

Deutsch



Fujitsu Software

BS2000 OS DX System Exits

Benutzerhandbuch

Stand der Beschreibung:
BS2000 V21.0A

Ausgabe Juni 2022

Kritik... Anregungen... Korrekturen...

Die Redaktion ist interessiert an Ihren Kommentaren zu diesem Handbuch. Ihre Rückmeldungen helfen uns, die Dokumentation zu optimieren und auf Ihre Wünsche und Bedürfnisse abzustimmen.

Sie können uns Ihre Kommentare per E-Mail an bs2000.info@fujitsu.com senden.

Zertifizierte Dokumentation nach DIN EN ISO 9001:2015

Um eine gleichbleibend hohe Qualität und Anwenderfreundlichkeit zu gewährleisten, wurde diese Dokumentation nach den Vorgaben eines Qualitätsmanagementsystems erstellt, welches die Forderungen der DIN EN ISO 9001:2015 erfüllt.

Copyright und Handelsmarken

Copyright © 2025 Fujitsu

Alle Rechte vorbehalten.

Liefermöglichkeiten und technische Änderungen vorbehalten.

Alle verwendeten Hard- und Softwarenamen sind Handelsnamen und/oder Warenzeichen der jeweiligen Hersteller.

Inhaltsverzeichnis

System Exits	5
1 Einleitung	6
1.1 Zielsetzung und Zielgruppen des Handbuchs	7
1.2 Konzept des Handbuchs	8
1.3 Änderungen gegenüber der vorherigen Ausgabe	9
1.4 Verwendete Metasprache	10
2 Einführung in die System Exits	11
2.1 Funktion des Exit-Mechanismus	15
2.2 TU-Makroschnittstelle EXCALL	17
2.3 Exit-Routinen	18
2.3.1 Programmierregeln für Exit-Routinen	19
2.3.2 Speicherverwaltung	20
2.3.3 Diagnose und Test	21
2.3.4 Überwachung von Dateizugriffen	22
2.4 Programmierbeispiel	23
2.4.1 Systemumgebung der System Exits	24
2.4.2 Konzept für den modularen Aufbau von Exit-Routinen	25
2.4.3 Beispiel für eine Basisroutine mit Subroutinen	26
2.5 Verwaltung von Exit-Routinen mit DSSM	28
3 Beschreibung der Exits	31
3.1 Termination-Exits (001/002)	32
3.2 SHUTDOWN-Exit (003)	38
3.3 OPEN-Element-Exits (005/006)	39
3.4 CLOSE-Element-Exits (007/008)	41
3.5 JOIN-Exit (010)	43
3.6 ACCOUNTING-Exit (015)	48
3.7 PSWORD-Exit (020)	52
3.8 SECURE-Exits (025/026)	55
3.9 LOGON-Exit (030)	59
3.10 Jobparameter-Exit (032)	61
3.11 JV-Exit (033)	63
3.12 BCAM-Exit (036)	65
3.13 OPEN-Exits (039/040/041)	67
3.14 CLOSE-Exits (042/043)	71
3.15 Spulenwechsel-Exits (044/045)	74
3.16 FILE-Exits (050/051)	77
3.17 ATTACH-Library-Exit (054)	80

3.18 DELETE-Element-Exits (055/056)	82
3.19 RENAME-Element-Exits (057/058)	84
3.20 ERASE-Exits (060/061)	86
3.21 CATALOG-Exits (070/071)	89
3.22 DCAM-Exit (075)	93
3.23 SYSCMD-Exit (080)	103
3.24 SDF-Exits (080/081/082/083)	107
3.24.1 SYSCMD-Exit 080	108
3.24.2 SYSCMD-Exit 081	109
3.24.3 SYSSTMT-Exit 082	110
3.24.4 SYSSTMT-Exit 083	111
3.25 SPOOL-Exits (090/091/092/094/095/096/097)	115
3.25.1 Allgemeines zu den SPOOL-Exits	116
3.25.2 SPOOL-Exit 090	118
3.25.3 SPOOL-Exits (091/092) für Deckblatt und Schlussblatt	124
3.25.4 SPOOL-Exit 094 (Ressourcen-Routine)	136
3.25.5 SPOOL-Exit 095 (Ausgabe-Routine)	138
3.25.6 SPOOL-Exit 096 (Server-Auswahl)	142
3.25.7 SPOOL-Exit 097 (Steuern File Transfer)	144
3.26 SAT-Exit 110	146
3.27 Makro \$SATANY	149
4 Eventing in Exit-Routinen	153
4.1 \$GTBOID - GET Bourse Identifier	154
4.2 \$NBSSIG - Send a Signal to Event Item Bourse	157
5 System Exits für Programmbibliotheken	161
6 Anhang	164
6.1 Systemtabellen	165
6.1.1 CATAL	166
6.1.2 FILE	172
6.1.3 ERASE	179
6.1.4 IDFCB	188
6.1.5 DCACATE	194
6.1.6 \$JTBP	212
6.1.7 SJMCHKZ	219
6.1.8 SSVSCB	224
6.2 SVC-Listen	247
6.2.1 Liste der in System Exits gültigen SVCs	248
6.2.2 Liste der SVCs, die in System Exits nicht verwendet werden dürfen	258
6.2.3 Hinweise	270
7 Literatur	271

System Exits

1 Einleitung

Im Handbuch werden die Systemausgänge von BS2000 (System Exits) beschrieben. Sie bieten die Möglichkeit, die Befehlsfolge einer Systemkomponente oder eines Benutzerprogramms an einer bestimmten Stelle zu unterbrechen und eine von der Systembetreuung erstellte Befehlsfolge ablaufen zu lassen. So wird eine Anpassung an spezielle Erfordernisse des RZ-Betriebs erreicht.

1.1 Zielsetzung und Zielgruppen des Handbuchs

Dieses Handbuch wendet sich an die Systembetreuung von BS2000 und an den Programmierer mit sehr guten Assembler- und Betriebssystem-Kenntnissen.

1.2 Konzept des Handbuchs

Die Beschreibung des Exit-Basismechanismus finden Sie in „[Einführung in die System Exits](#)“.

Die Beschreibung aller System Exits, die zurzeit angeboten werden finden Sie in „[Beschreibung der Exits](#)“.

Die darauf folgenden Kapitel beschreiben die Anwendung der Ereignissteuerung in Exit-Routinen und die System Exits zur Überwachung der Zugriffe auf Programmbibliotheken.

Am Ende des Handbuchs finden Sie verschiedene Verzeichnisse, die Ihnen das Arbeiten mit diesem Handbuch erleichtern.

Informationen unter BS2000

Das Kommando `/SHOW-INSTALLATION-PATH INSTALLATION-UNIT=<product>` zeigt, unter welcher Benutzerkennung die Dateien des Produkts abgelegt sind.

Ergänzende Produkt-Informationen

Aktuelle Informationen, Versions-, Hardware-Abhängigkeiten und Hinweise für Installation und Einsatz einer Produktversion enthält die zugehörige Freigabemitteilung. Solche Freigabemitteilungen finden Sie online unter <http://bs2manuals.ts.fujitsu.com>.

1.3 Änderungen gegenüber der vorherigen Ausgabe

1.4 Verwendete Metasprache

In diesem Handbuch werden folgende Darstellungsmittel verwendet:

Ausgabe DSECTS und Beispiele werden in Schreibmaschinenschrift dargestellt

i für Hinweise auf besonders wichtige Informationen

Literaturhinweise werden im Text in Kurztiteln und eckigen Klammern angegeben. Der vollständige Titel jeder Druckschrift, auf die verwiesen wird, ist im Literaturverzeichnis aufgeführt.

2 Einführung in die System Exits

Ein System Exit ist eine Stelle im Betriebssystem oder in einem TU-Programm (TU = Task unprivileged), an der die laufende Verarbeitung durch den Aufruf einer so genannten Exit-Routine unterbrochen werden kann.

Eine Exit-Routine ist eine Komponente, die in der Regel von der Systembetreuung von BS2000 erstellt und dem Betriebssystem hinzugefügt wird. Mit dieser Exit-Routine kann eine normalerweise nicht änderbare Betriebssystem-Funktion an spezielle Erfordernisse des RZ-Betriebs angepasst werden. Exit-Routinen können Funktionen zum normalen Systemverhalten hinzufügen oder existierende Funktionen verändern bzw. ersetzen. Sie erscheinen als integrierte Bestandteile des Betriebssystems.

Exit-Routinen können während des Systemlaufs geladen und aktiviert oder deaktiviert, also dynamisch verwaltet werden.

Der Anschluss von Exit-Routinen ist bei den in der nachfolgender Tabelle genannten Systemfunktionen vorgesehen. Für einige Funktionen sind zwei Exit-Punkte vorhanden:

- Ein Request Exit, der es ermöglicht, die Eingabe des Benutzers vor der Bearbeitung durch BS2000 zu modifizieren oder abzuweisen.
- Ein Return Exit, an dem die Bearbeitungsergebnisse des Systemmoduls vor dem Rücksprung in die anfordernde Task überprüft oder verändert werden können.

Eine Sonderstellung nimmt der von jedem TU-Programm aufrufbare TU-Exit ein. Wann und wie dieser Exit benutzt wird, liegt allein in der Verantwortung der Systembetreuung in Absprache mit seinen RZ-Kunden.

Übersicht der Exit-Punkte

Die Exits sind intern durch ihre Exit-Nummer gekennzeichnet. Die folgende Übersicht zeigt die Exit-Punkte und ihren Aufrufzeitpunkt:

Nr.	Exit	Typ / Zeitpunkt des Aufrufs
000	TU-Exit	
001 002	Termination Exits für Programm- bzw. Taskbeendigung	Request Return
003	SHUTDOWN-Exit	während Beendigung des Systemlaufs
005 006	OPEN-Element-Exit für Programmbibliotheken	Request Return
007 008	CLOSE-Element -Exit für Programmbibliotheken	Request Return
010	JOIN-, ADD-USER- und MODIFY-USER-ATTRIBUTES-Kommando	vor Schreiben des Benutzersatzes in den Benutzerkatalog
015	Accounting Exit	vor Schreiben eines Abrechnungssatzes
020	PSWORD-Kommando MODIFY-USER-PROTECTION-Kommando	vor dem Ändern eines Kennwortes im Benutzersatz des Benutzerkatalogs

025	SECURE-Kommando	Return
026	SECURE-RESOURCE-ALLOCATION-Kommando	Request
030	LOGON-Exit	bei den Kommandos ENTER-JOB, ENTER-PROCEDURE und SET-LOGON-PARAMETERS
032	Jobparameter-Exit	bei den Kommandos ENTER-JOB, ENTER-PROCEDURE, MODIFY-JOB und SET-LOGON-PARAMETERS zur Behandlung von Jobparametern
033	JV-Exit	bei lesendem Zugriff auf eine Sonderjobvariable
036	BCAM-Exit	Verbindungsaufbau zu Systemanwendungen
039	OPEN-Exit	Security
040	OPEN-Exit	Return
041		Request
042	CLOSE-Exit	Request
043		Return
044	Spulenwechsel bei Magnetband	Request
045		Return
050	CREATE-FILE-/ MODIFY-FILE-ATTRIBUTES-/ CREATE-FILE-GENERATION-/ MODIFY-FILE-GENERATION-SUPPORT-/ ADD-FILE-LINK-Kommando FILE-Makro	Return Request
051		
054	ATTACH-Library-Exit für Programmbibliotheken	nach Eröffnen Programmbibliothek
055	DELETE-Element-Exit für Programmbibliotheken	Request
056		Return
057	RENAME-Element-Exit für Programmbibliotheken	Request
058		Return
060	DELETE-FILE-/ DELETE-FILE-GROUP-/ DELETE-FILE-GENERATION-/ DELETE-SYSTEM-FILE-/ EXPORT-FILE-Kommando ERASE-Makro	Return Request
061		

070	CREATE-FILE-/ 071 CREATE-FILE-GENERATION-/ CREATE-FILE-GROUP-/ MODIFY-FILE-ATTRIBUTES-/ MODIFY-FILE-GENERATION-SUPPORT-/ MODIFY-FILE-GROUP-ATTRIBUTES- Kommando CATALOG-Makro	Return Request
075	DCAM	Request bzw. ereignisgesteuert
080 081 082 083	CMD (SDF-Exits) CMD STMT STMT	Request = vor Bearbeitung jedes Kommando Return = nach Analyse jedes Kommandos durch den Kommandoentschlüssler Request = vor Analyse jeder Anweisung Return = nach Analyse jeder Anweisung
090	SPOOL-Exit	vor Ausgabe jedes Satzes auf Drucker
091 092	SPOOL-Deckblatt SPOOL-Schlussblatt	vor Ausgabe des Deckblatts vor Ausgabe des Schlussblatts
094	SPOOL-Ressourcen-Routine	Konvertierung Client- / Server-Ressource
095	SPOOL-Exit	vor der Ausgabe des Kanalprogramms
096	SPOOL-Exit	Server-Auswahl
097	SPOOL-Exit	Steuern der Aufträge für den File-Transfer
110	SAT-Exit	vor Schreiben eines SAT-Protokollsatzes

Übersicht der verwendeten Makros und der zugehörigen Standardbibliothek

Mit den unten genannten Makros können die für die Exits benötigten DSECTs erzeugt werden. Die Makros werden in den genannten Bibliotheken ausgeliefert. Der Aufbewahrungsort kann von der Systemverwaltung geändert werden.

Makroname	Standardbibliothek
\$TERMRF, \$DOCSYS	SIPLIB.BS2CP.<ver>
EX005, EX007, EX054, EX055, EX057	SIPLIB.PLAM.<ver>
\$SRMSYE	SIPLIB.SRPMNUC.<ver>
NASXIT, NKSEPAR, EX030, EX032, EX050, EX060, EX070, EX080	SYSLIB.BS2CP.<ver>
EX033	SYSLIB.JV.<ver>
YDBBXD	SIPLIB.BCAM.<ver>
YDDEXPL	SYSLIB.DCAM

EX090, EX091, EX092, EX094, EX095, EX096, EX097	SYSLIB.SPOOL.<ver>
EX110	SIPLIB.SATCP.<ver>

<ver> ist die Version der aktuell installierten Komponente

2.1 Funktion des Exit-Mechanismus

Die Funktionsweise des Exit-Mechanismus ist im folgenden Bild schematisch dargestellt:

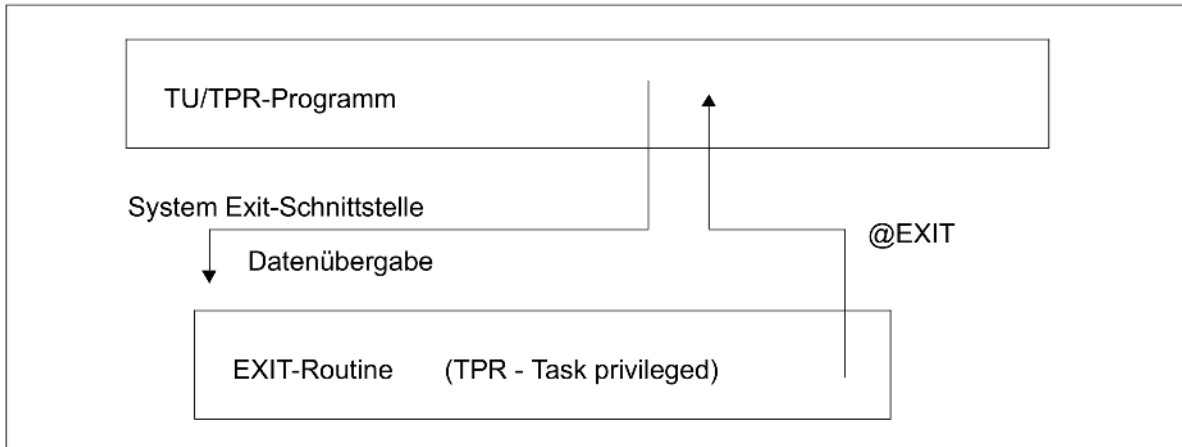


Bild 1: Exit-Routinen-Aufruf

Ein System-Modul, bei dem ein Exit-Punkt vorgesehen ist, verzweigt über eine TU- bzw. TPR-Makroschnittstelle in die dazugehörige Exit-Routine, falls diese existiert und aktiv ist. Dabei übergibt der Modul exitspezifische Parameter und Daten an die Exit-Routine. Diese kann, abhängig vom gewünschten Verwendungszweck, die Daten prüfen und evtl. modifizieren, eine Anforderung abweisen oder protokollieren. Anschließend erfolgt der Rücksprung in den rufenden System-Modul durch den Makro @EXIT in Programm-Manager-Umgebung (DSL-Environment) bzw. über BR R14 bei reinen Assembler-Routinen. Neue Exitroutinen sollten unbedingt in DSL-Environment entwickelt werden.

Bei der Verzweigung in eine Exit-Routine sind einige TPR-Register mit Informationen versorgt. Die Registerinhalte sind der Beschreibung des jeweiligen Exits zu entnehmen.

Die Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden; sie müssen sichergestellt und vor dem Rücksprung aus der Exit-Routine mit den ursprünglichen Inhalten geladen werden, d.h., sie müssen beim Rücksprung ihren ursprünglichen Inhalt haben.

Die Zugriffe auf Inhalte der EXVT und des TCB müssen über die entkoppelten Zugriffsfunktionen (siehe Entwickler-Handbuch) erfolgen.

Die Rückkehrinformation der Exitroutine muss entsprechend der jeweiligen Exit-Beschreibung entweder in Register 15 oder in der spezifizierten Parameterliste mit Standardheader hinterlegt werden.

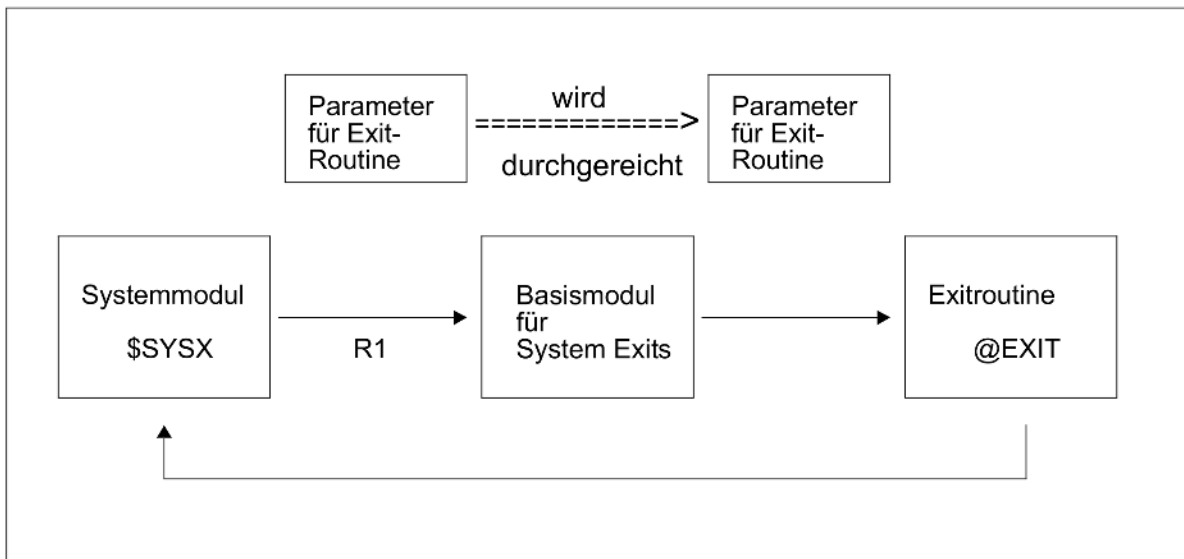


Bild 2: Exit-Mechanismus

2.2 TU-Makroschnittstelle EXCALL

EXCALL – EXit CALL für TU-Programme (O-Typ)

Der Makroaufruf EXCALL kann von TU-Programmen zum Aufruf der Exit-Routine 000 verwendet werden.

[name] EXCALL

Die TU-Register 0 und 15 werden zerstört.

Welche Parameter in Register 1 an die Exit-Routine 000 übergeben werden, liegt in der Verantwortung der Systembetreuung und ist abhängig von seinen Vereinbarungen mit dem Aufrufer. Die Information in Register 1 wird vom TU-PCB zum TPR-PCB weitergereicht.

Die übergebenen Parameter müssen von der Exit-Routine validiert werden (\$HVAL,\$VALID). Es muss sichergestellt sein, dass die angeführten Parameter zugreifbar sind.

Für die Exit-Routine 000 gelten ansonsten die gleichen Programmierregeln wie für alle anderen Exit-Routinen.

2.3 Exit-Routinen

- Programmierregeln für Exit-Routinen
- Speicherverwaltung
- Diagnose und Test
- Überwachung von Dateizugriffen

2.3.1 Programmierregeln für Exit-Routinen

Da Exit-Routinen wie die rufenden Systemkomponenten im privilegierten Zustand ablaufen, sollten sie mit besonderer Sorgfalt programmiert werden. Insbesondere können die Verwendung von TU-Makros oder andere Programmierfehler einen Task-Abbruch oder gar einen Systemausfall zur Folge haben. Fehler in Exit-Routinen werden wie Systemfehler behandelt. Die im folgenden Abschnitt aufgeführten Regeln und Einschränkungen sind unbedingt zu beachten:

- Exit-Routinen müssen in XS-Umgebung AMODE ANY und RMODE ANY enthalten und den XS-Programmierregeln genügen (siehe "[Beispiel für eine Basisroutine mit Subroutinen](#)").
- Exit-Routinen müssen reentrant und read only programmiert sein.
- Der CSECT-/ENTRY-Name einer Exit-Routine ist frei wählbar. Die verwendeten CSECT-/ENTRY-Namen müssen systemweit eindeutig sein.
- Probleme der Aufrufverschachtelung, Parallelbenutzung oder Rekursion müssen durch die Exit-Routine selbst abgefangen werden. Wir verweisen auf das Programmierbeispiel für Exit-Routinen.
- Existiert für eine Funktion sowohl ein TU-SVC als auch eine privilegierte Schnittstelle (z.B. REQM, \$REQM), muss die Exit-Routine den privilegierten Aufruf verwenden, Ausnahme: PRNT (siehe Hinweise zu SPOOLOUT-Exit).
- Felder in Systemtabellen (s. Anhang), die als „unused“ oder „reserved“ gekennzeichnet sind, dürfen von Exit-Routinen nicht verwendet werden. Sofern die Exit-Routine Task- bzw. Jobinformationen benötigt, empfehlen wir dringend, die Makros \$TINF bzw. \$RDUID, \$JINBAS, \$JINTSK, JOBINFO zu verwenden. Die entsprechenden Informationen können so problemlos aus den der Exit-Routine eigenen Parameterbereichen abgeholt bzw. abgefragt werden.
- Privilegierte Maschinenbefehle dürfen nicht verwendet werden.
- Die Exit-Routine darf für Systemtasks (interrupt driven tasks) keine Behandlung durchführen (entsprechendes Bit im TCB über Zugriffsmethode \$EZOTYP abfragen).
- Bei Dateibehandlung in Exit-Routinen muss eine eröffnete Datei unbedingt von dieser Routine auch wieder geschlossen werden, sonst führt es zu Komplikationen.
- Exit-Routinen laufen im privilegierten Zustand TPR (Task Privileged); sie können auf *eigenes Risiko* privilegierte Systemschnittstellen verwenden.
- Exit-Routinen werden zu einem Zeitpunkt aktiviert, zu dem im System kein „LOCK“- Zustand vorliegt.
- Exit-Routinen dürfen die Gleitpunktregister benutzen, wenn diese vor Verwendung sichergestellt und danach wieder zurückgeladen werden.
- Bei Verwendung von Makros, die die Einschaltungsmakros ##BAL und ##BALR benutzen, müssen diese mit der Assembler-Anweisung OPSYN nach BAS bzw. BASR umgelenkt werden.

2.3.2 Speicherverwaltung

Exit-Routinen belegen Klasse-4-Speicher des privilegierten Systemadressraums im BS2000. Sie werden als dynamisch zuladbarer Teil des Systems angesehen und daher nicht gegen Überschreiben geschützt.

Tasklokaler Speicherbereich, z.B. für Parameter, kann mithilfe der privilegierten Systemfunktionen \$GETMEM /\$RETMEM im Klasse-5-Speicher angefordert bzw. freigegeben werden.

Globaler Speicherbereich muss immer im Klasse-4-Speicher reserviert werden.

Über die Class-2-Optionen TASKVECT bzw. SYSVECT kann bereits bei der Systemgenerierung tasklokaler bzw. globaler Speicherplatz für die Exit-Routinen vorgesehen werden. Standardwert: 0 Seiten; maximale Größe: 15 Seiten

Taskvektor und Systemvektor werden beim ersten Aufruf einer Exit-Routine eingerichtet. Der Taskvektor wird bei LOGOFF oder Zerstörung der Task freigegeben. Der Systemvektor bleibt bis SHUTDOWN bestehen.

Die Adresse des Taskvektors erhält man durch den Aufruf von \$TANC mit den Operanden FUNCT=READ, OWNER=SYSEXIT. Die Adresse des Systemvektors erhält man durch den Aufruf \$SANC mit den Operanden FUNCT=READ, OWNER=SYSEXIT. Direktzugriffe auf die Felder ETCBSEXP (TCB) und EXVTSEXP (EXVT) sollten vermieden werden. Weitere Informationen dazu sind dem Entwickler-Handbuch zu entnehmen.

Die Länge eines Vektors in Bytes wird jeweils im ersten Halbwort des Vektors abgespeichert. Ist der Vektor größer als 7 Seiten, ist das Bit 2^{15} gesetzt; daher kann zum Laden der Länge in ein Register der Assembler-Befehl LH nicht verwendet werden!

System- und Taskvektor werden von allen Exits benützt. Die Koordination der Zugriffe liegt beim Exit-Programmierer /Systembetreuung.

Speicherengpässe, die durch Verwendung von Task- und Systemvektor bzw. durch \$REQM oder durch die im Klasse-4-Speicher liegenden Exit-Routinen verursacht werden, liegen in der Verantwortung des Benutzers.

2.3.3 Diagnose und Test

Bei Aufruf einer Exit-Routine wird im PCB (Process Control Block) im Byte ESTKIND das Bit 2^4 gesetzt:

```
ESTKEXIT EQU X'10'
```

Nach Ablauf der Exit-Routine wird das Bit wieder zurückgesetzt. Die Namen, die im PCB verwendet werden, können mit dem Makroaufruf DSTK generiert werden.

Exit-Routinen können mit den Produkten DAMP, HELGA und IDIAS getestet werden. DAMP ist im „Diagnosehandbuch“ [3] beschrieben. HELGA und IDIAS sind interne Diagnosetools für den Servicetechniker.

Gesetzte AUDIT-Indikatoren bleiben vom Aufruf einer Exit-Routine unberührt.

Bei der Erstellung von Fehlerunterlagen (Dumps) werden Exit-Routinen wie Systemteile behandelt. Sondermaßnahmen werden nicht durchgeführt.

2.3.4 Überwachung von Dateizugriffen

Mithilfe der System Exits des Datenverwaltungssystems für die Funktionen OPEN, FILE, ERASE und CATALOG kann die Systembetreuung den Zugriff zu Dateien überwachen und gegebenenfalls einschränken. Mit dem AUDIT-Operanden in den Kommandos CREATE-FILE-[GROUP]/MODIFY-FILE[-GROUP]-ATTRIBUTES erhalten Dateien ein Kennzeichen, das von Exit-Routinen ausgewertet wird. So kann z.B. festgehalten werden, wie oft welche Datei eröffnet wurde oder wie viele Dateien während einer Task gelöscht wurden. Die Exit-Routine kann zu diesem Zweck temporäre Informationen in den Taskvektor schreiben. Ebenso können Kommandos oder Makroaufrufe zurückgewiesen werden. Die geschilderte Überwachung der Dateizugriffe kann eingeschränkt werden auf bestimmte Benutzerkennungen und auf bestimmte Dateien.

ADD-USER-/MODIFY-USER-ATTRIBUTES-Kommando

Mit dem Operanden FILE-AUDIT wird für eine Benutzerkennung festgelegt, ob der Benutzer mit den Kommandos CREATE-FILE-[GROUP]/MODIFY-FILE[-GROUP]-ATTRIBUTES oder dem CATAL-Makro das AUDIT-Attribut für seine Dateien ändern darf; Standardwert: FILE-AUDIT=NO.

Mit dem Kommando SHOW-USER-ATTRIBUTES kann der Benutzer den Eintrag in seiner Kennung im Benutzerkatalog überprüfen.

CREATE-FILE-[GROUP]/MODIFY-FILE[-GROUP]-ATTRIBUTES-Kommando

Mit dem Operanden *AUDIT=*NONE/*SUCCESS/*FAILURE/*ALL* können Benutzer oder Systembetreuung für jede Datei festlegen, ob und wie Dateizugriffe durch Exit-Routinen überwacht werden sollen.

*NONE	Standardwert; keine Überwachung
*SUCCESS	erfolgreiche Kommando-/Makroverarbeitung wird überwacht
*FAILURE	fehlerhafte Kommando-/Makroverarbeitung wird überwacht
*ALL	jede Kommando/Makroverarbeitung wird überwacht

Mit dem SHOW-FILE-ATTRIBUTES-Kommando/-Makro kann der Inhalt des AUDIT-Feldes im Katalogeintrag überprüft werden.

Lage des AUDIT-Indikators in Systemtabellen (s. Anhang):

CATALOG Parameterbereich

IDKAUDIT	DC	X '00'	AUDIT PARAMETER
IDKNOAUD	EQU	X '00'	AUDIT NOT SPECIFIED
IDKAUDSU	EQU	X '01'	AUDIT=SUC
IDKAUDFA	EQU	X '02'	AUDIT=FAIL
IDKAUDAL	EQU	X '04'	AUDIT=ALL
IDKAUDNO	EQU	X '08'	AUDIT=NONE

2.4 Programmierbeispiel

- [Systemumgebung der System Exits](#)
- [Konzept für den modularen Aufbau von Exit-Routinen](#)
- [Beispiel für eine Basisroutine mit Subroutinen](#)

2.4.1 Systemumgebung der System Exits

Exit-Routinen werden als Bestandteile des Betriebssystems betrachtet. Privilegierte Systemschnittstellen werden adressierungsmodus-unabhängig im 31-Bit-Format angeboten. Zur Vermeidung von (hardware-abhängigen) Umschaltungen des Adressierungsmodus an den entsprechenden Schnittstellen müssen die Exit-Routinen in demselben Adressierungsmodus laufen wie das System.

Für Exit-Routinen gilt also:

- sie müssen in jedem Adressierungsmodus ablaufen können (AMODE ANY)
- sie müssen die für das Betriebssystem gültigen Programmierregeln einhalten
- sie müssen RMODE ANY als Attribut besitzen

Exit-Routinen können auf Adressfelder zugreifen, die von Benutzerprogrammen versorgt werden (z.B. in Parameterlisten). Diese Felder müssen von den System-Exit-Routinen vor der Weiterverarbeitung bereinigt bzw. für 31-Bit-Adressierung aufbereitet werden.

System Exits auf Anlagen mit x86-Architektur

Wie alle privilegiert ablaufenden Routinen müssen auch System Exit-Routinen direkt (native) auf der CPU ablaufen, d.h. als x86-Code direkt auf der x86-CPU.

Falls Sie System Exits einsetzen, muss der Quell-Code der System Exits angepasst und neu übersetzt werden. Dazu bietet Ihnen Fujitsu Technology Solutions Beratung und Unterstützung. Wenden Sie sich bitte an Ihren Service-Beauftragten.

2.4.2 Konzept für den modularen Aufbau von Exit-Routinen

Eine Exit-Routine enthält meist mehrere Funktionen (beim LOGON-Exit zum Beispiel: Prüfen der Benutzererkennung, Prüfen der Abrechnungsnummer und Prüfen des LOGON-Kennworts). Aber auch zusätzliche Prüfungen können über diese Schnittstelle realisiert werden. Da an einen System Exit nur eine Routine angeschlossen werden kann und jedes RZ andere Funktionen integriert, sind die Routinen in den verschiedenen Rechenzentren untereinander nicht austauschbar. Die einzelnen Funktionen wären aber austauschbar.

Wir schlagen daher als Konzept für den Aufbau von Exit-Routinen vor, dass Exit-Routinen sich aus einer Basisroutine und einer oder mehreren Subroutinen zusammensetzen sollen.

Basisroutine

Die Basisroutine ist eine allgemeine Routine, die bei allen System Exits benutzt werden kann. In ihr sind noch keine speziellen Servicefunktionen enthalten. Sie dient ausschließlich dazu, eine oder mehrere Subroutinen an einen System Exit anzuhängen. Die Subroutinen werden in der gewünschten Reihenfolge durchlaufen. Der Returncode aus der Subroutine entscheidet über den Aufruf weiterer Subroutinen oder die Beendigung der Basisroutine.

Subroutine

Eine Subroutine enthält jeweils nur eine funktionell in sich geschlossene Einheit. Mit geringem Programmieraufwand lässt sich aus bereits existierenden Subroutinen ein individuelles Paket für jeden System Exit zusammenstellen.

2.4.3 Beispiel für eine Basisroutine mit Subroutinen

Wir wählen bewusst das Hilfsmittel Columbus-Assembler, weil wir so auf explizites Registersichern und Rückladen verzichten können. Der Benutzer braucht nur das angebotene Gerüst zu übernehmen und kann nach Belieben den Rest der Routinen in Assembler oder Columbus-Assembler schreiben.

i Die Namen von Basisroutine und Subroutinen müssen systemweit eindeutig sein und dürfen nicht mit Namen in Systemmodulen kollidieren.

Basisroutine

```

        START
EXITCOD  AMODE  ANY
EXITCOD  RMODE  ANY
EXPARI   EX090                DSECT for Exit-090
EXITCOD  CSECT
*****
*        MAINROUTINE
*        R1 = EXPARI A(PARAMETERLIST OF EXIT#90
*****
EXIT090  @ENTR  TYP=E,ENV=SPLSPEC,RETURNS=NO
        ENTRY  EXIT090
        USING  EXPARI,R1                Use layout of parlist of Exit-090
*****
*        AUFRUF 1.SUBROUTINE
*****
        @PASS  EXTNAME=SUBROUT1        Call Subrout1
        @IF    NZ                        If returncode not ok
        LTR    R15,R15
        @THEN
        ST     R15,IEXRC                set returncode in exit paralist
        @EXIT                                then return to system
        @BEND
*****
*        AUFRUF 2.SUBROUTINE
*****
        @PASS  EXTNAME=SUBROUT2        Call Subrout2
        @IF    NZ                        If returncode not ok
        LTR    R15,R15
        @THEN
        ST     R15,IEXRC                set returncode in exit paralist
        @EXIT                                then return to system
        @BEND

```

```

*****
*      AUFRUF WEITERER ROUTINEN      *
*****
      :
      :
      @PASS  EXTNAME=SUBROUTN      Call Subrout2
      @IF    NZ                    If returncode not ok
      LTR    R15,R15
      @THEN
      ST     R15,IEXRC             set returncode in exit paralist
      @EXIT
      @BEND
*****
*      END OF MAINROUTINE          *
*****
      @EXIT
      @END
      END

```

Subroutine

```

SUBEX001 CSECT
      PRINT  GEN
SUBEX001 AMODE ANY
SUBEX001 RMODE ANY
EXITPL  EX090
SUBROUTN @ENTR  TYP=I,ENV=SPLSPEC,RETURNS=NO
      ENTRY  SUBROUTN
*****
*      Anwendungsspezifische Befehle      *
*****
      L     R15,IADD              RETURNCODE WEITERREICHEN
      @EXIT ,                    RETURN TO BASISROUTINE
      @END ,
      END

```

2.5 Verwaltung von Exit-Routinen mit DSSM

Exit-Routinen sind Subsysteme.

Mit den Funktionen des Dynamischen Subsystem Managements (DSSM) kann die Systembetreuung im laufenden Betrieb Subsysteme verfügbar machen, anhalten, fortsetzen oder entladen.

Die statische Subsystemkatalog-Verwaltung (SSCM) generiert einen statischen Subsystemkatalog (SSMCAT), in dem die Subsystem-Konfiguration definiert ist. Dieser Subsystemkatalog wird durch die dynamische Subsystem-Verwaltung (DSSM) zum Zeitpunkt der Systemeinleitung geladen. Durch das Laden wird aus dem statischen ein dynamischer Subsystemkatalog.

Während des Systemlaufs kann der Subsystemkatalog durch das DSSM-Kommando ADD-SUBSYSTEM verwaltet werden.

Mithilfe der Kommandos START-SUBSYSTEM, HOLD-SUBSYSTEM, RESUME-SUBSYSTEM, REMOVE-SUBSYSTEM und STOP-SUBSYSTEM können Subsysteme dynamisch verwaltet werden.

Mit dem Kommando SHOW-SUBSYSTEM-STATUS kann die Systembetreuung Informationen über Subsysteme anfordern.

Mit dem Kommando SET-DSSM-OPTIONS kann eine Logging-Funktion für Diagnosezwecke eingeschaltet werden.

Wenn System-Exit-Routinen Entrys enthalten, die in anderen Subsystemen aufgelöst werden, müssen diese in der Deklaration des Subsystems (Anweisung SET-SUBSYSTEM-ATTRIBUTES, Operand REFERENCED-SUBSYSTEM) angegeben werden.

Wenn System-Exit-Routinen privilegierte Schnittstellen benutzen, die mit einem Subsystem verbunden werden (z.B. mit SPOOL), muss das Subsystem mit STOP-SUBSYSTEM vor dem Shutdown explizit beendet werden.

Ausführlichere Informationen zu DSSM und zu den Kommandos finden Sie in den Handbüchern „Verwaltung von Subsystemen“ [16] und „Kommandos“ [8].

Beispiel

Die Exit-Routine EXJOI010 soll als Subsystem deklariert werden. Standardwerte sind in den Anweisungen nicht angegeben; die Anweisungen sind im Handbuch „Verwaltung von Subsystemen“ [16] beschrieben.

```

/START-SSCM _____ (1)
//START-CATALOG-CREATION CATALOG-NAME=$TSOS.SSMCAT.NEW _____ (2)
. . .
//SET-SUBSYSTEM-ATTRIBUTES -
//  SUBSYSTEM-NAME=EXJOIN(VERSION=01.0),LIBRARY=EXLIB, - _____ (3)
//  REP-FILE=*NO,DYNAMIC-CHECK-ENTRY=*NO, -
//  CREATION-TIME=*AFTER-SYSTEM-READY, -
//  MEMORY-CLASS=*SYSTEM-GLOBAL(SUBSYSTEM-ACCESS=*SYSTEM), - _____ (4)
//  SUBSYSTEM-ENTRIES=EXJOI010( - _____ (5)
//    MODE=SYSTEM-EXIT(NUMBER=10), -
//    CONNECTION-ACCESS=*SYSTEM, -
//    CONNECTION-SCOPE=*FREE), - _____ (6)
//  LINK-ENTRY=EXJOIN(AUTOLINK=*ALLOWED) _____ (7)
. . .
//CHECK-CATALOG
//SAVE-CATALOG
//END

```

(1) Aufruf des Subsystems SSCM zur Generierung eines statischen Subsystemkatalogs (SSMCAT).

- (2) Definieren der Datei, die den SSMCAT aufnehmen soll.
- (3) Festlegen der subsystemspezifischen festen Attribute; der Name des Subsystems (EXJOIN) ist frei wählbar.
- (4) Adressraumangaben; die Exit-Routine wird in den privilegierten Systemadressraum geladen.
- (5) Definieren von Einsprungstelle (EXJOI010), Art des Subsystems (System Exit), Nummer des Exits und Zugriffsberechtigung nur für privilegierte Programme.
- (6) Zusätzlich kann der Operand `CONNECTION-SCOPE=*FREE` verwendet werden, wenn die Exit-Routine nicht durch die Tasks, die sie benutzen, für das Entladen gesperrt werden soll.
- (7) Steuerung des Binderlaufs bei der Erzeugung des Subsystems EXJOIN.

Gemäß Subsystem-Deklaration im Beispiel oben wird die Exit-Routine nach System-Ready automatisch gestartet (Operand `CREATION-TIME=*AFTER-SYSTEM-READY`).

Falls `CREATION-TIME=*AT-CREATION-REQUEST` definiert wurde, muss die so deklarierte Exit-Routine mit dem Kommando

```
/START-SUBSYSTEM SUBSYSTEM-NAME=EXJOIN
```

im System verfügbar gemacht werden.

Mit dem Kommando

```
/HOLD-SUBSYSTEM SUBSYSTEM-NAME=EXJOIN
```

wird die Exit-Routine in den Wartezustand versetzt.

Die angehaltene Exit-Routine kann mit

```
/RESUME-SUBSYSTEM SUBSYSTEM-NAME=EXJOIN
```

wieder aktiviert werden.

Das Kommando

```
/STOP-SUBSYSTEM SUBSYSTEM-NAME=EXJOIN
```

bewirkt Deaktivierung der Exit-Routine und Freigabe des belegten Speicherplatzes.

Die Funktionen HOLD und DELETE sperren zunächst die Exit-Routine gegen weitere Aufrufe und warten anschließend, bis der letzte Aufrufer die Routine verlassen hat. Erst dann wird die gewünschte Funktion durchgeführt.

Informationen über den aktuellen Status der Exit-Routine können mit dem Kommando

```
/SHOW-SUBSYSTEM-STATUS SUBSYSTEM-NAME=EXJOIN
```

angefordert werden.

Die aktive Version der Exit-Routine kann durch eine neue Version ersetzt werden. Die Voraussetzung ist, dass die neue Version mit dem Attribut `VERSION-EXCHANGE=*ALLOWED` deklariert wurde, also die temporäre Koexistenz zweier Versionen eines Subsystems erlaubt.

Das Kommando lautet:

```
/START-SUBSYSTEM SUBSYSTEM-NAME=EXJOIN,
```

VERSION='02.2',VERSION-PARALLELISM=*EXCHANGE-MODE

3 Beschreibung der Exits

In diesem Kapitel sind die zurzeit freigegebenen Exits beschrieben.

Die in „[Exit-Routinen](#)“ enthaltenen Programmierregeln sind unbedingt einzuhalten. Wir verweisen an dieser Stelle nochmals auf das [Programmierbeispiel](#).

3.1 Termination-Exits (001/002)

Vor und nach Programmbeendigung bzw. vor Taskbeendigung kann eine Exit-Routine angeschlossen werden.

Am Termination Request Exit wird die Steuerung vor Programm-/ bzw. Taskbeendigung (TERM, ABEND, LOGOFF) an die entsprechende Exit-Routine übergeben.

Am Termination Return Exit wird eine Exit-Routine nach Programmbeendigung aufgerufen.

Die Exit-Routine kann

- eigene Abrechnungsdaten schreiben,
- eine abschließende Dateibehandlung durchführen,
- den globalen Speicherbereich (SYSVECT) auswerten oder
- bei Task-Termination, einen LOGOFF-/ABEND-Parameter verändern.

Request Exit (001)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A($TERMRF-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Return Exit (002)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A($TERMRF-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Die Rückkehrinformation an die rufende Systemkomponente muss im Feld ENRTMRET des Parameterbereichs hinterlegt werden.

Hinweise

- Allgemeines

Die Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.

Aufrufe, die zu einer weiteren Termination-Behandlung führen (EXEC, LOAD, TERM, \$TERM), sind verboten.

Ein Zurücksetzen oder ein Abbruch der Termination-Behandlung ist nicht möglich.

Der Makroaufruf TYPIO mit Anfordern einer Operateurantwort und CLOSE ALL sollen in der Exit-Routine nicht verwendet werden.

Falls in den Exit-Routinen Abrechnungsdaten erfasst werden, sollten die Exits vor SHUTDOWN nicht deaktiviert werden, solange noch Benutzertasks im System vorhanden sind. Da SHUTDOWN zur Zwangsbeendigung von Tasks führt, könnten sonst Daten verloren gehen.

- Dateiverarbeitung

Dateien, die von Exit-Routinen geführt werden, müssen mit dem Operanden `USER-ACCESS=*ALL-USERS` unter einer beliebigen Benutzerkennung im System eingerichtet werden. Grund: Programm- und Task-Beendigung laufen unter Steuerung der betroffenen Task ab und damit der entsprechenden Benutzerkennung. Der Zugriff auf die Datei muss also von allen Benutzerkennungen möglich sein.

Für die von Exit-Routinen geführten Dateien müssen alle `EXLST`-Ausgänge versorgt sein. Sonst besteht beim Termination Request für Programmbeendigung die Gefahr, dass das Benutzerprogramm mit dem DMS-Fehlercode der Exit-Routine abgebrochen wird.

Für das ordnungsgemäße Schließen der im Exit eröffneten Dateien ist die Exit-Routine selbst verantwortlich.

- Task Termination

Der Inhalt des Feldes `ENRTLGFC` des `$TERMRF`-Parameterbereichs (den der Benutzer modifizieren kann) wird übernommen, falls dieser einen gültigen Wert enthält.

Bei der Beendigung von Systemtasks oder `ARCHIVE`-Subtasks erfolgt kein Ansprung der Exit-Routinen.

- Return Exit bei Programmbeendigung

Zum Aufrufzeitpunkt sind alle Dateien des Benutzerprogramms bereits geschlossen (impliziter `CLOSE ALL` durch die Termination-Behandlung).

DSECT

```

$TERMRF MF=D
MFCHK MF=D,
    SUPPORT=(C,D,L),
    PREFIX=E,
    MACID=NRT,
    DMACID=NRT,
    DNAME=NRTPL,
    PARAM=
ENRTPL DSECT ,
    *,##### PREFIX=E, MACID=NRT #####
#INTF REFTYPE=REQUEST,
    INTNAME=NRTPARA,
    INTCOMP=001
*
* THIS MACRO GENERATES THE DSECT FOR THE PARAMETER LIST WHICH IS
* EXCHANGED BETWEEN THE MODULS NRTCTRL, NRTERM AND EXTERNAL MODULS.
*
* TO PREVENT ABUSE OF THE GIVEN EQUATES, THE TEST METHOD TO BE USED IS
* INDICATED BY THE ABBREVIATIONS:
* (TM) EACH BIT OF THE DATA FIELD HAS A MEANING OF ITS OWN,
* NO MEANING IS DISPLAYED AS A COMBINATION OF BITS.
* USE THE "TM"-STATEMENT WHEN TESTING IF A BIT IS SET.
* THIS ALSO ALLOWS TESTING OF TWO OR MORE DIFFERENT
* FEATURES AT THE SAME TIME.
* (CL) ONLY A CERTAIN COMBINATION OF SET BITS INDICATES A
* FEATURE. USE THE "CLI"-STATEMENT WHEN TESTING A
* COMBINATION.
*
FHDR MF=(C,ENRT),EQUATES=NO
DS 0A
ENRTFHE DS 0XL8 0 GENERAL PARAMETER AREA HEADER
*
ENRTIFID DS 0A 0 INTERFACE IDENTIFIER
ENRTFCTU DS AL2 0 FUNCTION UNIT NUMBER
* BIT 15 HEADER FLAG BIT,
* MUST BE RESET UNTIL FURTHER NOTICE
* BIT 14-12 UNUSED, MUST BE RESET
* BIT 11-0 REAL FUNCTION UNIT NUMBER
ENRTFCT DS AL1 2 FUNCTION NUMBER
ENRTFCTV DS AL1 3 FUNCTION INTERFACE VERSION NUMBER
*
ENRTRET DS 0A 4 GENERAL RETURN CODE
ENRTSRET DS 0AL2 4 SUB RETURN CODE
ENRTSR2 DS AL1 4 SUB RETURN CODE 2
ENRTSR1 DS AL1 5 SUB RETURN CODE 1
ENRTMRET DS 0AL2 6 MAIN RETURN CODE

```

```

ENRTMR2 DS AL1 6 MAIN RETURN CODE 2
ENRTMR1 DS AL1 7 MAIN RETURN CODE 1
ENRTFHL EQU 8 8 GENERAL OPERAND LIST HEADER LENGTH
*
* ORG ENRTRET
* *****
ENRTRC DS F * RETURN CODE *
* *****
* PLEASE USE THE RETURNCODE
* IN THE STANDARDHEADER
* (MAINCODE 1)!
* -----
* THIS IS THE ONLY PART OF
* NRTPARA THAT MAY BE MANI- 120
* PULATED BY THE CALLED
* ROUTINES
* ORG ENRTRC
ENRTSCD DS XL1 (CL) SECONDARY CODE
ENRTNACT EQU X'0C' NO ACTION PERFORMED
*
* DS XL2 UNUSED
*
ENTRCDS DS XL1 (CL) RETURN CODE
ENRTSUCC EQU X'00' FUNCT. SUCCESSFULLY PERFORMED
ENRTERR EQU X'04' FUNCT. FAULTY - TERM. TASK
*
ENRTRES1 DS 3F USED ONLY BY NRT - NOT FOR
* EXTERNAL ROUTINES!
*
ENRTWORK DS A GENERAL WORK AREA ADDRESS
*
ENRTOUT DS XL1 (CL) OUTPUT PARAMETER FIELD 120
*
ENRTDMPN EQU 0 CDUMP SHOULD BE NOT CALLED 120
ENRTDMPY EQU 1 CDUMP CAN BE CALLED 120
ENRTRETN EQU 2 RETURN TO PROGRAM 120
ENRTCMDP EQU 3 CALL COMMAND PROCESSING 120
ENRTCONT EQU 4 CONTINUE TERMINATION 120
*
ENRTDUMP DS XL1 (TM) DUMP REQUEST 120
*
ENRTDIAG EQU X'04' DIAGNOSIS DUMP REQUESTED 121
ENRTSDMP EQU X'02' SYSTEM DUMP REQUESTED 121
ENRTUDMP EQU X'01' USER DUMP REQUESTED 121
ENRTDPNN EQU X'00' NO DUMP REQUESTED 121
*
* DS XL2 RESERVED 120
*

```

```

ENRTINFO DS      A                SPECIFIC INFORMATION FOR THE
*                CALLED ROUTINE. CURRENTLY ONLY
*                NAME MANAGER INFORMATION IS
*                PROVIDED WHEN REQUIRED.

          ORG      ENRTINFO
ENRTNMCL DS      XL3              NAME MANAGER INFORMATION FIELD.
ENRTCLAS DS      AL1              (CL) INDICATES THE CURRENT NAME
*                MANAGER CLASS OR ZERO FOR
*                DYNAMIC CLASSES.
*
ENRTREQ  DS      XL1              (TM) TERMINATION REQUEST.
*
ENRTNPTR EQU      X'01'           NONPRIVILEGED TERMINATION
*                REQUEST (TERM,TERMD,TERMJ,TRMJD)
ENRTTERM EQU      X'02'           PRIVILEGED TERMINATION REQUEST
*                ($TERM) AND NONPRIVILEGED
*                REQUEST ISSUED IN P2
ENRTTRMT EQU      X'04'           TERMINATION REQUEST DUE TO
*                SYSTEM ERROR
*
*
*
ENRTFUNT DS      XL1              (CL) TERMINATION UNIT TO BE
*                EXECUTED.
ENRTPRGR EQU      X'01'           TERMINATION UNIT IS
*                PROGRAM.
ENRTSTEP EQU      X'02'           TERMINATION UNIT IS STEP.
ENRTTASK EQU      X'04'           TERMINATION UNIT IS TASK.
ENRTJOB  EQU      ENRTTASK        TERMINATION UNIT IS JOB.
*                AT PRESENT JOB IS EQUIVALENT
*                TO TASK
*
*
*
ENRTTYP  DS      XL1              (CL) TERMINATION TYP
*
ENRTNORT EQU      X'00'           NORMAL TERMINATION
ENRTABNT EQU      X'04'           ABNORMAL TERMINATION
*
*
ENRTCUNT DS      XL1              (CL) CURRENT TERMINATION UNIT.
*
*                USE THE EQUATES DEFINED UNDER
*                "TERMINATION UNIT TO BE
*                EXECUTED" FOR PROGRAM, JOB
*                STEP, TASK AND JOB TERMINATION!
*

```

120

```

          DS      0F
ENRRCMPC DS      CL4          COMPLETION CODE FOR JOB VARIABLE
*
*
ENRRCMD  DS      XL1          (TM) INFORMATION ON THE COMMAND THAT
*
ENRTLGOFF EQU     X'01'      LOGOFF COMMAND
ENRTABND EQU     X'02'      ABEND COMMAND
ENRTCAN  EQU     X'04'      CANCEL COMMAND
ENRTSHUT EQU     X'08'      SHUTDOWN COMMAND
ENRTEXLD EQU     X'10'      EXEC OR LOAD COMMAND
ENRTPOX  EQU     X'20'      EVENT=POSIX                      121
*
*
ENRTLGFC DS      XL1          (TM) INFORMATION ON LOGOFF COMMAND.
*
ENRTLGBT EQU     X'01'      LOGOFF BUT
ENRTLGTP EQU     X'02'      LOGOFF TAPE
ENRTLGNS EQU     X'10'      LOGOFF NOSPOOL
ENRTLGNO EQU     X'06'      EXIT-JOB - NO SYSOUT          201
ENRTLGNL EQU     X'0A'      EXIT-JOB - NO SYSLST          201
*
*
ENRTCANC DS      XL1          (TM) INFORMATION ON CANCEL COMMAND
*
ENRTOPEP EQU     X'02'      CANCEL BY OPERATOR-TASK
ENRTSOS  EQU     X'04'      CANCEL BY $TSOS
ENRTUSID EQU     X'08'      CANCEL BY SAME USER-ID
ENRTHOST EQU     X'10'      CANCEL BY ANOTHER HOST
*
*
          DS      CL1          RESERVED                      120
*
*
*
          DS      XL1
ENRTRMPC DS      CL7          TERMINATION CODE - 7 BYTES
          ORG     ENRTRMPC+3
ENRTCDC  DS      CL4          TERMINATION CODE - 4 BYTES
*
*
*
ENRTPLE  DS      0F          WORD ALIGNMENT IS SUFFICIENT 120
*
ENRT#    EQU     *-ENRTFHE

```

3.2 SHUTDOWN-Exit (003)

Der SHUTDOWN-Exit wird während der Beendigung eines Systemlaufs angestoßen.

Die entsprechende Exit-Routine kann eigene Abrechnungsdaten schreiben, statistische Werte festhalten usw.

Folgende Informationen werden an die EXIT-Routine übergeben:

```
R1 = A(Parameterbereich $SHUIND)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Hinweise

- Die Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Die SHUTDOWN-Exit-Routine läuft unter Steuerung der Operatortask ab. Alle Programmfehler führen deshalb zum abnormalen Systemende.
- Vor Ansprung der SHUTDOWN-Exit-Routine werden für jede zwangsweise zu beendende Task eventuell vorhandene Termination-Exit-Routinen angesprungen.

3.3 OPEN-Element-Exits (005/006)

Das Eröffnen von Elementen einer Programm-Bibliothek kann über zwei Exit-Punkte von der Systembetreuung überwacht werden.

- Am Request Exit kann die Systembetreuung den OPEN-Aufruf zurückweisen, bevor der OPEN ausgeführt wird.
- Am Return Exit ist das Element bereits eröffnet und die volle Identifikation des Elements (Typ, Name, Version, Variante) bekannt. Da beim Request Exit die Identifikation des Elements auf Grund der Suchlevel-Option noch unvollständig sein kann, bietet der Return Exit hier noch einmal die Möglichkeit, den OPEN-Aufruf zurückzuweisen. Das Element wird dann in den Originalzustand zurückversetzt.

Request Exit (005)

Folgende Informationen werden an die Exit-Routinen übergeben:

```
R1 = A(Exit-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

(siehe „System Exits für Programmbibliotheken“)

Rückkehrinformation im Register 15:

```
R15 = X'000000RR'
RR = Rückkehrinformation der Exit-Routine
00 kein Fehler
04 OPEN-Element abgewiesen
```

Rückkehrinformation im Exit-Parameterbereich:

Zusätzlich zum Returncode in Register 15 werden die Felder PMOPM (Open Modus), PMSLV (Search Level) und PMLVD (Search Default) versorgt.

Return Exit (006)

Der Return Exit ist identisch mit dem Request Exit.

Hinweise

- Der Search Level (PMSLV) ist die Hierarchie-Stufe innerhalb der Element-Identifikation. Da nur eine Variante erlaubt ist, sind „NO LEVEL“ und „VARIANT“ gleichbedeutend.
- Mit „SEARCH LEVEL = NO LEVEL“ wird ein vollständig beschriebenes Element erwartet.
- Der Search Default (PMLVD) beschreibt, ob das erste oder letzte Element auf dem angegebenen Search Level eröffnet werden soll.

DSECT

Die DSECT für den Parameterbereich kann mit dem Makro EX005 generiert werden:

```

name EX005 D[, [prefix], [ALL]]
        EX005 D
IEX005  DSECT
        DS      0F
*****
* OPEN    PARAMETER LIST FOR RZ ROUTINES          *
*****
PMRC      DC      F'0'          PLAM RETURN CODE
PMRC1     DC      F'0'          PLAM SECONDARY RETURN CODE
PMP LV    DC      CL8' '        VERSION OF PLAM: VZZ.ZBZZ
PMLIB     DC      A(0)          A(LIBRARY NAME): CL54
PMLINK    DC      A(0)          A(LIBRARY NAME LINK): CL8
PMOPM     DC      CL1' '        OPEN MODUS: I: INPUT
*                                     W: WRITE
*                                     U: UPDATE
PMSLV     DC      CL1' '        SEARCH LEVEL: L: LIBRARY
*                                     T: TYPE
*                                     N: NAME
*                                     V: VERSION
*                                     R: VARIANT
*                                     O: NO LEVEL
PMLVD     DC      CL1' '        SEARCH DEFAULT: N: NO
*                                     H: HIGH
*                                     L: LOW
          DC      CL1' '
PMTYP     DC      A(0)          A(MEMBER TYPE): CL8
PMNAM     DC      A(0)          A(MEMBER NAME): CL64
PMVER     DC      A(0)          A(MEMBER VERSION): CL24
PMVAR     DC      A(0)          A(MEMBER VARIANT): F
PMDAT     DC      A(0)          A(MODIFICATION DATE): CL14: DATE
*                                     ,CL8: TIME
          DC      A(0)
          DC      A(0)
          DC      A(0)
LEN        EQU      *-PMRC

```

3.4 CLOSE-Element-Exits (007/008)

Das Schließen von Programmbibliothekselementen kann über zwei Exit-Punkte überwacht werden:

- Am Request Exit kann sich die Systembetreuung Informationen über das Element verschaffen wie beim Return Exit des OPEN auf das Element; der CLOSE auf das Element kann nicht verhindert werden.
- Am Return Exit ist das Element bereits geschlossen. Die Systembetreuung erhält hier die volle Identifikation des Elements mit Modifikationsdatum neuerstellter oder geänderter Elemente.

Request Exit (007)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(Exit-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

(siehe „[System Exits für Programmbibliotheken](#)“)

Rückkehrinformation:

```
R15 = X'000000RR'
RR = Rückkehrinformation der Exit-Routine
00 kein Fehler
04 CLOSE-Element abgewiesen (von PLAM ignoriert)
```

Zusätzlich zum Returncode im Register 15 wird das Feld PMCLM versorgt.

Return Exit (008)

Der Return Exit ist identisch mit dem Request Exit.

DSECT

Die DSECT für den Parameterbereich kann mit dem Makro EX007 generiert werden:

```

name EX007 D[, [prefix], [ALL]]
        EX007 D
IEX007  DSECT
        DS      0F
*****
* CLOSE   PARAMETER LIST FOR PLAM RZ ROUTINES      *
*****
PMRC     DC      F'0'          PLAM RETURN CODE
PMRC1    DC      F'0'          PLAM SECONDARY RETURN CODE
PMPLV    DC      CL8' '       VERSION OF PLAM: VZZ.ZBZZ
PMLIB    DC      A(0)         A(LIBRARY NAME): CL54
PMLINK   DC      A(0)         A(LIBRARY NAME LINK): CL8
        DC      CL1' '
        DC      CL1' '
        DC      CL1' '
PMCLM    DC      CL1' '       CLOSE MODE: Y: KEEP
*                               N: DELETE
PMTYP    DC      A(0)         A(MEMBER TYPE): CL8
PMNAM    DC      A(0)         A(MEMBER NAME): CL64
PMVER    DC      A(0)         A(MEMBER VERSION): CL24
PMVAR    DC      A(0)         A(MEMBER VARIANT): F
PMDAT    DC      A(0)         A(MODIFICATION DATE): CL14: DATE
*                               ,CL8: TIME
        DC      A(0)
        DC      A(0)
        DC      A(0)
LEN      EQU     *-PMRC

```

3.5 JOIN-Exit (010)

Mit der JOIN-Exit-Routine kann ein /ADD-USER- oder /MODIFY-USER-ATTRIBUTES-Kommando angenommen oder abgewiesen werden. Die für die Entscheidung notwendigen Informationen werden in einer variablen Änderungsliste zur Verfügung gestellt. Die Änderungsliste enthält die durch das Kommando gestellten Änderungsanforderungen in Form einer variablen Liste. Der Aufbau ist dabei für den Neueintrag einer Benutzerkennung (ADD-USER) oder die Änderung der Attribute einer bestehenden Benutzerkennungen (MODIFY-USER-ATTRIBUTES) ähnlich. Die Adresse der variablen Änderungsliste wird im Exit-Parameterbereich \$SRMSYE nach dem Standardheader im Feld SRMSYEPT übergeben.

Zusätzliche Informationen über die betroffenen Benutzerkennungen oder andere Systemwerte können mit den entsprechenden Systemschnittstellen abgefragt werden.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(Parameterbereich $SRMSYE)
R2 .... R11 sind undefiniert.
R12 = A(TPR Program Manager)
R13 = A(Save Area)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Das Format der Einträge in der Änderungsliste ist im Makro SJMCHKZ beschrieben. Anhand der aufgebauten Liste kann auf die Operanden des angegebenen Kommandos rückgeschlossen werden.

Hinweis

Das neue Kennwort wird stets unverschlüsselt an der Schnittstelle bereitgestellt.

Die **Änderungsliste** für das **ADD-USER-Kommando** hat folgendes Format:

Benutzerkennung des Kommandogegers	8 Byte
Benutzer-Kennzeichen	X'01'
Zu bearbeitende Benutzerkennung	8 Byte
Gruppen-Kennzeichen	X'02'
Gruppen-Kennung	8 Byte
:	:
:	:
Ende-Kennzeichen	X'FF'

Die Liste wird fortgesetzt durch Werte, die in den Benutzer-Satz nicht eingehen, die aber zu Auswertungszwecken genutzt werden können.

Encrypt-Kennzeichen	1 Byte	"Encrypt-Kennzeichen"
---------------------	-----------	-----------------------

Encrypt	1 Byte	Encrypt-Wert
		01 - yes
		02 - no
Pubset-Kennzeichen	1 Byte	"Pubset-Kennzeichen"
Pubset-Kennung	4 Byte	Pubset (evtl. # für Home)

Die **Änderungsliste** für das **MODIFY-USER-ATTRIBUTES-Kommando** hat folgendes Format:

Benutzerkennung des Kommandogebers	8 Byte
Benutzer-Kennzeichen	X'01'
Zu bearbeitende Benutzerkennung	8 Byte
:	:
Änderungs-Kennzeichen	X'xx'
Änderungsdatum	Anzahl Bytes = Länge Änderungsdatum
:	:
:	:
Ende-Kennzeichen	X'FF'

Die Liste wird fortgesetzt durch Werte, die in den Benutzer-Satz nicht eingehen, die aber zu Auswertungszwecken genutzt werden können.

Encrypt-Kennzeichen	1 Byte	"Encrypt-Kennzeichen"
Encrypt	1 Byte	Encrypt-Wert
		01 - yes
		02 - no
Pubset-Kennzeichen	1 Byte	"Pubset-Kennzeichen"
Pubset-Kennung	4 Byte	Pubset (evtl. # für Home)

Bedeutung

Encrypt gibt an, ob das angegebene Passwort verschlüsselt (YES/01) oder unverschlüsselt (NO/02) in den Benutzer-Satz übernommen werden soll.

Die Pubset-Kennung gibt an, auf welches Pubset das Kommando wirken soll.

Format zusammengesetzter Einträge:

Einige Einträge in der Änderungsliste bestehen nicht aus einem einzelnen (in seiner Bedeutung offensichtlichen) Wert, sondern sind aus mehreren Einzelwerten zusammengesetzt. Es sind dies:

Kennzeichen		Feld	Länge
Name	Wert		
PASS	X'03'	Encryption (1=ja,0=nein)	1 Byte
		Neues Kennwort	8 Byte
		----- leer -----	8 Byte
TESTPRIV	X'10'	READ-Privileg	1 Byte
		WRITE-Privileg	1 Byte
		CONTROLLED-Indikator	1 Byte
USS	X'15'	Schalter ON	4 Byte
		Schalter OFF	4 Byte
		Schalter Wert	4 Byte

Im Falle eines MODIFY-USER-ATTRIBUTES-Kommandos werden die Accountnummerdaten mehrfach abgelegt, weil eine Accountnummer neuangelegt werden muss bzw. die Daten zu einer bestehenden Accountnummer nur geändert werden müssen. Dadurch kommt es zu Wiederholungen in der variablen Liste.

Die Accountnummern werden deshalb in der Liste folgendermaßen abgelegt:

- komplette Liste mit Werten für eine Neuaufnahme der Accountnummer
- angefügte Liste der angegebenen Werte zu einer bestehenden Accountnummer.

Vollständige Angabe der Werte für eine Accountnummer

Account-Kennzeichen	X'1A'
Account-Nummer	8 Byte'
NTL-Kennzeichen	x'1B'
NTL-Wert :	1 Byte

:	:
:	:
CLASS-Kennzeichen	X'21'
CLASS-Wert	1 Byte

Effektiv angegebene Werte zu einer Accountnummer

:	:
TIME-Kennzeichen	X'1F'
TIME-Wert :	4 Byte
:	:

Die Listen können anhand des Makros SJMCHKZ analysiert werden.

Die Rückkehrinformation an die rufende Systemkomponente erfolgt über den Maincode1 des Parameterbereichs \$SRMSYE im rechten Byte des Feldes SRMSYEHDR.

SRMMR1=X'00'	Das Kommando soll ausgeführt werden.
SRMMR1=X'01'	Das Kommando soll abgewiesen werden.

Hat die Exit-Routine das ADD-USER-/MODIFY-USER-ATTRIBUTES-Kommando zurückgewiesen, wird folgende Meldung ausgegeben:

% SRM2108 KOMMANDO WURDE DURCH EINE SYSTEM-EXIT-ROUTINE ABGEWIESEN

DSECT

```

        $SRMSYE MF=D
        MFTST MF=D,PREFIX=S,MACID=RMF,ALIGN=F,                                C
            DMACID=RMF,SUPPORT=(D,C,M,L),DNAME=RMF_MDL
SRMF_MDL DSECT ,
        *,##### PREFIX=S, MACID=RMF #####
*   parameterarea description
SRMFSYEHDR FHDR MF=(C,SRMF),EQUATES=NO                                     Standardheader
SRMFSYEHDR DS      0A
SRMFFHE DS      0XL8                0   GENERAL PARAMETER AREA HEADER
*
SRMFIFID DS      0A                0   INTERFACE IDENTIFIER
SRMFFCTU DS      AL2                0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*                               BIT 14-12 UNUSED, MUST BE RESET
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
SRMFFCT DS      AL1                2   FUNCTION NUMBER
SRMFFCTV DS      AL1                3   FUNCTION INTERFACE VERSION NUMBER
*
SRMFRET DS      0A                4   GENERAL RETURN CODE
SRMFSRET DS      0AL2               4   SUB RETURN CODE
SRMFSR2 DS      AL1                4   SUB RETURN CODE 2
SRMFSR1 DS      AL1                5   SUB RETURN CODE 1
SRMFMRET DS      0AL2               6   MAIN RETURN CODE
SRMFMR2 DS      AL1                6   MAIN RETURN CODE 2
SRMFMR1 DS      AL1                7   MAIN RETURN CODE 1
SRMFFHL EQU     8                8   GENERAL OPERAND LIST HEADER LENGTH
*
*   main return codes
SRMFSUCC EQU     0                                CMD accepted
SRMFREJE EQU     1                                CMD rejected
*
SRMFSYEPT DS     A                                POINTER
SRMF# EQU *-SRMFSYEHDR

```

3.6 ACCOUNTING-Exit (015)

Eine von der Systembetreuung anschließbare Accounting-Exit-Routine wird für jeden Abrechnungssatz aufgerufen, bevor dieser an die Accounting-Schreibtask übergeben und in die Abrechnungsdatei geschrieben wird.

Die Exit-Routine kann den Abrechnungssatz akzeptieren, ändern, unterdrücken oder zusätzliche Abrechnungssätze erzeugen, die vor oder hinter dem aktuellen Satz in die Abrechnungsdatei eingefügt werden.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(Parameterbereich NASXIT)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Die Rückkehrinformation an die rufende Komponente kann im Feld NASXRET des Parameterbereichs übergeben werden:

- Abrechnungssatz modifizieren (Returncode X'00')

Die Adresse des Original-Abrechnungssatzes wird im Feld NASXRECA des Parameterbereichs an die Exit-Routine übergeben. Die Routine kann beliebige Veränderungen des Satzes vornehmen, der Satz darf jedoch höchstens 496 Bytes lang sein. Das Layout der System-Abrechnungssätze ist im Handbuch „Einführung in die Systembetreuung“ [7] beschrieben.

- Abrechnungssatz unterdrücken (Returncode X'08')

Außer Setzen der Rückkehrinformation X'08' sind durch die Exit-Routine keine Aktionen erforderlich.

- Abrechnungssatz einfügen (Returncode X'00')

Die einzufügenden Sätze müssen von der Exit-Routine vor dem Rücksprung ins System mit dem Makroaufruf \$NAREC in die Abrechnungsdatei geschrieben werden. Die Satzkennungen kundenspezifischer Abrechnungssätze sollten mit den Buchstaben X, Y oder Z beginnen, um Namenskonflikte mit den Satzkennungen des Systems zu vermeiden.

- Abrechnungssatz anfügen (Returncode X'04')

Das System schreibt den Abrechnungssatz in die Abrechnungsdatei und gibt anschließend die Steuerung an die Exit-Routine zurück. Das Feld NASXRECA des Parameterbereichs enthält in diesem Fall keine Adresse. Nun können mit \$NAREC weitere Abrechnungssätze angeschlossen werden. Beim erneuten Rücksprung in das System wird der Returncode nicht mehr ausgewertet, da ein dritter Aufruf nicht vorgesehen ist.

- Schutz vor Rekursion

Die Accounting-Exit-Routine wird auch für Abrechnungssätze aufgerufen, die von der Routine selbst mit \$NAREC geschrieben werden. Eine mögliche Rekursion muss innerhalb der Exit-Routine verhindert werden.

DSECT

Der Makro NASXIT erzeugt das Layout des Exit-Parameterbereichs als Dsect oder als Datenbereich.

Der Exit-Parameterbereich enthält den Standardheader (mit Returncode) und das FeldNASRECA für die Adresse des Abrechnungssatzes, der in die Accounting-Datei eingetragen werden soll.

Die Rückkehrinformation der Exit-Routine muss in das Feld NASXRET („General Return Code“) im Parameterbereich eingetragen werden.

```

        NASXIT MF=D
        MFCHK MF=D,                                C
            SUPPORT=(C,D),                          C
            PREFIX=N,                                C
            MACID=ASX,                               C
            DMACID=ASX,                              C
            DNAME=ASXIT
NASXIT   DSECT ,
            * ,##### PREFIX=N, MACID=ASX #####
            #INTF INTNAME=$NASXIT,REFTYPE=REQUEST,INTCOMP=1
* PARAMETER LIST LAYOUT
NASXACC  FHDR MF=(C,NASX),EQUATES=NO
NASXACC  DS      0A
NASXFHE  DS      0XL8          0   GENERAL PARAMETER AREA HEADER
*
NASXIFID DS      0A          0   INTERFACE IDENTIFIER
NASXFCTU DS      AL2          0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*                               BIT 14-12 UNUSED, MUST BE RESET
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
NASXFCT  DS      AL1          2   FUNCTION NUMBER
NASXFCTV DS      AL1          3   FUNCTION INTERFACE VERSION NUMBER
*
NASXRET  DS      0A          4   GENERAL RETURN CODE
NASXSRET DS      0AL2         4   SUB RETURN CODE
NASXSR2  DS      AL1          4   SUB RETURN CODE 2
NASXSR1  DS      AL1          5   SUB RETURN CODE 1
NASXMRET DS      0AL2         6   MAIN RETURN CODE
NASXMR2  DS      AL1          6   MAIN RETURN CODE 2
NASXMR1  DS      AL1          7   MAIN RETURN CODE 1
NASXFHL  EQU     8           8   GENERAL OPERAND LIST HEADER LENGTH
*
*
* FOLLOWING RETURN CODES MAY BE SET :
*
*   SC2   SC1   MAIN           MEANING
*   00    00   0000           WRITE ACC RECORD AND DO NOT RECALL EXIT
*   00    00   0004           WRITE ACC RECORD AND RECALL EXIT
*   00    00   0008           DO NOT WRITE ACC RECORD
*
* EQUATES FOR SUBCODE1 :
*
NASXS1OK EQU  X'00'           NORMAL PROCESSING
*
* EQUATES FOR MAINCODE :
*
NASXMWR  EQU  X'00'           WRITE ACC RECORD
NASXMWRR EQU  X'04'           WRITE ACC RECORD AND RECALL
NASXMNWR EQU  X'08'           DO NOT WRITE THE RECORD
*
NASXRECA DS      A           ACC RECORD ADDRESS
*
* EQUATE FOR LENGTH OF PARAMETERLIST :
*
NASX#    EQU     *-NASXACC

```


3.7 PSWORD-Exit (020)

Eine PSWORD-Exit-Routine kann ein MODIFY-USER-PROTECTION-Kommando annehmen oder abweisen. Die zur Überprüfung notwendigen Daten werden in einer variablen Änderungsliste zur Verfügung gestellt. Die Adresse der Änderungsliste wird im Exit-Parameterbereich \$SRMSYE nach dem Standardheader im Feld SRMFSYEPT übergeben.

Zusätzliche Informationen über die betroffenen Benutzerkennungen oder andere Systemwerte können mit den entsprechenden Systemschnittstellen abgefragt werden.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(Parameterbereich $SRMSYE)
R2 ... R11 sind undefiniert.
R12 = A(TPR Program Manager)
R13 = A(Save Area)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Die Exit-Routine wird in zwei verschiedenen Formaten aufgerufen. Diese werden durch das Versionskennzeichen SRMFFCTV (siehe "DSECT") unterschieden.

SRMFFCTV=1

Mit diesem Format wird die Exit-Routine bei der Verarbeitung des Kommandos MODIFY-USER-PROTECTION aufgerufen, wenn SECOS ab Version V3.0 im Einsatz ist. In diesem Fall erfolgen zwei Aufrufe:

1. Eine Änderungsliste wird übergeben, anhand derer in der Exit-Routine entschieden werden kann, ob das Kommando ausgeführt werden soll oder nicht.

Die Änderungsliste hat folgendes Format:

Kennzeichen für Kennwortänderung	X'03'
----- unbenutzt -----	X'01'
neues Kennwort (unverschlüsselt) *	8 Byte
altes Kennwort *	8 Byte
Endekennzeichen	X'FF'
PVS-Kennzeichen	1 Byte
PVS-Id	4 Byte
Benutzerkennung	8 Byte
TID	4 Byte
RUN (1. oder 2. Aufruf)	1 Byte
RESULT(Aufrufergebnis)	1 Byte

*Das neue Kennwort wird stets unverschlüsselt an der Schnittstelle bereitgestellt. Das alte Kennwort ist gemäß der Class-2-Option ENCRYPT verschlüsselt.

Bei diesem Aufruf hat das Feld RUN den Wert 1 und das Feld RESULT den Wert 0.

Die Rückkehrinformation an die rufende Systemkomponente erfolgt über den Maincode1 des Parameterbereichs \$SRMSYE im rechten Byte des Feldes SRMFSYEHDR.

SRMFMR1=X'00'	Das Kommando soll ausgeführt werden.
SRMFMR1=X'01'	Das Kommando soll abgewiesen werden.

Danach finden im System weitere Überprüfungen statt, die entscheiden, ob das Kennwort geändert werden kann, z. B. Prüfung auf hinreichende Komplexität.

2. Im Feld RESULT der Änderungsliste wird übermittelt, ob das Kennwort erfolgreich geändert werden konnte oder nicht:

RESULT=1	Das Kennwort wurde erfolgreich geändert.
RESULT=2	Das Kennwort wurde nicht geändert.

Das Feld RUN hat den Wert 2.

Hat die Exit-Routine das MODIFY-USER-PROTECTION-Kommando abgewiesen, wird folgende Meldung ausgegeben:

```
% SRM2301 KOMMANDO WURDE DURCH EINE SYSTEM-EXIT-ROUTINE ABGEWIESEN
```

SRMFFCTV=0

Mit diesem Format wird die Exit-Routine bei der Verarbeitung des Kommandos MODIFY-USER-PROTECTION aufgerufen, wenn SECOS nicht eingesetzt wird oder eine SECOS-Version < V3.0 im Einsatz ist. Bei der Verarbeitung des ISP-Kommandos PSWORD wird die Exit-Routine ebenfalls in diesem Format aufgerufen, unabhängig davon ob und welche Version von SECOS eingesetzt wird.

Die Änderungsliste hat folgendes Format:

Kennzeichen für Kennwortänderung	X'03'
----- unbenutzt -----	X'01'
neues Kennwort (unverschlüsselt) *	8 Byte
altes Kennwort *	8 Byte
Endekennzeichen	X'FF'
PVS-Kennzeichen	1 Byte
PVS-Id	4 Byte
Benutzerkennung	8 Byte

*Das neue Kennwort wird stets unverschlüsselt an der Schnittstelle bereitgestellt. Das alte Kennwort ist gemäß der Class-2-Option ENCRYPT verschlüsselt.

Die Rückkehrinformation an die rufende Systemkomponente erfolgt über den Maincode1 des Parameterbereichs \$SRMSYE im rechten Byte des Feldes SRMFSYEHDR.

SRMFMR1=X'00'	Das Kommando soll ausgeführt werden.
SRMFMR1=X'01'	Das Kommando soll abgewiesen werden.

Hat die Exit-Routine das MODIFY-USER-PROTECTION-Kommando abgewiesen, wird folgende Meldung ausgegeben:

```
% SRM2301 KOMMANDO WURDE DURCH EINE SYSTEM-EXIT-ROUTINE ABGEWIESEN
```

DSECT

```

$SRMSYE MF=D
MFTST MF=D,PREFIX=S,MACID=RMF,ALIGN=F,                                C
    DMACID=RMF,SUPPORT=(D,C,M,L),DNAME=RMF_MDL
SRMF_MDL DSECT ,
    *,##### PREFIX=S, MACID=RMF #####
*   parameterarea description
SRMFSYEHDR FHDR MF=(C,SRMF),EQUATES=NO                               Standardheader
SRMFSYEHDR DS    0A
SRMFFHE DS    0XL8                0   GENERAL PARAMETER AREA HEADER
*
SRMFIFID DS    0A                0   INTERFACE IDENTIFIER
SRMFFCTU DS    AL2                0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*                               BIT 14-12 UNUSED, MUST BE RESET
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
SRMFFCT DS    AL1                2   FUNCTION NUMBER
SRMFFCTV DS    AL1                3   FUNCTION INTERFACE VERSION NUMBER
*
SRMFRET DS    0A                4   GENERAL RETURN CODE
SRMFSRET DS    0AL2              4   SUB RETURN CODE
SRMFSR2 DS    AL1                4   SUB RETURN CODE 2
SRMFSR1 DS    AL1                5   SUB RETURN CODE 1
SRMFMRET DS    0AL2              6   MAIN RETURN CODE
SRMFMR2 DS    AL1                6   MAIN RETURN CODE 2
SRMFMR1 DS    AL1                7   MAIN RETURN CODE 1
SRMFFHL EQU    8                8   GENERAL OPERAND LIST HEADER LENGTH
*
*   main return codes
SRMFSUCC EQU    0                CMD accepted
SRMFREJE EQU    1                CMD rejected
*
SRMFSYEPT DS    A                POINTER
SRMF# EQU    *-SRMFSYEHDR

```

3.8 SECURE-Exits (025/026)

Die Anwendung des Kommandos SECURE-RESOURCE-ALLOCATION kann an zwei Exit-Punkten von der Systembetreuung überwacht werden:

- Die Request Exit-Routine kann die SECURE-Anforderung abweisen.
- Die Return Exit-Routine kann nach der Bearbeitung der SECURE-Anforderung geeignete Aktionen zur Protokollierung bzw. Registrierung durchführen.

Return Exit (025)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(Parameterbereich NKSEPAR)
R12 = A(P2 Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Hinweise

- Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Das Feld NKSHMSG des NKSEPAR-Arbeitsbereiches enthält:
 - die Meldungsnummer (_NKSxxx), mit der das SECURE-RESOURCE-ALLOCATION-Kommando vom Kommandoprozessor abgewiesen wurde;
 - XL8'00' bei erfolgreicher Kommandobearbeitung.
- Das Feld NKSHPTR des NKSEPAR-Arbeitsbereiches enthält:
 - einen Zeiger auf das fehlerhafte Element der NKSEPAR-Datenliste oder A(0), wenn das fehlerhafte Element nicht eindeutig zu bestimmen ist;
 - A(0) bei erfolgreicher Kommandobearbeitung.

Request Exit (026)

Die Registerversorgung erfolgt wie beim Return Exit.

Die Rückkehrinformation an die rufende Systemkomponente erfolgt im Feld NKSHEXR des Parameterbereichs NKSEPAR.

NKSHEXRC=X'00'	/SECURE-RESOURCE-ALLOCATION erfolgreich.
NKSHEXRC=X'04'	/SECURE-RESOURCE-ALLOCATION von der Exit-Routine abgewiesen.

Hinweise

- Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Wird die SECURE-Anforderung von der Exit-Routine abgewiesen, erscheint die Meldung:
 % NKS0062 SECURE-ANFORDERUNG VOM SYSTEMVERWALTER ABGEWIESEN. GRUND: (&00)
 Ein Ablehnungsgrund wäre z.B., dass mehr als drei Geräte vom Typ 3590 reserviert werden sollen.

- Wird das SECURE-RESOURCE-ALLOCATION-Kommando von der Exit-Routine abgewiesen (NKSHEXRC = X'04'), kann das Feld NKSHPTR des NKSEPAR-Arbeitsbereichs von der Exit-Routine mit der Anfangsadresse des fehlerhaften Elements in der NKSEPAR-Datenliste versorgt werden. Der Benutzer erhält dann weitere Informationen in der Meldung NKS0062 (s.o.).

DSECT

```

NKSEPAR
MFCHK DNAME=KSHEAD,MF=D,MACID=KSH,ALIGN=F,PREFIX=N,          200C
      DMACID=KSH                                             200
NKSHHEAD DSECT ,
          *,##### PREFIX=N, MACID=KSH #####
*****
**      SYSTEM-EXIT FOR SECURE : HEADER DESCRIPTION      **
*****
*--
          1 SECURE_EXIT_HEADER,
NKSHEXRT DS      0F          2 EXIT_RETURN_INFO          200
          DS      XL3          FILLER                    200
NKSHEXRC DS      X          EXIT_RETURN_CODE            200
NKSHEXOK EQU     X'00'      SUCCESS                     200
NKSHEXSR EQU     X'04'      SEC_RES_REJECTED            200
NKSHMSG  DS      XL8          2 RETURN_EXIT_MSG_IDENTIFIER
*--
          (SEC_SUCCESS = XL8'00',
*--
          SEC_NO_SUCCESS = ' NKS...'),
NKSHPTR  DS      A          2 SECURE_EXIT_ERROR_ELEMENT
*--
          (DEFINED = A(BEGIN_OF_ERROR_ELEM,
*--
          NOT_DEFINED = A(0));
*--
          2 POINTERS_AND_COUNTERS_OF_ELEMENTS
*--
          (UNDEFINED=POINTER_AND_COUNTER_
*--
          _ARE_SET_TO_ZERO)
NKSHFILE DS      A          3 FILE_LIST_POINTER,
NKSHFIL# DS      H          3 FILE_ELEM_COUNTER,
NKSHVOL  DS      A          3 VOLUME_LIST_POINTER,
NKSHVOL# DS      H          3 VOLUME_ELEM_COUNTER,
NKSHDEV  DS      A          3 DEVICE_LIST_POINTER,
NKSHDEV# DS      H          3 DEVICE_ELEM_COUNTER,
NKSHUNIT DS      A          3 UNIT_LIST_POINTER,
NKSHUNI# DS      H          3 UNIT_ELEM_COUNTER,
NKSHDEVL DS      A          3 DEVICE_AT_LOCATION_LIST_POINTER 102
NKSHDE#L DS      H          3 DEVICE_AT_LOCATION_ELEM_COUNTER 102
          DS      0F
****      ELEMENT_DESCRIPTION SEE BELOW      ****
NKSHLEN  EQU     *-NKSHEXRT      LENGTH_OF_HEADER;

```

```

SPACE 2
MFCHK DNAME=KSELEM,MF=D,MACID=KSE,ALIGN=F,PREFIX=N,      200C
      DMACID=KSE                                          200
NKSELEM DSECT ,
      *,##### PREFIX=N, MACID=KSE #####
*****
**      SYSTEM-EXIT FOR SECURE : ELEMENT DESCRIPTION      **
*****
*--
      1 SECURE_EXIT_ELEMENTS ALIGN FW,
NKSETYP DS XL1      2 TYPE_OF_ELEMENT SET
NKSEFILE EQU X'01'      (FILE = 1,
NKSEVOL EQU X'02'      VOLUME = 2,
NKSEDEV EQU X'03'      DEVICE = 3,
NKSEUNIT EQU X'04'      UNIT = 4,
NKSEDEVL EQU X'05'      DEVICE AT LOCATION = 5),      102
      SPACE 1
NKSEDESC DS 0C      2 ELEMENTS_DESCRIPTION,
*** LAYOUT FOR FILE ELEMENT *** 3 FILE_ELEMENT,
NKSEFNAM DS CL54      4 FILE_NAME,
NKSEFTYP DS XL1      4 FILE_DEVICE_TYPE SET
NKSEFTDI EQU X'01'      (DEVICE_TYPE_DISK = 1,
NKSEFTTA EQU X'02'      DEVICE_TYPE_TAPE = 2),
NKSEFRES DS XL1      4 FILE_RESERVATION_TYPE SET
NKSEFRSH EQU X'00'      (SHAREABLE = 0,
NKSEFREX EQU X'01'      EXCLUSIVE = 1),
NKSEFACC DS XL1      4 FILE_ACCESS_IF_TAPE SET
NKSEFARD EQU X'00'      (READ = 0,
NKSEFAWR EQU X'01'      WRITE = 1),
*
NKSEFLEN EQU *-NKSETYP      LENGTH_OF_ONE_FILE_ELEMENT
      SPACE 2
      ORG NKSEDESC
      SPACE 1
*** LAYOUT FOR VOLUME ELEMENT *** 3 VOLUME_ELEMENT DEFINED FILE_ELEMENT,
NKSEVSN DS CL6      4 VOLUME_SERIAL_NUMBER,
NKSEVDT DS XL2      4 VOLUME_DEVICE_TYPE_CODE,
NKSEVTYP DS XL1      4 VOLUME_DEVICE_TYPE SET
NKSEVTDI EQU X'01'      (DEVICE_TYPE_DISK = 1,
NKSEVTTA EQU X'02'      DEVICE_TYPE_TAPE = 2),
NKSEVRSH DS XL1      4 VOLUME_RESERVATION_TYPE SET
NKSEVRSH EQU X'00'      (SHAREABLE = 0,
NKSEVREX EQU X'01'      EXCLUSIVE = 1),
NKSEVACC DS XL1      4 VOLUME_ACCESS_IF_TAPE SET
NKSEVARD EQU X'00'      (READ = 0,
NKSEVAWR EQU X'01'      WRITE = 1),
*
NKSEVLEN EQU *-NKSETYP      LENGTH_OF_ONE_VOLUME_ELEMENT
      SPACE 2

```

```

        ORG  NKSEDESC
        SPACE 1
*** LAYOUT FOR DEVICE ELEMENT *** 3 DEVICE_ELEMENT DEFINED FILE_ELEMENT,
NKSEDTC DS  XL2                4 DEVICE_TYPE_CODE,
NKSEDEV# DS  H                  4 NUMBER_OF_DEVICES,
*
NKSEDLEN EQU  *-NKSETYP          LENGTH_OF_ONE_DEVICE_ELEMENT
        SPACE 2
        ORG  NKSEDESC
        SPACE 1
*** LAYOUT FOR UNIT ELEMENT *** 3 UNIT_ELEMENT DEFINED FILE_ELEMENT,
NKSEUMN DS  CL4                4 UNIT_MNEMONIC,          MNE
*
NKSEULEN EQU  *-NKSETYP          LENGTH_OF_ONE_UNIT_ELEMENT
        SPACE 2
        ORG  NKSEDESC
        SPACE 1
*** LAYOUT FOR DEVICE_AT_LOCATION ELEMENT ***
NKSEDTCL DS  XL2                4 TYPE-CODE OF DEVICE_AT_LOCATION
NKSEDE#L DS  H                  4 NUMBER OF DEVICE_AT_LOCATION
NKSEDLOC DS  CL8                4 DEVICE-LOCATION
*
NKSELLEN EQU  *-NKSETYP
        SPACE 2
        ORG

```

Die übergebene Elementliste entspricht nicht unmittelbar den im SECURE-RESOURCE-ALLOCATION-Kommando angegebenen Parametern.

- Die VOLUME_ELEMENT-Teilliste enthält neben den explizit angegebenen DISK-/TAPE-Anforderungen auch die durch FILE spezifizierten implizit benötigten Volumes, die aus den Katalogeinträgen ergänzt wurden.
- Die DEVICE_AT_LOCATION-Teilliste ist eine rein additive verfeinerte Aufschlüsselung der DEVICE-Teilliste. Sie enthält die Device-Anforderungen, für die eine Lagerortzuordnung existiert. Diese Zuordnung kann entstanden sein durch:
 - die Angabe des Anwenders (LOCATION=<loc>)
 - die Default-Location bei *USER-DEFAULT für MAREN
 - eine im MAREN-EXIT6 ermittelte Location bei Angabe von LOCATION=*BY-FILE

3.9 LOGON-Exit (030)

Eine LOGON-Exit-Routine wird, falls vorhanden, in folgenden Fällen aufgerufen:

- Im Dialogbetrieb vor der Initialisierung der Task, wenn der Benutzer eine unzulässige Benutzerkennung, Abrechnungsnummer oder ein falsches LOGON-Kennwort eingegeben hat.
Die Exit-Routine kann z. B. feststellen, von welcher Datenstation das LOGON abgeschickt wurde.
- Nach Bearbeitung eines Stapel- oder Dialog-LOGON.
Die Exit-Routine kann veranlassen, dass z.B. die Task beendet/abgebrochen wird oder Systemverwalterprozeduren aufgerufen werden.
- Bei einem ENTER-JOB- oder ENTER-PROCEDURE-Kommando eines Benutzers (nicht beim Operator-ENTER!), wenn falsche Benutzerkennung, Abrechnungsnummer oder falsches Kennwort angegeben wurden.
Die Exit-Routine kann z.B. feststellen, von welcher Datenstation das ENTER-JOB- oder ENTER-PROCEDURE-Kommando im Dialog gegeben wurde.
- Nach erfolgreicher Bearbeitung eines ENTER-JOB- oder ENTER-PROCEDURE-Kommandos.
Die Exit-Routine kann z.B. die Aufruferdaten mit den ENTER-Daten vergleichen und falls nötig, den ENTER-Auftrag abweisen

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX030-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Rückkehrinformation an die rufende Systemkomponente wird im Parameterbereich übergeben.

Ein vom Betriebssystem bereits akzeptiertes LOGON- bzw. ENTER-JOB-Kommando kann von der Exit-Routine mit Returncode X'01' noch zurückgewiesen werden. Der Returncode sollte mit der Form MF=M des EX030-Makros übergeben werden:

```
EX030 MF=M,PARAM=... ,RC=*ACCEPT/*REJECT
```

Der Benutzer erhält dann folgende Meldung:

```
% JMS0152 ZUGANG ZUM SYSTEM DURCH 'LOGON-EXIT-ROUTINE' ABGEWIESEN
```

Hinweise

- Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Wird ein SET-LOGON-PARAMETERS, ENTER-PROCEDURE oder ENTER-JOB-Kommando mit falschen Benutzerdaten abgesetzt (JMSXLEJS=JMSXLEIU,JMSXLEIP,JMSXLEIA), so hat der Returncode keinen Einfluss mehr; das Kommando wird in jedem Fall mit der entsprechenden Fehlermeldung abgewiesen.
- Bei ENTER-Aufträgen wird die LOGON-Verarbeitung, die zum Aufruf der Exit-Routine führt, nach der SPOOLIN-Phase durchgeführt.
- Es wird empfohlen in LOGON-Exit-Routinen keinen Dialog zu führen.

DSECT

```

        EX030 MF=D
        MFTST MF=D,PREFIX=J,MACID=MSX,ALIGN=F,                C
            DMACID=MSX,SUPPORT=(E,D,C,M,L),DNAME=MSXLE
JMSXLE  DSECT ,
        *,##### PREFIX=J, MACID=MSX #####
*   exit parameterarea description
JMSXLER0 DS    XL6                reserved
JMSXLERC DS    H                  (main) return code
*   decision of the exit routine
JMSXLEAC EQU   0                  accept job
JMSXLERJ EQU   1                  reject job
*
JMSXLEJP DS    A                  pointer to JTBP
JMSXLEJS DS    FL1                job state
*   describes the acceptance state of the job at the time the exit is
*   called
JMSXLEJA EQU   0                  job accepted
JMSXLEIU EQU   4                  job rejected (invalid user
*                                  id)
JMSXLEIP EQU   8                  job rejected (invalid
*                                  password)
JMSXLEIA EQU   12                 job rejected (invalid account
*
JMSXLECL DS    FL1                instance calling the exit
*   describes which instance calls the exit
JMSXLEDL EQU   0                  dialog logon
JMSXLEBL EQU   4                  batch logon
JMSXLEEV EQU   8                  enter validation
JMSXLERL EQU   12                 rlogin
*
JMSXLER2 DS    XL2                reserved
JMSX#    EQU   *-JMSXLER0

```

Das Feld JMSXLEJP enthält die Adresse des JTBP eines Auftrags, der gerade definiert oder gestartet werden soll. Änderungen in dieser Datenstruktur sind verboten. Eine DSECT des JTBP erhält man mit dem Makroaufruf \$JTBP.

Nur bei Aktivierung der Exit-Routine während der Initialisierung einer Batch-Task zur Ausführung eines Stapelauftrags (JMSXLECL=JMSXLEBL) ist der JTBP vollständig.

Für einen akzeptierten Enterjob (JMSXLECL=JMSXLEEV) ist die TSN im JTBP noch nicht festgelegt. Das LOGON-Kennwort ist nur versorgt, wenn die Exit-Routine nach negativer Prüfung der Zutrittsberechtigung aktiviert wird (JMSXLEJS=JMSXLEIU,JMSXLEIP,JMSXLEIA).

Bei Aktivierung der Exit-Routine während der Initialisierung einer akzeptierten Dialog- oder Batch-Task (JMSXLEJS=JMSXLEJA, JMSXLECL=JMSXLEDL, JMSXLEBL), können die wesentlichen Daten über die anlaufende Task mit dem Makro JOBINFO besorgt werden (Handbuch „Makroaufrufe an den Ablaufteil“ [9]).

3.10 Jobparameter-Exit (032)

Dieser Exit ermöglicht die Verarbeitung so genannter Jobparameter, die die Systembetreuung bei der Definition von Jobklassen festlegen kann (siehe dazu Anweisung DEFINE-JOB-CLASS im Dienstprogramm JMU, Handbuch „Dienstprogramme“ [4]).

Ein Jobparameter ist eine bis zu 127 Zeichen lange Folge in beliebiger Syntax, die in einem der Kommandos ENTER-PROCEDURE, ENTER-JOB, MODIFY-JOB oder SET-LOGON-PARAMETERS als Operand angegeben werden kann.

Die Exit-Routine wird nur bei akzeptiertem SET-LOGON-PARAMETERS-Kommando im Dialogbetrieb bzw. bei den Kommandos ENTER-JOB, ENTER-PROCEDURE und MODIFY-JOB aufgerufen. Bei ENTER-JOB, ENTER-PROCEDURE oder MODIFY-JOB auf nicht gestartete Stapelaufträge werden die Jobparameter auch an den TU-Scheduler übergeben, der sie ebenfalls auswerten kann.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX032-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Die Rückkehrinformation sollte mit der Form MF=M des EX032-Makros an die rufende Systemkomponente übergeben werden.

Hinweis

Der Jobparameter-Exit wird vor einer eventuell vorhandenen LOGON-Exit-Routine aufgerufen. Zu diesem Zeitpunkt stehen die Informationen in den Systemtabellen noch nicht vollständig zur Verfügung (z.B. TSN im JTBP).

DSECT

```

        EX032 MF=D
        MFTST MF=D,PREFIX=J,MACID=MSX,ALIGN=F,                C
            DMACID=MSX,SUPPORT=(E,D,C,M,L),DNAME=MSXJE
JMSXJE  DSECT ,
        *,##### PREFIX=J, MACID=MSX #####
*   parameterarea description
JMSXJER0 DS    XL6                      reserved
JMSXJERC DS    H                        (main) return code
*   decision of the exit routine
JMSXJEAC EQU   0                        accept job
JMSXJERJ EQU   1                        reject job
*
JMSXJEJ  DS    0XL128
JMSXJEJP DS    CL128                    job parameter data
        ORG    JMSXJEJ
*
JMSXJEPD DS    0XL128
JMSXJEPL DS    X                        length of job parameters
JMSXJEPS DS    CL127                    job parameters
*
        ORG    JMSXJEJ+128
JMSXJEJC DS    0XL128
JMSXJECP DS    CL128                    job class parameter data
        ORG    JMSXJEJC
*
JMSXJECD DS    0XL128
JMSXJECL DS    X                        length of job class
*                                           parameters
JMSXJECS DS    CL127                    job class parameters
*
        ORG    JMSXJEJC+128
JMSX#    EQU   *-JMSXJER0

```

3.11 JV-Exit (033)

Die Exit-Routine wird aufgerufen, wenn eine vom System nicht angebotene Sonderjobvariable angesprochen wird. Der Zugriff kann durch die Kommandos SHOW-JV und MODIFY-JV oder durch die Makros GETJV und SETJV erfolgen. In der Exit-Routine können Werte für die Sonderjobvariable übergeben werden, die der Anwender des Kommandos erhalten soll. Auf diese Weise können eigene Sonderjobvariablen realisiert werden.

Die Adresse des Parameterbereichs EX033 wird in Register 1 übergeben. Das FeldJVSXJV des Parameterbereichs enthält den Namen der vom Benutzer gewünschten Sonderjobvariable.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX033-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Rückkehrinformationen

Die Rückinformation an die rufende Systemkomponente kann im Parameterbereich übergeben werden. Im Feld JVSXLEN kann die Länge der Information (1-256) und im Feld JVSXVAL der Inhalt der Sonderjobvariablen eingetragen werden. Der Maincode im Standardheader muss auf X'0000' und der Subcode1 auf X'00' gesetzt sein.

Die Meldung

```
% JVS0472 NAME DER SONDER-JV UNZULAESSIG. KOMMANDO KORRIGIEREN
```

wird ausgegeben, wenn keine Exit-Routine angemeldet, der Maincode ungleich X'0000' und/oder der Subcode ungleich X'00' ist.

DSECT

```

EX033 MF=D
MFTST MF=D,PREFIX=J,MACID=VSX,ALIGN=F,                                C
      DMACID=VSX,SUPPORT=(D,C,M,L),DNAME=VSXPL
JVSXPL DSECT ,
      *,##### PREFIX=J, MACID=VSX #####
JVSXMIN EQU 1                                MIN. LENGTH
*
JVSXMAX EQU 256                              MAX. LENGTH
*
*   end parameterarea
JVSXHDR FHDR MF=(C,JVSX),EQUATES=NO                                STANDARDHEADER
JVSXHDR DS 0A
JVSXFHE DS 0XL8                                0   GENERAL PARAMETER AREA HEADER
*
JVSXIFID DS 0A                                0   INTERFACE IDENTIFIER
JVSXFCTU DS AL2                                0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*
*                               BIT 14-12 UNUSED, MUST BE RESET
*
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
JVSXFCT DS AL1                                2   FUNCTION NUMBER
JVSXFCTV DS AL1                                3   FUNCTION INTERFACE VERSION NUMBER
*
JVSXRET DS 0A                                4   GENERAL RETURN CODE
JVSXSRET DS 0AL2                              4   SUB RETURN CODE
JVSXSR2 DS AL1                                4   SUB RETURN CODE 2
JVSXSR1 DS AL1                                5   SUB RETURN CODE 1
JVSXMRET DS 0AL2                              6   MAIN RETURN CODE
JVSXMR2 DS AL1                                6   MAIN RETURN CODE 2
JVSXMR1 DS AL1                                7   MAIN RETURN CODE 1
JVSXFHL EQU 8                                8   GENERAL OPERAND LIST HEADER LENGTH
*
*   main return codes
JVSXSUCC EQU 0                                NO ERROR DETECTED
JVSXREJE EQU 1                                INVALID SPECIAL JV
*
JVSXJV DS CL54                                NAME OF THE SPECIAL JV
JVSXLEN DS H                                  RETURNED LENGTH OF JV-VALUE
JVSXUNU DS XL2                                UNUSED
JVSXVAL DS CL256                              RETURNED VALUE OF THE SPECIAL
*
*                               JV
JVSXUNU1 DS XL2                                UNUSED
JVSX# EQU *-JVSXHDR

```

3.12 BCAM-Exit (036)

Dieser Exit ermöglicht die Kontrolle von Verbindungsaufbauwünschen zu Systemanwendungen (z.B. \$DIALOG). Die Exit-Routine wird aufgerufen, wenn BCAM erkennt, dass ein lokaler oder ferner Partner einen Verbindungsaufbau zu einer Systemanwendung wünscht.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A($SYSX-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Der Parameterbereich hat folgendes Format:

fun	typ1	id1	typ2	id2
0	1	2		

fun =	Funktion der Exitroutine (Inhalt des Feldes: X'01')
typ =	Art der Funktion, zunächst existiert nur ein Typ mit der Verschlüsselung X'01'. X'01' bedeutet Überwachen der Verbindung zu Systemanwendungen.
id =	Identifikation der Anwendung. id1 ist die gerufene, id2 die rufende Anwendung.

Geliefert werden Anwendungsname/Rechnername in folgender Struktur:

01	01	called appl	own proc	01	calling appl	partner proc
00	01	02	0A	12	13	1B 22

called appl:	Name der gerufenen Systemanwendung
own proc:	Name des eigenen Rechners
calling appl:	Name der rufenden Anwendung
partner proc:	Name des Partnerrechners, in dem die rufende Anwendung ist

i **Warnung!**

In der Exitroutine sollte nur ein Namensabgleich gemacht werden. Vor dem Zugriff auf die ID's (Namen) muss unbedingt der Typ abgeprüft werden, da sich mit einem Typ das Layout (insbesondere die Länge) der ID verändern kann. Eine fehlerhafte Exitroutine kann zum Absturz des Kommunikationssystems (DCM) führen.

Rückkehrinformation an die rufende Systemkomponente im \$SYSX-Parameterbereich.

3.13 OPEN-Exits (039/040/041)

Mithilfe von drei Exitpunkten kann die Anwendung der Systemfunktion „OPEN = Datei Eröffnen“ gesteuert und überwacht werden. Dabei ist es gleichgültig, ob der OPEN aus einem Programm des Benutzers, einem Dienstprogramm (z.B. LMS, EDT) oder der Bearbeitung eines Systemkommados (z.B. START-PROGRAM) stammt.

- Am Security-Exit (039) kann die Systembetreuung den OPEN-Aufruf zurückweisen, bevor er ausgeführt ist.
- Am Request Exit (041) kann die Systembetreuung den OPEN-Aufruf zurückweisen, bevor er ausgeführt wird. Die zu öffnende Datei ist entweder aus der Task File Table oder dem FCB zu bestimmen.
- Am Return Exit (040) ist die Datei bereits eröffnet. Die Exit-Routine kann das Eröffnen nicht mehr verhindern oder rückgängig machen. Alle Funktionen des Return Exits können auch vom Security-Exit übernommen werden.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A($DOCSYS-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Hinweise

- Die Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Die Exits 039/040 können in Abhängigkeit vom Wert des AUDIT-Anzeigers im Katalogeintrag Überwachungsmaßnahmen durchführen (siehe „[Überwachung von Dateizugriffen](#)“).
- Wenn die Exits 039/040 angesprungen werden, hat das Datenverwaltungssystem von BS2000 die Datei bereits eröffnet. In den Exits 039 und 041 ist der Katalogeintrag für die eröffnete Datei noch gesperrt. Falls die Exit-Routine auf Grund eines Programmierfehlers „hängt“ oder „schleift“, ohne sich fehlerhaft zu beenden (dies hätte Abnormal Task Termination und Freigabe der Datei zur Folge), bleibt die Datei bis zum Ende des Systemlaufs gesperrt.
- Im Exit 040 ist der Katalogeintrag für die eröffnete Datei bereits freigegeben.
- Die Exit-Routine darf selbst wieder Dateien eröffnen! Probleme, die sich aus einer solchen rekursiven Aufrufverschachtelung ergeben, müssen von der Exit-Routine selbst abgefangen werden.
- Für Banddateien kann unter der Benutzerkennung TSOS mithilfe dieser Exits über entsprechende Exit-Routinen die Verwendung von Bändern kontrolliert/registriert und somit eine eigene Bandverwaltung realisiert werden.
- Wird der OPEN abgewiesen, erhält der Benutzer die Meldung DMS0D10 auf SYSOUT, bzw. den Fehlercode X'0D10' im Feld ID1ECB des FCB:

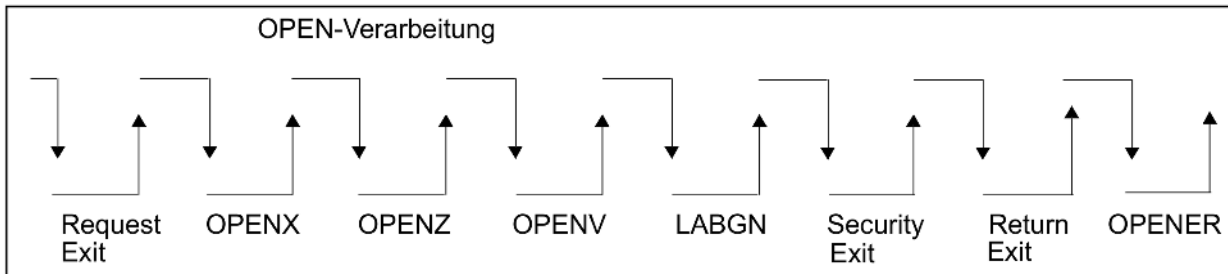
```
% DMS0D10 BENUTZUNG DES OPEN-MAKROS WURDE VON DEM SYSTEMADMINISTRATOR
EINGESCHRAENKT.
Die aktivierte System-EXIT-Routine hat den OPEN abgewiesen.
Das Programm wird mit dem EXLST-Ausgang 'OPENER' fortgesetzt, wenn dieser
definiert ist, ansonsten abgebrochen.
```

- Bei einem Fehler des Exit-Basismechanismus, wird am Bedienplatz die Meldung DMS0D15 ausgegeben. Danach geht die Verarbeitung weiter.

```
% DMS0D15 AUFRUF DES BASISMECHANISMUS FUER DEN SYSTEM EXIT '(&00)'  
LIEFERT FEHLERCODE '(&01)'  
(&00): EXIT-Nummer.  
(&01): Fehlercode der $SYSX-Schnittstelle.
```

Exit-Routinen und EXLST-Ausgänge

Die obenbeschriebenen Exit-Routinen und die bei der OPEN-Verarbeitung anschließbaren EXLST-Ausgänge werden, falls vorhanden, in der nachfolgend gezeigten Reihenfolge aktiviert.



Parameterbereich

Das Layout des Exit-Parameterbereichs kann mit dem Makro \$DOCSYS generiert werden.

```

        $DOCSYS MF=D
        MFTST MF=D,PREFIX=D,MACID=OCS,ALIGN=F,                                C
            DMACID=OCS,SUPPORT=(D,C,M,L),DNAME=OCS_MDL
DOCS_MDL DSECT ,
        *,##### PREFIX=D, MACID=OCS #####
*   parameterarea description
DOCSHDR  FHDR  MF=(C,DOCS),EQUATES=NO                                     Standardheader
DOCSHDR  DS    0A
DOCSFHE  DS    0XL8                0   GENERAL PARAMETER AREA HEADER
*
DOCSIFID DS    0A                0   INTERFACE IDENTIFIER
DOCSFCTU DS    AL2                0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*
*                               BIT 14-12 UNUSED, MUST BE RESET
*
*                               BIT 11-0  REAL FUNCTION UNIT NUMBER
DOCSFCT  DS    AL1                2   FUNCTION NUMBER
DOCSFCTV DS    AL1                3   FUNCTION INTERFACE VERSION NUMBER
*
DOCSRET  DS    0A                4   GENERAL RETURN CODE
DOCSSRET DS    0AL2              4   SUB RETURN CODE
DOCSSR2  DS    AL1                4   SUB RETURN CODE 2
DOCSSR1  DS    AL1                5   SUB RETURN CODE 1
DOCSMRET DS    0AL2              6   MAIN RETURN CODE
DOCSMR2  DS    AL1                6   MAIN RETURN CODE 2
DOCSMR1  DS    AL1                7   MAIN RETURN CODE 1
DOCSFHL  EQU   8                8   GENERAL OPERAND LIST HEADER LENGTH
*
*   main return codes
DOCSNOER EQU   0                  no error detected
DOCSRREJ EQU   4                  request rejected
*
DOCSFCB@ DS    A                  address of P1-FCB
DOCS#    EQU   *-DOCSHDR

```

Security Exit (039)

Die Rückinformation an den aufrufenden Systemmodul wird im Standard-Header des \$DOCSYS-Parameterbereichs im Feld DOCSMRET übergeben.

Returncode: DOCSNOER = X'0000' = kein Fehler

Returncode: DOCSRREJ = X'0004' = OPEN soll abgewiesen werden

Der Security-Exit wird nicht angesprungen, wenn ein OPEN wegen eines ungültigen Dateinamens abgewiesen wird.

Return Exit (040)

Der Return Exit wird auch dann angesprungen, wenn ein OPEN wegen eines ungültigen Dateinamens abgewiesen wird.

Die Exit-Routine kann keine Rückkehrinformation an den Systemmodul senden.

Request Exit (041)

Die Rückkehrinformation an den aufrufenden Systemmodul wird im Standard-Header des \$DOCSYS-Parameterbereichs im Feld DOCSMRET übergeben.

Returncode: DOCSNOER = X'0000' = kein Fehler

Returncode: DOCSRREJ = X'0004' = OPEN soll abgewiesen werden

3.14 CLOSE-Exits (042/043)

Für die Systemfunktion „CLOSE = Dateien schließen“ werden zwei Exits angeboten. Für Banddateien kann unter der Benutzerkennung TSOS mithilfe dieser Exits über entsprechende Exit-Routinen die Verwendung von Bändern kontrolliert/registriert und somit eine eigene Bandverwaltung realisiert werden.

- Der Request Exit wird zu Beginn der CLOSE-Verarbeitung angestoßen. Bei Banddateien ist das Band hinter dem letzten verarbeiteten Datenblock positioniert, Dateiendekennsätze sind noch nicht bearbeitet.
- Die Return Exit-Routine wird am Ende der CLOSE-Verarbeitung aufgerufen. Bei Banddateien ist das Band zwischen den beiden Abschnittsmarken positioniert, die das Dateiende kennzeichnen; bei Bändern mit Standardkennsätzen liegt diese Stelle hinter den EOF-Kennsätzen.

In beiden Fällen wird die Adresse des Exit-Parameterbereichs übergeben. Damit ist es möglich, auf weitere Informationen aus dem P1-FCB, P2-FCB oder der CVT (Current Volume Table) zuzugreifen.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A($DOCSYS-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Hinweise

- Die Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Ist der Returncode im Parameterbereich des Exit-Basismechanismus $\neq 0$, wird auf dem Bedienplatz die Meldung DMS0D15 ausgegeben. Danach geht die Verarbeitung weiter.

```
% DMS0D15 AUFRUF DES BASISMECHANISMUS FUER DEN SYSTEM EXIT '(&00)'
LIEFERT FEHLERCODE '(&01)'
(&00): EXIT-Nummer.
(&01): Fehlercode der $SYSX-Schnittstelle.
```

Informationsquellen

Die folgende Tabelle enthält Hinweise zum Zugriff auf Informationen, die sinnvoll von den Exit-Routinen verarbeitet werden können:

Information	DSECT	I/A	Name	steht in	Bemerkung
Exit-Parameterliste	\$DOCSYS	A		Register 1	
P1-FCB	IDFCB	A	P1FCB-ADDRESS	Exit-Parameterliste	
P2-FCB	IDFC2	A	D1P2LNK	P1-FCB	
CVT	DMACVT	A	D2CVTAD	P2-FCB	
VSN	DMADR	I	DRVVSN	TFT	mit RDTFT-Makro

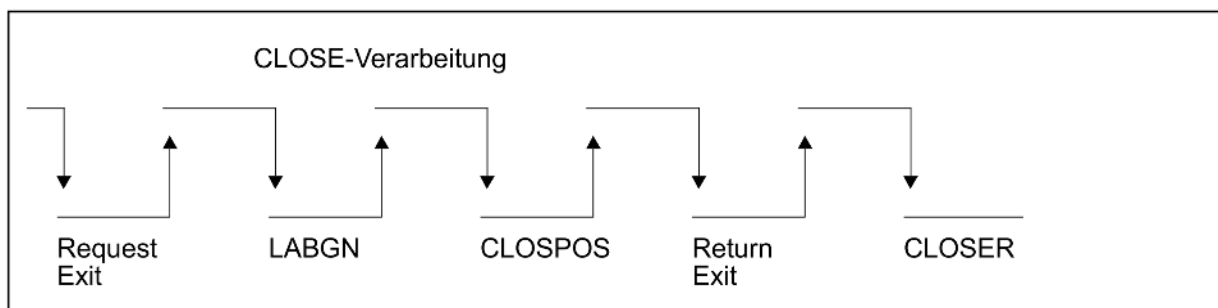
G Geräte-Zugriffsrecht	DMACVT	I	DCVRID	CVT	steht bei CLOSE Return Exit nicht zur Verfügung
Kennsätze	DTVOL	A	D2LGINF	P2-FCB	Return Exit
	DTHT1 DTHT2 DTHT3	A	D2LGINF	P2-FCB	Request Exit
Blockzähler für Datei gesamt		I	D2THPGN	P2-FCB	nur bei SAM und CLOSE Return Exit
Blockzähler für Band		I I	DIHPGNR D1BLK# / DIBLK#	P2-FCB P1-FCB / P2-FCB	BLKSIZE=STD BLKSIZE (STD,n) (alle nur bei SAM)

I = Information ist Inhalt von Feld „Name“

A= Adresse von Information in Feld „Name“

Exit-Routinen und EXLST-Ausgänge

Die obenbeschriebenen Exit-Routinen und die bei der CLOSE-Verarbeitung anschließbaren EXLST-Ausgänge werden, falls vorhanden, in der nachfolgend gezeigten Reihenfolge aktiviert.



Return Exit (042)

Die Exit-Routine kann keine Rückkehrinformation an den Systemmodul senden.

Request Exit (043)

Die Exit-Routine kann keine Rückkehrinformation an den Systemmodul senden.

Bei Ausgabedateien auf Band kann über die Adresse im Feld D2LGINF im P2-FCB auf die vom OPEN geschriebenen Kennsätze VOL1, HDR1, HDR2 und HDR3 zugegriffen werden. Die Länge des adressierten Bereichs ist $4 \times 80 = 320$ Bytes.

Der Blockzähler ist zum Aufrufzeitpunkt der Request Exit-Routine noch nicht versorgt.

Parameterbereich

Das Layout des Exit-Parameterbereichs kann mit dem Makro \$DOCSYS generiert werden.

```

    $DOCSYS MF=D
    MFTST MF=D,PREFIX=D,MACID=OCS,ALIGN=F,
    DMACID=OCS,SUPPORT=(D,C,M,L),DNAME=OCS_MDL
DOCS_MDL DSECT ,
    *,##### PREFIX=D, MACID=OCS #####
*   parameterarea description
DOCSHDR  FHDR  MF=(C,DOCS),EQUATES=NO           Standardheader
DOCSHDR  DS    0A
DOCSFHE  DS    0XL8                0   GENERAL PARAMETER AREA HEADER
*
DOCSIFID DS    0A                0   INTERFACE IDENTIFIER
DOCSFCTU DS    AL2                0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*                               BIT 14-12 UNUSED, MUST BE RESET
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
DOCSFCT  DS    AL1                2   FUNCTION NUMBER
DOCSFCTV DS    AL1                3   FUNCTION INTERFACE VERSION NUMBER
*
DOCSRET  DS    0A                4   GENERAL RETURN CODE
DOCSSRET DS    0AL2               4   SUB RETURN CODE
DOCSSR2  DS    AL1                4   SUB RETURN CODE 2
DOCSSR1  DS    AL1                5   SUB RETURN CODE 1
DOCSMRET DS    0AL2               6   MAIN RETURN CODE
DOCSMR2  DS    AL1                6   MAIN RETURN CODE 2
DOCSMR1  DS    AL1                7   MAIN RETURN CODE 1
DOCSFHL  EQU    8                8   GENERAL OPERAND LIST HEADER LENGTH
*
*   main return codes
DOCSNOER EQU    0                  no error detected
DOCSRREJ EQU    4                  request rejected
*
DOCSFCB@ DS    A                   address of P1-FCB
DOCS#    EQU    *-DOCSHDR

```

3.15 Spulenwechsel-Exits (044/045)

In Ergänzung der CLOSE-Exits (s.o. 042/043) werden zwei Exits für Eingriffe beim Spulenwechsel von Banddateien angeboten.

- Die Request-Exit-Routine wird zu Beginn jedes Spulenwechsels angestoßen, d.h. bei der Endbehandlung des abzuschließenden Bandes.
- Die Return-Exit-Routine wird nach der Anfangsbehandlung des neuen Bandes aufgerufen.

Wie bei den CLOSE-Exits für Banddateien wird in beiden Fällen die Adresse des Exit-Parameterbereichs übergeben, die die Adresse des P1FCB enthält.

Für Banddateien kann unter der Benutzerkennung TSOS mithilfe dieser Exits über entsprechende Exit-Routinen die Verwendung von Bändern kontrolliert/registriert und somit eine eigene Bandverwaltung realisiert werden.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A($DOCSYS-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Hinweise

- Die Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Bei einem Fehler des Exit-Basismechanismus, wird auf dem Bedienplatz die Meldung DMS0E15 ausgegeben. Danach geht die Verarbeitung weiter.

```
% DMS0E15 AUFRUF DES BASISMECHANISMUS FUER DEN SYSTEM EXIT '(&00)'  
LIEFERT FEHLERCODE '(&01)'  
(&00): EXIT-Nummer.  
(&01): Fehlercode der $SYSX-Schnittstelle.
```

Parameterbereich

Das Layout des Exit-Parameterbereichs kann mit dem Makro \$DOCSYS generiert werden.

```

        $DOCSYS MF=D
        MFTST MF=D,PREFIX=D,MACID=OCS,ALIGN=F,
        DMACID=OCS,SUPPORT=(D,C,M,L),DNAME=OCS_MDL
DOCS_MDL DSECT ,
        *,##### PREFIX=D, MACID=OCS #####
*   parameterarea description
DOCSHDR  FHDR  MF=(C,DOCS),EQUATES=NO           Standardheader
DOCSHDR  DS    0A
DOCSFHE  DS    0XL8                0   GENERAL PARAMETER AREA HEADER
*
DOCSIFID DS    0A                0   INTERFACE IDENTIFIER
DOCSFCTU DS    AL2                0   FUNCTION UNIT NUMBER
*
*                                     BIT 15   HEADER FLAG BIT,
*                                     MUST BE RESET UNTIL FURTHER NOTICE
*                                     BIT 14-12 UNUSED, MUST BE RESET
*                                     BIT 11-0   REAL FUNCTION UNIT NUMBER
DOCSFCT  DS    AL1                2   FUNCTION NUMBER
DOCSFCTV DS    AL1                3   FUNCTION INTERFACE VERSION NUMBER
*
DOCSRET  DS    0A                4   GENERAL RETURN CODE
DOCSSRET DS    0AL2               4   SUB RETURN CODE
DOCSSR2  DS    AL1                4   SUB RETURN CODE 2
DOCSSR1  DS    AL1                5   SUB RETURN CODE 1
DOCSMRET DS    0AL2               6   MAIN RETURN CODE
DOCSMR2  DS    AL1                6   MAIN RETURN CODE 2
DOCSMR1  DS    AL1                7   MAIN RETURN CODE 1
DOCSFHL  EQU   8                  8   GENERAL OPERAND LIST HEADER LENGTH
*
*   main return codes
DOCSNOER EQU   0                   no error detected
DOCSRREJ EQU   4                   request rejected
*
DOCSFCB@ DS    A                   address of P1-FCB
DOCS#    EQU   *-DOCSHDR

```

Return Exit (044)

Die Exit-Routine kann keine Rückkehrinformationen an den Systemmodul übergeben. Bei Ausgabedateien steht in D2LGINF im P2FCB die Adresse eines 320 Bytes großen Bereichs, der die Kennsätze VOL1, HDR1, HDR2, HDR3 enthält.

Request Exit (045)

Die Rückkehrinformation wird im Standard-Header des Parameterbereichs im Feld DOCSMRET übergeben.

Returncode: DOCSNOER = X'0000' = kein Fehler

Returncode: DOCSRREJ = X'0004' = OPEN soll abgewiesen werden

Wird der Spulenwechsel abgewiesen, erhält der Benutzer die Meldung DMS0E10 aufSYSOUT bzw. den Fehlercode X'0E10' im Feld ID1ECB des FCB:

% DMS0E10 DURCHFUEHRUNG DES SPULENWECHSELS WURDE VOM SYSTEMVERWALTER EINGESCHRAENKT

Bei Ausgabedateien steht in D2LGINF im P2FCB die Adresse eines 320 Byte großen Bereichs, der die Kennsätze VOL1, EOF1, EOF2, EOF3 enthält.

3.16 FILE-Exits (050/051)

Die Anwendung der folgenden Kommandos kann an zwei Exit-Punkten mit einer Exit-Routine überwacht werden:

- CREATE-FILE,
- MODIFY-FILE-ATTRIBUTES,
- CREATE-FILE-GENERATION,
- MODIFY-FILE-GENERATION-SUPPORT,
- ADD-FILE-LINK,
- CREATE-TAPE-SET,
- IMPORT-FILE (für Banddateien).

Außerdem ist die Überwachung des FILE-Makros möglich.

Die Request Exit-Routine kann die FILE-Anforderung abweisen oder modifizieren.

Die Return Exit-Routine kann nach Abarbeitung der FILE-Anforderung geeignete Aktionen zur Protokollierung bzw. Registrierung durchführen.

Return Exit (050)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX050-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Die Exit-Routine kann keine Rückkehrinformation an den Systemmodul senden. Nach der Rückkehr aus der Return-Exit-Routine erfolgt die Rückkehr aus der FILE-Verarbeitung. Der Maincode im Standard-Header des Parameterbereichs muss von der Exit-Routine mit X'0000' oder X'0004' versorgt werden. Enthält der Maincode andere Werte, wird auf dem Bedienungsplatz folgende Meldung ausgegeben:

```
% DMS06B9 DIE AN DEN SYSTEM-EXIT (&00) ANGESCHLOSSENE EXIT-ROUTINE LIEFERTE DEN
FEHLERCODE (&01)
```

Request Exit (051)

Registerversorgung wie Return Exit.

Die Rückinformation an den rufenden Systemmodul erfolgt im Maincode des Standardheaders des Exit-Parameterbereichs. Es sind folgende Werte möglich:

X'0000'	Die FILE-Verarbeitung wird mit dem eventuell von der Exit-Routine modifizierten FILE-Parameterbereich ausgeführt.
X'0004'	Die FILE-Anforderung wird mit Fehlercode X'0515' abgewiesen. Für Kommandos erfolgt auf SYSOUT die Meldung: <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"><pre>DMS0515 BENUTZUNG DES ANGEgebenEN KOMMANDOS ODER 'FILE' MAKROS WURDE VOM SYSTEMVERWALTER EINGESCHRAENKT. KOMMANDO ODER MAKRO ABGEWIESEN.</pre></div>
sonstige	Wie für X'0004'; zusätzlich wird folgende Meldung auf dem Bedienungsplatz ausgegeben: <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"><pre>DMS06B9 DIE AN DEN SYSTEM-EXIT (&00) ANGESCHLOSSENE EXIT-ROUTINE LIEFERTE DEN FEHLERCODE (&01)</pre></div>

Hinweis

Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.

DSECT

```

        EX050 MF=D
        MFTST MF=D,PREFIX=D,MACID=MAX,ALIGN=F,                                C
            DMACID=MAX,SUPPORT=(D,C,L),DNAME=MAXMDL
DMAXMDL DSECT ,
        *,##### PREFIX=D, MACID=MAX #####
*   Parameterarea
DMAXHDR FHDR MF=(C,DMAX),EQUATES=NO                                Standard-Header
DMAXHDR DS    0A
DMAXFHE DS    0XL8                0   GENERAL PARAMETER AREA HEADER
*
DMAXIFID DS    0A                0   INTERFACE IDENTIFIER
DMAXFCTU DS    AL2                0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*
*                               BIT 14-12 UNUSED, MUST BE RESET
*
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
DMAXFCT DS    AL1                2   FUNCTION NUMBER
DMAXFCTV DS    AL1                3   FUNCTION INTERFACE VERSION NUMBER
*
DMAXRET DS    0A                4   GENERAL RETURN CODE
DMAXSRET DS    0AL2              4   SUB RETURN CODE
DMAXSR2 DS    AL1                4   SUB RETURN CODE 2
DMAXSR1 DS    AL1                5   SUB RETURN CODE 1
DMAXMRET DS    0AL2              6   MAIN RETURN CODE
DMAXMR2 DS    AL1                6   MAIN RETURN CODE 2
DMAXMR1 DS    AL1                7   MAIN RETURN CODE 1
DMAXFHL EQU    8                8   GENERAL OPERAND LIST HEADER LENGTH
*
DMAXFRC DS    F                                FILE-Returncode
DMAXFPL DS    A                                Addr der FILE-PL
DMAX#   EQU    *-DMAXHDR

```

Beim Return-Exit enthält das Feld MAXFRC den Returncode der FILE-Verarbeitung.

FILE-Parameterbereich

Eine DSECT für den FILE-Parameterbereich kann erzeugt werden mit

```
[name] FILE MF=D[,PREFIX=prefix/*],VERSION=3
```

Der FILE-Parameterbereich muss mit dem seit BS2000/OSD-BC 3.0 gültigen Format (VERSION=3) erzeugt werden.

Das Feld MAXFPL der EX050-DSECT enthält die Adresse des FILE-Parameterbereichs.

3.17 ATTACH-Library-Exit (054)

Mit diesem Exit-Punkt kann die Systembetreuung festlegen, welche Programmbibliotheken überwacht werden sollen. Die Exit-Routine wird nach jedem ATTACH-Library (Eröffnen Programmbibliothek) angesprochen. Der Systembetreuer stellt mit der Exit-Routine fest, ob die Elementzugriffe auf diese aktuelle Bibliothek überprüft werden sollen.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(Exit-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

(siehe „[System Exits für Programmbibliotheken](#)“)

Versorgung des Parameterbereichs:

- Bibliotheksname
- LINK-Name der Bibliothek
- PLAM-Returncode1
- PLAM Returncode2

Die PLAM-Returncodes haben den Wert 0, wenn die Bibliothek korrekt eröffnet wurde.

Rückkehrinformation:

```
R15 = X'000000RR'
  RR = Rückkehrinformation der Exit-Routine
  00 Bibliothek soll überwacht werden
  04 Bibliothek soll nicht überwacht werden
```


3.18 DELETE-Element-Exits (055/056)

Das Löschen von Programmbibliothekselementen kann über zwei Exit-Punkte von der Systembetreuung überwacht werden.

- Am Request Exit kann die Systembetreuung den DELETE-Aufruf zurückweisen, bevor der Aufruf ausgeführt wird.
- Am Return Exit ist das Element bereits gelöscht. Die Systembetreuung erhält hier die volle Identifikation des gelöschten Elements.

Request Exit (055)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(Exit-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

(siehe „[System Exits für Programmbibliotheken](#)“)

Rückkehrinformation:

```
R15 = X'000000RR'
RR = Rückkehrinformation der Exit-Routine
00 keine Fehler
04 (Delete-Element abgewiesen)
```

Return Exit (056)

Der Return Exit ist identisch mit dem Request Exit.

DSECT

Die DSECT für den Parameterbereich kann mit dem Makro EX055 generiert werden:

```
name EX055 D[, [prefix], [ALL]]
      EX055 D
IEX055 DSECT
      DS      0F
*****
* DELETE  PARAMETER LIST FOR PLAM RZ ROUTINES      *
*****
PMRC      DC      F'0'          PLAM RETURN CODE
PMRC1     DC      F'0'          PLAM SECONDARY RETURN CODE
PMPLV     DC      CL8' '       VERSION OF PLAM: VZZ.ZBZZ
PMLIB     DC      A(0)         A(LIBRARY NAME):   CL54
PMLINK    DC      A(0)         A(LIBRARY NAME LINK):  CL8
          DC      CL1' '
          DC      CL1' '
          DC      CL1' '
          DC      CL1' '
PMTYP     DC      A(0)         A(MEMBER TYPE):     CL8
PMNAM     DC      A(0)         A(MEMBER NAME):     CL64
PMVER     DC      A(0)         A(MEMBER VERSION):  CL24
PMVAR     DC      A(0)         A(MEMBER VARIANT):  F
PMDAT     DC      A(0)         A(MODIFICATION DATE): CL14: DATE
*                                                ,CL8:  TIME
          DC      A(0)
          DC      A(0)
          DC      A(0)
LEN       EQU      *-PMRC
```

3.19 RENAME-Element-Exits (057/058)

Das Umbenennen von Programmbibliothekselementen kann über zwei Exit-Punkte von der Systembetreuung überwacht werden.

- Am Request Exit kann die Systembetreuung den RENAME-Aufruf zurückweisen, bevor die Funktion ausgeführt wird.
- Am Return Exit ist das Element bereits umbenannt. Die Systembetreuung erhält hier die vollen Identifikationen des Elements, d.h. sowohl die ursprüngliche als auch die neue Identifikation.

Request Exit (057)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(Exit-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

(siehe „[System Exits für Programmbibliotheken](#)“)

Rückkehrinformation:

```
R15 = X'000000RR'
  RR = Rückkehrinformation der Exit-Routine
  00 keine Fehler
  04 (RENAME-Element abgewiesen)
```

Return Exit (058)

Der Return Exit ist identisch mit dem Request Exit.

3.20 ERASE-Exits (060/061)

Bei der ERASE-Verarbeitung sind zwei Exit-Punkte vorgesehen, ein Request- und ein Return Exit. Sie klammern je einen Auftrag.

Ein Auftrag ist zum einen immer eines der Kommandos DELETE-FILE, DELETE-FILE-GROUP, DELETE-FILE-GENERATION, DELETE-SYSTEM-FILE, EXPORT-FILE bzw. der ERASE-Makroaufruf, zum anderen aber auch jeder Katalogeintrag, der von der ERASE-Funktion bearbeitet wird.

Eine Sonderstellung haben die Dateigenerationsgruppen: hier bilden nicht die Katalogeinträge der Generationen separate Aufträge, sondern die Dateigenerationsgruppe als Ganzes gilt als ein Auftrag.



- Am Request Exit kann die Exit-Routine den Auftrag prüfen und ggf. abweisen. Wenn der Return Exit eine AUDIT-Überwachung durchführen soll, muss der AUDIT-Indikator aus dem Katalogeintrag gesichert werden.
- Der Return Exit wird nach Abarbeitung eines Auftrags angesprungen. Die Exit-Routine kann den DMS-Fehlercode prüfen und ggf. Überwachungsmaßnahmen einleiten.

Im Register 1 enthalten beide Exits einen Zeiger auf eine Struktur, die vom Makro EX060 beschrieben wird und im Arbeitsbereich des aufrufenden Systemmoduls angelegt ist.

Will die Systembetreuung in der Exit-Routine Überwachungsmaßnahmen ergreifen, die vom AUDIT-Wert im Katalogeintrag abhängen, so muss diese Information bereits am Request Exit sichergestellt werden, da nach erfolgreichem Löschen (am Return Exit) der Katalogeintrag im Regelfall nicht mehr zur Verfügung steht.

Das Feld für den Dateinamen ist 80 Bytes lang. Es sollte in der Exit-Routine gelöscht werden, bevor es versorgt wird, da es sonst Schmierzeichen enthalten kann. Wird das Feld nicht gelöscht, muss das Endekriterium für den Dateinamen abgefragt werden (X'40').

Return Exit (060)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX060-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Der Return Exit soll den Erfolg eines Löschauftrages anzeigen. Er wird nicht aufgerufen, wenn die Request-Exit-Routine den ERASE abgewiesen hat.

Bei jedem Aufruf enthält das Feld &p.RET des ERASE-Parameterbereichs den Returncode, der beim Löschen bzw. Löschversuch aufgetreten ist; falls kein Fehler aufgetreten ist, steht dort der Wert X'00000000'. Der ERASE-Parameterbereich wird durch das Feld MAXEPL adressiert. Das Feld MAXFLC enthält den Wert, den das entsprechende MAXFLC-Feld des Request Exits enthielt. Ist MAXFLC Null, so handelt es sich um den Return Exit eines Kommandos bzw. Makros und &p.RET enthält den Sammelfehlercode.

Die Exit-Routine kann keine Rückkehrinformation an den Systemmodul senden. Nach der Rückkehr aus der Return-Exit-Routine erfolgt die Rückkehr aus der ERASE-Verarbeitung. Der Maincode im Standard-Header des Parameterbereichs muss von der Exit-Routine mit X'0000' oder X'0004' versorgt werden. Enthält der Maincode andere Werte, wird auf dem Bedienungsplatz folgende Meldung ausgegeben:

```
DMS06B9 DIE AN DEN SYSTEM-EXIT (&00) ANGESCHLOSSENE EXIT-ROUTINE LIEFERTE DEN
FEHLERCODE (&01)
```

Request Exit (061)

Registerversorgung wie Return Exit.

Der Request Exit wird genau einmal pro ERASE-Auftrag angesprungen. Der Exit-Parameterblock EX060 wird vorher wie folgt versorgt:

MAXEPL zeigt auf den ERASE-Parameterbereich (s.o.), MAXFLC enthält einen Zähler, der beim Kommando- bzw. Makro-Request mit Null initialisiert und bei jedem Auftrag um eins erhöht wird.

Die Rückinformation an den rufenden Systemmodul erfolgt im Maincode des Standardheaders des Exit-Parameterbereichs. Es sind folgende Werte möglich:

X'0000'	Die ERASE-Verarbeitung wird mit dem eventuell von der Exit-Routine modifizierten ERASE-Parameterbereich ausgeführt.
X'0004'	Die ERASE-Anforderung wird mit Fehlercode X'0514' abgewiesen. Für Kommandos erfolgt auf SYSOUT die Meldung: <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <pre>DMS0514 BENUTZUNG DES ANGEGEBENEN KOMMANDOS ODER 'ERASE' MAKROS WURDE VOM SYSTEMVERWALTER EINGESCHRAENKT. KOMMANDO ODER MAKRO ABGEWIESEN.</pre> </div>
sonstige	Wie für X'0004'; zusätzlich wird folgende Meldung auf dem Bedienungsplatz ausgegeben: <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <pre>DMS06B9 DIE AN DEN SYSTEM-EXIT (&00) ANGESCHLOSSENE EXIT-ROUTINE LIEFERTE DEN FEHLERCODE (&01)</pre> </div>

Hinweis

Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.

DSECT

```

        EX060 MF=D
        MFTST MF=D,PREFIX=D,MACID=MAX,ALIGN=F,                                C
            DMACID=MAX,SUPPORT=(D,C,L),DNAME=MAXMDL
DMAXMDL DSECT ,
        *,##### PREFIX=D, MACID=MAX #####
*   Parameterarea
DMAXHDR  FHDR  MF=(C,DMAX),EQUATES=NO                                Standard-Header
DMAXHDR  DS    0A
DMAXFHE  DS    0XL8                                0   GENERAL PARAMETER AREA HEADER
*
DMAXIFID DS    0A                                0   INTERFACE IDENTIFIER
DMAXFCTU DS    AL2                                0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*
*                               BIT 14-12 UNUSED, MUST BE RESET
*
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
DMAXFCT  DS    AL1                                2   FUNCTION NUMBER
DMAXFCTV DS    AL1                                3   FUNCTION INTERFACE VERSION NUMBER
*
DMAXRET  DS    0A                                4   GENERAL RETURN CODE
DMAXSR2  DS    0AL2                               4   SUB RETURN CODE
DMAXSR1  DS    AL1                                4   SUB RETURN CODE 2
DMAXSR1  DS    AL1                                5   SUB RETURN CODE 1
DMAXMRET DS    0AL2                               6   MAIN RETURN CODE
DMAXMR2  DS    AL1                                6   MAIN RETURN CODE 2
DMAXMR1  DS    AL1                                7   MAIN RETURN CODE 1
DMAXFHL  EQU   8                                8   GENERAL OPERAND LIST HEADER LENGTH
*
DMAXFLC  DS    F                                File Counter
DMAXEPL  DS    A                                Addr der ERASE-PL
DMAX#    EQU   *-DMAXHDR

```

Das Feld MAXEPL enthält einen Zeiger auf einen Parameterbereich der Version 3. Wurde ein ERASE der Version 3 verlangt, so ist dies die Kopie des Original-Parameterbereichs. Wurde ein ERASE der Version 0, 1 oder 2 verlangt, so wird das alte Format in das neue umgewandelt. Ein Zugriff auf den Original-Parameterbereich ist nicht möglich.

ERASE-Parameterbereich

Die DSECT für den ERASE-Parameterbereich kann erzeugt werden mit

```
[name] ERASE MF=D[,PREFIX=prefix/*],VERSION=3
```

Der ERASE-Parameterbereich muss mit dem seit BS2000/OSD-BC 3.0 gültigen Format (VERSION=3) erzeugt werden. Alle Exit-Routinen, die den ERASE-Parameterbereich verwenden, müssen an dieses neue Format angepasst werden.

3.21 CATALOG-Exits (070/071)

Die Anwendung der folgenden Kommandos kann an zwei Exit-Punkten mit einer Exit-Routine überwacht werden.

- CREATE-FILE,
- CREATE-FILE-GENERATION,
- CREATE-FILE-GROUP,
- MODIFY-FILE-ATTRIBUTES,
- MODIFY-FILE-GENERATION-SUPPORT,
- MODIFY-FILE-GROUP-ATTRIBUTES.

Außerdem kann der CATAL-Makro überwacht werden.

Die Request-Exit-Routine kann die CATALOG-Anforderung abweisen oder ändern.

Die Return-Exit-Routine kann nach Abarbeitung der CATALOG-Anforderung geeignete Aktionen zur Protokollierung bzw. Registrierung durchführen.

Return Exit (070)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX070 Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Die Exit-Routine kann keine Rückkehrinformation an den Systemmodul senden. Nach der Rückkehr aus der Return-Exit-Routine erfolgt die Rückkehr aus der CATALOG-Verarbeitung. Der Maincode im Standard-Header des Parameterbereichs muss von der Exit-Routine mit X'0000' oder X'0004' versorgt werden. Enthält der Maincode andere Werte, wird auf dem Bedienungsplatz folgende Meldung ausgegeben:

```
DMS06B9 DIE AN DEN SYSTEM-EXIT (&00) ANGESCHLOSSENE EXIT-ROUTINE LIEFERTE DEN
FEHLERCODE (&01)
```

Request Exit (071)

Registerversorgung wie Return Exit.

Die Rückinformation an den rufenden Systemmodul erfolgt im Maincode des Standardheaders des Exit-Parameterbereichs. Es sind folgende Werte möglich:

X'0000'	Die CATALOG-Verarbeitung wird mit dem eventuell von der Exit-Routine modifizierten CATALOG-Parameterbereich ausgeführt.
---------	---

X'0004' Die CATALOG-Anforderung wird mit Fehlercode X'0513' abgewiesen.
Für Kommandos erfolgt auf SYSOUT die Meldung:

```
DMS0513 DIE BENUTZUNG DES ANGEGBENEN KOMMANDOS ODER CATAL-MAKROS  
WURDE VOM SYSTEMVERWALTER EINGESCHRAENKT. AUFTRAG ABGEWIESEN.
```

sonstige	<p>Wie für X'0004!; zusätzlich wird folgende Meldung auf dem Bedienungsplatz ausgegeben:</p> <pre style="border: 1px solid black; padding: 5px;"> DMS06B9 DIE AN DEN SYSTEM-EXIT (&00) ANGESCHLOSSENE EXIT-ROUTINE LIEFERTE DEN FEHLERCODE (&01) </pre>
----------	---

Hinweise

- Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Die Exit-Routine kann den AUDIT-Parameter setzen oder verändern, ohne dass der Benutzer es beeinflussen oder verhindern kann.

DSECT

```

EX070 MF=D
MFTST MF=D,PREFIX=D,MACID=MAX,ALIGN=F, C
      DMACID=MAX,SUPPORT=(D,C,L),DNAME=MAXMDL
DMAXMDL DSECT ,
      *,##### PREFIX=D, MACID=MAX #####
*   Parameterarea
DMAXHDR FHDR MF=(C,DMAX),EQUATES=NO          Standard-Header
DMAXHDR DS    0A
DMAXFHE DS    0XL8          0  GENERAL PARAMETER AREA HEADER
*
DMAXIFID DS    0A          0  INTERFACE IDENTIFIER
DMAXFCTU DS    AL2          0  FUNCTION UNIT NUMBER
*
*          BIT 15  HEADER FLAG BIT,
*          MUST BE RESET UNTIL FURTHER NOTICE
*          BIT 14-12 UNUSED, MUST BE RESET
*          BIT 11-0  REAL FUNCTION UNIT NUMBER
DMAXFCT DS    AL1          2  FUNCTION NUMBER
DMAXFCTV DS    AL1          3  FUNCTION INTERFACE VERSION NUMBER
*
DMAXRET DS    0A          4  GENERAL RETURN CODE
DMAXSRET DS    0AL2        4  SUB RETURN CODE
DMAXSR2 DS    AL1          4  SUB RETURN CODE 2
DMAXSR1 DS    AL1          5  SUB RETURN CODE 1
DMAXMRET DS    0AL2        6  MAIN RETURN CODE
DMAXMR2 DS    AL1          6  MAIN RETURN CODE 2
DMAXMR1 DS    AL1          7  MAIN RETURN CODE 1
DMAXFHL EQU    8          8  GENERAL OPERAND LIST HEADER LENGTH
*
DMAXCRC DS    F              CATAL-Returncode
DMAXCPL DS    A              Addr der CATAL-PL
DMAX#   EQU    *-DMAXHDR

```

Beim Return-Exit enthält das Feld MAXCRC den Returncode der CATAL-Verarbeitung.

CATALOG-Parameterbereich

Eine DSECT für den CATALOG-Parameterbereich kann erzeugt werden mit

```
[name] CATAL MF=D[,PREFIX=prefix/*],VERSION=3
```

Der CATALOG-Parameterbereich muss mit dem seit BS2000/OSD-BC 3.0 gültigen Format (VERSION=3) erzeugt werden.

Das Feld MAXCPL des Exit-Parameterbereichs enthält einen Zeiger auf den CATALOG-Parameterbereich.

3.22 DCAM-Exit (075)

Der DCAM-Exit bietet dem Benutzer die Möglichkeit, DCAM-Funktionen zu erweitern, zu ändern und eigene spezielle Funktionen hinzuzufügen. Solche Funktionen sind z.B.:

- über TRANSDATA hinausgehender Datenschutz durch erweiterte Berechtigungsprüfungen, z.B. Öffnen von Anwendungen abhängig machen von der Benutzererkennung: Verbindungsaufbau für eine Anwendung nur mit definierten Partnern/Prozessoren zulassen
- zusätzliche Aktionen veranlassen, z.B. Jobs starten, Meldungen ausgeben

Die benutzerspezifische Exit-Routine wird beim Öffnen und Schließen von Anwendungen sowie beim Öffnen und Schließen von Verbindungen aufgerufen. Beim Öffnen von Anwendungen und Verbindungen kann die Exit-Routine den Aufruf annehmen, ablehnen oder ggfs. modifizieren. Die Aufrufe beim Schließen dienen nur zur Information der Exit-Routine.

Aufrufereignisse

Zurzeit gibt es 12 Aufrufereignisse für die DCAM-Exit-Routine. Zu unterscheiden sind Hauptereignisse (Exit Maincases) und Nebenereignisse (Exit Subcases), die jeweils durch eine 3 Zeichen lange Buchstabenfolge gekennzeichnet sind. Die 6-stellige Kombination aus Haupt- und Nebenereignis-Kennzeichnung bezeichnet eindeutig das Aufrufereignis.

i Es muss damit gerechnet werden, dass Exit Maincase und Exit Subcase in zukünftigen DCAM Versionen erweitert werden!

Auf den folgenden Seiten sind die Hauptereignisse APP und CON und ihre Nebenereignisse einander gegenübergestellt.

Hauptereignis APP

APP betrifft Ereignisse, die mit dem Öffnen und Schließen von Anwendungen (Application) zusammenhängen.

Exit Subcase	Aufruf der Exit-Routine	Funktion der Exit-Routine
APPOP	bei jedem YOPEN-Aufruf: nach Prüfung der Benutzerangaben vor Bearbeitung des Aufrufs	<ul style="list-style-type: none"> • zusätzliche Prüfungen, z.B.: Anwendung darf nur von bestimmten Benutzern eröffnet werden; Tabellen nachladen • YOPEN-Aufruf akzeptieren • YOPEN-Aufruf ablehnen • YOPEN-Parameter modifizieren • zusätzliche Aktionen veranlassen; z.B. Jobs starten, etc.

APPCLS	bei jedem YCLOSE-Aufruf: vor Abbau der DCAM-Datenstruktur	<ul style="list-style-type: none"> zusätzliche Aktionen veranlassen; z.B.: Meldung ausgeben, exitspezifische Tabellen aktualisieren etc.
APPSHU	bei jedem internen Schließen einer Anwendung, wenn der Operatorein BCEND-, BCLOSE- oder SHUTDOWN-Kommando eingegeben hat *)	
APPTRM	beim internen Schließen einer Anwendung, bei Programm- oder Taskterminierung	Ersatztask starten
APPFCL	bei unvorhergesehenem Schließen einer Anwendung wegen einer DCAM-Störung	

*) Der Exit wird nur beim Schließen, nicht bei einer eventuell vorangehenden Warnung aufgerufen. Er wird nicht aufgerufen, wenn vor der DCAM-Terminierung die TIAM-Terminierung abläuft, d.h. wenn das DCAM-Programm als Dialogtask gestartet wurde. In diesem Fall wird bereits durch TIAM eine Task-Terminierung eingeleitet, d.h. der Exit APPTRM wird aufgerufen.

Hauptereignis CON

CON betrifft Ereignisse, die mit dem Auf-/Abbau von Verbindungen zusammenhängen. Die Exit-Routine wird jedoch nicht aufgerufen, wenn beim Schließen einer Anwendung implizit alle noch bestehenden Verbindungen geschlossen werden.

Folgende Nebenereignisse sind zu unterscheiden:

- Auffordern zum Verbindungsaufbau (YOPNCON ACQUIRE):
 - beim YOPNCON-ACQUIRE-Aufruf des Benutzers
 - nach erfolgreichem Verbindungsaufbau
- Annehmen einer Verbindungsaufforderung
 - beim Eintreffen einer Aufforderung zum Verbindungsaufbau
 - nach Annahme der Aufforderung (YOPNCON ACCEPT)

- bei explizitem Verbindungsabbau oder nach einem DCAM-Fehler:
 - Verbindungsabbau durch den Benutzer
 - Erzwungener Verbindungsabbau durch Störung
 - Verbindungsabbau durch den Partner

Exit Subcase	Aufruf der Exit-Routine	Funktion der Exit-Routine
CONACQ	bei Senden einer Aufforderung zum Verbindungsaufbau (YOPNCON ACQUIRE), nach Prüfung der Benutzerangaben vor Bearbeitung des Aufrufs (d.h. vor BCAM-Call 'REQCON')	<ul style="list-style-type: none"> ◦ zusätzliche Prüfungen z.B. Verbindungsnachricht (LOGON MESSAGE) gemäß Vereinbarungen? ◦ Aufruf annehmen ◦ Aufruf ablehnen ◦ YOPNCON-Parameter ändern
CONACC	bei jedem Annehmen eines Verbindungswunsches (YOPNCON ACCEPT), nach Prüfung der Benutzerangaben, vor Bearbeitung des Aufrufs	<ul style="list-style-type: none"> ◦ zusätzliche Prüfungen ◦ Auftrag annehmen ◦ Auftrag ablehnen ◦ YOPNCON-Parameter ändern
CONCPL	nach erfolgreichem Verbindungsaufbau, d.h., wenn ein YOPNCON ACQUIRE vom Partner angenommen wurde	
CONREQ	beim Eintreffen eines Verbindungswunsches, nach Prüfung der Angaben, vor Bearbeitung des Verbindungswunsches	<ul style="list-style-type: none"> ◦ zusätzliche Prüfungen ◦ Verbindungswunsch annehmen / ablehnen
CONCLS	bei jedem Verbindungsabbau durch den Benutzer (YCLSCON), vor Freigabe der DCAM-Datenstruktur	
CONFCL	bei jedem erzwungenen Verbindungsabbau wegen NEA-Protokollfehler oder wegen Quittungsüberlauf	
CONBAD	bei jedem Verbindungsabbau durch den Partner	

Folgende Informationen werden an die Exit-Routine übergeben:

```

R1 = A(YDDEXPL) = DCAM-Exit-Parameterbereich
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
    
```

Rückkehrinformation an die rufende Systemkomponente erfolgt über den Parameterbereich im Feld YDDXRETC:

YDDXRETC	X'00FDBKRC'	
	FDBK =	Rückkehrinformation der Exit-Routine an den DCAM-Benutzer
	RC =	Rückkehrinformation der Exit-Routine an die rufende DCAM-Komponente.
		00 Aufruf annehmen
		04 Aufruf ablehnen
		08 Aufruf modifizieren

Das Feld FDBK muss versorgt werden, wenn RC=04 gesetzt ist. Die Angabe wird an den DCAM-Benutzer im Feld FDBK1/FDBK2 (siehe Handbuch „DCAM-Makroaufrufe“ [1]) übergeben. FDBK kann entweder aus einem für die Exit-Routine reservierten Bereich 'CC00'-'CCFF' genommen werden oder kann einer der gültigen DCAM-Returncodes sein (siehe Handbuch „Systemmeldungen“ [15]). FDBK darf nicht '0000' sein. Ist RC = 08, übergibt die Exit-Routine im Parameterbereich YDDEXPL die gültigen Daten für die Modifikation. Die Hinweise zur Programmierung von Exit-Routinen sind zu beachten.

Das Feld FDBK wird nur bei den Exitereignissen APPOP, CONACQ und CONACC an den Benutzer übergeben (entspricht YOPEN, YOPNCON ACQ und YOPNCON ACC), jedoch nicht an den Partner geschickt.

Bei Ablehnung eines Verbindungswunsches durch den Systemexit des Partners wird der YOPNCON ACQUIRE nicht mit dem dort angegebenen FDBK sondern mit X'0C40' beendet.

DSECT

```

          YDDEXPL D
YDDEXPL DSECT
*****
*
*   DCAM  EXIT PARAMETERLISTE
*
*****
*
*  EINGABEPARAMETER
*
YDDXDPAR DS      A          A(DARPPAR)
*                DCAM INTERNER ARBEITSBEREICH
YDDXMAIN DS      X          EXIT-MAIN-CASE
YDDXAPPL EQU     1          APPL
YDDXCONN EQU     2          CONN
YDDXSUB DS       X          EXIT-SUB-CASE
YDDXAOPN EQU     1          APPOPN
YDDXACLS EQU     2          APPCLS
YDDXASHU EQU     3          APPSHU
YDDXATRM EQU     4          APPTRM
YDDXAFCL EQU     5          APPFCL
YDDXCACQ EQU     6          CONACQ
YDDXCACC EQU     7          CONACC
YDDXCCLPL EQU    8          CONCPL
YDDXCREQ EQU     9          CONREQ
YDDXCCLS EQU    10          CONCLS
YDDXCFCL EQU    11          CONFCL
YDDXCBAD EQU    12          CONBAD
*
*  EINGABE-/RUECKGABEPARAMETER
*
YDDXLOGL DS      H          LAENGE DER LOGON MESSAGE
YDDXLOGM DS      CL80       LOGON MESSAGE
*
*  RUECKGABEPARAMETER
*
YDDXPW DS        XL4        PASSWORT
YDDXRETC DS      F          RETURNCODE '..FDBKRC'
*
YDDXL EQU        *-YDDEXPL  LAENGE DER DCAM EXIT PARAMETERLISTE
*                *,YDDEXPL  060    950330    55647211

```

Beschreibung der YDDEXPL-Felder

YDDXDPAR	A(DARPPAR)	
	Adresse eines DCAM-internen Arbeitsbereichs, in dem weitere aufrufspezifische Informationen an die Exit-Routine übergeben werden	
YDDXMAIN	Indikator für das Hauptereignis	
YDDXMAIN	Indikator für das Hauptereignis	
	YDDXAPPL	Hauptereignis ' APP' (betr. Anwendungen)
	YDDXCONN	Hauptereignis ' CON' (betr. Verbindungen)
YDDXSUB	Indikator für das Nebenereignis	
	YDDXAOPN	Exit-Case APPOP
	YDDXACLS	Exit-Case APPCLS
	YDDXASHU	Exit-Case APPSHU
	YDDXATRM	Exit-Case APPTRM
	YDDXAFCL	Exit-Case APPFCL
	YDDXCACQ	Exit-Case CONACQ
	YDDXCACC	Exit-Case CONACC
	YDDXCCLPL	Exit-Case CONCPL
	YDDXCREQ	Exit-Case CONREQ
	YDDXCCLS	Exit-Case CONCLS
	YDDXCFCFCL	Exit-Case CONFCL
	YDDXCBCAD	Exit-Case CONBAD
	YDDXLOGL	Länge der Verbindungsnachricht (LOGON-Message) im Feld YDDXLOGLM; die Verbindungsnachricht darf höchstens 80 Bytes lang sein. Dieses Feld kann von der Exit-Routine verändert werden. YDDXLOGL = 0: Es existiert keine Verbindungsnachricht
YDDXLOGM	enthält die Verbindungsnachricht (LOGON-Message), die von der Exit-Routine verändert werden kann.	
YDDXPW	enthält ein Kennwort, das die Exit-Routine beim Rücksprung an DCAM übergibt. Dieses Kennwort ersetzt das vom Benutzer angegebene RDF-Kennwort beim YOPEN oder das LOGON-Kennwort beim YOPNCON ACQUIRE.	

YDDXRETC	Returncode der Exit-Routine
----------	-----------------------------

Hinweise zur Programmierung von DCAM-Exit-Routinen

Zusätzlich zu den in „Exit-Routinen“ beschriebenen Regeln gelten für DCAM-Exit-Routinen folgende Einschränkungen

- Auf alle Felder der DCAM-Datenstruktur (DARPPAR, Kontrollblöcke, s.u.) darf nur lesend zugegriffen werden. Schreibzugriff ist nur für die Rückgabefelder im Parameterbereich YDDEXPL erlaubt.
- Es sind keine BCAM-Aufrufe erlaubt.
- Die Exit-Routine darf keine Börsen-Locks setzen.
- Die Exit-Routine darf ihren Prozess(PCB)-Level nicht ändern.

Auszug aus der DCAM-Datenstruktur

Die für die Programmierung von Exit-Routinen wichtigsten Elemente der DCAM-Datenstruktur sind nachfolgend aufgeführt. Da für das Layout der DCAM-Kontrollblöcke keine Kompatibilität garantiert wird, darf der Zugriff zur DCAM-Datenstruktur nur über die symbolischen Adressen erfolgen. Bei DCAM-Versionswechsel können Neuübersetzungen oder Sourceänderungen notwendig werden.

DCAM-interner Arbeitsbereich

DARPPAR

enthält taskspezifische Daten, z.B. die aktuellen Adressen der DCAM-Kontrollblöcke (siehe Handbuch „DCAM Makroaufrufe“ [1])

Eine DSECT für DARPPAR kann erzeugt werden mit dem Makroaufruf

```
[name] YDDDVEC D [,prefix]
```

Standardwerte:	name	YDDDVEC
	prefix	YDDV

DCAM Application Control Block

DACB

enthält anwendungsspezifische Daten, z.B. Anwendungsname, Attribute ... Der DACB wird beim YOPEN angelegt und bleibt bis zum Schließen der Anwendung bestehen.

Eine DSECT kann erzeugt werden mit dem Makroaufruf

```
[name] YDDDACB D [,prefix]
```

Standardwerte:	name	YDDDACB
	prefix	YDDH

DCAM Connection Control Block

DCCB

enthält verbindungspezifische Daten, z.B. Partner-, Prozessornamen. Pro Verbindung wird ein DCCB angelegt.

Eine DSECT kann generiert werden mit dem Makroaufruf

[name] YDDDCCB D [,prefix]

Standardwerte:	name	YDDDCCB
	prefix	YDDL

DCAM Request Parameter Block

DRPB

enthält bei asynchronen Aufrufen die aufrufspezifischen Informationen aus dem RPB.

Eine DSECT kann erzeugt werden mit dem Makroaufruf

[name] YDDDRPB D [,prefix]

Standardwerte:	name	YDDDRPB
	prefix	YDDD

Feldinhalte bei Aufruf der Exit-Routine

Abhängig vom Aufrufereignis (Exit Case) werden die in den folgenden Tabellen enthaltenen Felder des Exit-Parameterbereichs und des internen Arbeitsbereiches versorgt. Nicht versorgte Felder und Adressen werden mit X'00' vordefiniert.

YDDEXPL - Exit-Parameterbereich

Feldname	Feldinhalt	Exit Case												
		APP					CON							
		OPN	CLS	SHU	TRM	FCL	ACQ	ACC	CPL	REQ	CLS	FCL	BAD	
YDDXMAIN	Exit Main Case	x	x	x	x	x	x	x	x	x	x	x	x	x
YDDXSUB	Exit Sub Case	x	x	x	x	x	x	x	x	x	x	x	x	x
YDDXDPAR	A(DARPPAR)	x	x	x	x	x	x	x	x	x	x	x	x	x
YDDXLOGL	Länge LOGON Mess.						x	x	x	x				
YDDXLOGM	A(LOGON Message)						x	x	x	x				

YDDDVEC - DCAM-interner Arbeitsbereich

Feldname	Feldinhalt	Exit Case												
		APP					CON							
		OPN	CLS	SHU	TRM	FCL	ACQ	ACC	CPL	REQ	CLS	FCL	BAD	
YDDVXTID	A(TID)	x	x	x	x	x	x	x	x	x	x	x	x	x
YDDVADAC	A(DACB)	x	x	x	x	x	x	x	x	x	x	x	x	x

YDDVADCC	A(DCCB)						x	x	x			x	x
YDDVADRP	A(DRPB)						x	x	x				
YDDVARPB	A(RPB)						x	x	x		x		
YDDVAPCN	A(PCN) *)							x		x			
YDDVAPT	A(PTN) *)							x		x			

*) Bei Anstoß des Exits durch das Ereignis CONREQ existiert noch kein Verbindungssteuerblock (DCCB). Der Benutzer kann sich mithilfe von YDDVAPT über die Partneradresse informieren, jedoch nicht modifizieren.

PCN CL8 PROCESSOR NAME

PTN CL8 PARTNER NAME

Rückkehrinformation der Exit-Routine

Bei den Aufrufereignissen APPCLS, APPSHU, APPTRM, APPFCL, CONCPL, CONCLS, CONFCL und CONBAD muss die Exit-Routine das rechte Byte im Feld YDDXRETC auf null setzen (RC = X'00'). An diesen Aufrufereignissen kann nur Information an die Exit-Routine übergeben werden, die DCAM-Verarbeitung kann nicht durch die Exit-Routine beeinflusst werden.

Bei den Aufrufereignissen APPOP, CONACQ, CONACC und CONREQ kann die Exit-Routine über den Returncode im Feld YDDXRETC des Parameterbereichs die DCAM-Verarbeitung beeinflussen. Welche Returncodes zu setzen sind, zeigt die folgende Tabelle.

Returncodes (1)

		Bedingung	Exit Case			
			APPOP	CONACQ	CONACC	CONREQ
YDDXRETC	RC = 00		x	x	x	x
	RC = 04		x	x	x	x
	RC = 08		x	x	x	-
	FDBK	RC = 04	+	+	+	+
YDDEXPL	YDDXPW	RC = 08	*	*	-	-
	YDDXLOGL	RC = 08	-	*	*	-
	YDDXLOGM	YDDXLOGL > 0	-	*	*	-

Erläuterung:

- x: der Wert/das Feld kann angegeben werden
 -: der Wert/das Feld kann nicht angegeben werden
 +: das Feld muss angegeben werden, wenn die Bedingung zutrifft
 *: das Feld kann bei RC = 08 verändert an DCAM zurückgegeben werden

Die in der vorangehenden Tabelle mit '*' gekennzeichneten Fälle sind in der folgenden Tabelle näher erläutert.

Returncodes (2)

Exit-Case	Feld in YDDEXPL	Modifikation
APPOPN	YDDXPW	das RDF-Kennwort (Parameter USEPW im ACB) wird beim BCAM-Aufruf durch das übergebene Kennwort ersetzt; YDDXPW = 0: das Kennwort wird nicht verändert
CONACC CONACQ	YDDXLOGL	YDDXLOGL = 0: die Verbindungsnachricht wird nicht verändert YDDXLOGL = X' 7FFF' : die Verbindungsnachricht soll gelöscht werden
CONACC CONACQ	YDDXLOGM	die Verbindungsnachricht (LOGON Message) des Benutzers (Inhalt von AREA) wird vor dem BCAM-Aufruf durch die übergebenen Daten ersetzt
CONACQ	YDDXPW	das LOGON-Kennwort (Parameter LOGPW im CCB) wird vor dem BCAM-Aufruf durch das übergebene Kennwort ersetzt YDDXPW = X' 00' : das Kennwort wird nicht ersetzt

3.23 SYSCMD-Exit (080)

Dieser Exit wird jedes Mal bei der Eingabe eines Systemkommandos durchlaufen, sofern die Task aktiviert ist.

Drei Fälle der Kommandoeingabe werden unterschieden:

- im Dialog von der Datensichtstation (interaktiver Modus)
- aus einer katalogisierten Datei (SPOOLIN, ENTER-JOB, CALL-PROCEDURE) oder im Stapelbetrieb
- aus einem TU-Programm über die MCLP-Schnittstelle (siehe Handbuch „Makroaufrufe an den Ablaufteil“ [9])

Mithilfe der Exit-Routine kann der Kommandofluss aller aktiven Tasks während der OPEN-Session überwacht werden.

Die Exit-Routine kann

- das eingegebene Kommando unverändert akzeptieren (Returncode EX080CAC)
- das Kommando abweisen (Returncode EX080CRJ)
- das Kommando verändern oder ersetzen (Returncode EX080CRO)
- ein neues oder mehrere Kommandos einfügen (Returncodes EX080CRS/EX080CLR)

Die Rückinformation der Exit-Routine muss im Feld EX080RC des Parameterbereichs EX080 hinterlegt werden. Die Adresse des Parameterbereichs wird im R1 an die Exit-Routine übergeben.

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX080-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Hinweise

- Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Die SYSCMD-Exit-Routine sollte Kommandos, die über die verschiedenen Wege eingegeben werden, dennoch gleich verarbeiten.
- Wird die Exit-Routine deaktiviert, während ein Kommando durch mehrere ersetzt wird, wird der Dummy-Kommandosatz /REMARK SYSCMD EXIT NO MORE ACTIVE DURING PROCESSING an das Benutzerprogramm gesendet.
- Werden in einer Prozedur Kommandos mit /SKIP-COMMANDS übergangen, wird für jedes übergangene Kommando die Exit-Routine gerufen. Das Gleiche gilt, wenn durch Spinoff (Suche nach STEP, LOGOFF, SET-JOB-STEP) Kommandos übergangen werden.
- In CALL-PROCEDURE-Prozeduren ist die Variablensubstitution zum Zeitpunkt des Exit-Aufrufs bereits durchgeführt.
- Das Sprungziel bei Sprungmarken wird zweimal zur Verfügung gestellt.
- Die Rückinformation an die rufende Systemkomponente muss im Feld EX080RC des Parameterbereichs hinterlegt werden.

DSECT

```

          EX080 D
EX080PL DSECT
* -
* -      INPUT  INFORMATION
* -
EX080IND DC    AL1(0)          SOURCE OF COMMANDS
EX080TER EQU    1              -CMD READ FROM A TERMINAL
EX080FIL EQU    2              -CMD READ FROM A JCL FILE
EX080MCL EQU    4              -CMD READ VIA MCLP
EX080TRN EQU    8              -CMD READ VIA TRCMD          999
EX080BUF EQU   16              -CMD FOUND IN BS2000 BUFFER  999
* -
EX080BIT DC    X'00'          BIT_INFO          900
EX080SDF EQU   X'80'          -INPUT IN SDF STRUCTURED FORM002
EX080SD2 EQU   X'40'          -INPUT IN NEW SDF STRUCT FORM120
EX080STR EQU   X'00'          -INPUT IN STRING MODE          900
* -
EX080UNU DC    2X'00'          900
* -
EX080IR  DC    A(0)           A (INPUT RECORD)
* -
* -      OUTPUT INFORMATION
* -
EX080RC  DC    AL1(0)         SYSCMD EXIT'S RETURN CODE
EX080CAC EQU    0              -CMD REC ACCEPTED
EX080CRJ EQU    4              -CMD REC REJECTED
EX080CRO EQU    8              -CMD REC REPLACED BY ONE
EX080CRS EQU   12              -CMD REC REPLACED BY SEVERAL
EX080CLR EQU   16              -LAST CMD REC REPLACED
          DC    3X'00'         UNUSED IN VER=710
* -
EX080RR  DC    A(0)           A (RETURNED RECORD)
* -
* -      ADDITIONAL INPUT  INFORMATION
* -
EX080ILR DC    A(0)           A (LOWER CASE INPUT RECORD)
EX080STD DC    A(0)           A (STD FORM)
EX080PLL EQU   *-EX080PL     LENGTH OF SYSCMD EXIT  P/L

```

Erklärung der Felder

EX080IND	Art der Kommandoeingabe	
	EX080TER	Kommandoingabe von der Datenstation
	EX080FIL	Kommandoingabe aus einer JCL-Date
	EX080MCL	Kommandoingabe über MCLP aus einem Benutzerprogramm
EX080BIT	Indikator für SDF	
	EX080SDF	Eingabe in SDF-Struktur

	EX080STR	Eingabe nicht in SDF-Struktur
EX080IR	Adresse des Eingabe-Kommandosatzes	
EX080RC	Rückinformation	
	EX080CAC	Kommando akzeptiert
	EX080CRJ	Kommando abgewiesen
	EX080CRO	Kommando geändert oder ersetzt
	EX080CRS	Kommando wird durch mehrere Kommandos ersetzt
	EX080CLR	letzter Kommandosatz ersetzt
EX080RR	Adresse des Kommandosatzes, der an die rufende Systemkomponente zurückgegeben wird (bei Ändern/Ersetzen)	

Erklärung der Returncodes im Feld EX080RC

Returncode X'04': Kommando zurückweisen (EX080RC=EX080CRJ)

Im Dialogbetrieb wird keine Systemmeldung ausgegeben; die Exit-Routine kann jedoch eine Meldung an den Benutzer senden

Bei Kommandoeingabe aus einer katalogisierten Datei (PROC A oder PROC C bzw. entsprechende SDF-Kommandos) wird folgende Meldung ausgegeben:

```
/REMARK COMMAND REJECTED BY THE SYSTEM ADMINISTRATOR
```

Im Stapelbetrieb wird die gleiche Meldung in die Protokolldatei aufgenommen.

Returncode X'08': Kommando ersetzen oder ändern (EX080RC=EX080CRO)

Die Exit-Routine muss das neue oder geänderte Kommando selbst aufbauen und die Adresse des Kommandosatzes im Feld EX080RR des Parameterbereichs an die rufende Systemkomponente übergeben.

Aufbau des Kommandosatzes (sofern nicht normierter SDF-Übergabebereich)

0	2	4	SL-1
SL		Kommando mit Operanden	

SL = Satzlänge

Der Kommandosatz darf keinen ISAM-Schlüssel, kein Fortsetzungszeichen und keine symbolischen Parameter (&...) enthalten. Die Satzlänge darf verändert werden.

- Im Dialogbetrieb erfolgt keine Systemmeldung.
- Bei Kommandoeingabe aus einer katalogisierten Datei wird das neue oder geänderte Kommando ausgegeben bzw. in die Protokolldatei geschrieben.

- Bei Kommandoeingabe über MCLP muss das ersetzende Kommando ebenfalls MCLP-zulässig sein, sonst wird der Returncode X'14' (= ungültiges Kommando) an das Benutzerprogramm zurückgegeben. Es darf weder DO-, CALL-, LOAD-, LOGOFF-, ABEND- oder EXECUTE-Kommando zurückgegeben oder ein anders Kommando über MCLP aufgerufen werden, da dann das Anwenderprogramm sofort entladen wird. – Tritt bei Abarbeitung des neuen Kommandos ein Fehler auf, wird X'10' (unzulässiges Kommando) an das Anwenderprogramm gemeldet.

*Returncodes X'0C'/X'10': ein Kommandos durch mehrere Kommandos ersetzen
(EX080RC=EX080CRS/EX080CLR)*

Gibt die Exit-Routine einen Kommandosatz an das System zurück (siehe RC = X'08') und setzt gleichzeitig im Feld EX080RC den Returncode auf X'0C' (= EX080CRS), erhält sie sofort die Steuerung zurück und kann einen weiteren Kommandosatz übergeben. Ab dem zweiten Aufruf nach Returncode X'0C', ist die Adresse des Eingabe-Kommandosatzes (Feld EX080IR des Parameterbereichs) auf null gesetzt. Wenn die Exit-Routine das letzte Kommando zurückgibt, muss sie den Returncode im Parameterbereich auf X'10' (=EX080CLR) setzen.

- Im Dialogbetrieb erfolgt keine Systemmeldung, wenn ein Kommando durch mehrere ersetzt wurde.
- Im Stapelbetrieb werden die ersetzenden Kommandos in die Protokolldatei geschrieben.
- Bei Kommandoeingabe über MCLP sind die Hinweise zu beachten, die den Returncode '08' betreffen. Die o.g. Kommandos dürfen auch nicht in einer Kommandofolge zurückgegeben werden, da dann das Anwenderprogramm sofort entladen wird. Wenn im Anwenderprogramm ein Puffer für die Aufnahme der Bearbeitungsergebnisse des MCLP-Kommandos zur Verfügung gestellt wird, muss dieser Puffer die Ergebnisse der gesamten Kommandofolge aufnehmen können. Es ist zu erwarten, dass viele Anwenderprogramme, die MCLP-Aufrufe benutzen, nicht mehr korrekt ablaufen, wenn die Exit-Routine ein Kommando durch mehrere ersetzt!

3.24 SDF-Exits (080/081/082/083)

In Systemen mit SDF werden ergänzend zum Exit 080 (siehe SYSCMD-Exit (080)) drei weitere Exits angeboten, siehe folgende Übersicht:

Exit-Nr.	Eingabe von	Art d. Exits	Aufrufzeitpunkt
080	SYSCMD	request	vor Bearbeitung des Kommandos (entspricht Exit 080 ohne SDF)
081	SYSCMD	return	nach Bearbeitung des Kommandos durch den Entschlüssler, wenn kein Syntaxfehler festgestellt wurde
082	SYSSTMT	request	vor Bearbeitung der Anweisung durch den Entschlüssler
083	SYSSTMT	return	nach Bearbeitung der Anweisung durch den Entschlüssler, wenn kein Syntaxfehler festgestellt wurde

Alle vier Exits verwenden den Parameterbereich EX080. Die Adresse des Parameterbereichs wird im Register 1 an die Exitroutine übergeben.

Mit den Exits kann das Kommando akzeptiert, abgewiesen, geändert oder ersetzt werden.

Die Rückinformation der Exit-Routine muss im Feld EX080RC des Parameterbereichs hinterlegt werden.

Hinweis

Für sinnvollen Einsatz der Exits in SDF-Umgebung sind tiefere Kenntnisse der Schnittstelle erforderlich, deren Darstellung den Rahmen dieses Handbuchs sprengt. Bitte entnehmen Sie die entsprechenden Informationen den Handbüchern „SDF-A“ [10], „Einführung in die Dialogschnittstelle (SDF)“ [11], „Kommandos“ [8], „Einführung in die Systembetreuung“ [7].

3.24.1 SYSCMD-Exit 080

Es gilt die Beschreibung aus Exit 080. Zusätzlich ist zu beachten

- Die Exit-Routine wird im geführten SDF-Dialog nicht angesprungen.
- Der Ansprung der Exit-Routine erfolgt unabhängig davon, ob das Kommando intern vom Entschlüssler selbst gelesen wurde (EX080BUF) oder ob das Kommando mit TRCMD übergeben wurde (EX080TRN; siehe Erweiterung im Parameterbereich EX080).
- Wird eine Eingabe erkannt, die nicht für die aktuelle System-Syntaxdatei bestimmt ist, wird die Exit-Routine auch dann angesprungen, wenn die Eingabe zu einem untergeordneten „Kommandostrom“ gehört und keine Fortsetzungszeile erwartet wurde.
- Werden Datensätze gelesen, wenn ein Kommando erwartet wurde, so wird die Exit-Routine nicht aufgerufen und die Fehlerbehandlung angestoßen.
- Wird Dateiende beim Leseversuch erkannt, wird die Exit-Routine nicht aufgerufen und die EOF-Behandlung wird durchgeführt.

3.24.2 SYSCMD-Exit 081

Dieser Exit ist nur in Systemen mit SDF verwendbar. Zum Aufrufzeitpunkt wurde das zu behandelnde Kommando vom Entschlüssler als syntaktisch richtig erkannt. Parameterersetzung und Fortsetzungsbehandlung sind abgeschlossen.

Der Exit-Routine bekommt das Kommando im Normalfall in Form des Eingabestrings übergeben. Die Adresse steht im Feld EX080IR im Parameterbereich.

Wurde das Kommando mit der Operandenangabe IMPLEMENTOR=P2(...,CALL=NEW,...) in der SDF-A-Anweisung //ADD-COMMAND in die Syntaxdatei eingetragen, wird im Parameterbereich die Adresse des normierten Übergabebereichs an die Exit-Routine übergeben. Die Exit-Routine muss auf beide Formen der Übergabe entsprechend vorbereitet sein. Die Beschreibung des normierten Übergabebereichs, die Anweisung //ADD-COMMAND und viele weitere Informationen schlagen Sie bitte nach im Handbuch „SDF-A“ [\[10\]](#).

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX080-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Hinweise

- Werden Kommandos mit /SKIP-COMMANDS übersprungen, erhält die Exit-Routine erst dann die Steuerung, wenn das Sprungziel-Label erreicht ist.
- Das Sprungziel bei Sprungmarken wird zweimal zur Verfügung gestellt.
- Im Fehlerfall wird die Exit-Routine erst wieder beim „Aufsetz“-Kommando aufgerufen (/SET-JOB-STEP).

3.24.3 SYSSTMT-Exit 082

Dieser Exit hat die gleiche Funktion für Eingaben von SYSSTMT wie der Exit 080 für SYSCMD. Alle Bemerkungen und Hinweise, sowie die technischen Einzelheiten gelten entsprechend!

3.24.4 SYSSTMT-Exit 083

Dieser Exit entspricht dem Exit 081. Alle Aussagen gelten gleichermaßen für beide Exits.

Hinweise

- Register 12, 13 und 14 dürfen von der Exit-Routine nicht zerstört werden.
- Die SYSTMT-Exit-Routine sollte Anweisungen, die über die verschiedenen Wege eingegeben werden, dennoch gleich verarbeiten.
- Wird die Exit-Routine deaktiviert, während eine Anweisung durch mehrere ersetzt wird, wird der Dummy-Kommandosatz `/REMARK SYSCMD EXIT NO MORE ACTIVE DURING PROCESSING` an das Benutzerprogramm gesendet.
- Ein Kommando/eine Anweisung darf nur verändert werden durch ein Kommando oder eine Anweisung, die in der System-Syntaxdatei definiert sind.
- In Prozeduren ist die Variablensubstitution zum Zeitpunkt des Exit-Aufrufs bereits durchgeführt.

DSECT

```

EX080 D
EX080PL DSECT
* -
* -      INPUT  INFORMATION
* -
EX080IND DC    AL1(0)          SOURCE OF COMMANDS
EX080TER EQU    1              -CMD READ FROM A TERMINAL
EX080FIL EQU    2              -CMD READ FROM A JCL FILE
EX080MCL EQU    4              -CMD READ VIA MCLP
EX080TRN EQU    8              -CMD READ VIA TRCMD          999
EX080BUF EQU   16              -CMD FOUND IN BS2000 BUFFER  999
* -
EX080BIT DC    X'00'          BIT_INFO                          900
EX080SDF EQU   X'80'          -INPUT IN SDF STRUCTURED FORM02
EX080SD2 EQU   X'40'          -INPUT IN NEW SDF STRUCT FORM120
EX080STR EQU   X'00'          -INPUT IN STRING MODE          900
* -
EX080UNU DC    2X'00'          900
* -
EX080IR  DC    A(0)           A (INPUT RECORD)
* -
* -      OUTPUT INFORMATION
* -
EX080RC  DC    AL1(0)         SYSCMD EXIT'S RETURN CODE
EX080CAC EQU    0              -CMD REC ACCEPTED
EX080CRJ EQU    4              -CMD REC REJECTED
EX080CRO EQU    8              -CMD REC REPLACED BY ONE
EX080CRS EQU   12              -CMD REC REPLACED BY SEVERAL
EX080CLR EQU   16              -LAST CMD REC REPLACED
EX080CLR  DC   3X'00'         UNUSED IN VER=710
* -
EX080RR  DC    A(0)           A (RETURNED RECORD)
* -
* -      ADDITIONAL INPUT  INFORMATION
* -
EX080ILR DC    A(0)           A (LOWER CASE INPUT RECORD)
EX080STD DC    A(0)           A (STD FORM)
EX080PLL EQU   *-EX080PL     LENGTH OF SYSCMD EXIT  P/L

```

Erklärung der Felder:

EX080IND	Quelle des Kommandos	
	EX080TER	Kommandoeingabe vom Terminal
	EX080FIL	Kommandoeingabe aus einer JCL-Date
	EX080MCL	Kommandoeingabe über MCLP aus einem Benutzerprogramm
	EX080TRN	Kommandoeingabe von TRCMD
	EX080BUF	Kommando intern gelesen

EX080BIT	Indikator für SDF	
	EX080SDF	Eingabe in SDF-Struktur
	EX080STR	Eingabe nicht in SDF-Struktur
EX080IR EX080RC	Adresse des Eingabe-Kommandosatzes oder des normierten Übergabebereichs	
	Returncode des System Exits	
	EX080CAC	Kommando akzeptiert
	EX080CRJ	Kommando abgewiesen
	EX080CRO	Kommando geändert oder ersetzt
	EX080CRS	Kommando wird durch mehrere Kommandos ersetzt
	EX080CLR	letzter Kommandosatz ersetzt
EX080RR	Adresse des Kommandosatzes, der an die rufende Systemkomponente zurückgegeben wird (bei Ändern/Ersetzen)	
EX080ILR	Adresse des Eingabesatzes	
EX080STD	Adresse der Standardeingabe	

Erklärung der Returncodes im Feld EX080RC

Returncode X'04': Kommando zurückgewiesen (EX080RC=EX080CRJ)

Im Dialogbetrieb wird keine Systemmeldung ausgegeben; die Exit-Routine kann jedoch eine Meldung an den Benutzer senden

Bei Kommandoeingabe aus einer katalogisierten Datei wird folgende Meldung ausgegeben:

```
/REMARK COMMAND REJECTED BY THE SYSTEM ADMINISTRATOR
```

Im Stapelbetrieb wird die gleiche Meldung in die Protokolldatei aufgenommen.

Returncode X'08': Kommando ersetzen oder ändern (EX080RC=EX080CRO)

Die Exit-Routine muss das neue oder geänderte Kommando selbst aufbauen bzw. den normierten Übergabebereich versorgen (siehe Handbuch „SDF-A“ [10]). Die Adresse des Kommandosatzes (des normierten Übergabebereichs) wird im Feld EX080RR des Parameterbereichs an die rufende Systemkomponente übergeben.

Aufbau des Kommandosatzes (sofern nicht normierter SDF-Übergabebereich)

0	2	4	SL-1
SL	Kommando mit Operanden		

SL = Satzlänge

Der Kommandosatz darf keinen ISAM-Schlüssel, kein Fortsetzungszeichen und keine symbolischen Parameter (&...) enthalten. Die Satzlänge darf verändert werden.

Im Dialogbetrieb erfolgt keine Systemmeldung.

Bei Kommandoeingabe aus einer katalogisierten Datei wird das neue oder geänderte Kommando ausgegeben bzw. in die Protokolldatei geschrieben.

Bei Kommandoeingabe über MCLP muss das ersetzende Kommando ebenfalls MCLP-zulässig sein, sonst wird der Returncode X'14' (= ungültiges Kommando) an das Benutzerprogramm zurückgegeben. Es darf kein CALL-PROCEDURE-, LOGOFF-, LOAD-PROGRAM- oder START-PROGRAM-Kommando zurückgegeben oder ein anders Kommando über MCLP aufgerufen werden, da dann das Anwenderprogramm sofort entladen wird. – Tritt bei Abarbeitung des neuen Kommandos ein Fehler auf, wird X'10' (unzulässiges Kommando) an das Anwenderprogramm gemeldet.

*Returncode X'0C'/X'10': ein Kommando durch mehrere Kommandos ersetzen
(EX080RC=EX080CRS/EX080CLR)*

Gibt die Exit-Routine einen Kommandosatz an das System zurück (siehe RC = X'08') und setzt gleichzeitig im Feld EX080RC den Returncode auf X'0C' (= EX080CRS), erhält sie sofort die Steuerung zurück und kann einen weiteren Kommandosatz übergeben. Ab dem zweiten Aufruf nach Returncode X'0C', ist die Adresse des Eingabe-Kommandosatzes (Feld EX080IR des Parameterbereichs) auf null gesetzt. Wenn die Exit-Routine das letzte Kommando zurückgibt, muss sie den Returncode im Parameterbereich auf X'10' (=EX080CLR) setzen.

Im Dialogbetrieb erfolgt keine Systemmeldung, wenn ein Kommando durch mehrere ersetzt wurde.

Im Stapelbetrieb werden die ersetzenden Kommandos in die Protokolldatei geschrieben.

Bei Kommandoeingabe über MCLP sind die Hinweise zu beachten, die den Returncode X'08' betreffen. Die o.g. Kommandos CALL-PROCEDURE etc. dürfen auch nicht in einer Kommandofolge zurückgegeben werden, da dann das Anwenderprogramm sofort entladen wird. Wenn im Anwenderprogramm ein Puffer für die Aufnahme der Bearbeitungsergebnisse des MCLP-Kommandos zur Verfügung gestellt wird, muss dieser Puffer die Ergebnisse der gesamten Kommandofolge aufnehmen können.

Es ist zu erwarten, dass viele Anwenderprogramme, die MCLP-Aufrufe benutzen, nicht mehr korrekt ablaufen, wenn die Exit-Routine ein Kommando durch mehrere ersetzt!

3.25 SPOOL-Exits (090/091/092/094/095/096/097)

- Allgemeines zu den SPOOL-Exits
- SPOOL-Exit 090
- SPOOL-Exits (091/092) für Deckblatt und Schlussblatt
- SPOOL-Exit 094 (Ressourcen-Routine)
- SPOOL-Exit 095 (Ausgabe-Routine)
- SPOOL-Exit 096 (Server-Auswahl)
- SPOOL-Exit 097 (Steuern File Transfer)

3.25.1 Allgemeines zu den SPOOL-Exits

Bei Ausgabegeräten kann für jedes Gerät einzeln gesteuert werden, ob SPOOL-Ausgaben auf diesem Gerät durch SPOOL-Exitroutinen bearbeitet werden sollen, siehe Kommando START-PRINTER-OUTPUT.

Aktuelle SPOOL-Version abfragen

Die SPOOL- (und RSO-)Exit-Routinen müssen die Nummer der aktuellen SPOOL-Version im SCB abfragen: Exit-Routinen werden unabhängig von den DSSM-Kommandos START-SUBSYSTEM, STOP-SUBSYSTEM, HOLD-SUBSYSTEM und RESUME-SUBSYSTEM geladen, aktiviert und deaktiviert. Aus diesem Grund kann SPOOL entladen und durch eine andere Version ersetzt werden, während die Exit-Routinen aktiviert bleiben und später von der neu geladenen SPOOL-Version aufgerufen werden.

Die Exit-Routinen müssen diese Änderung der SPOOL-Version erkennen und prüfen daher die im SCB hinterlegte Versionsangabe.

Die Versionsnummer ist im Feld SCBESCB der DSECT enthalten, die mit dem Makro SSVSCB erzeugt werden kann.

Sind die System Exits 90, 91 und 92 gleichzeitig aktiv, entsteht folgende Reihenfolge der Aufrufe:

EXIT 90:	erster Aufruf
EXIT 91:	Routine für Deckblatt
EXIT 90:	Verarbeitungsroutine
EXIT 92:	Routine für Schlussblatt
EXIT 90:	Schlussaufruf.

Der erste Aufruf ist bereits vor Dateieröffnung abgearbeitet. Anschließend erfolgt der Aufbau des Deckblattes. Nach der Erstellung des Schlussblattes erfolgt der Schlusssaufruf, danach das Schließen der Datei.

Unabhängig davon kann der System Exit 95 aktiviert worden sein, um Kanalprogramme zu überwachen, die zu den Druckern gesendet werden.

Bei einer „Family“-Bearbeitung kann entweder nur für die erste Datei oder für jede Datei der „Family“ ein Deckblatt gedruckt werden. Über einen SPSINF-Aufruf (INFO=SPSGEN) in der Exit-Routine können Sie den GEN-Eintrag lesen und so die entsprechenden Informationen erhalten:

Ist im GEN-Eintrag GENFAM=YES gesetzt, wird für jede Datei des „Family“-PRINTs ein Deckblatt gedruckt, andernfalls nur für die erste Datei (d.h. wenn im SCB das Feld SCBESEQ#=0 gesetzt ist).

Mit dem Exit 094 kann die Systembetreuung eine Anwenderroutine erstellen, mit deren Hilfe die Konvertierung von Ressourcen innerhalb einer Client-Server-Umgebung bei Einsatz des Produkts „Distributed Print Services“ gesteuert wird.

Mit dem Exit 096 kann die Systembetreuung bei Einsatz des Produkts „Distributed Print Services“ eine Server-Auswahl durch Kombination bestimmter Auswahl-Kriterien vornehmen.

Mit dem Exit 097 kann die Anzahl der Aufträge des Produkts „Distributed Print Services“ für den File-Transfers gesteuert werden.

Hinweis

Die nachfolgenden Beschreibungen beziehen sich auf das seit SPOOL V3.0 neue Kommando PRINT-DOCUMENT bzw. den neuen Makro PRNTDOC. Die Exit-Funktionen können in gleicher Weise mit dem alten Kommando PRINT-FILE bzw. dem alten Makro PRNT genutzt werden. Nähere Informationen zum SPOOL können dem Handbuch „SPOOL“ [\[13\]](#) entnommen werden. Für Exits, die das Produkt „Distributed Print Services“ betreffen muss zusätzlich das Handbuch „Distributed Print Services“ [\[5\]](#) verwendet werden

3.25.2 SPOOL-Exit 090

Diese Exit-Routine wird aufgerufen, nachdem ein Datensatz der auszugebenden Datei gelesen wurde, aber bevor der Datensatz gemäß den Angaben im PRINT-DOCUMENT vom SPOOL weiterverarbeitet wird.

Der Exit ist verwendbar für die Ausgabe auf Zeilendrucker, Laserdrucker (Seitendrucker). Er kann nicht verwendet werden bei Ausgabe auf Magnetband oder Diskette.

Die Exit-Routine kann folgende Maßnahmen des Systems veranlassen:

- den gelesenen Datensatz akzeptieren (verändert oder unverändert)
- den gelesenen Datensatz ersetzen
- den gelesenen Datensatz übergehen
- den gelesenen Datensatz und alle bis Dateiende folgenden Datensätze übergehen

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX090-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Die Returncodes werden im Standardheader eingetragen:

Standardheader:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX090 wird im Standardheader ein Returncode übergeben:

```
aa=Maincode1
bb=Maincode2
cc=Subcode1
dd=Subcode2
```

X'aa'	Erläuterung
00	Gelesenen Datensatz akzeptieren
04	Gelesenen Datensatz durch Inhalt des Ausgabeblockes ersetzen
08	Gelesenen Datensatz übergehen
0C	Gelesenen Datensatz und alle Folgenden bis Dateiende übergehen

Hinweise

- Die Exit-Routine sollte nicht aktiviert bzw. deaktiviert werden, solange ein SPOOLOUT bearbeitet wird.
- Der Makroaufruf PRNTDOC darf in der Exit-Routine verwendet werden, nicht aber \$PRNT und \$SPRQ.

- Die Exit-Routine erhält die Steuerung bei jedem von der Eingabedatei gelesenen Satz, also auch beim Probedruck der ersten Seite, solange bis der Operator den Druck startet. Das Gleiche gilt für die RESPOOL-Verarbeitung (HOLD-PRINT-JOB-/RESUME-PRINT-JOB-Kommando), wobei die Exit-Routine die Lage des RESPOOL-Beginns berücksichtigen muss.
- Die notwendigen Informationen zur Verarbeitung von Dateien mit Steuerzeichen entnehmen Sie bitte dem PRINT-DOCUMENT-Kommando bzw. dem SPOOL-Control-Block.
- Wenn eigene Ausgabeblocks zur Verfügung gestellt werden, ist eine beliebige Speicherklasse erlaubt. Es empfiehlt sich aber, den Klasse-5-Speicher zu nehmen. Der Speicher wird vom Exit wieder freigegeben, sobald der Exit die Kontrolle wieder zurückbekommen hat.

DSECT

Eine DSECT/CSECT für den Parameterbereich (Adresse wird in Register 1 übergeben) kann mit dem Makro EX090 erzeugt werden, wobei „prefix“ max. 3 Zeichen lang sein darf. name EX090 D/C,[prefix]

Die Eingabefelder werden nach dem Rücksprung von der rufenden SPOOL-Komponente nicht mehr abgeprüft. Der erste und der letzte Aufruf der Exit-Routine erfolgt, ohne dass ein Satz der Datei bereitgestellt wird (ICLTYP = X'02' bzw. X'03'). Das Byte IPRTYP gibt das Ausgabegerät an.

Die Ausgabefelder werden von der Exit-Routine versorgt. Das Byte IEXRC steuert die von der Exit-Routine gewünschte Funktion. Die Funktionen sind im Anschluss an die DSECT näher beschrieben.

```

      EX090 D
* SPOOL INPUT RECORDS ROUTINE EXIT PARAMETER LIST
*
IRC      MFCHK MF=D,DMACID=E90,ALIGN=F,SUPPORT=(C,D),ENTRY=NO,          C
          PREFIX=I,MACID=E90,DNAME=C
IRC      DSECT ,
          *,##### PREFIX=I, MACID=E90 #####
*
* STANDARD HEADER RETURN INFORMATION
*
IBEG     DS      0H
00100510
          FHDR MF=(C,I),EQUATES=NO
00100600
          DS      0A
IFHE     DS      0XL8          0  GENERAL PARAMETER AREA HEADER
*
IIFID    DS      0A          0  INTERFACE IDENTIFIER
IFCTU    DS      AL2          0  FUNCTION UNIT NUMBER
*
*          BIT 15  HEADER FLAG BIT,
*          MUST BE RESET UNTIL FURTHER NOTICE
*          BIT 14-12 UNUSED, MUST BE RESET
*          BIT 11-0  REAL FUNCTION UNIT NUMBER
IFCT     DS      AL1          2  FUNCTION NUMBER
IFCTV    DS      AL1          3  FUNCTION INTERFACE VERSION NUMBER
*
IRET     DS      0A          4  GENERAL RETURN CODE
ISRET    DS      0AL2         4  SUB RETURN CODE
ISR2     DS      AL1          4  SUB RETURN CODE 2
ISR1     DS      AL1          5  SUB RETURN CODE 1
IMRET    DS      0AL2         6  MAIN RETURN CODE
IMR2     DS      AL1          6  MAIN RETURN CODE 2
IMR1     DS      AL1          7  MAIN RETURN CODE 1
IFHL     EQU     8            8  GENERAL OPERAND LIST HEADER LENGTH
*

```

```

*   EQUATES FOR EXIT RETURN INFORMATION
*
IACCP  EQU   0           ACCEPT RECORD AS IT IS
IADD    EQU   4           ADD RECORD(S)
IDEL    EQU   8           DELETE (SKIP) RECORD
IEOF    EQU  12           TERMINATE PRINT (SIMULATE EOF)
*
*   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
IREC@   DS    A           INPUT RECORD ADDRESS
ISCB@   DS    A           SCB ADDRESS
IRCLLEN DS    Y           INPUT RECORD LENGTH
IPRTYP  DS   AL1         TYPE OF PRINT
ILINES  EQU  32          LINES PRINTER TYPE           751
IPAGES  EQU  33          PAGE PRINTER TYPE            751
ILOADL  EQU  34          LOADABLE LINE PRINTER TYPE   751
IRBP    EQU  35          RBP PRINTER TYPE             751
IRSOPB  EQU  36          PUBLIC RSO PRINTER           752
IRSOPR  EQU  37          PRIVATE RSO PRINTER          752
IAPAPR  EQU  38          APA PRINTERS
IPUNCH  EQU  48          PUNCH TYPE                   751
ICLTYP  DS   AL1         CALL TYPE
ICLREC  EQU   1          RECORD CALL TYPE
ICLFST  EQU   2          NO RECORD FIRST TIME CALL
ICLLST  EQU   3          NO RECORD LAST TIME CALL
          DS   CL4         RESERVED FOR EXTENSION
*
*   OUTPUT FIELDS FROM THE INPUT EXIT ROUTINE
*
IBLK@   DS    A           DATA BLOCK ADDRESS
IBLKLE  DS    Y           DATA BLOCK LENGTH
IEXRC   DS   AL1         R-C FROM EXIT ROUTINE MAY BE
*
*                               STORED HERE IN ADDITION TO
*                               THE MAIN RETURN CODE
IBEXRC  DS   AL1         BASE R-C FROM EXIT ROUTINE
*
IBRCOK  EQU   0          BASE R-C O.K.
INEXS   EQU   4          EXIT ROUTINE NOT EXISTENT
INACT   EQU   8          EXIT ROUTINE NOT ACTIVE
IPARER  EQU  12          PARAMETER ERROR
          DS   CL8         RESERVED FOR EXTENSION
*
*   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
IPIR    DS    A           PRINTER INFORMATION RECORD ADDR.  755
IDVTYP  DS   AL1         DEVICE TYPE                       755
          DS   CL11        RESERVED FOR EXTENTION           755
ILEN    EQU  *-IRC        ROUTINE EXIT P/L LENGTH

```

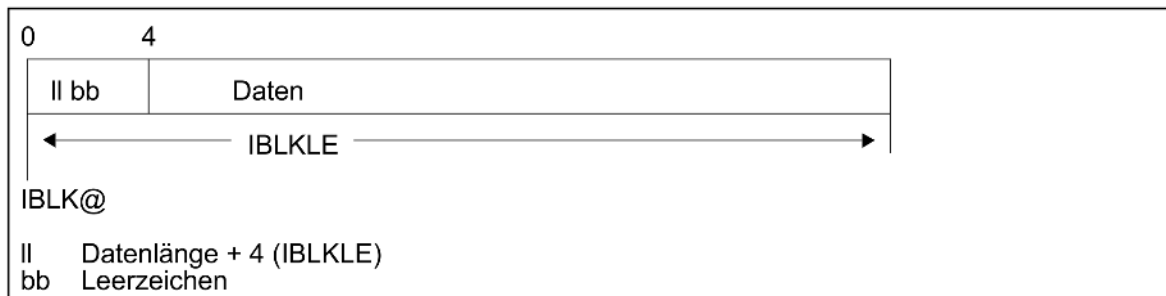
Funktionen der Exit-Routine

Returncode X'00': gelesenen Satz akzeptieren

Die Adresse des Satzes und die Satzlänge werden in den Feldern IREC@ bzw. IRCLLEN an die Exit-Routine übergeben. Der Satz enthält kein Satzlängenfeld.

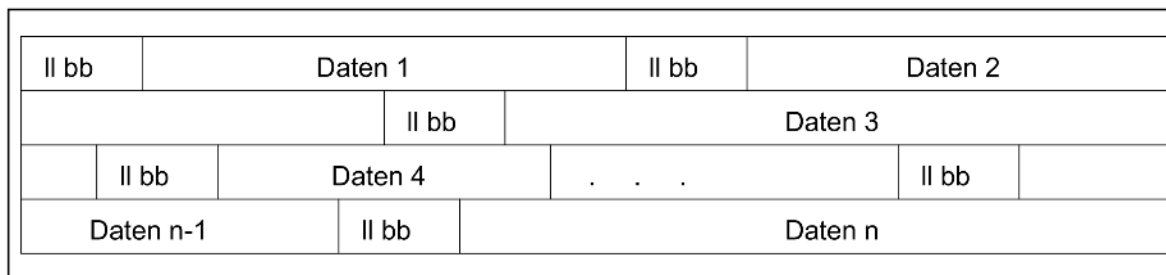
Die Exit-Routine kann den Satz unverändert vom SPOOL akzeptieren lassen oder ihn ändern. Wenn der Satz dabei nicht länger wird, kann er direkt im Eingabepuffer verändert werden. Die Felder IBLK@ und IBLKLE müssen in diesem Fall mit der Adresse des Satzes aus IREC@ und der gültigen Satzlänge versorgt werden. Wird der Satz durch die Änderung verlängert, muss die Exit-Routine den Satz in einem eigenen, über IBLK@ adressierten Puffer bereitstellen.

Aufbau des Puffers:



Returncode X'04': gelesenen Satz ersetzen

SPOOL ersetzt den eingelesenen Satz durch die Sätze, die von der Exit-Routine im Ausgabeblock (Adresse in IBLK@) bereitgestellt wurden:



Die Exit-Routine muss die Felder IBLK@ und IBLKLE versorgen. Soll der gelesene Satz Bestandteil des Ausgabeblocks sein, muss ihn die Exit-Routine dorthin übertragen.

Returncode X'08': gelesenen Satz übergehen

Die Exit-Routine muss nur den Returncode setzen, weitere Aktionen sind nicht erforderlich.

Returncode X'0C': alle Sätze bis Dateiende übergehen

Die Exit-Routine muss nur den Returncode setzen, weitere Aktionen sind nicht erforderlich.

Operanden im PRINT-DOCUMENT-Kommando/PRNTDOC-Makro

Die Exit-Routine muss alle in PRINT-DOCUMENT-Kommandos angegebenen Operanden beachten! Die entsprechenden Werte können dem SPOOL-Kontrollblock entnommen werden.

*RECORD-PART=*PARAMETERS(FIRST-CHARACTER = n):*

Alle von der Exit-Routine eingefügten Sätze müssen länger als n Bytes sein, sonst werden sie nicht gedruckt.

*OUTPUT-PART=*RANGE(FROM = +n):*

Die Sätze 1–n werden ebenfalls an die Exit-Routine übergeben.

*OUTPUT-PART=*LAST(LAST = -n):*

Die Exit-Routine erhält die Steuerung zunächst für alle Sätze von 1–m (Dateiende, keine Ausgabe). Danach setzt SPOOL den FROM-Wert auf $z = (m-n+1)$ und arbeitet die Datei ab wie bei FROM = +z.

ADDITIONAL-COPIES = n:

Die Exit-Routine kann bei wiederholter Ausgabe einer Datei nicht zwischen den einzelnen Kopien unterscheiden, d. h. es gibt keinen Unterschied in den übergebenen Parametern oder Datensätzen.

3.25.3 SPOOL-Exits (091/092) für Deckblatt und Schlussblatt

Mit diesen System Exits wird die Gestaltung der Deckblätter / Schlussblätter bei Ausgaben auf Drucker durch Exit-Routinen gesteuert.

Die Exit-Routinen können dabei die Standardausgabe des SPOOL-Systems unterdrücken, durch eine Alternativausgabe ersetzen oder zusätzlich zur Standardausgabe eine Alternativausgabe veranlassen.

Alternativausgaben für Deck- oder Schlussblätter werden von der Systembetreuung zur Verfügung gestellt und vom Anwender mit den Operanden HEADER-EXIT-NUMBER und TRAILER-EXIT-NUMBER im Kommando PRINT-DOCUMENT angefordert. Die Standardausgabe kann unabhängig von einer EXIT-Routine durch einen Text (Operand HEADER-PAGE-TEXT im Kommando PRINT-DOCUMENT) modifiziert werden.

Beispiel

Festgelegt werden muss ein Standardname, z.B.

```
HEADERPAGETEXT.n,
```

der eine SAM-, ISAM-Datei oder eine Jobvariable bezeichnet. Darin können Sie Texte hinterlegen, die Sie mit dem Kommando

```
PRINT-DOCUMENT . . . ,LAYOUT-CONTROL=*PARAMETERS ( COVER-PAGES= -
    *PARAMETERS (HEADER-EXIT-NUMBER= . . . ) )
```

ansprechen können.

Die Exit-Routine muss auf die Datei \$userid.HEADERPAGETEXT.n zugreifen und den Text in den Puffer für das Deckblatt eintragen. Eine Auswahl erfolgt erst, wenn die Exit-Routine aufgerufen ist. Diese Exit-Routinen sichern und stellen alle Register wieder her.

Die Routinen werden aufgerufen:

- wenn kein Deck- bzw. Schlussblatt erzeugt wurde
- bei bestehendem Deck- bzw. Schlussblatt, um den Verwaltungspuffer wieder aufzubauen

Deckblatt-Exit (091)

Folgende Informationen werden an die EXIT-Routine übergeben:

```
R1 = A(EX091-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Die Returncodes werden im Standardheader eingetragen:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX091 und EX092 wird im Standardheader ein Returncode übergeben:

aa=Maincode1
bb=Maincode2
cc=Subcode1
dd=Subcode2

X'aa'	Erläuterung
00	Kein Deck- bzw. Schlussblatt
01	Ohne Änderung Systemdeck- bzw. -schlussblatt drucken
02	Angegebenes alternatives Deck- bzw. Schlussblatt drucken
03	Das Systemdeck- bzw. Schlussblatt ohne Änderung und das angegebene alternative Deck- bzw. Schlussblatt drucken
04	Nur für Exit 92. Kein Schlussblatt. Der letzte Vorschub auf den Vertikaltabulator „Kanal 1“ soll nicht durchgeführt werden

DSECT

Eine DSECT/CSECT für den Parameterbereich (Adresse wird in Register 1 übergeben) kann mit dem Makro EX091 erzeugt werden, wobei „prefix“ max. 3 Zeichen lang sein darf.

```
EX091 D
*
* SPOOL HEADER PAGE ROUTINE EXIT PARAMETER LIST
*
IHD MFCHK MF=D,DMACID=E91,ALIGN=F,SUPPORT=(C,D),ENTRY=NO, C
    PREFIX=I,MACID=E91,DNAME=C
IHD DSECT ,
    *,##### PREFIX=I, MACID=E91 #####
*
* STANDARD HEADER RETURN INFORMATION
*
IBEG DS 0H
    FHDR MF=(C,I),EQUATES=NO
    DS 0A
IFHE DS 0XL8 0 GENERAL PARAMETER AREA HEADER
*
IIFID DS 0A 0 INTERFACE IDENTIFIER
IFCTU DS AL2 0 FUNCTION UNIT NUMBER
*
* BIT 15 HEADER FLAG BIT,
* MUST BE RESET UNTIL FURTHER NOTICE
* BIT 14-12 UNUSED, MUST BE RESET
* BIT 11-0 REAL FUNCTION UNIT NUMBER
IFCT DS AL1 2 FUNCTION NUMBER
IFCTV DS AL1 3 FUNCTION INTERFACE VERSION NUMBER
*
IRET DS 0A 4 GENERAL RETURN CODE
ISRET DS 0AL2 4 SUB RETURN CODE
ISR2 DS AL1 4 SUB RETURN CODE 2
ISR1 DS AL1 5 SUB RETURN CODE 1
IMRET DS 0AL2 6 MAIN RETURN CODE
IMR2 DS AL1 6 MAIN RETURN CODE 2
IMR1 DS AL1 7 MAIN RETURN CODE 1
IFHL EQU 8 8 GENERAL OPERAND LIST HEADER LENGTH
*
```

```

*
*   EQUATES FOR EXIT RETURN INFORMATION IN &P.MR1
*
IEXNH   EQU   0           NO HEADER PAGE OUTPUT
IEXSY   EQU   1           SYSTEM HEADER UNALTERED
IEXAL   EQU   2           ALTERNATIVE HEADER
IEXBO   EQU   3           SYSTEM AND ALTERNATIVE HEADER
*
*   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
IPRTYP  DS     C           PRINTER TYPE
ILINES  EQU   X'20'       LINES PRINTER TYPE
IPAGES  EQU   X'21'       PAGE PRINTER TYPE
ILOADL  EQU   X'22'       LOADABLE LINE PRINTER TYPE      753
IRBP    EQU   X'23'       RBP PRINTER TYPE                753
IRSOPB  EQU   X'24'       PUBLIC RSO PRINTER              753
IRSOPR  EQU   X'25'       PRIVATE RSO PRINTER            753
IAPAPR  EQU   X'26'       APA PRINTER
IPUNCH  EQU   X'30'       PUNCH TYPE                      753
ICLTYP  DS     C           CALL TYPE
INORM   EQU   X'01'       NORMAL CALL                    754
IRETRY  EQU   X'02'       RETRY                          754
INOBUF  EQU   X'04'       NO BUFFER                      754
ISMLHD  EQU   X'08'       SMALL HEADER                   802
        DS     CL2       RESERVED
ISCB@   DS     A           SCB ADDRESS
ISYIB@  DS     A           SYSTEM BUFFER
ILNPG   DS     F           # OF LINES PER PAGE
ICHLN   DS     F           # OF CHAR. PER LINE
*
*   OUTPUT FIELDS FROM THE INPUT EXIT ROUTINE
*
IUSOB@  DS     A           USER BUFFER ADDRESS
IUSOBL  DS     F           USER BUFFER LENGTH
ISKIP#  DS     F           # SKIP RECORDS
ISYDF   DS     F           SYSTEM DUPL FACT
IUSDF   DS     F           USER DUPL FACT
        DS     CL12      RESERVED FOR EXTENSION
*
*   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
IPIR    DS     A           PRINTER INFORMATION RECORD ADDR. 756
IDVTYP  DS     AL1        DEVICE TYPE                      756
        DS     XL1        HW ALIGNMENT
IFREQ#  DS     H           SEPARATOR PAGES FREQUENCE #      865
*       DS     H           FREQ# IS INPUT / OUTPUT          865
IFWFR@  DS     A           ADDR OF 10 FREE FW FOR EXIT 091  865
        DS     2F         RESERVED - FREE FOR USE          865
ILEN    EQU   *-IHD       ROUTINE EXIT P/L LENGTH

```

Vor Aufruf der Exit-Routine versorgte Felder:

IPRTYP	Ausgabegerät	
	ILINES	Zeilendrucker
	IPAGES	Laserdrucker / Seitendrucker
	ILOADL	Drucker mit ladbarem VFB
	IRBP	RBP-Drucker
	IRSOPB	gemeinschaftlicher RSO-Drucker
	IRSOPR	privater RSO-Drucker
	IAPAPR	APA-Drucker
ICLTYP	Art des Aufrufs	
	INORM	normaler Aufruf
	IRETRY	wiederholter Aufruf: wenn der Speicherbereich für das Alternativdeckblatt zerstört oder nicht korrekt aufgebaut ist
	INOBUF	kein System-Standarddeckblatt: es wird keine Pufferadresse ISYIB@ übergeben; die Exit-Routine kann eine Alternativausgabe veranlassen, entsprechend ILNPG und ICHLN (s.u.)
	ISMLHD	kurzes Deckblatt (nach einer Jobunterbrechung)
ISCB@	Adresse des SPOOL-Kontrollblocks (SCB)	
ISYIB@	Adresse des Systempuffers	
ILNPG	Anzahl Zeilen pro Deckblatt-Seite	
ICHLN	Anzahl Zeichen pro Deckblatt-Zeile	
Felder, die von der Exit-Routine versorgt werden können:		
IUSOB@	Klasse-3-Speicheradresse, an der das Alternativ-Deckblatt bereitgestellt ist	
IUSOBL	Länge des Puffers für das Alternativdeckblatt, entsprechend dem von der Exit-Routine mit \$GETMEM angeforderten Pufferbereich	
ISKIP#	Anzahl Sätze des Alternativdeckblatts, die bei einer Wiederholung übergangen, d. h. nicht gedruckt werden sollen	
ISYDF	Wiederholungsfaktor für die Ausgabe des Standarddeckblatts; Standard: 1	

IUSDF	Wiederholungsfaktor für die Ausgabe des Alternativdeckblatts; Standard: 1
IPIR	Adresse des speziellen Drucker-Informationen-Satzes. Für genauere Informationen wird die Hardware-Beschreibung benötigt. (Unterstützung zurzeit nur für LP65-Drucker)
IDVTYP	Gerätetyp Genauere Informationen dazu siehe SPOOL-Makro SPSINF.

Schlussblatt-Exit (092)

Folgende Informationen werden an die Exit-Routine übergeben:

```
R1 = A(EX092-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

DSECT

Eine DSECT/CSECT für den Parameterbereich (Adresse in R1) kann mit dem Makro EX092 erzeugt werden, wobei „prefix“ max. 3 Zeichen lang sein darf.

```
EX092 D
*
* SPOOL TRAILER PAGE ROUTINE EXIT PARAMETER LIST
*
ITR      MFCHK MF=D,DMACID=E92,ALIGN=F,SUPPORT=(C,D),ENTRY=NO,          C
        PREFIX=I,MACID=E92,DNAME=C
ITR      DSECT ,
        *,##### PREFIX=I, MACID=E92 #####
*
* STANDARD HEADER RETURN INFORMATION
*
IBEG     DS      0H
        FHDR MF=(C,I),EQUATES=NO
        DS      0A
IFHE     DS      0XL8          0   GENERAL PARAMETER AREA HEADER
*
IIFID    DS      0A          0   INTERFACE IDENTIFIER
IFCTU    DS      AL2          0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*                               BIT 14-12 UNUSED, MUST BE RESET
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
IFCT     DS      AL1          2   FUNCTION NUMBER
IFCTV    DS      AL1          3   FUNCTION INTERFACE VERSION NUMBER
*
IRET     DS      0A          4   GENERAL RETURN CODE
ISRET    DS      0AL2         4   SUB RETURN CODE
ISR2     DS      AL1          4   SUB RETURN CODE 2
ISR1     DS      AL1          5   SUB RETURN CODE 1
IMRET    DS      0AL2         6   MAIN RETURN CODE
IMR2     DS      AL1          6   MAIN RETURN CODE 2
IMR1     DS      AL1          7   MAIN RETURN CODE 1
IFHL     EQU     8            8   GENERAL OPERAND LIST HEADER LENGTH
```

```

*
*
*   EQUATES FOR EXIT RETURN INFORMATION IN &P.MR1
*
IEXNT   EQU   0           NO TRAIL. PAGE OUTPUT
IEXSY   EQU   1           SYSTEM TRAIL. UNALTERED
IEXAL   EQU   2           ALTERNATIVE TRAILER PAGE
IEXBO   EQU   3           SYSTEM AND ALTERNATIVE TRAILER
*   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
IPRTYP  DS     C           PRINTER TYPE
ILINES  EQU   X'20'       LINES PRINTER TYPE
IPAGES  EQU   X'21'       PAGE PRINTER TYPE           753
ILOADL  EQU   X'22'       LOADABLE LINE PRINTER TYPE  753
IRBP    EQU   X'23'       RBP PRINTER TYPE           753
IRSOPB  EQU   X'24'       PUBLIC RSO PRINTER         753
IRSOPR  EQU   X'25'       PRIVATE RSO PRINTER        753
IAPAPR  EQU   X'26'       APA PRINTERS
IPUNCH  EQU   X'30'       PUNCH TYPE                 753
ICLTYP  DS     C           CALL TYPE
INORM   EQU   X'01'       NORMAL CALL                754
IRETRY  EQU   X'02'       RETRY                      754
INOBUF  EQU   X'04'       NO BUFFER                 754
        DS     CL2       RESERVED
ISCB@   DS     A           SCB ADDRESS
ISYIB@  DS     A           SYSTEM BUFFER
ILNPG   DS     F           # OF LINES PER PAGE
ICHLN   DS     F           # OF CHAR. PER LINE
*
*   OUTPUT FIELDS FROM THE INPUT EXIT ROUTINE
*
IUSOB@  DS     A           USER BUFFER ADDRESS
IUSOBL  DS     F           USER BUFFER LENGTH
ISKIP#  DS     F           # SKIP RECORDS
ISYDF   DS     F           SYSTEM DUPL FACT
IUSDF   DS     F           USER DUPL FACT
        DS     CL12      RESERVED FOR EXTENSION
*
*   INPUT FIELDS FOR THE INPUT EXIT ROUTINE
*
IPIR    DS     A           PRINTER INFORMATION RECORD ADDR. 756
IDVTYP  DS     AL1        DEVICE TYPE                 756
        DS     CL11      RESERVED FOR EXTENTION      756
ILEN    EQU   *-ITR      ROUTINE EXIT P/L LENGTH

```

Vor Aufruf der Exit-Routine versorgte Felder:

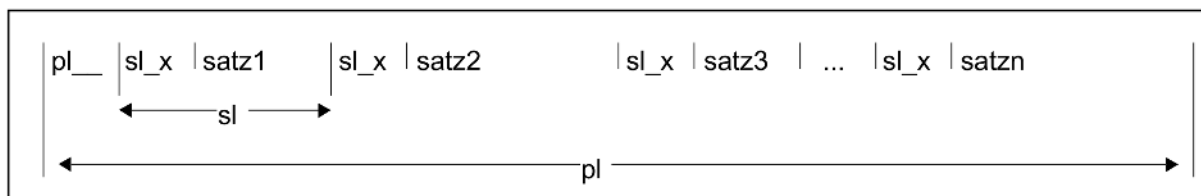
YDDXMAIN	Indikator für das Hauptereignis YDDXAPPL Hauptereignis ' APP' (betr. Anwendungen)	
IPRTYP	Ausgabegerät	
	ILINES	Zeilendrucker
	IPAGES	Laserdrucker / Seitendrucker
	ILOADL	Drucker mit ladbarem VFB
	IRBP	RBP-Drucker
	IRSOPB	gemeinschaftlicher RSO-Drucker
	IRSOPR	privater RSO-Drucker
	IAPAPR	APA-Drucker
ICLTYP	Art des Aufrufs	
	INORM	normaler Aufruf
	IRETRY	wiederholter Aufruf: wenn der Speicherbereich für das Alternativschlussblatt zerstört oder nicht korrekt aufgebaut ist
	INOBUF	kein System-Standardschlussblatt: es wird keine Pufferadresse ISYIB@ übergeben; die Exit-Routine kann eine Alternativausgabe veranlassen, entsprechend ILNPG und ICHLN (s.u.)
	ISMLHD	kurzes Schlussblatt (nach einer Jobunterbrechung)
ISCB@	Adresse des SPOOL-Kontrollblocks (SCB)	
ISYIB@	Adresse des Systempuffers	
ILNPG	Anzahl Zeilen pro Schlussblatt-Seite	
ICHLN	Anzahl Zeichen pro Schlussblatt-Zeile	
Felder, die von der Exit-Routine versorgt werden können:		
IUSOB@	Klasse-3-Speicheradresse, an der das Alternativ-Schlussblatt bereitgestellt ist	
IUSOBL	Länge des Puffers für das Alternativschlussblatt, entsprechend dem von der Exit-Routine mit \$GETMEM angeforderten Pufferbereich	
ISKIP#	Anzahl Sätze des Alternativschlussblatts, die bei einer Wiederholung übergangen, d.h. nicht gedruckt werden sollen	

ISYDF	Wiederholungsfaktor für die Ausgabe des Standardschlussblatts; Standard: 1
IUSDF	Wiederholungsfaktor für die Ausgabe des Alternativschlussblatts; Standard: 1
IPIR	Adresse des speziellen Drucker-Informationen-Satzes. Für genauere Informationen wird die Hardware-Beschreibung benötigt. (Unterstützung zurzeit nur für LP65-Drucker)
IDVTYP	Gerätetyp Genauere Informationen dazu siehe SPOOL-Makro SPSINF.

Aufbau der Puffer für Deck-/Schlussblätter

Alternativ-Ausgabe

Für den Aufbau von Alternativ-Ausgaben muss die Exit-Routine einen Puffer im Klasse-3-Speicher reservieren. Die Deck-/Schlussblätter werden aus Sätzen variabler Länge aufgebaut, d.h. mit vorangestelltem 4-Byte-Satzlängenfeld. Die ersten 4 Bytes des Puffer enthalten linksbündig die genutzte Pufferlänge, höchstens 32767 Bytes = X'7FFF'.



pl: genutzte Pufferlänge

sl: Satzlänge/Zeilenlänge + 4

_.: Leerzeichen (X' 40')

x: Kennzeichen für Ausgabotyp (nur SPOOL-intern ausgewertet)

X' 0A' Drucker

satz: bei PRINT-Ausgabe: 1 Byte Drucksteuerzeichen (s. unten) + Text

X' 40': Kein Vorschub vor dem Drucken

X' 41': 1 Zeile Vorschub vor dem Drucken

X' 42': 2 Zeilen Vorschub vor dem Drucken

X' 43': 3 Zeilen Vorschub vor dem Drucken

Andere Steuerzeichen können zu unerwünschten Ergebnissen führen, da sie nicht von jedem Gerätetyp verstanden werden.

Systemdeckblatt

Das Deckblattformat ist abhängig von Papiergröße und verwendetem Zeichensatz. Wieviele Zeilen auf eine Druckseite ausgegeben werden können, lässt sich folgendermaßen bestimmen:

Max.Zeilenzahl = Papierformat in Zoll * Zeilen / Zoll

Ausnahme: Wird bei Laserdruckern das Deckblatt unter LOOP-Steuerung ausgedruckt, ist die Zeilenzahl durch den VFB vorgegeben!

Die Zeilen des Deckblatts werden entsprechend der unten beschriebenen Rangfolge gedruckt (abhängig von der maximalen Zeilenzahl).

Rangfolge

Adress- und Identifikationsfeld (24 Zeilen)	
Benutzertext (10 Zeilen) + 2 Leerzeilen	in „Riesen“-Lettern
Auftragsname (10 Zeilen) + 2 Leerzeilen	in „Riesen“-Lettern
Benutzerkennung (10 Zeilen) + 2 Leerzeilen	in „Riesen“-Lettern
Abrechnungsnummer (10 Zeilen) + 2 Leerzeilen	in „Riesen“-Lettern

Stehen weniger als 24 Zeilen zur Verfügung, wird kein Standard-Deckblatt gedruckt. Die Exit-Routine kann jedoch geeignete Maßnahmen treffen.

Jede gedruckte Zeile wird als Satz variabler Länge gespeichert, d.h. mit einem vorangestellten Satzlängenfeld (4 Bytes). Leerzeilen enthalten ein Leerzeichen (X'40') als Satzinhalt.

Beispiel

Aufbau eines Standard-Deckblatts, abhängig von Zeichen- und Zeilendichte

Format (Seitengröße): 9 x 11 Zoll

Zeichensatz A: 15 Zeichen/Zoll; 8 Zeilen/Zoll (= 15 cpi / 8 lpi)

Zeichensatz B: 12 Zeichen/Zoll; 6 Zeilen/Zoll (= 12 cpi / 6 lpi)

	Zeichensatz A (15 cpi / 8 lpi)	Zeichensatz B (12 cpi / 6 lpi)
Anzahl Zeichen pro Zeile	11 * 15 = 165	11 * 12 = 132
Anzahl Zeilen pro Seite	9 * 8 = 72	9 * 6 = 54
Deckblatt-Aufbau		
Adressfeld (24 Zeilen)	freie Zeilen: 48	freie Zeilen: 30
Auftragsname (10 + 2 Zeilen)	freie Zeilen: 36	freie Zeilen: 18
Benutzerkennung (10 + 2 Zeilen)	freie Zeilen: 24	freie Zeilen: 6
Abrechnungsnummer (10 + 2 Zeilen)	freie Zeilen: 12	freie Zeilen: -
Bemerkung	Deckblatt wird vollständig ausgedruckt	Abrechnungsnummer fehlt
Länge der Sternchenzeilen	165 '*'	132 '*'

Papiervorschub

Nach Drucken des Systemschlussblatts wird das Folgeblatt auf die erste Zeile positioniert. Das gilt auch, wenn kein Deck- oder Schlussblatt gedruckt wird.

Anwendungshinweise

Die sinnvolle Verwendung der obenbeschriebenen Exits setzt geeignete Vereinbarungen zwischen Systembetreuung und Kunden voraus, z.B. einen Standardnamen für eine SAM- oder ISAM-Datei (oder auch eine Jobvariable). In dieser Datei kann sowohl der privilegierte als auch der nichtprivilegierte Benutzer ein Alternativ-Deckblatt aufbauen.

Beispiel für den Aufbau eines Standardnamens:

```
HEADERPAGETEXT.n, 0 <= n <= 2147.483.639
```

Das Alternativdeckblatt kann dann ausgedruckt werden mit PRINT-DOCUMENT-Kommando (Operand HEADER-EXIT-NUMBER = n) oder PRNTDOC-Makro (Operand COVPAGE).

Die Exit-Routine muss dann auf die Datei \$userid.HEADERPAGETEXT.n zugreifen und den Deckblatt-Puffer mit dem Alternativtext versorgen.

Ob ein Deck-/Schlussblatt gedruckt wird, hängt ab von der Anzahl der Zeichen, die pro Zeile gedruckt werden können.

Wenn bei den Exits 091/092 eigene Ausgabeblöcke zur Verfügung gestellt werden, ist der Klasse-3-Speicher erforderlich. Der Speicher wird von SPOOL, nicht von den Exits wieder freigegeben.

3.25.4 SPOOL-Exit 094 (Ressourcen-Routine)

Dieser SPOOL-Exit ermöglicht es der Systembetreuung, eine Anwenderroutine zu erstellen, die zwei Funktionen innerhalb einer Client-Server-Umgebung bei Einsatz des Produkts „Distributed Print Services“ erfüllen soll.

Konvertierung von Client-Ressourcen auf Server-Ressourcen (Funktion F0)

Diese Funktion modifiziert die Werte einiger Felder des SCB (z.B. FORM, DIA).

Die Eingabe besteht aus dem Standardheader, der Adresse des SCB und dem Funktionscode X'F0' oder C'0' im Feld E94INFOR.

Die Ausgabe besteht aus dem Returncode im Standardheader und dem modifizierten SCB.

Konvertierung von Server-Ressourcen auf Client-Ressourcen (Funktion F1)

Diese Funktion stellt die Originalwerte der Felder des SCB (z.B. FORM, DIA) wieder her. Die Eingabe besteht aus dem Standardheader, der Adresse des SCB und dem Funktionscode X'F1' oder C'1' im Feld E94INFOR.

Die Ausgabe besteht aus dem Returncode im Standardheader und dem modifizierten SCB.

Der Exit 094 muss auf dem Client- und dem Server-Rechner gestartet werden. Anderenfalls wird der SPOOLOUT-Auftrag abgewiesen mit der Meldung

```
% SCP1089 EXIT#094 FUER TSN '(&00)' OBLIGATORISCH. KOMMANDO ABGEWIESEN
```

Der Einsatz der Exit-Routine mit dem Funktionscode F1 zur Wiederherstellung des Originalnamens im SCB ist optional. Lediglich der Returncode muss gesetzt werden. Die vom Server durchgeführte Übersetzung ist jedoch auch in diesem Fall transparent für den Anwender am Client-Rechner. Bei allen SHOW-PRINT-JOB-Kommandos ist die Ausgabe entsprechend modifiziert. In jedem Fall werden dem Cluster-Verwalter die übersetzten Werte angezeigt.

Als Voreinstellung wird das Schlussblatt mit den Werten des Servers ausgedruckt. Will der Anwender sein Schlussblatt mit den Werten des Clients ausdrucken, muss er einen entsprechenden Trailer-Exit mit Aufruf des Exits 094 erstellen.

Tritt während der Verarbeitung des Funktionsaufrufs F0 ein Fehler auf (Returncode ungleich 0 im Standardheader), wird der SPOOLOUT-Auftrag mit der Fehlermeldung SCP1090 abgewiesen.

Tritt während der Verarbeitung des Funktionsaufrufs F1 ein Fehler auf (Returncode ungleich 0 im Standardheader), wird eine Fehlermeldung an der Bedienstation ausgegeben, der SPOOLOUT-Auftrag jedoch fortgesetzt.

Folgende Informationen werden an die Exit-Routine übergeben:

R1 = A (EX094-Parameterbereich)

Die Returncodes werden im Standardheader eingetragen:



Über die Ausführung von EX094 wird im Standardheader ein Returncode übergeben:

aa=Maincode1

bb=Maincode2

cc=Subcode1

dd=Subcode2

X'aa'	Erläuterung
00	Exit fehlerfrei ausgeführt
01	Exit nicht fehlerfrei ausgeführt

DSECT

Eine DSECT/CSECT für den Parameterbereich kann mit dem Makro EX094 erzeugt werden, wobei „prefix“ max. 3 Zeichen lang sein darf.

```

EX094 D
CLREXPL DSECT ,          RENAMING OF CLIENT RESOURCE EXIT P/L
E94EXBEG DS    0H
          FHDR MF=(C,E94),EQUATES=NO
          DS    0A
E94FHE   DS    0XL8          0   GENERAL PARAMETER AREA HEADER
*
E94IFID  DS    0A          0   INTERFACE IDENTIFIER
E94FCTU  DS    AL2          0   FUNCTION UNIT NUMBER
*
*          BIT 15   HEADER FLAG BIT,
*          MUST BE RESET UNTIL FURTHER NOTICE
*          BIT 14-12 UNUSED, MUST BE RESET
*          BIT 11-0  REAL FUNCTION UNIT NUMBER
E94FCT   DS    AL1          2   FUNCTION NUMBER
E94FCTV  DS    AL1          3   FUNCTION INTERFACE VERSION NUMBER
*
E94RET   DS    0A          4   GENERAL RETURN CODE
E94SRET  DS    0AL2         4   SUB RETURN CODE
E94SR2   DS    AL1          4   SUB RETURN CODE 2
E94SR1   DS    AL1          5   SUB RETURN CODE 1
E94MRET  DS    0AL2         6   MAIN RETURN CODE
E94MR2   DS    AL1          6   MAIN RETURN CODE 2
E94MR1   DS    AL1          7   MAIN RETURN CODE 1
E94FHL   EQU   8           8   GENERAL OPERAND LIST HEADER LENGTH
*
*
*   EQUATES FOR EXIT RETURN INFORMATION IN &P.RET
*
E94EXNOK EQU   1           EXIT NOK
E94EXOK  EQU   0           EXIT OK
E94INFOR DC   X'00'       INFORMATION GIVEN TO EXIT
E94CLTSV EQU   C'0'       TRANS. CLIENT RESSOURCE TO SERV
E94SVTCL EQU   C'1'       TRANS. SERVER RESSOURCE TO CLI.
E94CALCL EQU   C'2'       TRANS. CALL ON CLIENT
E94UNUS  DC   XL3'00'     ****  UNUSED  ****
E94SCBAD DC   A(0)        ADDRESS OF SCB
          DS    CL20        RESERVED FOR EXTENTION
E94PLEN  EQU   *-E94EXBEG LENGTH OF THE PARAMETER LIST

```

3.25.5 SPOOL-Exit 095 (Ausgabe-Routine)

Dieser SPOOL-Exit ermöglicht es der Systembetreuung, eine Anwenderroutine zu erstellen, die bei jeder Ausgabeoperation des SPOOL aufgerufen wird. Nur bei der Fehlerbehandlung erfolgt kein Aufruf. Die Exit-Routine wird aufgerufen, nachdem ein Kanalprogramm erstellt wurde und bevor dieses zum Drucker gesendet wird. Die Exit-Routine kann für Zeilen- und Laserdrucker verwendet werden. Die APA-Drucker, RSO, Remote-Batch-Verarbeitung, Magnetbänder und Floppy Disk werden nicht unterstützt.

Für die Exit-Routine bestehen folgende Möglichkeiten:

- das Kanalprogramm zu akzeptieren
- das Kanalprogramm zu verändern
- das Kanalprogramm zu unterdrücken.

Folgende Informationen werden an die Exit-Routine übergeben:

R1 = A (EX095-Parameterbereich)

Die Returncodes werden im Standardheader eingetragen:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX095 wird im Standardheader ein Returncode übergeben:

aa=Maincode1

bb=Maincode2

cc=Subcode1

dd=Subcode2

X'aa'	Erläuterung
00	Kanalprogramm ausgegeben
04	Kanalprogramm nicht ausgegeben
0C	Kanalprogramm nicht ausgegeben und EOF

Hinweise

- Die Exit-Routine hat Zugriff zu den SPOOL-eigenen Datenstrukturen. Diese Datenstrukturen sind versionsabhängig.
- Wird das Kanalprogramm verändert oder nicht ausgegeben, erfolgt keine Berichtigung des SPOOL-Abrechnungssatzes.
- Für die von der Exit-Routine übergebenen Daten wird keine „Code-Translation“ durchgeführt.
- Wird das Kanalprogramm verändert, sind folgende Regeln zu beachten:

- Das Kanalprogramm und der Datenpuffer dürfen nicht erweitert werden.
 - Werden die Daten, die von einem Kanalbefehl adressiert sind verkürzt, muss auch die Datenlängenangabe des Kanalbefehls korrigiert werden.
 - Wird das Kanalprogramm verkürzt, muss auch der „Kettenende-Indikator“ korrigiert werden.
 - Die Datenadresse kann verändert werden, muss aber immer im SPOOL-Ausgabebereich liegen.
- Die Adresse „Beginn Ausgabebereich“ ist im ersten Kanalbefehl der Kette enthalten. Die Adresse „Ende Ausgabebereich“ ergibt sich aus der Datenlänge plus Datenadresse des letzten Kanalbefehls.

DSECT

Eine DSECT/CSECT für den Parameterbereich kann mit dem Makro EX095 erzeugt werden, wobei „prefix“ max. 3 Zeichen lang sein darf.

```

          EX095 D
RECEXPL DSECT ,           I/O   EXIT ROUTINE P/L
*
*   STANDARD HEADER RETURN INFORMATION
*
SPOEXBEG DS    0H
          FHDR MF=(C,SPO),EQUATES=NO
          DS    0A
SPOFHE   DS    0XL8           0   GENERAL PARAMETER AREA HEADER
*
SPOIFID  DS    0A           0   INTERFACE IDENTIFIER
SPOFCTU  DS    AL2           0   FUNCTION UNIT NUMBER
*
*           BIT 15   HEADER FLAG BIT,
*           MUST BE RESET UNTIL FURTHER NOTICE
*           BIT 14-12 UNUSED, MUST BE RESET
*           BIT 11-0 REAL FUNCTION UNIT NUMBER
SPOFCT   DS    AL1           2   FUNCTION NUMBER
SPOFCTV  DS    AL1           3   FUNCTION INTERFACE VERSION NUMBER
*
SPORET   DS    0A           4   GENERAL RETURN CODE
SPOSRET  DS    0AL2          4   SUB RETURN CODE
SPOSR2   DS    AL1           4   SUB RETURN CODE 2
SPOSR1   DS    AL1           5   SUB RETURN CODE 1
SPOMRET  DS    0AL2          6   MAIN RETURN CODE
SPOMR2   DS    AL1           6   MAIN RETURN CODE 2
SPOMR1   DS    AL1           7   MAIN RETURN CODE 1
SPOFHL   EQU    8           8   GENERAL OPERAND LIST HEADER LENGTH
*
*
*   EQUATES FOR EXIT RETURN INFORMATION IN &P.MR1
*
SPOEXIO  EQU    0           SEND I/O REQUESTED
SPOEXNI  EQU    4           DO NOT SEND I/O
SPOEXEO  EQU    12          NO I/O DECLARE EOF
SPOPRYP  DC    AL1(0)      PRINTER TYPE
SPOPLINES EQU    32          LINES PRINTER TYPE           751
SPOPAGES EQU    33          PAGE PRINTER TYPE             751
SPOLOADL EQU    34          LOADABLE LINE PRINTER TYPE    751
SPOINFOR DC    X'00'        INFORMATION GIVEN TO EXIT
SPONOSCB EQU    X'01'        IND. SPOSCBAD NOT FILLED
SPOUNUS  DC    XL2'00'      **** UNUSED ****
SPOCHADD DC    A(0)         CCW'S CHAIN OR BUFFER ADDRESS
SPOSCBAD DC    A(0)         ADDRESS OF SCB
SPODV TYP DS    AL1         DEVICE TYPE
SPOPR TYP DS    CL2         PRINTER MNEMONIC
          DS    CL20        RESERVED FOR EXTENTION
SPOPLLEN EQU    *-SPOEXBEG  LENGTH OF THE PARAMETER LIST

```

Das Feld SPOCHADD enthält die Adresse des Kanalprogramms. Das Format eines Kanalbefehls ist unabhängig von Kanal- und Maschinentyp.

op	fl	count	address
----	----	-------	---------

op = Operationscode

fl = Flagbyte

count = Datenlänge

address = virtuelle Datenadresse

Hinweise

- Die Operationscodes des Kanalbefehls sind abhängig von Kanal- und Druckertyp und werden von SPOOL nach der Rückkehr aus der Exit-Routine nicht überprüft.
- Das Feld SPONOSCB zeigt an, dass das Feld SPOSCBAD keine SCB-Adresse enthält und somit nicht verwendet werden kann. Das kommt vor, wenn der SPOOL Ein/Ausgabeoperationen durchführt, die keinem Job zugeordnet sind.
- Das Feld SPODV TYP enthält den gleichen detaillierten Gerätetyp, der in der SPOOL-Parameterdatei benutzt wird. Für die Auflösung ist der Makro SPSDTYP zu verwenden.
- Das Feld SPOPRTMN enthält die mnemotechnische Bezeichnung des Druckers.

3.25.6 SPOOL-Exit 096 (Server-Auswahl)

Mit diesem System Exit kann die SPOOL-Verwaltung (TSOS bzw. PRINT-SERVICE-ADM) eine Server-Auswahl durch Kombination bestimmter Auswahl-Kriterien vornehmen. Die Server-Auswahl kann dabei einer individuellen Entscheidungstabelle angepasst sein, die beispielsweise vorsieht, dass die Aufträge von Anwender A mit dem Papierformat A auf den Server X geleitet werden sollen, die Aufträge von Anwender A mit dem Papierformat B auf Server Y u.s.w.

Der Exit 096 muss am Client-Rechner gestartet werden.

Bei Änderungen der Server-Liste muss auf deren Konsistenz geachtet werden. Der vom SPOOL-Verwalter für einen Auftrag ausgewählte Server muss in der dem Exit übergebenen Server-Liste enthalten sein. Um diese Anforderung sicherzustellen, kann der SPOOL-Verwalter

- den gewünschten Server an die erste Position der Liste setzen und die anderen Server beibehalten
- den gewünschten Server an die erste Position der Liste setzen und die Anzahl der in der Liste verfügbaren Server auf 1 setzen.

Folgende Informationen werden an die Exit-Routine übergeben:

R1 = A (EX096-Parameterbereich)

Die Returncodes werden im Standardheader eingetragen:

dd	cc	bb	aa
----	----	----	----

Über die Ausführung von EX096 wird im Standardheader ein Returncode übergeben:

aa=Maincode1

bb=Maincode2

cc=Subcode1

dd=Subcode2

X'aa'	Erläuterung
00	Exit fehlerfrei ausgeführt
01	Exit nicht fehlerfrei ausgeführt

DSECT

Eine DSECT/CSECT für den Parameterbereich kann mit dem Makro EX096 erzeugt werden, wobei „prefix“ max. 3 Zeichen lang sein darf.

```

EX096 D
SVSEXPL DSECT ,          SERVER SELECTION EXIT P/L
E96EXBEG DS  0H
          FHDR MF=(C,E96),EQUATES=NO
          DS  0A
E96FHE   DS  0XL8          0  GENERAL PARAMETER AREA HEADER
*
E96IFID  DS  0A          0  INTERFACE IDENTIFIER
E96FCTU  DS  AL2          0  FUNCTION UNIT NUMBER
*
*          BIT 15  HEADER FLAG BIT,
*          MUST BE RESET UNTIL FURTHER NOTICE
*          BIT 14-12 UNUSED, MUST BE RESET
*          BIT 11-0  REAL FUNCTION UNIT NUMBER
E96FCT   DS  AL1          2  FUNCTION NUMBER
E96FCTV  DS  AL1          3  FUNCTION INTERFACE VERSION NUMBER
*
E96RET   DS  0A          4  GENERAL RETURN CODE
E96SRET  DS  0AL2        4  SUB RETURN CODE
E96SR2   DS  AL1          4  SUB RETURN CODE 2
E96SR1   DS  AL1          5  SUB RETURN CODE 1
E96MRET  DS  0AL2        6  MAIN RETURN CODE
E96MR2   DS  AL1          6  MAIN RETURN CODE 2
E96MR1   DS  AL1          7  MAIN RETURN CODE 1
E96FHL   EQU  8          8  GENERAL OPERAND LIST HEADER LENGTH
*
*
*  EQUATES FOR EXIT RETURN INFORMATION IN &P.RET
*
E96EXNOK EQU  1          EXIT NOK
E96EXOK  EQU  0          EXIT OK
E96SCBAD DC  A(0)        ADDRESS OF SCB
E96SVLAD DC  A(0)        ADDRESS OF SERVER LIST
          DS  CL12        RESERVED FOR EXTENTION
E96PLLEN EQU  *-E96EXBEG LENGTH OF THE PARAMETER LIST
          SPACE 3
E96SVL   DSECT ,          SERVER LIST DESCRIPTION
E96SV#   DS  F            SERVERS#
E96SVSZ  DS  F            SIZE
E96SV    DS  0XL20        SERVER(SERVERS#) (1 ENTRY)
E96SVN   DS  CL8          NAME
E96APR#  DS  FL4          PRINTERS#
E96DT    DS  XL8          DEVICE_TYPES

```

3.25.7 SPOOL-Exit 097 (Steuern File Transfer)

Mit dem SPOOL-Exit 097 kann in einer Client-Server-Umgebung bei Einsatz des Produkts „Distributed Print Services“ die Anzahl der Dprint-Aufträge für den File Transfer gesteuert werden. Mit dem Einsatz des Exits ist es möglich für jeden Client eine maximale Anzahl von Aufträgen für den File Transfer festzulegen. Das führt dazu die Dprint-Systemtasks zu reduzieren, die diejenigen Dprint-Aufträge steuern, für die eine File-Transfer-Bearbeitung angestoßen wurde. Außerdem kann mit dem Exit vermieden werden, dass durch Dprint die File-Transfer-Ressourcen auf dem Client- und Server-Rechner allein beansprucht werden. Das Problem, dass Druckaufträge zu lange in einem Zwischen-Status (WFT oder FT) bleiben kann dadurch gelöst werden.

SPOOL-Hintergrund

Wurde ein von einem Client abgegebener Dprint-Auftrag von einem fernen Server angenommen, befindet er sich in der SPOOL-Auftragswarteschlange des Servers im Status WFT (wait file transfer). Auf die Anforderung der Dprint-File-Transfer-Einheit wählt SPOOL einen Auftrag aus der Warteschlange aus und übergibt ihn an Dprint. Dabei wird der Auftrag gewählt, der die höchste Priorität hat und dessen Original-Host (d.h. der Client Host, der den Auftrag abgegeben hat) verfügbar ist und freie Betriebsmittel besitzt. Hat die Dprint-File-Transfer-Einheit den Auftrag erhalten, informiert sie den Client-Host. Dann initiiert dieser den File Transfer der zu druckenden Datei und evtl. der Datei, die die für den Ausdruck benötigten Ressourcen enthält. Da die Dprint-File-Transfer-Einheit Aufträge ohne Unterbrechung anfordert, werden alle Aufträge, die vom Server angenommen wurden und File Transfer benötigen, sofort an File Transfer geleitet. Ist die Anzahl von Aufträgen sehr hoch und ist der Server nicht leistungsfähig genug, führt dies zu Performance-Problemen auf der Server-Seite und zu einer alleinigen Inanspruchnahme der File-Transfer-Ressourcen auf der Client-Seite. Der Exit 097 ist im SPOOL auf der Server-Seite lokalisiert und greift dort ein, wo ein Auftrag für die Dprint-File-Transfer-Einheit ausgewählt wird.

Folgende Informationen werden an die Exit-Routine übergeben:

R1 = A (EX096-Parameterbereich)

Im Parameterbereich sind folgende Felder versorgt:

E97HSTI@	Adresse einer Host-Tabelle, die die Namen der verfügbaren und nicht gesättigten Hosts enthält
E97HST#	Anzahl der Einträge in der Host-Tabelle
E97HSTL	Länge eines Eintrags in der Host-Tabelle
E97SVT@	Adresse der SPOOL-Vector-Tabelle

Der Returncode wird im Standardheader des Parameterbereichs abgelegt.

E97EXNOK	X'00000001' während der Bearbeitung ist ein Fehler aufgetreten.
E97EXOK	X'00000000' der Exit wurde ohne Fehler ausgeführt.

Die folgenden Felder können von der Exit-Routine versorgt werden:

E97HOK#	Anzahl in der Eingabe-Host-Liste* gebliebener Hosts
E97WFT#	Anzahl der Aufträge im Zustand WFT für die Hosts, die in der Eingabe-Host-Liste geblieben sind
E97FT#	Anzahl der Aufträge im Zustand FT für die Hosts, die in der Eingabe-Host-Liste geblieben sind

* Eingabe-Host-Liste = Eingabeliste, aus der einige Einträge entfernt wurden, d.h. die mit binären Nullen aufgefüllt wurden.

DSECT

Eine DSECT/CSECT für den Parameterbereich kann mit dem Makro EX097 erzeugt werden, wobei „prefix“ max. 3 Zeichen lang sein darf.

```

EX097 D
CLREXPL DSECT ,           File transfer balancing exit p/l
E97EXBEG DS    0H
          FHDR MF=(C,E97),EQUATES=NO
          DS    0A
E97FHE   DS    0XL8           0   GENERAL PARAMETER AREA HEADER
*
E97IFID  DS    0A           0   INTERFACE IDENTIFIER
E97FCTU  DS    AL2          0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*                               BIT 14-12 UNUSED, MUST BE RESET
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
E97FCT   DS    AL1           2   FUNCTION NUMBER
E97FCTV  DS    AL1           3   FUNCTION INTERFACE VERSION NUMBER
*
E97RET   DS    0A           4   GENERAL RETURN CODE
E97SRET  DS    0AL2         4   SUB RETURN CODE
E97SR2   DS    AL1          4   SUB RETURN CODE 2
E97SR1   DS    AL1          5   SUB RETURN CODE 1
E97MRET  DS    0AL2         6   MAIN RETURN CODE
E97MR2   DS    AL1          6   MAIN RETURN CODE 2
E97MR1   DS    AL1          7   MAIN RETURN CODE 1
E97FHL   EQU   8           8   GENERAL OPERAND LIST HEADER LENGTH
*
*
*   EQUATES FOR EXIT RETURN INFORMATION IN &P.RET
*
E97EXNOK EQU   1           EXIT NOK
E97EXOK  EQU   0           EXIT OK
E97UNUS  DC    XL4'00'     ****  UNUSED  ****
*
E97HSTL@ DS    A           ACCESSIBLE HOST LIST
E97SVT@  DS    A           SVT ADDRESS
E97HST#  DS    H           NBR OF ENTRIES IN THE LIST
E97HSTL  DS    H           LENGTH OF AN ENTRY IN THE LIST
*
E97HOK#  DS    H           OUTPUT AREA = NBR OF CORRECT HST
E97WFT#  DS    H           TOTAL NBR OF JOB IN WFT
E97FT#   DS    H           TOTAL NBR OF JOB IN FT
          DS    CL20        RESERVED FOR EXTENSION
E97PLEN  EQU   *-E97EXBEG  LENGTH OF THE PARAMETER LIST

```

3.26 SAT-Exit 110

Die SAT-Exit-Routine wird nur für Ereignisse aufgerufen, die protokolliert werden sollen, siehe Handbuch „SECOS“ [12]. Dies geschieht unmittelbar bevor das Ereignis in die SAT-Protokolldatei geschrieben wird. Die Exit-Routine muss von der Systembetreuung geladen und vom Sicherheitsbeauftragten aktiviert werden (/MODIFY-SAT-PRESELECTION EXIT=YES).

Die Exit-Routine kann

- gezielte Reaktionen einleiten (z.B. Sperrung einer Benutzerkennung nach einer bestimmten Anzahl fehlerhafter LOGON-Versuche)
- einen eigenen SATLOG-Satz schreiben (Ereignis ANY, Makro \$SATANY)
- bei Rückkehr zu SAT über den Returncode das Schreiben des analysierten SATLOG-Satzes zulassen oder unterdrücken.

Folgende Information wird an die Routine übergeben:

```
R1 = A(EX110-Parameterbereich)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Die Rückkehrinformation an die aufrufende Systemkomponente erfolgt über die Parameterliste SATZPMDL im Feld SATZSR1.

SATZSR1 = X'00' (das Ereignis soll in die SAT-Protokolldatei geschrieben werden)

SATZSR1 = X'04' (das Ereignis soll nicht in die SAT-Protokolldatei geschrieben werden)

Die Parameter-Liste, die an die Exit-Routine übergeben wird, enthält eine Kopie des Protokolldatensatzes. Dadurch wird sichergestellt, dass die Originalinformation nicht verändert werden kann.

Die Parameter-Liste hat folgenden Inhalt:

- Standardheader
- Länge des Datensatzes (2 Byte)
- reserviert (2 Byte)
- Protokolldatensatz (fester und variabler Teil)

Der Protokolldatensatz ist im Handbuch „SECOS“ [12] beschrieben. Er besteht aus einem festen Teil und einem variablen Teil. Der variable Teil enthält eine Liste von Informationsfeldern. Es gibt zwei Typen von Informationsfeldern:

Das Standard-Informationsfeld

- Längenangabe
- Identifikator (siehe Handbuch „SECOS“ [12])
- zugehöriges Datenfeld

Das *LNG-Feld

- Feld mit dem Wert 255 (Indikator für *LNG-Feld)

- negativen Indikator
- Längenangabe
- 2 Byte reserviert
- zugehöriges Datenfeld

Die Beschreibung des Parameterbereichs gibt nur das Layout für **ein** Informationsfeld (Standard-Informationsfeld bzw. *LNG-Feld) des variablen Teils an.

DSECT

```

        EX110 MF=D
        MFTST MF=D,PREFIX=S,MACID=ATZ,ALIGN=F,                                C
            DMACID=ATZ,SUPPORT=(D,C),DNAME=ATZPMDL
SATZPMDL DSECT ,
            *,##### PREFIX=S, MACID=ATZ #####
*   set for subcode_1
SATZLOG  EQU   0                               record-log requested
SATZRJ   EQU   4                               reject record
*
SATZIMDL EQU   5900                            max. length of logging record
*
SATZILFP EQU   28                             length of fixed part in
*                                           logging record
*
*   parameter list description
SATZHDR  FHDR  MF=(C,SATZ),EQUATES=NO                standard header
SATZHDR  DS    0A
SATZFHE  DS    0XL8                                0   GENERAL PARAMETER AREA HEADER
*
SATZIFID DS    0A                                0   INTERFACE IDENTIFIER
SATZFCTU DS    AL2                                0   FUNCTION UNIT NUMBER
*                                           BIT 15   HEADER FLAG BIT,
*                                           MUST BE RESET UNTIL FURTHER NOTICE
*                                           BIT 14-12 UNUSED, MUST BE RESET
*                                           BIT 11-0  REAL FUNCTION UNIT NUMBER
SATZFCT  DS    AL1                                2   FUNCTION NUMBER
SATZFCTV DS    AL1                                3   FUNCTION INTERFACE VERSION NUMBER
*
SATZRET  DS    0A                                4   GENERAL RETURN CODE
SATZSRET DS    0AL2                              4   SUB RETURN CODE
SATZSR2  DS    AL1                                4   SUB RETURN CODE 2
SATZSR1  DS    AL1                                5   SUB RETURN CODE 1
SATZMRET DS    0AL2                              6   MAIN RETURN CODE
SATZMR2  DS    AL1                                6   MAIN RETURN CODE 2
SATZMR1  DS    AL1                                7   MAIN RETURN CODE 1

```

```

SATZFHL EQU 8 8 GENERAL OPERAND LIST HEADER LENGTH
*
* main return codes
SATZOK EQU 0 no error
*
SATZLEN DS H length of logging record
SATZRS1 DS XL2 alignment
*
SATZREC DS 0XL5928 logging record
*
SATZFIX DS 0XL28 fixed part of logging record
SATZUID DS CL8 user id
SATZTSN DS CL4 TSN
SATZEVT DS CL3 event
SATZRES DS CL1 event result
*
SATZTSP DS 0XL8 time stamp
SATZDAT DS CL4 YYYYMMDD
SATZTIM DS CL4 HHMMSS00
*
SATZSAT DS X SAT version
SATZETT DS X reserved for extent
SATZSCF DS X caller
SATZSCV DS X caller version
*
SATZVAR DS 0XL5900 variable part of logging
* record
SATZDAR DS CL5900 variable part
ORG SATZVAR
*
SATZFLD DS 0XL258 layout of a standard field
SATZVLN DS X length of the field
SATZVID DS CL2 id of the field
SATZVDT DS CL255 data
*
ORG SATZVAR
*
SATZEFLD DS 0XL4007 layout of a *LNG field
SATZVLND DS X 255 to indicate *LNG-field
SATZNIDE DS CL2 negative id of the field
SATZFLEN DS CL2 length of *LNG-field
SATZABRL DS CL2 on exit 110 always zero
SATZLVAL DS CL4000 data
*
ORG SATZVAR+5900
*
SATZ# EQU *-SATZHDR

```

3.27 Makro \$SATANY

Anwendung: Systemsicherheit (TPR)

Makro-Typ: GC-Typ (C-Form, D-Form, E-Form, L-Form, M-Form)

SAT kann von einem System Exit aufgerufen werden. Die Schnittstelle \$SATANY bietet dem privilegierten Benutzer die Möglichkeit, eigene Informationen in die SAT-Protokolldatei zu schreiben. Wenn \$SATANY in der Exit-Routine 110 verwendet wird, verhindert ein Schutzmechanismus rekursive Aufrufe der Exit-Routine. Das Ereignis ANY kann zurückgewiesen werden, wenn es vom Sicherheitsbeauftragten nicht für die Protokollierung ausgewählt wurde.

Mit dem \$SATANY-Makro werden im privilegierten Status für sicherheitsrelevante, benutzerspezifische Ereignisse Protokolldatensätze in die SAT-Protokolldatei geschrieben.

Mit diesem Makro wird ein globales Ereignis ANY eingeführt und es kann ein benutzereigener, vier Zeichen langer Subcode für dieses Ereignis erstellt werden. Diese Möglichkeit erleichtert die Aufbereitung der Ereignisse mit dem Auswerter SATUT (siehe Handbuch „SECOS“ [12]).

Der Benutzer kann folgende Informationen protokollieren lassen:

- type of event
- result of event
- sub-event code
- data type
- data reference
- long data reference

Zusätzlich ergänzt SAT diese Information durch folgende Informationen:

- TSN der aufrufenden Task
- Benutzerkennung der aufrufenden Task
- time-stamp
- Gruppenkennung (falls vorhanden)
- Chipkarten-Id, wenn die Chipkarte verwendet wird
oder alternativ
die persönliche Benutzerkennung, wenn SRPM im Einsatz ist und für eine Benutzerkennung der Zugang mittels persönlicher Identifizierung festgelegt wird.

Hinweis

In einigen Fällen, besonders wenn ein LOGON-Versuch fehlschlägt, kann SAT keine Benutzerkennung protokollieren. Diese Schnittstelle bietet keine Möglichkeit, die Benutzerkennung der aufrufenden Task zu überschreiben.

Makroaufrufformat und Operandenbeschreibung

```
[name] $SATANY  
[,TYPE=*NONE/<variable>/*ANY]  
[,RESULT=*NONE/<variable>/*SUCC/*FAIL]  
[,SUBCOD=*NONE/<variable>/*TEXT/*HEXA/*BOTH]  
[,DATATYP=*NONE/<variable>]  
[,DATAPTR=*NONE/<variable>]  
[,DATALEN=*NONE/<variable>]  
[,LDTAPTR=*NONE/<variable>]  
[,LDTALEN=*NONE/<variable>]
```

TYPE=

Bestimmt den Ereignis-Typ

=*NONE

der Operand wird nicht verwendet (bei MF=L Standard)

= <variable>

symbolische Adresse eines drei Byte langen Feldes mit dem Ereignis-Typ (immer ANY)

=*ANY

Ereignis ANY

RESULT

Definiert das Ereignis-Ergebnis

=*NONE

der Operand wird nicht verwendet (bei MF=L Standard)

==<variable>

symbolische Adresse eines ein Byte langen Feldes mit dem Ereignis-Ergebnis (SUCC oder FAIL)

=*SUCC

das Ereignis wurde erfolgreich und komplett ausgeführt

=*FAIL

ein nicht behebbarer Fehler trat während des Ereignisses auf

SUBCOD

Subcode für ein Ereignis

Bestimmt den Namen des Sub-Ereignisses, innerhalb des Ereignisses. Das Feld muss linksbündig aufgefüllt werden. Es kann kleiner oder gleich vier Zeichen lang sein. Ist das Sub-Ereignis kürzer als vier Zeichen, muss das Feld mit Leerzeichen aufgefüllt werden.

=*NONE

der Operand wird nicht verwendet (bei MF=L Standard)

=<variable>

bestimmt die symbolische Adresse eines vier Byte langen Feldes

DATATYP

bestimmt den Typ der protokollierten Information

=*NONE

der Operand wird nicht verwendet (bei MF=L Standard)

=<variable>

bestimmt die symbolische Adresse eines zwei Byte langen Feldes, das den Typ der protokollierten Information enthält

X'0060' = Typ *TEXT

X'0061' = Typ *HEXA

X'0062' = Typ *BOTH

=*TEXT

das Feld wird als Zeichenkette ausgegeben

=*HEXA

das Feld wird hexadezimal ausgegeben

=*BOTH

gibt an, dass das Feld sowohl Text als auch hexadezimale Zeichen enthält. Die Ausgabe zeigt in der ersten Zeile Text und in den beiden folgenden Zeilen den entsprechenden Hexadezimalcode (entsprechend der EDT-Ausgabe im Hexa-Modus).

DATAPTR

Datenzeiger (Länge der Daten höchstens 255 Byte)

=*NONE

der Operand wird nicht verwendet (bei MF=L Standard)

=<variable>

symbolische Adresse eines vier Byte langen Feldes, das die Adresse des ersten Bytes des zu protokollierenden Datenfelds enthält.

DATALEN

Länge der Information

=*NONE

der Operand wird nicht verwendet (bei MF=L Standard)

=<variable>

symbolische Adresse eines ein Byte langen Feldes, das die Größe des zu protokollierenden Datenfelds in Byte enthält.

LDTAPTR

Zeiger auf Daten, deren Länge größer ist als 255 Byte

=*NONE

der Operand wird nicht verwendet (bei MF=L Standard)

=<variable>

symbolische Adresse eines vier Byte langen Feldes, das die Adresse des ersten Bytes des zu protokollierenden Datenfelds enthält.

LDTALEN

Länge der Information

=*NONE

der Operand wird nicht verwendet (bei MF=L Standard)

=<variable>

symbolische Adresse eines zwei Byte langen Feldes, das die Größe des durch LDTAPTR bestimmten Datenfelds in Byte enthält.

Beziehung zwischen den Operanden:

Die folgende Tabelle zeigt, welche Verbindung zwischen den Operanden von \$SATANY besteht, d.h. die Pflichtparameter und die optionalen Parameter für den Ereignis-Typ.

	SAT-Information							
Ereignis	TYPE	RESULT	SUBCODE	DATATYP	DATAPTR	DATALEN	LDTAPTR	LDTALEN
ANY event	ANY	M	O	O	O	O	O	O

M = Pflichtangabe O = Optional

Wenn entweder DATATYP, DATAPTR oder DATALEN den Wert *NONE hat, wird kein Datenfeld protokolliert. Das durch LDTAPTR und LDTALEN angegebene Datenfeld wird nur dann protokolliert, wenn der Sicherheitsbeauftragte die Protokollierung von Zusatzinformation zu Ereignissen eingeschaltet hat (/MODIFY-SAT-PRESELECTION LOGGING-QUANTITY=*EXTENDED).

Entry-Name: SATANY

4 Eventing in Exit-Routinen

Mit den Makros ENAEI, SOLSIG, POSSIG, ... (siehe Handbuch „Makroaufrufe“ [9]) nutzt der Programmierer in nicht privilegierten (TU) Benutzerprogrammen den Börsenmechanismus von BS2000 zur ereignisabhängigen Steuerung von Programmen (Eventing).

Die Anwendung dieser Makros ist in den privilegiert (TPR) ablaufenden Exit-Routinen nicht möglich.

Um auch dem Exit-Programmierer diese Funktionen zugänglich zu machen, werden im folgenden drei privilegierte Makros beschrieben. Sie stellen die Verbindung her zwischen

- einer Hilfstask unter TSOS-Kennung, welche eine Ereigniskennung einrichtet (mit ENAEI);
- über die Exit-Routine Nummer 0, die mit EXCALL innerhalb der Hilfstask aufgerufen wird;
- hin zu „normalen“ Exit-Routinen, die den Mechanismus nutzen wollen.

Diese Funktionen ermöglichen z.B., dass mehrere Exit-Routinen während eines Systemlaufs in dieselbe Protokolldatei schreiben können. Zu diesem Zweck wird unter TSOS-Kennung ein Programm gestartet, das ereignisgesteuert jedes Mal in Aktion tritt, wenn eine an dieselbe Ereigniskennung angeschlossene Exit-Routine eine Ausgabe wünscht.

4.1 \$GTBOID - GET Bourse Identifier

Der \$GTBOID-Makro ermittelt für eine Ereigniskennung die zugehörige Börsenkennung. \$GTBOID darf nur in der Exit-Routine Nummer 0 benutzt werden, die von der Hilftask unter TSOS aufgerufen wird und die die Ereigniskennung eingerichtet hat (ENAEI-Makro). Die Ereignis-Kurzkennung muss in der EXCALL-Parameterliste an die Exit-Routine übergeben werden.

Makroaufrufformat und Operandenbeschreibung

[label] \$GTBOID
MF=C / D / E / L
[,PREFIX=prefix]
[,PARAM=adr]

MF

beschreibt die Makroaufrufform.

=C

Die Parameterliste wird als Datenbereich generiert.

=D

Die Parameterliste wird als DSECT generiert mit dem für „label“ angegebenen Namen; Voreinstellung: label = NTEG.

=E

Aufruf der Funktion.

=L

Die Parameterliste wird generiert (ohne Feldnamen und Equates).

PREFIX

Angabe eines Zeichens für die Feldnamen in der Parameterliste. Nur in Verbindung mit MF=C/D möglich.

=prefix

Ein Zeichen, mit dem die Feldnamen in der Parameterliste beginnen sollen; Voreinstellung: N.

PARAM

bezeichnet die Adresse der Parameterliste, wenn MF=E angegeben wird.

=adr

Adresse der Parameterliste.

Hinweise

- R12 und R13 dürfen seit Exit-Routinenauf Ruf nicht verändert worden sein.
- R14, R15 und R1 werden durch MF=E verändert.
- Die Parameterliste muss entsprechend der Reentrant-Programmierung bearbeitet werden.

Returncode

Im Standardheader der Parameterliste wird folgender Returncode übergeben:

Maincode	Bedeutung
NTEGPAER	Parameter-Syntax-Fehler
NTEGEIER	Falsche Ereigniskennung, keine Börsenkennung zurückgegeben
X' 00'	Erfolgreicher Aufruf, Börsenkennung zurückgegeben

Programmierbeispiel

```

EXIT0000 @ENTR TYP=E,ENV=SPLSPEC,LOCAL=ADF
:
MVC CPARAM(NTEG#),LPARAM _____ (1)
MVC NTEGEIID(4),EIID _____ (2)
$GTBOID MF=E,PARAM=CPARAM _____ (3)
CLI NTEGMR1,X'00' _____ (4)
BNE exception
MVC BOID(4),NTEGBOID _____ (5)
:
@EXIT
:
LPARAM $GTBOID MF=L _____ (6)
NTEGDATA DS 0A
NTEGFHE DS 0XL8 GENERAL OPERAND LIST HEADER
NTEGFCTU DC AL2(81) FUNCTION UNIT NUMBER
NTEGFCT DC AL1(0) FUNCTION NUMBER
NTEGFCTV DC AL1(1) FUNCTION INTERFACE VERSION NUMBER
NTEGRET DC X'FFFFFFFF' RETURN CODE NOT VALID
NTEGMR1 EQU NTEGRET+3,1 MAIN RETURN CODE 1
*
NTEGEIID DC F'0' Event Item Identifier (EIID)
NTEGBOID DC F'0' Bourse Identifier (BOID)
NTEG# EQU *-NTEGDATA Size of Parameter List
:
BOID DC A(0)
ENTRY BOID
@END
ADF @PAR L=YES
:
CPARAM $GTBOID MF=C _____ (7)
ADF @PAR LEND=YES

```

- (1) Vorbereiten der aktuellen Parameterliste mit dem „Standardmuster“
- (2) Einsetzen der Ereigniskennung
- (3) Aufruf der Funktion \$GTBOID
- (4) Auswertung des Returncodes

- (5) Sicherstellung der Börsenkennung im Gut-Fall an „gut“ sichtbarer Stelle oder im Systemvektor
- (6) Standardmuster der Parameterliste
- (7) Platzhalter der aktuellen Parameterliste im Automatic-Speicherlayout

4.2 \$NBSSIG - Send a Signal to Event Item Bourse

Mit der \$NBSSIG-Funktion kann die Exit-Routine einer wartenden Task anzeigen, dass ein Börsenereignis eingetreten ist (analog POSSIG in TU). Ein Postcode wird an die mit SOLSIG auf das Ereignis wartende Task übergeben.

Voraussetzung: Eine Ereigniskennung muss existieren und die dazugehörige Börsenkennung mit \$GTBOID systemglobal zugreifbar abgespeichert sein.

Makroaufrufformat und Operandenbeschreibung

```
[label] $NBSSIG
MF=C / D / E / L / M
[,PREFIX=prefix]
[,PARAM=adr]
[,BOID=boid / (R1)]
[,SDATA=(1, adr / (R2))]
```

MF

beschreibt die Makroaufrufform.

=C

Die Parameterliste wird als Datenbereich generiert.

=D

Die Parameterliste wird als DSECT generiert mit dem für „label“ angegebenen Namen; Voreinstellung: label = NTEG.

=E

Aufruf der Funktion.

=L

Die Parameterliste wird generiert (ohne Feldnamen und Equates).

=M

Modifikation der Parameterliste mit aktuellen Werten.

PREFIX

Angabe eines Zeichens für die Feldnamen in der Parameterliste. Nur in Verbindung mit MF=C/D möglich.

=prefix

Ein Zeichen, mit dem die Feldnamen in der Parameterliste beginnen sollen; Voreinstellung: N.

PARAM

bezeichnet die Adresse der Parameterliste, wenn MF=E angegeben wird.

=adr

Adresse der Parameterliste.

BOID=

bezeichnet die Börsenkennung.

=boid

Börsenkennung, wie sie mit \$GTBOLD global hinterlegt wurde.

=(R1)

Register, das die Börsenkennung enthält.

SDATA

bezeichnet ein Feld mit der Adresse des 1-Wort-Postcode, der an die mit SOLSIG wartende Task übergeben wird. Das kann z.B. die Adresse der Daten sein, die in eine Protokolldatei ausgegeben werden sollen.

=(1,...)

adr = Adresse des Feldes mit dem 1-Wort-Postcode.

R2 = Register mit dem Adresswert des Feldes mit dem 1-Wort-Postcode.

Hinweise

- R12 und R13 dürfen seit Exit-Routinenaufruf nicht verändert worden sein.
- R14, R15 und R1 werden durch MF=E verändert.
- Die Parameterliste muss entsprechend der Reentrant-Programmierung bearbeitet werden.

Returncode

Im Standardheader der Parameterliste (mit \$NBSSIG MF=D/C) wird folgender Returncode übergeben:

Maincode	Bedeutung
NTBSM1OK	Erfolgreicher Aufruf
NTBSM1IC	Inkonsistente Parameter
NTBSM1BE	Fehlerhafte Börsendaten
NTBSM1IB	Ungültige Börsenkennung
NTBSM1BD	Börse bereits zerstört, d.h. Ereigniskennung existiert nicht mehr
NTBSM1QL	Überlauf in der Sende-Queue der Börse
NTBSM1NS	Keine Slots für Sende-Queue verfügbar

Keine Aktion bei allen Returncodes außer NTBSM1OK.

Programmierbeispiel

```

EXITXXXX @ENTR TYP=E,ENV=SPLSPEC,LOCAL=ADF
:
L    R1,BOIDX _____ (1)
L    R1,0(R1)
MVC  CPARAM(NTBS#),LPARAM _____ (2)
$NBSSIG MF=M,PARAM=CPARAM,BOID=(R1),SDATA=(1,SDATA) _____ (3)
$NBSSIG MF=E,PARAM=CPARAM _____ (4)
CLI  NTBSMR1,NTBSM1OK _____ (5)
BNE  exception
:
@EXIT
:
LPARAM $NBSSIG MF=L _____ (6)
NTBS#  DS    0A
NTBSFHE DS    0XL8          GENERAL OPERAND LIST HEADER
NTBSFCTU DC   AL2(93)      FUNCTION UNIT NUMBER
NTBSFCT  DC   AL1(6)       FUNCTION NUMBER

```

```

NTBSFCTV DC   AL1(1)       FUNCTION INTERFACE VERSION NUMBER
NTBSRET  DC   X'FFFFFFF'   RETURN CODE NOT VALID
NTBSMR1  EQU  NTBSRET+3,1  MAIN RETURN CODE 1
*
NTBSBOID DC   F'0'         BOID
          DC   X'00'        unused
          DC   X'00'        unused
NTBSVECT DC   B'01000000'  VECT. FIELD
NTBSDLTH DC   X'00'        DATA LENGTH
NTBSLTIM DC   H'600'       LIFETIME
NTBSPRIO DC   AL1(0)       PRIORITY
          DC   X'00'        unused
NTBSSIG  DC   F'0'         SIGNAL
          DC   F'0'        unused
          DC   F'0'        unused
NTBSSDTA DC   A(0)         SEND DATA ADDRESS
NTBSARSL DC   A(0)         AR Slot address (only for SIH user)
NTBS#    EQU  *-NTBS#      Length of parameterlist
:
BOIDX   DC   A(BOID) _____ (7)
          EXTRN BOID
SDATA   DC   A(MESS)
MESS    DC   Y(MESSEND-MESS)
          DC   C'  IRGEND EIN TEXT'
:
@END
ADF     @PAR  L=YES
:
CPARAM  $NBSSIG MF=C _____ (8)
:
ADF     @PAR  LEND=YES

```

(1) Besorgen der Börsenkennung

- (2) Vorbereiten der aktuellen Parameterliste mit dem Standardmuster
- (3) Einsetzen der Börsenkennung und des Postcodes
- (4) Aufruf der Funktion \$NBSSIG
- (5) Auswerten des Returncodes
- (6) Standardmuster der Parameterliste
- (7) Beispiel für die Börsenkennungsermittlung und Postcodeverwendung
- (8) Platzhalter der aktuellen Parameterliste im Automatic-Speicherlayout

5 System Exits für Programmbibliotheken

Mit den System Exits für Programmbibliotheken (PLAM-Exits) kann die Systembetreuung die Zugriffe auf die Bibliothekselemente überwachen und gegebenenfalls zurückweisen. Unterstützt werden die Funktionen: Element eröffnen, schließen, löschen und umbenennen. Für jede dieser Funktionen gibt es Request- und Return Exits, also vor Funktionsaufruf und nach Abschluss der Operation, siehe unten die Übersicht über die PLAM-Exits.

Die gezielte Auswahl einer Bibliothek wird über einen eigenen Exit getroffen, der bei jeder Funktion ATTACH-Library angesprochen wird. Die Exits für die Element-Zugriffsfunktionen können so auf ein vertretbares Minimum (Performance) beschränkt werden.

Funktion	Art des Exits	Exit-Nummer
Element eröffnen (OPEN ELEMENT)	Request	005
	Return	006
Element schließen (CLOSE ELEMENT)	Request	007
	Return	008
Element löschen (DELETE ELEMENT)	Request	055
	Return	056
Element umbenennen (RENAME ELEMENT)	Request	057
	Return	058
Programmbibliothek eröffnen (ATTACH LIBRARY)		054

Folgende Informationen werden an den Exit-Punkten übergeben:

```
R1 = A(Exit-Parameterliste)
R12 = A(TPR Program Manager)
R13 = A(Sicherungsbereich der rufenden Komponente)
R14 = A(indirekter Rücksprung)
R15 = A(Exit-Routine)
```

Rückkehrinformation:

```
R15 = X'000000RR'
RR = Rückkehrinformation der Exit-Routine:
00: kein Fehler (005-008, 055-058), Bibliothek soll überwacht werden (054)
04: Funktion wird abgewiesen (005, 006, 008, 055-058), Bibliothek soll
nicht überwacht werden (054)
```

Am Request Exit kann die Systembetreuung den Funktionsaufruf zurückweisen, bevor er ausgeführt wird (Exit-Punkte 005, 055, 057).

Am Return Exit ist die Aktion bereits ausgeführt (Exit-Punkte 006, 008, 056, 058). Mit Ausnahme des Exit-Punktes 006 (OPEN ELEMENT) kann die Aktion nicht mehr rückgängig gemacht werden.

DSECT

Sämtliche Exits für Programmbibliotheken werden mit einer Parameterliste identischer Struktur versorgt.

Mit dem Makro EXnnn kann eine DSECT für die Parameterliste generiert werden: `name EXnnn D[, [prefix], [ALL]]`

nnn = dreistellige Exitnummer

Ohne die Angabe ALL werden nur die von der Funktion versorgten Felder mit Namen versehen. Mit der Angabe ALL wird eine für alle Funktionen allgemein gültige Parameterliste erzeugt. Mit dem Parameter „prefix“ lassen sich die symbolischen Feldnamen um einen vorangestellten, freigewählten Buchstaben erweitern.

Alle Exits werden in der Parameterliste mit der aktuellen PLAM-Version und dem Bibliotheksnamen versorgt. Ein LINK=name wird hinterlegt, sofern vorher angegeben. Typ, Name, Version und Variante eines Bibliothekselements sind ebenfalls in der Parameterliste hinterlegt mit Ausnahme der Funktionen

- ATTACH LIBRARY: es besteht noch kein Bezug zum Bibliothekselement
- OPEN ELEMENT: beim Request Exit können die Angaben zum Bibliothekselement noch unvollständig sein (siehe auch OPEN-Element-Exit 005).

Welche Felder von PLAM versorgt werden, zeigt die folgende Tabelle.

von PLAM versorgte Felder	Exit-Routinen										
	ATTACH		OPEN			CLOSE		DELETE		RENAME	
	054	005	006	007	008	056	057	058	059		
PMRC	x	x	x	x	x	x	x	x	x	x	
PMRC1	x	x	x	x	x	x	x	x	x	x	
PMPLV	x	x	x	x	x	x	x	x	x	x	
PMLIB	x	x	x	x	x	x	x	x	x	x	
PMLINK	x	x	x	x	x	x	x	x	x	x	
PMOPM		x	x								
PMSLV		x	x								
PMLVD		x	x								
PMCLM				x	x						
PMTYP		x	x	x	x	x	x	x	x	x	
PMNAM		x	x	x	x	x	x	x	x	x	
PMVER		x	x	x	x	x	x	x	x	x	
PMVAR		x	x	x	x	x	x	x	x	x	
PMDAT		x	x	x	x	x	x	x	x	x	
PMTYP2								x	x		

System Exits

PMNAM2									x	x
PMVER2									x	x

6 Anhang

Der Anhang enthält die Systemtabellen und SVC-Liste.

6.1 Systemtabellen

Dieser Abschnitt enthält die DSECTs für verschiedene im Text angesprochene Systemtabellen. Die hier abgedruckten Makroauflösungen entsprechen dem Stand der Version BS2000/OSD-BC V5.0. Für die praktische Anwendung wird empfohlen, sich die jeweils aktuellen Tabellen selbst zu erzeugen.

6.1.1 CATAL

```

        CATAL MF=D,VERSION=3
*****      CATAL PARAMETER AREA  --  VERSION=3  *****
        MFTST MF=D,PREFIX=I,MACID=DK,ALIGN=F,          C
            DMACID=DK,SUPPORT=(C,D),DNAME=DKCATPL
IDKCATPL DSECT ,
            *,##### PREFIX=I, MACID=DK #####
IDKCATST EQU *
        FHDR MF=(C,IDK),EQUATES=NO
        DS    0A
IDKFHE   DS    0XL8          0   GENERAL PARAMETER AREA HEADER
*
IDKIFID  DS    0A           0   INTERFACE IDENTIFIER
IDKFCTU  DS    AL2          0   FUNCTION UNIT NUMBER
*
*                               BIT 15   HEADER FLAG BIT,
*                               MUST BE RESET UNTIL FURTHER NOTICE
*                               BIT 14-12 UNUSED, MUST BE RESET
*                               BIT 11-0   REAL FUNCTION UNIT NUMBER
IDKFCT   DS    AL1          2   FUNCTION NUMBER
IDKFCTV  DS    AL1          3   FUNCTION INTERFACE VERSION NUMBER
*
IDKRET   DS    0A           4   GENERAL RETURN CODE
IDKSRET  DS    0AL2         4   SUB RETURN CODE
IDKSR2   DS    AL1          4   SUB RETURN CODE 2
IDKSR1   DS    AL1          5   SUB RETURN CODE 1
IDKMRET  DS    0AL2         6   MAIN RETURN CODE
IDKMR2   DS    AL1          6   MAIN RETURN CODE 2
IDKMR1   DS    AL1          7   MAIN RETURN CODE 1
IDKFHL   EQU    8           8   GENERAL OPERAND LIST HEADER LENGTH
*
IDKPN1   DS    CL80          FILE          .#'008' 200
*
IDKPN2   DS    CL80          NEWNAME       .#'058' 200
*
*                               PASSWORDS
IDKRD PAS DS    F            READ  PASSWORD .#'0A8'
IDKWR PAS DS    F            WRITE PASSWORD .#'0AC'
IDKEXPAS DS    F            EXEC  PASSWORD .#'0B0'
*
IDKRETPD DS    H            RETENTION PERIOD .#'0B4'
*
IDKCFLAG DS    X            FLAG          .#'0B6'
IDKSTATE EQU    X'80'       7-7 :S: STATE=*UPDATE
*                               :R: STATE=*NEW OR STATE=*FOREIGN
IDKACCES EQU    X'40'       6-6 :S: ACCESS=*READ
*                               :R: ACCESS=*WRITE

```

```

IDKSHARE EQU X'20'          5-5 :S: SHARE=*YES / :R: SHARE = *NO
IDKRSPSP EQU X'10'          4-4 :S: RDPASS SPECIFIED          110
IDKWSPSP EQU X'08'          3-3 :S: WRPASS SPECIFIED          110
IDKRPDSP EQU X'04'          2-2 :S: RETPD SPECIFIED          110
IDKACCSP EQU X'02'          1-1 :S: ACCESS SPECIFIED          110
IDKSHASP EQU X'01'          0-0 :S: SHARE SPECIFIED          110
*
IDKCFLG1 DS X              FLAG1                .#'0B7'
IDKCRSP EQU X'80'          7-7 :S: RELSPAC SPECIFIED          110
IDKCBAS EQU X'40'          6-6 :S: BASE SPECIFIED          953
IDKNOENC EQU X'20'          5-5 :S: DO NOT ENCRYPT PASSWORDS
IDKSTATF EQU X'10'          4-4 :S: STATE = *FOREIGN
IDKKEEP EQU X'08'          3-3 :S: DISP = *KEEP          953
IDKREUS EQU X'04'          2-2 :S: DISP = *REUSE
IDKCYCL EQU X'02'          1-1 :S: DISP = *CYCLE
IDKDELT EQU X'06'          1-2 :S: DISP = *DELETE
IDKDISPS EQU X'0E'          1-2-3 : IF NOT ZERO, DISP PRESENT  200
IDKEXSP EQU X'01'          0-0 :S: EXPASS SPECIFIED          110
*
*              GENERATION FIELDS
IDKBASE DS H              BASE = +/-NNN OR NNNN        .#'0B8'
IDKGEN DS H              GEN = NNN                    .#'0BA'
IDKFRST DS H              FIRST = NNNN                .#'0BC'
IDKDEVTP DS CL8          NAME OF DEVICE-TYPE          .#'0BE' 910
IDKVOLN DS CL6           VOLUME NAME OF PRIVATE DISK .#'0C6' 004
*
IDKBKUP DS X              BACKUP                    .#'0CC' 005
IDKBKUPU EQU X'8F'          7-3-2-1-0 :R: -- UNUSED, MUST BE 0 --
IDKBKUPE EQU X'40'          6-6 :S: BACKUP LEVEL = E
IDKBKUPD EQU X'30'          4-5 :S: BACKUP LEVEL = D
IDKBKUPC EQU X'20'          5-5 :S: BACKUP LEVEL = C
IDKBKUPB EQU X'10'          4-4 :S: BACKUP LEVEL = B
IDKBKUPA EQU X'00'          BACKUP LEVEL = A
*
IDKCFLG2 DS X              FLAG2                    .#'0CD' 005
IDKRSPC EQU X'80'          7-7 :S: RELSPAC = *IGNORED  110
IDKSPECI EQU X'40'          6-6 :S: SHARE = *SPECIAL
IDKPWSK EQU X'20'          5-5 :S: PASSW = *YES
IDKBKUPF EQU X'10'          4-4 :S: BACKUP LEVEL SPECIFIED
IDKDESTU EQU X'0C'          2-3 :S: DESTROY = *UNCHANGED 200
IDKNDEST EQU X'08'          3-3 :S: DESTROY = *NO
IDKDEST EQU X'04'          2-2 :S: DESTROY = *YES
IDKNLARG EQU X'02'          1-1 :S: LARGE = *NO
IDKLARGE EQU X'01'          0-0 :S: LARGE = *YES
*
IDKAUDIT DS X              AUDIT                    .#'0CE'
IDKAUDUN EQU X'F0'          4-7 -- UNUSED, MUST BE 0 -- 200
IDKAUDNO EQU X'08'          3-3 :S: AUDIT = *NONE

```

```

IDKAUDAL EQU X'04'          2-2 :S: AUDIT          = *ALL
IDKAUDFA EQU X'02'          1-1 :S: AUDIT          = *FAILURE
IDKAUDSU EQU X'01'          0-0 :S: AUDIT          = *SUCCESS
IDKNOAUD EQU X'00'          AUDIT              NOT SPECIFIED
*
IDKFLAG3 DS X              FLAG3                .#'0CF'
IDKNERIM EQU X'80'          7-7 :S: DO NOT ERASE BY IMCAT (TP) 950
IDKSUSA EQU X'40'          6-6 :S: NO PUBSPACE ACCOUNTING(TP) 950
IDKOPNBY EQU X'20'          5-5 :S: OPNBACK=*YES                001
IDKOPNBN EQU X'10'          4-4 :S: OPNBACK=*NO                001
IDKACLPHY EQU X'08'          3-3 :S: ACLPROT=*YES                (TP) 005
                                15:29:06 2001-10-15 PAGE
0030
SOURCE STATEMENT
IDKACLPN EQU X'04'          2-2 :S: ACLPROT=*NO                (TP) 005
IDKBACLN EQU X'02'          1-1 :S: BASACL=*NONE                010
IDKBACLS EQU X'01'          0-0 :S: BASACL=*STD                014
*
IDKMIGRI DS X              MIGRATE              .#'0D0'
IDKMIGAL EQU X'01'          0-0 :S: MIGRATE = *ALLOWED          951
IDKMIUNU EQU X'02'          1-1 :R: -- UNUSED --                010
IDKMIGIN EQU X'03'          0-1 :S: MIGRATE = *INHIBITED          951
IDKMIGFB EQU X'04'          2-2 :S: MIGRATE = *FORBIDDEN          200
IDKMIGNO EQU X'00'          :R: MIGRATE NOT SPECIFIED          951
IDKMUNUS EQU X'F8'          -- UNUSED, MUST BE 0 --            951
*
IDKFAR DS 0X              ACCESS RIGHTS          003
IDKOWNER DS X              OWNER                .#'0D1' 003
IDKGROUP DS X              GROUP                .#'0D2' 003
IDKOTHER DS X              OTHERS              .#'0D3' 003
*
IDKFAUS EQU X'80'          7-7 :S: USER CLASS SPECIFIED          003
IDKFARS EQU X'40'          6-6 :S: READ SPECIFIED                003
IDKFAWS EQU X'20'          5-5 :S: WRITE SPECIFIED                003
IDKFAXS EQU X'10'          4-4 :S: EXEC SPECIFIED                003
IDKFARO EQU X'08'          3-3 :S: SET READ                    003
IDKFAWO EQU X'04'          2-2 :S: SET WRITE                    003
IDKFAXO EQU X'02'          1-1 :S: SET EXEC                    003
IDKFARU EQU X'01'          0-0 -- UNUSED, MUST BE 0 ---          003
*
IDKLAST DS H              LAST = NNNN          .#'0D4' 110
*
IDKPERF DS X              IO-PERFORMANCE        .#'0D6' 110
IDKPFUM EQU X'FF'          0-7 :S: IOPERF = *USER_MAX            110
IDKPFVH EQU X'03'          0-1 :S: IOPERF = *VERY_HIGH          110
IDKPFHI EQU X'02'          1-1 :S: IOPERF = *HIGH                110
IDKPFST EQU X'01'          0-0 :S: IOPERF = *STD                110
IDKPFNS EQU X'00'          IOPERF NOT SPECIFIED          110

```

```

*
IDKUSAG DS X IO-USAGE .#'0D7' 110
IDKUSU EQU X'FC' 2-7 -- MUST BE 0, RES. FOR FUTURE USE
IDKUSRW EQU X'03' 0-1 :S: IOUSAGE = *READ_WRITE 110
IDKUSWR EQU X'02' 1-1 :S: IOUSAGE = *WRITE 110
IDKUSRD EQU X'01' 0-0 :S: IOUSAGE = *READ 110
IDKUSNS EQU X'00' IOUSAGE NOT SPECIFIED 110
*
IDKDW DS X DISK-WRITE .#'0D8' 110
IDKDWU EQU X'FC' 2-7 -- MUST BE 0, RES. FOR FUTURE USE
IDKDWCL EQU X'02' 1-1 :S: DISKWR = *BY_CLOSE 110
IDKDWIM EQU X'01' 0-0 :S: DISKWR = *IMMEDIATE 110
IDKDWNS EQU X'00' DISKWR NOT SPECIFIED 110
*
IDKFLAG4 DS X FLAG 4 .#'0D9' 110
IDKGRDN EQU X'80' 7-7 :S: GUARDS = *NONE 117
IDKFTS EQU X'40' 6-6 -- RESERVED FOR FTS -- 117
IDKCCSST EQU X'20' 5-5 :S: CCS = *STD 117
IDKWRKGP EQU X'10' 4-4 :S: WORKGRP = *YES 201
IDKRDGS EQU X'08' 3-3 :S: READ GUARD SPECIFIED 110
IDKWRGS EQU X'04' 2-2 :S: WRITE GUARD SPECIFIED 110
15:29:06 2001-10-15 PAGE
0031
SOURCE STATEMENT
IDKXGS EQU X'02' 1-1 :S: EXEC GUARD SPECIFIED 110
IDKCCSS EQU X'01' 0-0 :S: CODED CHARACTER SET SPEC. 110
*
IDKRDG DS CL18 READ GUARD .#'0DA' 110
IDKWRG DS CL18 WRITE GUARD .#'0EC' 110
IDKXG DS CL18 EXEC GUARD .#'0FE' 110
*
IDKCHECK DS X CHECK .#'110' 200
IDKCKSTD EQU 0 *STD (DEFAULT) 200
IDKCKNO EQU 1 *NO 200
IDKCKERR EQU 2 *ERROR 200
IDKCKSIN EQU 3 *SINGLE 200
IDKCKMUL EQU 4 *MULTIPLE 200
IDKCKUID EQU 5 *USERID 200
IDKCKCAT EQU 6 *CATALOG 200
*
IDKLIST DS X LIST .#'111' 200
IDKLSNO EQU 0 *NO (DEFAULT) 200
IDKLSEOU EQU 1 *ERRORS_TO_SYSOUT 200
IDKLSOUT EQU 2 *SYSOUT 200
*
IDKAVAIL DS X AVAIL .#'112' 200
IDKAVANS EQU 0 NOT SPECIFIED 200
IDKAVAST EQU 1 *STD 200

```

IDKAVAHI	EQU	2		*HIGH		200
*						
IDKSOMIG	DS	X		SOMIGR	.# '113'	200
IDKSOMNS	EQU	0		NOT SPECIFIED		200
IDKSOMAL	EQU	1		*ALLOWED		200
IDKSOMFB	EQU	4		*FORBIDDEN		200
*						
IDKCCS	DS	CL8		CODED CHARACTER SET	.# '114'	110
*						
IDKFTAM	DS	XL8		RESERVED AREA FOR FTAM	.# '11C'	110
*						
IDKFLAG6	DS	X		FLAG 6 (ONLY FOR TPR CALL)	.# '124'	116
IDKFLG6U	EQU	X'F8'		4-7 MUST BE 0, RES. FOR FUTURE USE		116
IDKPECTC	EQU	X'04'		3-3 :S: RES. FOR DMS		204
IDKPCOPY	EQU	X'03'		1-2 :S: RES. FOR DMS		116
IDKPMODI	EQU	X'02'		2-2 :S: RES. FOR DMS		116
IDKPCREA	EQU	X'01'		1-1 :S: RES. FOR DMS		116
*						
IDKTPFG1	DS	X		TRR FLAG1	.# '125'	120
IDKDNERN	EQU	X'80'		7-7 :S: DNTERAS=*NO (TPR)		120
IDKDNERY	EQU	X'C0'		6-7 :S: DNTERAS=*YES (TPR)		120
IDKURSFN	EQU	X'20'		5-5 :S: USRSYSF=*NO (TPR)		120
IDKURSFY	EQU	X'30'		4-5 :S: USRSYSF=*YES (TPR)		120
IDKERF5N	EQU	X'08'		3-3 :S: ERADF5R=*NO (TPR)		120
IDKERF5Y	EQU	X'0C'		2-3 :S: ERADF5R=*YES (TPR)		120
IDKSRMBN	EQU	X'02'		1-1 :S: SRMBKUP=*NO (TPR)		201
IDKSRMBY	EQU	X'03'		0-1 :S: SRMBKUP=*YES (TPR)		201
*						
IDKUNUS1	DS	X		-- UNUSED --	.# '126'	200
*						
IDKPROTP	DS	X		PROTECT - TYPE	.# '127'	200
IDKPRONS	EQU	0		PROTECT NOT SPECIFIED		200
IDKPROFI	EQU	1		PROTECT=(*FROM_FILE , ...)		200
IDKPROST	EQU	2		PROTECT=*STD		200
*						
IDKSECRF	DS	CL54		PROTECT - REFERENCE	.# '128'	200
*						
IDKPRFLG	DS	X		PROTECT - FLAG	.# '15E'	200
IDKSHAUC	EQU	X'80'		7-7 :S: SHARE=*UNCHANGED		200
IDKACCUC	EQU	X'40'		6-6 :S: ACCESS=*UNCHANGED		200
IDKBALUC	EQU	X'20'		5-5 :S: BASACL=*UNCHANGED		200
IDKGRDUC	EQU	X'10'		4-4 :S: GUARDS=*UNCHANGED		200
IDKRSPUC	EQU	X'08'		3-3 :S: RELSPAC=*UNCHANGED		200
IDKEXDUC	EQU	X'04'		2-2 :S: EXDATE=*UNCHANGED		200
IDKPRFGU	EQU	X'03'		0-1 -- UNUSED, MUST BE 0 --		200
*						
IDKFLAG7	DS	X		FLAG 7	.# '15F'	200
IDKWCFBD	EQU	X'80'		7-7 :S: WILDCARDS FORBIDDEN		200

IDKEXDSP	EQU	X'40'	6-6 :S: EXDATE SPECIFIED	200
IDKDEDSP	EQU	X'20'	5-5 :S: DELDATE SPECIFIED	200
IDKMANS	EQU	X'10'	4-4 :S: MANCLAS SPECIFIED	200
IDKUSISP	EQU	X'08'	3-3 :S: USRINFO SPECIFIED	200
IDKADISP	EQU	X'04'	2-2 :S: ADMINFO SPECIFIED	200
IDKTBLTI	EQU	X'01'	0-0 :S: TIMEBASE LTI	210
IDKFLG7U	EQU	X'02'	1-1 -- UNUSED, MUST BE 0 --	210
*				
IDKEXDAT	DS	CL10	EXDATE DATE	.# '160' 200
IDKEXTIM	DS	CL8	EXDATE TIME	.# '16A' 210
*				
IDKDEDAT	DS	CL10	DELDATE DATE	.# '172' 200
IDKDETIM	DS	CL8	DELDATE TIME	.# '17C' 210
*				
IDKMANCL	DS	CL8	MANCLAS	.# '184' 200
*				
IDKUSINF	DS	CL8	USRINFO	.# '18C' 200
*				
IDKADINF	DS	CL8	ADMINFO	.# '194' 200
*				
IDKUNUS2	DS	XL1	-- UNUSED2, MUST BE 0 --	.# '19C' 200
*				
IDKSTCLF	DS	XL1	STOCLAS	.# '19D' 200
IDKSCNSP	EQU	0	NOT SPECIFIED	200
IDKSCNON	EQU	1	*NONE	200
IDKSCUPD	EQU	2	*UPDATE	200
IDKSCSTD	EQU	3	*STD	200
IDKSCNAM	EQU	4	NAME SPECIFIED	200
IDKSTCLN	DS	CL8	STOCLAS-NAME	.# '19E' 200
*				
IDKFUTUR	DS	2XL1	-- UNUSED, MUST BE 0 --	.# '1A6' 210
IDKPLEN	EQU	*-IDKCATST	LENGTH OF PARAMETERAREA = X'1A8' (424)	

6.1.2 FILE

```

FILE MF=D,VERSION=3
      *,VERSION 400
*-----*
* FILE MACRO PARAMETER LIST VERSION 3 *
*-----*
      #INTF INTNAME=FILE,
                                     C
15:29:06 2001-10-15 PAGE
0033
SOURCE STATEMENT
      REFTYPE=REQUEST,
                                     C
      INTCOMP=4
      MFPRE MF=D,PREFIX=,
                                     201C
      DNAME=DBPFLPL,MACID=DB,DMACID=DB,ALIGN=F
                                     201
IDBPFLPL DSECT ,
      *,##### PREFIX=I, MACID=DB #####
IDBPFLST EQU *
      FHDR MF=(C,IDBF),EQUATES=NO
      DS 0A
IDBFFHE DS 0XL8 0 GENERAL PARAMETER AREA HEADER
*
IDBFIFID DS 0A 0 INTERFACE IDENTIFIER
IDBFFCTU DS AL2 0 FUNCTION UNIT NUMBER
* BIT 15 HEADER FLAG BIT,
* MUST BE RESET UNTIL FURTHER NOTICE
* BIT 14-12 UNUSED, MUST BE RESET
* BIT 11-0 REAL FUNCTION UNIT NUMBER
IDBFFCT DS AL1 2 FUNCTION NUMBER
IDBFFCTV DS AL1 3 FUNCTION INTERFACE VERSION NUMBER
*
IDBFRET DS 0A 4 GENERAL RETURN CODE
IDBFSRET DS 0AL2 4 SUB RETURN CODE
IDBFSR2 DS AL1 4 SUB RETURN CODE 2
IDBFSR1 DS AL1 5 SUB RETURN CODE 1
IDBFMRET DS 0AL2 6 MAIN RETURN CODE
IDBFMR2 DS AL1 6 MAIN RETURN CODE 2
IDBFMR1 DS AL1 7 MAIN RETURN CODE 1
IDBFFHL EQU 8 8 GENERAL OPERAND LIST HEADER LENGTH
*
IDBLINK DS CL8 LINK
IDBFILE DS CL54 FILENAME
IDBRETPD DS Y RETENTION PERIOD
IDBRECSI DS Y RECORD SIZE
IDBBLKSZ DS Y BLOCK SIZE 004
IDBKEYPO DS Y KEY POSITION
IDBKEYLE DS AL1 KEY LENGTH

```

IDBCTRLI	DS	X	BLKCTRL-INDIKATOR	952
IDBCTRLN	EQU	X'80'	7-7 S BLKCTRL=NO	952
IDBCTRLP	EQU	X'40'	6-6 S BLKCTRL=PAMKEY	952
IDBCTRLD	EQU	X'20'	5-5 S BLKCTRL=DATA	952
IDBCTRL0	EQU	X'10'	4-4 S BLKCTRL=NULL	952
IDBCTRLB	EQU	X'08'	3-3 S BLKCTRL=*BY-PROG	200
IDBCTRLR	EQU	X'F8'	7-3 R BLKCTRL=NOT SPECIFIED	200
IDBCTRLU	EQU	X'04'	R -- RESERVED, MUST BE 0 --	200
IDBCF4K	EQU	X'02'	1-1 S BLOCK CONTROL FIELD 4K	200
IDBCF2K	EQU	X'01'	0-0 S BLOCK CONTROL FIELD 2K	200
IDBPAD	DS	AL1	PAD	
IDBOPEN	DS	X	OPEN TYPE	
IDBSINOT	EQU	X'80'	7-7 S SINOUT	
IDBOUTIN	EQU	X'40'	6-6 S OUTIN	
IDBINOUT	EQU	X'20'	5-5 S INOUT	
IDBUPDAT	EQU	X'10'	4-4 S UPDATE	
IDBEXTEN	EQU	X'08'	3-3 S EXTEND	
IDBOUTPU	EQU	X'04'	2-2 S OUTPUT	
IDBREVER	EQU	X'02'	1-1 S REVERSE	
IDBINPUT	EQU	X'01'	0-0 S INPUT	
IDBDEVL	DS	CL8	DEVICE TYP	950
IDBIND1	DS	X	INDICATORS (SET,RESET)	
IDBLABEL	EQU	X'60'	6-5 FILE LABEL OPTIONS	
IDBSTD	EQU	X'60'	R STD LABELS	
IDBNSTD	EQU	X'40'	S NSTD LABELS	
IDBLABNO	EQU	X'20'	S LABEL=NO	
IDBAMN	EQU	X'18'	4-3 PRINTER CONTROL	
IDBA	EQU	X'18'	S ASA CONTROL CHARS	
IDBM	EQU	X'08'	S EBCDIC CONTROL CHARS	
IDBN	EQU	X'18'	R NO CONTROL CHARS	
IDBRECFO	EQU	X'06'	2-1 RECORD FORM	
IDBF	EQU	X'04'	S FIXED LENGTH	
IDBV	EQU	X'02'	S VARIABLE LENGTH	
IDBU	EQU	X'06'	S UNDEFINED	
IDBTMRK	EQU	X'01'	0-0 S TPMARK=YES	
IDBIND1U	EQU	X'80'	R -- RESERVED, MUST BE 0 --	950
IDBIND2	DS	X	INDICATORS (SET,RESET)	
IDBFCBTY	EQU	X'C0'	7-6 FCB TYPE	
IDBSAM	EQU	X'C0'	R SAM	
IDBISAM	EQU	X'40'	S ISAM	
IDBPAM	EQU	X'C0'	S PAM	
IDBBTAM	EQU	X'80'	S BTAM	
IDBDUPKE	EQU	X'20'	5-5 S DUPLICATE KEYS ALLOWED	
IDBDUPKO	EQU	X'20'	5-5 R DUPEKY=NO	
IDBOVERL	EQU	X'10'	4-4 S OVERLAP=YES	
IDBOVERO	EQU	X'10'	4-4 R OVERLAP=NO	
IDBSHARW	EQU	X'08'	3-3 S SHARUPD=WEAK	050
IDBSHARU	EQU	X'04'	2-2 S SHARUPD=YES	

```

*
IDBSHARN EQU X'01' 3-2 R SHARUPD=NO
IDBIND2U EQU X'02' 0-0 S SHARUPD SPECIFIED
IDBIND3 DS X R -- RESERVED, MUST BE 0 -- 050
IDBSPAKE EQU X'80' 7-7 S SPACE=(...,*KEEP) 050
IDBBSSTD EQU X'40' 6-6 S BLKSIZE=STD 004
IDB2NDSP EQU X'20' 5-5 S 2ND SPACE PARAM SPECIFIED 004
IDBSPACE EQU X'10' 4-4 S SPACE=ABS
IDBSTATE EQU X'08' 3-3 S STATE=FOREIGN
IDBVOLUM EQU X'04' 2-2 S VOLUME=PRIVATE
IDBMNTQ0 EQU X'02' 1-1 S MOUNT=0
IDBDMYPR EQU X'01' 0-0 S *DUMMY & VOL=PRIVATE (TPR) 009
IDBIND4 DS X INDICATORS (SET,RESET)
IDBFILEN EQU X'80' 7-7 S FILE NOT SPECIFIED
IDBCLOSN EQU X'40' 6-6 S CLOSE NOT SPECIFIED 052
IDBBLKSN EQU X'20' 5-5 S BLKSIZE=NOT SPECIFIED
IDBDUPKN EQU X'10' 4-4 S DUPEKY=NOT SPECIFIED
IDBRETPN EQU X'08' 3-3 S RETPD=NOT SPECIFIED
IDBPADN EQU X'04' 2-2 S PAD=NOT SPECIFIED
IDBKEYPN EQU X'02' 1-1 S KEYPOS=NOT SPECIFIED
IDBOPENN EQU X'01' 0-0 S OPEN=NOT SPECIFIED
IDBIND5 DS X INDICATORS (SET,RESET)
IDBKEYLN EQU X'80' 7-7 S KEYLEN=NOT SPECIFIED
IDBLABEN EQU X'40' 6-6 S LABEL=NOT SPECIFIED
IDBTVSNT EQU X'20' 5-5 S TVSN=SPECIFIED
IDBOVERN EQU X'10' 4-4 S OVERLAP=NOT SPECIFIED
IDBRECSN EQU X'08' 3-3 S RECSIZE=NOT SPECIFIED
IDBDEVIN EQU X'04' 2-2 S DEVICE=NOT SPECIFIED
IDBFCBTN EQU X'01' 0-0 S FCBTYP=NOT SPECIFIED
IDBIND5U EQU X'02' R -- RESERVED, MUST BE 0 -- 950
IDBIND6 DS X INDICATORS (SET,RESET)
IDBRECFN EQU X'80' 7-7 S RECFORM=NOT SPECIFIED
IDBVOLUN EQU X'40' 6-6 S VOLUME=NOT SPECIFIED
IDBSPACN EQU X'20' 5-5 S SPACE=NOT SPECIFIED
IDBSTATN EQU X'08' 3-3 S STATE=NOT SPECIFIED
IDBMOUNN EQU X'02' 1-1 S MOUNT=NOT SPECIFIED
IDBIND6U EQU X'15' R -- RESERVED, MUST BE 0 -- 950
IDBVOL@ DS A ADDRESS OF VOLUME-LIST 004
IDBDVOL@ DS A ADDRESS OF DVOLUME-LIST 004
IDBTVSN@ DS A ADDRESS OF TVSN-LIST 004
IDBMNT@ DS A ADDRESS OF MOUNT-LIST 004
IDBVSQ@ DS A ADDRESS OF VSEQ-LIST 004
IDBNUSE1 DS XL16 -- RESERVED, MUST BE 0 -- 004
IDBPRMHP DS 0A PRIMARY # OF HALFPAGES (SPACE)
IDBFSTHP DS A FIRST HALFPAGE (SPACE=ABS)
IDBSCDHP DS 0A SECONDARY # OF HALFPAGES (SPACE) 004
IDBAMNT DS A NUMBER OF HALFPAGES (SPACE=ABS) 004
IDBPRVL# DS AL1 NUMBER OF ANY PRIVATE VOLUMES 004

```

IDBIND7	DS	X	INDICATOR	
IDBBLKNL	EQU	X'80'	7-7 S BLKSIZE=NULL	
IDBKYPNL	EQU	X'40'	6-6 S KEYPOS=NULL	
IDBKYLNL	EQU	X'20'	5-5 S KEYLEN=NULL	
IDBRESNL	EQU	X'10'	4-4 S RECSIZE=NULL	
IDBFCBNL	EQU	X'08'	3-3 S FCBTYP= NULL	
IDBREFNL	EQU	X'04'	2-2 S RECFORM=NULL	
IDBCODNL	EQU	X'02'	1-1 S CODE=NULL	
IDBBUFNL	EQU	X'01'	0-0 S BUFOFF=NULL	800
IDBIND8	DS	X	INDICATOR	
IDBVLPNL	EQU	X'80'	7-7 S VALPROP=NULL	
IDBLOGNL	EQU	X'40'	6-6 S LOGLEN=NULL	
IDBVALNL	EQU	X'20'	5-5 S VALLEN=NULL	
IDBWROUT	EQU	X'10'	4-4 S WROUT=YES	
IDBIND8U	EQU	X'0F'	R -- RESERVED, MUST BE 0 --	950
IDBIND9	DS	X	INDICATOR	
IDBDDEVN	EQU	X'80'	7-7 S DDEVICE=NOT SPECIFIED	
IDBDVOLN	EQU	X'40'	6-6 S DVOLUME=NOT SPECIFIED	
IDBDSPAN	EQU	X'20'	5-5 S DSPACE=NOT SPECIFIED	
IDBVL PNS	EQU	X'10'	4-4 S VALPROP=NOT SPECIFIED	
IDBLOGNS	EQU	X'08'	3-3 S LOGLEN=NOT SPECIFIED	
IDBVALNS	EQU	X'04'	2-2 S VALLEN=NOT SPECIFIED	
IDBCODEN	EQU	X'02'	1-1 S CODE=NOT SPECIFIED	
IDBWRONS	EQU	X'01'	0-0 S WROUT=NOT SPECIFIED	
IDBIIND	DS	X	ISAM INDICATOR	
IDBPREFS	EQU	X'80'	7-7 S POOLLNK SPECIFIED	950
IDB2NDDS	EQU	X'40'	6-6 S 2ND DSPACE PARAM SPECIFIED	004
IDBDSPAC	EQU	X'10'	4-4 S DSPACE=ABS	
IDBDVOLU	EQU	X'04'	2-2 S DVOLUME=PRIVATE	
IDBVMIN	EQU	X'01'	0-0 R MINIMUM FUNCTION FOR ISAM	
*			VALUE FLAG	
IDBVMAX	EQU	X'01'	0-0 S MAXIMUM FUNCTION FOR ISAM	
*			VALUE FLAG	
IDBIINDU	EQU	X'2A'	R -- RESERVED, MUST BE 0 --	004
IDBLOGLN	DS	AL1	ISAM LOGICAL FLAG LENGTH	
IDBVALLN	DS	AL1	ISAM VALUE FLAG LENGTH	
IDBDEVDL	DS	CL8	DEVICE TYPE FOR ISAM DATA VSN	950
IDBCODE	DS	X	CODE FOR TAPE	
IDBISOD	EQU	X'20'	5-5 S ISO7D CODE	200
IDBISO7	EQU	X'10'	4-4 S ISO7 CODE	
IDBEBCD	EQU	X'08'	3-3 S EBCDIC CODE	
IDBOWN	EQU	X'04'	2-2 S OWN CODE	
IDBNDEF	EQU	X'3C'	5-2 R CODE NOT DEFINED	200
IDBCODEU	EQU	X'C3'	R -- RESERVED, MUST BE 0 --	200
IDBIGNP	DS	X	IGNORE PROTECTION (TPR)	008
IDBIACC	EQU	X'80'	7-7 S IGNORE ACCESS-RIGHTS	008
IDBIEXD	EQU	X'40'	6-6 S IGNORE EXDATE	008
IDBIRDPW	EQU	X'20'	5-5 S IGNORE RDPASS	008

IDBIWRPW	EQU	X'10'	4-4 S IGNORE WRPASS	008
IDBIEXPW	EQU	X'08'	3-3 S IGNORE EXPASS	008
IDBIGNPU	EQU	X'07'	R -- RESERVED MUST BE 0 --	008
IDBPERF	DS	X	IOPERF-INDICATOR	050
IDBPFUM	EQU	X'FF'	7-0 S IOPERF=USER-MAX	050
IDBPFNL	EQU	X'FE'	7-1 S IOPERF=NULL	050
IDBPFVH	EQU	X'03'	1-0 S IOPERF=VERY-HIGH	050
IDBPFHI	EQU	X'02'	1-1 S IOPERF=HIGH	050
IDBPFST	EQU	X'01'	0-0 S IOPERF=STD	050
IDBPFNS	EQU	X'00'	S IOPERF NOT SPECIFIED	050
IDBUSAG	DS	X	IOUSAGE-INDICATOR	050
IDBUSNL	EQU	X'FF'	7-0 S IOUSAGE=NULL	050
IDBUSRW	EQU	X'03'	1-0 S IOUSAGE=RDWRT	050
IDBUSWR	EQU	X'02'	1-1 S IOUSAGE=WRITE	050
IDBUSRD	EQU	X'01'	0-0 S IOUSAGE=READ	050
IDBUSNS	EQU	X'00'	S IOUSAGE NOT SPECIFIED	050
IDBBUFOF	DS	AL1	BUFFER OFFSET VALUE	
IDBFSEQ	DS	Y	FSEQ VALUE	
IDBTSET	DS	CL4	TSETNAME	
IDBFSID	DS	CL6	FIRST VSN FOR TSET ENTRY	
IDBVSQVL	DS	AL1	VSEQ VALUE	004
IDBIND10	DS	X		
IDBBUFFN	EQU	X'80'	7-7 S BUFOFF NOT SPECIFIED	
IDBTYPLN	EQU	X'40'	6-6 S TYPLAB NOT SPECIFIED	
IDBTRANN	EQU	X'20'	5-5 S TRANS NOT SPECIFIED	
IDBFSEQN	EQU	X'10'	4-4 S FSEQ NOT SPECIFIED	
IDBSECLN	EQU	X'08'	3-3 S SECLEV NOT SPECIFIED	
IDBWRCHN	EQU	X'04'	2-2 S WRCHK NOT SPECIFIED	751
IDBIN10U	EQU	X'03'	R -- RESERVED, MUST BE 0 --	950
IDBIND11	DS	X		
IDBWRCHK	EQU	X'80'	7-7 S WRCHK =YES	751
IDBTRANS	EQU	X'40'	6-6 S TRANS =YES	
IDBOPR	EQU	X'20'	5-5 S SECLEV = (... ,OPR)	
IDBBLP	EQU	X'10'	4-4 S SECLEV = LOW	701
IDBDIN3	EQU	X'08'	3-3 S LABEL=(STD,3)	
IDBDIN2	EQU	X'04'	2-2 S LABEL=(STD,2)	
IDBDIN1	EQU	X'02'	1-1 S LABEL=(STD,1)	
IDBBS2	EQU	X'01'	0-0 S LABEL=(STD,0)	
IDBIND12	DS	X		
IDBBLPID	EQU	X'80'	7-7 S BYPASS LABEL PROCESSING	803
IDBRSDUM	EQU	X'40'	6-6 S DUMMY FOR RESTART	750
IDBCKPFE	EQU	X'20'	5-5 S CHKPT WHEN FEOV EXECUTED	750
IDBCKPBL	EQU	X'10'	4-4 S CHKPT WHEN BLIM OCCURS	750
IDBFSEQV	EQU	X'08'	2-2 S FSEQ VALUE SPECIFIED	200
IDBFSNEW	EQU	X'04'	1-1 S FSEQ = NEW	200
IDBFSUNK	EQU	X'02'	0-0 S FSEQ = UNK	200
IDBFSNUL	EQU	X'01'	7-7 S FSEQ = NULL	200
IDBIND13	DS	X		

IDBVSEQL	EQU	X'80'	7-7 S VSEQ IS A LIST	200
IDBUFL	EQU	X'40'	6-6 S BUFOFF = L	
IDBPOLTP	EQU	X'20'	5-5 S POOL TAPE:NO VOLSWITCH	801
IDBSTRMY	EQU	X'10'	4-4 S STREAM=YES	802
IDBUCON	EQU	X'08'	3-3 S ALLOC: USER CTRL ON (TPR)	750
IDBUOFF	EQU	X'04'	2-2 S ALLOC: USER CTRL OFF (TPR)	750
IDBL4CHK	EQU	X'02'	1-1 S ALLOC: LEVEL 4 CHECK (TPR)	750
IDBL5CHK	EQU	X'01'	0-0 S ALLOC: LEVEL 5 CHECK (TPR)	750
IDBIND14	DS	X		
IDBTSETN	EQU	X'80'	7-7 S TSET NOT SPECIFIED	
IDBVSEQN	EQU	X'40'	6-6 S VSEQ NOT SPECIFIED	
IDBIN14U	EQU	X'38'	R -- RESERVED, MUST BE 0 --	220
IDBVOLRU	EQU	X'04'	2-2 S VOLUME=REMOVE-UNUSED	220
IDBLKENX	EQU	X'02'	1-1 S LOCKENV=XCS	200
IDBLKENS	EQU	X'01'	0-0 S LOCKENV SPECIFIED	200
IDBIND15	DS	X		950
IDBDESCS	EQU	X'80'	7-7 S DESTOC SPECIFIED	950
IDBDESCY	EQU	X'40'	6-6 S DESTOC = YES	950
IDBCMSG	EQU	X'20'	5-5 S CLOSMG SPECIFIED	950
IDBCMSGY	EQU	X'10'	4-4 S CLOSMG = YES	950
IDBTAPWS	EQU	X'08'	3-3 S TAPEWR SPECIFIED	950
IDBTAPWY	EQU	X'04'	2-2 S TAPEWR = DEVICE-BUFFER	950
IDBX32GS	EQU	X'02'	1-1 S EXC32GB SPECIFIED	400
IDBX32GA	EQU	X'01'	0-0 S EXC32GB=ALLOWED	400
IDBBLIM	DS	AL3	BLOCK NUMBER LIMIT	750
IDBCHAIN	DS	Y	CHAIN VALUE (STREAMING TAPES)	802
IDBBLPPS	DS	Y	BYPASS POSITION	803
IDBBLPVL	DS	Y	BYPASS VALUE	803
IDBPLREF	DS	CL8	POOLLNK VALUE	950
IDBPHPID	DS	0A	PRIMARY # OF HALFPAGES (DSPACE)	
IDBFSTID	DS	A	FIRST HALFPAGE (DSPACE=ABS)	004
IDBSCDID	DS	0A	SECONDARY # OF HALFPAGES (DSPACE)	
IDBAMTID	DS	A	NUMBER OF HALFPAGES (DSPACE=ABS)	004
IDBPRDV#	DS	AL1	NUMBER OF ANY PRIVATE DVOLUMES	004
IDBCLOSE	DS	X	CLOSE TYPE	052
IDBCLSLV	EQU	X'00'	7-0 R CLOSE=LEAVE	052
IDBCLSDC	EQU	X'01'	0-0 S CLOSE=DISCON	052
IDBCLSRP	EQU	X'02'	1-1 S CLOSE=REPOS	052
IDBCLSRW	EQU	X'03'	1-0 S CLOSE=RWD	052
IDBCLSIV	EQU	X'05'	2,0 S CLOSE=INVAL	052
IDBCLSKD	EQU	X'06'	2-1 S CLOSE=KEEP-DATA-IN-CACHE	201
IDBIND16	DS	X		200
IDBWRKFY	EQU	X'80'	7-7 S WORKFIL=YES	200
IDBAVHI	EQU	X'40'	6-6 S AVAIL=HIGH	200
IDBIN16U	EQU	X'3E'	R -- RESERVED, MUST BE 0 --	240
IDBWRKFS	EQU	X'01'	0-0 S WORKFIL SPECIFIED	240
IDBIND17	DS	X		200
IDBSTCLS	EQU	X'80'	7-7 S STOCLAS SPECIFIED	200

IDBVSETS	EQU	X'40'	6-6 S VOLSET SPECIFIED	200
IDBVSLO	EQU	X'20'	5-5 S ONLY VSETLIST FROM SC (TPR)	200
IDBSCNO	EQU	X'10'	4-4 S STOCLAS=*NONE	202
IDBVSKS0	EQU	X'08'	3-3 S VOLSET: KEEP S0-MIGR (TPR)	260
IDBIN17U	EQU	X'07'	R -- RESERVED, MUST BE 0 --	260
IDBSTOCL	DS	CL8	STOCLAS VALUE	200
IDBVSET	DS	CL4	VOLSET VALUE	200
IDBVSCTL	EQU	C'*CVS'	VOLSET=*CONTROL	230
IDBIND18	DS	X		200
IDBBLSBP	EQU	X'80'	7-7 S BLKSIZE BY PROG	200
IDBKYPBP	EQU	X'40'	6-6 S KEYPOS BY PROG	200
IDBKYLPBP	EQU	X'20'	5-5 S KEYLEN BY PROG	200
IDBRESBP	EQU	X'10'	4-4 S RECSIZE BY PROG	200
IDBFCBBP	EQU	X'08'	3-3 S FCBTYP BY PROG	200
IDBREFFBP	EQU	X'04'	2-2 S RECFORM BY PROG	200
IDBCODBP	EQU	X'02'	1-1 S CODE BY PROG	200
IDBBUFBP	EQU	X'01'	0-0 S BUFOFF BY PROG	200
IDBIND19	DS	X		200
IDBVLBPBP	EQU	X'80'	7-7 S VALPROP BY PROG	200
IDBLOGBP	EQU	X'40'	6-6 S LOGLLEN BY PROG	200
IDBVALBP	EQU	X'20'	5-5 S VALLEN BY PROG	200
IDBLABBP	EQU	X'10'	3-3 S LABEL BY PROG	200
IDBIN19U	EQU	X'0F'	R -- RESERVED, MUST BE 0 --	200
IDBDSKW	DS	XL1	DISKWR	202
IDBDWCL	EQU	X'02'	1-1 S DISKWR=BY-CLOSE	202
IDBDWIM	EQU	X'01'	0-0 S DISKWR=IMMEDIATE	202
IDBDWNS	EQU	X'00'	S DISKWR NOT SPECIFIED	202
IDBDATRI	DS	X		200
IDBDATRU	EQU	X'FE'	R -- RESERVED, MUST BE 0 --	200
IDBDATFF	EQU	X'01'	0-0 S DATATTR FROM FILE	200
IDBDATRF	DS	CL54	DATATTR REFERENCE FILENAME	200
IDBIND20	DS	X		210
IDBNEWCE	EQU	X'80'	7-7 S CREATE NEW CE (TPR)	210
IDBNDADM	EQU	X'40'	6-6 S TSOS PRIVILEGE NEEDED (TPR)	220
IDBNDPHA	EQU	X'20'	5-5 S PHYSICAL ALLOC NEEDED (TPR)	220
IDBPFNAJ	EQU	X'10'	4-4 S NO IOPERF ADJUST (TPR)	260
IDBVSUIS	EQU	X'08'	3-3 S VOLSETSEL USERID SPEC (TPR)	260
IDBALTFN	EQU	X'04'	2-2 S NO ALTERNATIVE FORMAT (TPR)	262
IDBIN20U	EQU	X'03'	R -- RESERVED, MUST BE 0 --	262
IDBVSUID	DS	CL8	USERID FOR VOLSET SELECTION (TPR)	260
IDBIND21	DS	X		270
IDBDEFPN	EQU	X'80'	7-7 S NO DEFAULT PROTECTION	270
IDBIN21U	EQU	X'7F'	R -- RESERVED, MUST BE 0 --	270
IDBNUSE4	DS	XL4	-- RESERVED, MUST BE 0 --	270
IDBPARN	EQU	*-IDBPFLST	LENGTH OF PARAM LIST	004

6.1.3 ERASE

```

        ERASE MF=D,VERSION=3
        #INTF REFTYPE=REQUEST,INTNAME=ERASE,INTVERS=3
        MFPRE MF=D,DMACID=D,DNAME=ERAPL,PREFIX=IER
IERERAPL DSECT ,
        *,##### PREFIX=IER, MACID=D #####
*
IERERAST DS      0A
*
        FHDR MF=(C,IER),EQUATES=NO
        DS      0A
IERFHE  DS      0XL8          0   GENERAL PARAMETER AREA HEADER
*
IERIFID DS      0A          0   INTERFACE IDENTIFIER
IERFCTU DS      AL2          0   FUNCTION UNIT NUMBER
        BIT 15   HEADER FLAG BIT,
*
        MUST BE RESET UNTIL FURTHER NOTICE
*
        BIT 14-12 UNUSED, MUST BE RESET
*
        BIT 11-0  REAL FUNCTION UNIT NUMBER
IERFCT  DS      AL1          2   FUNCTION NUMBER
IERFCTV DS      AL1          3   FUNCTION INTERFACE VERSION NUMBER
*
IERRET  DS      0A          4   GENERAL RETURN CODE
IERSRET DS      0AL2         4   SUB RETURN CODE
IERSR2  DS      AL1          4   SUB RETURN CODE 2
IERSR1  DS      AL1          5   SUB RETURN CODE 1
IERMRET DS      0AL2         6   MAIN RETURN CODE
IERMR2  DS      AL1          6   MAIN RETURN CODE 2
IERMR1  DS      AL1          7   MAIN RETURN CODE 1
IERFHL  EQU      8          8   GENERAL OPERAND LIST HEADER LENGTH
*
IERPNAM DC      CL80' '      PATHNAME
*
*
IERACTIO DC     AL1(0)       ACTION -
IERSPCAT EQU    0           SPACE-CATALOG
IERSPACE EQU    1           SPACE
IERDATA  EQU    2           DATA
IERCATLG EQU    3           CATALOG
IEREXPRT EQU    3           EXPORT
IERSPCEX EQU    4           DELETE-OR-EXPORT
*
IERATTR  DC     B'00000000'  ATTRIBUTES -
IERDSTRO EQU    X'80' 7-7 S  DESTROY
IERMTALL EQU    X'40' 6-6 S  MOUNT ALL DISKS
IERMT1ST EQU    X'40' 6-6 R  MOUNT FIRST DISK

```

```

IERATTNU EQU  X'3C'   5-2 R  -- UNUSED, MUST BE 0 --          210
IERTBLTI EQU  X'02'   1-1 S      TIME BASE LTI              210
IERKPACL EQU  X'01'   0-0 S      KEEP ACL
*
IERIGNOR DC   B'00000000'      IGNORE -
IERIGACC EQU  X'80'   7-7 S      ACCESS
IERIGRTP EQU  X'40'   6-6 S      EXPIRATION DATE
IERIGWRP EQU  X'20'   5-5 S      WRITE PASSWORD
IERIGRDP EQU  X'10'   4-4 S      READ PASSWORD
IERIGEXP EQU  X'08'   3-3 S      EXEC PASSWORD
IERIGNNU EQU  X'07'   2-0 R  -- UNUSED, MUST BE 0 --
*
IEROUTPT DC   B'00000000'      OUTPUT -
IERLIST EQU   X'80'   7-7 S      LIST ADDRESS IS GIVEN
IERLIST@ EQU  X'40'   6-6 R      ABSOLUTE ADDRESS
IERLIST@ EQU  X'40'   6-6 S      ADDRESS IN REGISTER
IERLIST#A EQU  X'20'   5-5 R      ABSOLUTE LENGTH
IERLIST#R EQU  X'20'   5-5 S      LENGTH IN REGISTER
IERLISTSY EQU  X'10'   4-4 S      ERRORS TO SYSOUT
IEROUTNU EQU  X'0F'   3-0 R  -- UNUSED, MUST BE 0 --
*
IERCHECK DC   AL1(0)          CHECK -
IERCKNO EQU   0              NO
IERCKERR EQU  1              ERROR
IERCKSIN EQU  2              SINGLE
IERCKMUL EQU  3              MULTIPLE
IERCKUID EQU  4              USERID
IERCKPVS EQU  5              PVS
IERCKSTD EQU  255           STD
*
IERFUT1 DC    3X'00'         FOR FUTURE USE, MUST BE ZERO
*
IERLIST@ DC   A(0)           LIST ADDRESS
IERLIST# DC   A(0)           LIST SIZE
*
IERPWARR DC   A(0,0,0)       3 PW'S
*
IERNSTEP DC   Y(0,0,0)       3 ERR'S
*
IERPOSF DC    B'00000000'     POS-FLAG -
IERPOSB EQU   X'80'   7-7 S     POS=BEFORE
IERPOSA EQU   X'40'   6-6 S     POS=AFTER
IERPOSNU EQU  X'3F'   5-0 R  -- UNUSED, MUST BE 0 --
*
IERFUT2 DC    5X'00'         FOR FUTURE USE, MUST BE ZERO
*
DMSSEL MF=D,PREFIX=IER,VERSION=3
MFCHK MF=C,SUPPORT=(D,C),PREFIX=I,MACID=XX,DMACID=YYY

```

```
DS 0F
*,##### PREFIX=I, MACID=XXY #####
*
*
DS 0A
*
IERSELI0 DC B'00000000' SELECT IND0
IERSIACC EQU X'80' ACCESS
IERSIBKP EQU X'40' BACKUP-LEVEL
IERSIFCB EQU X'20' FCBTYPE
IERSIPWP EQU X'10' PASSWORD-PROTECTED
IERSISHR EQU X'08' SHARE
IERSICRD EQU X'04' CREATION-DATE
IERSIEXD EQU X'02' EXPIRATION-DATE
IERSILAD EQU X'01' LAST-ACCESS-DATE
*
IERSELI1 DC B'00000000' SELECT IND1
IERSIEXT EQU X'80' EXTENTS
IERSIFSZ EQU X'40' FREESIZE
IERSISIZ EQU X'20' SIZE
IERSISUP EQU X'10' SUPPORT
IERSIVOL EQU X'08' VOLUME
IERSIACL EQU X'04' ACL
IERSIACT EQU X'02' ACCESS-COUNT
IERSIBAC EQU X'01' BASIC-ACL
*
15:29:06 2001-10-15 PAGE
0041
SOURCE STATEMENT
IERSELI2 DC B'00000000' SELECT IND2
IERSIBKC EQU X'80' BLOCK-CONTROL
IERSIBCT EQU X'40' BLOCK-COUNT
IERSIDWR EQU X'20' DISK-WRITE
IERSIGUA EQU X'10' GUARDS
IERSIIOP EQU X'08' IO-PERFORMANCE
IERSILCD EQU X'04' LAST-CHANGE-DATE
IERSIIOU EQU X'02' IO-USAGE
IERSILPA EQU X'01' LAST-PAGE
*
IERSELI3 DC B'00000000' SELECT IND3
IERSIMIG EQU X'80' MIGRATE
IERSIPRO EQU X'40' PROTECTION-LEVEL
IERSIRSP EQU X'20' RELEASE-SPACE
IERSISTA EQU X'10' STATE
IERSISTL EQU X'08' STORAGE-LEVEL
IERSITYP EQU X'04' TYPE
IERSICCS EQU X'02' CCS
IERSI3UU EQU X'01' -- UNUSED, MUST BE 0 --
```

```

*
IERSELI4 DC      B'00000000'    SELECT IND4                200
IERSIVST EQU    X'80'          VOLUME-SET                 200
IERSIWKF EQU    X'40'          WORK-FILE                           200
IERSIAVA EQU    X'20'          AVAILABILITY                       200
IERSIS0M EQU    X'10'          S0-MIGRATION                       200
IERSIDL D EQU    X'08'          DELETION-DATE                     200
IERSIUIN EQU    X'04'          USER-INFO                          200
IERSIAIN EQU    X'02'          ADM-INFO                            200
IERSI4UU EQU    X'01'          -- UNUSED, MUST BE 0 --           200
*
IERSELI5 DC      B'00000000'    SELECT IND5                200
IERSISCL EQU    X'80'          STORAGE-CLASS                     200
IERSIMCL EQU    X'40'          MANAGEMENT-CLASS                   200
IERSIPFO EQU    X'20'          FILE-PREFORMAT                     220
IERSI5UU EQU    X'1F'          -- UNUSED, MUST BE 0 --           220
*
IERUNUS1 DC      X'0000'        200
*
IERACC  DC      A(0)           ACCESS-COUNT - FROM
IERACC2 DC      A(0)           - TO
*
IERBACL DC      B'00000000'    BASIC-ACL
IERBACL EQU    X'80'          YES
IERBACL EQU    X'40'          NONE
IERBACLU EQU   X'2F'          -- UNUSED, MUST BE 0 --
*
IERBOW  DC      B'00000000'    BASIC-ACL-OWNER
IERBOWS EQU    X'80'          READ-RIGHT-SPECIFIED
IERBOWS EQU    X'40'          WRITE-RIGHT-SPECIFIED
IERBOWS EQU    X'20'          EXEC-RIGHT-SPECIFIED
IERBOWRY EQU   X'10'          READ = YES
IERBOWRY EQU   X'08'          WRITE = YES
IERBOWRY EQU   X'04'          EXEC = YES
IERBOWNO EQU   X'02'          NO-ACCESS
IERBOWUU EQU   X'01'          -- UNUSED, MUST BE 0 --
*
IERBGR  DC      B'00000000'    BASIC-ACL-GROUP
IERBGRS EQU    X'80'          READ-RIGHT-SPECIFIED
IERBGRS EQU    X'40'          WRITE-RIGHT-SPECIFIED
IERBGRS EQU    X'20'          EXEC-RIGHT-SPECIFIED
IERBGRRY EQU   X'10'          READ = YES
IERBGRRY EQU   X'08'          WRITE = YES
IERBGRRY EQU   X'04'          EXEC = YES
IERBGRNO EQU   X'02'          NO-ACCESS
IERBGRUU EQU   X'01'          -- UNUSED, MUST BE 0 --
*
IERBOT  DC      B'00000000'    BASIC-ACL-OTHERS

```

```

IERBOTRS EQU X'80'          READ-RIGHT-SPECIFIED
IERBOTWS EQU X'40'          WRITE-RIGHT-SPECIFIED
IERBOTXS EQU X'20'          EXEC-RIGHT-SPECIFIED
IERBOTRY EQU X'10'          READ = YES
IERBOTWY EQU X'08'          WRITE = YES
IERBOTXY EQU X'04'          EXEC = YES
IERBOTNO EQU X'02'          NO-ACCESS
IERBOTUU EQU X'01'          -- UNUSED, MUST BE 0 --
*
IERBKL  DC  B'00000000'     BACKUP LEVEL FLAG
IERBKLA EQU X'80'  7-7 S    A
IERBKLB EQU X'40'  6-6 S    B
IERBKLC EQU X'20'  5-5 S    C
IERBKLD EQU X'10'  4-4 S    D
IERBKLE EQU X'08'  3-3 S    E
IERBKLN EQU X'07'  2-0 R    -- UNUSED, MUST BE 0 --
*
IERBLKCT DC  B'00000000'     BLKCTRL -
IERBCTNN EQU X'80'  7-7 S    NONE
IERBCTPK EQU X'40'  6-6 S    PAMKEY
IERBCTDA EQU X'20'  5-5 S    DATA2K
IERBCTNO EQU X'10'  4-4 S    NO
IERBCTD4 EQU X'08'  3-3 S    DATA4K
IERBCTN2 EQU X'04'  2-2 S    NK2
IERBCTN4 EQU X'02'  1-1 S    NK4
IERBCTNU EQU X'01'  0-0 R    -- UNUSED, MUST BE 0 --
*
IERFUT3  DC  X'0000'        SPACE FOR FUTURE USE, MUST BE 0
*
IERBLCT  DC  A(0)           BLOCK-COUNT - FROM
IERBLCT2 DC  A(0)           - TO
*
IERCRD   DC  CL10' '        CREATION DATE - FROM DATE
IERCRT   DC  CL8' '         TIME
IERCRD2  DC  CL10' '        - TO DATE
IERCRT2  DC  CL8' '         TIME
*
IEREXD   DC  CL10' '        EXPIRATION DATE - FROM DATE
IEREXT   DC  CL8' '         TIME
IEREXD2  DC  CL10' '        - TO DATE
IEREXT2  DC  CL8' '         TIME
*
IEREXS   DC  A(0)           EXTENTS - FROM
IEREXS2  DC  A(0)           - TO
*
IERFSIZE DC  A(0)           FSIZE - FROM
IERFSIZ2 DC  A(0)           - TO
*

```

```

IERDW   DC   B'00000000'   DISK WRITE -
IERDWIM EQU  X'80'   7-7 S   IMMEDIATE
IERDWBC EQU  X'40'   6-6 S   BY-CLOSE
IERDWNU EQU  X'3F'   5-0 R   -- UNUSED, MUST BE 0 --
*
IERFCB  DC   B'00000000'   FCB TYPE FLAG -
IERFCBP EQU  X'80'   7-7 S   PAM
IERFCBS EQU  X'40'   6-6 S   SAM
IERFCBI EQU  X'20'   5-5 S   ISAM
IERFCBB EQU  X'10'   4-4 S   BTAM
IERFCBN EQU  X'08'   3-3 S   NONE
IERFCBNU EQU X'07'   2-0 R   -- UNUSED, MUST BE 0 --
*
IERGUA  DC   B'00000000'   GUARDS SPECIFIED FLAG
IERGUARS EQU  X'80'   7-7 S   READ-SPECIFIED
IERGUAWS EQU  X'40'   6-6 S   WRITE-SPECIFIED
IERGUAXS EQU  X'20'   5-5 S   EXEC-SPECIFIED
IERGUUNU EQU  X'10'   4-4 S   UNUSED, MUST BE ZERO --           200
IERGUANS EQU  X'08'   3-3 S   GUARDS=NONE SPECIFIED
IERGUAYS EQU  X'04'   2-2 S   GUARDS=YES SPECIFIED
IERGUAUU EQU  X'03'   1-0 R   -- UNUSED, MUST BE ZERO --
IERGUAR  DC   CL18' '   GUARDS-READ
IERGUAW  DC   CL18' '   GUARDS-WRITE
IERGUAX  DC   CL18' '   GUARDS-EXEC
*
IERIP   DC   B'00000000'   IO-PERFORMANCE
IERIPVH EQU  X'80'   7-7 S   VERY-HIGH
IERIPHI EQU  X'40'   6-6 S   HIGH
IERIPST EQU  X'20'   5-5 R   STD
IERIPNU EQU  X'1F'   4-0 R   -- UNUSED, MUST BE 0 --
*
IERIU   DC   B'00000000'   IO-USAGE
IERIURW EQU  X'80'   7-7 S   READ-WRITE
IERIURD EQU  X'40'   6-6 S   READ
IERIUWR EQU  X'20'   5-5 R   WRITE
IERIUNU EQU  X'1F'   4-0 R   -- UNUSED, MUST BE 0 --
*
IERUNUS2 DC   X'00'
*
IERLAD  DC   CL10' '   LAST ACCESS DATE - FROM DATE
IERLAT  DC   CL8' '   TIME
IERLAD2 DC   CL10' '   - TO DATE
IERLAT2 DC   CL8' '   TIME
*
IERLAPA DC   A(0)   LAST-PAGE - FROM
IERLAPA2 DC  A(0)   - TO
*
IERLCD  DC   CL10' '   LAST CHANGE DATE - FROM DATE

```

System Exits

```

IERLCT  DC      CL8'  '          TIME
IERLCD2 DC      CL10' '          - TO DATE
IERLCT2 DC      CL8'  '          TIME
*
IERDLDD DC      CL10' '          DELETION DATE - FROM DATE          200
IERDLT  DC      CL8'  '          TIME          200
IERDLDD2 DC     CL10' '          - TO DATE          200
IERDLT2 DC      CL8'  '          TIME          200
*
IERMIGRT DC     B'00000000'      MIGRATE -
IERMGALL EQU    X'80'  7-7 S      ALLOWED
IERMGINH EQU    X'20'  5-5 S      INHIBIT
IERMGFOB EQU    X'10'  4-4 S      FORBIDDEN          200
IERMGANY EQU    X'B0'  7,4 R      ANY          200
IERMIGNU EQU    X'4F'  3-0 R      -- UNUSED, MUST BE 0 --          200
*
IERPWP  DC      B'00000000'      PW PROTECT FLAG -
IERPWPRD EQU    X'80'  7-7 S      READ
IERPWPR EQU    X'40'  6-6 S      WRITE
IERPWPEX EQU    X'20'  5-5 S      EXEC
IERPWPNO EQU    X'10'  4-4 S      NONE
IERPWPNU EQU    X'0F'  3-0 R      -- UNUSED, MUST BE 0 --
*
IERPROL DC      B'00000000'      PROTECTION-LEVEL-FLAG
IERPROL0 EQU    X'80'  7-7 S      LEVEL 0
IERPROL1 EQU    X'40'  6-6 S      LEVEL 1
IERPROL2 EQU    X'20'  5-5 S      LEVEL 2
IERPROUU EQU    X'1F'  4-0 R      -- UNUSED, MUST BE 0 --
*
IERRLSP DC      B'00000000'      RELEASE-SPACE-FLAG
IERRLSPA EQU    X'80'  7-7 S      ALLOWED
IERRLSPI EQU    X'40'  6-6 S      IGNORED
IERRLSPU EQU    X'3F'  5-0 R      -- UNUSED, MUST BE 0 --          200
*
IERSELF0 DC     B'00000000'      SELECT FLAG 0 -
IERTYPEF EQU    X'80'  7-7 S      TYPE = FILE
IERTYPEG EQU    X'40'  6-6 S      TYPE = FGG
IERTYPEA EQU    X'C0'  7-6 S      TYPE = ANY
IERTYPEP EQU    X'20'  5-5 S      TYPE = PLAM
IERSTCNS EQU    X'10'  4-4 S      STATE/CACHE NOT SAV
IERAVAST EQU    X'08'  3-3 S      AVAILABILITY = STD          200
IERAVAH1 EQU    X'04'  2-2 S      AVAILABILITY = HIGH          200
IERSF0NU EQU    X'03'  1-0 R      -- UNUSED, MUST BE 0 --          200
*
IERSELF1 DC     B'00000000'      SELECT FLAG 1 -
IERACCRD EQU    X'80'  7-7 S      ACCESS = READ
IERACCWR EQU    X'40'  6-6 S      ACCESS = WRITE
IERSHARY EQU    X'20'  5-5 S      SHARE = YES

```

```

IERSHARN EQU X'10' 4-4 S SHARE = NO
IERSHARS EQU X'08' 3-3 S SHARE = SPECIAL
IERACLY EQU X'04' 2-2 S ACL = YES
IERACLN EQU X'02' 1-1 S ACL = NO
IERSF1NU EQU X'01' 0-0 R -- UNUSED, MUST BE 0 --
*
IERSELF2 DC B'00000000' SELECT FLAG 2 -
IERSEQFS EQU X'80' 7-7 S SIZE = FSIZE
IERFSEQS EQU X'40' 6-6 S FSIZE = SIZE
IERCRDNO EQU X'20' 5-5 S CRDATE = NONE
IEREXDNO EQU X'10' 4-4 S EXDATE = NONE
IERLADNO EQU X'08' 3-3 S LADATE = NONE
IERLCDNO EQU X'04' 2-2 S LCDATE = NONE
IERDLNO EQU X'02' 1-1 S DELDATE = NONE 200
IERSF2UU EQU X'01' 0-0 R -- UNUSED, MUST BE 0 -- 200
*
IERSTA DC B'00000000' STATE FLAG -
IERSTNOC EQU X'80' 7-7 S NOCLOS
IERSTCLO EQU X'40' 6-6 S CLOSED
IERSTREP EQU X'20' 5-5 S REPAIR NEEDED
IERSTCAC EQU X'10' 4-4 S CACHED
IERSTNCA EQU X'08' 3-3 S NOT CACHED
IERSTNOA EQU X'04' 2-2 S NO OPEN ALLOWED
IERSTOPA EQU X'02' 1-1 S OPEN ALLOWED
IERSTDFR EQU X'01' 0-0 S DEFECT-REPORTED
*
IERSTLEV DC B'00000000' SLEVEL -
IERSLVL0 EQU X'80' 7-7 S S0
IERSLVL1 EQU X'40' 6-6 S S1
IERSLVL2 EQU X'20' 5-5 S S2
IERSLANY EQU X'E0' 7-5 R ANY
IERSLVNU EQU X'1F' 4-0 R -- UNUSED, MUST BE 0 --
*
IERSUP DC B'00000000' SUPPORT TYPE FLAG -
IERSUPPU EQU X'80' 7-7 S PUBLIC
IERSUPPR EQU X'40' 6-6 S PRDISC
IERSUPTP EQU X'20' 5-5 S TAPE
IERSUPNU EQU X'1F' 4-0 R -- UNUSED, MUST BE 0 --
*
IERSMS DC B'00000000' SMS FLAG - 200
IERSOMAL EQU X'80' 7-7 S S0-MIGRATION = ALLOWED 200
IERSOMFO EQU X'40' 6-6 S S0-MIGRATION = FORBIDDEN 200
IERWKFYE EQU X'20' 5-5 S WORK-FILE = YES 200
IERVCTRL EQU X'10' 4-4 S VOL-SET = CONTROL 210
IERSMSNU EQU X'0F' 3-0 R -- UNUSED, MUST BE 0 -- 210
*
IERUNUS3 DC X'00' 200
*

```

```

IERSIZE  DC    A(0)          SIZE  - FROM
IERSIZE2 DC    A(0)          - TO
*
IERVOL   DC    CL6' '       VOLUME
IERCCS   DC    CL8' '       CODED-CHARACTER-SET
IERVOLST DC    CL4' '       VOLUME-SET (SMS)                200
IERUSINF DC    CL8' '       USER-INFORMATION                200
IERADINF DC    CL8' '       ADM-INFORMATION                  200
IERMGMCL DC    CL8' '       MANAGEMENT-CLASS                200
IERSTOCL DC    CL8' '       STORAGE-CLASS                   200
IERPFORM DC    B'00000000'   FILE-PREFORMAT           220
IERPFONO EQU    X'80' 7-7 S   NONE                          220
IERPFOK  EQU    X'40' 6-6 S   K                              220
IERPFON2 EQU    X'20' 5-5 S   NK2                            220
IERPFON4 EQU    X'10' 4-4 S   NK4                            220
IERPFOAN EQU    X'F0' 7-4 R   ANY                            220
IERPFOUN EQU    X'0F' 3-0 R   -- UNUSED, MUST BE 0 --       220
IERFUT4  DC    61X'00'      SPACE FOR FUTURE USE, MUST BE 0
IERPLLN  EQU    *-IERERAST   LENGTH
*
```

6.1.4 IDFCB

```

IDFCB D
*
***** FCB (P1 REGION) *****
*
#INTF REFTYPE=REQUEST, C
      INTNAME=IDFCB,INTCOMP=1
MFPRE MF=D,PREFIX=,MACID=D1,DNAME=D1FCB,ALIGN=F 200
ID1FCB DSECT ,
      *,##### PREFIX=I, MACID=D1 #####
*
ID1P1FST EQU *
ID1CONAR DS 0F
ID1IOAR1 DS F ADDR OF IOAREA1 ISQP
ID1IOA1N EQU X'80' S IOAREA1=NO ISQP
ID1IOA1B EQU X'40' S IOAREA1= ISQP
ID1IOAR2 DS F ADDR OF IOAREA2 ISQP
ID1IOA2N EQU X'80' S IOAREA2=NO ISQP
ID1IOA2B EQU X'40' S IOAREA2= ISQP
ID1EXIT DS F ADDR OF EXIT LIST ISQP
ID1NOXIT EQU X'80' S EXIT= ISQP
ID1MSTXT EQU X'40' S MASTER EXIT LIST ISQP
ID1CLBT EQU X'20' S EXIT MODIFY BIT ISQP
ID1KEYAR DS F ADDR OF KEY S
ID1KEYAN EQU X'80' S KEYARG= S
*
ID1PASS DS XL4 PASSWORD ISQPM
ID1RETPD DS H RETENTION PERIOD ISQPM
ID1RECSI DS H RECORD SIZE IS M
ID1BLKSI DS H BLOCK SIZE IS PM
ID1BLSTD EQU X'80' S BLKSIZE=STD IS M
ID1KEYPO DS H KEY POSITION S M
ID1KEYLE DS HL1 KEY LENGTH S M
ID1INDEX DS HL1 INDEX S M
ID1PAD DS HL1 I.S. PADDING S M
*
ID1OPEN DS X OPEN TYPE : ISQPM
ID1SINOT EQU X'80' S SINOUT P
ID1OUTIN EQU X'40' S OUTIN IS PM
ID1INOUT EQU X'20' S INOUT IS PM
ID1UPDAT EQU X'10' S UPDATE I Q M
ID1EXTEN EQU X'08' S EXTEND IS PM
ID1OUTPU EQU X'04' S OUTPUT ISQPM
ID1REVER EQU X'02' S REVERSE I M
ID1INPUT EQU X'01' S INPUT ISQPM
*

```

```

ID1IND0 DS X INDICATOR 0 :
ID1IIND EQU ID1IND0 ISAM INDICATOR
ID1VLPNL EQU X'08' S VALPROP=NULL
ID1LOGNL EQU X'04' S LOGLEN=NULL
ID1VALNL EQU X'02' S VALLEN=NULL
ID1WROUT EQU X'01' S WROUT=YES

15:29:06 2001-10-15 PAGE

0047
SOURCE STATEMENT
ID1FRRC EQU X'80' S RECON=YES
ID1FRRS EQU X'40' S RESET=YES
ID1IPP EQU X'20' S VALPROP/VALLEN/LOGLEN SPEC.
*
ID1IND1 DS X INDICATOR 1 : ISQPM
ID1TRANS EQU X'80' S TRANSLATE TAPE
ID1LABEL EQU X'60' FILE LABEL OPTIONS ISQPM
ID1STD EQU X'60' R STD LABELS ISQPM
ID1NSTD EQU X'40' S NSTD LABELS I PM
ID1LABEL EQU X'20' S LABEL=NO
ID1AMN EQU X'18' PRINTER CONTROL ISQPM
ID1A EQU X'18' S ASA CONTROL CHAR ISQPM
ID1M EQU X'08' S EBCDIC CONTROL ISQPM
ID1N EQU X'18' R NO CONTROL CHAR ISQPM
ID1RECFO EQU X'06' RECORD FORM ISQPM
ID1F EQU X'04' S FIXED LENGTH ISQPM
ID1V EQU X'02' S VARIABLE LENGTH IS M
ID1U EQU X'06' S UNDEFINED IS PM
ID1TPMRK EQU X'01' S TPMARK=YES
*
ID1IND2 DS X INDICATOR 2 : (SET,RESET)
ID1FCBTY EQU X'C0' FCB TYPE ISQPM
ID1SAM EQU X'C0' R SAM I M
ID1ISAM EQU X'40' S ISAM S M
ID1PAM EQU X'C0' S PAM Q M
ID1BTAM EQU X'80' S BTAM PM
ID1DUPKE EQU X'20' S DUP KEY ALLOWED S M
ID1OVERL EQU X'10' S OVERLAP=YES S M
ID1INDPR EQU X'08' S INDEX=(ABSEXP) S M
ID1SHB EQU X'04' S SPECIAL HANDLING BIT Q
ID1FORM EQU X'02' S FORM=SHORT QP
ID1VMIN EQU X'01' R MINIMUM FUNCT.FOR ISAM VALUE FLAG
ID1VMAX EQU X'01' S MAXIMUM FUNCT.FOR ISAM VALUE FLAG
*
ID1IND3 DS X INDICATOR 3 :
ID1NULLP EQU X'FF' NULL FILE CHARACTERISTIC PARAMETERS
ID1RECSN EQU X'80' S RECSIZE=,
ID1BLKSN EQU X'40' S BLKSIZE=,
ID1KEYPN EQU X'20' S KEYPOS=,

```

```

ID1KEYLN EQU X'10'      S KEYLEN=,
ID1RECFN EQU X'08'      S RECFORM=,
ID1FCBTN EQU X'04'      S FCBTYPE=,
ID1CODEN EQU X'02'      S CODE=,
ID1BUFNL EQU X'01'      S BUFOFF3=,
*
ID1IOREG DS HL1          IOREG                                IS
ID1VARBL DS HL1          VARBLD                              IS
ID1LINK DS CL8           FCB/LINK NAME                       ISQPM
ID1FILE DS CL54          FILE NAME                           ISQPM
ID1VARAR DS 0F
ID1P2LNK DS F            LINK TO P2 REGION                   ISQP
ID1UFCBS EQU *-ID1CONAR  SIZE OF CONSTANT REGION
*
ID1COMMN DS 11F          COMMON REGION                        ISQP
ID1CMEND EQU *           END COMMON REGION -----
*
          ORG ID1COMMN   *** S A M ***                        I
ID1LHECN DS F            LHECON CURRENT                      I
ID1AFNAL DS F            LHECON MAXIMUM                      I
ID1BLKCT DS PL4          BLOCK COUNT                          I
ID1RPTR DS F            LOG BUFF/RCD PTR                     I
ID1RGSV DS 5F           SAVE AREA                            I
ID1TRADR DS A            A(USER TRANS TAB) READ MODE        I P
ID1TRADW DS A            A(USER TRANS TAB) WRITE MODE        I P
*
          ORG ID1COMMN   ACCESS METHOD ISAM
ID1INHKB@ DS A           @ HIGHKEY
ID1INNR@ DS A           @ NEXT RECORD
ID1INCR@ DS A           @ CURRENT RECORD
ID1INFL0 DS X           FLAGB. DISUSNT: WIE ISAM-ALT
ID1INSTF EQU X'01'      D1SHAR1 ! WIE ISAM-ALT !
ID1INLTF EQU X'04'      D1DLOK1 ! WIE ISAM-ALT !
ID1INLSP EQU X'10'      D1DLOK ! WIE ISAM-ALT !
ID1INFL1 DS X
ID1INHK EQU X'01'       @ HIGHKEY VALID
ID1INFL2 DS X           UNBENUTZT
ID1INSTA DS X           P1LOGICAL-STATUS
ID1INNS EQU X'01'      NOT_SEQUENTIAL
ID1INSR EQU X'02'      SEQUENTIAL_READ (GET/GETR MIT IOAREA)
ID1INGT EQU X'04'      SEQUENTIAL_READ_GET
ID1INGR EQU X'08'      SEQUENTIAL_READ_GETR
ID1INSL EQU X'10'      SEQUENTIAL_LOAD (PUT MIT IOAREA)
ID1INPC EQU X'20'      PUTX_LOCATE_TO_BE_CLOSED
ID1INSPB EQU X'40'      SEQ_READ_AFTER_PTXLCB
ID1INPCB EQU X'80'      PUTX_LOC_TO_BE_CLOSED_BLK
ID1INMRL DS CL4        MINIMALE RECORD-LAENGE
ID1INPOL DS H          SUMME UEBER POINTERLAENGEN BEI PUT

```

```

ID1INVLN DS      X      ENTSPRICHT  D1VALLN ! WIE ISAM-ALT !
ID1INLLN DS      X      ENTSPRICHT  D1LOGLN ! WIE ISAM-ALT !
ID1INTBS DS      A      BLKSIZE OHNE TRAILER (N*2032 - 12)
ID1INPBS DS      A      BLKSIZE OHNE PAD-SIZE
ID1INBS  DS      A      BLKSIZE = BLOCKFAKTOR * 2048
ID1PRECS DS      H      RECSIZE DES VORGAENGER PUT-LOCATE    951
*
      ORG  ID1COMMN      ***  I S A M  ***                S
ID1GRPTR DS      A(0)    'GETR' ADDRESS                S
ID1GPTR  DS      A(0)    'GET' ADDRESS                 S
ID1GETSW DS      0F      GET | GETR SWITCHES           S
ID1SETL  EQU     X'80'    S INDICATE SETL                S
ID1GET   EQU     X'40'    S INHIBIT GET EOB              S
ID1GETR  EQU     X'20'    S INHIBIT GETR EOB             S
ID1GETER EQU     X'10'    S ERROR DURING GET          S
ID1NULLF EQU     X'08'    S NULL FILE                    S
ID1FIRST DS      A(0)    'FIRST' RECORD POINTER        S
*
ID1SUSNT DS      X      SHARED UPDATE SENTINEL :
ID1SHAR1 EQU     X'01'    S SHARUPD=YES FOR THIS FILE
ID1SHAR2 EQU     X'02'    S SHARUPD=YES FOR SOME FILE
ID1DLOK1 EQU     X'04'    S DATA PG LOCKED THIS FILE
ID1DLOK2 EQU     X'08'    S DATA PG LOCKED SOME FILE
ID1DLOK  EQU     X'10'    S LOCK SPECIFIED IN MACRO
*
      R NOLOCK SPECIFIED IN THE MACRO
ID1PGLOK EQU     X'20'    S GET,GETR,GETFL ARE ILLEGAL,
*
      SET WHEN USER TAKES PGLOCK EXIT
ID1ISLK  EQU     X'40'    S SAM LOCK EXISTS
ID1SWEAK EQU     X'80'    S SHARUPD=WEAK FOR THIS FILE
*
      ORG  ID1SUSNT
ID1CURBF DS      A(0)    CURRENT BUFFER ADDRESS                S
ID1PUTSW DS      0F      PUT SWITCH                        S
ID1PUT   EQU     X'80'    S EXTENDED 'PUT'                S
ID1FPUT  EQU     X'40'    S PUT TO EMPTY BUFFER           S
ID1NPUT  EQU     X'20'    S PUT TO EMPTY PAGE              S
ID1SPUT  EQU     X'10'    S SEQ CHK NOT ALLOWED            S
ID1PPTR  DS      A(0)    'PUT' POINTER                    S
*
ID1SPREM EQU     ID1PPTR
ID1SWORK DS      H      ALIGNMENT CONVERSION              S
ID1VALLN DS      X      ISAM VALUE FLAG LENGTH
ID1LOGLN DS      X      ISAM LOGICAL FLAG LENGTH
ID1REGSV DS      4F      REGISTER SAVE AREA                S
ID1CMPD  DS      1F      LOCATE MODE COMPARAND
*
      ORG  ID1COMMN      ***  P A M  ***
ID1LWB   DS      F      LAST WAITED BUFFER                Q

```

ID1NOWT	EQU	X'FF'	NO BUFFER WAITED	Q
	ORG	ID1COMMN	RFA PAM FIELD	
ID1PLPK	DS	H	# OF LOCKED PAM PAGES	
ID1PUPK	DS	H	# OF UNLOCKED PAM PAGES	
ID1KEY1	DS	CL16	KEY	Q
ID1KEY2	DS	CL16	KEY	Q
ID1REQ	DS	0H	# PAM REQUESTS	Q
ID1NBPP	DS	CL1	# PROCESSED HP'S	Q
ID1PRQ	DS	CL1	PAMREQS	Q
ID1TOUT	DS	AL2	PAM TIME OUT	Q
ID1CHERR	DS	A(0)	CHAIN ELEMENT IN ERR	Q
*				
	ORG	ID1COMMN	*** B T A M ***	P
*			BTAM ALSO USES FIELDS &P.TRADR AND &P.TRADW	
*			OF THE SAM WORK AREA	
	ORG	ID1CMEND		
*				
*****		COMMON TO ALL ACCESS METHODS	*****	
*				
ID1RTNAD	DS	0F	RETURN ADDR	ISQP
ID1LGINF	DS	A(0)	ADDRESS OF LOGINFO	
ID1ECB	DS	XL2	ERROR CODE	ISQP
*				
ID1XITB	DS	X	ERROR EXIT BYTE :	ISQP
ID1OPENX	EQU	4	OPEN EXIT	ISQP
ID1OPENE	EQU	8	OPEN ERROR	ISQP
ID1PASSE	EQU	12	PASSWORD ERROR	ISQP
ID1LOCK	EQU	16	LOCKED FILE	ISQP
ID1NODEV	EQU	20	NO DEVICE	ISQP
ID1OPENZ	EQU	24	OPEN EXIT AFTER LABEL	ISQP
ID1OPENV	EQU	28	NON-STANDARD TAPE LABELS	
ID1RECNX	EQU	32	RECON EXIT	
ID1LABGN	EQU	36	GEN/RD HDR LAB VOL1	I P
ID1LABEO	EQU	40	GEN/RD EOF LAB	I P
ID1CLOSE	EQU	44	CLOSE ERROR	
ID1LABEN	EQU	48	GEN/RD EOF LAB	I P
ID1EOVCT	EQU	52	VOLUME SWAPPING DONE	I P
ID1PLKXT	EQU	56	LOCKED PAGE(S) EXIT	SQ
ID1DEDLK	EQU	60	DEADLOCK EXIT	SQ
ID1EOFAD	EQU	64	EOF ADDR	IS P
ID1ERRAD	EQU	68	ERROR ADDR	ISQP
ID1ERROP	EQU	72	READ ERROR OPTIONS	I
ID1NOSPA	EQU	76	FOR SECONDARY ALLOC	
ID1ISPER	EQU	80	INSUFFICIENT SPACE	S
ID1DUPEK	EQU	84	DUPLICATE KEY	S
ID1NOFIN	EQU	88	RECORD NOT FOUND	S
ID1USERE	EQU	92	USER ERROR	ISQP
ID1SEQCH	EQU	96	SEQUENCE ERROR	S

ID1WLRER	EQU	100	WRONG LENGTH RECORD	
ID1OPENC	EQU	104	OPENC EXIT	
ID1LABER	EQU	108	LABERR EXIT	
ID1CLOSP	EQU	112	CLOSPOS EXIT	
*				
ID1STAT	DS	0CL5	STATUS BYTES	QP
ID1SDB	DS	C	STANDARD DEVICE BYTE	QP
ID1SB1	DS	C	SENSE BYTE 1	QP
ID1SB2	DS	C	SENSE BYTE 2	QP
ID1SB3	DS	C	SENSE BYTE 3	QP
ID1EFB	DS	C	EXECUTIVE FLAG BYTE	QP
*				
ID1SNT1	DS	X	SENTINEL1 : (SET,RESET)	ISQP
ID1REMOT	EQU	X'80'	S FILE ACCESS IN RFA MODE	
ID1BLOGS	EQU	X'40'	S SAM BUFFER LOGGED	
ID1NEWV	EQU	X'20'	S EXTENSION FLAG	
ID1CHECK	EQU	X'01'	S NORMAL CHECK COMPLETION	
ID1SYCLS	EQU	X'02'	S SYSTEM FILE	
*				
ID1CODE	EQU	ID1SNT1	CODE OF TAPE :	
ID1ISO7	EQU	X'10'	S ISO7 CODE	
ID1EBCD	EQU	X'08'	S EBCDI CODE	
ID1OWN	EQU	X'04'	S OWN CODE	
ID1NDEF	EQU	X'1C'	R CODE NOT DEFINED	
*				
ID1SNT2	DS	X	SENTINEL2 : (SET,RESET)	
ID1UPACT	EQU	X'08'	S UPDATED BLOCK	I
ID1RAW	EQU	X'04'	S READ AFTER WRITE	
ID1OPN	EQU	X'40'	S FILE OPEN	ISQP
ID1GSET	EQU	X'20'	S LAST ACTION (GET)	S
*				
S SETL=V B				
*				
R SETL=E				
*				
S ELIM				
ID1GRSET	EQU	X'18'	S LAST ACTION (GETR)	S
ID1GRST1	EQU	X'10'	S SETL=V(V FOUND)	S
*				
R SETL=V(V N/FOUND)				
*				
R SETL=B				
*				
R ELIM				
ID1LPUTX	EQU	X'08'	S LOCATE PUTX	S
ID1EXCAT	EQU	X'02'	S EXCAT IN PROCESS	
*				
ID1DFCBS	EQU	*-ID1VARAR	SIZE OF VARIABLE REGION	
ID1VIND	DS	H	EXTENTION LENGTH	
	DS	0F		
ID1FCBS	EQU	*-ID1CONAR	SIZE OF FCB WITHOUT LOGICALS	
ID1LOGIC	DS	128F	LOGICAL ROUTINES	IS
	ORG	ID1LOGIC+X'48'		950
ID1SAVE	DS	16F	SAVE AREA FOR P1-LOGICAL USE	950
	ORG			
ID1LFCBS	EQU	*-ID1LOGIC	SIZE OF LOGICAL PROCEDURE AREA	
ID1FCBSZ	EQU	*-ID1P1FST	SIZE OF P1 FCB	

6.1.5 DCACATE

```

DCACATE MF=D
MFCHK SUPPORT=(C,D),PREFIX=D,MACID=CAC,MF=D,          C
          DMACID=CAC,DNAME=CACATF
DCACATF  DSECT  ,
          * ,##### PREFIX=D, MACID=CAC #####
*
*-----*
  DECLARE FIXED PART OF CATALOG ENTRY
*-----*
*
DCACFFP    DSECT
DCACFCEL   DS      XL2          LENGTH OF CATALOG ENTRY
*
DCACFCE#   EQU     4079         maximal length of catalog entry
DCACFEVE   DS      0XL2        VERSION OF CATALOG ENTRY LAYOUT
DCACFEV1   EQU     X'FF01',2
DCACFEV2   EQU     X'FF02',2   BS2000/OSD-BC V3
DCACFEV3   EQU     X'FF03',2   BS2000/OSD-BC V5          319
*
DCACFEFM   DS      X           MARKER OF CE FORMAT
DCACFENW   EQU     X'FF'       "NEW" FORMAT INTRODUCED IN V1
*
DCACFVER   DS      X           CE VERSION
DCACFVRC   EQU     X'03'       current CE version          319
DCACFEF1   EQU     X'01'       319
DCACFEF2   EQU     X'02'       319
DCACFEF3   EQU     X'03'       319
*
DCACFOMP   DS      XL2         OFFSET MAIN PART
*
                                JV ENTRY: MUST BE ZERO
DCACFOOP   DS      XL2         OFFSET OLD PART          IDJE/IDCE
DCACFOFJ   DS      XL2         OFFSET FILE-/JV-NAME
DCACFOEX   DS      XL2         OFFSET CE EXTENSION        IDCEX
*
                                JV ENTRY: MUST BE ZERO
DCACFOVE   DS      0XL2        OFFSET VOLUME EXTENT LIST IDVT, IDEE
DCACFOFI   DS      XL2         OR FGG-INDEX IDCEG
*
                                JV ENTRY: MUST BE ZERO
DCACFOCP   DS      XL2         OFFSET COMMON PART
DCACFODP   DS      XL2         OFFSET DATE PART
DCACFOUN   DS      XL2         OFFSET UNUSED
*
                                JV ENTRY: MUST BE ZERO
DCACFOHP   DS      XL2         OFFSET HSMS PART
*
                                JV ENTRY: MUST BE ZERO
DCACFOPP   DS      XL2         OFFSET PROFILE PART
*

```

```

DCACFOF#   EQU   (*-DCACFOMP)/L'DCACFOMP      number of offsets
           DS   (((*-DCACFFP+3)/4)*4+DCACFFP-*)XL1      310
DCACFFP#   EQU   *-DCACFFP
* -----
* HERE STARTS THE NEW VARIABLE PART OF THE CATALOG ENTRY
* (SEE DESCRIPTION FOR COMMON STRUCTURE, DATE STRUCTURE ETC.)
*
* THE OLD CE LAYOUT STARTS AFTER VARIABLE PART
* (SEE OLD IDCE, IDCEX ETC.)
* -----
*
* -----*
  DECLARE DATE STRUCTURE FOR FILE-/JV-ENTRY (UTC-Format)
* -----
*
DCACFD P    DSECT
DCACFFPL   DS   XL2           LENGTH OF DATE STRUCTURE FILE
DCACFJPL   DS   XL2           LENGTH OF DATE STRUCTURE JV
DCACFCRD   DS   0CL14        CREATION DATE PART
DCACFCRY   DS   CL4           YEAR
DCACFCRM   DS   CL2           MONTH
DCACFCRT   DS   CL2           DAY
DCACFCRH   DS   CL2           HOURS
DCACFCRI   DS   CL2           MINUTES
DCACFCRS   DS   CL2           SECONDS
*           DATE/TIME UNDEFINED: C'
DCACFEXD   DS   0CL14        EXPIRATION DATE PART
DCACFEXY   DS   CL4           YEAR
DCACFEXM   DS   CL2           MONTH
DCACFEXT   DS   CL2           DAY
DCACFEXH   DS   CL2           HOURS
DCACFEXI   DS   CL2           MINUTES
DCACFEXS   DS   CL2           SECONDS
*           DATE/TIME UNDEFINED: C'
DCACFLAD   DS   0CL14        LAST ACCESS DATE
DCACFLAY   DS   CL4           YEAR
DCACFLAM   DS   CL2           MONTH
DCACFLAT   DS   CL2           DAY
DCACFLAH   DS   CL2           HOURS
DCACFLAI   DS   CL2           MINUTES
DCACFLAS   DS   CL2           SECONDS
*           DATE/TIME UNDEFINED: C'
DCACFLMD   DS   0CL14        LAST DATE MODIFIED
DCACFLMY   DS   CL4           YEAR
DCACFLMM   DS   CL2           MONTH
DCACFLMT   DS   CL2           DAY
DCACFLMH   DS   CL2           HOURS
DCACFLMI   DS   CL2           MINUTES

```

```

DCACFLMS    DS    CL2                SECONDS
*
*          DATE/TIME UNDEFINDED: C'      '
DCACFDED    DS    0CL14             DELETION DATE
DCACFDEY    DS    CL4                YEAR
DCACFDEM    DS    CL2                MONTH
DCACFDET    DS    CL2                DAY
DCACFDEH    DS    CL2                HOURS
DCACFDEI    DS    CL2                MINUTES
DCACFDES    DS    CL2                SECONDS
*
*          DATE/TIME UNDEFINDED: C'      '
          DS    XL2                RESERVED; MUST BE ZERO          310
          DS    (( (*-DCACFDP+3)/4)*4+DCACFDP-*)XL1          310
DCACFDP#    EQU    *-DCACFDP
*
*-----*
*  DECLARE COMMON STRUCTURE FOR FILE/JV-ENTRY
*-----*

DCACFCP     DSECT
DCACFCPL    DS    XL2                LENGTH OF COMMON PART
DCACFEHC    DS    CL8                EXTENDED HOST CODE
*          DC    C'                  '                UNDEFINED
*          DC    C'DF04              '                EBCDIC.DF.04
DCACFCTS    DS    XL2                CHARACTER TYPE SET
DCACFCUD    EQU    X'0000',2         UNDEFINED CHAR SET
DCACFEIN    DS    X                  INDICATOR
DCACFECC    EQU    X'80'             CONVERTED CE
DCACFEUN    EQU    X'7F'             UNUSED
DCACFCWT    DS    XL4                TASKID OF LAST WRITE
DCACFCWS    DS    XL1                SYSID OF LAST WRITE
DCACFWCC    DS    XL1                WRITE CHECK COUNTER
* The following fields are present only if CE_IDENTIFICATION.VERSION
* in the fixed part of the CE is at least 2.
DCACFCO0    DS    XL5                RESERVED: MUST BE ZERO
DCACFMGC    DS    CL8                SMS MANAGEMENT CLASS
          DS    (( (*-DCACFCP+3)/4)*4+DCACFCP-*)XL1          310
DCACFCP#    EQU    *-DCACFCP
*
          EJECT ,
*
*-----*
*  DECLARE NEW MAIN STRUCTURE
*-----*
*
DCACFMP     DSECT
DCACFMPL    DS    XL2                LENGTH OF MAIN STRUCTURE
DCACFIND    DS    X                  MAIN INDICATOR
DCACFPDF    EQU    X'80'             PD FILE

```

DCACFSNR	EQU	X'40'	SPACE NOT RELEASABLE	
DCACFGUF	EQU	X'20'	GUARDS FILE	
DCACFARW	EQU	X'10'	ARCHIVE WORK FILE	
DCACFEX4	EQU	X'08'	EXTENT-LIST WITH 4 BYTE	319
DCACFS0W	EQU	X'04'	WORK FILE FOR S0 MIGRATION	
DCACFCMF	EQU	X'02'	CMS FILE	
DCACFSRB	EQU	X'01'	SYSSRPM.BACKUP	
DCACFACC	DS	XL4	ACCESS COUNTER	
DCACFHCF	DS	0XL4	HSMS SAVED CODED FILE ID	
DCACFTAP	DS	0X	TAPE CRITERION	
DCACFBA#	DS	XL4	# BLOCKS OF TAPE FILE	
DCACFBL#	DS	XL4	# BLOCKS RECORDED ON LAST TAPE	
*			OF MULTIVOLUME TAPE FILE	
DCACFPER	DS	X	PERFORMANCE ATTRIBUTE	
DCACFPVH	EQU	X'03'	VERY HIGH	
DCACFPHI	EQU	X'02'	HIGH	
DCACFPST	EQU	X'01'	STANDARD (DEFAULT VALUE)	
DCACFUSE	DS	X	USE ATTRIBUTE	
DCACFURW	EQU	X'03'	READ-WRITE (DEFAULT VALUE)	
DCACFUWR	EQU	X'02'	WRITE	
DCACFURE	EQU	X'01'	READ	
DCACFTPE	DS	X	PERFORMANCE ATTRIBUTE OPEN/CLOSE	
DCACFTVH	EQU	X'03'	VERY HIGH	
DCACFTHI	EQU	X'02'	HIGH	
DCACFTST	EQU	X'01'	STANDARD (DEFAULT VALUE)	
DCACFTUS	DS	X	USE ATTRIBUTE OPEN/CLOSE	
DCACFTRW	EQU	X'03'	READ-WRITE (DEFAULT VALUE)	
DCACFTWR	EQU	X'02'	WRITE	
DCACFTRE	EQU	X'01'	READ	
DCACFTYA	DS	X	DISK WRITE (FILE TYPE)	
DCACFTCR	EQU	X'02'	BY CLOSE (NOT SENSITIVE)	
DCACFTNC	EQU	X'01'	IMMEDIATE (SENSITIVE, DEFAULT)	
DCACFCAI	DS	X	CACHE INDICATOR	
DCACFCAU	EQU	X'80'	S: CACHE USED	
*			R: CACHE NOT AVAILBALE	
DCACFDIG	EQU	X'40'	S: DATA IN GLOBAL STAORAGE	
*			R: NO DATA IN GLOBAL STORAGE	
DCACFDNO	EQU	X'20'	S: DENY OPEN REQUEST	
*			R: OPEN POSSIBLE	
DCACFDIC	EQU	X'10'	S: DATA STILL IN CACHE	
*			R: DATA NO LONGER IN CACHE	
DCACFWPF	EQU	X'08'	S: WRITE PERFORMED	
*			R: WRITE NOT PERFORMED	
DCACFFTS	DS	XL8	FILE TRANSFER INFORMATION	
DCACFSIN	DS	X	SMS INDICATOR	
DCACFAVA	EQU	X'C0'	AVAILABILITY:	
DCACFAVS	EQU	X'00'	STANDARD	
DCACFAVH	EQU	X'40'	HIGH	

```

DCACFMIF EQU X'20' S: S0 MIGRATION FORBIDDEN
* R: S0 MIGRATION ALLOWED
DCACFSIR EQU X'1F' RESERVED: MUST BE ZERO
DCACFSAT DS X SMS FILE ATTRIBUTES
DCACFWRK EQU X'08' S: FILE IS A USER WORKFILE
DCACFFMT EQU X'07' FILE PRE-FORMAT:
DCACFFMU EQU X'00' UNDEFINED
DCACFFMK EQU X'01' K
DCACFFM2 EQU X'02' NK2
DCACFFM4 EQU X'03' NK4
DCACFIN1 DS X INDICATOR1
DCACFNER EQU X'80' DO NOT ERASE
DCACFUSF EQU X'40' USER-SPECIFIC SYSTEM FILE
DCACFD5R EQU X'20' DELETE DURING F5 RECONSTRUCTION
DCACFCSF EQU X'10' DELETE SPACEOPT H-FILES 318
DCACFNDA EQU X'08' NO DMS ACCESS
DCACFIDF EQU X'04' INCR/DECR FILE-VERSION-NR 319
DCACFI1R EQU X'03' RESERVED
*
* The following fields are present only if CE_IDENTIFICATION.VERSION
* in the fixed part of the CE is at least 2.
DCACFUTU DS CL8 USER INFORMATION FOR STATE TU
DCACFUTP DS CL8 USER INFORMATION FOR STATE TPR
DCACFSTC DS CL8 STORAGE CLASS
DCACFTAG DS X FILE TAG (IDENTIFIES SPECIFIC FILE)
DCACFTNO EQU 0 NONE
DCACFGCF EQU 1 GENERIC CATALOG
DCACFSIC DS X SYSID OF WRITE CACHER 317
*
DCACFFS4 DS XL4 FILE SIZE (4 BYTE !) 319
DCACFLP4 DS XL4 LAST HALF PAGE POINTER (4 BYTE !) 319
DS XL2 RESERVED; MUST BE ZERO 319
DS (( *-DCACFMP+3)/4)*4+DCACFMP-*)XL1 310
DCACFMP# EQU *-DCACFMP LENGTH OF STRUCTURE
*
*
* -----
* DECLARE NEW HSMS STRUCTURE FOR CATALOG ENTRY
* -----
*
DCACFHP DSECT
DCACFHVS DS XL6 HSMS VOLUME SERIAL NUMBER
DCACFHVE DS XL1 VERSION OF HSMS STRUCTURE
DCACFHVL DS XL2 VARIABLE LENGTH OF HSMS STRUCTURE
DCACFHP# EQU *-DCACFHP
*
* -----
* DECLARE NEW PROFILE PART (VARIABLE IN LENGTH)

```

```

* -----
*
DCACFPP      DSECT
DCACFPPL    DS      XL2          LENGTH OF WHOLE PROFILE PART
DCACFPVP    DS      XL1          VERSION OF PROFILE PART
DCACFPPU    DS      XL4          :* Field has never been used
* -----
* - HERE STARTS THE VARIABLE PART OF PROFILE NAME LIST
* - (THREE ENTRIES ARE DEFINED, 18 BYTES PER ENTRY)
* - PROFILE NAME: $USERID.NAME OR BLANK (MEANS UNDEFINED)
* -----
DCACFPRD    DS      CL18         NAME OF READ PROFILE
DCACFPWR    DS      CL18         NAME OF WRITE PROFILE
DCACFPEX    DS      CL18         NAME OF EXEC PROFILE
            DS      XL3          ***      RESERVED      ***
DCACFPP#    EQU     *-DCACFPP
            EJECT ,
            IDCE D,D,NEWONLY=*YES
*
* -----
* - OLD CATALOG ENTRY FIXED PART
* -----
            MFPRE DNAME=DCES,MF=D,ALIGN=X,PREFIX=D                300
DDCES      DSECT ,
            *,##### PREFIX=D, MACID= #####
DDCEES    DS      XL2          0-1 ENTRY SIZE
DDCEPBN   DS      XL2          2-3 PRIMARY BLOCK NUMBER
DDCELBN   DS      XL2          4-5 BLOCK NUMBER
DDCEDMS   DS      XL1          6 DMS INDICATOR 1
DDCEINDX  EQU     X'01'        S FILE GENERATION GROUP INDEX -V4.0-
DDCEEPP   EQU     X'02'        S EXEC PASSWORD
DDCESF    EQU     X'04'        S SHARED FILE
DDCEROA   EQU     X'08'        S READ ONLY ACCESS
DDCERPP   EQU     X'10'        S READ PASSWORD
DDCEWPP   EQU     X'20'        S WRITE PASSWORD
DDCEAMN   EQU     X'C0'        PRINTER CONTROL
DDCEN     EQU     X'C0'        R NO CONTROL CHARACTER
DDCEM     EQU     X'40'        S MACHINE CODE CONTROL CHARACTER
DDCEA     EQU     X'C0'        S ASA CONTROL CHARACTER
DDCEDMS2  DS      XL1          7 DMS INDICATOR 2
*          EQU     X'80'        S USED BY FSTAT MACRO/CMD-NON SHARAB
*                               (ONLY VERSIONS < V8.0)          V10.0
DDCEFAR   EQU     X'80'        S UDS - FAR INTERFACE          V10.0
DDCEFGEN  EQU     X'40'        S FILE IS A GENERATION
DDCEFRAP  EQU     X'20'        S FOREIGN RANDOM ACCESS PROCESSING
*                               USED BY FILE MACRO AND CMS
DDCEISID  EQU     X'10'        S ISAM INDEX AND DATA VOLUMES ARE
*                               PRESENT IN CATALOG ENTRY

```

```

DDCERLK EQU X'08' S: RELEASE LOCK
DDCEELK EQU X'04' S: ERASE LOCK
DDCERPI EQU X'02' S: FILE TO REPAIR INDICATOR (RPI)
15:29:06 2001-10-15 PAGE

```

0058

SOURCE STATEMENT

```

DDCEENCR EQU X'01' S: ENTRY WIYH ENCRYPTED PASSW
DDCERD DS CL4 8-11 READ PASSWORD
DDCEWRT DS CL4 12-15 WRITE PASSWORD
DDCESA DS XL2 16-17 SECONDARY ALLOCATION
DDCESAC DS XL1 18 SECONDARY ALLOCATION COUNT
DDCEXTD DS XL1 19 DISPLACEMENT TO CATALOG ENTRY
*
DS CL6 UNDEFINDED DATA
DDCEMACL DS XL1 26 MINI - ACL V10.0
DDCEURD EQU X'80' S USER: READ-PRIVILEG V10.0
DDCEUWR EQU X'40' S USER: WRITE-PRIVILEG V10.0
DDCEUEX EQU X'20' S USER: EXEC-PRIVILEG V10.0
DDCEGRD EQU X'10' S GROUP: READ-PRIVILEG V10.0
DDCEGWR EQU X'08' S GROUP: WRITE-PRIVILEG V10.0
DDCEGEX EQU X'04' S GROUP: EXEC-PRIVILEG V10.0
DDCEORD EQU X'02' S OTHERS: READ-PRIVILEG V10.0
DDCEOWR EQU X'01' S OTHERS: WRITE-PRIVILEG V10.0
DDCEDTVC DS XL1 27 DATA ON TAPE VOLUME COUNT
DDCEFTP DS XL1 28 FCB FILE TYPE
DDCEFCBT EQU X'C0' FCB FILE TYPE
DDCESAM EQU X'C0' R SAM FILE
DDCEISAM EQU X'40' S ISAM FILE
DDCEBTAM EQU X'80' S BTAM FILE
DDCEPAM EQU X'C0' S PAM FILE
DDCERB DS XL1 29 RESTRICT BYTE
DDCEPRV EQU X'01' S PRIVATE VOLUME
DDCEPUB EQU X'01' R PUBLIC VOLUME
DDCERES EQU X'02' S SYSRES RESTRICTED

```

15:29:06 2001-10-15 PAGE

0059

SOURCE STATEMENT

```

DDCEFOR EQU X'04' S FOREIGN TAPE VOLUME
DDCEVIF EQU X'08' S VERY IMPORTANT FILE ( VIF )
DDCEIDVC EQU X'40' S-IGNORE DATA VOLUME COUNT
*
FOR TAPE EOF PROCESSING
* EQU X'10' S ERASE/CATAL COMMAND OR MACRO V10.0
* IS ACTIVE V10.0
* (ONLY VERSIONS < V8.0) V10.0
* EQU X'20' S OPEN/CLOSE IN PROCESS V10.0
* (ONLY VERSIONS < V9.0) V10.0
DDCEDFB EQU X'80' S File contains defective block
DDCEBLK DS XL2 30-31 BUFFER SIZE

```

```

DDCESTD EQU X'80' S PAM CREATED FILE
DDCCODE DS XL1 32 CODE OF TAPE
DDCISO7 EQU X'10' ISO7 CODE
DDCEBCD EQU X'08' EBCDIC CODE
DDCOWN EQU X'04' OWN CODE ISO7
DDCNDEF EQU X'1C' CODE NOT DEFINED
DDCERT DS XL1 33 RECORD TYPE
DDCEVFU EQU X'06' RECORD TYPE
DDCERTV EQU X'02' S VARIABLE
DDCERTF EQU X'04' S FIXED
DDCERTU EQU X'06' S UNDEFINED
DDCEREC DS XL2 34-35 RECORD LENGTH
DDCEKLC DS XL2 36-37 KEY LOCATION
DDCEKLN DS XL1 38 KEY LENGTH
DDCEFSZ3 DS XL3 39-41 FILE SIZE 306
DDCELRP DS XL2 42-43 LAST RECORD POINTER
DDCEIRA DS XL1 44 EDT VERSION-COUNTER
DDCELPP3 DS XL3 45-47 LAST 1/2 PAGE POINTER 306
DDCEFND DS XL4 48-51 FILE NUMBER
DDCEVTD DS XL2 52-53 DISPLACEMENT TO THE LAST 020
* - VSN TABLE ENTRY 020
DDCEVN1 DS XL1 54 CURRENT VERSION NUMBER
DDCEVN2 DS XL1 55 BACK-UP VERSION NUMBER
DDCETYP DS XL1 56 DEVICE TYPE
DDCEVC DS XL1 57 VSN COUNT
DDCEF1 DS XL1 58 FORMAT 1 POINTER
DDCEFNC DS XL1 59 FILENAME COUNT
DDCELEN EQU *-DDCEES LENGTH OF FIXED PORTION
DDCEFNM DS CL1 60-VAR FILENAME
* -
DDCEESF EQU 1881 MAXIMAL SIZE OF EXTENTLIST WITH 3 BYTE
* - OR FILABEL 307
DDCEES4 EQU 2496 MAXIMAL SIZE OF EXTENTLIST WITH 4 BYTE 306
EJECT ,
IDCEX D,D,NEWONLY=*YES
*
*****
*** CATALOG ENTRY EXTENSION
*****
MFPRE DNAME=DCEXS,MF=D,ALIGN=X,PREFIX=D 300
DDCEXS DSECT ,
*,##### PREFIX=D, MACID= #####
* CEL
DDCEEXTS EQU * START OF CATALOG ENTRY EXTENSION
DDCECNT DS XL2 FILE RECONSTRUCTION COUNT CEL
ORG DDCECNT CEL
DDCESESN DS XL1 SESSION NUMBER CEL
DDCEOICI DS XL1 OPEN-CLOSE INDICATOR -- OCI CEL

```

DDCECMSI	DS	XL1	CMS INDICATOR FLAGS	CEL
DDCEWRCH	EQU	X'C0'	7-6 WRITE COUNT (MODULO 4)	CEL
DDCEINCR	EQU	X'40'	COUNT INCREMENT	CEL
DDCEEXCL	EQU	X'20'	5 S=EXCLUSIVE SECURE	CEL
DDCESVAC	EQU	X'10'	4 S=ACCESSIBLE FOR SERVICE	CEL
DDCEKEYZ	EQU	X'0C'	3-2 S: BLKCTRL=NONE	9.5
*			R: BLKCTRL=PAMKEY	9.5
DDCEKEYD	EQU	X'08'	3 S: BLKCTRL=NO	9.5
DDCEKEYV	EQU	X'04'	2 S: BLKCTRL=DATA	9.5
DDCESPCE	EQU	X'02'	1 RESERVED FOR SPANNED CE'S	CEL
DDCESPC1	EQU	X'01'	0 RESERVED FOR SPANNED CE'S	CEL
*				CEL
	DS	XL1	OPEN COUNTER	V10.0
*			(ONLY VERSIONS < V9.0)	V10.0
DDCEDMS3	DS	X	DMS INDICATOR 3	CEL
DDCEPWSK	EQU	X'80'	SKIP PASSWORDS	CEL
DDCEPSCL	EQU	X'40'	SET: PSEUDO - CLOSE	CEL
DDCETEMP	EQU	X'20'	SET: TEMPORARY INDICATOR	CEL
DDCENIMC	EQU	X'10'	SET: NOT TO BE ERASED BY IMCAT	CEL
DDCESPEC	EQU	X'08'	SET: PLAM-FILE INDICATOR	9.5
DDCESUSA	EQU	X'04'	SPEC_USER_SPACE_ACCG	CEL
15:29:06 2001-10-15 PAGE				
0062				
SOURCE STATEMENT				
DDCEOEX	EQU	X'02'	S: MINI-ACL: OTHERS: EXEC-PRIV. V10.0	
DDCEFACP	EQU	X'01'	S: FULL ACL PRESENT	032
DDCAUDIT	DC	X'00'	AUDIT KEY	CEL
DDCAUDNO	EQU	X'00'	AUDIT=NONE	CEL
DDCAUDSU	EQU	X'01'	AUDIT=SUCC	CEL
DDCAUDFA	EQU	X'02'	AUDIT=FAIL	CEL
DDCAUDAL	EQU	X'04'	AUDIT=ALL	CEL
DDCEFRS	DS	XL1	FILE RECONSTRUCTION SYSTEM CONTROL	CEL
*			BYTE	CEL
DDCEFRRC	EQU	X'80'	S RECON=YES	CEL
DDCEFRRS	EQU	X'40'	S RESET=YES	CEL
DDCEOPT	EQU	X'20'	S CANNOT OPEN FILE OUTPUT/OUTIN	CEL
DDCECHNG	EQU	X'10'	S CALL FILE RECONSTRUCTION SYSTEM	CEL
*			BEFORE OPEN	CEL
DDCEMODE	EQU	X'08'	S MOVE MODE ONLY	CEL
DDCERLSK	EQU	X'04'	S RESET FUNCTION PENDING--FILE	CEL
*			CANNOT BE OPENED	CEL
DDCMIGFB	EQU	3	MIGRATION FORBIDDEN	
DDCERESS	EQU	2	MIGRATION INHIBIT	
DDCETRES	EQU	1	MIGRATION LIMITED	
DDCEIIND	DS	XL1	FILE INDICATOR BYTE	CEL
DDCEMACP	EQU	X'80'	S MINI ACL PRESENT	032
DDCESUW1	EQU	X'40'	SHARED UPDATE WEAK	100
DDCESUW2	EQU	X'20'	SHARED UPDATE WEAK	

```

DDCESUW3 EQU X'10'          SHARED UPDATE WEAK
DDCESUW4 EQU X'08'          SHARED UPDATE WEAK
DDCENKI4 EQU X'02'          NK-ISAM 4K-BLOCK-FILE          100
DDCEVMIN EQU X'01'          R MIN FUNCTION FOR ISAM VALUE FLAG
DDCEVMAX EQU X'01'          S MAX FUNCTION FOR ISAM VALUE FLAG
DDCELOGL DS XL1             ISAM LOGICAL FLAG LENGTH          CEL
DDCEVALL DS XL1             ISAM VALUE FLAG LENGTH          CEL
DDCESAD DS XL2              SECONDARY ALLOCATION(DATA)          CEL
DDCEITN8 DS XL1             ITN OF LOCKER WITH VERSION # < 9 CEL
DDCELOCK DS XL1             LOCKS                              CEL
*                               XXXX.... OUTPUT LOCKS          CEL
*                               ....XXXX CATALOG LOCKS          CEL
*                               CEL
DDCE$UID DS CL8              ALPHA-NUMERIC $USERID            CEL
          DS XL4              UNDEFINED DATA
DDCEEX DS XL4                EXEC PASSWORD                    CEL
DDCEFLX DS XL1              LEFTMOST BYTE OF F1 POINTER        CEL
*                               CEL
DDCEMEMB DS 4CL2            SHARER-ID/INPUT COUNTER (4 TIMES) CEL
* SHARER-1 : SHARER-ID : 1 BYTE ; INPUT COUNTER : 1 BYTE      CEL
* SHARER-2 : SAME --> 2 BYTES                                  CEL
* SHARER-3 : SAME --> 2 BYTES                                  CEL
* SHARER-4 : SAME --> 2 BYTES                                  CEL
          ORG DDCEMEMB                                          CEL
* THE BYTES USED BY THE PRIVATE DISK FILES ARE REDEFINED FOR USE BY CEL
* TAPE FILES.                                                  CEL
DDCETPID DS X                TAPE INDICATORS                  CEL
DDCEPFT EQU X'80'            S - PERMANENT FOREIGN TAPE        CEL
DDCESTD EQU X'40'            S - STD LABELS                    CEL
DDCENSTD EQU X'20'           S -NSTD LABELS                    CEL
DDCENOLB EQU X'10'           S -NO LABELS                      CEL
DDCEDIN3 EQU X'08'           S -LABEL TYPE=DIN3                CEL
DDCEDIN2 EQU X'04'           S -          DIN2                  CEL
DDCEDIN1 EQU X'02'           S -          DIN1                  CEL
DDCEBS2 EQU X'01'           S -          BS2000                 CEL
DDCEBUFO DS X                BUFFER OFFSET                    CEL
DDCEFSEQ DS XL2              FILE SEQUENCE NUMBER              CEL
DDCEDLAB DS X                DIN LABEL                          100
DDCEDL00 EQU X'00'           DIN LABEL UNDEFINED              100
          DS CL3              UNUSED                            CEL
*                               CEL
*                               CEL
DDCEARLV DC X'00'            CEL
* THE FIRST 4 BITS ARE RESERVED FOR SAVE/RESTORE              CEL
DDCEARCA EQU X'00'           BACKUP LEVEL=A                    CEL
DDCEARCB EQU X'10'           BACKUP LEVEL=B                    CEL
DDCEARCC EQU X'20'           BACKUP LEVEL=C                    CEL
DDCEARCD EQU X'30'           BACKUP LEVEL=D                    CEL

```



```

                DMACID=CAE , SUPPORT=( D , C ) , DNAME=CAEE4
DCAEE4  DSECT ,
                * ,##### PREFIX=D, MACID=CAE #####
*   CATALOG ENTRY EXTENT LIST ENTRY WITH 4 BYTE
DCAELHP4 DS    XL4                4 BYTE LOGICAL HALF-PAGE
DCAEPHP4 DS    XL4                4 BYTE PHYSICAL HALF-PAGE
DCAE#    EQU   *-DCAELHP4
                EJECT ,
                IDCEG D,D
*
* -----
* - CATALOG ENTRY EXTENSION FOR FILE GENERATION GROUP
* - THIS ENTRY IS LINKED AFTER CATALOG ENTRY EXTENSION (IDCEX)
* - (INSTEAD OF VOLUME TABLE (IDVT))
* -----
                MFPRE DNAME=DCEG , MF=D , ALIGN=X , PREFIX=D                300
DDCEG   DSECT ,
                * ,##### PREFIX=D, MACID= #####
DDCEGBAS DS    XL2                BASE FOR RELATIVE ADDRESSING
DDCEGCUR DS    XL2                CURRENT GENERATION
DDCEGBEG DS    XL2                FIRST GENERATION
DDCEGGEN DS    XL2                GEN=
DDCEGDIS DS    X                  DISP=
DDCEGCYC EQU   X'01'              =CYCLE
DDCEGDEL EQU   X'02'              =DELETE
DDCEGREU EQU   X'03'              =REUSE
DDCEGKEE EQU   X'04'              =KEEP                700
DDCEGDEV DS    X                  DEVICE TYPR (PR.DISK)    003
DDCEGVSN DS    CL6                VOLUME # (PR.DISK)      003
DDCEGNUS DS    CL5                NOT USED                003
DDCEGLEN EQU   *-DDCEGBAS
                IDCE D,D,NEWONLY=*YES
*
* -----
* - OLD CATALOG ENTRY FIXED PART
* -----
                MFPRE DNAME=DCES , MF=D , ALIGN=X , PREFIX=D                300
DDCES   DSECT ,
                * ,##### PREFIX=D, MACID= #####
DDCEES  DS    XL2                0-1 ENTRY SIZE
DDCEPBN DS    XL2                2-3 PRIMARY BLOCK NUMBER
DDCELBN DS    XL2                4-5 BLOCK NUMBER
DDCEDMS DS    XL1                6 DMS INDICATOR 1
DDCEINDX EQU   X'01'              S FILE GENERATION GROUP INDEX -V4.0-
DDCEEPP EQU   X'02'              S EXEC PASSWORD
                                           15:29:06 2001-10-15 PAGE
0069
SOURCE STATEMENT

```

DDCESF	EQU	X'04'	S SHARED FILE	
DDCEROA	EQU	X'08'	S READ ONLY ACCESS	
DDCERPP	EQU	X'10'	S READ PASSWORD	
DDCEWPP	EQU	X'20'	S WRITE PASSWORD	
DDCEAMN	EQU	X'C0'	PRINTER CONTROL	
DDCEN	EQU	X'C0'	R NO CONTROL CHARACTER	
DDCEM	EQU	X'40'	S MACHINE CODE CONTROL CHARACTER	
DDCEA	EQU	X'C0'	S ASA CONTROL CHARACTER	
DDCEDMS2	DS	XL1	7 DMS INDICATOR 2	
*	EQU	X'80'	S USED BY FSTAT MACRO/CMD-NON SHARAB	
*			(ONLY VERSIONS < V8.0)	V10.0
DDCEFAR	EQU	X'80'	S UDS - FAR INTERFACE	V10.0
DDCEFGEN	EQU	X'40'	S FILE IS A GENERATION	
DDCEFRAP	EQU	X'20'	S FOREIGN RANDOM ACCESS PROCESSING	
*			USED BY FILE MACRO AND CMS	
DDCEISID	EQU	X'10'	S ISAM INDEX AND DATA VOLUMES ARE	
*			PRESENT IN CATALOG ENTRY	
DDCERLK	EQU	X'08'	S: RELEASE LOCK	
DDCEELK	EQU	X'04'	S: ERASE LOCK	
DDCERPI	EQU	X'02'	S: FILE TO REPAIR INDICATOR (RPI)	
DDCEENCR	EQU	X'01'	S: ENTRY WIYH ENCRYPTED PASSW	
DDCERD	DS	CL4	8-11 READ PASSWORD	
DDCEWRT	DS	CL4	12-15 WRITE PASSWORD	
DDCESA	DS	XL2	16-17 SECONDARY ALLOCATION	
DDCESAC	DS	XL1	18 SECONDARY ALLOCATION COUNT	
DDCEEXTD	DS	XL1	19 DISPLACEMENT TO CATALOG ENTRY	
*			EXTENSION	
	DS	CL6	UNDEFINDED DATA	
DDCEMACL	DS	XL1	26 MINI - ACL	V10.0
DDCEURD	EQU	X'80'	S USER: READ-PRIVILEG	V10.0
DDCEUWR	EQU	X'40'	S USER: WRITE-PRIVILEG	V10.0
DDCEUEX	EQU	X'20'	S USER: EXEC-PRIVILEG	V10.0
DDCEGRD	EQU	X'10'	S GROUP: READ-PRIVILEG	V10.0
DDCEGWR	EQU	X'08'	S GROUP: WRITE-PRIVILEG	V10.0
DDCEGEX	EQU	X'04'	S GROUP: EXEC-PRIVILEG	V10.0
DDCEORD	EQU	X'02'	S OTHERS: READ-PRIVILEG	V10.0
DDCEOWR	EQU	X'01'	S OTHERS: WRITE-PRIVILEG	V10.0
DDCEDTVC	DS	XL1	27 DATA ON TAPE VOLUME COUNT	
DDCEFTP	DS	XL1	28 FCB FILE TYPE	
DDCEFCBT	EQU	X'C0'	FCB FILE TYPE	
DDCESAM	EQU	X'C0'	R SAM FILE	
DDCEISAM	EQU	X'40'	S ISAM FILE	
DDCEBTAM	EQU	X'80'	S BTAM FILE	
DDCEPAM	EQU	X'C0'	S PAM FILE	
DDCERB	DS	XL1	29 RESTRICT BYTE	
DDCEPRV	EQU	X'01'	S PRIVATE VOLUME	
DDCEPUB	EQU	X'01'	R PUBLIC VOLUME	
DDCERES	EQU	X'02'	S SYSRES RESTRICTED	

```

DDCEFOR EQU X'04' S FOREIGN TAPE VOLUME
DDCEVIF EQU X'08' S VERY IMPORTANT FILE ( VIF )
DDCEIDVC EQU X'40' S-IGNORE DATA VOLUME COUNT
* FOR TAPE EOF PROCESSING
* EQU X'10' S ERASE/CATAL COMMAND OR MACRO V10.0
* IS ACTIVE V10.0
* (ONLY VERSIONS < V8.0) V10.0
* EQU X'20' S OPEN/CLOSE IN PROCESS V10.0
* (ONLY VERSIONS < V9.0) V10.0
DDCEDFB EQU X'80' S File contains defective block
DDCEBLK DS XL2 30-31 BUFFER SIZE
DDCESTD EQU X'80' S PAM CREATED FILE
DDCCODE DS XL1 32 CODE OF TAPE
DDCISO7 EQU X'10' ISO7 CODE
DDCEBCD EQU X'08' EBCDIC CODE
DDCOWN EQU X'04' OWN CODE ISO7
DDCNDEF EQU X'1C' CODE NOT DEFINED
DDCERT DS XL1 33 RECORD TYPE
DDCEVFU EQU X'06' RECORD TYPE
DDCERTV EQU X'02' S VARIABLE
DDCERTF EQU X'04' S FIXED
DDCERTU EQU X'06' S UNDEFINED
DDCEREC DS XL2 34-35 RECORD LENGTH
DDCEKLC DS XL2 36-37 KEY LOCATION
DDCEKLH DS XL1 38 KEY LENGTH
DDCEFSZ3 DS XL3 39-41 FILE SIZE 306
DDCELRP DS XL2 42-43 LAST RECORD POINTER
DDCEIRA DS XL1 44 EDT VERSION-COUNTER
DDCELPP3 DS XL3 45-47 LAST 1/2 PAGE POINTER 306
DDCEFN DS XL4 48-51 FILE NUMBER
DDCEVTD DS XL2 52-53 DISPLACEMENT TO THE LAST 020
* _ VSN TABLE ENTRY 020
DDCEVN1 DS XL1 54 CURRENT VERSION NUMBER
DDCEVN2 DS XL1 55 BACK-UP VERSION NUMBER
DDCETYP DS XL1 56 DEVICE TYPE
DDCEVC DS XL1 57 VSN COUNT
DDCEF1 DS XL1 58 FORMAT 1 POINTER
DDCEFNC DS XL1 59 FILENAME COUNT
DDCELEN EQU *-DDCEES LENGTH OF FIXED PORTION
DDCEFNM DS CL1 60-VAR FILENAME
* _
DDCEESF EQU 1881 MAXIMAL SIZE OF EXTENTLIST WITH 3 BYTE
* _ OR FLLABEL 307
DDCEES4 EQU 2496 MAXIMAL SIZE OF EXTENTLIST WITH 4 BYTE 306
IDCEX D,D,NEWONLY=*YES
*
*****
*** CATALOG ENTRY EXTENSION

```

```

*****
MFPRE DNAME=DCEXS,MF=D,ALIGN=X,PREFIX=D 300
DDCEXS DSECT ,
      *,##### PREFIX=D, MACID= #####
*
DDCEEXTS EQU * START OF CATALOG ENTRY EXTENSION CEL
DDCECNT DS XL2 FILE RECONSTRUCTION COUNT CEL
      ORG DDCECNT CEL
DDCESESN DS XL1 SESSION NUMBER CEL
DDCEOCI DS XL1 OPEN-CLOSE INDICATOR -- OCI CEL
DDCECMSI DS XL1 CMS INDICATOR FLAGS CEL
DDCEWRCH EQU X'C0' 7-6 WRITE COUNT (MODULO 4) CEL
DDCEINCR EQU X'40' COUNT INCREMENT CEL
DDCEEXCL EQU X'20' 5 S=EXCLUSIVE SECURE CEL
DDCESVAC EQU X'10' 4 S=ACCESSIBLE FOR SERVICE CEL
DDCEKEYZ EQU X'0C' 3-2 S: BLKCTRL=NONE 9.5
* R: BLKCTRL=PAMKEY 9.5
DDCEKEYD EQU X'08' 3 S: BLKCTRL=NO 9.5
DDCEKEYV EQU X'04' 2 S: BLKCTRL=DATA 9.5
DDCESPCE EQU X'02' 1 RESERVED FOR SPANNED CE'S CEL
DDCESPC1 EQU X'01' 0 RESERVED FOR SPANNED CE'S CEL
* CEL
      DS XL1 OPEN COUNTER V10.0
* (ONLY VERSIONS < V9.0) V10.0
DDCEDMS3 DS X DMS INDICATOR 3 CEL
DDCEPSK EQU X'80' SKIP PASSWORDS CEL
DDCEPSCL EQU X'40' SET: PSEUDO - CLOSE CEL
DDCETEMP EQU X'20' SET: TEMPORARY INDICATOR CEL
DDCENIMC EQU X'10' SET: NOT TO BE ERASED BY IMCAT CEL
DDCESPEC EQU X'08' SET: PLAM-FILE INDICATOR 9.5
DDCESUSA EQU X'04' SPEC_USER_SPACE_ACCG CEL
DDCEOEX EQU X'02' S: MINI-ACL: OTHERS: EXEC-PRIV. V10.0
DDCEFACP EQU X'01' S: FULL ACL PRESENT 032
DDCAUDIT DC X'00' AUDIT KEY CEL
DDCAUDNO EQU X'00' AUDIT=NONE CEL
DDCAUDSU EQU X'01' AUDIT=SUCG CEL
DDCAUDFA EQU X'02' AUDIT=FAIL CEL
DDCAUDAL EQU X'04' AUDIT=ALL CEL
DDCEFRS DS XL1 FILE RECONSTRUCTION SYSTEM CONTROL CEL
* BYTE CEL
DDCEFRR EQU X'80' S RECON=YES CEL
DDCEFRRS EQU X'40' S RESET=YES CEL
DDCEOPT EQU X'20' S CANNOT OPEN FILE OUTPUT/OUTIN CEL
DDCECHNG EQU X'10' S CALL FILE RECONSTRUCTION SYSTEM CEL
* BEFORE OPEN CEL
DDCEMODE EQU X'08' S MOVE MODE ONLY CEL
DDCERLSK EQU X'04' S RESET FUNCTION PENDING--FILE CEL
* CANNOT BE OPENED CEL

```

DDCMIGFB	EQU	3	MIGRATION FORBIDDEN	
DDCERESS	EQU	2	MIGRATION INHIBIT	
DDCETRES	EQU	1	MIGRATION LIMITED	
DDCEIIND	DS	XL1	FILE INDICATOR BYTE	CEL
DDCEMACP	EQU	X'80'	S MINI ACL PRESENT	032
DDCESUW1	EQU	X'40'	SHARED UPDATE WEAK	100
DDCESUW2	EQU	X'20'	SHARED UPDATE WEAK	
DDCESUW3	EQU	X'10'	SHARED UPDATE WEAK	
DDCESUW4	EQU	X'08'	SHARED UPDATE WEAK	
DDCENKI4	EQU	X'02'	NK-ISAM 4K-BLOCK-FILE	100
DDCEVMIN	EQU	X'01'	R MIN FUNCTION FOR ISAM VALUE FLAG	
DDCEVMAX	EQU	X'01'	S MAX FUNCTION FOR ISAM VALUE FLAG	
DDCELOGL	DS	XL1	ISAM LOGICAL FLAG LENGTH	CEL
DDCEVALL	DS	XL1	ISAM VALUE FLAG LENGTH	CEL
DDCESAD	DS	XL2	SECONDARY ALLOCATION(DATA)	CEL
DDCEITN8	DS	XL1	ITN OF LOCKER WITH VERSION # < 9	CEL
DDCELOCK	DS	XL1	LOCKS	CEL
*			XXXX... OUTPUT LOCKS	CEL
*			...XXXX CATALOG LOCKS	CEL
*				CEL
DDCE\$UID	DS	CL8	ALPHA-NUMERIC \$USERID	CEL
	DS	XL4	UNDEFINDED DATA	
DDCEEX	DS	XL4	EXEC PASSWORD	CEL
DDCEFLX	DS	XL1	LEFTMOST BYTE OF F1 POINTER	CEL
*				CEL
DDCEMEMB	DS	4CL2	SHARER-ID/INPUT COUNTER (4 TIMES)	CEL
*			SHARER-1 : SHARER-ID : 1 BYTE ; INPUT COUNTER : 1 BYTE	CEL
*			SHARER-2 : SAME --> 2 BYTES	CEL
*			SHARER-3 : SAME --> 2 BYTES	CEL
*			SHARER-4 : SAME --> 2 BYTES	CEL
	ORG	DDCEMEMB		CEL
*			THE BYTES USED BY THE PRIVATE DISK FILES ARE REDEFINED FOR USE BY	CEL
*			TAPE FILES.	CEL
DDCETPID	DS	X	TAPE INDICATORS	CEL
DDCEPFT	EQU	X'80'	S - PERMANENT FOREIGN TAPE	CEL
DDCESTD	EQU	X'40'	S - STD LABELS	CEL
DDCENSTD	EQU	X'20'	S -NSTD LABELS	CEL
DDCENOLB	EQU	X'10'	S -NO LABELS	CEL
DDCEDIN3	EQU	X'08'	S -LABEL TYPE=DIN3	CEL
DDCEDIN2	EQU	X'04'	S - DIN2	CEL
DDCEDIN1	EQU	X'02'	S - DIN1	CEL
DDCEBS2	EQU	X'01'	S - BS2000	CEL
DDCEBUFO	DS	X	BUFFER OFFSET	CEL
DDCEFSEQ	DS	XL2	FILE SEQUENCE NUMBER	CEL
DDCEDLAB	DS	X	DIN LABEL	100
DDCEDL00	EQU	X'00'	DIN LABEL UNDEFINED	100
	DS	CL3	UNUSED	CEL
*				CEL

```

*
DDCEARLV DC      X'00'                                CEL
* THE FIRST 4 BITS ARE RESERVED FOR SAVE/RESTORE      CEL
DDCEARCA EQU    X'00'      BACKUP LEVEL=A            CEL
DDCEARCB EQU    X'10'      BACKUP LEVEL=B            CEL
DDCEARCC EQU    X'20'      BACKUP LEVEL=C            CEL
DDCEARCD EQU    X'30'      BACKUP LEVEL=D            CEL
DDCEARCE EQU    X'40'      BACKUP LEVEL=E            CEL
*
*              ORG      DDCEARLV                        CEL
DDCEARFL DC      X'00'      FLAGS                      CEL
DDCELARG EQU     X'01'      LARGE FILE                  CEL
DDCEDEST EQU     X'02'      TO BE ERASED WITH DESTROY  CEL
DDCEPART EQU     X'04'      PART RESTORE                CEL
DDCEMIGR EQU     X'08'      S: MIGRATED (HSMS)          CEL
*
*              R: NOT MIGRATED (HSMS)                  CEL
*
*
*
DDCEEXTL EQU     *-DDCEEXTS          LENGTH OF CATALOG ENTRY EXTENSION
                IDVT D,D
*
* -----
* - VOLUME TABLE ENTRY
* -----
                MFPRE DNAME=DVTS, MF=D, ALIGN=X, PREFIX=D          300
DDVTS      DSECT ,
                *,##### PREFIX=D, MACID= #####
DDVTVSN   DS      CL6              0-5 VOLUME SERIAL NUMBER
DDVTLENT  EQU     *-DDVTVSN        LENGTH OF FIXED PORTION-TAPE
DDVTTYP   DS      XL1              6 DEVICE TYPE
DDVTIDV   EQU     X'40'            6-6 S ISAM DATA VOLUME
DDVTIIV   EQU     X'40'            6-6 R ISAM INDEX VOLUME
DDVT#EE   DS      XL1              7 NUMBER OF EXTENT ENTRIES FOR VSN
DDVTLEN   EQU     *-DDVTVSN        LENGTH OF FIXED PORTION-R.A.
                IDEE D,D
*
* -----
* - CATALOG ENTRY EXTENT LIST ENTRY WITH 3 BYTE
* -----
                MFPRE DNAME=DEES, MF=D, ALIGN=X, PREFIX=D          300
DDEES     DSECT ,
                *,##### PREFIX=D, MACID= #####
*
DDEELHP   DS      XL3              LOGICAL 1/2 PG
DDEEPHP   DS      XL3              PHYSICAL 1/2 PG
*
DDEELEN   EQU     *-DDEELHP        LENGTH OF ENTRY
                IDCEG D,D

```

```
*
* -----
* - CATALOG ENTRY EXTENSION FOR FILE GENERATION GROUP
* - THIS ENTRY IS LINKED AFTER CATALOG ENTRY EXTENSION (IDCEX)
* - (INSTEAD OF VOLUME TABLE (IDVT))
* -----
          MFPRE DNAME=DCEG,MF=D,ALIGN=X,PREFIX=D                300
DDCEG    DSECT  ,
          *,##### PREFIX=D, MACID= #####
DDCEGBAS DS    XL2      BASE FOR RELATIVE ADDRESSING
DDCEGCUR DS    XL2      CURRENT GENERATION
DDCEGBEG DS    XL2      FIRST GENERATION
DDCEGGEN DS    XL2      GEN=
DDCEGDIS DS    X        DISP=
DDCEGCYC EQU   X'01'    =CYCLE
DDCEGDEL EQU   X'02'    =DELETE
DDCEGREU EQU   X'03'    =REUSE
DDCEGKEE EQU   X'04'    =KEEP                                700
DDCEGDEV DS    X        DEVICE TYPR (PR.DISK)                003
DDCEGVSN DS    CL6      VOLUME #   (PR.DISK)                003
DDCEGNUS DS    CL5      NOT USED                               003
DDCEGLEN EQU   *-DDCEGBAS
```

6.1.6 \$JTBP

```

$JTBP MF=D
MFTST MF=D,PREFIX=E,MACID=JT,ALIGN=F, C
      DMACID=JT,SUPPORT=(D,C),DNAME=JTBP
EJTBP  DSECT ,
      *,##### PREFIX=E, MACID=JT #####
EJTPLEN      EQU    608
*
EJTPIDNT      DS     CL4      String 'JTBP' identifies
*              itself
EJTPVERS      DS     CL3      BS2000-Version that created
*              this JTBP
EJTPJOMT      DS     FL1      indicates which structure of
*              union 'miscellaneous' (see
*              below) is valid
* indicates which of the structures in union 'miscellaneous' is
* valid
EJTPSMUN      EQU    0        X'00'
EJTSMST       EQU    40       ' ': start
EJTSMCA       EQU    195      'C': calendar
EJTSMRP       EQU    217      'R': repeat
EJTSMSSU      EQU    226      'S': subtask
EJTRBP        EQU    231      'X': remote batch
*
*
EJTPJOB      DS     0XL44     JOBBAS: basic job data
EJTPUSR      DS     CL8       (USERID) user id
EJTPACT      DS     CL8       (ACT#) account number
EJTSPSWD     DS     CL8       password of user Id - only if
*                             invalid! (for use in
*                             exit#030)
EJTPJCLS     DS     CL8       job class
EJTJNAM      DS     CL8       (JOBNAME) job name
EJTPTS      DS     CL4       TSN
*
*
EJTPST       DS     0XL12     STATE: data on job state
EJTPSTCI     DS     A         (CANCEL_INFO@) pointer to
*                             cancel context
EJTPSTST     DS     F         time the job started
EJTPSTQU     DS     X         (QUE#) queue number
EJTPSTUP     DS     FL1       (PRIVG_USER, PRI_USER)
*                             indicates normal user,
*                             system administrator or
*                             operator
* indicates normal user, system administrator or operator

```

EJTPSUTO	EQU	1	TSOS
EJTPSUUS	EQU	2	normal user
EJTPSUOP	EQU	4	operator
*			
EJTPSTF	DS	0XL1	
EJTPSTFL	DS	AL1	
EJTPSTFI	EQU	X'80'	job held: /HOLD-JOB given
EJTPSTFT	EQU	X'40'	a /CANCEL-JOB command has
*			been given
EJTPSTFS	EQU	X'20'	a /CAN-JOB STEPS=*CURR has
*			been given
EJTPSTFC	EQU	X'10'	a /CAN-JOB
*			STEPS=*ALL-CALENDAR-REPET
*			has been given
EJTPSTIE	EQU	X'08'	job is being extracted
EJTPSTTE	EQU	X'04'	special termination for
*			jobs-in-extract
EJTPSTFJ	EQU	X'02'	job held: missing cal cat
EJTPSTFR	EQU	X'01'	unused
ORG	EJTPSTF		
EJTPSTOL	DS	AL1	
EJTPSTOI	EQU	X'80'	(HOLD) job is held
EJTPSTOR	EQU	X'7F'	unused
ORG	EJTPSTF+1		
EJTPSTRS	DS	XL1	unused
*			
*			
EJTPCA	DS	0XL32	CALDAT: data of caller
EJTPCACR	DS	CL8	userid that created this task
EJTPCATS	DS	CL4	(ORIG_TSN) TSN of the job
*			that created this one
EJTPCAAU	DS	XL16	audit id (chipcard id of
*			original dialog job)
EJTPCAJT	DS	X	job type: see JATOTYP in
*			macro D\$JAT (resp.
*			JMS_JO_TYPE_SET in JMSSETI,
*			SPL)
EJTPCAFL	DS	AL1	
EJTPCAFA	EQU	X'80'	privileged acceptance or not
EJTPCAFC	EQU	X'40'	enter from console or not
EJTPCAFT	EQU	X'20'	rfa servertask or not
EJTPCAFR	EQU	X'1F'	unused
EJTPCARS	DS	XL2	unused
*			
*			
EJTPED	DS	0XL120	EDAT: batch data
EJTPSPOT	DS	F	time of job acceptance
EJTPEDFN	DS	CL54	(NON_CONV_TASK) name of enter

*			(SYSCMD) file
EJTPEDFP	DS	CL4	(ENTER_PSWD) password of
*			enter file
EJTPEDON	DS	CL54	name of the file that was
*			copied into an S.IN-file
EJTPEDRC	DS	H	repeat count
EJTPEDFL	DS	AL1	
EJTPEDF1	EQU	X'80'	(ENTER_ER) erase enter file
EJTPEDF2	EQU	X'40'	(NOT_ER) do not erase enter
*			file
EJTPEDFS	EQU	X'20'	don't count job in job class
EJTPEDFX	EQU	X'10'	S.OUT. files created by
*			forked tasks are printed
EJTPJNFE	EQU	X'08'	take JOB_NAME from ENTER-JOB
*			command
EJTPNLTM	EQU	X'04'	tasklocal time shall not
*			apply
EJTPDFE	EQU	X'02'	syscmd file is execute only
EJTPSPAR	EQU	X'01'	repeat jobs permit start time
*			in the past
EJTPEDRS	DS	XL1	unused
*			
*			
EJTPPA	DS	0XL204	PARLST: some other data
EJTPPAMN	DS	CL54	(JOB_VAR_NAME) name of
*			monitoring JV
EJTPPAMP	DS	CL4	(JV_PSWD) password of
*			monitoring JV
EJTPPARP	DS	X	(PRI) run priority for task
*			management
EJTPPAJP	DS	X	job priority
EJTPPASL	DS	F	line limit for SYSLST
EJTPPASO	DS	F	card limit for SYSOPT
EJTPPATL	DS	F	(TIME_VALUE) cpu time limit
*			for job
EJTPPARS	DS	XL2	unused
EJTPPAJC	DS	CL128	job class parameters
EJTPPAMS	DS	0XL1	message attributes
EJTPPAMD	DS	X	message attributes
ORG	EJTPPAMS		
EJTPPAMB	DS	XL1	unused
ORG	EJTPPAMS		
EJTPPAFM	DS	AL1	Meldungsattribute
EJTPPAM1	EQU	X'C0'	unused
EJTPPAML	EQU	X'20'	copy SYSOUT to SYSLST
EJTPPAM2	EQU	X'10'	unused
EJTPPAMT	EQU	X'08'	terminal hardcopy, unused
EJTPPAMH	EQU	X'04'	display console messages on

EJTPLIMM	DS	X	minutes
EJTPLIMD	DS	CL10	date
*			
*			
	ORG	EJTPMISC	
*			
EJTSPSU	DS	0XL60	SUBTASK: temporary data
*			during subtask proliferation
EJTSPSUOR	DS	CL8	name of subsystem that
*			created the subtask
EJTSPSUCA	DS	CL7	(CATEG) task start category
EJTSPSUSP	DS	X	(PRIVIL) set privileges
*			according to
*			PRIVILEGE_VECTOR
EJTSPSUPV	DS	XL42	(SRPM_DATA) privilege vector
*			for subtask
EJTSPSUTY	DS	X	(TASK_TYPE) subtask type
EJTSPSUCS	DS	X	(CMDCON) is task to be
*			connected to SDF?
*			
	ORG	EJTPMISC	
*			
EJTSPRB	DS	0XL34	RBP: data for Remote Batch
*			Processing
EJTSPRBNA	DS	CL8	job name of an RBP job
EJTSPRBSS	DS	CL8	(REM_STATION) name of station
*			that submitted the RB job
EJTSPRBDS	DS	CL8	(DEST_STATION) destination
*			station for SPOOL
EJTSPRBAU	DS	CL8	(ALT_USER) Who knows?
EJTSPRBBS	DS	X	(BTCH_STN) some data for
*			spool?
EJTSPRBU	DS	0XL1	
EJTRB_OUT_DEF	DS	CL1	remote terminal control
	ORG	EJTSPRBU	
EJTSPRBFL	DS	AL1	
EJTSPRBF1	EQU	X'F8'	unused
EJTSPRBF2	EQU	X'04'	send output to remote station
EJTSPRBF3	EQU	X'02'	unused
EJTSPRBF4	EQU	X'01'	hold spoolout until requested
	ORG	EJTSPRBU+1	
*			
	ORG	EJTPMISC+92	
*			
EJTSPRE	DS	0XL16	REPEAT: repeat job data
EJTSPRETY	DS	X	repeat type: see JATPJRT in
*			macro D\$JAT (resp.
*			JMS_REPEAT_SET in JMSSETI,

```

*
EJTPRERM          DS      0XL1          SPL)
EJTPREIN          DS      X            repeat indicator
      ORG      EJTPRERM
EJTPREIB          DS      AL1
EJTPREMF          EQU     X'FE'        unused
EJTPREMC          EQU     X'01'        monjv is definitively closed
*
      ORG      EJTPRERM+1
EJTPREIV          DS      H            repeat interval
EJTPRETI          DS      F            planned start time of next
*
      image
EJTPRETL          DS      F            time of last repeat
EJTPRETU          DS      0XL4
EJTPRETS          DS      CL4         tsn of last repeat
      ORG      EJTPRETU
EJTPRETF          DS      A
EJTPRETB          EQU     X'FFFFFFF'   tsn of last repeat is set
      ORG      EJTPRETU+4
*
*
EJTPJD            DS      0XL34        JDRES: job distribution data
EJTPJDHO          DS      CL8         (SUBMITT_JOB) name of host
*
      that sent the job
EJTPJDOX          DS      AL1
EJTPJDOE          EQU     X'80'        old parameter EXPRESS
EJTPJDXR          EQU     X'7F'        unused
EJTPJDOM          DS      XL1         old parameter MSG
EJTPJDOP          DS      X           old parameter PRIO
EJTPJDRH          DS      X           repeat hours
EJTPJDRM          DS      X           repeat minutes
EJTPJDSD          DS      X           start hours
EJTPJDSE          DS      X           start minutes
EJTPJDRS          DS      XL1        unused
EJTPJDSD          DS      CL10       start date
EJTPJDDE          DS      XL4        expl default-ind of JAP
EJTPJDDI          DS      XL4        impl default-ind of JAP
*
*
EJTPSQ            DS      0XL8        STATUS QUO: data describing
*
      rights that are recorded
*
      during job acceptance and
*
      used for task initialization
EJTPSQSC          DS      X           (CLASS_OUT)
EJTPSQMC          DS      CL3        (MAX_ALLOWED_CATEGORY)
EJTPSQMP          DS      X           (MAX_PRIORITY) maximum run
*
      priority
EJTPSQAR          DS      X           (READ) test option

```

```
EJTPSQAW          DS    X          (WRITE) test option
EJTPSQ00          DS    XL1         unused
*
*
EJTPTD            DS    0XL12       TO DELETE: data that should
*                    be eliminated from the JTBP,
*                    but are currently used by
*                    someone
EJTSPoolIN_TIME_STCK DS    XL8       (SPOOL_IN_TIME_STCK) kann aus
*                    SPOOLIN_TIME berechnet
*                    werden
EJTPTDBL          DS    H          (BUFF_PARAM) data available
*                    from $JCBRW
EJTPTDFL          DS    AL1
EJTPTDFE          EQU    X'80'      unused
EJTPTDFD          EQU    X'40'      syscmd is a /DATA file
EJTPTDFR          EQU    X'3F'      unused
EJTPTD00          DS    XL1         unused
*
*
EJTPTMO           DS    0XL4        MOVE: data for EXTRACT/IMPORT
EJTPTMOTS         DS    CL4        TSN of the job before
*                    EXTRACT/IMPORT
*
EJTPTMPTY         DS    XL14        unused
EJTPTM#           EQU    *-EJTPTDNT
```

6.1.7 SJMCHKZ

```

SJMCHKZ MF=D
*****
* ,SJMCHKZ': DATA FOR THE INTERFACE ,JOIN-CMD'. *
*-----*
* THE FOLLOWING DATA IS TRANSFERRED: *
* | ID | DATA | ID | DATA | .... | END_ID | *
* *
* THE 1-BYTE IDS AND THE IMPLICIT LENGTHS (WITH THE ID IN *
* EACH CASE) ARE DEFINED HERE: ID = KZ IN NAMES *
*****
*
*****
* 1. ID FOR THE ,USER PART' OF THE ,JOIN RECORD'. *
*****
*
* RECORD KEY
SJKZUSR EQU 1 ,USERID'
SJKZUSRL EQU 9
*
* GROUP
SJKZGRP EQU 2 ,GROUP='
SJKZGRPL EQU 9
*
* PASSWORD /PASSWORD OPERATIONS
SJKZPAS EQU 3 ,PASS='
SJKZPASL EQU 18 L=( ID,ENCR-ID,PSWORD1,PSWORD2 )
SJKZPNO EQU 0 ENCR-ID: DO NOT ENCRYPT,
SJKZPEN EQU 1 ENCRYPT.
*
SJKZECCM EQU 4 -FREE-
*
SJKZPSW EQU 5 ,PSWORD='
SJKZPWN EQU 1 TRANSFER VALUES.
SJKZPWM EQU 2
SJKZPWY EQU 3
SJKZPSWL EQU 2
*
* SEVER/LOCK INDIKATOR
SJKZSVR EQU 6 ,LOCK='
SJKZVRY EQU 1 TRANSFER VALUE
SJKZSVRL EQU 2
*
* PRIMARY CATALOG BLOCK NUMBER
SJKZPBN EQU 7 ,PBN'
SJKZPBNL EQU 3
*

```

		DEFAULT CATID		
*				
SJKZDPS	EQU	8		,DEF-PUBSET= '
SJKZDPSL	EQU	5		
*				
SJKZUPC	EQU	9		,USER PRIVILEGE CODE '
SJKZADMN	EQU	1		TRANSFER VALUES
SJKZNUSR	EQU	2		
SJKZUPCL	EQU	2		
*				
SJKZTPR	EQU	10		,TESTPRIV= '
SJKZPRY	EQU	1		
SJKZPRN	EQU	2		
SJKZTPRL	EQU	4		L= (ID , AID1 , AID2 , Y/N)
*				
SJKZCST	EQU	11		,CSTMP-MAC= '
SJKZCSY	EQU	1		
SJKZCSN	EQU	2		
SJKZCSTL	EQU	2		
*				
SJKZADT	EQU	12		,AUDIT= '
SJKZADY	EQU	1		
SJKZADN	EQU	2		
SJKZADTL	EQU	2		
*				
SJKZENF	EQU	13		,ENF= ' EXTENDED IN V11
SJKZEFY	EQU	1		DOES NOT MATCH VALUE STORED
SJKZEFN	EQU	2		IN USER RECORD.
SJKZEFT	EQU	3		
SJKZENFL	EQU	2		
*				
SJKZTPI	EQU	14		,TPIGNORE= '
SJKZTPIL	EQU	2		
*				
SJKZUSS	EQU	15		,USER SWITCHES '
SJKZUSSL	EQU	13		L= (KZ , ON-SW , OFF-SW , INV-SW)
*				
*				RESOURCES (USER TEIL)
SJKZADR	EQU	16		,ADDRSPACE= '
SJKZADRL	EQU	3		
*				
SJKZPUBS	EQU	17		,PUBSPACE= ' OR SPACE-LIMIT
SJKZPUBL	EQU	5		
*				
SJKZSUS	EQU	19		,SPACE-USED '
SJKZSUSL	EQU	5		
*				
SJKZRPGL	EQU	20		,RES-PAGES= '
SJKZRPGL	EQU	3		

```
*
SJKZMAR    EQU    21                ,MAX-ACC-REC='
SJKZMARL   EQU    3
*
*                MESSAGE HANDLING
SJKZMSG    EQU    22                ,MESSAGE SEARCH='
SJKZTSK    EQU    1
SJKZALL    EQU    2
SJKZMSGSL  EQU    2
*
SJKZDML    EQU    23                ,DEF-MSG-LAN='
SJKZDMLL   EQU    2
*
*                MAILING ADDRESS/ COMMAND FILE NAME
SJKZMAL    EQU    24                ,MAIL='
SJKZMALL   EQU    65
*
SJKZCMF    EQU    25                ,COMMANDS=' NEW: PROFILE ID
SJKZCMFL   EQU    55
*
*****
* EXTENSIONS IN V11: QUOTA, TUNING MEASURES *
*                EXTENDED HOST CODE. *
*****
*
SJKZTSL    EQU    35                ,TEMP-SPACE-LIMIT='
SJKZTSLL   EQU    5
*
SJKZTSU    EQU    36                ,TEMP SPACE USED'
SJKZTSUL   EQU    5
*
SJKZFLI    EQU    37                ,FILE-LIMIT='
SJKZFLIL   EQU    5
*
SJKZFLA    EQU    38                ,FILE AMOUNT'
SJKZFLAL   EQU    5
*
SJKZJVL    EQU    39                ,JV-LIMIT='
SJKZJVLL   EQU    5
*
SJKZJVA    EQU    40                ,JV AMOUNT'
SJKZJVAL   EQU    5
*
SJKZXHC    EQU    41                ,EXTENDED-HOST-CODE='
SJKZXHCL   EQU    9
*
SJKZDTR    EQU    42                ,DMS-TUNING-RESOURCES='
SJKZDTRN   EQU    1                NONE
```

```

SJKZDTRP EQU 2 PAGEABLE
SJKZDTRR EQU 3 RESIDENT
SJKZDTRL EQU 2
*
SJKZPAL EQU 44 , 'PHYSICAL-ALLOCATION=' :* V12.0
SJKZPAN EQU 1
SJKZPAA EQU 2
SJKZPALL EQU 2
*
SJKZHAS EQU 45 ,HIGH-AVAILABLE-SPACE=' :* V12.0
SJKZHASL EQU 5
*
SJKZS1U EQU 46 ,S1_LEVEL_USED' :* V12.0
SJKZS1UL EQU 5
*
SJKZS2U EQU 47 ,S2_LEVEL_USED' :* V12.0
SJKZS2UL EQU 5
*
SJKZHAU EQU 48 , 'HARDWARE-AUDIT' :* V13.0
SJKZHAA EQU 1
SJKZHAN EQU 2
SJKZHAUL EQU 2
*
SJKZLAU EQU 49 , 'LINKAGE-AUDIT' :* V13.0
SJKZLAA EQU 1
SJKZLAN EQU 2
SJKZLAUL EQU 2
*
*
*****
* 2. ID FOR THE ,ACCOUNT PART' OF THE ,JOIN RECORD'. *
* ----- *
* - WHEN MODIFYING AN EXISTING ACCOUNT NUMBER, THE DATA *
* ITEMS ARE PASSED INDIVIDUALLY. THIS MODIFICATION DATA *
* REFERS TO THE ACCOUNT NUMBER IN ,SJKZACT'! *
*****
*
* ACCOUNT-SPECIFIC PRIVILEGES
SJKZACT EQU 26 ,ACCOUNT='
SJKZACTL EQU 9
*
SJKZNTL EQU 27 ,NTL='
SJKZNTY EQU 1
SJKZNTN EQU 2
SJKZNTLL EQU 2
*
SJKZEXP EQU 28 ,EXPRESS='
SJKZEXY EQU 1

```

```
SJKZEXN    EQU    2
SJKZEXPL   EQU    2
*
SJKZINH    EQU    29          , INHD= '
SJKZINY    EQU    1
SJKZINN    EQU    2
SJKZINHL   EQU    2
*
SJKZTTY    EQU    30          , TTYPL= '
SJKZTSTD   EQU    1
SJKZTTP    EQU    2
SJKZTSYS   EQU    3
SJKZTTYL   EQU    2
*
*          ACCOUNT-SPECIFIC RESOURCES
SJKZTIM    EQU    31          , TIME= '
SJKZTIML   EQU    5
*
SJKZPRI    EQU    32          , PRIORITY= '
SJKZPRIL   EQU    2
*
SJKZCLA    EQU    33          , CLASS= '
SJKZCLAL   EQU    2
*
SJKZPOS    EQU    43          , POSIX= '          :* V11.2 POSIX
SJKZPOY    EQU    1
SJKZPON    EQU    2
SJKZPOSL   EQU    2
*
SJKZDEL    EQU    34          , DEL= '
SJKZDELL   EQU    81          (ALWAYS WHOLE AREA)
*
SJKZMAX    EQU    49          FOR QUERIES, MAXIMUM VALUE
*
SJKZEOF    EQU    255
```



```

*
SCBETYP2 DS      X          3RD BYTE OF THE TYPE
SCBELP3  EQU    X'01'      - PRINTER 136
SCBELP6  EQU    X'02'      - PRINTER 160
SCBETND  EQU    X'04'      - ND2
SCBETSD  EQU    X'08'      - SD23
SCBETCD  EQU    X'10'      - CARD PUNCHER
SCBETFD  EQU    X'20'      - FD
*
SCBEFSEQ DS      CL2       FSEQ OF FILE ON TAPE
SCBEIND  DS      X         INDICATOR BYTE
SCBEICF  EQU    X'01'      - CONTINUATION FILE
*
SCBETPRI DS      X         PRIORITY FROM ORIGINAL CMD
SCBETTSN DS      CL4       TSN FROM ORIGINAL CMD
SCBESQ#  DS      X         SEQUENCE # FROM ORIGINAL CMD
          DS      CL7       UNUSED
SCBEKLEN EQU    32         KEY LENGTH
*-----*
*                               LOGON TYPE INFORMATIONS                               *
*-----*
SCBEJNAM DS      CL8       JOB NAME (RJOB, LOGON OR DEFAULT)
SCBEUSR  DC      XL8'00'   ALPHANUMERIC USERID <LOGON
SCBEACT  DC      XL8'00'   ACCOUNT NUMBER <LOGON
SCBEPSWD DC      XL8'00'   PASSWORD < LOGON
SCBEPVG  DC      X'00'     PRIVILEGED USER
SCBEADM  EQU    X'01'     INDICATES SYSTEM ADMINISTRATOR
SCBEUSER EQU    X'02'     INDICATES NORMAL USER
SCBEOPR  EQU    X'04'     INDICATES OPERATOR
*
SCBESCLA DC      X'00'     SPOOLOUT CLASS DEFINITION
SCBESPCS DS      X         INITIAL SPACE VALUE OF PRINT CMD
SCBECTCS DS      X         INITIAL CONTROL CHAR VALUE OF PRINT
SCBECCST DS      X         CCS TYPE
SCBECCSV DS      X         CCS VARIANT
*-----*
*                               CATALOG, ACCOUNTING, ... INFORMATIONS                               *
*-----*
SCBEUTID DC      F'0'      USER TID                                     002
SCBES#IO DC      F'0'      #I/O DURING SPOOLIN
SCBEUID  DC      2F'0'     /ACCOUNT SUPPLIED INFORMATION
SCBECODE DC      H'00'     PRIMARY CATAL BLOCK NUMBER
SCBEDLOF DC      XL3'00'   DATE LOGOFF WAS PROCESSED
SCBETLOF DC      XL2'00'   TIME LOGOFF WAS PROCESSED
*-----*
*                               RBP INFORMATIONS                               *
*-----*
SCBEXBSE DC      X'00'     INDEX OF BATCH ENTRY STATIONS

```

```

SCBEXBJE DC      X'00'          INDEX OF BATCH JOB ENTRIES
*-----*
*                   GENERAL INFORMATIONS                   *
*-----*
SCBEMSG DC      X'00'          OPTION BYTE
SCBEFLMG EQU    X'01'          F = FULL MESSAGE OPTION
SCBECDMG EQU    X'02'          C = SHORT MESSAGE OPTION
SCBEHCMG EQU    X'04'          H = HARD COPY OPTION
SCBEHCT EQU     X'08'          T = TERMINAL'S HARD-COPY
SCBEDSTP EQU    X'10'          --- SWITCH FROM DISC TO TAPE
SCBEMSLO EQU    X'20'          L = LOGGING REQUESTED INDICATOR
SCBEBCPU EQU    X'80'          --- INSUFFICIENT CPU TIME LEFT OR
*                               --- BIAS TIME VALUE IS SET LOWER
*                               --- THEN TIME VALUE GIVEN OR
*                               --- DEFAULTTED
SCBESW DC       X'00'          JTBP SWITCH FIELD
SCBEPVT EQU     X'01'          SET: ENTER ERASE=YES GIVEN
SCBESCHD EQU    X'02'          SET: JOB IS SCHEDULED
SCBEDVQ EQU     X'04'          SET: TASK IS ON THE DEVICE QUEUE
SCBENTL EQU     X'08'          SET: NTL REQUESTED
SCBEENTR EQU    X'10'          JOB CREATED VIA ENTER
SCBERBP EQU     X'20'          JOB CREATED VIA REMOTE
SCBERCJ EQU     X'40'          JOB CREATED VIA RCARD
SCBENER EQU     X'80'          DO NOT ERASE FILE DURING LOGOFF
*                               ALSO, DO NOT SET UP TCB DURING
*                               TSN ASSIGNMENT, ONLY THE JTBP
SCBELRI DS      X              MAX PRI ALLOWED FOR AN ACCOUNT #
*
SCBESW2 DC      X'00'          SYSFILE SWITCH FIELD 2
SCBETEMF EQU    X'80'          TEMP FILE USED
SCBETEMJ EQU    X'40'          TEMP JV USED
SCBEEX94 EQU    X'20'          EXIT 94 LOADED
SCBEUFSF EQU    X'08'          UFS FILE
SCBEUFSC EQU    X'04'          UFS COPY CREATED
SCBERSRL EQU    X'02'          RESOURCE LOCATION : ON=SERVER
*                               OFF=CLIENT
SCBELIBM EQU    X'01'          LMS LIBRARY MEMBER
*
SCBEPARE DC     X'00'          SPARAM EDIT VALUE
SCBEPARN EQU    X'00'          EDIT= NO
SCBEPARP EQU    X'01'          EDIT= PARTIAL
SCBEPART EQU    X'02'          EDIT= TOTAL
*
SCBEMLN DC     CL1' '          MESSAGE LANGUAGE
SCBEMLNS DC    CL1' '          MESSAGE LANGUAGE STANDARD
SCBECNAM DS     CL8            JOBCLASS NAME
SCBEJPRI DS     X              JOB PRIORITY
SCBEHLDI DS     X              JOB HOLD INDICATOR

```

```

SCBEHLDY EQU X'80' - JOB IN HOLD
SCBECENT DS X CENTURY FOR THE DATE 036
SCBESW3 DS X INDICATOR 3 042
SCBEFTIP EQU X'01' - FT IN PROGRESS(NO DELETE AT
SCBESICK EQU X'02' - SINIX CLIENT KEEP 061
SCBEFILT EQU X'04' - JOB FILTERED 081
SCBEUSTT EQU X'08' - USER TRANSLATION TABLE 081
SCBESTPR EQU X'10' - START-PROC NE IMMEDIATE 082
SCBERDCM EQU X'20' - RDIR COMMAND FROM SYSFILE 082
*
CLIENT STARTUP
SCBEFPP# DS F TOTAL PPAGE# IN A FAMILY 005
SCBECLN EQU *-SCBEJTC
SCBETLP EQU X'40' - 3337 TYPE (SEE &P.TYP2)
SCBETHP EQU X'80' - HP TYPE (SEE &P.TYP2)
EJECT
SSVSCBE ,E
MFPRE MF=,PREFIX=SCB,MACID=E,ALIGN=F,DNAME=ESCB 010
SCBESCB DS OF
*****
* SPOOL SUBSYSTEM BLOCK *
*****
*
*
*****
* SOURCE INFORMATION *
*****
*
*
*-----*
* LOCAL AND REMOTE SPOOL COMMON FIELDS *
*-----*
SCBESCB# DS CL3 SCB VERSION NUMBER
SCBEV30A EQU X'100' START SCB VERSION NUMBER OF VERSION 3.0A
SCBESCBV EQU X'103' CURRENT SCB VERSION NUMBER, 95-06-07
SCBELST# DS X SYSLST NUMBER
SCBEOTSN DC F'0' TSN OF TASK REQUESTOR
SCBEFILN DC A(0) INTERNAL FILENAME OF FILE TO BE PRINTED
SCBESIZE DS F FILE SIZE (# OF PAM PAGES) 100
SCBERPSW DC XL4'00' FILE READ PASSWORD
SCBEFID DC CL54' ' FILE TO BE PRINTED(OR FILE# FOR EAM)
SCBEFTYP DC X'00' FILE TYPE TO BE OUTPUTTED
SCBESEAM EQU X'01' SYSTEM FILE(EAM)
SCBEOMF EQU X'02' OBJECT MODULE FILE(EAM)
SCBESOUT EQU X'04' SYSOUT
SCBESLST EQU X'08' SYSLST
SCBETEMP EQU X'10' TEMPORARY FILE SBC
SCBESOPT EQU X'20' SYSOPT SBC
SCBEUEAM EQU X'40' USER CREATED EAM FILE

```

SCBEFAM	EQU	X'80'	FIRST MEMBER OF A FAMILY	
*				
SCBEFFCB	DC	X'00'	FILE TYPE	
SCBEFSAM	EQU	X'00'	SAM FILE	
SCBEFTAP	EQU	X'01'	FILE RESIDES ON TAPE	101
SCBEFEAM	EQU	X'20'	EAM FILE (TO KEEP FOR COMPATIBILITY)	
SCBEFISA	EQU	X'40'	ISAM FILE	
SCBEFBTA	EQU	X'80'	BTAM FILE	
SCBEFPAM	EQU	X'C0'	PAM FILE	
*				
SCBERTYP	DC	X'00'	FILE RECORD FORMAT FROM CATAL	
SCBERVAR	EQU	X'02'	VARIABLE LENGTH RECORDS(V-TYPE)	
SCBERFIX	EQU	X'04'	FIXED LENGTH RECORDS(F-TYPE)	
SCBERUND	EQU	X'06'	UNDEFINED RECORDS	
*				
SCBEBITF	DS	X	BIT MAP INDICATOR	
SCBELOER	EQU	X'80'	LOCK ERROR	
SCBECKAV	EQU	X'40'	CHECKPOINT APA AVAILABLE	
SCBEAPPL	EQU	X'20'	ON : APPLICATION NOT ALLOWED	011
SCBEBITN	EQU	X'01'	DO NOT RESET BIT MAP	
*				
SCBERSIZ	DS	H	RECSIZE	
SCBEBSIZ	DS	H	BLOCKSIZE	
*				
SCBEMAIL	DS	CL64	MAILING ADDRESS	
*				
SCBEFOBS	DS	H	FOB SIZE.	
*				
	DS	0F		102
	DS	H	EAM TAPE FILENAME (TO KEEP FOR COMPAT.)	
SCBEOFIL	DS	X	ORIGINAL FILE SPECIFICATIONS	
SCBER164	EQU	X'80'	RECORD 164 PRESENT FOR LIB.ELEM.	
SCBEPEF1	EQU	X'40'	FCB TYPE OF PLAM ELEMENT	
SCBEPEF2	EQU	X'20'	IDEM	
SCBEASAC	EQU	X'10'	ASA-CODE SPECIFIED FOR FILE	
SCBEPGMO	EQU	X'08'	PAGE-MODE	
SCBEIAS6	EQU	X'04'	\$SYSSPOOL IMAGE APA FILE (600 DPI)	022
SCBEIAU6	EQU	X'02'	\$USER IMAGE APA FILE (600 DPI)	022
SCBEOEAM	EQU	X'01'	EAM FILE	
*				
SCBEIMFL	DS	X	IMAGE ND OR HP FILE USED	
SCBEIMNO	EQU	X'00'	NO INFO (OLD SCB)	
SCBEIMUS	EQU	X'01'	\$USER IMAGE HP FILE	
SCBEIMTS	EQU	X'02'	\$TSOS IMAGE HP FILE	
SCBEIDUS	EQU	X'04'	\$USER IMAGE ND FILE	
SCBEIDTS	EQU	X'08'	\$TSOS IMAGE ND FILE	
SCBEIAUS	EQU	X'10'	\$USER IMAGE APA FILE	
SCBEIASY	EQU	X'20'	\$SYSSPOOL IMAGE APA FILE	

SCBEIAU3	EQU	X'40'	\$USER IMAGE APA FILE (300 DPI)	022
SCBEIAS3	EQU	X'80'	\$SYSSPOOL IMAGE APA FILE (300 DPI)	022
*				
SCBE#VSN	DS	H	# OF VSN'S IN VSNLIST (FIELD SCBETPVL)	102
SCBEFSQ	DS	H	FSEQ FOR TAPE PROCESSING	102
SCBERTPD	DS	H	RETENTION PERIOD	102
SCBESTSN	DS	CL4	TSN REPLAY SAVE AREA	102
SCBETPTY	DS	X	TAPE TYPE	102
SCBETPR	EQU	X'01'	* REPLAY TAPE	102
SCBETPUS	EQU	X'02'	* USER TAPE (DEVIN=STAPE)	102
SCBERTPD	EQU	X'04'	* DMS TAPE (DEVIN=TAPE)	102
SCBETPP	EQU	X'08'	* POOLER TAPE	102
*				
SCBETFL	DS	X	TAPE FLAG	102
SCBETFLC	EQU	X'01'	* TAPE PROCESSED IN COPY MODE	102
SCBETFLD	EQU	X'02'	* TAPE PROCESSED IN DIRECT MODE	102
SCBETFLI	EQU	X'04'	* INPUT TAPE	102
SCBETFLO	EQU	X'08'	* OUTPUT TAPE	102
SCBETFLF	EQU	X'10'	STATE=FOREIGN	
SCBETFLM	EQU	X'80'	MIXED MODE	
*				
SCBETD	DS	X	TAPE DENSITY	102
SCBETY	EQU	X'B0'	* TAPE = YES	
SCBET9N	EQU	X'B1'	* TAPE = T9N	
SCBET9P	EQU	X'B2'	* TAPE = T9P	
SCBET9G	EQU	X'B4'	* TAPE = T9G	
*				
SCBEPBLK	DS	X	BLOCKSIZE FOR POOLER TAPE	102
SCBETPVL	DS	4CL6	VSN LIST FOR CURRENT FILE	
SCBETPLG	EQU	*-SCBETPVL	VSN LIST LENGTH	
SCBECHD1	DS	X	CHARS DENSITY FOR ND2 OFF LINE	
SCBELPI	EQU	X'C0'	LINES PER INCH	
SCBECPI	EQU	X'30'	CHARS PER INCH	
SCBE12L	EQU	X'80'	12 LPI	
SCBE8L	EQU	X'40'	8 LPI	
SCBE6L	EQU	X'00'	6 LPI	
SCBE15C	EQU	X'30'	15 CPI	
SCBE12C	EQU	X'20'	12 CPI	
SCBE10C	EQU	X'10'	10 CPI	
*				
SCBECHD2	DS	X	CHARS2 DENSITY	
SCBECHD3	DS	X	CHARS3 DENSITY	
SCBECHD4	DS	X	CHARS4 DENSITY	
SCBEFOBN	DS	CL4	FOB/FOBPPOOL NAME	
SCBEUPDI	DS	X	DPRINT INDICATOR	
SCBEFDIS	EQU	X'80'	- FILE CAN BE DISTRIBUTED	
SCBEPVCS	EQU	X'40'	- PVS SHARED BETWEEN CLIENT AND SERVER	
SCBEPVCG	EQU	X'20'	- PVS SHARED BETWEEN CLIENT AND GATEWAY	

```

SCBEBLKT EQU X'10' - BLOCKTYPE = STD
SCBEERAU EQU X'08' - ERASE OF FILE AUTHORIZED
SCBESTUP EQU X'04' - STARTUP
SCBELCK EQU X'02' - LOCK
SCBEDEL EQU X'01' - DELETE
*
SCBEFTST DS X FILE TRANSFER STATUS BYTE
SCBEFTNK EQU X'01' - FT ENDED NOT OK
SCBEFTOK EQU X'02' - FT ENDED OK
SCBEFTIN EQU X'04' - FT IN PROGRESS
SCBEFTRQ EQU X'08' - FT REQUESTED
SCBEFTWT EQU X'10' - FT WAITING
SCBEFTRE EQU X'20' - FT RECOVERY 074
*
SCBEJBTY DS X JOB TYPE
SCBEJBSP EQU X'80' - SPOOL JOB (LOCAL)
SCBEJBDP EQU X'04' - DPRINT JOB (LOCAL)
SCBECPCLEQU X'02' - JOB SENT TO ANOTHER HOST (CLIENT COPY)
SCBECPSV EQU X'01' - JOB COMING FROM ANOTHER HOST (SERVER C.)
*
SCBECLTT DS X CLUSTER TYPE
SCBECLHO EQU X'00' - LOCAL CLUSTER
SCBECLSX EQU X'01' - SINIX CLUSTER
SCBECLBS EQU X'02' - BS2000 CLUSTER
*
*****
* DESTINATION INFORMATION *
*****
*
*
*-----*
* LOCAL SPOOL SPECIFIC FIELDS *
*-----*
SCBEODEV DC H'0' OUTPUT DEVICE MNEMONIC
SCBEDENT DC X'00' OUTPUT DEVICE TYPE
SCBEPRT EQU X'20' PRINTER
SCBEPUN EQU X'30' PUNCH
SCBEPFD EQU X'90' FLOPPY DISK
SCBETAP EQU X'B0' TAPE
SCBEDESF DS X DESTINATION FLAG
SCBEDGRO EQU X'01' DEVICE GROUP
SCBERDIR EQU X'02' COMING FROM RDIR
SCBEDIST EQU X'04' JOB SCHEDULED ON A REMOTE SERVER (DP)
SCBEIDOM EQU X'08' JOB IDOM 078
SCBEBSDC EQU X'10' SBSDC CALLER 078
SCBELHST EQU X'20' LONG HOSTNAME SUPPORT 078
*-----*
* REMOTE SPOOL SPECIFIC FIELDS *

```

```

*-----*
SCBERUID DC    CL8'00'          REMOTE OUTPUT USERID          211
          ORG    SCBERUID
SCBENUID DS    CL8              NEW REMOTE OUTPUT USERID
*****
*      REQUIREMENT INFORMATION      *
*****
*
*-----*
*      PRINT COMMAND PARAMETERS--COMMON TO LOCAL AND REMOTE      *
*      (AND ALSO COMMON WITH PUNCH COMMAND FIELDS)                *
*-----*
SCBEPNAM DS    XL8              PRINT OR PUNCH JOB NAME
SCBECNTR DS    X                RECORD BYTE FOR CONTROL PHYSICAL
SCBEREAD DS    X                REMAIN ADVANCE FROM PCC
SCBESTRT DC    F'0'            STARTNO: BINARY LHE RECORD POSITION
SCBESALL EQU   X'00'           ZERO=BEGIN AT FIRST POSITION/BLOCK
SCBEEND DC    F'0'            ENDNO: BINARY RHE STOP RECORD POSITION
SCBEEALL EQU   X'00'           ZERO=END AT 132 OR 160 FOR PRINTER
SCBEFRM DC    XL6'40'          FORM SIZE CODE
SCBEFRMS EQU   C'STD'          INDICATE SYSTEM STANDART OR DEFAULT
SCBEHREC DC    H'0'            FREQUENCY OF HISTORICAL RECORDS
SCBEHPRT EQU   336             PRINTER DEFAULT IS 336 LINES
SCBEHPUN EQU   26             PUNCH DEFAULT IS 26 CARDS
*
SCBEER DC     X'00'            ERASE OPTION
SCBEERN EQU   X'00'            DO NOT ERASE FILE AFTER PROCESSING
SCBEERD EQU   X'FE'            DESTROY OPTION
SCBEERY EQU   X'FF'            ERASE FILE AFTER PROCESSING
*
SCBEERR DC    X'00'            READ ERROR OPTION
SCBEERT EQU   X'00'            TERMINATE UPON READ ERROR
SCBEERS EQU   X'40'            SKIP BAD BLOCK UPON READ ERROR
SCBEERA EQU   X'80'            OUTPUT BAD BLOCK AS IT IS
SCBEERF EQU   X'C0'            OUTPUT BAD BLOCK AND FLAG IT
*-----*
*      PRINT COMMAND PARAMETERS--COMMON TO LOCAL AND REMOTE      *
*      (BUT NO LONGER COMMON WITH PUNCH COMMAND FIELDS)          *
*-----*
SCBELINE DC    H'0'            NUMBER OF LINES MAXIMUM PER PAGE    210
SCBEPSIZ DC    Y(132)          PRINTER SIZE IN BYTES PER LINES
SCBEP132 EQU   132             INDICATES 132 CHARACTER PRINTER
SCBEP160 EQU   160             INDICATES 160 CHARACTER PRINTER
*
SCBEHDR DC     X'00'            HEADER INFORMATION ON EACH PAGE
SCBEHRDN EQU   X'00'            NOTHING IS PRINTED AS PART OF HEADER
SCBEHRP EQU   X'04'            PAGE NUMBER PART OF HEADER
SCBEHDRS EQU   X'10'            STANDARD HEADER-ID, PG, TSN, DAT, FNAME

```

SCBEHDRD	EQU	X'40'	DATE PART OF HEADER	
SCBEHVRT	EQU	X'80'	FIRST LOGICAL RCD=HEADER/PAGE	
*				
SCBEHEAD	DC	X'00'	CMD INFO	212
SCBEHNCS	EQU	X'00'		212
SCBELSPE	EQU	X'01'	LINE SPECIFIED IN PRINT REQUEST	210
SCBEHNSP	EQU	X'02'	JOB NOT REALLY SUSPENDED (FROM)	LPM
SCBESHSP	EQU	X'04'	SHIFT SPECIFIED IN PRINT REQUEST	212
*				
SCBESPRT	DC	X'00'	SEPARATOR PAGE TO BE OUTPUT	
SCBESYES	EQU	X'00'	YES	
SCBESNO	EQU	X'01'	NO	
*				
SCBETRLR	DC	X'00'	TRAILER PAGE NECESSARY	
SCBETYES	EQU	X'00'	YES	
SCBETNO	EQU	X'01'	NO	
*				
SCBEEDIT	DC	X'01'	EDIT CHARACTER OR SPACING	
SCBEEDTE	EQU	X'00'	SPACE=E(1ST RCD CHAR=PRNT CNTRL)	
SCBEEDT1	EQU	X'01'	LINES=1(SPACE 1 LINE BETWEEN RCDS)	
SCBEEDT2	EQU	X'02'	LINES=2(2)	
SCBEEDT3	EQU	X'03'	LINES=3(3)	
SCBEEDTN	EQU	X'04'	SPACE=NO (FOR RSO 2.0)	
*				
SCBEETYP	DC	X'00'	CONTROL CHAR TYPE	
SCBEEIBM	EQU	X'01'	CONTROL CHAR IS IBM	
SCBEEASA	EQU	X'02'	CONTROL CHAR IS ASA	
SCBEPALN	EQU	X'04'	PARAM LINE SPECIFIED	
SCBEETE	EQU	X'80'	SPACE=TE (FOR RSO 2.0)	
SCBEETI	EQU	X'81'	SPACE=TI "	
SCBEETA	EQU	X'82'	SPACE=TA "	
*				
SCBECSW1	DS	XL1	OUTPUT CONTROLLER SWITCH 1	213
SCBENOER	EQU	X'04'	S: NO ERASE ALLOWED	213
SCBENODL	EQU	X'02'	S: NO DEL. FROM SHUT. Q	213
*				
SCBECSW2	DS	XL1	OUTPUT CONTROLLER SWITCH 2	213
SCBEIOER	EQU	X'01'	S: I/O ERROR HAPPENED ON PRINTER	213
SCBEDMSE	EQU	X'04'	S: DMS ERROR HAPPENED ON INPUT FILE	213
SCBEUSRE	EQU	X'08'	S: USER ERROR HAPPENED	213
SCBERTDR	EQU	X'80'	S: REPLAY TAPE DIRECT	213
*				
SCBEEXIT	DC	X'00'	EXIT CALL REQUEST	210
SCBEEX95	EQU	X'02'	OUTPUT RECORD	210
SCBEEX90	EQU	X'01'	INPUT RECORD	210
SCBEBIN	DC	X'00'	BINARY OPTION	
SCBEXPBN	EQU	X'00'	GRAPHIC ONLY IS OUTPUTTED	
SCBEXPBY	EQU	X'02'	BOTH HEX AND GRAPHIC ARE OUTPUTTED	

```

*
SCBEPPFL DC    F'0'          NO OF PAM PAGES IN FILE
SCBEPPPR DC    F'0'          NO OF PAM PAGES PRINTED
SCBEFRNB DC    F'0'          VALUE OF FROM OR LAST PARAM
SCBETONB DC    F'0'          VALUE OF TO PARAM
SCBESEL  DC    X'00'         SELECTIVE PRINT OPTION FEATURE INDICATOR
SCBENSEL EQU   X'00'         NO SELECTIVE PRINT FEATURE
SCBESELA EQU   X'02'         LAST OPTION
SCBESELT EQU   X'04'         TO OPTION
SCBESELF EQU   X'08'         FROM OPTION
SCBESELL EQU   X'40'         TYPE=L
SCBESELP EQU   X'80'         TYPE=P
*
SCBEFCPY DC    X'00'         NUMBER OF FILE COPIES
SCBECOPY EQU   SCBEFCPY
SCBEPCPY DC    X'00'         NUMBER OF PAGE COPIES
*-----*
*          PRINT COMMAND PARAMETERS-ONLY MEANINGFULL FOR LOCAL          *
*-----*
SCBES2  DC    X'00'         SWITCH 2
SCBESD  EQU   X'01'         JOB MAY BE DIRECTED ON SD23
SCBEND  EQU   X'02'         JOB MAY BE DIRECTED ON ND2
SCBEMEX EQU   X'04'         MXM EXTENSION INDIRECTLY REQUESTED
SCBECVFB EQU   X'08'         CHECK VFB INDIRECTLY REQUESTED
SCBEMND EQU   X'10'         JOB MUST BE DIRECTED ON ND2
SCBENOLD EQU   X'20'         JOB MAY NOT GO ON OLD PRINTER
SCBE160 EQU   X'40'         JOB MUST BE REDIRECTED ON 160          CLEAN
SCBE37  EQU   X'80'         MAY ON 3337
*
SCBELOOP DS    XL3          LOOP PARAMETER          102
SCBEFT1 DS    XL3          FONT 1 KEY          102
SCBEFT2 DS    XL3          FONT 2 KEY          102
SCBEFT3 DS    XL3          FONT 3 KEY          102
SCBEFT4 DS    XL3          FONT 4 KEY          102
SCBEDIA DC    XL2'00'       DIA IDENTIFICATION
SCBEIMAG DC    XL4'00'       IMAGE IDENTIFICATION-ND2 PREFIX
SCBESHIF DC    X'00'         SHIFT COLUMN DELAY
SCBECC  DC    X'00'         CONTROL PARAM
SCBECCN EQU   X'01'         CONTROL = NO
SCBECCP EQU   X'02'         = PHYSICAL
SCBEC12 EQU   X'08'         = 12LPI
SCBECCL EQU   X'04'         = LOGICAL
SCBELNMD EQU   X'10'         = LINEMODE          210
SCBECCTR EQU   X'20'         = TRANSPARENT
SCBECAPA EQU   X'40'         = APA
*-----*
*          SQ COMMAND AND ABNORMAL I/O MSG PARAMETERS          *
*-----*

```

SCBEAND	DC	X'00'	ACTION INDICATOR	102
SCBEANDR	EQU	X'01'	RESPOOL SPECIFIED	102
SCBEANDS	EQU	X'02'	SUSPEND SPECIFIED	102
SCBEANDH	EQU	X'04'	HOLD "H" SPECIFIED	102
SCBEANDK	EQU	X'08'	HOLD "K" SPECIFIED	102
SCBEIMEX	EQU	X'10'	MPVS IMPORT-EXPORT	MPVS
*				
SCBESUSI	DC	X'00'	SUSPEND INDICATOR	
SCBESUSL	EQU	X'01'	< SPECIFIED	
SCBESUSP	EQU	X'02'	P SPECIFIED	
SCBESUSB	EQU	X'04'	B SPECIFIED	
SCBESUSS	EQU	X'08'	S SPECIFIED	
SCBESUSM	EQU	X'10'	PAGE # SPECIFIED	
SCBESUSY	EQU	X'20'	PRIORITY SPECIFIED	102
SCBECRLG	EQU	X'40'	- DEFAULT FORM FROM GEN REC	070
*				
SCBELPHC	DS	CL3	REQUESTED CROSS LOOP FOR HP (ROTAT.)	
SCBESUSG	DC	F'0'	PAGE # SPECIFIED IN THE CMD	
SCBESUSV	DC	X'00'	PRIORITY SPECIFIED IN THE CMD	102
SCBESEC#	DS	CL2	SECOND # AT PRINT START	

*			SPECIAL PARAMETERS FOR PUNCH FLOPPY DISC	*

SCBEFDNM	DS	CL8	FILENAME SPECIFIED	
SCBECH#P	DS	X	ORDER# OF CHARS REQUESTED IN POOL (HP)	
SCBEFDSC	DC	H'0'	NUMBER OF RESERVED SECTORS	
SCBEFDRP	DC	H'0'	RETENTION PERIOD	
SCBERCZ	DS	H	RECSIZE FROM PUNCH CMD	
SCBEFDCB	DC	X'00'	CONTROL BYTE	
SCBEFDBP	EQU	X'01'	BYPASS=Y	
SCBEFDAC	EQU	X'02'	ACCESS=N	
SCBEFDWP	EQU	X'04'	WRITEPR=Y	
SCBEFDSU	EQU	X'08'	SKEL=US	
SCBEFDSN	EQU	X'10'	SKEL=NS	
SCBEFLST	EQU	X'20'	REPORT REQUESTED	CLEAN
*				
SCBEFDB2	DC	X'00'	CONTROL BYTE 2	102
SCBEFDNO	EQU	X'01'	OWNERID = NO	102
*				
SCBEFDB3	DC	X'00'	CONTROL BYTE 3	103
SCBEFDSA	EQU	X'01'	JOB COPIED SHUTDOWN->ACTIVE Q	103
*				
SCBESAER	DS	CL1	SAVE ERASE OPTION	103

*			VSN LIST USED BY FLOPPY DISCS	*

SCBEFDVS	DS	10CL6	VSN LIST FOR FLOPPY DISC	
SCBEFDLG	EQU	*-SCBEFDVS	VSN LIST LENGTH	

```

*
*****
*      CONTROL-STATUS INFORMATION      *
*****
*
*-----*
*                                *
*                                *
*-----*
SCBELHI  DS      X      LOOP LENGTH IN 1/6 INCH
SCBELHIC DS      X      CROSS LOOP LENGTH IN 1/6 INCH
SCBEROE  DS      A      @ OF ROE                                102
SCBECOUN DC     F'0'    SAVED LINES/CARDS COUNT(AFTER /SQ)
SCBEIO   DC     F'0'    SAVED I/O COUNT ON PUB DEVICE(AFTER /SQ)
SCBECNT  DC     F'0'    LINES/CARDS # TO PRINT/PUNCH
SCBEDSPO DC    XL3'00'  DATE JOB SPOOLED OUT
*                                UNUSED (SCBETSPO) FOR COMPATIBIL.
SCBECTTY DS      X      CONTROL TYPE
SCBECTCP EQU    X'01'    = COMPATIBLE
SCBECTHP EQU    X'02'    = HP
SCBECEDA DS      X      CENTURY FOR DATE JOB SPOOLED OUT END
SCBEDSPF DC    XL3'00'  DATE JOB SPOOLED OUT FINISH
*                                UNUSED (SCBETSPF) FOR COMPATIBIL.  211
SCBEPTY  DS      0XL2   PRINTER TYPE
SCBEPTY1 DS      X      FIRST BYTE
SCBEPTY2 DS      X      SECOND BYTE
SCBEPT65 EQU    X'08'    = LP65
SCBEPTHP EQU    X'04'    = HP
SCBEPTAP EQU    X'02'    = APA
SCBEPTAN EQU    X'01'    = ANY
*
SCBEEPDC DS      X      ERROR PAGE CODE                                102
SCBEDMSC EQU    X'01'    DMS ERROR CODE
SCBERTPR EQU    X'02'    ERROR PAGE INDICATOR
SCBESPND EQU    X'04'    SUSPENDED
SCBEHLD  EQU    X'08'    HOLDED
SCBECNCL EQU    X'10'    CANCELLED
SCBEWCB  EQU    X'20'    WRONG CONTROL BYTE
SCBELTR  EQU    X'40'    LINE TRUNCATION OCCURED
SCBETMSG EQU    X'80'    SCBEEPMS IS A PRINTABLE MSG
*
SCBEEPMS DS     CL40    ERROR PAGE MESSAGE                                102
SCBEDSCD DS     CL1     DELAY SCHEDULING BYTE
SCBENSCD EQU    X'FF'   DELAY SCHEDULING IN EFFECT
SCBEFMND DS      X      FAMILY INDICATOR
SCBEFFM  EQU    X'01'   -FIRST MEMBER OF FAMILY
SCBEFMNQ EQU    X'02'   -ALL FAMILY NOT YET IN Q
SCBEFBIT EQU    X'04'   -DO NOT RESET BIT MAP

```

```

*-----*
*                LOCAL SPOOL SPECIFIC FIELDS                *
*-----*
SCBEROTV DS      X          ROTATION VALUES
SCBER00 EQU     X'F0'      0    DEGREE
SCBER90 EQU     X'F1'      90   DEGREES
SCBER180 EQU    X'F2'     180   DEGREES
SCBER270 EQU    X'F3'     270   DEGREES
SCBER018 EQU    X'F4'     (0,180) DEGREES
SCBER927 EQU    X'F5'     (90,270) DEGREES
SCBE1800 EQU    X'F6'     (180,0) DEGREES
SCBER279 EQU    X'F7'     (270,90) DEGREES
SCBERYES EQU    C'Y'      ROTATION=YES
SCBERNO EQU    X'00'      ROTATION=NO
*
SCBESAMP DC     H'0'      SAMPLE=VALUE USED FOR FORMS ALIGN.
SCBENSAM EQU    X'00'      ZERO=NO FORMS ALIGNMENT
SCBEXPND DC     X'00'      SQ/CANCEL INDICATOR
SCBEXINT EQU    X'08'      OTHER RESPOOL/SUSPEND NOT ALLOWED
SCBEXPC EQU     X'10'      CANCEL SPOOL-OUT W/NO SHUTDOWN ENTRY
SCBEXPCH EQU    X'20'      SPOOL SHUTDOWN IN PROGRESS          210
*
SCBECPL DC     X'00'      SPOOLOUT COMPLETION SWITCH
SCBELOK EQU    X'00'      NORMAL COMPLETION
SCBELCAN EQU    X'01'      SPOOLOUT TASK CANCELLED
SCBELER1 EQU    X'02'      ABNORMAL OUTPUT I/O TERMINATED TASK
SCBELER2 EQU    X'04'      ABNORMAL INPUT I/O TERMINATED TASK
SCBELRES EQU    X'08'      TASK WAS STOPPED AND RESPOOLED
*
SCBEDIRE DS     H          DISPLACEMENT IN RECORD (MLR)
SCBEHRC DS      0F        SKIP TYPE HISTORICAL RECORD
SCBEDISP DC     H'0'      DISPLACEMENT WITHIN I/P RECORD
SCBEPOS DC     H'0'      # OF GET LOOPS ON I/P RECORD
SCBESEL DC     F'0'      SEL PRINT CNT(#LINES)PAGES CORRESPONDING
*                          TO THE I/O COUNT FOR THE SUSPEND FEATURE)
SCBESAVP DC     F'0'      SAVED CURRENT PAGE# FOR HEADER(SQ)
SCBETFLN DC     XL12'00'  NEW FILE NAME GIVEN TO THE FILE WHEN CATA-
*                          LOGING FROM TAPE TO DISC
*-----*
*                ND2 SPECIFIC FIELDS                *
*-----*
SCBEREC DC     F'0'      RECORD # (HIST. RCD)
SCBEUF DC      F'0'      BUFFER # (HIST. RCD)
SCBELNA DC     F'0'      LINE ADDR (HIST. RCD)
SCBEVFBL DC    X'00'      # OF LINES FROM THE VFB
SCBECH1 DS     X          FONT# & DENSITY OF 1ST SET 12LPI
SCBECH2 DS     X          FONT# & DENSITY OF 2ND SET 12LPI
SCBEDEFV DC    X'00'      DEFAULT VALUES GIVEN BY THE SPOOL

```

```

SCBECHAR EQU X'80'      CHARS
SCBELPRO EQU X'40'      LOOP
SCBECLPD EQU X'20'      CROSS LOOP
*
SCBEFNAM DS CL10        REPLAY FILE MNEMONIC
SCBEIDEV DS H           INPUT DEVICE MNEMONIC
SCBEVBSD DC X'00'      # OF LINES FROM VFB FOR SD23
SCBEVBLP DC X'00'      # OF LINES FROM VFB FOR 3337
SCBEVBND DC X'00'      # OF LINES FROM VFB FOR ND2
SCBELCH1 DC X'00'      NUMBER OF LINES BEFORE CHANEL 1
SCBEITN DC F'0'        ITN (USABLE BY STATUS ONLY ! )
*-----*
*                REMOTE CONTROL-TO BE UPDATED IN 1.2                *
*-----*
SCBEPAGE DC F'0'        CURRENT PAGE# FOR HEADER
SCBECHPN DS CL4         CHARS POOL NAME
SCBECRLO DS CL3         CROSS LOOP PARAMETER
SCBEFDNA DS C           FD NAME FROM DEVICE PARAMETER
SCBE3171 EQU X'00'      DEVICE=FD3171 (0 FOR COMPATIBILITY)
SCBEFACO EQU X'9B'      FD75407 (FALCON) REQUESTED
SCBEHPID DS CL4         USED HPFILE CODED FILE ID.
SCBECHAN DS H           CHARS NUMBER IN POOL
SCBESTID DS X           STREAM INTERNAL ID (SLOT SCHEDULING)
SCBELOID EQU X'00'      STREAM FOR LOCAL SCHEDULING
SCBERSID EQU X'01'      STREAM FOR RSO SCHEDULING
SCBESPRI DS X           PRIORITY BEFORE SCHEDULING
*-----*
*                TEMPORARY FOR TAPES                *
*-----*
SCBEFLTP DS X           FLAG BYTE FOR TAPES ****                102
SCBEUSTP EQU X'11'      * USER TAPE
SCBERLTP EQU X'10'      * REPLAY TAPE
SCBERRCD DS CL8         RSO PRINTER ERROR CODE
SCBEDFES DS X           TO SAVE DFER (RBP CONTROL SWITCH).
*
SCBEDSTN DS CL8         DESTINATION STATION                JSI & 211
SCBEVBHC DS H           # OF LINES FROM CROSS VFB FOR HP
*-----*
*                REMOTE CONTROL SPOOL V2                *
*-----*
SCBEINF DS 0F           HISTORICAL RECORD                211
SCBEDMEC DS CL8         DCAM ERROR CODE (..SPOL:..DMEC)      211
SCBERJBN DS CL8         REMOTE JOB NAME                    211
SCBERINF DS X           REMOTE INFORMATION                211
SCBETERM EQU X'01'      SET: JOB ABNORMALLY TERMINATED      211
SCBERPRT EQU X'02'      SET: PRINT TO THE REMOTE            211
*                      RESET: LOGOFF PROCESSING            211
SCBEGRPC EQU X'08'      SET: GROUP IN CONFIGURATION

```

SCBERSO	EQU	X'10'	SET: RSO DESTINATION	211
SCBEDFER	DC	X'00'	CONTROL SWITCH	211
SCBEDRMT	EQU	X'01'	SET: REMOTE DESTINATION	211
*			RESET: CENTRAL DESTINATION	211
SCBEDFRR	EQU	X'02'	SET: DEFERRED OUTPUT	211
*			RESET: IMMEDIATE OUTPUT	211
SCBERESV	EQU	X'04'	SET: RESERVED OUTPUT	211
*			RESET: NOT RESERVED OUTPUT	211
SCBESUBM	EQU	X'08'	SET: RESERVED BY SUBMITTER	211
*			RESET: RESERVED BY ALTERNATE	211
-----*				
*			NEW FIELDS FOR SPOOL V2	*
-----*				
SCBECH1C	DS	H	# OF LINES BEF. CHANNEL1, CROSS HP	
SCBETRLN	DC	F'0'	TRAILER NUMBER	211
SCBEHDRN	DC	F'0'	HEADER NUMBER	211
SCBEUSTX	DC	CL32'	USER TEXT IN HEADER-TRAILER	213
SCBELIND	DS	H	TOTAL LINES/PAGE FOR ND2	
SCBELISD	DS	H	TOTAL LINES/PAGE FOR SD23	
SCBELILP	DS	H	TOTAL LINES/PAGE FOR LP3337	
SCBELIHP	DS	H	TOTAL LINES/PAGE FOR HP (3351, 3353)	
SCBEVBHP	DS	H	# OF LINES FROM VFB FOR HP	
SCBES2B	DS	X	SWITCH 2 BIS	
SCBEHP	EQU	X'01'	JOB MAY BE DIRECTED ON HP	
SCBEMHP	EQU	X'02'	JOB MUST BE DIRECTED ON HP	
SCBECVFH	EQU	X'04'	CHECK VFB REQUESTED FOR HP	
SCBENDX	EQU	X'08'	JOB MAY BE DIRECTED ON HP54 (=NDX)	
SCBEMNDX	EQU	X'10'	JOB MUST BE DIRECTED ON HP54	
SCBECVFX	EQU	X'20'	CHECK VFB REQUESTED FOR HP54	
SCBEMUNW	EQU	X'40'	MUST NEW, FROM SPOOL 2.5	
*				
SCBERDT	DS	X	REQUESTED DEVICE TYPE	
SCBEDTAL	EQU	X'00'	*ALL	
SCBEDTND	EQU	X'01'	ND PRINTER	
SCBEDTHP	EQU	X'02'	HP PRINTER	
*				
SCBEHPFL	DS	X	HP FLAG	
SCBEFOBP	EQU	X'01'	FOB POOL USED	
SCBECHPO	EQU	X'02'	CHARS POOL USED	
SCBEROTF	EQU	X'04'	ROTATION SPECIFIED IN REQUEST	
SCBEH2SP	EQU	X'08'	HP TO SPS JOB	074
*				
SCBELPHP	DS	CL3	REQUESTED LOOP FOR HP	
SCBELPND	DS	CL3	REQUESTED LOOP FOR ND2	
SCBELPSD	DS	CL3	REQUESTED LOOP FOR SD23	
SCBELPLP	DS	CL3	REQUESTED LOOP FOR 3337	
SCBEFDTY	DS	X	FDTYPE PARAM	
SCBEFSRA	EQU	X'00'	SCRATCH	

```

SCBE3170 EQU X'40'      FD3170
SCBE1S1D EQU X'F1'      FD1S1D (SINGLE SIDE, SINGLE DENSITY)
SCBE11BD EQU X'40'      FD11BD (SINGLE SIDE, SINGLE DENSITY)
SCBE2S1D EQU X'F2'      FD2S1D (DOUBLE SIDE, SINGLE DENSITY)
SCBE2S2D EQU X'D4'      FD2S2D (DOUBLE SIDE, DOUBLE DENSITY)
*
SCBECH1H DS H          # OF LINES BEFORE CHANEL 1 FOR HP
SCBEVFBH DS H          SCBEVFB FOR HP
SCBEC#DH DS CL1       LPI AND CPI (CPI - 6) FOR HP FROM
*                       1RST CMD OR POOL CHARS OR DEFAULT CH.
SCBELIRD EQU X'80'     RESERVED FOR CHARS# FROM FIRST CHARS
SCBELPI6 EQU X'10'     6 LPI FOR HP
SCBELPI8 EQU X'20'     8 LPI FOR HP
SCBELP10 EQU X'30'    10 LPI FOR HP
SCBELP12 EQU X'40'    12 LPI FOR HP
SCBELP24 EQU X'50'    24 LPI FOR HP
SCBEHCPI EQU X'0F'    CPI VALUE IN 4 LAST BITS
*
SCBEHPOS DS CL4       $TSOS HPFILE CODED FILE ID
SCBEFIFL DS X         FILE FLAG
SCBEFIPL EQU 1        PLAM LIB
SCBEPRER EQU 2        SECTION PREPROCESSING REQUESTED.
SCBEPREI EQU 4        SECTION IN PREPROCESS
SCBEPRES EQU 8        SECTION PREPROCESSED WITH SUCCESS
SCBEPREF EQU 16       PREPROCESSING FAILED WITH DMS ERROR
*                       (WHICH IS IN &P.SDMS).
SCBEPREL EQU 32       PREPROCESSING LOGICAL FAILURE
SCBECOFI EQU 64       FILE COPIES IS REQUESTED
SCBECOPA EQU 128      PAGE COPIES IS REQUESTED
*
SCBEPLRF DS X        PLAM RECORD TYPE FROM
SCBEPLRT DS X        PLAM RECORD TYPE TO
SCBEERP DS X         ERROR PROCESSING=
SCBETRIG EQU 1        TRUNC=IGNORE | RESET =
SCBETRKE EQU 2        TRUNC=KEEP   |         STD
SCBEMGPY EQU 4        MSGPAGE=YES  | RESET =
SCBEMGPN EQU 8        MSGPAGE=NO   |         STD
*
SCBELP48 DS CL3      REQUESTED LOOP FOR LP48
SCBELI48 DS H        TOTAL LINES/PAGE FOR LP48
SCBEVB48 DS H        # OF LINES FROM VFB FOR LP48
SCBELI65 DS H        TOTAL LINES/PAGE FOR LP65 (=ND50)
SCBEVB65 DS H        # OF LINES FROM VFB FOR LP65
SCBELIUL DS H        TOTAL LINES/PAGE FOR EMULATED
SCBEVBUL DS H        # OF LINES FROM VFB FOR EMULATED
SCBELP65 DS CL3     REQUESTED LOOP FOR LP65
SCBELPUL DS CL3     REQUESTED LOOP FOR EMULATED PRINTER
SCBEDSEM DS H       DISPLACEMENT TO DSEM PRINT PARAMETER

```

```

SCBEFACR DS      X          NUMBER OF FILES IN A FAMILY(1RST MEM.)
SCBESEFL DS      X          SECTION FLAG
SCBESERE EQU     1          SECTION REQUESTED IN THE COMMAND
SCBESEBF EQU     2          FROM BEGIN OF FILE
SCBESEFY EQU     4          FROM *ANY
SCBESEEF EQU     8          TO END OF FILE
SCBESETY EQU    16          TO *ANY
SCBESCTY EQU    32          SET = SECT  IN CHR, RESET = HEXA
SCBEFITY EQU    64          SET = FIRST IN CHR, RESET = HEXA
SCBELATY EQU   128          SET = LAST  IN CHR, RESET = HEXA
*-----*
*                                CHARS FOR RSO                                *
*-----*
SCBEFT5  DS      XL3          FONT 5 KEY (SRC SEQ # MODIF)
SCBEFT6  DS      XL3          FONT 6 KEY
SCBEFT7  DS      XL3          FONT 7 KEY                                211
SCBEFT8  DS      XL3          FONT 8 KEY                                211
SCBEFT9  DS      XL3          FONT 9 KEY                                211
SCBEFT10 DS      XL3          FONT 10 KEY                               211
SCBEFT11 DS      XL3          FONT 11 KEY                               211
SCBEFT12 DS      XL3          FONT 12 KEY                               211
SCBEFT13 DS      XL3          FONT 13 KEY                               211
SCBEFT14 DS      XL3          FONT 14 KEY                               211
SCBEFT15 DS      XL3          FONT 15 KEY                               211
SCBEFT16 DS      XL3          FONT 16 KEY                               211
SCBEFTTL EQU    *-SCBEFT5+12    TOTAL FONTS LENGTH                    211
SCBEFT#  EQU    SCBEFTTL/L'SCBEFT1  FONTS NUMBER                      211
SCBEC#D1 DS      CL(SCBEFT#)  DENSITIES CORR. TO (16) FONTS          211
*-----*
*                                GENERAL OUTPUT INFORMATIONS                    *
*-----*
SCBETSPO DC      XL3'000'      TIME JOB SPOOLED OUT HHMMSS            211
SCBETSPP DC      XL3'000'      TIME JOB SPOOLED OUT FINISH HHMMSS    211
SCBELIHC DS      H            TOTAL LINES/CROSS PAGE FOR HP
SCBELPP# DS      F            LAST RETRIEVAL @: BLOCK#(BBBBBBFF)      SBC
SCBE#PP  DS      F            FILE SIZE IN PRINTER PAGES              SBC
SCBECH1# DS      F            COUNTER OF SKIP TO CHANNEL 1            SBC
SCBESBC# DS      H            PRINT SEQUENCE #                        SBC
*-----*
*                                RSO 2.0 SPECIFIC FIELDS                        *
*-----*
SCBETRA# DS      X            INPUT TRAY # FOR RSO & NDY
SCBEITMA EQU     X'80'        = MANUAL                                011
SCBEITA3 EQU     X'81'        = A3                                    011
SCBEITA4 EQU     X'82'        = A4                                    011
SCBEITA5 EQU     X'83'        = A5                                    011
SCBEITB4 EQU     X'84'        = B4                                    011
SCBEITB5 EQU     X'85'        = B5                                    011

```

SCBEITDO EQU	X'86'	= DOUBLE-LETTER	011
SCBEITEX EQU	X'87'	= EXEC	011
SCBEITFO EQU	X'88'	= FOLIO	011
SCBEITIV EQU	X'89'	= INVOICE	011
SCBEITLG EQU	X'8A'	= LEGAL	011
SCBEITLT EQU	X'8B'	= LETTER	011
SCBEITMO EQU	X'8C'	= MONARCH	011
SCBEIT10 EQU	X'8D'	= COM10	011
SCBEITDL EQU	X'8E'	= DL	011
SCBEITC5 EQU	X'8F'	= C5	011
SCBEITIG EQU	X'FF'	= IGNORE	011
*			
SCBESQSA DS	X	PREVIOUS "SQ" KIND SAVED	
SCBEXSEC DS	F	TDFILE SECTION NUMBER	
SCBESHRO DS	D	RSO SCHEDULING	
SCBETRAT DS	CL8	TRANSLATION TABLE NAME	
SCBERDST DS	CL8	RSO AND LOCAL DESTINATION GROUP	
SCBEPINF DS	X	RSO PRINT INFO	
SCBECHMO EQU	X'01'	CHARS MODIFICATION	
SCBERSMP EQU	X'02'	SET: SAMPLE=YES GIVEN	
SCBERDI EQU	X'04'	SET: RDIR=NO	
*			
SCBEXTYP DS	X	MODE TYPE	
SCBEXRSO EQU	X'00'	XTYPE=RSO (DEFAULT)	
SCBEXRTD EQU	X'01'	XTYPE=TDSPPOOL	
*			
SCBESPC2 DS	X	SPACE OPERAND FOR RSO 2.0	
SCBESPE EQU	X'00'	SPACE=E	
SCBESP1 EQU	X'01'	SPACE=1	
SCBESP2 EQU	X'02'	SPACE=2	
SCBESP3 EQU	X'03'	SPACE=3	
SCBESPN EQU	X'04'	SPACE=NO	
*			
SCBETYP3 DS	X	SPACE OPERAND FOR RSO 2.0	
SCBESPI EQU	X'01'	SPACE=IBM	
SCBESPA EQU	X'02'	SPACE=ASA	
SCBELN2 EQU	X'04'	LINES=SPECIFIED	
SCBESPTI EQU	X'80'	SPACE=TE	
SCBESPTI EQU	X'81'	SPACE=TI	
SCBESPTA EQU	X'82'	SPACE=TA	
*			
SCBECTL2 DS	X	CONTROL OPERAND FOR RSO 2.0	
SCBECTNO EQU	X'01'	CONTROL=NO	
SCBECTPH EQU	X'02'	CONTROL=PHYSICAL	
SCBECTLG EQU	X'04'	CONTROL=LOGICAL	
SCBECTLN EQU	X'10'	CONTROL=LINEMODE	
SCBECTTR EQU	X'20'	CONTROL=TRANSPARENT	
*			

```

SCBETTIM DS    CL4          IMAGE FOR TRANSLATION TABLE
SCBEEEXHC DS   CL8          EXTENDED HOST CODE
SCBESRMD DS    X            SORT MODE                                011
SCBESMNO EQU   X'01'        = NO                                011
SCBESMGR EQU   X'02'        = GROUP                                011
SCBESMCO EQU   X'03'        = COLLATE                            011
SCBESMST EQU   X'04'        = STACKER                            011
SCBESMAU EQU   X'05'        = AUTOMATIC                          011
*
SCBERPPD DS    H            RSO PRINT PARAM. DISPLACEMENT
SCBEESTS DS    F            ESTIMATE SIZE FOR STATUS
SCBETOOS DS    H            TOP OFFSET                                011
SCBELEOS DS    H            LEFT OFFSET                             011
SCBEMTSN DS    CL4          OLD TSN DURING /MOVE-PR-JOB COMMAND    077
SCBEMOVE DS    X            INDICATOR FOR /MOVE-PR-JOB COMMAND    077
SCBENEW EQU    X'80'        NEW TSN HAS BEEN ASSIGNED DURING MOVE 077
SCBEMVPJ EQU   X'40'        MOVE-PR-JOB COMMAND GIVEN            077
SCBEMPJA EQU   X'20'        MODIFY-PR-JOB-ATTR COMMAND GIVEN     077
SCBENSCH EQU   X'10'        JOB IS NOT SCHEDULABLE                077
          DS    CL1          RESERVED FOR RSO 2.0
*-----*
*                GENERAL INFORMATIONS                               *
*-----*
SCBECPUT DS    XL8          CPU TIME USED BY SPOOL
SCBEUTAP DS    XL8          USER PARAM TAPE=
          ORG   SCBEUTAP
SCBENFNM DS    CL54          NEW SAM FILENAME
SCBEFUT DS    0F            UNUSED IN VER = 102                    102
          ORG   SCBETOP+X'3F8'                                     102
*-----*
* ATTENTION:  THE REAL LENGTH OF THE SCB IS IN &P.TOP. !!!!!!! *
*-----*
SCBELIEX DS    CL1024       MAXIMUM EXTENSION LENGTH
          ORG   SCBELIEX     FIXED PART OF THE EXTENSION
SCBEPLRA DS    F            MAXIMUM RECSIZE FOR SATZART A
SCBEPLRB DS    F            MAXIMUM RECSIZE FOR SATZART B
SCBEPLD DS    H            DISPLACEMENT FOR PLAM ELEMENT NAME
*                FROM THE BEGIN OF THE COMPLETE SCB.
SCBEPLVD DS    H            IDEM FOR PLAM ELEMENT VERSION.
SCBEPLTD DS    H            IDEM FOR PLAM ELEMENT TYPE
SCBEJVDI DS    H            IDEM FOR JOB VARIABLE
SCBEJPWD DS    CL4          USER JOB VARIABLE PASSWORD
SCBECPY# DS    H            NUMBER OF PRINT EXEMPLARS
SCBEPARM DS    H            DISPL. TO PRINT PARAMETERS
*
SCBEJVEX DS    X            JOB VARIABLE INDICATOR
SCBEJVXY EQU   1            JV EXISTS
SCBEJVXN EQU   2            JV INACCESSIBLE

```

SCBEJUWP	EQU	4	USER PASSWORD TO RESTORE	
SCBEJVCR	EQU	8	JV CREATED BY SPOOL	
*				
SCBESDMS	DS	X	SECTION DMS ERROR CODE INDICATOR	
SCBESDEF	EQU	1	EMPTY FILE	
SCBESDOP	EQU	2	DMS CODE IN SCBEPERR	
SCBESDSE	EQU	4	SETL ERROR	
*				
SCBEPERR	DS	CL4	DMS ERROR CODE OF PREPROCESSING	
SCBEDIAR	DS	X	DIA RECTO FOR LP65 (=ND50)	
SCBEDIAV	DS	X	DIA VERSO FOR LP65	
SCBESIDD	DS	H	DISPLACEMENT FOR SECTION ID	
SCBESEPO	DS	H	SECTION ID POSITION	
SCBESF#	DS	F	SECTION FROM NUMBER	
SCBESFDI	DS	H	DISPLACEMENT OF FROM SECTION ID	
SCBESFPO	DS	H	SECTION FROM POSITION	
SCBEST#	DS	F	SECTION TO NUMBER	
SCBESTDI	DS	H	DISPLACEMENT OF TO SECTION ID	
SCBESTPO	DS	H	SECTION TO POSITION	
SCBESLB#	DS	F	SETL BLOCK NUMBER	
SCBESLR#	DS	F	SETL RECORD NUMBER	
SCBER#BF	DS	F	RECORD NUMBER FROM BEGIN OF FILE	
SCBEFSCR	DS	F	FROM (FIRST) STRING COUNTER	
SCBETSCR	DS	F	TO (LAST) STRING COUNTER	
SCBESSCR	DS	F	SECTION STRING COUNTER	
SCBEISKY	DS	H	ISAM KEY DISPLACEMENT FOR SECTIONS	
SCBEIMDI	DS	H	DISPLACEMENT FOR IMAGE FILENAME	
*				
SCBECKPT	DS	X	CHEKPOINT TO DO	
SCBECKPP	EQU	0	AT PAGE	
SCBECKPS	EQU	1	AT SECTION	
*				
SCBEDPLX	DS	X	DUPLEX =	
SCBEDUPD	EQU	0	STD	
SCBEDUPY	EQU	1	YES	
SCBEDUPT	EQU	2	TUMBLE	
SCBEDUPN	EQU	4	NO	
*				
SCBEPACC	DS	X	PAGE CONTROL CHARACTER (TOP OF PAGE)	
SCBEPACS	EQU	0	STD	
SCBEPACN	EQU	1	NO	
*				
SCBEOTRA	DS	X	OUTPUT TRAY # LP65 (=ND50,NDY)	
SCBEOTIG	EQU	X'80'	OUT-TRAY = NO (RSO)	011
SCBEOTSO	EQU	X'81'	OUT-TRAY = SORTER (RSO)	011
*				
SCBENPAD	DS	F	PAGEDEF VALUE FOR LP65	
SCBEPCCD	DS	F	PRINT CONTROL CHARACTER POSITION	

SCBETTID DS	H	TRANSLATION TABLE IMAGE DISPL.	
SCBEVBNX DS	H	USABLE LINES FROM VFB (HP54=NDX)	
SCBELINX DS	H	TOTAL LINES#/PAGE (HP54)	
SCBECH1X DS	H	LINES# BEFORE CHANNEL 1 (HP54)	
SCBELPNX DS	CL3	REQUESTED LOOP FOR HP54	
SCBELXI DS	X	LOOP LENGTH IN 1/6 INCH (HP54)	
SCBERPMX DS	F	RASTER PATTERN MEMORY IN KB (HP54)	
SCBETWUX DS	X	TWO-UP MODE FRO: RECORD FORM	
SCBETW1X EQU	X'01'	MODE 1	
SCBETW2X EQU	X'02'	MODE 2	
SCBETW3X EQU	X'04'	MODE 3	
SCBETWNX EQU	X'80'	NO TWO-UP	
*			
SCBEC#DX DS	CL1	LPI AND CPI (CPI-6) FOR HP54 FROM 1RST	
*		CMD OR POOL OR DEFAULT CHARS	
*		TO USE WITH THE EQU OF SCBOC#DH.	
SCBEITRA DS	X	INPUT TRAY # (LP65=NDY,ND50)	
SCBERESR DS	X	RESOURCE PRINTER	
SCBEND65 EQU	X'80'	LP65	
SCBED65R EQU	X'40'	DIA RECTO FOR LP65 SPECIFIED	
SCBED65V EQU	X'20'	DIA VERSO FOR LP65 SPECIFIED	
SCBERAPA EQU	X'10'	APA	
SCBERFOD EQU	X'08'	FORMDEF SPECIFIED FOR APA	
SCBERPAD EQU	X'04'	PAGEDEF SPECIFIEC FOR APA	
SCBETRCY EQU	X'02'	TAB-REF-CHAR = YES	
SCBELIP EQU	X'01'	LIP-2090	002
*			
SCBEVBXC DS	H	USABLE LINES FROM CROSS VFB (HP54=NDX)	
SCBELIXC DS	H	CROSS TOTAL LINES#/PAGE (HP54)	
SCBECX1C DS	H	CROSS LINES# BEFORE CHANNEL 1 (HP54)	
SCBECRLX DS	CL3	REQUESTED CROSS LOOP FOR HP54	
SCBELXIC DS	X	CROSS LOOP LENGTH IN 1/6 INCH (HP54)	
SCBETRCD DS	H	DISPLACEMENT TO TABLE REFERENCE CHAR	
*		[A(SCB) + &P.TRCD] POINTS TO AN AREA	
*		OF (4 * 8) BYTES.	
SCBETSAV DS	F	TO (LAST) STRING COUNTER SAVED	
SCBESSAV DS	F	SECTION COUNTER SAVED	
SCBE50FO DS	CL8	FORMDEF FOR APA-2050	
SCBE50PA DS	CL8	PAGEDEF FOR APA-2050	
SCBE90FO DS	CL8	FORMDEF FOR APA-2090	
SCBE90PA DS	CL8	PAGEDEF FOR APA-2090	
SCBEKPOS DS	H	KEYPOS OF FILE TO BE PRINTED	
SCBEFKLN DS	X	KEYLEN OF FILE TO BE PRINTED	
SCBELAGX DS	X	LANGUAGE EXTENSION	
SCBEARTL EQU	X'01'	= ARABIC RIGHT TO LEFT	
SCBEALTR EQU	X'02'	= ARABIC LEFT TO RIGHT	
SCBEFRTL EQU	X'03'	= FARSI RIGHT TO LEFT	
SCBEFLTR EQU	X'04'	= FARSI LEFT TO RIGHT	

```

*
SCBEFOND DS      H      DISPLACEMENT TO FORMAT NAME
SCBETMBS DS      C      START TIME JOB SPOOLED OUT (SUMMER/WINTER)
SCBETMES DS      C      END TIME JOB SPOOLED OUT (SUMMER/WINTER)
SCBELOTS DS      C      LOGOFF TIME (SUMMER/WINTER)
SCBEDOCF DS      X      DOCUMENT-FORMAT                                001
SCBEDCFT EQU     X'00'   = TEXT
SCBEDCFP EQU     X'01'   = PAGE-FORMAT
SCBEDCFS EQU     X'02'   = SPECIAL-FORMAT
*
SCBEUFND DS      H      DISPLACEMENT TO UFS FILENAME
SCBEUCND DS      H      DISPLACEMENT TO UFS COPY NAME
SCBECFRM DS      CL6    CLIENT FORM FOR EXIT 94
SCBEMLSV DS      H      MAX LINES OF PRINT REQUEST
SCBEDIFD DS      H      DISPLACEMENT TO DIA-IN-FORM STRUCTURE          002
SCBESPSD DS      H      DISPLACEMENT TO SPS PRINT PARAMETER           022
SCBEDPRD DS      H      DISPLACEMENT TO DPRINT PRINT PARAMETER        022
SCBEFLTD DS      H      DISPLACEMENT TO FILTER WORK FILE INFORM.      022
SCBEFRMD DS      H      DISPLACEMENT TO DOCUMENT FORMAT               022
SCBESEND DS      F      END OF SECTION (# RECORDS)                     022
SCBECMXI DS      0XL20  CMX_INFORMATION
SCBECMXH DS      F      HEADER
SCBECMXN DS      CL8    HOST NAME
SCBECMXA DS      CL8    APPLICATION
SCBEOHST DS      CL8    ORIGINAL HOST NAME
SCBECTSN DS      CL4    ORIGINAL TSN
SCBEIHST DS      CL8    ICRM HOST NAME
SCBESERV DS      CL8    ORIGINAL SERVER NAME
SCBEDHST DS      CL8    DESTINATION HOST NAME
SCBEDTSN DS      CL4    SERVER TSN
SCBECIFT DS      CL4    CATID OF FILE TO BE TRANSFER
SCBECICT DS      CL4    CATID OF CONTAINER
SCBERDSB DS      CL8    ORIGINAL SCHEDULING BITS FOR RDIR
SCBEVIDS DS      H      DISPLACEMENT TO PROGRAM NAME AND STRING        021
SCBEAPSC DS      H      DISPLACEMENT TO SPS SCHEDULING AREA            063
SCBEPCLD DS      H      DISPLACEMENT TO PCL INFORMATION STRUCTURE      071
SCBEAIDD DS      H      DISPLACEMENT TO ACCOUNT-ID                     073
SCBESDAT DS      F      SCHEDULING DATE                                074
SCBESTIM DS      H      SCHEDULING TIME                                074
      ORG SCBESDAT
SCBESDCD DS      CL6    DATE AND TIME IN CHAR                           074
      ORG SCBESDAT
SCBESDCB DS      XL6    DATE AND TIME IN HEXA                           074
SCBELUID DS      H      DISPLACEMENT TO LONG USER-ID                    076
SCBEAPAD DS      H      APA OVERLAY DISPLACEMENT                       078
      DS      H      DISPL FOR NEW EXTENSIONS - DON'T TOUCH !          078
SCBEVARI DS      CL728  VARIABLE PART OF THE EXTENSION
SCBEVAR2 DS      CL1024 EXTENSION FOR POSIX FILENAME

```

```
*****
*
* THE VARIABLE PART MUST BE USED FOR PARAMETERS OF VARIABLE *
* LENGTH, I.E. PLAMLIB ELEMENT NAME, JV NAME, SECTION ID. *
* EACH VARIABLE NAME MUST BE PRECEDED BY ITS LENGTH (1 BYTE). *
* THE ADDRESS OF THIS FIELD = SCB @ + SCB DISPLACEMENT FIELD. *
* THE DISPLACEMENT FIELDS ARE EITHER IN THE FIXED PART OR IN *
* THE EXTENSION FIXED PART OF THE SCB. *
* FOR EXAMPLE : *
* &P.PLEL DS X LENGTH OF PLAM ELEMENT NAME 930 *
* &P.PLEN DS CL(&P.PLEL) PLAM ELEMENT NAME 930 *
* THE ADDRESS OF &P.PLEL = ADDR(SCB) + SCBOPLED. *
*
*****
SCBESLEN EQU *-SCBESCB
SCBELEN EQU *-SCBETOP
END
```

6.2 SVC-Listen

- [Liste der in System Exits gültigen SVCs](#)
- [Liste der SVCs, die in System Exits nicht verwendet werden dürfen](#)
- [Hinweise](#)

6.2.1 Liste der in System Exits gültigen SVCs

Sortiert nach SVC-Nummern

SVC-Nummer		Makroname (in System Exits gültig)	Bemerkung (Tabelle sortiert nach SVC-Nummer)
dezimal	sedezimal		
1	1	RELM (31 Bit)	
1	1	REQM (31 Bit)	
1	1	DISMP (31 Bit)	
1	1	ENAMP (31 Bit)	
1	1	RELMP (31 Bit)	
1	1	REQMP (31 Bit)	
1	1	MINF (31 Bit)	
1	1	CSTMP (31 Bit)	
1	1	CSTAT (31 Bit)	
2	2	LPOV	
7	7	DCLVAR	
7	7	DELVAR	
7	7	SETVAR	
7	7	GETVAR	
7	7	GETNVAR	
7	7	SETNVAR	
7	7	POSVAR	
23	17	GETOD	
25	19	DUMP (24 Bit)	
26	1A	CDUMP (31 Bit)	
27	1B	CDUMP	
28	1C	QUIET	
30	1E	EXCALL	
31	1F	EXECITC	
32	20	DELPOOL	s. Hinweise

32	20	CREAPOOL	s. Hinweise
33	21	STAMCE	
35	23	GEPRT	
36	24	BCAM-SVC	
40	28	EAM (31 Bit)	
42	2A	SWITCH	
43	29	MAREN	LOGON-Umgebung nötig
45	2D	FITC	
46	2E	ADAM	
49	31	GETUGR	
49	31	SRMSUG	
49	31	RDUID	
56	38	XHCS	
64	40	SETUS	
65	41	GETUS	
70	46	TMODE	
70	46	TSTAT	
70	46	TCHNG	
71	47	SETSW	
72	48	GETSW	
76	4C	PASS	
79	4F	SVC79	
84	54	PFMON	nur auf Sonderfreigabe
87	57	GTMAP	
89	59	VPASS	
95	5F	AUDIT	
96	60	MSG7	
99	63	AINF	
99	63	AREC	

99	63	ASPC	
100	64	HOSTPR	
102	66	NKGTYPE	
103	67	VMGINF	
104	68	ISREQ	
105	69	OSTAT	
117	75	VSVI	s. Hinweise
117	75	PBUNLD	s. Hinweise
119	77	DSSMTST1	
120	78	DSSMTST2	
127	7F	MRSINF	
129	81	AIDTRAP	
130	82	TAM	
135	87	SINF	
135	87	TINF	
138	8A	DHSRCL	
140	8C	JOBINFO	
140	8C	JOBINFOL	
140	8C	JOBINFOS	
144	90	FSTAT (31 Bit)	
144	90	RDTFT (31 Bit)	
144	90	ERASE (31 Bit)	
144	90	CATAL (31 Bit)	
144	90	IMPORT (31 Bit)	
144	90	FILE (31 Bit)	
146	92	GEPRT (31 Bit)	
148	94	\$ITRM	
149	95	\$IGPT	
150	96	TYPIO	

168	A8	ELIM (31 Bit)	s. Hinweise
168	A8	GET (31 Bit)	s. Hinweise
168	A8	GETFL (31 Bit)	s. Hinweise
168	A8	GETKY (31 Bit)	s. Hinweise
168	A8	GETR (31 Bit)	s. Hinweise
168	A8	GOTO (31 Bit)	s. Hinweise
168	A8	INSRT (31 Bit)	s. Hinweise
168	A8	ISREQ (31 Bit)	s. Hinweise
168	A8	OSTAT (31 Bit)	s. Hinweise
168	A8	PUT (31 Bit)	s. Hinweise
168	A8	PUTX (31 Bit)	s. Hinweise
168	A8	RETRY (31 Bit)	s. Hinweise
168	A8	SETL (31 Bit)	s. Hinweise
168	A8	STORE (31 Bit)	s. Hinweise
169	A9	RELSE (24 Bit)	
170	AA	PAM (31 Bit)	
171	AB	BTAM (31 Bit)	
172	AC	IOSID	
177	B1	EAM (24 Bit)	s. Hinweise
179	B3	PAM (24 Bit)	s. Hinweise
181	B5	CHNGE	
182	B6	COPY	
183	B7	LINK (31 Bit)	s. Hinweise
183	B7	TABLE (31 Bit)	s. Hinweise
183	B7	ITABL (31 Bit)	s. Hinweise
183	B7	SULNK (31 Bit)	s. Hinweise
186	BA	LBRET (31 Bit)	
186	BA	OPEN (31 Bit)	
186	BA	CLOSE (31 Bit)	

186	BA	EXRTN (31 Bit)	
186	BA	FEOV(31 Bit)	
188	BC	GET (31 Bit)	
188	BC	PUT (31 Bit)	
188	BC	SETL (31 Bit)	
188	BC	PUTX (31 Bit)	
188	BE	RELSE (31 Bit)	
190	BE	GETJV (31 Bit)	s. Hinweise
190	BE	SETJV (31 Bit)	s. Hinweise
190	BF	STAJV (31 Bit)	s. Hinweise
191	BF	JSATTCH (31 Bit)	
191	BF	JSDETC (31 Bit)	
191	BF	JSEXPCT (31 Bit)	
191	BF	JSINFO (31 Bit)	
191	BF	JSWAKE (31 Bit)	
191	BF	JSRUNJB (31 Bit)	
191	BF	JINF (31 Bit)	
191	CE	TMODE (31 Bit)	
206	CE	§NKACALL (31 Bit)	
208	D0	RDUID	
221	DD	\$VRIF	
228	E4	\$REQM	
229	E5	\$RELM	
237	ED	\$TERM	

Sortiert nach Makronamen

SVC-Nummer		Makroname (in System Exits gültig)	Bemerkung (Tabelle sortiert nach Mkrname)
dezimal	sedezimal		
149	95	\$IGPT	

148	94	\$ITRM	
206	CE	\$NKACALL (31 Bit)	
229	E5	\$RELM	
228	E4	\$REQM	
237	ED	\$TERM	
221	DD	\$VRIF	
46	2E	ADAM	
129	81	AIDTRAP	
99	63	AINF	
99	63	AREC	
99	63	ASPC	
95	5F	AUDIT	
36	24	BCAM-SVC	
171	AB	BTAM (31 Bit)	
144	90	CATAL (31 Bit)	
26	1A	CDUMP (31 Bit)	
27	1B	CDUMP	
181	B5	CHNGE	
186	BA	FEOV(31 Bit)	
182	B6	COPY	
32	20	CREAPOOL	s. Hinweise
1	1	CSTAT (31 Bit)	
1	1	CSTMP (31 Bit)	
7	7	DCLVAR	
32	20	DELPOOL	s. Hinweise
7	7	DELVAR	
138	8A	DHSRCL	
1	1	DISMP (31 Bit)	
119	77	DSSMTST1	

120	78	DSSMTST2	
25	19	DUMP (24 Bit)	
40	28	EAM (31 Bit)	
177	B1	EAM (24 Bit)	s. Hinweise
168	A8	ELIM (31 Bit)	s. Hinweise
1	1	ENAMP (31 Bit)	
144	90	ERASE (31 Bit)	
30	1E	EXCALL	
31	1F	EXECITC	
186	BA	EXRTN (31 Bit)	
186	BA	FEOV(31 Bit)	
144	90	FILE (31 Bit)	
45	2D	FITC	
144	90	FSTAT (31 Bit)	
35	23	GEPRT	
146	92	GEPRT (31 Bit)	
168	A8	GET (31 Bit)	s. Hinweise
188	BC	GET (31 Bit)	
168	A8	GETFL (31 Bit)	s. Hinweise
190	BE	GETJV (31 Bit)	s. Hinweise
168	A8	GETKY (31 Bit)	s. Hinweise
7	7	GETNVAR	
23	17	GETOD	
168	A8	GETR (31 Bit)	s. Hinweise
72	48	GETSW	
49	31	GETUGR	
65	41	GETUS	
7	7	GETVAR	
168	A8	GOTO (31 Bit)	s. Hinweise

87	57	GTMAP	
100	64	HOSTPR	
144	90	IMPORT (31 Bit)	
168	A8	INSRT (31 Bit)	s. Hinweise
172	AC	IOSID	
104	68	ISREQ	
168	A8	ISREQ (31 Bit)	s. Hinweise
183	B7	ITABL (31 Bit)	s. Hinweise
191	BF	JINF (31 Bit)	
140	8C	JOBINFO	
140	8C	JOBINFOL	
140	8C	JOBINFOS	
191	BF	JSATTCH (31 Bit)	
191	BF	JSDETC (31 Bit)	
191	BF	JSEXPCT (31 Bit)	
191	BF	JSINFO (31 Bit)	
191	BF	JSRUNJB (31 Bit)	
191	BF	JSWAKE (31 Bit)	
186	BA	LBRET (31 Bit)	
183	B7	LINK (31 Bit)	s. Hinweise
2	2	LPOV	
43	29	MAREN	LOGON-Umgebung nötig
1	1	MINF (31 Bit)	
127	7F	MRSINF	
96	60	MSG7	
102	66	NKGTYPE	
186	BA	OPEN (31 Bit)	
105	69	OSTAT	
168	A8	OSTAT (31 Bit)	s. Hinweise

170	AA	PAM (31 Bit)	
179	B3	PAM (24 Bit)	s. Hinweise
76	4C	PASS	
117	75	PBUNLD	s. Hinweise
84	54	PFMON	nur auf Sonderfreigabe
7	7	POSVAR	
168	A8	PUT (31 Bit)	s. Hinweise
188	BC	PUT (31 Bit)	
168	A8	PUTX (31 Bit)	s. Hinweise
188	BC	PUTX (31 Bit)	
28	1C	QUIET	
144	90	RDTFT (31 Bit)	
49	31	RDUID	
208	D0	RDUID	
1	1	RELM (31 Bit)	
1	1	RELMP (31 Bit)	
169	A9	RELSE (24 Bit)	
188	BE	RELSE (31 Bit)	
1	1	REQM (31 Bit)	
1	1	REQMP (31 Bit)	
168	A8	RETRY (31 Bit)	s. Hinweise
190	BE	SETJV (31 Bit)	s. Hinweise
168	A8	SETL (31 Bit)	s. Hinweise
188	BC	SETL (31 Bit)	
7	7	SETNVAR	
71	47	SETSW	
64	40	SETUS	
7	7	SETVAR	
135	87	SINF	

49	31	SRMSUG	
190	BF	STAJV (31 Bit)	s. Hinweise
33	21	STAMCE	
168	A8	STORE (31 Bit)	s. Hinweise
183	B7	SULNK (31 Bit)	s. Hinweise
79	4F	SVC79	
42	2A	SWITCH	
183	B7	TABLE (31 Bit)	s. Hinweise
130	82	TAM	
70	46	TCHNG	
135	87	TINF	
70	46	TMODE	
191	CE	TMODE (31 Bit)	
70	46	TSTAT	
150	96	TYPIO	
103	67	VMGINF	
89	59	VPASS	
117	75	VSVI	s. Hinweise
56	38	XHCS	

6.2.2 Liste der SVCs, die in System Exits nicht verwendet werden dürfen

Sortiert nach SVC-Nummern

SVC-Nummer		Makroname (in System Exits nicht erlaubt)	Bemerkung (Tabelle sortiert nach SVC-Nummer)
dezimal	sedezimal		
3	3	NPCSVK	
5	5	WRCPT (31 Bit)	
6	6	Mess-SVC	
8	8	SRMASVC	
9	9	TERM	\$TERM verwenden
10	A	NCAN	
11	B	IOTRACE	
12	C	NPCHK	
13	D	VMMUSER	
14	E	NKDINF	
15	F	IPPMBASE	
16	10	CRYPT	
22	16	TERMD	
24	18	GEPRT (24 Bit)	
29	1D	SCA	
36	24	BCAM-SVC	
37	25	REQSPF	
38	26	MSG (31 Bit)	
39	27	RDCRD	s. Hinweise
39	27	RDATA	s. Hinweise
39	27	WROUT	s. Hinweise
39	27	WRLST	s. Hinweise
39	27	WRTRD	s. Hinweise
39	27	WRTOT	s. Hinweise
39	27	\$EOUT	s. Hinweise

System Exits

41	29	SAT	
44	2C	YTCMAC	
47	2F	PLAM	
48	30	WARTOPT	
50	32	OPCOM	
51	33	SEVNT	
52	34	REVNT	
53	35	RELBF	
54	36	CLCOM	
57	39	CSSCN	
59	3B	MONTB	
60	3C	ERFLG	
61	3D	WRTOT	
62	3E	RDCRD	
63	3F	TERMJ	
63	3F	TERMJD	
66	42	RDATA	
67	43	WROUT	
68	44	WRTRD	
69	45	WRLST	
73	49	REQM (24 Bit)	
74	4A	RELM (24 Bit)	
75	4B	CSTAT (24 Bit)	
81	51	JSINF (24 Bit)	
81	51	JSINFO (24 Bit)	
81	51	JSRUNDB (24 Bit)	
81	51	JSFXCPT (24 Bit)	
81	51	JSDETC (24 Bit)	
81	51	JSATTCH (24 Bit)	

83	53	SM2	nur von SM2 verwendbar
85	55	IC02000	
86	56	SETBF	
88	58	CMD (24 Bit)	
92	5C	BKPT	
93	5D	MSG (24 Bit)	
94	5E	FASTACC	
94	5E	FASTPAM	
98	62	DBSVC	
106	6A	UNLOD	
107	6B	PRNT	
108	6C	\$SPRQ	
108	6C	PNCH	
109	6D	ITABL (24 Bit)	
110	6E	LINK (24 Bit)	
110	6E	SULNK	
111	6F	TABLE	
112	70	VOLIN	
113	71	INITMB	
114	72	SFDLOAD	
116	74	NLMSERVE	
118	76	\$SPORQ (RSO)	
121	79	DISSI	Serialisierung
121	79	ENASI	Serialisierung
121	79	CHKSI	Serialisierung
121	79	DEQAR	Serialisierung
121	79	ENQAR	Serialisierung
122	7A	ENAMP (24 Bit)	
122	7A	REQMP (24 Bit)	

System Exits

122	7A	RELMP (24 Bit)	
122	7A	DISMP (24 Bit)	
123	7B	SUSPEND	Contingency
123	7B	LEVCO	Contingency
123	7B	RETCO	Contingency
123	7B	DISCO	Contingency
123	7B	ENACO	Contingency
124	7C	POSSIG	\$EXDEV, \$EXEVI, EEXPOS verwenden (Eventing)
124	7C	SOLSIG	
124	7C	ENAEI	
124	7C	CHKEI	
124	7C	DISEI	
125	7D	(DCAM-Aufrufe)	
126	7E	DIV	
128	80	STXIT	
128	80	CONXT	
128	80	EXIT	
131	83	ARCHIVE	
132	84	SPOOLFREE	
133	85	STAM (24 Bit)	
133	85	ONEVT (24 Bit)	
133	85	STAJV (24 Bit)	
133	85	DCLJV (24 Bit)	
133	85	CATJV (24 Bit)	
133	85	ERAJV (24 Bit)	
133	85	GETJV (24 Bit)	
133	85	SETJV (24 Bit)	
134	86	KDCEND	
139	8B	SIR	

System Exits

141	8D	IOFCOPY	
142	8E	LOADPM	
143	8F	FDDRL	
145	91	CMD (31 Bit)	
147	93	FEOV (24 Bit)	
151	97	GET (24 Bit)	
152	98	PUT (24 Bit)	
153	99	PUTX (24 Bit)	
154	9A	SETL (24 Bit)	
155	9B	STORE (24 Bit)	
156	9C	OPEN (24 Bit)	
157	9D	CATAL (24 Bit)	
158	9E	ERASE (24 Bit)	
159	9F	FILE (24 Bit)	
160	A0	FSTAT (24 Bit)	
160	A0	RDTFT (24 Bit)	
161	A1	RELEASE (24 Bit)	
162	A2	CLOSE (24 Bit)	
163	A3	GOTO (24 Bit)	
164	A4	GETFL (24 Bit)	
165	A5	VERIF (24 Bit)	
166	A6	IMPORT (24 Bit)	
173	AD	INSRT (24 Bit)	
174	AE	ELIM (24 Bit)	
175	AF	GETKY (24 Bit)	
176	B0	GETR (24 Bit)	
178	B2	EXRTN (24 Bit)	
178	B2	LBRET (24 Bit)	
180	B4	BTAM (24 Bit)	

184	B8	POSIX	
185	B9	SRMUINF	
187	BB	RSOFEI	Eventing
187	BB	DELFEI	Eventing
187	BB	DSOFEI	Eventing
187	BB	RPOFEI	Eventing
187	BB	DPOFEI	Eventing
189	BD	FT-BS2000	
199	C7	\$DOER	
213	D5	\$EXCP	
214	D6	\$WAIT	
215	D7	\$XCPW	
216	D8	\$PGFX	
216	D8	\$PGFR	
222	DE	\$AUDSW	
233	E9	\$FNAT	
234	EA	\$FNDT	
235	EB	\$PEND	
236	EC	\$UNPN	
238	EE	\$PNUP	
241	F1	\$PNDT	
245	F5	\$MASK	
246	F6	\$UNMASK	
250	FA	\$BOWT	
255	FF	DIAGNOSE SVC	

Sortiert nach Makronamen

SVC-Nummer	Makroname (in System Exits nicht erlaubt)	Bemerkung (Tabelle sortiert nach Makronamen)
------------	--	---

dezimal	sedezimal		
222	DE	\$AUDSW	
250	FA	\$BOWT	
199	C7	\$DOER	
39	27	\$EOUT	s. Hinweise
213	D5	\$EXCP	
233	E9	\$FNAT	
234	EA	\$FNDT	
245	F5	\$MASK	
235	EB	\$PEND	
216	D8	\$PGFR	
216	D8	\$PGFX	
241	F1	\$PNDT	
238	EE	\$PNUP	
118	76	\$SPORQ (RSO)	
108	6C	\$SPRQ	
246	F6	\$UNMASK	
236	EC	\$UNPN	
214	D6	\$WAIT	
215	D7	\$XCPW	
125	7D	(DCAM-Aufrufe)	
131	83	ARCHIVE	
36	24	BCAM-SVC	
92	5C	BKPT	
180	B4	BTAM (24 Bit)	
157	9D	CATAL (24 Bit)	
133	85	CATJV (24 Bit)	
124	7C	CHKEI	
121	79	CHKSI	Serialisierung

54	36	CLCOM	
162	A2	CLOSE (24 Bit)	
88	58	CMD (24 Bit)	
145	91	CMD (31 Bit)	
128	80	CONXT	
16	10	CRYPT	
57	39	CSSCN	
75	4B	CSTAT (24 Bit)	
98	62	DBSVC	
133	85	DCLJV (24 Bit)	
187	BB	DELFEI	Eventing
121	79	DEQAR	Serialisierung
255	FF	DIAGNOSE SVC	
123	7B	DISCO	Contingency
124	7C	DISEI	
122	7A	DISMP (24 Bit)	
121	79	DISSI	Serialisierung
126	7E	DIV	
187	BB	DPOFEI	Eventing
187	BB	DSOFEI	Eventing
174	AE	ELIM (24 Bit)	
123	7B	ENACO	Contingency
124	7C	ENAEI	
122	7A	ENAMP (24 Bit)	
121	79	ENASI	Serialisierung
121	79	ENQAR	Serialisierung
133	85	ERAJV (24 Bit)	
158	9E	ERASE (24 Bit)	
60	3C	ERFLG	

System Exits

128	80	EXIT	
178	B2	EXRTN (24 Bit)	
94	5E	FASTACC	
94	5E	FASTPAM	
143	8F	FDDRL	
147	93	FEOV (24 Bit)	
159	9F	FILE (24 Bit)	
160	A0	FSTAT (24 Bit)	
189	BD	FT-BS2000	
24	18	GEPRT (24 Bit)	
151	97	GET (24 Bit)	
164	A4	GETFL (24 Bit)	
133	85	GETJV (24 Bit)	
175	AF	GETKY (24 Bit)	
176	B0	GETR (24 Bit)	
163	A3	GOTO (24 Bit)	
85	55	IC02000	
166	A6	IMPORT (24 Bit)	
113	71	INITMB	
173	AD	INSRT (24 Bit)	
141	8D	IOCFCOPY	
11	B	IOTRACE	
15	F	IPPMBASE	
109	6D	ITABL (24 Bit)	
81	51	JSATTCH (24 Bit)	
81	51	JSDETC (24 Bit)	
81	51	JSFXCPT (24 Bit)	
81	51	JSINF (24 Bit)	
81	51	JSINFO (24 Bit)	

81	51	JSRUNDB (24 Bit)	
134	86	KDCEND	
178	B2	LBRET (24 Bit)	
123	7B	LEVCO	Contingency
110	6E	LINK (24 Bit)	
142	8E	LOADPM	
59	3B	MONTB	
93	5D	MSG (24 Bit)	
38	26	MSG (31 Bit)	
6	6	Mess-SVC	
10	A	NCAN	
14	E	NKDINF	
116	74	NLMSERVE	
12	C	NPCHK	
3	3	NPCSV	
133	85	ONEVT (24 Bit)	
50	32	OPCOM	
156	9C	OPEN (24 Bit)	
47	2F	PLAM	
108	6C	PNCH	
184	B8	POSIX	
124	7C	POSSIG	
107	6B	PRNT	
152	98	PUT (24 Bit)	
153	99	PUTX (24 Bit)	
39	27	RDATA	s. Hinweise
66	42	RDATA	
39	27	RDCRD	s. Hinweise
62	3E	RDCRD	

160	A0	RDTFT (24 Bit)	
53	35	RELBF	
161	A1	RELEASE (24 Bit)	
74	4A	RELM (24 Bit)	
122	7A	RELMP (24 Bit)	
73	49	REQM (24 Bit)	
122	7A	REQMP (24 Bit)	
37	25	REQSPF	
123	7B	RETCO	Contingency
52	34	REVNT	
187	BB	RPOFEI	Eventing
187	BB	RSOFEI	Eventing
41	29	SAT	
29	1D	SCA	
86	56	SETBF	
133	85	SETJV (24 Bit)	
154	9A	SETL (24 Bit)	
51	33	SEVNT	
114	72	SFDLOAD	
139	8B	SIR	
83	53	SM2	nur von SM2 verwendbar
124	7C	SOLSIG	
132	84	SPOOLFREE	
8	8	SRMASVC	
185	B9	SRMUINF	
133	85	STAJV (24 Bit)	
133	85	STAM (24 Bit)	
155	9B	STORE (24 Bit)	
128	80	STXIT	

110	6E	SULNK	
123	7B	SUSPEND	Contingency
111	6F	TABLE	
9	9	TERM	\$TERM verwenden
22	16	TERMD	
63	3F	TERMJ	
63	3F	TERMJD	
106	6A	UNLOD	
165	A5	VERIF (24 Bit)	
13	D	VMMUSER	
112	70	VOLIN	
48	30	WARTOPT	
5	5	WRCPT (31 Bit)	
39	27	WRLST	s. Hinweise
69	45	WRLST	
39	27	WROUT	s. Hinweise
67	43	WROUT	
39	27	WRTOT	s. Hinweise
61	3D	WRTOT	
39	27	WRTRD	s. Hinweise
68	44	WRTRD	
44	2C	YTCMAC	

6.2.3 Hinweise

SYSFILE-Umgebung

Die Makroaufrufe RDATA und WRTRD erzeugen Wartezustände und haben bei ihrer Abarbeitung wegen der Verknüpfung mit dem Break-/Escape-Processing (K2-Taste) entsprechende Regeln zur Verwendung von Prozess-Level und Aufrufhierarchien. Die korrekte Ausführung dieser Funktionen ist daher nur bei einem Aufruf aus dem Zustand TU gewährleistet.

Ebenso wie RDATA/WRTRD braucht der Makroaufruf WROUT eine initialisierte SYSFILE-Umgebung.

DCAM-Aufrufe

Die DCAM-Aufrufe können im Zustand TPR und damit in System Exits nicht genutzt werden, da sie bei ihrer Ausführung sowohl Prozesslevel verändern als auch TPR-Contingencies verwenden. Der SVC selbst muss daher auf einem Prozesslevel < 156 (= Zustand TU) aufgerufen werden.

ISAM SVCs

Die ISAM-SVCs 32 und 168 können grundsätzlich in System Exits verwendet werden. Bezüglich des SVC 168 ist jedoch im SHARUPD-Modus Folgendes zu beachten:

Durch die meisten ISAM-Makroaufrufe werden vom Auftrag gehaltene externe ISAM-Sperren freigegeben, sodass stets nur eine Sperre gehalten wird und deadlock-freie Verarbeitung gewährleistet ist. Durch einen ISAM-Aufruf in System Exits kann also u.U. eine vom Anwenderprogramm gesetzte Sperre zurückgesetzt werden, ohne dass dieses vom Anwenderprogramm registriert wird.

Beispiel

Anwenderprogramm:	GET	FCB1,AREA1,LOCK
System Exit:	GET	FCB2,AREA2,LOCK
Anwenderprogramm:	PUTX	FCB1,AREA1

Der PUTX wird ohne System Exit korrekt ausgeführt, mit System Exit ist die Sperre aufgehoben, es können unerwartete Ergebnisse auftreten.

Daher sollten bei Shared-Update-Verarbeitung in System Exits nur dann ISAM-Makros, die eine Sperre aufheben können, aufgerufen werden, wenn gewährleistet ist, dass kein Anwenderprogramm geladen ist. Selbst in diesem Fall muss sichergestellt werden, dass es keine unerwünschte Wechselwirkung mit Systemdateien gibt.

Die 24-Bit-ISAM-SVCs dürfen nicht mehr in System Exits verwendet werden. Ihre Anwendung in XS-Umgebung führt sofort zum Fehler, da System Exits in TPR-Umgebung laufen, also im 31-Bit-Modus.

JMS-/JV-SVCs

Die JMS-/JV-SVCs sind in TPR-Umgebung und damit auch in System Exits ablauffähig. Sie liefern jedoch nicht generell sinnvolle Ergebnisse (z.B. der Aufruf JINF nicht, wenn keine Taskumgebung existiert).

DSSM/SULNK

Nur die SULNK-Schnittstelle ist nutzbar, wenn die System Exits von DSSM oder SULNK in privilegierten Speicher geladen wurden. Wenn die System Exits im nichtprivilegierten Speicher geladen werden, können die SVCs VSVI, PBUNLD, ITABL, TABLE und LINK verwendet werden. Der Makro \$LOAD kann nicht verwendet werden.

7 Literatur

Die Handbücher finden Sie im Internet unter <http://manuals.ts.fujitsu.com>. Handbücher, die mit einer Bestellnummer angezeigt werden, können Sie auch in gedruckter Form bestellen.

- [1] **DCAM**
Makroaufrufe
Benutzerhandbuch

- [2] **DCAM**
Programmschnittstellen
Beschreibung

- [3] **BS2000 OS DX**
Diagnosehandbuch
Benutzerhandbuch

- [4] **BS2000 OS DX**
Dienstprogramme
Benutzerhandbuch

- [5] **Distributed Print Services**
Drucken in Computernetzen
Benutzerhandbuch

- [6] **BS2000 OS DX**
DVS-Makros
Benutzerhandbuch

- [7] **BS2000 OS DX**
Einführung in die Systembetreuung
Benutzerhandbuch

- [8] **BS2000 OS DX**
Kommandos
Benutzerhandbücher

- [9] **BS2000 OS DX**
Makroaufrufe an den Ablaufteil
Benutzerhandbuch

- [10] **SDF-A**
Benutzerhandbuch

- [11] **SDF**
Einführung in die Dialogschnittstelle SDF

Benutzerhandbuch

