =**

Legt fest, in welchen Teil des Adreßraums PERCON geladen wird (gilt nur bei XS-Anlagen).

**PROGRAM-MODE = 24**

Die gesamte Ladeeinheit wird unterhalb der 16-Mbyte-Grenze geladen. Die Programmausführung erfolgt im 24-Bit-Adressierungsmodus. Extern-Verweise werden als 24-Bit-Adressen interpretiert.

**PROGRAM-MODE = \*ANY**

Die Module der Ladeeinheit können sowohl unterhalb als auch oberhalb der 16-Mbyte-Grenze liegen.

**VERSION = \*STD / <product-version> / <product-version without-corr> / <product-version without-man>**

Gibt die Version von PERCON an, die aufgerufen werden soll.

Die Version kann verschieden detailliert angegeben werden. Folgende Angaben sind zugelassen:

nn.nann	n.nann	(z.B. 02.6A30)	volle Versionsangabe
nn.na	bzw. n.na	(z.B. 02.6A)	ohne Korrekturstand
nn.n	n.n	(z.B. 02.6)	ohne Korrektur- und Freigabestand

wobei n eine Ziffer und a ein Buchstabe ist

*Hinweis* Das Voranstellen des Zeichens V ist möglich, ebenso das Einschließen in Hochkommata.

Das Weglassen des Operanden VERSION bzw. die Angabe \*STD bedeuten, daß der Benutzer keine spezielle Version anfordert. Stehen aufgrund einer zu wenig detaillierten Versionsangabe mehrere installierte PERCON-Versionen zur Auswahl, so werden die auf Seite 192 genannten Kriterien für eine Auswahl herangezogen. Steht keine Version, die der Versions-Anforderung genügt, zur Verfügung, wird mit der Fehlermeldung PER0104 abgebrochen.

Die zweite Möglichkeit, eine PERCON-Version anzufordern, bietet das Kommando /SELECT-PRODUCT-VERSION. Damit kann der Benutzer vor dem PERCON-Aufruf (bei Unterprogramm-Aufruf vor dem Aufruf des zugehörigen Hauptprogramms) die gewünschte PERCON-Version spezifizieren. Verzichtet er in diesem Fall auf eine explizite Versionsangabe beim Start von PERCON, wird die im Kommando /SELECT-PRODUCT-VERSION angegebene PERCON-Version ausgewählt.

Sollten bei einer expliziten Versionsangabe aufgrund einer wenig detaillierten Formulierung mehrere installierte Versionen in Frage kommen, wird das eingeschränkte Versions-Angebot entsprechend der festgelegten Reihenfolge (siehe Seite 192) geprüft, bis eine eindeutige Version gefunden worden ist.

**VERSION = \*STD**

Der Anwender fordert keine spezielle Version an, bzw. der Operand VERSION wurde nicht angegeben.

## 3.2 Beenden des PERCON-Laufs

Um PERCON zu beenden, muß die Anweisung END angegeben werden.

### 3.3 Kommando-Returncode

Beim Hauptprogramm-Aufruf setzt PERCON einen Kommando-RC, aus dem hervorgeht, ob alle Anweisungen von PERCON fehlerfrei ausgeführt wurden oder nicht bzw. ob PERCON normal oder abnormal endete. Bei Warnungen oder Abbrüchen wird der Kommando-RC mit dem entsprechenden Meldungsschlüssel versorgt.

Beim Unterprogramm-Aufruf von PERCON wird kein Kommando-RC gesetzt.

Beim Festlegen des Kommando-RC werden folgende Fälle unterschieden:

1. normale Beendigung: SC2=X'00' / SC1=X'00' / MC=C'CMD0001'
2. normale Beendigung mit Warnung: SC2=X'02' / SC1=X'00' / MC=C'PERxxxx'
3. abnormale Beendigung nach Abbruch von mindestens einem Konvertierungsschritt:  
SC2=X'00' / SC1=X'01' / MC=C'PERxxxx'

Liegt keine PER-Meldung vor (z.B. bei Syntaxfehlern in fester Anweisungsfolge), gilt:  
MC=C'CMD0202'.

4. abnormale Beendigung nach einem Fehler, der das Nachbinden des Großmoduls von PERCON (und damit den eigentlichen PERCON-Lauf) verhindert:  
SC2=X'00' / SC1=X'40' / MC=C'PERxxxx'

Treten in einem PERCON-Lauf mehrere Situationen auf, die den Kommando-RC beeinflussen, so wird der schwerste aufgetretene Fehler gemeldet. Das bedeutet, daß z.B. der einer Warnung zugeordnete Kommando-RC nicht gemeldet wird, wenn ein Konvertierungsschritt abgebrochen wurde. Bei mehreren gleichwertigen Fehlern wird der erste gemeldet.

Das Festlegen des Maincodes erfolgt unabhängig davon, ob die entsprechende Meldung wirklich ausgegeben wird oder ob ihre Ausgabe entsprechend den Festlegungen in der Anweisung MODIFY-PERCON-OPTIONS unterdrückt wird, d.h. das Auftreten eines Ereignisses bestimmt den Maincode, nicht die gerade eingestellte Art der Meldungs-Ausgabe.

PERxxxx	SC2	SC1	Maincode	Bedeutung
0000 0029 0030 0070	-	-	-	kein RC gesetzt
0001	-	-	-	wenn eine andere PERCON-Meldung den Abbruch beschreibt
	00	01	CMD0202	sonst
0031	-	-	-	wenn eine andere PERCON-Meldung eine Warnung ausgibt
	00	00	CMD0001	sonst
0009 0012 0014 0016 0024 0040 0045 0051 0052 0054 0059 0065 0071 0084 0086 0092 0093 0094 0097 0109	02	00	PERxxxx	Warnungen
0021	02	00	PER0021	bei Antwort HN oder SN
	00	01	PER0021	bei Antwort HA oder SA

PERxxxx	SC2	SC1	Maincode	Bedeutung
0022	00	01	PER0022	<ul style="list-style-type: none"> <li>– bei Ausgabe-Dateien</li> <li>– bei Eingabe-Dateien mit TERMINATION = abnormal</li> </ul>
	02	00	PER0022	bei Eingabedateien mit TERMINATION = normal
0063	02	00	PER0063	Antwort I oder S
	00	01	PER0063	Antwort H
0005 0006 0007 0008 0020 0025 0026 0027 0028 0034 0043 0046 0047 0056 0062	-	-	-	bei Eingabe der Anweisungen über Terminal
	00	01	PERxxxx	bei fester Anweisungs-Folge
0010 0033 0049 0050 0055 0058 0067 0072 0088 0089 0090 0091	-	-	-	bei Eingabe der Anweisungen über Terminal
	02	00	PERxxxx	bei fester Anweisungs-Folge
0013 0066 0080 0087	-	-	-	bei Eingabe der Anweisungen über Terminal
	02	00	PERxxxx	ISP-Anweisungen, bei fester Anweisungs-Folge
	00	01	PERxxxx	SDF-Anweisungen, bei fester Anweisungs-Folge
0002 0004 0011 0015 0017 0018 0019 0023 0032 0035 0036 0037 0038 0039 0041 0042 0044 0048 0053 0057 0060 0061 0064 0068 0069 0073 0074 0075 0076 0077 0078 0079 0081 0082 0083 0085 0096 0098 0099 0100 0101 0102 0103	00	01	PERxxxx	Abbruch des Konvertierungsschritts
0095 0104 0105 0106 0107 0110 0111	00	64	PERxxxx	Abbruch von PERCON
0003	00	64	PER0003	HS-Mangel beim Nachbinden des Großmoduls
	00	01	PER0003	Abbruch des Konvertierungsschritts bei HS-Mangel im PERCON-Lauf

## 3.4 Unterbrechen des PERCON-Laufs

Der PERCON-Lauf kann unterbrochen werden:

- im Dialog durch die Programmunterbrechungstaste der Datensichtstation (K2-Taste) und durch Eingabe des Kommandos /SEND-MESSAGE TO=\*PROGRAM,
- im ENTER-Betrieb durch Eingabe von /SEND-MESSAGE von der Konsole.

Nach der Eingabe des Kommandos /SEND-MESSAGE wird die Meldung PER0070 ausgegeben. Um den weiteren Ablauf zu steuern, muß einer der folgenden Operationscodes als Antwort eingegeben werden:

Operationscode	Bedeutung
DISP	Der aktuelle Bearbeitungszustand wird ausgegeben (Satz-/ Blockzähler).
CONT	Die Bearbeitung wird fortgesetzt.
START	Der aktuelle Konvertierungsschritt wird abgebrochen und der nächste Konvertierungsschritt gestartet.
FIN	Das Lesen aus der aktuellen Eingabedatei wird beendet und, falls vorhanden, mit der folgenden Eingabedatei fortgesetzt.
TERM	Der PERCON-Lauf wird beendet.

Wird DISP eingegeben, wird nach den Informationen über den aktuellen Bearbeitungszustand die gleiche Meldung wie nach dem Kommando /SEND-MESSAGE ausgegeben. Für den weiteren Ablauf muß erneut eine Antwort in Form eines Operationscodes eingegeben werden.

### *Hinweis*

Die Angaben DISP, CONT, START, FIN, TERM dürfen nicht abgekürzt werden.



### 3.5 Konvertierungsschritt

PERCON arbeitet in Konvertierungsschritten. Ein Konvertierungsschritt ist ein Abschnitt des PERCON-Laufs, der mit den Anweisungen zum Übertragungsvorgang der Daten beginnt und mit einer abschließenden Meldung von PERCON endet. In einem PERCON-Lauf können beliebig viele Konvertierungsschritte gestartet werden.

Durch eine der Anweisungen START-CONVERSION oder END wird das Ende der Angaben zu einem Konvertierungsschritt gekennzeichnet.

Bei der RESET-INPUT-Anweisung werden alle bis dahin gemachten Angaben zurückgesetzt und der Konvertierungsschritt wird beendet.

Während nach einer der Anweisungen START-CONVERSION oder RESET-INPUT weitere Konvertierungsschritte möglich sind, beendet die END-Anweisung den PERCON-Lauf.

Beim Kopieren von Dateien können pro Konvertierungsschritt beliebig viele Ein- und Ausgabedateien zugewiesen werden. Alle Anweisungen eines Konvertierungsschrittes werden gesammelt und in einem Übertragungsvorgang ausgeführt.

Beim Editieren eines Bandes können pro Konvertierungsschritt nur ein Eingabeband aber mehrere Ausgabedateien zugewiesen werden. Jede Anweisung eines Konvertierungsschrittes wird sofort ausgeführt. So können Ergebnisse einer Anweisung in den folgenden Anweisungen verwertet werden.

Beim Duplizieren von Bändern kann pro Konvertierungsschritt nur ein Eingabeband oder MF/MV-Set, aber es können mehrere Ausgabebänder oder MF/MV-Sets zugewiesen werden. Die Ausgabe erfolgt gleichzeitig auf alle Ausgabebänder.

#### Beispiel

```
/START-PERCON
//ASSIGN-INPUT-FILE...
//ASSIGN-OUTPUT-FILE...
//SELECT-INPUT-RECORDS...
//START-CONVERSION
```

```
PER0030 .....
PER0030 ..... (1)
```

```
//ASSIGN-INPUT-FILE
//ASSIGN-OUTPUT-FILE...
//SET-PAGE-LAYOUT...
//END
```

```
PER0030 .....
PER0030 ..... (2)
```

```
PER0031 PERCON NORMAL BEENDET (3)
```

- (1) Ende 1. Konvertierungsschritt
- (2) Ende 2. Konvertierungsschritt
- (3) Ende des PERCON-Laufes

Tritt während der Bearbeitung eines Konvertierungsschrittes ein logischer Fehler auf,

- wird in Prozeduren und Stapelaufträgen auf die nächste STEP- oder END-Anweisung verzweigt,
- werden im Dialog die nachfolgenden Anweisungen zum nächsten Konvertierungsschritt ausgewertet.

Ein fehlerhafter Konvertierungsschritt führt zur fehlerhaften Beendigung (TERM MODE=\*ABNORMAL) des gesamten PERCON-Laufs (Verzweigung auf /SET-JOB-STEP), auch wenn alle Konvertierungsschritte nach dem fehlerhaften normal beendet wurden.

## 3.6 Reihenfolge der Anweisungen

Zuerst müssen die Ein- und Ausgabeeinheiten wie Dateien, Bänder usw. zugewiesen werden, damit spätere Anweisungen sich auf diese beziehen können (Anweisungen ASSIGN-INPUT-FILE, ASSIGN-OUTPUT-FILE, ASSIGN-INPUT-TAPE, ASSIGN-OUTPUT-TAPE).

Beim Duplizieren eines Bandes muß die Anweisung ASSIGN-INPUT-TAPE vor der Anweisung ASSIGN-OUTPUT-TAPE erfolgen.

Die Anweisung START-TAPE-PROCESSING kann nur nach einer Anweisung ASSIGN-INPUT-TAPE gegeben werden.

Im geführten Dialog werden die Anweisungen ASSIGN-OUTPUT-TAPE, CHANGE-INPUT-TAPEPOSITION und START-TAPE-PROCESSING erst **nach** Eingabe der Anweisung ASSIGN-INPUT-TAPE angeboten.

Die Anweisung CHANGE-INPUT-TAPEPOSITION darf nur nach der Anweisung ASSIGN-INPUT-TAPE gegeben werden, kann aber vor oder nach der Anweisung ASSIGN-OUTPUT-FILE stehen.

Folgende Anweisungen können sich auf eine, auf ausgewählte oder auf alle bisher angegebenen Ausgabedateien eines Konvertierungsschrittes beziehen: START-TAPE-PROCESSING, SELECT-INPUT-RECORDS, SET-RECORD-MAPPING, SET-PAGE-LAYOUT und SET-GROUP-ATTRIBUTES. Soll sich eine Anweisung nur auf eine oder auf ausgewählte Ausgabedateien beziehen, müssen in der Anweisung die zugehörigen Dateikettungsamen dieser Ausgabeeinheiten angegeben werden.

Wird eine der genannten Anweisungen ohne Dateikettungsamen angegeben, bezieht sie sich auf alle bereits angegebenen ASSIGN-OUTPUT-FILE-Anweisungen. ASSIGN-OUTPUT-FILE-Anweisungen, die später angegeben werden, sind nicht betroffen.

Mit Ausnahme der Anweisung MODIFY-PERCON-OPTIONS gelten alle Anweisungen nur für einen Konvertierungsschritt. Sie müssen vollständig angegeben werden, da sie nicht nachträglich ergänzt werden können.

Die Anweisung MODIFY-PERCON-OPTIONS gilt, bis sie neu definiert oder der PERCON-Lauf beendet wird.

Über ein /SET-FILE-LINK-Kommando erzeugte TFT-Einträge bleiben während des gesamten PERCON-Laufs erhalten, wenn nicht explizit eine Freigabe der Dateikettungsamen gefordert wird (Operand REMOVE-FILE-LINK in den Anweisungen ASSIGN-INPUT-FILE und ASSIGN-OUTPUT-FILE).

**Beispiel**

```

/SET-FILE-LINK LINK-NAME=AUS1,-
/          FILE-NAME=AUSDAT,-
/          ACCESS-METHOD=SAM _____ (1)
/SET-FILE-LINK LINK-NAME=AUS2,-
/          FILE-NAME=TESTDAT,-
/          ACCESS-METHOD=SAM _____ (2)
/START-PERCON _____ (3)
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=EINDAT),-
//          LINK-NAME=EIN1 _____ (4)
//ASSIGN-OUTPUT-FILE LINK-NAME=AUS1 _____ (5)
//ASSIGN-OUTPUT-FILE LINK-NAME=AUS2 _____ (6)
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=AUS1,-
//          CONDITION=((15,1)=ALPHA) _____ (7)
//START-CONVERSION _____ (8)

```

- (1) Die Ausgabedatei AUSDAT wird mit dem Dateikettungsnamen AUS1 und ihren Dateieigenschaften zugewiesen.
- (2) Die Ausgabedatei TESTDAT wird mit dem Dateikettungsnamen AUS2 und ihren Dateieigenschaften zugewiesen.
- (3) PERCON wird aufgerufen.
- (4) Die Datei EINDAT wird als Eingabedatei mit dem Dateikettungsnamen EIN1 zugewiesen.
- (5) Die Ausgabedatei AUSDAT wird über den Dateikettungsnamen AUS1 mit PERCON verknüpft.
- (6) Die Ausgabedatei TESTDAT wird über den Dateikettungsnamen AUS2 mit PERCON verknüpft.
- (7) Die SELECT-INPUT-RECORDS-Anweisung bezieht sich zur Satzauswahl über den Dateikettungsnamen AUS1 auf die Ausgabedatei AUSDAT.
- (8) Der Übertragungsvorgang wird gestartet.

## 3.7 Zuweisen der Ein- und Ausgabe

Das folgende Kapitel beschreibt, wie Platten-Eingabe- bzw. Ausgabedateien oder Band-Eingabe- bzw. Ausgabedateien zugewiesen werden. Die dafür verwendeten BS2000-Kommandos sind in den Handbüchern „Kommandos“ Band 1 - 4 [11] beschrieben.

### Wertigkeit der Zuweisungen

Die Angabe von Dateimerkmalen mit dem SET-FILE-LINK-Kommando hat Vorrang vor Angaben in ASSIGN-INPUT-FILE- bzw. ASSIGN-OUTPUT-FILE-Anweisungen und vor Katalogeinträgen. Bei bereits katalogisierten Eingabedateien ist darauf zu achten, daß die Angaben im SET-FILE-LINK-Kommando nicht den Katalogeinträgen oder den ASSIGN-INPUT-FILE-Angaben widersprechen. In diesem Fall werden immer die SET-FILE-LINK-Angaben ausgewertet.

Die Angabe von Dateimerkmalen über ASSIGN-INPUT-FILE- bzw. ASSIGN-OUTPUT-FILE-Anweisungen hat Vorrang vor Katalogeinträgen. Bei bereits katalogisierten Eingabedateien ist auch hier zu beachten, daß sich die Angaben in den ASSIGN-INPUT-FILE-Anweisungen und die Katalogeinträge nicht widersprechen, da sonst Fehler auftreten.

Dateimerkmale, die weder über das SET-FILE-LINK-Kommando, noch über ASSIGN-INPUT-FILE bzw. ASSIGN-OUTPUT-FILE-Anweisungen zugewiesen wurden, werden mit den Katalogeinträgen - soweit vorhanden - versorgt.

Zusammenfassend läßt sich die Wertigkeit der Zuweisungen durch folgende Hierarchie darstellen:

- SET-FILE-LINK-Kommando
- PERCON-Steuer-Anweisungen
- Katalogeintrag
- Standardwerte von PERCON

### Eingabedatei

Um Platten- oder Band-Eingabedateien zuzuweisen, sind folgende BS2000-Kommandos und PERCON-Anweisungen von Bedeutung:

#### 1. IMPORT-FILE-Kommando

Es ist anzugeben, wenn sich die zu bearbeitende Datei auf einem privaten Datenträger befindet und noch keinen Katalogeintrag hat.

## 2. SET-FILE-LINK-Kommando

Es ist anzugeben, wenn Standardverarbeitungsoperanden, wie z.B. OPEN-MODE=\*INPUT, verändert werden sollen. Unterschiedliche Angaben zu den Dateimerkmalen im Katalogeintrag führen zu einem Dateieröffnungsfehler. Eine einzige Datei kann über den Standarddateikettungsname PCIN zugewiesen werden.

## 3. ASSIGN-INPUT-FILE-Anweisung

Sie ist anzugeben, wenn nicht der Standarddateikettungsname PCIN verwendet wird. Bei mehreren Eingabedateien ist jede über eine eigene ASSIGN-INPUT-FILE-Anweisung zuzuweisen und für jede muß ein anderer Dateikettungsname verwendet werden. Die Dateien werden in der Reihenfolge ihrer Angabe nacheinander bearbeitet. Über den Dateikettungsname wird der Bezug zu einem angegebenen SET-FILE-LINK-Kommando hergestellt.

Wird der Name einer katalogisierten Datei sowohl im SET-FILE-LINK-Kommando als auch in der ASSIGN-INPUT-FILE-Anweisung angegeben, hat der Dateiname im SET-FILE-LINK-Kommando Vorrang. Stimmen die Dateinamen nicht überein, wird die Meldung PER0012 ausgegeben.

Soll die Eingabe über SYSDTA oder DISKETTE erfolgen, so sind lediglich Angaben in der ASSIGN-INPUT-FILE-Anweisung nötig.

## Ausgabedatei

Um Ausgabedateien auf Platte oder Band zuzuweisen, sind folgende BS2000-Kommandos und PERCON-Anweisungen von Bedeutung:

### 1. CREATE-FILE-Kommando

Es ist anzugeben, wenn die Datei noch nicht existiert und

- sie auf private Datenträger auszugeben ist,
- für PRIMARY-ALLOCATION oder SECONDARY-ALLOCATION von der Standardzuweisung abweichende Werte angegeben werden sollen.

### 2. SET-FILE-LINK-Kommando

Es ist anzugeben, wenn die Dateimerkmale

- einer bereits vorhandenen Datei von den Einträgen im Katalog abweichen sollen,
- einer neu einzurichtenden Datei von den Standardwerten des PERCON abweichen sollen.

### 3. ASSIGN-OUTPUT-FILE-Anweisung

Sie ist anzugeben, wenn nicht der Standarddateikettungsname PCOUT verwendet wird. Bei mehreren Dateien ist jede über eine eigene ASSIGN-OUTPUT-FILE-Anweisung zuzuweisen und für jede muß ein anderer Dateikettungsname verwendet werden. In die Dateien wird gleichzeitig geschrieben. Über den Dateikettungsnamen wird der Bezug zu einem angegebenen SET-FILE-LINK-Kommando hergestellt.

Wird der Name einer katalogisierten Datei sowohl im SET-FILE-LINK-Kommando als auch in der ASSIGN-OUTPUT-FILE-Anweisung angegeben, hat der Dateiname im SET-FILE-LINK-Kommando Vorrang. Stimmen die Dateinamen nicht überein, wird die Meldung PER0012 ausgegeben.

Soll die Ausgabe über SYSLST, SYSOUT oder DISKETTE erfolgen, so sind lediglich Angaben in der ASSIGN-OUTPUT-FILE-Anweisung nötig.

### **Eingabeband**

Beim Editieren kann ein Band, beim Duplizieren können mehrere Bänder zugewiesen werden. Dazu muß die Archivnummer in der ASSIGN-INPUT-TAPE-Anweisung angegeben werden. Die maximale Länge der Eingabeblocks wird im Operanden BUFFER-LENGTH angegeben. Ist ein Block länger, wird die im Operanden LENGTH-ERROR angegebene Aktion ausgeführt. Dies gilt nicht für Magnetbandkassetten und für Bandgeräte mit Typ-2-Kanalsteuerungen, bei denen die Eingabeblocks auf den BUFFER-LENGTH-Wert gekürzt werden.

### **Ausgabeband**

Beim Duplizieren von Bändern können mehrere Ausgabebänder zugewiesen werden. Dazu müssen die Archivnummern in der ASSIGN-OUTPUT-TAPE-Anweisung angegeben werden.

### 3.8 Maßnahmen im OPENC-Ausgang

Kann eine Datei nicht eröffnet werden, weil sie während einer vorhergehenden Bearbeitung nicht ordnungsgemäß geschlossen wurde (z.B. bei einem System-Absturz), wird zum EXLST-Ausgang OPENC verzweigt. Dort kann dann der Makro VERIF genutzt werden, um die Datei zu entsperren und die Konsistenz wiederherzustellen.

Bei ISAM-Dateien kann es durch die Wiederherstellung zu Änderungen am Datei-Inhalt kommen. Bei den anderen Zugriffsmethoden wird die Datei nur entsperrt und eventuell der LAST-PAGE-POINTER aktualisiert. Aus diesem Grunde geht PERCON, abhängig von der Zugriffsmethode, im OPENC-Ausgang unterschiedlich vor.

Folgende Maßnahmen sind im OPENC-Ausgang vorgesehen:

- Nicht-ISAM-Dateien:  
Da es in diesem Fall zu keiner Veränderung des Datei-Inhalts kommt, wird immer eine Wiederherstellung mit dem Makro VERIF versucht. Der Benutzer wird davon mit der Meldung PER0092 informiert. Tritt dabei ein Fehler auf, wird zusätzlich die Meldung PER0016 ausgegeben.
- ISAM-Dateien im Dialogbetrieb:  
Der Benutzer wird mit der Meldung PER0093 gefragt, ob eine Wiederherstellung versucht werden soll oder nicht. Stimmt er zu (Antwort YES), so wird mit dem Makro VERIF die Wiederherstellung versucht, anderenfalls (Antwort NO) wird der OPENC-Ausgang verlassen. Tritt beim Abarbeiten von VERIF ein Fehler auf, wird zusätzlich die Meldung PER0016 ausgegeben.
- ISAM-Dateien und kein Dialogbetrieb:  
Der Benutzer wird mit der Meldung PER0094 von der Notwendigkeit einer Wiederherstellung informiert. Da aber keine Zustimmung des Benutzers eingeholt werden kann, erfolgt keine Wiederherstellung. Danach wird der OPENC-Ausgang verlassen.

Wird im Falle einer Wiederherstellung einer Datei kein Fehler gemeldet, versucht PERCON nach dem Verlassen des OPENC-Ausgangs ein erneutes OPEN .

Wird keine Wiederherstellung durchgeführt oder tritt bei der Wiederherstellung ein Fehler auf, so wird bei Eingabedateien entsprechend den Festlegungen im Operanden OPEN-ERROR der Anweisung ASSIGN-INPUT-FILE vorgefahren. Bei Ausgabedateien wird der Konvertierungsschritt abgebrochen.



## 3.9 Bearbeiten katalogisierter Dateien

### 3.9.1 Dateimerkmale

Alle im folgenden beschriebenen Dateimerkmale können über ein SET-FILE-LINK-Kommando (siehe Handbücher „Kommandos“ Band 1 - 4 [11]) zugewiesen werden.

Wenn keine Angaben gemacht werden und das Dateimerkmal auch nicht dem Katalogeintrag bzw. den Bandkennsätzen entnommen werden kann, dann tritt der von PERCON festgelegte Standardwert in Kraft, siehe auch Abschnitt „Wertigkeit der Zuweisungen“ auf Seite 29.

#### ACCESS-METHOD

Zugriffsmethode der Datei. PERCON kann die Zugriffsmethoden SAM, ISAM, PAM und BTAM verarbeiten.

Standardwert:

Eingabedateien werden mit derjenigen Zugriffsmethode gelesen, die im Katalog eingetragen ist.

Für Ausgabedateien gilt:

Kopieren einer Datei in eine Plattendatei: `ACCESS-METHOD=*ISAM`

Kopieren einer Datei in eine Banddatei: `ACCESS-METHOD=*SAM`

Editieren eines Bandes: `ACCESS-METHOD=*SAM`

Formataufbereitete Dateien sind immer SAM-Dateien.

Soll die Ausgabedatei eine ISAM-Datei sein, müssen die auszugebenden Sätze nach dem angegebenen Schlüssel aufsteigend sortiert sein. Als Schlüssel eignet sich z.B. der Satz-zähler (RECORD-COUNTER), der über eine SET-RECORD-MAPPING-Anweisung in den Ausgabesatz eingefügt werden kann.

#### BLOCK-CONTROL-INFO

Beschreibung der Blockstruktur. Dieses Dateimerkmal ist aufgrund der PAMKEY-Eliminierung (siehe Seite 38) entstanden. Es wirkt sich auf die Zugriffsmethoden SAM, ISAM und PAM aus.

Es wird unterschieden zwischen:

PAMKEY: Jedem 2K-Block sind 16 byte Verwaltungsinformation (PAM-Schlüssel) zugeordnet, die nicht Bestandteil des 2K-Blocks sind.

**WITHIN-DATA:** Jeder logische Block beinhaltet 12 byte Systeminformation. Die Systeminformation enthält im wesentlichen diejenigen Daten, die bei BLOCK-CONTROL-INFO=\*PAMKEY im PAM-Schlüssel untergebracht sind.

**NO:** Die Datei enthält keine Systeminformation.

Das Dateimerkmal BLOCK-CONTROL-INFO ist Bestandteil des Katalogeintrages. Neu zu erstellende Dateien erhalten das Merkmal aufgrund einer Defaultregelung abhängig von der BS2000-Version, dem Datenträger, dem Klasse2-Systemparameter BLKCTRL und der Dateizugriffsmethode.

## **BUFFER-LENGTH**

Blockgröße der Datei. Dateien bestehen aus Standardblöcken (BUFFER-LENGTH=\*STD). Nur bei Banddateien können auch Nicht-Standardblöcke (BUFFER-LENGTH=\*NSTD) vorkommen.

Die Blockgröße hat Einfluß auf die maximale Satzlänge. Ist die Länge des auszugebenden Satzes zu groß, wird er abgeschnitten.

Für formataufbereitete Dateien gilt immer BUFFER-LENGTH=\*STD(SIZE=n), wobei der Wert für SIZE abhängt vom Pubsetformat:

SIZE=1 bei 2K-formatierten Pubsets,

SIZE=2 bei 4K-formatierten Pubsets.

Für neu angelegte Ausgabedateien gilt dieser Wert standardmäßig.

Die Primärgröße einer Datei sollte mindestens die zweifache BUFFER-LENGTH, die Sekundärzuweisung mindestens die einfache BUFFER-LENGTH aufweisen.

## **RECORD-FORMAT**

Satzformat der Dateien. PERCON unterstützt die Satzformate FIXED, VARIABLE und UNDEFINED.

Standardwert: RECORD-FORMAT=\*VARIABLE

Formataufbereitete Dateien haben immer variables Satzformat.

## **RECORD-SIZE**

Satzlänge. Bei RECORD-FORMAT=\*FIXED wird die Länge, bei RECORD-FORMAT=\*VARIABLE die maximale Länge der Sätze angegeben. Ist ein Ausgabesatz länger als im Operand RECORD-SIZE angegeben, wird der Satz abgeschnitten. Ist bei RECORD-FORMAT=\*FIXED der Ausgabesatz kürzer als im Operand RECORD-SIZE angegeben, wird der Satz mit Leerzeichen (X'40') aufgefüllt.

Wurde keine Satzlänge angegeben oder überragt sie den laut nachstehender Tabelle aus BUFFER-LENGTH ermittelten Maximalwert, wird dieser Maximalwert für die Ausgabesatzlänge angenommen.

ACCESS-METHOD	BLOCK-CONTROL-INFO	RECORD-FORMAT	Maximalwert
SAM ohne Formataufbereitung	PAMKEY	VARIABLE FIXED UNDEFINED	BUFFER-LENGTH - 4 BUFFER-LENGTH BUFFER-LENGTH <sup>1)</sup>
	WITHIN-DATA-BLOCK	VARIABLE FIXED UNDEFINED	BUFFER-LENGTH - 16 BUFFER-LENGTH - 16 BUFFER-LENGTH - 16 <sup>1)</sup>
	NO <sup>2)</sup>	VARIABLE FIXED UNDEFINED	BUFFER-LENGTH - 4 BUFFER-LENGTH BUFFER-LENGTH <sup>1)</sup>
ISAM	PAMKEY	VARIABLE FIXED	BUFFER-LENGTH BUFFER-LENGTH - 4
	WITHIN-DATA-BLOCK	VARIABLE FIXED	BUFFER-LENGTH BUFFER-LENGTH - 4
PAM <sup>1)</sup>	PAMKEY		BUFFER-LENGTH 2048 <sup>3)</sup>
	WITHIN-DATA-BLOCK		BUFFER-LENGTH - 12
	NO		BUFFER-LENGTH
BTAM		VARIABLE FIXED UNDEFINED	BUFFER-LENGTH <sup>1)</sup> BUFFER-LENGTH BUFFER-LENGTH <sup>1)</sup>

- <sup>1)</sup> Die RECORD-SIZE-Angabe wird zur Ermittlung des Maximalwerts nicht herangezogen.
- <sup>2)</sup> Gilt nur für Banddateien.
- <sup>3)</sup> Gilt nur, wenn die aktuelle Eingabedatei ACCESS-METHOD=\*PAM und BLOCK-CONTROL-INFO=\*PAMKEY hat und keine Ausgabedatei BLOCK-CONTROL-INFO=\*WITHIN-DATA-BLOCK oder \*NO aufweist.

## KEY-POSITION

Position des ISAM-Schlüssels im Satz (nur bei ISAM-Dateien von Bedeutung).

Standardwert:

bei RECORD-FORMAT=\*FIXED: KEY-POSITION=1

bei RECORD-FORMAT=\*VARIABLE: KEY-POSITION=5

## KEY-LENGTH

Länge des ISAM-Schlüssels im Satz (nur bei ISAM-Dateien von Bedeutung).

Standardwert: KEY-LENGTH=8

## OPEN-MODE

Eröffnungsart der Datei. Über den Wert \*EXTEND können die auszugebenden Sätze an eine bestehende SAM- bzw. BTAM-Datei angefügt werden. Bei SAM-Dateien können die eingelesenen Sätze wieder in die Eingabedatei zurückgeschrieben werden, wenn Ein- und Ausgabedatei mit OPEN-MODE=\*UPDATE eröffnet wurden. Hierbei hat der aktuelle Ausgabesatz die gleiche Länge wie der aktuelle Eingabesatz. Gegebenenfalls wird er auf diese Länge gekürzt oder mit Leerzeichen aufgefüllt. Bei ISAM-Dateien können die Ausgabesätze auf Grund ihrer Schlüsselfelder an der richtigen Stelle eingefügt werden, wenn OPEN-MODE=\*OUTIN (bei neu zu erstellenden Dateien) oder OPEN-MODE=\*INOUT (bei bestehenden Dateien) angegeben wurde.

Standardwert für Eingabe: OPEN-MODE=\*INPUT

Standardwert für Ausgabe: PAM-Dateien: OPEN-MODE=\*OUTIN  
sonst: OPEN-MODE=\*OUTPUT

### 3.9.1.1 4K-Pubsets

Mit BS2000 V11 werden erstmals Platten mit 4Kbyte-Blockstruktur unterstützt. Hierbei muß die Blockgröße der Datei ein Vielfaches von 4K betragen, d.h. BUFFER-LENGTH=\*STD (SIZE=n) mit n geradzahlig. Außerdem muß die Blockstruktur ungleich BLOCK-CONTROL-INFO=\*PAMKEY sein.

Bei Ausgabedateien ist der Standardwert für BUFFER-LENGTH abhängig vom Pubset, auf welchem die Datei angelegt wird:

- 2K-formatierte Pubsets: BUFFER-LENGTH=\*STD(SIZE=1);
- 4K-formatierte Pubsets: BUFFER-LENGTH=\*STD(SIZE=2).

Zwischendateien für die Ausgabe auf Diskette werden immer mit BUFFER-LENGTH=\*STD(SIZE=2) angelegt.

### 3.9.1.2 4K-fähiges ISAM

Eine 4K-ISAM-Datei weist folgende Dateimerkmale auf:

BUFFER-LENGTH=\*STD(SIZE=n) mit n geradzahlig  
BLOCK-CONTROL-INFO=\*WITHIN-DATA-4K-BLOCK

Beide Dateimerkmale können über das Kommando SET-FILE-LINK oder FILE-Makro angegeben werden.

Eine 4K-ISAM-Datei wird unter folgenden Bedingungen angelegt:

Pubset: 4K  
BUFFER-LENGTH: nicht spezifiziert /  
\*STD(SIZE=n) mit n geradzahlig  
BLOCK-CONTROL-INFO: nicht spezifiziert /  
\*WITHIN-DATA-BLOCK /  
\*WITHIN-DATA-4K-BLOCK

Wird FILE-ATTRIBUTES=\*INPUT-FILE in der ASSIGN-OUTPUT-FILE-Anweisung angegeben, werden die Dateimerkmale Blockgröße und BLOCK-CONTROL-INFO der Eingabedatei in die Ausgabe übernommen.

Falls die Eingabedatei eine ISAM-Datei ist und BLOCK-CONTROL-INFO=\*WITHIN-DATA-2K-BLOCK aufweist, wird sie mit BLOCK-CONTROL-INFO=\*WITHIN-DATA-BLOCK eingetragen. Dadurch kann die Datei, abhängig vom Pubset, auf dem sie angelegt wird, mit WITHIN-DATA-2K-BLOCK oder WITHIN-DATA-4K-BLOCK eingerichtet werden.

## 3.9.2 Pamkey-Eliminierung

Vom BS2000 werden neben den CKD-Platten Platten mit fester Blockgröße (2Kbyte, 4Kbyte...) unterstützt. Diese festen Blockgrößen verhindern die einfache Unterbringung der PAM-Schlüssel. Aus diesem Grund wird der PAM-Schlüssel entfallen. Dieser Vorgang wird als Pamkey-Eliminierung bezeichnet.

Die im bisherigen PAM-Schlüssel enthaltene Systeminformation wird entweder im Datenblock in einem gesonderten Blockkontrollfeld untergebracht oder sie entfällt. Das Blockkontrollfeld belegt die ersten zwölf Bytes des Datenblocks und enthält u.a. die Coded File Identification, die zum Lückentest verwendet wird. Durch das Blockkontrollfeld reduziert sich die maximal nutzbare Satzlänge des Datenblocks.

Abhängig von der im SET-FILE-LINK-Kommando definierten Dateizugriffsmethode ergeben sich durch die Pamkey-Eliminierung neue Dateiformate.

Ab der BS2000 Version 9.5 können keylose ISAM-Dateien, ab der BS2000 Version 10.0 keylose SAM- und PAM-Dateien bearbeitet werden.

Im folgenden werden die Änderungen beschrieben, die für das Arbeiten mit PERCON von Bedeutung sind. Nähere Einzelheiten zu den neuen Dateiformaten siehe Handbuch „Einführung in das DVS“ [1].

SAM und ISAM-Dateien werden weiterhin satzweise gelesen und geschrieben. Dadurch ergeben sich für PERCON keine Änderungen in der Arbeitsweise.

BTAM-Dateien sind von der Pamkey-Eliminierung nicht betroffen.

Bei PAM-Dateien sind folgende Erweiterungen zu beachten:

### 3.9.2.1 PAM-Eingabedateien

- Bearbeitung von Leerblöcken - Operand EMPTY-BLOCK

Bisher werden Leerblöcke in der Eingabedatei überlesen. Durch die Einführung des neuen Operanden EMPTY-BLOCK in der ASSIGN-INPUT-FILE-Anweisung wird nun gesteuert, ob wie bisher konvertiert wird oder ein Leerblock (innerhalb einer Datei vor dem LAST-PAGE-POINTER bzw. vor HIGHEST-USED-PAGE) als Eingabesatz mit Inhalt binär 0 (X'00') verarbeitet wird.

- Blockstruktur - Operand BLOCK-CONTROL-INFO

BLOCK-CONTROL-INFO=\*PAMKEY

Jeder logische Block bildet einen Eingabesatz mit der Länge BUFFER-LENGTH. Zum Lückentest wird die im PAM-Schlüssel des ersten PAM-Blocks des logischen Blocks befindliche CFID (Coded File Identification) mit der im Katalog eingetragenen verglichen.

Bei Ungleichheit wird der logische Block gemäß dem Operanden EMPTY-BLOCK verarbeitet.

Die Anwenderdatenbereiche in den PAM-Schlüsseln werden, falls die Ausgabedatei ACCESS-METHOD=\*PAM und BLOCK-CONTROL-INFO=\*PAMKEY aufweist, an diese weitergegeben, sie sind jedoch nicht Bestandteil des Satzes.

*Hinweis*

Wenn die PAM-Eingabedatei BLOCK-CONTROL-INFO=\*PAMKEY und keine Ausgabedatei BLOCK-CONTROL-INFO=\*WITHIN-DATA bzw. \*NO aufweist, dann bildet jeder PAM-Block einen Eingabesatz mit einer Länge von 2048 byte. Zum Lückentest wird die im PAM-Schlüssel jedes PAM-Blocks befindliche Coded File Identification herangezogen. Bei Ungleichheit wird der PAM-Block gemäß dem Operanden EMPTY-BLOCK in der ASSIGN-INPUT-FILE-Anweisung verarbeitet.

**BLOCK-CONTROL-INFO=\*WITHIN-DATA**

Jeder logische Block bildet einen Eingabesatz mit der Länge BUFFER-LENGTH - 12. Das Blockkontrollfeld ist nicht Bestandteil des Eingabesatzes. Zum Lückentest wird die CFID im Blockkontrollfeld des logischen Blocks herangezogen. Bei Ungleichheit wird der logische Block gemäß dem Operanden EMPTY-BLOCK verarbeitet.

**BLOCK-CONTROL-INFO=\*NO**

Jeder logische Block bildet einen Eingabesatz mit der Länge BUFFER-LENGTH. Da kein Lückentest durchgeführt werden kann, werden alle logischen Blöcke verarbeitet. Blöcke am Ende dieser Datei, die nicht zu dieser gehören (Blocknummer > LAST-PAGE-POINTER bzw. HIGHEST-USED-PAGE), werden nicht übernommen.

### 3.9.2.2 PAM-Ausgabedateien

Die Arbeitsweise von PERCON bezüglich PAM-Ausgabedateien ist im wesentlichen vom Dateimerkmal BLOCK-CONTROL-INFO abhängig.

BLOCK-CONTROL-INFO=\*PAMKEY

Jeder Ausgabesatz hat die Größe eines logischen Blocks. Die Anwenderdatenbereiche der PAM-Schlüssel werden übernommen, wenn die aktuelle Eingabedatei ACCESS-METHOD=\*PAM aufweist. Sonst werden sie mit binär 0 versorgt.

#### *Hinweis*

Wenn die aktuelle Eingabedatei eine PAM-Datei mit BLOCK-CONTROL-INFO=\*PAMKEY ist und keine Ausgabedatei BLOCK-CONTROL-INFO=\*WITHIN-DATA bzw. \*NO aufweist, dann hat jeder Ausgabesatz die Größe eines PAM-Blocks (fixe Länge von 2048 byte). Der Blockungsfaktor n gibt die Anzahl der zu einem logischen Block zu kettenenden PAM-Blöcke an.

BLOCK-CONTROL-INFO=\*WITHIN-DATA

Jeder Ausgabesatz wird auf Blockanfang + 12 eingetragen. Dadurch hat der Ausgabesatz die Länge BUFFER-LENGTH - 12.

BLOCK-CONTROL-INFO=\*NO

Jeder Ausgabesatz hat die Größe eines logischen Blocks (= feste Länge von BUFFER-LENGTH).



### 3.9.2.3 Beispiele zum Kopieren von PAM-Dateien

#### Beispiel 1

Es soll eine PAM-Datei mit BLOCK-CONTROL-INFO=\*PAMKEY in eine PAM-Datei gleichen Dateiformats konvertiert werden. In der Eingabedatei bilden 2 PAM-Blöcke einen logischen Block, in der Ausgabedatei sollen 4 PAM-Blöcke einen logischen Block bilden.

Bei dieser Umwandlung werden die Daten des Anwenders in den PAM-Schlüsseln mitübertragen, sie sind aber nicht Bestandteil des Satzes und können deshalb nicht verändert werden.

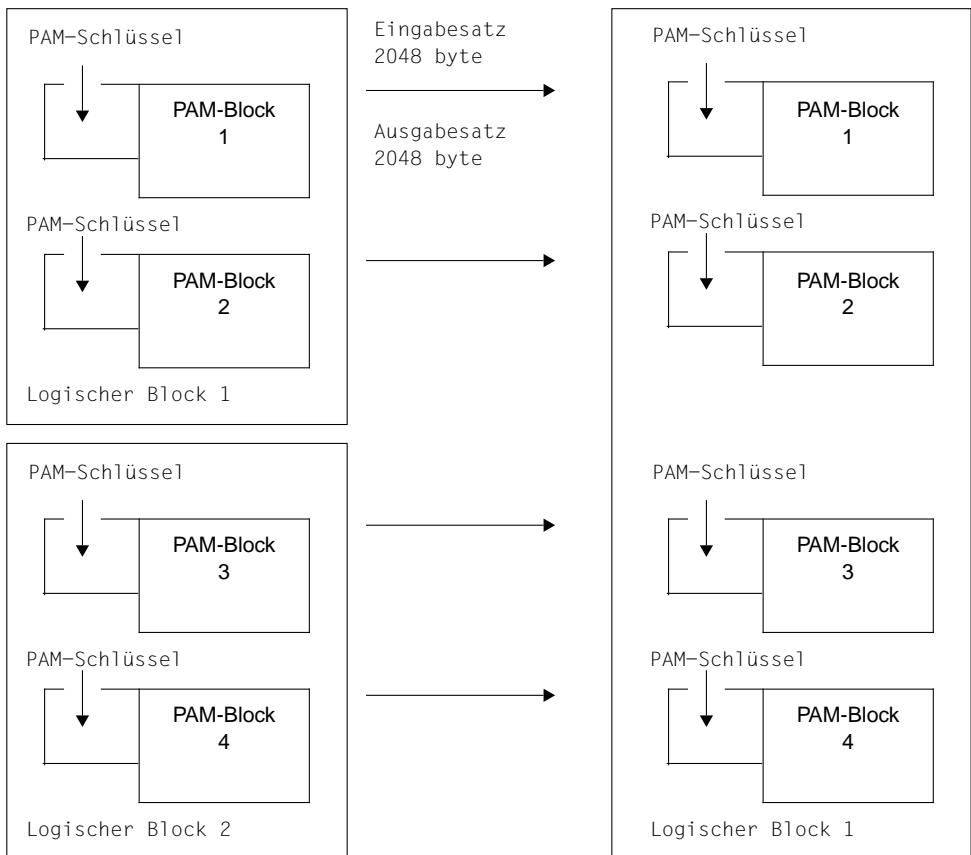
Folgende Dateimerkmale sind bei dieser Konvertierung von Bedeutung:

Eingabedatei:

```
ACCESS-METHOD=*PAM  
BUFFER-LENGTH=*STD(SIZE=2)  
BLOCK-CONTROL-INFO=*PAMKEY
```

Ausgabedatei:

```
ACCESS-METHOD=*PAM  
BUFFER-LENGTH=*STD(SIZE=4)  
BLOCK-CONTROL-INFO=*PAMKEY
```



## Beispiel 2

Es soll eine pamkeylose PAM-Datei in eine PAM-Datei mit PAM-Schlüssel konvertiert werden. In der Eingabedatei bilden 2 Standardblöcke einen logischen Block, in der Ausgabedatei sollen 4 PAM-Blöcke einen logischen Block bilden.

Die Größe des Eingabesatzes errechnet sich zu:

$$n \times \text{Standardgröße} - \text{Blockkontrollfeld} = 2 \times 2048 - 12 = 4084 \text{ byte.}$$

Das Blockkontrollfeld ist nicht Bestandteil des Eingabesatzes.

Die Größe des Ausgabesatzes ergibt sich aus:

$$m \times \text{Standardgröße} = 4 \times 2048 = 8192 \text{ byte.}$$

Bei dieser Konvertierung werden die Anwenderdatenbereiche in den PAM-Schlüsseln mit X'00' versorgt.

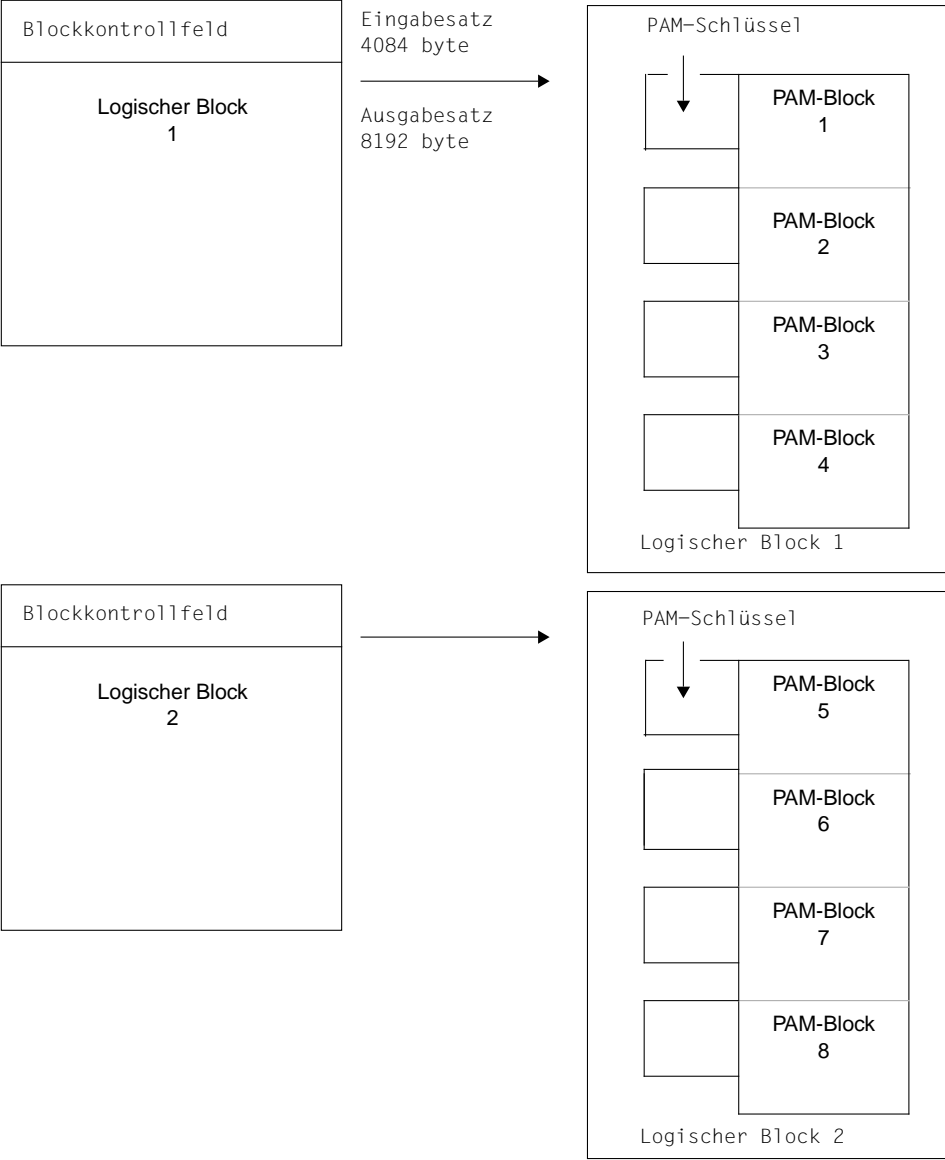
Folgende Dateierkmale sind bei dieser Konvertierung von Bedeutung:

Eingabedatei:

ACCESS-METHOD=\*PAM  
BUFFER-LENGTH=\*STD(SIZE=2)  
BLOCK-CONTROL-INFO=\*WITHIN-DATA

Ausgabedatei:

ACCESS-METHOD=\*PAM  
BUFFER-LENGTH=\*STD(SIZE=4)  
BLOCK-CONTROL-INFO=\*PAMKEY



Beispiel 3

Es soll eine PAM-Datei mit BLOCK-CONTROL-INFO=\*PAMKEY in eine pamkeylose PAM-Ausgabedatei mit BLOCK-CONTROL-INFO=\*NO konvertiert werden. In der Eingabedatei bilden 2 PAM-Blöcke einen logischen Block, in der Ausgabedatei soll ein logischer Block 4096 byte groß sein.

Bei dieser Umwandlung werden die Daten des Anwenders in den PAM-Schlüsseln nicht übertragen.

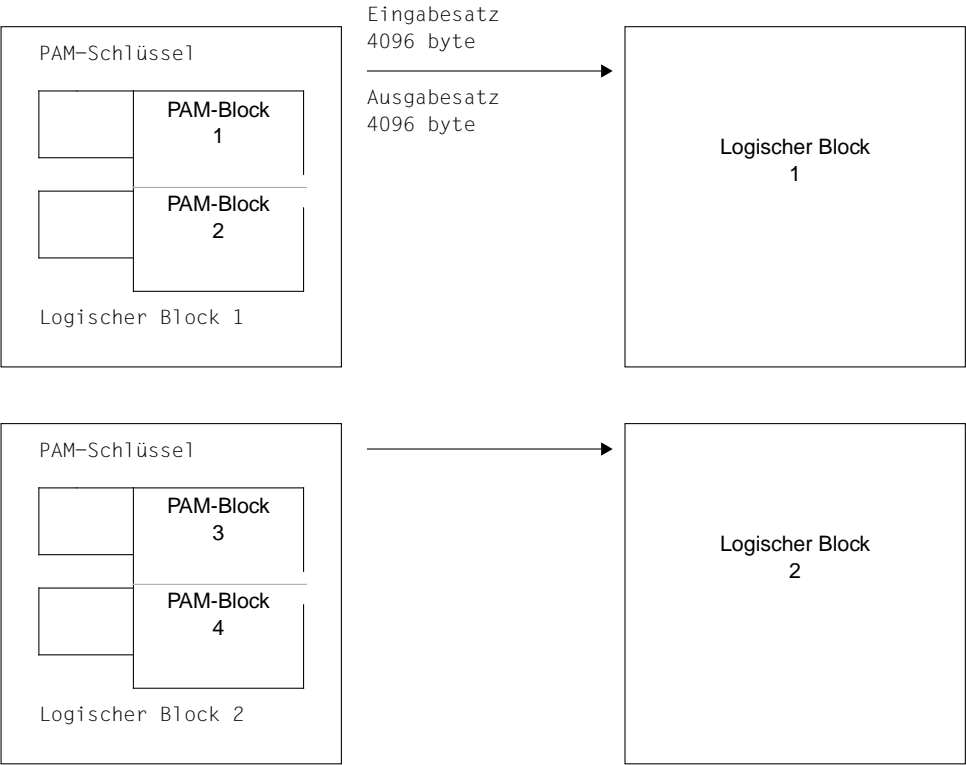
Folgende Dateierkmale sind bei dieser Konvertierung von Bedeutung:

Eingabedatei:

ACCESS-METHOD=\*PAM  
BUFFER-LENGTH=\*STD(SIZE=2)  
BLOCK-CONTROL-INFO=\*PAMKEY

Ausgabedatei:

ACCESS-METHOD=\*PAM  
BUFFER-LENGTH=\*STD(SIZE=2)  
BLOCK-CONTROL-INFO=\*NO



Beispiel 4

Es soll eine keylose PAM-Datei in eine ebensolche Ausgabedatei konvertiert werden. Ein- und Ausgabedatei weisen jedoch verschiedene Blockgröße und Blockstruktur auf:

Eingabedatei:

```
ACCESS-METHOD=*PAM
BUFFER-LENGTH=*STD(SIZE=3)
BLOCK-CONTROL-INFO=*NO
```

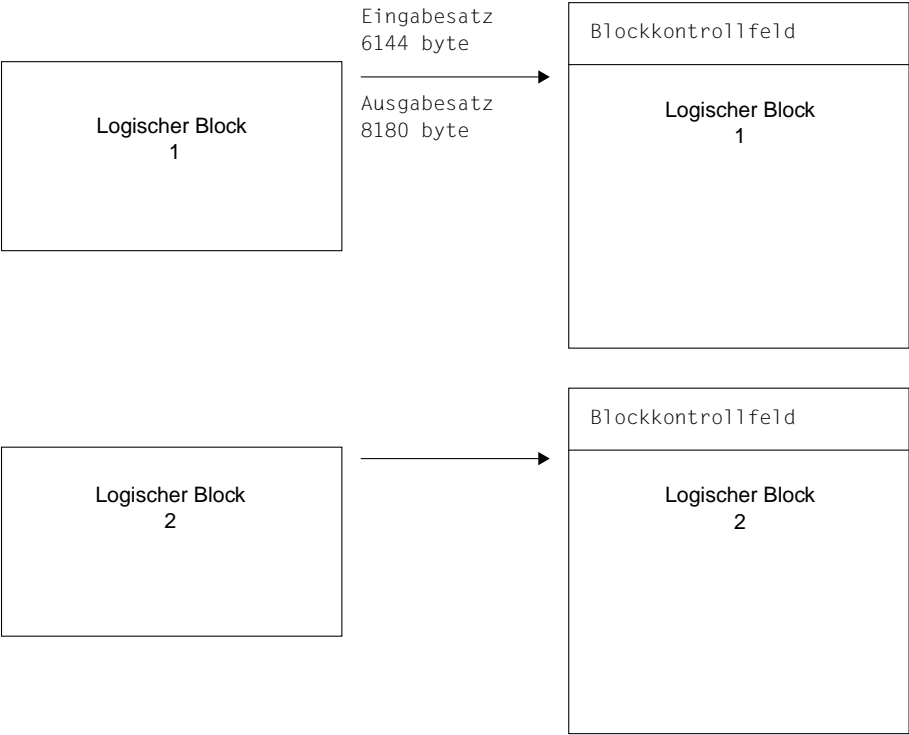
Ausgabedatei:

```
ACCESS-METHOD=*PAM
BUFFER-LENGTH=*STD(SIZE=4)
BLOCK-CONTROL-INFO=*WITHIN-DATA
```

Die Größe eines Ausgabesatzes errechnet sich zu:

$$n \times \text{Standardgröße} - \text{Blockkontrollfeld} = 4 \times 2048 - 12 = 8180 \text{ byte.}$$

Der Ausgabesatz wird hinter das Blockkontrollfeld in den logischen Block geschrieben.



### 3.9.3 Übernehmen von Sekundärschlüsseln

Beim Umsetzen von NK-ISAM-Dateien werden eventuell existierende Sekundärschlüssel der Eingabedateien nicht automatisch für die Ausgabedateien erzeugt. Bei Bedarf ermittelt aber PERCON die Sekundärschlüssel von Eingabedateien und baut diese nach dem Umsetzen für Ausgabedateien wieder auf. Da i. a. mehrere Ein- und mehrere Ausgabedateien möglich sind, muß der Benutzer mitteilen, ob und für welche Dateien er diese Funktion nutzen will.

Der Aufbau der Sekundärschlüssel für Ausgabedateien erfolgt unmittelbar vor dem Ende eines Konvertierungsschritts, wenn bis dahin kein Ereignis eingetreten ist, das eine abnormale Beendigung des Konvertierungsschritts verlangt, d.h. alle Ausgabedateien müssen erfolgreich erzeugt worden sein.

Ist das der Fall, so werden von der angegebenen Eingabedatei für alle Sekundärschlüssel die vereinbarten Werte für den Namen, die Position und die Länge sowie die DUPKEY-Eigenschaft ermittelt. Danach werden alle Sekundärschlüssel für die Ausgabedatei mit diesen Werten aufgebaut.

Ist die Ausgabedatei bereits mit einem der zu übernehmenden Sekundärschlüssel versehen und die vereinbarten Werte für Position, Länge und DUPKEY-Eigenschaft stimmen mit den Werten in der Eingabedatei überein, so wird ohne weitere Reaktion die Verarbeitung fortgesetzt.

Kommt es beim Übernehmen der Sekundärschlüssel zu einem Fehler, so wird eine der Meldungen PER0099 bis PER0103 ausgegeben. Anschließend wird der Konvertierungsschritt abgebrochen.

Ist die Eingabedatei jedoch eine NK-ISAM-Datei ohne Sekundärschlüssel, so wird nach einer Warnung PER0097 die Verarbeitung normal fortgesetzt.

Folgende Ausnahmesituationen können bei der Übernahme der Sekundärschlüssel auftreten:

1. Ein- oder Ausgabedatei ist keine NK-ISAM-Datei.
2. Eingabedatei besitzt keine Sekundärschlüssel (nur Warnung PER0097).
3. Die Ausgabedatei besitzt mehrfache Primärschlüssel oder logische Markierungen oder Wertmarkierungen.
4. Die Ausgabedatei ist bereits mit einem der zu übernehmenden Sekundärschlüssel versehen, die vereinbarten Werte für Position, Länge oder DUPKEY-Eigenschaft weichen jedoch von den Werten in der Eingabedatei ab.
5. Bei der Übernahme wird die Maximalzahl von 30 Sekundärschlüsseln überschritten.
6. Für den Sekundärschlüssel ist DUPKEY=\*NO vereinbart, es kommen jedoch gleiche Werte des Sekundärschlüssels in verschiedenen Sätzen der Ausgabedatei vor.

7. Mindestens ein Satz in der Ausgabedatei ist zu kurz, um den Sekundärschlüssel vollständig aufnehmen zu können.
8. Sonstige unerwartete Systemfehler.

Bei der Grundfunktion „Editieren“ wird der Operand ALTERNATE-INDEX ignoriert.

### 3.10 PERCON in XS-Umgebung (31-Bit-Adressierung)

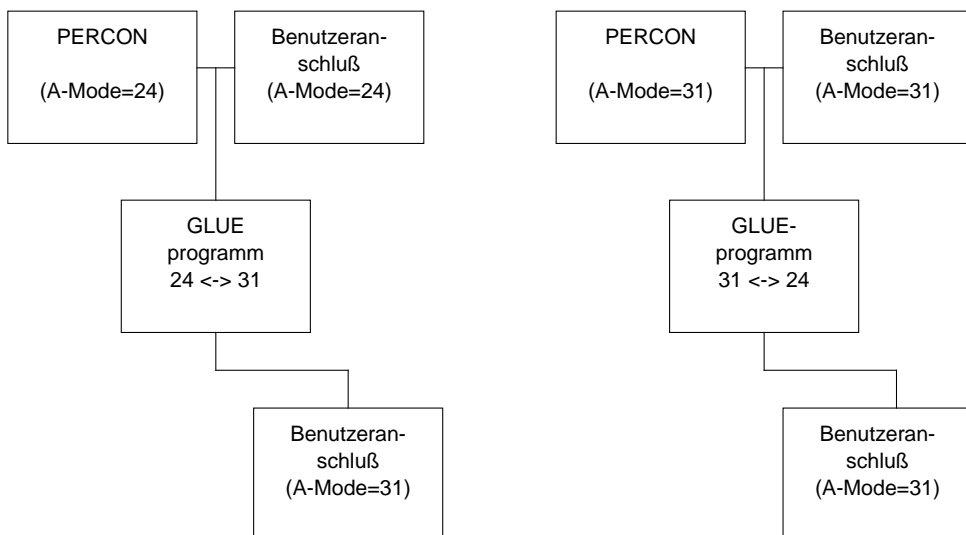
PERCON ist im 24- und 31-Bit-Adressierungsmodus ablauffähig. Sinnvoll ist die Verwendung des 31-Bit-Adressierungsmodus, wenn ein entsprechender Bedarf an virtuellem Speicher vorhanden ist (>16 Mbyte) und PERCON als Unterprogramm aufgerufen werden soll oder Benutzermodule angeschlossen werden sollen.

Standardmäßig läuft PERCON bei autonomem Aufruf im 24-Bit-Adressierungsmodus. Soll der 31-Bit-Adressierungsmodus eingeschaltet werden, muß PERCON mit `/START-PERCON PROGRAM-MODE=ANY` gestartet werden.

Wird PERCON als Unterprogramm aufgerufen und muß der zum Hauptprogramm hinzugebundene Bindemodul PCROOT (siehe Seite 171) den Adressierungsmodus wechseln (weil z.B. PERCON als Subsystem in den oberen Adreßraum geladen wurde), so wird bei Rückkehr ins Hauptprogramm wieder der alte Adressierungsmodus eingestellt.

Bei Verwendung von Benutzeranschlüssen muß folgendes beachtet werden:

1. PERCON verändert nicht den Adressierungsmodus beim Aufruf eines Benutzeranschlusses. Stimmen die Adressierungsmodi von PERCON und Benutzeranschluß nicht überein,
  - muß entweder der Benutzeranschluß an den Adressierungsmodus von PERCON angepaßt werden,
  - oder, falls das nicht möglich ist, muß ein sogenanntes GLUE-Programm dazwischengeschaltet werden, um die Modus-Umschaltung vorzunehmen (siehe Handbuch „Einführung in die XS-Programmierung“ [7]).





2. Unabhängig vom Adressierungsmodus benutzt PERCON die 31-Bit-Systemschnittstelle (PARMOD=31). Deshalb wird bei Benutzeranschlüssen für Kennsatzbearbeitung ein FILE CONTROL BLOCK (FCB) im 31-Bit-Format angeboten.

### 3.11 Verwendung erweiterter Zeichensätze in PERCON

Rechenanlagen (Hosts) und Datensichtstationen arbeiten mit je einem Satz von Buchstaben, Ziffern und Zeichen, aus denen Wörter und andere elementare Bestandteile einer Sprache aufgebaut sind, dem sogenannten **Zeichensatz (character set)**.

Durch die Erweiterung dieser Zeichensätze können landesspezifische Zeichendarstellungen, wie z.B. Umlaute (deutsch) oder Akzente (französisch), innerhalb eines Zeichensatzes gleichzeitig angeboten werden.

Ein **codierter Zeichensatz (coded character set, CCS)** ist die eindeutige Darstellung der Zeichen eines Zeichensatzes in binärer Form. Der Inhalt eines codierten Zeichensatzes und seine Regeln, wie z.B. die Sortierreihenfolge und Konvertierungsvorschriften, sind durch internationale Normen festgelegt.

Beispiel: Das Zeichen „ä“ ist im codierten Zeichensatz EBCDIC.DF.03-DRV (deutsche Referenzversion) durch das Byte X'FB', in EBCDIC.DF.04-1 durch X'43' dargestellt.

Jeder codierte Zeichensatz, kurz Code, wird durch seinen eindeutigen Namen (**coded character set name, CCSN**) bestimmt.

Beispiel: Der Code EBCDIC.DF.03-IRV (Internationale Referenzversion) hat den Namen „EDF03IRV“.

Eine Liste der existierenden Codes befindet sich im Anhang des Handbuchs „XHCS“ [2].

Seit BS2000 V10.0 besteht die Möglichkeit, **erweiterte Codes** zu benutzen. Erweiterte Codes ergänzen die bestehenden

7-bit-Codes     EBCDIC.DF.03 für Hosts und  
                  ISO646 für Datensichtstationen  
                  mit etwa 90 genutzten Zeichen

um

8-bit-Codes     EBCDIC.DF.04-x für Hosts und  
                  ISO8859-x für Datensichtstationen  
                  mit etwa 190 genutzten Zeichen.

*Hinweis*

In BS2000 V10.0 steht nur ein erweiterter Code je Host zur Verfügung. Dieser wird von der Systemverwaltung aus den möglichen Codes ausgewählt und als Systemstandard-Code eingerichtet.

Ab BS2000/OSD-BC V1.0 können mehrere Codes parallel genutzt werden.

### 3.11.1 Hard- und Softwarevoraussetzungen

Sollen die erweiterten Codes für PERCON benutzt werden, müssen die nachfolgenden Voraussetzungen erfüllt sein:

Software:

- PERCON ab V2.5A
- BS2000 ab OSD-BC V1.0
- Softwareprodukt XHCS (optionales Subsystem XHCS-SYS)
- Softwarekomponente VTSU

Zur Erstellung erweiterter Codes im Host und zur Datenübertragung zwischen Host und Datensichtstation wird das Softwareprodukt **XHCS** (e**X**tended **H**ost **C**ode **S**upport) benötigt. Im Handbuch „XHCS“ [2] befindet sich eine detaillierte Beschreibung der Prinzipien und Funktionen von XHCS sowie eine Auflistung der Code-Tabellen und der Namen von Standard-Codes.

Als Hardware werden für die Ein-/Ausgabe erweiterter Zeichensätze auf Datensichtstationen sogenannte 8-bit-Terminals benötigt.

Geprüft wird die 8-bit-Fähigkeit von Datensichtstationen mit der Softwarekomponente VTSU.

### 3.11.2 PERCON-spezifische Anwendung erweiterter Zeichensätze

Die Verwendung erweiterter Codes ist ab PERCON V2.5A möglich.

Zur richtigen Interpretation der an Schnittstellen übergebenen Daten (Anweisungen, Eingabesätze, Ausgabesätze) muß man wissen, welchem CCS sie angehören.

Kann den Übergabewerten einer Schnittstelle kein CCSN zugeordnet werden, so wird intern wie vor der Einführung von XHCS gearbeitet.

Beim Aufbereiten einer Datei wird, wenn vorhanden, die Tabelle der druckbaren Zeichen des Ausgabe-CCS verwendet. Nichtdruckbare Zeichen werden durch Leerzeichen ersetzt. Hat die Datei kein Ausgabe-CCS, so werden wie bisher die Zeichen X'00' bis X'3F' durch Leerzeichen ersetzt.

ISO-CCS werden von PERCON nicht unterstützt.

## Dateiorientierte Eingabe

PERCON beachtet das CCS der Eingabequelle.

Innerhalb einer über ASSIGN-INPUT-FILE zugeordneten Eingabequelle darf der CCSN nicht wechseln. Das ist beim Einlesen über SYSDTA bei einem Wechsel der Zuordnung zu beachten.

Eine Datei auf einer Diskette kann keinen CCSN erhalten, da dort in den Kennsätzen kein Platz zum Speichern dieser Information vorhanden ist. Dateien auf Disketten haben keinen Katalogeintrag.

Dateien auf Privatplatten haben ebenfalls keinen CCSN, da in deren Katalogeintrag kein Platz dafür vorgesehen ist.

## Datenträgerorientierte Eingabe

Die Blöcke des Bandes werden ohne CCS bereitgestellt.

Mit dem Operanden CODE der ASSIGN-INPUT-TAPE-Anweisung wird der Bandcode beschrieben:

CODE=\*EBCDIC:

Beim Lesen findet keine Code-Umsetzung statt.

CODE=\*ISO7:

Das Band ist im ISO6461-IRV erstellt. Alle Bytes eines Blocks werden beim Lesen in den EBCDI-Code umcodiert.

CODE=\*OWN:

Mit einer Umsetztabelle wird der Bandcode beschrieben. Alle Bytes eines Blocks werden beim Lesen umcodiert.

## Dateiorientierte Ausgabe

- Katalogisierte Datei  
FILE-ATTRIBUTES=\*STD Der im Katalogeintrag vermerkte CCSN wird verwendet. Existiert noch kein Katalogeintrag, so wird er von PERCON mit dem CCSN der ersten Eingabe des Konvertierungsschrittes erzeugt.  
FILE-ATTRIBUTES=\*INPUT-FILE Die Datei erhält den CCSN der ersten Eingabe des Konvertierungsschrittes.
- SYSOUT  
Ist SYSOUT dem Terminal zugewiesen, wird als Ausgabe-CCSN der CCSN der ersten Eingabe des Konvertierungsschrittes verwendet.  
Ist SYSOUT einer katalogisierten Datei zugewiesen, so wird deren CCSN verwendet.

- SYSLST  
Es wird der CCSN der SYSLST-Datei verwendet.
- Diskette  
Die Datei, die nach der Konvertierung an SPOOL weitergegeben wird, erhält den CCSN der ersten Eingabe des Konvertierungsschrittes.
- Bei Dateien auf Privatplatten kann im Katalogeintrag kein CCSN abgespeichert werden.

### **Datenträgerorientierte Ausgabe**

Den Blöcken des Bandes wird kein CCS zugeordnet.

Mit dem Operanden CODE der ASSIGN-OUTPUT-TAPE Anweisung wird der Bandcode beschrieben.

CODE=\*EBCDIC:

Beim Schreiben findet keine Code-Umsetzung statt.

CODE=\*ISO7:

Das Band wird im ISO6461-IRV erstellt. Alle Bytes eines Blocks werden beim Schreiben umcodiert.

CODE=\*OWN:

Mit einer Umsetztabelle wird der Bandcode beschrieben. Alle Bytes eines Blocks werden beim Schreiben umcodiert.

## **3.11.3 Verarbeitungsfunktionen**

PERCON berücksichtigt bei der Verarbeitung das Anweisungs-, das Eingabe- und das Ausgabe-CCS. Haben alle CCS den gleichen Namen, so sind keine besonderen Regeln zu beachten. Werden unterschiedliche Namen vergeben, gelten folgende Regeln:

### **Eingabe-CCS ungleich Ausgabe-CCS**

Ist das Eingabe-CCS ungleich dem Ausgabe-CCS, so muß folgendes beachtet werden:

Voraussetzung für eine Konvertierung ist, daß Eingabe-CCS und Ausgabe-CCS kompatible CCS sind.

Wird keine SET-RECORD-MAPPING-Anweisung angegeben, so werden die Zeichen des Eingabesatzes bei der Übertragung in den Ausgabebereich vom Eingabe-CCS in Zeichen des Ausgabe-CCS umgewandelt, sofern die Eingabedatei ein CCS ungleich EDF03IRV besitzt.

Bei Angabe von FILLER=\*INPUT in der SET-RECORD-MAPPING-Anweisung wird der Ausgabe-Satz mit dem Eingabe-Satz, unabhängig von deren CCS, vorbelegt.

Die Übertragung von Feldern des Eingabe-Satzes mit INPUT-FORMAT=\*CHARACTER ist vom OUTPUT-FORMAT abhängig:

- OUTPUT-FORMAT=\*CHARACTER / \*HEXADECIMAL / \*BINARY  
Das Eingabefeld wird in das Ausgabe-CCS umgesetzt und ggf. aufbereitet.
- OUTPUT-FORMAT=\*TRANSLATION  
Die Bytes des Eingabe-Feldes werden nach einer Umsetztabelle umcodiert und in das Ausgabe-Feld übertragen. Eingabe- und Ausgabe-CCSN werden nicht berücksichtigt. Die Tabelle kann als Datei oder in Form von Wertepaaren vorliegen. Werden die Bytes als c-strings über INPUT-CHARACTER / OUTPUT-CHARACTER angegeben, so werden sie als Zeichen des Anweisungs-CCS interpretiert und dementsprechend in die Umsetztabelle eingetragen.
- OUTPUT-FORMAT=\*NO-TRANSLATION  
Dieses Format wird verwendet, wenn das Eingabefeld nicht in das Ausgabe-CCS umgesetzt werden soll (z.B. bei gepackten Zahlen).

### **Anweisungs-CCS ungleich Eingabe-CCS**

Ist das Anweisungs-CCS ungleich dem Eingabe-CCS, so muß folgendes beachtet werden:

Werden in der SELECT-INPUT-RECORDS-Anweisung c-strings verwendet, so müssen Anweisungs-CCS und Eingabe-CCS derselben Code-Familie angehören und alle Zeichen der c-strings im Eingabe-CCS vorhanden sein.

### **Anweisungs-CCS ungleich Ausgabe-CCS**

Ist das Anweisungs-CCS ungleich dem Ausgabe-CCS, so muß folgendes beachtet werden:

Werden in den Anweisungen SET-GROUP-ATTRIBUTES, SET-RECORD-MAPPING oder SET-PAGE-LAYOUT c-strings verwendet, so müssen Anweisungs-CCS und Ausgabe-CCS derselben Code-Familie angehören und alle Zeichen der c-strings im Ausgabe-CCS vorhanden sein.

### **Eingabe-CCS(n) ungleich Eingabe-CCS(n+1)**

Ist das CCS der aktuellen Eingabedatei ungleich dem CCS der vorherigen Eingabedatei des Konvertierungsschrittes, so muß folgendes beachtet werden:

Der Gruppenwechsel wird durch Vergleich von Feldern aufeinanderfolgender Eingabesätze, unabhängig vom CCS, festgestellt.

Bei Vergleich auf ASCENDING oder DESCENDING mit der SELECT-INPUT-RECORDS-Anweisung wird das Eingabe-CCS nicht beachtet.

## 3.12 PERCON und ACS

Wenn das Subsystem ACS (Alias Catalog Service) eingesetzt ist, sind folgende Auswirkungen zu beachten:

1. Existiert ein vom Benutzer angelegter TFT-Eintrag, so kann nicht festgestellt werden, ob eine Ersetzung des Dateinamens vorgenommen wurde oder nicht. Deshalb wird bei Verwendung des dort entnommenen Dateinamens (z.B. bei SHOW-FILE-ATTRIBUTES) ACS unterbrochen (HOLD-ALIAS-SUBSTITUTION).
2. Legt PERCON selbst einen TFT-Eintrag mit dem FILE-Makro an, so bleibt ACS in dem Zustand wie beim Start von PERCON, d.h. es können Ersetzungen von Dateinamen durchgeführt werden. Damit besteht kein Unterschied bezüglich der Dateinamen zwischen den TFT-Einträgen des Benutzers und denen von PERCON.
3. Gibt ein Benutzer in einer ASSIGN-Anweisung einen Dateinamen an, so muß er, falls bereits ein TFT-Eintrag vorhanden ist, bedenken, ob eine Ersetzung des Dateinamens durchgeführt wurde oder nicht. Der Dateiname in der Anweisung wird zum Dateinamen-Vergleich in jedem Fall ohne irgendwelche Ersetzungen herangezogen.
4. Ändert der Benutzer zwischen dem Anlegen der TFT-Einträge und dem Start von PERCON den Zustand von ACS (z.B. durch verspätetes Laden, Unterbrechen usw.), so kann es zu Unklarheiten bei den letztendlich verwendeten Dateinamen kommen.

---

## 4 PERCON-Anweisungen

Dieses Kapitel beschreibt die Anweisungen, die während des PERCON-Laufs eingegeben werden können. Nach der Übersicht der PERCON-Anweisungen und der SDF-Standardanweisungen, der Gegenüberstellung von SDF- und ISP-Anweisungen, der Syntaxbeschreibung der SDF-Benutzeroberfläche und der Formate, folgt die Beschreibung der PERCON-Anweisungen in alphabetischer Reihenfolge.

### 4.1 Übersicht der PERCON-Anweisungen

Die Eingabe von Anweisungen kann im ungeführten oder geführten Dialog, aus Prozedurdateien oder im Stapelbetrieb erfolgen (Einzelheiten zu den Eingabemöglichkeiten sind im Handbuch „Einführung in die Dialogschnittstelle SDF“ [8] beschrieben).

Im geführten Dialog werden dem Benutzer nur die Anweisungen angeboten, die im Moment zugelassen sind. So wird z.B. START-TAPE-PROCESSING nur nach ASSIGN-INPUT-TAPE angeboten (siehe Reihenfolge der Anweisungen Seite 27).

Anweisung	Bedeutung
ASSIGN-INPUT-FILE	Zuweisen einer Eingabedatei
ASSIGN-INPUT-TAPE	Zuweisen eines Eingabebandes
ASSIGN-OUTPUT-FILE	Zuweisen einer Ausgabedatei
ASSIGN-OUTPUT-TAPE	Zuweisen eines Ausgabebandes
CHANGE-INPUT-TAPEPOSITION	Positionieren des Eingabebandes
END	Starten eines Konvertierungsschrittes, Beenden des PERCON-Laufs
EXECUTE-SYSTEM-COMMAND	Absetzen eines Systemkommandos
MODIFY-PERCON-OPTIONS	Steuern der Meldungsausgabe
RESET-INPUT	Rücksetzen von Anweisungen
SELECT-INPUT-RECORDS	Festlegen von Auswahlbedingungen
SET-GROUP-ATTRIBUTES	Festlegen von Gruppenwechselbedingungen
SET-PAGE-LAYOUT	Beschreiben des Ausgabeformates
SET-RECORD-MAPPING	Festlegen des Ausgabesatzaufbaus
START-CONVERSION	Starten eines Konvertierungsschrittes
START-TAPE-PROCESSING	Steuern der Bandausgabe

Folgende Standardanweisungen von SDF können zusätzlich gegeben werden:

Anweisung	Funktion
EXECUTE-SYSTEM-CMD	Ausführen von Systemkommandos
MODIFY-SDF-OPTIONS	Ändern der SDF-Einstellungen
REMARK	Kommentar
RESTORE-SDF-INPUT	Wiederanzeigen der letzten Eingabe
SHOW-SDF-OPTIONS	Anzeigen der SDF-Einstellungen
STEP	Definieren eines Wiederaufsetzpunktes
WRITE-TEXT	Text nach SYSOUT ausgeben

Die SDF-Standardanweisungen (bis auf END) werden nicht in diesem Handbuch beschrieben. Eine Beschreibung findet sich im Handbuch „Einführung in die Dialogschnittstelle SDF“ [8].



## 4.2 Gegenüberstellung von SDF- und ISP-Anweisungen

In der folgenden Tabelle werden den Anweisungen im SDF-Format die entsprechenden Anweisungen im ISP-Format und ihre wichtigsten Unterschiede gegenübergestellt.

SDF-Anweisungen	bisherige Anweisungen	Funktionsunterschiede
ASSIGN-INPUT-FILE	FILIN	Dateimerkmale, die von den Standardwerten abweichen sollen, <ul style="list-style-type: none"> <li>– müssen im SDF-Format über ein /SET-FILE-LINK-Kommando zugewiesen werden</li> <li>– können im ISP-Format über ein /FILE-Kommando oder eine FILIN-Anweisung zugewiesen werden</li> </ul>
ASSIGN-INPUT-TAPE	VOLIN	---
ASSIGN-OUTPUT-FILE	FILOUT	Dateimerkmale, die von den Standardwerten abweichen sollen, <ul style="list-style-type: none"> <li>– müssen im SDF-Format über ein /SET-FILE-LINK-Kommando zugewiesen werden</li> <li>– können im ISP-Format über ein /FILE-Kommando oder eine FILOUT-Anweisung zugewiesen werden</li> </ul>
ASSIGN-OUTPUT-TAPE	VOLOUT	---
CHANGE-INPUT-TAPEPOSITION	POSIT	---
END	END/HALT	---
EXECUTE-SYSTEM-COMMAND	---	keine entsprechende ISP-Anweisung vorhanden
MODIFY-PERCON-OPTIONS	PARAM	---
RESET-INPUT	RESET	Funktion ist im SDF-Format unabhängig vom Eingabemedium
SELECT-INPUT-RECORDS	SELECT	---
SET-GROUP-ATTRIBUTES	GROUP	---
SET-PAGE-LAYOUT	FORMAT	---
SET-RECORD-MAPPING	RECORD	---
START-CONVERSION	START	---
START-TAPE-PROCESSING	EDIT	---

## 4.3 Syntaxbeschreibung

Diese Syntaxbeschreibung basiert auf der SDF-Version 4.0A. Die Syntax der SDF-Kommando-/Anweisungssprache wird im folgenden in drei Tabellen erklärt.

### *Zu Tabelle 1: Metasyntax*

In den Kommando-/Anweisungsformaten werden bestimmte Zeichen und Darstellungsformen verwendet, deren Bedeutung in Tabelle 1 erläutert wird.

### *Zu Tabelle 2: Datentypen*

Variable Operandenwerte werden in SDF durch Datentypen dargestellt. Jeder Datentyp repräsentiert einen bestimmten Wertevorrat. Die Anzahl der Datentypen ist beschränkt auf die in Tabelle 2 beschriebenen Datentypen.

Die Beschreibung der Datentypen gilt für alle Kommandos und Anweisungen. Deshalb werden bei den entsprechenden Operandenbeschreibungen nur noch Abweichungen von Tabelle 2 erläutert.

### *Zu Tabelle 3: Zusätze zu Datentypen*

Zusätze zu Datentypen kennzeichnen weitere Eingabevorschriften für Datentypen. Die Zusätze schränken den Wertevorrat ein oder erweitern ihn. Im Handbuch werden folgende Zusätze in gekürzter Form dargestellt:

cat-id	cat
completion	compl
construction	constr
correction-state	corr
generation	gen
lower-case	low
manual-release	man
odd-possible	odd
path-completion	path-compl
separators	sep
underscore	under
user-id	user
version	vers
wildcards	wild

Für den Datentyp integer enthält Tabelle 3 außerdem kursiv gesetzte Einheiten, die nicht Bestandteil der Syntax sind. Sie dienen lediglich als Lesehilfe.

Die Beschreibung der Zusätze zu den Datentypen gilt für alle Kommandos und Anweisungen. Deshalb werden bei den entsprechenden Operandenbeschreibungen nur noch Abweichungen von Tabelle 3 erläutert.

## Metasyntax

Kennzeichnung	Bedeutung	Beispiele
GROSSBUCHSTABEN	Großbuchstaben bezeichnen Schlüsselwörter. Schlüsselwörter für konstante Operandenwerte beginnen mit *	<b>HELP-SDF</b>  <b>SCREEN-STEPS = <u>*NO</u></b>
<b>GROSSBUCHSTABEN in Halbfett</b>	Großbuchstaben in Halbfett kennzeichnen garantierte bzw. vorgeschlagene Abkürzungen der Schlüsselwörter.	<b>GUIDANCE-MODE = *YES</b>
=	Das Gleichheitszeichen verbindet einen Operandennamen mit den dazugehörigen Operandenwerten.	<b>GUIDANCE-MODE = <u>*NO</u></b>
< >	Spitze Klammern kennzeichnen Variablen, deren Wertevorrat durch Datentypen und ihre Zusätze beschrieben wird (siehe Tabellen 2 und 3).	<b>SYNTAX-FILE = &lt;full-filename 1..54&gt;</b>
<u>Unterstreichug</u>	Der Unterstrich kennzeichnet den Default-Wert eines Operanden.	<b>GUIDANCE-MODE = <u>*NO</u></b>
/	Der Schrägstrich trennt alternative Operandenwerte.	<b>NEXT-FIELD = <u>*NO</u> / *YES</b>
(...)	Runde Klammern kennzeichnen Operandenwerte, die eine Struktur einleiten.	<b>,UNGUIDED-DIALOG = <u>*YES</u> (...) / *NO</b>
[ ]	Eckige Klammern kennzeichnen struktureinleitende Operandenwerte, deren Angabe optional ist. Die nachfolgende Struktur kann ohne den einleitenden Operandenwert angegeben werden.	<b>SELECT = [<u>*BY-ATTRIBUTES</u>](...)</b>
Einrückung	Die Einrückung kennzeichnet die Abhängigkeit zu dem jeweils übergeordneten Operanden.	<b>,GUIDED-DIALOG = <u>*YES</u> (...)</b> <b>*YES(...)</b> <b>SCREEN-STEPS = <u>*NO</u> / *YES</b>

Tabelle 1: Metasyntax (Abschnitt 1 von 2)

--	--	--

## Datentypen

Datentyp	Zeichenvorrat	Besonderheiten
alphanum-name	A...Z 0...9 \$, #, @	
cat-id	A...Z 0...9	maximal 4 Zeichen; darf nicht mit der Zeichenfolge PUB beginnen
command-rest	beliebig	
composed-name	A...Z 0...9 \$, #, @ Bindestrich Punkt Katalogkennung	alphanumerische Zeichenfolge, die in mehrere durch Punkt oder Bindestrich getrennte Teilzeichenfolgen gegliedert sein kann. Ist auch die Angabe eines Dateinamens möglich, so kann die Zeichenfolge mit einer Katalogkennung im Format :cat: beginnen (siehe Datentyp full-filename).
c-string	EBCDIC-Zeichen	ist in Hochkommata einzuschließen; der Buchstabe C kann vorangestellt werden; Hochkommata innerhalb des c-string müssen verdoppelt werden
date	0...9 Strukturkennzeichen: Bindestrich	Eingabeformat: yyyy-mm-tt  yyyy: Jahr; wahlweise 2- oder 4stellig mm: Monat tt: Tag
device	A...Z 0...9 Bindestrich	Zeichenfolge, die maximal 8 Zeichen lang ist und einem im System verfügbaren Gerät entspricht. In der Dialogführung zeigt SDF die zulässigen Operandenwerte an. Hinweise zu möglichen Geräten sind der jeweiligen Operandenbeschreibung zu entnehmen.
fixed	+, - 0...9 Punkt	Eingabeformat: [zeichen][ziffern].[ziffern]  [zeichen]: + oder - [ziffern]: 0...9  muß mindestens eine Ziffer, darf aber außer dem Vorzeichen maximal 10 Zeichen (0...9, Punkt) enthalten

Tabelle 2: Datentypen (Abschnitt 1 von 6)

Datentyp	Zeichenvorrat	Besonderheiten
full-filename	A...Z 0...9 \$, #, @ Bindestrich Punkt	<p>Eingabeformat:</p> $[:cat:][\$user.] \left\{ \begin{array}{l} \text{datei} \\ \text{datei(nr)} \\ \text{gruppe} \\ \text{gruppe} \left\{ \begin{array}{l} (*abs) \\ (+rel) \\ (-rel) \end{array} \right\} \end{array} \right\}$ <p>:cat:</p> <p>wahlfreie Angabe der Katalogkennung; Zeichenvorrat auf A...Z und 0...9 eingeschränkt; max. 4 Zeichen; ist in Doppelpunkte einzuschließen; voreingestellt ist die Katalogkennung, die der Benutzerkennung laut Eintrag im Benutzerkatalog zugeordnet ist.</p> <p>\$user.</p> <p>wahlfreie Angabe der Benutzerkennung; Zeichenvorrat ist A...Z, 0...9, \$, #, @; max. 8 Zeichen; darf nicht mit einer Ziffer beginnen; \$ und Punkt müssen angegeben werden; voreingestellt ist die eigene Benutzerkennung.</p> <p>\$. (Sonderfall) System-Standardkennung</p> <p>datei</p> <p>Datei- oder Jobvariablenname; kann durch Punkt in mehrere Teilnamen gegliedert sein: name<sub>1</sub>[.name<sub>2</sub>[...]] name<sub>i</sub> enthält keinen Punkt und darf nicht mit Bindestrich beginnen oder enden; datei ist max. 41 Zeichen lang, darf nicht mit \$ beginnen und muß mindestens ein Zeichen aus A...Z enthalten.</p>

Tabelle 2: Datentypen (Abschnitt 2 von 6)

Datentyp	Zeichenvorrat	Besonderheiten
full-filename (Forts.)		<p>#datei (Sonderfall) @datei (Sonderfall) # oder @ als erstes Zeichen kennzeichnet je nach Systemgenerierung temporäre Dateien und Jobvariablen.</p> <p>datei(nr) Banddateiname nr: Versionsnummer; Zeichenvorrat ist A...Z, 0...9, \$, #, @. Klammern müssen angegeben werden.</p> <p>gruppe Name einer Dateigenerationsgruppe (Zeichenvorrat siehe unter "datei")</p> <p>gruppe <math>\left\{ \begin{array}{l} (*abs) \\ (+rel) \\ (-rel) \end{array} \right\}</math></p> <p>(*abs) absolute Generationsnummer (1..9999); * und Klammern müssen angegeben werden.</p> <p>(+rel) (-rel) relative Generationsnummer (0..99); Vorzeichen und Klammern müssen angegeben werden.</p>
integer	0...9, +, -	+ bzw. - kann nur erstes Zeichen sein (Vorzeichen).
name	A...Z 0...9 \$, #, @	darf nicht mit einer Ziffer beginnen.

Tabelle 2: Datentypen (Abschnitt 3 von 6)

Datentyp	Zeichenvorrat	Besonderheiten
partial-filename	A...Z 0...9 \$, #, @ Bindestrich Punkt	Eingabeformat: [:cat:][\$user.][teilname.]  :cat: siehe full-filename \$user. siehe full-filename  teilname wahlfreie Angabe des gemeinsamen ersten Namensteils von Dateien und Dateigenera- tionsgruppen in der Form: name <sub>1</sub> . [name <sub>2</sub> . [...]] name <sub>i</sub> siehe full-filename. Das letzte Zeichen von teilname muß ein Punkt sein. Es muß mindestens einer der Teile :cat:, \$user. oder teilname angegeben werden.
posix-filename	A...Z 0...9 Sonderzeichen	Zeichenfolge, die maximal 255 Zeichen lang ist. Besteht entweder aus einem oder zwei Punk- ten, oder aus alphanumerischen Zeichen und Sonderzeichen; Sonderzeichen sind mit dem Zeichen \ zu entwerten. Nicht erlaubt ist das Zei- chen /. Muß in Hochkommata eingeschlossen werden, wenn alternative Datentypen zulässig sind, Se- paratoren verwendet werden oder das erste Zei- chen ? bzw. ! ist. Zwischen Groß- und Kleinschreibung wird un- terschieden.
posix-pathname	A...Z 0...9 Sonderzeichen Strukturkennzeichen: Schrägstrich	Eingabeformat: [/]part <sub>1</sub> /.../part <sub>n</sub> wobei part <sub>i</sub> ein posix-filename ist; maximal 1024 Zeichen; muß in Hochkommata eingeschlossen werden, wenn alternative Datentypen zulässig sind, Se- paratoren verwendet werden oder das erste Zei- chen ? bzw. ! ist.

Tabelle 2: Datentypen (Abschnitt 4 von 6)



Datentyp	Zeichenvorrat	Besonderheiten
product-version	A...Z 0...9 Punkt Hochkomma	<p>Eingabeformat: <code>[[C']][V][n]n.nann[']</code></p> <div style="text-align: right; margin-right: 50px;"> <div style="display: inline-block; vertical-align: middle; text-align: center;"> <div style="border-left: 1px solid black; height: 40px; width: 10px; margin: 0 auto;"></div> <div style="font-size: 8px; margin-top: 5px;">Korrekturstand</div> </div> <div style="display: inline-block; vertical-align: middle; text-align: center;"> <div style="border-left: 1px solid black; height: 40px; width: 10px; margin: 0 auto;"></div> <div style="font-size: 8px; margin-top: 5px;">Freigabestand</div> </div> </div> <p>wobei n eine Ziffer und a ein Buchstabe ist. Freigabe- und Korrekturstand müssen angegeben werden, wenn product-version keinen Zusatz enthält (siehe Tabelle 3, Zusatz without-corr bzw. without-man). product-version kann in Hochkommata eingeschlossen werden, wobei der Buchstabe C vorangestellt werden kann. Die Versionsangabe kann mit dem Buchstaben V beginnen.</p>
structured-name	A...Z 0...9 \$, #, @ Bindestrich	<p>alphanumerische Zeichenfolge, die in mehrere durch Bindestrich getrennte Teilzeichenfolgen gegliedert sein kann; erstes Zeichen: A...Z oder \$, #, @</p>
text	beliebig	Das Eingabeformat ist den jeweiligen Operandenbeschreibungen zu entnehmen.
time	0...9 Strukturkennzeichen: Doppelpunkt	<p>Angabe einer Tageszeit</p> <p>Eingabeformat: <math>\left\{ \begin{array}{l} \text{hh:mm:ss} \\ \text{hh:mm} \\ \text{hh} \end{array} \right\}</math></p> <div style="display: flex; align-items: center;"> <div style="margin-right: 10px;"> <p>hh:    Stunden</p> <p>mm:    Minuten</p> <p>ss:    Sekunden</p> </div> <div style="font-size: 2em; margin-right: 10px;">}</div> <div> <p>führende Nullen können weggelassen werden</p> </div> </div>
vsn	<p>a) A...Z 0...9</p> <p>b) A...Z 0...9 \$, #, @</p>	<p>a) Eingabeformat: pvsid.folgenummer max. 6 Zeichen;  pvsid:            2-4 Zeichen; Eingabe von PUB nicht erlaubt folgenummer: 1-3 Zeichen</p> <p>b) max. 6 Zeichen; PUB darf vorangestellt werden, dann dürfen jedoch nicht \$, #, @ folgen.</p>

Tabelle 2: Datentypen (Abschnitt 5 von 6)

Datentyp	Zeichenvorrat	Besonderheiten
x-string	Sedezimal: 00...FF	ist in Hochkommata einzuschließen; der Buchstabe X muß vorangestellt werden; die Anzahl der Zeichen darf ungerade sein.
x-text	Sedezimal: 00...FF	ist nicht in Hochkommata einzuschließen; der Buchstabe X darf nicht vorangestellt werden; die Anzahl der Zeichen darf ungerade sein.

Tabelle 2: Datentypen (Abschnitt 6 von 6)

## Zusätze zu Datentypen

Zusatz	Bedeutung								
x..y <i>unit</i>	<p>a) beim Datentyp integer: Intervallangabe</p> <p>x      Mindestwert, der für integer erlaubt ist. x ist eine ganze Zahl, die mit einem Vorzeichen versehen werden darf.</p> <p>y      Maximalwert, der für integer erlaubt ist. y ist eine ganze Zahl, die mit einem Vorzeichen versehen werden darf.</p> <p><i>unit</i>    nur bei Datentyp integer: zusätzliche Einheiten. Folgende Angaben werden verwendet:</p> <table> <tr> <td><i>days</i></td><td><i>byte</i></td></tr> <tr> <td><i>hours</i></td><td><i>2Kbyte</i></td></tr> <tr> <td><i>minutes</i></td><td><i>4Kbyte</i></td></tr> <tr> <td><i>seconds</i></td><td><i>Mbyte</i></td></tr> </table> <p>b) bei den übrigen Datentypen: Längenangabe</p> <p>x      Mindestlänge für den Operandenwert; x ist eine ganze Zahl.</p> <p>y      Maximallänge für den Operandenwert; y ist eine ganze Zahl.</p> <p>x=y    Der Operandenwert muß genau die Länge x haben.</p>	<i>days</i>	<i>byte</i>	<i>hours</i>	<i>2Kbyte</i>	<i>minutes</i>	<i>4Kbyte</i>	<i>seconds</i>	<i>Mbyte</i>
<i>days</i>	<i>byte</i>								
<i>hours</i>	<i>2Kbyte</i>								
<i>minutes</i>	<i>4Kbyte</i>								
<i>seconds</i>	<i>Mbyte</i>								
with	Erweitert die Angabemöglichkeiten für einen Datentyp.								
-compl	Bei Angaben zu dem Datentyp date ergänzt SDF zweistellige Jahresangaben der Form jj-mm-tt zu:								
	<table> <tr> <td>20jj-mm-tt</td><td>falls jj &lt; 60</td></tr> <tr> <td>19jj-mm-tt</td><td>falls jj ≥ 60</td></tr> </table>	20jj-mm-tt	falls jj < 60	19jj-mm-tt	falls jj ≥ 60				
20jj-mm-tt	falls jj < 60								
19jj-mm-tt	falls jj ≥ 60								
-low	Groß- und Kleinschreibung wird unterschieden.								
-under	Erlaubt Unterstriche '_' beim Datentyp name.								

Tabelle 3: Zusätze zu Datentypen (Abschnitt 1 von 6)

Zusatz	Bedeutung
with (Forts.)	
-wild(n)	Teile eines Namens dürfen durch die folgenden Platzhalter ersetzt werden. n bezeichnet die maximale Eingabelänge bei Verwendung von Platzhaltern. Mit Einführung der Datentypen posix-filename und posix-pathname werden neben den bisher im BS2000 üblichen Platzhaltern auch Platzhalter aus der UNIX-Welt (nachfolgend POSIX-Platzhalter genannt) akzeptiert. Innerhalb einer Musterzeichenfolge sollten entweder nur BS2000- oder nur POSIX-Platzhalter verwendet werden. Bei den Datentypen posix-filename und posix-pathname sind nur POSIX-Platzhalter erlaubt. Ist eine Musterzeichenfolge mehrdeutig auf einen String abbildbar, gilt der erste Treffer.
BS2000-Platzhalter	Bedeutung
*	Ersetzt eine beliebige, auch leere Zeichenfolge. Ein * an erster Stelle muß verdoppelt werden, sofern dem * weitere Zeichen folgen und die eingegebene Zeichenfolge nicht mindestens einen weiteren Platzhalter enthält.
Punkt am Ende	Teilqualifizierte Angabe eines Namens. Entspricht implizit der Zeichenfolge ".*", d.h. nach dem Punkt folgt mindestens ein beliebiges Zeichen.
/	Ersetzt genau ein beliebiges Zeichen.
<s <sub>x</sub> :s <sub>y</sub> >	Ersetzt eine Zeichenfolge, für die gilt: <ul style="list-style-type: none"> <li>– sie ist mindestens so lang wie die kürzeste Zeichenfolge (s<sub>x</sub> oder s<sub>y</sub>)</li> <li>– sie ist höchstens so lang wie die längste Zeichenfolge (s<sub>x</sub> oder s<sub>y</sub>)</li> <li>– sie liegt in der alphabetischen Sortierung zwischen s<sub>x</sub> und s<sub>y</sub>; Zahlen werden hinter Buchstaben sortiert (A...Z 0...9)</li> <li>– s<sub>x</sub> darf auch die leere Zeichenfolge sein, die in der alphabetischen Sortierung an erster Stelle steht</li> <li>– s<sub>y</sub> darf auch die leere Zeichenfolge sein, die an dieser Stelle für die Zeichenfolge mit der höchst möglichen Codierung steht (enthält nur die Zeichen X' FF' )</li> </ul>
<s <sub>1</sub> ,...>	Ersetzt alle Zeichenfolgen, auf die eine der mit s angegebenen Zeichenkombinationen zutrifft. s kann auch die leere Zeichenfolge sein. Jede Zeichenfolge s kann auch eine Bereichsangabe "s <sub>x</sub> :s <sub>y</sub> " sein (siehe oben).

Tabelle 3: Zusätze zu Datentypen (Abschnitt 2 von 6)

Zusatz	Bedeutung	
with-wild(n) (Forts.)	-s	Ersetzt alle Zeichenfolgen, die der angegebenen Zeichenfolge s nicht entsprechen. Das Minuszeichen darf nur am Beginn der Zeichenfolge stehen. Innerhalb der Datentypen full-filename bzw. partial-filename kann die negierte Zeichenfolge -s genau einmal verwendet werden, d.h., -s kann einen der drei Namensteile cat, user oder datei ersetzen.
	Platzhalter	sind in Generations- und Versionsangaben von Dateinamen nicht erlaubt. In Benutzerkennungen ist die Angabe von Platzhaltern der Systemverwaltung vorbehalten. Platzhalter können nicht die Begrenzer der Namensteile cat (Doppelpunkte) und user (\$ und Punkt) ersetzen.
	POSIX-Platzhalter	Bedeutung
	*	Ersetzt eine beliebige, auch leere Zeichenfolge. Ein * an erster Stelle muß verdoppelt werden, sofern dem * weitere Zeichen folgen und die eingegebene Zeichenfolge nicht mindestens einen weiteren Platzhalter enthält.
	?	Ersetzt genau ein beliebiges Zeichen. Ist als erstes Zeichen außerhalb von Hochkommata nicht zulässig.
	[c <sub>x</sub> -c <sub>y</sub> ]	Ersetzt genau ein Zeichen aus dem Bereich c <sub>x</sub> und c <sub>y</sub> einschließlich der Bereichsgrenzen. c <sub>x</sub> und c <sub>y</sub> müssen einfache Zeichen sein.
	[s]	Ersetzt genau ein Zeichen aus der Zeichenfolge s. Die Ausdrücke [c <sub>x</sub> -c <sub>y</sub> ] und [s] können kombiniert werden zu [s <sub>1</sub> c <sub>1</sub> -c <sub>2</sub> s <sub>2</sub> ]
	[!c <sub>x</sub> -c <sub>y</sub> ]	Ersetzt genau ein Zeichen, das nicht in dem Bereich c <sub>x</sub> und c <sub>y</sub> einschließlich der Bereichsgrenzen enthalten ist. c <sub>x</sub> und c <sub>y</sub> müssen einfache Zeichen sein. Die Ausdrücke [!c <sub>x</sub> -c <sub>y</sub> ] und [!s] können kombiniert werden zu [!s <sub>1</sub> c <sub>1</sub> -c <sub>2</sub> s <sub>2</sub> ]
	[!s]	Ersetzt genau ein Zeichen, das nicht in der Zeichenfolge s enthalten ist. Die Ausdrücke [!s] und [!cx-cy] können kombiniert werden zu [!s <sub>1</sub> c <sub>1</sub> -c <sub>2</sub> s <sub>2</sub> ]

Tabelle 3: Zusätze zu Datentypen (Abschnitt 3 von 6)

Zusatz	Bedeutung										
with (Forts.) -constr	<p>Angabe einer Konstruktionszeichenfolge, die angibt, wie aus einer zuvor angegebenen Auswahlzeichenfolge mit Musterzeichen (siehe with-wild) neue Namen zu bilden sind.</p> <p>Die Konstruktionszeichenfolge kann aus konstanten Zeichenfolgen und Musterzeichen bestehen. Ein Musterzeichen wird durch diejenige Zeichenfolge ersetzt, die durch das entsprechende Musterzeichen in der Auswahlzeichenfolge ausgewählt wird.</p> <p>Folgende Platzhalter können zur Konstruktionsangabe verwendet werden:</p> <table> <tr> <th>Platzhalter</th><th>Bedeutung</th></tr> <tr> <td>*</td><td>Entspricht der Zeichenfolge, die durch den Platzhalter * in der Auswahlzeichenfolge ausgewählt wird.</td></tr> <tr> <td>Punkt am Ende</td><td>Entspricht der teilqualifizierten Angabe eines Namens in der Auswahlzeichenfolge. Entspricht der Zeichenfolge, die durch den Punkt am Ende der Auswahlzeichenfolge ausgewählt wird.</td></tr> <tr> <td>/ oder ?</td><td>Entspricht dem Zeichen, das durch den Platzhalter / oder ? in der Auswahlzeichenfolge ausgewählt wird.</td></tr> <tr> <td>&lt;n&gt;</td><td>Entspricht der Zeichenfolge, die durch den n-ten Platzhalter in der Auswahlzeichenfolge ausgewählt wird; n = &lt;integer&gt;</td></tr> </table> <p>Zuordnung der Platzhalter zu entsprechenden Platzhaltern in der Auswahlzeichenfolge:</p> <p>In der Auswahlzeichenfolge werden alle Platzhalter von links nach rechts aufsteigend numeriert (globaler Index).</p> <p>Gleiche Platzhalter in der Auswahlzeichenfolge werden zusätzlich von links nach rechts aufsteigend numeriert (platzhalter-spezifischer Index).</p> <p>In der Konstruktionsangabe können Platzhalter auf zwei, sich gegenseitig ausschließende Arten angegeben werden:</p> <ol style="list-style-type: none"> <li>1. Platzhalter werden über den globalen Index angegeben: &lt;n&gt;</li> <li>2. Angabe desselben Platzhalters, wobei die Ersetzung gemäß dem platzhalter-spezifischen Index entsprechend erfolgt: z.B. der zweite "/" entspricht der Zeichenfolge, die durch den zweiten "/" in der Auswahlzeichenfolge ausgewählt wird.</li> </ol>	Platzhalter	Bedeutung	*	Entspricht der Zeichenfolge, die durch den Platzhalter * in der Auswahlzeichenfolge ausgewählt wird.	Punkt am Ende	Entspricht der teilqualifizierten Angabe eines Namens in der Auswahlzeichenfolge. Entspricht der Zeichenfolge, die durch den Punkt am Ende der Auswahlzeichenfolge ausgewählt wird.	/ oder ?	Entspricht dem Zeichen, das durch den Platzhalter / oder ? in der Auswahlzeichenfolge ausgewählt wird.	<n>	Entspricht der Zeichenfolge, die durch den n-ten Platzhalter in der Auswahlzeichenfolge ausgewählt wird; n = <integer>
Platzhalter	Bedeutung										
*	Entspricht der Zeichenfolge, die durch den Platzhalter * in der Auswahlzeichenfolge ausgewählt wird.										
Punkt am Ende	Entspricht der teilqualifizierten Angabe eines Namens in der Auswahlzeichenfolge. Entspricht der Zeichenfolge, die durch den Punkt am Ende der Auswahlzeichenfolge ausgewählt wird.										
/ oder ?	Entspricht dem Zeichen, das durch den Platzhalter / oder ? in der Auswahlzeichenfolge ausgewählt wird.										
<n>	Entspricht der Zeichenfolge, die durch den n-ten Platzhalter in der Auswahlzeichenfolge ausgewählt wird; n = <integer>										

Tabelle 3: Zusätze zu Datentypen (Abschnitt 4 von 6)

Zusatz	Bedeutung
with-constr (Forts.)	<p>Bei Konstruktionsangaben sind folgende Regeln zu beachten:</p> <ul style="list-style-type: none"> <li>– Die Konstruktionsangabe muß mindestens einen Platzhalter der Auswahlzeichenfolge enthalten.</li> <li>– Ist die Anzahl gleicher Platzhalter größer als in der Auswahlzeichenfolge, muß die Index-Schreibweise gewählt werden.</li> <li>– Soll die Zeichenkette, die der Platzhalter &lt;...&gt; bzw. [...] auswählt, in der Konstruktionsangabe verwendet werden, muß die Index-Schreibweise gewählt werden.</li> <li>– Die Index-Schreibweise muß gewählt werden, wenn die Zeichenkette, die der Platzhalter "*" bezeichnet, vervielfacht werden soll: Statt "***" muß z.B. "&lt;n&gt;&lt;n&gt;" angegeben werden.</li> <li>– Der Platzhalter * kann auch die leere Zeichenkette sein. Insbesondere ist zu beachten, daß bei mehreren Sternen in Folge (auch mit weiteren Platzhaltern) nur der letzte Stern eine nicht leere Zeichenfolge sein kann: z.B. bei "*****" oder "*//*".</li> <li>– Aus der Konstruktionsangabe sollten gültige Namen entstehen. Darauf ist sowohl bei der Auswahlangabe als auch bei der Konstruktionsangabe zu achten.</li> <li>– Abhängig von der Konstruktionsangabe können aus unterschiedlichen Namen, die in der Auswahlangabe ausgewählt werden, identische Namen gebildet werden: z.B. "A/*" wählt die Namen "A1" und "A2" aus; die Konstruktionsangabe "B*" erzeugt für beide Namen denselben neuen Namen "B". Um dies zu vermeiden, sollten in der Konstruktionsangabe alle Platzhalter der Auswahlangabe mindestens einmal verwendet werden.</li> <li>– Wird die Auswahlzeichenfolge mit einem Punkt abgeschlossen, so muß auch die Konstruktionsangabe mit einem Punkt enden (und umgekehrt).</li> </ul>

Tabelle 3: Zusätze zu Datentypen (Abschnitt 5 von 6)

Zusatz	Bedeutung																				
with-constr (Forts.)	Beispiele:																				
	<table><tr><th>Auswahlmuster</th><th>Auswahl</th><th>Konstruktionsmuster</th><th>neuer Name</th></tr><tr><td>A/*</td><td>AB1 AB2 A.B.C</td><td>D&lt;3&gt;&lt;2&gt;</td><td>D1 D2 D.CB</td></tr><tr><td>C.&lt;A:C&gt;/&lt;D,F&gt;</td><td>C.AAD C.ABD C.BAF C.BBF</td><td>G.&lt;1&gt;.&lt;3&gt;.XY&lt;2&gt;</td><td>G.A.D.XYA G.A.D.XYB G.B.F.XYA G.B.F.XYB</td></tr><tr><td>C.&lt;A:C&gt;/&lt;D,F&gt;</td><td>C.AAD C.ABD C.BAF C.BBF</td><td>G.&lt;1&gt;.&lt;2&gt;.XY&lt;2&gt;</td><td>G.A.A.XYA G.A.B.XYB G.B.A.XYA G.B.B.XYB</td></tr><tr><td>A//B</td><td>ACDB ACEB AC.B A.CB</td><td>G/XY/</td><td>GCXYD GCXYE GCXY. G.XYC</td></tr></table>	Auswahlmuster	Auswahl	Konstruktionsmuster	neuer Name	A/*	AB1 AB2 A.B.C	D<3><2>	D1 D2 D.CB	C.<A:C>/<D,F>	C.AAD C.ABD C.BAF C.BBF	G.<1>.<3>.XY<2>	G.A.D.XYA G.A.D.XYB G.B.F.XYA G.B.F.XYB	C.<A:C>/<D,F>	C.AAD C.ABD C.BAF C.BBF	G.<1>.<2>.XY<2>	G.A.A.XYA G.A.B.XYB G.B.A.XYA G.B.B.XYB	A//B	ACDB ACEB AC.B A.CB	G/XY/	GCXYD GCXYE GCXY. G.XYC
	Auswahlmuster	Auswahl	Konstruktionsmuster	neuer Name																	
	A/*	AB1 AB2 A.B.C	D<3><2>	D1 D2 D.CB																	
	C.<A:C>/<D,F>	C.AAD C.ABD C.BAF C.BBF	G.<1>.<3>.XY<2>	G.A.D.XYA G.A.D.XYB G.B.F.XYA G.B.F.XYB																	
	C.<A:C>/<D,F>	C.AAD C.ABD C.BAF C.BBF	G.<1>.<2>.XY<2>	G.A.A.XYA G.A.B.XYB G.B.A.XYA G.B.B.XYB																	
A//B	ACDB ACEB AC.B A.CB	G/XY/	GCXYD GCXYE GCXY. G.XYC																		
1) Punkt am Ende des Namens kann Namenskonvention widersprechen (z.B bei vollqualifizierten Dateinamen)																					
without	Schränkt die Angabemöglichkeiten für einen Datentyp ein.																				
	-cat	Die Angabe einer Katalogkennung ist nicht erlaubt.																			
	-corr	Eingabeformat: <code>[[C]' ][V][n]n.na[' ]</code> Angaben zum Datentyp product-version dürfen den Korrekturstand nicht enthalten.																			
	-gen	Die Angabe einer Dateigeneration oder Dateigenerationsgruppe ist nicht erlaubt.																			
	-man	Eingabeformat: <code>[[C]' ][V][n]n.n[' ]</code> Angaben zum Datentyp product-version dürfen weder Freigabe- noch Korrekturstand enthalten.																			
	-odd	Der Datentyp x-text erlaubt nur eine gerade Anzahl von Zeichen.																			
	-sep	Beim Datentyp text ist die Angabe der folgenden Trennzeichen nicht erlaubt: ; = ( ) < > ? (also Strichpunkt, Gleichheitszeichen, runde Klammer auf und zu, Größerzeichen, Kleinerzeichen und Leerzeichen)																			
	-user	Die Angabe einer Benutzerkennung ist nicht erlaubt.																			
-vers	Die Angabe der Version (siehe "datei(nr)") ist bei Banddateien nicht erlaubt.																				

Tabelle 3: Zusätze zu Datentypen (Abschnitt 6 von 6)

## **Folgezeilen**

Anweisungen können sich über mehrere Eingabezeilen erstrecken. Für die Trennung gelten die Konventionen der BS2000-Kommandosprache. Als Trennzeichen wird ein Bindestrich (-) verwendet. Anweisungszeilen können maximal 32763 Zeichen lang sein.

## **Abkürzungsmöglichkeiten**

Die garantierten Abkürzungsmöglichkeiten aller Anweisungen, Operanden und Operandenwerte sind in den Syntaxbeschreibungen der Anweisungen (ab Seite 81) durch halbfette Schreibweise gekennzeichnet. Es können aber auch kürzere Angaben (Abkürzungen bis zur Eindeutigkeit innerhalb einer Struktur) gemacht werden.

Um Doppeldeutigkeiten durch Funktionserweiterungen in kommenden Versionen vorzubeugen und die Lesbarkeit anderen Benutzern zu ermöglichen, sollten in Prozeduren Abkürzungen vermieden werden.

## **Verhalten bei formalen Eingabefehlern**

Unter formalen Eingabefehlern versteht man sowohl Verstöße gegen die Anweisungssyntax als auch widersprüchliche Operanden einer Anweisung.

Widersprüche zwischen verschiedenen Anweisungen werden wie logische Fehler behandelt (siehe Seite 26).

Tritt ein formaler Fehler auf,

- bietet PERCON im Dialog die Möglichkeit, Hilfsmenüs zu den Anweisungen anzufordern und bei fehlerhaften Anweisungen einen Korrekturdialog über SDF zu führen (siehe Handbuch „Einführung in die Dialogschnittstelle SDF“ [8]).
- wird im Stapelbetrieb auf die nächste STEP- oder END-Anweisung verzweigt.

## **Kommentare**

Zeichenfolgen, die in Anführungszeichen eingeschlossen sind, werden als Kommentare interpretiert und ignoriert.



## 4.4 Literale

PERCON kann als Vergleichskriterien oder zum Einfügen in Ausgabesätze konstante Werte verarbeiten. Diese werden als Literale bezeichnet.

### Aufbau eines Literals bei

SET-GROUP-ATTRIBUTES SET-PAGE-LAYOUT SET-RECORD-MAPPING	SELECT-INPUT-RECORDS	Bedeutung
[C] 'c-string'		Folge von bis zu 256 Groß-, Kleinbuchstaben, Ziffern oder Sonderzeichen. Soll innerhalb dieser Folge ein Hochkomma stehen, ist dieses doppelt anzugeben. Speicherbedarf: 1 byte pro Zeichen
X 'x-string'		Folge von bis zu 512 Sedezimalziffern ('0',..., 'F'). Wird keine geradzahlige Anzahl von Ziffern angegeben, wird die Folge linksbündig mit 0 ergänzt. Speicherbedarf: 1 byte pro Sedezimalziffernpaar
	P 'integer'	bis zu 16 Ziffern umfassende Dezimalzahl (wahlweise mit Vorzeichen). Sie wird von PERCON in gepacktem Format abgelegt.
	Z 'integer'	bis zu 32 Ziffern umfassende Dezimalzahl (wahlweise mit Vorzeichen). Sie wird von PERCON in gezontem Format abgelegt.
integer		Dezimalzahl im Bereich $-2^{31}$ bis $2^{31} - 1$

## 4.5 Kennwörter

PERCON führt intern eine Reihe von Informationen, auf die der Benutzer über Kennwörter zugreifen kann. Diese Kennwörter können je nach Bedarf in Ausgabesätze oder Druckseiten eingefügt oder als Vergleichskriterium verwendet werden. Diese Informationen werden von PERCON mehrfach sowohl für die Eingabe(n) als auch für die Ausgabe(n) zur Verfügung gestellt.

Die von PERCON mitgeführten Zähler werden jeweils nach der entsprechenden Aktion erhöht oder verändert. Sie sind mit 0 initialisiert. Z.B. wird der Satzzähler der Eingabedatei nach dem Lesen eines Satzes, der Satzzähler der Ausgabedatei nach dem Schreiben eines Satzes in die Ausgabedatei erhöht. Dies ist zu beachten, wenn man sich in Anweisungen auf die Kennwörter bezieht.

Folgende Kennwörter stehen zur Verfügung:

Bedeutung	Bezeichnung bei		Format	Länge
	SET-GROUP-ATTRIBUTES SET-PAGE-LAYOUT SET-RECORD-MAPPING	SELECT-INPUT-RECORDS		
Satzlänge	RECORD-LENGTH	RECLen	ZONED-DECIMAL	8 Zeichen
Blockzähler	BLOCK-COUNTER	BLKcNT	ZONED-DECIMAL	8 Zeichen
Zeichenzähler	BYTE-COUNTER	BYTCNT	ZONED-DECIMAL	10 Zeichen
Satzzähler	RECORD-COUNTER	RECCNT	ZONED-DECIMAL	8 Zeichen
Seitenzähler	PAGE-COUNTER	nicht angebbbar	ZONED-DECIMAL	6 Zeichen
Datum	DATE		CHARACTER	*
Zeit	TIME		CHARACTER	8 Zeichen hh-mm-ss

- \* Abhängig von CENTURY im Operanden DATE kann zwischen 2- bzw. 4stelligen Jahresangaben gewählt werden.

Bisheriges PERCON-Format:                   jj-mm-tt                   ( 8 byte lang)

Erweitertes PERCON-Format:               jjjj-mm-tt               (10 byte lang)

### Aufbereitung von Kennwörtern ab Ausgabelänge 1

Bei der Aufbereitung von Kennwörtern (außer DATE und TIME) in den Anweisungen SET-RECORD-MAPPING und SET-GROUP-ATTRIBUTES ist der Minimalwert für den Operanden OUTPUT-LENGTH gleich 1.

Die Angabe von OUTPUT-LENGTH=1 in Verbindung mit den Ausgabe-Formaten DECIMAL oder SIGNED-DECIMAL führt allerdings zu einem Semantikfehler (Meldung PER0007), da wegen der bei diesem Format vorhandenen Vorzeichenstelle mindestens der Wert 2 als Länge erforderlich ist.

Bei der Aufbereitung eines Kennworts dürfen gültige Stellen nicht abschnitten werden. Paßt ein aufzubereitender Wert eines Kennworts aufgrund einer zu kleinen Ausgabelänge nicht in das Ausgabefeld, so wird der Konvertierungsschritt mit der Meldung PER0042 abgebrochen.

**Hinweis zum Kennwort GROUP-COUNTER**

Beim Gruppieren von Eingabesätzen werden Zähler geführt, die sogenannten GROUP-COUNTER (siehe SET-GROUP-ATTRIBUTES-Anweisung Seite 141).

GROUP-COUNTER ist für die Gruppenstufen 1 bis 8 jeder Ausgabeeinheit verfügbar. Jeder Zähler ist durch seinen Dateikettungsnamen und durch eine Gruppenstufe eindeutig bestimmt.

Die Zähler sind mit 0 voreingestellt und werden zu folgenden Zeitpunkten aktualisiert:

- Zähler auf 1 gesetzt  
Nach dem Schreiben des Gruppenvorlaufs bzw. bei GROUP-HEADER=\*NONE, wenn das Kriterium zum Schreiben des Gruppenvorlaufs gegeben ist.
- Zähler um 1 erhöht  
Nach dem Lesen eines Eingabesatzes, wenn das Kriterium eines Gruppenwechsels nicht erfüllt ist.

**4.6 Formate**

Ein Feld ist ein Teil des Ein- oder Ausgabesatzes, das charakterisiert ist durch

- seine Anfangsposition im Satz,
- seine Länge,
- sein Format.

Das Format gibt an, wie das Feld im Eingabesatz zu interpretieren bzw. wie es bei seiner Übertragung in den Ausgabesatz umzuformen ist. Kennwörter und Literale besitzen ein implizit festgelegtes Format. PERCON kann folgende Formate bearbeiten:

**CHARACTER**

Das bezeichnete Feld enthält Zeichen oder soll Zeichen aufnehmen.  
Unter 'Zeichen' sind Buchstaben, Ziffern und Sonderzeichen zu verstehen.

**Beispiel**

Inhalt vom Eingabefeld Eingabeformat CHARACTER	Inhalt vom Ausgabefeld Ausgabeformat CHARACTER
X'C1C2F1F2F3F4' (6 byte lang)	C'AB1234' (6 byte lang)

## HEXADECIMAL

Nur Ausgabeformat. Das Eingabefeld wird sedezimal aufbereitet und in das bezeichnete Ausgabefeld ausgegeben, d.h. jedes Zeichen des Eingabefeldes wird in den entsprechenden Sedezimal-Code umgewandelt und nimmt im Ausgabefeld 2 byte ein.

### Beispiel

Inhalt vom Eingabefeld Eingabeformat CHARACTER	Inhalt vom Ausgabefeld Ausgabeformat HEXADECIMAL
X'C1C2F1F2F3F4' (6 byte lang)	C'C1C2F1F2F3F4' (12 byte lang)

## BINARY

Nur Ausgabeformat. Das Eingabefeld wird binär aufbereitet und in das bezeichnete Ausgabefeld ausgegeben, d.h. jedes Zeichen des Eingabefeldes wird in den entsprechenden Binär-Code umgewandelt und nimmt im Ausgabefeld 8 byte ein.

### Beispiel

Inhalt vom Eingabefeld Eingabeformat CHARACTER	Inhalt vom Ausgabefeld Ausgabeformat BINARY
X'C1C2' (2 byte lang)	C'1100000111000010' (16 byte lang)

## ZONED-DECIMAL

Das bezeichnete Feld enthält eine entpackte Dezimalzahl im EBCDI-Code oder soll eine solche aufnehmen. Das Vorzeichen der Zahl wird nicht berücksichtigt.

### Beispiel

Inhalt vom Eingabefeld Eingabeformat ZONED-DECIMAL	Inhalt vom Ausgabefeld Ausgabeformat ZONED-DECIMAL
X'F1F2F3F4' (4 byte lang)	C'1234' (4 byte lang)

Inhalt vom Eingabefeld Eingabeformat PACKED-DECIMAL	Inhalt vom Ausgabefeld Ausgabeformat ZONED-DECIMAL
X'123F' (2 byte lang)	C'123' (3 byte lang)

**SIGNED-DECIMAL**

Nur Ausgabeformat. Das Ausgabefeld soll eine entpackte Dezimalzahl aufnehmen, bei der das Vorzeichen (+ oder –) in jedem Fall im ersten Byte des Feldes enthalten ist.

Dies ist für die Längenangabe des Feldes zu berücksichtigen.

**Beispiel**

Inhalt vom Eingabefeld Eingabeformat <b>ZONED-DECIMAL</b>	Inhalt vom Ausgabefeld Ausgabeformat <b>SIGNED-DECIMAL</b>
X'F1F2F3F4' (4 byte lang)	C'+1234' (5 byte lang)

Inhalt vom Eingabefeld Eingabeformat <b>PACKED-DECIMAL</b>	Inhalt vom Ausgabefeld Ausgabeformat <b>SIGNED-DECIMAL</b>
X'123F' (2 byte lang)	C'+123' (4 byte lang)

**DECIMAL**

Nur Ausgabeformat. Das Ausgabefeld soll eine entpackte Dezimalzahl enthalten, bei der das Vorzeichen nur bei negativem Wert der Zahl ausgegeben wird. Bei positivem Wert wird an Stelle des Vorzeichens ein Leerzeichen ausgegeben. Dies ist bei der Längenangabe zu berücksichtigen.

**Beispiel**

Inhalt vom Eingabefeld Eingabeformat <b>ZONED-DECIMAL</b>	Inhalt vom Ausgabefeld Ausgabeformat <b>DECIMAL</b>
X'F1F2F3F4' (4 byte lang)	C' 1234' (5 byte lang)

Inhalt vom Eingabefeld Eingabeformat <b>PACKED-DECIMAL</b>	Inhalt vom Ausgabefeld Ausgabeformat <b>DECIMAL</b>
X'123F' (2 byte lang)	C' 123' (4 byte lang)

## PACKED-DECIMAL

Das bezeichnete Feld enthält eine gepackte Dezimalzahl bzw. soll eine solche aufnehmen. Die Länge einer gepackten Zahl darf maximal 16 byte betragen.

### Beispiel

Inhalt vom Eingabefeld Eingabeformat ZONED-DECIMAL	Inhalt vom Ausgabefeld Ausgabeformat PACKED-DECIMAL
X'F1F2F3F4' (4 byte lang)	X'01234F' (3 byte lang)

Inhalt vom Eingabefeld Eingabeformat PACKED-DECIMAL	Inhalt vom Ausgabefeld Ausgabeformat PACKED-DECIMAL
X'123F' (2 byte lang)	X'123F' (2 byte lang)

## ZONED-DECIMAL-LEFT

Nur Ausgabeformat für Kennwörter. Das Ausgabefeld soll eine linksbündige Dezimalzahl aufnehmen. Der rechte Teil des Ausgabefeldes wird mit Leerzeichen aufgefüllt.

### Beispiel

Inhalt vom	Ausgabeformat ZONED-DECIMAL-LEFT
RECORD-COUNTER	C'82579... ' (8 byte lang)

## TRANSLATION

Diese Angabe ist nur zulässig innerhalb der Feldbeschreibung des Operanden OUTPUT-FIELDS in der SET-RECORD-MAPPING-Anweisung. Die einzelnen Zeichen werden entsprechend einer Code-Tabelle umgesetzt.

## NO-TRANSLATION

Diese Angabe ist nur zulässig innerhalb der Feldbeschreibung des Operanden OUTPUT-FIELDS in der SET-RECORD-MAPPING-Anweisung. NO-TRANSLATION ist zu verwenden, wenn das Eingabefeld nicht in den Ausgabezeichensatz umgesetzt werden soll, z.B. bei gepackten Zahlen.

## Formatierte Zahlen mit einer Aufbereitungsmaske

Nur Ausgabeformat für numerische Ausgabefelder. Neben den Ausgabeformaten ZONED-DECIMAL, SIGNED-DECIMAL und ZONED-DECIMAL-LEFT kann für numerische Ausgabefelder auch eine Aufbereitungsmaske angegeben werden. In einer Maske können führende Nullen unterdrückt, Nachkommastellen angezeigt oder eine Zifferngruppierung durchgeführt werden.

Eine Maske wird in den Anweisungen SET-GROUP-ATTRIBUTES oder SET-RECORD-MAPPING mit dem Operanden OUTPUT-FORMAT=<c-string> vereinbart. Zudem muß der Operand OUTPUT-LENGTH den Wert \*STD haben.

In einer Maske können folgende Zeichen stehen:

Füllzeichen	:	alle abdruckbaren Zeichen
Ziffernzeichen	:	Ziffernselektor ('Z') Signifikanzstarter ('N')
Einfügungen	:	Zeichen ungleich 'Z' und 'N'

### – Maskenaufbau

Das 1. Zeichen der Maske wird als Füllzeichen interpretiert. Für Maskenzeichen rechts vom Füllzeichen gilt, daß die Anzahl der Ziffernzeichen ungerade sein muß und nur ein oder kein Signifikanzstarter vorhanden sein darf. Ist dies nicht der Fall, so wird die Maske mit einem Syntaxfehler als fehlerhaft zurückgewiesen. Zwischen den Ziffernzeichen können Einfügungen vorhanden sein.

### – Maskengröße

Enthält die Maske mehr Ziffernzeichen als in der aufzubereitenden Zahl Ziffern vorhanden sind, so wird die aufzubereitende Zahl linksbündig mit Nullen erweitert.

Ist die Anzahl der Ziffern der aufzubereitenden Zahl größer als die der Ziffernzeichen in der Maske, so wird die aufzubereitende Zahl (von links beginnend) gekürzt. Werden bei diesem Kürzungsvorgang relevante Ziffern ( $\neq$  führende Nullen) abgeschnitten, so wird der Konvertierungsschritt sofort fehlerhaft abgebrochen (siehe Meldung PER0042).

Bei den bisherigen Ausgabeformaten von PERCON gilt weiterhin die Einschränkung, daß die Anzahl der Ziffernstellen des Ausgabefeldes größer gleich der des Eingabefeldes sein muß.

### – Maskenverarbeitung

Die Maske wird von links nach rechts abgearbeitet. Alle Zeichen vor dem ersten Ziffernzeichen ('Z', 'N') werden durch das Füllzeichen ersetzt.

In weiterer Folge werden alle Zeichen, auch die Einfügungszeichen, solange durch das Füllzeichen ersetzt, bis entweder ein Ziffernselektor durch eine Ziffer  $\neq 0$  ersetzt wird oder in der Maske ein Signifikanzstarter auftritt.

Danach werden die Ziffernzeichen durch die entsprechenden Ziffern ersetzt, einzufügende Zeichen bleiben unverändert.

Einzufügende Zeichen hinter dem letzten Ziffernzeichen bleiben unverändert, wenn die aufzubereitende Zahl negativ ist, sonst werden sie durch das Füllzeichen ersetzt.

### Beispiel 1

```
FIELD (INPUT-POSITION=...,-
      INPUT-LENGTH=5,-
      INPUT-FORMAT=*PACKED-DECIMAL,-
      OUTPUT-POSITION=...,-
      OUTPUT-LENGTH=*STD,-
      OUTPUT-FORMAT='*ZZN.ZZZ-')
```

Es ergibt sich folgender Umsetzungsvorgang:

Aufzubereitendes Feld:	0 0 0 0 0 0 0 3 8 C
Angegebene Maske:	* Z Z N Z . Z Z Z -
Ausgabefeld:	* * * * 0 . 0 3 8 *

Die gepackte Zahl ist positiv, deshalb wird das Minuszeichen hinter der Zahl durch das Füllzeichen ersetzt. Beim aufzubereitenden Feld wird wegen der Anpassung an die Maske das führende Byte, das Null enthält, weggeschnitten. Das Abschneiden von Feldteilen ungleich Null würde zu einem Abbruch mit Fehlermeldung PER0042 führen.

### Beispiel 2

```
FIELD (INPUT-POSITION=...,-
      INPUT-LENGTH=4,-
      INPUT-FORMAT=*ZONED-DECIMAL,-
      OUTPUT-POSITION=...,-
      OUTPUT-LENGTH=*STD,-
      OUTPUT-FORMAT='xZ ZNZ,ZZZ -')
```

Es ergibt sich folgender Umsetzungsvorgang:

Aufzubereitendes Feld:	F 3 F 9 F 7 D 1
Angegebene Maske:	x Z Z N Z , Z Z Z -
Ausgabefeld:	x x x x x 3 , 9 7 1 -

Da die gezonte Zahl negativ ist, wird das Leer- und Minuszeichen hinter der Zahl in das Ausgabefeld übernommen.



## 4.7 Beschreibung der Anweisungen

Die PERCON-Anweisungen sind in alphabetischer Reihenfolge ihrer Namen nach folgender Gliederung beschrieben:

- Anweisungsname und Funktion
- Beschreibung der Anweisungsfunktion
- Darstellung des Anweisungsformats
- Beschreibung der Operanden

## ASSIGN-INPUT-FILE

### Zuweisen einer Eingabedatei

Diese Anweisung ist nur beim Kopieren von Dateien zulässig. Sie weist die Eingabedatei zu. Für Fehlersituationen können ferner das Fehlverhalten und die anzuschließenden Benutzermodule bestimmt werden.

(Abschnitt 1 von 3)

#### ASSIGN-INPUT-FILE

**FILE** = \*DISK-FILE (...) / \*TAPE-FILE(...) / \*SYSDTA(...) / \*DISKETTE(...)

\*DISK-FILE (...)

**NAME** = \*STD / <full-filename 1..54>

**REMOVE-FILE-LINK** = \*STD / \*NO / \*YES(...)

        \*YES(...)

**RELEASE-DEVICE** = \*YES / \*NO

**OPEN-ERROR** = \*STD / \*FINISH-INPUT(...) / \*CONTINUE-NEXT-FILE(...) / \*DIALOG /  
                    \*MODULE(...)

    \*FINISH-INPUT(...)

**TERMINATION** = \*ABNORMAL / \*NORMAL

    \*CONTINUE-NEXT-FILE(...)

**TERMINATION** = \*ABNORMAL / \*NORMAL

    \*MODULE(...)

**NAME** = <name 1..8>

**PARITY-ERROR** = \*TERMINATE-ABNORMAL / \*IGNORE-ERROR / \*CONTINUE-NEXT-BLOCK /  
                    \*DIALOG / \*MODULE(...)

    \*MODULE(...)

**NAME** = <name 1..8>

**LENGTH-ERROR** = \*TERMINATE-ABNORMAL / \*IGNORE-ERROR / \*CONTINUE-NEXT-BLOCK /  
                    \*DIALOG / \*MODULE(...)

    \*MODULE(...)

**NAME** = <name 1..8>

**EMPTY-BLOCK** = \*STD / \*TAKE / \*SKIP

Fortsetzung ➡

```

*TAPE-FILE(...)
  NAME = *STD / <full-filename 1..54>
  ,REMOVE-FILE-LINK = *STD / *NO / *YES(...)
    *YES(...)
      | RELEASE-DEVICE = *YES / *NO
      | ,UNLOAD-RELEASED-TAPE = *NO / *YES
  ,END-POSITION = *BEGIN-OF-TAPE / *BEGIN-OF-FILE / *END-OF-FILE / *UNLOAD-TAPE
  ,CODE-TRANSLATION = *NO / *MODULE(...)
    *MODULE(...)
      | NAME = <name 1..8>
  ,LABEL-EXIT = *NO / *MODULE(...) / *SYSOUT(...)
    *MODULE(...)
      | NAME = <name 1..8>
      | ,CONTROLLED-LABEL = *STD / list-poss(2000): *VOL-USER-LABEL /
        | *HDR-USER-LABEL / *EOV-USER-LABEL / *END-USER-LABEL /
        | *CLOSE-POSITION / *LABEL-ERROR
    *SYSOUT(...)
      | CONTROLLED-LABEL = *STD / list-poss(2000): *VOL-USER-LABEL /
        | *HDR-USER-LABEL / *EOV-USER-LABEL / *END-USER-LABEL
  ,OPEN-ERROR = *STD / *FINISH-INPUT(...) / *CONTINUE-NEXT-FILE(...) / *DIALOG /
    | *MODULE(...)
  *FINISH-INPUT(...)
    | TERMINATION = *ABNORMAL / *NORMAL
  *CONTINUE-NEXT-FILE(...)
    | TERMINATION = *ABNORMAL / *NORMAL
  *MODULE(...)
    | NAME = <name 1..8>

```

Fortsetzung ➡

(Abschnitt 3 von 3)

```

,PARITY-ERROR = *TERMINATE-ABNORMAL / *IGNORE-ERROR / *CONTINUE-NEXT-BLOCK /
                  *DIALOG / *MODULE(...)

    *MODULE(...)
    |   NAME = <name 1..8>

,LENGTH-ERROR = *TERMINATE-ABNORMAL / *IGNORE-ERROR / *CONTINUE-NEXT-BLOCK /
                  *DIALOG / *MODULE(...)

    *MODULE(...)
    |   NAME = <name 1..8>

,EMPTY-BLOCK = *STD / *TAKE / *SKIP

*SYSDTA(...)

    RECORD-SIZE = 32767 / <integer 4..32767>

,INPUT-ERROR = *TERMINATE-ABNORMAL / *IGNORE-ERROR / *CONTINUE-NEXT-RECORD /
                *DIALOG / *MODULE(...)

    *MODULE(...)
    |   NAME = <name 1..8>

*DISKETTE(...)

    VOLUME = list-poss(10): <alphanum-name 1..6>

,FILE-NAME = *NONE / <name 1..8>

,RECORD-SIZE = 128 / <integer 1..2048>

,INPUT-ERROR = *TERMINATE-ABNORMAL / *IGNORE-ERROR / *CONTINUE-NEXT-RECORD /
                *DIALOG / *MODULE(...)

    *MODULE(...)
    |   NAME = <name 1..8>

,LINK-NAME = PCIN / <full-filename 1..8 without-gen>

,INPUT-EXIT = *NO / *MODULE(...)

    *MODULE(...)
    |   NAME = <name 1..8>

```

## FILE =

Mit FILE entscheidet man sich für ein Eingabemedium.

Bei FILE=\*DISK-FILE oder \*TAPE-FILE können ergänzende Angaben über das SET-FILE-LINK-Kommando gemacht werden. Beim Zuweisen einer Eingabedatei über ein SET-FILE-LINK-Kommando oder über eine ASSIGN-INPUT-FILE-Anweisung wird ein Eintrag in der Prozeßdateitabelle (TFT) unter dem Dateikettungsamen erzeugt.

**FILE = \*DISK-FILE (...)**

Die Eingabedatei liegt auf Platte.

**NAME = \*STD / <full-filename 1..54>**

Name der Eingabedatei.

Er wird nicht benötigt, wenn der Dateiname bereits in einem SET-FILE-LINK-Kommando angegeben wurde, auf das sich die ASSIGN-INPUT-FILE-Anweisung über den Operanden LINK-NAME bezieht. Der Dateiname im SET-FILE-LINK-Kommando hat Vorrang vor dem hier angegebenen.

**NAME = \*STD**

Standardwert ist der zum LINK-NAME gehörende Dateiname, der im SET-FILE-LINK-Kommando festgelegt wurde. Wurde kein SET-FILE-LINK-Kommando abgesetzt, wird der durch den Operand LINK-NAME vereinbarte Name als Dateiname verwendet.

**REMOVE-FILE-LINK =**

Legt fest, ob und in welcher Weise der TFT-Eintrag nach der Dateibearbeitung gelöscht oder erhalten werden soll.

**REMOVE-FILE-LINK = \*STD**

PERCON löscht den TFT-Eintrag nur dann, wenn er über eine ASSIGN-INPUT-FILE-Anweisung von PERCON selbst angelegt wurde. Wurde die Datei mit einem SET-FILE-LINK-Kommando zugewiesen, bleibt der TFT-Eintrag erhalten.

**REMOVE-FILE-LINK = \*NO**

Der TFT-Eintrag bleibt erhalten, unabhängig davon, wie er angelegt wurde.

**REMOVE-FILE-LINK = \*YES(...)**

Der TFT-Eintrag wird gelöscht, unabhängig davon, wie er angelegt wurde.

**RELEASE-DEVICE = \*YES / \*NO**

Dient zur Verwaltung des Plattengeräts nach der Bearbeitung.

Nur für Privatplatten.

Mit \*YES wird das Plattengerät freigegeben, mit NO bleibt es dem Prozeß zugeordnet.

**OPEN-ERROR =**

Steuert das Verhalten von PERCON bei Eröffnungsfehlern der Eingabedatei.

**OPEN-ERROR = \*STD**

Dialogbetrieb: \*DIALOG

Stapelbetrieb: \*FINISH-INPUT(TERMINATION=\*ABNORMAL)

**OPEN-ERROR = \*FINISH-INPUT(...)**

Die Eingabedateien, die bis zu der angesprochenen Eingabedatei eröffnet werden konnten, werden bearbeitet.

**TERMINATION = \*ABNORMAL / \*NORMAL**

Der Konvertierungsschritt wird entweder fehlerhaft (\*ABNORMAL) oder normal (\*NORMAL) beendet.

**OPEN-ERROR = \*CONTINUE-NEXT-FILE(...)**

Die nächste Eingabedatei wird bearbeitet.

**TERMINATION = \*ABNORMAL / \*NORMAL**

Der Konvertierungsschritt wird entweder fehlerhaft (\*ABNORMAL) oder normal (\*NORMAL) beendet.

**OPEN-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf.

Folgende Angaben können gemacht werden:

HA: Der Konvertierungsschritt wird fehlerhaft beendet.

HN: Der Konvertierungsschritt wird normal beendet.

SA: Die nächste Eingabedatei wird bearbeitet.  
Der Konvertierungsschritt wird fehlerhaft beendet.

SN: Die nächste Eingabedatei wird bearbeitet.  
Der Konvertierungsschritt wird normal beendet.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert für Stapelbetrieb tritt in Kraft.

**OPEN-ERROR = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 190, Anschluß für die Eröffnungsfehlerbehandlung).

**PARITY-ERROR =**

Steuert das Verhalten von PERCON, wenn ein Block einer SAM-Datei wegen eines Paritätsfehlers nicht oder nur unvollständig gelesen werden kann.

**PARITY-ERROR = \*TERMINATE-ABNORMAL**

Die Ein- und Ausgabedateien werden geschlossen und der Konvertierungsschritt wird fehlerhaft beendet.

**PARITY-ERROR = \*IGNORE-ERROR**

Der Fehler wird ignoriert. PERCON verhält sich so, als ob der fragliche Block gelesen werden konnte. Der Block wird fehlerhaft in die Ausgabedatei übernommen.

**PARITY-ERROR = \*CONTINUE-NEXT-BLOCK**

Der fehlerhafte Block wird übergangen.

**PARITY-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf.

Folgende Angaben können gemacht werden:

H: Der Konvertierungsschritt wird fehlerhaft beendet.

I: Der Fehler wird ignoriert.

S: Der Block wird übergangen.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert tritt in Kraft.

**PARITY-ERROR = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder Einsprungpunkt in dem Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 189, Anschluß für die Lesefehlerbehandlung).

**LENGTH-ERROR =**

Steuert das Verhalten von PERCON, wenn ein Block gelesen werden soll, dessen Länge von der Summe der in ihm enthaltenen Sätze abweicht.

**LENGTH-ERROR = \*TERMINATE-ABNORMAL**

Die Ein- und Ausgabedateien werden geschlossen und der Konvertierungsschritt wird fehlerhaft beendet.

**LENGTH-ERROR = \*IGNORE-ERROR**

Der Fehler wird ignoriert und der fragliche Block wird trotzdem gelesen.

**LENGTH-ERROR = \*CONTINUE-NEXT-BLOCK**

Der fehlerhafte Block wird übergangen.

**LENGTH-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf.

Folgende Angaben können gemacht werden:

H: Der Konvertierungsschritt wird fehlerhaft beendet.

I: Der Fehler wird ignoriert.

S: Der Block wird übergangen.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert tritt in Kraft.

### **LENGTH-ERROR = \*MODULE(...)**

**NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 189, Anschluß für die Längenfehlerbehandlung).

### **EMPTY-BLOCK =**

Bearbeitung der Leerblöcke (siehe Seite 38, PAM-Eingabedateien).

### **EMPTY-BLOCK = \*STD**

\*SKIP bei BLOCK-CONTROL-INFO = \*PAMKEY,

\*TAKE bei BLOCK-CONTROL-INFO = \*WITHIN-DATA.

### **EMPTY-BLOCK = \*TAKE**

Leerblöcke werden als Eingabesätze behandelt.

### **EMPTY-BLOCK = \*SKIP**

Leerblöcke werden überlesen.

### **FILE = \*TAPE-FILE(...)**

Die Eingabedatei liegt auf Band.

### **NAME = \*STD / <full-filename 1..54>**

Name der Eingabedatei.

Er wird nicht benötigt, wenn der Dateiname bereits in einem SET-FILE-LINK-Kommando angegeben wurde, auf das sich die ASSIGN-INPUT-FILE-Anweisung über den Operanden LINK-NAME bezieht. Der Dateiname im SET-FILE-LINK-Kommando hat Vorrang vor dem hier angegebenen.

### **NAME = \*STD**

Standardwert ist der zum LINK-NAME gehörende Dateiname, der im SET-FILE-LINK-Kommando festgelegt wurde. Wurde kein SET-FILE-LINK-Kommando abgesetzt, wird der durch den Operand LINK-NAME vereinbarte Name als Dateiname verwendet.

### **REMOVE-FILE-LINK =**

Legt fest, ob und in welcher Weise der TFT-Eintrag nach der Dateibearbeitung gelöscht oder erhalten werden soll.

### **REMOVE-FILE-LINK = \*STD**

PERCON löscht den TFT-Eintrag nur dann, wenn er über eine ASSIGN-INPUT-FILE-Anweisung von PERCON selbst angelegt wurde. Wurde die Datei mit einem SET-FILE-LINK-Kommando zugewiesen, bleibt der TFT-Eintrag erhalten.

### **REMOVE-FILE-LINK = \*NO**

Der TFT-Eintrag bleibt erhalten, unabhängig davon, wie er angelegt wurde.

### **REMOVE-FILE-LINK = \*YES(...)**

Der TFT-Eintrag wird gelöscht, unabhängig davon, wie er angelegt wurde.



**RELEASE-DEVICE = \*YES / \*NO**

Dient zur Verwaltung des Bandgeräts nach der Bearbeitung. Mit \*YES wird das Bandgerät freigegeben, mit \*NO bleibt es diesem Prozeß zugeordnet.

**UNLOAD-RELEASED-TAPE = \*NO / \*YES**

Dient zur Verwaltung des Bandes nach der Bearbeitung. Das Band bleibt geladen, wenn \*NO eingegeben wird. Das Band wird entladen, wenn \*YES eingegeben wird.

**END-POSITION =**

Gibt an, wie das Band nach der Bearbeitung der Datei zu positionieren ist.

**END-POSITION = \*BEGIN-OF-TAPE**

Das Band wird zum Bandanfang zurückgespult.

**END-POSITION = \*BEGIN-OF-FILE**

Das Band wird auf den Anfang der gerade bearbeiteten Datei positioniert.

**END-POSITION = \*END-OF-FILE**

Die Position des Bandes wird nach der Bearbeitung nicht verändert.

**END-POSITION = \*UNLOAD-TAPE**

Das Band wird zurückgespult und entladen.

**CODE-TRANSLATION =**

Gibt an, welche Code-Umsetzung der Eingabesätze erfolgen soll.

**CODE-TRANSLATION = \*NO**

Es erfolgt keine Code-Umsetzung.

**CODE-TRANSLATION = \*MODULE(...)**

Die Code-Umsetzungstabelle befindet sich in der über SET-TASKLIB zugewiesenen Modulbibliothek. Ihr Inhalt sind die 256 Sedezimalverschlüsselungen (X'00' bis X'FF') der Zeichen, in die die Eingabezeichen umzuwandeln sind.

Bei der Umwandlung wird vom sedezimalen Wert des Eingabezeichens ausgegangen. Dieser Wert wird zur Anfangsadresse der Umsetzungstabelle addiert; das Zeichen an der sich ergebenden Position der Umsetzungstabelle ersetzt das Eingabezeichen.

**NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die Code-Umsetzungstabelle enthält.

**LABEL-EXIT =**

Beschreibt die Behandlung von Benutzerkennsätzen.

**LABEL-EXIT = \*NO**

Benutzerkennsätze werden nicht gesondert bearbeitet.

**LABEL-EXIT = \*MODULE(...)**

**NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 182, Anschluß für die Kennsatzbearbeitung).

**CONTROLLED-LABEL =**

Der Benutzermodul wird bei den angegebenen Bearbeitungsabschnitten angesprungen.

**CONTROLLED-LABEL = \*STD**

Der Benutzermodul wird bei der Bearbeitung folgender vier Benutzer-Kennsätze angesprungen:

VOL-USER-LABEL, HDR-USER-LABEL, EOVS-USER-LABEL, END-USER-LABEL

**CONTROLLED-LABEL = \*VOL-USER-LABEL**

Der Benutzermodul wird bei der Bearbeitung der Benutzer-VOLUME-Kennsätze angesprungen.

**CONTROLLED-LABEL = \*HDR-USER-LABEL**

Der Benutzermodul wird bei der Bearbeitung der Benutzer-HEADER-Kennsätze angesprungen.

**CONTROLLED-LABEL = \*EOVS-USER-LABEL**

Der Benutzermodul wird bei der Bearbeitung der Benutzer-END-OF-VOLUME-Kennsätze bei Bandwechsel angesprungen.

**CONTROLLED-LABEL = \*END-USER-LABEL**

Der Benutzermodul wird bei der Bearbeitung der Benutzer-END-Kennsätze angesprungen.

**CONTROLLED-LABEL = \*CLOSE-POSITION**

Der Benutzermodul wird zur Positionierung des Bandes bei der CLOSE-Bearbeitung angesprungen.

**CONTROLLED-LABEL = \*LABEL-ERROR**

Der Benutzermodul wird bei fehlerhaften Bandendekennsätzen angesprungen.

**LABEL-EXIT = \*SYSOUT(...)**

PERCON protokolliert die Benutzerkennsätze nach SYSOUT. Diese Angabe ist nur für Dateien mit Standardkennsätzen zulässig.

**CONTROLLED-LABEL =**

Die angegebenen Benutzerkennsatzgruppen werden nach SYSOUT ausgegeben.

**CONTROLLED-LABEL = \*STD**

Folgende Benutzerkennsatzgruppen werden nach SYSOUT ausgegeben:

VOL-USER-LABEL, HDR-USER-LABEL, EOVS-USER-LABEL, END-USER-LABEL

**CONTROLLED-LABEL = \*VOL-USER-LABEL**

Benutzer-VOLUME-Kennsätze werden nach SYSOUT ausgegeben.

**CONTROLLED-LABEL = \*HDR-USER-LABEL**

Benutzer-HEADER-Kennsätze werden nach SYSOUT ausgegeben.

**CONTROLLED-LABEL = \*EOV-USER-LABEL**

Benutzer-END-OF-VOLUME-Kennsätze werden nach SYSOUT ausgegeben.

**CONTROLLED-LABEL = \*END-USER-LABEL**

Benutzer-END-Kennsätze werden nach SYSOUT ausgegeben.

**OPEN-ERROR =**

Steuert das Verhalten von PERCON bei Eröffnungsfehlern der Eingabedatei.

**OPEN-ERROR = \*STD**

Dialogbetrieb: \*DIALOG

Stapelbetrieb: \*FINISH-INPUT(TERMINATION=\*ABNORMAL)

**OPEN-ERROR = \*FINISH-INPUT(...)**

Die Eingabedateien, die bis zu der angesprochenen Eingabedatei eröffnet werden konnten, werden bearbeitet.

**TERMINATION = \*ABNORMAL / \*NORMAL**

Der Konvertierungsschritt wird entweder fehlerhaft (\*ABNORMAL) oder normal (\*NORMAL) beendet.

**OPEN-ERROR = \*CONTINUE-NEXT-FILE(...)**

Die nächste Eingabedatei wird bearbeitet.

**TERMINATION = \*ABNORMAL / \*NORMAL**

Der Konvertierungsschritt wird entweder fehlerhaft (\*ABNORMAL) oder normal (\*NORMAL) beendet.

**OPEN-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf. Folgende Angaben können gemacht werden:

HA: Der Konvertierungsschritt wird fehlerhaft beendet.

HN: Der Konvertierungsschritt wird normal beendet.

SA: Die nächste Eingabedatei wird bearbeitet.  
Der Konvertierungsschritt wird fehlerhaft beendet.

SN: Die nächste Eingabedatei wird bearbeitet.  
Der Konvertierungsschritt wird normal beendet.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert für Stapelbetrieb tritt in Kraft.

**OPEN-ERROR = \*MODULE(...)**

**NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 190, Anschluß für die Eröffnungsfehlerbehandlung).

**PARITY-ERROR =**

Steuert das Verhalten von PERCON, wenn ein Block einer SAM-Datei oder BTAM-Datei wegen eines Paritätsfehlers nicht oder nur unvollständig gelesen werden kann.

**PARITY-ERROR = \*TERMINATE-ABNORMAL**

Die Ein- und Ausgabedateien werden geschlossen und der Konvertierungsschritt wird fehlerhaft beendet.

**PARITY-ERROR = \*IGNORE-ERROR**

Der Fehler wird ignoriert. PERCON verhält sich so, als ob der fragliche Block gelesen werden konnte. Der Block wird fehlerhaft in die Ausgabedatei übernommen.

**PARITY-ERROR = \*CONTINUE-NEXT-BLOCK**

Der fehlerhafte Block wird übergangen.

**PARITY-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf. Folgende Angaben können gemacht werden:

H: Der Konvertierungsschritt wird fehlerhaft beendet.

I: Der Fehler wird ignoriert.

S: Der Block wird übergangen.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert tritt in Kraft.

**PARITY-ERROR = \*MODULE(...)**

**NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 189, Anschluß für die Lesefehlerbehandlung).

**LENGTH-ERROR =**

Steuert das Verhalten von PERCON, wenn ein Block gelesen werden soll, dessen Länge von der Länge abweicht, die im Operand BUFFER-LENGTH des SET-FILE-LINK-Kommandos festgelegt wurde.

**LENGTH-ERROR = \*TERMINATE-ABNORMAL**

Die Ein- und Ausgabedateien werden geschlossen und der Konvertierungsschritt wird fehlerhaft beendet.

**LENGTH-ERROR = \*IGNORE-ERROR**

Der Fehler wird ignoriert und der fragliche Block wird trotzdem verarbeitet.

**LENGTH-ERROR = \*CONTINUE-NEXT-BLOCK**

Der fehlerhafte Block wird übergangen.

**LENGTH-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf. Folgende Angaben können gemacht werden:

H: Der Konvertierungsschritt wird fehlerhaft beendet.

I: Der Fehler wird ignoriert.

S: Der Block wird übergangen.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert tritt in Kraft.

**LENGTH-ERROR = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 189, Anschluß für die Längenfehlerbehandlung).

**EMPTY-BLOCK =**

Bearbeitung der Leerblöcke (siehe Seite 38, PAM-Eingabedateien).

**EMPTY-BLOCK = \*STD**

\*SKIP bei BLOCK-CONTROL-INFO = \*PAMKEY,

\*TAKE bei BLOCK-CONTROL-INFO = \*WITHIN-DATA.

**EMPTY-BLOCK = \*TAKE**

Leerblöcke werden als Eingabesätze behandelt.

**EMPTY-BLOCK = \*SKIP**

Leerblöcke werden überlesen.

**FILE = \*SYSDTA(...)**

Die Eingabe erfolgt aus der Systemdatei SYSDTA. Die Eingabe beginnt mit dem Datensatz, der nach einer der Anweisungen START-CONVERSION oder END von SYSDTA gelesen wird und endet mit Erreichen der Dateiendebedingung. Diese ist erfüllt, wenn

- ein Datensatz mit '/EOF' beginnt
- das Ende einer katalogisierten Datei erreicht wird
- in Prozeduren oder Stapelaufträgen ein BS2000-Kommando erkannt wird.

*Hinweise*

- Bei Erreichen der Dateiendebedingung beim Lesen von einer Datei wird SYSDTA wieder SYSCMD zugewiesen.
- Bei Eingabe der Daten über Terminal wird die EOF-Bedingung durch folgenden Ablauf signalisiert:
  - Unterbrechungstaste K2
  - /EOF
  - /RESUME-PROGRAM

**RECORD-SIZE = 32767 / <integer 4..32767>**

Gibt die maximale Länge der zu bearbeitenden Sätze an.

**INPUT-ERROR =**

Steuert das Verhalten von PERCON, wenn bei der Eingabe ein Fehler auftritt.

**INPUT-ERROR = \*TERMINATE-ABNORMAL**

Die Ein- und Ausgabedateien werden geschlossen und der Konvertierungsschritt wird fehlerhaft beendet.

**INPUT-ERROR = \*IGNORE-ERROR**

Der Fehler wird ignoriert. PERCON verhält sich so, als ob der fragliche Satz gelesen werden konnte. Der Satz wird fehlerhaft übernommen.

**INPUT-ERROR = \*CONTINUE-NEXT-BLOCK**

Der fehlerhafte Satz wird übergangen.

**INPUT-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf. Folgende Angaben können gemacht werden:

H: Der Konvertierungsschritt wird fehlerhaft beendet.

I: Der Fehler wird ignoriert.

S: Der Satz wird übergangen.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert tritt in Kraft.

**INPUT-ERROR = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 189, Anschluß für die Lese- und Längenfehlerbehandlung).

**FILE = \*DISKETTE(...)**

Die Eingabedatei liegt auf Diskette.

**VOLUME = list-poss: <alphanum-name 1..6>**

Archivnummer(n) der Diskette(n).

**FILE-NAME = \*NONE / <name 1..8>**

Dateiname

**FILE-NAME = \*NONE**

Es wird kein Dateiname angegeben, sondern der im Operand LINK-NAME angegebene Name wird verwendet.

**RECORD-SIZE = 128 / <integer 1..128>**

Gibt die Länge der zu bearbeitenden Sätze an.

**INPUT-ERROR =**

Steuert das Verhalten von PERCON, wenn bei der Eingabe ein Fehler auftritt.

**INPUT-ERROR = \*TERMINATE-ABNORMAL**

Die Ein- und Ausgabedateien werden geschlossen und der Konvertierungsschritt wird fehlerhaft beendet.

**INPUT-ERROR = \*IGNORE-ERROR**

Der Fehler wird ignoriert. PERCON verhält sich so, als ob der fragliche Satz gelesen werden konnte. Der Satz wird fehlerhaft in die Ausgabedatei übernommen.

**INPUT-ERROR = \*CONTINUE-NEXT-RECORD**

Der fehlerhafte Satz wird übergangen.

**INPUT-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf.

Folgende Angaben können gemacht werden:

H: Der Konvertierungsschritt wird fehlerhaft beendet.

I: Der Fehler wird ignoriert.

S: Der Satz wird übergangen.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert tritt in Kraft.

**INPUT-ERROR = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 189, Anschluß für die Lese- und Längenfehlerbehandlung).

**LINK-NAME = PCIN / <full-filename 1..8 without-gen>**

PERCON-Dateikettungsname zur Bezugnahme in Kennwörtern und im SET-FILE-LINK-Kommando.

**INPUT-EXIT =**

Name des Benutzermoduls oder des Einsprungpunktes in dem Benutzermodul, der die Eingabesätze unmittelbar nach dem Lesen bearbeitet.

**INPUT-EXIT = \*NO**

Keine Bearbeitung durch einen Benutzermodul.

**INPUT-EXIT = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 184, Anschluß für die Eingabe).



## ASSIGN-INPUT-TAPE

### Zuweisen eines Eingabebandes

Diese Anweisung ist nur beim Editieren bzw. Duplizieren von Bändern zulässig. Sie weist das Eingabeband oder MF/MV-Set zu. Für Fehlersituationen können ferner das Fehlerverhalten und die anzuschließenden Benutzermodule bestimmt werden.

#### ASSIGN-INPUT-TAPE

```

VOLUME = list-poss(100): <alphanum-name 1..6>
, LINK-NAME = PCIN / <full-filename 1..8 without-gen>
, END-POSITION = *BEGIN-OF-TAPE / *END-OF-TAPE / *UNLOAD-TAPE
, CODE = *EBCDIC / *ISO7 / *OWN(...)
    *OWN(...)
        | NAME = <name 1..8>
, DEVICE-TYPE = T1600 / <structured-name 1..8>
, BUFFER-LENGTH = *STD / <integer 18..32768>
, PARITY-ERROR = *TERMINATE-ABNORMAL / *IGNORE-ERROR / *CONTINUE-NEXT-BLOCK /
    *DIALOG / *MODULE(...)
    *MODULE(...)
        | NAME = <name 1..8>
, LENGTH-ERROR = *TERMINATE-ABNORMAL / *IGNORE-ERROR / *CONTINUE-NEXT-BLOCK /
    *DIALOG / *MODULE(...)
    *MODULE(...)
        | NAME = <name 1..8>

```

#### **VOLUME = list-poss(100): <alphanum-name 1..6>**

Archivnummer des Eingabebandes.

Beim Duplizieren eines Bandes können mit diesem Operanden die Archivnummern eines Eingabe-MF/MV-Sets angegeben werden, das bearbeitet werden soll.

#### **LINK-NAME = PCIN / <full-filename 1..8 without-gen>**

Interner PERCON-Dateikettungsname zur Bezugnahme in Kennwörtern.

#### **END-POSITION =**

Position des Bandes nach der Bearbeitung.

**END-POSITION = \*BEGIN-OF-TAPE**

Das Band wird zur Anfangsmarke zurückgespult.

**END-POSITION = \*END-OF-TAPE**

Die Position des Bandes wird nach der Bearbeitung nicht verändert.

**END-POSITION = \*UNLOAD-TAPE**

Das Band wird zurückgespult und entladen.

**CODE =**

Code, in dem das Band beschrieben ist. Bei der Code-Umsetzung ist zu beachten, daß alle Daten (auch Kennsätze) des Bandes umcodiert werden.

**CODE = \*EBCDIC**

Das Band ist im EBCDI-Code beschrieben und wird ohne Code-Umsetzung bearbeitet.

**CODE = \*ISO7**

Das Band ist im ISO7-Code beschrieben. Die Eingabedaten werden in den EBCDI-Code umgesetzt.

**CODE = \*OWN(...)**

Der Bandinhalt soll über eine benutzereigene Tabelle (Code-Umsetzungstabelle) umgesetzt werden. Sie befindet sich in der über SET-TASKLIB zugewiesenen Modulbibliothek. Ihr Inhalt sind die 256 Sedezimalverschlüsselungen (X'00' bis X'FF') der Zeichen, in die die Eingabezeichen umzuwandeln sind.

Bei der Umwandlung wird vom sedezimalen Wert des Eingabezeichens ausgegangen. Dieser Wert wird zur Anfangsadresse der Umsetzungstabelle addiert; das Zeichen an der sich ergebenden Position der Umsetzungstabelle ersetzt das Eingabezeichen.

**NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die Code-Umsetzungstabelle enthält.

**DEVICE-TYPE =**

Bandgerätetyp

**DEVICE-TYPE = T1600**

9-Spur-Bandgerät mit 1600 Bpl.

**DEVICE-TYPE = <structured-name 1..8>**

Explizite Angabe des Bandgerätetyps.

**BUFFER-LENGTH = \*STD / <integer 18..32768>**

Maximale Blocklänge des Eingabebandes oder MF/MV-Sets. Kommen Blöcke mit größerer Länge vor, wird die im Operanden LENGTH-ERROR angegebene Aktion ausgeführt.

**BUFFER-LENGTH = \*STD**

Standardwert: 32768 byte

**PARITY-ERROR =**

Steuert das Verhalten von PERCON, wenn ein Block wegen eines Paritätsfehlers nicht oder nur unvollständig gelesen werden kann.

**PARITY-ERROR = \*TERMINATE-ABNORMAL**

Der Konvertierungsschritt wird fehlerhaft beendet.

**PARITY-ERROR = \*IGNORE-ERROR**

Der Fehler wird ignoriert. PERCON verhält sich so, als ob der fragliche Block gelesen werden konnte. Der Block wird fehlerhaft in die Ausgabedatei/das Ausgabeband übernommen.

**PARITY-ERROR = \*CONTINUE-NEXT-BLOCK**

Der fehlerhafte Block wird übergangen.

**PARITY-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf.

Folgende Angaben können gemacht werden:

H: Der Konvertierungsschritt wird fehlerhaft beendet.

I: Der Fehler wird ignoriert.

S: Der Block wird übergangen.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert tritt in Kraft.

**PARITY-ERROR = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 189, Anschluß für die Lesefehlerbehandlung).

**LENGTH-ERROR =**

Steuert das Verhalten von PERCON, wenn ein Block gelesen werden soll, dessen Länge die im Operand BUFFER-LENGTH angegebene überschreitet. Dies gilt nicht für Magnetbandkassetten und Bandgeräte mit Typ-2-Kanalsteuerungen, bei denen die Eingabeblocks auf den BUFFER-LENGTH-Wert gekürzt werden.

**LENGTH-ERROR = \*TERMINATE-ABNORMAL**

Der Konvertierungsschritt wird fehlerhaft beendet.

**LENGTH-ERROR = \*IGNORE-ERROR**

Der Fehler wird ignoriert und der fragliche Block übernommen.

**LENGTH-ERROR = \*CONTINUE-NEXT-BLOCK**

Der fehlerhafte Block wird übergangen.

**LENGTH-ERROR = \*DIALOG**

Vorgesehen nur bei Eingabe über die Datensichtstation.

PERCON erwartet von der Datensichtstation Angaben über den weiteren Ablauf.

Folgende Angaben können gemacht werden:

H: Der Konvertierungsschritt wird fehlerhaft beendet.

I: Der Fehler wird ignoriert.

S: Der Block wird übergangen.

Wird dieser Operandenwert in einem Stapelauftrag angegeben, wird die Angabe ignoriert, eine Warnung ausgegeben und der Standardwert tritt in Kraft.

**LENGTH-ERROR = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 189, Anschluß für die Längenfehlerbehandlung).

## ASSIGN-OUTPUT-FILE

### Zuweisen einer Ausgabedatei

Diese Anweisung ist nur beim Kopieren von Dateien bzw. Editieren von Bändern zulässig. Sie weist eine Ausgabedatei zu. Für Fehlersituationen können ferner das Fehlverhalten und die anzuschließenden Benutzermodule bestimmt werden.

(Abschnitt 1 von 2)

#### ASSIGN-OUTPUT-FILE

**FILE** = \*DISK-FILE (...) / \*TAPE-FILE(...) / \*SYSLSST(...) / \*SYSOUT(...) / \*DISKETTE(...)

##### \*DISK-FILE (...)

**NAME** = \*STD / <full-filename 1..54>

,REMOVE-FILE-LINK = \*STD / \*NO / \*YES(...)

\*YES(...)

| **RELEASE-DEVICE** = \*YES / \*NO

,FILE-ATTRIBUTES = \*STD / \*INPUT-FILE

,OVERWRITE = \*YES / \*NO

,ALTERNATE-INDEX = \*NONE / \*FROM-INPUT-FILE(...)

\*FROM-INPUT-FILE(...)

| **LINK-NAME** = PCIN / <full-filename 1..8 without-gen>

##### \*TAPE-FILE(...)

**NAME** = \*STD / <full-filename 1..54>

,REMOVE-FILE-LINK = \*STD / \*NO / \*YES(...)

\*YES(...)

| **RELEASE-DEVICE** = \*YES / \*NO

| **UNLOAD-RELEASED-TAPE** = \*NO / \*YES

,FILE-ATTRIBUTES = \*STD / \*INPUT-FILE

,OVERWRITE = \*YES / \*NO

,END-POSITION = \*BEGIN-OF-TAPE / \*BEGIN-OF-FILE / \*END-OF-FILE / \*UNLOAD-TAPE

,CODE-TRANSLATION = \*NO / \*MODULE(...)

\*MODULE(...)

| **NAME** = <name 1..8>

Fortsetzung ➡

(Abschnitt 2 von 2)

```

, LABEL-EXIT = *NO / *MODULE(...)
    *MODULE(...)
        NAME = <name 1..8>
        , CONTROLLED-LABEL = *STD / list-poss(2000): *VOL-USER-LABEL /
            *HDR-USER-LABEL / *EOV-USER-LABEL / *END-USER-LABEL /
            *CLOSE-POSITION / *LABEL-ERROR
*SYSLST(...)
    RECORD-FORMAT = *UNDEFINED / *VARIABLE(...) / *FIXED(...)
        *VARIABLE(...)
            RECORD-SIZE = 32768 / <integer 4..32768>
        *FIXED(...)
            RECORD-SIZE = 32768 / <integer 1..32768>
*SYSOUT(...)
    RECORD-FORMAT = *UNDEFINED / *VARIABLE(...) / *FIXED(...)
        *VARIABLE(...)
            RECORD-SIZE = 32768 / <integer 4..32768>
        *FIXED(...)
            RECORD-SIZE = 32768 / <integer 1..32768>
*DISKETTE(...)
    VOLUME = *ANY / list-poss(10): <alphanum-name 1..6>
    , FILE-NAME = *NONE / <name 1..8>
    , RECORD-SIZE = 128 / <integer 1..2048>
    , RETENTION-PERIOD = 0 / <integer 0..32766>
, LINK-NAME = PCOUT / <full-filename 1..8 without-gen>
, OUTPUT-EXIT = *NO / *MODULE(...)
    *MODULE(...)
        NAME = <name 1..8>

```

**FILE =**

Mit FILE entscheidet man sich für ein Ausgabemedium.

Bei FILE=\*DISK-FILE oder \*TAPE-FILE können ergänzende Angaben über das SET-FILE-LINK-Kommando gemacht werden. Beim Zuweisen einer Ausgabedatei über ein SET-FILE-LINK-Kommando oder über eine ASSIGN-OUTPUT-FILE-Anweisung wird ein Eintrag in der Prozeßdateitabelle (TFT) unter dem Dateikettungsnamen erzeugt.

**FILE = \*DISK-FILE(...)**

Die Ausgabedatei liegt auf Platte.

**NAME = \*STD / <full-filename 1..54>**

Name der Ausgabedatei.

Er wird nicht benötigt, wenn der Dateiname bereits in einem SET-FILE-LINK-Kommando angegeben wurde, auf das sich die ASSIGN-OUTPUT-FILE-Anweisung über den Operanden LINK-NAME bezieht. Der Dateiname im SET-FILE-LINK-Kommando hat Vorrang vor dem hier angegebenen.

**NAME = \*STD**

Standardwert ist der zum LINK-NAME gehörende Dateiname, der im SET-FILE-LINK-Kommando festgelegt wurde. Wurde kein SET-FILE-LINK-Kommando abgesetzt, wird der durch den Operand LINK-NAME vereinbarte Name als Dateiname verwendet.

**REMOVE-FILE-LINK =**

Legt fest, ob und in welcher Weise der TFT-Eintrag nach der Dateibearbeitung gelöscht oder erhalten werden soll.

**REMOVE-FILE-LINK = \*STD**

PERCON löscht den TFT-Eintrag nur dann, wenn er über eine ASSIGN-OUTPUT-FILE-Anweisung von PERCON selbst angelegt wurde. Wurde die Datei mit einem SET-FILE-LINK-Kommando zugewiesen, bleibt der TFT-Eintrag erhalten.

**REMOVE-FILE-LINK = \*NO**

Der TFT-Eintrag bleibt erhalten, unabhängig davon, wie er angelegt wurde.

**REMOVE-FILE-LINK = \*YES(...)**

Der TFT-Eintrag wird gelöscht, unabhängig davon, wie er angelegt wurde.

**RELEASE-DEVICE = \*YES / \*NO**

Dient zur Verwaltung des Plattengeräts nach der Bearbeitung.

Nur für Privatplatten.

Mit \*YES wird das Plattengerät freigegeben, mit \*NO bleibt es dem Prozeß zugeordnet.

**FILE-ATTRIBUTES =**

Legt für eine neu anzulegende Ausgabedatei Dateimerkmale fest, die nicht über ein SET-FILE-LINK-Kommando spezifiziert wurden.

**FILE-ATTRIBUTES = \*STD**

PERCON nimmt für die Dateimerkmale folgende Standardwerte:

ACCESS-METHOD = \*ISAM

BLOCK-CONTROL-INFO = Defaultregelung

BUFFER-LENGTH = \*STD(SIZE = n)

mit n = 1 bei 2K-formatiertem Pubset

mit n = 2 bei 4K-formatiertem Pubset

RECORD-FORMAT = \*VARIABLE

KEY-POSITION = 5

KEY-LENGTH = 8

**FILE-ATTRIBUTES = \*INPUT-FILE**

Die Dateimerkmale ACCESS-METHOD, BLOCK-CONTROL-INFO, BUFFER-LENGTH, RECORD-FORMAT, RECORD-SIZE, KEY-LENGTH sowie KEY-POSITION werden von der ersten Eingabedatei übernommen.

Liegt die erste Eingabedatei auf SYSDTA oder Diskette, werden die unter \*STD beschriebenen Standardwerte eingesetzt, bei BUFFER-LENGTH wird jedoch immer SIZE=2 eingesetzt.

Weist die erste Eingabedatei variables Satzformat auf und wird für die Ausgabedatei eine Aufbereitung gefordert, so ist zu beachten, daß der unaufbereitete Ausgabesatz ein zusätzliches Satzlängelfeld im Datenteil erhält.

Falls die Eingabedatei eine ISAM-Datei ist und BLOCK-CONTROL-INFO=\*WITHIN-DATA-2K-BLOCK aufweist, wird sie mit BLOCK-CONTROL-INFO=\*WITHIN-DATA-BLOCK eingetragen. Dadurch kann die Datei, abhängig vom Pubset, auf dem sie angelegt wird, mit WITHIN-DATA-2K-BLOCK oder WITHIN-DATA-4K-BLOCK eingerichtet werden.

**OVERWRITE =**

Legt fest, wie bei einer bereits bestehenden ISAM-Ausgabedatei mit den Eingabesätzen verfahren werden soll, deren Schlüssel bereits in der Ausgabedatei vorhanden sind.

**OVERWRITE = \*YES**

Sätze mit gleichen Schlüsseln können verarbeitet werden.

Abhängig von der DUPLICATE-KEY-Angabe gilt:

Wenn DUPLICATE-KEY=\*NO angegeben ist, wird der Satz mit gleichem Schlüssel überschrieben. Sind mehrere Sätze mit diesem Schlüssel in der Ausgabedatei enthalten, so wird der erste Satz überschrieben.

Wenn DUPLICATE-KEY=\*YES, wird der Satz nach den Sätzen mit gleichem Schlüssel als letzter eingefügt.



**OVERWRITE = \*NO**

Sätze mit gleichem Schlüssel können nicht verarbeitet werden.

Nur Sätze mit neuen Schlüsseln können eingefügt werden. Diese Festlegung ist unabhängig von der DUPLICATE-KEY-Angabe. Tritt ein Eingabesatz mit gleichem Schlüssel auf, wird der Konvertierungsschritt mit der Meldung PER0098 abgebrochen. Dies gilt auch, wenn in der Eingabedatei mehrere Sätze mit gleichem Schlüssel vorliegen (der erste Satz wird eingefügt, der zweite führt zum Fehler).

**ALTERNATE-INDEX =**

Legt fest, ob für die Ausgabedatei die Sekundärschlüssel einer Eingabedatei übernommen werden sollen oder nicht.

**ALTERNATE-INDEX = \*NONE**

Für die Ausgabedatei sind keine Sekundärschlüssel zu übernehmen.

**ALTERNATE-INDEX = \*FROM-INPUT-FILE(...)**

Für die Ausgabedatei sind die Sekundärschlüssel einer Eingabedatei zu übernehmen.

**LINK-NAME = PCIN / <full-filename 1..8 without-gen>**

Dateikettungsname der Eingabedatei, deren Sekundärschlüssel übernommen werden sollen.

**FILE = \*TAPE-FILE(...)**

Die Ausgabe erfolgt in eine Banddatei.

**NAME = \*STD / <full-filename 1..54>**

Name der Ausgabedatei.

Er wird nicht benötigt, wenn der Dateiname bereits in einem SET-FILE-LINK-Kommando angegeben wurde, auf das sich die ASSIGN-OUTPUT-FILE-Anweisung über den Operanden LINK-NAME bezieht. Der Dateiname im SET-FILE-LINK-Kommando hat Vorrang vor dem hier angegebenen.

**NAME = \*STD**

Standardwert ist der zum LINK-NAME gehörende Dateiname, der im SET-FILE-LINK-Kommando festgelegt wurde. Wurde kein SET-FILE-LINK-Kommando abgesetzt, wird der durch den Operand LINK-NAME vereinbarte Name als Dateiname verwendet.

**REMOVE-FILE-LINK =**

Legt fest, ob und in welcher Weise der TFT-Eintrag nach der Dateibearbeitung gelöscht oder erhalten werden soll.

**REMOVE-FILE-LINK = \*STD**

PERCON löscht den TFT-Eintrag nur dann, wenn er über eine ASSIGN-OUTPUT-FILE-Anweisung von PERCON selbst angelegt wurde. Wurde die Datei mit einem SET-FILE-LINK-Kommando zugewiesen, bleibt der TFT-Eintrag erhalten.

**REMOVE-FILE-LINK = \*NO**

Der TFT-Eintrag bleibt erhalten, unabhängig davon, wie er angelegt wurde.

**REMOVE-FILE-LINK = \*YES(...)**

Der TFT-Eintrag wird gelöscht, unabhängig davon, wie er angelegt wurde.

**RELEASE-DEVICE = \*YES / \*NO**

Dient zur Verwaltung des Bandgeräts nach der Bearbeitung. Mit \*YES wird das Bandgerät freigegeben, mit \*NO bleibt es diesem Prozeß zugeordnet.

**UNLOAD-RELEASED-TAPE = \*NO / \*YES**

Dient zur Verwaltung des Bandes nach der Bearbeitung. Das Band bleibt geladen, wenn \*NO eingegeben wird. Das Band wird entladen, wenn \*YES eingegeben wird.

**FILE-ATTRIBUTES =**

Legt für eine neu anzulegende Ausgabedatei Dateimerkmale fest, die nicht über ein SET-FILE-LINK-Kommando spezifiziert wurden.

**FILE-ATTRIBUTES = \*STD**

PERCON nimmt für die Dateimerkmale folgende Standardwerte:

ACCESS-METHOD = \*SAM

BLOCK-CONTROL-INFO = \*PAMKEY

BUFFER-LENGTH = \*STD(SIZE = 1)

RECORD-FORMAT = \*VARIABLE

**FILE-ATTRIBUTES = \*INPUT-FILE**

Die Dateimerkmale ACCESS-METHOD, BLOCK-CONTROL-INFO, BUFFER-LENGTH, RECORD-FORMAT und RECORD-SIZE werden von der ersten Eingabedatei übernommen.

Liegt die erste Eingabedatei auf SYSDTA oder Diskette, werden die unter \*STD beschriebenen Standardwerte eingesetzt.

Weist die erste Eingabedatei variables Satzformat auf und wird für die Ausgabedatei eine Aufbereitung gefordert, so ist zu beachten, daß der unaufbereitete Ausgabesatz ein zusätzliches Satzlängengebiet im Datenteil erhält.

**OVERWRITE =**

Legt fest, ob eine bereits bestehende Ausgabedatei überschrieben oder nur fortgeschrieben werden kann.

**OVERWRITE = \*YES**

Die Ausgabedatei kann sequentiell neu geschrieben werden.

**OVERWRITE = \*NO**

Die Ausgabedatei kann nicht überschrieben, sondern nur erweitert werden.

**END-POSITION =**

Gibt an, wie das Band nach der Bearbeitung der Datei zu positionieren ist.

**END-POSITION = \*BEGIN-OF-TAPE**

Das Band wird zum Bandanfang zurückgespult.

**END-POSITION = \*BEGIN-OF-FILE**

Das Band wird auf den Anfang der gerade bearbeiteten Datei positioniert.

**END-POSITION = \*END-OF-FILE**

Die Position des Bandes wird nach der Bearbeitung nicht verändert.

**END-POSITION = \*UNLOAD-TAPE**

Das Band wird zurückgespult und entladen.

**CODE-TRANSLATION =**

Gibt an, welche Code-Umsetzung der Ausgabesätze erfolgen soll.

**CODE-TRANSLATION = \*NO**

Es erfolgt keine Code-Umsetzung.

**CODE-TRANSLATION = \*MODULE(...)**

Die Code-Umsetztabelle befindet sich in der über SET-TASKLIB zugewiesenen Modulbibliothek. Ihr Inhalt sind die 256 Sedezimalverschlüsselungen (X'00' bis X'FF') der Zeichen, in die die Ausgabezeichen umzuwandeln sind.

Bei der Umwandlung wird vom sedezimalen Wert des Ausgabezeichens ausgegangen. Dieser Wert wird zur Anfangsadresse der Umsetztabelle addiert; das Zeichen an der sich ergebenden Position der Umsetztabelle ersetzt das Ausgabezeichen.

**NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die Code-Umsetztabelle enthält.

**LABEL-EXIT =**

Beschreibt die Behandlung von Benutzerkennsätzen.

**LABEL-EXIT = \*NO**

Benutzerkennsätze werden nicht gesondert bearbeitet.

**LABEL-EXIT = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 182, Anschluß für die Kennsatzbearbeitung).

**CONTROLLED-LABEL =**

Der Benutzermodul wird bei den angegebenen Bearbeitungsabschnitten angesprungen.

**CONTROLLED-LABEL = \*STD**

Der Benutzermodul wird bei der Bearbeitung der vier folgenden Benutzer-Kennsätze angesprungen:

VOL-USER-LABEL, HDR-USER-LABEL, EOVS-USER-LABEL, END-USER-LABEL

**CONTROLLED-LABEL = \*VOL-USER-LABEL**

Der Benutzermodul wird bei der Bearbeitung der Benutzer-VOLUME-Kennsätze angesprungen.

**CONTROLLED-LABEL = \*HDR-USER-LABEL**

Der Benutzermodul wird bei der Bearbeitung der Benutzer-HEADER-Kennsätze angesprungen.

**CONTROLLED-LABEL = \*EOV-USER-LABEL**

Der Benutzermodul wird bei der Bearbeitung der Benutzer-END-OF-VOLUME-Kennsätze bei Bandwechsel angesprungen.

**CONTROLLED-LABEL = \*END-USER-LABEL**

Der Benutzermodul wird bei der Bearbeitung der Benutzer-END-Kennsätze angesprungen.

**CONTROLLED-LABEL = \*CLOSE-POSITION**

Der Benutzermodul wird zur Positionierung des Bandes bei der CLOSE-Bearbeitung angesprungen.

**CONTROLLED-LABEL = \*LABEL-ERROR**

Der Benutzermodul wird bei fehlerhaften Bandendekennsätzen angesprungen.

**FILE = \*SYSLSL(...)**

Die Ausgabe erfolgt in die Systemdatei SYSLSL gemäß einem vorgegebenen Standardformat, das über die SET-PAGE-LAYOUT-Anweisung abgeändert werden kann.

**RECORD-FORMAT =**

Bestimmt das Satzformat des unaufbereiteten Ausgabesatzes.

**RECORD-FORMAT = \*UNDEFINED**

Vereinbart Sätze undefinierter Länge.

**RECORD-FORMAT = \*VARIABLE(...)**

Es werden Sätze variabler Länge vereinbart.

**RECORD-SIZE = 32768 / <integer 4..32768>**

Maximale Länge der unaufbereiteten Ausgabesätze.

**RECORD-FORMAT = \*FIXED(...)**

Es werden Sätze fester Länge vereinbart.

**RECORD-SIZE = 32768 / <integer 1..32768>**

Länge der unaufbereiteten Ausgabesätze.

**FILE = \*SYSOUT(...)**

Die Ausgabe erfolgt in die Systemdatei SYSOUT gemäß einem vorgegebenen Standardformat, das über die SET-PAGE-LAYOUT-Anweisung abgeändert werden kann.

**RECORD-FORMAT =**

Bestimmt das Satzformat des unaufbereiteten Ausgabesatzes.

**RECORD-FORMAT = \*UNDEFINED**

Vereinbart Sätze undefinierter Länge.

**RECORD-FORMAT = \*VARIABLE(...)**

Es werden Sätze variabler Länge vereinbart.

**RECORD-SIZE = 32768 / <integer 4..32768>**

Maximale Länge der unaufbereiteten Ausgabesätze.

**RECORD-FORMAT = \*FIXED(...)**

Es werden Sätze fester Länge vereinbart.

**RECORD-SIZE = 32768 / <integer 1..32768>**

Länge der unaufbereiteten Ausgabesätze.

**FILE = \*DISKETTE(...)**

Die Ausgabe erfolgt in eine Diskettendatei.

**VOLUME = \*ANY / list-poss: <alphanum-name 1..6>**

Archivnummer(n) der Diskette(n), auf die geschrieben wird. Wird \*ANY angegeben, erfolgt die Ausgabe auf eine freie Diskette.

**FILE-NAME = \*NONE / <name 1..8>**

Dateiname der auszugebenden Datei. Wird \*NONE angegeben, wird der im Operand LINK-NAME angegebene Name verwendet.

**RECORD-SIZE = 128 / <integer 1..2048>**

Gibt die Länge der zu bearbeitenden Sätze an.

**RETENTION-PERIOD = 0 / <integer 0..32766>**

Zeitraum in Tagen, in dem die Datei nicht verändert oder gelöscht werden kann.

**LINK-NAME = PCOUT / <full-filename 1..8 without-gen>**

PERCON-Dateikettungsname zur Bezugnahme in Kennwörtern und im SET-FILE-LINK-Kommando.

**OUTPUT-EXIT =**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die Ausgabesätze unmittelbar vor der Ausgabe bearbeitet. Er darf nur beim Kopieren von Dateien angegeben werden.

**OUTPUT-EXIT = \*NO**

Es erfolgt keine Bearbeitung durch einen Benutzermodul.

**OUTPUT-EXIT = \*MODULE(...)****NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die weitere Bearbeitung übernimmt (siehe Seite 186, Anschluß für die Ausgabe).

## ASSIGN-OUTPUT-TAPE

### Zuweisen eines Ausgabebandes

Diese Anweisung ist nur beim Duplizieren von Bändern zulässig. Sie weist das Ausgabeband/die Ausgabebänder zu. Für Fehlersituationen können ferner das Fehlerverhalten und die anzuschließenden Benutzermodule bestimmt werden.

#### ASSIGN-OUTPUT-TAPE

```

VOLUME = list-poss(2000): <alphanum-name 1..6>
, LINK-NAME = PCOUT / <full-filename 1..8 without-gen>
, END-POSITION = *BEGIN-OF-TAPE / *UNLOAD-TAPE
, CODE = *EBCDIC / *ISO7 / *OWN(...)
    *OWN(...)
        | NAME = <name 1..8>
, DEVICE-TYPE = *TAPE / <structured-name 1..8>
, RENAME-VOLUME = *NO / list-poss: <alphanum-name 1..6>

```

**VOLUME = list-poss: <alphanum-name 1..6>**

Archivnummer(n) des Ausgabebandes/der Ausgabebänder.

**LINK-NAME = PCOUT / <full-filename 1..8 without-gen>**

Interner PERCON-Dateikettungsname.

**END-POSITION =**

Position des Bandes nach der Bearbeitung.

**END-POSITION = \*BEGIN-OF-TAPE**

Das Band wird zur Anfangsmarke zurückgespult.

**END-POSITION = \***UNLOAD-TAPE****

Das Band wird zurückgespult und entladen.

**CODE =**

Code, in dem das Band zu beschreiben ist. Bei der Code-Umsetzung ist zu beachten, daß alle Daten (auch Kennsätze) des Bandes umcodiert werden.

**CODE = \*EBCDIC**

Das Band ist im EBCDI-Code zu beschreiben und wird ohne Code-Umsetzung bearbeitet.

**CODE = \*ISO7**

Das Band ist im ISO7-Code zu beschreiben. Die Ausgabedaten werden in den ISO7-Code umgesetzt.

**CODE = \*OWN (...)**

Der Bandinhalt soll über eine benutzereigene Tabelle (Code-Umsetztabelle) umgesetzt werden. Sie befindet sich in der über SET-TASKLIB zugewiesenen Modulbibliothek. Ihr Inhalt sind die 256 Sedezimalverschlüsselungen (X'00' bis X'FF') der Zeichen, in die die Ausgabezeichen umzuwandeln sind.

Bei der Umwandlung wird vom sedezimalen Wert des Ausgabezeichens ausgegangen. Dieser Wert wird zur Anfangsadresse der Umsetztabelle addiert; das Zeichen an der sich ergebenden Position der Umsetztabelle ersetzt das Ausgabezeichen.

**NAME = <name 1..8>**

Name des Benutzermoduls oder des Einsprungpunkts im Benutzermodul, der die Code-Umsetztabelle enthält.

**DEVICE-TYPE =**

Bandgerätetyp

**DEVICE-TYPE = \*TAPE**

Bandgerät mit der größten Schreiddichte.

**DEVICE-TYPE = <structured-name 1..8>**

Explizite Angabe des Bandgerätetyps.

**RENAME-VOLUME =**

Nur für den Systemverwalter.

Neue Archivnummer(n) des Ausgabebandes/der Ausgabebänder. Die Angabe des Parameters ist nur sinnvoll, wenn auf dem Eingabeband oder MF/MV-Set Standardkennsätze stehen.

**RENAME-VOLUME = \*NO**

Die Archivnummern der Bänder, die mit dem Operand VOLUME angegeben wurden, bleiben erhalten.

**RENAME-VOLUME = list-poss: <alphanum-name 1..6>**

Legt die neue(n) Archivnummer(n) eines MF/MV-Sets fest.



## CHANGE-INPUT-TAPEPOSITION

### Positionieren des Eingabebandes

Diese Anweisung ist nur beim Editieren eines Bandes zulässig. Mit ihr kann das Eingabeband vor der Aufbereitung von Ausgabedaten positioniert werden. Das Eingabeband muß über die ASSIGN-INPUT-TAPE-Anweisung zugewiesen werden.

Die CHANGE-INPUT-TAPEPOSITION-Anweisung darf nur nach der ASSIGN-INPUT-TAPE-Anweisung gegeben werden, sie kann aber vor oder nach der ASSIGN-OUTPUT-FILE-Anweisung stehen.

#### CHANGE-INPUT-TAPEPOSITION

**DIRECTION** = \*BACKWARD (...) / \*FORWARD(...)

\*BACKWARD (...)

**DESTINATION** = \*BEGIN-OF-TAPE / \*DOUBLE-TAPE-MARK / \*BLOCKS(...) / \*TAPE-MARKS(...)

\*BLOCKS(...)

**BLOCKS** = <integer 1..999999>

\*TAPE-MARKS(...)

**TAPE-MARKS** = <integer 1..999999>

\*FORWARD(...)

**DESTINATION** = \*DOUBLE-TAPE-MARK / \*PAST-END-OF-TAPE / \*END-OF-TAPE / \*BLOCKS(...) / \*TAPE-MARKS(...)

\*BLOCKS(...)

**BLOCKS** = <integer 1..999999>

\*TAPE-MARKS(...)

**TAPE-MARKS** = <integer 1..999999>

#### **DIRECTION =**

Richtung, in die das Band positioniert werden soll.

#### **DIRECTION = \*BACKWARD (...)**

Das Eingabeband soll rückwärts - Richtung Bandanfang - positioniert werden.

#### **DESTINATION =**

Gewünschte Position des Bandes

#### **DESTINATION = \*BEGIN-OF-TAPE**

Das Band wird auf den Bandanfang positioniert.

**DESTINATION = \*DOUBLE-TAPE-MARK**

Das Band wird auf die nächste doppelte Bandmarke positioniert.

**DESTINATION = \*BLOCKS (...)**

**BLOCKS = <integer 1..999999>**

Anzahl der Blöcke, um die zurückpositioniert werden soll.

**DESTINATION = \*TAPE-MARKS (...)**

**TAPE-MARKS = <integer 1..999999>**

Anzahl der Abschnittsmarken, um die zurückpositioniert werden soll.

**DIRECTION = \*FORWARD(...)**

Das Eingabeband soll vorwärts - Richtung Bandende - positioniert werden.

**DESTINATION =**

Gewünschte Position des Bandes.

**DESTINATION = \*DOUBLE-TAPE-MARK**

Das Band wird auf die nächste doppelte Bandmarke positioniert.

**DESTINATION = \*PAST-END-OF-TAPE**

Nur für den Systemverwalter.

Das Band wird hinter das logische Bandende positioniert. Befindet sich keine doppelte Bandmarke mehr auf dem Band, wird das Band ausgespult.

Bei Magnetbandkassetten nicht möglich.

**DESTINATION = \*END-OF-TAPE**

Das Band wird auf das logische Bandende positioniert.

**DESTINATION = \*BLOCKS (...)**

**BLOCKS = <integer 1..999999>**

Anzahl der Blöcke, um die vorpositioniert werden soll.

**DESTINATION = \*TAPE-MARKS (...)**

**TAPE-MARKS = <integer 1..999999>**

Anzahl der Abschnittsmarken, um die vorpositioniert werden soll.

**END****Beenden des PERCON-Laufs, Starten eines Konvertierungsschrittes**

Die END-Anweisung beendet die Anweisungseingabe an PERCON.

Beim Kopieren von Dateien und Duplizieren von Bändern dient die END-Anweisung zum Starten des Übertragungsvorganges und zum Beenden von PERCON.

Beim Editieren von Bändern sind zum Starten von Übertragungsvorgängen zusätzliche Angaben über die START-TAPE-PROCESSING-Anweisung erforderlich; die END-Anweisung dient in diesem Modus nur zum Beenden von PERCON.

Falls nur die END-Anweisung angegeben wurde, wird ein Standardumsetzlauf ausgeführt. Sind keine SET-FILE-LINK-Kommandos vorausgegangen, d.h. keine Ein- bzw. Ausgabedateien wurden definiert, verwendet PERCON die Dateien mit den Namen PCIN bzw. PCOUT.

END

Die END-Anweisung hat keine Operanden.

## EXECUTE-SYSTEM-COMMAND

### Absetzen eines Systemkommandos

Diese Anweisung ermöglicht dem Benutzer, ein BS2000-Systemkommando abzusetzen. Nachdem das Systemkommando ausgeführt wurde, können weitere PERCON-Anweisungen eingegeben werden.

<b>EXECUTE-SYSTEM-COMMAND</b>
<b>COMMAND</b> = <command-rest 1..1800>

#### **COMMAND = <command-rest 1..1800>**

Dient zur Eingabe eines BS2000-Kommandos.

Ab BS2000/OSD-BC V2.0 wird diese Anweisung im Dialog nicht mehr unterstützt. Als Ersatz ist die SDF-Standardanweisung EXECUTE-SYSTEM-CMD zu verwenden.

In Prozeduren und im Stapelbetrieb wird die Anweisung weiterhin unterstützt (Kompatibilität).

## MODIFY-PERCON-OPTIONS

### Steuern der Meldungsausgabe

Diese Anweisung steuert den Umfang der Meldungen, die PERCON nach SYSOUT und SYSLST ausgibt.

#### MODIFY-PERCON-OPTIONS

**SYSOUT-LOGGING** = \*UNCHANGED / \*ALL / \*ERRORS / \*NO

**,SYSLST-LOGGING** = \*UNCHANGED / \*ALL / \*ERRORS / \*NO

#### **SYSOUT-LOGGING =**

Legt den Umfang der Meldungen fest, die nach SYSOUT ausgegeben werden.  
Standardwerte:

Dialog-, Batchbetrieb: SYSOUT-LOGGING = \*ALL

Unterprogramm: SYSOUT-LOGGING = \*NO

#### **SYSOUT-LOGGING = \*UNCHANGED**

Die Einstellung von SYSOUT-LOGGING wird nicht verändert.

#### **SYSOUT-LOGGING = \*ALL**

PERCON gibt alle Meldungen nach SYSOUT aus.

#### **SYSOUT-LOGGING = \*ERRORS**

PERCON gibt bis auf den Abschlußbericht über die Anzahl der bearbeiteten Sätze/Blöcke alle Meldungen nach SYSOUT aus. (Im „Abschlußbericht“ werden für jede Datei bzw. für jeden Datenträger des Konvertierungsschritts die Meldungen PER0029 bzw. PER0030 ausgegeben.)

#### **SYSOUT-LOGGING = \*NO**

PERCON gibt keine Meldungen nach SYSOUT aus. Bei logischen Fehlern wird kein Fehlerdialog geführt.

#### **SYSLST-LOGGING =**

Legt den Umfang des Protokolls fest, das nach SYSLST ausgegeben wird.  
Standardwerte:

Dialogbetrieb, Unterprogramm: SYSLST-LOGGING = \*NO

Batchbetrieb: SYSLST-LOGGING = \*ALL

#### **SYSLST-LOGGING = \*UNCHANGED**

Die Einstellung von SYSLST-LOGGING wird nicht verändert.

**SYSLST-LOGGING = \*ALL**

PERCON gibt bis auf die Startmeldung alle Meldungen nach SYSLST aus.

**SYSLST-LOGGING = \*ERRORS**

PERCON gibt bis auf die Startmeldung und den Abschlußbericht über die Anzahl der bearbeiteten Sätze/Blöcke alle Meldungen nach SYSLST aus.

**SYSLST-LOGGING = \*NO**

PERCON gibt keine Meldungen nach SYSLST aus.

## RESET-INPUT

### Rücksetzen von Anweisungen

Mit dieser Anweisung werden alle Angaben des laufenden Konvertierungsschrittes zurückgesetzt. Danach können neue Anweisungen für den folgenden Konvertierungsschritt gegeben werden.

RESET-INPUT

Die RESET-INPUT-Anweisung hat keine Operanden.

## SELECT-INPUT-RECORDS

### Festlegen von Auswahlbedingungen

Beim Kopieren von Dateien werden Sätze einer Eingabedatei nach bestimmten Kriterien ausgewählt und in eine oder mehrere Ausgabedateien übernommen. Beim Editieren eines Bandes werden Blöcke des Eingabebandes ausgewählt. Die Positionsangaben beziehen sich dann nicht auf den Satzanfang, sondern auf den Blockanfang.

Eingabesätze, für die keines der Auswahlkriterien zutrifft, können in spezielle Ausgabedateien übernommen werden. Diese Ausgabedateien, im folgenden als „Restdateien“ bezeichnet, werden über das Schlüsselwort REMAINING-RECORDS des Operanden CONDITION vereinbart.

Es ist möglich, für jede Ausgabedatei mit der Anweisung SELECT-INPUT-RECORDS ein eigenes Auswahlkriterium festzulegen.

Beziehen sich in einem Konvertierungsschritt mehrere SELECT-INPUT-RECORDS-Anweisungen auf den gleichen Dateikettungsamen, wird für diesen Dateikettungsamen nur die zuletzt gegebene Anweisung ausgeführt.

#### SELECT-INPUT-RECORDS

**OUTPUT-LINK-NAME = \*STD** / list-poss(2000): <full-filename 1..8 without-gen>  
**,CONDITION = \*REMAINING-RECORDS** / <text 7..1800 with-low>

#### OUTPUT-LINK-NAME =

Dateikettungsname der Ausgabedateien, auf die sich diese Anweisung beziehen soll.

#### OUTPUT-LINK-NAME = \*STD

Diese Anweisung bezieht sich auf alle bisher angegebenen Ausgabedateien.

#### OUTPUT-LINK-NAME = list-poss(2000): <full-filename 1..8 without-gen>

Soll sich die Anweisung nur auf einige Ausgabedateien beziehen, müssen die Dateikettungsamen dieser Dateien angegeben werden.

#### CONDITION =

Über diesen Operanden werden entweder die Auswahlkriterien (Bedingungen) festgelegt oder „Restdateien“ vereinbart.

#### CONDITION = \*REMAINING-RECORDS

Die im Operanden OUTPUT-LINK-NAME angegebenen Ausgabedateien werden als „Restdateien“ verwendet. Das heißt, alle Eingabesätze, für die keines der Auswahlkriterien zutrifft, werden in die „Restdateien“ ausgegeben.

In einem Konvertierungsschritt können mehrere Restdateien vereinbart werden.



*Hinweis*

Die Anweisungen SET-RECORD-MAPPING, SET-GROUP-ATTRIBUTES und SET-PAGE-LAYOUT können ohne Einschränkungen für „Restdateien“ verwendet werden, ebenso die Benutzeranschlüsse.

Wird in einem Benutzeranschluß für Eingabe entschieden, einen Satz nicht zu übernehmen, so gilt das für alle Ausgabedateien einschließlich der Restdateien.

Genügt ein Satz den Selektierkriterien und wird dann im Benutzeranschluß für Ausgabe entschieden, ihn nicht zu übernehmen, gilt er dennoch als ausgewählt und wird in keine Restdatei ausgegeben.

Bei der Anweisung START-TAPE-PROCESSING wird durch den Operanden OUTPUT-LINK-NAME festgelegt, für welche der vereinbarten Ausgabedateien die Anweisung wirksam werden soll. D.h. bei jedem Aufruf dieser Anweisung können andere Ausgabedateien angesprochen werden. Die Angabe \*STD gilt für alle zuvor angegebenen Ausgabedateien mit Ausnahme der Restdateien. Werden die Dateikettungsamen der Ausgabedateien explizit bei diesem Operanden angegeben, so ist die Verwendung des LINK-Namens einer Restdatei nicht zulässig und wird als Fehler zurückgewiesen (Meldung PER0062). Beim Abarbeiten einer Anweisung START-TAPE-PROCESSING wird nur dann ein Block in die Restdateien ausgegeben, wenn für keine der im Operanden OUTPUT-LINK-NAME angesprochenen Ausgabedateien eine Ausgabe möglich war, d.h. für keine dieser Ausgabedateien erfüllte der Block die Selektierkriterien.

Die Vereinbarung einer Restdatei ist nur sinnvoll, wenn für alle „normalen“ Ausgabedateien eine Anweisung SELECT-INPUT-RECORDS mit Angabe einer Bedingung wirksam ist, da sonst keine Restsätze entstehen können.

Ein Beispiel zur sinnvollen Verwendung einer Restdatei ist auf Seite 203 gegeben.

**CONDITION = <text 7..1800 with-low>**

Über diesen Operanden wird die Bedingung angegeben, unter der ein Eingabesatz für die Ausgabe ausgewählt wird. Ein Satz wird übernommen, wenn die Bedingung erfüllt wird, d.h. den Wert 'wahr' annimmt.

Die Bedingung kann aus einem oder mehreren durch AND oder OR logisch verknüpften Vergleichen bestehen. Die AND-Verknüpfung hat Vorrang vor der OR-Verknüpfung. Diese Vorrangsteuerung kann durch Klammern geändert werden. Dadurch ergibt sich eine bessere Lesbarkeit und eine kürzere Programmlaufzeit, da einzelne Vergleiche nicht mehrfach abgearbeitet werden.

Die Bedingung wird von links nach rechts abgearbeitet.

Durch gezielte Reihung der Vergleiche (insbesondere Vergleiche, die meist dasselbe Ergebnis liefern) wird die Laufzeit reduziert:

z.B.:           a OR b OR c  
                  Der am häufigsten erfüllte Vergleich soll links stehen.  
  
                  a AND b AND c  
                  Der am häufigsten erfüllte Vergleich soll rechts stehen.

Einschränkung der Klammerschachtelung: 1800 byte für CONDITION.

Mehrere durch AND verknüpfte Vergleiche (=Relation) ergeben den Wert 'wahr', wenn alle Vergleiche 'wahr' sind.

Mehrere durch OR verknüpfte Relationen oder Vergleiche, die keiner Relation angehören, ergeben den Wert 'wahr', wenn mindestens eine Relation oder ein Vergleich 'wahr' ist.

Bei zwei Vergleichen gilt:

1. Vergleich	Verknüpfung	2. Vergleich	Ergebnis
w	AND	w	w
w	AND	f	f
f	AND	w	f
f	AND	f	f
w	OR	w	w
w	OR	f	w
f	OR	w	w
f	OR	f	f

f = falsch

w = wahr

Die Syntax, mit der die Bedingung im Operanden CONDITION angegeben wird, lautet folgendermaßen:

```

bedingung  : =  (vergleich [AND/OR vergleich] ...)
vergleich  : =  (position[,länge]) EQ/NE N[UMERIC]/AL[PHA]
              oder (position[,länge]) AS[CENDING]/D[ESCENDING]/M[ODULE](modname)
              oder (position[,länge[,C]]) op <c-string 1..256>/<x-string 1..512>
              oder (position,länge,Z/P) op Z'zahl'/P'zahl'
              oder RECCNT(dateikettungsname) op Z'zahl'/P'zahl'
              oder BYTCNT(dateikettungsname) op Z'zahl'/P'zahl'
              oder BLKCNT(dateikettungsname) op Z'zahl'/P'zahl'
              oder RECLN(dateikettungsname) op Z'zahl'/P'zahl'
              oder bedingung
position    : =  <integer 1..32768>
länge       : =  <integer 1..256>
              oder RECLN (Satzlänge), wobei 1 <= RECLN <= 256 gelten muß
              oder 1 <= RECLN-reduction <= 256
              Bei Verletzung der Bedingung wird der Konvertierungsschritt
              mit der Meldung PER0042 abgebrochen.
op          : =  EQ/NE/GT/LT/GE/LE

```

Die Angaben für Vergleichsoperatoren sind gleichwertig und haben folgende Bedeutung:

Sonderzeichen	Buchstabenkombinationen	Bedeutung
=	EQ	gleich
>	GT	größer
<	LT	kleiner
>=	GE	größer gleich
<=	LE	kleiner gleich
<>	NE	ungleich

## Satzauswahl

Die Auswahl eines Satzes erfolgt durch

- Überprüfen eines Feldes auf numerisches oder alphabetisches Zeichenformat
- Vergleich auf aufsteigend oder absteigend sortierte Felder
- Überprüfen eines Feldes auf zulässige Zeichen anhand einer Benutzertabelle
- Vergleich eines Feldes mit einem Literal (Konstante)
- Vergleich eines Kennwortes mit einem Literal (Konstante).

zu a) *Überprüfen eines Feldes auf das Zeichenformat:*

Das bezeichnete Feld wird auf das gewünschte Zeichenformat geprüft.

NUMERIC	PERCON prüft, ob das angegebene Feld nur Ziffern enthält.
ALPHA	PERCON prüft, ob das angegebene Feld ausschließlich Großbuchstaben, Kleinbuchstaben, Umlaute und Leerzeichen enthält.

**Beispiele**

```
COND=((5,3) EQ NUMERIC)
```

```
COND=((20,4) EQ NUMERIC AND (25,5) EQ ALPHA)
```

zu b) *Überprüfen eines Feldes, ob auf- bzw. absteigende Reihenfolge besteht:*

PERCON prüft das angegebene Feld auf aufsteigende bzw. absteigende Reihenfolge, d.h. ob das Feld einen höheren bzw. niedrigeren Wert hat als das Feld des vorhergehenden Satzes. Wird ein Satz aufgrund der SELECT-INPUT-RECORDS-Bedingung nicht in die Ausgabedatei übernommen, wird auch das Feld dieses Satzes nicht als Vergleichsfeld zum Vergleich mit dem Feld des nächsten Satzes herangezogen.

ASCENDING PERCON überprüft auf aufsteigende Reihenfolge

DESCENDING PERCON überprüft auf absteigende Reihenfolge

zu c) *Überprüfen eines Feldes auf zulässige Zeichen anhand einer Benutzertabelle:*

MODULE	PERCON prüft, ob das angegebene Feld nur die Zeichen enthält, die vom Benutzer zugelassen sind.
modname	Gibt den Namen eines Benutzermoduls (1. CSECT-Name) oder den Namen (ENTRY-Name) eines Bereichs in einem Benutzermodul an, der aus 256 Zeichen X'00' bzw. X'FF' besteht. PERCON addiert den Wert jedes Zeichens aus dem Feld auf die Anfangsadresse des Benutzermoduls bzw. des Benutzerbereichs auf. Steht an der sich ergebenden Position X'00', ist das Zeichen zugelassen, steht an der Stelle X'FF', ist das Zeichen nicht zugelassen. Entspricht ein Byte des Feldes nicht der Bedingung, ist der Vergleich nicht erfüllt.

zu d) *Vergleich eines Feldes mit einem Literal:*

Das Feld wird mit dem angegebenen Literal verglichen. Es ist zu beachten, daß entsprechend dem Format und der Länge verglichen wird. Beachte die Tabelle am Ende des Abschnittes.

zu e) *Vergleich eines Kennwortes mit einem Literal:*

Der aktuelle Wert von Kennwort (siehe Seite 73) wird mit dem angegebenen Literal verglichen.

Es ist zu beachten, daß entsprechend dem Format und der Länge dieser Kennwörter verglichen wird. Beachte die Tabelle am Ende des Abschnitts.

**Beispiel**

Es sollen zwei Eingabedateien (IN1 und IN2) in eine Ausgabedatei konvertiert werden. Von der ersten Eingabedatei sollen alle Sätze und von der zweiten Eingabedatei alle Sätze ab dem fünften übertragen werden. Die Bedingung hierfür lautet:

```
COND=(RECCNT(IN1) GT Z'0' AND RECCNT(IN2) EQ Z'0'
      OR RECCNT(IN2) GE Z'5')
```

**Beispiele zur Klammerschachtelung:**

Im ersten Beispiel werden zwei Felder verglichen.

Wenn im ersten Feld eine entpackte Zahl > 100 **oder** < 50 steht **und** zugleich im zweiten Feld die Zeichenkette '1.1' **oder** '2.2', dann wird der Satz übernommen.

```
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=02,-
//          CONDITION=(-
//      (    (10,5,Z) GT Z'100' -
//      OR    (10,5,Z) LT Z'50' -
//      ) -
// AND (    (20,3,C) EQ C'1.1' -
//      OR    (20,3,C) EQ C'2.2' -
//      ) )
```

Im zweiten Beispiel werden die Möglichkeiten der Klammerschachtelung aufgezeigt. Die Vergleiche sind von V1 bis V12 durchnummeriert.

Ein Satz wird nur dann ausgegeben, wenn:

- (1) V1, V2 **und** V3 wahr sind **und**  
mindestens einer der Vergleiche V4, V5 oder V6 wahr ist **und**  
V7 wahr ist.
- (2) V1, V2 **und** V3 wahr sind **und**  
mindestens einer der Vergleiche V4, V5 oder V6 wahr ist **und**  
V8 und V9 wahr sind **und**  
mindestens einer der Vergleiche V10, V11 **oder** V12 wahr ist.

```
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=03,-
//          CONDITION=( -
//          RECCNT(IN1) LE Z'5000' - V1
// AND      RECCNT(03) LT Z'300' - V2
// AND      (28,5) EQ ASCENDING - V3
// AND (    (6,1,C) EQ 'P' - V4
//          OR (6,2,C) EQ 'ST' - V5
//          OR (6,3,C) EQ 'MOD' - V6
//          ) -
// AND (    (20,5) EQ NUMERIC - V7
//          OR RECLLEN(IN1) GT Z'120' - V8
//          AND (35,2) EQ MODULE(ABISK) - V9
//          AND ( (37,3,P) LT Z'230' - V10
//                OR (37,3,P) EQ Z'241' - V11
//                OR (37,3,P) GT Z'260' - V12
//          ) -
//          )
```

Konvertierungsregeln bei ungleichem Zeichenformat und verschiedenen Längen der Felder

Im CONDITION-Operanden werden Operanden spezifiziert, die miteinander zu vergleichen sind. Haben zwei zu vergleichende Operanden verschiedenes Zeichenformat oder verschiedene Länge, ist folgendes zu beachten:

literal		C,X	Z,P	Bemerkung
feld	kennwort			
C		+	-	Bei unterschiedlichen Längen von feld und literal wird für den Vergleich das kürzere Feld rechts mit Leerzeichen aufgefüllt. Liegt feld ganz oder teilweise außerhalb eines Eingabesatzes variabler Länge, gilt bei der Abfrage '<>' der Vergleich als erfüllt, bei allen anderen Vergleichsoperatoren als nicht erfüllt. Der Vergleich erfolgt entsprechend der EBCDIC-Verschlüsselung.
P,Z	RECCNT, BYTCNT, BLKCNT, RECLLEN	-	+	Bei unterschiedlichen Längen von feld, literal und kennwort wird das kürzere Feld links mit 0 aufgefüllt. Wird Format P mit Format Z oder umgekehrt verglichen, wird die entpackte Zahl für den Vergleich gepackt. Ist der aktuelle Wert des Feldes nicht vom spezifizierten Format (kein numerischer Wert), gilt bei Abfrage auf '<>' der Vergleich als erfüllt, bei allen übrigen Vergleichsoperatoren als nicht erfüllt. Es erfolgt ein numerischer Vergleich unter Beachtung des Vorzeichens.

- + zulässiger Vergleich
- unzulässiger Vergleich

## SET-GROUP-ATTRIBUTES

### Festlegen von Gruppenwechselbedingungen

Diese Anweisung ist nur beim Kopieren von Dateien zulässig. Sie dient zum Bilden von Gruppen durch Zusammenfassung von Eingabesätzen mit gleichem Gruppenbegriff. Die Gruppenbildung bewirkt implizit eine Formataufbereitung. Es können Gruppenvorlauf- und Gruppennachlaufbenutzerzeilen gestaltet werden. Benutzerzeilen sind Zeilen, in die ein vom Benutzer vorgegebener Text geschrieben wird. Dieser Text kann aus Feldern des Eingabesatzes, aus Kennwörtern und Literalen bestehen. Je nachdem ob diese Zeilen vor oder nach einer Gruppe stehen, werden sie Gruppenvorlauf- bzw. Gruppennachlaufbenutzerzeilen genannt.

Beziehen sich mehrere SET-GROUP-ATTRIBUTES-Anweisungen mit gleicher Gruppenstufe auf den gleichen Dateikettungsnamen, wird in der angegebenen Gruppenstufe für diesen Dateikettungsnamen nur die zuletzt gegebene Anweisung ausgeführt.

(Abschnitt 1 von 6)

#### SET-GROUP-ATTRIBUTES

```

OUTPUT-LINK-NAME = *STD / list-poss(2000): <full-filename 1..8 without-gen>
, GROUP-LEVEL = 1 / <integer 1..8>
, GROUP-CONTROL = *ALL / list-poss(2000): *FIELD(...)
    *FIELD(...)
        | POSITION = <integer 1..32767>
        | LENGTH = 1 / <integer 1..256>
, GROUP-HEADER = *NONE / list-poss(2000): *FIELD(...) / *RECORD-LENGTH(...) / *BYTE-COUNTER(...) /
    *RECORD-COUNTER(...) / *PAGE-COUNTER(...) / *DATE(...) / *TIME(...) /
    <c-string 1..204 with-low>(...) / <x-string 1..408>(...) /
    <integer -2147483648..2147483647>(...) / *NEW-PAGE / *SPACING(...) / *MODULE(...)
*FIELD(...)
    | INPUT-POSITION = <integer 1..32768>
    | INPUT-LENGTH = 1 / <integer 1..204>
    | INPUT-FORMAT = *CHARACTER / *ZONED-DECIMAL / *PACKED-DECIMAL
    | OUTPUT-POSITION = <integer 1..204>
    | OUTPUT-LENGTH = *STD / <integer 1..204>
    | OUTPUT-FORMAT = *STD / *CHARACTER / *HEXADECIMAL / *BINARY /
        <c-string 2..204 with-low> / *ZONED-DECIMAL / *SIGNED-DECIMAL / *DECIMAL

```

Fortsetzung ➡

(Abschnitt 2 von 6)

**\*RECORD-LENGTH(...)****LINK-NAME** = <full-filename 1..8 without-gen>**,OUTPUT-POSITION** = <integer 1..203>**,OUTPUT-LENGTH** = **\*STD** / <integer 1..204>**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..204 with-low> / **\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL****\*BYTE-COUNTER(...)****LINK-NAME** = <full-filename 1..8 without-gen>**,OUTPUT-POSITION** = <integer 1..203>**,OUTPUT-LENGTH** = **\*STD** / <integer 1..204>**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..204 with-low> / **\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL****\*RECORD-COUNTER(...)****LINK-NAME** = <full-filename 1..8 without-gen>**,OUTPUT-POSITION** = <integer 1..203>**,OUTPUT-LENGTH** = **\*STD** / <integer 1..204>**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..204 with-low> / **\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL****\*PAGE-COUNTER(...)****LINK-NAME** = <full-filename 1..8 without-gen>**,OUTPUT-POSITION** = <integer 1..203>**,OUTPUT-LENGTH** = **\*STD** / <integer 1..204>**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..204 with-low> / **\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL****\*DATE(...)****OUTPUT-POSITION** = <integer 1..197>**,OUTPUT-LENGTH** = **\*STD** / <integer 8..204>**,OUTPUT-FORMAT** = **\*CHARACTER** / **\*HEXADECIMAL** / **\*BINARY****,CENTURY** = **\*NO** / **\*YES**

Fortsetzung ➡



(Abschnitt 3 von 6)

```
*TIME(...)
|   OUTPUT-POSITION = <integer 1..197>
|   ,OUTPUT-LENGTH = *STD / <integer 8..204>
|   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY
<c-string with-low>(...)
|   OUTPUT-POSITION = <integer 1..204>
|   ,OUTPUT-LENGTH = *STD / <integer 1..204>
|   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY
<x-string>(...)
|   OUTPUT-POSITION = <integer 1..204>
|   ,OUTPUT-LENGTH = *STD / <integer 1..204>
|   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY
<integer>(...)
|   OUTPUT-POSITION = <integer 1..204>
|   ,OUTPUT-LENGTH = *STD / <integer 1..204>
|   ,OUTPUT-FORMAT = *ZONED-DECIMAL / <c-string 2..204 with-low> / *SIGNED-DECIMAL /
|   *DECIMAL
*SPACING(...)
|   LINES = 1 / <integer 1..15>
*MODULE(...)
|   NAME = <name 1..8>
```

Fortsetzung ➔

(Abschnitt 4 von 6)

```
,GROUP-TRAILER = *NONE / list-poss(2000): *FIELD(...) / *SUM-FIELD(...) / *GROUP-COUNTER(...) /
  *RECORD-LENGTH(...) / *BYTE-COUNTER(...) / *RECORD-COUNTER(...) / *PAGE-COUNTER(...) /
  *DATE(...) / *TIME(...) / <c-string 1..204 with-low>(...) / <x-string 1..408>(...) /
  <integer -2147483648..2147483647>(...) / *NEW-PAGE / *SPACING(...) / *MODULE(...)
```

**\*FIELD(...)**

```
  INPUT-POSITION = <integer 1..32768>
  ,INPUT-LENGTH = 1 / <integer 1..204>
  ,INPUT-FORMAT = *CHARACTER / *ZONED-DECIMAL / *PACKED-DECIMAL
  ,OUTPUT-POSITION = <integer 1..204>
  ,OUTPUT-LENGTH = *STD / <integer 1..204>
  ,OUTPUT-FORMAT = *STD / *CHARACTER / *HEXADECIMAL / *BINARY /
    <c-string 2..204 with-low> / *ZONED-DECIMAL / *SIGNED-DECIMAL /
    *DECIMAL
```

**\*SUM-FIELD(...)**

```
  INPUT-POSITION = <integer 1..32768>
  ,INPUT-LENGTH = 1 / <integer 1..16>
  ,INPUT-FORMAT = *ZONED-DECIMAL / *PACKED-DECIMAL
  ,OUTPUT-POSITION = <integer 1..204>
  ,OUTPUT-LENGTH = *STD / <integer 1..204>
  ,OUTPUT-FORMAT = *ZONED-DECIMAL / <c-string 2..204 with-low> / *SIGNED-DECIMAL /
    *DECIMAL
```

**\*GROUP-COUNTER(...)**

```
  LINK-NAME = <full-filename 1..8 without-gen>
  ,GROUP-LEVEL = <integer 1..8>
  ,OUTPUT-POSITION = <integer 1..203>
  ,OUTPUT-LENGTH = *STD / <integer 1..204>
  ,OUTPUT-FORMAT = C 'ZZZ' / <c-string 2..204 with-low> / *ZONED-DECIMAL-LEFT
```

Fortsetzung ➔

**\*RECORD-LENGTH(...)****LINK-NAME** = <full-filename 1..8 without-gen>**,OUTPUT-POSITION** = <integer 1..203>**,OUTPUT-LENGTH** = **\*STD** / <integer 1..204>**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..204 with-low> / **\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL****\*BYTE-COUNTER(...)****LINK-NAME** = <full-filename 1..8 without-gen>**,OUTPUT-POSITION** = <integer 1..203>**,OUTPUT-LENGTH** = **\*STD** / <integer 1..204>**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..204 with-low> / **\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL****\*RECORD-COUNTER(...)****LINK-NAME** = <full-filename 1..8 without-gen>**,OUTPUT-POSITION** = <integer 1..203>**,OUTPUT-LENGTH** = **\*STD** / <integer 1..204>**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..204 with-low> / **\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL****\*PAGE-COUNTER(...)****LINK-NAME** = <full-filename 1..8 without-gen>**,OUTPUT-POSITION** = <integer 1..203>**,OUTPUT-LENGTH** = **\*STD** / <integer 1..204>**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..204 with-low> / **\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL****\*DATE(...)****OUTPUT-POSITION** = <integer 1..197>**,OUTPUT-LENGTH** = **\*STD** / <integer 8..204>**,OUTPUT-FORMAT** = **\*CHARACTER** / **\*HEXADECIMAL** / **\*BINARY****,CENTURY** = **\*NO** / **\*YES**

Fortsetzung ➡

(Abschnitt 6 von 6)

```

*TIME(...)
    |   OUTPUT-POSITION = <integer 1..197>
    |   ,OUTPUT-LENGTH = *STD / <integer 8..204>
    |   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY
<c-string>(...)
    |   OUTPUT-POSITION = <integer 1..204>
    |   ,OUTPUT-LENGTH = *STD / <integer 1..204>
    |   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY
<x-string>(...)
    |   OUTPUT-POSITION = <integer 1..204>
    |   ,OUTPUT-LENGTH = *STD / <integer 1..204>
    |   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY
<integer>(...)
    |   OUTPUT-POSITION = <integer 1..204>
    |   ,OUTPUT-LENGTH = *STD / <integer 1..204>
    |   ,OUTPUT-FORMAT = *ZONED-DECIMAL / <c-string 2..204 with-low> / *SIGNED-DECIMAL /
    |   *DECIMAL
*SPACING(...)
    |   LINES = 1 / <integer 1..15>
*MODULE(...)
    |   NAME = <name 1..8>

```

**OUTPUT-LINK-NAME =**

Dateikettungsamen der Ausgabedateien, auf die sich diese Anweisung beziehen soll.

**OUTPUT-LINK-NAME = \*STD**

Diese Anweisung bezieht sich auf alle bisher angegebenen Ausgabedateien.

**OUTPUT-LINK-NAME = list-poss(2000): <full-filename 1..8 without-gen>**

Soll sich die Anweisung nur auf einige Ausgabedateien beziehen, müssen die Dateikettungsamen dieser Dateien angegeben werden.

**GROUP-LEVEL = 1 / <integer 1..8>**

Gruppenstufe, für die diese SET-GROUP-ATTRIBUTES-Anweisung gilt. Es sind bis zu acht geschachtelte Gruppenstufen möglich. Durch die Angabe der Gruppenstufen wird eine Wertung der Gruppenbegriffe vorgenommen. Für jede Gruppenstufe ist eine eigene SET-GROUP-ATTRIBUTES-Anweisung anzugeben. Bei einem Gruppenwechsel einer Stufe werden auch Gruppen mit niedrigerem GROUP-LEVEL einem Wechsel unterzogen.

1: höchste Stufe

8: niedrigste Stufe

**GROUP-CONTROL =**

Definiert den Gruppenbegriff.

**GROUP-CONTROL = \*ALL**

Die gesamte Eingabedatei wird als eine Gruppe interpretiert. Diese Angabe ist nur in der höchsten Gruppenstufe (GROUP-LEVEL=1) möglich.

**GROUP-CONTROL = \*FIELD(...)**

Ein Feld des Eingabesatzes wird als Gruppenbegriff festgelegt.

**POSITION = <integer 1..32767>**

Anfangsposition des Vergleichsfeldes bezogen auf den Anfang des Eingabesatzes.

**LENGTH = 1 / <integer 1..256>**

Länge des Vergleichsfeldes begrenzt durch die Satzlänge des Eingabesatzes.

**GROUP-HEADER =**

Definiert Benutzerzeilen, die den Gruppenvorlauf bilden. Diese werden beim Gruppenwechsel vor der ersten Zeile der entsprechenden Gruppe ausgegeben.

**GROUP-HEADER = \*NONE**

Es werden keine Benutzerzeilen ausgegeben.

**GROUP-HEADER = \*FIELD(...)**

Übernimmt ein Feld des Eingabesatzes (Eingabefeld) in ein Feld der Benutzerzeile (Ausgabefeld).

**INPUT-POSITION = <integer 1..32768>**

Anfangsposition des Eingabefeldes bezogen auf den Anfang des Eingabesatzes.

**INPUT-LENGTH = 1 / <integer 1..204>**

Länge des Eingabefeldes begrenzt durch die Satzlänge des Eingabesatzes.

**INPUT-FORMAT = \*CHARACTER / \*ZONED-DECIMAL / \*PACKED-DECIMAL**

Format des Eingabefeldes im Eingabesatz (siehe Seite 75 ff).

Das Feld des Eingabesatzes, welches mittels Maske aufbereitet werden soll, darf bei INPUT-FORMAT=\*ZONED-DECIMAL nicht länger als 203, bei INPUT-FORMAT=\*PACKED-DECIMAL nicht länger als 102 byte sein.

**OUTPUT-POSITION = <integer 1..204>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH = \*STD / <integer 1..204>**

Länge des Ausgabefeldes.

Für den Standardwert gilt folgende Tabelle:

OUTPUT-FORMAT	INPUT-FORMAT		
	CHARACTER	ZONED-DECIMAL	PACKED-DECIMAL
CHARACTER	INPUT-LENGTH		
HEXADECIMAL	2 * INPUT-LENGTH		
BINARY	8 * INPUT-LENGTH		
<c-string> (Maske)		Maskenlänge	Maskenlänge
SIGNED-DECIMAL		INPUT-LENGTH + 1	2*INPUT-LENGTH
DECIMAL		INPUT-LENGTH + 1	2*INPUT-LENGTH
ZONED-DECIMAL		INPUT-LENGTH	2*INPUT-LENGTH - 1

**OUTPUT-FORMAT = \*STD / \*CHARACTER / \*HEXADECIMAL / \*BINARY / <c-string 2..204 with-low> / \*ZONED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL**

Format des Ausgabefeldes (siehe Seite 75 ff).

Erlaubt sind nur die Kombinationen:

OUTPUT-FORMAT	INPUT-FORMAT		
	CHARACTER	ZONED-DECIMAL	PACKED-DECIMAL
CHARACTER	+		
HEXADECIMAL	+		
BINARY	+		
<c-string> (Maske)		+	+
SIGNED-DECIMAL		+	+
DECIMAL		+	+
ZONED-DECIMAL		+	+

Für den Standardwert gilt folgende Tabelle:

INPUT-FORMAT	OUTPUT-FORMAT
CHARACTER	CHARACTER
PACKED-DECIMAL	ZONED-DECIMAL
ZONED-DECIMAL	ZONED-DECIMAL

### GROUP-HEADER = \*RECORD-LENGTH(...)

**LINK-NAME** = <full-filename 1..8 without-gen>

Dateikettungsname der Ein-/Ausgabedatei, deren aktuelle Satzlänge man in der Benutzerzeile verwenden will.

**OUTPUT-POSITION** = <integer 1..203>

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH** = \*STD / <integer 1..204>

Länge des Ausgabefeldes.

Für den Standardwert gilt folgende Tabelle:

OUTPUT-FORMAT	OUTPUT-LENGTH
ZONED-DECIMAL-LEFT	8 Zeichen
ZONED-DECIMAL	8 Zeichen
SIGNED-DECIMAL	9 Zeichen
DECIMAL	9 Zeichen

**OUTPUT-FORMAT** = \*ZONED-DECIMAL-LEFT / <c-string 2..204 with-low> / \*ZONED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL

Format des Ausgabefeldes (siehe Seite 75 ff).

### GROUP-HEADER = \*BYTE-COUNTER(...)

**LINK-NAME** = <full-filename 1..8 without-gen>

Dateikettungsname der Ein-/Ausgabedatei, deren aktuellen Bytezähler man in der Benutzerzeile verwenden will.

**OUTPUT-POSITION** = <integer 1..203>

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH** = \*STD / <integer 1..204>

Länge des Ausgabefeldes.

Für den Standardwert gilt folgende Tabelle:

OUTPUT-FORMAT	OUTPUT-LENGTH
ZONED-DECIMAL-LEFT	10 Zeichen
ZONED-DECIMAL	10 Zeichen
SIGNED-DECIMAL	11 Zeichen
DECIMAL	11 Zeichen

**OUTPUT-FORMAT** = \*ZONED-DECIMAL-LEFT / <c-string 2..204 with-low> /  
\*ZONED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL

Format des Ausgabefeldes (siehe Seite 75 ff).

**GROUP-HEADER** = \*RECORD-COUNTER(...)

**LINK-NAME** = <full-filename 1..8 without-gen>

Dateikettungsname der Ein-/Ausgabedatei, deren aktuellen Satzzählerwert man in der Benutzerzeile verwenden will.

**OUTPUT-POSITION** = <integer 1..203>

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH** = \*STD / <integer 1..204>

Länge des Ausgabefeldes.

Für den Standardwert gilt folgende Tabelle:

OUTPUT-FORMAT	OUTPUT-LENGTH
ZONED-DECIMAL-LEFT	8 Zeichen
ZONED-DECIMAL	8 Zeichen
SIGNED-DECIMAL	9 Zeichen
DECIMAL	9 Zeichen

**OUTPUT-FORMAT** = \*ZONED-DECIMAL-LEFT / <c-string 2..204 with-low> /  
\*ZONED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL

Format des Ausgabefeldes (siehe Seite 75 ff).

**GROUP-HEADER** = \*PAGE-COUNTER(...)

**LINK-NAME** = <full-filename 1..8 without-gen>

Dateikettungsname der Ausgabedatei, deren aktuellen Seitenzählerwert man in der Benutzerzeile verwenden will.



**OUTPUT-POSITION = <integer 1..203>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH = \*STD / <integer 1..204>**

Länge des Ausgabefeldes.

Für den Standardwert gilt folgende Tabelle:

OUTPUT-FORMAT	OUTPUT-LENGTH
ZONED-DECIMAL-LEFT	6 Zeichen
ZONED-DECIMAL	6 Zeichen
SIGNED-DECIMAL	7 Zeichen
DECIMAL	7 Zeichen

**OUTPUT-FORMAT = \*ZONED-DECIMAL-LEFT / <c-string 2..204 with-low> / \*ZONED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL**

Format des Ausgabefeldes (siehe Seite 75 ff).

**GROUP-HEADER = \*DATE(...)**

Datum als 8stellige Zeichenfolge im Format jj-mm-tt.

**OUTPUT-POSITION = <integer 1..197>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH = \*STD / <integer 8..204>**

Länge des Ausgabefeldes.

Für den Standardwert gilt folgende Tabelle:

OUTPUT-FORMAT	OUTPUT-LENGTH
CHARACTER	8 Zeichen
HEXADECIMAL	16 Zeichen
BINARY	64 Zeichen

**OUTPUT-FORMAT = \*CHARACTER / \*HEXADECIMAL / \*BINARY**

Format des Ausgabefeldes (siehe Seite 75 ff).

**CENTURY = \*NO / \*YES**

Bei der Angabe von CENTURY=\*YES wird das Datum als 10stellige Zeichenfolge mit der Angabe des Jahrhunderts im Format jjjj-mm-tt ausgegeben, damit verringert sich der Maximalwert für OUTPUT-POSITION um 2 auf 195 und der Minimalwert für OUTPUT-LENGTH erhöht sich um 2 auf 10.

**GROUP-HEADER = \*TIME(...)**

Uhrzeit als 8stellige Zeichenfolge im Format hh:mm:ss.

**OUTPUT-POSITION = <integer 1..197>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH = \*STD / <integer 8..204>**

Länge des Ausgabefeldes.

Für den Standardwert gilt folgende Tabelle:

OUTPUT-FORMAT	OUTPUT-LENGTH
CHARACTER	8 Zeichen
HEXADECIMAL	16 Zeichen
BINARY	64 Zeichen

**OUTPUT-FORMAT = \*CHARACTER / \*HEXADECIMAL / \*BINARY**

Format des Ausgabefeldes (siehe Seite 75 ff).

**GROUP-HEADER = <c-string 1..204 with-low>(...)****OUTPUT-POSITION = <integer 1..204>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH = \*STD / <integer 1..204>**

Länge des Ausgabefeldes.

Für den Standardwert gilt folgende Tabelle:

OUTPUT-FORMAT	OUTPUT-LENGTH
CHARACTER	Länge des c-string
HEXADECIMAL	2 * Länge des c-string
BINARY	8 * Länge des c-string

**OUTPUT-FORMAT = \*CHARACTER / \*HEXADECIMAL / \*BINARY**

Format des Ausgabefeldes (siehe Seite 75 ff).

**GROUP-HEADER = <x-string 1..408>(...)****OUTPUT-POSITION = <integer 1..204>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH = \*STD / <integer 1..204>**

Länge des Ausgabefeldes.

Für den Standardwert gilt folgende Tabelle:

OUTPUT-FORMAT	OUTPUT-LENGTH
CHARACTER	0,5 * Länge des x-string
HEXADECIMAL	Länge des x-string
BINARY	4 * Länge des x-string

**OUTPUT-FORMAT = \*CHARACTER / \*HEXADECIMAL / \*BINARY**

Format des Ausgabefeldes (siehe Seite 75 ff).

**GROUP-HEADER = <integer -2147483648..2147483647>(…)**

**OUTPUT-POSITION = <integer 1..204>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH = \*STD / <integer 1..204>**

Länge des Ausgabefeldes.

Standardwert ist die Mindestlänge, die erforderlich ist, um die auszugebene Zahl im angegebenen Format unterzubringen.

**OUTPUT-FORMAT = \*ZONED-DECIMAL / <c-string 2..204 with-low> / \*SIGNED-DECIMAL / \*DECIMAL**

Format des Ausgabefeldes (siehe Seite 75 ff).

**GROUP-HEADER = \*NEW-PAGE**

Bewirkt einen Vorschub auf die nächste Seite und die Ausgabe der bisher aufgebauten Benutzerzeile. Nach dem Ausführen kann eine weitere Benutzerzeile erstellt werden.

**GROUP-HEADER = \*SPACING(…)**

Bewirkt die Ausgabe der bisher aufgebauten Benutzerzeile und einen Vorschub um 1 bis 15 Zeilen. Nach dem Ausführen des Operanden kann eine weitere Benutzerzeile erstellt werden.

**LINES = 1 / <integer 1..15>**

Anzahl der Zeilenvorschübe.

**GROUP-HEADER = \*MODULE(...)****NAME = <name 1..8>**

Die Aufbereitung einer Gruppenvorlaufbenutzerzeile erfolgt in einem Benutzermodul. Anzugeben ist der Name des Benutzermoduls oder der Einsprungpunkt in dem Benutzermodul. Eine Benutzerzeile, die bereits erstellt ist, aber noch nicht ausgedruckt wurde, wird der Benutzerroutine zur Verfügung gestellt (siehe Seite 187, Anschluß für die Gruppenbearbeitung).

**GROUP-TRAILER = \*NONE / list-poss(2000): \*FIELD(...) / \*SUM-FIELD(...) / \*GROUP-COUNTER(...) / \*RECORD-LENGTH / \*BYTE-COUNTER(...) / \*RECORD-COUNTER(...) / \*PAGE-COUNTER(...) / \*DATE(...) / \*TIME(...) / <c-string 1..204 with-low>(...) / <x-string 1..408>(...) / <integer -2147483648..2147483647>(...) / \*NEW-PAGE / \*SPACING(...) / \*MODULE(...)**

Definiert Benutzerzeilen, die den Gruppennachlauf bilden. Diese werden beim Gruppenwechsel nach der letzten Zeile der entsprechenden Gruppe ausgegeben.

Bei GROUP-TRAILER=\*FIELD wird die FIELD-Information dem letzten Satz der Gruppe entnommen.

Die Beschreibung der Operanden außer SUM-FIELD(...) und GROUP-COUNTER(...) siehe GROUP-HEADER (ab Seite 133).

**GROUP-TRAILER = \*SUM-FIELD(...)**

Die Werte des Zahlenfeldes aller Sätze der zuletzt durchlaufenen Gruppe werden summiert. Diese Summe wird in das Summenfeld der Benutzerzeile übertragen.

Das Summenfeld muß ausreichende Länge haben. Es können mehrere Felder summiert werden.

**INPUT-POSITION = <integer 1..32768>**

Anfangsposition des Zahlenfeldes bezogen auf den Anfang des Eingabesatzes.

**INPUT-LENGTH = 1 / <integer 1..16>**

Länge des Zahlenfeldes.

**INPUT-FORMAT = \*ZONED-DECIMAL / \*PACKED-DECIMAL**

Format des Zahlenfeldes (siehe Seite 75 ff).

**OUTPUT-POSITION = <integer 1..204>**

Anfangsposition des Summenfeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH = \*STD / <integer 1..204>**

Länge des Summenfeldes.

Standardwert ist die Mindestlänge, die erforderlich ist, um die auszugebende Zahl im angegebenen Format unterzubringen. \*STD ist bei Ausgabe mit Maske die Maskenlänge. Bei den nachfolgenden aufgeführten Umwandlungen werden folgende Werte für \*STD eingesetzt:

OUTPUT-FORMAT	INPUT-FORMAT	
	ZONED-DECIMAL	PACKED-DECIMAL
SIGNED-DECIMAL	INPUT-LENGTH + 1	2 * INPUT-LENGTH
DECIMAL	INPUT-LENGTH + 1	2 * INPUT-LENGTH
ZONED-DECIMAL		2 * INPUT-LENGTH-1

In allen anderen Fällen wird INPUT-LENGTH angenommen.

**OUTPUT-FORMAT = \*ZONED-DECIMAL / <c-string 2..204 with-low> / \*SIGNED-DECIMAL / \*DECIMAL**

Format des Summenfeldes (siehe Seite 75 ff).

**GROUP-TRAILER = \*GROUP-COUNTER(...)**

Zähler je Gruppe. GROUP-COUNTER ist für die Gruppenstufe 1 bis 8 jeder Ausgabeeinheit verfügbar. Jeder Zähler ist durch seinen Dateikettungsnamen und durch eine Gruppenstufe eindeutig bestimmt.

**LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ein-/Ausgabedatei, von der man einen Gruppenzähler in der Benutzerzeile verwenden will.

**GROUP-LEVEL = <integer 1..8>**

Angabe, welcher Gruppenzähler der mit LINK-NAME bezeichneten Ein-/Ausgabedatei verwendet werden soll.

**OUTPUT-POSITION = <integer 1..203>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**OUTPUT-LENGTH = \*STD / <integer 4..204>**

Länge des Ausgabefeldes.

Der Standardwert ist entweder 4 bei OUTPUT-FORMAT=\*ZONED-DECIMAL-LEFT oder die Maskenlänge.

**OUTPUT-FORMAT = C' ZZZ' / <c-string 2..204 with-low> / \*ZONED-DECIMAL-LEFT**

Format des Ausgabefeldes (siehe Seite 75 ff).

SET-PAGE-LAYOUT  
Beschreiben des Ausgabeformates

Diese Anweisung beschreibt die seitengerechte Gestaltung der Ausgabedatei.

Die SET-PAGE-LAYOUT-Anweisung kann entfallen, wenn in der ASSIGN-OUTPUT-FILE-Anweisung FILE=\*SYSOUT bzw. FILE=\*SYSLST oder wenn eine SET-GROUP-ATTRIBUTES-Anweisung angegeben wurde. In diesen Fällen wird eine Aufbereitung entsprechend der Standardwerte durchgeführt (siehe Seite 13).

Beziehen sich in einem Konvertierungsschritt mehrere SET-PAGE-LAYOUT-Anweisungen auf den gleichen Dateikettungsnamen, wird für diesen Dateikettungsnamen nur die zuletzt gegebene Anweisung ausgeführt.

Beim Aufbereiten einer Ausgabe für SYSOUT werden folgende Zeilen nicht berücksichtigt:

- Kopfzeilen (unabhängig von der Angabe im Operanden HEADER-LINE)
- Leerzeilen
- Benutzerzeilen (unabhängig von der Angabe im Operanden USER-LINE)

Beim Aufbereiten eines Ausgabesatzes werden die in ihm enthaltenen Steuerzeichen X'00' bis X'3F' behandelt. Dies kann vom Benutzer durch den Operanden REPLACE-CONTROL-CHAR unterdrückt werden, wenn er z.B. selbst Steuerzeichen zur Druckersteuerung in den Satz einfügt.

(Abschnitt 1 von 3)

SET-PAGE-LAYOUT
<p><b>OUTPUT-LINK-NAME</b> = <u>*STD</u> / list-poss(2000): &lt;full-filename 1..8 without-gen&gt;</p> <p><b>,HEADER-LINE</b> = <u>*STD</u> / *NONE / *PAGE-COUNTER / *TITLE</p> <p><b>,OUTPUT-AREA</b> = list-poss(2000): <u>*LINE-RANGE</u> (...)</p> <p>    <u>*LINE-RANGE</u> (...)</p> <p>        <b>FIRST-LINE</b> = <u>5</u> / &lt;integer 1..112&gt;</p> <p>        <b>,LAST-LINE</b> = <u>66</u> / &lt;integer 1..112&gt;</p> <p><b>,SPACING</b> = <u>0</u> / &lt;integer 0..3&gt;</p> <p><b>,OUTPUT-FORMAT</b> = <u>*STD</u> / *CHARACTER / *HEXADECIMAL / *BOTH</p> <p><b>,LINE-SIZE</b> = <u>*STD</u> / &lt;integer 1..204&gt;</p> <p><b>,COLUMN-SIZE</b> = <u>*LINE-SIZE</u> / &lt;integer 1..204&gt;</p> <p><b>,SUPPRESS-EQUAL-LINES</b> = *YES / *NO</p> <p><b>,REPLACE-CONTROL-CHAR</b> = <u>*YES</u> / *NO</p>

Fortsetzung ➡

(Abschnitt 2 von 3)

```

,USER-LINE = *NONE / list-poss(2000): *RECORD-LENGTH(...) / *BLOCK-COUNTER(...) /
  *BYTE-COUNTER(...) / *RECORD-COUNTER(...) / *PAGE-COUNTER(...) / *DATE(...) / *TIME(...) /
  <c-string 1..204 with-low>(...) / <x-string 1..408>(...)

*RECORD-LENGTH(...)
  | LINK-NAME = <full-filename 1..8 without-gen>
  | ,OUTPUT-LINE = <integer 1..112>
  | ,OUTPUT-COLUMN = <integer 1..197>

*BLOCK-COUNTER(...)
  | LINK-NAME = <full-filename 1..8 without-gen>
  | ,OUTPUT-LINE = <integer 1..112>
  | ,OUTPUT-COLUMN = <integer 1..197>

*BYTE-COUNTER(...)
  | LINK-NAME = <full-filename 1..8 without-gen>
  | ,OUTPUT-LINE = <integer 1..112>
  | ,OUTPUT-COLUMN = <integer 1..195>

*RECORD-COUNTER(...)
  | LINK-NAME = <full-filename 1..8 without-gen>
  | ,OUTPUT-LINE = <integer 1..112>
  | ,OUTPUT-COLUMN = <integer 1..197>

*PAGE-COUNTER(...)
  | LINK-NAME = <full-filename 1..8 without-gen>
  | ,OUTPUT-LINE = <integer 1..112>
  | ,OUTPUT-COLUMN = <integer 1..199>

*DATE(...)
  | OUTPUT-LINE = <integer 1..112>
  | ,OUTPUT-COLUMN = <integer 1..197>
  | ,CENTURY = *NO / *YES

```

Fortsetzung ➡

```
*TIME(...)
|
|   OUTPUT-LINE = <integer 1..112>
|   ,OUTPUT-COLUMN = <integer 1..197>
|
|<c-string>(...)
|   OUTPUT-LINE = <integer 1..112>
|   ,OUTPUT-COLUMN = <integer 1..204>
|
|<x-string>(...)
|   OUTPUT-LINE = <integer 1..112>
|   ,OUTPUT-COLUMN = <integer 1..204>
```

**OUTPUT-LINK-NAME =**  
Dateikettungsnamen der Ausgabedateien, auf die sich diese Anweisung beziehen soll.

**OUTPUT-LINK-NAME = \*STD**  
Diese Anweisung bezieht sich auf alle bisher angegebenen Ausgabedateien.

**OUTPUT-LINK-NAME = list-poss(2000): <full-filename 1..8 without-gen>**  
Soll sich die Anweisung nur auf einige Ausgabedateien beziehen, müssen die Dateiket-  
tungsnamen dieser Dateien angegeben werden.

**HEADER-LINE =**  
Kopfzeile, die pro Seite ausgegeben wird.

**HEADER-LINE = \*STD**  
Beim Kopieren von Dateien: HEADER-LINE=\*PAGE-COUNTER  
Beim Editieren von Bändern: HEADER-LINE=\*TITLE

**HEADER-LINE = \*NONE**  
PERCON gibt keine Kopfzeile aus.

**HEADER-LINE = \*PAGE-COUNTER**  
Der Text PAGE xxxxx wird pro Seite in Zeile 1, Spalte 119 eingefügt.  
xxxxx ist der aktuelle Stand des Seitenzählers.

**HEADER-LINE = \*TITLE**  
PERCON gibt eine Standard-Kopfzeile in Zeile 1 aus.

1	36	41	86	91	119
PERCON	DATE	jjjj-mm-tt	TIME	hh:mm:ss	PAGE xxxxx



**OUTPUT-AREA =**

Definiert über FIRST-LINE und LAST-LINE die Anfangs- und Endzeile eines Zeilenbereichs pro Druckseite, in den PERCON die Ausgabesätze ausgibt. Es können mehrere solcher Zeilenbereiche pro Druckseite definiert werden.

Mit diesem Operand kann die Kopfzeile überschrieben werden.

**OUTPUT-AREA = \*LINE-RANGE (...)**

Der Zeilenbereich wird festgelegt.

**FIRST-LINE = 5 / <integer 1..112>**

Anfangszeile für die Datenausgabe.

**LAST-LINE = 66 / <integer 1..112>**

Endzeile für die Datenausgabe.

**SPACING = 0 / <integer 0..3>**

Anzahl der Leerzeilen, die nach jeder Druckzeile ausgegeben werden.

**OUTPUT-FORMAT =**

Ausgabeformat der auszugebenden Daten.

**OUTPUT-FORMAT = \*STD**

Beim Kopieren von Dateien: OUTPUT-FORMAT = \*CHARACTER

Beim Editieren von Bändern: OUTPUT-FORMAT = \*BOTH

**OUTPUT-FORMAT = \*CHARACTER**

Die Daten werden als Zeichen ausgegeben. Nicht abdruckbare Zeichen werden nur dann durch Leerzeichen ersetzt, wenn REPLACE-CONTROL-CHAR=\*YES gilt und keine von XHCS gelieferte Tabelle der druckbaren Zeichen verfügbar ist.

**OUTPUT-FORMAT = \*HEXADECIMAL**

Die Daten werden in sedezimaler Form ausgegeben.

**OUTPUT-FORMAT = \*BOTH**

Die Daten werden in sedezimaler Form ausgegeben. Abdruckbare Zeichen werden zusätzlich in einer zweiten Zeile über den entsprechenden Sedezimalverschlüsselungen ausgegeben. Nicht abdruckbare Zeichen werden nur dann durch Leerzeichen ersetzt, wenn REPLACE-CONTROL-CHAR=\*YES gilt und keine von XHCS gelieferte Tabelle der druckbaren Zeichen verfügbar ist. Es werden demnach von jedem Ausgabesatz (bzw. jedem Teil desselben, wenn er länger als die Zeile ist, siehe Operand LINE-SIZE) zwei Datenzeilen belegt.

**LINE-SIZE = \*STD / <integer 1..204>**

Anzahl der Zeichen des aufzubereitenden Satzes, die pro Zeile auszugeben sind. Dadurch kann ein Ausgabesatz auf mehrere Zeilen aufgeteilt werden. Neben dem Operand LINE-SIZE bestimmen die Operanden COLUMN-SIZE und OUTPUT-FORMAT die Länge einer Ausgabezeile.

Haben die aufzubereitenden Sätze variables Satzformat, wird auch das Satzlängengebiet ein Teil der auszugebenden Daten.

Am Anfang jeder Zeile wird zusätzlich ein Vorspann ausgegeben, dessen Länge in folgender Weise vom gewählten Ausgabemedium und der ausgewählten Funktion abhängt.

Funktion	Ausgabedatei	
	SYSLST, katalogisierte Datei	SYSOUT
Datei kopieren	kein Vorspann	9 Zeichen (Satzposition) (BYTCNT)
Band editieren	21 Zeichen (Bandabschnittsmarkenzähler) (TMC) (Blockzähler) (BLKCNT) (Satzposition) (BYTCNT)	9 Zeichen (Satzposition) (BYTCNT)

**COLUMN-SIZE = \*LINE-SIZE / <integer 1..204>**

Anzahl der Zeichen des Ausgabesatzes, die zu einer Gruppe zusammenzufassen und durch Leerzeichen von der nächsten Gruppe zu trennen sind.

**COLUMN-SIZE = \*LINE-SIZE**

Es erfolgt keine Aufteilung in Gruppen.

**SUPPRESS-EQUAL-LINES =**

Beeinflusst das Ausdrucken von Zeilen mit gleichen Inhalt. Die letzte Zeile eines aufbereiteten Satzes wird immer ausgegeben, unabhängig von der Belegung des Parameters SUPPRESS-EQUAL-LINES.

**SUPPRESS-EQUAL-LINES = \*YES**

Folgen zwei oder mehr identische Zeilen aufeinander, wird die sich wiederholende Zeile nur einmal ausgedruckt. Es folgt in der nächsten Zeile der Text

'xxxx IDENTICAL LINES SUPPRESSED',

wobei xxxx die Anzahl der Zeilen mit gleichem Inhalt angibt.

**SUPPRESS-EQUAL-LINES = \*NO**

Es werden alle, auch sich wiederholende Zeilen, ausgegeben.

**REPLACE-CONTROL-CHAR =**

Regelt die Steuerzeichenbehandlung für die Druckersteuerung.

**REPLACE-CONTROL-CHAR = \*YES**

Die Steuerzeichen X'00' bis X'3F' in den Ausgabesätzen werden durch Leerzeichen ersetzt, wenn entweder das Subsystem XHCS nicht aktiv ist oder wenn bei aktivem XHCS die Ausgabedatei keinen CCSN besitzt.

Ist das Subsystem XHCS aktiv und hat die Ausgabedatei einen CCSN, so wird die XHCS-Tabelle der druckbaren Zeichen für dieses CCS zum Ersetzen herangezogen.

**REPLACE-CONTROL-CHAR = \*NO**

In den Ausgabesätzen sind Zeichen enthalten, die nicht verändert werden sollen.

Auf nichtdruckbare Zeichen wird nicht geprüft. Zeichen werden nicht durch andere ersetzt.

**USER-LINE =**

Gibt Kennwörter und Literale an bestimmten Seitenpositionen aus. Kopfzeile und Zeilenbereiche, die durch den Operand OUTPUT-AREA festgelegt wurden, können überschrieben werden.

**USER-LINE = \*NONE**

Es werden keine Benutzerzeilen ausgegeben.

**USER-LINE = \*RECORD-LENGTH(...)****LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ein-/Ausgabedatei oder des Eingabebandes, deren (dessen) aktuelle Satzlänge man in der Benutzerzeile verwenden will.

Dargestellt als 8stellige Dezimalzahl.

**OUTPUT-LINE = <integer 1..112>**

Nummer der Zeile, in die die Satzlänge geschrieben wird.

**OUTPUT-COLUMN = <integer 1..197>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**USER-LINE = \*BLOCK-COUNTER(...)****LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname des Eingabebandes, dessen aktuellen Blockzählerwert man in der Benutzerzeile verwenden will.

Dargestellt als 8stellige Dezimalzahl.

**OUTPUT-LINE = <integer 1..112>**

Nummer der Zeile, in die der Wert des Blockzählers geschrieben wird.

**OUTPUT-COLUMN = <integer 1..197>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**USER-LINE = \*BYTE-COUNTER(...)****LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ein-/Ausgabedatei oder des Eingabebandes, deren (dessen) aktuellen Bytezählerwert man in der Benutzerzeile verwenden will.

Dargestellt als 10stellige Dezimalzahl.

**OUTPUT-LINE = <integer 1..112>**

Nummer der Zeile, in die der Wert des Bytezählers geschrieben wird.

**OUTPUT-COLUMN = <integer 1..195>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**USER-LINE = \*RECORD-COUNTER(...)****LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ein-/Ausgabedatei, deren aktuellen Satzzählerwert man in der Benutzerzeile verwenden will.

Dargestellt als 8stellige Dezimalzahl.

**OUTPUT-LINE = <integer 1..112>**

Nummer der Zeile, in die der Wert des Satzzählers geschrieben wird.

**OUTPUT-COLUMN = <integer 1..197>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**USER-LINE = \*PAGE-COUNTER(...)****LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ausgabedatei, deren aktuellen Seitenzählerwert man in der Benutzerzeile verwenden will.

Dargestellt als 6stellige Dezimalzahl.

**OUTPUT-LINE = <integer 1..112>**

Nummer der Zeile, in die die Seitennummer geschrieben wird.

**OUTPUT-COLUMN = <integer 1..199>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**USER-LINE = \*DATE(...)**

Datum als 8stellige Zeichenfolge im Format jj-mm-tt.

**OUTPUT-LINE = <integer 1..112>**

Nummer der Zeile, in die das Datum geschrieben wird.

**OUTPUT-COLUMN = <integer 1..197>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**CENTURY = \*NO / \*YES**

Bei der Angabe von CENTURY=\*YES wird das Datum als 10stellige Zeichenfolge mit der Angabe des Jahrhunderts im Format jjjj-mm-tt ausgegeben. Damit verringert sich der Maximalwert für OUTPUT-COLUMN um 2.

**USER-LINE = \*TIME(...)**

Uhrzeit als 8stellige Zeichenfolge im Format hh:mm:ss.

**OUTPUT-LINE = <integer 1..112>**

Nummer der Zeile, in die die Uhrzeit geschrieben wird.

**OUTPUT-COLUMN = <integer 1..197>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**USER-LINE =<c-string 1..204 with-low>(...)****OUTPUT-LINE = <integer 1..112>**

Nummer der Zeile, in die die Zeichenkonstante geschrieben wird.

**OUTPUT-COLUMN = <integer 1..204>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

**USER-LINE =<x-string 1..408>(...)****OUTPUT-LINE = <integer 1..112>**

Nummer der Zeile, in die die Zeichenkonstante geschrieben wird.

**OUTPUT-COLUMN = <integer 1..204>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang der Benutzerzeile.

## SET-RECORD-MAPPING

### Festlegen des Ausgabesatzaufbaus

Diese Anweisung ist nur beim Kopieren von Dateien zulässig. Mit ihr kann der Aufbau des Ausgabesatzes gegenüber dem Eingabesatz verändert werden. Der Ausgabesatz kann aus Feldern des Eingabesatzes, aus Literalen und Kennwörtern zusammengestellt werden.

### Überschreiben des Ausgabesatzes

Ein bereits existierender Ausgabesatz kann entweder mit dem Eingabesatz oder mit einem aus den Daten des Eingabesatzes, aus Kennwörtern und Literalen aufgebauten Satz überschrieben werden. Im zweiten Fall wird der Eingabesatz oder ein Füllzeichen zur Vorbelegung des Ausgabebereichs verwendet.

### Aktualisieren des Ausgabesatzes

Einzelne Felder in einem bereits existierenden Ausgabesatz können aktualisiert werden (Schlüsselwort \*OUTPUT des Operanden FILLER). Ein Satz der Ausgabedatei wird dazu zur Vorbelegung des Ausgabebereichs herangezogen. Folgende Randbedingungen sind dabei zu beachten:

1. Zwischen Eingabesatz und zugehörigem Ausgabesatz muß sich eine Zuordnung herstellen lassen. Da an die Eingabedatei keine Bedingungen bezüglich des Sortierens gestellt werden, muß ein wahlfreier Zugriff auf die Ausgabedatei möglich sein. Deshalb wird als Ausgabedatei nur eine bereits existierende ISAM-Datei zugelassen. Über einen Schlüssel wird auf den Ausgabesatz zugegriffen. Dabei kann auch ein Sekundärschlüssel angegeben werden. Die Ausgabedatei wird mit dem OPEN-Modus INOUT eröffnet. Wird keine ISAM-Datei zur Ausgabe verwendet, bricht der Konvertierungsschritt mit der Meldung PER0053 ab.
2. Die Eingabedatei kann von beliebigem Dateityp sein. Sie muß allerdings in jedem Satz ab gleicher Position ein Feld aufweisen, das den Schlüssel enthält, über den der Bezug zum Ausgabesatz hergestellt wird.
3. Das Schlüsselfeld der Eingabedatei muß die gleiche Länge haben wie der verwendete Schlüssel der Ausgabedatei.
4. Jeder Eingabesatz muß mindestens so lang sein, daß das Schlüsselfeld vollständig in ihm enthalten ist. Überragt das Schlüsselfeld das Ende des Eingabesatzes, wird der Konvertierungsschritt mit der Meldung PER0044 abgebrochen.
5. Ist die Eingabedatei eine ISAM-Datei, so muß das Schlüsselfeld zur Zuordnung des Satzes der Ausgabedatei nicht notwendig ein ISAM-Schlüsselfeld sein.
6. Enthält die Ausgabedatei mehrere Sätze mit gleichen Werten im angesprochenen Schlüssel, so wird der erste Satz zur Vorbelegung herangezogen.

7. Die Satzlänge des neuen Ausgabesatzes kann geändert werden. Aber beim Rückschreiben eines verlängerten Satzes kann sich dessen Position innerhalb einer Folge von Sätzen mit gleichem Schlüssel ändern (Eigenschaft von ISAM). Damit ist nicht ausgeschlossen, daß bei einem weiteren Zugriff mit diesem Schlüssel ein anderer Satz bearbeitet wird.
8. Die Operandenwerte in der Anweisung SET-RECORD-MAPPING müssen so festgelegt werden, daß das Schlüsselfeld des Primärschlüssels im Ausgabesatz nicht verändert wird (Forderung von ISAM). Wird diese Forderung nicht eingehalten, wird der Konvertierungsschritt mit der Meldung PER0085 abgebrochen.
9. Wird ein Schlüssel in der Ausgabedatei nicht gefunden, wird die Warnung PER0054 ausgegeben, die den Schlüssel als Insert enthält. Die Verarbeitung wird mit dem nächsten Eingabesatz fortgesetzt. Diese Meldung wird höchstens zehnmal für jeden Dateikettungsnamen ausgegeben.
10. Wenn der entsprechend der SET-RECORD-MAPPING-Anweisung aufgebaute Satz länger ist als der zur Vorbelegung gelesene Satz, so werden diejenigen Bereiche mit Leerzeichen aufgefüllt, die weder durch den Vorbelegungssatz noch durch den Operanden OUTPUT-FIELDS beschrieben werden.

Bei der Verarbeitung eines Eingabesatzes laufen dann im Normalfall folgende Schritte ab:

1. Lesen eines Eingabesatzes und Bestimmen des Schlüssels für den zugehörigen Ausgabesatz.
2. Lesen des zugehörigen Ausgabesatzes über den Schlüssel und Vorbelegen des Ausgabebereiches mit diesem Satz.
3. Aufbauen des vollständigen Ausgabesatzes entsprechend den Angaben in der Anweisung SET-RECORD-MAPPING.
4. Rückschreiben des Ausgabesatzes in die Ausgabedatei.

### **Mindest-Satzlänge festlegen**

Beim Aufbauen eines Ausgabesatzes wird dessen Länge, sofern keine feste Satzlänge vorliegt, von der höchsten Satzposition bestimmt, die im Operanden OUTPUT-FIELDS der Anweisung SET-RECORD-MAPPING angesprochen wird. Über den Operanden MIN-RECORD-LENGTH kann für die Daten eine Mindestlänge angegeben werden, die entweder fest sein kann oder von der Länge des jeweiligen Eingabesatzes abhängt. Ist der aufgebaute Satz kürzer als diese Mindestlänge, so wird er entweder mit dem Füllzeichen oder mit Leerzeichen bei einer Vorbelegung des Ausgabebereiches mit dem Ein- bzw. bereits existierenden Ausgabesatz auf diese Länge erweitert.

Die Anweisung SET-RECORD-MAPPING ist wahlweise. Fehlt sie, wird der Eingabesatz unverändert als Ausgabesatz übernommen.

*Hinweis*

Eine eventuelle Konvertierung in ein Ausgabe-CCS wird in diesem Sinne nicht als Veränderung aufgefaßt.

Beziehen sich in einem Konvertierungsschritt mehrere SET-RECORD-MAPPING-Anweisungen auf den gleichen Dateikettungsnamen, wird für diesen Dateikettungsnamen nur die zuletzt gegebene Anweisung ausgeführt.

(Abschnitt 1 von 3)

**SET-RECORD-MAPPING**

```

OUTPUT-LINK-NAME = *STD / list-poss(2000): <full-filename 1..8 without-gen>
FILLER = C ' ' / <c-string 1..1 with-low> / <x-string 1..2> / *INPUT / *OUTPUT(...)
  *OUTPUT(...)
    |
    | KEY-NAME = *PRIMARY / <name 1..8>
    | KEY-VALUE = *BY-INPUT-RECORD (...)
    |   *BY-INPUT-RECORD(...)
    |     |
    |     | KEY-POSITION = *STD / <integer 1..32768>
    |
    | OUTPUT-FIELDS = *COMPLETE-RECORD / list-poss(2000): *FIELD(...) / *GROUP-COUNTER(...) /
    |   *RECORD-LENGTH(...) / *BYTE-COUNTER(...) / *RECORD-COUNTER(...) / *PAGE-COUNTER(...) /
    |   *DATE(...) / *TIME(...) / <c-string 1..256 with-low>(…) / <x-string 1..512>(…) /
    |   <integer -2147483648..2147483647>(…)
    |
    | *FIELD(...)
    |   |
    |   | INPUT-POSITION = <integer 1..32768>
    |   | INPUT-LENGTH = 1 / <integer 1..32767> / *RECORD-LENGTH(...)
    |   |   *RECORD-LENGTH(...)
    |   |     |
    |   |     | REDUCTION = 0 / <integer 0..32767>
    |   |
    |   | INPUT-FORMAT = *CHARACTER / *ZONED-DECIMAL / *PACKED-DECIMAL
    |   | OUTPUT-POSITION = <integer 1..32768>
    |   | OUTPUT-LENGTH = *STD / <integer 1..32767> / *RECORD-LENGTH(...)
    |   |   *RECORD-LENGTH(...)
    |   |     |
    |   |     | REDUCTION = 0 / <integer 0..32767>
    |   |
    |   | OUTPUT-FORMAT = *INPUT-FORMAT / *CHARACTER / *HEXADECIMAL / *BINARY /
    |   |   *TRANSLATION / *NO-TRANSLATION / *ZONED-DECIMAL / *PACKED-DECIMAL /
    |   |   <c-string 2..256 with-low> / *SIGNED-DECIMAL / *DECIMAL

```

Fortsetzung ➡



(Abschnitt 2 von 3)

**\*GROUP-COUNTER(...)**

**LINK-NAME** = <full-filename 1..8 without-gen>  
**,GROUP-LEVEL** = <integer 1..8>  
**,OUTPUT-POSITION** = <integer 1..32767>  
**,OUTPUT-LENGTH** = **\*STD** / <integer 1..32767>  
**,OUTPUT-FORMAT** = **C 'ZZZ'** / <c-string 2..256 with-low> / **\*ZONED-DECIMAL-LEFT**

**\*RECORD-LENGTH(...)**

**LINK-NAME** = <full-filename 1..8 without-gen>  
**,OUTPUT-POSITION** = <integer 1..32767>  
**,OUTPUT-LENGTH** = **\*STD** / <integer 1..32767>  
**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..256 with-low> / **\*PACKED-DECIMAL** /  
**\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL**

**\*BYTE-COUNTER(...)**

**LINK-NAME** = <full-filename 1..8 without-gen>  
**,OUTPUT-POSITION** = <integer 1..32767>  
**,OUTPUT-LENGTH** = **\*STD** / <integer 1..32767>  
**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..256 with-low> / **\*PACKED-DECIMAL** /  
**\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL**

**\*RECORD-COUNTER(...)**

**LINK-NAME** = <full-filename 1..8 without-gen>  
**,OUTPUT-POSITION** = <integer 1..32767>  
**,OUTPUT-LENGTH** = **\*STD** / <integer 1..32767>  
**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..256 with-low> / **\*PACKED-DECIMAL** /  
**\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL**

**\*PAGE-COUNTER(...)**

**LINK-NAME** = <full-filename 1..8 without-gen>  
**,OUTPUT-POSITION** = <integer 1..32767>  
**,OUTPUT-LENGTH** = **\*STD** / <integer 1..32767>  
**,OUTPUT-FORMAT** = **\*ZONED-DECIMAL-LEFT** / <c-string 2..256 with-low> / **\*PACKED-DECIMAL** /  
**\*ZONED-DECIMAL** / **\*SIGNED-DECIMAL** / **\*DECIMAL**

Fortsetzung ➔

(Abschnitt 3 von 3)

```

*DATE(...)
    |   OUTPUT-POSITION = <integer 1..32761>
    |   ,OUTPUT-LENGTH = *STD / <integer 8..32767>
    |   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY
    |   ,CENTURY = *NO / *YES

*TIME(...)
    |   OUTPUT-POSITION = <integer 1..32761>
    |   ,OUTPUT-LENGTH = *STD / <integer 8..32767>
    |   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY

<c-string>(...)
    |   OUTPUT-POSITION = <integer 1..32768>
    |   ,OUTPUT-LENGTH = *STD / <integer 1..32767>
    |   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY

<x-string>(...)
    |   OUTPUT-POSITION = <integer 1..32768>
    |   ,OUTPUT-LENGTH = *STD / <integer 1..32767>
    |   ,OUTPUT-FORMAT = *CHARACTER / *HEXADECIMAL / *BINARY

<integer>(...)
    |   OUTPUT-POSITION = <integer 1..32768>
    |   ,OUTPUT-LENGTH = *STD / <integer 1..32767>
    |   ,OUTPUT-FORMAT = *ZONED-DECIMAL / <c-string 2..256 with-low> / *PACKED-DECIMAL /
    |                   *SIGNED-DECIMAL / *DECIMAL

,CODE-TRANSLATION = *NONE / *TABLE(...) / list-poss(2000): *REPLACE-CHARACTER(...)

*TABLE(...)
    |   FILE-NAME = <full-filename 1..54>

*REPLACE-CHARACTER(...)
    |   INPUT-CHARACTER = <c-string 1..1 with-low> / <x-string 1..2>
    |   ,OUTPUT-CHARACTER = <c-string 1..1 with-low> / <x-string 1..2>

,MIN-RECORD-LENGTH = *NONE / *BY-INPUT-RECORD(...) / <integer 1..32768>

*BY-INPUT-RECORD(...)
    |   ADDITION = 0 / <integer -32767..32767>

```

**OUTPUT-LINK-NAME =**

Dateikettungsnamen der Ausgabedateien, auf die sich diese Anweisung beziehen soll.

**OUTPUT-LINK-NAME = \*STD**

Diese Anweisung bezieht sich auf alle bisher angegebenen Ausgabedateien.

**OUTPUT-LINK-NAME = list-poss(2000): <full-filename 1..8 without-gen>**

Soll sich die Anweisung nur auf einige Ausgabedateien beziehen, müssen die Dateikettungsnamen dieser Dateien angegeben werden.

**FILLER =**

Füllzeichen für die nicht über den Operand OUTPUT-FIELDS beschriebenen Bereiche des Ausgabesatzes.

**FILLER = C' '**

Das Füllzeichen ist ein Leerzeichen.

**FILLER = <c-string 1..1 with-low>**

Füllzeichen im CHARACTER-Format.

**FILLER = <x-string 1..2>**

Füllzeichen im HEXADECIMAL-Format.

**FILLER = \*INPUT**

Der Ausgabesatz wird mit dem Eingabesatz vorbesetzt. Ist der Ausgabesatz länger als der Eingabesatz, wird der Rest des Ausgabesatzes mit Leerzeichen aufgefüllt.

**FILLER = \*OUTPUT (...)**

Legt fest, daß ein Satz der Ausgabedatei zur Vorbelegung des Ausgabebereichs verwendet werden soll.

**KEY-NAME =**

Nennt den Schlüssel der Ausgabedatei, der beim Beschaffen des Ausgabesatzes benutzt werden soll.

**KEY-NAME = \*PRIMARY**

Der Primärschlüssel ist zu verwenden.

**KEY-NAME = <name 1..8>**

Der Name des zu verwendenden Sekundärschlüssels ist anzugeben.

Existiert der angegebene Sekundärschlüssel nicht, wird der Konvertierungsschritt mit einer Meldung abgebrochen.

**KEY-VALUE = \*BY-INPUT-RECORD (...)**

Gibt an, daß sich der Schlüssel für den Zugriff auf die Ausgabedatei im Eingabesatz befindet. Als Schlüssellänge wird die Länge des unter KEY-NAME angegebenen Schlüssels der Ausgabedatei verwendet.

**KEY-POSITION = \*STD**

Besitzt die Eingabedatei variables Satzformat, so gilt KEY-POSITION=5, ansonsten KEY-POSITION=1.

**KEY-POSITION = <integer 1.. 32768>**

Gibt die Position an, ab der im Eingabesatz das Schlüsselfeld beginnt.

Wird in der Anweisung SET-RECORD-MAPPING für eine Ausgabedatei der Operand FILLER=\*OUTPUT angegeben, so sind für diese Ausgabedatei die Anweisungen SET-GROUP-ATTRIBUTES und SET-PAGE-LAYOUT nicht zugelassen. Falls sie dennoch angegeben werden, wird der Anwender durch die Meldung PER0056 darauf hingewiesen.

Wenn in der Anweisung ASSIGN-OUTPUT-FILE der Operand OVERWRITE=\*NO angegeben ist und für eine Ausgabedatei der Operand FILLER=\*OUTPUT vereinbart ist, wird dies als Widerspruch erkannt und mit der Meldung PER0056 als Fehler abgelehnt.

In der Anweisung SET-RECORD-MAPPING ist die Angabe von FILLER=\*OUTPUT mit der Angabe von OUTPUT-FIELDS=\*COMPLETE-RECORD unverträglich und wird mit der Meldung PER0087 als Fehler abgelehnt.

**OUTPUT-FIELDS =**

Baut den Ausgabesatz aus Teilen des Eingabesatzes, Kennwörtern und Literalen auf. Bei unterschiedlichen Angaben zu Länge und Format von Ein- und Ausgabefeld ist zu beachten, daß PERCON Umwandlungen vornimmt.

Bei Ausgabedateien mit RECORD-FORMAT=\*VARIABLE ist zu beachten:

- In der SET-RECORD-MAPPING-Anweisung dürfen beim Übertragen von Daten die ersten 4 Zeichen des Ausgabesatzes nicht angesprochen werden, da diese das Satz-längenfeld belegt (wird von PERCON versorgt).
- Die Ausgabesatzlänge wird bestimmt durch das letzte Feld im Ausgabesatz, das im Operand OUTPUT-FIELDS angegeben wurde, oder durch den Operanden MIN-RECORD-LENGTH. Ohne Operand OUTPUT-FIELDS wird die Eingabesatzlänge bzw. die Angabe im Operanden MIN-RECORD-LENGTH übernommen.

**OUTPUT-FIELDS = \*COMPLETE-RECORD**

Der Eingabesatz wird in die Ausgabedatei übernommen. Ist der Operand FILLER ungleich INPUT, werden alle Zeichen des Eingabesatzes durch das Füllzeichen ersetzt.

**OUTPUT-FIELDS = \*FIELD (...)**

Übernimmt ein Feld des Eingabesatzes (Eingabefeld) in ein Feld des Ausgabesatzes (Ausgabefeld). Das angegebene Satzfeld muß innerhalb des Eingabesatzes liegen.

**INPUT-POSITION = <integer 1..32768>**

Anfangsposition des Eingabefeldes bezogen auf den Anfang des Eingabesatzes.

**INPUT-LENGTH = 1 / <integer 1..32767> / \*RECORD-LENGTH (...)**

Länge des Eingabefeldes.

**INPUT-LENGTH = \*RECORD-LENGTH (...)**

Die Satzlänge des aktuellen Eingabesatzes wird verwendet; bei variablem Satzformat ohne Satzlängenfeld. Dieser Operand darf nur bei INPUT-FORMAT=\*CHARACTER angegeben werden.

**REDUCTION = 0 / <integer 0..32767>**

Gibt an, um wieviele Zeichen die aktuelle Eingabesatzlänge verkürzt werden soll.

**INPUT-FORMAT = \*CHARACTER / \*ZONED-DECIMAL / \*PACKED-DECIMAL**

Format des Eingabefeldes (siehe Seite 75 ff).

Das Feld des Eingabesatzes, welches mittels Maske aufbereitet werden soll, darf bei INPUT-FORMAT=\*ZONED-DECIMAL nicht länger als 255, bei INPUT-FORMAT=\*PACKED-DECIMAL nicht länger als 128 byte sein.

**OUTPUT-POSITION = <integer 1..32768>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 1..32767> / \*RECORD-LENGTH (...)**

Länge des Ausgabefeldes.

**OUTPUT-LENGTH = \*STD**

Als Standardwert wird die INPUT-LENGTH genommen.

Ausnahmen bei folgenden Kombinationen:

OUTPUT-FORMAT	CHARACTER INPUT-LENGTH ≠ *RECORD-LENGTH	INPUT-FORMAT	
		ZONED-DECIMAL	PACKED-DECIMAL
HEXADECIMAL	2 * INPUT-LENGTH		
BINARY	8 * INPUT-LENGTH		
SIGNED-DECIMAL		INPUT-LENGTH + 1	2 * INPUT-LENGTH
DECIMAL		INPUT-LENGTH + 1	2 * INPUT-LENGTH
ZONED-DECIMAL		INPUT-LENGTH	2 * INPUT-LENGTH - 1
PACKED-DECIMAL		INPUT-LENGTH / 2 + 1	
<c-string> (Maske)		Maskenlänge	Maskenlänge

**OUTPUT-LENGTH = \*RECORD-LENGTH (...)**

Die Satzlänge wird festgelegt. Dieser Operand darf nur bei INPUT-FORMAT=\*CHARACTER angegeben werden.

**REDUCTION = 0 / <integer 0..32767>**

Gibt an, um wieviele Zeichen die aktuelle Eingabesatzlänge verkürzt werden soll.

**OUTPUT-FORMAT = \*INPUT-FORMAT / \*CHARACTER / \*HEXADECIMAL / \*BINARY / \*TRANSLATION / \*NO-TRANSLATION / \*ZONED-DECIMAL / \*PACKED-DECIMAL / <c-string 2..256 with-low> / \*SIGNED-DECIMAL / \*DECIMAL**  
\*NO-TRANSLATION ist zu verwenden, wenn keine Umsetzung des Eingabefeldes in den Ausgabezeichensatz erfolgen soll (z.B. bei gepackten Zahlen).  
Format des übrigen Ausgabefeldes (siehe Seite 75 ff).

Erlaubt sind nur die Kombinationen:

OUTPUT-FORMAT	INPUT-FORMAT		
	CHARACTER	ZONED-DECIMAL	PACKED-DECIMAL
INPUT-FORMAT	+	+	+
CHARACTER	+		
HEXADECIMAL	+		
BINARY	+		
TRANSLATION	+		
NO-TRANSLATION	+	+	+
ZONED-DECIMAL	+	+	+
PACKED-DECIMAL		+	+
<c-string>		+	+
SIGNED-DECIMAL		+	+
DECIMAL		+	+

**Dezimale Aufbereitung von Feldern**

Die Umwandlung des binären Wertes eines Feldes in die dezimale Form ist durch die Kombination der Operanden INPUT-FORMAT=\*CHARACTER und OUTPUT-FORMAT=\*ZONED-DECIMAL möglich.

Wenn binäre Werte eines Eingabefeldes in dezimale Werte im Ausgabefeld konvertiert werden sollen, sind folgende Festlegungen zu beachten:

1. Das Eingabefeld darf maximal vier Zeichen lang sein (Operand INPUT-LENGTH). Wird ein größerer Wert angegeben, wird die Anweisung als syntaktisch falsch mit der Meldung PER0005 zurückgewiesen.
2. Das Eingabefeld wird als nicht vorzeichenbehaftet angenommen.
3. Die Länge des Ausgabefeldes (Operand OUTPUT-LENGTH) muß so gewählt werden, daß alle gültigen Stellen in das Feld passen. Ansonsten wird der Konvertierungsschritt mit der Meldung PER0042 abgebrochen.

4. Bei OUTPUT-LENGTH=\*STD wird der Wert 10 als Standard angenommen.
5. Bei INPUT-LENGTH=\*RECORD-LENGTH kann die Einhaltung der maximalen Eingabelänge von vier Byte erst bei der Verarbeitung geprüft werden. Eine Verletzung dieser Bedingung führt zum Abbruch des Konvertierungsschritts mit der Meldung PER0042.

*Beispiel:*

In einer Datei befindet sich ab Position 12 ein Feld mit der Länge 2. Der binäre Wert dieses Feldes soll im Ausgabesatz ab Position 8 in der Länge 6 in dezimaler Form erscheinen.

Inhalt des Eingabefeldes: X'02C3'

Operanden der SET-RECORD-MAPPING-Anweisung:

```
OUTPUT-FIELDS=FIELD( -
                    INPUT-POSITION =12,INPUT-LENGTH =2, -
                    OUTPUT-POSITION= 8,OUTPUT-LENGTH=6, -
                    INPUT-FORMAT  =*CHARACTER, -
                    OUTPUT-FORMAT=*ZONED-DECIMAL)
```

Inhalt des Feldes im Ausgabesatz ab Position 8: 000707

### **OUTPUT-FIELDS = \*GROUP-COUNTER(...)**

Zähler je Gruppe. GROUP-COUNTER ist für die Gruppenstufe 1 bis 8 jeder Ausgabeeinheit verfügbar. Jeder Zähler ist durch seinen Dateikettungsnamen und durch eine Gruppenstufe eindeutig bestimmt.

#### **LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ein-/Ausgabedatei, von der man einen Gruppenzähler in der Benutzerzeile verwenden will.

#### **GROUP-LEVEL = <integer 1..8>**

Angabe, welcher Gruppenzähler der mit LINK-NAME bezeichneten Ein-/Ausgabedatei verwendet werden soll.

#### **OUTPUT-POSITION = <integer 1..32767>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

#### **OUTPUT-LENGTH = \*STD / <integer 1..32767>**

Länge des Ausgabefeldes. Standardwert ist entweder 4 bei OUTPUT-FORMAT=\*ZONED-DECIMAL oder die Maskenlänge.

#### **OUTPUT-FORMAT = C 'ZZZ' / <c-string 2..256 with-low> / \*ZONE-DECIMAL-LEFT**

Format des Ausgabefeldes (siehe Seite 75 ff).

**OUTPUT-FIELDS = \*RECORD-LENGTH(...)****LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ein-/Ausgabedatei, deren aktuelle Satzlänge man im Ausgabesatz verwenden will.

**OUTPUT-POSITION = <integer 1..32767>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 1..32767>**

Länge des Ausgabefeldes.

**OUTPUT-LENGTH = \*STD**

OUTPUT-FORMAT	OUTPUT-LENGTH
ZONED-DECIMAL-LEFT	8 Zeichen
ZONED-DECIMAL	8 Zeichen
SIGNED-DECIMAL	9 Zeichen
DECIMAL	9 Zeichen
PACKED-DECIMAL	5 Zeichen
<c-string> (Maske)	Maskenlänge

**OUTPUT-FORMAT = \*ZONED-DECIMAL-LEFT / <c-string 2..256 with-low> / \*PACKED-DECIMAL / \*ZONED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL**

Format des Ausgabefeldes (siehe Seite 75 ff).

**OUTPUT-FIELDS = \*BYTE-COUNTER(...)****LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ein-/Ausgabedatei, deren aktuellen Bytezählerwert man im Ausgabesatz verwenden will.

**OUTPUT-POSITION = <integer 1..32767>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 1..32767>**

Länge des Ausgabefeldes.



**OUTPUT-LENGTH = \*STD**

OUTPUT-FORMAT	OUTPUT-LENGTH
ZONED-DECIMAL-LEFT	10 Zeichen
ZONED-DECIMAL	10 Zeichen
SIGNED-DECIMAL	11 Zeichen
DECIMAL	11 Zeichen
PACKED-DECIMAL	6 Zeichen
<c-string> (Maske)	Maskenlänge

**OUTPUT-FORMAT = \*ZONED-DECIMAL-LEFT / <c-string 2..256 with-low> / \*PACKED-DECIMAL / \*ZONED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL**  
 Format des Ausgabefeldes (siehe Seite 75 ff).

**OUTPUT-FIELDS = \*RECORD-COUNTER(...)**

**LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ein-/Ausgabedatei, deren aktuellen Satzzählerwert man im Ausgabesatz verwenden will.

**OUTPUT-POSITION = <integer 1..32767>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 1..32767>**

Länge des Ausgabefeldes.

**OUTPUT-LENGTH = \*STD**

OUTPUT-FORMAT	OUTPUT-LENGTH
ZONED-DECIMAL-LEFT	8 Zeichen
ZONED-DECIMAL	8 Zeichen
SIGNED-DECIMAL	9 Zeichen
DECIMAL	9 Zeichen
PACKED-DECIMAL	5 Zeichen
<c-string> (Maske)	Maskenlänge

**OUTPUT-FORMAT = \*ZONED-DECIMAL-LEFT / <c-string 2..256 with-low> / \*PACKED-DECIMAL / \*ZONED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL**  
 Format des Ausgabefeldes (siehe Seite 75 ff).

**OUTPUT-FIELDS = \*PAGE-COUNTER(...)****LINK-NAME = <full-filename 1..8 without-gen>**

Dateikettungsname der Ausgabedatei, deren aktuellen Seitenzählerwert man im Ausgabesatz verwenden will.

**OUTPUT-POSITION = <integer 1..32767>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 1..32767>**

Länge des Ausgabefeldes.

**OUTPUT-LENGTH = \*STD**

OUTPUT-FORMAT	OUTPUT-LENGTH
ZONED-DECIMAL-LEFT	6 Zeichen
ZONED-DECIMAL	6 Zeichen
SIGNED-DECIMAL	7 Zeichen
DECIMAL	7 Zeichen
PACKED-DECIMAL	4 Zeichen
<c-string> (Maske)	Maskenlänge

**OUTPUT-FORMAT = \*ZONED-DECIMAL-LEFT / <c-string 2..256 with-low> / \*PACKED-DECIMAL / \*ZONED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL**

Format des Ausgabefeldes (siehe Seite 75 ff).

**OUTPUT-FIELDS = \*DATE(...)**

Datum als 8stellige Zeichenfolge im Format jj-mm-tt.

**OUTPUT-POSITION = <integer 1..32761>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 8..32767>**

Länge des Ausgabefeldes.

**OUTPUT-LENGTH = \*STD**

	CENTURY = *NO	CENTURY = *YES
OUTPUT-FORMAT	OUTPUT-LENGTH	OUTPUT-LENGTH
CHARACTER	8 Zeichen	10 Zeichen
HEXADECIMAL	16 Zeichen	20 Zeichen
BINARY	64 Zeichen	80 Zeichen

**OUTPUT-FORMAT = \*CHARACTER / \*HEXADECIMAL / \*BINARY**

Format des Ausgabefeldes (siehe Seite 75 ff).

**CENTURY = \*NO / \*YES**

Bei der Angabe von CENTURY=\*YES wird das Datum als 10stellige Zeichenfolge mit der Angabe des Jahrhunderts im Format jjjj-mm-tt ausgegeben. Damit verringert sich der Maximalwert für OUTPUT-POSITION um 2 auf 32759 und der Minimalwert für OUTPUT-LENGTH erhöht sich um 2 auf 10.

**OUTPUT-FIELDS = \*TIME(...)**

Uhrzeit als 8stellige Zeichenfolge im Format hh:mm:ss.

**OUTPUT-POSITION = <integer 1..32761>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 8..32767>**

Länge des Ausgabefeldes.

**OUTPUT-LENGTH = \*STD**

OUTPUT-FORMAT	OUTPUT-LENGTH
CHARACTER	8 Zeichen
HEXADECIMAL	16 Zeichen
BINARY	64 Zeichen

**OUTPUT-FORMAT = \*CHARACTER / \*HEXADECIMAL / \*BINARY**

Format des Ausgabefeldes (siehe Seite 75 ff).

**OUTPUT-FIELDS = <c-string 1..256 with-low>(...)****OUTPUT-POSITION = <integer 1..32768>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 1..32767>**

Länge des Ausgabefeldes.

**OUTPUT-LENGTH = \*STD**

OUTPUT-FORMAT	OUTPUT-LENGTH
CHARACTER	Länge des c-string
HEXADECIMAL	2 * Länge des c-string
BINARY	8 * Länge des c-string

**OUTPUT-FORMAT = \*CHARACTER / \*HEXADECIMAL / \*BINARY**

Format des Ausgabefeldes (siehe Seite 75 ff).

**OUTPUT-FIELDS = <x-string 1..512>(...)****OUTPUT-POSITION = <integer 1..32768>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 1..32767>**

Länge des Ausgabefeldes.

**OUTPUT-LENGTH = \*STD**

OUTPUT-FORMAT	OUTPUT-LENGTH
CHARACTER	0,5 * Länge des x-string
HEXADECIMAL	Länge des x-string
BINARY	4 * Länge des x-string

**OUTPUT-FORMAT = \*CHARACTER / \*HEXADECIMAL / \*BINARY**

Format des Ausgabefeldes (siehe Seite 75 ff).

**OUTPUT-FIELDS = <integer -2147483648..2147483647>(...)****OUTPUT-POSITION = <integer 1..32768>**

Anfangsposition des Ausgabefeldes bezogen auf den Anfang des Ausgabesatzes.

**OUTPUT-LENGTH = \*STD / <integer 1..32767>**

Länge des Ausgabefeldes.

**OUTPUT-LENGTH = \*STD**

Standardwert ist die Mindestlänge, die erforderlich ist, um die auszugebende Zahl im angegebenen Format darzustellen, bei <c-string> die Maskenlänge.

**OUTPUT-FORMAT = \*ZONED-DECIMAL / <c-string 2..256 with-low> / \*PACKED-DECIMAL / \*SIGNED-DECIMAL / \*DECIMAL**

Format des Ausgabefeldes (siehe Seite 75 ff).

**CODE-TRANSLATION =**

Ist anzugeben, wenn im Operanden OUTPUT-FIELDS=\*FIELD der Strukturoperand OUTPUT-FORMAT=\*TRANSLATION verwendet wurde.

**CODE-TRANSLATION = \*NONE**

Es erfolgt keine Code-Umsetzung, auch wenn OUTPUT-FORMAT=\*TRANSLATION angegeben wurde.

**CODE-TRANSLATION = \*TABLE(...)**

Datei mit der Code-Umsetztabelle.

**FILE-NAME = <full-filename 1..54>**

Diese Datei enthält 16 Sätze, die die sedezimale Verschlüsselung von jeweils 16 Zeichen enthalten. In diese Zeichen werden die entsprechenden Zeichen des Eingabefeldes umgewandelt und in das Ausgabefeld übertragen. Die Zuordnung erfolgt über den sedezimalen Wert innerhalb der Code-Umsetztabelle.

Die Datei kann eine SAM- oder ISAM-Datei sein. Bei ISAM-Dateien muß der ISAM-Satzschlüssel am Anfang des Satzes stehen und 8 byte lang sein (bei fester Satzlänge Position 1, bei variabler Satzlänge Position 5).

**CODE-TRANSLATION = \*REPLACE-CHARACTER(...)**

Ersetzt zeichenweise.

**INPUT-CHARACTER = <c-string 1..1> / <x-string 1..2>**

Umcodierendes Zeichen.

**OUTPUT-CHARACTER = <c-string 1..1> / <x-string 1..2>**

Ersetzungszeichen für das umcodierende Zeichen.

**MIN-RECORD-LENGTH =**

Festlegen der minimalen Satzlänge für den Ausgabesatz.

Bei variablem Satzformat der Ausgabedatei berücksichtigt PERCON zusätzlich das Satzlängenfeld.

**MIN-RECORD-LENGTH = \*NONE**

Es wird keine minimale Ausgabesatzlänge gefordert.

**MIN-RECORD-LENGTH = <integer 1..32768>**

Es wird eine feste minimale Ausgabesatzlänge gefordert.

**MIN-RECORD-LENGTH = \*BY-INPUT-RECORD(...)**

Die Satzlänge des aktuellen Eingabesatzes wird verwendet; bei variablem Satzformat der Eingabedatei ohne Satzlängenfeld.

**ADDITION = <integer -32767..32767>**

Gibt an, um wieviele Zeichen die aktuelle Eingabesatzlänge verändert werden soll.

*Hinweis*

- Ist die geforderte Minimallänge größer als die maximal mögliche Länge eines Ausgabesatzes, so wird die Minimallänge auf diesen Wert gesetzt. Bei negativem Ergebnis oder Null wird für den betreffenden Satz keine Minimallänge berücksichtigt.
- Bei fester Länge der Ausgabesätze hat der Operand MIN-RECORD-LENGTH keine Bedeutung. Er wird im Rahmen der Syntaxprüfung zwar ausgewertet, bei der Verarbeitung dann aber ignoriert.
- Beim Aufbereiten von Sätzen mit der Anweisung SET-PAGE-LAYOUT oder bei einer Standard-Aufbereitung bewirkt die Angabe einer Mindestlänge, daß rechts vor dem Satz-Ende stehende Leerzeichen nicht unterdrückt werden. D.h. in diesem Fall werden Sätze beim Aufbereiten nicht gekürzt.

## START-CONVERSION

### Starten eines Konvertierungsschrittes

Beim Kopieren von Dateien und Duplizieren von Bändern dient die START-CONVERSION-Anweisung zum Starten des Übertragungsvorganges und zum Beenden des Konvertierungsschrittes.

Beim Editieren von Bändern sind zum Starten von Übertragungsvorgängen zusätzliche Angaben über die START-TAPE-PROCESSING-Anweisung erforderlich; die START-CONVERSION-Anweisung dient in diesem Modus nur zur Beendigung des Konvertierungsschrittes.

Im Gegensatz zu der Anweisung END wird der PERCON-Lauf nicht beendet. Es kann sofort ein weiterer Konvertierungsschritt durchgeführt werden.

<b>START-CONVERSION</b>

Die Anweisung START-CONVERSION hat keine Operanden.

## START-TAPE-PROCESSING

### Steuern der Bandausgabe

Diese Anweisung ist nur beim Editieren eines Bandes von einem Band auf eine Datei zulässig. Sie steuert den Umfang der Ausgabe und startet den Übertragungsvorgang.

Die Anzahl der bearbeiteten Blöcke wird erst nach der Eingabe einer der Anweisungen START-CONVERSION oder END gemeldet.

Als Ausgabemedium muß SYSLST bzw. SYSOUT (siehe ASSIGN-OUTPUT-FILE-Anweisung, Seite 101, Operanden FILE=\*SYSLST bzw. FILE=\*SYSOUT) oder eine katalogisierte Datei zugewiesen werden.

Die Ausgabe kann auf eines der genannten Ausgabemedien oder gleichzeitig auf mehrere, auch verschiedene Ausgabemedien erfolgen.

#### START-TAPE-PROCESSING

**OUTPUT-LINK-NAME** = \*STD / list-poss(2000): <full-filename 1..8 without-gen>

**INPUT-RANGE** = \*NONE / **\*BLOCKS(...)** / **\*TAPE-MARKS(...)**

**\*BLOCKS(...)**

    | **BLOCKS** = <integer 1..999999>

**\*TAPE-MARKS(...)**

    | **TAPE-MARKS** = <integer 1..999999>

**TERMINATION** = \*END-OF-TAPE / **\*DOUBLE-TAPE-MARK**

**END-POSITION** = \*LEAVE-POSITION / **\*START-POSITION**

#### OUTPUT-LINK-NAME =

Dateikettungsamen der Ausgabedateien, auf die sich diese Anweisung beziehen soll.

#### OUTPUT-LINK-NAME = \*STD

Diese Anweisung bezieht sich auf alle bisher angegebenen Ausgabedateien mit Ausnahme der „Restdateien“.

#### OUTPUT-LINK-NAME = list-poss: <full-filename 1..8 without-gen>

Soll sich die Anweisung nur auf einige Ausgabedateien beziehen, müssen die Dateikettungsamen dieser Dateien angegeben werden. Der Dateikettungsamen einer „Restdatei“ darf nicht angegeben werden. Dies wird als Fehler zurückgewiesen (PER0062).

#### INPUT-RANGE =

Das Eingabeband wird von der augenblicklichen Position gelesen. Die gelesenen Blöcke werden auf das zugewiesene Ausgabemedium übertragen.



**INPUT-RANGE = \*NONE**

Der Operandenwert, der bei TERMINATION angegeben wird, oder der dortige Standardwert hat Gültigkeit.

**INPUT-RANGE = \*BLOCKS(...)**

**BLOCKS = <integer 1..999999>**

Anzahl der zu bearbeitenden Blöcke.

**INPUT-RANGE = \*TAPE-MARKS(...)**

**TAPE-MARKS = <integer 1..999999>**

Alle bis zum Erreichen der angegebenen Anzahl von TAPE-MARKS gelesenen Blöcke werden bearbeitet.

**TERMINATION =**

Zusätzliche Begrenzung der Bearbeitung.

**TERMINATION = \*END-OF-TAPE**

Die Bearbeitung endet spätestens bei Erreichen des logischen Bandendes. Leere Dateien werden erkannt, wenn der doppelten Abschnittsmarke ein Standard-HDR3-Kennsatz vorausgeht.

**TERMINATION = \*DOUBLE-TAPE-MARK**

Das Einlesen endet spätestens bei Erreichen einer doppelten Abschnittsmarke.

**END-POSITION =**

Steuert die Positionierung des Bandes nach der Bearbeitung.

**END-POSITION = \*LEAVE-POSITION**

Das Eingabeband wird nach der Bearbeitung nicht positioniert.

**END-POSITION = \*START-POSITION**

Das Eingabeband wird nach der Bearbeitung zurückpositioniert.



---

## 5 Aufruf von PERCON als Unterprogramm

PERCON kann als Unterprogramm aufgerufen werden. Es werden folgende Einsprungpunkte angeboten:

- PERCONU  
Zum Hauptprogramm wird ein kleiner Bindemodul (PCROOT) hinzugebunden, der den Großmodul PCNSR9 aus der Modulbibliothek OMLPERCON bzw. \$.SYSLNK.PERCON.026 dynamisch nachlädt.
- PCNSR9 (=PCNSR7,PCNSR)  
Der Großmodul wird fest an das aufrufende Programm gebunden.

Der Eingang PCNSR9 soll nur verwendet werden, wenn ein Einbinden des gesamten PERCON notwendig ist.

Die Anweisungen können

- von SYSDTA gelesen oder
- vom Hauptprogramm übergeben werden.

### *Hinweis*

PERCON arbeitet mit dem SPL-Laufzeitsystem. Dabei wird ein Stack mit dem Namen PCNSRSTK angelegt. Dieser Name wird in die Binder-Lader-Struktur eingetragen und darf daher nicht von den Oberprogrammen verwendet werden.

### **Zuweisen der Bindemodulbibliothek OMLPERCON bzw. \$.SYSLNK.PERCON.026**

Die Auflösung von Externverweisen erfolgt nach dem Aufruf des Binders BINDER bzw. TSOSLNK über INCLUDE-MODULES bzw. \*RESOLVE.

Weitere Informationen zum Binden eines Programms siehe Handbuch „BINDER“ [6].

## 5.1 Versionsbezogener UP-Aufruf von PERCON

Ab BS2000/OSD-BC V2.0 können bei Verwendung von IMON-GPN die Dateien von PERCON in beliebigen Kennungen und auch unter anderen Namen abgelegt sein. Damit wird es problematisch, beim Binden mit BINDER bzw. TSOSLNK oder bei Verwendung des dynamischen Bindeladers DBL (siehe Handbuch „Bindelader-Starter“ [14]) die externe Referenz PERCONU aufzulösen.

Damit der DBL den über PERCONU angesprochenen Startermodul von PERCON finden kann, muß der Name der zugehörigen Datei im Kommando /SET-TASKLIB mitgeteilt werden. Bisher war das durch den fest vergebenen Namen OMLPERCON unproblematisch. Da sich der Benutzer aber nicht mehr darauf verlassen kann, bietet sich künftig an, über SDF-P erstellte S-Prozeduren zu nutzen.

Mit Hilfe des Kommandos

```
/SET-VARIABLE VAR = INSTALLATION-PATH -
/          (LOGICAL-ID      = 'SYSLNK' -
/          ,INSTALLATION-UNIT = 'PERCON'-
/          ,VERSION          = '*STD'-
/          ,DEFAULT-PATH-NAME = '$.SYSLNK.PERCON.026')
```

wird der Variablen VAR der Name der aktuellen PERCON-Bibliothek zugewiesen.

Im anschließenden Kommando

```
/SET-TASKLIB LIBRARY = &VAR
```

wird der auf diese Weise ermittelte Name verwendet.

Das folgende Beispiel zeigt einen Ausschnitt aus einer Prozedur, bei der nach dem erfolgreichen Übersetzen des Hauptprogramms die erzeugte \*OMF-Datei aufgerufen werden soll.

```
:
Hauptprogramm übersetzen
:

/SET-VARIABLE LIB-NAME = '$.SYSLNK.PERCON.026'
/IF (SDF-P-VERSION >= 'V02.0A00')
/  LIB-NAME = INSTALLATION-PATH -
/          (LOGICAL-ID      = 'SYSLNK' -
/          ,INSTALLATION-UNIT = 'PERCON' -
/          ,VERSION          = '*STD' -
/          ,DEFAULT-PATH-NAME = '&LIB-NAME')
/END-IF
/SET-TASKLIB LIBRARY = &LIB-NAME
/START-PROGRAM FROM-FILE=*MODULE(LIBRARY=*OMF)
:
```

In entsprechender Weise müssen S-Prozeduren aufgebaut werden, die das statische Binden mit BINDER oder auch noch TSOSLNK bewerkstelligen. Das gilt auch für den Fall, daß der Großmodul von PERCON mit dem Einsprungpunkt PCNSR9 an das Hauptprogramm gebunden werden soll.

Bei Verwendung von PERCONU wird der Startermodul von PERCON, der Bindemodul PCROOT, hinzugebunden. Dieser Startermodul arbeitet beim UP-Aufruf versionsunabhängig, d.h. er verzweigt nicht zwangsläufig zu dem Großmodul der Version, der er selbst entstammt. Er ist so ausgelegt, daß er den Aufruf der PERCON-Version veranlaßt, die zum Zeitpunkt der Ausführung vom Benutzer über Angaben im Parameterbereich festgelegt wird:

1. Läßt der Benutzer den Parameterbereich so wie er ihn in früheren PERCON-Versionen übergab, dann wird die Version wie bei einem Kommando /START-PERCON ohne Parameter VERSION aufgerufen. Durch Aufruf des Kommandos /SELECT-PRODUCT-VERSION kann der Benutzer vor dem PERCON-Aufruf auch eine andere Version als zu verwendende Version festlegen.
2. Die vom Hauptprogramm bereitzustellende Adreßleiste wird um ein Wort, das auf einen Versionsbereich zeigt, erweitert. In dem Versionsbereich kann der Benutzer die gewünschte Version mitgeben.

Die Adreßleiste hat dann folgenden Aufbau:

Länge	Bedeutung
4	Adreßkonstante, die auf den Parameterbereich zeigt.
4	Adreßkonstante, die auf den Bereich für die Rückkehrinformationen zeigt (oder Null).
4	Adreßkonstante, die auf den CCSN-Bereich zeigt (oder Null).
4	Adreßkonstante, die auf den Versionsbereich zeigt.
4	X'80000000' als Endekriterium der Adreßleiste.

Der Versionsbereich selbst ist entweder vier oder sieben Zeichen lang, wobei die Versions-Angabe linksbündig im Bereich erwartet wird.

Im ersten Fall ist nur die Angabe \*STD möglich. In diesem Fall wird wie im Fall „keine Versions-Angabe“ vorgegangen.

Im zweiten Fall ist eine Version anzugeben (z.B. 02.6A43). Die Version kann auch in einem verkürzten Format (siehe Seite 21) angegeben werden. In diesem Fall sind anstelle der fehlenden Positionen Leerzeichen zu übergeben.

Gibt es mehrere passende Versionen, so erfolgt die Auswahl nach den auf Seite 192 genannten Kriterien. Ist die Versions-Angabe falsch, wird mit einer Fehlermeldung abgebrochen.

In Betriebssystemen, in denen noch Anwendungen mit Startermodulen bis einschließlich PERCON V2.5A im Einsatz sind, muß eine \$.OMLPERCON installiert sein oder es sind die Möglichkeiten des Alias-Katalogs zu nutzen.

## 5.2 Registerkonventionen

Beim Aufruf von PERCON als Unterprogramm sind folgende Registerkonventionen zu beachten:

Register 1: Adresse der Adreßleiste

Register 13: Adresse eines Sicherstellungsbereichs von 18 Worten, der vom Hauptprogramm zur Verfügung gestellt werden muß. Der Bereich wird von PERCON als Sicherstellungsbereich, u.a. für die Register des aufrufenden Programms verwendet.

Register 14: Rücksprungadresse in das Hauptprogramm

Register 15: Adresse des Einsprungpunktes PERCONU oder PCNSR9

### *Hinweis*

PERCON führt einen eigenen STXIT-Aufruf durch. Dadurch wird ein ggf. im Hauptprogramm angelegter STXIT-Verwaltungsblock (spezifizierte Zuordnung "STXIT-Ereignis-klasse - STXIT-Routine") modifiziert und ergänzt. Die im Hauptprogramm definierten STXIT-Routinen werden nicht mehr aktiviert. (Ausnahme: SVC, RTIMER-Zuordnung wird von PERCON nicht verändert).

## 5.3 Aufbau der Adreßleiste

Die Adreßleiste, auf die die Adresse in Register 1 zeigt, besteht aus mindestens 8 byte.

Ist nur der Parameterbereich zu übergeben, hat die Adreßleiste eine Länge von 8 byte.

Byte	Länge	Bedeutung
0-3	4	Adreßkonstante, die auf den Parameterbereich zeigt
4-7	4	X'80000000' als Endekriterium der Adreßleiste

Soll ein Bereich für Rückkehrinformationen angelegt werden, hat die Adreßleiste eine Länge von mindestens 12 byte.

Byte	Länge	Bedeutung
0-3	4	Adreßkonstante, die auf den Parameterbereich zeigt
4-7	4	Adreßkonstante, die auf den Bereich für die Rückkehrinformationen zeigt
8-11	4	X'80000000' als Endekriterium der Adreßleiste

Werden die Anweisungen vom Hauptprogramm im Speicher übergeben und sind sie mit einem CCS zu verarbeiten, hat die Adreßleiste eine Länge von mindestens 16 byte.

Byte	Länge	Bedeutung
0-3	4	Adreßkonstante, die auf den Parameterbereich zeigt
4-7	4	Adreßkonstante, die auf den Bereich für die Rückkehrinformationen zeigt (oder Null)
8-11	4	Adreßkonstante, die auf den CCSN-Bereich zeigt
12-15	4	X'80000000' als Endekriterium der Adreßleiste

Soll ein Versionsbereich übergeben werden, hat die Adreßleiste eine Länge von 20 byte.

Byte	Länge	Bedeutung
0-3	4	Adreßkonstante, die auf den Parameterbereich zeigt
4-7	4	Adreßkonstante, die auf den Bereich für die Rückkehrinformationen zeigt (oder Null)
8-11	4	Adreßkonstante, die auf den CCSN-Bereich zeigt (oder Null)
12-15	4	Adreßkonstante, die auf den Versionsbereich zeigt
16-20	4	X'80000000' als Endekriterium der Adreßleiste



### Aufbau des Parameterbereichs

Der Parameterbereich muß angegeben werden, da sich in ihm das Kennzeichen für die SDF-Versorgung befindet.

Byte	Länge	Bedeutung
0-3	4	Kennzeichen für die Art der Übergabe der Anweisungen: X'00000080': Die Anweisungen werden von SYSDTA gelesen X'00000084': Die Anweisungen werden vom Hauptprogramm übergeben
4-n	beliebig	In diesem Bereich werden die Anweisungen in Form variabler Sätze abgelegt. Die einzelnen Sätze müssen auf Halbwortgrenze ausgerichtet sein.

### Aufbau des Bereichs mit Rückkehrinformation

Byte	Länge	Bedeutung
0-3	4	DVS-Meldungsnummer der letzten aufgetretenen DVS-Meldung im Fehlerfall, sonst bleibt der Inhalt unverändert
4-15	12	Hier werden alle während des Programmlaufs aufgetretenen Meldungen registriert, indem die entsprechenden Bits gesetzt werden. Einer Meldung PERnnnn entspricht das Bit nnnn innerhalb der 12 byte, gezählt von links. nnnn kann zwischen 0 und 95 liegen. Der Bereich wird nicht gelöscht.

### Aufbau des CCSN-Bereichs

Byte	Länge	Bedeutung
0-7	8	Anweisungs-CCSN

Wird den Anweisungen im Speicher kein codierter Zeichensatzname mitgegeben, werden sie ohne Beachtung eines CCS verarbeitet. Alle Anweisungen müssen demselben CCS angehören.



---

## 6 Benutzeranschlüsse

In den Anweisungen ASSIGN-INPUT-FILE, ASSIGN-OUTPUT-FILE, ASSIGN-INPUT-TAPE und SET-GROUP-ATTRIBUTES hat der Benutzer die Möglichkeit, in Benutzermodule zu verzweigen. Die Steuerung des PERCON-Laufs wird somit vorübergehend an den Benutzer übergeben.

PERCON stellt folgende Benutzeranschlüsse zur Verfügung:

- Anschluß für Kennsatzbearbeitung  
(Anweisungen: ASSIGN-INPUT-FILE, ASSIGN-OUTPUT-FILE)
- Eingabe-Anschluß (ASSIGN-INPUT-FILE-Anweisung)
- Ausgabe-Anschluß (ASSIGN-OUTPUT-FILE-Anweisung)
- Anschluß für Gruppenbearbeitung (SET-GROUP-ATTRIBUTES-Anweisung)
- Anschluß für Lese-/Längenfehlerbehandlung  
(Anweisungen: ASSIGN-INPUT-FILE, ASSIGN-INPUT-TAPE)
- Anschluß für Eröffnungsfehlerbehandlung (ASSIGN-INPUT-FILE-Anweisung)

Die Verbindung zwischen PERCON und der Adresse des Benutzermoduls wird über einen Modulnamen (1. CSECT-Name) bzw. über einen Namen innerhalb eines Moduls (ENTRY-Name) hergestellt.

Der Benutzermodul muß in einer Bibliothek stehen, die vorher mit folgendem Kommando zugewiesen wurde:

/SET-TASKLIB LIBRARY=bibliothek

## 6.1 Registerkonventionen

Wenn die Steuerung von PERCON an einen Benutzermodul übergeben wird, haben die Inhalte der Register 1, 13, 14 und 15 folgende Bedeutung:

- Register 1: Enthält die Adresse einer Folge von 4-byte-langen Adreßkonstanten, die auf Parameterbereiche zeigen. Die Anzahl der Parameterbereiche ist für die verschiedenen Anschlüsse unterschiedlich. Das Ende der Adreßleiste wird durch eine Konstante in der Form X'80000000' angezeigt.
- Register 13: Mit diesem Register übergibt PERCON dem Benutzermodul die Adresse eines 18 Worte großen Sicherstellungsbereichs. In diesem Bereich sind die PERCON-Register mit dem Makro SAVE sicherzustellen. Der Makro SAVE muß am Beginn des Benutzermoduls stehen. Der Rücksprung geschieht über den Makro RETRN (siehe Handbuch „Makroaufrufe an den Ablaufteil“ [10]). Dieser Makro stellt auch den Inhalt der PERCON-Register wieder her, indem er auf den Sicherstellungsbereich zurückgreift. Aus diesem Grund darf das Register 13 vom Benutzer nicht überschrieben werden bzw. muß vor dem Aufruf von RETRN wieder hergestellt werden.
- Register 14: Rücksprungadresse
- Register 15: Ansprungadresse des Benutzermoduls

## 6.2 Gemeinsamer Parameterbereich der Anschlüsse

Verzweigt PERCON in einen Benutzermodul, enthält Register 1 die Adresse einer Folge von Adreßkonstanten, deren Bedeutung für die einzelnen Ausgänge unterschiedlich ist. Der erste Eintrag dieser Adreßfolge zeigt jedoch immer auf den Beginn eines Parameterbereichs, der für alle Anschlüsse folgendes Aussehen hat:

Byte	Länge	Bedeutung
0-1	2	Länge des Parameterbereichs (einschließlich des Längenfeldes). Der Parameterbereich ist auf Wortgrenze ausgerichtet.
2	1	Kennzeichen des Anschlusses
3	1	Zusatzinformation zur Art des Anschlusses
4	1	Returncode, der den Benutzermodul an PERCON übergibt.
5	1	reserviert
6-13	8	Dateikettungsname der aktuellen Datei
14-21	8	CCSN des übergebenen Satzes/Blocks

Bei allen Benutzeranschlüssen gilt, daß der CCSN des Satzes/Blocks als Übergabeparameter angeboten wird. Jeglicher Datentransfer an einem Benutzeranschluß ist dann an den angegebenen CCSN gebunden.

- Eingabe-, Lesefehler-, Längenfehler-Benutzeranschluß  
Satz/Block wird im Eingabe-CCS angeboten.
- Ausgabe-, Gruppenwechsel-Benutzeranschluß  
Satz wird im Ausgabe-CCS angeboten.

Die von Benutzer-Routinen gelieferten Returncodes werden auf Gültigkeit geprüft. Zulässig sind alle im Handbuch beschriebenen Returncodes. Wird ein unzulässiger Returncode erkannt, so wird der Konvertierungsschritt mit der Meldung PER0048 abgebrochen.

## 6.3 Beschreibung der Anschlüsse

Die im folgenden beschriebenen Benutzeranschlüsse besitzen folgende Kennzeichen:

Anschluß für die ...	Kennzeichen
Kennsatzbearbeitung	X'00'
Eingabe	X'04'
Ausgabe	X'08'
Gruppenbearbeitung	X'0C'
Lese-/Längenfehlerbehandlung	X'10'
Eröffnungsfehlerbehandlung	X'14'

### 6.3.1 Anschluß für die Kennsatzbearbeitung

Wird in den Anweisungen ASSIGN-INPUT-FILE bzw. ASSIGN-OUTPUT-FILE im Operanden LABEL-EXIT ein Benutzerausgang angegeben, verzweigt PERCON nach dem Lesen bzw. vor dem Schreiben eines Kennsatzes in den angegebenen Benutzermodul.

Register 1 zeigt auf folgende Adreßleiste:

```
A(Parameterbereich)
A(FCB)
X'80000000'
```

Der Aufbau des Parameterbereichs ist auf Seite 181 beschrieben.

**Der Parameterbereich wird mit folgenden Werten versehen:**

Kennzeichen des Anschlusses: X'00'
------------------------------------

Zusatzinformation	Bedeutung
X'00'	Ausgang OPENV für UVL-Kennsätze
X'04'	Ausgang LABGN für UHL-Kennsätze
X'08'	Ausgang LABEOV für UTL-Kennsätze bei Bandwechsel
X'0C'	Ausgang LABEND für UTL-Kennsätze
X'10'	Ausgang LABERR für fehlerhafte Bandendekennsätze
X'14'	Ausgang CLOSPOS zur Positionierung des Bandes bei der CLOSE-Bearbeitung

Beschreibung der einzelnen Ausgänge, siehe Handbuch „DVS Makros“ [13], EXLST-Makro.

Bei Nicht-Standard-Kennsätzen liest/schreibt der Benutzer seine Kennsätze mit der Zugriffsmethode BTAM selbst und gibt die Steuerung an PERCON ohne Returncode zurück.

Bei Standard-Kennsätzen zeigt das Register 0 auf den Anfang des Benutzerkennsatzes, der Benutzer prüft/erstellt den Kennsatz selbst und gibt die Steuerung an PERCON mit dem Returncode X'F1' zurück, wenn die Benutzer-Kennsatzbearbeitung abgeschlossen werden soll, bzw. mit X'F2', wenn die Kennsatzbearbeitung durch PERCON fortgesetzt werden soll.

Für Dateien mit Standard-Kennsätzen wird zum LABERR-Ausgang verzweigt, wenn während der Bandendebearbeitung ein Fehler aufgetreten ist. Der Fehlercode ist aus dem ID1ECB-Feld des FCB ersichtlich.

Über den Ausgang CLOSPOS kann der Benutzer Positionierungen des Bandes zur Zeit der CLOSE-Bearbeitung vornehmen. Nachdem der Benutzer das Band mit BTAM-Makroaufrufen positioniert hat, gibt er die Steuerung ohne Returncode an PERCON zurück. (Nähere Beschreibung dieser Ausgänge siehe Handbuch „DVS Makros“ [13]).

Die Rückkehr zu PERCON ist mit folgenden Returncodes möglich:

Returncode	Bedeutung
X'00'	Spulenwechsel durchführen, als ob die richtigen EOY/EOF-Kennsätze gelesen worden wären.
X'01'	Prozeß mit Fehler beenden.
X'02'	Dateiendebearbeitung durchführen.
X'F1'	Benutzerkennsatzbearbeitung beenden.
X'F2'	Benutzerkennsatzbearbeitung durch PERCON fortsetzen.

6.3.2 Anschluß für die Eingabe

Wird in der ASSIGN-INPUT-FILE-Anweisung im Operand INPUT-EXIT der Wert \*MODULE(NAME=modulname) angegeben, verzweigt PERCON nach jedem Lesen eines Satzes aus dieser Datei in den Benutzermodul.

Register 1 zeigt auf folgende Adreßleiste:

```
A(Parameterbereich)
A(aktueller Eingabesatz)
A(Benutzereingabesatz)
X'80000000'
```

Der Aufbau des Parameterbereichs ist auf Seite 181 beschrieben.

Der Parameterbereich wird mit folgenden Werten versehen:

Kennzeichen des Anschlusses: X'04'

Zusatzinformationen	Bedeutung
X'00'	Der Eingabesatz wird an den Benutzermodul übergeben.
X'04'	Ende der Eingabedatei. Es wird kein Satz übergeben. In diesem Fall sind nur die Returncodes X'0C' bis X'18' von Bedeutung.



Der Benutzer übergibt im Returncode des Parameterbereichs Informationen an PERCON.

Returncode	Bedeutung
X'00'	Satz übernehmen. PERCON übernimmt den zur Verfügung gestellten Satz. Dieser kann vom Benutzer verändert worden sein. Bei Sätzen variabler Länge ist es verboten, die ersten 4 Byte (=Satzlängengebiet) zu überschreiben.
X'04'	Satz ersetzen. PERCON übernimmt als Eingabesatz den Satz, der bei der Adresse des Benutzereingabesatzes hinterlegt ist. Das Adreßfeld ist vom Benutzer zu versorgen. Bei Sätzen variabler Länge erwartet PERCON in den ersten 4 Byte das Satzlängengebiet. Dieser Returncode ist für OPEN=UPDATE unzulässig. Bei RECORD-FORMAT=U wird der ersetzte Satz in der Länge des übergebenen Satzes erwartet.
X'08'	Satz nicht übernehmen. Der Satz wird nicht weiterbearbeitet. PERCON liest den nächsten Satz der Eingabedatei.
X'0C'	Satz einfügen. Der Benutzer kann hier einen Satz vor dem von PERCON gelesenen Satz einfügen. Die Adresse des einzufügenden Satzes ist als dritter Eintrag in die Adreßliste einzutragen. Bei RECORD-FORMAT=U wird der einzufügende Satz in der Länge des übergebenen Satzes erwartet. Ist der einzufügende Satz von PERCON bearbeitet worden, verzweigt PERCON erneut in den Benutzermodul und stellt den vor der Einfügung gelesenen Eingabesatz nochmals zur Verfügung. Dieser Returncode ist für OPEN=UPDATE unzulässig.
X'10'	Nicht wieder in den Benutzermodul verzweigen. PERCON steuert den Benutzerausgang für diese Eingabedatei nicht mehr an. Der bereitgestellte Satz wird wie bei Returncode X'00' übernommen.
X'14'	Lesen der Eingabedatei beenden. PERCON schließt diese Eingabedatei und eröffnet (falls vorhanden) die nächste Eingabedatei dieses Konvertierungsschrittes. Der bereitgestellte Satz wird nicht mehr übernommen.
X'18'	Konvertierungsschritt sofort fehlerhaft beenden.

6.3.3 Anschluß für die Ausgabe

Wird in der ASSIGN-OUTPUT-FILE-Anweisung im Operand OUTPUT-EXIT der Wert \*MODULE(NAME=modulname) angegeben, verzweigt PERCON direkt vor dem Ausgeben des Satzes in den Benutzermodul modulname.

Register 1 zeigt auf folgende Adreßleiste:

```
A(Parameterbereich)
A(aktueller Ausgabesatz)
X'80000000'
```

Der Aufbau des Parameterbereichs ist auf Seite 181 beschrieben.

Der Parameterbereich wird mit folgenden Werten versehen:

Kennzeichen des Anschlusses: X'08'
------------------------------------

Der Benutzer übergibt im Returncode des Parameterbereichs Informationen an PERCON.

Returncode	Bedeutung
X'00'	Satz übernehmen. Der Ausgabesatz kann vom Benutzer inhaltlich verändert werden. Eine Längenänderung ist nicht möglich. Bei Sätzen variabler Länge ist es verboten, die ersten 4 Byte zu überschreiben.
X'08'	Satz nicht übernehmen. Der Satz wird nicht in die aktuelle Ausgabedatei übernommen.
X'10'	Benutzermodul nicht mehr anspringen. PERCON steuert den Benutzerausgang für diese Ausgabedatei nicht mehr an. Der aktuelle Satz wird noch übernommen.
X'14'	Schreiben in die aktuelle Ausgabedatei beenden. PERCON gibt in die aktuelle Ausgabedatei keine Sätze mehr aus. Auch der zuletzt an den Benutzermodul übergebene Satz wird nicht mehr übernommen.
X'18'	Konvertierungsschritt sofort fehlerhaft beenden. Der aktuelle Satz wird nicht mehr übernommen.

6.3.4 Anschluß für die Gruppenbearbeitung

Wird in den Operanden GROUP-HEADER oder GROUP-TRAILER der SET-GROUP-ATTRIBUTES-Anweisung der Strukturoperand \*MODULE(NAME=modulname) angegeben, verzweigt PERCON bei Gruppenwechsel für diese Stufe in den Benutzermodul modulname. Hier kann der Benutzer eine Gruppenvorlauf- bzw. Gruppennachlaufzeile zusammen mit einem Druckervorschub-Steuerzeichen aufbauen. Wurde die vorher von PERCON aufbereitete Zeile noch nicht ausgegeben (mit \*SPACING oder \*NEW-PAGE), wird dem Benutzer die bisher aufbereitete Zeile zur Verfügung gestellt. Wurde die Vor- bzw. Nachlaufzeile vor dem Aufruf des Benutzermoduls noch nicht versorgt, enthält sie Leerzeichen (X'40').

Register 1 zeigt auf folgende Adreßleiste:

```
A(Parameterbereich)
A(Druckzeile)
X'80000000'
```

Der Aufbau des Parameterbereichs ist auf Seite 181 beschrieben.

Der Parameterbereich wird mit folgenden Werten versehen:

Kennzeichen des Anschlusses: X'0C'

Zusatzinformation	Bedeutung
X'00'	Gruppenvorlauf
X'04'	Gruppennachlauf

Der Benutzer übergibt im Returncode des Parameterbereichs Informationen an PERCON.

Returncode	Bedeutung
X'00'	Aufbereitete Druckzeile ausgeben.
X'04'	Nicht wieder in den Benutzermodul verzweigen. PERCON steuert den Benutzerausgang für diese Ausgabedatei nicht mehr an.
X'08'	Beenden der Ausgabe in diese Druckdatei. PERCON stößt die Nachlauf-Routinen für sämtliche eröffneten Gruppenstufen an und beendet die Ausgabe in diese Druckdatei.

Die von PERCON übergebene Druckzeile ist 205 Zeichen lang. Das 1. Zeichen ist ein Druckervorschub-Steuerzeichen, die restlichen 204 Zeichen sind die Datenzeichen der Vor- bzw. Nachlaufzeile. Wird das Druckersteuerzeichen vom Benutzermodul versorgt, ist es,

unabhängig von dem im Operand PRINT-CONTROL des SET-FILE-LINK-Kommandos angegebenen Format, als EBCDIC-Steuerzeichen zu übergeben. Die Bedeutung der einzelnen Bits des EBCDIC-Steuerzeichens kann der folgenden Auflistung entnommen werden.

Die maximale Länge einer Gruppenwechselzeile ergibt sich aus dem Zeilenvorspann (siehe SET-PAGE-LAYOUT-Anweisung, Seite 142) und den Operanden COLUMN-SIZE, LINE-SIZE, OUTPUT-FORMAT der SET-PAGE-LAYOUT-Anweisung. Dahinter liegende Daten werden nicht übernommen.

### EBCDIC-Steuerzeichen

Die einzelnen Bytes haben folgende Bedeutung:

#### Zeilenvorschub:

vor	nach	dem Drucken
Byte	Byte	Anzahl der Zeilen
X'40'	X'00' *	kein Zeilenvorschub
X'41'	X'01'	1 Zeile
X'42'	X'02'	2 Zeilen
X'43'	X'03'	3 Zeilen
.	.	.
.	.	.
.	.	.
X'4F'	X'0F'	15 Zeilen

\* Mit dem Steuerbyte X'00' kann der Vorschub nur einmal unterdrückt werden.

In dieser Übersicht ist die Zeile, die automatisch vorgeschoben wird, noch nicht berücksichtigt.

Kanalvorschub:

vor	nach	dem Drucken
Byte	Byte	Vorschub zum Kanal
X'C0'	X'80'	ist nicht zulässig
X'C1'	X'81'	1
X'C2'	X'82'	2
X'C3'	X'83'	3
.	.	.
.	.	.
.	.	.
X'C8'	X'88'	8
X'CA'	X'8A'	10
X'CB'	X'8B'	11

6.3.5 Anschluß für die Lese-/Längenfehlerbehandlung

Ist in den Anweisungen ASSIGN-INPUT-FILE oder ASSIGN-INPUT-TAPE einer der Operanden PARITY-ERROR, LENGTH-ERROR oder INPUT-ERROR mit der Angabe \*MODULE(NAME=modulname) gesetzt worden, verzweigt PERCON bei Lese- oder Längenfehlern in den Benutzermodul modulname.

Register 1 zeigt auf folgende Adreßleiste:

```
A(Parameterbereich)
A(fehlerhafter Block)
X'80000000'
```

Der Aufbau des Parameterbereichs ist auf Seite 181 beschrieben.

Der Parameterbereich wird mit folgenden Werten versehen:

Kennzeichen des Anschlusses: X'10'
------------------------------------

Zusatzinformation	Bedeutung
X'00'	Längenfehler
X'04'	Lesefehler

Der Benutzer übergibt im Returncode des Parameterbereichs Informationen an PERCON.

Returncode	Bedeutung
X'00'	Fehler ignorieren. PERCON übergibt an das DVS die Information, daß der Fehler ignoriert werden soll. Der fehlerhafte Block wird vom DVS zur Verfügung gestellt und wird bearbeitet, als wäre er korrekt gelesen worden. Die Bearbeitung wird normal fortgesetzt.
X'04'	Block übergehen. PERCON übergibt an das DVS die Information, daß der fehlerhafte Block übergangen werden soll. Das DVS stellt den nächsten Block zur Verfügung. Die Bearbeitung wird normal fortgesetzt.
X'08'	Bearbeitung der Datei abbrechen. Die Bearbeitung der Eingabedatei wird abgebrochen. Alle Ausgabedateien werden geschlossen. Der Konvertierungsschritt wird fehlerhaft beendet.

### 6.3.6 Anschluß für die Eröffnungsfehlerbehandlung

Wird in der ASSIGN-INPUT-FILE-Anweisung der Operand OPEN-ERROR = \*MODULE(NAME=modulname) angegeben, verzweigt PERCON bei Eröffnungsfehlern einer Eingabedatei in den Benutzermodul modulname.

Register 1 zeigt auf folgende Adreßleiste:

```
A(Parameterbereich)
X'80000000'
```

Der Aufbau des Parameterbereichs ist auf Seite 181 beschrieben.

**Der Parameterbereich wird mit folgenden Werten versehen:**

Kennzeichen des Anschlusses: X'14'
------------------------------------

Der Benutzer übergibt im Returncode des Parameterbereichs Informationen an PERCON.

Returncode	Bedeutung
X'00'	Der Konvertierungsschritt wird fehlerhaft beendet.
X'04'	Der Konvertierungsschritt wird normal beendet.
X'08'	Die nächste Eingabedatei wird bearbeitet. Der Konvertierungsschritt wird fehlerhaft beendet.
X'0C'	Die nächste Eingabedatei wird bearbeitet. Der Konvertierungsschritt wird normal beendet.

---

## 7 Installation

Ab Version BS2000/OSD-BC V2.0 ist die Installation von Softwareprodukten nicht mehr wie bisher an fest vorgegebene Dateinamen und Kennungen gebunden. Das bedeutet, daß in Zukunft die Produkte den Standort aller Bestandteile einer Version selbst dynamisch ermitteln müssen. Als Werkzeug steht dafür das Subsystem IMON-GPN zur Verfügung.

In den BS2000-Versionen bis einschließlich BS2000/OSD-BC V1.0 wird PERCON wie bisher installiert, d.h. die Dateien von PERCON werden unter ihrem festen Namen in der Standard-Benutzerkennung installiert. Außerdem kann PERCON als Subsystem mit der Möglichkeit der Austauschbarkeit (EXCHANGE) deklariert werden.

Ab BS2000/OSD-BC V2.0 ist eine Installation von PERCON nicht mehr an feste Dateinamen und Kennungen gebunden. Das bedeutet u.a. für PERCON, sich von seinem traditionellen Dateinamen OMLPERCON zu trennen und zukünftig mit Hilfe von IMON den Standort seiner Dateien selbst zu ermitteln.

Damit kann PERCON auch als Subsystem mit den Möglichkeiten für Austauschbarkeit (EXCHANGE) und Koexistenz (COEXISTENCE) von Produktversionen eingesetzt werden.

Voraussetzung dafür ist, daß die Subsysteme DSSM (ab Version 3.5) und IMON-GPN (ab Version 1.5) installiert sind.

### *Hinweis*

Die bisherigen PERCON-Versionen einschließlich V2.5A sind nicht koexistenz-fähig, da sie noch feste Dateinamen enthalten und auch organisatorische Hilfsmittel (IMON-GPN) fehlen. D.h. PERCON V2.6A kann nicht mit seinen Vorgänger-Versionen koexistieren.

Die vollständige Koexistenz von Produktversionen (d.h. der Module von PERCON und seiner Satelliten SDF- und Meldungsdateien) ist erst ab der Version BS2000/OSD-BC V3.0 gewährleistet, da diese Funktionen erst dann durch DSSM und SSCM vollständig unterstützt werden. In BS2000/OSD-BC V2.0 ist die Möglichkeit von Koexistenz auf die Module von PERCON beschränkt.

Die vollständigen Pfadnamen aller Produktdateien befinden sich in der durch IMON erzeugten Installations-Tabelle und können von dort abgerufen werden.

Zur Ermittlung des Standortes einer Produktdatei werden drei Informationen benötigt:

1. der Produktname, der PERCON sein soll,
2. der logische Name (logical ID) der Datei (z.B. SYSLNK)
3. die Produktversion.

Bei vollständiger Versions-Angabe im VERSION-Parameter des Startkommandos oder in der Parameterleiste beim UP-Aufruf wird mit dieser Version gearbeitet (vorausgesetzt, diese Version ist ordnungsgemäß installiert). Bei fehlender (d.h. \*STD) bzw. zu wenig detaillierter Versionsangabe kommen u.U. mehrere Versionen in Frage. In diesem Fall werden vom Kommandoserver bzw. von PERCON beim UP-Aufruf alle passenden Versionen ermittelt. Die endgültige Entscheidung für eine Version wird dann unter Beachtung folgender Prioritäten getroffen:

1. Die Version wurde mit dem Kommando /SELECT-PRODUCT-VERSION vorgegeben.
2. Die höchste mit IMON installierte PERCON-Version.

Ist in einem Betriebssystem ab BS2000/OSD-BC V2.0 das Subsystem IMON-GPN nicht installiert, so wird der Großmodul von PERCON aus der Bibliothek \$.SYSLNK.PERCON.026 nachgebunden. Das gilt auch, wenn das Subsystem IMON-GPN aktiv, das Produkt PERCON aber in seiner Installations-Tabelle nicht enthalten ist.

Ist das Subsystem IMON-GPN beim Start von PERCON nicht aktiv, so wird mit der Meldung PER0106 abgebrochen.



## Produkt-Installation für die einzelnen BS2000-Versionen

Die PERCON-Version V02.6A ist unter dem Betriebssystem BS2000 ab Version V10.0A einsetzbar. Die folgende Übersicht zeigt, welche Bestandteile dieser PERCON-Version bei den einzelnen BS2000-Versionen zu installieren sind:

Datei - Name	BS2000 V10.0A	BS2000/ OSD- BC V1.0	BS2000/ OSD- BC V2.0	ab BS2000/ OSD- BC V3.0	logical ID
PERCON	X	X	X	X	SYSPRG
OMLPERCON	X	X			SYSOML
SYSLNK.PERCON.026			X	X	SYSLNK
SYSREP.PERCON.026			X	X	SYSREP
SYSSDF.PERCON.026.SYSTEM	X				SYSSDF.SYSTEM
SYSSDF.PERCON.026.TSOS	X				SYSSDF.TSOS
SYSSDF.PERCON.026.110		X			SYSSDF
SYSSDF.PERCON.026.112			X	X	SYSSDF
SYMSV.PERCON.026	X				SYMSV
SYSMES.PERCON.026		X	X	X	SYSMES
SYSSSD.PERCON.026.CL4	X				SYSSSD.CL4
SYSSSD.PERCON.026.CL5	X				SYSSSD.CL5
SYSSSD.PERCON.026.XS	X				SYSSSD.XS
SYSSSC.PERCON.026.110		X			SYSSSC
SYSSSC.PERCON.026.110.LOW		X			SYSSSC.LOW
SYSSSC.PERCON.026.112			X		SYSSSC
SYSSSC.PERCON.026.112.LOW			X		SYSSSC.LOW
SYSSSC.PERCON.026.120				X	SYSSSC
SYSSSC.PERCON.026.120.LOW				X	SYSSSC.LOW
SYSSII.PERCON.026 <sup>1)</sup>			X	X	SYSSII
SYSFGM.PERCON.026.D	X	X	X	X	SYSFGM.D
SYSFGM.PERCON.026.E	X	X	X	X	SYSFGM.E

<sup>1)</sup> Diese IMON-Verwaltungsdatei wird eventuell auch in den anderen (d.h. älteren) BS2000-Versionen benötigt, dann allerdings nur im Rahmen der Installation von PERCON.

### Ergänzende Bemerkungen:

1. Kommen ab BS2000/OSD-BC V2.0 Anwendungen zum Einsatz, die PERCON als Unterprogramm aufrufen und dabei einen Startermodul einer Version bis einschließlich PERCON V2.5A eingebunden haben, so ist zu beachten, daß diese Startermodule eine Bibliothek \$.OMLPERCON erwarten.

#### Lösungsmöglichkeiten für das Problem:

- Die betroffenen Anwendungen mit dem Startermodul der Version V2.6A neu binden.
  - Zusätzlich die Bibliothek OMLPERCON in der DEFLUID-Kennung installieren bzw. eine Kopie von SYSLNK.PERCON.026 unter dem Namen OMLPERCON in der DEFLUID-Kennung ablegen.
  - Mit den ACS-Möglichkeiten OMLPERCON als Aliasnamen für SYSLNK.PERCON.026 bereitstellen.
2. In den SYSSSC-Dateien mit der logical-ID SYSSSC wurde der Parameter MEMORY-CLASS so vereinbart, daß das Subsystem in den oberen Adreßraum (d.h. > 16 Mbyte) geladen wird. Im Gegensatz dazu wird das Subsystem PERCON bei Verwendung der SYSSSC-Dateien mit der logical-ID SYSSSC.LOW in den unteren Adreßraum (d.h. < 16 Mbyte) geladen.

### *Hinweise zur Defaultregelung*

Bis einschließlich BS2000/OSD-BC V1.0 wird die Modulbibliothek von PERCON unter dem Namen OMLPERCON installiert, ab BS2000/OSD-BC V2.0 unter dem Namen SYSLNK.PERCON.026. Wird bei einem Zugriff auf die Modulbibliothek die Anwendung der Defaultregelung erforderlich, wird entsprechend dieser Festlegung auch der Standardname für die Modulbibliothek gebildet.

## 8 Anwendungsbeispiele

Die in den nächsten Abschnitten dargestellten Beispiele laufen mit diesen beiden Eingabedateien ab:

MIT.AB.1 und MIT.AB.2

/SHOW-FILE-ATTRIBUTES FILE-NAME=MIT.AB.1, INFORMATION=ALL-ATTRIBUTES

00000003 :catid:\$userid.MIT.AB.1

```
----- HISTORY -----
CRE-DATE   = 1995-10-11  ACC-DATE   = 1995-10-11  CHANG-DATE = 1995-10-11
CRE-TIME   =   10:40:16  ACC-TIME   =   14:10:02  CHANG-TIME =   10:40:16
ACC-COUNT  = 8          S-ALLO-NUM = 0
----- SECURITY -----
READ-PASS  = NONE       WRITE-PASS = NONE       EXEC-PASS  = NONE
USER-ACC   = OWNER-ONLY ACCESS      = WRITE      ACL       = NO
AUDIT      = NONE       DESTROY     = NO        EXPIR-DATE = 1995-10-11
SP-REL-LOCK= NO        EXPIR-TIME  =   00:00:00
----- BACKUP -----
BACK-CLASS = A          SAVED-PAG  = COMPL-FILE  VERSION   = 1
MIGRATE    = ALLOWED
----- ORGANIZATION -----
FILE-STRUC = SAM        BUF-LEN    = STD(1)      BLK-CONTR  = PAMKEY
IO(USAGE)  = READ-WRITE IO(PERF)   = STD        DISK-WRITE = IMMEDIATE
REC-FORM   = (V,N)      REC-SIZE   = 0
----- ALLOCATION -----
SUPPORT    = PUB        S-ALLOC   = 3          HIGH-US-PA = 1
EXTENTS    VOLUME      DEVICE-TYPE EXTENTS    VOLUME      DEVICE-TYPE
  1        PUBFOO      D348F
NUM-OF-EXT = 1
:catid:    PUBLIC:      1 FILE RES=           3 FREE=           2 REL=           0 PAGES
/
```

### Inhalt der Datei MIT.AB.1

BACH	JOHANN	MUENCHEN	BRUNNENSTR.4	AB 1
BERGER	NORBERT	MUENCHEN	TORWEG 10	AB 2
FINK	SUSANNE	NUERNBERG	RINGSTR. 11	AB 2
GROEBL	WOLFGANG	BASEL	SONNENLAENG 7	AB 1
KOLL	MONIKA	FRANKFURT	IN DER BREITE 61	AB 3
LIEDL	ERIKA	MUENCHEN	BAUMALLEE 8	AB 1
PRIES	ALFONS	MUENCHEN	TAL 4	AB 1
WAGNER	RICHARD	MUENCHEN	AM TEICH 24	AB 3

```
/SHOW-FILE-ATTRIBUTES FILE-NAME=MIT.AB.2, INFORMATION=ALL-ATTRIBUTES
0000003 :catid:$userid.MIT.AB.2

----- HISTORY -----
CRE-DATE   = 1995-10-11  ACC-DATE   = 1995-10-11  CHANG-DATE = 1995-10-11
CRE-TIME   = 10:40:19    ACC-TIME   = 14:10:03  CHANG-TIME = 10:40:19
ACC-COUNT  = 8           S-ALLO-NUM = 0
----- SECURITY -----
READ-PASS  = NONE        WRITE-PASS = NONE        EXEC-PASS  = NONE
USER-ACC   = OWNER-ONLY  ACCESS    = WRITE        ACL         = NO
AUDIT      = NONE        DESTROY   = NO          EXPIR-DATE = 1995-10-11
SP-REL-LOCK= NO          EXPIR-TIME = 00:00:00
----- BACKUP -----
BACK-CLASS = A           SAVED-PAG  = COMPL-FILE  VERSION    = 1
MIGRATE    = ALLOWED
----- ORGANIZATION -----
FILE-STRUC = SAM         BUF-LEN    = STD(1)        BLK-CONTR  = PAMKEY
IO(USAGE)  = READ-WRITE  IO(PERF)   = STD          DISK-WRITE = IMMEDIATE
REC-FORM   = (V,N)       REC-SIZE    = 0
----- ALLOCATION -----
SUPPORT    = PUB         S-ALLOC    = 3           HIGH-US-PA = 1
EXTENTS    VOLUME       DEVICE-TYPE  EXTENTS    VOLUME       DEVICE-TYPE
1          PUBFOO       D348F
NUM-OF-EXT = 1
:catid:    PUBLIC:      1 FILE RES=      3 FREE=      2 REL=      0 PAGES
/
```

**Inhalt der Datei MIT.AB.2**

BAUER	FRANK	HAMBURG	WALDWEG 29	AB 4
HOERMANN	MORITZ	MUENCHEN	SONNENSTR. 149	AB 4
RAVEL	ROBERT	MUENCHEN	LEOPOLDSTR. 74	AB 4
SONNTAG	ELLEN	KOELN	WILHELM-BUSCH-STR. 8	AB 4

**Hinweis zur Spalteneinteilung für MIT.AB.1 und MIT.AB.2**

Spalte				
5.....	17.....	29.....	44.....	69.....
Name	Vorname	Ort	Straße	Abteilung

## 8.1 Umsetzen zweier SAM-Dateien in eine ISAM-Datei und Ausgeben auf Drucker

### Eingabe:

SAM-Datei MIT.AB.1 auf Platte

SAM-Datei MIT.AB.2 auf Platte

### Ausgabe:

ISAM-Datei MIT.AB,RECORD-FORMAT=FIXED,RECORD-SIZE=100 auf Platte

SAM-Datei MIT.TA,RECORD-FORMAT=VARIABLE auf Band

### Ablaufprotokoll:

```

/SET-FILE-LINK FILE-NAME=MIT.AB,- _____ (1)
/          RECORD-FORMAT=FIXED(RECORD-SIZE=100),-
/          SUPPORT=DISK,-
/          LINK-NAME=OUT1,-
/          ACCESS-METHOD=ISAM
/CREATE-FILE FILE-NAME=MIT.TA,- _____ (2)
/          SUPPORT=TAPE(VOLUME=BD0002,-
/          DEVICE-TYPE=T6250)
/SET-FILE-LINK FILE-NAME=MIT.TA,-
/          SUPPORT=TAPE,-
/          LINK-NAME=OUT2,-
/          ACCESS-METHOD=SAM
/START-PERCON _____ (3)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=MIT.AB.1),- _____ (4)
//          LINK-NAME=IN1
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=MIT.AB.2),- _____ (5)
//          LINK-NAME=IN2

```

- (1) Die Plattendatei MIT.AB wird mit ihren Dateieigenschaften zugewiesen.
- (2) Die Banddatei MIT.TA wird mit ihren Dateieigenschaften zugewiesen.
- (3) PERCON wird aufgerufen.
- (4) Die Eingabedatei MIT.AB.1 wird mit dem Dateikettungsnamen IN1 zugewiesen.
- (5) Die Eingabedatei MIT.AB.2 wird mit dem Dateikettungsnamen IN2 zugewiesen.

```

//ASSIGN-OUTPUT-FILE FILE=DISK-FILE(NAME=MIT.AB),- _____ (6)
//                               LINK-NAME=OUT1
//ASSIGN-OUTPUT-FILE FILE=TAPE-FILE(NAME=MIT.TA),- _____ (7)
//                               LINK-NAME=OUT2
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=OUT1,- _____ (8)
//                               OUTPUT-FIELDS=(RECORD-COUNTER(LINK-NAME=OUT1,-
//                               OUTPUT-POSITION=1,-
//                               OUTPUT-LENGTH=8,-
//                               OUTPUT-FORMAT=ZONED-DECIMAL),-
//                               FIELD(INPUT-POSITION=5,-
//                               INPUT-LENGTH=68,-
//                               OUTPUT-POSITION=14,-
//                               OUTPUT=LENGTH=68))
//END _____ (9)
% DMS0DE3 TAPE WITH VSN BD0002 FOR FILE :catid:$userid.MIT.TA IS _____ (10)
    MOUNTED ON DEVICE AF
% DMS0DE7 SAM FILE CLOSED: FILE NAME=:catid:$userid.MIT.TA, LINKNAME=OUT2,
    BLOCK COUNT=000001 _____ (11)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=IN1 (FILE=MIT.AB.1): 8 (12)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=IN2 (FILE=MIT.AB.2): 4
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=OUT1 (FILE=MIT.AB): 12
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=OUT2 (FILE=MIT.TA): 12
% PER0031 PERCON TERMINATED NORMALLY _____ (13)
/PRINT-FILE FILE-NAME=MIT.AB
% SCP0810 SPOOLOUT OF FILE ':catid:$userid.MIT.AB' ACCEPTED: TSN: '2203',
    PNAME: 'TINE'
/

```

- (6) Die Ausgabedatei MIT.AB wird mit dem Dateikettungsnamen OUT1 zugewiesen.
- (7) Die Ausgabedatei MIT.TA wird mit dem Dateikettungsnamen OUT2 zugewiesen.
- (8) Die Werte des Satzzählers werden in die Ausgabedatei MIT.AB übertragen.  
Der Satzzähler wird ab Byte 1 in der Länge von 8 byte in gezontem Format in den Ausgabesatz übertragen.  
Für die ISAM-Datei MIT.AB wird auf diese Weise der ISAM-Schlüssel erzeugt.  
Der Eingabesatz wird mit 68 byte Länge ab Position 14 in den Ausgabesatz geschrieben.
- (9) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.
- (10) DVS-Meldung: Das Band für die Datei MIT.TA ist montiert.
- (11) DVS-Meldung: Die SAM-Datei MIT.TA wurde geschlossen.
- (12) PERCON-Meldungen: Die Anzahl der pro Datei übertragenen Sätze wird ausgegeben. Die Dateien werden mit dem Dateikettungsnamen und dem Dateinamen aufgeführt.
- (13) PERCON wurde normal beendet.

**Ausdruck der Datei MIT.AB**

00000000	BACH	JOHANN	MUENCHEN	BRUNNENSTR.4	AB 1
00000001	BERGER	NORBERT	MUENCHEN	TORWEG 10	AB 2
00000002	FINK	SUSANNE	NUERNBERG	RINGSTR. 11	AB 2
00000003	GROEBL	WOLFGANG	BASEL	SONNENLAENG 7	AB 1
00000004	KOLL	MONIKA	FRANKFURT	IN DER BREITE 61	AB 3
00000005	LIEDL	ERIKA	MUENCHEN	BAUMALLEE 8	AB 1
00000006	PRIES	ALFONS	MUENCHEN	TAL 4	AB 1
00000007	WAGNER	RICHARD	MUENCHEN	AM TEICH 24	AB 3
00000008	BAUER	FRANK	HAMBURG	WALDWEG 29	AB 4
00000009	HOERMANN	MORITZ	MUENCHEN	SONNENSTR. 149	AB 4
00000010	RAVEL	ROBERT	MUENCHEN	LEOPOLDSTR. 74	AB 4
00000011	SONNTAG	ELLEN	KOELN	WILHELM-BUSCH-STR. 8	AB 4

## 8.2 Umsetzen und Ausgeben einer nichtkatalogisierten Banddatei

### Eingabe:

SAM-Datei MIT.TA auf Band

### Ausgabe:

- ISAM-Datei MIT.TAB  
Der ISAM-Satzschlüssel wird mit dem Satzzähler aufgebaut.  
Die Ausgabesätze werden umstrukturiert.
- Ausgabe auf SYSLST  
Für die Druckausgabe werden alle Sätze ausgewählt, in denen in Spalte 29 beginnend, C'MUENCHEN' steht.

### Ablaufprotokoll:

```

/EXPORT-FILE FILE=NAME(FILE-NAME=MIT.TA) _____ (1)
/IMPORT-FILE SUPPORT=TAPE(FILE-NAME=MIT.TA,-
/                               VOLUME=BD0002,-
/                               DEVICE-TYPE=T6250)
/SET-FILE-LINK FILE-NAME=MIT.TA,-
/               SUPPORT=TAPE,-
/               LINK-NAME=IN
/SET-FILE-LINK FILE-NAME=MIT.TAB,- _____ (2)
/               RECORD-FORMAT=FIXED(RECORD-SIZE=82),-
/               SUPPORT=DISK,-
/               LINK-NAME=OUT1
/START-PERCON _____ (3)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=TAPE-FILE(NAME=MIT.TA),- _____ (4)
//               LINK-NAME=IN
//ASSIGN-OUTPUT-FILE FILE=DISK-FILE(NAME=MIT.TAB),- _____ (5)
//               LINK-NAME=OUT1

```

- (1) Die nichtkatalogisierte Banddatei MIT.TA wird mit ihren Dateieigenschaften zugewiesen.
- (2) Die Ausgabedatei MIT.TAB wird mit ihren Dateieigenschaften zugewiesen.
- (3) PERCON wird aufgerufen.
- (4) Die nichtkatalogisierte Banddatei MIT.TA wird zugewiesen.
- (5) Die Ausgabedatei MIT.TAB wird mit dem Dateikettungsnamen OUT1 zugewiesen.



```

//ASSIGN-OUTPUT-FILE FILE=SYSLST,- _____ (6)
//                               LINK-NAME=OUT2
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=OUT1,- _____ (7)
//                               OUTPUT-FIELDS=(RECORD-COUNTER(LINK-NAME=OUT1,-
//                                               OUTPUT-POSITION=1,-
//                                               OUTPUT-LENGTH=8,-
//                                               OUTPUT-FORMAT=ZONED-DECIMAL),-
//                               ' |(OUTPUT-POSITION=10),-
//                               FIELD(INPUT-POSITION=5,-
//                                     INPUT-LENGTH=10,-
//                                     OUTPUT-POSITION=12),-
//                               ' |(OUTPUT-POSITION=24),-
//                               FIELD(INPUT-POSITION=17,-
//                                     INPUT-LENGTH=10,-
//                                     OUTPUT-POSITION=26),-
//                               ' |(OUTPUT-POSITION=37),-
//                               FIELD(INPUT-POSITION=29,-
//                                     INPUT-LENGTH=10,-
//                                     OUTPUT-POSITION=39),-
//                               ' |(OUTPUT-POSITION=51),-
//                               FIELD(INPUT-POSITION=44,-
//                                     INPUT-LENGTH=20,-
//                                     OUTPUT-POSITION=53),-
//                               ' |(OUTPUT-POSITION=75),-
//                               FIELD(INPUT-POSITION=69,-
//                                     INPUT-LENGTH=4,-
//                                     OUTPUT-POSITION=77),-
//                               ' |(OUTPUT-POSITION=82))
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=OUT2,- _____ (8)
//                               CONDITION=((29,8)=C'MUENCHEN')
//END _____ (9)

```

- (6) Die Druckausgabedatei wird mit dem Dateikettungsnamen OUT2 zugewiesen.
- (7) Mit der SET-RECORD-MAPPING-Anweisung werden der Satzzähler und bestimmte Felder des Eingabesatzes in den Ausgabesatz übertragen.
- (8) Die SELECT-INPUT-RECORDS-Anweisung wählt alle Sätze aus, die ab Spalte 29 C'MUENCHEN' enthalten.  
Die Sätze werden in die Druckdatei mit dem Dateikettungsnamen OUT2 übernommen.
- (9) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.

```

% DMS0DE3 TAPE WITH VSN BD0002 FOR FILE :catid:$userid.MIT.TA IS —— (10)
      MOUNTED ON DEVICE AF
% DMS0DE7 SAM FILE CLOSED: FILE NAME=:catid:$userid.MIT.TA, LINKNAME=IN,
      BLOCK COUNT=000001 —— (11)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=IN (FILE=MIT.TA): 12 (12)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=OUT1 (FILE=MIT.TAB): 12
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=OUT2 : 7
% PER0031 PERCON TERMINATED NORMALLY —— (13)
/PRINT-FILE FILE-NAME=MIT.TAB
% SCP0810 SPOOLOUT OF FILE ':catid:$userid.MIT.TAB' ACCEPTED: TSN: '2205',
      PNAME: 'TINE'
/

```

- (10) DVS-Meldung: Das Band mit der Archivnummer BD0002, auf dem die Eingabedatei steht, ist montiert.
- (11) DVS-Meldung: Die SAM-Datei MIT.TA wurde geschlossen.
- (12) PERCON-Meldungen: Die Anzahl der pro Datei übertragenen Sätze wird ausgegeben. Die Dateien werden mit dem Dateikettungsnamen und dem Dateinamen aufgeführt.
- (13) PERCON wurde normal beendet.

### Ausdruck der Datei MIT.TAB

00000000	BACH	JOHANN	MUENCHEN	BRUNNENSTR.4	AB 1
00000001	BERGER	NORBERT	MUENCHEN	TORWEG 10	AB 2
00000002	FINK	SUSANNE	NUERNBERG	RINGSTR. 11	AB 2
00000003	GROEBL	WOLFGANG	BASEL	SONNENLAENG 7	AB 1
00000004	KOLL	MONIKA	FRANKFURT	IN DER BREITE 61	AB 3
00000005	LIEDL	ERIKA	MUENCHEN	BAUMALLEE 8	AB 1
00000006	PRIES	ALFONS	MUENCHEN	TAL 4	AB 1
00000007	WAGNER	RICHARD	MUENCHEN	AM TEICH 24	AB 3
00000008	BAUER	FRANK	HAMBURG	WALDWEG 29	AB 4
00000009	HOERMANN	MORITZ	MUENCHEN	SONNENSTR. 149	AB 4
00000010	RAVEL	ROBERT	MUENCHEN	LEOPOLDSTR. 74	AB 4
00000011	SONNTAG	ELLEN	KOELN	WILHELM-BUSCH-STR. 8	AB 4

### Ausdruck der ausgewählten Sätze

BACH	JOHANN	MUENCHEN	BRUNNENSTR.4	AB 1
BERGER	NORBERT	MUENCHEN	TORWEG 10	AB 2
LIEDL	ERIKA	MUENCHEN	BAUMALLEE 8	AB 1
PRIES	ALFONS	MUENCHEN	TAL 4	AB 1
WAGNER	RICHARD	MUENCHEN	AM TEICH 24	AB 3
HOERMANN	MORITZ	MUENCHEN	SONNENSTR. 149	AB 4
RAVEL	ROBERT	MUENCHEN	LEOPOLDSTR. 74	AB 4

## 8.3 Ausgeben von Sätzen in eine Restdatei

Aus einer Datei sind alle Sätze mit den Orts-Angaben „MUENCHEN“ in die Datei MIT.MUENCHEN und „NUERNBERG“ in die Datei MIT.NUERNBERG zu übernehmen. Die restlichen Sätze sind in die Datei REST-DER-WELT zu bringen. Als Eingabedatei wird die Datei MIT.AB.1 verwendet.

### Ablaufprotokoll:

```

/SET-FILE-LINK FILE-NAME=MIT.MUENCHEN,SUPPORT=DISK,- _____ (1)
/                LINK-NAME=OUT1,ACCESS-METHOD=SAM
/SET-FILE-LINK FILE-NAME=MIT.NUERNBERG,SUPPORT=DISK,-
/                LINK-NAME=OUT2,ACCESS-METHOD=SAM
/SET-FILE-LINK FILE-NAME=REST-DER-WELT,SUPPORT=DISK,-
/                LINK-NAME=OUT3,ACCESS-METHOD=SAM
/START-PERCON _____ (2)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=MIT.AB.1),- _____ (3)
//                LINK-NAME=IN1
//ASSIGN-OUTPUT-FILE FILE=DISK-FILE(NAME=MIT.MUENCHEN),- _____ (4)
//                LINK-NAME=OUT1
//ASSIGN-OUTPUT-FILE FILE=DISK-FILE(NAME=MIT.NUERNBERG),- _____ (5)
//                LINK-NAME=OUT2
//ASSIGN-OUTPUT-FILE FILE=DISK-FILE(NAME=REST-DER-WELT),- _____ (6)
//                LINK-NAME=OUT3
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=OUT1,- _____ (7)
//                CONDITION=((29,9)='MUENCHEN ')
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=OUT2,- _____ (8)
//                CONDITION=((29,9)='NUERNBERG')
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=OUT3,- _____ (9)
//                CONDITION=*REMAINING-RECORDS
//END _____ (10)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='IN1' _____ (11)
%                (FILE=:catid:$userid.MIT.AB.1): 8
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='OUT1'
%                (FILE=:catid:$userid.MIT.MUENCHEN): 5
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='OUT2'
%                (FILE=:catid:$userid.MIT.NUERNBERG): 1
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='OUT3'
%                (FILE=:catid:$userid.REST-DER-WELT): 2
% PER0031 PERCON TERMINATED NORMALLY _____ (12)

```

- (1) Die drei Ausgabedateien werden mit ihren Datei-Eigenschaften mittels Kommando /SET-FILE-LINK vereinbart.
- (2) PERCON wird aufgerufen.
- (3) Die Eingabedatei MIT.AB.1 wird mit dem Dateikettungsnamen IN1 zugewiesen.
- (4) Die Ausgabedatei MIT.MUENCHEN wird mit dem Dateikettungsnamen OUT1 zugewiesen.
- (5) Die Ausgabedatei MIT.NUERNBERG wird mit dem Dateikettungsnamen OUT2 zugewiesen.
- (6) Die Ausgabedatei REST-DER-WELT wird mit dem Dateikettungsnamen OUT3 zugewiesen.
- (7) Die Anweisung SELECT-INPUT-RECORDS wählt alle Sätze aus, die ab Spalte 29 die Angabe C'MUENCHEN ' enthalten. Die Sätze werden in die Ausgabedatei mit dem Dateikettungsnamen OUT1 übernommen.
- (8) Die Anweisung SELECT-INPUT-RECORDS wählt alle Sätze aus, die ab Spalte 29 die Angabe C'NUERNBERG' enthalten. Die Sätze werden in die Ausgabedatei mit dem Dateikettungsnamen OUT2 übernommen.
- (9) Die Anweisung SELECT-INPUT-RECORDS bestimmt die Datei mit dem Dateikettungsnamen OUT3 als Restdatei, d.h. in diese Datei werden alle Eingabesätze übernommen, die ab Spalte 29 weder die Angabe C'MUENCHEN ' noch C'NUERNBERG' enthalten.
- (10) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON. Dabei werden die Sätze, die in keine der beiden Ausgabedateien mit den Dateikettungsnamen OUT1 und OUT2 ausgegeben werden, in die Ausgabedatei mit dem Dateikettungsnamen OUT3 übernommen.
- (11) PERCON-Meldungen:  
Die Anzahl der pro Datei übertragenen Sätze wird ausgegeben.
- (12) PERCON wurde normal beendet.

#### Ausdruck der Datei MIT.MUENCHEN

BACH	JOHANN	MUENCHEN	BRUNNENSTR.4	AB 1
BERGER	NORBERT	MUENCHEN	TORWEG 10	AB 2
LIEDL	ERIKA	MUENCHEN	BAUMALLEE 8	AB 1
PRIES	ALFONS	MUENCHEN	TAL 4	AB 1
WAGNER	RICHARD	MUENCHEN	AM TEICH 24	AB 3

**Ausdruck der Datei MIT.NUERNBERG**

FINK	SUSANNE	NUERNBERG	RINGSTR. 11	AB 2
------	---------	-----------	-------------	------

**Ausdruck der Datei REST-DER-WELT**

GROEBL	WOLFGANG	BASEL	SONNENLAENG 7	AB 1
KOLL	MONIKA	FRANKFURT	IN DER BREITE 61	AB 3

## 8.4 Aktualisieren Felder in einem existierenden Ausgabesatz

Als bereits existierende Ausgabedatei steht die ISAM-Datei PERSONAL mit den folgenden Eigenschaften zur Verfügung:

RECORD-FORMAT = FIXED	RECORD-SIZE = 60
KEY-POSITION = 1	KEY-LENGTH = 4

**Inhalt der Datei PERSONAL vor dem PERCON-Lauf:**

Spalte					
1-4	5-17	18-29	30-34	35-39	41-60
Pers.-Nr.	Name	Vorname	Abt.	Gehalt	Ort
0005	MUELLER	HUBERT	A	06500	MUENCHEN
0008	BREMER	KARIN	B1	04800	FREISING
0012	ILTENAU	OTTO	A23	03450	ULM
0015	ALBERTUS	ILONA	B12	02880	MUENCHEN
0023	BERGER	ALBERT	A1	04250	NUERNBERG
0036	BOETTGE	BAERBEL	K2	04300	WUERZBURG

In einem Aktualisierungslauf sollen Gehalts- und Ortsveränderungen in die Datei eingearbeitet werden. Die Änderungsdaten befinden sich in der Datei PERS.UPDATE. Mit einem dreistelligen Kennzeichen wird festgelegt, welches Feld aktualisiert werden soll. Über die Personalnummer (Schlüselfeld) wird die Beziehung zwischen Ein- und Ausgabesätzen hergestellt:

**Aufbau der Änderungssätze (variables Satzformat):**

Spalte			
5-7	8-11	12 - max. 31	
Kennz.	Pers.-Nr.	Änderungsdaten	(abhängig vom Kennzeichen)
ORT	0015	REGENSBURG	Änderung des Ortes
GEH	0008	05300	Gehalts-Änderung
ORT	0011	STUTTGART	Änderung des Ortes, aber Pers.-Nr. nicht vorhanden

**PERCON - Lauf:**

```

/SET-FILE-LINK LINK-NAME=PCOUT1,FILE-NAME=PERSONAL,- _____ (1)
/              OPEN-MODE=INOUT
/SET-FILE-LINK LINK-NAME=PCOUT2,FILE-NAME=PERSONAL,-
/              OPEN-MODE=INOUT
/START-PERCON _____ (2)
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=PERS.UPDATE) _____ (3)
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT1 _____ (4)
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT2
//SELECT-INPUT-RECORDS CONDITION=((5,3)='GEH'),- _____ (5)
//              OUTPUT-LINK-NAME=PCOUT1
//SELECT-INPUT-RECORDS CONDITION=((5,3)='ORT'),- _____ (6)
//              OUTPUT-LINK-NAME=PCOUT2
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT1,- _____ (7)
//      FILLER=*OUTPUT(KEY-NAME=*PRIMARY,-
//              KEY-VALUE=*BY-INPUT-RECORD(KEY-POSITION=8)), -
//      OUTPUT-FIELDS=FIELD(INPUT-POSITION=12,INPUT-LENGTH=5,-
//              INPUT-FORMAT=ZONED-DECIMAL,-
//              OUTPUT-POSITION=35)
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT2,- _____ (8)
//      FILLER=*OUTPUT(KEY-NAME=*PRIMARY,-
//              KEY-VALUE=*BY-INPUT-RECORD(KEY-POSITION=8)), -
//      OUTPUT-FIELDS=FIELD(INPUT-POSITION=12,-
//              INPUT-LENGTH=RECORD-LENGTH(REDUCTION=7),-
//              OUTPUT-POSITION=41,OUTPUT-LENGTH=20)
//END _____ (9)

```

- (1) Bei ISAM-Dateien können für dieselbe Ausgabedatei mehrere SET-FILE-LINK-Kommandos mit dem Eröffnungsmodus OPEN-MODE=INOUT und verschiedenen Dateikettungsamen abgesetzt werden. Über diese Dateikettungsamen werden dann unterschiedliche PERCON-Anweisungen zugeordnet.

*Hinweis*

Wenn FILLER=\*OUTPUT in der Anweisung SET-RECORD-MAPPING angegeben wird, werden die betroffenen Ausgabedateien immer mit OPEN-MODE=INOUT eröffnet, d.h. die Angabe in /SET-FILE-LINK ist nicht unbedingt erforderlich.

- (2) PERCON wird gestartet.
- (3) Die Eingabedatei PERS.UPDATE mit den Aktualisierungsdaten wird zugewiesen.
- (4) Die Ausgabedatei wird mit ihren Dateikettungsamen mehrfach zugewiesen.
- (5) Mit dieser Anweisung werden aus der Eingabedatei die Sätze selektiert, die mit dem Kennzeichen GEH eine Änderung des Feldes „Gehalt“ bewirken sollen. Über den LINK-Namen PCOUT1 wird die Verbindung zur Anweisung SET-RECORD-MAPPING, die das Feld „Gehalt“ ändert, hergestellt.

- (6) Mit dieser Anweisung werden aus der Eingabedatei die Sätze selektiert, die mit dem Kennzeichen ORT eine Änderung des Feldes „Ort“ bewirken sollen. Über den LINK-Namen PCOUT2 wird die Verbindung zur Anweisung SET-RECORD-MAPPING, die das Feld „Ort“ ändert, hergestellt.
- (7) Diese Anweisung bewirkt, daß ein Satz der Ausgabedatei zur Vorbelegung des Ausgabebereiches benutzt wird. Die Auswahl des Satzes erfolgt über das im Änderungssatz enthaltene Feld „Pers.-Nr.“. Die eigentlichen Änderungsdaten sind in das Feld „Gehalt“ des Ausgabesatzes zu übertragen.
- (8) Diese Anweisung bewirkt, daß ein Satz der Ausgabedatei zur Vorbelegung des Ausgabebereiches benutzt wird. Die Auswahl des Satzes erfolgt über das im Änderungssatz enthaltene Feld „Pers.-Nr.“. Die eigentlichen Änderungsdaten sind in das Feld „Ort“ des Ausgabesatzes zu übertragen.
- (9) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.

#### Inhalt der Datei PERSONAL nach dem PERCON-Lauf:

Geändert wurden die zwei Sätze mit den Personal-Nummern 0008 und 0015.

Spalte 1-4	5-17	18-29	30-34	35-39	41-60	
Pers.-Nr.	Name	Vorname	Abt.	Gehalt	Ort	
0005	MUELLER	HUBERT	A	06500	MUENCHEN	
0008	BREMER	KARIN	B1	05300	FREISING	(10)
0012	ILTENAU	OTTO	A23	03450	ULM	
0015	ALBERTUS	ILONA	B12	02880	REGENSBURG	(11)
0023	BERGER	ALBERT	A1	04250	NUERNBERG	
0036	BOETTGE	BAERBEL	K2	04300	WUERZBURG	

(10) In diesem Satz wurde das Feld „Gehalt“ geändert.

(11) In diesem Satz wurde das Feld „Ort“ geändert.

In der Datei PERSONAL fehlt der Satz mit der Pers.-Nr. 0011. Deshalb wird bei der Verarbeitung des entsprechenden Änderungssatzes die Meldung PER0054 ausgegeben.

## 8.5 Ausgeben einer Datei auf Band mit Umsetzen des Codes

**Eingabe:** ISAM-Datei MIT.TAB

**Ausdruck der Eingabedatei MIT.TAB.**

00000000	BACH	JOHANN	MUENCHEN	BRUNNENSTR.4	AB 1
00000001	BERGER	NORBERT	MUENCHEN	TORWEG 10	AB 2
00000002	FINK	SUSANNE	NUERNBERG	RINGSTR. 11	AB 2
00000003	GROEBL	WOLFGANG	BASEL	SONNENLAENG 7	AB 1
00000004	KOLL	MONIKA	FRANKFURT	IN DER BREITE 61	AB 3
00000005	LIEDL	ERIKA	MUENCHEN	BAUMALLEE 8	AB 1
00000006	PRIES	ALFONS	MUENCHEN	TAL 4	AB 1
00000007	WAGNER	RICHARD	MUENCHEN	AM TEICH 24	AB 3
00000008	BAUER	FRANK	HAMBURG	WALDWEG 29	AB 4
00000009	HOERMANN	MORITZ	MUENCHEN	SONNENSTR. 149	AB 4
00000010	RAVEL	ROBERT	MUENCHEN	LEOPOLDSTR. 74	AB 4
00000011	SONNTAG	ELLEN	KOELN	WILHELM-BUSCH-STR. 8	AB 4

**Ausgabe:** SAM-Datei MIT.UMS, mit umgesetztem Code

### Umsetztabelle

Diese Tabelle wird als Assemblerquellprogramm mit Konstantendefinition erstellt und mit dem Assembler-Compiler übersetzt.

```

TRANS      START
TRANS      AMODE ANY
TRANS      RMODE ANY
DC          X'000102030405060708090A0B0C0D0E0F'
DC          X'101112131415161718191A1B1C1D1E1F'
DC          X'202122232425262728292A2B2C2D2E2F'
DC          X'303132333435363738393A3B3C3D3E3F'
DC          X'404142434445464748494A4B4C4D4E5C'
DC          X'505152535455565758595A5B4F5D5E5F'
DC          X'606162636465666768696A6B6C6D6E6F'
DC          X'707172737475767778797A7B7C7D7E7F'
DC          X'808182838485868788898A8B8C8D8E8F'
DC          X'909192939495969798999A9B9C9D9E9F'
DC          X'A0A1A2A3A4A5A6A7A8A9AAABACADAEAF'
DC          X'B0B1B2B3B4B5B6B7B8B9BABBBCBDBEBF'
DC          X'C0C1C2C3C4C5C6C7C8C9CACBCCCDCECF'
DC          X'D0D1D2D3D4D5D6D7D8D9DADBDCDDDEDF'
DC          X'E0E1E2E3E4E5E6E7E8E9EAEBECEDEEEF'
DC          X'F0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF'
END        TRANS

```



Die Übersetzung der Tabelle ist der Bindemodul TRANS. Er wird in eine mit LMS erstellte Bindemodulbibliothek TRANS.LIB gebracht. PERCON greift auf den Bindemodul TRANS als Element der Bibliothek TRANS.LIB zu und benutzt TRANS als Code-Umsetztabelle. Vor dem PERCON-Aufruf muß deshalb die Bibliothek TRANS.LIB dem Betriebssystem als TA-SKLIB bekanntgegeben (zugewiesen) werden.

### Ablaufprotokoll:

```

/SET-TASKLIB LIBRARY=TRANS.LIB _____ (1)
/CREATE-FILE FILE-NAME=MIT.UMS,- _____ (2)
/      SUPPORT=TAPE(VOLUME=BD0002,-
/      DEVICE-TYPE=T6250)
/SET-FILE-LINK FILE-NAME=MIT.UMS,-
/      LINK-NAME=AUS,-
/      SUPPORT=TAPE,-
/      ACCESS-METHOD=SAM,-
/      BUFFER-LENGTH=2048
/START-PERCON _____ (3)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=MIT.TAB) _____ (4)
//ASSIGN-OUTPUT-FILE FILE=TAPE-FILE(NAME=MIT.UMS,- _____ (5)
//      CODE-TRANSLATION=MODULE(NAME=TRANS)), -
//      LINK-NAME=AUS
//END _____ (6)
% DMS0DE3 TAPE WITH VSN BD0002 FOR FILE :catid:$userid.MIT.UMS IS _____ (7)
%      MOUNTED ON DEVICE AF
% DMS0DE7 SAM FILE CLOSED: FILE NAME=:catid:$userid.MIT.UMS, LINKNAME=AUS,
%      BLOCK COUNT=000001 _____ (8)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCIN (FILE=MIT.TAB): 12 (9)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=AUS (FILE=MIT.UMS): 12
% PER0031 PERCON TERMINATED NORMALLY _____ (10)
/

```

- (1) Die Bibliothek TRANS.LIB wird als TASKLIB zugewiesen. In dieser Bibliothek steht die Code-Umsetztabelle TRANS als Bindemodul.
- (2) Die Ausgabedatei MIT.UMS wird mit ihren Dateieigenschaften zugewiesen.
- (3) PERCON wird aufgerufen.
- (4) Die Eingabedatei MIT.TAB wird zugewiesen.
- (5) Die Ausgabedatei MIT.UMS wird zugewiesen. CODE-TRANSLATION=MODULE(NAME=TRANS) bedeutet, daß die Code-Umsetztabelle, mit deren Code die Ausgabedatei geschrieben werden soll, im Bindemodul TRANS (Modulname) steht.
- (6) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.

- (7) DVS-Meldung: Das Band ist für die Ausgabedatei MIT.UMS montiert.
- (8) DVS-Meldung: Die SAM-Datei MIT.UMS wurde geschlossen.
- (9) PERCON-Meldungen: Die Anzahl der pro Datei übertragenen Sätze wird ausgegeben. Die Dateien werden mit dem Dateikettungsnamen und dem Dateinamen aufgeführt.
- (10) PERCON wurde normal beendet.

### Ausdruck der Ausgabedatei MIT.UMS

00000000	*	BACH	*	JOHANN	*	MUENCHEN	*	BRUNNENSTR.4	*	AB 1	*
00000001	*	BERGER	*	NORBERT	*	MUENCHEN	*	TORWEG 10	*	AB 2	*
00000002	*	FINK	*	SUSANNE	*	NUERNBERG	*	RINGSTR. 11	*	AB 2	*
00000003	*	GROEBL	*	WOLFGANG	*	BASEL	*	SONNENLAENG 7	*	AB 1	*
00000004	*	KOLL	*	MONIKA	*	FRANKFURT	*	IN DER BREITE 61	*	AB 3	*
00000005	*	LIEDL	*	ERIKA	*	MUENCHEN	*	BAUMALLEE 8	*	AB 1	*
00000006	*	PRIES	*	ALFONS	*	MUENCHEN	*	TAL 4	*	AB 1	*
00000007	*	WAGNER	*	RICHARD	*	MUENCHEN	*	AM TEICH 24	*	AB 3	*
00000008	*	BAUER	*	FRANK	*	HAMBURG	*	WALDWEG 29	*	AB 4	*
00000009	*	HOERMANN	*	MORITZ	*	MUENCHEN	*	SONNENSTR. 149	*	AB 4	*
00000010	*	RAVEL	*	ROBERT	*	MUENCHEN	*	LEOPOLDSTR. 74	*	AB 4	*
00000011	*	SONNTAG	*	ELLEN	*	KOELN	*	WILHELM-BUSCH-STR. 8	*	AB 4	*

## 8.6 Ein- und Ausgeben einer Datei von bzw. auf Diskette

### 8.6.1 Ausgabe auf Diskette

#### Eingabe:

Datei MIT.AB auf Platte

#### Ausgabe:

Datei MIT.DIS auf Diskette

#### Ablaufprotokoll:

```

/START-PERCON _____ (1)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=MIT.AB) _____ (2)
//ASSIGN-OUTPUT-FILE FILE=DISKETTE(VOLUME=C0000J,- _____ (3)
//                                     FILE-NAME=DIS)
//END _____ (4)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCIN (FILE=MIT.AB): 8 (5)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCOUT (FILE=PERCON.
      TPWORK.D.3327.VS0001.DIS): 8
% SCP0810 SPOOLOUT OF FILE ':catid:$userid.PERCON.TPWORK.D.3327.
      VS0001.DIS' ACCEPTED: TSN: '3331', PNAME: 'TINE'
% PER0031 PERCON TERMINATED NORMALLY _____ (6)
/

```

- (1) PERCON wird aufgerufen.
- (2) Die Eingabedatei MIT.AB wird zugewiesen.
- (3) Die Ausgabedatei DIS wird auf das Ausgabemedium Diskette zugewiesen.
- (4) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.
- (5) PERCON-Meldungen: Die Anzahl der pro Datei übertragenen Sätze wird ausgegeben. Die Dateien werden mit dem Dateikettensnamen und dem Dateinamen aufgeführt.
- (6) PERCON wurde normal beendet.

## 8.6.2 Eingabe von Diskette

### Eingabe:

Datei MIT.DIS auf Diskette

### Ausgabe:

Datei MIT.OUT auf Platte

### Ablaufprotokoll:

```
/START-PERCON _____ (1)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=DISKETTE(VOLUME=C0000J,- _____ (2)
//                                     FILE-NAME=DIS)
//ASSIGN-OUTPUT-FILE FILE=DISK-FILE(NAME=MIT.OUT) _____ (3)
//END _____ (4)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCIN (FILE=DIS): 8 _____ (5)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCOUT (FILE=MIT.OUT): 8
% PER0031 PERCON TERMINATED NORMALLY _____ (6)
/
```

- (1) PERCON wird aufgerufen.
- (2) Die Eingabedatei DIS wird auf die Diskette mit der Archivnummer C0000J zugewiesen.
- (3) Die Ausgabedatei MIT.OUT wird zugewiesen.
- (4) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.
- (5) PERCON-Meldungen: Die Anzahl der pro Datei übertragenen Sätze wird ausgegeben. Die Dateien werden mit dem Dateikettungsamen und dem Dateinamen aufgeführt.
- (6) PERCON wurde normal beendet.

## 8.7 Sortieren einer ISAM-Datei

**Eingabe:**

SAM-Datei MIT.DAT

**Ausgabe:**

ISAM-Datei MIT.LIS

**Ausdruck der Eingabedatei MIT.DAT**

FINK	SUSANNE	NUERNBERG	RINGSTR.11	AB 2
BACH	JOHANN	MUENCHEN	BRUNNENSTR.4	AB 1
GROEBL	WOLFGANG	BASEL	SONNENLAENG 7	AB 1
LIEDL	ERIKA	MUENCHEN	BAUMALLEE 8	AB 1
WAGNER	RICHARD	MUENCHEN	AM TEICH 24	AB 3
BERGER	NORBERT	MUENCHEN	TORWEG 10	AB 2
PRIES	ALFONS	MUENCHEN	TAL 4	AB 1
KOLL	MONIKA	FRANKFURT	IN DER BREITE 61	AB 3

**Ablaufprotokoll:**

```
/SET-FILE-LINK LINK-NAME=PCOUT,- _____ (1)
/ FILE-NAME=MIT.LIS,-
/ ACCESS-METHOD=ISAM(KEY-LENGTH=25,-
/ KEY-POSITION=5),-
/ OPEN-MODE=OUTIN
/START-PERCON _____ (2)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=MIT.DAT) _____ (3)
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT,- _____ (4)
// FILE=DISK-FILE(OVERWRITE=YES)
//END _____ (5)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCIN (FILE=MIT.DAT): 8 (6)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCOUT (FILE=MIT.LIS): 8
% PER0031 PERCON TERMINATED NORMALLY _____ (7)
/
```

- (1) Die Ausgabedatei MIT.LIS wird mit ihren Dateieigenschaften über ein SET-FILE-LINK-Kommando zugewiesen.
- (2) PERCON wird aufgerufen.
- (3) Die Eingabedatei MIT.DAT wird zugewiesen.

- (4) Die Ausgabedatei MIT.LIS wird mit ihren Dateieigenschaften zugewiesen.
- (5) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.
- (6) PERCON-Meldungen: Die Anzahl der pro Datei übertragenen Sätze wird ausgegeben. Die Dateien werden mit dem Dateikettungsnamen und dem Dateinamen aufgeführt.
- (7) PERCON wurde normal beendet.

**Ausdruck der Ausgabedatei MIT.LIS**

BACH	JOHANN	MUENCHEN	BRUNNENSTR.4	AB 1
BERGER	NORBERT	MUENCHEN	TORWEG 10	AB 2
FINK	SUSANNE	NUERNBERG	RINGSTR.11	AB 2
GROEBL	WOLFGANG	BASEL	SONNENLAENG 7	AB 1
KOLL	MONIKA	FRANKFURT	IN DER BREITE 61	AB 3
LIEDL	ERIKA	MUENCHEN	BAUMALLEE 8	AB 1
PRIES	ALFONS	MUENCHEN	TAL 4	AB 1
WAGNER	RICHARD	MUENCHEN	AM TEICH 24	AB 3

## 8.8 Bilden von Gruppen

### Eingabe:

SAM-Datei MIT.KONTO

### Ausgabe:

SAM-Datei MIT.BERICHT

### Ausdruck der Eingabedatei MIT.KONTO

1987	JANUAR	DM	83000
1987	JANUAR	DM	4000
1987	JANUAR	DM	17600
1987	JANUAR	DM	12110
1987	FEBRUAR	DM	16900
1987	FEBRUAR	DM	43000
1987	FEBRUAR	DM	78000
1987	APRIL	DM	93500
1987	APRIL	DM	26000
1987	JULI	DM	11450
1987	JULI	DM	98000
1987	JULI	DM	40500
1987	JULI	DM	13000
1987	SEPTEMBER	DM	32500
1987	SEPTEMBER	DM	72500
1987	DEZEMBER	DM	21500
1987	DEZEMBER	DM	73000
1987	DEZEMBER	DM	33000
1987	DEZEMBER	DM	43000
1988	JANUAR	DM	81000
1988	JANUAR	DM	92500
1988	JANUAR	DM	33300
1988	FEBRUAR	DM	99000
1988	FEBRUAR	DM	93000
1988	MAERZ	DM	14000
1988	MAERZ	DM	86700
1988	MAERZ	DM	36400
1988	MAI	DM	23000
1988	MAI	DM	23000
1988	MAI	DM	65500
1988	JUNI	DM	93000
1988	JUNI	DM	8400
1988	OKTOBER	DM	95000
1988	OKTOBER	DM	7100
1988	DEZEMBER	DM	89300
1988	DEZEMBER	DM	3000
1988	DEZEMBER	DM	43000

**Ablaufprotokoll:**

```

/SET-FILE-LINK FILE-NAME=MIT.KONTO,- _____ (1)
/          LINK-NAME=PCIN
/SET-FILE-LINK FILE-NAME=MIT.BERICHT,- _____ (2)
/          ACCESS-METHOD=SAM,-
/          LINK-NAME=PCOUT
/START-PERCON _____ (3)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//SET-GROUP-ATTRIBUTES GROUP-LEVEL=1,- _____ (4)
//          GROUP-CONTROL=FIELD(POSITION=5,-
//                                LENGTH=4),-
//          GROUP-HEADER=(SPACING(LINES=3),-
//                        'JAHRESBERICHT'(OUTPUT-POSITION=2),-
//                        FIELD(INPUT-POSITION=5,-
//                              INPUT-LENGTH=4,-
//                              INPUT-FORMAT=CHARACTER,-
//                              OUTPUT-POSITION=16),-
//                        SPACING(LINES=1)),-
//          GROUP-TRAILER=(SPACING(LINES=2),-
//                        'JAHRESSUMME VON '(OUTPUT-POSITION=2),-
//                        FIELD(INPUT-POSITION=5,-
//                              INPUT-LENGTH=4,-
//                              INPUT-FORMAT=CHARACTER,-
//                              OUTPUT-POSITION=18),-
//                        ': DM '(OUTPUT-POSITION=22),-
//                        SUM-FIELD(INPUT-POSITION=23,-
//                              INPUT-LENGTH=5,-
//                              INPUT-FORMAT=ZONED-DECIMAL,-
//                              OUTPUT-POSITION=27,-
//                              OUTPUT-LENGTH=6,-
//                              OUTPUT-FORMAT=ZONED-DECIMAL))

```

- (1) Die Eingabedatei MIT.KONTO wird mit dem Standarddateikettungsamen PCIN über ein SET-FILE-LINK-Kommando zugewiesen.
- (2) Die Ausgabedatei MIT.BERICHT wird mit dem Standarddateikettungsamen PCOUT über ein SET-FILE-LINK-Kommando zugewiesen.
- (3) PERCON wird aufgerufen.
- (4) Diese SET-GROUP-ATTRIBUTES-Anweisung legt die Gruppenstufe 1, die Jahreszahl als Gruppenbegriff und den Gruppenvor- und -nachlauf für diese Gruppe fest.



```

//SET-GROUP-ATTRIBUTES GROUP-LEVEL=2,- _____ (5)
//
//      GROUP-CONTROL=FIELD(POSITION=10,-
//                          LENGTH=9),-
//
//      GROUP-HEADER=(SPACING(LINES=1),-
//                      'MONAT:'(OUTPUT-POSITION=4),-
//                      FIELD(INPUT-POSITION=10,-
//                          INPUT-LENGTH=9,-
//                          INPUT-FORMAT=CHARACTER,-
//                          OUTPUT-POSITION=11),-
//                      SPACING(LINES=1)),-
//
//      GROUP-TRAILER=(SPACING(LINES=1),-
//                      'MONATSSUMME VON '(OUTPUT-POSITION=2),-
//                      FIELD(INPUT-POSITION=10,-
//                          INPUT-LENGTH=9,-
//                          INPUT-FORMAT=CHARACTER,-
//                          OUTPUT-POSITION=18),-
//                      ': DM '(OUTPUT-POSITION=27),-
//                      SUM-FIELD(INPUT-POSITION=23,-
//                          INPUT-LENGTH=5,-
//                          INPUT-FORMAT=ZONED-DECIMAL,-
//                          OUTPUT-POSITION=32,-
//                          OUTPUT-LENGTH=6,-
//                          OUTPUT-FORMAT=ZONED-DECIMAL))
//END _____ (6)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCIN (FILE=MIT.KONTO):
37 _____ (7)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCOUT(FILE=MIT.BERICHT): 37
% PER0031 PERCON TERMINATED NORMALLY _____ (8)
/

```

- (5) Diese SET-GROUP-ATTRIBUTES-Anweisung legt die Gruppenstufe 2, den Monat als Gruppenbegriff und den Gruppenvor- und -nachlauf für diese Gruppe fest.
- (6) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.
- (7) PERCON-Meldungen: Die Anzahl der pro Datei übertragenen Sätze wird ausgegeben. Die Dateien werden mit dem Dateikettungsnamen und dem Dateinamen aufgeführt.
- (8) PERCON wurde normal beendet.

**Ausdruck der Ausgabedatei MIT.BERICHT**

PAGE 1

JAHRESBERICHT 1987

MONAT: JANUAR

1987 JANUAR	DM 83000
1987 JANUAR	DM 4000
1987 JANUAR	DM 17600
1987 JANUAR	DM 12110

MONATSSUMME VON JANUAR : DM 116710

MONAT: FEBRUAR

1987 FEBRUAR	DM 16900
1987 FEBRUAR	DM 43000
1987 FEBRUAR	DM 78000

MONATSSUMME VON FEBRUAR : DM 137900

MONAT: APRIL

1987 APRIL	DM 93500
1987 APRIL	DM 26000

MONATSSUMME VON APRIL : DM 119500

MONAT: JULI

1987 JULI	DM 11450
1987 JULI	DM 98000
1987 JULI	DM 40500
1987 JULI	DM 13000

MONATSSUMME VON JULI : DM 162950

MONAT: SEPTEMBER

1987 SEPTEMBER	DM 32500
1987 SEPTEMBER	DM 72500

MONATSSUMME VON SEPTEMBER: DM 105000

MONAT: DEZEMBER

1987 DEZEMBER	DM 21500
1987 DEZEMBER	DM 73000
1987 DEZEMBER	DM 33000
1987 DEZEMBER	DM 43000

MONATSSUMME VON DEZEMBER : DM 170500

JAHRESSUMME VON 1987: DM 812560

## JAHRESBERICHT 1988

## MONAT: JANUAR

1988 JANUAR	DM 81000
1988 JANUAR	DM 92500
1988 JANUAR	DM 33300

MONATSSUMME VON JANUAR : DM 206800

PAGE 2

## MONAT: FEBRUAR

1988 FEBRUAR	DM 99000
1988 FEBRUAR	DM 93000

MONATSSUMME VON FEBRUAR : DM 192000

## MONAT: MAERZ

1988 MAERZ	DM 14000
1988 MAERZ	DM 86700
1988 MAERZ	DM 36400

MONATSSUMME VON MAERZ : DM 137100

## MONAT: MAI

1988 MAI	DM 23000
1988 MAI	DM 23000
1988 MAI	DM 65500

MONATSSUMME VON MAI : DM 111500

## MONAT: JUNI

1988 JUNI	DM 93000
1988 JUNI	DM 8400

MONATSSUMME VON JUNI : DM 101400

## MONAT: OKTOBER

1988 OKTOBER	DM 95000
1988 OKTOBER	DM 7100

MONATSSUMME VON OKTOBER : DM 102100

MONAT: DEZEMBER

1988 DEZEMBER DM 89300

1988 DEZEMBER DM 3000

1988 DEZEMBER DM 43000

MONATSSUMME VON DEZEMBER : DM 135300

JAHRESSUMME VON 1988: DM 986200

## 8.9 Ausgeben eines definierten Bereichs eines Bandes

### Eingabe:

Band mit der Archivnummer BD0002

### Ausgabe:

- Ausgeben des ersten Blocks des Bandes im Zeichenmodus nach SYSOUT
- Ausgeben des ersten Blocks im Hexadezimalmodus nach SYSLST

### Ablaufprotokoll:

```

/START-PERCON _____ (1)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-TAPE VOLUME=BD0002 _____ (2)
% DMS0DE3 TAPE WITH VSN BD0002 FOR FILE :catid:$userid.PERCON.
      TPWORK.D.2076.VS0001 IS MOUNTED ON DEVICE AF
//ASSIGN-OUTPUT-FILE FILE=SYSOUT,- _____ (3)
//                      LINK-NAME=OUT1
//ASSIGN-OUTPUT-FILE FILE=SYSLST,- _____ (4)
//                      LINK-NAME=OUT2
//SET-PAGE-LAYOUT OUTPUT-LINK-NAME=OUT1,- _____ (5)
//                      OUTPUT-FORMAT=CHARACTER,-
//                      LINE-SIZE=65
//CHANGE-INPUT-TAPEPOSITION DIRECTION=FORWARD(DESTINATION=TAPE-MARKS- — (6)
//                      (TAPE-MARKS=1))

```

- (1) PERCON wird aufgerufen.
- (2) Das Eingabeband mit der Archivnummer BD0002 wird zugewiesen.
- (3) SYSOUT wird als Ausgabemedium mit dem Dateikettungsnamen OUT1 zugewiesen.
- (4) SYSLST wird als Ausgabemedium mit dem Dateikettungsnamen OUT2 zugewiesen.
- (5) Für die Ausgabe nach SYSOUT wird der Zeichenmodus und die Anzahl der aufzubereitenden Zeichen pro Zeile festgelegt.
- (6) Hiermit werden Kennsätze übergangen.

```
//START-TAPE-PROCESSING INPUT-RANGE=BLOCKS(BLOCKS=1) _____ (7)
      TMCNT: 001      BLOCK: 00000001 _____ (8)
(00000) 1036008600000000 * BACH * JOHANN * MUENCHEN * BRUNN
(00065) ENSTR.4 * AB 1 *008600000001 * BERGER * NORBERT
(00130) * MUENCHEN * TORWEG 10 * AB 2 *008600000002 * FINK
(00195) * SUSANNE * NUERNBERG * RINGSTR. 11 * AB 2
(00260) *008600000003 * GROEBL * WOLFGANG * BASEL * SONNENL
(00325) AENG 7 * AB 1 *008600000004 * KOLL * MONIKA *
(00390) FRANKFURT * IN DER BREITE 61 * AB 3 *008600000005 * LIEDL
(00455) * ERIKA * MUENCHEN * BAUMALLEE 8 * AB 1 *
(00520) 008600000006 * PRIES * ALFONS * MUENCHEN * TAL 4
(00585) * AB 1 *008600000007 * WAGNER * RICHARD * MU
(00650) ENCHEN * AM TEICH 24 * AB 3 *008600000008 * BAUER
(00715) * FRANK * HAMBURG * WALDWEG 29 * AB 4 *00
(00780) 8600000009 * HOERMANN * MORITZ * MUENCHEN * SONNENSTR.
(00845) 149 * AB 4 *008600000010 * RAVEL * ROBERT * MUEN
(00910) CHEN * LEOPOLDSTR.74 * AB 4 *008600000011 * SONNTAG
(00975) * ELLEN * KOELN * WILHELM-BUSCH-STR. 8 * AB 4 *
//END _____ (9)
% PER0029 NUMBER OF PROCESSED BLOCKS FOR LINK=PCIN: 8(INCL. 7
LABELS) _____ (10)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=OUT1: 8(INCL. 7 LABELS)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=OUT2: 8(INCL. 7 LABELS)
% PER0031 PERCON TERMINATED NORMALLY _____ (11)
/
```

- (7) Ein Block des Eingabebandes wird auf die zugewiesenen Ausgabemedien ausgegeben.
- (8) Ausgabe nach SYSOUT:  
Vor den Daten werden die aktuellen Werte des Bandmarkenzählers TMCNT und des Blockzählers ausgegeben. Den Datenzeilen sind die aktuellen Werte des Bytezählers BYTCNT vorangestellt.
- (9) Die END-Anweisung beendet PERCON.
- (10) PERCON-Meldungen: Die Anzahl der für jedes Ein- und Ausgabemedium übertragenen Blöcke wird ausgegeben.
- (11) PERCON wurde normal beendet.

## Ausgabe auf Drucker

[illegible]

## 8.10 Duplizieren eines Bandes

### Eingabe:

Band mit der Archivnummer BD0002

### Ausgabe:

Band mit der Archivnummer C5414A

Band mit der Archivnummer C5426A

### Ablaufprotokoll:

```

/START-PERCON _____ (1)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-TAPE VOLUME=BD0002 _____ (2)
% DMS0DE3 TAPE WITH VSN BD0002 FOR FILE :catid:$userid.PERCON.
  TPWORK.D.2076.VS0001IS MOUNTED ON DEVICE AF
//ASSIGN-OUTPUT-TAPE VOLUME=C5414A,- _____ (3)
//                               LINK-NAME=AUS1
% DMS0DE3 TAPE WITH VSN C5414A FOR FILE :catid:$userid.PERCON.
  TPWORK.D.2076.VS0002 IS MOUNTED ON DEVICE AG
//ASSIGN-OUTPUT-TAPE VOLUME=C5426A,- _____ (4)
//                               LINK-NAME=AUS2
% DMS0DE3 TAPE WITH VSN C5426A FOR FILE :catid:$userid.PERCON.
  TPWORK.D.2076.VS0003 IS MOUNTED ON DEVICE AH
//END _____ (5)
% PER0029 NUMBER OF PROCESSED BLOCKS FOR LINK=PCIN:  8 (INCL.  7 LABELS) (6)
% PER0029 NUMBER OF PROCESSED BLOCKS FOR LINK=AUS1:  8 (INCL.  7 LABELS)
% PER0029 NUMBER OF PROCESSED BLOCKS FOR LINK=AUS2:  8 (INCL.  7 LABELS)
% PER0031 PERCON TERMINATED NORMALLY _____ (7)
/

```

- (1) PERCON wird aufgerufen.
- (2) Das Eingabeband mit der Archivnummer BD0002 wird zugewiesen.
- (3) Das Ausgabeband mit der Archivnummer C5414A und dem Dateikettungsnamen AUS1 wird zugewiesen.
- (4) Das Ausgabeband mit der Archivnummer C5426A und dem Dateikettungsnamen AUS2 wird zugewiesen.
- (5) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.
- (6) PERCON-Meldungen: Die Anzahl der pro Band übertragenen Blöcke wird ausgegeben.
- (7) PERCON wurde normal beendet.



## 8.11 Beispiel eines COBOL-Hauptprogramms

Ein COBOL-Hauptprogramm soll PERCON als Unterprogramm aufrufen.  
Die Anweisungen werden im Programm zur Verfügung gestellt.

### Ausdruck des Quellprogramms

```

IDENTIFICATION DIVISION.
PROGRAM-ID. PERCONCO.
*      BEISPIEL FUER AUFRUF VON PERCON AUS COBOL-PROGRAM.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SPECIAL-NAMES.
        TERMINAL IS SCHIRM.
DATA DIVISION.
WORKING-STORAGE SECTION.
01 PARAM. _____ (1)
    02 KONTROLL-INFORMATION      PIC S9(5) COMP SYNC VALUE 132. _____ (2)
    02 ANWEISUNG-1.
        03 LAENGE                PIC S99 COMP VALUE 44. _____ (3)
        03 FILLER                PIC S99 COMP VALUE 0.
        03 INHALT                PIC X(40) VALUE _____ (4)
                                "MODIFY-PERCON-OPTIONS SYSOUT-LOG
                                "GING=ALL".
-
    02 ANWEISUNG-2.
        03 LAENGE                PIC S99 COMP VALUE 50.
        03 FILLER                PIC S99 COMP VALUE 0.
        03 INHALT                PIC X(46) VALUE
                                "ASSIGN-INPUT-FILE FILE=DISK-FILE
                                "(NAME=MIT.TAB)".
-
    02 ANWEISUNG-3.
        03 LAENGE                PIC S99 COMP VALUE 40.
        03 FILLER                PIC S99 COMP VALUE 0.
        03 INHALT                PIC X(36) VALUE
                                "ASSIGN-OUTPUT-FILE LINK-NAME=AUS
                                "GABE".
-

```

- (1) Symbolische Adresse des Parameterbereichs.
- (2) Im 4. Byte steht das Kennzeichen für die Art der Übergabe der Anweisungen.  
X'84': Die PERCON-Anweisungen werden vom Hauptprogramm in Form variabler Sätze übergeben.
- (3) Mit LAENGE und FILLER wird das Satzlängenfeld der Anweisung definiert.
- (4) Mit INHALT wird das Feld für den Satzinhalt der Anweisung definiert.

```

02 ANWEISUNG-4.
03 LAENGE                      PIC S99 COMP VALUE 7.
03 FILLER                      PIC S99 COMP VALUE 0.
03 INHALT                      PIC X(3) VALUE "END".
01 RETCODE. _____ (5)
02 DMS-CODE                    PIC S9(5) COMP SYNC. _____ (6)
02 FEHLER-NR. _____ (7)
03 BYTE-1-2                    PIC 9(4) COMP.
03 BYTEAN-1-2                  REDEFINES BYTE-1-2.
04 BYTEAN-1                    PIC X.
04 FILLER                      PIC X.
03 BYTES-REST                  PIC X(10).
01 RESERVIERT.
02 FILLER                      PIC S9(5) COMP.
02 FILLER                      PIC S9(5) COMP.
77 HILFSFELD                   PIC 9(5).
77 BIT-2                       PIC 9.
77 VIERZEHN-RECHTS             PIC 9(5) VALUE 16384.
PROCEDURE DIVISION.
AUFRUF-PERCON SECTION.
AR-1.
  CALL "PERCONU" USING PARAM RETCODE RESERVIERT. _____ (8)
  SUBTRACT 32768 FROM BYTE-1-2.
  IF BYTEAN-1 NOT < SPACE
  MOVE 1 TO BIT-2
  ELSE MOVE ZERO TO BIT-2.
  IF BIT-2 = 0
    DISPLAY "PERCON-LAUF ERFOLGREICH" UPON SCHIRM
  ELSE
    DISPLAY "PERCON-LAUF FEHLERHAFT" UPON SCHIRM.
  STOP RUN.

```

- (5) Symbolische Adresse des Bereichs für Rücksprunginformation.
- (6) Diese vier Byte sind für die letzte DVS-Meldung reserviert.
- (7) Reservierter Bereich für die PERCON-Meldungen.
- (8) PERCON wird aufgerufen. Der Einsprungpunkt ist PERCONU.

Als Daten werden übergeben:

- Die Datengruppe PARAM, bestehend aus einem Feld, das den Code für die Aufrufstufe enthält, gefolgt von den Steueranweisungen.
- Die Datengruppe RETCODE, bestehend aus vier Feldern, kann den Return-code aufnehmen.
- Die Datengruppe RESERVIERT, bestehend aus zwei Feldern, dient für zukünftige Erweiterungen.

**Übersetzen, Binden und Aufrufen des Programms (Ablaufprotokoll)**

```

/DELETE-SYSTEM-FILE FILE-NAME=OMF
/START-PROGRAM FROM-FILE=COBOL85 _____ (9)
% BLS0500 PROGRAM 'COBOL85', VERSION 'V2.1A10' OF '1993-07-14' LOADED.
9000 Copyright (C) Siemens Nixdorf Informationssysteme AG 1993
*COMOPT LISTS=(DIAG,SOURCE) _____ (10)
9099 COMOPT LISTS=(DIAG,SOURCE)
*END COB.TEST _____ (11)
9099 END COB.TEST
9017 COMPILATION INITIATED, VERSION IS V02.1A10
9097 COMPILATION COMPLETED WITHOUT ERRORS
9004 COMPILATION OF PERCONCO USED 0.4211 CPU SECONDS
% SCP0810 SPOOLOUT OF FILE ':catid:$userid.S.SPS.3213.09.07.88251.131405'
      ACCEPTED: TSN: '3218', PNAME: 'TINE'
% EXC0392 ASSIGNMENT IS THE SAME AS SYSCMD. COMMAND IGNORED
/START-BINDER _____ (12)
% BLS0500 PROGRAM 'BINDER', VERSION 'V01.1A01' OF '1992-11-12' LOADED.
% BLS0552 COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992.
      ALL RIGHTS RESERVED
//START-LLM-CREATION INTERNAL-NAME=COB.PROG _____ (13)
//INCLUDE-MODULES LIBRARY=*OMF,ELEMENT=*ALL
//RESOLVE-BY-AUTOLINK LIBRARY=$.OMLPERCON _____ (14)
//RESOLVE-BY-AUTOLINK LIBRARY=$.SYSLNK.CRTE
//SAVE-LLM LIBRARY=COB.PROG
% BND3101 SOME EXTERNAL REFERENCES UNRESOLVED
% BND3102 SOME WEAK EXTERNS UNRESOLVED
% BND1501 LLM FORMAT: '1'
//END
% BND1101 BINDER NORMALLY TERMINATED. SEVERITY CLASS: 'UNRESOLVED EXTERNAL'

```

- (9) Der COBOL-Übersetzer wird aufgerufen.
- (10) Mit der COMOPT-Anweisung werden Fehler- und Quellprogramm-Liste angefordert.
- (11) Mit der END-Anweisung wird die Datei COB.TEST als Eingabedatei zugewiesen. Am Ende der Übersetzung wird SYSDTA automatisch wieder auf SYSCMD gesetzt.
- (12) Der BINDER wird aufgerufen.
- (13) Der Name des Programms wird bestimmt.
- (14) Die Bindemodulbibliothek OMLPERCON und das Laufzeitsystem SYSLNK.CRTE werden zugewiesen.

```

/SET-FILE-LINK LINK-NAME=AUSGABE,- _____ (15)
/          FILE-NAME=TEST,-
/          ACCESS-METHOD=ISAM(KEY-LENGTH=8,-
/                                KEY-POSITION=5),-
/          RECORD-FORMAT=VARIABLE
/START-PROGRAM FROM-FILE=*MODULE(LIBRARY=COB.PROG,ELEMENT=COBPROG1,- (16)
              RUN-MODE=ADVANCED)
% BLS0523 ELEMENT 'COBPROG1', VERSION '@' FROM LIBRARY ':catid:$userid.
          COB.PROG' IN PROCESSING
% BLS0524 LLM 'COBPROG1', VERSION ' ' OF '1995-10-12:09:13:47' LOADED.
% BLS0551 COPYRIGHT (C) XXX 1995. ALL RIGHTS RESERVED
ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=MIT.TAB) _____ (17)
ASSIGN-OUTPUT-FILE LINK-NAME=AUSGABE
END
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCIN (FILE=MIT.TAB): 12 (18)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=AUSGABE (FILE=TEST): 12
% PER0031 PERCON TERMINATED NORMALLY
PERCON-LAUF ERFOLGREICH _____ (19)
/

```

- (15) Die Ausgabedatei TEST wird mit dem Dateikettungsnamen AUSGABE und ihren Dateieigenschaften zugewiesen.
- (16) Das soeben gebundene Programm wird geladen und gestartet.
- (17) Die folgenden Anweisungen werden vom COBOL-Programm bereitgestellt.
- (18) Die Meldungen von PERCON als Unterprogramm.
- (19) Die Ausgabe des Hauptprogramms.

## 8.12 Beispiel eines Assembler-Hauptprogramms

Ein Assembler-Hauptprogramm soll PERCON als Unterprogramm aufrufen.

Die Anweisungen werden von SYSDTA gelesen.

### Ausdruck des Quellprogramms

```

PERUP    START
PERUP    AMODE ANY
PERUP    RMODE ANY
          BALR 8,0
          USING *,8
          LA 1,ATAB                ADR. PARAMETERLEISTE _____ (1)
          LA 13,SAV               ADR. SICHERSTELLUNGSBEREICH _____ (2)
          L 15,=V(PERCONU)        _____ (3)
          BALR 14,15              UNTERPROGRAMMSPRUNG _____ (4)
          TERM
          SPACE 3
SAV      DS 20F                  SICHERSTELLUNGSBEREICH _____ (5)
          SPACE
ATAB     DC A(PARAM)              PARAMETERLEISTE _____ (6)
          DC A(RETCODE)           _____ (7)
          DC X'80000000'           _____ (8)
          SPACE
RETCODE  DC F'0'                  _____ (9)
          DC 12X'00'              _____ (10)
          SPACE
PARAM    DC X'00000080'           ANWEISUNGEN VON SYSDTA _____ (11)
          END PERUP

```

- (1) Die symbolische Adresse ATAB der Adreßleiste wird in Register 1 geladen.
- (2) Die symbolische Adresse SAV des Sicherstellungsbereichs wird in Register 13 geladen.
- (3) Eine V-Konstante mit der Adresse des Einsprungpunktes PERCONU wird in Register 15 geladen.
- (4) Hier erfolgt der Sprung ins Unterprogramm. Die Rücksprungadresse wird in Register 14 geladen.
- (5) Für den Sicherstellungsbereich werden 20 Worte reserviert.

Die Adreßleiste wird definiert:

- (6) Adreßkonstante PARAM, die auf den Parameterbereich zeigt.
- (7) Adreßkonstante RETCODE, die auf den Bereich für Rücksprunginformation zeigt.

- (8) Endekriterium der Adreßleiste.

Bereich für Rücksprunginformation:

- (9) Dieser Bereich wird für die Meldungsnummer der letzten auftretenden DVS-Meldung reserviert.
- (10) In diesem Bereich werden alle während des Programmlaufes auftretenden PERCON-Meldungen im Bereich PER0000 bis PER0095 registriert.
- (11) Parameterbereich

Im 4. Byte steht X'80', das bedeutet, die Anweisungen werden von SYSDTA im SDF-Format gelesen.

### Übersetzen, Binden und Aufrufen des Programms (Ablaufprotokoll)

```

/ASSIGN-SYSDTA TO-FILE=ASS.TEST
/DELETE-SYSTEM-FILE FILE-NAME=OMF
/START-PROGRAM FROM-FILE=$ASSEMB
% BLS0500 PROGRAM 'ASSEMB', VERSION '300' OF '1989-11-03' LOADED.
V30.0A20 OF SIEMENS BS 2000 ASSEMBLER READY
GIVE ASSEMBLER OPTIONS !
FLAGS IN 00000 STATEMENTS, 000 PRIVILEGED FLAGS, 000 MNOTES
HIGHEST ERROR-WEIGHT : -
SYSTEM MACROLIBRARY : MACROLIB
ASSEMBLY TIME : 0.7149 SEC.
/ASSIGN-SYSDTA TO-FILE=*PRIMARY
/START BINDER
% BLS0500 PROGRAM 'BINDER', VERSION 'V01.1A01' OF '1992-11-12' LOADED.
% BLS0552 COPYRIGHT (C) SIEMENS NIXDORF INFORMATIONSSYSTEME AG 1992.
ALL RIGHTS RESERVED
//START-LLM-CREATION INTERNAL-NAME=ASSPROG1
//INCLUDE-MODULES LIBRARY=*OMF,ELEMENT=*ALL
//RESOLVE-BY-AUTOLINK LIBRARY=$.OMLPERCON (12)
//SAVE-LLM LIBRARY=ASS.PROG (13)
% BND1501 LLM FORMAT: '1'
//END
% BND1101 BINDER NORMALLY TERMINATED. SEVERITY CLASS: 'OK'
/SET-FILE-LINK FILE-NAME=MIT.FRA,- (14)
/ ACCESS-METHOD=SAM,-
/ LINK-NAME=AUS
/START-PROGRAM FROM-FILE=*MODULE(LIBRARY=ASS.PROG,ELEMENT=ASSPROG1, (15)
RUN-MODE=ADVANCED)
% BLS0523 ELEMENT 'ASSPROG1', VERSION '@' FROM LIBRARY ':catid:$userid.
ASS.PROG' IN PROCESSING
% BLS0524 LLM 'ASSPROG1', VERSION ' ' OF '1995-10-12:12:38:05' LOADED.
% BLS0551 COPYRIGHT (C) XXX 1995. ALL RIGHTS RESERVED

```

```

//MODIFY-PERCON-OPTIONS SYSOUT-LOGGING=ALL _____ (16)
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=MIT.AB) _____ (17)
//ASSIGN-OUTPUT-FILE FILE=DISK-FILE(NAME=MIT.FRA),- _____ (18)
//                               LINK-NAME=AUS
//SELECT-INPUT-RECORDS CONDITION=((38,9)='FRANKFURT') _____ (19)
//END _____ (20)
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=PCIN (FILE=MIT.AB): 12
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK=AUS (FILE=MIT.FRA): 1
% PER0031 PERCON TERMINATED NORMALLY _____ (21)
/

```

- (12) Die Bindemodulbibliothek OMLPERCON wird zum Binden zugewiesen.
- (13) Das übersetzte und gebundene Programm ASSPROG1 wird in die Bibliothek ASS.PROG geschrieben.
- (14) Die Ausgabedatei MIT.FRA wird mit dem Dateikettungsnamen AUS und ihren Dateieigenschaften zugewiesen.
- (15) Das Programm ASSPROG1 in der Bibliothek ASS.PROG wird aufgerufen.
- (16) Der Sprung ins Unterprogramm ist erfolgt.  
Es werden Anweisungen erwartet.  
Die MODIFY-PERCON-OPTIONS-Anweisung steuert die Ausgabe der Meldungen nach SYSOUT: Alle Meldungen werden im vollen Umfang nach SYSOUT ausgegeben.
- (17) Die Eingabedatei MIT.AB wird zugewiesen.
- (18) Die Ausgabedatei MIT.FRA wird über den Dateikettungsnamen AUS mit PERCON verknüpft.
- (19) Sätze, die ab Spalte 38 C'FRANKFURT' enthalten, werden in die Ausgabedatei übernommen.
- (20) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.
- (21) PERCON wurde normal beendet.  
Es erfolgt ein Programmsprung vom Unterprogramm ins Hauptprogramm zurück.

## 8.13 Beispiel zu formatierten Zahlen und Zähler je Gruppe

Die über SYSDTA eingegebenen Daten werden aufbereitet in die Datei LIST ausgegeben.

```
//SET-FILE-LINK LINK-NAME=PCOUT, FILE-NAME=LIST, ACCESS-METHOD=SAM (1)
//START-PERCON (2)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=SYSDTA (3)
//ASSIGN-OUTPUT-FILE FILE=DISK-FILE(NAME=LIST) (4)
//SET-RECORD-MAPPING OUTPUT-FIELDS=(GROUP-COUNTER(LINK-NAME=PCOUT,- (5)
//                                     GROUP-LEVEL=1,-
//                                     OUTPUT-POSITION=10),-
//                                     '/'(OUTPUT-POSITION=16),-
//                                     FIELD(INPUT-POSITION=5,-
//                                     INPUT-LENGTH=10,-
//                                     OUTPUT-POSITION=22),-
//                                     '/'(OUTPUT-POSITION=35),-
//                                     FIELD(INPUT-POSITION=15,-
//                                     INPUT-LENGTH=3,-
//                                     INPUT-FORMAT=ZONED-DECIMAL,-
//                                     OUTPUT-POSITION=40,-
//                                     OUTPUT-FORMAT=' ZNZ'))
//SET-GROUP-ATTRIBUTES GROUP-TRAILER=(SPACING(LINES=4),- (6)
//                                     'Gesamt Posten:'(OUTPUT-POSITION=10),-
//                                     GROUP-COUNTER(LINK-NAME=PCOUT,-
//                                     GROUP-LEVEL=1,-
//                                     OUTPUT-POSITION=25,-
//                                     OUTPUT-FORMAT=' ZZZNZ'),-
//                                     SPACING,-
//                                     'Gesamt Anzahl:'(OUTPUT-POSITION=10),-
//                                     SUM-FIELD(INPUT-POSITION=15,-
//                                     INPUT-LENGTH=3,-
//                                     INPUT-FORMAT=ZONED-DECIMAL,-
//                                     OUTPUT-POSITION=25,-
//                                     OUTPUT-FORMAT=' ZZZNZ'))
//SET-PAGE-LAYOUT HEADER-LINE=NONE (7)
//END (8)
*Anzug 014 (9)
*Hose 053
*Hemd 162
*Bluse 064
*Schuhe 136
*/EOF
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK 'PCIN' : 5
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK 'PCOUT' (FILE=LIST): 5
% PER0031 PERCON TERMINATED NORMALLY (10)
```



- (1) Die Ausgabedatei LIST wird mit dem Dateikettungsnamen PCOUT zugewiesen.
- (2) PERCON wird aufgerufen.
- (3) Die Eingabe soll über SYSDTA erfolgen.
- (4) Die Ausgabedatei LIST wird zugewiesen.
- (5) Mit der SET-RECORD-MAPPING-Anweisung werden der Gruppenzähler und bestimmte Felder des Eingabesatzes in den Ausgabesatz übertragen.
- (6) Die SET-GROUP-ATTRIBUTES-Anweisung legt Aufbau und Inhalt der Gruppennachlaufzeilen fest.
- (7) Die SET-PAGE-LAYOUT-Anweisung bestimmt, daß keine Kopfzeile ausgegeben werden soll.
- (8) Die END-Anweisung startet den Übertragungsvorgang und beendet PERCON.
- (9) Eingabe der Daten über SYSDTA.
- (10) PERCON wurde normal beendet.

**Inhalt der Ausgabedatei LIST:**

Die Positionsleiste ist nicht in der Datei enthalten und bezieht sich nur auf die Daten, d.h. Satzlängenfeld und Vorschub-Steuerzeichen werden nicht beachtet.

5	10	15	20	25	30	35	40	45	50	(Positionsleiste)
		1	/	Anzug		/		14		
		2	/	Hose		/		53		
		3	/	Hemd		/		162		
		4	/	Bluse		/		64		
		5	/	Schuhe		/		136		
Gesamt Posten:				5						
Gesamt Anzahl:				429						

## 8.14 Spezielle Anwendungen

### 8.14.1 Kopieren eines FDDRL-Sicherungsbandes

Beim Kopieren von FDDRL-Sicherungsbandsets sind die von FDDRL angebotenen Benutzeranschlüsse zur Kennsatzbearbeitung zu verwenden.

```
/SET-TASKLIB LIBRARY=SYSLNK.FDDRL.110.PERCON
/IMPORT-FILE TAPE(VOLUME=(EIN1,EIN2),-
/          DEVICE=T6250,-
/          FILE-NAME=FDDRL.V11.0A)
/SET-FILE-LINK LINK-NAME=PCIN,-
/          FILE-NAME=FDDRL.V11.0A,-
/          ACCESS-METHOD=BTAM
/CREATE-FILE FILE-NAME=FDDRL.V11.0A.COPY,-
/          SUPPORT=TAPE(VOLUME=(AUS1,AUS2),-
/          DEVICE=T6250)
/SET-FILE-LINK LINK-NAME=PCOUT,-
/          FILE-NAME=FDDRL.V11.0A.COPY
/START-PERCON
//ASSIGN-INPUT-FILE FILE=TAPE-FILE(NAME=FDDRL.V11.0A,-
//          LABEL-EXIT=MODULE(NAME=USERIN,-
//          CONTROLLED-LABEL=(HDR,EOV,END)))
//ASSIGN-OUTPUT-FILE FILE=TAPE-FILE(NAME=FDDRL.V11.0A.COPY,-
//          LABEL-EXIT=MODULE(NAME=USEROUT,-
//          CONTROLLED-LABEL=(HDR,EOV,END)),-
//          FILE-ATTRIBUTES=INPUT-FILE)
//END
```

Einzelne FDDRL-Sicherungsbänder können auch ohne diese Anschlußroutinen dupliziert werden. Dabei darf aber pro Eingabeband nur ein Ausgabedatenträger entstehen.

## 8.14.2 Kopieren einer SLED-Datei auf Platte

Die folgende Prozedur ist für K-Platten ausgelegt.

```
/IMPORT-FILE TAPE(VOLUME=EIN,-  
/                DEVICE=T6250,-  
/                FILE-NAME=SLEDFILE)  
/SET-FILE-LINK LINK-NAME=PCIN,-  
/                FILE-NAME=SLEDFILE,-  
/                ACCESS-METHOD=SAM,-  
/                BLOCK-CONTROL-INFO=BY-CATALOG,-  
/                BUFFER-LENGTH=BY-CATALOG,-  
/                RECORD-FORMAT=BY-CATALOG  
/CREATE-FILE FILE-NAME=SLEDTEST,-  
/                SUPPORT=PUBLIC-DISK(SPACE=RELATIVE(PRIMARY-ALLOCATION=2000,-  
/                                                    SECONDARY-ALLOCATION=500))  
/SET-FILE-LINK LINK-NAME=PCOUT,-  
/                FILE-NAME=SLEDTEST,-  
/                ACCESS-METHOD=PAM  
/START-PERCON  
//END
```

Nähere Informationen über den Umgang mit SLEDFILES bei anderen Plattenformaten können dem Handbuch „Einführung in die Systembetreuung“ [3] entnommen werden.

### 8.14.3 Auswählen und Verändern von Sätzen

Sätze können mit der selektiven SET-RECORD-MAPPING-Anweisung ausgewählt und verändert werden.

Für eine Ausgabedatei können mehrere SET-FILE-LINK-Kommandos mit demselben Eröffnungsmodus und verschiedenen Dateikettungsamen abgesetzt werden. Über diese Dateikettungsamen werden dann die PERCON-Anweisungen zugeordnet. So werden Sätze mit SELECT-INPUT-RECORDS ausgewählt und mit SET-RECORD-MAPPING verändert. Die restlichen Sätze bleiben unverändert oder werden in anderer Weise verändert. Zu beachten sind die Unterschiede, wenn Ein- und Ausgabedatei identisch sind oder die Ausgabedatei neu zu erstellen ist.

#### Ein- und Ausgabedatei sind identisch

Bei SAM- bzw. ISAM-Dateien können für dieselbe Datei mehrere SET-FILE-LINK-Kommandos mit OPEN-MODE=UPDATE bzw. OPEN-MODE=INOUT und verschiedenen Dateikettungsamen abgesetzt werden. Bei identischer Ein- und Ausgabedatei bleibt die Satzanzahl dieser Datei unverändert. Werden Sätze über mehrere SELECT-INPUT-RECORDS-Anweisungen ausgewählt und über mehrere SET-RECORD-MAPPING-Anweisungen verändert, werden die Sätze entsprechend oft überschrieben. Nur die letzte Änderung bleibt bestehen.

Bei **SAM-Dateien**, die mit OPEN-MODE=UPDATE geöffnet werden, ändert sich die Satzlänge nicht. Verlängerte Sätze werden auf ihre ursprüngliche Länge gekürzt; verkürzte Sätze mit dem FILLER-Zeichen der SET-RECORD-MAPPING-Anweisung auf ihre alte Länge gebracht.

```
/SET-FILE-LINK FILE-NAME=DATEI,-
/              LINK-NAME=PCIN1,-
/              OPEN-MODE=UPDATE
/SET-FILE-LINK FILE-NAME=DATEI,-
/              LINK-NAME=PCOUT1,-
/              OPEN-MODE=UPDATE
/SET-FILE-LINK FILE-NAME=DATEI,-
/              LINK-NAME=PCOUT2,-
/              OPEN-MODE=UPDATE
/SET-FILE-LINK FILE-NAME=DATEI,-
/              LINK-NAME=PCOUT3,-
/              OPEN-MODE=UPDATE
/START-PERCON
//ASSIGN-INPUT-FILE LINK-NAME=PCIN1
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT1
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=PCOUT1,-
//              CONDITION=((5,1)='A')
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT1,-
//              FILLER=INPUT,-
```

```

//                                OUTPUT-FIELDS=C'X'(OUTPUT-POSITION=5)  _____ (1)
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT2
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=PCOUT2,-
//                                CONDITION=((5,1)='B')
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT2,-
//                                FILLER=INPUT,-
//                                OUTPUT-FIELDS=C'Y'(OUTPUT-POSITION=5)  _____ (2)
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT3
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=PCOUT3,-
//                                CONDITION=((6,1)<='2')
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT3,-
//                                FILLER=INPUT,-
//                                OUTPUT-FIELDS=C'9'(OUTPUT-POSITION=6)  _____ (3)
//END

```

Datei vorher: nachher:

A0 *	A9
B1 *	B9
B2 *	B9
B3	Y3
A4	X4
C5	C5
B6	Y6

\* Die Änderungen durch (1) und (2) bleiben wegen (3) unberücksichtigt.

- (1) In Spalte 5 wird 'A' durch 'X' ersetzt.
- (2) In Spalte 5 wird 'B' durch 'Y' ersetzt.
- (3) In Spalte 6 werden alle Zeichen  $\leq$  '2' durch '9' ersetzt.

Bei **ISAM-Dateien** kann es zu unerwünschten Schleifen kommen, wenn

- der Satzschlüssel verändert wird,
- mit DUPLICATE-KEY=\*YES gearbeitet wird,
- der ausgegebene Satz vor dem eingelesenen Satz liegt, da die Schreibposition die nächste Leseposition festlegt.

## Ein- und Ausgabedatei verschieden

Bei ISAM- bzw. PAM-Dateien können für dieselbe Ausgabedatei mehrere SET-FILE-LINK-Kommandos mit OPEN-MODE=OUTPUT bzw. OPEN-MODE=OUTIN und verschiedenen Dateikettungsnamen abgesetzt werden.

Die Anzahl der Ausgabesätze kann sich reduzieren, wenn keine Eingabesätze durch eine SELECT-INPUT-RECORDS-Anweisung ausgewählt wurden. Sie kann sich erhöhen, wenn Eingabesätze durch mehrere SELECT-INPUT-RECORDS-Anweisungen ausgewählt wurden.

Wird bei ISAM-Dateien DUPLICATE-KEY=YES angegeben, kann ein Eingabesatz in mehrere Ausgabesätze aufgespalten werden.

```
/SET-FILE-LINK FILE-NAME=EINGABEDATEI,-
/          LINK-NAME=PCIN1,-
/          OPEN-MODE=INPUT
/SET-FILE-LINK FILE-NAME=AUSGABEDATEI,-
/          LINK-NAME=PCOUT1,-
/          OPEN-MODE=OUTPUT,-
/          ACCESS-METHOD=ISAM(DUPLICATE-KEY=YES)
/SET-FILE-LINK FILE-NAME=AUSGABEDATEI,-
/          LINK-NAME=PCOUT2,-
/          OPEN-MODE=OUTPUT,-
/          ACCESS-METHOD=ISAM(DUPLICATE-KEY=YES)
/SET-FILE-LINK FILE-NAME=AUSGABEDATEI,-
/          LINK-NAME=PCOUT3,-
/          OPEN-MODE=OUTPUT,-
/          ACCESS-METHOD=ISAM(DUPLICATE-KEY=YES)

/START-PERCON
//ASSIGN-INPUT-FILE LINK-NAME=PCIN1
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT1,-
//          FILE=DISK-FILE(FILE-ATTRIBUTES=INPUT-FILE)
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=PCOUT1,-
//          CONDITION=((13,1)='A')
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT1,-
//          FILLER=INPUT,-
//          OUTPUT-FIELDS=C'X'(OUTPUT-POSITION=13)
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT2,-
//          FILE=DISK-FILE(FILE-ATTRIBUTES=INPUT-FILE)
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=PCOUT2,-
//          CONDITION=((13,1)='B')
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT2,-
//          FILLER=INPUT,-
//          OUTPUT-FIELDS=C'Y'(OUTPUT-POSITION=13)
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT3,-
//          FILE=DISK-FILE(FILE-ATTRIBUTES=INPUT-FILE)
```

```
//SELECT-INPUT-RECORDS OUTPUT-LINK-NAME=PCOUT3,-  
//                      CONDITION=((14,1)<='2')  
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT3,-  
//                      FILLER=INPUT,-  
//                      OUTPUT-FIELDS=C'9'(OUTPUT-POSITION=14)  
//END
```

Eingabedatei:           Ausgabedatei:

00000001A0 (1)   00000001X0

00000002B1 (1)   00000001A9

00000003B2 (1)   00000002Y1

00000004B3       00000002B9

00000005A4       00000003Y2

00000006C5 (2)   00000003B9

00000007B6       00000004Y3

00000005X4

00000007Y6

- (1)   Diese Sätze werden jeweils durch die erste und dritte bzw. zweite und dritte SELECT-INPUT-RECORDS-Anweisung ausgewählt und erscheinen deshalb zweimal in der Ausgabedatei.
- (2)   Dieser Satz wird durch keine SELECT-INPUT-RECORDS-Anweisung ausgewählt und erscheint deshalb nicht in der Ausgabedatei.

## Aufspalten eines Satzes

Soll ein Satz in drei kürzere Sätze aufgespalten werden, ist dazu eine Zwischendatei erforderlich. Da diese eine ISAM-Datei ist, wird ein Pseudoschlüssel (für alle Sätze '1' in Spalte 5) eingeführt. Bei der Weiterverarbeitung der Zwischendatei wird dieser Schlüssel ignoriert.

```
/SET-FILE-LINK FILE-NAME=EINGABEDATEI,-
/
/      LINK-NAME=PCIN1,-
/      OPEN-MODE=INPUT
/SET-FILE-LINK FILE-NAME=ZWISCHENDATEI,-
/
/      LINK-NAME=PCOUT1,-
/      OPEN-MODE=OUTPUT,-
/      ACCESS-METHOD=ISAM(DUPLICATE-KEY=YES,-
/
/                               KEY-LENGTH=1,-
/                               KEY-POSITION=5)
/SET-FILE-LINK FILE-NAME=ZWISCHENDATEI,-
/
/      LINK-NAME=PCOUT2,-
/      OPEN-MODE=OUTPUT,-
/      ACCESS-METHOD=ISAM(DUPLICATE-KEY=YES,-
/
/                               KEY-LENGTH=1,-
/                               KEY-POSITION=5)
/SET-FILE-LINK FILE-NAME=ZWISCHENDATEI,-
/
/      LINK-NAME=PCOUT3,-
/      OPEN-MODE=OUTPUT,-
/      ACCESS-METHOD=ISAM(DUPLICATE-KEY=YES,-
/
/                               KEY-LENGTH=1,-
/                               KEY-POSITION=5)
/SET-FILE-LINK FILE-NAME=ZWISCHENDATEI,-
/
/      LINK-NAME=PCIN2
/SET-FILE-LINK FILE-NAME=AUSGABEDATEI,-
/
/      LINK-NAME=PCOUT4,-
/      ACCESS-METHOD=SAM
/START-PERCON
//ASSIGN-INPUT-FILE LINK-NAME=PCIN1
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT1
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT1,-
//
//      OUTPUT-FIELDS=(C'1'(OUTPUT-POSITION=5),-
//
//                               FIELD(INPUT-POSITION=5,-
//
//                                     INPUT-LENGTH=2,-
//                                     OUTPUT-POSITION=6))
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT2
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT2,-
//
//      OUTPUT-FIELDS=(C'1'(OUTPUT-POSITION=5),-
//
//                               FIELD(INPUT-POSITION=7,-
//
//                                     INPUT-LENGTH=2,-
//                                     OUTPUT-POSITION=6))
//
```



```
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT3
//SET-RECORD-MAPPING OUTPUT-LINK-NAME=PCOUT3,-
//      OUTPUT-FIELDS=(C'1'(OUTPUT-POSITION=5),-
//      FIELD(INPUT=POSITION=9,-
//      INPUT-LENGTH=2,-
//      OUTPUT-POSITION=6))
//START-CONVERSION
//ASSIGN-INPUT-FILE LINK-NAME=PCIN2
//ASSIGN-OUTPUT-FILE LINK-NAME=PCOUT4
//SET-RECORD-MAPPING OUTPUT-FIELDS=FIELD(INPUT-POSITION=6,-
//      INPUT-LENGTH=2,-
//      OUTPUT-POSITION=5)
//END
```

Eingabedatei:	Zwischendatei:	Ausgabedatei:
S1S2S3	1S1	S1
S4S5S6	1S2	S2
S7S8S9	1S3	S3
	1S4	S4
	1S5	S5
	1S6	S6
	1S7	S7
	1S8	S8
	1S9	S9

## 8.15 Arbeiten mit MF/MV-Sets

Die drei Plattendateien DATEI.01, DATEI.02 und DATEI.03 sollen auf Band gesichert werden. Da die Kapazität eines Bandes nicht ausreichen würde, werden zwei Datenträger für die Ausgabe zur Verfügung gestellt.

Die beiden Datenträger werden als TAPE-SET vereinbart, um mehrfach erforderliches Montieren und Positionieren der Datenträger zu vermeiden.

```

/CREATE-TAPE-SET TAPE-SET-NAME=SET1,VOLUME=(TAPEKA,BD0150) _____ (1)
/CREATE-FILE FILE-NAME=DATEI.01.BAND,- _____ (2)
/      SUPPORT=TAPE(VOLUME=*NO,DEVICE-TYP=T6250)
/SET-FILE-LINK LINK-NAME=PCOUT1,FILE-NAME=DATEI.01.BAND,-
/      ACCESS-METHOD=SAM,-
/      SUPPORT=TAPE(-
/      VOLUME-LIST=*BY-TAPE-SET(TAPE-SET-NAME=SET1),-
/      FILE-SEQUENCE=1)
/CREATE-FILE FILE-NAME=DATEI.02.BAND,- _____ (3)
/      SUPPORT=TAPE(VOLUME=*NO,DEVICE-TYP=T6250)
/SET-FILE-LINK LINK-NAME=PCOUT2,FILE-NAME=DATEI.02.BAND,-
/      ACCESS-METHOD=SAM,-
/      SUPPORT=TAPE(-
/      VOLUME-LIST=*BY-TAPE-SET(TAPE-SET-NAME=SET1),-
/      FILE-SEQUENCE=NEW)
/CREATE-FILE FILE-NAME=DATEI.03.BAND,-
/      SUPPORT=TAPE(VOLUME=*NO,DEVICE-TYP=T6250)
/SET-FILE-LINK LINK-NAME=PCOUT3,FILE-NAME=DATEI.03.BAND,-
/      ACCESS-METHOD=SAM,-
/      SUPPORT=TAPE(-
/      VOLUME-LIST=*BY-TAPE-SET(TAPE-SET-NAME=SET1),-
/      FILE-SEQUENCE=NEW)
/START-PERCON _____ (4)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=DATEI.01) _____ (5)
//ASSIGN-OUTPUT-FILE FILE=TAPE-FILE,LINK-NAME=PCOUT1 _____ (6)
//START-CONVERSION _____ (7)
% DMS0DE3 TAPE WITH VSN 'TAPEKA' FOR FILE ':catid:$userid.DATEI.01.BAND'
%      IS MOUNTED ON DEVICE 'TO'
% DMS0DE7 SAM FILE CLOSED: FILE NAME=:catid:$userid.DATEI.01.BAND, LINK
%      NAME=PCOUT1, BLOCK COUNT=00000001
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='PCIN'
%      (FILE=:catid:$userid.DATEI.01): 10
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='PCOUT1'
%      (FILE=:catid:$userid.DATEI.01.BAND): 10
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=DATEI.02) _____ (8)
//ASSIGN-OUTPUT-FILE FILE=TAPE-FILE,LINK-NAME=PCOUT2 _____ (9)

```

```
//START-CONVERSION _____ (10)
% DMSODE3 TAPE WITH VSN 'TAPEKA' FOR FILE ':catid:$userid.DATEI.02.BAND'
  IS MOUNTED ON DEVICE 'T0'
% DMSODE3 TAPE WITH VSN 'BD0150' FOR FILE ':catid:$userid.DATEI.02.BAND'
  IS MOUNTED ON DEVICE 'T1' _____ (11)
% DMSODE8 END OF TAPE FOR FILE ':catid:$userid.DATEI.02.BAND' WITH LINK
  NAME 'PCOUT2', BLOCK COUNT '000000' ON VOLUME WITH VSN 'TAPEKA'
% DMSODE8 END OF TAPE FOR FILE ':catid:$userid.DATEI.02.BAND' WITH LINK
  NAME 'PCOUT2', BLOCK COUNT '000060' ON VOLUME WITH VSN 'BD0150'
% DMSODE7 SAM FILE CLOSED: FILE NAME=:catid:$userid.DATEI.02.BAND, LINK
  NAME=PCOUT2, BLOCK COUNT=00000060
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='PCIN'
  (FILE=:catid:$userid.DATEI.02): 60
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='PCOUT2'
  (FILE=:catid:$userid.DATEI.02.BAND): 60
//ASSIGN-INPUT-FILE FILE=DISK-FILE(NAME=DATEI.03) _____ (12)
//ASSIGN-OUTPUT-FILE FILE=TAPE-FILE,LINK-NAME=PCOUT3 _____ (13)
//END _____ (14)
% DMSODE3 TAPE WITH VSN 'BD0150' FOR FILE ':catid:$userid.DATEI.03.BAND'
  IS MOUNTED ON DEVICE 'T1'
% DMSODE7 SAM FILE CLOSED: FILE NAME=:catid:$userid.DATEI.03.BAND, LINK
  NAME=PCOUT3, BLOCK COUNT=00000001
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='PCIN'
  (FILE=:catid:$userid.DATEI.03): 10
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='PCOUT3'
  (FILE=:catid:$userid.DATEI.03.BAND): 10
% PER0031 PERCON TERMINATED NORMALLY
//DELETE-TAPE-SET TAPE-SET-NAME=SET1 _____ (15)
```

- (1) Die beiden Datenträger (TAPEKA und BD0150) werden als TAPE-SET vereinbart.
- (2) Mit den Kommandos CREATE-FILE und SET-FILE-LINK wird die erste Ausgabedatei (FILE-SEQUENCE=1) vereinbart, die auf Band geschrieben werden soll. Die beiden Datenträger werden über den Parameter TAPE-SET-NAME zugeordnet.
- (3) Weitere Ausgabedateien (FILE-SEQUENCE=NEW) werden analog zu (2) vereinbart.
- (4) PERCON wird gestartet.
- (5) Die erste Eingabedatei wird zugewiesen.
- (6) Die erste Ausgabedatei wird zugewiesen.
- (7) Start der Übertragung. Die Eingabedatei paßt vollständig auf das erste Band.
- (8) Eine weitere Eingabedatei wird zugewiesen.
- (9) Eine weitere Ausgabedatei wird zugewiesen.
- (10) Start der Übertragung.

- (11) Die Eingabedatei paßt nicht vollständig auf das erste Band. Für den Rest der Eingabedatei wird das zweite Band verwendet.
- (12) Eine weitere Eingabedatei wird zugewiesen.
- (13) Eine weitere Ausgabedatei wird zugewiesen.
- (14) Start der Übertragung. Die Eingabedatei wird vollständig auf dem zweiten Band abgelegt. Nach Abschluß der Übertragung wird PERCON beendet.
- (15) Das vereinbarte TAPE-SET von Datenträgern wird gelöscht.

### Einlesen einer Datei von einem MF/MV-SET

Die auf zwei Bändern abgelegte Datei DATEI.02.BAND soll gelesen werden.

```

/DELETE-FILE FILE-NAME=DATEI.02.BAND _____ (1)
% DMS0800 SPECIFIED FILE ':catid:$userid.DATEI.02.BAND' DELETED
/SET-JOB-STEP
/IMPORT-FILE SUPPORT=TAPE(FILE-NAME=DATEI.02.BAND,- _____ (2)
/          VOLUME=(TAPEKA,BD0150),DEVICE-TYPE=T6250)
/SET-FILE-LINK LINK-NAME=PCIN,FILE-NAME=DATEI.02.BAND,- _____ (3)
/          SUPPORT=TAPE(FILE-SEQUENCE=UNKNOWN)
/SET-FILE-LINK LINK-NAME=PCOUT,FILE-NAME=DATEI.02,- _____ (4)
/          ACCESS-METHOD=SAM,-
/          RECORD-FORMAT=FIXED(RECORD-SIZE=2000)
/START-PERCON _____ (5)
% BLS0517 MODULE 'PCROOT' LOADED
% PER0000 PERCON STARTED, VERSION V02.6A00
//ASSIGN-INPUT-FILE FILE=TAPE-FILE _____ (6)
//END _____ (7)
% DMS0DE3 TAPE WITH VSN 'TAPEKA' FOR FILE ':catid:$userid.DATEI.02.BAND'
%          IS MOUNTED ON DEVICE 'T0'
% DMS0DE3 TAPE WITH VSN 'BD0150' FOR FILE ':catid:$userid.DATEI.02.BAND'
%          IS MOUNTED ON DEVICE 'T1'
% DMS0DE8 END OF TAPE FOR FILE ':catid:$userid.DATEI.02.BAND' WITH LINK
%          NAME 'PCIN', BLOCK COUNT '000000' ON VOLUME WITH VSN 'TAPEKA'
% DMS0DE8 END OF TAPE FOR FILE ':catid:$userid.DATEI.02.BAND' WITH LINK
%          NAME 'PCIN', BLOCK COUNT '000060' ON VOLUME WITH VSN 'BD0150'
% DMS0DE7 SAM FILE CLOSED: FILE NAME=:catid:$userid.DATEI.02.BAND, LINK
%          NAME=PCIN, BLOCK COUNT=00000060
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='PCIN'
%          (FILE=:catid:$userid.DATEI.02.BAND):          60
% PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='PCOUT'
%          (FILE=:catid:$userid.DATEI.02):          60
% PER0031 PERCON TERMINATED NORMALLY

```

- (1) Ein eventuell vorhandener Katalogeintrag für die anzulegende Datei wird gelöscht.
- (2) Die einzulesende Datei wird von den Datenträgern des MF/MV-Sets importiert.
- (3) Zuweisen der Eingabedatei mit dem Kommando SET-FILE-LINK.  
Da nicht bekannt ist, welchen Platz die Eingabedatei in der Folge der Dateien auf den Datenträgern einnimmt, wird der Parameter FILE-SEQUENCE=UNKNOWN angegeben.
- (4) Eine Ausgabedatei auf Platte wird zugewiesen.
- (5) PERCON wird gestartet.
- (6) Zuweisen der Eingabedatei:  
Das Zuweisen der Eingabedatei mit ASSIGN-INPUT-FILE ist notwendig, da FILE=TAPE-FILE nicht Standard ist.  
Die Anweisung ASSIGN-OUTPUT-FILE kann entfallen, da die Ausgabedatei durch das Kommando SET-FILE-LINK und die Standardeinstellungen der ASSIGN-Anweisung vollständig beschrieben ist.
- (7) Start der Übertragung. Die Eingabedatei wird von zwei Datenträgern gelesen. Anschließend wird PERCON beendet.



---

## 9 Meldungen

### Ausgabe von Meldungen in S-Variablen

Für eine Reihe von Meldungen werden Meldungsschlüssel und Inserts (Nummer und Semantik) als unveränderliche Bestandteile für künftige PERCON- und BS2000/OSD-BC-Versionen garantiert. Solche Meldungen werden als garantierte Meldungen bezeichnet.

#### Garantierte Meldungen des Produkts PERCON

Garantierte Meldungen von PERCON sind

- Meldungen zum Start und zur Beendigung von PERCON
- Meldungen über die Anzahl bearbeiteter Sätze bzw. Blöcke
- Meldungen mit Inserts wie DVS-Fehlercodes, Dateinamen, usw.
- Meldungen zu Fehlern, deren Ursache nicht in der Syntax bzw. Semantik der Anweisungen liegt (z.B. zu kurze Sätze in einer Datei).

In PERCON werden folgende Meldungen garantiert:

PER0000	PERCON startet
PER0001	PERCON abgebrochen
PER0002	Fehler in einem Systemmakro
PER0003	Speicherplatzmangel
PER0004	Datei auf falschem Speichermedium
PER0009	Ausgabe-Satzlänge zu groß
PER0012	Widerspruch in Dateinamen
PER0014	Sekundärschlüssel nicht übernommen
PER0016	DVS-Fehler bei Dateiwiederherstellung
PER0017 – 19	XHCS-Probleme
PER0021 – 23	OPEN- / CLOSE-Probleme
PER0028	LINK-Fehler bei Benutzermoduln
PER0029 – 30	Anzahl bearbeiteter Sätze bzw. Blöcke
PER0031	PERCON normal beendet
PER0035	ISAM-Schlüssel nicht aufsteigend
PER0036	DVS-Fehler

PER0038	Abbruch wegen Parity- oder Längenfehler
PER0039	CCSN-Unverträglichkeit bei SYSOUT
PER0041 – 42	fehlerhafter Datensatz
PER0044	Datensatz zu kurz
PER0045	Datensätze nicht in Folge
PER0048	Returncode einer Benutzer-Routine unzulässig
PER0053 – 54	Fehler der Ausgabedatei bei FILLER=*OUTPUT
PER0057	Gerätefehler beim Positionieren
PER0069	Fehler beim Bandduplizieren
PER0092 – 94	Dateiwiederherstellung mittels VERIF
PER0095	Großmodul von PERCON nicht ladbar
PER0098	ISAM-Schlüssel bereits vorhanden
PER0099 – 103	Fehler bei Übernahme von Sekundärschlüsseln
PER0104 – 107	PERCON-Version nicht verfügbar
PER0109	CCSN für Ausgabedatei nicht vermerkbar
PER0110	unzulässige BS2000-Version
PER0111	BIND-Fehler beim Nachbinden des Großmoduls

Ab BS2000/OSD-BC V1.0 und bei Einsatz des kostenpflichtigen Produkts SDF-P (ab Version 2.0A) besteht die Möglichkeit, diese garantierten Meldungen in strukturierte S-Variablen auszugeben. Über S-Variablen kann direkt auf bestimmte Meldungsdaten zugegriffen werden, ohne daß das Ausgabelay-out der Meldungen bekannt sein muß. Damit ist es möglich, in SDF-P-Prozeduren, abhängig vom Inhalt dieser S-Variablen, die Weiterverarbeitung zu steuern.

## Vorgehensweise

Eine S-Variable, die Meldungen aufnehmen kann, muß als Liste von Strukturen aufgebaut sein. Ihr Name ist frei wählbar und wird im folgenden durch *varname* dargestellt.

Die S-Variable muß vom Benutzer mit folgendem Kommando vereinbart werden:

```
/DECLARE-VARIABLE NAME=varname(TYPE=*STRUCTURE(DEFINITION=*DYNAMIC)), -
/                               MULTIPLE-ELEMENTS=*LIST, SCOPE=*TASK
```

### *Hinweis*

Die Angabe SCOPE=\*TASK ist nur erforderlich, wenn die S-Variable auch in Prozeduren gültig sein soll, die nach der Vereinbarung der S-Variablen aufgerufen werden.



Für jede Struktur (jede Meldung) sind folgende Struktur-Elemente vorgesehen:

Name des Strukturelements	Inhalt
varname(*LIST).MSG-TEXT	vollständiger Meldungstext
varname(*LIST).MSG-ID	Meldungsschlüssel
varname(*LIST).I0	Insert 0
varname(*LIST).I1	Insert 1
:	:
varname(*LIST).I14	Insert 14
varname(*LIST).REPLY	eingegebene Antwort

Die Anzahl der Inserts hängt von der Meldung ab, ebenso, ob eine Antwort vermerkt wird oder nicht.

Nach der Vereinbarung der S-Variablen wird ihr der Meldungsstrom mit folgendem Kommando zugewiesen:

```
/ASSIGN-STREAM STREAM-NAME=SYSMSG,T0=*VARIABLE(varname)
```

Danach kann PERCON oder ein Programm, das PERCON als Unterprogramm aufruft, gestartet werden. Während des PERCON-/Programm-Laufs wird die S-Variable gefüllt.

Nach Beendigung von PERCON oder des Programms, wird der Meldungsstrom in die S-Variable gestoppt und wieder dem Standard-Ausgabemedium für Meldungen zugewiesen:

```
/ASSIGN-STREAM STREAM-NAME=SYSMSG,T0=*STD
```

Die Variable `varname` enthält alle garantierten Meldungen, die zwischen den beiden ASSIGN-STREAM-Kommandos ausgegeben wurden. Dazu gehören insbesondere auch die BLS-Meldungen.

Wenn PERCON als Unterprogramm aufgerufen wird, enthält die S-Variable zusätzlich alle garantierten Meldungen, die vom Hauptprogramm ausgegeben wurden.

Um die Ausgabe zusätzlicher Meldungen zu unterdrücken, können die beiden ASSIGN-STREAM-Kommandos auch über CMD-Makro unmittelbar vor und nach dem PERCON-Aufruf im Hauptprogramm ausgeführt werden.

Die Ausgabe in die S-Variablen erfolgt unabhängig davon, ob bei der Meldungsabgabe die Ausgabe über SYSOUT bzw. SYSLST unterdrückt wird oder nicht.

Eine detaillierte Beschreibung der S-Variablen ist im Handbuch „SDF-P“ [15] zu finden.

## Auswertung von S-Variablen

Der Inhalt einer S-Variablen kann mit dem Kommando SHOW-VARIABLE ausgegeben werden. Nach Abschluß eines PERCON-Laufs könnte eine S-Variable mit dem Namen pervar wie folgt belegt sein:

```
/show-variable variable-name=pervar
```

```
PERVAR(*LIST).MSG-TEXT = % BLS0517 MODULE 'PCROOT' GELADEN
PERVAR(*LIST).MSG-ID = BLS0517
PERVAR(*LIST).IO = PCROOT
PERVAR(*LIST).MSG-TEXT = % PER0000 PERCON GESTARTET, VERSION V02.6A00
PERVAR(*LIST).MSG-ID = PER0000
PERVAR(*LIST).IO = V02.6A00
PERVAR(*LIST).MSG-TEXT = % PER0044 FELD IN 'SET-RECORD-MAPPING'-ANWEISUNG
UEBERRAGT EIN-/AUSGABE-SATZENDE, LINK='PCOUT'
PERVAR(*LIST).MSG-ID = PER0044
PERVAR(*LIST).IO = SET-RECORD-MAPPING
PERVAR(*LIST).I1 = PCOUT
PERVAR(*LIST).MSG-TEXT = % PER0030 ANZAHL VERARBEITETER SAETZE FUER
LINK='PCIN' :          3
PERVAR(*LIST).MSG-ID = PER0030
PERVAR(*LIST).IO = PCIN
PERVAR(*LIST).I1 =
PERVAR(*LIST).I2 =          3
PERVAR(*LIST).MSG-TEXT = % PER0030 ANZAHL VERARBEITETER SAETZE FUER
LINK='PCOUT' (FILE=:catid:$userid.XXX):          2
PERVAR(*LIST).MSG-ID = PER0030
PERVAR(*LIST).IO = PCOUT
PERVAR(*LIST).I1 = (FILE=:catid:$userid.XXX)
PERVAR(*LIST).I2 =          2
PERVAR(*LIST).MSG-TEXT = % PER0001 PERCON-LAUF ABGEBROCHEN
PERVAR(*LIST).MSG-ID = PER0001
```

Die einzelnen Struktur-Elemente der Variablen können gezielt angesprochen werden (z.B. wird mit /show-variable variable-name=pervar#3.MSG-ID der Meldungsschlüssel der dritten Meldung ermittelt).

Folgendes Beispiel zeigt einen Ausschnitt aus einer SDF-P-Prozedur, in der eine FOR-Schleife verwendet wird, um garantierte Meldungen in S-Variablen abzulegen:

```
/DECLARE-VARIABLE NAME=PERVAR(TYPE=*STRUCTURE), - _____ (1)
/      MULTIPLE-ELEMENTS=*LIST
/DECLARE-VARIABLE NAME=PERSTR(TYPE=*STRUCTURE) _____ (2)
/ASSIGN-STREAM STREAM-NAME=SYSMSG,TO=*VARIABLE(PERVAR) _____ (3)
/START-PERCON _____ (4)
.
... (PERCON-Anweisungen usw.)
.
```

```

/ASSIGN-STREAM STREAM-NAME=SYMSG,TO=*STD _____ (5)
/FOR PERSTR= *LIST(PERVAR) _____ (6)
/   IF (PERSTR.MSG-ID = 'PER0001')
/       WRITE-TEXT '*-----*'
/       WRITE-TEXT '* ABBRUCH PERCON / MASSNAHME ERFORDERLICH *'
/       WRITE-TEXT '*-----*'
/   END-IF
/END-FOR

```

- (1) Die S-Variable PERVAR wird als Liste von Strukturen vereinbart.
- (2) Die Strukturvariable PERSTR wird für die folgende FOR-Schleife vereinbart.
- (3) Ab jetzt werden garantierte Meldungen in PERVAR abgelegt.
- (4) PERCON wird gestartet.
- (5) Das Ablegen der garantierten Meldungen in PERVAR wird gestoppt.
- (6) Mit Hilfe einer FOR-Schleife wird die gesamte Liste von Strukturen PERVAR durchsucht, ob der Meldungsschlüssel PER0001 als MSG-ID auftritt. Wenn ja, wird mit WRITE-TEXT eine Information ausgegeben.

Die Fehlermeldungen werden 7stellig (PERnnnn) mit ihrer Bedeutung und eventuell erforderlicher Maßnahmen in deutscher bzw. englischer Sprache ausgegeben. Garantierte Meldungen sind mit Warranty: Y gekennzeichnet.

Den Umfang der Dialogmeldungen kann man durch Angaben in der MODIFY-PERCON-OPTIONS-Anweisung steuern.

```

PER0000 PERCON STARTED, VERSION (&00)
PER0000 PERCON GESTARTET, VERSION (&00)
Warranty: Y

```

### Bedeutung

(&00): PERCON-Version (nn.nann).

```

PER0001 PERCON RUN ABORTED
PER0001 PERCON-LAUF ABGEBROCHEN
Warranty: Y

```

### Bedeutung

Auskunft ueber die Fehlerursache gibt:

- bei autonomem PERCON-Aufruf die vorherige Fehlermeldung und
- bei Aufruf von PERCON als Unterprogramm die Rueckkehrinformation.

Bei autonomem Aufruf in einer Prozedur wird auf das naechste /STEP-, /SET-STEP-Kommando oder an das Prozedurende verzweigt.

### Maßnahme

PERCON-Lauf nach Beseitigung des Fehlers neu starten.

PER0002 ERROR CODE X'(&01)' DURING '(&00)' MACRO  
PER0002 FEHLER-CODE X'(&01)' BEI '(&00) '-MAKRO  
Warranty: Y

**Bedeutung**

Ein Lese- oder Schreibfehler in einem System-Makro fuehrt zum Abbruch des Konvertierungsschrittes.

(&00): RDATA, WROUT, WRLST

(&01): Makro-Fehlerschluessel.

**Maßnahme**

Fehlerart der Beschreibung 'Makroaufrufe' entnehmen und Lauf wiederholen.

PER0003 INSUFFICIENT CLASS 6 MEMORY SPACE  
PER0003 KLASSE-6-SPEICHER NICHT AUSREICHEND  
Warranty: Y

**Bedeutung**

Der Konvertierungsschritt wird fehlerhaft beendet.

**Maßnahme**

Ursache fuer Speichermangel beseitigen und Lauf wiederholen.

PER0004 FILE NOT ON '(&01)', LINK='(&00)'  
PER0004 DATEI NICHT AUF '(&01)', LINK='(&00)'  
Warranty: Y

**Bedeutung**

In der Anweisung ASSIGN-INPUT-FILE bzw. ASSIGN-OUTPUT-FILE wurde mit FILE=DISK-FILE() keine Plattendatei bzw. mit FILE=TAPE-FILE() keine Banddatei angegeben.

(&00): Dateikettungsname

(&01): DISK, TAPE.

**Maßnahme**

Anweisung ASSIGN-INPUT-FILE bzw. ASSIGN-OUTPUT-FILE oder /SET-FILE-LINK (FILE) -Kommando korrigieren und Konvertierungsschritt wiederholen.

PER0005 INVALID FORMAT CONVERSION  
PER0005 UNZULAESSIGE FORMAT-UMWANDLUNG

**Bedeutung**

In der Anweisung SET-RECORD-MAPPING bzw. SET-GROUP-ATTRIBUTES wurde eine unerlaubte Kombination der Strukturoperanden INPUT-FORMAT und OUTPUT-FORMAT im Operandenwert FIELD angegeben oder bei der Kombination von INPUT-FORMAT=CHARACTER und OUTPUT-FORMAT=ZONED-DECIMAL wurde ein Wert > 4 fuer INPUT-LENGTH angegeben.

**Maßnahme**

Im Dialog die widerspruechlichen Operanden korrigieren,  
im Batch den Konvertierungsschritt mit korrigierter Anweisung wiederholen.

PER0006 OPERAND VALUE 'RECORD-LENGTH' ONLY PERMISSIBLE FOR 'INPUT-FORMAT=CHARACTER'  
PER0006 OPERAND-WERT 'RECORD-LENGTH' NUR BEI 'INPUT-FORMAT=CHARACTER' ERLAUBT

**Bedeutung**

Der Strukturoperandenwert RECORD-LENGTH des Operandenwertes FIELD der SET-RECORD-MAPPING-Anweisung ist nur bei INPUT-FORMAT=CHARACTER zulaessig.

**Maßnahme**

Im Dialog die widerspruechlichen Operanden korrigieren,  
im Batch den Konvertierungsschritt mit korrigierter Anweisung wiederholen.

PER0007 STRUCTURE OPERAND 'OUTPUT-LENGTH' TOO SMALL  
PER0007 STRUKTUR-OPERAND 'OUTPUT-LENGTH' ZU KLEIN

**Bedeutung**

Im Operanden OUTPUT-FIELDS der Anweisung SET-RECORD-MAPPING oder im Operanden GROUP-HEADER bzw. GROUP-TRAILER der Anweisung SET-GROUP-ATTRIBUTES ist ein Feld oder Kennwort angegeben, dessen Strukturoperand OUTPUT-LENGTH fuer die Ausgabe und eine eventuelle Formataenderung nicht ausreicht.

**Maßnahme**

Im Dialogbetrieb den fehlerhaften Operanden korrigieren,  
im Batch den Konvertierungsschritt mit korrigierter Anweisung wiederholen.

PER0008 OPERAND VALUE 'REPLACE-CHARACTER' SPECIFIED MORE THAN ONCE  
PER0008 OPERANDEN-WERT 'REPLACE-CHARACTER' MEHRFACH ANGEGEBEN

**Bedeutung**

Im CODE-TRANSLATION-Operanden der SET-RECORD-MAPPING-Anweisung wurde ein Zeichen mehrfach als Strukturoperand INPUT-CHARACTER angegeben.

**Maßnahme**

Im Dialogbetrieb den fehlerhaften Operanden korrigieren,  
im Batch den Konvertierungsschritt mit korrigierter Anweisung wiederholen.

PER0009 WARNING: OUTPUT RECORD LENGTH > MAXIMUM RECORD LENGTH, LINK='(&00)'  
 PER0009 WARNUNG: AUSGABE-SATZLAENGE > MAXIMALE SATZLAENGE, LINK='(&00)'  
 Warranty: Y

### Bedeutung

Ausgabesaetze, deren Laenge die maximal zulaessige Ausgabesatzlaenge ueberschreiten, werden auf diese durch Abschneiden auf der rechten Seite gekuerzt. Die Meldung wird nur beim ersten Satz, auf den dies zutrifft ausgegeben.

(&00): Dateikettungsname der Ausgabedatei.

### Maßnahme

Saetze ueber SET-RECORD-MAPPING (RECORD)-Anweisung kuerzen bzw. Satz- und / oder Blocklaenge der Ausgabedatei vergroessern.

PER0010 NOT A PERCON STATEMENT  
 PER0010 KEINE PERCON-ANWEISUNG

### Bedeutung

Die zuletzt eingelesene Anweisung kann nicht entschlusselt werden.

### Maßnahme

Operationsteil korrigieren.

PER0011 STATEMENT READ ERROR  
 PER0011 ANWEISUNGS-LESEFEHLER

### Bedeutung

Beim Einlesen einer Anweisung ist ein Fehler aufgetreten.

Moegliche Ursachen:

- Kommando statt Anweisung gelesen oder
- SDF nicht verfuegbar oder
- PERCON-Anweisungen fehlen in der Syntax-Datei.

### Maßnahme

Prozedur ueberpruefen, eventuell Systemverwalter verstaendigen.

PER0012 WARNING: INCONSISTENCY IN FILE NAMES, LINK='(&00)'  
 PER0012 WARNUNG: WIDERSPRUCH IN DEN DATEINAMEN, LINK='(&00)'  
 Warranty: Y

### Bedeutung

Dateiname im TFT-Eintrag stimmt nicht mit jenem in der ASSIGN-INPUT-FILE (FILIN) bzw. ASSIGN-OUTPUT-FILE (FILOUT) Anweisung ueberein oder eine Dateigeneration wird verwendet.

Gearbeitet wird mit der Datei, die im /SET-FILE-LINK (FILE) Kommando angegeben wurde. (&00): Dateikettungsname.

### Maßnahme

Dateiname nur einmal oder gleiche Namen verwenden.

PER0013 SYNTACTICAL ERROR IN '(&00)' STATEMENT  
PER0013 SYNTAX-FEHLER IN '(&00)'-ANWEISUNG

**Bedeutung**

Die angegebene Anweisung ist syntaktisch fehlerhaft. Der fehlerhafte Teil (Operand oder Operandenwert) ist der Protokollierung zu entnehmen.

(&00): Bezeichnung der fehlerhaften Anweisung.

**Maßnahme**

Im Dialogbetrieb die korrigierte Anweisung erneut eingeben,  
im Batch den Konvertierungsschritt nach Korrektur der Anweisung wiederholen.

PER0014 WARNING: ALTERNATE KEYS NOT TRANSFERRED, LINK='(&00)'  
PER0014 WARNUNG: SEKUNDAERSCHLUESSEL NICHT UEBERNOMMEN, LINK='(&00)'  
Warranty: Y

**Bedeutung**

Die Eingabedatei enthaelt SekundaerschluesSEL, die angegebene Ausgabedatei nicht.  
SekundaerschluesSEL koennen fuer NK-ISAM-Dateien im nachhinein mit dem Kommando  
/CREATE-ALTERNATE-INDEX aufgebaut werden, fuer K-ISAM-Dateien nicht.

(&00): Dateikettungsname.

**Maßnahme**

Gegebenenfalls /CREATE-ALTERNATE-INDEX-Kommando nutzen.

PER0015 ILLEGAL COMBINATION OF INPUT, OUTPUT AND TRANSPOSE  
PER0015 UNZULAESSIGE KOMBINATION VON EINGABE, AUSGABE UND UMSETZUNG

**Bedeutung**

Eine Konvertierung kann nicht stattfinden, da Eingabe-, Ausgabe- und Umsetzparameter  
nicht zueinander passen.

**Maßnahme**

Dateimerkmale und Konvertierungsanweisungen ueberpruefen.

PER0016 WARNING: DMS ERROR '(&00)' WHILE REPAIRING FILE (LINK='(&01)'). FURTHER  
INFORMATION: /HELP-MSG DMS(&00)  
PER0016 WARNUNG: DVS-FEHLER '(&00)' BEI WIEDERHERSTELLUNG VON DATEI (LINK='(&01)').  
WEITERE INFORMATION: /HELP-MSG DMS(&00)  
Warranty: Y

**Bedeutung**

Beim Oeffnen der Datei mit Dateikettungsnamen (&01) wurde ein Fehler festgestellt.  
Bei der folgenden Wiederherstellung im OPENC-Exit mittels Makro VERIF trat der DVS-Fehler (&00) auf. Die Verarbeitung der fehlerhaften Datei wird fortgesetzt.  
Naehere Information ueber den DVS-Fehlerschluessel kann mit /HELP-MSG erfragt bzw. dem BS2000-Handbuch 'Systemmeldungen' entnommen werden.  
(&00): DVS-Fehlerschluessel  
(&01): Dateikettungsname.

**Maßnahme**

Ausgabe auf Vollstaendigkeit ueberpruefen.

PER0017 C STRING OF STATEMENT '(&00)' NOT CONVERTIBLE, LINK='(&01)'  
PER0017 C-STRING VON ANWEISUNG '(&00)' NICHT UMSETZBAR, LINK='(&01)'  
Warranty: Y

**Bedeutung**

Werden in der Anweisung SELECT-INPUT-RECORDS C-Strings verwendet, so muessen Anweisungs-CCS und Eingabe-CCS derselben Codefamilie angeh hoeren, und alle Zeichen der C-Strings muessen im Eingabe-CCS vorhanden sein.  
Werden in der Anweisung SET-GROUP-ATTRIBUTES bzw. SET-RECORD-MAPPING oder SET-PAGE-LAYOUT C-Strings verwendet, so muessen Anweisungs-CCS und Ausgabe-CCS derselben Codefamilie angeh hoeren, und alle Zeichen der C-Strings muessen im Ausgabe-CCS vorhanden sein.  
(&00): Moegliche fehlerhafte Anweisung:  
SELECT-INPUT-RECORDS, SET-GROUP-ATTRIBUTES,  
SET-RECORD-MAPPING, SET-PAGE-LAYOUT,  
(&01): Dateikettungsname der Ausgabe.

**Maßnahme**

Die Anweisungen in das Eingabe-CCS oder in das Ausgabe-CCS umsetzen.



PER0018 EXTENDED HOST CODE TABLE NOT AVAILABLE, LINK='(&01)'  
PER0018 'EXTENDED HOST CODE' TABELLE NICHT VERFUEGBAR, LINK='(&01)'  
Warranty: Y

**Bedeutung**

PERCON kann nicht feststellen, welche Zeichen des Eingabe-CCS Buchstaben oder Ziffern sind bzw. welche Zeichen des Ausgabe-CCS abdruckbar sind.  
(&01): Dateikettungsname.

**Maßnahme**

Systemverwalter verstaendigen. XHCS-Tabellen des Systems pruefen.

PER0019 CONVERSION OF INPUT DATA INTO OUTPUT CCS NOT POSSIBLE, LINK='(&01)'  
PER0019 UMSETZUNG DER EINGABE-DATEN IN DAS AUSGABE-CCS NICHT MOEGlich, LINK='(&01)'  
Warranty: Y

**Bedeutung**

Bedingung fuer eine Konvertierung ist, dass das Eingabe-CCS und das Ausgabe-CCS denselben Namen haben oder dass das Eingabe-CCS und das Ausgabe-CCS derselben Codefamilie angehoren und das Ausgabe-CCS ein vollstaendig kompatibles CCS ist.  
(&01): Dateikettungsname der Ausgabe.

**Maßnahme**

Anderes Ausgabe-CCS verwenden.

PER0020 ILLEGAL OVERLAPPING OF OUTPUT AREAS  
PER0020 UNZULAESSIGE UEBERSCHNEIDUNG DER AUSGABE-BEREICHE

**Bedeutung**

Bei der SET-PAGE-LAYOUT-Anweisung sind im Operanden OUTPUT-AREA Bereiche angegeben, die sich ueberschneiden.

**Maßnahme**

SET-PAGE-LAYOUT-Anweisung ueberpruefen.

PER0021 OPEN ERROR ON INPUT FILE X'(&01)', LINK='(&00)'. REPLY (HA = CONV. TERM. ABNORMALLY; HN = CONV. TERM.; SA = SKIP INPUT FILE, PERCON ABORTED; SN = SKIP INPUT FILE, PERCON TERM.)

PER0021 EINGABE-DATEI EROEFFNUNGS-FEHLER X'(&01)', LINK='(&00)'. ANTWORT (HA=KONV. ENDE FEHLERH.; HN=KONV. ENDE; SA=EINGABE-DATEI UEBERGEHEN, PERCON ABBRECHEN; SN=EINGABE-DATEI UEBERGEHEN, PERCON ENDE)

Warranty: Y

### Bedeutung

Beim Eroeffnen der Eingabedatei mit Dateikettungsname (&00) trat der Fehler (&01) auf.  
 (&00): Dateikettungsname  
 (&01): DVS-Fehlerschluessel.

### Maßnahme

Moegliche Massnahmen:

- HA: Konvertierungsschritt beenden, PERCON fehlerhaft beenden oder
- HN: Konvertierungsschritt beenden, PERCON normal beenden oder
- SA: Fehlerhafte Eingabedatei uebergehen, PERCON fehlerhaft beenden oder
- SN: Fehlerhafte Eingabedatei uebergehen, PERCON normal beenden.

PER0022 FILE OPEN ERROR X'(&01)', LINK='(&00)'. FURTHER INFORMATION: /HELP-MSG DMS(&01)

PER0022 DATEI-EROEFFNUNGS-FEHLER X'(&01)', LINK='(&00)'. WEITERE INFORMATION: /HELP-MSG DMS(&01)

Warranty: Y

### Bedeutung

Beim Eroeffnen der Datei mit Dateikettungsname (&00) trat der Fehler (&01) auf.  
 Ist die fehlerhafte Datei eine Eingabedatei, so wird abhaengig von der Einstellung des OPEN-ERROR (OPENER) -Operanden der ASSIGN-INPUT-FILE (FILIN)-Anweisung weiterverfahren. Eroeffnungsfehler von Ausgabedateien fuehren zur Beendigung des Konvertierungsschrittes.  
 (&00): Dateikettungsname  
 (&01): DVS-Fehlerschluessel.

PER0023 FILE CLOSE ERROR X'(&01)', LINK='(&00)'. FURTHER INFORMATION: /HELP-MSG DMS(&01)

PER0023 DATEI-ABSCHLUSS-FEHLER X'(&01)', LINK='(&00)'. WEITERE INFORMATION: /HELP-MSG DMS(&01)

Warranty: Y

### Bedeutung

Beim Schliessen der Datei mit Dateikettungsname (&00) trat der Fehler X'01' auf.  
 Der Konvertierungsschritt wird beendet.  
 (&00): Dateikettungsname  
 (&01): DVS-Fehlerschluessel.

PER0024 WARNING: INVALID STATEMENT IGNORED  
PER0024 WARNUNG: FEHLERHAFTE ANWEISUNG IGNORIERT

**Bedeutung**

Die eingelesene Anweisung ist fehlerhaft.

Die Art des Fehlers wird durch die zuletzt ausgegebene Meldung erläutert.

PER0025 INVALID USE OF A LINK NAME  
PER0025 UNGUELTIGE VERWENDUNG EINES DATEIKETTUNGSNAMENS

**Bedeutung**

In der eingelesenen Anweisung ist ein Dateikettungsname

- mehrfach angegeben oder
- keiner Datei zugeordnet oder
- nicht den BS2000 Namenskonventionen entsprechend.

**Maßnahme**

Verwendete Dateikettungsnamen ueberpruefen.

PER0026 NO DATA LINES FOR PRINT EDITING  
PER0026 KEINE DATEN-ZEILEN FUER DRUCK-AUFBEREITUNG

**Bedeutung**

In der ueber die SET-PAGE-LAYOUT (FORMAT) -Anweisung aufgebauten Ausgabemaske sind keine Zeilen fuer Datenaufnahme frei.

**Maßnahme**

Im OUTPUT-AREA (OUTLINE)-Operanden Datenzeilen zufuegen und Lauf wiederholen.

PER0027 GROUP CONTROL MAY ONLY BE MISSING ON GROUP LEVEL 1  
PER0027 GRUPPEN-BEGRIFF DARF NUR BEI STUFE 1 FEHLEN

**Bedeutung**

In der SET-GROUP-ATTRIBUTES (GROUP) -Anweisung wurde auf einer Gruppenstufe groesser als 1 kein Gruppenbegriff ueber den Operanden GROUP-CONTROL (CHANGE) angegeben.

PER0028 USER MODULE '(&00)' WITH LINK ERROR '(&01)'  
PER0028 BENUTZER-MODUL '(&00)' MIT LINK-FEHLER '(&01)'

Warranty: Y

**Bedeutung**

Beim Nachladen des Benutzermoduls (&00) mit LINK trat der Fehler (&01) auf.

(&00): Name bzw. Entry punkt der Benutzer routine

(&01): Returncode des LINK-Makros.

**Maßnahme**

Ueberpruefen, ob die Modulbibliothek mit /SET-TASKLIB (/SYSEFILE TASKLIB) zugewiesen ist und ob der Modul in der zugewiesenen Bibliothek vorhanden ist.

PER0029 NUMBER OF PROCESSED BLOCKS FOR LINK='(&00)':(&01) (&02)  
 PER0029 ANZAHL VERARBEITETER BLOECKE FUER LINK='(&00)':(&01) (&02)  
 Warranty: Y

### Bedeutung

Anzahl der verarbeiteten Bloecke des Bandes/der Bandmenge.

Beim Lesen/Schreiben etikettierter Baender wird zusaetzlich die Anzahl der verarbeiteten Kennsaetze gezeigt.

(&00): Dateikettungsname

(&01): Anzahl Bloecke

(&02): (INCL. n LABELS), n: Anzahl der Kennsaetze.

PER0030 NUMBER OF PROCESSED RECORDS FOR LINK='(&00)' (&01):(&02)  
 PER0030 ANZAHL VERARBEITETER SAETZE FUER LINK='(&00)' (&01):(&02)  
 Warranty: Y

### Bedeutung

Bei Ende des Konvertierungsschrittes wird fuer benannte Dateien zusaetzlich der Dateiname ausgegeben.

(&00): Dateikettungsname

(&01): Dateiname

(&02): Anzahl Saetze.

PER0031 PERCON TERMINATED NORMALLY  
 PER0031 PERCON NORMAL BEENDET  
 Warranty: Y

PER0032 CONVERSION STEP STARTED IN SPITE OF INVALID STATEMENT  
 PER0032 TROTZ FEHLERHAFTER ANWEISUNG KONVERTIERUNGS-SCHRITT GESTARTET

### Bedeutung

In einer Prozedur, einem Stapel-Auftrag wurde trotz fehlerhafter Anweisung ein Konvertierungsschritt gestartet. Die Fehlerstelle wird mit der letzten Meldung angezeigt.

### Maßnahme

Fehlerhafte Anweisung korrigieren und PERCON-Lauf wiederholen.

PER0033 LINK NAME '(&00)' USED MORE THAN ONCE. LAST STATEMENT IS VALID  
 PER0033 DATEIKETTUNGSNAME '(&00)' MEHRFACH VERWENDET. LETZTE ANWEISUNG GUELTIG

### Bedeutung

Fuer den gleichen Dateikettungsnamen wurden zwei FILIN oder FILOUT -Anweisungen angegeben.

### Maßnahme

Die erste Anweisung eventuell mit anderem Dateikettungsnamen wiederholen.

PER0034 VOLUME IDENTIFICATION MISSING, LINK='(&00)'  
PER0034 DATENTRAEGER-KENNZEICHEN FEHLT, LINK='(&00)'

**Bedeutung**

In der VOLIN bzw. VOLOUT -Anweisung bzw. im zugehoerigen /SET-FILE-LINK (FILE) -Kommando wurde der Operand VOLUME nicht angegeben.  
(&00): Dateikettungsname.

**Maßnahme**

VOLUME-Operand angeben und Lauf wiederholen.

PER0035 ISAM KEY NOT ASCENDING, LINK='(&00)'  
PER0035 ISAM-SCHLUESSEL NICHT AUFSTEIGEND, LINK='(&00)'  
Warranty: Y

**Bedeutung**

Bei der Ausgabe auf eine ISAM-Datei wird festgestellt, dass die ISAM-Schluesssel nicht in aufsteigender Folge sind.  
(&00): Dateikettungsname.

**Maßnahme**

SAM-Ausgabedatei verwenden oder Saetze ueber Dienstprogramm SORT nach Schluesssel sortieren.

PER0036 DMS ERROR CODE X'(&01)', LINK='(&00)'. FURTHER INFORMATION: /HELP-MSG DMS(&01)  
PER0036 DVS-FEHLER-CODE X'(&01)', LINK='(&00)'. WEITERE INFORMATION: /HELP-MSG DMS(&01)  
Warranty: Y

**Bedeutung**

Bei der Verarbeitung der Datei wurde ein DVS-Fehler erkannt.  
Der Konvertierungsschritt wird abgebrochen.  
(&00): Dateikettungsname  
(&01): DVS-Fehlerschluesssel.

PER0037 SWITCHOVER THE CHARACTER SET NOT PERMITTED  
PER0037 WECHSEL DES ZEICHENSATZES NICHT ZULAESSIG

**Bedeutung**

Alle Anweisungen muessen demselben CCS angehoren.  
Alle Saetze einer Eingabedatei muessen demselben CCS angehoren.

**Maßnahme**

Nur einen Zeichensatz verwenden.

PER0038 FILE CLOSED BECAUSE OF (&01) ERROR, LINK='(&00)'  
 PER0038 DATEI WEGEN (&01)-FEHLER GESCHLOSSEN, LINK='(&00)'  
 Warranty: Y

### Bedeutung

Beim Lesen der Datei mit Kettungsname (&00) ist ein (&01)-Fehler aufgetreten. Der Konvertierungsschritt wird beendet.

(&00): Dateikettungsname

(&01): PARITY, WLRERR.

### Maßnahme

Informationen zu diesem Fehler siehe DMS-Manual.

Das Verhalten im Fehlerfall kann mit den Operanden PARITY-ERROR, LENGTH-ERROR bzw. INPUT-ERROR der Anweisungen ASSIGN-INPUT-FILE bzw. ASSIGN-INPUT-TAPE gesteuert werden.

PER0039 CCS '(&00)' INCOMPATIBLE WITH SYSOUT, LINK='(&01)'  
 PER0039 CCS '(&00)' UNVERTRÄGLICH MIT SYSOUT, LINK='(&01)'  
 Warranty: Y

### Bedeutung

Es ist nicht möglich, mit dem CCS auf SYSOUT auszugeben.

(&00): Name des erweiterten Zeichensatzes

(&01): Dateikettungsname.

### Maßnahme

Prüfen des verwendeten CCS.

PER0040 WARNING: ILLEGAL DIALOG REQUEST. DEFAULT VALUE ASSUMED IN BATCH MODE  
 PER0040 WARNUNG: UNZULAESSIGE DIALOG-ANFORDERUNG. BEI STAPEL-VERARBEITUNG WIRD STANDARDWERT ANGENOMMEN

### Bedeutung

Der Operandenwert DIALOG (USER(CONV)) ist nur im Dialogbetrieb sinnvoll.

Bei Stapelverarbeitung wird er durch den jeweiligen Standardwert ersetzt.

PER0041 INVALID FIELD FORMAT '(&00)' AT RECORD (&01), LINK='(&02)'  
 PER0041 UNGÜELTIGES FELD-FORMAT '(&00)' BEI SATZ (&01), LINK='(&02)'  
 Warranty: Y

### Bedeutung

Das verarbeitete Feld hat nicht das in der SET-RECORD-MAPPING (RECORD)- oder SET-GROUP-ATTRIBUTES (GROUP)-Anweisung angegebene Format.

(&00): PACKED-DECIMAL (P), ZONED-DECIMAL (Z)

(&01): Satzzaehler

(&02): Dateikettungsname.

PER0042 ILLEGAL LENGTH COMBINATION IN '(&00)' STATEMENT, LINK='(&01)'  
PER0042 UNZULAESSIGE LAENGEN-KOMBINATION IN '(&00)'-ANWEISUNG, LINK='(&01)'  
Warranty: Y

**Bedeutung**

In der genannten Anweisung stimmen die Laengen der beteiligten Felder nicht ueberein oder die Maskenlaenge passt nicht zur Datenlaenge.

(&00): Moegliche fehlerhafte Anweisung:  
SELECT-INPUT-RECORDS (SELECT)  
SET-RECORD-MAPPING (RECORD)  
SET-GROUP-ATTRIBUTES (GROUP)

(&01): Dateikettungsname.

PER0043 LINE LENGTH ((&00)) > MAXIMUM LENGTH (204), LINK='(&01)'  
PER0043 ZEILENLAENGE ((&00)) > MAXIMALER LAENGE (204), LINK='(&01)'

**Bedeutung**

Aus der SET-PAGE-LAYOUT (FORMAT)-Anweisung ergibt sich eine Zeilenlaenge, die groesser ist als die maximal moegliche (204 Zeichen).

(&00): Falsche Zeilenlaenge

(&01): Dateikettungsname.

**Maßnahme**

Operanden LINE-SIZE (LLENGTH), COLUMN-SIZE (LGROUP) und OUTPUT-FORMAT (MODE) ueberpruefen.

Laenge des Satzvorspannes beruecksichtigen.

PER0044 FIELD IN '(&00)' STATEMENT EXCEEDS INPUT/OUTPUT RECORD END, LINK='(&01)'  
PER0044 FELD IN '(&00)'-ANWEISUNG UEBERRAGT EIN-/AUSGABE-SATZENDE, LINK='(&01)'  
Warranty: Y

**Bedeutung**

In der SELECT-INPUT-RECORDS (SELECT) oder SET-RECORD-MAPPING (RECORD) Anweisung wird ein Feld verwendet, welches teilweise oder ganz ausserhalb des Ein-/Ausgabesatzes liegt.

(&00): Fehlerhafte Anweisung

(&01): Dateikettungsname.

PER0045 RECORD '(&00)' NOT IN SEQUENCE, LINK='(&01)'  
PER0045 SATZ '(&00)' NICHT IN FOLGE, LINK='(&01)'  
Warranty: Y

**Bedeutung**

Die Nummer des ersten Satzes, der die in der SELECT-INPUT-RECORDS (SELECT)-Anweisung geforderte Reihenfolge nicht einhaelt wird ausgegeben.

Bei den nachfolgenden Saetzen, die nicht in Folge liegen, wird keine Meldung gebracht.

(&00): Satzzaehler

(&01): Dateikettungsname.

PER0046 ILLEGAL 'VOLIN'/'VOLOUT' STATEMENT DURING FILE PROCESSING  
 PER0046 UNZULAESSIGE 'VOLIN'/'VOLOUT'-ANWEISUNG BEI DATEI-VERARBEITUNG

### **Bedeutung**

Eine zuvor angegebene FILIN-Anweisung hat einen dateiorientierten Konvertierungsschritt eingeleitet.

Im Dialog wird die VOLIN/VOLOUT-Anweisung ignoriert,  
 im Batch wird der Konvertierungsschritt abgebrochen.

PER0047 ILLEGAL 'FILIN'/'FILOUT' STATEMENT DURING VOLUME PROCESSING  
 PER0047 UNZULAESSIGE 'FILIN'/'FILOUT'-ANWEISUNG BEI DATENTRAEGER-VERARBEITUNG

### **Bedeutung**

Eine zuvor angegebene VOLIN-/VOLOUT-Anweisung hat einen datentraeger-orientierten Konvertierungsschritt eingeleitet.

Im Dialog wird die FILIN/FILOUT-Anweisung ignoriert,  
 im Batch wird der Konvertierungsschritt abgebrochen.

PER0048 RETURN CODE X'(&00)' OF USER ROUTINE '(&02)' NOT PERMITTED, LINK='(&01)'  
 PER0048 RETURN-CODE X'(&00)' DER BENUTZER-ROUTINE '(&02)' NICHT ZULAESSIG, LINK='(&01)'  
 Warranty: Y

### **Bedeutung**

Die Benutzerroutine liefert einen unzuverlässigen Returncode.

Der Konvertierungsschritt wird abgebrochen.

(&00): Gelieferter Returncode

(&01): Dateikettungsname

(&02): Name bzw. Entry punkt der Benutzerroutine.

### **Maßnahme**

Benutzerroutine korrigieren und Konvertierungsschritt wiederholen.

PER0049 ONLY ONE 'VOLIN' STATEMENT IS ALLOWED  
 PER0049 NUR EINE 'VOLIN'-ANWEISUNG IST ZULAESSIG

### **Bedeutung**

In einem Konvertierungsschritt ist nur eine VOLIN-Anweisung erlaubt.

### **Maßnahme**

VOLIN-Anweisungen zu einer zusammenfassen oder auf mehrere Konvertierungsschritte verteilen.

PER0050 ILLEGAL 'EDIT'/'POSIT' STATEMENT DURING FILE PROCESSING  
 PER0050 UNZULAESSIGE 'EDIT'/'POSIT'-ANWEISUNG BEI DATEI-VERARBEITUNG

### **Bedeutung**

Eine zuvor eingegebene FILIN Anweisung hat einen dateiorientierten Konvertierungsschritt eingeleitet.

Im Dialog wird die EDIT/POSIT-Anweisung ignoriert,  
 im Batch wird der Konvertierungsschritt abgebrochen.



PER0051 WARNING: XHCS NOT ACTIVE, STATEMENT CCSN '(&00)' IGNORED  
PER0051 WARNUNG: XHCS NICHT AKTIV, ANWEISUNGS-CCSN '(&00)' IGNORIERT

**Bedeutung**

Das Subsystem XHCS ist nicht aktiv, es gibt aber ein Anweisungs-CCSN.

Der CCSN wird ignoriert.

(&00): Anweisungs-CCSN.

PER0052 WARNING: XHCS NOT ACTIVE, FILE CCSN '(&00)' IGNORED, LINK='(&01)'  
PER0052 WARNUNG: XHCS NICHT AKTIV, DATEI-CCSN '(&00)' IGNORIERT, LINK='(&01)'

**Bedeutung**

Das Subsystem XHCS ist nicht aktiv, die Datei besitzt aber einen CCSN.

Der CCSN wird ignoriert.

(&00): CCSN der Datei

(&01): Dateikettungsname.

PER0053 FILE TYPE OR OPEN MODE WRONG FOR FILE '(&00)', LINK='(&01)'  
PER0053 DATEITYP ODER OPEN-MODUS FEHLERHAFT FUER DATEI '(&00)', LINK='(&01)'  
Warranty: Y

**Bedeutung**

Die Ausgabedatei muss eine ISAM- bzw. NK-ISAM-Datei sein und im OPEN-Modus INOUT eroeffnet sein, da FILLER=OUTPUT in der Anweisung SET-RECORD-MAPPING angegeben wurde.

Der Konvertierungsschritt wird abgebrochen.

(&00): Dateiname

(&01): Dateikettungsname.

**Maßnahme**

Anweisung korrigieren und Konvertierungsschritt wiederholen.

PER0054 WARNING: KEY '(&00)' MISSING IN FILE '(&02)', LINK='(&01)'  
PER0054 WARNUNG: SCHLUESSEL '(&00)' IN DATEI '(&02)' NICHT VORHANDEN, LINK='(&01)'  
Warranty: Y

**Bedeutung**

In der Datei gibt es keinen Satz mit dem angegebenen Schluessel.

Der Konvertierungsschritt wird fortgesetzt.

Diese Meldung wird hoechstens zehnmal fuer jeden Dateikettungsnamen gesendet.

(&00): Gesuchter Schluessel

(&01): Dateikettungsname

(&02): Dateiname.

PER0055 'VOLIN' STATEMENT MISSING FOR VOLUME PROCESSING  
PER0055 'VOLIN'-ANWEISUNG FEHLT BEI DATENTRAEGER-VERARBEITUNG

**Bedeutung**

Es wurde eine VOLOUT- aber keine VOLIN-Anweisung angegeben.

PER0056 ILLEGAL COMBINATION OF FILLER=OUTPUT IN SET-RECORD-MAPPING STATEMENT WITH OTHER  
OPERANDS OR STATEMENTS, LINK='(&00)'

PER0056 UNZULAESSIGE KOMBINATION VON FILLER=OUTPUT IN ANWEISUNG SET-RECORD-MAPPING MIT  
ANDEREN OPERANDEN ODER ANWEISUNGEN, LINK='(&00)'

**Bedeutung**

Fuer die Datei mit dem Kettungsnamen (&00) wurde in der Anweisung SET-RECORD-MAPPING FILLER=OUTPUT angegeben. Diese Angabe darf nicht mit

Anweisung SET-PAGE-LAYOUT,

Anweisung SET-GROUP-ATTRIBUTES oder

Operand OVERWRITE=NO in Anweisung ASSIGN-OUTPUT-FILE

fuer dieselbe Datei kombiniert werden.

(&00): Dateikettungsname.

**Maßnahme**

Im Dialogbetrieb die fehlerhaften Anweisungen korrigieren,

im Batchbetrieb den Konvertierungsschritt mit korrigierten

Anweisungen wiederholen.

PER0057 UNRECOVERABLE DEVICE ERROR DURING POSITIONING

PER0057 NICHTBEHEBBARER GERAETE-FEHLER BEIM POSITIONIEREN

Warranty: Y

**Bedeutung**

Beim Positionieren des Eingabedatentraegers trat ein nichtbehebbarer Geraetefehler auf.  
Der Konvertierungsschritt wird abgebrochen.

**Maßnahme**

Geraet ueberpruefen oder Operator verstaendigen.

PER0058 OPERAND '(&00)' ONLY ALLOWED FOR SYSTEM ADMINISTRATOR

PER0058 OPERAND '(&00)' NUR FUER SYSTEMVERWALTER ZUGELASSEN

**Bedeutung**

(&00): VSNNEW = Operand der VOLOUT-Anweisung

PEOT = Operandenwert des Operanden FORWARD der POSIT-Anweisung.

**Maßnahme**

Systemverwalter verstaendigen.

PER0059 '(&00)' REACHED AFTER (&01) (&02) ON INPUT TAPE  
PER0059 '(&00)' NACH (&01) (&02) AUF DEM EINGABE-BAND ERREICHT

**Bedeutung**

Auf dem Eingabe-Band wurde vorzeitig Bandende/Bandanfang erreicht.

(&00): BEGIN-OF-TAPE (BOT)  
          END-OF-TAPE (EOT)  
          DOUBLE-TAPE-MARK (DTM)  
(&01): Anzahl Bloেকে/Bandmarken  
(&02): BLOCKS  
          TAPEMARKS.

PER0060 ERROR IN ASSIGNMENT OF 'SYSDTA' TO DISK DRIVE, LINK='(&00)'  
PER0060 FEHLER BEI ZUWEISUNG VON 'SYSDTA' AUF DISKETTEN-STATION, LINK='(&00)'

**Bedeutung**

(&00): Dateikettungsname.

**Maßnahme**

Zustand des Diskettenlaufwerks ueberpruefen.

PER0061 ERROR IN ASSIGNMENT OF 'SYSDTA' TO CARD READER, LINK='(&00)'  
PER0061 FEHLER BEI ZUWEISUNG VON 'SYSDTA' AUF LOCHKARTEN-LESER, LINK='(&00)'

**Bedeutung**

(&00): Dateikettungsname.

**Maßnahme**

Zustand des Kartenlesers ueberpruefen.

PER0062 OUTPUT INTO REMAIN FILE NOT ALLOWED, LINK='(&00)'  
PER0062 AUSGABE IN RESTDATEI NICHT ERLAUBT, LINK='(&00)'

**Bedeutung**

In der Anweisung START-TAPE-PROCESSING wird der Dateikettungsname einer Restdatei angegeben.

(&00): Dateikettungsname.

**Maßnahme**

Verwendete Dateikettungsnamen ueberpruefen.

PER0063 (&00) ERROR FOR (&01) (&02), LINK='(&03)'. REPLY (I=IGNORE; S=SKIP; H=HALT)  
 PER0063 (&00)-FEHLER BEI (&01) (&02), LINK='(&03)'. ANTWORT (I=IGNORE; S=SKIP; H=HALT)

### Bedeutung

Es ist ein Paritäts- oder Laengenfehler bei Satz-/Block-Nummer (&02) der Eingabedatei mit dem Dateikettungsnamen (&03) aufgetreten.

(&00): PARITY

WLRERR

(&01): BLOCK

RECORD

(&02): Block-/Satzzaehler

(&03): Dateikettungsname.

### Maßnahme

Moegliche Massnahmen:

- I: Fehler ignorieren oder
- S: Fehlerhaften Block/Satz ueberlesen oder
- H: Konvertierungsschritt abbrechen.

PER0064 RECORD LENGTH FIELD DESTROYED, LINK='(&00)'  
 PER0064 SATZLAENGEN-FELD ZERSTOERT, LINK='(&00)'

### Bedeutung

Bei der Zusammenstellung eines variablen Ausgabesatzes ueber die SET-RECORD-MAPPING (RECORD) -Anweisung wurde das Satzlaengenfeld ueberschrieben.

(&00): Dateikettungsname.

### Maßnahme

Mit Ausgabesatzposition 5 beginnen oder Ausgabedatei mit fixer Satzlaenge verwenden.

PER0065 WARNING: GROUP BREAK USER LINE INCOMPLETE, LINK='(&00)'  
 PER0065 WARNUNG: GRUPPENWECHSEL-BENUTZERZEILE UNVOLLSTAENDIG, LINK='(&00)'

### Bedeutung

Ein Feld der Gruppenwechselzeile ueberschreitet die Zeilenlaenge, welche sich durch LINE-SIZE (LLENGTH), COLUMN-SIZE (LGROUP) und OUTPUT-FORMAT (MODE) der SET-PAGE-LAYOUT (FORMAT) Anweisung ergibt, bzw. liegt ausserhalb des Eingabesatzes.

Das angegebene Feld wird nicht uebertragen. Die Meldung wird nur bei der ersten Zeile auf die dies zutrifft ausgegeben.

(&00): Dateikettungsname.

### Maßnahme

- Gruppenwechsel-Zeile kuerzen oder
- formataufbereitete Zeilen verlaengern oder
- einen anderen Teil des Eingabesatzes verwenden.

PER0066 CODE TRANSLATION TABLE IS MISSING OR INVALID  
PER0066 CODE-UMSETZTABELLE FEHLT ODER FEHLERHAFT

**Bedeutung**

Bei der SET-RECORD-MAPPING (RECORD) -Anweisung wurde im CODE-TRANSLATION (ALTER)-Operanden eine Umsetztabelle-Datei angegeben, die nicht vorhanden ist oder nicht gelesen werden kann.

**Maßnahme**

Datei der Umsetztabelle ueberpruefen.

PER0067 ILLEGAL 'DEVICE' OPERAND IN 'EDIT'/'POSIT' STATEMENT  
PER0067 UNZULAESSIGER DEVICE-OPERAND BEI 'EDIT'/'POSIT'-ANWEISUNG

**Bedeutung**

Wird beim Bandeditieren eine VOLOUT-Anweisung angegeben, so muss diese den Operanden DEVICE=PRINTER bzw. DISPLAY enthalten.

PER0068 ILLEGAL OPERATION IN USER ROUTINE, LINK='(&00)'  
PER0068 UNZULAESSIGE OPERATION IN BENUTZER-ROUTINE, LINK='(&00)'

**Bedeutung**

Bei OPEN=UPDATE in der FILIN-Anweisung bzw. im /SET-FILE-LINK (FILE) Kommando kann bei der Eingabe-Satzbearbeitung in einem Benutzermodul weder der gelesene Satz ersetzt noch ein Satz eingefuegt werden.  
(&00): Dateikettungsname.

**Maßnahme**

Benutzer-Routine ueberpruefen und Lauf wiederholen.

PER0069 ERROR DURING TAPE DUPLICATION, LINK='(&00)'  
PER0069 FEHLER BEI BAND-DUPLIZIERUNG, LINK='(&00)'

Warranty: Y

**Bedeutung**

Wegen eines Fehlers konnte keine vollstaendige Kopie der Eingabebeaender erstellt werden.  
(&00): Dateikettungsname.

**Maßnahme**

Dupliziertvorgang mit anderem Ausgabeband oder mit mehreren Ausgabebaendern wiederholen, oder auf freie Bandstation warten.

PER0070 PERCON INTERRUPTED. REPLY (DISP = DISPLAY STATE; CONT = CONTINUE; START = NEXT  
CONV.; FIN = NEXT INPUT FILE; TERM = ABORT)  
PER0070 PERCON UNTERBROCHEN. ANTWORT (DISP=STATUS ANZEIGEN; CONT=FORTSETZEN;  
START=NAECHSTE KONV.BEG.; FIN=NAECHSTE EING.DATEI; TERM=ABBRECHEN)

**Bedeutung**

PERCON wurde mit der Unterbrechungstaste und dem /INTR- bzw. SEND-MESSAGE-Kommando unterbrochen.

Eine Aktion wird als Massnahme angefordert.

**Maßnahme**

Moegliche Massnahmen:

- DISP: Ablaufzustand anzeigen (aktuelle Zaehler) oder
- CONT: Verarbeitung fortsetzen oder
- START: Konvertierungsschritt beenden oder
- FIN: Lesen aus aktueller Eingabedatei beenden oder
- TERM: Verarbeitung abbrechen.

PER0071 WARNING: ILLEGAL SUM FIELD FORMAT, FORMAT 'Z' ASSUMED  
PER0071 WARNUNG: UNZULAESSIGES SUMMENFELD-FORMAT, FORMAT 'Z' ANGENOMMEN

**Bedeutung**

Im SUM-Operanden der GROUP-Anweisung wurde ein Summenfeld angegeben, dessen Eingabeformat ungleich 'P'/'Z' oder dessen Ausgabeformat ungleich 'Z'/'Z'/'-Z' ist.

PER0072 DUPLICATE OPERAND IN '(&00)' STATEMENT  
PER0072 OPERAND IN '(&00) '-ANWEISUNG MEHRFACH

**Bedeutung**

PERCON meldet, dass ein Operand in der Anweisung (&00) mehrfach angegeben wurde. (&00): Bezeichnung der fehlerhaften Anweisung.

**Maßnahme**

Im Dialogbetrieb die fehlerhafte Anweisung korrigieren,  
im Batchbetrieb den Konvertierungsschritt mit korrigierter Anweisung wiederholen.

PER0073 STATEMENTS WITH ISO-CCSN '(&00)' NOT SUPPORTED  
PER0073 ANWEISUNGEN MIT ISO-CCSN '(&00)' NICHT UNTERSTUETZT

**Bedeutung**

Anweisungen mit einem ISO-CCSN werden nicht unterstuetzt.  
(&00): Anweisungs-CCSN.

**Maßnahme**

Anweisungs-CCS korrigieren.

PER0074 FILE WITH ISO-CCSN '(&00)' NOT SUPPORTED, LINK='(&01)'  
PER0074 DATEI MIT ISO-CCSN '(&00)' NICHT UNTERSTUETZT, LINK='(&01)'

**Bedeutung**

Dateien mit einem ISO-CCSN werden nicht unterstuetzt.

(&00): CCSN der Datei

(&01): Dateikettungsname.

**Maßnahme**

Datei-CCS korrigieren.

PER0075 STATEMENT CCSN '(&00)' UNKNOWN  
PER0075 ANWEISUNGS-CCSN '(&00)' UNBEKANNT

**Bedeutung**

Der CCSN der Anweisungen ist nicht bekannt.

(&00): Anweisungs-CCSN.

**Maßnahme**

Anweisungs-CCS korrigieren.

PER0076 FILE CCSN '(&00)' UNKNOWN, LINK='(&01)'  
PER0076 DATEI-CCSN '(&00)' UNBEKANNT, LINK='(&01)'

**Bedeutung**

Der CCSN der Datei ist nicht bekannt.

(&00): CCSN der Datei

(&01): Dateikettungsname.

**Maßnahme**

Datei-CCS korrigieren.

PER0077 INSTALLATION ERROR OF STATEMENT CCS '(&00)'  
PER0077 INSTALLATIONS-FEHLER BEI ANWEISUNGS-CCS '(&00)'

**Bedeutung**

Das CCS der Anweisungen wurde fehlerhaft installiert.

(&00): Anweisungs-CCSN.

**Maßnahme**

Systemverwalter verstaendigen.

PER0078 INSTALLATION ERROR OF FILE CCS '(&00)', LINK='(&01)'  
PER0078 INSTALLATIONS-FEHLER BEI DATEI-CCS '(&00)', LINK='(&01)'

**Bedeutung**

Das CCS der Datei wurde fehlerhaft installiert.

(&00): CCSN der Datei

(&01): Dateikettungsname.

**Maßnahme**

Systemverwalter verstaendigen.

PER0079 RECORD LENGTH SPECIFIKATION TOO SMALL, LINK='(&00)'  
PER0079 SATZLAENGEN-ANGABE ZU KLEIN, LINK='(&00)'

**Bedeutung**

Fuer eine variable Ein-/Ausgabedatei wurde eine Satzlaenge kleiner als 5 angegeben oder bei einer ISAM-Datei liegt der ISAM-Schluessel nicht vollstaendig im Satz.

(&00): Dateikettungsname.

**Maßnahme**

Satzlaenge ueber /SET-FILE-LINK (FILE) -Kommando oder FILIN/FILEOUT -Anweisung ver-groessern.

PER0080 DEVICE TYPE UNKNOWN  
PER0080 GERAETETYP UNBEKANNT

**Bedeutung**

Der angeforderte Geraetetyp ist in dieser BS2000-Version nicht verfuegbar.

**Maßnahme**

Systemverwalter befragen, welche Geraetetypen in diesem System generiert sind.

PER0081 NO STANDARD LABELS ON VOLUME '(&00)' FOR LINK='(&01)'  
PER0081 KEINE STANDARD-KENNSAETZE AUF BAND '(&00)', LINK='(&01)'

**Bedeutung**

Ein Folgeband eines Eingabe-MF/MV-Sets traegt im Gegensatz zum ersten Eingabeband keine Standardsaetze.

(&00): VSN

(&01): Dateikettungsname.

**Maßnahme**

Ueberpruefen, ob das Folgeband wirklich zum MF/MV-Set gehoert.



PER0082 WRONG FILE CONTINUATION ON VOLUME '(&00)', LINK='(&01)'  
PER0082 FALSCHER DATEI-FORTSETZUNG AUF BAND '(&00)', LINK='(&01)'

**Bedeutung**

Der erste Dateiabchnitt auf einem Folgeband ist keine Fortsetzung des letzten Dateiabchnitts auf dem vorhergehenden Band.

(&00): VSN

(&01): Dateikettungsname.

**Maßnahme**

Überprüfen, ob die genannten Eingabebänder in dieser Reihenfolge wirklich ein MF/MV-Set bilden.

PER0083 ONLY OWNER OR SYSTEM ADMINISTRATOR MAY USE VOLUME '(&00)', LINK='(&01)'  
PER0083 NUR EIGENTÜMER ODER SYSTEMVERWALTER DARF BAND '(&00)' VERWENDEN, LINK='(&01)'

**Bedeutung**

Der Zugriffsvermerk im VOL1-Kennsatz gestattet dem Benutzer keine Verarbeitung des Eingabebandes.

(&00): VSN

(&01): Dateikettungsname.

**Maßnahme**

Das Band vom Eigentümer oder vom Systemverwalter verarbeiten lassen.

PER0084 NOT ENOUGH OUTPUT TAPES SPECIFIED, \*SCRATCH IS REQUESTED, LINK='(&00)'  
PER0084 NICHT GENÜGENDE AUSGABEBÄNDER ANGEZEIGT, \*SCRATCH WIRD ANGEFORDERT, LINK='(&00)'

**Bedeutung**

Für eine vollständige Kopie wurde die Ausgabebandmenge zu klein angegeben. Vom Operator wird ein \*SCRATCH- (=Arbeits-) Band angefordert. Die Band-Duplizierung wird dann auf diesem Band fortgesetzt.

(&00): Dateikettungsname.

PER0085 ISAM KEY FIELD DESTROYED, LINK='(&00)'  
PER0085 ISAM-SCHLÜSSEL-FELD ZERSTÖRT, LINK='(&00)'

**Bedeutung**

Bei der Zusammenstellung eines Ausgabesatzes über die Anweisung SET-RECORD-MAPPING mit FILLER=OUTPUT oder im Benutzer-Anschluss für die Ausgabe wurde das ISAM-Schlüsselfeld überschrieben.

Der Konvertierungsschritt wird abgebrochen.

(&00): Dateikettungsname.

**Maßnahme**

Anweisung korrigieren und Konvertierungsschritt wiederholen.

PER0086 WARNING: BYTE COUNTER OVERFLOW AT RECORD (&00), LINK='(&01)'  
 PER0086 WARNUNG: BYTE-ZAEHLER-UEBERLAUF BEI SATZ (&00), LINK='(&01)'

### Bedeutung

Der intern mitgefuehrte Byte-Zaehler hat den angegebenen Maximalwert von  $2^{32} - 1$  erreicht und kann nicht mehr erhoeht werden. Der Konvertierungsschritt wird jedoch fortgesetzt.

(&00): Satzzaehler

(&01): Dateikettungsname.

### Maßnahme

Ueberpruefen, ob die Datei ab dem angegebenen Satzzaehler richtig verarbeitet wurde.

PER0087 CONTRADICTORY OPERANDS '(&00)' AND '(&01)' IN '(&03)' STATEMENT, LINK='(&02)'  
 PER0087 WIDERSPRUECHLICHE OPERANDEN '(&00)' UND '(&01)' IN '(&03)'-ANWEISUNG,  
 LINK='(&02)'

### Bedeutung

(&00): Operand-1

(&01): Operand-2

(&02): Dateikettungsname

(&03): Anweisung.

### Maßnahme

Einer der beiden angegebenen Operanden muss weggelassen oder korrigiert werden.

PER0088 ILLEGAL 'DEVICE' OPERAND IN FILOUT STATEMENT, LINK='(&00)'  
 PER0088 UNZULAESSIGER DEVICE-OPERAND IN FILOUT-ANWEISUNG, LINK='(&00)'

### Bedeutung

Wenn eine VOLIN-Anweisung angegeben wurde ist der Operand 'DEVICE=PRINTER bzw. =DISPLAY' in der FILOUT-Anweisung nicht erlaubt.

### Maßnahme

FILOUT-Anweisung mit 'FILENAME=Dateiname' angeben um das Band in eine Datei auszugeben oder VOLOUT-Anweisung mit 'DEVICE=PRINTER oder =DISPLAY' um das Band auf den Drucker bzw. Bildschirm auszugeben.

PER0089 DUPLICATION PROHIBITED DURING TAPE EDITING  
 PER0089 WAEHREND DES BAND-EDITIERENS IST DUPLIZIEREN NICHT ERLAUBT

### Bedeutung

In einem Konvertierungsschritt kann entweder nur dupliziert oder nur editiert werden.

PER0090 'FORMAT' STATEMENT NOT ALLOWED DURING TAPE DUPLICATION  
PER0090 'FORMAT'-ANWEISUNG BEI BAND-DUPLIZIERUNG UNZULAESSIG

**Bedeutung**

Beim Duplizieren (Kopieren von Baendern) ist eine FORMAT-Anweisung nicht zulaessig.

**Maßnahme**

FORMAT-Anweisung weglassen.

PER0091 ILLEGAL 'DEVICE=PRINTER OR DISPLAY' OPERAND DURING DUPLICATION  
PER0091 UNZULAESSIGER 'DEVICE=PRINTER BZW. DISPLAY'-OPERAND BEI DUPLIZIERUNG

**Bedeutung**

Beim Duplizieren ist die Angabe des Operanden DEVICE=PRINTER oder =DISPLAY in der VOLOUT-Anweisung nicht zulaessig.

PER0092 WARNING: REPAIRING OF FILE '(&00)', LINK='(&01)'  
PER0092 WARNUNG: DATEI '(&00)' WIRD WIEDERHERGESTELLT, LINK='(&01)'  
Warranty: Y

**Bedeutung**

Beim Oeffnen der Datei (&00) mit dem Dateikettungsnamen (&01) wurde ein Fehler festgestellt. Im OPENC-Ausgang wird mittels Makro VERIF die Datei wiederhergestellt.

(&00): Dateiname

(&01): Dateikettungsname.

**Maßnahme**

Ausgabe auf Vollstaendigkeit ueberpruefen.

PER0093 REPAIRING OF FILE '(&00)', LINK='(&01)'. REPLY (Y=YES; N=NO)  
PER0093 DATEI '(&00)' WIEDERHERSTELLEN, LINK='(&01)'. ANTWORT (Y=JA; N=NEIN)  
Warranty: Y

**Bedeutung**

Beim Oeffnen der Datei (&00) mit dem Dateikettungsnamen (&01) wurde ein Fehler festgestellt. Die Datei kann im OPENC-Ausgang mit dem Makro VERIF wiederhergestellt werden.

(&00): Dateiname

(&01): Dateikettungsname.

**Maßnahme**

Moegliche Massnahmen:

- Y: Datei wiederherstellen

- N: Datei nicht wiederherstellen

PER0094 FILE '(&00)' HAS TO BE REPAIRED, LINK='(&01)'  
PER0094 DATEI '(&00)' MUSS WIEDERHERGESTELLT WERDEN, LINK='(&01)'  
Warranty: Y

**Bedeutung**

Beim Oeffnen der Datei (&00) mit dem Dateikettungsnamen (&01) wurde ein Fehler festgestellt. Die Datei wird im OPENC-Ausgang nicht mit dem Makro VERIF wiederhergestellt.

(&00): Dateiname

(&01): Dateikettungsname.

**Maßnahme**

Datei wiederherstellen und Konvertierungsschritt wiederholen.

PER0095 OBJECT MODULE PCNSR9 WITH LINK ERROR X'(&00)'. RUN IS ABORTED  
PER0095 BINDEMODUL PCNSR9 MIT LINK-FEHLER X'(&00)'. LAUF WIRD ABGEBROCHEN  
Warranty: Y

**Bedeutung**

Beim Nachladen des Bindemoduls PCNSR9 trat ein LINK-Fehler auf.

(&00): Fehlerschlüssel des LINK-Makros.

**Maßnahme**

Modulbibliothek OMLPERCON ueberpruefen.

PER0096 FILE '(&02)' WITHOUT ALTERNATE KEY '(&00)', LINK='(&01)'  
PER0096 DATEI '(&02)' OHNE SEKUNDAER-SCHLUESSEL '(&00)', LINK='(&01)'

**Bedeutung**

Der in der Anweisung SET-RECORD-MAPPING angegebene Sekundaerschlüssel ist nicht vorhanden. Der Konvertierungsschritt wird abgebrochen.

(&00): Sekundaerschlüssel

(&01): Dateikettungsname

(&02): Dateiname.

**Maßnahme**

Anweisung korrigieren und Konvertierungsschritt wiederholen.

PER0097 WARNING: FILE '(&01)' HAS NO ALTERNATE INDEX, LINK='(&00)'. PROCESSING CONTINUES  
PER0097 WARNUNG: DATEI '(&01)' BESITZT KEINEN SEKUNDAERSCHLUESSEL, LINK='(&00)'.  
VERARBEITUNG WIRD FORTGESETZT

**Bedeutung**

(&00): Dateikettungsname

(&01): Dateiname.

PER0098 ISAM KEY '(&01)' ALREADY EXISTS, LINK='(&00)'  
PER0098 ISAM-SCHLUESSEL '(&01)' EXISTIERT BEREITS, LINK='(&00)'  
Warranty: Y

**Bedeutung**

Ein Satz konnte in die Ausgabedatei nicht eingefuegt werden, weil der Satz mit dem Schluessel bereits existiert und in der Anweisung ASSIGN-OUTPUT-FILE OVERWRITE =NO angegeben wurde.

(&00): Dateikettungsname

(&01): ISAM-Schluessel.

PER0099 DMS ERROR CODE X'(&01)' DURING DETERMINING ALTERNATE KEYS, LINK='(&00)'. FURTHER  
INFORMATION: /HELP-MSG DMS(&01)  
PER0099 DVS-FEHLER-CODE X'(&01)' BEIM ERMITTELN DER SEKUNDAER-SCHLUESSEL, LINK='(&00)'.  
WEITERE INFORMATION: /HELP-MSG DMS(&01)  
Warranty: Y

**Bedeutung**

Beim Ermitteln der Sekundaerschluessel wurde ein DVS-Fehler erkannt.  
Der Konvertierungsschritt wird abgebrochen.

(&00): Dateikettungsname

(&01): DVS-Fehlerschluessel.

PER0100 DMS ERROR CODE X'(&01)' DURING CREATING ALTERNATE KEYS, LINK='(&00)'. FURTHER  
INFORMATION: /HELP-MSG DMS(&01)  
PER0100 DVS-FEHLER-CODE X'(&01)' BEIM AUFBAUEN DER SEKUNDAER-SCHLUESSEL, LINK='(&00)'.  
WEITERE INFORMATION: /HELP-MSG DMS(&01)  
Warranty: Y

**Bedeutung**

Beim Aufbauen der Sekundaerschluessel wurde ein DVS-Fehler erkannt.  
Der Konvertierungsschritt wird abgebrochen.

(&00): Dateikettungsname

(&01): DVS-Fehlerschluessel.

PER0101 ALTERNATE INDEX '(&01)' ALREADY DEFINED WITH OTHER ATTRIBUTES, LINK='(&00)'  
PER0101 SEKUNDAER-SCHLUESSEL '(&01)' MIT ANDEREN EIGENSCHAFTEN BEREITS DEFINIERT,  
LINK='(&00)'  
Warranty: Y

**Bedeutung**

Der zu erzeugende Sekundaerschluessel existiert bereits. Fuer ihn sind jedoch andere Eigenschaften definiert als in der Eingabedatei, die in der Anweisung ASSIGN-OUTPUT-FILE im Parameter ALTERNATE-INDEX angegeben wurde.

Der Konvertierungsschritt wird abgebrochen.

(&00): Dateikettungsname

(&01): Sekundaerschluessel.

PER0102 DUPLICATE VALUES FOR ALTERNATE INDEX '(&01)', LINK='(&00)'  
PER0102 GLEICHE WERTE FUER SEKUNDAER-SCHLUESSEL '(&01)', LINK='(&00)'  
Warranty: Y

**Bedeutung**

Gleiche Werte des Sekundaerschlüssels kommen in verschiedenen Sätzen vor.  
Der Sekundaerschlüssel ist jedoch mit DUPKEY=NO zu erzeugen.  
Der Konvertierungsschritt wird abgebrochen.

(&00): Dateikettungsname

(&01): Sekundaerschlüssel.

PER0103 RECORD TOO SMALL FOR ALTERNATE INDEX '(&01)', LINK='(&00)'  
PER0103 SATZ ZU KLEIN FUER SEKUNDAER-SCHLUESSEL '(&01)', LINK='(&00)'  
Warranty: Y

**Bedeutung**

Beim Erstellen des Sekundaerschlüssels wurde ein Satz gelesen, der zu kurz ist, um den Sekundaerschlüssel enthalten zu können.  
Der Konvertierungsschritt wird abgebrochen.

(&00): Dateikettungsname

(&01): Sekundaerschlüssel.

PER0104 \*\*\* PERCON VERSION '(&00)' NOT AVAILABLE. RUN IS ABORTED \*\*\*  
PER0104 \*\*\* PERCON-VERSION '(&00)' NICHT VERFUEGBAR. LAUF WIRD ABGEBROCHEN \*\*\*  
Warranty: Y

**Bedeutung**

In einem Startkommando oder in einem Unterprogramm-Aufruf von PERCON wurde eine Version angefordert, die nicht installiert ist.  
Der Lauf wird abgebrochen.

(&00): Angeforderte PERCON-Version.

**Maßnahme**

Andere Version auswählen oder die gewünschte Version im System installieren lassen.

PER0105 SYNTACTICAL ERROR IN VERSION '(&00)'. RUN IS ABORTED  
PER0105 VERSION '(&00)' SYNTAKTISCH FALSCH. LAUF WIRD ABGEBROCHEN  
Warranty: Y

**Bedeutung**

Die Syntax der Version, die als Parameter beim Unterprogramm-Aufruf an PERCON uebergeben wurde, ist fehlerhaft.

Folgende Syntax ist fuer die Versions-Angabe erlaubt:

n.n / n.na / n.nann

nn.n / nn.na / nn.nann

(n ... Dezimalziffer / a ... Buchstabe)

(&00): falsche Version.

**Maßnahme**

Die Versions-Angabe beim Unterprogramm-Aufruf von PERCON pruefen und den Lauf nach der Korrektur wiederholen.

PER0106 SUBSYSTEM '(&00)' NOT AVAILABLE AT THE MOMENT. RUN IS ABORTED  
PER0106 SUBSYSTEM '(&00)' ZUR ZEIT NICHT VERFUEGBAR. LAUF WIRD ABGEBROCHEN  
Warranty: Y

**Bedeutung**

(&00): Name des Subsystems.

**Maßnahme**

PERCON-Lauf wiederholen, wenn Subsystem verfuegbar.

PER0107 ERROR CODE X'(&00)' DURING MACRO '(&01)'. RUN IS ABORTED  
PER0107 FEHLER-CODE X'(&00)' BEI MAKRO '(&01)'. LAUF WIRD ABGEBROCHEN  
Warranty: Y

**Bedeutung**

(&00): Makro-Fehlerschluessel

(&01): Name des Makros.

**Maßnahme**

Fehlerschluessel des Makros ueberpruefen und entsprechend reagieren.  
Lauf wiederholen.

PER0109 WARNING: DMS ERROR X'(&01)'. OUTPUT FILE '(&02)' WITHOUT CCSN '(&03)',  
 LINK='(&00)'. FURTHER INFORMATION: /HELP-MSG DMS(&01)  
 PER0109 WARNUNG: DVS-FEHLER X'(&01)'. AUSGABEDATEI '(&02)' OHNE CCSN '(&03)',  
 LINK='(&00)'. WEITERE INFORMATION: /HELP-MSG DMS(&01)  
 Warranty: Y

**Bedeutung**

Der CCSN konnte nicht in den Katalog-Eintrag der Ausgabedatei geschrieben werden.  
 PERCON wird fortgesetzt.

(&00): Dateikettungsname

(&01): DVS-Fehlerschlüssel

(&02): Name der Ausgabedatei

(&03): CCSN.

**Maßnahme**

Kommando /MODIFY-FILE-ATTRIBUTES zum Eintragen des CCSN in den  
 Katalog-Eintrag verwenden.

PER0110 BS2000-VERSION LOWER THAN VERSION '(&00)'. RUN IS ABORTED  
 PER0110 BS2000-VERSION KLEINER VERSION '(&00)'. LAUF WIRD ABGEBROCHEN  
 Warranty: Y

**Bedeutung**

PERCON benötigt mindestens die BS2000-Version (&00).

PER0111 OBJECT MODULE PCNSR9 IN LIBRARY '(&01)' WITH BIND ERROR X'(&00)'. RUN IS ABORTED  
 PER0111 BINDEMOMUL PCNSR9 IN BIBLIOTHEK '(&01)' MIT BIND-FEHLER X'(&00)'. LAUF WIRD  
 ABGEBROCHEN  
 Warranty: Y

**Bedeutung**

Beim Nachbinden des Bindemoduls PCNSR9 trat ein BIND-Fehler auf.

(&00): Fehlerschlüssel des BIND-Makros

(&01): benutzte Modulbibliothek.

**Maßnahme**

Benutzte Modulbibliothek ueberpruefen.



---

# Literatur

[1] **BS2000/OSD-BC V2.0**

Einführung in das DVS  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an den nichtprivilegierten Anwender und an die Systembetreuung.

*Inhalt*

Es beschreibt die Dateiverwaltung und -verarbeitung im BS2000.

Themenschwerpunkte:

- Datei- und Katalogverwaltung
- Dateien und Datenträger
- Datei- und Datenschutz
- OPEN-, CLOSE-, EOV-Verarbeitung
- DVS-Zugriffsmethoden (SAM, ISAM,...)

[2] **XHCS (BS2000/OSD)**

8-bit-Code-Verarbeitung im BS2000/OSD  
Benutzerhandbuch

*Zielgruppe*

Anwender der Zugriffsmethoden DCAM, TIAM und UTM sowie Systemverwalter, Anwender, die von EHCS auf XHCS umstellen.

*Inhalt*

XHCS (Extended Host Code Support) ist ein Softwareprodukt des BS2000/OSD.

Es ermöglicht Ihnen erweiterte Zeichensätze bei 8-bit-Datenstationen zu nutzen.

XHCS ist die zentrale Informationsquelle über die codierten Zeichensätze im BS2000/SD.

XHCS löst EHCS ab.

- [3] **BS2000/OSD-BC V2.0A**  
Einführung in die Systembetreuung  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an die Systembetreuung und das Operating des Betriebssystems BS2000/OSD.

*Inhalt*

Es sind u.a. folgende Themen zur Verwaltung und Überwachung des BS2000/OSD-Grundausbaus enthalten: Systemeinführung, Parameterservice, Job- und Tasksteuerung, Speicher-, Geräte-, Benutzer- und Dateiverwaltung, Privilegienvergabe, Accounting und Operatorfunktionen.

- [4] **SORT (BS2000)**  
**SDF-Format**  
Tabellenheft

*Zielgruppe*

Mit SORT (SDF-Format) vertraute Anwender

*Inhalt*

- Genereller Ablauf einer Sortierung
- Auflistung aller Anweisungen
- Beschreibung der Unterprogrammchnittstelle
- Zugriffsmethode SORTZM

- [5] **PERCON (BS2000/OSD)**  
Anweisungsformate (SDF-Format)  
Tabellenheft

*Zielgruppe*

BS2000/OSD-Anwender, insbesondere der mit PERCON vertraute Benutzer

*Inhalt*

Das Tabellenheft dient als Nachschlagewerk und enthält eine Zusammenstellung aller PERCON-Anweisungen in alphabetischer Reihenfolge.

- [6] **BINDER (BS2000/OSD)**  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an Software-Entwickler

*Inhalt*

Es beschreibt die BINDER-Funktionen und enthält Beispiele dazu. Im Nachschlageteil sind die BINDER-Anweisungen und der Makroaufruf BINDER beschrieben. BINDER V1.1A kann auch in BS2000 V10.0A eingesetzt werden.

[7] **Einführung in die XS-Programmierung  
(für ASSEMBLER-Programmierer) (BS2000)**

Benutzerhandbuch

*Zielgruppe*

- Programmierer
- System-Programmierer

*Inhalt*

- Die Adressierungsmodi der ab BS2000 V9.0 unterstützten Zentraleinheiten und wie sie sich auf die ASSEMBLER-Befehle auswirken
- Programmierhinweise für ASSEMBLER-Programmierer, die den erweiterten Adreßraum von XS-Anlagen nutzen oder ihre Programme adressierungsmodus-unabhängig und portabel gestalten wollen;

*Einsatz*

TU-, TPR-Programme

[8] **SDF V4.0A (BS2000/OSD)**

Einführung in die Dialogschnittstelle SDF

Benutzerhandbuch

*Zielgruppe*

BS2000/OSD-Anwender

*Inhalt*

Das Handbuch beschreibt die Dialog-Eingabe von Kommandos und Anweisungen im SDF-Format. Ein Schnelleinstieg mit leicht nachvollziehbaren Beispielen und weitere umfangreiche Beispiele erleichtern die Anwendung. SDF-Syntaxdateien werden erklärt.

[9] **JV V11.2A (BS2000/OSD)**

Jobvariablen

Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich sowohl an den nichtprivilegierten Anwender als auch an die Systembetreuung.

*Inhalt*

Es beschreibt die Verwaltung und die verschiedenen Einsatzmöglichkeiten von Jobvariablen. Die Kommandobeschreibungen sind getrennt nach den Funktionsbereichen der JVs aufgeführt. Die Makroaufrufe sind in einem eigenen Kapitel beschrieben.

- [10] **BS2000/OSD-BC V2.0A**  
Makroaufrufe an den Ablaufteil  
Benutzerhandbuch
- Zielgruppe*  
Das Handbuch wendet sich an alle BS2000/OSD-Assembler-Programmierer.
- Inhalt*  
Das Handbuch enthält eine Zusammenstellung der Makroaufrufe an den Ablaufteil, die ausführliche Beschreibung jedes Makroaufrufs mit Hinweisen und Beispielen, einschließlich der Jobvariablen-Makros, sowie einen ausführlichen allgemeinen Lernteil.
- [11] **BS2000/OSD-BC V2.0**  
Kommandos Band 1 bis 4  
Benutzerhandbuch
- Zielgruppe*  
Das Handbuch wendet sich sowohl an den nichtprivilegierten BS2000/OSD-Anwender als auch an die Systembetreuung.
- Inhalt*  
Es enthält die BS2000/OSD-Kommandos (Grundausbau und ausgewählte Produkte). Die Einleitung gibt Hinweise zur Kommandoeingabe. Die Kommandobeschreibung zeigt u.a. die Privilegien zur Nutzung der Kommandos.  
Mit SDF-P können Anwender bei SHOW-Kommandos die Ausgabe in strukturierte S-Variablen nutzen, die im Band 4 beschrieben sind.
- [12] **ARCHIVE (BS2000/OSD)**  
Benutzerhandbuch
- Zielgruppe*
- BS2000/OSD-Benutzer
  - BS2000/OSD-Systemverwalter
  - BS2000/OSD-Operator
- Inhalt*  
Funktionen und Anweisungen des Programms ARCHIVE zur logischen Datensicherung. ARCHIVE dient zur Sicherung, Rekonstruktion und zum Transfer von Dateien und Jobvariablen.
- [13] **BS2000/OSD-BC V2.0**  
DVS-Makros  
Benutzerhandbuch
- Zielgruppe*  
Das Handbuch wendet sich an Assembler-Programmierer.
- Inhalt*  
Es beschreibt die Makroschnittstellen des DVS für den BS2000/OSD-Grundausbau. Es werden kurz die programmierungsrelevanten zugriffsmethoden-spezifischen Besonderheiten und anschließend die Makros in lexikalischer Form beschrieben.

- [14] **BS2000/OSD-BC**  
Bindelader-Starter  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an Software-Entwickler und geübte BS2000/OSD-Benutzer.

*Inhalt*

Es beschreibt die Funktionen, die Unterprogrammschnittstelle, die XS-Unterstützung und den Aufruf des Bindeladers DBL. Daran anschließend sind die Kommandos zum Aufruf des Laders ELDE beschrieben.

- [15] **SDF-P V2.0A (BS2000/OSD)**  
Programmieren in der Kommandosprache  
Benutzerhandbuch

*Zielgruppe*

Das Handbuch wendet sich an BS2000/OSD-Anwender und die Systembetreuung.

*Inhalt*

SDF-P ist eine strukturierte Prozedursprache im BS2000. Nach einer Einführung werden Kommandos, Funktionen und Makros ausführlich beschrieben. SDF-P V2.0A kann unter BS2000/OSD-BC V1.0 nur mit VAS 2.0A und SDF V4.0 eingesetzt werden.

## Bestellen von Handbüchern

Die aufgeführten Handbücher finden Sie mit ihren Bestellnummern im *Druckschriftenverzeichnis* der Siemens Nixdorf Informationssysteme AG. Neu erschienene Titel finden Sie in den *Druckschriften-Neuerscheinungen*.

Beide Veröffentlichungen erhalten Sie regelmäßig, wenn Sie in den entsprechenden Verteiler aufgenommen sind. Wenden Sie sich bitte hierfür an Ihre zuständige Geschäftsstelle. Dort können Sie auch die Handbücher bestellen.



# Stichwörter

24-Bit-Adressierungsmodus 48  
 31-Bit-Adressierungsmodus 48  
 4K-Blockstruktur 36  
 4K-fähiges ISAM 36  
 4K-ISAM-Datei 37  
 4K-Pubset 36  
 4stellige Jahresangabe 74

## A

Abschlußbericht 117  
 ACCESS-METHOD 33  
 ACS Subsystem 54  
 Adressierungsmodus 48  
 Adreßleiste aufbauen 176  
 Adreßleiste bei erweiterten Zeichensätzen 176  
 Aliasname 60  
 alphabetisches Zeichenformat (ALPHA) 124  
 alphanum-name (Datentyp) 61  
 AND-Verknüpfung 122  
 Angeben Version 21  
 Anschluß  
   für Ausgabe 186  
   für Eingabe 184  
   für Eröffnungsfehlerbehandlung 190  
   für Fehlerbehandlung 189  
   für Gruppenbearbeitung 187  
   für Kennsatzbearbeitung 182  
 Anweisungen  
   Gegenüberstellung SDF-ISP 57  
   Übersicht 56  
 Anweisungsbeschreibungen 81  
 Anwendungsbeispiele 195  
 Arbeiten mit PERCON 19  
 ARCHIVE-Sicherungsbandsets 10  
 Assembler-Hauptprogramm 229

- ASSIGN-INPUT-FILE-Anweisung 30, 82
- ASSIGN-INPUT-TAPE-Anweisung 97
- ASSIGN-OUTPUT-FILE-Anweisung 31, 101
- ASSIGN-OUTPUT-TAPE-Anweisung 111
- Aufbau
  - Adreßleiste 176
  - Gruppenwechselzeilen 17
  - Literal 73
  - Parameterbereich 177
- Aufbauen von Ausgabesätzen 12
- Aufbereiten
  - Gruppenwechselzeilen 17
  - Kennwörter 74
- Aufbereitungsmaske 79
- Ausgabe
  - gestalten 14
  - über DISKETTE 31
  - über SYSLST 31
  - über SYSOUT 31
- Ausgabeband 31
  - zuweisen 31
- Ausgabedatei 9
- Ausgabedatei, PAM 40
- Ausgabefeld, numerisch 79
- Ausgabesatz
  - aktualisieren (Randbedingungen) 150
  - Aufbau festlegen 150
  - aufbauen 12
  - bereits existierenden aktualisieren 150
  - maximale Länge 15
  - überschreiben 150
- Ausgeben
  - Bandbereich nach SYSLST 221
  - Bandbereich nach SYSOUT 221
  - Banddatei auf Drucker 200
  - Meldungen in S-Variablen 247
- Auswählen von Sätzen 236
- Auswertung von S-Variablen 250



**B****Band**

- duplizieren 224
- positionieren 18

**Bandbereich**

- nach SYSLSST ausgeben 221
- nach SYSOUT ausgeben 221

**Banddatei**

- auf Drucker ausgeben 200
- auf Platte umsetzen 200

Bedingung für die Ausgabe eines Satzes 122

Beenden des PERCON-Laufs 21

Beendigung fehlerhaft 26

Benutzeranschlüsse 179

- Kennzeichen 182

Benutzermodul 180

Beschreibung der Anweisungen 81

Bilden von Gruppen 215

Bindemodul PCROOT 171

**Bindemodulbibliothek**

- \$.SYSLNK.PERCON.026 zuweisen 171
- OMLPERCON zuweisen 171

BLOCK-CONTROL, Dateimerkmal 40

BLOCK-CONTROL-INFO 33

BLOCK-CONTROL-INFO, Operand Blockstruktur 38

BLOCK-COUNTER 74

Blockgröße 34

Blockkontrollfeld 38

Blockstruktur 33, 38

BUFFER-LENGTH 34

BUFFER-LENGTH-Operand 98

BYTE-COUNTER 74

**C**

cat (Zusatz zu Datentypen) 71

cat-id (Datentyp) 61

CCS (coded character set) 49

CCSN (coded character set name) 49

CFID (Coded File Identification) 38

CHANGE-INPUT-TAPE-Anweisung 18

CHANGE-INPUT-TAPEPOSITION-Anweisung 113

COBOL-Hauptprogramm 225

Code 49

CODE-Operand 98, 111

CODE-TRANSLATION-Operand 165  
Code-Umsetzen einer Banddatei 208  
COLUMN-SIZE-Operand 146  
COMMAND-Operand 116  
command-rest (Datentyp) 61  
compl (Zusatz zu Datentypen) 66  
composed-name (Datentyp) 61  
CONDITION-Operand 120  
    Syntax der Bedingung 123  
constr (Zusatz zu Datentypen) 70  
corr (Zusatz zu Datentypen) 71  
CPU-LIMIT-Operand 19  
CREATE-FILE-Kommando 30  
c-string (Datentyp) 61

**D**

Darstellen von Daten 15  
date (Datentyp) 61  
DATE (Kennwort) 74  
Datei, Eröffnungsart 36  
Dateikettungsname PCIN 96, 97  
Dateikettungsname PCOUT 109, 111  
Dateimerkmal  
    ACCESS-METHOD 33  
    BLOCK-CONTROL-INFO 33  
    BUFFER-LENGTH 34  
    KEY-LENGTH 36  
    KEY-POSITION 35  
    OPEN-MODE 36  
    RECORD-FORMAT 34  
    RECORD-SIZE 34  
Dateiname fest vorgegeben 191  
Dateiorientierte Ausgabe 51  
Dateiorientierte Eingabe 51  
Dateizugriffsmethode 33  
Datenblöcke übertragen 9  
Datenformat 15  
    prüfen 11  
Datenträgerorientierte Ausgabe 52  
Datenträgerorientierte Eingabe 51  
Datentypen SDF 58, 61  
    Zusätze 58  
Definieren  
    Leerzeilen 15

Zeichenanzahl 15  
device (Datentyp) 61  
DEVICE-TYPE-Operand 98, 112  
Dezimale Aufbereitung von Feldern 158  
DIRECTION-Operand 113  
Diskettendatei  
    ausgeben 211  
    eingeben 211  
Duplizieren Band 10, 224

## E

EBCDIC-Steuerzeichen 188  
Einfügen von Text 15  
Eingabe  
    über DISKETTE 30  
    über SYSDTA 30  
Eingabeband 31  
    zuweisen 31  
Eingabedatei 9  
Eingabedatei, PAM 38  
Eingabefehler 72  
Eingabefehler, formal 72  
EMPTY-BLOCK, Operand -siehe Leerblöcke 38  
END-Anweisung 24  
Ende fehlerhaft 26  
END-POSITION-Operand 97, 111, 169  
Erweitertes PERCON-Format 74  
EXECUTE-SYSTEM-COMMAND-Anweisung 116

## F

FDDRL-Sicherungsband kopieren 234  
FDDRL-Sicherungsbandsets 10  
Fehler, formal 72  
Fehler, logisch 26  
fehlerhafte Beendigung 26  
Feld mit Literal vergleichen 11, 124  
feste  
    Dateinamen 191  
    Kennungen 191  
Festlegen  
    Spaltenbereich 14  
    Zeilenbereich 14  
FILE-Operand 103  
FILLER=\*OUTPUT 156

FILLER-Operand 155  
fixed (Datentyp) 61  
Format  
    BINARY 76  
    CHARACTER 75  
    DECIMAL 77  
    HEXADECIMAL 76  
    NO-TRANSLATION 78  
    PACKED-DECIMAL 78  
    SIGNED-DECIMAL 77  
    TRANSLATION 78  
    ZONED-DECIMAL 76  
    ZONED-DECIMAL-LEFT 78  
Formataufbereitete Dateien 33  
Formate 75  
formatierte Zahlen mit Aufbereitungsmaske 79  
formatierte Zahlen und Zähler, Beispiel 232  
full-filename (Datentyp) 62  
Funktionen von PERCON 1

## G

garantierte Meldungen 247  
Gegenüberstellung SDF-ISP-Anweisungen 57  
gen (Zusatz zu Datentypen) 71  
Gestalten  
    der Ausgabe 14  
    der Seite 14  
    der Zeilen 14  
GLUE-Programm 48  
GROUP-CONTROL-Operand 133  
GROUP-COUNTER-Operand 141  
GROUP-HEADER-Operand 133  
GROUP-LEVEL-Operand 133  
GROUP-TRAILER-Operand 140  
Grundfunktionen 7  
Gruppen bilden 215  
Gruppenbegriff 17  
Gruppendefinitionen 18  
Gruppennachlauf 17  
Gruppenstufen 18  
Gruppenvorlauf 17  
Gruppenwechsel 17  
Gruppenwechselzeilen aufbereiten 17  
Gruppieren von Sätzen 17

**H**

HEADER-LINE-Operand 144

**I**

IMPORT-FILE-Kommando 29

Index

global 69

Konstruktionszeichenfolge 69

platzhalter-spezifisch 69

-Schreibweise 70

INPUT-EXIT-Operand 96

INPUT-RANGE-Operand 168

Installation 191

Produktdateien für einzelne BS2000-Versionen 193

integer (Datentyp) 63

ISAM-Datei

sortieren 213

Wiederherstellung 32

ISAM-Schlüssel

Länge 36

Position 35

**J**

Jahresangabe, 4stellig 74

**K**

Kanalvorschub 189

Kennwort 73

Aufbereitung 74

mit Literal vergleichen 11, 125

Kennwort GROUP-COUNTER 75

KEY-LENGTH 36

KEY-POSITION 35

Klammerschachtelung 125

Beispiele 125

Koexistenz von Produktversionen 191

Kommando-RC (Returncode)

festlegen 22

Übersicht 22

Kommentare 72

Konstruktionsangabe 70

Konstruktionszeichenfolge 69

Konvertierungsregeln 126

Konvertierungsschritt 25

Kopieren

FDDRL-Sicherungsband 234

SLED-Datei 235

Kurzname 60

## L

Laufzeitsystem SPL 171

Leerblöcke bearbeiten 38

Leerzeilen definieren 15

LINE-SIZE-Operand 15, 146

LINK-NAME-Operand 96, 97, 109, 111

Literale 73

logischer Fehler 26

logischer Operator AND 122

logischer Operator OR 122

low (Zusatz zu Datentypen) 66

Lückentest 38

## M

Magnetbandkassetten 7

man (Zusatz zu Datentypen) 71

Maske, siehe Aufbereitungsmaske 79

Maskenaufbau 79

Maskengröße 79

Maskenverarbeitung 79

Meldung PER0012 31

Meldung PER0084 10

Meldungen 251

garantierte 247

in S-Variablen ausgeben 247

Meldungsstrom

stoppen 249

zuweisen 249

Metasyntax SDF 58, 59

MF/MV-Set 10

Mindest-Satzlänge festlegen 151, 165

MODIFY-PERCON-OPTIONS-Anweisung 117

MONJV-Operand 20

Multifile/Multivolume-Set (MF/MV-Set) 10

**N**

name (Datentyp) 63  
Nebenfunktionen 7  
NK-ISAM-Dateien mit Sekundärschlüssel 46  
numerisches Zeichenformat (NUMERIC) 124

**O**

odd (Zusatz zu Datentypen) 71  
OMLPERCON zuweisen 171  
OPENC-Ausgang 32  
OPEN-MODE 36  
Operand BLOCK-CONTROL-INFO 38  
Operand EMPTY-BLOCK 38  
OR-Verknüpfung 122  
OUTPUT-AREA-Operand 145  
OUTPUT-EXIT-Operand 109  
OUTPUT-FIELDS-Operand 156  
OUTPUT-FORMAT-Operand 15, 145  
OUTPUT-LINK-NAME-Operand 120, 132, 144, 155, 168

**P**

PAGE-COUNTER 74  
Pamkey-Eliminierung 38  
pamkeylose ISAM-Datei 38  
pamkeylose PAM-Datei 38  
pamkeylose SAM-Datei 38  
PAM-Schlüssel 33  
Parameterbereich  
    aufbauen 177  
    der Anschlüsse 181  
partial-filename (Datentyp) 64  
PCIN (Dateikettungsname) 96, 97  
PCNSR 171  
PCNSR7 171  
PCNSR9 171  
PCNSRSTK 171  
PCOUT (Dateikettungsname) 109, 111  
PERCON  
    als Unterprogramm 171, 225  
    in XS-Umgebung 48  
    und ACS (Hinweise) 54  
PERCON-Bibliothek, aktuelle zuweisen 172  
PERCON-Funktionen 1  
PERCON-Lauf

- beenden 21
- starten 19
- unterbrechen 24
- PERCONU 171
- PERCON-Version aufrufen 173
- Positionieren von Bändern 18
- posix-filename (Datentyp) 64
- posix-pathname (Datentyp) 64
- POSIX-Platzhalter 67
- product-version (Datentyp) 65
- Produktdateien, Übersicht 193
- Produktversionen, Koexistenz 191
- PROGRAM-MODE-Operand 20
- Prüfen
  - Datenformat 11
  - Sortierung (auf-, absteigend) 11, 124
  - Zeichen 11, 124
  - Zeichenformat 124

## R

- Readme-Datei 3
- RECORD-COUNTER 14, 33, 74
- RECORD-FORMAT 34
- RECORD-LENGTH 74
- RECORD-SIZE 34
- Registerkonventionen
  - für Benutzeranschlüsse 180
  - für Unterprogramm 175
- Reihenfolge der Anweisungen 27
- RENAME-VOLUME-Operand 112
- RESET-INPUT-Anweisung 25, 119
- Restdateien 120
  - bei START-TAPE-PROCESSING 121
  - vereinbaren 120
- Returncode 22
- Returncode des Parameterbereichs 186, 190



**S**

SAM- in ISAM-Datei umsetzen 197  
Satzauswahl 123  
Satzauswahlbedingung 121  
Sätze gruppieren 17  
Satzfelder summieren 17  
Satzformat 34  
Satzlänge 34  
Satzzähler 14, 33  
Seiten gestalten 14  
Seitengestaltung  
    für Bandinhalte 16  
    für Dateien 16  
Sekundärschlüssel bei NK-ISAM-Datei 46  
SELECT-INPUT-RECORDS-Anweisung 120  
SEND-MESSAGE-Kommando 24  
sep (Zusatz zu Datentypen) 71  
SET-FILE-LINK-Kommando 30  
SET-GROUP-ATTRIBUTES-Anweisung 127  
SET-PAGE-LAYOUT-Anweisung 142  
SET-RECORD-MAPPING-Anweisung 150  
SLED-Datei kopieren 235  
Softwarekomponente VTSU 50  
Sortieren einer ISAM-Datei 213  
Sortierung prüfen (auf-, absteigend) 11, 124  
SPACING-Operand 15, 145  
Spaltenbereich festlegen 14  
Spezielle Anwendungen 234  
Stack PCNSRSTK 171  
Standard-Kopfzeile 144  
Standard-Seitengestaltung 16  
START-CONVERSION-Anweisung 167  
START-TAPE-PROCESSING-Anweisung 168  
Steuerzeichen zur Druckersteuerung 142  
Steuerzeichenbehandlung 147  
structured-name (Datentyp) 65  
Subsystem  
    ACS 54  
    XHCS-SYS 50  
Summieren von Satzfeldern 17  
SUPPRESS-EQUAL-LINES-Operand 15, 146  
S-Variable  
    auswerten 250  
    vereinbaren 248

SYSLST-LOGGING-Operand 117  
SYSOUT-LOGGING-Operand 117  
Systeminformation 33

## T

TERMINATION-Operand 169  
text (Datentyp) 65  
Text einfügen 15  
TFT-Eintrag  
    erhalten 85, 88, 105  
    erzeugen 84, 103  
    löschen 85, 88, 106  
time (Datentyp) 65  
TIME (Kennwort) 74

## U

Übersicht  
    Kommando-Returncodes 22  
    PERCON-Anweisungen 56  
    Produkdateien 193  
Übertragen von Datenblöcken 9  
Übertragungsvorgang 25  
Umsetzen  
    Banddatei auf Platte 200  
    SAM- in ISAM-Datei 197  
under (Zusatz zu Datentypen) 66  
Unterbrechen des PERCON-Laufs 24  
Unterdrücken von Zeilen 15  
Unterprogramm PERCON 225  
Unterprogrammschnittstelle 171  
user (Zusatz zu Datentypen) 71  
USER-LINE-Operand 15, 147

## V

Verändern von Sätzen 236  
    bei identischer Ein- und Ausgabedatei 236  
    bei verschiedenen Ein- und Ausgabedateien 238  
Verarbeitungsfunktionen 52  
Vergleich Feld mit Literal 11, 123, 124  
Vergleich Kennwort mit Literal 11, 123, 125  
Vergleichsoperatoren 123  
vers (Zusatz zu Datentypen) 71  
Versions-Angabe 21  
Versionsauswahl, Prioritäten 192

Versionsbereich in Adreßleiste 173  
Versionsbezogener UP-Aufruf 172  
VOLUME-Operand 97, 111  
Voraussetzungen für erweiterte Codes 50  
vsn (Datentyp) 65  
VTSU (virtual terminal support) 50

## W

Wertigkeit der Zuweisungen 29  
Wiederherstellung von ISAM-Dateien 32  
wild(n) (Zusatz zu Datentypen) 67  
with (Zusatz zu Datentypen) 66  
without (Zusatz zu Datentypen) 71

## X

XHCS (extended host code support) 49  
x-string (Datentyp) 66  
x-text (Datentyp) 66

## Z

Zeichen prüfen 11, 124  
Zeichenanzahl definieren 15  
Zeichenformat prüfen 124  
Zeichensatz  
    codierter 49  
    erweiterter 49  
Zeilen  
    gestalten 14  
    unterdrücken 15  
Zeilenbereich festlegen 14  
Zugriffsmethode  
    BTAM 33  
    ISAM 33  
    PAM 33  
    SAM 33  
Zusätze zu Datentypen 58, 66  
Zuweisen  
    \$.SYSLNK.PERCON.026 171  
    OMLPERCON 171



# Inhalt

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Kurzbeschreibung des Produkts	1
1.2	Zielsetzung und Zielgruppen des Handbuchs	2
1.3	Konzept des Handbuchs	2
1.4	Änderungen gegenüber der vorigen Ausgabe	4
	Inkompatibilitäten	5
1.5	Verwendete Metasprache	6
<b>2</b>	<b>Funktionen</b>	<b>7</b>
2.1	Grundfunktionen	9
2.1.1	Kopieren von Dateien	9
2.1.2	Editieren von Bandinhalten	9
2.1.3	Duplizieren von Bändern	10
2.2	Nebenfunktionen	11
2.2.1	Auswählen von Sätzen oder Blöcken	11
2.2.2	Aufbauen von Sätzen	12
2.2.3	Aufbereiten von Sätzen oder Blöcken	13
2.2.4	Gruppieren von Sätzen	17
2.2.5	Positionieren von Bändern	18
<b>3</b>	<b>Arbeiten mit PERCON</b>	<b>19</b>
3.1	Starten des PERCON-Laufs	19
3.2	Beenden des PERCON-Laufs	21
3.3	Kommando-Returncode	22
3.4	Unterbrechen des PERCON-Laufs	24
3.5	Konvertierungsschritt	25
3.6	Reihenfolge der Anweisungen	27
3.7	Zuweisen der Ein- und Ausgabe	29
3.8	Maßnahmen im OPENC-Ausgang	32
3.9	Bearbeiten katalogisierter Dateien	33
3.9.1	Dateimerkmale	33
3.9.1.1	4K-Pubsets	36
3.9.1.2	4K-fähiges ISAM	36
3.9.2	Pamkey-Eliminierung	38
3.9.2.1	PAM-Eingabedateien	38
3.9.2.2	PAM-Ausgabedateien	40

3.9.2.3	Beispiele zum Kopieren von PAM-Dateien . . . . .	41
3.9.3	Übernehmen von Sekundärschlüsseln . . . . .	46
3.10	PERCON in XS-Umgebung (31-Bit-Adressierung) . . . . .	48
3.11	Verwendung erweiterter Zeichensätze in PERCON . . . . .	49
3.11.1	Hard- und Softwarevoraussetzungen . . . . .	50
3.11.2	PERCON-spezifische Anwendung erweiterter Zeichensätze . . . . .	50
3.11.3	Verarbeitungsfunktionen . . . . .	52
3.12	PERCON und ACS . . . . .	54
<b>4</b>	<b>PERCON-Anweisungen . . . . .</b>	<b>55</b>
4.1	Übersicht der PERCON-Anweisungen . . . . .	55
4.2	Gegenüberstellung von SDF- und ISP-Anweisungen . . . . .	57
4.3	Syntaxbeschreibung . . . . .	58
4.4	Literale . . . . .	73
4.5	Kennwörter . . . . .	73
4.6	Formate . . . . .	75
4.7	Beschreibung der Anweisungen . . . . .	81
	ASSIGN-INPUT-FILE	
	Zuweisen einer Eingabedatei . . . . .	82
	ASSIGN-INPUT-TAPE	
	Zuweisen eines Eingabebandes . . . . .	97
	ASSIGN-OUTPUT-FILE	
	Zuweisen einer Ausgabedatei . . . . .	101
	ASSIGN-OUTPUT-TAPE	
	Zuweisen eines Ausgabebandes . . . . .	111
	CHANGE-INPUT-TAPEPOSITION	
	Positionieren des Eingabebandes . . . . .	113
	END	
	Beenden des PERCON-Laufs, Starten eines Konvertierungsschrittes . . . . .	115
	EXECUTE-SYSTEM-COMMAND	
	Absetzen eines Systemkommandos . . . . .	116
	MODIFY-PERCON-OPTIONS	
	Steuern der Meldungsausgabe . . . . .	117
	RESET-INPUT	
	Rücksetzen von Anweisungen . . . . .	119
	SELECT-INPUT-RECORDS	
	Festlegen von Auswahlbedingungen . . . . .	120
	SET-GROUP-ATTRIBUTES	
	Festlegen von Gruppenwechselbedingungen . . . . .	127
	SET-PAGE-LAYOUT	
	Beschreiben des Ausgabeformates . . . . .	142
	SET-RECORD-MAPPING	
	Festlegen des Ausgabesatzaufbaus . . . . .	150

START-CONVERSION	
Starten eines Konvertierungsschrittes	167
START-TAPE-PROCESSING	
Steuern der Bandausgabe	168
<b>5 Aufruf von PERCON als Unterprogramm</b>	<b>171</b>
5.1 Versionsbezogener UP-Aufruf von PERCON	172
5.2 Registerkonventionen	175
5.3 Aufbau der Adreßleiste	176
<b>6 Benutzeranschlüsse</b>	<b>179</b>
6.1 Registerkonventionen	180
6.2 Gemeinsamer Parameterbereich der Anschlüsse	181
6.3 Beschreibung der Anschlüsse	182
6.3.1 Anschluß für die Kennsatzbearbeitung	182
6.3.2 Anschluß für die Eingabe	184
6.3.3 Anschluß für die Ausgabe	186
6.3.4 Anschluß für die Gruppenbearbeitung	187
6.3.5 Anschluß für die Lese-/Längenfehlerbehandlung	189
6.3.6 Anschluß für die Eröffnungsfehlerbehandlung	190
<b>7 Installation</b>	<b>191</b>
<b>8 Anwendungsbeispiele</b>	<b>195</b>
8.1 Umsetzen zweier SAM-Dateien in eine ISAM-Datei und Ausgeben auf Drucker	197
8.2 Umsetzen und Ausgeben einer nichtkatalogisierten Banddatei	200
8.3 Ausgeben von Sätzen in eine Restdatei	203
8.4 Aktualisieren Felder in einem existierenden Ausgabesatz	205
8.5 Ausgeben einer Datei auf Band mit Umsetzen des Codes	208
8.6 Ein- und Ausgeben einer Datei von bzw. auf Diskette	211
8.6.1 Ausgabe auf Diskette	211
8.6.2 Eingabe von Diskette	212
8.7 Sortieren einer ISAM-Datei	213
8.8 Bilden von Gruppen	215
8.9 Ausgeben eines definierten Bereichs eines Bandes	221
8.10 Duplizieren eines Bandes	224
8.11 Beispiel eines COBOL-Hauptprogramms	225
8.12 Beispiel eines Assembler-Hauptprogramms	229
8.13 Beispiel zu formatierten Zahlen und Zähler je Gruppe	232
8.14 Spezielle Anwendungen	234
8.14.1 Kopieren eines FDDRL-Sicherungsbandes	234
8.14.2 Kopieren einer SLED-Datei auf Platte	235
8.14.3 Auswählen und Verändern von Sätzen	236
8.15 Arbeiten mit MF/MV-Sets	242

9      **Meldungen** ..... **247**  
Ausgabe von Meldungen in S-Variablen ..... 247  
Vorgehensweise ..... 248

**Literatur** ..... **281**

**Stichwörter** ..... **287**



---

# PERCON V2.6A (BS2000/OSD)

## Benutzerhandbuch

### *Zielgruppe*

BS2000/OSD-Anwender

### *Inhalt*

Das Handbuch beschreibt die Vorgehensweise und die Anweisungen, wie Daten auf beliebige Datenträger übertragen und umgesetzt werden können. Häufige Anwendungsfälle werden durch Beispiele erklärt.

**Ausgabe: Dezember 1995**

**Datei: PERCON.PDF**

BS2000 ist ein eingetragenes Warenzeichen der Siemens Nixdorf Informationssysteme AG

Copyright © Siemens Nixdorf Informationssysteme AG, 1995.

Alle Rechte vorbehalten.

Liefermöglichkeiten und technische Änderungen vorbehalten.

Alle verwendeten Hard- und Softwarenamen sind Handelsnamen und/oder Warenzeichen der jeweiligen Hersteller.