

z/OS



DFSMSdfp 拡張サービス

z/OS



DFSMSdfp 拡張サービス

お願い

本書および本書で紹介する製品をご使用になる前に、特記事項に記載されている情報をお読みください。

本書は、z/OS™ バージョン 1 (プログラム番号 5694-A01) リリース 7 に適用されます。また、改訂版などで特に断りのない限り、これ以降のすべてのリリースおよびモディフィケーションにも適用されます。

本書は、SD88-6462-00 の改訂版です。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： SC26-7400-05
z/OS
DFSMSdfp Advanced Services

発 行： 日本アイ・ピー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

第1刷 2006.4

この文書では、平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 1979, 2005. All rights reserved.

© Copyright IBM Japan 2006

目次

図	vii
表	ix
本書について	xi
必要な製品知識	xi
参照される資料	xi
インターネットの z/OS DFSMS 資料へのアクセス	xiii
LookAt を使用してメッセージの説明を検索する	xiii
構文図の読み方	xiv
アドレスおよびレジスターの規則	xvi
変更の要約	xix
z/OS バージョン 1 リリース 7 用 SD88-6462-01 で の変更の要約	xix
新規情報	xix
変更情報	xix
削除情報	xx
SD88-6462-00 (英文原典: SC26-7400-04) での変更の 要約	xx
変更情報	xx
z/OS バージョン 1 リリース 6 用 SC26-7400-03 で の変更の要約	xxi
新規情報	xxi
削除情報	xxi
変更情報	xxi
第 1 章 ポリリューム目録の使用	1
VTOC コンポーネント	1
データ・セット制御ブロック (DSCB) のタイプ	2
DASD スペースの割り振りと解放	13
VTOC 索引	14
VTOC 索引レコード	16
DADSM マクロを使用した VTOC へのアクセス	17
LSPACE を使用した DASD ポリリューム情報の要 求	17
OBTAIN を使用した VTOC からの DSCB の読 み取り	26
PARTREL を使用した DASD データ・セットか らの未使用スペースの解放	29
REALLOC を使用した DASD データ・セットの 作成 (割り振り)	37
CVAF マクロを使用した VTOC へのアクセス	43
逐次化と更新	44
ポリリュームの識別	44
CVPL (CVAF パラメーター・リスト) の生成	45
バッファー・リストの使用	47
マクロ ICVEDT02 を使用したエクステンション領域 のマップ	49
DSCB への直接アクセス	50
DSN または DSCB への順次アクセス	53

CVAF フィルターを使用した DSCB セットの読 み取り	55
VTOC エラー応答	62
システム・エラーまたはユーザー・エラーからの リカバリー	62
GTF トレース	62
VTOC と VTOC 索引のリスト表示	63

第 2 章 ポリリューム目録の管理 65

VTOC および VTOC 索引の作成	65
VTOC および VTOC 索引の保護	65
RACF	65
APF	65
パスワード保護	66
VTOC のコピー/復元/初期化	66
索引なし VTOC を含むポリリューム	66
索引付き VTOC を含むポリリューム	66
VTOC からのデータ・セットの削除	67
影響を受けるポリリュームの指定	67
機密データの消去	67
システム管理ストレージに関する考慮事項	68
一般的な考慮事項および制約事項	68
SCRATCH および CAMLST マクロ仕様	70
例	70
SCRATCH パラメーター・リスト	71
SCRATCH からの戻りコード	72
SCRATCH からの状況コード	73
VTOC 内のデータ・セットの名前変更	74
影響を受けるポリリュームの指定	74
システム管理ストレージに関する考慮事項	74
一般的な考慮事項および制約事項	75
RENAME および CAMLST マクロ仕様	77
例	77
RENAME パラメーター・リスト	78
RENAME からの戻りコード	78
RENAME からの状況コード	79

第 3 章 カタログ管理マクロの使用 81

アプリケーション・プログラムに関する考慮事項	81
カタログ検索順序	81
カタログからの情報の検索	82
データ・セット名による情報の検索 (LOCATE と CAMLST NAME)	82
世代別データ・セット名による情報の検索 (LOCATE と CAMLST NAME)	84
別名による情報の検索 (LOCATE と CAMLST NAME)	86
相対ブロック・アドレスによるブロックの読み取 り (LOCATE と CAMLST BLOCK)	87
非 VSAM データ・セット・カタログ項目の使用	88

DFSMS レベル判別の呼び出し	263
フォーマット	263
パラメーター	263
戻りコード	265
例	266
データ・セット属性検索の呼び出し	266
フォーマット	266
パラメーター	267
戻りコード	268
例	268
データ・セット Backup-While-Open サポートの呼び出し	268
フォーマット	268
パラメーター	269
戻りコード	270
例	271
Backup-While-Open 機能の使用	271
DFSMSdfp 共用属性の呼び出し	276
フォーマット	276
パラメーター	276
戻りコード	277
IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード	277
レコード・レベル共用照会の呼び出し	278
フォーマット	278
パラメーター	279
戻りコード	280
例	282
第 10 章 DESERV 出口の使用	283
タスク出口とグローバル出口の説明	284
タスク・レベル出口	284
グローバル出口	285
タスク・レベル出口とグローバル出口の相互作用	285
複数のタスク・レベル出口または複数のグローバル出口の確立	287
DESERV FUNC=EXIT の発行 (呼び出し環境)	287
呼び出し構文	288
DESERV 出口のインストールまたは置換	291
DESERV 出口の削除	292
DESERV 出口がアクティブかどうかの判別	292
DESERV 出口の作成	293
出口処理	293
GET 機能に関連したパラメーター	294
PUT 機能に関連したパラメーター	308
PUT の戻りコードと理由コード	309
DELETE 機能に関連したパラメーター	312
RENAME 機能に関連したパラメーター	313
UPDATE 機能に関連したパラメーター	315
出口ルーチンの入り口環境	317
出口ルーチンの出口環境	317
DESERV 出口に入るときのレジスター	318
DESERV 出口から戻るときのレジスター	319
DESERV 出口の戻りコードと理由コード	319
DESERV FUNC=EXIT 戻りコードと理由コード	319
DESERV 出口の例	321

第 11 章 システム・イメージ・ライブラリーの保守	327
SYS1.IMAGELIB 内の UCS イメージ	329
UCS イメージのコーディング例	331
UCS イメージ別名	334
SYS1.IMAGELIB 内の UCS イメージ・テーブル	334
UCS イメージ・テーブル内の別名	335
UCS イメージ・テーブル項目の追加または変更	339
UCS イメージの検査	343
SYS1.IMAGELIB 内の FCB イメージ	344
イメージ・ライブラリーへの FCB イメージの追加	346
FCB イメージの変更	348
3211 指標付け機能に対する JES サポート	350
第 12 章 階層ファイル・システム・データ・セット	351
階層ファイル・システム・データ・セットの作成	351
ルート・ファイル・システムの定義	352
ルート・ファイル・システムの作成とマウント	352
追加のファイル・システムおよびディレクトリーの作成	352
ルート・ファイル・システムへのファイル・システムの追加とマウント	353
ファイル・システム・サイズの管理	354
ファイル・システム・アクティビティの管理	355
HFS データ・セット属性へのアクセス	355
ファイル・システムのトランスポート	355
ファイル・システムの除去 (削除)	356
ファイル・システムのマイグレーション	356
ファイル・システムのバックアップ	356
バックアップ・ファイル・システムのリカバリー	357
HFS 据え置きファイル・システム同期	357
SYNC 値の指定方法	359
HFS に対する pfscctl (BPX1PCT) 物理ファイル・システム制御の使用	359
フォーマット	359

第 13 章 サブシステムの統計、状況、およびカウント情報へのユーザー・アクセス	369
レジスター 1 パラメーター・リスト	369
渡される引数リスト -- SSGARGL	369

付録 A. CVAF VTOC アクセス・マクロ	379
CVAF マクロ命令のコーディング	379
CVAFDIR マクロの概要および仕様	379
CVAFDSM マクロの概要および仕様	392
CVAFFILT マクロの概要および仕様	400
CVAFSEQ マクロの概要および仕様	411
CVAFTST マクロの概要および仕様	424
VTOC 索引エラー・メッセージと関連コード	425
エラー・メッセージ	425
CVSTAT フィールドに入るコード	426

付録 B. 制御ブロック	427
データ・エクステンション・ブロック (DEB) フィールド	427
データ機能域 (DFA) フィールド	435
付録 C. アクセシビリティ	439
支援機能の使用	439
ユーザー・インターフェースのキーボード・ナビゲーション	439

z/OS 情報	439
特記事項	441
プログラミング・インターフェース情報	442
商標	442
用語集	443
索引	447



1. ボリューム目録の位置決め	1	27. サンプル &IHADFARELS プログラム	242
2. VTOC の内容 - VTOC 索引がないボリュームの データ・セットが記述されている DSCB	3	28. シンボル定義を判別する例	242
3. VTOC とその索引の関係の例	15	29. IGWASYS CALL ステートメントの例	266
4. DADSM LSPACE フリー・スペース情報 MF=(D,MSG)	24	30. IGWASMS LINK ステートメント呼び出しの 例	268
5. DADSM LSPACE フリー・スペース情報フォー マット、MF=(D,EXPMSG).	25	31. LOAD および CALL ステートメントを使用し た IGWABWO の例	271
6. CVAF フィルター・サービスに必要な制御プロ ック	56	32. LOAD および CALL ステートメントを使用し た IGWARLS 照会呼び出しの例	282
7. EXCP のデータ制御ブロックのフォーマット (OPEN 後)	116	33. 出口ルーチン呼び出しシーケンス	286
8. DEVD=DA および DSORG=PS (または DSORG=PO) の DCB の装置依存の部分	122	34. PUT の戻りコードと理由コード	310
9. DEVD=DA および DSORG=DA の DCB の装 置依存の部分	123	35. 出口 DESERV 機能の戻りコードと理由コード	320
10. DEVD=TA および DSORG=PS の DCB の装 置依存の部分	123	36. タスク・レベル DESERV 出口の確立と削除	322
11. DEVD=PR および DSORG=PS の DCB の装 置依存の部分	124	37. サンプル DESERV 出口ルーチン	324
12. DEVD=PC または RD および DSORG=PS の DCB の装置依存の部分	124	38. 1403 UCS イメージを SYS1.IMAGELIB に追 加するコード	331
13. 入出力ブロック (IOB) フォーマット	128	39. 3203 UCS イメージを SYS1.IMAGELIB に追 加するコード	332
14. IOBE のフォーマット (IOSDIOBE マクロによ ってマップされます)	132	40. 3211 UCS イメージを SYS1.IMAGELIB に追 加するサンプル・コード	333
15. IEDB のフォーマット (IOSDIEDB マクロによ ってマップされます)	135	41. UCS イメージ・テーブル項目フォーマット	336
16. 完了コードの通知後のイベント制御ブロック	136	42. 4245 UCS イメージ・テーブル (UCS5) への 新規バンド ID の追加	342
17. ADD 機能のパラメーター・リスト	173	43. 4248 UCS イメージ・テーブル (UCS6) への 新規デフォルト項目の追加	343
18. REPLACE 機能のパラメーター・リスト	174	44. 標準 STD1 FCB イメージのフォーマット	345
19. DELETE 機能のパラメーター・リスト	175	45. 標準 STD2 FCB イメージのフォーマット	346
20. LIST 機能のパラメーター・リスト	176	46. FCB ロード・モジュールをアセンブルして SYS1.IMAGELIB に追加するためのサンプ ル・コード	348
21. 標準形式の RDJFCB マクロの例	207	47. VTOC での CVAFDIR マクロの例	388
22. RDJFCB マクロを使用したコード例	208	48. 索引付き VTOC での CVAFDIR マクロの使 用例	391
23. EXCP によるマルチボリューム・データ・セ ットの処理	213	49. CVAFFILT マクロの例	407
24. 割り振り情報検索のコード例	218	50. 索引付き VTOC での CVAFSEQ マクロの使 用例	418
25. 標準形式の OPEN TYPE=J マクロの例	224	51. 索引なし VTOC での CVAFSEQ マクロの使 用例	421
26. IOB チェーン	229		

表

1. 参照資料	xi	43. XDAP で使用されるチャンネル・プログラム・	
2. DSCB Format-1	5	コマンド・ワード	163
3. DSCB Format-3	9	44. PROTECT 戻りコード	177
4. DSCB Format-4	10	45. 領域の最小サイズ	187
5. DSCB Format-5	13	46. INFO=AMCAP 32 バイトの戻りデータ	195
6. LSPACE パラメーター・リストのフォーマット		47. EXCP またはアクセス方式の大容量ブロッ	
(MF=D)	23	ク・インターフェースを使用している場合に	
7. DADSM LSPACE メッセージ戻り域の内容	25	サポートされる最適および最大ブロック・サ	
8. LSPACE データ戻り域フォーマット	25	イズ	195
9. DADSM OBTAIN 戻りコード	28	48. シミュレートされた装置 UCBTYP 情報	196
10. DADSM OBTAIN 戻りコード	29	49. シミュレートされた装置特性情報	197
11. DADSM PARTREL 戻りコード	35	50. DEVTYPE マクロからの出力	200
12. REALLOC パラメーター・リスト	41	51. DEVTYPE マクロからの出力 - DASD 装置	200
13. DADSM CREATE 戻りコード	42	52. RDJFCB マクロからの戻りコード	209
14. CVAF パラメーター・リストのフォーマット	45	53. 割り振り検索リストのフォーマット (IHAARL	
15. CVPL の CVFCTN フィールド - 内容と定義	47	マクロによってマップ)	214
16. バッファ・リスト・ヘッダーのフォーマット	48	54. 割り振り検索域のフォーマット (IHAARA マ	
17. バッファ・リスト項目のフォーマット	49	クロによってマップ)	216
18. ICVEDT02 マッピング・マクロのフォーマット	49	55. Backup-While-Open 標識	274
19. フィルター基準リスト・ヘッダーのフォーマッ		56. IGWASYS、IGWASMS、IGWABWO、	
ト	56	IGWLSHR の戻りコードと理由コード	277
20. フィルター基準リスト項目のフォーマット	58	57. IGWARLS の戻りコードと理由コード	280
21. SCRATCH パラメーター・リスト	71	58. DESERV スクリーン・テーブル構造	291
22. SCRATCH 戻りコード	72	59. DESERV 出口の削除	292
23. SCRATCH 状況コード	73	60. DESERV 出口がアクティブかどうかの判別	292
24. 2 次状況コード	73	61. DESX 構造マッピング DESERV 出口パラメ	
25. CAMLST によって生成される RENAME パラ		ーター・リスト	293
メーター・リスト	78	62. DESERV GET 呼び出しの DESP の構造	295
26. DADSM RENAME 戻りコード	79	63. DESL 構造	297
27. RENAME 状況コード	79	64. DESN パラメーター・リスト	298
28. LOCATE 戻りコード	87	65. DESB パラメーター・リスト	299
29. CATALOG 戻りコード	93	66. SMDE フォーマット	299
30. 戻りコード	94	67. ディレクトリー項目名セクション	301
31. DCBOFLGS の使用法	110	68. ディレクトリー項目注釈リスト・セクション	
32. DCBE の存在を示す DCB ビット	121	(PDS のみ)	301
33. DCB DEVD オプション	121	69. ディレクトリー項目トークン・セクション	301
34. IOBE 構造のマッピング	132	70. ディレクトリー項目基本名セクション	301
35. IBDB 構造のマッピング	135	71. ディレクトリー項目名前セクション	302
36. EXCP 付加ルーチン	138	72. プログラム・オブジェクトに固有の LSLoader	
37. 付加ルーチンのエントリー・ポイント、戻		属性	306
り、および使用可能な作業レジスター	139	73. ロード・モジュールに固有の属性 (PDS のみ)	306
38. 相対アドレスから実アドレスに変換するため		74. 不定形式の別名	307
のレジスターとその使用法	151	75. DESERV PUT DESP フィールド	308
39. 相対変換ルーチンから実変換ルーチンへの戻		76. DESD パラメーター・リスト	311
りコード	152	77. DESERV DELETE DESP フィールド	312
40. 実アドレスから相対アドレスに変換するため		78. DESERV RENAME DESP フィールド	313
のレジスターとその使用法	153	79. DESERV UPDATE DESP フィールド	315
41. 実変換ルーチンから相対変換ルーチンへの戻		80. SYS1.IMAGELIB の内容	328
りコード	153	81. UCS5 イメージ・テーブルの内容	336
42. セクター変換ルーチンのレジスターとその使		82. UCS6 イメージ・テーブルの内容	337
用法	155	83. 3262 モデル 5 印刷バンド	338

84. BPX1PCT - 戻りコードと理由コード	363	88. ExtendFS コマンドの構造 (GFUMPCTL)	367
85. DisplayBufferLimits および ChangeBufferLimits コマンドの構造 (GFUMPCTL) 365	89. サポートされる拡張単位の定数 367
86. DisplayGlobalStats コマンドの構造 (GFUMPCTL) 365	90. DEB フィールドの部分リスト 427
87. DisplayFSStats コマンドの構造 (GFUMPCTL)	366	91. DFA フィールド 435
		92. DFA エlement名 438

本書について

文書は、システム・プログラマーがオペレーティング・システムのデータ管理機能を変更および拡張したり、プログラマーが拡張アプリケーション・プログラムを書くのを支援することを目的としています。

身体に障害のある方のための z/OS のアクセシビリティ機能については、439 ページの『付録 C. アクセシビリティ』を参照してください。

必要な製品知識

本書を効果的に使用するには、以下の知識が必要です。

- アセンブラー言語
- 標準プログラム・リンケージ規則
- ユーティリティー・プログラム IEHLIST および IEHPROGM
- データ管理アクセス方式およびマクロ命令
- DFSMSdss™ および DFSMSHsm™ によって提供されるストレージ管理機能。
DFSMSdss は、1 つの装置から別の装置にデータを移動し、データ・セットのバックアップとリカバリーを行い、DASD ボリューム上のフリー・スペースのフラグメント化を削減します。DFSMSHsm は、階層式のストレージ・デバイスを通してデータ・セットのマイグレーションおよび再呼び出しを行い、ストレージを管理します。

物理ストレージの管理については、「z/OS DFSMSdfp ストレージ管理リファレンス、SC88-8974」を参照してください。

参照される資料

本書では、以下の資料を参照しています。

表 1. 参照資料

資料名	資料番号
<i>IBM High Level Assembler/MVS & VM & VSE Programmer's Guide</i>	SC26-4941
<i>IBM High Level Assembler/MVS & VM & VSE Language Reference</i>	SC26-4940
<i>z/OS MVS プログラミング: アセンブラー・サービス ガイド</i>	SA88-8577
<i>z/OS MVS プログラミング: アセンブラー・サービス 解説書 ABE-HSP</i>	SA88-8578
<i>z/OS MVS Programming: Authorized Assembler Services Guide</i>	SA22-7608
<i>z/OS MVS Programming: Authorized Assembler Services Reference ALE-DYN</i>	SA22-7609
<i>z/OS MVS Programming: Authorized Assembler Services Reference ENF-IXG</i>	SA22-7610
<i>z/OS MVS Programming: Authorized Assembler Services Reference LLA-SDU</i>	SA22-7611

表 1. 参照資料 (続き)

資料名	資料番号
<i>z/OS MVS Programming: Authorized Assembler Services Reference SET-WTO</i>	SA22-7612
<i>z/OS MVS 計画: グローバル・リソース逐次化</i>	SA88-8572
<i>z/OS MVS プログラム管理: ユーザーズ・ガイドおよび解説書</i>	SA88-8688
<i>z/OS MVS プログラム管理: 拡張機能</i>	SA88-8689
<i>z/Architecture 解説書</i>	SA88-8773
<i>IBM 2821 Control Unit Component Description</i>	GA24-3312
<i>IBM 3203 Printer Component Description and Operator's Guide</i>	GA33-1515
<i>IBM 3211 Printer, 3216 Interchangeable Train Cartridge, and 3811 Printer Control Unit Component Description and Operator's Guide</i>	GA24-3543
<i>IBM 3262 Model 5 Printer Product Description</i>	GA24-3936
<i>IBM 3800 Printing Subsystem Programmer's Guide</i>	GC26-3846
<i>IBM 3800 Printing Subsystem Model 3 Programmer's Guide: Compatibility</i>	SH35-0051
<i>IBM 3800 Printing Subsystem Models 3 and 8 Programmer's Guide</i>	SH35-0061
<i>3900 Advanced Function Printer Product Description</i>	GA32-0135
<i>IBM 4245 Printer Model 1 Component Description and Operator's Guide</i>	GA33-1541
<i>IBM 4248 Printer Model 1 Description</i>	GA24-3927
<i>IBM 6262 Printer Model 014 Product Description</i>	GA24-4134
<i>IBM 6262 Printer Model 014 User's Guide</i>	GA24-4132
<i>IBM 6262 Printer Print Band Manual</i>	GA24-4131
<i>Device Support Facilities User's Guide and Reference</i>	GC35-0033
<i>z/OS MVS JCL 解説書</i>	SA88-8569
<i>z/OS MVS JCL ユーザーズ・ガイド</i>	SA88-8570
<i>z/OS JES2 初期設定およびチューニング ガイド</i>	SA88-8533
<i>z/OS JES3 初期設定およびチューニング ガイド</i>	SA88-8543
<i>z/OS DFSMS カタログのためのアクセス方式サービス・プログラム</i>	SC88-9109
<i>z/OS DFSMSdfp 診断解説書</i>	GY88-8507
<i>z/OS DFSMS Installation Exits</i>	SC26-7396
<i>z/OS DFSMS Macro Instructions for Data Sets</i>	SC26-7408
<i>z/OS DFSMS データ・セットの使用法</i>	SC88-9114
<i>z/OS DFSMS Using Magnetic Tapes</i>	SC26-7412
<i>z/OS DFSMSdfp ユーティリティ</i>	SC88-8979
<i>MVS/ESA SML: Managing Data</i>	SC26-3124
<i>z/OS DFSMS システム管理ストレージのインプリメンテーション</i>	SC88-9113
<i>z/OS MVS 初期設定およびチューニング ガイド</i>	SA88-8563
<i>z/OS ハードウェア構成定義(HCD) 計画</i>	GA88-8529
<i>z/OS Security Server RACF ユーザーズ・ガイド</i>	SA88-8615
<i>z/OS MVS システム・メッセージ 第 1 巻 (ABA-AOM)</i>	SA88-8597
<i>z/OS MVS システム・メッセージ 第 2 巻 (ARC-ASA)</i>	SA88-8598

表 1. 参照資料 (続き)

資料名	資料番号
z/OS MVS システム・メッセージ 第 3 巻 (ASB-BPX)	SA88-8599
z/OS MVS システム・メッセージ 第 4 巻 (CBD-DMO)	SA88-8600
z/OS MVS システム・メッセージ 第 5 巻 (EDG-GFS)	SA88-8601
z/OS MVS システム・メッセージ 第 6 巻 (GOS-IEA)	SA88-8602
z/OS MVS システム・メッセージ 第 7 巻 (IEB-IEE)	SA88-8603
z/OS MVS システム・メッセージ 第 8 巻 (IEF-IGD)	SA88-8604
z/OS MVS システム・メッセージ 第 9 巻 (IGF-IWM)	SA88-8605
z/OS MVS システム・メッセージ 第 10 巻 (IXC-IZP)	SA88-8606
z/OS TSO/E コマンド解説書	SA88-8628

インターネットの z/OS DFSMS 資料へのアクセス

CD-ROM でソフトコピー文書を提供しているほかに、IBM では、インターネットでも、ライセンス不要な z/OS ソフトコピー文書にアクセスできるようにしています (ただし英語版のみ)。z/OS 資料を表示し、検索し、さらに印刷するには、z/OS インターネット・ライブラリーを参照します。

www.ibm.com/servers/eserver/zseries/zos/bkserv/

LookAt を使用してメッセージの説明を検索する

LookAt は、お客様が受け取る IBM® のほとんどのメッセージや、数々のシステムの異常終了およびコードに関する説明を検索できる、オンライン機能です。LookAt では、通常、該当メッセージの説明がただちに表示されるため、従来の方法よりも短時間で、必要な情報を検索することができます。

お客様は LookAt を使用して、以下のロケーションから z/OS®、z/VM®、および VSE の要素や機能についての IBM のメッセージの説明を検索することができます。

- インターネット。お客様は、<http://www.ibm.com/eserver/zseries/zos/bkserv/lookat/>にある LookAt Web サイトから、直接、IBM メッセージの説明をご利用できます。
- ご使用の z/OS TSO/E ホスト・システム。お客様は、z/OS または z/OS.e にコードをインストールして、TSO/E コマンド・ライン (例えば、TSO/E プロンプト、ISPF、または OMVS を実行する z/OS UNIX® システム・サービス) から LookAt を使用し、IBM メッセージの説明をご利用できます。
- ご使用の Windows® ワークステーション。お客様はコードをインストールして、Windows DOS コマンド・ラインから LookAt を使用し、「z/OS Collection」(SK3T-4269) 上の IBM メッセージの説明をご利用できます。
- ご使用のワイヤレス・ハンドヘルド・デバイス。お客様は、無線アクセスとインターネット・ブラウザを備えたハンドヘルド・デバイス (例えば、ポケット PC 用 Internet Explorer、Palm OS 用 Blazer または Eudora、あるいは Linux ハンドヘルド・デバイス用 Opera など) から LookAt Mobile Edition をご利用できます。LookAt Web サイトから、LookAt Mobile Edition へリンクしてください。

ご使用の「z/OS Collection」(SK3T-4269) のディスク、または LookAt Web サイトからホスト・システム、または Windows ワークステーションに LookAt をインストールするコードを入手できます (「ダウンロード」を選択して、要求に適合するプラットフォーム、リリース、コレクション、およびロケーションを選択します)。より詳細な情報については、ダウンロード・プロセス中に参照可能な LOOKAT.ME ファイルをご覧ください。

構文図の読み方

このライブラリーの全体を通して、プログラミングの構文図を示しています。キーワード・パラメーターは、定位置パラメーターの後に続くパラメーターです。特に指示がない限り、キーワード・パラメーターは任意の順序でコーディングできます。以下のリストは、構文図の解釈の仕方を示しています。

- 構文図は、メインパスの線に沿って、左から右、上から下の順に読みます。それぞれの構文図は、左方の 2 つの矢印で始まり、右方の向き合う 2 つの矢印で終了します。

▶▶ | 構文図 | ◀◀

- 構文図が 1 行より長い場合は、各行を 1 つの矢印で終了して継続させ、次の行を 1 つの矢印で開始します。

▶▶ | 最初の行 | ◀◀

▶▶ | 2 番目の行 | ◀◀

▶▶ | 最後の行 | ◀◀

- 必須のキーワードと値は、メインパスの線上に表示します。必須のキーワードと値は、コーディングしなければなりません。

▶▶—REQUIRED_KEYWORD—◀◀

相互に排他的な複数の必須キーワードまたは値が存在する場合は、英数字順に縦にスタックして表示します。

▶▶—REQUIRED_KEYWORD_OR_VALUE_1—◀◀
└─REQUIRED_KEYWORD_OR_VALUE_2─┘

- オプションのキーワードと値は、メインパスの線の下方に表示します。オプションのキーワードと値はコーディングしなくても構いません。

▶▶ └─KEYWORD─┘ ◀◀

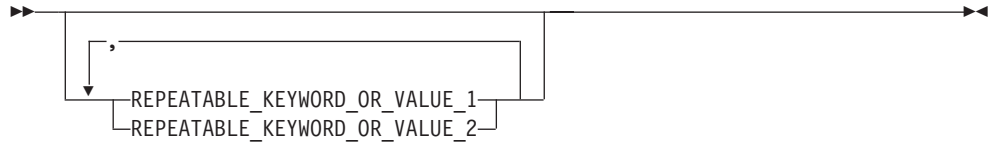
相互に排他的なオプションのキーワードまたは値が複数ある場合は、メインパスの線の下方に英数字順に縦にスタックして表示します。

▶▶ └─KEYWORD_OR_VALUE_1─┘ ◀◀
└─KEYWORD_OR_VALUE_2─┘

- メインパスの線上のキーワードまたは値の上方にある左に戻る矢印は、そのキーワードまたは値は反復できることを示しています。コンマは、各キーワードまたは値をコンマで区切る必要があることを示しています。



- キーワードまたは値のグループの上方にある左に戻る矢印は、複数を選択することも、1 つだけを繰り返すこともできることを示します。



- すべてが大文字のワードは、表示どおりに正確なスペルで入力する必要があるキーワードまたは値です。この例では、**KEYWORD** とコーディングする必要があります。



キーワードまたは値を省略できる場合、省略形についてはその構文図に関連した本文の中で説明します。

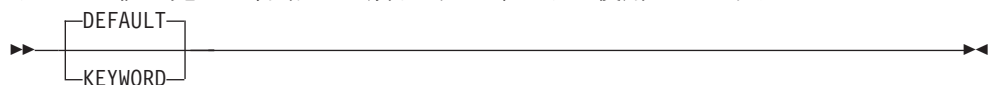
- 構文図の中に英数字以外の文字 (括弧、ピリオド、コンマ、等号など) が表示されている場合、その文字を構文の一部としてコーディングする必要があります。この例では、**KEYWORD=(001,0.001)** とコーディングしなければなりません。



- 構文図にブランク・スペースが表示されている場合、ブランク・スペースを構文の一部としてコーディングする必要があります。この例では、**KEYWORD=(001 FIXED)** とコーディングしなければなりません。



- デフォルトのキーワードと値は、メインパスの線の上方に表示します。キーワードまたは値を完全に省略した場合、デフォルトが使用されます。



- すべてが小文字のイタリック体で示されたワードは、変数 です。構文図の中に変数が表示されている場合、本文の中で定義されている許容される名前または値でそれを置き換える必要があります。



- 構文に関する注記への参照は、線の上方に括弧で囲んだ番号で表示します。この括弧や番号はコーディングしないでください。



注:

1 構文に関する注記の例。

- 一部の構文図には、構文フラグメント が含まれています。これは、長すぎたり、複雑すぎたり、反復が多すぎたりする構文図を分割する役目を果たします。構文

フラグメントの名前は、大/小文字混合で表され、構文図の中とフラグメントの見出しに表示されています。フラグメントは、主構文図の下に示されています。

→| 構文フラグメントへの参照 |←

構文フラグメント:

|—1ST_KEYWORD,2ND_KEYWORD,3RD_KEYWORD—|

アドレスおよびレジスタの規則

オペランドをコーディングするために使用する表記は、以下のフォーマットで表示します。

symbol

オペランドは、任意の有効なアセンブラ言語のシンボルを使用できます。

- (0) 汎用レジスタ 0 は、オペランドとして使用できます。マクロ命令内のオペランドとして使用する場合は、表示のように、レジスタを括弧で囲んだ 10 進数 0 として指定する必要があります。
- (1) 汎用レジスタ 1 は、オペランドとして使用できます。マクロ命令内のオペランドとして使用する場合は、表示のように、レジスタを括弧で囲んだ 10 進数 1 として指定する必要があります。レジスタ 1 を使用する場合、それをロードする命令は、マクロ展開には組み込まれません。

(2-12)

指定するオペランドは、汎用レジスタ 2 から 12 のいずれも可能です。オペランドとしてのレジスタは、括弧で囲んでコーディングする必要があります。例えば、レジスタ 3 をコーディングする場合は、**(3)** としてコーディングします。レジスタ 2 から 12 は、10 進数、シンボル (10 進数と等価)、または結果が 2 から 12 の値になる式としてコーディングできます。

RX-Type Address

オペランドは、以下の例に示すように、任意の有効なアセンブラ言語 RX タイプ・アドレスとして指定できます。

名前	演算	オペランド
ALPHA1	L	1,39(4,10)
ALPHA2	L	REG1,39(4,TEN)
BETA1	L	2,ZETA(4)
BETA2	L	REG2,ZETA(REG4)
GAMMA1	L	2,ZETA
GAMMA2	L	REG2,ZETA
GAMMA3	L	2,=F'1000'
LAMBDA1	L	3,20(,5)

ALPHA 命令は両方とも明示アドレスを指定します。REG1 と TEN は、絶対シンボルとして定義済みです。BETA 命令は両方とも暗黙アドレスを指定し、どちらも指標レジスタを使用します。ZETA は再配置可能シンボルです。指標付けは GAMMA 命令では省略されます。GAMMA1 と GAMMA2 は、暗黙アドレスを指定します。GAMMA3 の第 2 オペランドはリテラルです。

LAMBDA1 は指標を付けずに明示アドレスを指定します。

A-Type Address

オペランドは、任意の有効なアセンブラー言語 A タイプ・アドレス定数として書くことができる任意のアドレスを指定できます。A タイプ・アドレス定数は、絶対値、再配置可能シンボル、または再配置可能式として書くことができます。A タイプ・アドレスを必要とするオペランドは、マクロ展開処理時に A タイプ・アドレス定数に挿入されます。A タイプ・アドレス定数の詳細については、「*High Level Assembler/MVS & VM & VSE Language Reference*」を参照してください。

absexp

オペランドは、絶対値または式を使用できます。絶対式は、絶対項または絶対項の算術組み合わせ使用できます。絶対項には、再配置不可シンボル、自己定義項、または長さ属性参照を使用できます。絶対式の詳細については、「*High Level Assembler/MVS & VM & VSE Language Reference*」を参照してください。

relexp

オペランドは、再配置可能シンボルまたは式を使用できます。再配置可能シンボルまたは式を含むプログラムが、最初に割り当てられたストレージ域から n バイト離れた位置に再配置された場合、再配置可能シンボルまたは式の値は n だけ変更されます。再配置可能シンボルおよび式の詳細については、「*High Level Assembler/MVS & VM & VSE Language Reference*」を参照してください。

変更の要約

本書には、用語、細かな修正、および編集上の変更が含まれています。本文または図表に対して技術的な変更または追加が行われている場合には、その個所の左側に縦線を引いて示してあります。

z/OS バージョン 1 リリース 7 用 SD88-6462-01 での変更の要約

本書には、以前に「z/OS DFSMSdfp 拡張サービス, SD88-6462-00」に掲載されていた情報が含まれています。

以下の節で、その情報に対する変更内容を要約します。

新規情報

以下は、本書に含まれている新規情報の要約です。

- 107 ページの『無効な終了状況』が 95 ページの『第 4 章 ユーザー独自のチャンネル・プログラムの実行』に追加されました。

変更情報

以下は、本書に含まれている変更情報の要約です。

- 150 ページの『相対トラック・アドレスの実トラック・アドレスへの変換』および 152 ページの『実トラック・アドレスの相対トラック・アドレスへの変換』は、トラック変換ルーチンがラージ・フォーマット・データをサポートするように改良されたのを反映して更新されました。
- 168 ページの『PASSWORD データ・セットの特性』は更新され、PASSWORD データ・セットをラージ・フォーマットまたは拡張フォーマットにできないことが記載されています。
- 427 ページの『データ・エクステント・ブロック (DEB) フィールド』は更新され、以下が新しく追加されました。
 - ラージ・フォーマット・データ・セットを示す DS1FLAG1 内の DS1LARGE ビット。
 - DS1LSTAR にトラック番号の高位バイトを含むオフセット X'68' の DS1TTTHI バイト。 DS1LARGE がオンの場合に有効です。
 - 再呼び出しされるデータ・セットを示す DS1FLAG1 内の DS1RECAL ビット。
 - 最後の再呼び出し以降にデータ・セットが変更されたことを示す DS1DSIND 内の DS1IND08 ビット。
 - エクステントにある 3 バイトのトラック番号の高位バイトを含む、直接アクセス・ストレージ・デバイス・セクションのオフセット X'05' の DEBNMTRKHI バイト。低位の 2 バイトは DEBNMTRK 内にあります。
- 『DEB 直接アクセス・ストレージ・デバイス』の節のオフセット X'04' の DEBBINUM フィールドは廃止されました。

ライブラリーの変更

本書の一部の内容は、スタイルや構成が変更されています。例えば、見出しが作業本位になっている、注記の見出しが具体的で意図が分かりやすくなっている、情報検索を容易にするために索引項目が追加されている、手順の体裁や形式が異なっているなどです。

削除情報

この版から ISAM データ・セットのほとんどの参照が、オペランドで作成または開くことができるようになったため、削除されました。現在、ISAM データ・セットを作成したり開いたりすることはできませんが、それらの VTOC 情報を表示または削除することは可能です。その他の削除された項目は次のとおりです。

- Format-2 DSCB についての説明が 2 ページの『データ・セット制御ブロック (DSCB) のタイプ』から削除されました。このフォーマットは、ISAM データ・セットのみに適用されていました。
- **F2DSCB** パラメーターについての説明が、38 ページの『REALLOC - 実行形式』および 41 ページの『REALLOC - リスト形式』から削除されました。このパラメーターは現在もアセンブル可能ですが、プログラムの実行においてサポートされません。
- 180 ページの『DEBCHK マクロ仕様』で、ISAM が **AM** パラメーターの有効なアクセス方式の値リストから削除されました。

マイグレーションの操作については、「z/OS マイグレーション」を参照してください。

SD88-6462-00 (英文原典 : SC26-7400-04) での変更の要約

本書には、以前に「z/OS DFSMSdfp Advanced Services, SC26-7400-03」に掲載されていた情報が含まれています。

以下の節で、その情報に対する変更内容を要約します。

変更情報

- 369 ページの『第 13 章 サブシステムの統計、状況、およびカウント情報へのユーザー・アクセス』が更新され、このインターフェースを通して新規に LINK、RANK、および SEGMENT の統計を利用できます。
- CVAF パラメーター・リスト・マッピング・マクロ (CVPL) が更新され、追加フィールドが定義されました。45 ページの『CVPL (CVAF パラメーター・リスト) の生成』を参照してください。
- 渡される引数リスト SSGARGL が更新され、追加フィールドが定義されました。369 ページの『渡される引数リスト -- SSGARGL』を参照してください。
- CVAFDSM マクロに新規キーワード HADSCB (高位割り振り DSCB) が追加されました。392 ページの『CVAFDSM マクロの概要および仕様』を参照してください。

z/OS バージョン 1 リリース 6 用 SC26-7400-03 での変更の要約

本書には、以前に「z/OS DFSMSdfp Advanced Services, SD88-6462-02」に掲載されていた情報が含まれています。

以下の節で、その情報に対する変更内容を要約します。

新規情報

以下は、本書に含まれている新規情報の要約です。

- 369 ページの『第 13 章 サブシステムの統計、状況、およびカウント情報へのユーザー・アクセス』が追加されました。この情報は、以前は「3990/9390 Operations & Recovery Guide (GA32-0253-03)」に掲載されていました。
- 154 ページの『RPS 装置上のブロックのセクター番号の入手』は、前版では省略されていましたが、復元されました。
- 357 ページの『HFS 据え置きファイル・システム同期』が追加されました。

削除情報

ANTRQST マクロに関する情報は除去され、現在は「z/OS DFSMS Advanced Copy Services, SC35-0428」に含まれています。

変更情報

なし。

第 1 章 ボリューム目録の使用

この章は、ユーザーがシステム・マクロを使用してボリューム目録 (VTOC) および VTOC 索引にアクセスし、変更するのを支援することを目的としています。直接アクセス装置ストレージ管理 (DADSM) ルーチンは、ボリュームの VTOC を通して、また索引が存在する場合は VTOC 索引を通して、直接アクセス・ボリューム上のスペース割り振りを制御します。ストレージ管理者は、ICKDSF ユーティリティを使用して、これらの構造を構築します。65 ページの『VTOC および VTOC 索引の作成』を参照してください。

VTOC コンポーネント

VTOC は、常駐する直接アクセス・ボリュームの内容を記述するデータ・セットです。これは連続したデータ・セットです。つまり、ボリューム上の単一エクステンツト内にあり、シリンダー 0、トラック 0 の後、トラック 65,535 の前から開始します。VTOC のアドレスは、標準ボリューム・ラベルの VOLVTOC フィールドにあります (図 1 を参照)。ボリューム・ラベルについては、「*JOS DFSMS* データ・セットの使用法」で説明しています。VTOC は、完全なトラックで構成されます。

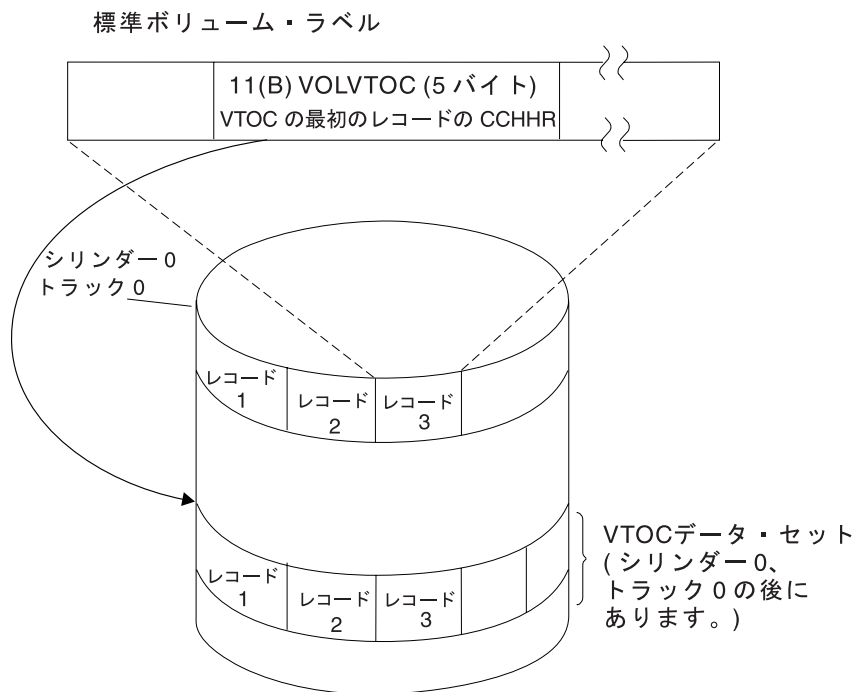


図 1. ボリューム目録の位置決め

VTOC は、140 バイト¹のデータ・セット制御ブロック (DSCB) から構成されます。これは、現在ボリューム上にあるデータ・セット、あるいはボリューム上の連続した未割り当てトラックのいずれかに対応します。

階層ファイル・システム (HFS) データ・セットには、特殊なタイプのデータが入ります。各データ・セットには、IEEE POSIX 標準および OSF XPG/4.2 標準に準拠した UNIX タイプのファイル・システムが入ります。一部の線形データ・セットには、z/OS ファイル・システム (zFS) ファイル・システムが入ります。それらのデータ・セットも上記の標準に準拠しています。HFS または zFS ファイル・システムを含む各データ・セットには、複数ユーザーが共有している UNIX ファイルおよびディレクトリを入れることができます。ファイルへのアクセスには、z/OS UNIX システム・サービス、BSAM、QSAM、および VSAM を使用します。「z/OS DFSMS データ・セットの使用法」を参照してください。データ・セットまたは VSAM データ・スペースの DSCB に、それらの特性と場所を記述します。連続した未割り当てトラックの DSCB では、それらの場所を示します。

データ・セット制御ブロック (DSCB) のタイプ

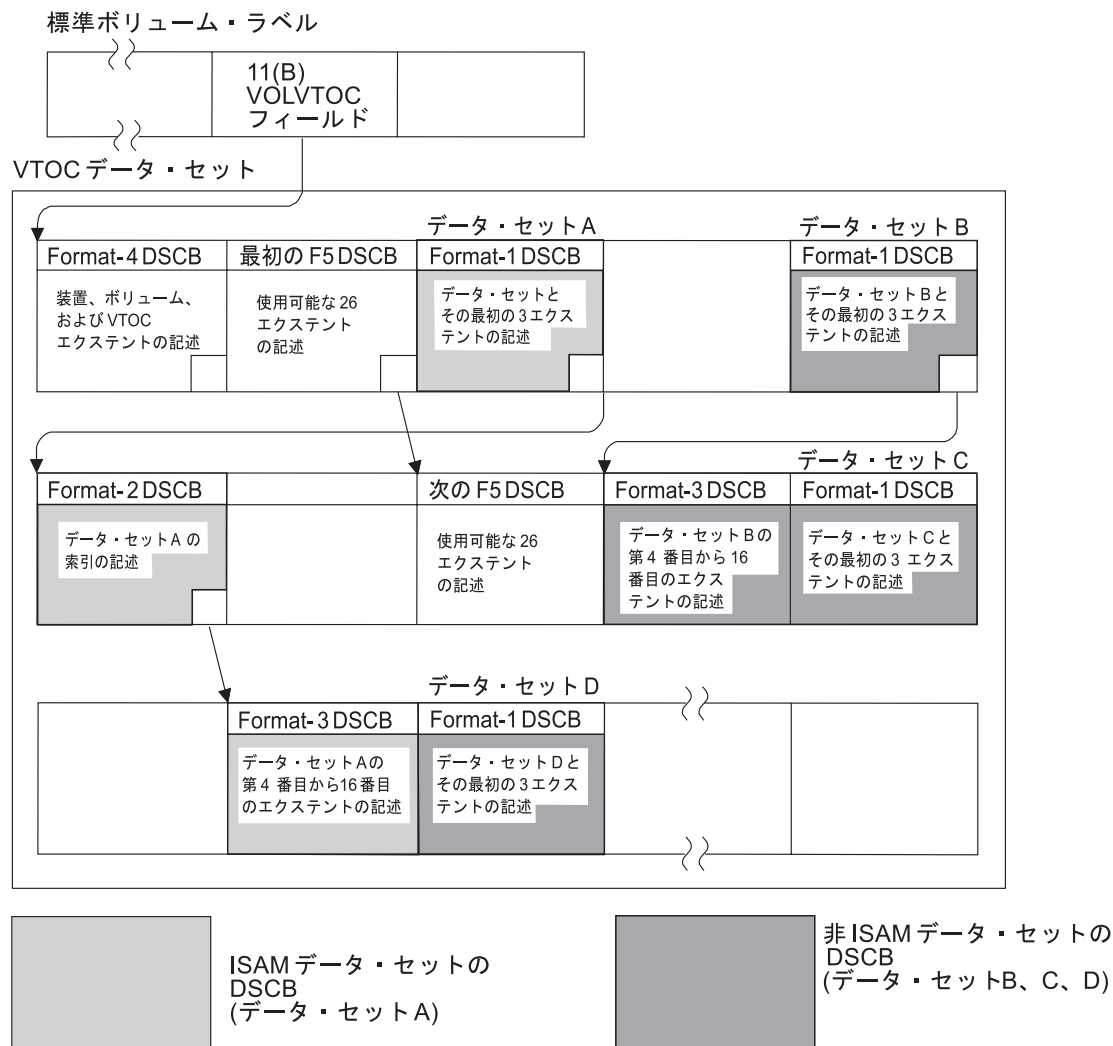
VTOC は、数種類の DSCB を含んでいます。ここでは、DSCB の用途、ボリューム上に存在する個数、およびその位置を見つける方法について説明します。DSCB のレイアウトを 5 ページの表 2 から 13 ページの表 5 に示します。

各 VTOC の最初のレコードは、VTOC DSCB (format-4) です。このレコードは、そのボリュームが常駐する装置、ボリュームの属性、および VTOC データ・セットのサイズと内容を記述します。VTOC データ・セットの次の DSCB は、フリー・スペース DSCB (format-5) です (フリー・スペースが format-7 DSCB によって記述される場合でも)。VTOC の 3 番目以降の DSCB は、任意の順序に配置できます。

VTOC 内の 1 つ以上の DSCB で、データ・セットがある各ボリュームのデータ・セットを定義します。データ・セットを定義するのに必要な DSCB の数は、データ・セットが占めるエクステントの数で決まります。3 エクステントを超えるデータ・セットには、1 つ以上の format-3 DSCB が必要です。

3 ページの図 2 には、4 つのデータ・セットを定義するために必要な VTOC と DSCB が示されています。データ・セット A (図 2 内) は 29 エクステントを含むため、基本フォーマット・データ・セット、直接データ・セット、または区分データ・セット (PDS) にはできません。エクステントの数が多いため、format-1 DSCB および 2 つの format-3 DSCB が必要となります。データ・セット B は 16 エクステントからなるため、format-1 および format-3 DSCB の両方が必要です。データ・セット C と D は、3 エクステント以下なので、format-1 DSCB のみ必要です。データ・セット B、C、および D は、いかなるタイプのデータ・セットでもかまいません。

1. 140 バイトは、44 バイトのキー部分とそれに続く 96 バイトのデータ部分に分けられます。ユーザーは、論理 140 バイト DSCB またはそのいずれかの部分を参照できます。



注: VTOC データ・セットの空のボックスは、フリー VTOC レコード (Format-0 DSCB) を表します。

図 2. VTOC の内容 - VTOC 索引がないボリュームのデータ・セットが記述されている DSCB

format-1、format-2、format-3、format-4、および format-5 DSCB のマッピング・マクロは、IECSDSL1 です。マップする DSCB のフォーマットを指定する定位置パラメーターをコーディングします。例えば、次の指定は、format-1 と format-3 をマップします。

```
IECSDSL1 1,3
```

マクロは DSECT ステートメントを生成しません。そのため、ユーザー独自の DSECT または CSECT に容易に組み込めます。次のような技法を使用して、別の DSECT ステートメントを入手できます。

```
F1AREA DSECT
        IECSDSL1 1
F3AREA DSECT
        IECSDSL1 3
```

各フォーマットで生成される最初のシンボルは IECSDSLn 形式です。ここで、n は、フォーマットの番号です。

Format-0 DSCB

名前: フリー VTOC レコード

機能: VTOC 内の未使用レコード (140 バイトの 2 進ゼロを含む) を記述します。VTOC から DSCB を削除するには、format-0 DSCB を上書きします。

個数: VTOC 上の未使用 140 バイト・レコードごとに 1 つ。format-4 DSCB の DS4DSREC フィールドは、VTOC 上の format-0 DSCB の個数です。索引付き VTOC の場合、このフィールドは保守されません。

検索方法: X'00' (場合によっては、X'00000000') に等しいキーに基づいて検索します。

Format-1 DSCB

名前: ID

機能: データ・セットまたは VSAM データ・スペースの最初の 3 エクステントを記述します。

個数: ボリューム上のデータ・セットまたはデータ・スペースごとに 1 つ (VTOC を除く)。

検索方法: OBTAIN マクロまたは CVAF マクロを使用します。

5 ページの表 2 は、format-1 DSCB の内容を示しています。

DS1REFD フィールドに関する注:

1. VSAM データ・セットの場合、FMT-1 DSCB の DS1REFD フィールドの日付が現在日付より前の場合には、OPEN は現在日付を使用してフィールドを更新しません。
2. マルチボリューム VSAM データ・セットの場合、OPEN は基本クラスターのデータ・コンポーネントの最初のボリュームの DS1REFD フィールドのみを更新します。
3. 非 SMS 管理の非 VSAM マルチボリューム・データ・セットの場合、OPEN は、OPEN が使用することを選択した最初のボリュームの DS1REFD フィールドを更新します。
4. 非拡張フォーマットの非 VSAM マルチボリューム SMS 管理データ・セットの場合、OPEN は、OPEN が使用することを選択した最初のボリュームおよびデータ・セットの最初のボリュームの DS1REFD フィールドを更新します。
5. 拡張フォーマットの単一ストライプ、非 VSAM データ・セットの場合、OPEN は、OPEN が使用することを選択した最初のボリュームおよびデータ・セットの最初のボリュームの DS1REFD フィールドを更新します。
6. 拡張フォーマットの複数ストライプ、非 VSAM データ・セットの場合、OPEN は、データ・セットのすべてのボリュームの DS1REFD フィールドを更新します。

表 2. DSCB Format-1

オフセット 10 進数 (16 進数)	タイプまたはビット・マスク	長さ	名前	説明
0(X'0')	文字	44	DS1DSNAM	データ・セット名。
44(X'2C')	文字	1	DS1FMTID	フォーマット ID (X'F1')。
45(X'2D')	文字	6	DS1DSSN	データ・セット通し番号 (データ・セット/スペースを含んでいる最初または唯一のボリュームを識別します)。
51(X'33')	符号なし	2	DS1VOLSQ	ボリューム・シーケンス番号。
53(X'35')	文字	3	DS1CREDIT	作成日 ('YDD')、不連続 2 進数。1900 と Y バイトの値を加算すると、年を算定できます。非 SMS 管理の VSAM データ・セットの場合、有効期限がカタログに示されています。
56(X'38')	文字	3	DS1EXPDT	有効期限 ('YDD')、不連続 2 進数。1900 と Y バイトの値を加算すると、年を算定できます。
59(X'3B')	符号なし	1	DS1NOEPV	ボリューム上のエクステンツ数。
60(X'3C')	符号なし	1	DS1NOBDB	最後のディレクトリー・ブロックに使用されたバイト数。
61(X'3D')	ビット・ストリング	1	DS1FLAG1	フラグ・バイト
	1		DS1COMPR	圧縮可能形式のデータ・セット (DS1STRP も 1)。
	. 1		DS1CPOIT	チェックポイント・データ・セット
	. . 1		DS1EXPBY	保存期間によって指定される VSE 有効期限 (現在、z/OS では使用されていません)
	. . . 1		DS1RECAL	再呼び出しされるデータ・セット。
 1		DS1LARGE	ラージ・フォーマット・データ・セット。
 x x x x			予約済み。
62(X'3E')	文字	13	DS1SYS CD	システム・コード。
75(X'4B')	文字	3	DS1REFD	最後の参照日 ('YDD' またはゼロ (保守されていない場合))。1900 と Y バイトの値を加算すると、年を算定できます。
78(X'4E')	ビット・ストリング	1	DS1SMSFG	システム管理ストレージ標識。
	1		DS1SMSDS	システム管理データ・セット。IEHLIST は、このビットを文字「S」として表示します。
	. 1		DS1SMSUC	アンカカタログ・システム管理データ・セット (システム管理ボリューム上のすべての一時データ・セットと同様に、VTOC 索引はアンカカタログ・システム管理データ・セットです)。IEHLIST は、このビットを文字「U」として表示します。
	. . 1		DS1REBLK	システムがブロック・サイズを決定し、データ・セットをブロック化し直すことができます (データ・セットのブロック化し直しは、ユーザーまたはシステムが行えます)。IEHLIST は、このビットを文字「R」として表示します。
	. . . 1		DS1CRSDB	DADSM が元のブロック・サイズを作成し、データ・セットは出力のためにオープンされていませんでした。IEHLIST は、このビットを文字「B」として表示します。
 1		DS1PDSE	データ・セットは、PDSE または HFS データ・セットです (HFS の場合、DS1PDSEX も 1)。PDSE の場合、IEHLIST は、このビットを文字「I」として表示します。無効なビットの組み合わせを検出した場合、IEHLIST は「?」を表示します。
 1		DS1STRP	順次拡張フォーマット・データ・セット。IEHLIST は、このビットを文字「E」として表示します。無効なビットの組み合わせを検出した場合、IEHLIST は「?」を表示します。
 1		DS1PDSEX	HFS データ・セット (DS1PDSE も 1 でなければなりません)。IEHLIST は、このビットを文字「H」として表示します。

VTOC の使用

表 2. DSCB Format-1 (続き)

オフセット 10 進 数 (16 進数)	タイプまたは ビット・マスク	長さ	名前	説明	
I 79(X'4F') 1	3	DS1DSAE	拡張属性はカタログ項目にあります。	
	文字		DS1SCEXT	2 次スペース拡張。DS1EXT がオンの場合にのみ有効です (オフセット 94(X'5E) を参照)。	
	ビット・ストリン グ		DS1SCXTF	2 次スペース拡張フラグ・バイト (最初の 4 ビットの 1 つのみがオン)。	
	1	DS1SCAVB	1 の場合、DS1SCXTV は元のブロック長です。0 の場合、DS1SCXTV は平均レコード長です。		
	. 1	DS1SCMB	1 の場合、DS1SCXTV は M バイト単位です。		
	.. 1	DS1SCKB	1 の場合、DS1SCXTV は K バイト単位です。		
	... 1	DS1SCUB	1 の場合、DS1SCXTV はバイト単位です。		
 1	DS1SCCP1	1 の場合、DS1SCXTV は 256 の係数で圧縮されました。		
 1 ..	DS1SCCP2	1 の場合、DS1SCXTV は 65,536 の係数で圧縮されました。		
	80(X'50')	符号なし	2	DS1SCXTV	平均レコード長または平均ブロック長の 2 次スペース拡張値。
	82(X'52')	ビット・ストリン グ	2	DS1DSORG	データ・セット編成。
		DS1DSORG の最初 のバイト。			
		1000 000x	DS1DSGIS	索引順次編成。	
		0100 000x	DS1DSGPS	物理順次編成。	
		0010 000x	DS1DSGDA	直接編成。	
0001 000x		DS1DSGCX	BTAM または QTAM 回線グループ。		
.... xx ..			予約済み。		
0000 001x		DS1DSGPO	区分編成。		
..... 1		DS1DSGU	移動不能。データは位置依存情報を含んでいます。		
DS1DSORG の 2 番目のバイト					
100x 00xx		DS1DSGGS	グラフィックス編成。		
010x 00xx		DS1DSGTX	TCAM 回線グループ。		
001x 00xx		DS1DSGTQ	TCAM メッセージ・キュー。		
000x 10xx		DS1ACBM	VSAM データ・セット/スペース。		
000x 10xx		DS1ORGAM	VSAM データ・セット/スペース。		
000x 01xx	DS1DSGTR	TCAM 3705。			
84(X'54')	文字	1	DS1RECFM	予約済み。	
10	DS1RECFE		レコード・フォーマット。		
01	DS1RECFV		固定長。		
11	DS1RECFU		可変長。		
.. 1	DS1RECFW		未定義長。		
... 1	DS1RECFX		トラック・オーバーフロー。		
.... 1	DS1RECFB		ブロック化。未定義では行えません。		
..... 1	DS1RECFB		固定長。標準ブロック。最後のブロックおよびトラックを除いて、切り捨てられたブロックや未充てんのトラックはありません。可変長: スパン・レコード。		
..... 00 ..			制御文字なし。		
..... 10 ..	DS1RECFA		ISO/ANSI 制御文字。		
..... 01 ..	DS1RECMC	マシン制御文字。			
..... 11 ..		予約済み。			
..... x		予約済み。			
85(X'55')	文字	1	DS1OPTCD	オプション・コード。	
BDAM OPTCD フィールド割り当て (DS1DSGDA がオンの場合にのみ適用):					
1				書き込み妥当性検査。	
. 1				トラック・オーバーフロー。	
.. 1				拡張検索。	

表 2. DSCB Format-1 (続き)

オフセット 10 進数 (16 進数)	タイプまたはビット・マスク	長さ	名前	説明
	... 1			フィードバック。
 1			実アドレッシング。
 1 . . .			動的バッファ方式。
 1 . .			排他的読み取り。
 1 .			相対ブロック・アドレッシング。
ISAM OPTCD	フィールド割り当て (DS1DSGIS がオンの場合にのみ適用):			
	1			書き込み妥当性検査。
	. 1			累算トラック索引項目。
	. . 1			マスター索引。
	. . . 1			独立オーバーフロー域。
 1			シリンダー・オーバーフロー域。
 1 . . .			予約済み。
 1 . .			削除オプション。
 1 .			再編成基準。
BPAM, BSAM, QSAM OPTCD	フィールド割り当て (DS1DSGPO または DS1DSGPS がオンの場合にのみ適用):			
	1			書き込み妥当性検査。
	. 1			データ・チェック許可 (プリンター上の場合)。
	. . 1			チェーン・スケジューリング。
	. . . 1			VSE/MVS 交換機能 (テープ上)。
 1			EOF を EOVS として扱う (テープ)。
 1 . . .			直接検索。
 1 . .			ユーザー・ラベル合計。
 1 .			各レコードは、テーブル参照文字を含んでいます。
85(X'55')	ビット・ストリング	1	DS1OPTAM	VSAM OPTCD 設定。
VSAM OPTCD	フィールド割り当て (DS1ORGAM がオンの場合にのみ適用):			
	1		DS1OPTIC	データ・セットは、統合カタログ機能カタログにカタログされています。
	. 1		DS1OPTBC	データ・セットは、統合カタログ機能カタログです。
	. . xx xxxx			予約済み。
86(X'56')	2 進数	2	DS1BLKL	ブロック長 (タイプ F 非ブロック化レコード)、または最大ブロック・サイズ (F ブロック化、U または V レコード)。
88(X'58')	2 進数	2	DS1LRECL	論理レコード長。固定長レコード長、未定義長ゼロ、可変非スパン最大レコード長、可変スパンおよび < 32757 バイト最大レコード長、可変スパンおよび > 32756 バイト - X'8000'。
90(X'5A')	2 進数	1	DS1KEYL	キー長 (0 から 255)。
91(X'5B')	2 進数	2	DS1RKP	相対キー位置。
93(X'5D')	文字	1	DS1DSIND	データ・セット標識。
	1		DS1IND80	このデータ・セットのデータを含んでいる最後のボリューム。
	. 1		DS1IND40	データ・セットは、個別プロファイルで定義された RACF™ (Security Server for z/OS のコンポーネント) です。
	. . 1		DS1IND20	ブロック長は、8 バイトの倍数です。
	. . . 1		DS1IND10	パスワードは、読み取りまたは書き込み (あるいは、その両方) が必要です。DS1IND04 を参照。
 1		DS1IND08	最後の再呼び出し以降にデータ・セットが変更されました。
 1 . . .		DS1IND04	DS1IND10 が 1 で、DS1IND04 が 1 の場合、パスワードは書き込みに必要ですが、読み取りには不要です。DS1IND10 が 1 で、DS1IND04 が 0 の場合、パスワードは書き込みと読み取りの両方に必要です。

VTOC の使用

表 2. DSCB Format-1 (続き)

オフセット 10 進数 (16 進数)	タイプまたはビット・マスク	長さ	名前	説明
 1 .		DS1IND02	最後のバックアップ・コピーが作成された後、入力以外の目的でデータ・セットがオープンされました。
			DS1DSCHA	DS1IND02 と同じ。
 1		DS1IND01	セキュア・チェックポイント・データ・セット。
			DS1CHKPT	DS1IND01 と同様。
94(X'5E')	2 進数	4	DS1SCALO	2 次割り振りスペース・パラメーター。
94(X'5E')	文字	1	DS1SCAL1	フラグ・バイト。
	xxxx xxxx		DS1DSPAC	スペース要求ビット。次のように定義されます。
	00 . 0		DS1DSABS	絶対トラック要求。
	00 . 1		DS1EXT	2 次スペースへの拡張が存在します (オフセット 79 (X'4F') の DS1SCEXT を参照)。
	11		DS1CYL	シリンダー要求。
	10		DS1TRK	トラック要求。
	01 x		DS1AVR	平均ブロック長要求。
	01 1		DS1AVRND	平均ブロックおよび丸め要求。
	. . . 1		DS1MSGP	大容量記憶ボリューム・グループ (MSVGP - サポートされなくなりました)。
 1		DS1CONTG	連続要求。
 1		DS1MXIG	MXIG 要求。
 1		DS1ALX	ALX 要求。
 1		-	丸め要求。
95(X'5F')	2 進数	3	DS1SCAL3	2 次割り振り数量。
98(X'62')	2 進数	3	DS1LSTAR	最後に使用されたトラックとトラック上のブロック (TTR)。VSAM、PDSE、HFS、および直接 (BDAM) の場合は定義されません。+61 のビット DS1LARGE および +104 のバイト DS1TTTHI を参照。
101(X'65')	2 進数	2	DS1TRBAL	拡張フォーマットでない場合、これは使用された最後のトラックの残りスペースを示す TRKCALC からの値です。拡張フォーマットのデータ・セットの場合、これは使用された最後のトラック番号の 4 バイトの高位 2 バイト (TT) です。DS1LSTAR を参照。VSAM、PDSE、および HFS の場合は、ゼロです。
103(X'67')	文字	1		予約済み。
104(X'68')	文字	1	DS1TTTHI	DS1LSTAR 内のトラック番号の高位バイト。DS1Large がオンの場合は有効です。
105(X'69')	文字	30	DS1EXNTS	3 エクステントのフィールド。
105(X'69')	文字	10	DS1EXT1	最初のエクステントの記述。
	文字	1		エクステント・タイプ標識。
	X'81'			シリンダー境界のエクステント。
	X'80'			記述されたエクステントはシリンダーを共用しています (サポートされなくなりました)。
	X'40'			最初のエクステントはユーザー・ラベルを記述し、DS1NOEPV にはカウントされません。
	X'04'			索引域エクステント (ISAM)。
	X'02'			オーバーフロー域エクステント (ISAM)。
	X'01'			ユーザーのデータ・ブロック・エクステント、または基本区域エクステント (ISAM)。
	X'00'			これはエクステントではありません。
		1		エクステント・シーケンス番号。
		4		下限 (CCHH)。CC バイトは、16 ビット符号なし 2 進値として扱ってください。
		4		上限 (CCHH)。
115(X'73')	文字	10	DS1EXT2	2 番目のエクステントの記述。
125(X'7D')	文字	10	DS1EXT3	3 番目のエクステントの記述。

表 2. DSCB Format-1 (続き)

オフセット 10 進 数 (16 進数)	タイプまたは ビット・マスク	長さ	名前	説明
135(X'87')	文字	5	DS1PTRDS	format-2 または format-3 DSCB へのポインター (CCHHR)、またはゼロ。
140(X'8C')	文字		DS1END	-

Format-2 DSCB

このフォーマットは、ISAM データ・セットのみに適用されていました。現在、作成およびオープンは不可能です。

Format-3 DSCB

名前: 拡張

機能: 非 VSAM データ・セットまたは VSAM データ・スペースの 3 番目のエクステントの後のエクステントを記述します。

個数: 3 エクステントを超えるボリューム上のデータ・セットまたは VSAM データ・スペースごとに 1 つ。統合カタログ機能カタログにカタログされている PDSE、HFS、拡張フォーマット順次データ・セット、または VSAM データ・セットの場合、最大 10 個まで可能です。PDSE、HFS、および拡張フォーマット・データ・セットは、最大 123 エクステントを含むことができます。統合カタログ機能カタログにカタログされた VSAM データ・セットの各コンポーネントは、ボリューム当たり最大 123 エクステントを含むことができます。その他のデータ・セットはすべて、ボリューム当たり 16 エクステントに制限されています。

検索方法: データ・セットを表す format-1 または format-2 DSCB からチェーニングされます。PDSE、HFS データ・セット、拡張フォーマット・データ・セット、または VSAM データ・セットの場合、先行する format-3 DSCB からチェーニングされる場合があります。

表 3 は、format-3 DSCB の内容を示しています。

表 3. DSCB Format-3

オフセット 10 進数 (16 進数)	タイプ	長さ	名前	説明
0(X'00')	ビット・ストリ ング	4	-	キー ID (X'03030303')。
4(X'04')	ビット・ストリ ング	40	DS3EXTNT	4 エクステントの記述。
		1		エクステント・タイプ標識。
		1		エクステント・シーケンス番号。
		4		下限 (CCHH)。
		4		上限 (CCHH)。
44(X'2C')	文字	1	DS3FMTID	フォーマット ID (X'F3')。
45(X'2D')	ビット・ストリ ング	90	DS3ADEXT	追加 9 エクステントの記述。
135(X'87')	ビット・ストリ ング	5	DS3PTRDS	次の format-3 DSCB へのポインター (CCHHR)、ま たはゼロ。

VTOC の使用

表 3. DSCB Format-3 (続き)

オフセット 10 進数 (16 進数)	タイプ	長さ	名前	説明
140(X'8C')			DS3END	-

Format-4 DSCB

名前: VTOC

機能: VTOC のエクステンと内容を記述し、ボリュームと装置の特性を提供します。この DSCB には、ボリュームが SMS 管理であるかどうかを示すフラグを含んでいます。

個数: 各ボリューム上に 1 つ。

検索方法: 標準ボリューム・ラベルの VOLVTOC フィールドは、format-4 アドレスを含んでいます。format-4 DSCB は、常に VTOC の最初のレコードです。

表 4 は、format-4 DSCB の内容を示しています。

例外: format-4 DSCB は、X'04' バイトの 44 バイト・キーを持っています (表 4 には示されていません)。

表 4. DSCB Format-4

オフセット 10 進数 (16 進数)	タイプ	長さ	名前	説明
0(X'00')	文字	1	DS4IDFMT	フォーマット ID (X'F4')。
1(X'01')	文字	5	DS4HPCHR	format-1 DSCB の最高アドレス (CCHHR)。
6(X'06')	符号なし	2	DS4DSREC	使用可能な DSCB の数。
8(X'08')	文字	4	DS4HCCHH	次の使用可能な代替トラックの CCHH。
12(X'0C')	符号なし	2	DS4NOATK	残りの代替トラックの数。
14(X'0E')	ビット・ストリング	1	DS4VTOCI	VTOC 標識。
	1		DS4DOSBT	VSE ビット。無効な format-5 DSCB または索引付き VTOC のいずれか。以前の DOS(VSE) ビット。DS4IVTOC を参照。
	. 1		DS4DVTOC	索引は使用不可にされました。
	. . 1		DS4EFVLD	拡張フリー・スペース管理フラグ。DS4EFVLD がオンのとき、ボリュームは OSVTOC フォーマットで、有効なフリー・スペース情報が format-7 DSCB に入ります。DS4EFLVL と DS4EFPTR も参照。
	. . . 1		DS4DSTKP	VSE スタック・パック。
 1		DS4DOCVT	VSE 変換 VTOC。
 1		DS4DIRF	DIRF ビット。VTOC 変更は未完了です。
 1		DS4DICVT	DIRF が再利用されました。
 1		DS4IVTOC	ボリュームは索引付き VTOC を使用します。
15(X'0F')	符号なし	1	DS4NOEXT	VTOC(X'01') 内のエクステンとの数。
16(X'10')	ビット・ストリング	1	DS4SMSFG	システム管理ストレージ標識。
	00		DS4NTSMS	非システム管理ボリューム。
	01		DS4SMSCV	初期状態のシステム管理ボリューム。

表 4. DSCB Format-4 (続き)

オフセット 10				
進数 (16 進数)	タイプ	長さ	名前	説明
	10			予約済み。
	11		DS4SMS	システム管理ボリューム。
	11		DS4SMSTS	システム管理ボリューム。
	. . xx xxxx			予約済み。
17(X'11')	2 進数	1	DS4DEVAC	ボリュームのフォーマット時の代替シリンダーの数。DS4DEVSZ の最初の 2 バイトから減算して、使用可能なシリンダー数を入力します (0 の場合もあります)。DS4DEVAV がオンの場合にのみ有効です。
18(X'12')	文字	14	DS4DEVCT	装置定数。現行のサポートされる装置の場合、このフィールドはトラック上で使用されたスペース量を計算するための十分な情報を提供しません。TRKCALC を使用してください (229 ページの『トラック計算の実行』を参照)。
18(X'12')	文字	4	DS4DEVSZ	装置サイズ。
18(X'12')	2 進数	2	DS4DSCYL	代替 (ある場合) を含めた論理シリンダーの数。符号なし数値。DS4DEVAC の説明を参照。
20(X'14')	2 進数	2	DS4DSTRK	論理シリンダーのトラックの数。
22(X'16')	符号なし	2	DS4DEVTK	装置のトラックの長さ。
24(X'18')	2 進数	2	DS4DEV OV	キー付きレコード・オーバーヘッド。
24(X'18')	2 進数	1	DS4DEVI	最後でないキー付きレコード・オーバーヘッド。
25(X'19')	2 進数	1	DS4DEVL	最後のキー付きレコード・オーバーヘッド。
26(X'1A')	2 進数	1	DS4DEVK	キーなしレコード・オーバーヘッド差分。
27(X'1B')	ビット・ストリング	1	DS4DEVFG	フラグ・バイト。
	. . . 1		DS4DEVAV	DS4DEVAC 内に値。代替シリンダーの数は有効です。そうでない場合、この DSCB 内に数値は得られません。
 1 . . .			キー付きレコード・オーバーヘッド・フィールド (DS4DEV OV) は、キー付きレコードに必要なオーバーヘッドを指定するための 2 バイト・フィールドとして使用されます。
 1 . .			絶対アドレスの CCHH は、連続した 2 進値として使用されます。
 1 .			絶対アドレスの CCHH は、4 つの分離した 2 進値として使用されます。
 1			許容度係数は、トラックの最終ブロック以外のすべてのブロックに適用する必要があります。
	xxxx			予約済み。
28(X'1C')	2 進数	2	DS4DEVTL	装置の許容度。
30(X'1E')	2 進数	1	DS4DEVDT	トラック当たりの DSCB の数。
31(X'1F')	2 進数	1	DS4DEVDB	トラック当たりの PDS ディレクトリー・ブロックの数。
32(X'20')	2 進数	8	DS4AMTIM	VSAM タイム・スタンプ。
40(X'28')	文字	3	DS4AMCAT	VSAM カタログ標識。
40(X'28')	ビット・ストリング	1	DS4VSIND	VSAM 標識。
	1		DS4VSREF	VSAM カタログはこのボリュームを参照します。

VTOC の使用

表 4. DSCB Format-4 (続き)

オフセット 10 進数 (16 進数)	タイプ	長さ	名前	説明
	. 1		DS4VSBAD	ボリュームの MSS CONVERTV コマンドが正常に完了しなかったので、このボリュームの VSAM データ・セットは利用不可です。(もう設定されていません。)
	. . xx xxxx			予約済み。
41(X'29')	符号なし	2	DS4VSCRA	CRA の相対トラック位置。
43(X'2B')	2 進数	8	DS4R2TIM	VSAM ボリューム/カタログ突き合わせタイム・スタンプ。
51(X'33')	文字	5		予約済み。
56(X'38')	文字	5	DS4F6PTR	最初の format-6 DSCB へのポインター、またはゼロ。(ゼロ以外はサポートされなくなりました。)
61(X'3D')	文字	10	DS4VTOCE	VTOC エクステンツの記述。
71(X'47')	文字	10		予約済み。
81(X'51')	文字	1	DS4EFLVL	VTOC エクステンツの記述。
				拡張フリー・スペース管理レベル。X'00' は、このボリュームには拡張フリー・スペース管理は使用されないことを示します。X'07' は、このボリュームには拡張フリー・スペース管理が使用されることを示します (DS4EFVLD も参照)。
82(X'52')	文字	5	DS4EFPTR	拡張フリー・スペース情報へのポインター。
				DS4EFLVL=X'00' の場合、これはゼロです。
				DS4EFLVL=X'07' の場合、これは最初の FMT-7 DSCB への CCHHR で、format-5 DSCB にはフリー・スペース情報は入りません。
87(X'57')	文字	9		予約済み。
96(X'60')			DS4END	-

Format-5 DSCB

名前: フリー・スペース

機能: 索引なし VTOC では、データ・セットに割り当てられていないボリューム上のスペース (使用可能なスペース) を記述します。索引付き VTOC の場合、VTOC 内に単一の空の format-5 DSCB があります。フリー・スペースは索引に記述され、DS4IVTOC は通常はオンです。

個数: 索引なし VTOC の場合、ボリューム上の使用可能なスペースの 26 不連続エクステンツごとに 1 つあります。索引付き VTOC の場合は、1 つだけ存在します。

検索方法: ボリュームの最初の format-5 DSCB は、常にその VTOC の 2 番目の DSCB です。複数の format-5 DSCB がある場合は、DS5PTRDS フィールドを使用して、直前の format-5 DSCB からチェーニングされます。

13 ページの表 5 は、format-5 DSCB の内容を示しています。

表 5. DSCB Format-5

オフセット 10 進数 (16 進数)	タイプ	長さ	名前	説明
0(X'00')	ビット・ストリ ング	4	DS5KEYID	キー ID (X'05050505')。
4(X'04')	ビット・ストリ ング	5	DS5AVEXT	使用可能なエクステント。
		2		エクステントの最初のトラックの相対トラック・アドレス。ボリュームの開始に対する相対値です。
		2		エクステント内の未使用シリンダーの数。
		1		追加の未使用トラックの数。
9(X'09')	ビット・ストリ ング	35	DS5EXTAV	7 つの使用可能なエクステント。
44(X'2C')	文字	1	DS5FMTID	フォーマット ID (X'F5')。
45(X'2D')	ビット・ストリ ング	90	DS5MAVET	18 の使用可能なエクステント。
135(X'87')	ビット・ストリ ング	5	DS5PTRDS	次の format-5 DSCB へのポインタ (CCHHR)、またはゼロ。
140(X'8C')			DS5END	-

Format-7 DSCB

名前: 特定の装置のフリー・スペース。

format-7 DSCB の 1 つのフィールドのみが、意図されたインターフェースです。このフィールドは、DSCB が format-7 DSCB かどうかを示します。ユーザーはそのフィールドを DS1FMTID または DS5FMTID として参照できます。文字 7 は DSCB が format-7 DSCB であることを示し、ユーザーのプログラムでそれを変更してはなりません。

問題を診断する場合は、「z/OS DFSMSdfp 診断解説書」に記載されている Format-7 DSCB のレイアウトを参照してください。

DASD スペースの割り振りと解放

DADSM 割り振りルーチンおよび拡張ルーチンは、データ・セットの直接アクセス・ボリューム上にトラックとシリンダーを割り当てます。DADSM 拡張ルーチンは、1 次割り振りを越えたデータ・セットのために追加スペースを取得します。DADSM スクラッチ・ルーチンおよび部分リリース・ルーチンは、直接アクセス・ボリューム上の不要になったスペースを解放します。

DADSM ルーチンは、DSCB を追加、削除、および変更し、VTOC 索引を更新することによって、スペースを割り振ったり、解放したりします。スペースが解放されると、スクラッチ・ルーチンは、削除されたデータ・セットまたはデータ・スペースの DSCB を解放します。

VTOC 索引

VTOC 索引は、VTOC アクセスのパフォーマンスを向上します。VTOC 索引は、関連 VTOC と同じボリューム上にある物理順次データ・セットです。これは VTOC に含まれている format-1 DSCB 内のデータ・セット名の索引と、ボリューム・フリー・スペース情報から構成されます。

SMS 管理ボリュームは、索引付き VTOC が必要です。あるいは、VTOC 索引の使用を強くお勧めします。SMS 管理ボリュームの追加情報については、「z/OS DFSMS システム管理ストレージのインプリメンテーション」を参照してください。

注: REFORMAT REFVTOC コマンドを使用して、VTOC 索引を再作成することにより、必要なくなった索引スペースを再利用し、アクセス時間の向上を図ることができます。

MVS™ は、VTOC 索引を含む非 SMS 管理ボリュームを、非 MVS システムと共用することをサポートします。他のシステムが VTOC を更新して DS4DOSBT をオンにした場合、DADSM は、後で MVS を使用して VTOC を変更するときに、索引が有効でなくなっていることを検出することができます。

装置サポート機能 (ICKDSF) は、VTOC 索引を、VTOC 索引レコード (VIR) という名前の 2048 バイトの物理ブロックとして初期設定します。VIR はいくつかの方法で使用されます。VTOC 索引には、以下の種類の VIR が含まれます。

VTOC 索引項目レコード (VIER) は、format-1 DSCB と format-4 DSCB の位置を識別します。

VTOC パック・スペース・マップ (VPSM) は、ボリューム上のフリー・スペースと割り振りスペースを識別します。

VTOC 索引マップ (VIXM) は、VTOC 索引に割り振られた VIR を識別します。

VTOC の DSCB マップ (VMDS) は、VTOC に割り振られた DSCB を識別します。

VTOC の format-1 DSCB は、VTOC 索引の名前とエクステンション情報を含んでいます。索引名は 'SYS1.VTOCIX.xxxxxxxx' でなければなりません。ここで、'xxxxxxx' は、標準データ・セット命名規則に準拠し、通常は VTOC とその索引を含んでいるボリュームの通し番号です。ENQ の競合を避けるために、名前はシステム内で固有でなければなりません。

1 つのボリューム上には、名前が「SYS1.VTOCIX.」で始まるデータ・セットは 1 つしか作成 (割り振り) できません。VTOC がアクティブのときに VTOC 索引データ・セットを名前変更するには、「SYS1.VTOCIX.」で始まる名前を使用します。「SYS1.VTOCIX.」データ・セットがすでに存在する場合、そのボリューム上の別のデータ・セットをこの修飾子を持つ名前に変更することはできません。VTOC 索引がアクティブの場合、VTOC 索引データ・セットを消去することはできません。

VTOC とその索引の関係を 15 ページの図 3 に示します。

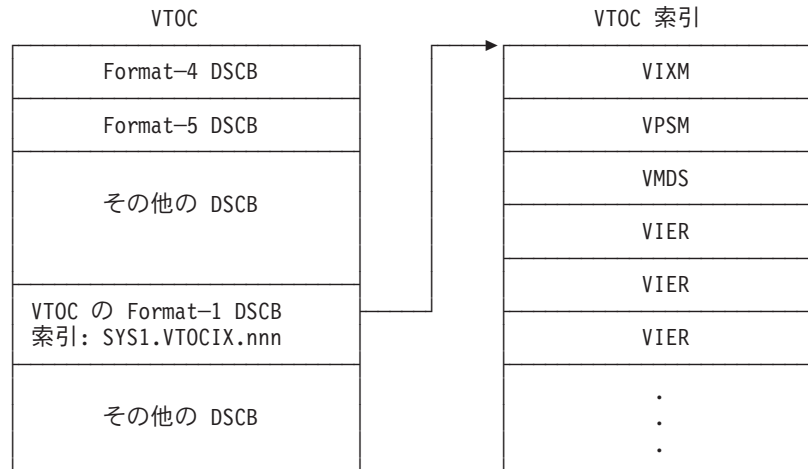


図 3. VTOC とその索引の関係の例

VTOC 索引レコード

VTOC 索引レコードは、以下のタイプで構成されます。

VTOC 索引項目レコード

最初のレベル 1 VIER は、VTOC 索引を使用して作成されます。以降の VIER は、VIER が満杯になってデータ・セット名を VTOC 索引に追加できなくなるたびに作成されます。VIER には、次のような特性があります。

- VIER は、1 つの VIR を使用し、可変長の索引項目を含みます。索引の VIER の数は、ボリューム上のデータ・セットの数によって決まります。
- VIER 内の索引項目は、すべて同じ索引レベルです。VTOC 索引内の VIER は、複数のレベルが可能で、階層関係を持つことができます。上位の VIER 内の索引項目は、下位の VIER を指します。レベル 1 VIER (最下位) の索引項目は、ボリューム上のデータ・セットの format-1 DSCB を指します。
- 同じレベルに 4 番目の VIER が作成されるたびに、上位に VIER が 1 つ作成されます。上位 VIER が下位 VIER へのポインターで満杯になると、同じレベルに別の VIER が作成されます。

VTOC パック・スペース・マップ

VPSM は、ボリューム上の割り振りスペースと残りのフリー・スペースを表示します。マップは、ボリューム上のシリンダーとトラックのビットマップを含みます。1 の値は、シリンダーまたはトラックが割り振り済みであることを示します。ゼロは、未割り振りであることを示します。代替トラックとして割り当てられたトラックは、割り振り済みとしてマークされます。

VTOC 索引マップ

VIXM はビットマップを含み、各ビットは 1 つの VTOC 索引レコードを表します。ビットの状況は、VIR が割り振り済み (1) または未割り振り (0) であることを示します。

VIXM の領域が VTOC 記録機能 (VRF) データ用に予約されています。(この機能は、索引付き VTOC の一部のエラーを検出し、リカバリーすることができます。)

索引付き VTOC の構造

索引付きと索引なし VTOC は同様の構造を持っていますが、索引付き VTOC には、次のような相違点があります。

- 単一の、空の format-5 DSCB のみが存在します。
- 一部の format-4 DSCB データ (format-0 DSCB の数、および最高 format-1 DSCB の CCHHR) は、DADSM によって保守されません。VSE ビット (フィールド DS4VTOCI のビット 0) は、format-4 DSCB に 1 に設定されています。これは、これらのフィールド (および format-5 DSCB) の内容が無効であることを示しています。索引ビット (フィールド DS4VTOCI のビット 7) は、format-4 DSCB に設定されています。これは、VTOC 索引が存在することを示します。

DADSM マクロを使用した VTOC へのアクセス

DADSM または共通 VTOC アクセス機能 (CVAF) マクロを使用して、VTOC とその索引にアクセスできます。(CVAF によるアクセスについては、43 ページの『CVAF マクロを使用した VTOC へのアクセス』で説明します。) この章で説明する DADSM マクロおよびタスクは、以下のもので構成されます。

- LSPACE は、DASD ボリュームのフリー・スペース、ボリュームのフラグメント化、および VTOC 状況情報を提供します。LSPACE は 4 つのユーザー指定領域に情報を戻します。
- OBTAIN は、VTOC から DSCB を読み取ります。
- PARTREL は、順次または区分データ・セット、または PDSE から未使用スペースを解放します。
- REALLOC は、DASD スペースを割り振ります。

DSCB を仮想ストレージに読み取るには、OBTAIN および CAMLST マクロ命令を使用します。DSCB に関連したデータ・セット名、または DSCB の絶対トラック・アドレスを使用して、読み取る DSCB を識別します。DSCB を含めるために、140 バイトのデータ域を仮想ストレージに提供します。データ・セット名を指定すると、ID (format-1 または format-4) DSCB が仮想ストレージに読み取られます。format-1 または format-4 DSCB 以外の DSCB を読み取るには、絶対トラック・アドレスを指定します (29 ページの『例』の例を参照してください)。

制約事項: SYSIN または SYSOUT データ・セットに対して OBTAIN マクロ命令を使用することはできません。

順次、区分、またはキー順データ・セットあるいは PDSE から未使用スペースを解放するには、PARTREL マクロ命令を使用します。プログラムは APF 許可が必要です。

以下のマクロ命令の説明では、コーディングの例、プログラミングに関する注記、および例外戻りコードについて説明します。

LSPACE を使用した DASD ボリューム情報の要求

LSPACE マクロを使用すると、直接アクセス・ストレージ・デバイス (DASD) ボリュームのフリー・スペース、ボリュームのフラグメント化、およびボリューム目録 (VTOC) の状況情報を入手できます。LSPACE マクロは、パラメーター・リストに状況情報 (LSPACE 副次機能、戻りコード、理由コードなど) を戻します。LSPACE マクロはレジスター 15 に戻りコードも戻します。9999 個を超えるシリンダーが構成されているボリュームの場合、EXPMSG オプションを使用して、LSPACE マクロが必要とする拡張メッセージ戻り域を作成できます。23 ページの表 6 は、LSPACE パラメーター・リストのフォーマットを示しています。LSPACE は、ボリューム上のフリー・エクステントの総数やフラグメント化指標などの追加情報も戻すことができます。この追加情報は、以下の領域に戻されます。

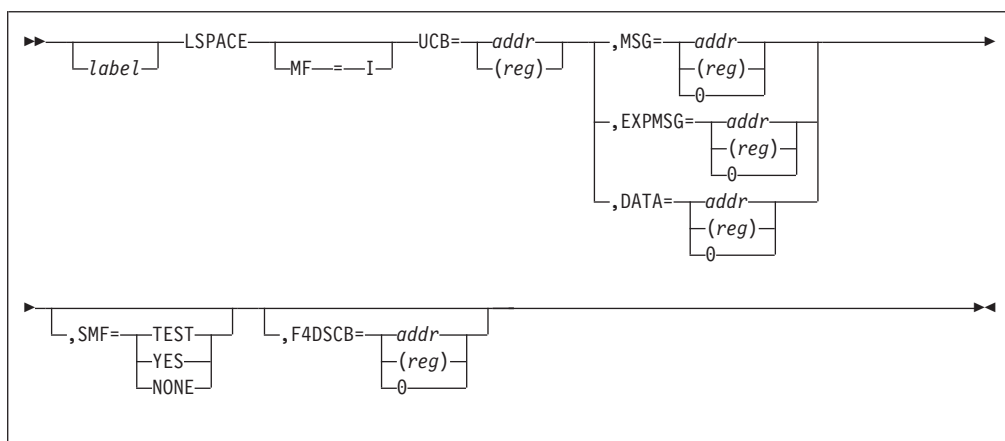
- **メッセージ戻り域:** 24 ページで説明します。
- **拡張メッセージ戻り域:** 25 ページで説明します。
- **データ戻り域:** 25 ページで説明します。
- **Format-4 DSCB 戻り域:** 26 ページで説明します。

呼び出し側プログラムは、LSPACE 処理時に、処理するボリュームがマウントされたままであることを確認する必要があります。ボリュームは割り振り済みである必要はありません。

LSPACE の発行時に装置が作動不能であり、作動不能のままであった場合、LSPACE は最終的に、タイムアウト・メッセージと共に戻りコード 4 を戻します。(25 ページの表 7 を参照してください。) 現行レベルのシステムでは、LSPACE はデフォルトで 240 秒間待ちます。この時間の量は、パラメーター・リストのオフセット 7 にバイトを設定することにより変更できます。(23 ページの表 6 を参照してください。) マクロはこのためのパラメーターを備えていないため、リスト形式および実行形式のマクロを使用する必要があります。

LSPACE - 標準形式

標準形式の LSPACE マクロのフォーマットは、次のとおりです。



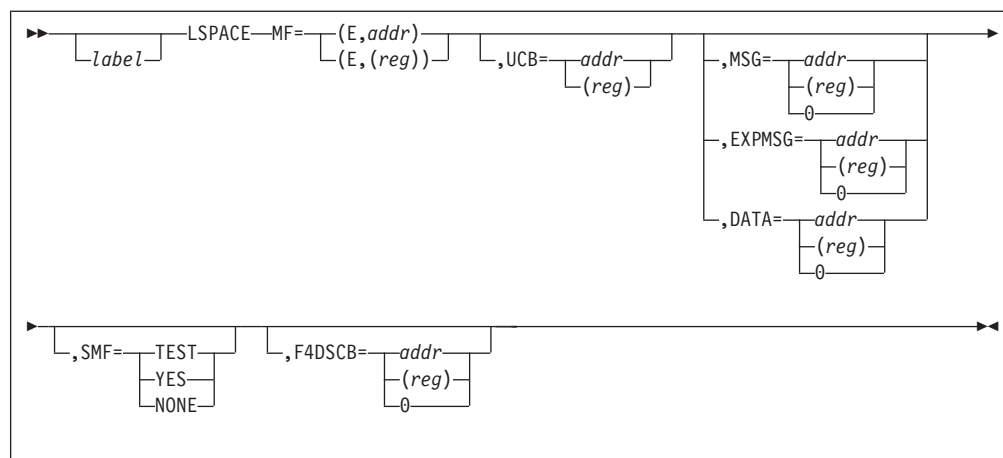
MF=I 標準形式の LSPACE マクロを指定します。

- I** 標準形式のマクロを指定します。これは、必要な変数を含んでいる標準パラメーター・リストを生成し、パラメーター・リストのアドレスをレジスター 1 にロードして、監視プログラム呼び出しを発行します。標準形式がデフォルトです。

要件: MF=I を使用するときは、UCB を指定する必要があります。

LSPACE - 実行形式

実行形式の LSPACE マクロのフォーマットは、次のとおりです。

**MF=(E,addr) または (E,(reg))**

実行形式の LSPACE マクロを指定します。

通常は、パラメーター・リストを作成するために、実行形式の前に MF=L 形式が発行されます。

(E,addr) *addr* で指定されたパラメーター・リストのアドレスをレジスター 1 にロードした後、監視プログラム呼び出しを発行します。

(E,(reg)) *(reg)* で指定されたパラメーター・リストのアドレスをレジスター 1 にロードした後、監視プログラム呼び出しを発行します。

UCB=addr または **(reg)**

フリー・スペース情報を要求しているボリュームの UCB のアドレスを指定します。UCB アドレスは、キャプチャーされた UCB のアドレスでも、16 MB 境界より上または下の実 UCB のアドレスでも構いません。呼び出し側が 31 ビットの場合、高位バイトは UCB アドレスの一部であり、24 ビットの UCB アドレスが渡される場合は、ゼロにクリアする必要があります。

addr-RX タイプ・アドレス

UCB アドレスを含んでいるフルワードのアドレスを指定します。

(reg)-(2-12) 装置の UCB アドレスが入っているレジスターを指定します。これは **addr** 形式のパラメーターとは異なることに注意してください。

標準 (MF=I) 形式のマクロを使用している場合は、UCB アドレスを提供する必要があります。

制約事項:

1. LSPACE マクロは、UCB または UCB コピーのアドレスを受け入れます。無許可プログラムは、UCBSCAN マクロを使用し、COPY および UCBAREA キーワードを指定することにより、UCB のコピーを入手できます。UCB コピーは、16 MB 境界より上または下、およびワード境界上に置くことができます。詳細については、「z/OS ハードウェア構成定義(HCD)計画」を参照してください。
2. LSPACE は、VIO UCB をサポートしません。

MSG=addr or (reg) または **0** or **EXPMSG=addr or (reg) or 0** または **DATA= addr or (reg)** または **0**

LSPACE がフリー・スペース情報を戻す方法を指定します。(オプション)

制約事項: MSG、EXPMSG、および DATA パラメーターは、相互に排他的です。

MSG=addr または **(2-12)** または **0**

LSPACE がフリー・スペース・メッセージまたは状況情報 (要求が失敗した場合) のいずれかを戻す、呼び出し側提供の 30 バイトのメッセージ戻り域のアドレスを指定します。この領域について詳しくは、24 ページの『メッセージ戻り域』を参照してください。

addr-RX タイプ・アドレス

メッセージ戻り域のアドレスを指定します。

(reg)-(2-12) メッセージ戻り域のアドレスが入っているレジスターを指定します。

0 フリー・スペース・メッセージは不要であることを指定します。これは、実行形式以外のすべての形式のマクロのデフォルトです。

EXPMSG=addr または **(reg)** または **0**

LSPACE がフリー・スペース・メッセージまたは状況情報 (失敗した要求の場合)

合) のいずれかを戻す、呼び出し側提供の 40 バイトの拡張メッセージ戻り域のアドレスを指定します。この領域についての詳細は、25 ページの『拡張メッセージ戻り域』を参照してください。

addr-RX タイプ・アドレス

メッセージ戻り域のアドレスを指定します。

(*reg*)-(2-12) メッセージ戻り域のアドレスが入っているレジスターを指定します。

0 フリー・スペース・メッセージは不要であることを指定します。これは、実行形式以外のすべての形式のマクロのデフォルトです。

DATA=addr または (*reg*) または **0**

LSPACE がフリー・スペースまたはボリューム情報を戻す、呼び出し側提供のデータ戻り域のアドレスを指定します。この領域についての詳細は、25 ページの『データ戻り域』を参照してください。

addr-RX タイプ・アドレス

データ戻り域のアドレスを指定します。

(*reg*)-(2-12) データ戻り域のアドレスが入っているレジスターを指定します。

0 フリー・スペースおよびボリューム情報は不要であることを指定します。

SMF=TEST または **YES** または **NONE**

SMF 処理のタイプを指定します。

TEST LSPACE はアクティブ SMF システムをテストすること、および SMF ボリューム情報が必要かどうかを指定します。これらの条件を満たす場合、SMF レコードが書き込まれます。監視プログラム状態、記憶保護キー 0-7、または APF 許可のもとで実行されるプログラムのみ、このオペランドを指定できます。

YES ボリューム情報を含む SMF レコードを書き込む必要があることを指定します。監視プログラム状態、記憶保護キー 0-7、または APF 許可のもとで実行されるプログラムのみ、このオペランドを指定できます。

NONE ボリューム情報を含む SMF レコードを書き込む必要がないことを指定します。これは、実行形式以外のすべての形式のマクロのデフォルトです。

F4DSCB=addr または (*reg*) または **0**

LSPACE がボリュームの format-4 DSCB を戻す、呼び出し側プログラム提供の 96 バイト DSCB 戻り域のアドレスを指定します。format-4 DSCB フィールドについての説明は、10 ページの表 4 を参照してください。

addr-RX タイプ・アドレス

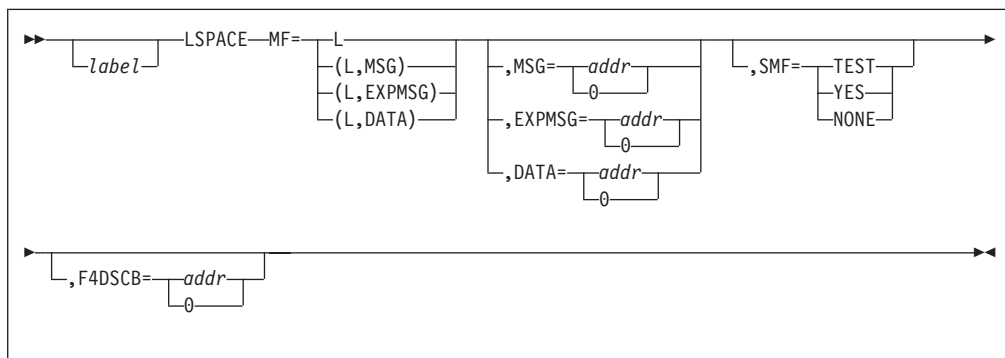
format-4 DSCB 戻り域のアドレスを指定します。

(*reg*)-(2-12) format-4 DSCB 戻り域のアドレスが入っているレジスターを指定します。

- 0 ボリュームの format-4 DSCB の部分は不要であることを指定します。これは、実行形式以外のすべての形式のマクロのデフォルトです。

LSPACE - リスト形式

リスト形式の LSPACE マクロのフォーマットは、次のとおりです。



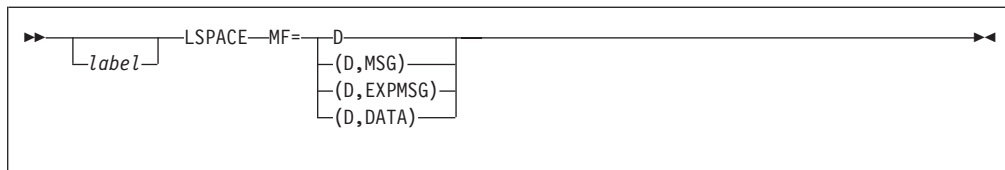
MF=L または (L,MSG) または (L,EXPMSG) または (L,DATA)

リスト形式の LSPACE マクロを指定します。

- L** 呼び出し側プログラム内で、必要な定数を生成します。その後で、その定数を使用する実行形式のマクロを発行できます。
- (L,MSG)** 呼び出し側プログラム内で、必要なメッセージ戻り域定数を生成します。これ以外のパラメーターは許可されません。
- (L,EXPMSG)** 呼び出し側プログラム内で、必要な拡張メッセージ戻り域定数を生成します。これ以外のパラメーターは許可されません。
- (L,DATA)** 呼び出し側プログラム内で、必要なデータ戻り域定数を生成します。これ以外のパラメーターは許可されません。

LSPACE: DSECT 形式

DSECT 形式の LSPACE マクロのフォーマットは、次のとおりです。



MF=D または (D,MSG) または (D,EXPMSG) または (D,DATA)

DSECT 形式の LSPACE マクロを指定します。

- D** LSPACE パラメーター・リストをマップする DSECT を生成します。これ以外のパラメーターは許可されません。LSPACE パラメーター・リストのフォーマットについては、23 ページの表 6 を参照してください。

- (D,MSG)** メッセージ戻り域をマップする DSECT を生成します。これ以外のパラメーターは許可されません。この領域のフォーマットについては、24 ページの『メッセージ戻り域』を参照してください。
- (D,EXPMSG)** 拡張メッセージ戻り域をマップする DSECT を生成します。これ以外のパラメーターは許可されません。この領域のフォーマットについては、25 ページの『拡張メッセージ戻り域』を参照してください。
- (D,DATA)** データ戻り域をマップする DSECT を生成します。これ以外のパラメーターは許可されません。この領域のフォーマットについては、25 ページの『データ戻り域』を参照してください。

表 6. LSPACE パラメーター・リストのフォーマット (MF=D)

名前	オフセット	バイト	説明
LSPAPL			
LSPAPLID	0(X'00')	4	EBCDIC 'LSPA'
LSPANGTH	4(X'04')	2	パラメーター・リストの長さ
LSPAFLAG	6(X'06')	1	パラメーター・フラグ・バイト
LSPASMFY		10	SMF=YES
LSPASMFT		01	SMF=TEST
LSPADATA		. . 1 0 0 . . .	フリー・スペース・データ要求
LSPAMSG		. . 0 1 0 . . .	メッセージ・データが戻されました
LSPAEMSG		. . 0 0 1 . . .	拡張メッセージ・データが要求されました
LSPARSVB	 xxx	予約済み。
LSPAXTIM	7(X'07')	1	最初のチャンネル・プログラムの入出力 (I/O) タイムアウト値 (秒) (デフォルトは、240 秒)
LSPAERCD	8(X'08')	1	LSPACE 戻りコード。24 ページの図 4 を参照
LSPASFID	9(X'09')	1	LSPACE 結果を詳しく説明する LSPACE 副次機能コード
LSPASFPC		X'00'	処理が完了
LSPASFVP		X'01'	妥当性検査パラメーター
LSPASFUS		X'02'	UCB 状況を検査
LSPAFNQ		X'03'	SYSZDMNT での ENQ
LSPASF45		X'04'	F4 と最初の F5 DSCB (EXCP) の読み取り
LSPAFN5		X'05'	次の F5 DSCB (EXCP) の読み取り
LSPASFRV		X'06'	ボリューム・ラベル (EXCP) の読み取り
LSPASF07		X'07'	タイムアウトを指定したボリューム・ラベルの読み取り
LSPASF4X		X'80'	F4 を読み取りとマップ (CVAFDIR)
LSPASFEX		X'81'	フリー・エクステンツの取得 (CVAFDSM)
LSPASFF0		X'82'	F0 カウントの取得 (CVAFDSM)
LSPASFVR		X'83'	VIR カウントの取得 (CVAFDSM)
LSPASFVD		X'84'	VRF の検査 (CVAFVRF)
LSPASF85		X'85'	入った ESTAE ルーチン。LSPACE の処理エラー。
LSPASFRT	10(X'0A')	1	副次機能戻りコード
LSPASFRS	11(X'0B')	1	副次機能理由コード
LSPARS01		X'01'	無効なパラメーター・リスト・ストレージ・キー
LSPARS02		X'02'	無効なパラメーター・リスト ID
LSPARS03		X'03'	無効な LSPACE フラグ
LSPARS04		X'04'	システム管理機能 (SMF) の無効な許可フラグ

VTOC の使用

表 6. LSPACE パラメーター・リストのフォーマット (MF=D) (続き)

名前	オフセット	バイト	説明
LSPARS05		X'05'	無効なメッセージまたはデータ戻り域ストレージ・キー
LSPARS06		X'06'	無効な format-4 DSCB 戻り域ストレージ・キー
LSPARS07		X'07'	無効な UCB アドレス
LSPARS08		X'08'	無効な仮想 UCB アドレス
LSPARS09		X'09'	無効なゼロ VTOC ポインター
LSPAUCB	12(X'0C')	4	UCB アドレス
LSPAFRSP	16(X'10')	4	メッセージまたはデータ戻り域のアドレス
LSPAFMT4	20(X'14')	4	format-4 DSCB のアドレス

LSPACE 戻りコード、副次機能コード、および副次機能の戻りコードと理由コードについての詳細は、25 ページの表 7 および「z/OS DFSMSdfp 診断解説書」を参照してください。

LSPACE からの戻りコード

制御は、LSPACE マクロによって生成された命令の後の命令に戻ります。

LSPACE 戻りコードについての説明は、25 ページの表 7 を参照してください。

LSPACE は、4 バイトの診断情報をレジスター 0 に戻します。この情報についての説明は、「z/OS DFSMSdfp 診断解説書」の『DADSM/CVAF Diagnostic Aids』のセクションを参照してください。

LSPACE 情報戻り域

LSPACE マクロは、パラメーター・リストに状況情報を戻し、オプションで、呼び出し側が要求した次の 4 つの戻り域のいずれかにボリューム情報を戻します。

MSG、EXPMSG、および DATA 域に対する要求は、相互に排他的です。LSPACE は、各情報戻り域のストレージ・キーが呼び出し側のキーと等しいかどうか、または呼び出し側が使用前に許可を得ているかどうかを検査して確認します。

メッセージ戻り域: LSPACE は、30 バイトのメッセージ戻り域 (図 4) に情報を戻します。MSG オプションを付けてメッセージ戻り域を提供すると、LSPACE は、25 ページの表 7 に示すように、戻りコードで修飾された EBCDIC テキストを戻します。

LSPMSG	DSECT	メッセージ領域
LSPMTEXT	DS CL30	メッセージ・テキスト

図 4. DADSM LSPACE フリー・スペース情報 MF=(D,MSG)

表 7. DADSM LSPACE メッセージ戻り域の内容

戻りコード	説明
0(X'00')	<p>テキスト: SPACE=aaaa,bbbb,cccc/dddd,eeee ここで、</p> <p>aaaa = フリー・シリンダーの総数</p> <p>bbbb = 追加フリー・トラックの総数</p> <p>cccc = フリー・エクステントの総数</p> <p>dddd = 最大フリー・エクステント内のシリンダー数</p> <p>eeee = 最大フリー・エクステント内の追加トラック数</p> <p>規則: 3つのコンマのそれぞれの後に (テキスト内の変数の後に) ブランクが1つ入ります。</p> <p>aaaa、bbbb、および cccc では、実際の値はそれより大きい場合でも、示される最大値は 9999 になります。最大値 9999 の使用を避けるには、EXPMSG を使用してください。</p>
4(X'04')	テキスト: LSPACE-PERMANENT I/O ERROR
4(X'04')	テキスト: LSPACE-I/O TIMEOUT ERROR
8(X'08')	テキスト: LSPACE-NON-STANDARD OS VOLUME
12(X'0C')	<p>テキスト: LSPACE-UCB NOT READY</p> <p>テキスト: LSPACE-UCBVTOC IS ZERO</p> <p>テキスト: LSPACE-INVALID PARAMETER</p> <p>テキスト: LSPACE-NOT A DIRECT ACCESS VOL</p>
16(X'10')	テキストは戻されません (無効なパラメーター・リストまたは SMF 標識)
20 (X'14')	テキストは戻されません (LSPACE の処理エラー)

拡張メッセージ戻り域: LSPACE は、40 バイトの拡張メッセージ戻り域 (図 5) に情報を戻します。EXPMSG オプションを付けて拡張メッセージ戻り域を提供すると、LSPACE は、表 7 に示すように、戻りコードで修飾された EBCDIC テキストを戻します。

拡張メッセージ戻り域は、標準メッセージ戻り域と同じ戻りコードを表示しますが、戻りコード・ゼロで提供されるスペース情報が 6 桁の値 (aaaa の代わりに aaaaaa) になります。

LSPMSG	DSECT	拡張メッセージ域
LSPETEXT	DS CL40	拡張メッセージ・テキスト

図 5. DADSM LSPACE フリー・スペース情報フォーマット、MF=(D,EXPMSG)

データ戻り域: DATA オプションを付けてデータ戻り域を提供すると、LSPACE は表 8 で説明するような情報を戻します。

表 8. LSPACE データ戻り域フォーマット

名前	オフセット	バイト	説明
LSPDRETN	0(X'00')	1	戻り域状況バイト。
LSPDSPAC		1... ..	戻されたスペース情報。
LSPDF0CN		.1.	戻された format-0 DSCB カウント。
LSPDVRCN		.1.	戻されたフリー VIR カウント。
LSPDRRES		. . . x xxxx	予約済み。
LSPDSTAT	1(X'01')	1	状況バイト。

表 8. LSPACE データ戻り域フォーマット (続き)

名前	オフセット	バイト	説明
LSPDIXDS		X'80'	VTOC 索引付きデータ・セットが存在します。LSPDIXAC フラグを見て、索引がアクティブかどうかを判別します。
LSPDIXAC		X'40'	索引付き VTOC がアクティブです。
LSPDIRES		. . xx xxxx	予約済み。
LSPDRSV1	2(X'02')	2	予約済み。
LSPDNEXT	4(X'04')	4	フリー・エクステントの数。
LSPDTCYL	8(X'08')	4	フリー・シリンダーの総数。
LSPDTTRK	12(X'0C')	4	追加のフリー・トラックの総数。
LSPDLCYL	16(X'10')	4	最大フリー・エクステント内のシリンダーの数。
LSPDLTRK	20(X'14')	4	最大フリー・エクステント内の追加トラックの数。
LSPDFOS	24(X'18')	4	Format-0 DSCB カウント。
LSPDVIRS	28(X'1C')	4	フリー VIR カウント。
LSPDFRAG	32(X'20')	4	フラグメント化指標。フラグメント化指標とは、ボリューム上のフリー・スペースの相対サイズと分布の数値表現です。指標の値が大きいと、フラグメント化の度合いが高いことを示します。

Format-4 DSCB 戻り域: F4DSCB オプションを付けて format-4 DSCB 戻り域を提供すると、F4DSCB は 10 ページの表 4 で説明するような情報を戻します。

LSPACE の例

次の例は、メッセージ戻り域にフリー・スペース情報を戻します。

```
LSPAMFIM LSPACE MSG=MYMSG,UCB=(R10),MF=I
```

次の例は、データ戻り域にフリー・スペース情報を戻します。

```
LSPAMFID LSPACE DATA=MYDATA,UCB=(R10),MF=I
```

次の例は、パラメーター・リストを定義するためのリスト形式のマクロと、同じパラメーター・リストを参照するための実行形式のマクロを使用します。

```
LSPALIST LSPACE MSG=MYDATA,MF=L
      .
      .
      .
LSPAEX  LSPACE MF=(E,LSPALIST),UCB=(R10)
```

OBTAIN を使用した VTOC からの DSCB の読み取り

以下では、OBTAIN ルーチンを使用して DSCB を読み取る方法について説明します。データ・セット名または絶対装置アドレスのいずれかを指定できます。

OBTAIN は HFS ファイルをサポートしません。HFS ファイルに対して OBTAIN を発行すると、予測不能の結果を受け取ります。

を含んでいるボリューム通し番号が入っている 6 バイト領域の仮想ストレージ位置を指定します。WORKAREA は、DSCB の戻し先の 140 バイト作業域の仮想ストレージ位置を指定します。

制御は、プログラムの OBTAIN マクロ命令の後の次の実行可能命令に戻されます。DSCB が作業域に正常に読み取られた場合、レジスター 15 にゼロが入ります。そうでない場合、レジスター 15 には、表 9 に示す戻りコードのいずれかが入ります。

OBTAIN からの戻りコード (データ・セット名による読み取り)

表 9. DADSM OBTAIN 戻りコード

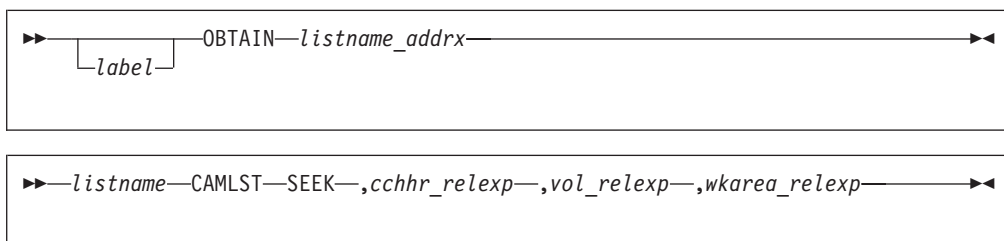
戻りコード	説明
0(X'00')	OBTAIN ルーチンの正常終了。
4(X'04')	必要なボリュームがマウントされていませんでした。
8(X'08')	指定されたボリュームの VTOC 内で format-1 DSCB が見つかりませんでした。
12(X'0C')	永続的な入出力エラーが発生したか、指定されたボリュームの処理時に無効な format-1 DSCB が見つかったか、または CVAF (共通 VTOC アクセス機能) から予期しないエラー戻りコードを受け取りました。
16(X'10')	無効な作業域ポインター。

これらのマクロ命令の実行後、作業域の最初の 96 バイトには、format-1 または format-4 DSCB のデータ部分が入ります。次の 5 バイトには、DSCB の絶対トラック・アドレス (CCHHR) が入ります。VSAM または VIO データ・セットの場合、この 5 バイトにはゼロが入ります。

絶対装置アドレスによる DSCB の読み取り

OBTAIN と CAMLST SEEK オプションを使用して、VTOC から DSCB を読み取ることができます。SEEK オプションを指定するには、SEEK を CAMLST マクロの第 1 オペランドとしてコーディングし、読み取る DSCB の絶対装置アドレスを提供します (VIO データ・セットの DSCB でない場合)。VIO データ・セットの DSCB の読み取りには、SEARCH オプションのみ使用できます。

OBTAIN および CAMLST マクロのフォーマットは、次のとおりです。



listname_addrx CAMLST マクロ命令によってセットアップされたパラメーター・リスト (ラベル付きリスト名) を指します。

SEEK このオペランドは表示どおりにコーディングしてください。

cchhr_relexp DSCB の 5 バイトの絶対装置アドレス (CCHHR) の仮想ストレージ位置を指定します。

vol_relexp DSCB が置かれている 6 バイトのボリューム通し番号の仮想ストレージ位置を指定します。

wkarea_relexp ユーザーが定義する必要がある 140 バイトの作業域の仮想ストレージ位置を指定します。

例: 次の例では、SEEK オプションを使用して、実装置アドレス X'00 00 00 01 07' にある DSCB が仮想ストレージ位置 READAREA に戻されます。DSCB は、ボリューム通し番号 108745 にあります。

```

          OBTAIN  ACTADDR          READ DSCB FROM
*                               LOCATION SHOWN IN CCHHR
*                               INTO STORAGE AT LOCATION
*                               NAMED READAREA

ACTADDR  CAMLST  SEEK,CCHHR,VOLSER,READAREA
CCHHR    DC      XL5'0000000107' ABSOLUTE TRACK ADDRESS
VOLSER   DC      CL6'108745'     VOLUME SERIAL NUMBER
READAREA DS      140C           140-BYTE WORK AREA

```

推奨: 戻りコードを確認してください。

OBTAIN マクロは、CAMLST パラメーター・リストを指示します。SEEK (CAMLST の第 1 オペランド) は、DSCB を仮想ストレージに読み取ること指定します。CCHHR は、DSCB の 5 バイト実装置アドレスを含んでいる保管場所を指定します。VOLSER は、DSCB が常駐するボリューム通し番号を含んでいる保管場所を指定します。READAREA は、140 バイト DSCB の戻り先の保管場所を指定します。

制御は、プログラムの OBTAIN マクロ命令の後の次の実行可能命令に戻されます。DSCB が作業域に読み取られた場合、レジスター 15 にゼロが入ります。そうでない場合、レジスター 15 には、表 10 に示す戻りコードのいずれかが入ります。

OBTAIN からの戻りコード (絶対装置アドレスによる読み取り)

表 10. DADSM OBTAIN 戻りコード

戻りコード	説明
0(X'00')	OBTAIN ルーチンの正常終了。
4(X'04')	必要なボリュームがマウントされていませんでした。
8(X'08')	指定されたボリュームの VTOC 内で format-1 DSCB が見つかりませんでした。
12(X'0C')	永続的な入出力エラーが発生したか、または CVAF から予期しないエラー戻りコードを受け取りました。
16(X'10')	無効な作業域ポインター。
20(X'14')	SEEK オプションが指定されましたが、絶対トラック・アドレス (CCHHR) が VTOC の境界内ではありません。

PARTREL を使用した DASD データ・セットからの未使用スペースの解放

DADSM は、順次または区分データ・セット、PDSE、順次拡張形式データ・セット、および VSAM 拡張形式データ・セットに割り振られている未使用スペースの解放をサポートします。部分リリース機能は、次の時点で呼び出されます。

- データ・セットを閉じるとき (DD ステートメントで SPACE の RLSE サブパラメーターが指定された場合、またはストレージ管理者が管理クラス定義の部分リリース・オプションで該当する値を指定した場合)。
- データ・セットの拡張前に取られたチェックポイントからの再始動処理時。
- DFSMSHsm がスペース管理サイクルを実行し、管理クラス定義で該当する値が指定されている場合。
- PARTREL マクロが発行されたとき。

PARTREL マクロ

PARTREL マクロは、索引付き VTOC を持つまたは持たないボリューム上の順次および区分データ・セットをサポートします。また、索引付き VTOC が必要な SMS 管理ボリューム上の PDSE および拡張形式データ・セットもサポートします。

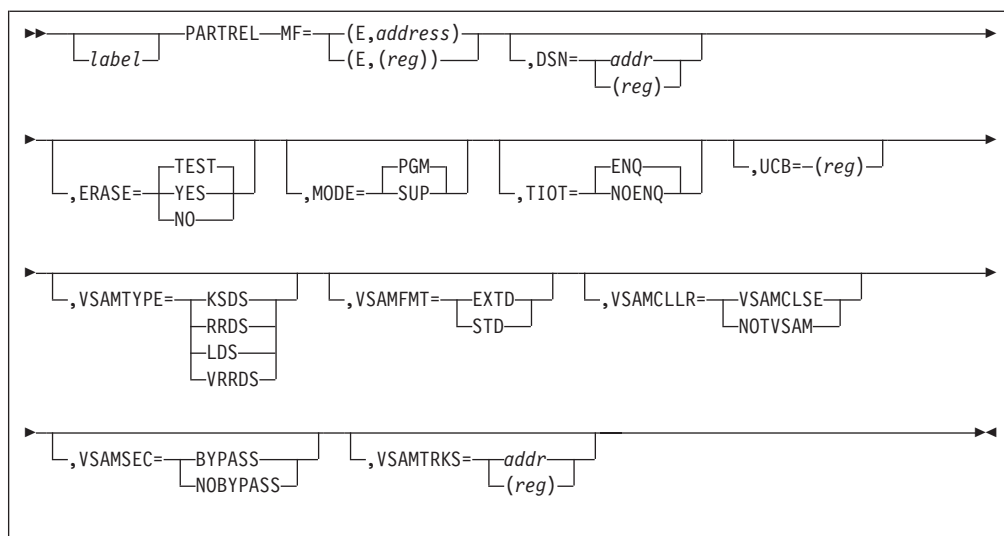
PARTREL は HFS ファイルをサポートしません。HFS ファイルに対して PARTREL を発行すると、予測不能の結果を受け取ります。

PARTREL は、実行形式、DSECT、およびリスト形式でコーディングできますが、標準形式は使用できません。呼び出し側プログラムには、以下の要件が適用されます。

- APF 許可が必要です。
- このタスクに対してボリュームが割り振り済みであり、PARTREL 機能の実行中マウントされたままであることを確認する必要があります。
- データ・セットがオープンされていないことを確認する必要があります。
- VTOC 上にロックまたは ENQ が保留されてはなりません。
- 汎用レジスター 13 に使用可能な標準レジスター保管域のアドレスを提供する必要があります。
- 関連のパラメーター・リストとパラメーターは、16 MB 仮想境界より上または下のストレージに提供できます。
- 任意のストレージ・キーを使用できます。
- 監視プログラム状態または問題プログラム状態で実行できます。
- 24 ビットまたは 31 ビット・アドレッシング・モードを使用できます。
- UCB コピーではなく、UCB のアドレスが必要です。

PARTREL - 実行形式

実行形式の PARTREL マクロのフォーマットは、次のとおりです。



MODE を除いて、すべてのパラメーターのデフォルトは、パラメーター・リストの現行内容になります。MODE パラメーターのデフォルトは、PGM です。

以下のパラメーターの説明には、DADSM 処理に関する情報が含まれています。説明の中では「値」という用語を使用して、DADSM に渡されるパラメーター値を指定しています。値は、次の方法で指定できます。

- PARTREL マクロのパラメーターとして指定する。
- パラメーター・リスト内に、そのパラメーターに関連した値として提供する。
- 要求の中で提供された情報から DADSM によって定義する。

MF=(E,addr) または (E,(reg))

実行形式のマクロと既存の PARTREL パラメーター・リストのアドレスを指定します。

addr-RX タイプ・アドレス、(reg)-(0-12)

PARTREL パラメーター・リストのアドレスを指定します。

DSN=addr または (reg)

データ・セット名を含んでいる 44 バイト域のアドレスを指定します。名前は左寄せし、未使用バイトはブランクとして定義します。

addr-RX タイプ・アドレス、(reg)-(0)、(2-12)

DSN の値を提供する必要があります。

ERASE=YES または NO または TEST

残余データ消去属性を指定します (消去属性については、67 ページの『VTOC からのデータ・セットの削除』を参照してください)。ERASE=YES と ERASE=NO は、相互に排他的です。デフォルトは ERASE=TEST です。VSAMTYPE を指定した場合、ERASE は無視されます。

ERASE=YES 新規割り振り用に使用可能にする前に、解放される領域を消去する (ゼロで上書きする) 必要があることを指定します。

ERASE=NO 領域を消去してはならないことを指定します。この指定は RACF 消去属性をオーバーライドします。

ERASE=TEST 関連 RACF 消去属性を使用することを指定します。

MODE=PGM または **SUP**

PARTREL は、問題プログラム状態 (**MODE=PGM**) または監視プログラム状態 (**MODE=SUP**) の呼び出し側によって要求されることを指定します。**MODE=PGM** がデフォルトです。

呼び出し側プログラムが監視プログラム状態である (かつ、監視プログラム状態で制御を戻す必要がある) 場合、**MODE** の値は **SUP** でなければなりません。呼び出し側プログラムが問題プログラム状態である場合、**MODE** の値は **PGM** でなければなりません。

TIOT=ENQ または **NOENQ**

部分リリース・ルーチン内で必要な **SYSZTIOT** および **SYSDSN ENQ** 処理を指定します。デフォルトは **ENQ** です。**VSAMTYPE** を指定した場合、**TIOT** は無視されます。

TIOT=ENQ 部分リリースは、**SYSZTIOT** および **SYSDSN** に対して通常の排他的 **ENQ** を行うことを指定します。これらの **ENQ** のいずれかが失敗した場合、**PARTREL** は要求を終了し、戻りコード 'X'08' を戻します。

TIOT=NOENQ

呼び出し側が必要な逐次化を準備したことを指定します。呼び出し側が **SYSDSN** の排他的使用を取得していないことを部分リリースが検出した場合、**PARTREL** は要求を終了し、戻りコード 'X'24' を戻します。

TIOT=NOENQ が指定され、フラグ **PRLTIOTX** がオンに設定されている場合、**ENQ TEST** は実行されません。これは危険な選択であり、データ破壊の原因になります。IBM では **PRLTIOTX** を設定しないことをお勧めします。

UCB=(reg)

対象データ・セットが常駐するボリューム用の **UCB** のアドレスを指定します。**UCB** アドレスは、キャプチャーされた **UCB** のアドレスでも、16 MB 境界より上または下の実 **UCB** のアドレスでも構いません。呼び出し側が 31 ビットの場合、高位バイトは **UCB** アドレスの一部であり、24 ビットの **UCB** アドレスが渡される場合は、ゼロにクリアする必要があります。ボリュームはマウント済みでなければならず、ユーザーはそれがマウントされたままであることを確認する必要があります。**UCB** コピーではなく、**UCB** のアドレスを使用してください。

(reg)-(0), (2-12)

装置の **UCB** アドレスが入っているレジスターを指定します。

VSAMTYPE=KSDS または **ESDS** または **RRDS** または **LDS** または **VRRDS**
VSAM データ・セットのタイプを指定します。

現行では、拡張キー順データ・セット (KSDS) が、サポートされる唯一の VSAM データ・セットのタイプです。それ以外のタイプを指定すると、無効なデータ・セット・タイプを示す戻りコードを受け取ります。

VSAMTYPE を指定する場合、VSAMFMT、VSAMCLLR、VSAMSEC、および VSAMTRKS も指定する必要があります。

VSAMTYPE を指定した場合、ERASE= パラメーターは無視されません。

VSAMTYPE=KSDS

VSAM キー順データ・セットを指定します。

VSAMTYPE=ESDS

VSAM 入力順データ・セットを指定します。

VSAMTYPE=RRDS

VSAM 固定長相対レコード・データ・セットを指定します。

VSAMTYPE=LDS

VSAM 線形データ・セットを指定します。

VSAMTYPE=VRRDS

VSAM 可変長相対レコード・データ・セットを指定します。

VSAMFMT=EXTD または STD

データ・セットのフォーマットを指定します。

VSAMFMT=EXTD

これは拡張フォーマットのデータ・セットであることを指定します。

VSAMFMT=STD

これは拡張フォーマットのデータ・セットではないことを指定します。STD カタログ・サービスを指定した場合、部分リリースにエラーが戻されます。

VSAMCLLR=VSAMCLSE または NOTVSAM

呼び出し側を指定します。

VSAMCLLR=VSAMCLSE

呼び出し側は VSAM CLOSE であることを指定します。

VSAMCLLR=NOTVSAM

呼び出し側は VSAM CLOSE 以外であることを指定します。

VSAMSEC=BYPASS または NOBYPASS

セキュリティー検査を実行するかどうかを指定します。

VSAMSEC=BYPASS

セキュリティー検査を実行しないことを指定します。

VSAMSEC=NOBYPASS

セキュリティー検査を実行することを指定します。

VSAMTRKS=addr または (reg)

このデータ・セット用に解放されたトラック数を入れるために使用される 4 バイト域のアドレスを指定します。

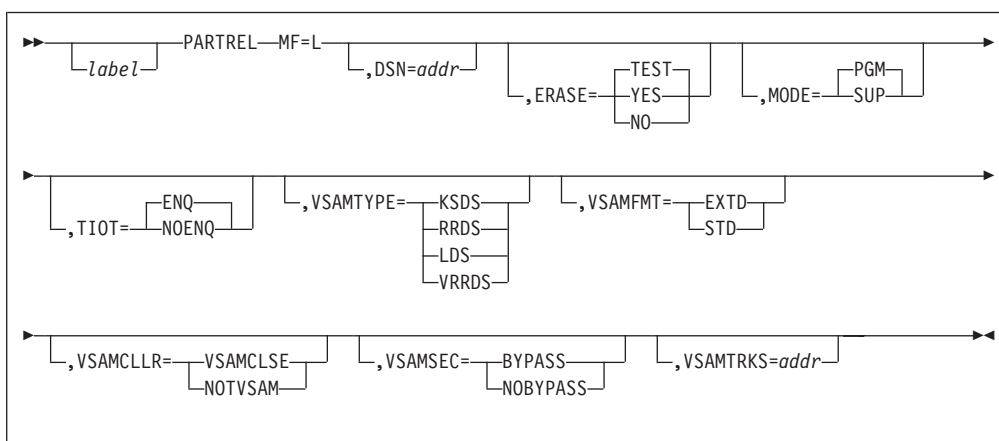
VSAMTRKS のアドレスは、VSAMTYPE をコーディングした場合にのみ提供する必要があります。

addr-RX タイプ・アドレス、(reg)-(0)、(2-12)

VSAM 拡張フォーマット・データ・セットの場合、PARTREL は、データ・セットのすべての部分の解放されたすべてのスペースの合計値を戻します。

PARTREL - リスト形式

リスト形式の PARTREL マクロのフォーマットは、次のとおりです。



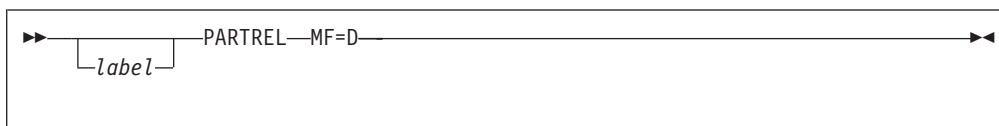
制約事項:

1. 実行形式の UCB パラメーターは、リスト形式では指定できません。
2. リスト形式の MODE パラメーターは、文書専用です。MODE の値は、実行形式での指定またはデフォルトと同じです。

パラメーターの説明は、実行形式の項を参照してください。

PARTREL - DSECT 形式

DSECT 形式の PARTREL マクロのフォーマットは、次のとおりです。



説明: 次の例は、パラメーター・リストの記述を示しています。

```

++PRLPLID DS    CL4    EBCDIC 'PREL' FOR PARTREL
+PRLNGTH DS    AL2    LENGTH OF PARAMETER LIST
+PRERRCDE DS    H      ERROR CODE RETURNED FROM
+*                               PARTIAL RELEASE
+*
+PRLFLAG DS    XL1    PARAMETER FLAG BYTE
+PRLPGM EQU    X'00'  MODE=PGM (PROBLEM PROGRAM)
+PRLSUP EQU    X'80'  MODE=SUP (SUPERVISOR STATE)
+PRLTIOT EQU   X'40'  TIOT=NOENQ
+PRLNERAS EQU  X'20'  ERASE=NO
    
```

```

+PRLERASE EQU X'10' ERASE=YES
+PRLVCLOS EQU X'08' VSAMCLLR=VSAMCLSE
+PRLVCNV EQU X'04' VSAMCLLR=NOTVSAM
+PRLBYPS EQU X'02' VSAMSEC=BYPASS
+PRLNBYP EQU X'01' VSAMSEC=NOBYPASS
+PRLFLAG2 DS XL1
+PRLVKSDS EQU X'80' VSAMTYPE=KSDS
+PRLVESDS EQU X'40' VSAMTYPE=ESDS
+PRLVRRDS EQU X'20' VSAMTYPE=RRDS
+PRLVVRRD EQU X'10' VSAMTYPE=VRRDS
+PRLVLDS EQU X'08' VSAMTYPE=LDS
+PRLXTD EQU X'04' VSAMFMT=EXTD
+PRLSTD EQU X'02' VSAMFMT=STD
+PRLTIOTX EQU X'01' No ENQ TEST when TIOT=NOENQ is specified
+PRLRSVD DS XL2 RESERVED
+PRLDSN DS A DATA SET NAME POINTER
+PRLUCB DS A UCB POINTER
+PRLTRKS DS A ADDRESS OF NUMBER OF TRACKS RETURNED
(OONLY VALID FOR VSAM REQUESTS)
+PRLCTGR DS F CATALOG REASON CODE
+PREND EQU * END OF PARAMETER LIST
+PRENGTH EQU PREND-PRELPL LENGTH OF PARAMETER LIST

```

PARTREL からの戻りコード

制御は、PARTREL マクロによって生成された最後の命令の後の命令に戻ります。レジスター 15 に該当する PARTREL 戻りコードが入ります。

エラーが発生した場合、PARTREL は障害関連の状況情報からなるメッセージ IEC614I を発行します。

- CVAF 機能の結果としてエラーが生じた場合、副次機能戻りコード・フィールドに CVAF 戻りコードが入り、副次機能理由コード・フィールドに CVAF 状況コード (CVSTAT) が入ります。
- EXCP チャンネル・プログラムの実行の結果としてエラーが生じた場合、副次機能戻りコードと理由コード・フィールドには、次のいずれかが入ります。
 - ECB 完了コードと CSW チャンネル状況 (ECB 完了コードが X'41' でなく、チャンネル状況がゼロでない場合)
 - センス・バイト (2 バイト) 形式の IOB (ECB 完了コードが X'41' で、チャンネル状況がない場合)
- RACF 呼び出しの結果としてエラーが生じた場合、副次機能戻りコードと理由コード・フィールドには、RACF 戻りコードと理由コードが入ります。

表 11 は、PARTREL 戻りコードによって示される状態について説明しています。

例外: これは、DADSM 部分リリース戻りコードの累積リストです。一部のコードは PARTREL マクロには適用されない場合があります。

表 11. DADSM PARTREL 戻りコード

戻りコード	説明
0(X'00')	レジスター 0 に戻された 4 バイトの診断情報がすべてゼロの場合、これは部分リリースが正常に行われたことを示しています。ゼロでない場合は、「z/OS DFSMSdfp 診断解説書」の『DADSM/CVAF』のセクションを参照して、障害関連の状態を判別してください。
2(X'02')	VTOC でエクステントを見つけることができません。
4(X'04')	VTOC でエクステントを見つけることができません。

表 11. DADSM PARTREL 戻りコード (続き)

戻りコード	説明
8(X'08')	必要な SYSZTIOT または SYSDSN ENQ が失敗したか、あるいは無関係の DEB が、データ・セットに対して別の DCB がオープンされていることを示しています。
12(X'0C')	無効なパラメーター・リスト。
16(X'10')	以下の状態のいずれかが発生しました。 <ul style="list-style-type: none"> • 永続的な入出力エラーが発生した。 • CVAF が予期しない戻りコードを提供した。 • インストール・システム出口が要求をリジェクトした。 • 解放するトラックの消去中に入出力エラーが発生した (ERASE-on-SCRATCH の場合)。
20(X'14')	DSN、または DSN ポインターが無効です。
24(X'18')	無効な UCB ポインター。
28(X'1C')	指定された DSORG はサポートされません。
32(X'20')	VTOC 内に余地がありません。
36(X'24')	無効な TIOT=NOENQ 要求。呼び出し側が SYSDSN の排他使用を取得していません。
40(X'28')	SMS が要求を処理していたときにエラーが発生しました。
44(X'2C')	CLOSE が呼び出し側です。ユーザーが PREEXIT ルーチンを使用して部分リリースをリジェクトしました。
48(X'30')	CCHH から相対トラック・アドレスへの変換中にエラーが発生しました。
52(X'34')	エクステント記述子テーブルのソート変換中にエラーが発生しました。
56(X'38')	新しいエクステント記述子テーブルから古いエクステント記述子テーブルへの変換中にエラーが発生しました。
60(X'3C')	相対トラック・アドレスから CCHH への変換中にエラーが発生しました。
64(X'40')	Format-7 からエクステント記述子テーブルへの変換中にエラーが発生しました。
68(X'44')	format-5 からエクステント記述子テーブルへの変換中にエラーが発生しました。
72(X'48')	入力 DSCB が format-5 または format-7 DSCB ではありません。
76(X'4C')	エクステント記述子テーブルから Format-7 への変換中にエラーが発生しました。
80(X'50')	エクステント記述子テーブルから Format-5 への変換中にエラーが発生しました。
84(X'54')	VSAM 標準形式データ・セットの指定が間違っています。データ・セットが VSAM の場合は、拡張フォーマットでなければなりません。
88(X'58')	VSAMTYPE を指定した場合、VSAMTRKS アドレス・パラメーターを指定する必要があります。
92(X'5C')	指定された VSAM データ・セット名は、VSAM クラスター・データ・コンポーネントではありません。
96(X'60')	部分リリースへのカタログ呼び出しが失敗しました。
100(X'64')	要求は、保証スペースを使用するように定義された VSAM データ・セットに送信されました。
104(X'68')	カタログ・サービスへの呼び出しから戻された、予期しない理由コードを受け取りました。

REALLOC を使用した DASD データ・セットの作成 (割り振り)

REALLOC マクロは、新規データ・セットを割り振るために、パラメーター・リストを作成します。マクロは、実行形式、DSECT 形式、またはリスト形式でコーディングできますが、標準形式は使用できません。呼び出し側プログラムには、次の要件とオプションが適用されます。

- APF 許可が必要です。
- ボリュームをこのアドレス・スペースに割り振る必要があり、REALLOC 機能の実行中マウントされたままであることを確認する必要があります。
- ロックを保持してはなりません。
- 関連のパラメーター・リストとパラメーターは、16 MB より上または下のストレージに提供できます。
- 任意のストレージ・キーを使用できます。
- 監視プログラム状態または問題プログラム状態で実行できます。
- 24 ビットまたは 31 ビット・アドレッシング・モードを使用できます。
- REALLOC は RACF またはカタログ管理を呼び出さないことに注意する必要があります。
- REALLOC は SMS 管理ボリューム上にはデータ・セットを作成できないことに注意する必要があります (SMS 管理データ・セットは、JCL または動的割り振りを通して作成できます)。
- REALLOC は PDSE、HFS データ・セット、または拡張フォーマット・データ・セットを作成できないことに注意する必要があります。
- UCB コピーではなく、UCB のアドレスが必要です。

これに加えて、呼び出し側プログラムは、1 つ以上のモデル DSCB を備えた REALLOC マクロを提供する必要があります。ユーザーは、OBTAIN マクロを使用して他のデータ・セットから DSCB を取得し、それを要求用に変更することができます。DADSM はこれらのモデル DSCB を使用して、割り振り要求の妥当性検査を行い、要求された割り振りのために VTOC に書き込まれる DSCB を構成します。

REALLOC マクロの ALLOC パラメーターは、割り振り要求を絶対 (ABS) または移動可能 (MOV) のいずれかとして定義します。

要求されたデータ・セットの割り振りは、ボリューム上の配置には依存しません。これは、ALLOC=MOV 要求の部分 DSCB でオン/オフに設定できる、format-1 DSCB ビット DS1DSGU (移動不能ビット) への参照ではありません。つまり、後でデータ・セットに位置依存の情報を含めることができます。

絶対要求は、索引付き VTOC をサポートする単一ボリュームに限定されます。絶対要求では、割り振りパラメーター・セット、完全な format-1 DSCB、およびオプションの format-3 DSCB (必要なデータ・セットの属性を記述) を提供します。

- ユーザー・ラベル・エクステントを持つデータ・セットに対するサポートが提供されていますが、このタイプだけに限りません。
- 割り振られるエクステントの数とそのボリューム上の絶対配置は、format-1 DSCB と 1 つの (オプション) format-3 DSCB によって定義されます。

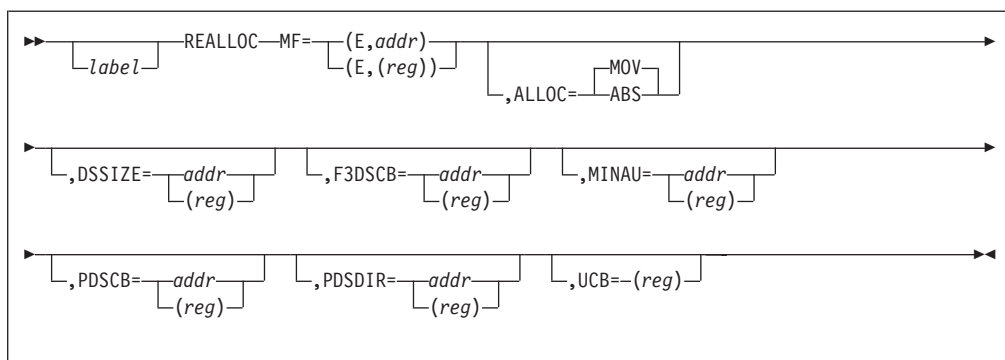
部分 DSCB (IECPDSCB マクロによってマップ) は、最初の 106 バイトの format-1 DSCB と、その後続く 10 バイト (1 次スペース要求とディレクトリー・ブロック数を指定する) から構成されます。

移動可能要求は、索引付き VTOC をサポートする、またはしない単一ボリュームに限定されます。移動可能要求では、割り振りパラメーター・セットと、必要なデータ・セットの属性を記述する部分 DSCB を提供します。

- 絶対トラック割り振りデータ・セットはサポートされません。
- 割り振ることができるエクステントの最大数は、部分 DSCB のデータ・セット編成 (PD1DSORG) とシステム管理ストレージ標識 (PD1SMSFG) バイトによって決まります。PD1DSORG が VSAM データ・セット編成を示している場合は、エクステントの最大数は、複数ボリューム全体で 7257、単一ボリュームでは 123 です。

REALLOC - 実行形式

実行形式の REALLOC マクロのフォーマットは、次のとおりです。



ALLOC を除いて、すべてのパラメーターのデフォルトは、パラメーター・リストの現行内容になります。ALLOC パラメーターのデフォルトは MOV です。

MF=(E,addr) または (E,(reg))

実行形式のマクロと REALLOC パラメーター・リストのアドレスを指定します。

addr-RX タイプ・アドレス、(reg)-(0-12)

REALLOC パラメーター・リストのアドレスを指定します。

ALLOC=ABS または MOV

REALLOC 要求が、絶対エクステント (ALLOC=ABS) であるか、移動可能割り振り (ALLOC=MOV) であるかを指定します。ALLOC=MOV がデフォルトです。

DSSIZE=addr または (reg)

割り振るデータ・セットのサイズをトラック数で指定します。DSSIZE パラメーターは ALLOC=ABS 要求に対しては無効です。

addr-RX タイプ・アドレス

データ・セット・サイズを含んでいるワードのアドレスを指定します。

(reg)-(0), (2-12)

データ・セットのサイズが入っているレジスターを指定します。

ALLOC=MOV 要求の場合、DSSIZE の値を提供してください。部分 DSCB の PDPRIQTY フィールドは無視されます。

REALLOC は、部分 DSCB の PD1SCALO フラグ・バイトがシリンダー要求 (X'C0') または平均ブロック要求 (X'40') を示している場合でも、ユーザーが DSSIZE 値をトラック数で提供したものと想定します。

部分 DSCB の PD1SCALO フラグ・バイトがシリンダー要求 (X'C0') または丸め付き平均ブロック要求 (X'41') を示している場合、DSSIZE の値は、その次の完全シリンダーに丸められます。

F2DSCB

このパラメーターは、アセンブリ用に現在もサポートされていますが、プログラムの実行時にシステムはパラメーターをサポートしません。これは、索引順次データ・セットを作成できないためです。

F3DSCB=addr または (reg)

ストレージ内の format-3 DSCB のアドレスを指定します。この DSCB は、割り振られるデータ・セットの format-3 DSCB を構成するためのモデルとして使用されます。

F3DSCB パラメーターは ALLOC=MOV 要求に対しては無効です。

addr-RX タイプ・アドレス、(reg)-(0), (2-12)

format-1 DSCB の DS1NOEPV バイトが 3 エクステントを超えることを示している場合 (または、DS1NOEPV バイトが 2 エクステントを超えることを示し、DS1EXT1 エクステント・タイプ標識が X'40' (ユーザー・ラベル・エクステント) の場合)、ALLOC=ABS 要求で F3DSCB に対して値を提供します。

F3DSCB キーワードが指定されている場合、REALLOC 要求は最大 16 エクステントに限定されます。format-3 DSCB は 1 つしか指定できません。

format-1 DSCB の DS1NOEPV バイトが 4 エクステント未満を示している場合 (または、DS1NOEPV バイトが 3 エクステント未満を示し、DS1EXT1 エクステント・タイプ標識が X'40' (ユーザー・ラベル・エクステント) の場合)、ALLOC=ABS 要求で F3DSCB に対してゼロの値を入力します。

MINAU=addr または (reg)

最小割り振り単位のサイズをトラック数で指定します。このデータ・セットの 1 次エクステントはすべてこの最小割り振り単位の倍数です。この最小値はデータ・セットの後続の拡張には適用されません。

MINAU パラメーターは ALLOC=ABS 要求に対しては無効です。

addr-RX タイプ・アドレス

最小割り振り単位を含んでいるワードのアドレスを指定します。

(reg)-(0), (2-12)

最小割り振り単位が入っているレジスターを指定します。

次の場合、MINAU パラメーターは、要求された割り振りに影響を与えません。

- ゼロの値を提供した場合。

- 部分 DSCB の PD1SCALO フラグ・バイトが、シリンダー要求 (X'C0') または丸め付き平均ブロック要求 (X'41') を示している場合。

そうでない場合、DSSIZE の値は MINAU の値の倍数でなければなりません。

PDSCB=addr または *(reg)*

部分 DSCB のアドレス (ALLOC=MOV の場合) またはストレージ内の完全 format-1 DSCB のアドレス (ALLOC=ABS の場合) を指定します。この DSCB は、割り振られるデータ・セットの format-1 DSCB を構成するためのモデルとして使用されます。

addr-RX タイプ・アドレス、 *(reg)*-(0)、(2-12)

PDSCB パラメーターに対して値を提供し、PD1FMTID フィールドを X'F1' に初期化します。

PDSDIR=addr または *(reg)*

区分データ・セット (PDS) の場合、256 バイト・ディレクトリー・ブロックの数を指定します。

addr-RX タイプ・アドレス

ストレージ内にある、256 バイト PDS ディレクトリー・ブロックの数を含んでいるフルワードのアドレスを指定します。

(reg)-(0)、(2-12)

256 バイト PDS ディレクトリー・ブロックの数が入っているレジスターを指定します。

区分編成が指定されている場合、PDSDIR に対して値を提供します。

- format-1 DSCB の DS1DSORG フラグ・バイトは X'02' (ALLOC=ABS) です。
- 部分 DSCB の PD1DSORG フラグ・バイトは X'02' (ALLOC=MOV) です。

ALLOC=MOV 要求の場合、部分 DSCB の PDDIRQTY フィールドに PDSDIR の値を指定できます。PDDIRQTY フィールドは、REALLOC パラメーター・リストの PDSDIR の値がゼロの場合にのみ使用されます。

PDS が示されていない場合、PDSDIR に対して値を指定しないでください。

UCB=(reg)

データ・セットを割り振るボリュームの UCB のアドレスを指定します。

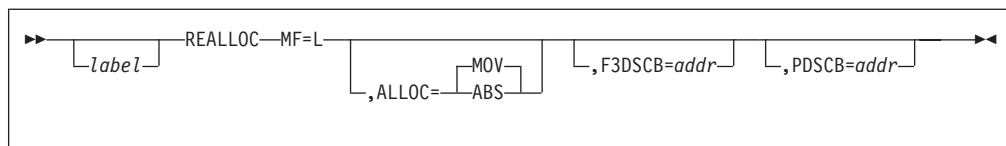
UCB アドレスは、キャプチャーされた UCB のアドレスでも、16 MB 境界より上または下の実 UCB のアドレスでも構いません。呼び出し側が 31 ビットの場合、高位バイトは UCB アドレスの一部であり、24 ビットの UCB アドレスが渡される場合は、ゼロにクリアする必要があります。ボリュームはマウント済みであり、マウントされたままでなければなりません。UCB コピーではなく、UCB のアドレスを使用してください。

(reg)-(0)、(2-12)

装置の UCB アドレスが入っているレジスターを指定します。

REALLOC - リスト形式

リスト形式の REALLOC マクロのフォーマットは、次のとおりです。パラメーターの説明は、実行形式の項を参照してください。

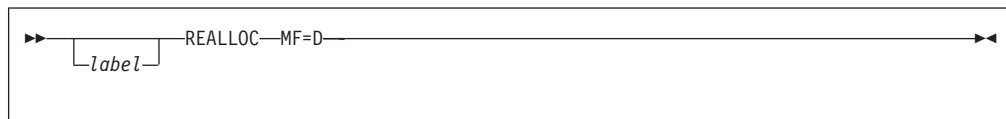


制約事項:

1. 実行形式のパラメーター DSSIZE、MINAU、PDSDIR、および UCB は、リスト形式では指定できません。
2. リスト形式の ALLOC パラメーターは、アセンブリー時に REALLOC マクロによって行われるテスト、およびパラメーター・リストの内容に影響を与えます。
3. ALLOC の値は、実行形式での指定またはデフォルトと同じです。

REALLOC - DSECT 形式

DSECT 形式の REALLOC マクロのフォーマットは、次のとおりです。



REALLOC パラメーター・リスト

拡張 DSECT 形式のパラメーター・リストは、次のとおりです。

表 12. REALLOC パラメーター・リスト

オフセット	長さ、またはビット・パターン	フィールド名	説明
0(0)		REALPL	DSECT 名
0(0)	4	RALPLID	"REALLOC" に対する EBCDIC "REAL"
4(4)	2	RALNGTH	パラメーター・リストの長さ
6(2)	2	RAERRCDE	DADSM 作成からのエラー・コード
8(1)	1	RALFLAG	フラグ・バイト
	1... ..	RALABS	0 は ALLOC=MOV を意味します。1 は ALLOC=ABS を意味します。
	.1.. ..	RALVTOCE	VTOCENQD=YES
9(9)	3		予約済み
12(C)	4	RALDSSZ	データ・セット・サイズ
16(10)	4	RALMAU	最小割り振り単位
20(14)	4	RALPDSCB	部分 DSCB のアドレス
24(18)	4	RALUCB	UCB のアドレス
28(1C)	4	RALDQTY	PDS ディレクトリーの数量
32(20)	4	RAL2DSCB	format 2 DSCB のアドレス

表 12. REALLOC パラメーター・リスト (続き)

オフセット	長さ、またはビット・パターン	フィールド名	説明
36(24)	4	RAL3DSCB	format 3 DSCB のアドレス
40(28)	0	RALEND	リストの終わりの後のバイト
	40	RALENGTH	パラメーター・リストのシンボルの長さ

REALLOC からの戻りコード

制御は、REALLOC マクロによって生成された命令の後の命令に戻ります。表 13 に示す該当の REALLOC 戻りコードがレジスター 15 に入ります。

REALLOC は、4 バイトの診断情報をレジスター 0 に戻します。「z/OS DFSMSdfp 診断解説書」を参照してください。

エラーが発生した場合、REALLOC は、障害に関連した状況情報から成るメッセージ IEC614I を発行します。REALLOC 戻りコードによって示される状態についての説明は、表 13 を参照してください。

例外: これは DADSM CREATE 戻りコードの累積のリストです。一部のコードは REALLOC マクロには適用されない場合があります。

表 13. DADSM CREATE 戻りコード

戻りコード	説明
00(X'00')	レジスター 0 に戻された 4 バイトの診断情報がすべてゼロの場合、これはデータ・セットが正常に作成されたことを示しています。ゼロでない場合は、「z/OS DFSMSdfp 診断解説書」の『DADSM/CVAF Diagnostic Aids』のセクションを参照して、障害関連の状態を判別してください。
04(X'04')	重複するデータ・セット名。
08(X'08')	VTOC または VTOC 索引内に余地がありません。
12(X'0C')	永続入出力エラーまたは CVAF エラー。
16(X'10')	要求された絶対トラックが利用不可です。
20(X'14')	要求された数量が利用不可です。
24(X'18')	平均レコード長が 65,535 バイトを超えています。
28(X'1C')	現在、ISAM はサポートされていません。要求は失敗します。
48(X'30')	無効な DADSM REALLOC パラメーター・リスト。
52(X'34')	無効な JFCB または部分 DSCB ポインター。
56(X'38')	ボリューム上にディレクトリー用の十分なスペースがありません。
60(X'3C')	REALLOC ALLOC=ABS は OS VTOC 上ではサポートされません。
64(X'40')	無効なユーザー・ラベル要求。
68(X'44')	無効な UCB ポインター。UCB コピーではなく、UCB のアドレスが必要です。
72(X'48')	VSE VTOC を OS VTOC に変換できません。
76(X'4C')	新規データ・セット用のスペース・パラメーターが与えられていないか、絶対トラック・ゼロにゼロ・スペースを要求しました。
104(X'68')	無効なスペース・サブパラメーター。
108(X'6C')	ABSTR 要求。
116(X'74')	ユーザー・ラベルがサポートされません。

表 13. DADSM CREATE 戻りコード (続き)

戻りコード	説明
120(X'78')	REALLOC パラメーターにおける DSSIZE と MINAU の無効な組み合わせ。
124(X'7C')	DSSIZE が MINAU の倍数ではありません。
128(X'80')	要求されたディレクトリー・スペースが 1 次スペースより大きくなっています。
136(X'88')	無効な FMT3 DSCB ポインター。
148(X'94')	VTOC 内でエクステントが重複しています。
156(X'9C')	おそらく VTOC エラーが原因で DADSM CREATE が終了しました。
164(X'A4')	VSE スタック・パック・フォーマットが原因で、割り振りが終了しました。
168(X'A8')	RACDEF が失敗しました。データ・セットはすでに定義済みです。
172(X'AC')	ユーザーはデータ・セットを定義する許可が与えられていません。
176(X'B0')	インストール・システム出口はこの要求をリジェクトし、戻りコード 8 を戻しました。
180(X'B4')	インストール・システム出口は要求をリジェクトし、戻りコード 4 を戻しました。
184(X'B8')	RACF は指定のモデリングを使用するように定義されていますが、モデルが見つかりません。
188(X'BC')	無効な FMT2 DSCB ポインター。
192(X'C0')	要求されたデータ・セット作成は、SMS によって許可されませんでした。
196(X'C4')	要求されたデータ・セット作成は不可能でした。レジスター 0 に追加の診断情報が入っています。
200(X'C8')	PDSE ディレクトリーを作成できませんでした。
204(X'CC')	VTOC ENQ 関連の障害。
208(X'D0')	データ・セットの割り振り中に入出力エラーが発生しました。データ・セットは削除されます。
212(X'D4')	ボリューム上に分割シリンダー・データ・セットが存在するために、要求は失敗しました。
216(X'D8')	このタイプのデータ・セットの場合、要求された 1 次数量は 65,535 トラックを超えることはできません。

CVAF マクロを使用した VTOC へのアクセス

この節で説明する共通 VTOC アクセス機能 (CVAF) マクロおよびトラックは、以下のもので構成されます。

- CVAFDIR は、DSCB に直接アクセスします。
- CVAFDSM は、ボリューム・フリー・スペース情報を取得します。
- CVAFFILT は、1 つ以上の DASD データ・セットの一連の DSCB を読み取ります。
- CVAFSEQ は、以下を検索します。
 - アクティブ VTOC 索引からデータ・セット名。
 - 物理順に DSCB
 - データ・セット名順に DSCB (索引が必要)
- CVAFST は、DASD ボリュームがアクティブ VTOC 索引を持っているかどうかを判別します。

379 ページの『付録 A. CVAF VTOC アクセス・マクロ』に、これらのマクロの説明とその使用例があります。

CVAF を呼び出すときは、プログラムは 24 ビットまたは 31 ビット・アドレッシング・モードのいずれも使用できます。31 ビット・モードの場合、56 ページの図 6 に示された制御ブロックは、16 MB 境界より上にあります。この領域には、プログラムのストレージ・キーでアクセス可能でなければなりません。

注: 呼び出し側の AMODE に一致する UCB アドレスを提供する必要があります。つまり、AMODE=24 は、24 ビット UCB アドレスを必要とし、一方、AMODE=31 は、31 ビット UCB アドレスを必要とします。

逐次化と更新

CVAF では、要求のために必要なすべてのシステム・リソースの逐次化をユーザーが準備する必要があります。システム・リソースを適切に逐次化して、DSCB または VIR セットに対する複数の CVAFFILT 要求を避けることができる場合にのみ、CVAF から戻される複数のデータ・エレメント(DSCB または VIR セット) の健全性が確保されます。逐次化によるシステム・パフォーマンスの損失の可能性と、データ健全性の損失の可能性とを比較検討してください。

適切な逐次化を行わずに更新すると、ボリュームの VTOC、VTOC 索引、または関連データ・セットの健全性が損なわれる危険があります。

CVAF は、許可プログラムからの、ボリュームの VTOC または索引を変更するための要求にのみ応じます。

CVAF は、許可プログラムが、SYSTEMS の有効範囲で、SYSVTOC の qname (大分類名) およびボリュームの通し番号の rname (小分類名) に関する排他的 RESERVE (または ENQ) を保持するものと想定しています。グローバル・リソース逐次化または同等の機能がアクティブの場合、この RESERVE をより効率的にすることができます。

SYSVTOC qname は、データ・セットの format-1 DSCB へのアクセスを逐次化しません。ユーザーは、処理 OLD、MOD、または NEW (SHR ではなく) を使用してデータ・セットを割り振ることにより、逐次化を提供できます。これによって、適正な ENQ が行われ、他のジョブはそのデータ・セットの format-1 DSCB を更新できないことが確実にになります。

ボリュームの識別

プログラムが許可されている場合、UCB のアドレスを指定することによって、CVAFDIR、CVAFDSM、CVAFFILT、および CVAFSEQ マクロに対してボリュームを識別できます。これらのマクロは、UCB コピーのアドレスは受け入れられません。プログラムが許可されていない場合は、ボリュームの VTOC に対してオープンされた SAM または EXCP DEB のアドレスを指定します。

データ・エクステント・ブロック (DEB) は、RDJFCB および OPEN TYPE=J マクロを使用して INPUT 用に DCB をオープンすることにより入手できます。(OPEN TYPE=J マクロを発行した後、DCB の DCBDEBA フィールドに DEB のアドレスが入ります。) DCB の DDNAME は、アクセスする VTOC の装置に割り振られた

DD ステートメントを識別しなければなりません。プログラムは、RDJFCB マクロを発行した後、format-4 DSCB (44 バイトの X'04') のデータ・セット名を使用して JFCBDSNM フィールドを初期化する必要があります。RDJFCB マクロについては、206 ページの『RDJFCB マクロ仕様』で説明しています。OPEN マクロについては、223 ページの『OPEN - JFCB 処理のためのデータ制御ブロックの初期化』で説明しています。

CVAF マクロ呼び出しが IOAREA=KEEP を指定している場合、異なる CVAF パラメーター・リスト (CVPL) を使用する後続の CVAF 呼び出しは、UCB および DEB キーワードを省略し、IOAREA キーワードを使用して他の CVPL から IOAREA アドレスを提供できます。

CVPL (CVAF パラメーター・リスト) の生成

CVAFDIR、CVAFDSM、CVAFFILT、および CVAFSEQ マクロは、CVPL を使用してパラメーターを CVAF に渡します。CVAFTST マクロが展開して、その唯一のパラメーター (UCB アドレス) をレジスター 1 に提供し、該当する CVAF モジュールを呼び出します。CVAF は、CVAF 要求に関連した情報を CVPL に戻します。CVAFTST マクロは、UCB または UCB コピーのアドレスを受け入れます。無許可プログラムは、UCBSCAN マクロを使用して COPY、UCBAREA、CMXTAREA、および DCEAREA キーワードを指定することにより、UCB のコピーを入手できません。UCB コピーは、拡張も含めて、16 MB 境界より下のワード境界上になければなりません。DCEAREA でアクセスされるデータは、16 MB 境界より上にあっても構いません。詳細については、「z/OS ハードウェア構成定義(HCD) 計画」を参照してください。

MF=L または MF=I (デフォルト) をサブパラメーターとして付けて CVAFDIR、CVAFDSM、CVAFFILT、または CVAFSEQ マクロを指定すると、CVPL を生成します。戻り時に、CVPL 内の CVHVT ビットは、アクセスした VTOC が索引付きか、索引なしかを示します。エラーが発生した場合、CVSTAT フィールドにフィールドバック・データが入ります。CVAF 入出力域アドレスが CVIOAR フィールドに戻され、CVAF フィルター保管域が CVFSA フィールドに戻れます。生成された CVPL を使用して、当初に指定したのとは別の機能を実行するには、MF=E キーワードを使用します。

CVAF フィルター要求を指定するには、CVAFFILT マクロによって生成される CVPL を使用します。CVAFFILT マクロは、他の CVAF マクロによって生成される CVPL (全長 = X'40') より 4 バイト長い CVPL (全長 = X'44') を生成します。

ICVAFPL マクロは、CVPL をマップします。CVPL のフォーマットを表 14 に示します。

注: CVCTAR5 で始まる区域が生成されるのは、ICVAFPL の呼び出しで CVPLX=YES マクロ変数が指定されている場合のみです。

表 14. CVAF パラメーター・リストのフォーマット

オフセット	長さ、またはビット・パターン	名前	説明
0 (X'00')	4	CVPL CVLBL	EBCDIC 'CVPL'

表 14. CVAF パラメーター・リストのフォーマット (続き)

オフセット	長さ、またはビット・パターン	名前	説明
4 (X'04')	2	CVLTH	CVPL の長さ: CVAFFILT 以外のマクロの場合は 64 (X'40')。CVAFFILT マクロの場合は 68 (X'44')。
6 (X'06')	1	CVFCTN	機能バイト (47 ページの表 15 を参照)。
7 (X'07')	1	CVSTAT	状況情報。
8 (X'08')	1	CVFL1	最初のフラグ・バイト。
	1	CV1IVT	索引付き VTOC にアクセスしました。
	. 1	CV1IOAR	IOAREA=KEEP
	. . 1	CV1PGM	BRANCH=(YES,PGM)
	. . . 1	CV1MRCDS	MAPRCDS=YES
 1	CV1IRCDS	IXRCDS=KEEP
 100	CV1MAPIX	MAP=INDEX
 010	CV1MAPVT	MAP=VTOC
 001	CV1MAPVL	MAP=VOLUME
9 (X'09')	1	CVFL2	2 番目のフラグ・バイト。
	1	CV2HIVIE	HIVIER=YES
	. 1	CV2VRF	VRF データが存在します。
	. . 1	CV2CNT	COUNT=YES
	. . . 1	CV2RCVR	RECOVER=YES
 1	CV2SRCH	SEARCH=YES
 1	CV2DSNLY	DSNONLY=YES
 1 .	CV2VER	VERIFY=YES
 1	CV2NLEVEL	新規の最高レベル VIER (出力)。
10 (X'0A')	1	CVFL3	3 番目のフラグ・バイト。
	1	CV3FILT	FLTAREA=KEEP
	. 1	CV3IXERR	索引エラーが検出されました。
	. . 1	CV3PTR31	AMODE31
	. . . 1	CV3AVT	書き込み禁止バイパス・エラー。
 1	CV3ENQD	CVAF 呼び出し側は、その VTOC を予約した別のタスクを使用して逐次化しました。
 1 . .	CV3NOPIN	呼び出し側は UCBPIN なしを要求しました。
 1 .	CV3HLIXP	上位レベル索引のブルーニングが行われました。
 1	CV3RTA4B	CVAFDSM RTA4BYTE が指定されました。
11 (X'0B')	1		予約済み。
12 (X'0C')	4	CVUCB	UCB アドレスは、キャプチャーされたアドレスでも、16 MB 境界より上または下の実アドレスでも構いません。
16 (X'10')	4	CVDSN	データ・セット・アドレス。
16 (X'10')	4	CVFCL	フィルター基準リスト・アドレス。
20 (X'14')	4	CVBUFL	バッファー・リスト・アドレス。
24 (X'18')	4	CVIRCDS	索引 VIR のバッファー・リスト・アドレス。
28 (X'1C')	4	CVMRCDS	マップ VIR のバッファー・リスト・アドレス。
32 (X'20')	4	CVIOAR	入出力域アドレス。
36 (X'24')	4	CVDEB	DEB アドレス。

表 14. CVAF パラメーター・リストのフォーマット (続き)

オフセット	長さ、またはビット・パターン	名前	説明
40 (X'28')	4	CVARG	引数アドレス。
44 (X'2C')	4	CVSPACE	SPACE パラメーター・リスト・アドレス。
48 (X'30')	4	CVEXTS	エクステンツ・テーブル・アドレス。
52 (X'34')	4	CVBUFL2	新規の VRF VIXM バッファー・リスト・アドレス。
56 (X'38')	4	CVVRFDA	VRF データ・アドレス。
60 (X'3C')	4	CVCTAR	カウント域アドレス。
64 (X'40')	4	CVFSA	フィルター保管域。
68 (X'44')	4	CVCTAR5	CCHHR 域戻りアドレス。
72 (X'48')	28	*	未使用。

CVPL の CVFCTN フィールドの可能な内容とその意味は、次のとおりです。

表 15. CVPL の CVFCTN フィールド - 内容と定義

値	名前	説明
X'01'	CVDIRD	CVAFDIR ACCESS=READ
X'02'	CVDIWR	CVAFDIR ACCESS=WRITE
X'03'	CVDIRLS	CVAFDIR ACCESS=RLSE
X'04'	CVSEQGT	CVAFSEQ ACCESS=GT
X'05'	CVSEQGTE	CVAFSEQ ACCESS=GTEQ
X'06'	CVDMIXA	CVAFDSM ACCESS=IXADD
X'07'	CVDMIXD	CVAFDSM ACCESS=IXDLT
X'08'	CVDMALC	CVAFDSM ACCESS=ALLOC
X'09'	CVDMRLS	CVAFDSM ACCESS=RLSE
X'0A'	CVDMMAP	CVAFDSM ACCESS=MAPDATA
X'0E'	CVFIRD	CVAFFILT ACCESS=READ
X'0F'	CVFIRES	CVAFFILT ACCESS=RESUME
X'10'	CVFIRLS	CVAFFILT ACCESS=RLSE
X'AA'	CVDMMAPX	CVAFDSM ACCESS=MAPDATA RTA4BYTE=YES

バッファー・リストの使用

バッファー・リストは、プログラムによって取得されて初期化された、1 つ以上のチェーニングされた制御ブロック (それぞれ、ヘッダーとバッファー・リスト項目を持つ) から構成されます。ヘッダーは、バッファー・リストが DSCB 用かどうかを示します。項目は、バッファーを指し示し、記述します。バッファー・リストは、2 とおりの方法で作成できます。

- 直接に。この場合は、読み取りまたは書き込み DSCB の引数とバッファー・アドレスを入力します。
- 間接的に (CVAF によって)。この場合は、IXRCDS=KEEP および/または MAPRCDS=YES キーワードをコーディングします。

ICVAFBFL マクロは、CVAF バッファー・リストをマップします。48 ページの表 16 は、バッファー・リスト・ヘッダーのフォーマットを示しています。49 ページの表 17 は、バッファー・リスト項目のフォーマットを示しています。

バッファ・リスト・ヘッダー

バッファ・リスト・ヘッダーは、バッファ・リストが DSCB 用のバッファを記述しているかどうかを示します。CVAF が DSCB の読み取りまたは書き込み要求を処理するには、DSCB ビットが 1 に設定され、VIR ビットがゼロに設定されている必要があります。ユーザーのプログラムの記憶保護キーにバッファ・リストとバッファを提供してください。ユーザーが ACCESS=RLSE をコーディングした場合にのみ、CVAF は記憶保護キーとバッファ・リスト・ヘッダー内のサブプール・フィールドを使用します。

各バッファ・リスト・ヘッダーは、ヘッダーの直後のバッファ・リストに項目数が入ります。

順方向チェーン・アドレスは、バッファ・リスト同士を連結します。バッファ・リスト・ヘッダーのフォーマットを表 16 に示します。

表 16. バッファ・リスト・ヘッダーのフォーマット

オフセット	長さ、またはビット・パターン	名前	説明
0 (X'00')	8	BFLHDR	バッファ・リスト・ヘッダー。
0 (X'00')	1	BFLHNOE	項目の数。
1 (X'01')	1	BFLHFL	キーおよびフラグ・バイト。
	xxxx	BFLHKEY	バッファ・リストとバッファの記憶保護キー。
 1	BFLHVIR	予約済み。
 1 . . .	BFLHDSCB	バッファ・リスト項目は DSCB を記述します。
 xx		予約済み。
2 (X'02')	1		予約済み。
3 (X'03')	1	BFLHSP	バッファ・リストとバッファのサブプールを識別します。
4 (X'04')	4	BFLHFCHN	次のバッファ・リストの順方向チェーン・アドレス。

バッファ・リスト項目

バッファ・リストには、1 つ以上の項目が含まれています。各項目は、バッファ・アドレス、DSCB バッファの長さ、および引数を提供し、引数が RBA、TTR、または CCHHR のいずれであるかを示します。フィールドおよびビットの使用法を以下に示します。

- DSCB バッファの場合、RBA ビットは 0 でなければならず、TTR または CCHHR ビットのどちらかを 1 に設定する必要があります (両方が 1 であってはなりません)。
- BFLESKIP ビットは、項目を無視させます。
- BFLEIOER ビットは、CVAF からの出力標識で、DSCB の読み取りまたは書き込み中に入出力エラーが発生したことを示します。
- BFLELTH フィールドは、バッファの長さです。DSCB バッファの場合、長さは 96 または 140 でなければなりません。
- BFLEARG フィールドは、DSCB の引数 (アドレス) です。BFLECHR、BFLETTR、または BFLERBA ビットを 1 に設定して、5 バイト・

フィールドのフォーマットを指定します。それぞれの BFLEARG 値とフォーマットは、次のとおりです。

値	フォーマット
CCHHR	5 バイト CCHHR
TTR	0TTR0
RBA	1 バイトの 0 と、その後続く 4 バイトの相対バイト・アドレス。

BFLEARG の値は、その要求に関連した変数によって決まります。これについては、以下で、要求に関するトピックの中で説明します。

バッファ・リスト項目のフォーマットを表 17 に示します。

表 17. バッファ・リスト項目のフォーマット

オフセット	長さ、またはビット・パターン	名前	説明
0 (X'00')	12	BFLE	バッファ・リスト項目。
0 (X'00')	1	BFLEFL	フラグ・バイト。
	100	BFLE RBA	引数は RBA です。
	010	BFLE CHR	引数は CCHHR です。
	001	BFLE TTR	引数は TTR です。
	. . . 1	BFLE AUPD	CVAF は引数フィールドを更新しました。
 1	BFLE MOD	バッファ内のデータが変更されました。
 1	BFLE SKIP	この項目を飛ばします。
 1	BFLE IOER	入出力エラー。
 1	BFLE NOVR	CVAFDIR を使用して 96 バイト DSCB を書き込む場合、DSCB キーとデータ・セット名の比較をバイパスします。
1 (X'01')	1		予約済み。
2 (X'02')	1	BFLE LTH	DSCB バッファの長さ。
3 (X'03')	5	BFLE ARG	DSCB の引数。
4 (X'04')	3	BFLE ATTR	DSCB の TTR。
4 (X'04')	4	BFLE ARBA	予約済み。
8 (X'08')	4	BFLE BUF	バッファ・アドレス。

マクロ ICVEDT02 を使用したエクステント領域のマッピング

CVAFDSM マクロを使用し、索引付き VTOC または索引なし VTOC に対して RTA4BYTE=YES を指定した場合、ICVEDT02 マッピング・マクロを使用してエクステント領域がマッピングされます。

ICVEDT02 マッピング・マクロのフォーマットは、次のとおりです。

表 18. ICVEDT02 マッピング・マクロのフォーマット

オフセット	バイト	名前	説明
0 (X'00')	8	DT2X7EYE	ID=『ICVEDT02』
8 (X'08')	4	DT2X7LEN	ICVEDT02 の長さ。
12 (X'0C')	1	DT2X7LEV	レベル番号。

表 18. ICVEDT02 マッピング・マクロのフォーマット (続き)

オフセット	バイト	名前	説明
13 (X'0D')	1	DT2X7FLG	フラグ・バイト。
14 (X'0E')	2	DT2X7NF0	作成された format-0 DSCB の数。
16 (X'10')	5	DT2X7CSR	CCHHR カーソル・フィールド。システムによって使用されます。呼び出し側によってゼロに初期化されます-その後は変更しないでください。
21 (X'15')	3	DT2X7RES	予約済み。
24 (X'18')	4	DT2X7RTA	相対トラック・アドレス (RTA) カーソル・フィールド。システムによって使用されます。呼び出し側によってゼロに初期化されます-その後は変更しないでください。
28 (X'1C')	4	DT2X7RE2	予約済み。
32 (X'20')	4	DT2X7ENT	エクステント記述子項目の数。
-	-	-	エクステント記述子配列。
36 (X'24')	8	DT2ENTRY	エクステントごとに 1 項目。
-	4	DT2RTAST	エクステント開始の RTA。
-	4	DT2RTAED	エクステント終了の RTA+1。

DSCB への直接アクセス

CVAFDIR マクロを使用して、DSCB の読み取りまたは書き込みを行うことができます。これについては、379 ページの『CVAFDIR マクロの概要および仕様』で説明します。CVAFDIR 呼び出しの後、CVPL の CV1IIVT ビットをテストすると、VTOC が索引付きか、索引なしかを判別できます。

DSCB バッファが 96 バイトの場合、操作が書き込みで BFLENOVR ビットがオンでない限り、CVAF はチャンネル・プログラムを発行して、DSCB 内のキーがユーザーの入力した 44 バイトのデータ・セット名に一致するかどうかを検査します。

バッファが 96 バイト書き込み用であり、BFLEFL の BFLENOVR ビットが 1 に設定されている場合、CVAF はキー検査をスキップし、パフォーマンスが向上します。ユーザーが提供するデータ・セット名が正しいと確信できない場合は、BFLENOVR ビットを 0 に設定してください。BFLENOVR ビットが 0 に設定されている場合、CVAF はキーが一致しない限り書き込みを実行しません。

CVAF がキー検査を実行しており、DSCB キーがユーザー提供のデータ・セット名に一致していない場合、CVAF は指定された BFLEARG を無視し、次節の『DSCB の読み取りまたは書き込みのためのデータ・セット名の指定』で説明する規則を使用して、DSCB を書き込みます。

DSCB の読み取りまたは書き込みのためのデータ・セット名の指定

データ・セット名のみを指定して単一の DSCB の読み取りまたは書き込みを行うには (つまり、BFLEARG がゼロ)、ACCESS=READ または ACCESS=WRITE を指定します。

データ・セット名のアドレスは DSN キーワードで指定します。バッファ・リストのアドレスは BUFLIST キーワードで指定します。これらの領域と関連バッファは、ユーザーのプログラムの記憶保護キー内になければなりません。

バッファ・リストには、スキップ・ビットがオフで、96 バイト・バッファへのポインターを持つ、少なくとも 1 つのバッファ・リスト項目を含める必要があります。140 バイト・バッファは使用しないでください。バッファ・リストは相互にチェーニングできますが、CVAF は最初の適格項目のみを使用します。

索引付き VTOC の場合、CVAF はデータ・セット名の索引を検索します。データ・セット名が見つかり、DSCB 引数がバッファ・リスト項目に入れられ、DSCB の読み取りまたは書き込みに使用されます。索引内でデータ・セット名が見つからない場合、CVAF は VTOC のキー検索を実行します。

索引なし VTOC の場合、CVAF はチャンネル・プログラムを使用して VTOC のキー検索を実行し、データ・セット名を見つけて、DSCB の読み取りまたは書き込みを行います。データ・セット名が見つかり、CVAF は DSCB 引数をバッファ・リスト項目に入れます。

バッファ・リスト項目に戻される DSCB 引数は、バッファ・リスト項目内の BFLECHR または BFLETR ビットによって決められたフォーマットになります。

CVAF が VTOC 内でデータ・セット名を見つけられない場合、戻りコード 4 がレジスター 15 に示され、エラー・コード 1 が CVSTAT フィールドに入ります。

DSCB の位置の指定

DSCB の位置 (つまり、BFLEARG) を指定して DSCB の読み取りまたは書き込みを行うには、ACCESS=READ または ACCESS=WRITE を指定します。

データ・セット名のアドレスは DSN キーワードで指定し、バッファ・リストのアドレスは BUFLIST キーワードで指定します。これらの領域と関連バッファは、ユーザーのプログラムの記憶保護キー内になければなりません。

バッファ・リストは、スキップ・ビットがオフで、96 バイト・バッファへのポインターを含むバッファ・リスト項目を持っている必要があります。バッファ・リストは相互にチェーニングできますが、CVAF は最初の適格項目のみを使用します。

バッファが 140 バイト読み取りまたは書き込み用である場合、CVAF はチャンネル・プログラムを発行して、バッファ・リスト項目で指定された位置にある DSCB の読み取りまたは書き込みを行います。CVAF は、指定されたデータ・セット名を無視します。VERIFY=YES を指定した場合、CVAF は書き込みチャンネル・プログラムを発行する前に、指定の DSCB が format-0 DSCB であるかどうかを検査します。

CVAF で取得したバッファとバッファ・リストの解放

CVAF によって獲得されたバッファとバッファ・リストは、次の方法で解放できます。

- ACCESS=RLSE を指定して CVAF 呼び出しを発行し、BUFLIST キーワードでバッファ・リスト・アドレスを指定する。
- MAPRCDS=NO または MAPRCDS=(NO,addr) をコーディングし、任意の ACCESS を指定して、MAP レコード・バッファ・リストを解放する。

VTOC の使用

- IXRCDS=NOKEEP または IXRCDS=(NOKEEP,*addr*) をコーディングし、任意の ACCESS を指定して、索引レコード・バッファ・リストを解放する。

CVAF は、空になった適格バッファとバッファ・リストをすべて解放します。適格バッファとは、スキップ・ビットがオフのバッファ・リスト項目によって指し示されているバッファを言います。CVAF は、スキップ・ビットがオンのバッファ・リスト項目を含んでいない場合、バッファ・リストを解放します。バッファ・リストが相互にチェーニングされている場合、CVAF は該当するバッファ・リストをすべて検査して、解放します。複数の場所にアドレスを指定して、CVAF に対して同じバッファ・リストを 2 度解放するように要求してはなりません。

DSN または DSCB への順次アクセス

CVAFSEQ マクロを使用して、データ・セットに関する情報を戻すことを要求できます。

- 索引 (データ・セット名) 順に 1 つの format-1 または format-4 DSCB。
- 物理順に 1 つ以上の DSCB。(無許可の場合は、1 つの DSCB しか要求できません。)
- 索引内の次のデータ・セット名。

CVAF は、BUFLIST キーワードで識別されたバッファー内に DSCB を読み取ります。マクロについての追加情報は、411 ページの『CVAFSEQ マクロの概要および仕様』を参照してください。

バッファー・リストを使用して、読み取る各 DSCB の引数を指定します。索引アクセスの場合、バッファー・リストで 96 バイト DSCB を要求します。物理順次アクセスの場合は、140 バイト DSCB を要求します。

索引順を選択した場合、CVAF は索引によって指示された各 format-1 または format-4 DSCB を戻します。索引内のデータ・セット名のみを戻す (DSCB は戻さない) 場合は、DSNONLY=YES を指定します。ユーザーが CVAFSEQ を発行するたびに、CVAF は CVAFSEQ マクロで指定された DSN 域を、読み取った各 DSCB のデータ・セット名で更新します。CVAF は、ARG キーワードで提供された引数域に DSCB の CCHHR も戻します。

注: バッファーに対して提供された長さが 140 バイトの場合 (フィールド BFLELTH)、CVAFSEQ から戻される DSCB には、常に DSCB のキー部分が含まれています。ただし、マクロ IECSDSL1 内の構造 IECSDSF4 のマッピングには、DSCB のキー部分は含まれません。そのため、この構造に定義されている format-4 DSCB フィールドを使用する場合 (フィールドはすべて DS4 で始まります)、IECSDSF4 構造の最初のフィールドの開始点を調整して、戻されたバッファーの 44 バイト内側の位置に一致させることが必要です。

索引アクセス (DSN 順) の開始

索引アクセス (DSN 順) を開始するには、DSN 域に 44 バイトの 2 進ゼロを提供するか (索引内の最初のデータ・セット名を示すため)、または見出し検索の開始点になるデータ・セット名を指定します。

DSN 域に戻される名前は、ACCESS キーワードの選択に応じて、提供された DSN に等しいか、それより大きい値になります。

DSNONLY=NO を指定した場合、CVAF は BUFLIST キーワードで提供されたバッファー・リストを使用して、DSCB と引数を戻します。CVAF はスキップ・ビットが 0 に設定され、非ゼロのバッファー・アドレスを持つ、バッファー・リスト内の最初の項目を使用します。バッファー・リスト項目で TTR または CCHHR ビットを 1 に設定することにより、引数のフォーマットを指定できます。どちらのビットも設定されていない場合、CVAF は CCHHR 引数を戻します。索引アクセスの場合、バッファー・リスト項目内の DSCB サイズは 96 バイトでなければなりません。

DSNONLY=YES を指定する場合は、ARG 域に CCHHR 引数を指定してください。

format-4 DSCB のデータ・セット名が索引内にあり、CVAF がその名前 (44 バイトの X'04') を戻すことがあります。format-4 DSCB の名前は、VTOC 索引の最初のデータ・セット名である可能性があります。

物理順次アクセスの開始

物理順次アクセスを開始するには、DSN=0 を指定するか、または DSN パラメーターをまったく指定しないかのいずれかです。読み取りを開始するには、最初のバッファー・リスト項目の引数フィールドをゼロ、または DSCB の引数に初期化します。引数がゼロの場合、CVAF は VTOC の開始の引数を使用します。

ACCESS キーワードは、CVAF が、引数が提供されている DSCB を読み取るのか、その後続の DSCB を読み取るのかを決めます。例えば、VTOC 内の最初の DSCB (format-4 DSCB) を読み取るには、最初のバッファー・リスト項目の BFLEARG をゼロに設定し、CVAFSEQ マクロで ACCESS=GTEQ を指定できます。その後に ACCESS=GT を指定すると、CVAF は 2 番目の DSCB (最初の format-5 DSCB) を読み取ります。バッファー・リスト項目で DSCB サイズを 140 に設定します。

プログラムが許可されている場合、CVAF は、単一の CVAF 呼び出しで、バッファー・リスト内にある限りの項目の DSCB を読み取ります。許可されていない場合、CVAF は 1 つだけ DSCB を読み取ります。

CVAF は 1 つのバッファー・リストを使用し、最初のバッファー・リストからチェーニングされた 2 番目のバッファー・リストは検査しません。プログラムが許可されている場合、CVAF はバッファー・リスト内のすべての項目を使用します。許可されていない場合、CVAF は最初の項目のみを使用します。CVAF は、スキップ・ビットは検査しません。各項目は、バッファー・アドレスと 140 に設定された長さフィールドを含んでいなければなりません。CVAF は、各バッファー・リスト項目の引数フィールドを DSCB の引数で更新します。バッファー・リスト項目で TTR または CCHHR ビットを 1 に設定することにより、引数のフォーマットを指定できます。どちらのビットも設定されていない場合、CVAF は CCHHR 引数を戻します。

CVAF は、最初の項目内の引数のみを使用して検索を開始し、後続の項目内の引数は検査しません。最初の項目にゼロ以外の引数値を指定した場合、その引数を持つ DSCB が存在しなければなりません。

CVAF は、戻りコード 4 をレジスター 15 に戻し、X'20' の値を CVSTAT フィールドに入れることにより、データ終了状態を示します。CVAF は、最後の DSCB の読み取り後のすべてのバッファー・リスト項目の引数フィールドをゼロに設定します (CVAF が 1 つも DSCB を読み取らない場合は、最初の項目はゼロです)。

CVAF は、format-0 DSCB を含めて、すべての DSCB を読み取ります。CVAF が VTOC 全体を読み取るまでは、ユーザーにはすべての DSCB が読み取られたことが確実に分かりません。索引なし VTOC の場合、format-4 DSCB の DS4HPCHR フィールドに、最後の format-1 DSCB の CCHHR が入ります。format-1 以外の DSCB は、その場所を超えた位置にあることがあります。索引付き VTOC の場合、どの DSCB が format-0 DSCB であるかに関する情報が VMDS に入ります。

CVAF フィルターを使用した DSCB セットの読み取り

CVAFFILT マクロを使用して DSCB セットを取り出し、呼び出し側プログラムによって提供されたバッファーに入れることができます。CVAF フィルター・サービスは、索引付きと索引なし VTOC の両方をサポートします。このマクロのフォーマットとパラメーターについては、400 ページの『CVAFFILT マクロの概要および仕様』で説明しています。以下では、このサービスとその要件の要約を示します。

- 1 つ以上の完全修飾データ・セット名、または 1 つの部分修飾名を指定して、DSCB を要求します。詳しくは、56 ページの『フィルター基準リスト (FCL)』、および 405 ページの『CVAFFILT の部分修飾名』を参照してください。
- CVPL に単一 DASD ボリュームを識別します。
- 修飾データ・セットごとに、CVAF フィルターは VTOC 内にチェーニングされた順序で (format-1、format-2、format-3 の順に) DSCB を戻します。CVAF は他のフォーマットの DSCB は戻しません。
- CVAF フィルター・サービスは、1 つ以上の修飾データ・セットの DSCB 情報を、呼び出し側が提供したバッファーに戻します。詳しくは、60 ページの『CVAFFILT マクロ・シーケンスの例』、および 406 ページの『CVAFFILT マクロの使用例』を参照してください。CVAF フィルター・サービスは、以下の場合、部分 DSCB チェーンは戻しません。
 - 要求された DSCB を保持するのに十分なバッファーが提供されていない場合、CVAF フィルター・サービスは、1 つ以上の完全な DSCB チェーンおよび/または状況コード (CVPL 内の CVSTAT) を戻します。状況コードは、RESUME CVAF 呼び出しを使用して、残り (または、追加) の DSCB を取得できるかどうかを示します。具体的な情報については、『RESUME 機能』を参照してください。
 - バッファーの総数がデータ・セットの完全な DSCB チェーンを含めるには不十分である場合、CVAF フィルター・サービスは FCL 内に FCLDSNST バイトを設定して、そのデータ・セットを無視し、次の修飾データ・セットを処理します。この状態を避けるために、最小 11 個の DSCB バッファ (123 エクステンション限度のデータ・セットを入れるのに十分な大きさ) を提供してください。

RESUME 機能

提供されたバッファーが不十分であるためにフィルター・サービスが終了した場合、RESUME 機能に必要な情報がフィルター保管域に保管されます。(初期 CVAFFILT で FLTAREA=KEEP を指定すると、フィルター保管域が割り振られます。)

RESUME 処理を正しく実行できるようにするには、要求されたボリューム (CVDEB、CVUCB、または保管された IOAREA によって識別) と、ユーザーの FCL と、CVAF の FSA との間関係を保守する必要があります。この要件を順守すれば、複数の CVAF フィルター・サービス操作を 1 つ以上の DASD ボリューム上で同時に開始したり、再開したりすることができます。各論理要求に関連した READ/RESUME/RELEASE シーケンスの期間中、固有の CVPL と FCL を提供することにより、この関係を確保できます。

表 19. フィルター基準リスト・ヘッダーのフォーマット (続き)

オフセット	バイト	名前	説明
	...	x xxxx	予約済み。
9 (X'09')	1	FCL2FLAG	状況フラグ・バイト。
	1	FCL2SEQ	CVAFFILT は順次 VTOC アクセスを実行しました。
	FCL2SDIR	CVAFFILT は順次 VTOC アクセスを実行しましたが、少なくとも 1 回は直接 DSCB 読み取りを行いました。
		予約済み。
10 (X'0A')	6	FCLDRSV	予約済み。

FCLID 'FCLbb' の 4 文字の EBCDIC 定数でなければなりません。(ここで bb は、ブランクを表します。)

FCLCOUNT リストに提供されたデータ・セット名項目 (FCLDSN) の数を指定します。初期 CVAFFILT 呼び出しと後続の RESUME 操作の間に、このパラメーターを変更しないでください。

- 部分修飾データ・セット名を指定する場合は、FCLCOUNT = 1 を指定します。部分修飾データ・セット名のフォーマットは、405 ページの『CVAFFILT の部分修飾名』を参照してください。
- 完全修飾名のリストを指定した場合、CVAFFILT は FCLCOUNT で指定された個数の名前のみを処理します。

FCLDSCBR 単一の CVAFFILT 呼び出しによって呼び出し側のバッファーに戻される DSCB 項目の総数 (format-1、format-2、および format-3 を含む) を示します。

データ・セットを正常に処理した後で CVAF にエラーが発生した場合、ユーザーは以下のようにすることができます。

1. 各 READ および RESUME 呼び出しの前に、FCLDSCBR を 0 に初期化します。
2. CVAF フィルター・サービスから戻った時点で、FCLDSCBR によって示された個数の DSCB を処理します。
3. その後で、CVAF 戻りコードと CVSTAT を解釈します。

FCL1FLAG このフラグ・バイトを使用して、ACCESS=READ の要求を定義します。後続の RESUME 要求は、フィルター保管域 (FSA) 内のこれらのビットのコピーを参照します。

FCL1LIST 完全修飾データ・セット名のリストを指定する場合、このビットを 1 に設定します。単一の部分修飾データ・セット名を指定する場合、このビットを 0 に設定します。

FCL1ORDR CVAF が FCLDSN エレメントの配置によって暗黙指定されるデータ・セット名順に DSCB チェーンを戻すことを指定する場合は、このビットを 1 に設定します。次のことに注意してください。

- これにより、CVAF は format-1 DSCB を戻す順序を判別できるため、パフォーマンスが向上することがあります。
- CVAF は所定のデータ・セットの DSCB を format-1、format-2、format-3 の順で戻します。
- 単一の部分修飾データ・セット名を指定した場合、このフィールドは使用されません。

FCL1EQF1 CVAF がデータ・セット名の format-1 DSCB のみを戻すようにするには、このビットを 1 に設定します。

FCL2FLAG CVAF フィルターは、このバイトで以下の状況条件を示します。

FCL2SEQ 順次 VTOC アクセス・パスが最も効率的である場合は、このビットを 1 に設定します。CVAF フィルターが直接 VTOC アクセス・パスを選択する場合は、このフィールドを 0 に設定します。

FCL2SDIR 順次 VTOC アクセス・パス内のストレージ制限のために直接 DSCB 読み取りが必要な場合は、このビットを 1 に設定します。各 ACCESS=READ および ACCESS=RESUME 要求時に、CVAF はこのビットを 0 に初期化します。CVAF フィルターが制御を戻すときにこのビットをテストすることにより、ユーザーがストレージ制限を変更する必要があるかどうかを示すことができます。

FCL 項目のフォーマットを表 20 に示します。

表 20. フィルター基準リスト項目のフォーマット

オフセット	バイト	名前	説明
0 (X'00')	8	FCLDSN	データ・セット名情報項目。
	1	FCLDSNST	データ・セット名状況。
	X'00'		データ・セット名はまだ処理されていません。
	X'01'		DSCB が正常に戻されました。
	X'02'		データ・セット名が見つかりません。
	X'03'		DSCB チェーン内のエラー。RESUME 機能が推奨されます。
1 (X'01')	X'04'		CVAFFILT 処理中のエラー。RESUME は推奨されません。
	X'05'		ユーザー・バッファー・リスト・エレメントが不十分です。RESUME 機能が推奨されます。
	1	FCLDSNLG	データ・セット名の長さ。
2 (X'02')	1	FCL3FLAG	フラグ・バイト。
	1	FCL3UPDT	このデータ・セット名はこの呼び出し中に処理されました。
3 (X'03')	. xxx xxxx		予約済み。
	1	FCLDSNRV	予約済み。
4 (X'04')	4	FCLDSNA	データ・セット名アドレス。

FCLDSN データ・セット名情報を含みます。FCL 内では、このフィールドと後続のフィールドをセットとして、FCLCOUNT の値によって示された回数だけ繰り返されます。

FCLDSNST DSCB 検索状況を示します。

- ACCESS=READ 要求の場合、CVAF フィルターはこのバイトを 0 に初期化します。
- ACCESS=READ または ACCESS=RESUME の場合、データ・セット名の処理後に、CVAF フィルターはこのバイトを更新します。
- ACCESS=RESUME 要求は、FCLDSNST フィールドがゼロでないデータ・セット名は処理しません。そのため、このフィールドを変更すると、結果が予測不能になることがあります。
- 部分修飾データ・セット名の要求の場合、すべての修飾データ・セットのすべての DSCB チェーンを戻すまでは、CVAF フィルターは FCLDSNST フィールドに記入しません。CVAF フィルターは、その処理中に適用した最高の数値を記入します。
- 完全修飾データ・セット名の要求の場合、CVAF フィルターは、それぞれのデータ・セット名ごとに FCLDSNST バイトを戻します。値が 1 より大きい場合、CVAF フィルターは関連データ・セット名の DSCB を戻しませんでした。

このフィールドの値についての説明は、58 ページの表 20 を参照してください。

FCLDSNLG データ・セット名の長さを示します。この値は必要です。

FCL3FLAG FCLDSNA によって指示されたデータ・セット名に関連した状況フラグ・バイト。

FCL3UPDT

このビットは、CVAFFILT の現行呼び出し中に CVAF フィルターが関連データ・セット名を処理したことを示します。

- READ または RESUME 要求のために初期化する際に、CVAF フィルターはこのビットを 0 に設定します。
- CVAF フィルターは、関連データ・セット名の処理を完了すると、このビットを 1 に設定します。

FCL3DSNRV 予約済み、未使用。

FCLDSNA 完全修飾データ・セット名のアドレスを指定するか、またはこれが唯一のデータ・セット名であり、FCL1LIST が 0 の場合は、部分修飾データ・セッ

ト名のアドレスを指定します。このアドレスとそれが指すストレージ域の両方を提供する必要があります。

CVAFFILT マクロ・シーケンスの例

以下の例は、次のタスクを完了させるために CVAFFILT マクロ呼び出しを発行する順序を示します。

- 一連のデータ・セットの DSCB を要求する。
- ユーザーのバッファが不十分であったために中断された CVAFFILT 処理を再開する。
- 保持していたフィルター保管域を解放する。

この例では、以下の条件を想定しています。

- ユーザーは許可された呼び出し側である (つまり、UCB アドレスと IOAREA=KEEP を指定している)。
- CVAF バッファ・リストを以下の特性を持つように初期化した。
 - 4 つのバッファ。
 - プログラム内のバッファ・リスト・アドレスはラベル BUFADDR を持っている。
 - ACCESS=READ と ACCESS=RESUME 処理に同じバッファ・リストを使用する。
- フィルター基準リストを次のように初期化した。
 - FCLCOUNT = 6 (6 つのデータ・セット名の DSCB チェーンを要求しています。)
 - FCL1LIST = '1'B (データ・セット名は完全修飾名です。)
 - FCL1ORDR = '1'B (FCL 内のデータ・セット名エレメントによって暗黙指定された順序で DSCB チェーンを戻す必要があります。)
 - 6 つのデータ・セット名エレメントは、SYS1.A、SYS2.B、SYS3.C、SYS4.D、SYS5.E、および SYS6.F を要求するリストを形成するように初期化されている。
- 最初の 5 つのデータ・セットは、ボリューム上の DSCB チェーンの長さ (それぞれ、1、5、2、3、1) を持っている。
- 6 番目のデータ・セット (SYS6.F) は、ボリューム上に定義されていない。

初期化された CVPL を入手するには、次の CVAFFILT マクロを発行できます (リスト形式-CVAF を呼び出しません)。この例は、CVAF への分岐項目を要求し、呼び出し側は監視プログラム状態であることを指定します。

```
CVPLIST  CVAFFILT BRANCH=(YES,SUP),MF=L
```

DSCB チェーンの最初のセットを入手するには、次の CVAFFILT マクロを発行できます (実行形式-CVAF を呼び出します)。この例では、ACCESS=RESUME 呼び出しを可能にするために、フィルター保管域を保持することを指定します。IOAREA は、効率を良くするために保持する必要があります。

```
CVAFFILT ACCESS=READ,BUFLIST=bufaddr,FCL=fcladdr,
          UCB=ucbaddr,FLTAREA=KEEP,IOAREA=KEEP,
          MF=(E,CVPLIST)
```

この CVAFFILT 呼び出しは、次の DSCB を戻します。

Buffer	Contents of Buffer
1	Format-1 DSCB, SYS1.A
2	Format-1 DSCB, SYS3.C
3	Format-3 DSCB, SYS3.C
4	Undefined (unused)

CVAF フィルターは、戻りコード = 4、CVSTAT = X'40' (RESUME 推奨)、および FCLDSCBR = 3 を生成します。(CVAF は 2 つのデータ・セットの合計 3 つの DSCB を戻します。) CVAF は、データ・セット SYS2.B の DSCB は戻しません。そのチェーンには、提供されたバッファの総数を超える DSCB が含まれているからです。SYS2.B の DSCB を取得するには、少なくとも 5 つのバッファを指定し、別の ACCESS=READ を実行する必要があります。(CVAF は、それぞれの READ または RESUME ごとに異なるバッファ・リストを指定したり、READ と RESUME 呼び出しの間に既存のリストを変更したりすることを許可しますが、FCL を変更すると予測不能の結果になります。) SYS4.D の DSCB チェーン・サイズは、残りのバッファの数より大きいため、バッファ項目 4 には DSCB が戻されていません。FCL 状況情報は、次のようになります。

DSN	FCLDSNST	FCL3UPDT	Comments
SYS1.A	1	1	DSCBs returned from this call
SYS2.B	5	1	DSCB chain exceeds total buffers
SYS3.C	1	1	DSCBs returned from this call
SYS4.D	0	0	DSCBs can be returned by RESUME
SYS5.E	0	0	DSCBs can be returned by RESUME
SYS6.F	0	0	DSCBs can be returned by RESUME

この CVAFFILT 呼び出しは RESUME を推奨しており、ユーザーは FLTAREA=KEEP を指定したので、次のような実行形式の CVAFFILT を使用して、追加の DSCB チェーンを取得できます。

```
CVAFFILT ACCESS=RESUME,MF=(E,CVPLIST)
```

この CVAFFILT 呼び出しは、次のような DSCB を戻します。

Buffer	Contents of Buffer
1	Format-1 DSCB, SYS4.D
2	Format-2 DSCB, SYS4.D
3	Format-3 DSCB, SYS4.D
4	Format-1 DSCB, SYS5.E

CVAF フィルターは、戻りコード = 0、CVSTAT = 0 (要求完了) を生成し、FCL 状況を次のように更新します。

DSN	FCLDSNST	FCL3UPDT	Comments
SYS1.A	1	0	DSCBs returned from prior call
SYS2.B	5	0	DSCB chain exceeds total buffers
SYS3.C	1	0	DSCBs returned from prior call
SYS4.D	1	1	DSCBs returned from this call
SYS5.E	1	1	DSCBs returned from this call
SYS6.F	2	1	Data set name not found

FCLDSCBR には 4 が入ります。(この CVAFFILT 呼び出しは、合計 4 つの DSCB を戻します。) CVAF フィルターは SYS6.F の DSCB は戻しません。format-1 DSCB をボリューム上 (FCLDSNST = '2') で見つけることができなかつたからです。

この状況は CVAF フィルターが要求された DSCB をすべて戻したことを示しており、ユーザーは前の呼び出しで FLTAREA=KEEP および IOAREA=KEEP を要求したので、次のように RLSE 機能を要求します。

```
CVAFFILT ACCESS=RLSE,FLTAREA=NOKEEP,IOAREA=NOKEEP,  
MF=(E,CVPLIST)
```

VTOC エラー応答

VTOC 内でエラーが発生した場合、以下のアクションが取られます。

- 索引構造エラーが検出され、VTOC 上のアドレス・スペースがエンキューされている場合は、DADSM または CVAF が VTOC 索引を使用不可にします。
format-4 DSCB 内の索引付き VTOC ビットはゼロにリセットされ、ソフトウェア・エラー・レコードが SYS1.LOGREC に書き込まれ、DADSM 作成または拡張のための次の呼び出し時にシステム・ダンプが取られます。次の DADSM 作成または拡張呼び出し時に、VTOC は索引なしフォーマットに変換されます。
- VTOC アクセス・マクロの使用中にプログラム・チェック、マシン・チェック、または他のエラーが発生した場合、SYS1.LOGREC レコードが書き込まれ、システム・ダンプが取られます。
- エラー・コードが CVPL の CVSTAT フィールドに入ります。これらのエラー・コードの値と説明は、425 ページの『VTOC 索引エラー・メッセージと関連コード』にリストされています。

システム・エラーまたはユーザー・エラーからのリカバリー

無許可ユーザーは VTOC を変更できないため、無許可ユーザーが原因でのエラーの場合は、VTOC も VTOC 索引もリカバリーの必要はありません。

システム・エラーは、更新中の DADSM を中断して、VTOC または VTOC 索引 (あるいは、その両方) を部分的に更新された状態のままにするため、VTOC と VTOC 索引に影響を与えることがあります。VTOC と VTOC 索引は両方とも、DADSM がそのような中断からリカバリーできます。

索引なし VTOC (または、使用不可にされた索引を持つ VTOC) の場合、DADSM 作成または拡張のための後続の呼び出し時に、VTOC 変換ルーチンがフリー・スペース DSCB チェーンを再確立します。

索引付き VTOC の場合、任意の DADSM 機能への後続の呼び出し時に、直前の中断がリカバリーされます (中断された機能を取り消すか、完了させることによって)。

GTF トレース

すべての VTOC 索引出力 I/O、すべての VTOC 出力 I/O、およびすべての VTOC 索引およびスペース・マップ変更のためのすべての CVAF 呼び出しをトレースするために、トレース機能が存在します。この機能の詳細については、「z/OS DFSMSdfp 診断解説書」を参照してください。

VTOC と VTOC 索引のリスト表示

IEHLIST ユーティリティー・プログラムの LISTVTOC コマンドを使用すると、VTOC および VTOC 索引のダンプ (定様式) または要約リストを入手できます。DFSMSdss プリント・コマンドも VTOC および索引のリスト表示オプションを備えています。ISMF データ・セット・アプリケーションは、VTOC に表されたデータ・セットに関する情報を表示します。

第 2 章 ポリリューム目録の管理

この章では、ポリリューム目録 (VTOC) を作成し、管理するために使用するプログラムと手順について説明します。

VTOC および VTOC 索引の作成

アクティビティーのためにポリリュームを初期化して準備するには、装置サポート機能 (ICKDSF) ユーティリティーを使用して VTOC を作成します。ICKDSF INIT コマンドを使用して INDEX キーワードを指定すると、ポリリュームを初期化するとき VTOC 索引を作成できます。

索引なし VTOC から索引付き VTOC への変換は、BUILDIX コマンドを使用して IXVTOC キーワードを指定して行います。その逆の操作は、BUILDIX コマンドを使用して OSVTOC キーワードを指定することによって行えます。詳細については、「*Device Support Facilities User's Guide and Reference*」を参照してください。

VTOC および VTOC 索引の保護

VTOC および VTOC 索引を保護するために、以下の方法を使用します。

RACF[®]

リソース・アクセス管理機能 (RACF) (Security Server for z/OS のコンポーネント) を使用して、RACF DASDVOL クラスのもとにポリリューム通し番号エンティティーを定義することにより、VTOC および VTOC 索引を保護できます。保護された VTOC および VTOC 索引を変更するには、プログラムは以下のレベルの許可が必要です。

- 出力処理のために VTOC、または SYS1.VTOCIX で始まる名前のデータ・セットをオープンする場合は、UPDATE レベル。
- SYS1.VTOCIX で始まる名前のデータ・セットの割り振り、名前変更、またはスクラッチを行う場合、あるいはデータ・セットを SYS1.VTOCIX で始まる名前に変更する場合は、ALTER レベル。

VTOC と VTOC 索引はどちらも、DASDVOL/通し番号エンティティーによって入力処理のためにオープンされることから保護されません。また VTOC と VTOC 索引はどちらも、RACF DATASET クラスを通して保護することはできません。RACF の使用についての追加情報は、「*z/OS Security Server RACF ユーザーズ・ガイド*」を参照してください。

APF

許可プログラム機能 (APF) は、以下の操作を行うための許可をプログラムに与える必要があります。

- 出力処理のための VTOC のオープン

- 名前が SYS1.VTOCIX で始まるデータ・セットの出力処理、割り振り、名前変更、またはスクラッチのためのオープン
- SYS1.VTOCIX で始まる名前へのデータ・セットの名前変更

APF の使用についての追加情報は、「*z/OS MVS Programming: Authorized Assembler Services Guide*」を参照してください。

パスワード保護

VTOC 索引付きデータ・セットは、パスワードで保護することができます。保護は、パワー保護データ・セットの場合と同じです。VTOC 索引を含むボリュームが RACF によって DASDVOL クラスを使用して保護されている場合、パスワード検査はバイパスされます。追加情報については、165 ページの『第 6 章 パスワード保護データ・セットの使用』を参照してください。

VTOC のコピー/復元/初期化

以下では、ボリュームの更新時の VTOC に関する考慮事項について説明します。

索引なし VTOC を含むボリューム

索引なし VTOC を含んでいるボリュームを更新する場合、以下の考慮事項に留意してください。

ダンプ・テープからのボリュームの復元: ボリューム通し番号を変更する場合、または VTOC を変更しない部分復元を行う場合は、操作上の要件はありません。復元を行って、ボリューム通し番号を変更せずに VTOC サイズを変更する場合は、復元した後でボリュームをオフラインに変更する必要があります。索引付き VTOC を含むボリュームでは復元を試みないでください。

ボリュームのコピー: ボリューム通し番号を変更する場合、または受け取り側ボリュームの VTOC を変更しない場合は、操作上の要件はありません。コピーを行って、ボリューム通し番号を変更せずに VTOC サイズを変更する場合は、コピーした後でボリュームをオフラインに変更する必要があります。索引付き VTOC を含むボリュームからのコピーは試みないでください。

共用 DASD に関する考慮事項: 共用 DASD 環境では、VTOC 索引を再配置する場合、またはボリュームを索引付き VTOC から OS VTOC に、または OS VTOC から索引付き VTOC に変更する場合、操作を開始する前に、共用システムに対して装置をオフラインに変更する必要があります。

索引付き VTOC を含むボリューム

ボリュームを更新するために VTOC を索引なしフォーマットに変換する場合、装置サポート機能 (ICKDSF) を使用してください。これを利用しない場合は、以下の考慮事項に留意してください。

ボリュームの初期化: ボリューム通し番号を変更しない場合は、ジョブを開始する前にボリュームをオフラインに変更する必要があります。

ダンプ・テープからのボリュームの復元: ボリューム通し番号を変更する場合、あるいは VTOC または VTOC 索引を変更しない部分復元を行う場合は、操作上の要件はありません。復元を行って、ボリューム通し番号を変更せずに VTOC または VTOC 索引を変更する場合は、復元した後でボリュームをオフラインに変更する必要があります。

ボリュームのコピー: 受け取り側ボリュームのボリューム通し番号を変更する場合、あるいは VTOC または VTOC 索引を変更せずに部分ダンプを行う場合は、操作上の要件はありません。ボリューム通し番号を変更せずに VTOC または VTOC 索引を変更する場合は、コピーした後で受け取り側ボリュームをオフラインに変更する必要があります。

VTOC からのデータ・セットの削除

SCRATCH および CAMLST マクロを使用して、非 VSAM データ・セットまたは一時 VSAM データ・セットを削除できます。SCRATCH 処理は、データ・セットが占めていたスペースを再割り振り用に使用可能にします。この処理は、ディスクからデータを自動的に消去しません。詳しくは、『機密データの消去』を参照してください。

影響を受けるボリュームの指定

データ・セットを削除するときは、仮想ストレージ内にボリューム・リストを作成します。ボリューム・リストは、データ・セットが常駐する各ボリュームの項目から構成されます。SMS 管理データ・セットを削除する場合は、リスト内に少なくとも 1 つの SMS 管理ボリュームを指定します。リストの最初の 2 バイトは、リスト内の項目の数を示します。各 12 バイト項目は、4 バイトの装置コード (ボリュームの UCB からの UCBTYP フィールド)、6 バイトのボリューム通し番号、および 2 バイトのスクラッチ状況情報で構成されます。スクラッチ状況情報は、2 次状況コードと状況コードからなり、両方ともゼロに初期化されていなければなりません。

ボリュームは、ボリューム・リスト内の順序に従って処理されます。ボリュームがマウントされていない場合、ボリュームをマウントするように要求するメッセージがオペレーターに出されます。メッセージが出されるのは、ユーザーが直接アクセス装置を指示した場合に限られ、レジスター 0 にその装置の UCB のアドレスをロードして、アンマウントされたボリュームをマウントする必要があります。(装置は、ジョブに対して割り振り済みでなければなりません。) **レジスター 0 に UCB アドレスをロードしていない場合、その内容はゼロでなければならない**、また SCRATCH マクロ命令が発行される前に、ボリューム・リスト内のボリュームの少なくとも 1 つがマウントされている必要があります。このマクロでは、レジスター 0 には UCB (UCB のコピーではなく) のアドレスを使用してください。31 ビット・プログラムは、RENAME で clean UCB addr を渡す必要があります (該当する場合)。

要求されたボリュームをマウントできない場合、オペレーター応答で、要求を満たせないことを示します。これにより、利用不可のボリュームのボリューム・リスト項目の最後のバイトに状況コードが設定され (スクラッチ状況コードの 2 番目のバイト)、ボリューム・リスト内の次のボリュームが処理されます。

機密データの消去

機密データを含むデータ・セットは、そのスペースを使用可能にする前に、ゼロで上書きして消去する必要があります。これは SCRATCH マクロを発行する前に、またはスクラッチ処理内で要求して、次のいずれかを実行することにより達成できます。

- 関連 RACF プロファイルの ERASE 属性を提供する。
- SCRATCH パラメーター・リストのビット 21 (X'00 00 04 00') をアクティブにする。71 ページの表 21 を参照してください。

許可を受けている SCRATCH の呼び出し側は、ビット 22 を 1 に設定することにより (これは RACF プロファイルの ERASE 属性をオーバーライドします)、データの消去を防止できます。

システム管理ストレージに関する考慮事項

SMS は、データ・セットの SCRATCH 要求をすべて審査します。ボリューム・リスト内のボリュームが SMS 管理対象である場合、SMS はカタログの LOCATE を行って、実際のボリューム通し番号を判別し、常駐しているすべてのボリュームからデータ・セットを削除します。SMS は、VTOC、VTOC 索引、およびカタログに対して必要な変更を調整します。

SMS がアクティブで、リスト内のすべてのボリュームが SMS 管理対象である場合、DADSM が処理エラーを検出すると、SMS は障害が発生したボリュームを判別します。リスト内の最初の項目は、要求が失敗したボリュームの項目でオーバーレイされます。

エラーがあると示されたボリュームは、プログラムが提供したボリューム・リストには指定されていない場合があります。この状態が起きるのは、リスト内のボリュームがデータ・セットのカタログ項目内のボリュームと異なっている場合です。

SMS がアクティブでない場合、SMS 管理データ・セットは削除できません。

SMS 管理 VSAM データ・セットは、アクセス方式サービス・プログラムの DELETE コマンドを使用して削除できます。詳しくは、「z/OS DFSMS カatalogのためのアクセス方式サービス・プログラム」を参照してください。

一般的な考慮事項および制約事項

format-1 DSCB 内の有効期限が過ぎていない場合、その有効期限をオーバーライドしない限り、データ・セットを削除することはできません。CAMLST マクロ命令で OVRD オプションを指定することにより、有効期限を無視して SCRATCH を要求できます。SCRATCH 処理は、3 つの never-scratch (スクラッチされない) 日付をサポートします。データ・セットがスクラッチされるのを防止するには、次の有効期限のいずれかを指定します。

1999.365
1999.366
1999.999

仮想入出力 (VIO) データ・セットを削除するには、ジョブ・ステップで使用するデータ・セットが割り振り済みでなければなりません。

SYSIN または SYSOUT データ・セットあるいは HFS ファイルに対しては、SCRATCH マクロ命令を使用できません。HFS ファイルに対して SCRATCH を使用すると、予測不能の結果を受け取ります。

RACF 保護されていない、パスワード保護データ・セットを削除しようとすると、オペレーティング・システムは、コンソールのオペレーターまたは TSO コンソール

ルの端末オペレーターにメッセージ IEC301A を出して、パスワードの入力を求めます。提供されたパスワードが WRITE 保護モード標識に関連付けられている場合、データ・セットはスクラッチされます。保護モード標識については、165 ページの『第 6 章 パスワード保護データ・セットの使用』で説明します。

データ・セットが RACF に定義されている場合 (その format-1 DSCB に示されるか、RACF プロファイルによって記述)、あるいはそれが常駐するボリュームが RACF に定義されている場合、DATASET クラスのデータ・セット/ボリューム通し番号または DASDVOL クラスのボリューム通し番号に対する ALTER アクセス権限を持っている場合にのみ、データ・セットをスクラッチできます。

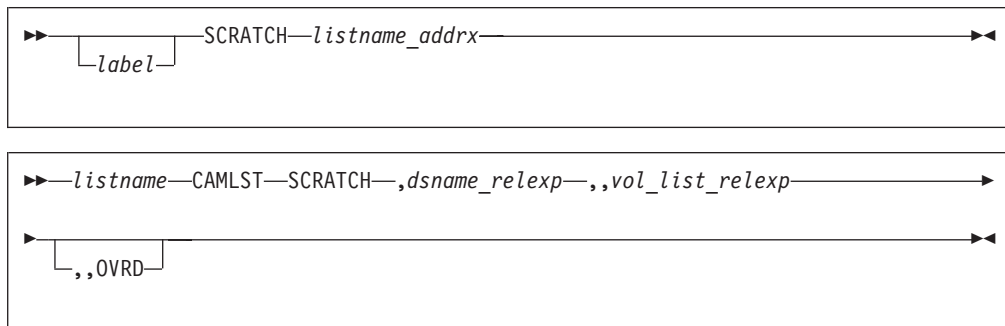
要件: SMS 管理の非 VSAM データ・セットの場合、それを削除するにはデータ・セットまたはカタログに対する RACF 権限が必要です。

非 SMS 管理の非 VSAM データ・セットの場合、DADSM は RACF を呼び出して許可を検査します。ユーザーが DATASET クラスのデータ・セット/ボリューム通し番号に対する ALTER アクセス権限を持っている場合、DADSM はボリュームからデータ・セットを削除します。ユーザーが DATASET クラスのデータ・セット/ボリューム通し番号に対する ALTER 権限、または DATASET クラスのカタログ/ボリューム通し番号に対する UPDATE アクセス権限を持っている場合、ユーザーはカタログ項目を削除できます。

PDS または PDSE のメンバーの削除や名前変更を行うには、STOW マクロを使用します。STOW については、「z/OS DFSMS Macro Instructions for Data Sets」および「z/OS DFSMS データ・セットの使用法」で説明しています。IEHPROGM ユーティリティを使用してメンバーを削除することもできます (「z/OS DFSMSdfp ユーティリティ」を参照してください)。

SCRATCH および CAMLST マクロ仕様

SCRATCH および CAMLST マクロのフォーマットは、次のとおりです。



listname_addrx CAMLST マクロ命令によってセットアップされたパラメーター・リスト (ラベル付きリスト名) を指します。

SCRATCH このオペランドは表示どおりにコーディングしてください。

dsname_relexp 完全修飾データ・セット名の仮想ストレージ位置を指定します。名前を含んでいる領域は 44 バイトの長さでなければなりません。

vol_list_relexp ポリウム・リストを含んでいる領域の仮想ストレージ位置を指定します。領域はハーフワード境界で開始しなければなりません。

OVRD 表示どおりにコーディングされている場合、DSCB 内の有効期限を無視する必要があることを指定します。

例

次の例では、データ・セット A.B.C が 2 つのポリウムから削除されます。ID (format-1) DSCB 内の有効期限は無視されます。

```

          SR      0,0          SHOW NO UCB IS SUPPLIED
          SCRATCH DELABC      DELETE DATA SET A.B.C
*
*
*
          DELABC  CAMLST  SCRATCH,DSABC,,VOLIST,,OVRD
          DSABC   DC      CL44'A.B.C'    DATA SET NAME
          VOLIST  DC      H'2'          NUMBER OF VOLUMES
          DC      X'3030200E'          3380 DISK DEVICE CODE
          DC      CL6'000017'          VOLUME SERIAL NO.
          DC      H'0'                SCRATCH STATUS CODE
          DC      X'3030200E'          3380 DISK DEVICE CODE
          DC      CL6'000018'          VOLUME SERIAL NO.
          DC      H'0'                SCRATCH STATUS CODE
    
```

推奨: 戻りコードおよび SCRATCH 状況コードを確認してください。

SCRATCH マクロ命令は CAMLST マクロ命令を指します。SCRATCH オペランドは、データ・セットを削除することを指定します。DSABC は、削除するデータ・セットの完全修飾名を含んでいる 44 バイト域の仮想ストレージ位置を指定します。VOLIST は、ユーザーが作成したポリウム・リストの仮想ストレージ位置を指定します。OVRD は、削除するデータ・セットの DSCB 内の有効期限を無視する必要があることを指定します。

SCRATCH パラメーター・リスト

CAMLST マクロは SCRATCH パラメーター・リストを生成しますが、ユーザーのコードでは CAMLST がサポートしないオプションをいくつか設定できます。表 21 を参照してください。

表 21. SCRATCH パラメーター・リスト

オフセット	長さ、または ビット・パターン	説明
0(0)	1	フラグ。常に X'41' です。
1(1)	1	フラグ。
	.1.	リソース・アクセス管理機能 (RACF) プロファイルを削除しないでください。JSCBPASS がオンの場合にのみ有効です。プログラムのプロパティ・テーブルでセキュリティをバイパスする権限が与えられている場合、JSCBPASS はオンです。SYS1.PARMLIB の SCHEDxx メンバーの 1 次 POI タスク (PPT) ステートメントの NOPASS オプションを参照してください (「 <i>/OS MVS 初期設定およびチューニング 解説書</i> 」で説明しています)。
	x.xx xxxx	予約済み。
2(2)	1	フラグ。
	1...	SYSZTIOT はすでにエンキューされています。このビットがオンで、呼び出し側が APF 許可されているか、またはシステム・キーまたは監視プログラム状態で実行されている場合、SCRATCH は SYSZTIOT のエンキューをバイパスします。SYSZTIOT がすでにエンキューされていない場合、システムに損傷を与える可能性があります。
	.01.	SCRATCH に対して常に設定されます。
	...1	CAMLST で OVRD がコーディングされています。
 1...	RACF プロファイル検査をバイパスします。このビットがオンで、呼び出し側が APF 許可されているか、またはシステム・キーまたは監視プログラム状態で実行されている場合、SCRATCH は RACF プロファイル検査をバイパスします。
1..	データ・セットの割り振りスペースをすべて消去します。
1.	割り振りスペースを消去しません。このビットがオンで、呼び出し側が APF 許可されているか、またはシステム・キーまたは監視プログラム状態で実行されている場合、SCRATCH はスペース消去をバイパスします。
3(3)	1	予約済み。
4(4)	4	44 バイトのデータ・セット名のアドレス。
8(8)	4	予約済み。
12(C)	4	ボリューム・リストのアドレス。

SCRATCH からの戻りコード

制御は、SCRATCH マクロによって生成された命令の後続の命令に戻ります。レジスタ 15 に、表 22 に示すような SCRATCH 戻りコードが入ります。

SCRATCH は、4 バイトの診断情報をレジスタ 0 に戻します。エラーが発生した場合、DADSM は、障害関連の情報からなるメッセージ IEC614I を発行します。これには、戻りコードと 4 バイトの診断情報が含まれています。この情報についての説明は、「z/OS DFSMSdfp 診断解説書」を参照してください。

ボリューム・リスト項目の最後の 2 バイトには、2 次状況コードとスクラッチ状況コードが入ります。2 次状況コードを 73 ページの表 24 に、スクラッチ状況コードを 73 ページの表 23 に示します。データ・セットが各ボリュームから削除されたかどうかを判別するには、スクラッチ状況コードを検査します。(ボリューム・リストの最初の項目の場合、スクラッチ状況コードがゼロでも、2 次状況コードはゼロでないことがあります。)

表 22 は、SCRATCH 戻りコードによって示される状態について説明しています。

表 22. SCRATCH 戻りコード

戻りコード	説明
000 (X'00')	レジスタ 0 に戻された 4 バイトの診断情報がすべてゼロの場合、データ・セットが正常に削除されたことを示しています。ゼロでない場合は、「z/OS DFSMSdfp 診断解説書」内の SCRATCH 診断情報テーブルを参照して、障害関連の状態を判別してください。
004 (X'04')	データ・セットのどの部分を含んでいるボリュームもマウントされていませんでした。データ・セットは、ジョブ・ステップに割り振られていない VIO データ・セットである可能性があります。
008 (X'08')	1 つ以上のボリュームで異常な状態が検出されました。
012 (X'0C')	以下の状態のいずれかが発生しました。 <ul style="list-style-type: none"> • SCRATCH パラメーター・リストが無効である。 • ボリューム・リストが無効である。 • SCRATCH への入り口で、レジスタ 0 がゼロでなく、有効な UCB を指していなかった。アドレスは、UCB (UCB のコピーではなく) のアドレスでなければなりません。

SCRATCH 状況コードは設定されていません。

SCRATCH からの状況コード

SCRATCH マクロ命令が実行された後 (SCRATCH 戻りコード 0、4、および 8 の場合のみ)、ボリューム・リスト内の各 12 バイト項目の最後のバイトは、以下の状態のいずれかを示します。

表 23. SCRATCH 状況コード

状況コード	意味
0 (X'00')	データ・セットはこのボリュームから削除されました。
1 (X'01')	このボリュームの VTOC は、削除する format-1 DSCB を含んでいません。
2 (X'02')	以下の状態のいずれかが発生しました。 <ul style="list-style-type: none"> 許可された 2 回の試行で正しいパスワードが指定されないため、データ・セットをスクラッチできなかった。 ユーザーは VSAM データ・スペースまたは統合カタログ機能 VSAM データ・セットをスクラッチしようとした。 ユーザーは VTOC 索引付きデータ・セットをスクラッチしようとした。 SMS 妥当性検査で障害が発生した。 最後の参照日の検査が失敗した。
3 (X'03')	OVRD オプションが指定されなかったか、または保存期間の有効期限が切れていなかったため、データ・セットは削除されませんでした。
4 (X'04')	以下の状態のいずれかが発生しました。 <ul style="list-style-type: none"> このボリュームの処理中に無効な format-1 DSCB が検出された。 予期しない CVAE エラー戻りコードが戻された。 インストール・システム出口が要求をリジェクトした。 データ・セットによって占められていた DASD トラックを消去しているときに入出力エラーが発生した。スクラッチ・パラメーター・リストで ERASE オプションが指定されたか、または RACF 定義データ・セットに対して ERASE 属性が指定されたかのいずれかです。
5 (X'05')	このボリュームがマウントされていることも、ボリュームをマウントするのに使用可能な装置があることも確認できませんでした。
6 (X'06')	オペレーターはこのボリュームをマウントできませんでした。
7 (X'07')	データ・セットがオープンされていたので削除できませんでした。
8 (X'08')	format-1 DSCB はデータ・セットが RACF に定義されていることを示していますが、ユーザーがデータ・セットまたはボリュームに対して許可されていないか、またはデータ・セットが VSAM データ・スペースであるかのいずれかです。

SCRATCH マクロ命令が実行された後、ボリューム・リストの最初の項目の最後から 2 番目のバイトは、以下の状態のいずれかを示します。

表 24. 2 次状況コード

状況コード	意味
0 (X'00')	このボリュームには、2 次状況はありません。

表 24. 2 次状況コード (続き)

状況コード	意味
128 (X'80')	データ・セットは RACF で保護され、呼び出し側プログラムは RACF DATASET クラスによってデータ・セットのスクラッチが許可されていました。これは、少なくとも 1 つのボリューム項目が保護されていたことを意味します。

VTOC 内のデータ・セットの名前変更

RENAME および CAMLST マクロを使用して、非 VSAM データ・セットの名前を変更できます。名前変更を処理すると、すべての format-1 DSCB 内のデータ・セット名が、ユーザーの提供する新しい名前置き換えられます。新しいデータ・セット名は、標準データ・セット命名規則に準拠している必要があります。

影響を受けるボリュームの指定

データ・セットを名前変更するときは、仮想ストレージ内にボリューム・リストを作成します。ボリューム・リストは、データ・セットが常駐する各ボリュームの項目から構成されます。SMS 管理データ・セットを名前変更する場合は、リスト内に少なくとも 1 つの SMS 管理ボリュームを指定します。リストの最初の 2 バイトは、リスト内の項目の数を示します。各 12 バイト項目は、4 バイトの装置コード (ボリュームの UCB からの UCBTYP フィールド)、6 バイトのボリューム通し番号、および 2 バイトの名前変更状況コード (これはゼロに初期化されていなければなりません) から構成されます。

ボリュームは、ボリューム・リスト内の順序に従って処理されます。ボリュームがマウントされていない場合、ボリュームをマウントするように要求するメッセージがオペレーターに出されます。メッセージが出されるのは、ユーザーが直接アクセス装置を指示した場合に限られ、レジスター 0 にその装置の UCB のアドレスをロードして、アンマウントされたボリュームをマウントする必要があります。(装置は、ジョブに対して割り振り済みでなければなりません。) **レジスター 0 に UCB アドレスをロードしていない場合、その内容はゼロでなければならない**、また RENAME マクロ命令が発行される前に、ボリューム・リスト内のボリュームの少なくとも 1 つがマウントされている必要があります。このマクロでは、レジスター 0 には UCB (UCB のコピーではなく) のアドレスを使用してください。31 ビット・プログラムは、RENAME で clean UCB addr を渡す必要があります (該当する場合)。

要求されたボリュームをマウントできない場合、オペレーター応答で、要求を満たせないことを示します。これにより、利用不可のボリュームのボリューム・リスト項目の最後のバイトに状況コードが設定され (名前変更状況コードの 2 番目のバイト)、ボリューム・リスト内の次のボリュームが処理されます。

システム管理ストレージに関する考慮事項

SMS は、データ・セットの RENAME 要求をすべて審査します。ボリューム・リストで指定されたボリュームが SMS 管理対象である場合、SMS はカタログの LOCATE を行って、実際のボリューム通し番号を判別し、VTOC、VTOC 索引、およびカタログに対して必要な変更を調整します。

SMS がアクティブで、リスト内のすべてのボリュームが SMS 管理対象である場合、DADSM が処理エラーを検出すると、SMS は障害が発生したボリュームを判別します。リスト内の最初の項目は、要求が失敗したボリュームの項目でオーバーレイされます。

エラーがあると示されたボリュームは、プログラムが提供したボリューム・リストには指定されていない場合があります。この状態が起きるのは、リスト内のボリュームがデータ・セットのカタログ項目内のボリュームと異なっている場合です。

SMS がアクティブでない場合、SMS 管理データ・セットは名前変更できません。

一般的な考慮事項および制約事項

マルチボリュームに関する考慮事項

複数のボリューム上に保管されているデータ・セットの名前変更を行うには、すべてのボリュームがマウントされている必要があります。

名前変更できないデータ・セットと UNIX ファイル

SYSIN または SYSOUT データ・セット、あるいは UNIX ファイル (HFS ファイルなど) に対しては、RENAME マクロは使用できません。UNIX ファイルに対して RENAME を使用すると、予測不能の結果を受け取ります。

VIO データ・セットは、名前変更することができません。

データ・セットのセキュリティ

DATASET クラスのデータ・セットに対する ALTER アクセス権を持っている場合のみ、RACF 定義データ・セットの名前変更を行うことができます。

パスワード保護されているデータ・セットを名前変更しようとする、オペレーティング・システムはメッセージ IEC301A を出して、オペレーターまたは TSO オペレーターにパスワードを検査するように求めます。提供されたパスワードが WRITE 保護モード標識に関連付けられている場合、データ・セットは名前変更されます。保護モード標識については、165 ページの『第 6 章 パスワード保護データ・セットの使用』で説明します。

使用中の可能性のあるデータ・セットの名前変更

現行アドレス・スペースに割り振られているデータ・セットを名前変更することはできますが、それをオープンすることはできません。

一般に、同じシステム内の別のアドレス・スペースに割り振られているデータ・セット、または SYSDSN エンキューの有効範囲内にあるデータ・セットと同じ名前のデータ・セットは、名前変更することができません。以下の条件がすべて真の場合、システムはこの制限をバイパスします。

- プログラムが CAMLST マクロ展開の特定のビットをオンに設定する。この命令は、OI listname+2,X'10' とコーディングできます。
- STGADMIN.DPDSRN.oddname という名前の RACF 機能クラスに対して少なくとも読み取り権限を持っている (ここで、oddname は、最大 23 文字の既存のデータ・セット名です)。STGADMIN.DPDSRN.SYS2.* のような総称クラス名を使

用できます。IBM では、STGADMIN.DPDSRN.* は範囲が広すぎるので、これに対する権限を与えることはお勧めできません。

- データ・セットが SMS 管理対象ではない。

代わりに、PDF のデータ・セット名前変更オプションを使用することもできます。非 SMS 管理の非 VSAM データ・セットの名前変更を試みたときに、そのデータ・セット名が使用中であり、ユーザーが適切な RACF 機能クラス権限を持っている場合、PDF はユーザーに処理を進めるかどうかを尋ねます。ユーザーにはそのデータ・セットが実際にオープンされていないことが分かるからです。名前変更されるデータ・セットがどのシステムでもオープンされていないことが分かっている場合にのみ、名前変更の処理を進めてください。

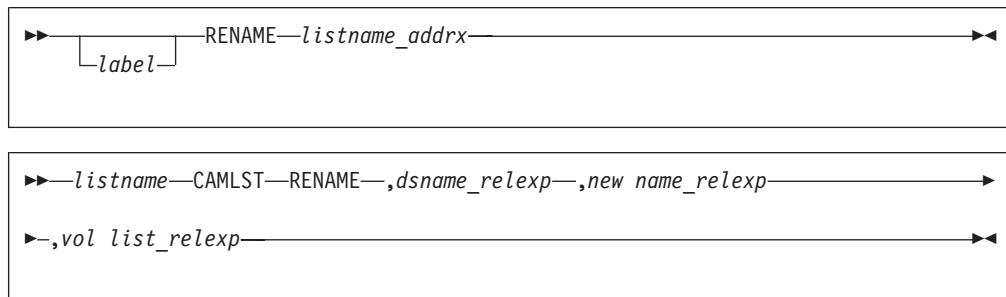
重要: このオプションは、十分に注意して使用する必要があります。

STGADMIN.DPDSRN.oldsname に対する RACF 権限を与えるのはごく少数に限る必要があります。データ・セットがどのシステムでもオープンされないことが分かっている限り、このオプションは使用しないでください。データ・セットが名前変更された後で、誰かが異なるアドレス・スペースでそれを削除する可能性があります。別の誰かがそれを古い名前でオープンしていた場合、新しいデータ・セットはディスク上のその場所に表示されます。これは、セキュリティ違反になりますが、システムはそれを検出しません。

データ・セット名前変更機能は、タイプ 18 SMF レコードを書き込んで、ストレージ管理者、システム・プログラマー、および監査員に情報を提供します。この重複名オーバーライド機能の使用により、レコードには名前変更が成功したかどうかを示す標識が含まれています。CAMLST マクロ展開でこのオプションを要求したがデータ・セット名が使用中でない場合には、SMF 指標はオンになりません。

RENAME および CAMLST マクロ仕様

RENAME および CAMLST マクロのフォーマットは、次のとおりです。



listname_addrx CAMLST マクロ命令によってセットアップされたパラメーター・リスト (ラベル付きリスト名) を指します。

RENAME このオペランドは表示どおりにコーディングしてください。

dsname_relexp 置換される完全修飾データ・セット名の仮想ストレージ位置を指定します。名前を含んでいる領域は 44 バイトの長さでなければなりません。

new name_relexp 新しい名前として使用される完全修飾データ・セット名の仮想ストレージ位置を指定します。名前を含んでいる領域は 44 バイトの長さでなければなりません。

vol list_relexp ボリューム・リストを含んでいる領域の仮想ストレージ位置を指定します。領域はハーフワード境界で開始しなければなりません。

例

次の例では、データ・セット A.B.C が D.E.F に名前変更されます。データ・セットは 2 つのボリューム上にあります。

```

SR      0,0          SET REG 0 TO ZERO
RENAME  DSABC       CHANGE DATA SET
                          NAME A.B.C TO D.E.F

DSABC   CAMLST     RENAME,OLDNAME,NEWNAME,VOLIST
OLDNAME DC        CL44'A.B.C'   OLD DATA SET NAME
NEWNAME DC        CL44'D.E.F'   NEW DATA SET NAME
VOLIST  DC        H'2'         TWO VOLUMES
DC      DC        X'3030200E'   3380 DISK DEVICE CODE
DC      DC        CL6'000017'   VOLUME SERIAL NO.
DC      DC        H'0'         RENAME STATUS CODE
DC      DC        X'3030200E'   3380 DISK DEVICE CODE
DC      DC        CL6'000018'   VOLUME SERIAL NO.
DC      DC        H'0'         RENAME STATUS CODE

```

推奨: 戻りコードおよび RENAME 状況コードを確認してください。

RENAME マクロ命令は CAMLST マクロ命令を指します。RENAME オペランドは、データ・セットを名前変更することを指定します。OLDNAME は、名前変更するデータ・セットの完全修飾名を入れた 44 バイト域の仮想ストレージ位置を指定

します。NEWNAME は、データ・セットの新しい名前を入れた 44 バイト域の仮想ストレージ位置を指定します。VOLIST は、ユーザーが作成したボリューム・リストの仮想ストレージ位置を指定します。

RENAME パラメーター・リスト

CAMLST マクロは RENAME パラメーター・リストを生成しますが、ユーザーのコードでは CAMLST がサポートしないオプションをいくつか設定できます。表 25 を参照してください。

表 25. CAMLST によって生成される RENAME パラメーター・リスト

オフセット	長さ、または ビット・パターン	説明
0(0)	1	フラグ。常に X'CI' です。
1(1)	1	フラグ。
	.1..	RACF プロファイルを定義しないでください。
	..1.	format-1 DSCB 内の変更ビット (DS1DSCHA) は更新しないでください。
	x..x xxxx	予約済み。
2(2)	1	フラグ。
	..1.	RENAME の場合、常に設定されます。
	...1	75 ページを参照。
	xx.. xxxx	予約済み。
3(3)	1	予約済み。
4(4)	4	44 バイトの既存のデータ・セット名のアドレス。
8(8)	4	44 バイトの新規データ・セット名のアドレス。
12(C)	4	ボリューム・リストのアドレス。

RENAME からの戻りコード

制御は、RENAME マクロによって生成された命令の後続の命令に戻ります。レジスター 15 に、79 ページの表 26 に示すような DADSM 戻りコードが入ります。

DADSM RENAME は、4 バイトの診断情報をレジスター 0 に戻します。エラーが発生した場合、DADSM は、障害関連の情報からなるメッセージ IEC614I を発行します。これには、戻りコードと 4 バイトの診断情報が含まれています。この情報についての説明は、「z/OS DFSMSdfp 診断解説書」を参照してください。

各ボリュームのボリューム・リスト項目には、79 ページの表 27 に示すような名前変更状況コードが入ります。各ボリューム上のデータ・セットが正常に名前変更されたかどうかを判別するには、ボリューム・リストの各項目の最後のバイトに含まれている名前変更状況コードを検査します。

79 ページの表 26 は、DADSM 戻りコードによって示される状態について説明しています。

表 26. DADSM RENAME 戻りコード

戻りコード	説明
000 (X'00')	レジスター 0 に戻された 4 バイトの診断情報がすべてゼロの場合、データ・セットは正常に名前変更されました。ゼロでない場合は、「z/OS DFSMSdfp 診断解説書」内の DADSM RENAME 診断情報テーブルを参照して、障害に関する状態を判別してください。
004 (X'04')	データ・セットのどの部分を含んでいるボリュームもマウントされていませんでした。データ・セットは VIO データ・セットでも構いませんが、名前変更することはできません。
008 (X'08')	1 つ以上のボリュームで異常な状態が検出されました。診断情報はレジスター 0 にあります。「z/OS DFSMSdfp 診断解説書」内の DADSM RENAME 診断情報テーブルを参照して、障害に関する状態を判別してください。
012 (X'0C')	以下の状態のいずれかが発生しました。 <ul style="list-style-type: none"> • DADSM RENAME パラメーター・リストが無効である。 • ボリューム・リストが無効である。 • RENAME への入り口で、レジスター 0 がゼロでなく、有効な UCB を指していなかった。アドレスは、UCB (UCB のコピーではなく) のアドレスでなければなりません。

RENAME 状況コードは設定されていません。

RENAME からの状況コード

RENAME マクロ命令が実行された後 (RENAME 戻りコード 0、4、および 8 の場合のみ)、ボリューム・リスト内の各 12 バイト項目の最後のバイトは、表 27 で説明している状態のいずれかを示します。

表 27. RENAME 状況コード

状況コード	意味
0 (X'00')	このボリューム上の VTOC で、データ・セットの format-1 DSCB が名前変更されました。
1 (X'01')	このボリュームの VTOC は、名前変更するデータ・セットの format-1 DSCB を含んでいません。
2 (X'02')	以下の状態のいずれかが発生しました。 <ul style="list-style-type: none"> • データ・セットはパスワード保護されており、許可された 2 回の試行で正しいパスワードが提供されないため、データ・セットを名前変更できなかった。 • VSAM データ・スペースまたは統合カタログ機能 VSAM データ・セットを名前変更しようとした。 • VTOC 索引付きデータ・セットを名前変更しようとした。 • SMS 妥当性検査で障害が発生した。
3 (X'03')	新規データ・セット名を含んでいる format-1 DSCB がすでにこのボリュームの VTOC に存在するか、またはデータ・セットを SYS1.VTOCIX で始まる名前に変更しようとした。

表 27. RENAME 状況コード (続き)

状況コード	意味
4 (X'04')	以下の状態のいずれかが発生しました。 <ul style="list-style-type: none">• このボリューム上のデータ・セットを名前変更しようとしているときに永続入出力エラーが発生した。• このボリュームの処理中に無効な format-1 DSCB が検出された。• 索引 VIER 内に新しい名前のために使用可能なスペースがないか、追加の VIER が利用不可である。
5 (X'05')	このボリュームがマウントされていることも、ボリュームをマウントするのに使用可能な装置があることも確認できませんでした。
6 (X'06')	オペレーターはこのボリュームをマウントできませんでした。
7 (X'07')	データ・セットは、現在処理のためにオープンされているため、名前変更できませんでした。
8 (X'08')	データ・セットは RACF に対して定義されていますが、ユーザーがデータ・セットに対して許可されていないか、またはデータ・セットが複数のボリューム上で RACF に対して定義されています。

第 3 章 カタログ管理マクロの使用

この章では、互換性の目的でのみ使用されるカタログ管理マクロ命令について説明します。カタログ管理マクロ命令は、以下の機能を実行するのに使用できます。

- 統合カタログ機能カタログから情報を検索する。
- 以下のタイプのデータ・セットを統合カタログ機能カタログにカタログ、アンカタログ、または再カタログする。
 - 非 VSAM または SMS 管理の DASD データ・セット
 - テープ・データ・セット

アプリケーション・プログラムに関する考慮事項

カタログ管理要求は、統合カタログ機能カタログで満たすことができます。ユーザーのアプリケーション・プログラムとの関係では、以下の制約および制限を考慮してください。

- カタログ管理要求は、レジスター 1 によって指示されるパラメーター・リストに表示されています。パラメーター・リストは、CAMLST マクロを使用して生成します。CAMLST と関連のフィールドは、読み取り専用ストレージにあってはならず、16 MB 境界より下に常駐していなければなりません。
- レジスター 15 に戻りコードが入ります。戻りコードについては、「z/OS MVS システム・メッセージ 第 7 巻 (IEB-IEE)」および「z/OS MVS システム・メッセージ 第 8 巻 (IEF-IGD)」の IDC3009I の項で説明しています。

カタログ検索順序

カタログの項目の検索は、以下の方法で行われます。

1. マクロでカタログが指定されている場合、そのカタログのみが検索されます。
2. 項目が修飾項目名を使用して識別されており、その最初の修飾子がユーザー・カタログの名前または別名と同じである場合、ユーザー・カタログが検索されます。項目が見つかり、その他のカタログは検索されません。
3. マスター・カタログが検索されます。

詳細については、「z/OS DFSMS カタログのためのアクセス方式サービス・プログラム」を参照してください。

カタログからの情報の検索

カタログから項目を読み取るには、LOCATE および CAMLST マクロ命令を使用します。次の情報を使用して、ユーザーの出力域に読み取る項目を指定できます。

- データ・セットの完全修飾名または部分修飾名
- 項目を含んでいるブロックの相対ブロック・アドレス

完全修飾データ・セット名を指定した場合、データ・セットが常駐するボリュームのリストがユーザーの出力域に読み取られます。このボリューム・リストは、常にリスト内のボリューム数を示す 2 バイト項目で始まります。データ・セットが 20 を超えるボリューム上に常駐している場合、ボリューム・リスト項目の後にボリューム制御ブロックのアドレスが続きます。

制約事項: CAMLST を使用して、長さが 20 ボリュームを超えるデータ・セットを見つけた場合、最初の 20 ボリュームからの情報のみが戻されます。

カタログ・インターフェースでは、完全修飾名とは、単一のデータ・セットを表す名前です。部分修飾名とは、複数の修飾子を含めることができるが、完全なデータ・セット名は指定しない名前をいいます。

例えば、LEVEL1.LEVEL2.LEVEL3.LEVEL4 がデータ・セットである場合、LEVEL1.LEVEL2.LEVEL3.LEVEL4 が完全修飾名です。以下のデータ・セットは、部分修飾名と見なされます。

```
LEVEL1.LEVEL2.LEVEL3
LEVEL1.LEVEL2
LEVEL1
```

制約事項: カタログ・インターフェースでは、部分修飾データ・セット名を指定するために、アスタリスク (*) またはアンパーサンド (&) を指定することはできません。

ボリューム制御ブロックおよび他のカタログ・データ域の内容の説明は、「z/OS DFSMSdfp 診断解説書」を参照してください。

LOCATE 戻りコードの説明は、87 ページの『LOCATE からの戻りコード』を参照してください。

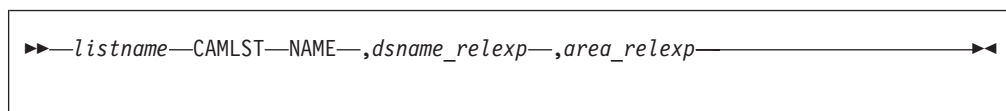
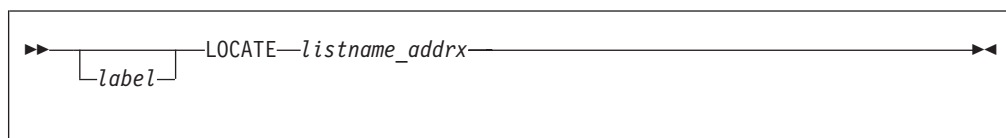
データ・セット名による情報の検索 (LOCATE と CAMLST NAME)

データ・セット名を指定して、ボリューム・リストをユーザーの出力域に戻します。ボリューム・リストは、データ・セットの部分が常駐する各ボリュームの項目から構成されます。カウント・フィールドの後に、可変数の 12 バイト項目が続きます。各 12 バイト項目は、4 バイトの装置コード、6 バイトのボリューム通し番号、および 2 バイトのボリューム・シーケンス番号から構成されます。この 12 バイト項目を 20 個だけ、ユーザーの出力域に作成できます。LOCATE は、そのデータ・セットの最初または唯一のボリューム上の DSCB の相対トラック・アドレスを含んでいる、ユーザー領域のバイト 252 から 254 を戻すことができます。そうでない場合、これらのバイトはゼロです。バイト 242 から 251 は予約済みで、バイト 255 にはゼロが入り、バイト 256 から 264 も予約済みで、プログラミング・インターフェースの対象ではありません。

データ・セットが 5 つを超えるボリュームに常駐する場合、ユーザーの出力域内のボリューム・リストは、ユーザーの出力域に読み取られたボリューム制御ブロック (VCB) です。VCB 内のカウント・フィールドには、この VCB と後続の VCB 内のボリューム項目の数が入ります。例えば、カウント 41 は、それぞれ、20 のカウントを持つ 2 つの後続の VCB と 1 を示します。次の VCB の相対トラック・アドレス (TTR) は、ユーザーの出力域のバイト 252 から 254 に入ります。データ・セットの最後の VCB は、バイト 252 から 254 に 2 進ゼロが入ります。

VSAM クラスタ上での CAMLST LOCATE は、すべてのコンポーネントのボリュームを戻します。このボリューム・リストのフォーマットは、前述の通りです。出力は、呼び出し側に戻される前に、前述の文の 2 つの段落で説明したフォーマットに変換されます。元の VSAM または統合カタログ機能の戻りコードは、レジスター 0 に保管されます。

LOCATE および CAMLST NAME マクロのフォーマットは、次のとおりです。



listname_addrx CAMLST マクロ命令によってセットアップされたパラメーター・リスト (ラベル付きリスト名) を指します。

NAME 名前によってカタログから情報を検索するには、このオペランドを表示通りにコーディングします。

dsname_relexp 完全修飾データ・セット名の仮想ストレージ位置を指定します。名前を含んでいる領域は 44 バイトの長さでなければなりません。名前は C タイプ定義定数 (DC) 命令によって定義できます。

area_relexp ユーザーが定義する必要がある 256 バイト出力域の仮想ストレージ位置を指定します。出力域はダブルワード境界で開始しなければなりません。

例

次の例では、データ・セット A.B が常駐するボリュームのリストを含んでいるカタログ項目が仮想ストレージに読み取られます。

```

                LOCATE    INDAB                READ CATALOG ENTRY FOR DATA SET A.B
*
*
*
                INTO VIRTUAL STORAGE AREA NAMED LOCAREA.
                LOCAREA MAY ALSO CONTAIN A 3-BYTE
                TTR OR A 6-BYTE SERIAL NUMBER
    
```

Check Return Codes

INDAB	CAMLST	NAME,AB,,LOCAREA
AB	DC	CL44 'A.B'
LOCAREA	DS	0D
	DS	265C

LOCATE マクロ命令は CAMLST マクロ命令を指します。NAME (CAMLST の第 1 オペランド) は、システムがデータ・セットの名前を使用してカタログ項目を検索することを指定します。AB (第 2 オペランド) は、完全修飾データ・セット名の仮想ストレージ位置を指定し、LOCAREA (第 4 オペランド) は、ユーザーが仮想ストレージ内に予約した 265 バイト域を指定します。

これらのマクロ命令を実行した後、265 バイト域にはデータ・セット A.B のボリューム・リストまたはボリューム制御ブロックが入ります。項目が見つかり、正常に読み取られた場合、レジスター 15 にゼロが入ります。そうでない場合は、レジスター 15 に戻りコードが入ります (87 ページの『LOCATE からの戻りコード』を参照してください)。

世代別データ・セット名による情報の検索 (LOCATE と CAMLST NAME)

データ・セットの完全修飾世代索引名と相対世代番号を使用して、世代別データ・セットの名前を指定します。相対世代番号の値は、世代別データ・グループ内のデータ・セットの位置を反映します。世代別データ・グループ内のデータ・セットを識別するには、以下の値を使用できます。

- ゼロ-世代別データ・グループにカタログされている最新のデータ・セット (最高の世代番号) を指定します。
- 負の数値-最新のデータ・セットより前にカタログされたデータ・セットを指定します。

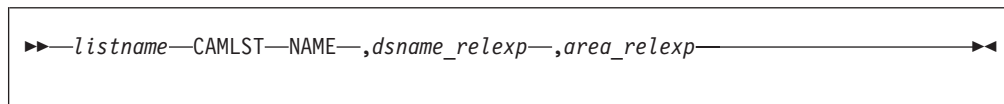
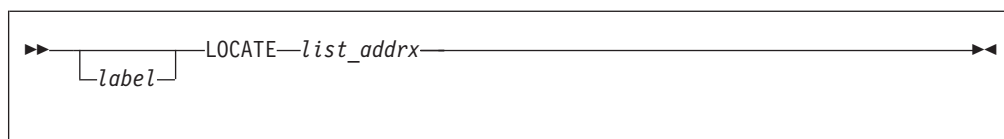
規則: DISP (処理) が他のデータ・セット用の余地をつくる DELETE であり、世代別データ・グループが存在しない場合、ジョブは完了し、削除された世代名 (G0000V00) が示されます。世代別データ・グループは存在するが、削除のために指定された範囲内にはない場合、このステップは失敗します。

- 正の数値-世代別データ・グループにまだカタログされていないデータ・セットを指定します。

ゼロまたは負の数値を相対世代番号として使用すると、ボリューム・リスト (または、ボリューム制御ブロック) をユーザーの出力域に置いて、相対世代番号を絶対世代名で置き換えます。

正の数値を相対世代番号として使用すると、絶対世代名を作成して、相対世代番号を置き換えます。カタログ内に項目がないため、ユーザー出力域の最初の 256 バイトには、ゼロが読み取られません。

LOCATE および CAMLST NAME マクロのフォーマットは、次のとおりです。



list_addrx CAMLST マクロ命令によってセットアップされたパラメーター・リスト (ラベル付きリスト名) を指します。

NAME 世代別データ・セット名によってカタログからブロックを読み取るには、このオペランドを表示通りにコーディングします。

dsname_relexp 世代索引名と相対世代番号の仮想ストレージ位置を指定します。これらを含む領域は 44 バイトの長さでなければなりません。

area_relexp ユーザーが定義する必要がある 256 バイト出力域の仮想ストレージ位置を指定します。出力域はダブルワード境界で開始しなければなりません。出力域にはカタログから作成されたボリューム・リストが入ります。データ・セットが 1 つのボリューム上に常駐する場合、バイト 252 から 254 には、DSCB の相対トラック・アドレスを入れることができます。このアドレスは、ボリュームの開始に対する相対値です。

例

次の例では、世代別データ・セット A.PAY(-3) を含んでいるボリュームのリストが仮想ストレージに読み取られます。

```

          LOCATE    INDGX          READ CATALOG ENTRY
*                                     FOR DATA SET A.PAY(-3)
*                                     INTO YOUR STORAGE
*                                     AREA NAMED LOCAREA

```

Check Return Codes

INDGX	CAMLST	NAME, APAY, , LOCAREA
APAY	DC	CL44'A.PAY(-3)'
LOCAREA	DS	0D
	DS	265C

LOCATE マクロ命令は CAMLST マクロ命令を指します。NAME (CAMLST の第 1 オペランド) は、データ・セットの名前を使用してカタログ項目の検索を開始します。APAY (第 2 オペランド) は、世代別データ・グループ内のデータ・セットの世代索引名と相対世代番号の仮想ストレージ位置を指定します。LOCAREA (第 4 オペランド) は、ユーザーがカタログ情報を受け取るために予約した 265 バイト域を指定します。

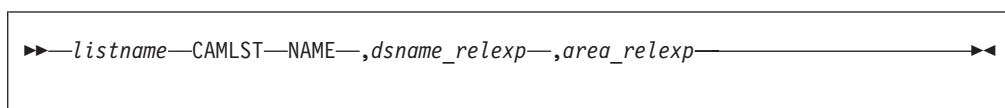
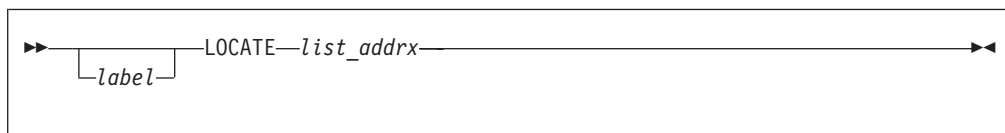
このマクロ命令を実行した後、システムはユーザーが指定した相対世代番号をデータ・セットの絶対世代名で置き換えます。制御は、プログラムの LOCATE マクロ命令の後の次の実行可能命令に戻されます。項目が見つかり、正常に読み取られた場合、レジスター 15 にゼロが入ります。そうでない場合は、レジスター 15 に戻

りコードが入ります (87 ページの『LOCATE からの戻りコード』を参照してください)。出力域の内容についての説明は、82 ページの『データ・セット名による情報の検索 (LOCATE と CAMLST NAME)』を参照してください。

別名による情報の検索 (LOCATE と CAMLST NAME)

前述の各機能では、データ・セットの名前として別名を指定できます。真の名前が指定の別名で置き換えられる点を除いて、機能は前述のとおりに行われます。

LOCATE および CAMLST NAME マクロのフォーマットは、次のとおりです。



- list_addrx* CAMLST マクロ命令によってセットアップされたパラメーター・リスト (ラベル付きリスト名) を指します。
- NAME** カタログから情報を検索するには、このオペランドを表示通りにコーディングします。
- dsname_relexp* 完全修飾データ・セット名 (最初または唯一の名前が別名) の仮想ストレージ位置を指定します。名前を含んでいる領域は、44 バイトの長さでなければなりません。名前は C タイプ DC 命令によって定義できます。
- area_relexp* ユーザーが定義する必要がある 256 バイト出力域の仮想ストレージ位置を指定します。出力域はダブルワード境界で開始しなければなりません。出力域の最初の 256 バイトには、カタログから読み取られたボリューム・リストが入ります。データ・セットが 1 つのボリューム上に常駐する場合、バイト 252 から 254 には、DSCB の相対トラック・アドレスを入れることができます。このアドレスは、ボリュームの開始に対する相対値です。

例

次の例では、データ・セット A.B.C が常駐するボリュームのリストを含んでいるカタログ項目が仮想ストレージに読み取られます (ただし、データ・セット A.B.C は別名 X.B.C によってアドレス指定されています)。

```

LOCATE    INDAB          READ CATALOG ENTRY
*                                     FOR DATA SET X.B.C
*                                     INTO VIRTUAL STORAGE
*                                     AREA NAMED LOCAREA.

```

Check Return Codes

```

INDAB    CAMLST    NAME,ABC,,LOCAREA
ABC      DC        CL44'X.B.C'
LOCAREA  DS        0D
          DS        265C

```

LOCATE マクロ命令は CAMLST マクロ命令を指します。NAME (CAMLST の第 1 オペランド) は、データ・セットの名前を使用してカタログで項目の検索を開始します。ABC (第 2 オペランド) は、データ・セットの完全修飾名の仮想ストレージ位置を指定します (この場合、データ・セット A.B.C は、その別名である X.B.C によってアドレス指定されています)。LOCAREA (第 4 オペランド) は、ユーザーが仮想ストレージ内に予約した 265 バイト域を指定します。

LOCATE 戻りコードの説明は、『LOCATE からの戻りコード』を参照してください。

相対ブロック・アドレスによるブロックの読み取り (LOCATE と CAMLST BLOCK)

このフォーマットは現在はサポートされなくなっており、結果はエラーになります。

LOCATE からの戻りコード

制御は、プログラムの LOCATE マクロ命令の後の次の実行可能命令に戻されます。カタログから情報が正常に読み取られた場合、レジスターにゼロが入ります。そうでない場合、レジスター 15 には、次の戻りコードのいずれかが入ります。表 28 の LOCATE 戻りコードを参照してください。

以下のコードの代わりに、94 ページの表 30 にリストされているコードのいずれかを受け取る場合があります。

表 28. LOCATE 戻りコード

コード	意味
0 (X'00')	操作が正常に行われました。
4 (X'04')	必要なカタログが存在しないか、オープンできないかのいずれかです。
8 (X'08')	以下のいずれかが発生しました。 <ul style="list-style-type: none"> 項目が見つからなかった。レジスター 0 にカタログ戻りコードが入ります。 ユーザーはこの操作を実行することが許可されていない。レジスター 0 に 16 進数 38 が入ります。 世代別データ・グループ (GDG) 別名が検出された。レジスター 0 に有効な索引レベルの数が入ります。別名は真の名前で置き換えられました。 世代別データ・グループ基本項目が検出された。レジスター 0 にゼロが入ります。
12 (X'0C')	無効な低レベル GDG 名が検出されました。

表 28. LOCATE 戻りコード (続き)

コード	意味
16 (X'10')	データ・セットは、指定された最低索引レベル以外のレベルに存在します。レジスター 0 に、データ・セットが検出された索引レベル番号が入ります。
20 (X'14')	構文エラー。名前フィールドの最初のバイトにブランクが検出されました。
24 (X'18')	以下のいずれかが発生しました。 <ul style="list-style-type: none"> 永続的な入出力エラーが発生した。レジスター 0 に VSAM または統合カタログ機能戻りコードが入ります。 ゼロ以外の ESTAE 戻りコード。 パラメーター・リスト内のエラー。
28 (X'1C')	予約済み。
32 (X'20')	予約済み。
36 (X'24')	DSCB 上で見つからなかったデータ・セットは仮想記憶アクセス方式 (VSAM) データ・セットを示しています。
38 (X'26')	DFSMSHsm LOCATE プリプロセッサでエラーが発生しました。
注: 統合カタログ機能カタログおよび VSAM カタログの戻りコードの資料は、「z/OS MVS システム・メッセージ 第 7 巻 (IEB-IEE)」および「z/OS MVS システム・メッセージ 第 8 巻 (IEF-IGD)」、メッセージ IDC3009I を参照してください。	

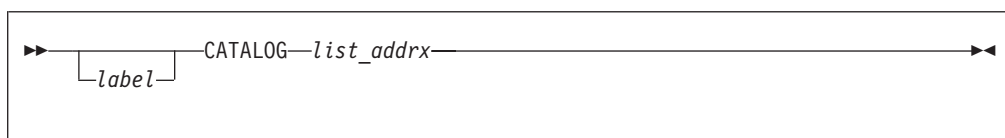
非 VSAM データ・セット・カタログ項目の使用

CATALOG および CAMLST マクロ命令を使用して、非 VSAM データ・セットのカタログ、アンカタログ、および再カタログを行うことができます。CATALOG マクロ命令を使用して CAMLST マクロ命令を指し、カタログ・オプションを指定します。

非 VSAM データ・セットのカタログ、アンカタログ、および再カタログに使用される検索アルゴリズムについての説明は、「z/OS DFSMS カタログのためのアクセス方式サービス・プログラム」内の DEFINE および DELETE コマンドの項を参照してください。

非 VSAM データ・セットのカタログ (CATALOG と CAMLST CAT)

CATALOG および CAMLST マクロのフォーマットは、次のとおりです。



UNCAT または UCATDX

このオペランドは表示どおりにコーディングしてください。UNCAT または UCATDX をコーディングできますが、いずれの場合も、不要な索引 (最高レベルの索引を除く) がデータ・セット参照と共に除去されます。

name_relexp

データ・セットの完全修飾名または索引レベルの仮想ストレージ位置を指定します。名前は 44 文字を超えることはできません。名前が 44 文字より少ない場合、後ろに少なくとも 1 つのブランクを含める必要があります。DFSMSHsm 環境では、データ・セット名が 44 文字より少ない場合、44 文字の長さに達するまでブランクを埋め込む必要があります。

例

次の例では、データ・セット A.B.C のカタログ項目が、カタログから除去されます。

```
*          CATALOG REMOVE          REMOVE REFERENCES TO DATA SET A.B.C
                                         FROM CATALOG
```

Check Return Codes

REMOVE	CAMLST	UNCAT,DSNAME	
DSNAME	DC	CL6'A.B.C'	ONE BLANK FOR DELIMITER

CATALOG マクロ命令は CAMLST マクロ命令を指します。UNCAT (CAMLST の第 1 オペランド) は、データ・セットへの参照をカタログから除去することを指定します。DSNAME (第 2 オペランド) は、参照を除去するデータ・セットの完全修飾名の仮想ストレージ位置を指定します。

制御は、プログラムの CATALOG マクロ命令の後続の命令に戻されます。レジスター 15 には、93 ページの『CATALOG からの戻りコード』で説明されている戻りコードのいずれかが入ります。

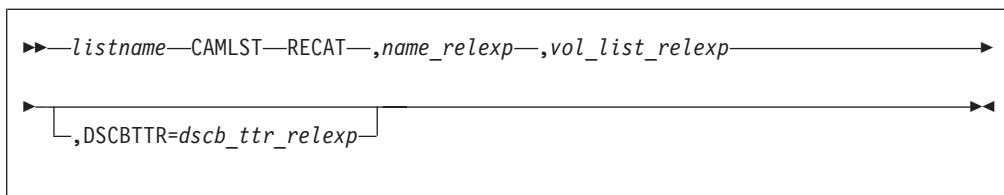
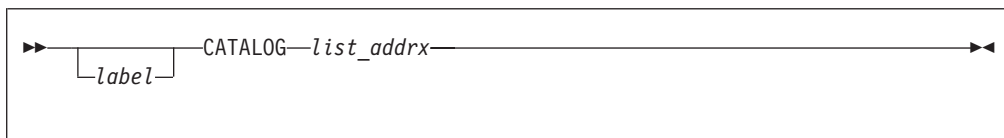
制約事項: CAMLST UNCAT または UCATDX 機能は、システム管理データ・セットに対してはサポートされません。これらは無視されます。機能は実行されず、戻りコードは 0 です。

非 VSAM データ・セットの再カタログ (CATALOG と CAMLST RECAT)

CATALOG および CAMLST マクロ命令を使用して、非 VSAM データ・セットを再カタログすることができます。再カタログは通常、データ・セットが新規ボリュームに拡張された場合に必要になります。

データ・セットが常駐する各ボリュームの項目からなる完全なボリューム・リストを仮想ストレージ内に作成します。リストの最初の 2 バイトは、リスト内の項目の数を示します。この数値はゼロであってはなりません。各 12 バイトのボリューム・ポインターは、4 バイトの装置コード、6 バイトのボリューム通し番号、および 2 バイトのデータ・セット・シーケンス番号から構成されます。直接アクセス・ボリュームの場合、シーケンス番号は常にゼロです。

CATALOG および CAMLST マクロのフォーマットは、次のとおりです。



list_addrx CAMLST マクロ命令によってセットアップされたパラメーター・リスト (ラベル付きリスト名) を指します。

RECAT このオペランドは表示どおりにコーディングしてください。

name_relexp データ・セットの完全修飾名の仮想ストレージ位置を指定します。名前は 44 文字を超えることはできません。名前が 44 文字より少ない場合、後ろに少なくとも 1 つの空白を含める必要があります。DFSMSHsm 環境では、データ・セット名が 44 文字より少ない場合、44 文字の長さまで空白を埋め込む必要があります。C タイプ DC 命令によって名前を定義できます。

vol_list_relexp ボリューム・リストを含んでいる領域の仮想ストレージ位置を指定します。領域はハーフワード境界で開始しなければなりません。

DSCBTTR=dscb_ttr_relexp

ID DSCB の 3 バイトの相対トラック・アドレス (TTR) の仮想ストレージ位置を指定します。この DSCB は、データ・セットの最初または唯一のボリューム上にあります。アドレスは、ボリュームの開始に対する相対値です。

例

次の例では、2 つのボリューム上にある A.B.C という名前のデータ・セットが、3 番目のボリュームを追加するために再カタログされます。以前は 2 つの項目のみを含んでいたボリューム・リストに、項目が 1 つ追加されます。

```

          CATALOG RECATLG          RECATALOG DATA SET
*                                     A.B.C ADDING A NEW
*                                     VOLUME
    
```

Check Return Codes

```

RECATLG  CAMLST  RECAT,DSNAME,,VOLUMES
DSNAME   DC      CL6'A.B.C '    FOR DELIMITER ONE BLANK
VOLUMES  DC      H'3'          THREE VOLUMES
          DC      X'3010200E'    3380 DISK DEVICE CODE
          DC      CL6'000014'    VOLUME SERIAL NUMBER
          DC      H'0'          SEQUENCE NUMBER
          DC      X'3010200E'    3380 DISK DEVICE CODE
          DC      CL6'000015'    VOLUME SERIAL NUMBER
          DC      H'0'          SEQUENCE NUMBER
          DC      X'3010200E'    3380 DISK DEVICE CODE
          DC      CL6'000016'    VOLUME SERIAL NUMBER
          DC      H'0'          SEQUENCE NUMBER
    
```

CATALOG マクロ命令は CAMLST マクロ命令を指します。RECAT (CAMLST の第 1 オペランド) は、データ・セットを再カタログすることを指定します。DSNAME (第 2 オペランド) は、再カタログするデータ・セットの完全修飾名の仮想ストレージ位置を指定します。VOLUMES (第 4 オペランド) は、ユーザーが作成したボリューム・リストの仮想ストレージ位置を指定します。

制御は、プログラムの CATALOG マクロ命令の後続の命令に戻されます。データ・セットが正常に再カタログされた場合、レジスター 15 にゼロが入ります。そうでない場合、レジスター 15 には、『CATALOG からの戻りコード』で説明されている戻りコードのいずれかが入ります。

CATALOG からの戻りコード

制御は、CATALOG マクロ命令の後続の命令に戻されます。レジスター 15 には、次の戻りコードのいずれかが入ります。

以下のコードの代わりに、94 ページの表 30 にリストされているコードのいずれかを受け取る場合があります。

表 29. CATALOG 戻りコード

コード	意味
0 (X'00')	操作が正常に行われました。
4 (X'04')	必要なカタログが存在しないか、オープンされていないかのいずれかです。
8 (X'08')	以下のいずれかが発生しました。 <ul style="list-style-type: none"> 既存のカタログ構造は、要求された操作と矛盾している。レジスター 0 にはカタログ戻りコードが入り、レジスター 1 にはゼロが入ります。 ユーザーは、操作を実行することが許可されていない。レジスター 0 には 10 進数 56 ('38') が入り、レジスター 1 にはゼロが入ります。
12 (X'0C')	予約済み。
16 (X'10')	予約済み。
20 (X'14')	カタログ・データ・セット内のスペースが不十分です。
24 (X'18')	予約済み。

表 29. CATALOG 戻りコード (続き)

コード	意味
28 (X'1C')	以下のいずれかが発生しました。 <ul style="list-style-type: none"> 永続的入出力エラーまたはリカバリー不能エラーが発生した。 パラメーター・リストでエラーが検出された。 ESTAE または GETMAIN からゼロ以外の戻りコードが戻された。

カタログ・プロセッサ戻りコード

表 30 は、カタログ・プロセッサからの戻りコードについて説明しています。

表 30. 戻りコード

コード	意味
0	成功。
4	必要なカタログを割り振れません。
8	データ・セットが見つからなかったか、カタログをオープンできなかったかのいずれかです。
24	検索中に入出力エラーまたはリカバリー不能エラーが発生しました。
28	検索以外の時点で、入出力エラーまたはリカバリー不能エラーが発生しました。
40	提供されたスペースが不十分です。計算されたスペースのサイズが、カタログ・パラメーター・リストの CTGFDBK 域に戻されます。
44	提供されたスペースが不十分です。必要なサイズは不明です。
48	機能が無効でした。
56	セキュリティーまたはパスワード検査に失敗しました。
84	ページ日付が満了していないことが原因で SCRATCH が失敗したので、DELETE が失敗しました。
164	ESTAE または GETMAIN 戻りコードがゼロ以外でした。
168	装置タイプの不一致が原因で SCRATCH が失敗したので、DELETE が失敗しました。
184	データ・セットが現在オープンされていることが原因で SCRATCH が失敗したので、DELETE が失敗しました。

第 4 章 ユーザー独自のチャネル・プログラムの実行

この章には、execute-channel-program (EXCP) および execute-channel-program-virtual-real (EXCPVR) マクロ命令に関する情報が含まれており、この情報は他の IBM オペレーティング・システムとの互換性のために提供されています。特に指示がない限り、EXCP に対する参照は、EXCPVR にも同様に適用されます。EXCPVR と EXCP の差については、112 ページの『実ストレージ内の固定チャネル・プログラムの実行 (EXCPVR)』を参照してください。IBM では、EXCP の代わりに、VSAM などのアクセス方式を使用することをお勧めします。

EXCP マクロ命令を使用すると、装置特性に基づいてデータ編成を制御できます。この機能の例外は、拡張区分データ・セット (PDSE)、拡張フォーマット・データ・セット、スプールおよびダミー・データ・セット、TSO 端末、および HFS (階層ファイル・システム) データ・セットです。これらは、EXCP を使用したユーザー作成アプリケーションではサポートされません。この章では、EXCP マクロ命令のアプリケーションと機能について説明し、特定の制御ブロックおよびマクロ命令の説明を含めます。装置の差異やプログラムの変更など、EXCP の操作に影響を与える要因についても説明します。

この章をお読みになる前に、ユーザーのチャネル・プログラムに必要な入出力装置の操作特性を十分に理解しておく必要があります。操作特性については、それぞれの入出力装置の IBM 資料で説明しています。資料の多くは、xi ページの表 1 にリストされています。また、以下の資料に記載されている情報も理解しておくことが必要です。

- 「*HLASM Programmer's Guide*」および「*Assembler H V2 Programming Guide*」は、アセンブラ言語でプログラムをコーディングするために必要な情報を提供しています。
- 「*z/Architecture 解説書*」(SA88-8773) は、チャネル・コマンド・ワード (CCW) およびチャネル・プログラムについて説明しています。
- 「*z/OS DFSMS データ・セットの使用法*」には、オペレーティング・システム下での入出力処理の標準手順が示されています。
- 「*z/OS DFSMS Macro Instructions for Data Sets*」は、アセンブラ言語でコーディングされたプログラムで使用できるシステム・マクロ命令について説明しています。

EXCP は、主として標準アクセス方式では処理できない入出力プログラミング状態を対象としています。ユーザー独自のアクセス方式を作成する際には、入出力操作用の EXCP を組み込んでください。また、標準外磁気テープ・ラベルの処理用の EXCP (ラベルの読み取りと書き込み、ボリュームの位置決めを含む) も使用する必要があります。

EXCP を発行するには、ユーザーのプログラム域にチャネル・プログラム (チャネル・コマンド・ワードのリスト) と制御ブロックを提供します。これにより、入出

カプロセスは、指定された装置の入出力要求をスケジュールし、コマンドを実行し、割り込みを処理し、エラー・リカバリー手順を指示し、入出力要求の結果を通知します。

EXCP の使用

ここでは、EXCP マクロ命令を発行する際に必要な手順について説明します。EXCP マクロ命令を直接発行するには、以下のタスクを実行します。

1. DCB および OPEN マクロ命令を使用して、データ制御ブロック (DCB) を構成し、オープンする。113 ページの『データ制御ブロック (DCB) フィールド』を参照してください。
2. 該当する装置上の入出力操作のチャンネル・コマンドを含むチャンネル・プログラムを作成する。
3. チャンネル・プログラムに関する情報が入った入出力ブロック (IOB) を構成する。127 ページの『入出力ブロック・フィールド』を参照してください。
4. チャンネル・プログラムが終了したときに完了コードを通知するイベント制御ブロック (ECB) を構成する。136 ページの『イベント制御ブロック・フィールド』を参照してください。
5. EXCP マクロ命令を発行して、入出力操作を開始および監視するルーチンに IOB のアドレスを渡す。

入出力プロセスは通常、チャンネル・プログラムのスケジューリングと完了コードの通知を扱います。

EXCP を発行した後、ECB のアドレスを指定した WAIT または EVENTS マクロ命令を発行して、チャンネル・プログラムが終了したかどうかを判別します。ボリューム切り替えが必要な場合 (装置例外または DASD エクステンション終了のため)、EOV マクロ命令を発行します。データ・セット処理が完了したら、CLOSE マクロ命令を発行して DCB を復元します。

EXCP 用のプログラミング・インターフェースは、MVS/370 および MVS/ESA™ または z/OS との間で互換性があります (ただし、16 MB より上の仮想ストレージおよび中央ストレージ・アドレスのサポートを除きます)。EXCP は、24 ビットまたは 31 ビット・アドレッシング・モードで実行されるプログラムに対して使用可能です。以下の制約事項が適用されます。

- EXCP は、format-1 CCW、IDAW、DCBE、および XTIOB 項目を除く、24 ビット仮想ストレージ・インターフェースのみをサポートします。また、入出力操作に関連した領域 (例えば、チャンネル・コマンド・ワード、IOB、DEB、付加ルーチンなど) はすべて、24 ビット仮想アドレス可能のままではなりません。EXCP (チャンネル・コマンド・ワード変換プログラム) は、24 ビット仮想入出力バッファを 16 MB より上の実ストレージに固定できます。チャンネル・コマンド・ワード (CCW) が 16 MB より上の中央ストレージ・アドレスを参照している場合、CCW 変換プログラムは、その CCW 用に 1 つ以上の間接アドレッシング・ワード (IDAW) を作成します。仮想アドレスは、IDAW または format-1 CCW 内のものを除いて、すべて 16 MB より下になければなりません。V=R ユーザーの場合、CCW と IDAW は、16 MB より下の実ストレージになければなりません。

- ユーザー指定の付加ルーチンは、24 ビット・アドレッシング・モードで制御が与えられ、同じモードで制御を戻す必要があります。

IDAW 要件

EXCP 呼び出し側によって提供される仮想チャネル・プログラムは、IDA フラグが設定された 1 つ以上の CCW を持ち、これらの CCW のアドレス部分は単一の 31 ビットまたは 64 ビット IDAW を指すことができます。この EXCP 機能は、仮想 IDAW と呼ばれます。31 ビット IDAW は、最大 31 ビットまでの有効な仮想アドレスを含むことができます。仮想 IDAW は、以下を除くすべての仮想 CCW でサポートされます。

- チャネル内転送 (TIC) コマンド
- スキップ標識が設定された、読み取り、逆方向読み取り、およびセンス・コマンド
- すべての非データ転送タイプのコマンド。例えば、再調整、巻き戻し、前送り、折り返し、ブロック・データ・チェック、ノーオペレーション、制御コマンド

仮想 IDAW によって指されるストレージは、2 GB より上または下にある中央ストレージでバックアップできます。どちらのタイプの仮想 IDAW も、2 GB より上のバックアップ・ストレージを指すことができます。

注: 64 ビット IDAW は、z/Architecture モードでのみサポートされ、直接アクセス・ストレージ・デバイス (DASD) および IBM 提供のすべてのカートリッジ磁気テープ装置に対してのみサポートされます。ユーザーは、UCB 内のビット UCBEIDAW を調べて、64 ビット IDAW がその装置でサポートされているかどうかを判別する必要があります。z/OS 1.6 以降は、システムは常に z/Architecture モードで稼働します。

EXCPVR 入力に対しても、これと同じアドレッシング上の制約事項が適用されますが、例外として、ユーザー作成の CCW で IDAW を使用することにより、31 ビットまたは 64 ビットの実データ域を指定できます。フォーマット 0 CCW および IDAW は、すべて 16 MB より下の実データ域になければなりません。フォーマット 1 CCW は、2 GB より下の実データ域になければなりません。

EXCP 要件

ここでは、EXCP を発行するために必要なチャネル・プログラムと、直接に、またはマクロ命令を使用して構成される制御ブロックについて説明します。

EXCP/EXCPVR 入出力操作に関連した領域 (例えば、CCW、IDAW、IOB、DEB、DCB、付加ルーチンなど) は、すべて 24 ビット・アドレス可能のままでなければなりません。ただし、装置がそれをサポートする場合には、EXCP プロセッサは 24 ビット、31 ビット、および 64 ビット仮想入出力バッファを実 2 GB より上に固定できます。

EXCPVR のページの固定化に関する追加情報は、142 ページの『ページ固定 (Page Fix) および EXCPVR 入出力開始 (Start I/O) 付加ルーチン』を参照してください。

チャンネル・プログラム

ユーザーが提供するチャンネル・プログラムは、ダブルワード境界上の CCW で構成されます。各チャンネル・コマンド・ワードは、実行するコマンドを指定し、データ転送コマンドの場合は、ソースまたは宛先の領域を指定します。CCW 命令コードについては、それぞれの入出力装置の IBM 資料で説明しています。

データ・チェーニングとコマンド・チェーニングはどちらも、チャンネル・コマンド・ワードで該当するチェーニング・ビットを設定し、IOB 内にチェーニングのタイプを示すことによって指定できます。チャンネル・プログラムの実行中に入出力エラーが発生した場合、IOB 内の対応するチェーニング・ビットを設定することが必要です。そうしないと、エラー・リカバリーが不可能になることがあります。ユーザーのデータの保全性が損なわれる恐れがあります。(追加情報については、127 ページの『入出力ブロック・フィールド』を参照してください。) 同じチャンネル・コマンド・ワード内でデータ・チェーニングとコマンド・チェーニングの両方を指定した場合は、データ・チェーニングが優先されます。

EXCP は、変更の方法に関係なく、自分自身を変更するチャンネル・プログラムはサポートしません(ただし、100 ページの『仮想および中央ストレージ・アドレス』の注記を除きます)。仮想ストレージ内での意図された変更は、実チャンネル・プログラムの実行には影響を与えません(100 ページの『実行時のチャンネル・プログラムの変更』を参照してください)。

EXCP で使用される制御ブロック

EXCP を使用するには、IOB、ECB、DCB、DEB、およびオプションの IDAW、IOBE、DCBE、UCB、および IEDB の機能と構造を十分に理解している必要があります。DCB、IOB、IOBE、IEDB、および ECB フィールドについては、113 ページの『制御ブロック・フィールド』で、図解して説明しています。EXCP および EXCPVVR で使用される DEB フィールドは、427 ページの『付録 B. 制御ブロック』に示されています(すべての DEB フィールドについては、「z/OS DFSMSdfp 診断解説書」で解説しています)。IDAW の取り扱いは、143 ページの『SIO 付加ルーチン』で説明しています。UCB については、「z/OS MVS Data Areas, Vol 5 (SSAG-XTLST)」で解説しています。

ユーザー提供の EXCP 制御ブロックは、ユーザー・プログラムの記憶保護キーでプログラムの変更が許可されているストレージ内に配置する必要があります。これらの制御ブロックについて、以下で説明します。

入出力ブロック

IOB は、問題プログラムとシステム間の通信に使用されます。これは、他の制御ブロックのアドレスを提供し、チェーニングのタイプや入出力操作の進行状況など、チャンネル・プログラムに関する情報を保守します。ユーザーは IOB を定義し、そのアドレスを EXCP または EXCPVVR マクロ命令の唯一のパラメーターとして指定する必要があります。127 ページの『入出力ブロック・フィールド』を参照してください。

イベント制御ブロック

ECB は、ユーザーに完了コードを提供し、チャンネル・プログラムの完了にエラーが伴っているか、いないかを説明します。WAIT または EVENTS マクロ命令では(入

出力操作と問題プログラムを同期するのに使用できます)、ECB を識別する必要があります。ユーザーは ECB を定義し、そのアドレスを IOB に指定する必要があります。136 ページの『イベント制御ブロック・フィールド』を参照してください。

データ制御ブロック

DCB は、チャンネル・プログラムによって読み取られる、または書き込まれるデータ・セットの特性や処理要件に関する情報をシステムに提供します。DCB は、EXCP 用のパラメーターを含む DCB マクロ命令によって生成する必要があります。ユーザーが付加ルーチンを使用していない場合、短形式の DCB が構成されず。このような DCB は、縮小エラー・リカバリーをサポートしません。ユーザーは、DCB のアドレスを IOB に指定します。

拡張データ制御ブロック

DCBE は、追加の処理オプションを提供します。現在 EXCP によってサポートされるオプションは、EOV マクロを発行する場合の EODAD パラメーターと、OPEN および EOVS マクロを発行する場合の BLKSIZE パラメーターだけです。125 ページの『拡張データ制御ブロック』を参照してください。

データ・エクステント・ブロック

DEB は、関連データ・セットおよび他の制御情報の 1 つ以上のエクステント項目を含みます。エクステントは、特定のデータ・セットが使用 (または、予約) する入出力装置上の物理境界の全部または一部を定義します。各エクステント項目には、入出力装置のタイプと位置に関する情報を提供する装置制御ブロック (UCB) のアドレスが入ります。複数のエクステント項目が同じ UCB アドレスを含むこともできます。オペレーティング・システムがサポートするすべての入出力装置で、DCB に対する OPEN マクロ命令の実行時に DEB が生成されます。システムは DEB のアドレスを DCB に入れます。138 ページの『データ・エクステント・ブロック・フィールド』を参照してください。

共通拡張入出力ブロック

IOBE は、ERP 処理のレベルを制御することができ、IEDB に対するアンカーを提供します。IOBE は IOB の拡張であり、IOB と同様に、EXCP のユーザーとシステム間の通信を提供します。この制御ブロックはオプションです。131 ページの『共通拡張入出力ブロック・フィールド』を参照してください。

入出力エラー・データ・ブロック

システムは IEDB を使用して拡張エラー情報を提供します。この制御ブロックはオプションです。134 ページの『入出力エラー・データ・ブロック・フィールド』を参照してください。

チャンネル・プログラムの実行

ここでは、ユーザーが EXCP を発行した後、システムがユーザーのチャンネル・プログラムと制御ブロックを使用する方法について説明します。

チャンネル・プログラムの開始

EXCP を発行すると、IOB に指定されたチャンネル・プログラムの実行を要求します。入出力プロセスは、この要求に関連した制御ブロックのフィールドを検査し

て、要求の妥当性検査を行います。入出力プロセスが制御ブロック内で無効な情報を検出した場合、異常終了手順を開始します。EXCP プロセッサは、次のアドレスを取得します。

- IOB から DCB
- DCB から DEB
- DEB から UCB

ユーザーが入出力開始 (SIO) 付加ルーチンを提供した場合、EXCP プロセッサはそれに制御を渡します。SIO 付加ルーチンからの戻りアドレスによって、EXCP プロセッサが入出力操作を実行するか、スキップするかが決まります。SIO 付加ルーチンおよびその EXCP プロセッサへのリンケージについての説明は、141 ページの『入出力開始 (Start-I/O) 付加ルーチン』を参照してください。

仮想および中央ストレージ・アドレス

ユーザーが構成するチャンネル・プログラムは、仮想アドレスを含んでいます。

ユーザーのプログラムが、JCL EXEC ステートメントで ADDRSPC=REAL (V=R) を指定して APF 許可のもとで実行されているのでない限り、EXCP プロセッサは、以下のタスクを実行します。

- 仮想チャンネル・プログラムをコピーし、そのコピーを変換して、中央ストレージ・アドレスのみを使用するようにする。
- チャンネル・プログラムで指定されたデータ転送操作の入出力域として使用されるページを、実ストレージに固定する。

EXCP プロセッサは、実ストレージの一部に、変換済み (実) チャンネル・プログラムを構築します。

V=R アドレス・スペース内の APF 許可プログラムの入出力操作にユーザーが構成したチャンネル・プログラムは、変換されません。アドレス・スペースが V=R であるため、ユーザーによって作成される CCW はすべて正しい実データ・アドレスを持つからです。(変換はチャンネル・プログラムを再作成するだけで、CCW は直接使用されます。)

アクティブ・チャンネル・プログラムが、V=R アドレス・スペース内に読み取られたデータまたはプロセッサ命令によって変更された場合、APF 許可アドレス・スペースには影響を与えますが、V=V アドレス・スペースには影響を与えません。

実行時のチャンネル・プログラムの変更

V=R アドレス・スペース内の APF 許可プログラムを除いて、システムはユーザーのチャンネル・プログラムの変換コピーを実ストレージ内に作成します。ユーザーのページ固定 (page-fix) または入出力開始 (start-I/O) 付加ルーチンによって行われるチャンネル・プログラムの変更は、実チャンネル・プログラムに影響を与えます。その後、プロセッサ命令または入出力操作によるデータの読み取りを使用して、チャンネル・プログラムのコピーに対して行われる変更は、変換された実チャンネル・プログラムには影響を与えません。このように、V=V アドレス・スペースでは、アクティブ・チャンネル・プログラムに対する変更の試みは、チャンネル・プログラムの仮想イメージにのみ影響を与え、チャンネル・サブシステムによって実行されている実チャンネル・プログラムには影響を与えません。

変更されたチャンネル・プログラムを実行したい場合は、プログラムで別の EXCP マクロを発行するか、またはチャンネル終了 (channel-end) 付加ルーチンをオフセット 8 で戻すことができます。

装置タイプに関する考慮事項

さまざまな装置クラスまたはタイプで、以下で説明する要因を考慮してください。

DASD チャンネル・プログラム接頭部コマンド

直接アクセス装置の場合、IOB にシーク・アドレスを指定します。CKD 装置の場合は、システムが CCW チェーンを構成します。CCW チェーンは DEB に指定された seek と set file mask を発行し、制御をユーザーのチャンネル・プログラムの実バージョンに渡します。ユーザーは初期シークまたはファイル・マスク設定 CCW を発行することはできません。ECKD 装置の場合、システムは、seek および set file mask コマンドの代わりに、define extent または prefix コマンドを構成します。システムは、define extent または prefix コマンドを構築する際に、DEBXDEF の内容 (427 ページの『付録 B. 制御ブロック』で説明) を使用します。ユーザーは、チャンネル・プログラムの先頭で define extent コマンドを発行できます。システムは、データ域をコピーし、開始と終了エクステント・アドレスおよびファイル・マスク・バイトを置き換えます。ユーザーのプログラムは DEVTYPE マクロを発行して、装置が ECKD をサポートするかどうかを確認できます。最小タイプの呼び出し、つまり INFO=DASD を指定した INFOLIST タイプを使用してください。

CKD と ECKD の両方とも、ファイル・マスクは prohibit seek-cylinder CCW に設定されるか、またはスペースが全シリンダーに割り振られていない場合は、seek-head コマンドに設定されます。データ・セットが INPUT のためにオープンされる場合、write CCW も禁止されます。ECKD 装置の場合は、チャンネル・プログラムに locate-record CCW を含めることができます。

INFO=DASD パラメーターを持つ DEVTYPE マクロは、装置が locate record extended コマンドをサポートするかどうか、およびそれがコントローラー・キャッシュを持っているかどうかユーザーのプログラムに知らせます。193 ページの『DEVTYPE - INFO 形式』を参照してください。

DASD 回転位置感知機構

チャンネル・プログラムでセクター数を提供すると、装置やチャンネルの使用がより効率的になります。プログラムは、read sector コマンドを使用してセクター数を読み取ったり、セクター変換ルーチンを使用してセクター数を計算したりすることができます。このルーチンについては、154 ページの『RPS 装置上のブロックのセクター番号の入手』で説明しています。

コマンド再試行に関する考慮事項

コマンド再試行は、多くの IBM 3990 ストレージ・コントローラーが持つ機能の 1 つです。チャンネル・サブシステムは、再試行要求を受け取ると CCW の実行を繰り返し、追加の入出力割り込みを行う必要はありません。例えば、ストレージ管理機能は、一時的エラーから回復するために再試行手順を開始できます。

チャンネル・プログラムの実行時のコマンド再試行は、開始プログラムによって以下の状態が検出される原因になります。

- **CCW の変更:** V=R アドレス・スペースで実行されている APF 許可プログラムの場合、チャンネル・プログラム内で使用される CCW は、CCW 操作が正常に完了する前に変更されてはなりません。コマンド再試行機能を使用していない場合、コマンドはチャンネルによって 1 度だけストレージから取り出されます。この場合、プログラムは、条件コードまたはプログラム制御割り込み (PCI) を通して、CCW がチャンネルによって取り出され、受け入れられたことを判別できます。CCW は実行前に変更される可能性があります。コマンド再試行機能を使用している場合、チャンネルはコマンド再試行シーケンスで再度 CCW をストレージから取り出すので、この手順を繰り返すことができません。データ・チェーンの場合、チャンネルはデータ・チェーンの最初の CCW から始めて、コマンドを再試行します。
- **プログラム制御割り込み (PCI):** PCI フラグを含んでいる CCW は、複数のプログラム制御割り込みの原因になります。これが起きるのは、PCI フラグ付き CCW がコマンド再試行手順で再試行され、CCW が実行されるたびに PCI が生成された場合です。
- **残余カウント:** チャンネル・プログラムがコマンドの再試行時に早期終了した場合、チャンネル状況ワード (CSW) 内の残余カウントは、必ずしもストレージの使用量を示していません。例えば、ストレージ管理が誤長レコード (wrong-length record) エラー条件を検出した場合、コマンド再試行が成功するまでは、間違った残余カウントが CSW に保管されています。再試行が成功すると、CSW 内の残余カウントは、データ転送の正しい長さを反映するようになります。
- **コマンド・アドレス:** コマンド再試行を伴うデータ・チェーンの場合、CSW は PCI の時点で実行済みの CCW の数を示していない場合があります。次の例を見てください。

CCW#	チャンネル・プログラム
1	Read, data chain
2	Read, data chain
3	Read, data chain, PCI
4	Read, command chain

この例では、ストレージ管理は Read #3 でコマンド再試行をシグナルし、チャンネルがコマンド再試行のためにコマンド・アドレスを Read #1 にリセットした後で、プロセッサが PCI を受け入れるものと想定します。PCI 用に保管される CSW には、チャンネルが実際に Read #3 まで進行した時点で、Read #1 のコマンド・アドレスが入ります。

- **データ読み取り時のバッファ内容のテスト:** バッファをテストして CCW が実行された時期を判別し、このデータに基づいて実行を継続するプログラムは、エラーが検出されて CCW が再試行された場合、間違った結果を得る可能性があります。

磁気テープに関する考慮事項

磁気テープ装置の場合、システムが CCW チェーンを構成して、DEB で指定されたモードに設定し、制御をユーザーのチャンネル・プログラムの実バージョンに渡します。(ユーザー自身がモードを設定することはできません。) カートリッジ磁気テープ装置の場合、モード・バイトは、モード設定コマンドなどの監視プログラム・チャンネル・コマンド・ワードを禁止します。プログラムが入力または逆方向読み取りのためにテープをオープンし、ユーザーがボリュームに書き込むための RACF 権限

を持っていない場合、システムは通常、書き込みを防止します。リール・テープを使用している場合、システムはオペレーターに対して書き込みリングを除去するように要求することによって、これを行います。カートリッジ・テープを使用している場合、システムは Mode Set コマンドを使用してこれを行います。テープの取り扱いについては、「z/OS DFSMS Using Magnetic Tapes」を参照してください。

IBM 3800 上のデータ脱落条件

IBM 3800 印刷サブシステムを使用している場合、取り消しキーまたはシステム再始動が必要な (system-restart-required) 紙詰まりが発生すると、DCBIFLGS にデータ脱落標識が設定され、さらに脱落ページ数およびチャンネル・ページ ID が拡張 UCB に保管されます。プリンターに要求を再発行する前に、データ脱落標識ビット (DCBIFLDT) と DCBIFLGS フィールドの最初の 2 バイトをゼロにリセットしてください。追加情報については、「IBM 3800 Printing Subsystem Programmer's Guide」および「IBM 3800 Printing Subsystem Models 3 and 8 Programmer's Guide」を参照してください。

実行の完了

サブチャンネル状況ワード (SCSW) でチャンネル終了 (channel-end) 条件を受け取った場合、システムはチャンネル・プログラムが完了したものと見なします。チャンネル終了 (CHE) または異常終了 (ABE) 付加ルーチンで別の指示をしていない限り、完了コードが ECB、IOB、および IEDB に入ります。完了コードは、チャンネル終了に関連したエラーを参照します。装置終了とチャンネル終了が同時に発生した場合、装置終了に関連したエラー (つまり、装置例外または装置チェック) も考慮されます。

チャンネル終了の後に装置終了が続き、エラーが装置終了に関連している場合、ECB 内の完了コードはエラーを示しません。ただし、装置とチャンネルの状況は、装置のためにシステムによって保管されます。任意のアドレス・スペースからその入出力装置に送られる次の入出力要求は、代行受信 (intercepted) としてマークされます。エラーは永続と見なされ、代行受信された要求の ECB 内の完了コードが代行受信を示します。DCB の DCBIFLGS フィールドも永続エラーを示します。

write-tape-mark または erase-long-gap CCW が、チャンネル・プログラムの最後またはは唯一の CCW である場合、入出力プロセスは装置終了エラーのためにリカバリー手順を試行しないので注意してください。このような環境では、NOP CCW をユーザーの write-tape-mark または erase-long-gap CCW にチェーニングするコマンドが、装置終了エラー・リカバリー手順を開始します。

装置終了エラーに備えるために、ユーザーはそうしたエラーの原因になる装置特性を十分に理解しておく必要があります。チャンネル・プログラムの 1 つが終了した後、その装置に対する次の要求が代行受信されたことが判別されるまでは、バッファ・スペースを解放しないでください。代行受信された要求は、再発行できません。

割り込み処理およびエラー・リカバリー手順

入出力割り込みを使用して、プロセッサは、入出力操作のフェーズの終了または装置上の外部アクションを示す入出力装置からの信号に応答できます。入出力割り

込みに関する詳しい解説は、資料「z/Architecture 解説書」に記載されています。特定の装置による割り込みについての説明は、それぞれの装置の IBM 資料を参照してください。

エラー条件が割り込みに関連している場合、システムは該当の装置依存エラー・ルーチンをスケジュールします。その後、オペレーティング・システムは、エラーのあるチャンネル・プログラムに関係のない別の要求を開始できることもあります。エラー・リカバリー手順 (ERP) でエラーを訂正するのに失敗した場合、システムはエラー・コードを ECB、IOB、および IEDB に入れます。

チャンネル・プログラムは、前のチャンネル・プログラムの正常終了に依存する場合があります (例えば、1 つのチャンネル・プログラムが、別のプログラムを構築するために使用するデータを取得する場合など)。前のチャンネル・プログラムは、関連要求が呼び出されており、システムに対して識別されていなければなりません。この手順についての説明は、127 ページの『入出力ブロック・フィールド』の IOBFLAGS の項、および 224 ページの『入出力要求のパーズと復元』を参照してください。

関連要求のチャンネル・プログラム内で永続エラーが発生した場合、システムはすべての従属チャンネル・プログラムの要求キュー・エレメントを削除し、要求を実行せずにそれを呼び出し側に戻します。エラーが生じたチャンネル・プログラムに依存するすべての要求について、システムは完了コードを ECB に入れます。また、システムは DCB の DCBIFLGS フィールドの最初の 2 ビット位置に 1 を入れます。エラー・フラグを持つ DCB に対する要求は、実行せずに完了が通知されます。エラーが生じたチャンネル・プログラムに依存する要求を再発行するには、DCB の DCBIFLGS フィールドの最初の 2 ビットをゼロにリセットします。その後で、必要なチャンネル・プログラムごとに EXCP を再発行します。

エラー・リカバリー手順によるチャンネル・プログラムの再実行: 一部の環境では、ERP がチャンネル・プログラムを最初から再実行できることがあります。ユーザーは、CCW に障害が起きた後に ERP を再始動できるチャンネル・プログラムを作成することが必要な場合があります。

CCW が前の CCW によって使用されたデータ域を変更すると、チャンネル・プログラムは正常に再実行されない場合があります。

以下は、V=R アドレス・スペース内の APF 許可プログラムによって再実行された場合に、チャンネル・プログラムが正しい結果を与えられない可能性があるいくつかの状態を示しています (100 ページの『実行時のチャンネル・プログラムの変更』を参照してください)。

- チャンネル・プログラムがそれ自体を変更する。
- チャンネル・プログラムが完了したという通知を受け取る前に、アプリケーション・プログラムまたは PCI 付加ルーチンがチャンネル・プログラムまたはデータ域を変更する。通常は、チャンネル・プログラムの最後に CCW を追加することを試みることができます。

例: 以下は、常に正常に実行されるとは限らない DASD チャンネル・プログラムの例です。このようなチャンネル・プログラムの使用は、お勧めできません。

```
CCW  X'47',LRArea,X'60',L'LRArea  Locate Record
CCW  X'86',Data,X'60',L'Data      Read Data
CCW  X'92',Search,X'60',L'Search  Read Count
```


	CCW	X'22',Sector,X'20',1	Read Sector for this count
LRArea	DC	0XL16	Area for Locate Record command
	DC	X'06010002'	Operation, auxiliary, block count (2)
Seek	DC	X'xxxxxxxx'	BBCCHH for seek
Search	DC	X'xxxxxxxx'	Search argument
Sector	DC	X'FF0000'	Sector and transfer length factor
Data	DC	----	

アプリケーション・プログラムは、EXCP を発行する前に、ラベル「Seek」と「Search」で該当する値を保管する必要があります。(「Sector」でも該当の値が保管されていると、より効率的に実行されます。)

3 番目の CCW の全部または一部を実行する場合 (Read Count)、ERP によるチャンネル・プログラムの再実行は、異なる結果を与えることとなります。

このプログラミングの例は、IBM 高水準アセンブラーが扱う大/小文字混合の使用の例を示しています。

拡張エラー情報の要求: EXCP および EXCPVR 処理が拡張エラー情報を戻すことを要求できます。この拡張エラー情報は、センス情報と完了コードで構成され、後に検出された入出力エラーをより正確に診断するのに役立ちます。拡張エラー情報は、すべての装置について入手できます。

拡張エラー情報を要求するには、以下のステップを実行する必要があります。

- DCB、IOB、および ECB の作成と初期化を含めて、127 ページの『入出力ブロック・フィールド』で説明している EXCP/EXCPVR 初期化手順に従います。
- IOSDIOBE および IOSDIEDB マッピング・マクロを使用して、共通拡張入出力ブロック (IOBE) と入出力エラー・データ・ブロック (IEDB) を定義します。それぞれをワード境界で定義する必要があります。拡張エラー情報を受け取るには、IOBE と IEDB の両方を提供する必要があります。131 ページの『共通拡張入出力ブロック・フィールド』および 134 ページの『入出力エラー・データ・ブロック・フィールド』の説明に従って、フィールドを初期化します。
- レジスター 0 を IOBE のアドレスに設定します。
- レジスター 1 を IOB のアドレスに設定します。
- IOB 内のフラグ IOBCEF をオンに設定して、拡張 IOB が存在することを示します。
- EXCP または EXCPVR マクロを発行します (それぞれ、107 ページの『EXCP での実行可能マクロ命令の使用』と 112 ページの『実ストレージ内の固定チャンネル・プログラムの実行 (EXCPVR)』を参照してください)。

IEDB には、以下の状況情報が入ります。

- 状況を提供する、入出力処理からの完了コード。完了コードとその意味は、136 ページの図 16 に示されています。

完了コードは IEDBCOD に入ります。このフィールドは入出力要求の結果を使用して更新されます。システムは、異常終了 (abnormal-end)、正常終了 (normal-end)、PCI、およびエクステント終了 (end-of-extent) 付加ルーチンを呼び出す前に、IEDBCOD フィールドを設定します。ユーザーが付加ルーチンを設定

しているかどうかに関係なく、ECB が通知された場合、システムもこのフィールドを設定します。システムは、イベントの発生に応じて複数回 IEDBCOD を設定できます。

- センス情報。装置チェックの後にのみ設定されます。設定された後、次の装置チェックによってオーバーレイされるまで、センス情報はそのまま残ります。システムはこの領域を決してクリアしないので、ユーザーは EXCP を発行する前に、それをクリアすることが必要になる場合があります。センス情報についての説明は、該当する装置の資料を参照してください。

異なるレベルの ERP 処理の要求: ユーザーはシステムに対して、選択された機能のエラー・リカバリー手順 (ERP) の処理を制限するように要求できます。これは通常、EXCP または EXCPVR 処理で入出力エラーが発生した場合です。使用可能な処理の選択は、装置タイプによって異なります。

すべての装置の場合 (カートリッジを使用する磁気テープ・サブシステムを除く)、以下の処理レベルが使用可能です。

- **ERP 処理なし。** ERP 処理は、エラー・リカバリーを試行したり、メッセージを発行したりしません。ただし、ロギング、強制ロギング・モード、およびバッファ・ログ・オーバーフローの non-error 装置チェックの場合は、ERP 処理はリカバリーを実行できます。「ERP 処理なし」を要求するには、DCB 内の DCBIFLGS フィールドの DCBIFIOE ビットの少なくとも 1 つをオンに設定します。
- **完全 ERP 処理。** ERP 処理は、エラーのロギング、装置のために制御装置によって収集されたデータのロギング、エラーの再試行、エラー・メッセージの発行、装置からの要求の処理などの機能を実行します。ユーザーは「完全 ERP 処理」を要求する必要はありません。これはシステム・デフォルトです。

カートリッジを使用する磁気テープ・サブシステム (3490 または 3590-1 など) の場合、以下の処理レベルが使用可能です。

- **基本 ERP 処理。** ERP 処理は、エラーをログに記録し、装置のために制御装置によって収集されたデータをログに記録して、装置からの要求を処理します。この場合、システムはメッセージの発行やエラーの再試行は行いません。「基本 ERP 処理」を要求するには、DCB 内の DCBIFLGS フィールドの DCBIFIOE ビットの 1 つをオンに設定します。
- **中間 ERP 処理。** ERP 処理は、基本処理で提供される機能を実行し、永続エラー・メッセージも発行します。「中間 ERP 処理」を要求するには、IOBE を定義して、IOBEERPM 内の IOBEPMSG をオンに設定します。さらに、DCB 内の DCBIFLGS フィールドの DCBIFIOE ビットの少なくとも 1 つをオンに設定します。
- **完全 ERP 処理。** ERP 処理は、エラーのロギング、装置のために制御装置によって収集されたデータのロギング、エラーの再試行、エラー・メッセージの発行、装置からの要求の処理などの機能を実行します。ユーザーは「完全 ERP 処理」を要求する必要はありません。これはシステム・デフォルトです。DCBIFIOE 内のビットは 0 でなければなりません。DCB マクロは、それらをゼロとしてアセンブルします。

VIO に関する考慮事項: ユーザーが VIO データ・セットに対して EXCP または EXCPVR を発行すると、システムはすべての共通チャンネル・コマンドをシミュレー

トします。VIO データ・セットを処理する場合、ユーザーが IEDB を提供すると、システムはその妥当性を検査しますが、現行レベルのオペレーティング・システムは、どのフィールドも設定しません。

無効な終了状況: チャンネル・プログラムの終了状況がチャンネル・エラーであった場合、チャンネル・プログラムの終了アドレスは予測不能です。通常、終了アドレスは IOBCMD31 内に保管されるか、あるいは IOBCSW の先頭の 3 バイトです。チャンネル・エラーが出るのは、サブチャンネル状況にプログラム・チェック、保護チェック、チャンネル・データ・チェック、チャンネル制御チェック、インターフェース制御チェック、またはチェーニング・チェックが示されている場合です。

EXCP での実行可能マクロ命令の使用

EXCP マクロ命令を使用しているときには、OPEN、CLOSE、場合によっては、EOV マクロ命令も使用します。EOV および EXCP マクロ命令のパラメーターについて、ここで説明します。OPEN (TYPE=J なし) および CLOSE マクロ命令のパラメーターと各種の形式については、「z/OS DFSMS Macro Instructions for Data Sets」で説明しています。TYPE=J パラメーター付きの OPEN マクロは、本書で説明します。EXCPVR マクロの説明は、112 ページの『実ストレージ内の固定チャンネル・プログラムの実行 (EXCPVR)』を参照してください。

データ・セットまたは装置の割り振り

データ・セットまたは装置の割り振りは、ステップ割り振り (「z/OS MVS JCL 解説書, SA88-8569」を参照)、または動的割り振り (「z/OS MVS Programming: Authorized Assembler Services Guide, SA22-7608」または「z/OS TSO/E コマンド解説書, SA88-8628」を参照) を使用して行います。

動的割り振りでは、XTIOT オプション (ビット S99TIOEX) を使用できます。この場合は、TIOT 内の項目の代わりに、XTIOT 項目が作成されます。XTIOT は 16 MB 境界より上に常駐します。このオプションは、ユーザーのプログラムが APF 許可、監視プログラム状態、またはシステム・キー (0 - 7) で実行されている必要があります。他の DCB アクセス方式は XTIOT をサポートしません。

動的割り振りでは、NOCAPTURE オプション (ビット S99ACUCB) を使用できます。実 UCB アドレスが 16 MB 境界より上にある場合、OPEN と EOV はそれをキャプチャーし、EOV と CLOSE はそれをキャプチャー解除します。(キャプチャーとは、16 MB 境界より上に実アドレスを持つ UCB の 24 ビット・アドレスを作成することをいいます。このオプションも TIOT 項目の代わりに XTIOT を作成しますが、プログラムは許可を取得している必要はありません。) また、S99DSABA ビットも使用できます。これは、DSAB 制御ブロックを 16 MB 境界より上に常駐させることができます。他の DCB アクセス方式は、S99DSABA をサポートしません。

データ制御ブロックの初期化

OPEN マクロ命令は、1 つ以上の DCB を初期化して、関連データ・セットを処理できるようにします。チャンネル・プログラムで使用されるすべての DCB に対して OPEN を発行します。(EXCP では、ダミー・データ・セットをオープンすることはできません。) OPEN によって実行される手順の一部を、以下に示します。

- データ・セットのアクセス許可を検査する
- データ・エクステント・ブロック (DEB) を組み立てる
- DCB および DCB 内のフィールドに記入する
- 標準ラベルを検査または作成する
- テープを位置決めする
- 付加ルーチンをロードする
- UCB をキャプチャーする (実 UCB が 16 MB 境界より上にあり、割り振りが動的であり、NOCAPTURE オプションが使用されている場合)

OPEN マクロ命令のパラメーターと各種の形式については、「z/OS DFSMS Macro Instructions for Data Sets」および 223 ページの『OPEN - JFCB 処理のためのデータ制御ブロックの初期化』で説明しています。

直接データ・セットに関する考慮事項

EXCP を使用してマルチボリューム直接データ・セット (BDAM) を処理するには、オープン・ルーチンを使用してボリュームごとにデータ・エクステント・ブロックを構築し、そのマウント・メッセージを発行します。これは、RDJFCB マクロ命令を使用して JFCB 内で読み取り、データ・セットの各ボリュームを別々の DCB を使用してオープンすることにより行います。これを実行するルーチンのコーディング方法については、211 ページの『マルチボリューム・データ・セットへのランダム入出力のための BSAM または EXCP の使用』を参照してください。RDJFCB マクロの使用法についての詳細は、204 ページの『ジョブ・ファイル制御ブロックの読み取りおよび変更 (RDJFCB マクロ)』を参照してください。

VSAM データ・セットに関する考慮事項

VSAM データ・セットのコンポーネントをオープンするのに使用される DCB では、以下のタスクを実行できます。

- アプリケーションがマスター・パスワード、またはデータ・セットに対する RACF 変更権限を持っているかどうかを検査する。
- 通常の VSAM 処理ができない場合、データ・セットの読み取りまたは書き込みを行って、データ・セットまたはカタログの損傷を修復する。有効なデータ・セットまたはカタログを損傷する可能性があるため、このインターフェースを使用してアプリケーションを書く場合は、細心の注意が必要です。

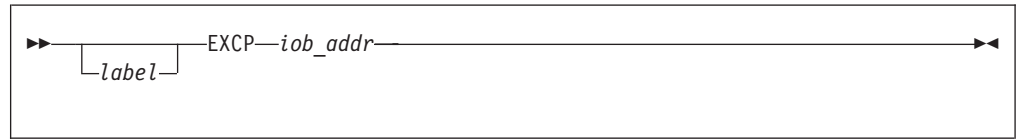
以下の条件を満たす場合、VSAM データ・セットのコンポーネント (データまたは索引) をオープンするときに DCB を指定できます。

- アプリケーションがマスター・パスワード、またはデータ・セットに対する RACF 変更権限を持っている。
- コンポーネントは複数のボリューム上に常駐してはならない。
- コンポーネントは連結のメンバーであってはならない。
- DCB は EXCP アクセス方式を指定しなければならない。
- データ・セットの処理は (OLD,KEEP,KEEP) または (SHR,KEEP,KEEP) でなければならない。
- DCB は INPUT または UPDAT オプションを指定しなければならない。
- プログラムは、APF 許可または監視プログラム状態でなければならない。

チャンネル・プログラムの実行

EXCP マクロ命令は、チャンネル・プログラム入出力操作を開始します。1 つのチャンネル・プログラムを実行するたびに、EXCP を発行します。

EXCP マクロのフォーマットは、次のとおりです。



iob_addr - **RX** タイプ・アドレス、(2-12)、または (1)

実行するチャンネル・プログラムの入出力ブロックのアドレス。

プログラムが IOBE も提供する場合は、105 ページの『拡張エラー情報の要求』で説明したようにレジスター 0 を設定してください。

ボリュームの終わり

EOV マクロ命令は、ボリューム終了 (end-of-volume) およびデータ・セット終了 (end-of-data-set) 条件を識別します。end-of-volume 条件の場合、EOV はボリュームを切り替え、標準ラベルを検査または作成します。end-of-data-set 条件の場合 (別のデータ・セットが連結されている場合を除く)、EOV はユーザーのデータ・セット終了 (end-of-data set) ルーチンに入ります。テープ入力データ・セット上のトレーラー・ラベルを処理する前に、DCBBLKCT フィールドを減分してください。プログラムは、次のいずれかの理由で EOV を発行します。

- 磁気テープまたは直接アクセス・ボリュームの切り替えが必要である。
- 出力のためにオープンされた直接アクセス・データ・セットと同じボリューム上または別のボリューム上で 2 次割り振りを実行する。
- 次の連結データ・セットへの切り替えが必要である。

磁気テープの場合、テープ・マークが読み取られたか、または書き込みコマンドが装置例外条件を受け取った場合、EOV を発行します。また、テープ・マークは読み取られず、テープ終了 (end-of-tape) にも達していなくても、次のボリュームまたはデータ・セットに進むために EOV を発行することもできます。DCB 内の 1 バイト DCBOFLGS フィールドのビット設定が、EOV が実行されたときに取られるアクションを決めます。磁気テープまたは DASD の EOV を発行する前に、DCBOFLGS に適切なビットが設定されていることを確認してください。DCBOFLGS のビット位置 2、3、6、および 7 は、システム専用です。ユーザーに関係があるのは、ビット位置 0、1、4、および 5 です。これらの DCBOFLGS ビット位置の使用は、次のとおりです。

表 31. DCBOFLGS の使用法

位置	意味
0	1 の設定は、書き込みコマンドが実行され、テープ・マークまたは DASD ファイル・マークが書き込まれることを示します。OPEN オプションが OUTPUT、OUTIN、OUTINX、または EXTEND の場合、OPEN はこのビットを 1 に設定します。その他の OPEN オプションの場合、OPEN はこれを 0 に設定します。詳しくは、123 ページの を参照してください。
1	逆方向読み取りが最後の入出力操作であったことを示します。
4	連結データ・セットは非類似として扱う必要があることを示します。詳細については、「z/OS DFSMS データ・セットの使用法」を参照してください。
5	テープ・マークが読み取られたことを示します (テープのみ)。

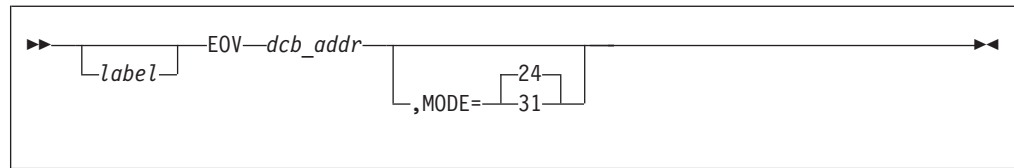
EOV が実行されたときに DCBOFLGS のビット 0 と 5 が両方ともオフの場合、新旧両方のボリューム上で、テープがテープ・マークを超えた EOV スペースと標準ラベル (ある場合) が検査されます。スペーシングの方向は、ビット 1 によって決まります。ビット 1 がオフの場合、テープは前送りされています。ビット 1 がオンの場合、テープは後送りされています。

テープの場合、ビット 0 がオンでビット 5 がオフのとき、EOV は現在位置でテープ・マークを書き込みます (現在位置は、現行ボリューム上のデータ・セットの最後のデータ・レコードの後と想定されます)。ラベル付きテープの場合、EOV はラベルも書き込みます。ラベルの処理については、「z/OS DFSMS Using Magnetic Tapes」を参照してください。DASD の場合、ビット 0 がオンのとき、EOV はファイル・マークの書き込みを試みます。詳しくは、123 ページの DCBFDAD の説明を参照してください。ラベル (指定されている場合) は、古いボリュームと新しいボリュームに作成されます。

EOV を発行すると、システムは別の場所に DEB を再ビルドする場合があります。各 EOV の後で、DCB から再ビルドされた DEB アドレスを入手してください。データ・セットが動的割り振りの nocapture オプションを使用して割り振られた場合、EOV は前ボリュームの UCB をキャプチャー解除した可能性があります。その場合、その UCB は無効になる可能性があります。新規 UCB の 24 ビット・アドレスは、前の UCB と同じ場合があることに注意してください。

直接アクセス・ボリューム上の順次編成出力データ・セットに対する EOV を発行した後、ユーザーは、追加スペースが同じボリューム上で取得されたか、異なるボリューム上で取得されたかを判別することができます。これは、DEB と UCB を調べることによって行います。UCB 内のボリューム通し番号が変更されなかった場合、追加スペースは同じボリューム上で取得されました。そうでない場合、スペースは異なるボリューム上で取得されています。

EOV マクロのフォーマットは、次のとおりです。



dcb_addr - RX タイプ・アドレス、(2-12)、または (1)

データ・セット用にオープンされる DCB のアドレス。このパラメーターが (1) として指定されている場合、レジスター 1 には、このアドレスが入っていない必要はありません。

MODE=24 または 31

レジスターの設定方法を指示します。どちらかの値を持つ場合、プログラムは 24 ビットまたは 31 ビット・モードが可能です。モードは、次のとおりです。

24 MODE オペランドを指定しない場合、このモードが想定されます。EOV マクロの展開により、DCB アドレスがレジスター 1 に保管されます。レジスター 1 の高位バイトは、EOV 処理時に無視されます。DCB は 16 MB 境界より下でなければなりません²、呼び出し側プログラムは 16 MB 境界より上にあっても構いません。

31 EOV マクロ展開コードは、DCB アドレスをレジスター 15 に入れ、レジスター 1 をゼロに設定します。DCB アドレスは 16 MB より下でなければなりません²。EOV マクロで指定されるアドレスの高位バイトはゼロでなければなりません。

EOV マクロが発行されたときにシステムがテープ・ボリュームを処理する方法を調べるには、「z/OS DFSMS Macro Instructions for Data Sets」内の OPEN マクロの DISP パラメーターの説明を参照してください。

データ制御ブロックの復元

CLOSE マクロ命令は、1 つ以上の DCB を復元して、関連データ・セットの処理を終了できるようにします。チャンネル・プログラムによって使用されたすべての DCB に対して CLOSE を発行します。CLOSE の実行時に行われる手順の一部を、以下に示します。

- データ・エクステント・ブロックを解放する
- OPEN の実行時に DCB フィールドに転送された情報を除去する
- 標準ラベルを検査または作成する
- ボリュームの処理
- プログラマー作成の付加ルーチンを解放する
- UCB をキャプチャー解除する (実 UCB が 16 MB 境界より上にあり、nocapture オプションを使用して割り振りが行われた場合)

磁気テープ・ボリューム上のデータ・セットに対して CLOSE が発行された場合、ラベルは DCB の DCBOFLGS フィールドのビット設定に従って処理されます。磁

² DCB を 16 MB より上に提供すると、ABEND50D が出されます。

気テープに対して CLOSE を発行する前に、DCBOFLGS に適切なビットを設定してください。有効な DCBOFLGS ビット位置は、EOV マクロ命令の説明文の中にリストされています。

直接アクセス装置上で出力のためにオープンされたデータ・セットに対して CLOSE が発行された場合、フィールドが 121 ページの『装置依存のパラメーター』の説明どおりであれば、システムはファイル・マークの書き込みを試みます。

CLOSE マクロ命令のパラメーターと各種の形式については、「z/OS DFSMS Macro Instructions for Data Sets」で説明しています。

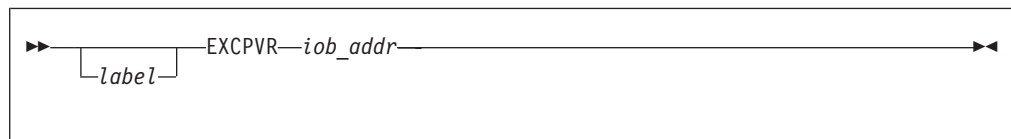
実ストレージ内の固定チャンネル・プログラムの実行 (EXCPVR)

EXCPVR マクロ命令は、EXCP マクロ命令と同じ機能 (つまり、入出力操作を実行する装置依存の手段) を提供します。さらに、プログラムは、自身の仮想チャンネル・プログラムを実チャンネル・プログラムに変換することにより、ページング環境における入出力操作の効率を改善できます。EXCPVR を発行するには、プログラムが記憶保護キー 0 から 7 で実行されているか、監視プログラム状態で実行されているか、または APF 許可が与えられている必要があります。

ユーザーは、IDAW を指す CCW を作成するか、またはフォーマット 1 CCW を使用してビット IOBEFMT1 を設定することにより、31 ビット実データ域を指定できます。このビットについては、131 ページの『共通拡張入出力ブロック・フィールド』で説明しています。

EXCPVR を発行するプログラムは、チャンネル・プログラムが完了するまで許可状態のままではなりません。プログラムに許可を与える方法についての説明は、「z/OS MVS Programming: Authorized Assembler Services Guide」を参照してください。

EXCPVR マクロのフォーマットは、次のとおりです。



iob_addr - **RX** タイプ・アドレス、(2-12)、または (1)

実行するチャンネル・プログラムの入出力ブロックのアドレス。

プログラムが IOBE も提供する場合は、105 ページの『拡張エラー情報の要求』で説明したようにレジスター 0 を設定してください。

EXCPVR を使用するには、EXCP 要求を実行するために必要な手順に従い、さらに以下にリストする手順にも従ってください。V=R アドレス・スペースでは、ステップ 1 と 2 は不要です。

1. EXCPVR 要求に関連した DCB に PGFX=YES をコーディングし、DCB で SIOA=symbol を指定して、ページ固定 (PGFX) 付加ルーチンを提供します。V=R アドレス・スペースでは、このステップは不要です。

2. ユーザーのチャンネル・プログラムを含んでいるデータ域、チャンネル・プログラムによって参照されるデータ域、PCI 付加ルーチン (ユーザーのプログラムがプログラム制御割り込みを生成できる場合)、および PCI 付加ルーチンによって参照されるすべての領域 (DEB、IOB などを含む) を固定します。これらの領域を固定するには、PGFX 付加ルーチン内に、これらの仮想領域のアドレスを含むリストを作成します。入出力の期間中すでに固定ストレージ内にあることが分かっている領域は、ページ固定リストから省略できます。

以下の 2 つの項目は、ユーザーの入出力開始 (SIO) 付加ルーチンで実行する必要があります。SIO 付加ルーチンについては、141 ページの『入出力開始 (Start-I/O) 付加ルーチン』で説明します。

3. ユーザーの CCW のアドレス・フィールドで指定された仮想ストレージのデータ域がページ境界を超えているかどうかを判別します。超えている場合は、間接データ・アドレス・リスト (IDAL) を作成し、影響を受ける CCW に IDAL のアドレスを入れて、CCW 内の IDA ビットをオンにします。詳しくは、「z/Architecture 解説書」を参照してください。
4. ユーザーの CCW 内のアドレスを仮想ストレージ・アドレスから中央ストレージ・アドレスに変換します。

EXCP/EXCPVR 入出力操作に関連したその他の領域 (つまり、CCW、IDAW、IOB、DEB、DCB、付加ルーチンなど) はすべて、24 ビット・アドレス可能のままではなければなりません。システムは、24 ビットおよび 31 ビット仮想入出力バッファのどちらも、16 MB より上の実ストレージに固定することを許可します。

制御ブロック・フィールド

ここでは、DCB のフィールド、入出力ブロック、IOBE、IEDB、イベント制御ブロック、およびデータ・エクステンション・ブロックについて、図解して説明します。DCBD マクロを使用して DCB をマップする方法については、125 ページの『DCBD マクロの DSORG パラメーター』を参照してください。

データ制御ブロック (DCB) フィールド

EXCP 形式の DCB マクロ命令は、EXCP マクロ命令で使用できる DCB を生成します。チャンネル・プログラムで処理するデータ・セットごとに DCB マクロ命令をコーディングしてください。(DCB マクロ命令の表記規則とフォーマットの図は、「z/OS DFSMS Macro Instructions for Data Sets」に記載されています。) EXCP に適用される DCB パラメーターは、生成される DCB の以下のエレメントによって決まります。

- 基礎ブロック: この部分は必要であり、常に 12 バイトの長さです。DCBMACRF フィールドが、これより前の DCB 部分が欠落したり、短いことを示している場合、システムはそれらのフィールド内の値を無視します。
- EXCP インターフェース: この部分はオプションです。このカテゴリーのパラメーターを指定すると、20 バイトが生成されます。
- 拡張基礎ブロックおよび共通インターフェース: この部分はオプションで、常に 20 バイトの長さです。この部分が生成される場合、装置依存の部分も生成されます。

- 装置依存: この部分はオプションで、拡張基礎ブロックおよび共通インターフェース部分が生成される場合にのみ生成されます。そのサイズは、DEVD パラメーターの指定に応じて、4 から 20 バイトの範囲です。ユーザーが DEVD パラメーターを指定せず、拡張基礎および共通インターフェース部分が生成される場合、この部分用の 20 バイトも生成されます。

DCB を開いたり閉じたりするときにシステムによって実行される手順の一部では (直接アクセス・ボリューム上の出力データ・セットにファイル・マークを書き込むなど)、オプションの DCB フィールドからの情報を必要とします。DCB は、システムが処理する手順のために必要な情報を提供できる十分な大きさであることを確認してください。

116 ページの図 7 は、オープンされた DCB の各部分の相対位置を表示していません。DDNAME を除いて、DCB マクロ命令パラメーターに対応したフィールドも識別されています。(DDNAME は、オープンされた DCB には含まれません。) 括弧内のフィールドは、DCB マクロ命令のパラメーターには関連のないシステム情報を表します。

DCB マクロ命令以外の DCB フィールドの情報ソースは、データ定義 (DD) ステートメントまたは動的割り振りパラメーター、データ・セット・ラベル、および DCB 変更ルーチンです。ユーザーは、これらの任意のソースを使用して、DCB パラメーターを指定できます。ただし、DCB の特定の部分が DCB マクロ命令によって生成されない場合、システムはその部分を対象とした代替ソースからの情報は受け入れません。

DCBD マクロをコーディングしてダミー制御セクション (DSECT) を生成することにより、1 つ以上の EXCP DCB のフィールドのシンボル名を提供できます。詳しくは、125 ページの『DCBD マクロの DSORG パラメーター』を参照してください。

EXCP DCB には、現行または最大ブロック・サイズを入れるフィールドがありませんが、DCBE にはそのようなフィールドがあります。

マクロ・パラメーターを持たない DCB フィールド

DCBOFLGS: 109 ページの『ボリュームの終わり』を参照してください。

DCBIFLGS: 103 ページの『実行の完了』、および 103 ページの『割り込み処理およびエラー・リカバリー手順』を参照してください。

以下は、DCB の OPEN 後に、EXCP が DCBIFLGS 内で使用するビットです。

11..	DCBIFPEC	無関係のビットがオンにされていない IOB が永続入出力エラーを受け取りました。EXCP は、ユーザーが DCBIFPEC をクリアするまで、後続の関連要求の ECB 内に X'48' を通知します。OPEN 前の DCBIFLG 内の DCBIBEC に対応します。
..11	DCBIFPCT	プリンター上でチャンネル (つまり、12) コードが検出されました。OPEN 前の DCBMACR 内の DCBIFC9 および DCBIFC12 に対応します。

.... 11..	DCBIFIOE	完全なシステム ERP (エラー・リカバリー手順) 処理より少ない要求。ユーザーによって設定されます。106 ページの『異なるレベルの ERP 処理の要求』を参照してください。
-----------	----------	--

DCBTIOT: データ・セット割り振りが動的であり、nocapture オプション S99ACUCB が指定されていた場合、このフィールドにはゼロが入ります。ユーザーは、DEBXTNP、DEBXDSAB、および DSABTIOT を使用して、TIOT 項目アドレスを入手できます。割り振りが動的でなかった場合、または nocapture オプションが指定されていなかった場合、このフィールドは TIOT 内のその項目への符号なし (16 ビット) オフセットです。

0		装置 依存	
データ制御ブロックの装置依存部分の長さフォーマットは、DSORG および DEVD パラメーターの指定によって異なります。 各装置タイプのこの部分の説明は、DEVD パラメーターの説明に含まれています。			
20	BUFNO	共通 インターフェース	
BUFCB			
24	BUFL		DSORG
28		基礎 ブロック 拡張	
IOBAD			
32	BFTEK, BFALN	EODAD	
36	RECFM	EXLST	
40		基礎 ブロック	
(TIOT)			MACRF
44	(IFLGS)		(DEB アドレス)
48	(OFLGS)	予約済み	
52	OPTCD	予約済み	
56		EXCP インターフェース	
予約済み			
60	EOEA		PCIA
64	SIOA		CENDA
68	XENDA		予約済み

図7. EXCP のデータ制御ブロックのフォーマット (OPEN 後)

括弧内のフィールドは、DCB マクロ命令のパラメーターには関連のないシステム情報を表します。

基礎ブロック・パラメーター

DDNAME=*symbol*

処理するデータ・セットを記述したデータ定義 (DD) ステートメントの名前。このパラメーターは必要であり、ユーザーのプログラムが OPEN マクロを発行する前に提供する必要があります。

MACRF=E EXCP マクロ命令をデータ・セットの処理に使用する必要がありま

す。このオペランドはコーディングする必要があり、ユーザーのプログラムが OPEN マクロを発行する前に提供する必要があります。

REPOS=Y または N

磁気テープ・ボリューム: このパラメーターは、ユーザーが正確なブロック数を保持しているかどうかをシステムに示します。(ブロック数を DCBBLKCT フィールドに保守します。) プログラムが実行中であり、装置が磁気テープでない場合、システムはこのパラメーターを無視します。

EOV および CLOSE 機能は、ブロック数を IBM 標準または ISO/ANSI 標準トレーラー・ラベルに記録できます。

入力時に、EOV および CLOSE 機能は DCB ブロック数を標準トレーラー・ラベル内のブロック数と比較して、欠落または重複しているブロックを検出できます。これは、リール・テープおよびカートリッジ・テープでサポートされます。

入力または出力時に、EOV および CLOSE 機能は DCB ブロック数をテープ・カートリッジ・サブシステムからの計算されたブロック数と比較して、欠落または重複しているブロックを検出できます。システムはこれをすべてのラベル・タイプおよびすべてのラベルなしテープに対して行います。「z/OS DFSMS Using Magnetic Tapes」を参照してください。

オープンされたリール付き (カートリッジではなく) 磁気テープ装置の場合、REPOS=Y をコーディングすると、動的装置再構成 (DDR) 機能は、ドライブに障害が起きた場合に、ボリュームを別の磁気テープ・ドライブに移動できます。カートリッジ付きの場合、ユーザーのプログラムでブロック数を保守しなくても、再始動によってこれを行うことができます。

オープンされたリール付き (カートリッジではなく) 磁気テープ装置の場合、再始動時に、テープをチェックポイントの位置に復元するためにブロック数が必要になります。カートリッジ付きの場合、ユーザーがブロック数を保守する必要はなく、再始動によりこれを行うことができます。

システムは、以下のタスクを実行します。

- Y - ブロック数は正確です。
- N - ブロック数は不正確です。

このオペランドが省略されている場合、N が想定されます。

EXCP インターフェース・パラメーター

このセクションのパラメーターを何もコーディングしない場合、DCB マクロはブロックのこの部分を作成せず、OPEN はユーザーのプログラムが設定する値を無視します。値をコーディングすると、マクロ展開に影響を与え、DCB OPEN 出口ルーチンが終了する前に、これらのフィールドを変更することができます。EXCP 付加ルーチンについて詳しくは、140 ページの『付加ルーチンをシステムに対して使用可能にする』を参照してください。

- EOEA=***symbol* ユーザーが SYS1.LPALIB または SYS1.SVCLIB に入力した EOE 付加ルーチン名の最後の 2 文字。
- PCIA=***symbol* ユーザーが SYS1.LPALIB または SYS1.SVCLIB に入力した PCI 付加ルーチン名の最後の 2 文字。
- SIOA=***symbol* ユーザーが SYS1.LPALIB または SYS1.SVCLIB に入力した SIO 付加ルーチン名の最後の 2 文字。
- PGFIX=****YES** SIO 付加ルーチンは、ページ固定 (page-fix) エントリー・ポイントを含みます。112 ページの『実ストレージ内の固定チャネル・プログラムの実行 (EXCPVR)』、および 142 ページの『ページ固定 (Page Fix) および EXCPVR 入出力開始 (Start I/O) 付加ルーチン』を参照してください。
- CENDA=***symbol*
ユーザーが SYS1.LPALIB または SYS1.SVCLIB に入力した CHE 付加ルーチン名の最後の 2 文字。
- XENDA=***symbol*
ユーザーが SYS1.LPALIB または SYS1.SVCLIB に入力した ABE 付加ルーチン名の最後の 2 文字。
- OPTCD=****Z** 磁気テープ・リールを持つ装置の場合 (入力のみ)、データ・チェックが検出されたときに縮小エラー・リカバリー手順 (4 回の読み取りのみ) が実行されることを示します。このオペランドを指定するのは、テープにエラーが含まれていることが分かっており、アプリケーションはすべてのレコードを処理する必要がない場合に限ってください。SYNAD ルーチンにエラー頻度分析を組み込むのも適切な用法です。このパラメーターを指定すると、拡張基礎ブロックも生成されます。カートリッジを使用する磁気テープ・サブシステムには、このパラメーターは適用されません。ユーザーのプログラムは、このパラメーター値をいつでも変更できます。
- IMSK=***value* この指定は、システムはシステムのエラー・ルーチンを使用しないことを示します。

拡張基礎ブロックおよび共通インターフェース・パラメーター

EXLST=*address*

ユーザーが例外条件用に作成した出口リストのアドレス。出口リストのフォーマットと目的は、「z/OS DFSMS データ・セットの使用法」に記載されています。

EODAD=*address*

入力データ・セットのデータ・セット終了 (end-of-data-set) ルーチンのアドレス。このルーチンが必要なときに利用不可であった場合、タスクは強制終了します。

これはデータを含む最後のボリュームに対して EOV マクロを発行する場合に必要であり、DCBOFLGS のビット 0 は、最後の操作が読み取られたことを示します。(EOV について詳しくは、109 ページの『ボリュームの終わり』を参照してください。) この例外は、現行データ・セットの後に別のデータ・セットが連結されている場合です。

ここで使用される EODAD アドレスは、DCB 内の 24 ビット・アドレスです。拡張 DCB (DCBE) を使用する場合は、その 31 ビット EODAD アドレスが使用されます。

DSORG=PS または **PO** または **DA** または **IS**

データ・セット編成 (以下のコードのいずれか)。各コードは、DCB の装置依存部分のフォーマットは特定のアクセス方式で生成されるフォーマットと同じであることを示しています。

コード	DCB フォーマット:
PS	QSAM または BSAM
PO	BPAM
DA	BDAM
IS	QISAM または BISAM

区分または順次 DASD データ・セットに書き込む場合、EOV または CLOSE でファイル・マークを書き込むことができます。詳しくは、121 ページの『装置依存のパラメーター』を参照してください。

EXCP DCB に対して STOW マクロを発行しないでください。ディレクトリーの内容を破壊するおそれがあります。

IOBAD=address

入出力ブロックのアドレス。このフィールドは任意の用途に使用できます。

RECFM=code

データ・セットのレコード・フォーマット。(レコード・フォーマット・コードは、「z/OS DFSMS Macro Instructions for Data Sets」に記載されています。) 後で読み取るためにデータ・セットを書き込む場合、RECFM、LRECL、および BLKSIZE を指定してデータ・セット属性を識別する必要があります。LRECL と BLKSIZE は、DD ステートメント、SVC 99 呼び出し、または JFCB でのみ指定できます。これらのフィールドは、EXCP で使用される DCB には存在しないからです。

IBM では、ユーザーがデータ・セットを作成する場合、OPEN TYPE=J を発行する前に、RDJFCB マクロを発行して JFCLRECL と JFCBLKSI を設定することをお勧めします。これにより、LRECL と BLKSIZE がデータ・セット・ラベルに組み込まれます。

RECFM パラメーターは、EXCP ルーチンでは使用されません。これは、データ・セットの読み取りまたは更新を行うアクセス方式が後で使用するために保管された情報を提供します。

以下のパラメーターはオプションです。システムは、記述されているバッファーを管理しません。

BFALN=F or D

各バッファーのワード境界位置合わせ (ワードまたはダブルワード)。

BUFL=length 各バッファの長さ (バイト)。最大長は 32760。

BUFCB=address

バッファ・プール制御ブロックのアドレス。つまり、バッファ・プール内のバッファに先行する 8 バイト・フィールド。下位ビットがオンの場合、アドレスは無効であり、ユーザーが FREEPOOL マクロを発行しても、効果はありません。FREEPOOL マクロは下位ビットをオンに設定します。DCB マクロは、共通インターフェース部分が有効の場合、下位ビットをオンに設定します。

BUFNO=number

関連データに割り当てられるバッファの数。最大数は 255。EXCP DCB の場合、OPEN はこのパラメータと、BFALN、BUFL、および BUFCB パラメータを無視します。これらのパラメータは、ユーザーのプログラムがこれらのフィールドを使用する場合、またはプログラムが GETPOOL または GETBUF マクロを発行する場合にのみ影響を与えます。

バッファ数とブロック・サイズは、データ転送速度およびオペレーティング・システムのブロック当たりのオーバーヘッドに影響を与えます。より多くのバッファを使用すると、(ブロック転送当たりの) システム・オーバーヘッドが削減され、DASD がシリンダーを選択して、要求されたレコードを回転させるまでにかかる時間 (装置待ち時間) が短縮されます。プログラムが効率的に処理できる数を超えるバッファを割り当てると、それらのバッファを含んでいる仮想ページがページアウトし、ジョブのシステム・オーバーヘッドが追加される可能性があります。また、バッファ数が多いと、データが転送されている間、ジョブのために大量の実ストレージが割り振られることにもなります。

低パフォーマンス・グループのジョブは、優先順位の高いジョブよりも頻繁にスワップアウトされる可能性があります。ジョブに対して割り振られるバッファの数は、スワップアウトが必要になるページ数に影響を与えます。

小さいブロック・サイズ (例えば、80) を持つデータ・セットにアクセスするプログラムは、30 個のバッファ (多くても 2 つの 4096 バイト・ページに収まる) を効率的に使用できます。この場合は、30 個のバッファを使用すると 5 個のバッファを使用するよりも効率的で、6 つのチャンネル・プログラムの代わりに、1 つだけのチャンネル・プログラムを使用して 30 ブロックを転送できることとなります。

3380 または 3390 上でのハーフトラック・ブロッキングのような大きなブロック化因数を持つデータ・セットの使用は、3 個か 4 個だけのバッファ (5 個以上ではなく) が指定されている場合に有効です。制約された環境では、DASD のパフォーマンスのわずかな低下やシステム命令コストの多少の増加は、ページングやスワッピングの削減によって十分に埋め合わせられることが必要です。

DCB OPEN インストール・システム出口は、EXCP DCB に対してインストール・システム基準のデフォルト・バッファ数を使用できます (OPEN インストール・システム出口について説明は、

「z/OS DFSMS データ・セットの使用法」を参照してください。
 ユーザーのプログラムは、OPEN インストール・システム出口が設定する BUFNO 値を使用できます。

装置依存のパラメーター

DCBE= DCBE= パラメーターは、装置から独立したパラメーターですが、DCBE= が指定されている場合、DCB の装置依存セクション全体が生成されるため、ここに含めてあります。DCBE= は、拡張 DCB (DCBE) が必要な場合に指定します。以下のように、オフセット 32 の 2 ビット (X'20') がオンの場合、DCB の最初のワード (オフセット +0) は DCBE を指します。

表 32. DCBE の存在を示す DCB ビット

名前	ビット
DCBH1	X'80'
DCBH0	X'04'

DCB マクロで DCBE= をコーディングすると、この 2 ビットをオンに設定します。DCBE について詳しくは、125 ページの『拡張データ制御ブロック』を参照してください。DCB マクロの DCBE= パラメーターについての詳細は、「z/OS DFSMS Macro Instructions for Data Sets」を参照してください。

DEV D=code データ・セットが常駐する装置。コードは、DCB のスペース所要量の降順にリストされています。

表 33. DCB DEV D オプション

コード	装置
DA	直接アクセス
TA	磁気テープ
PR	プリンター
PC	カード・パンチ
RD	カード・リーダー

ジョブのセットアップ時間まで特定の装置を選択したくない場合は、最大の領域を必要とする装置タイプ (つまり、DEV D=DA) を指定してください。

次の図は、DEV D パラメーターで指定された装置タイプと DSORG パラメーターで指定されたデータ・セット編成の組み合わせごとに、DCB の装置依存の部分を示しています。DEV D に加えて、装置依存パラメーターに対応するフィールドが、パラメーター名によって示されています。

連結データ・セットを処理している場合、システムは、新規データ・セットに達するたびに、DCBDEV T 内の値を該当する値に変更します。これらの値については、「z/OS DFSMS Macro Instructions for Data Sets」で説明しています。

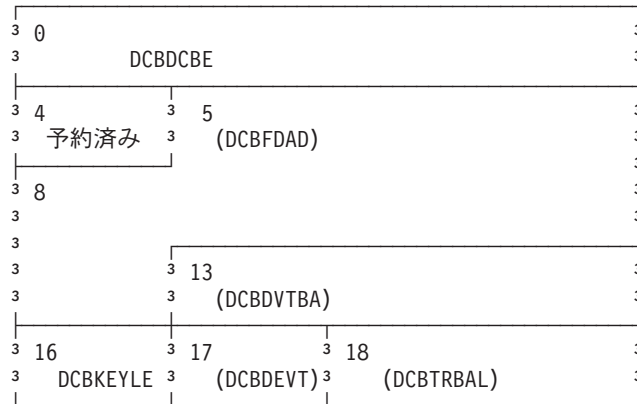


図8. DEVD=DA および DSORG=PS (または DSORG=PO) の DCB の装置依存の部分

括弧内のフィールドは、DCB マクロ命令のパラメーターには関連のない情報を表します。EOV は、これらのフィールドのすべてを設定します。OPEN は、DCBDVTBA と DCBDEVT を設定しますが、DCBFDAD は設定しません。ユーザーのプログラムは、本文で説明しているように、DCBFDAD、DCBKEYLE、または DCBTRBAL を変更できます。

DASD 上で書き込みをする場合、DCB の装置依存部分の特定フィールドを保守してください。システムはその情報を以下の場合に使用します。

- 出力データ・セット用のファイル・マークを書き込むため。
- 割り振り領域の終了時に未使用スペースを解放する場合。

ユーザーは、管理クラス、DD SPACE パラメーターの RLSE、TSO/E ALLOCATE コマンドの RELEASE パラメーター、または PARTREL マクロを使用して、部分スペース解放を要求できます。

- DISP=MOD または OPEN の EXTEND または OUTINX オプションを使用して、データ・セットに追加する場合。
- DFSMSdss がデータ・セットをコピーする場合。

DASD 上で書き込みをする場合は、DCB の装置依存部分の以下のフィールドを保守してください。

- 現行トラック上の残りのバイト数を示す符号なし 2 進数を含んでいるトラック・バランス (DCBTRBAL) フィールド。ユーザーのプログラムでは、この数値を直接計算せずに、TRKCALC マクロを使用することをお勧めします (229 ページの『トラック計算の実行』を参照してください)。計算値とアルゴリズムの両方も、装置タイプによって異なります。
- 現行レコードの位置を示す完全ディスク・アドレス (DCBFDAD) フィールド。アドレスは、MBBCCCHR 形式です。MBBCCHR の実際の形式は、151 ページの表 38 を参照してください。

出力データ・セット用のスペースが使用可能の場合、システムは DCB がクローズされる時、または EOV が発行されたときに、完全なディスク・アドレス (DCBFDAD) フィールドの内容を使用し、それに 1 を加算して、ファイル・マーク

を書き込みます。トラック・バランス (DCBTRBAL) フィールドが 8 より小さい場合、ファイル・マークは、次の順次トラックに書き込まれます。

最後のデータ・ブロックを含んでいるトラック (DCBFDAD で識別) がファイル・マークに最適な場所とはいええない場合があるので注意が必要です。以下を考慮してください。

- ファイル・マークがトラックの終わり近くにあり、将来のユーザーが DD ステートメントで DISP=MOD を使用するか、OPEN EXTEND または OUTINX オプションを使用してデータ・セットを拡張した場合、最初の新規ブロックは、次のトラック上になる可能性があります。この場合、ファイル・マークはデータの内側に残されます。
- BSAM および QSAM との互換性のために固定標準レコードを書き込む場合、すべてのブロックがフルサイズであるかのようにして、次のブロックが書き込まれた場所に必ずファイル・マークを書き込むようにする必要があります。ファイル・マークを現行トラックに「無理に押し込んで」はなりません。

システムがファイル・マークを書き込む場合、ユーザーはこの 2 バイトの内容を保守し、DCBOFLGS のビット 0 をオンに設定する必要があります。DCBOFLGS について詳しくは、109 ページの『ボリュームの終わり』を参照してください。OPEN マクロ命令を使用して、DCBDVTBA および DCBDEVT を初期化します。

TRKCALC マクロの DEVTAB パラメーターで、DCBDVTBA または DCBDVTBL を使用できます (TRKCALC の説明は、229 ページの『トラック計算の実行』を参照してください)。

16 DCBKEYLE	17 予約済み
----------------	------------

図 9. DEVD=DA および DSORG=DA の DCB の装置依存の部分

12 (DCBBLKCT)			
16 DCBTRTCH	17 DCBDEVT	18 DCBDEN	19 予約済み

括弧内のフィールドは、DCB マクロ命令のパラメーターには関連のない情報を表します。これらは、OPEN および EOVS によって設定されます。上記の REPOS パラメーターで説明したように、ユーザーのプログラムは DCBBLKCT を変更できます。

図 10. DEVD=TA および DSORG=PS の DCB の装置依存の部分

システムは、DCB がクローズされる時、または EOVS マクロ命令が発行されたときに、ブロック数 (DCBBLKCT) フィールドの内容を使用して、トレーラー・ラベルにブロック数を書き込みます。テープ・カートリッジの場合、システムはこのカウントとハードウェア情報から計算されたカウントとの比較も行います。OPEN と EOVS は、この DCB フィールドをゼロに設定します。ただし、標準ラベル・テープを逆方向に読み取る場合は例外です。その場合には、OPEN または EOVS は、DCBBLKCT をトレーラー・ラベル内のブロック数に設定します。

入出力プロセスは、それぞれの入出力要求が完了した時点で、このフィールドを IOB の IOBINCAM フィールドの内容だけ増分します。

EXCP を使用して、チェックポイントでオープンされた磁気テープ・リール・データ・セットを処理する場合、正しいカウントを保守する必要があります。そうしないと、再始動のときに、データ・セットが間違った位置に置かれる可能性があります。REPOS=Y をコーディングした場合は、動的装置再構成時の位置変更のためのカウントを保守する必要があります。DCB を変更しないでください。

16 DCBPRTSP	17 予約済み
----------------	------------

図 11. DEVD=PR および DSORG=PS の DCB の装置依存の部分

16 DCBMODE DCBSTACK	17 予約済み
---------------------------	------------

図 12. DEVD=PC または RD および DSORG=PS の DCB の装置依存の部分

以下の DCB オペランドは特定の装置に関するもので、DEVD パラメーターが指定されている場合にのみ指定できます。

KEYLEN=length

直接アクセス装置の場合、物理レコードのキーの長さ (バイト) を指定し、最大値は 255 です。ブロックが読み取りまたは書き込みされる場合、伝送されるバイトの数は、キーの長さにレコード長を加算した値になります。このパラメーターは EXCP 処理に直接影響は与えませんが、データ・セット・ラベルに保管されます。

DEN=value

磁気テープの場合、テープ記録密度 (ビット/インチ) を下の表に示します。

値	密度
2	800 (NRZI)
3	1600 (PE)
4	6250 (GCR)

NRZI-Non return-to-zero change to ones recording PE-phase encoded recording GCR-group coded recording

このパラメーターが省略されている場合、その装置で使用可能な最高密度が想定されます。DEN についての詳細は、「z/OS DFSMS Macro Instructions for Data Sets」を参照してください。

TRTCH=value

改良データ記録機能を備えた磁気テープ・サブシステムの場合、テープ記録手法は、次の値で構成されます。

値	テープ記録手法
COMP	データを圧縮形式で記録します。
NOCOMP	データを標準の非圧縮形式で記録します。

7トラック磁気テープの場合、テープ記録手法は、次のとおりです。

値	テープ記録手法
C	データ変換機能が使用可能です。
E	偶数パリティが使用されます。(省略されている場合、奇数パリティが想定されます。)
T	BCDIC から EBCDIC への変換が必要です。

MODE=value カード・リーダーまたはパンチの場合、操作のモード。C (カラム・バイナリー・モード) または E (EBCDIC コード) のいずれかを指定できます。このフィールドとパラメーターは EXCP 処理には直接影響を与えませんが、ユーザーのプログラムはこのフィールドを使用できます。これは、DD ステートメントで値を指定できるので便利です。

STACK=value カード・パンチまたはカード・リーダーの場合、カードを受け取るスタッカー・ピンを指定し、1 または 2 のいずれかの値です。このフィールドとパラメーターは EXCP 処理には直接影響を与えませんが、ユーザーのプログラムはこのフィールドを使用できます。これは、DD ステートメントで値を指定できるので便利です。

PRTSP=value プリンターの場合、行送りは 0 から 3 の値です。このフィールドとパラメーターは EXCP 処理には直接影響を与えませんが、ユーザーのプログラムはこのフィールドを使用できます。これは、DD ステートメントで値を指定できるので便利です。

DCBD マクロの DSORG パラメーター

DCBD マクロの DSORG パラメーターに対しては、「z/OS DFSMS Macro Instructions for Data Sets」に記載されているオペランドに加えて、以下のオペランドを指定できます。

DSORG=XA または **XE**

マップする DCB のセクションを指定します。

XA 基礎ブロックと EXCP インターフェースを備えた DCB を指定します。

XE 共通インターフェース、拡張基礎ブロック、および基礎ブロックを備えた DCB を指定します。

DCB の 4 つのセクションをすべてマップするには、DSORG=(XA,XE) をコーディングします。

拡張データ制御ブロック

拡張データ制御ブロック (DCBE) は、追加の処理オプションを提供します。EXCP は次のオプションをサポートします。

- EOVS マクロを発行するときの EODAD パラメーター。

- OPEN および EOV マクロを発行するときの BLKSIZE パラメーター。
- OPEN、EOV、または CLOSE マクロを発行するときに、IBM 3590 上のバッファ・テープ・マークを制御するための SYNC パラメーター。
- OPEN マクロを発行するときに、IBM 3490 をエミュレートする IBM 3590 磁気テープ・サブシステム上に追加データを書き込むための CAPACITYMODE パラメーター。
- ご使用のプログラムでラージ・フォーマット属性を持つ DASD データ・セットが処理可能であることを示す BLOCKTOKENSIZE パラメーター。データ・セットは必ずしも大規模である必要はありません。BLOCKTOKENSIZE=LARGE をコーディングする場合に、プログラムが 65535 トラックを超えるデータ・セット、またはそれを越えて拡張する可能性があるデータ・セットを処理できることを意味します。これらの違いは、次のとおりです。
 - データ・セットがラージ・フォーマット・データ・セットの場合、DSCB において DS1LARGE ビットがオンになり、関連トラック番号を含む DS1LSTAR 内の 2 バイトは、論理的に DS1TTTHI バイトが加えられて拡張されます。1 ページの『第 1 章 ボリューム目録の使用』に記載されている format 1 DSCB についての説明を参照してください。
 - 各 DEBNMTRK フィールド内の 2 バイトは、DEBNMTRKHI バイトによって論理的に拡張されます。427 ページの『付録 B. 制御ブロック』を参照してください。
 - プログラムが CVTPRLTV または CVTPCNVT がポイントするトラック変換ルーチンのいずれかを呼び出した場合、プログラムは、+0 エントリー・ポイントの代わりに +12 エントリー・ポイントを使用します。

DCBE マクロで BLOCKTOKENSIZE=LARGE をコーディングしない場合は、次のようになります。

- OPEN マクロ・オプションが INPUT ではなく、また、データ・セットがラージ・フォーマットである場合は、OPEN によって 213-14 ABEND が出されません。
- OPEN マクロ・オプションが INPUT で、かつ、データ・セットがボリューム上に 65535 を超えるトラックを持っている場合は、OPEN により 213-16 ABEND が出されます。

ユーザー・プログラムが 24 ビット・モードで実行されていても、DCBE は 16 MB 境界より上に置けることに注意してください。DCBE に関して詳しくは、「z/OS DFSMS Macro Instructions for Data Sets」を参照してください。

データ・セットのブロック・サイズの設定と検索

DCBE には、データ・セットの現行および最大ブロック・サイズを入れるフィールドがあります。

このフィールドを使用するには、プログラムは、DCBE マクロで BLKSIZE キーワードをコーディングするか、または DCBEULBI ビットをオンにする必要があります。キーワードの値は、数値または再配置不可シンボリック式を使用できます。値は 0 でも構いません。

ゼロの値をコーディングするか、DCBEULBI をオンにした場合、OPEN は、DISP=MOD (ディスクまたはテープ) を使用して入力または出力用にオープンした場

合、DCBEBLKSI にデータ・セット・ラベルからのブロック・サイズ値を保管します。プログラムが OUTPUT または OUTIN オプションを使用して OPEN マクロを発行した場合、OPEN は、BSAM および QSAM の場合と同様に、BLKSIZE の最適値の計算を試みます。システムが決定するブロック・サイズ機能については、「z/OS DFSMS データ・セットの使用法」で説明しています。基本原理は、ユーザーのプログラムが LRECL 値と RECFM 値 (U ではなく) を提供し、OPEN が装置用の最適な BLKSIZE 値を計算して、それを DCBEBLKSI に保管することです。

注意: OPEN が算定した DCBE 内の BLKSIZE 値は、通常のアクセス方式またはユーザー・プログラムのデータ・セットを読み取る他のプログラムによってサポートされる最大値を超えることがあります。例えば、磁気テープ上では、おそらく値は 32760 を超過します。これが問題になる場合は、次のようにできます。

「z/OS DFSMS データ・セットの使用法」で説明しているように、DCB OPEN 出口ルーチンを提供します。出口ルーチンが、DCBEBLKSI フィールドがまだ 0 であることを検出した場合、データ・セット・ラベルおよび DD ステートメントは BLKSIZE の値を持っていないことを意味しています。ユーザーのプログラムが出力用にオープンしている場合は、DCBE だけを残して、OPEN に最適な BLKSIZE 値を計算させるか、またはプログラムが最適値を計算できます。これは、システムが BLKSIZE 値をさまざまなシステム制御ブロックやデータ・セット・ラベルに保管する前に、ユーザーのプログラムがそれを設定できる最後のチャンスです。DCB OPEN 出口ルーチンは、INFO=AMCAP パラメーターを指定した DEVTYPE マクロを発行して、その装置の BLKSIZE の最適値と最大値を確認できます。値が大きすぎる場合、プログラムはより小さい有効な値を計算できます。OPEN が値を設定する前に、値を DCBEBLKSI に保管してください。DEVTYPE の INFO=AMCAP パラメーターについては、193 ページの『DEVTYPE - INFO 形式』で説明します。

ユーザーのプログラムがこれを行う必要がある理由

- プログラムの装置タイプへの依存度を軽減する。
- システムがこの値をデータ・セット・ラベルに保管して、z/OS およびその他のシステムで使用できるようにする。
- システムがこの値をさまざまな SMF レコードに保管して、システム・リソースのモニターを改善する。
- DFSMSrmm™ は、テープ・レポート用にこの情報を保持する。

通常のアクセス方式とは異なり、EXCP 処理は、EXCP DCBE 内の BLKSIZE 値の完全な検査は行いません。

入出力ブロック・フィールド

入出力ブロック (IOB) は、マクロ命令によって自動的に構成されません。一連の定数として定義し、ワード境界に置く必要があります。ユニット・レコード装置および磁気テープ装置の場合、IOB は 32 バイトの長さです。直接アクセス装置、テレプロセッシング装置、およびグラフィック装置の場合、追加の 8 バイトを提供する必要があります。システム・マッピング・マクロ IEZIOB を使用すると (これは

DSECT に展開されます)、IOB を構成するのに役立ちます。ここに記述されていない IEZIOB フィールドは、プログラミング・インターフェースの一部ではありません。

図 13 で、陰影付きの領域は、ユーザーが情報を指定する必要があるフィールドを示しています。その他のフィールドはシステムによって使用され、すべてゼロとして定義しなければなりません。ユーザーはこれらのフィールドに情報を入れることはできませんが、その内容を調べることはできます。

以下の IOB フィールドは、システム自体が値を設定するので、ユーザーは EXCP を発行する前に特定の値に設定する必要はありません。

- IOBSENS0
- IOBSENS1
- IOBECBCC
- IOBCSW
- IOBSIOCC

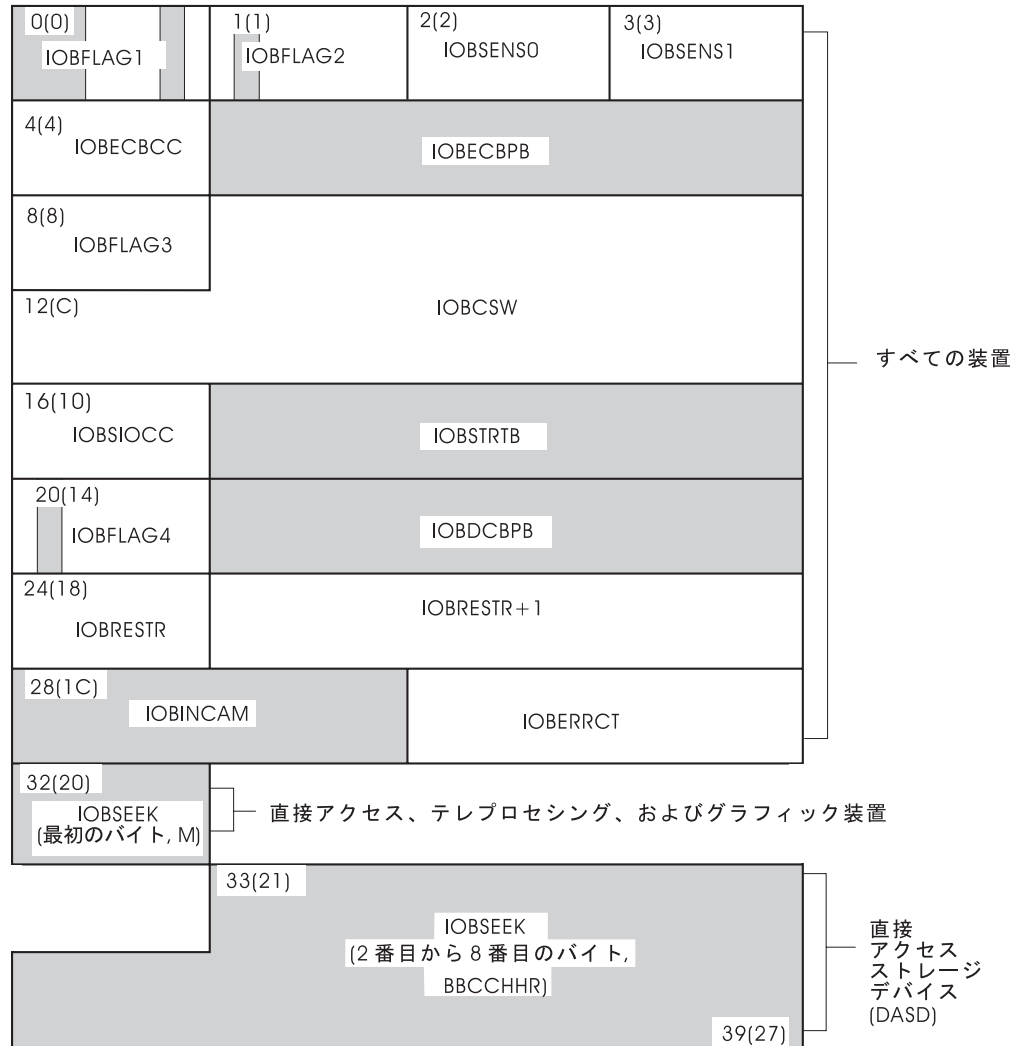


図 13. 入出力ブロック (IOB) フォーマット

IOBFLAG1 (1 バイト)

ビット位置 0、1、6、および 7 を設定します。位置 0 と 1 (IOBDATCH と IOBCMDCH) 内の 1 ビットは、それぞれデータ・チェーニングとコマンド・チェーニングを示します。(データ・チェーニングとコマンド・チェーニングの両方を指定した場合、直接アクセス装置と磁気テープ装置を除いて、システムはエラー・リカバリー・ルーチンを使用しません。) チャンネル・プログラムの実行中に入出力エラーが発生した場合、CCW 内のチェーニング・ビットに対応したチェーニング・ビットが IOB に設定されていないと、成功するエラー・リカバリーも不可能になることがあります。ユーザーのデータの保全性が損なわれる恐れがあります。

位置 6 (IOBUNREL) 内の 1 ビットは、チャンネル・プログラムは関連要求ではないこと、つまりそのチャンネル・プログラムは他のチャンネル・プログラムに関連していないことを示します。下記の IOBFLAG2 のビット 2 と 3 を参照してください。

BSAM、QSAM、または BPAM DCB を含む EXCP (IOBSPSVC) マクロを発行する意図がある場合は、ビット 7 (IOBSPSVC) をオンにして、アクセス方式 (access-method) 付加ルーチンが入出力要求を処理するのを防止する必要があります。

IOBFLAG2 (1 バイト)

IOBFLAG1 フィールドのビット 6 をゼロに設定した場合、このフィールドのビット 2 と 3 (IOBRRT3 と IOBRRT2) は、以下のいずれかに設定する必要があります。

- 00 (関連要求に関連した任意のチャンネル・プログラムまたは付加ルーチンがこの IOB またはチャンネル・プログラムを変更できる場合)。
- 01 (00 設定を必要とする条件は適用されないが、CHE または ABE 付加ルーチンが正常に完了した場合、あるいは CSW 内の装置例外 (unit-exception) または誤長レコード (wrong-length-record) ビットをオンに設定して完了した場合には、このチャンネル・プログラムを再試行できる場合)。
- 10 (その他のすべての場合)。

ビット 2 と 3 の組み合わせは関連要求を表し、タイプ 1 (00)、タイプ 2 (01)、およびタイプ 3 (10) と呼ばれます。ユーザーが使用するタイプによって、EXCP プロセッサが関連要求の処理をオーバーラップできる度合いが決まります。タイプ 3 は最大のオーバーラップを許容し、通常はチャンネル終了 (channel-end) 割り込み後にすぐに装置が再利用できるようになります。(MVS 以前のシステム上で実行された関連要求は、変更されていない場合、タイプ 1 要求として実行されます。)

IOBSENS0 および IOBSENS1 (2 バイト)

装置チェックが発生した場合、EXCP プロセッサによって設定されます。これは最初の 2 つのセンス・バイトです。時折、センス・コマンドが発行されたときに、装置チェックが原因でシステムがセ

ンス・バイトを入手できないことがあります。この場合、システムは X'10FE' を IOBSENS0 と IOBSENS1 に移動してセンス・バイトをシミュレートします。

2 バイトより多くのセンス・バイトを取得したい場合は、103 ページの『割り込み処理およびエラー・リカバリー手順』で説明しているように、IOBE と IEDB を提供してください。

IOBECBCC (1 バイト)

チャンネル・プログラムの完了コードの最初のバイト。チャンネル・プログラムが完了を通知すると、システムはこのコードをイベント制御ブロックの高位バイトに入れます。完了コードとその意味は、136 ページの『イベント制御ブロック・フィールド』のもとにリストされています。

IOBECBPT (3 バイト)

ユーザーが提供した 4 バイトのイベント制御ブロック (ECB) のアドレス。

IOBFLAG3 (1 バイト)

システムによって使用されます。チャンネル・プログラムがフォーマット 1 CCW の場合、IOBFLAG3 のフィールドは次のとおりです。IOBCMD31 は 4 バイトで、チャンネル・プログラムの終了アドレスが入っています。IOBCSW8 は 8 バイトで、IOBCSW のようなエミュレートされた CSW を含みます。

IOBCSW (7 バイト)

チャンネル終了 (channel-end) または PCI 割り込みが発生するたびにこのフィールドに入れられる、シミュレートされたチャンネル状況ワードの下位 7 バイト。最初の 3 バイトは、ユーザーのチャンネル・プログラム内の最後に実行された (last-executed) CCW の後を指す仮想アドレスです。その次の 2 バイトは、状況バイトです。最後の 2 バイトは残余カウントで、これは最後の CCW で転送されなかったバイト数を表します。

IOBSIOCC (1 バイト)

ビット 0 と 1 には、命令長コードが入ります。ビット 2 と 3 には、システムがチャンネル・プログラムを開始するために発行する命令のサブチャンネル開始 (SSCH) 条件コードが入ります。ビット 4 から 7 には、プログラム・マスクが入ります。

IOBSTART (3 バイト)

実行するチャンネル・プログラムの開始アドレス。

IOBFLAG4 (1 バイト)

ビット 3 (IOBCEF) を設定して、ユーザーが共通拡張 IOB を提供しているかどうかを示します。このビットが 1 の場合、ユーザーが EXCP を発行すると、レジスター 0 に IOBE アドレスが入ります。105 ページの『拡張エラー情報の要求』および 106 ページの『異なるレベルの ERP 処理の要求』を参照してください。

IOBDCBPT (3 バイト)

チャンネル・プログラムによって読み取りまたは書き込みするデータ・セットの DCB のアドレス。

予約済み (1 バイト)

システムによって使用されます。

IOBRESTR+1 (3 バイト)

関連チャンネル・プログラムが永続エラーの場合、このフィールドを使用して、従属チャンネル・プログラムを表す IOB を相互にチェーニングします。チェーンを構築する条件についての詳細は、224 ページの『入出力要求のパーズと復元』を参照してください。

IOBINCAM (2 バイト)

磁気テープの場合、システムが DCB の装置依存部分のブロック数 (DCBBLKCT) フィールドを増分する量。このバイトはいつでも変更できます。順方向操作の場合、このバイトには 2 進の正整数 (通常は、+1) が入ります。逆方向操作の場合は、2 進の負の整数が入ります。このバイトを使用しない場合は、すべてゼロを指定する必要があります。123 ページの図 10 を参照してください。

IOBERRCT (2 バイト)

システムによって使用されます。

IOBSEEK (最初のバイト、M)

直接アクセス装置の場合、チャンネル・プログラムに関連したデータ・エクステント・ブロック内のエクステント項目 (0 は最初の項目を示し、1 は 2 番目の項目を示す、という具合で進みます)。テレプロセッシングおよびグラフィック装置の場合、ここには UCB 索引が入ります。

IOBSEEK (最後の 7 バイト、BBCCHHR)

直接アクセス装置の場合、ユーザーのチャンネル・プログラムのシーク・アドレス。

共通拡張入出力ブロック・フィールド

拡張エラー情報を受け取ったり、エラー・リカバリー手順 (ERP) 処理のレベルを制御するために、共通拡張 IOB (IOBE) ブロックを構成できます。IOBE を提供するには、ユーザーが EXCP または EXCPVR を発行するときに、IOBFLAG4 内の IOBCEF をオンに設定し、レジスター 0 に IOBE アドレスを設定します。(IOBE は IOSDIOBE マクロによってマップされます。)

予約済みバイトは X'00' に設定してください。IOBE は、16 MB より上または下の仮想ストレージに常駐できます。この制御ブロックの図は、132 ページの図 14 を参照してください。

0 IOBEID			
4 IEDBVERS	5 IOBEFLG1	6 IOBEFLG2	7 IOBEERPM
8-11 IOBEUSER, IOBEUPTR			
12-15 IOBEIEDB			
16 IOBEFLG3	17-47 予約済み		

図 14. IOBE のフォーマット (IOSDIOBE マクロによってマップされます)

表 34. IOBE 構造のマッピング:

オフセット	長さ、 または ビット・ パターン	名前	説明
0 (X'0')	4	IOBEID	目印フィールド (「IOBE」)
4 (X'4')	1	IOBEVERS	バージョン番号 (X'01')
5 (X'5')	1	IOBEFLG1	予約済み。

表 34. IOBE 構造のマッピング (続き):

オフセット	長さ、 または ビット・ パターン	名前	説明
6 (X'6')	1	IOBEFLG2	フラグ・フィールド 2。EXCP または EXCPVR の発行者によって設定されま す。
x...			予約済み。
.1.		IOBEP	ユーザーは CCW、IDAW、およびデータの事前取り出しを無制限に許可しま す。ゼロの場合、事前取り出しは出力時のデータ・チェーニングに対してのみ 許可されます。この場合、データ域を記述する 1 つの CCW が許可されま す。IDAW およびデータの事前取り出しは、FICON™ チャンネルにのみ適用さ れます。Enterprise Systems Connection (ESCON) または並列チャンネルに対 しては影響を与えません。このビットは EXCPVR に適用されます。EXCP はこの ビットをオンと想定します。
..1.		IOBECPNM	ユーザーは実行中にチャンネル・プログラムが変更されないことを保証しま す (最後に CCW を追加することを除いて)。APF 許可アドレス・スペースの EXCPVR に対してのみ影響を与えます。それ以外は、現行リリースには影響 を与えません。
...1		IOBEEIDA	IDAW はそれぞれ 8 バイトで、4096 バイト境界を指します。ただし、各リス トの最初の IDAW は例外の可能性があります。現在は、直接アクセス・スト レージ・デバイス (DASD) と IBM 提供のカートリッジ磁気テープ装置でのみ サポートされます。UCB 内のビット UCBEIDAW は、その装置が 64 ビット IDAW をサポートするかどうかを示します。このビットは、システムが z/Architecture モードで実行されている場合にのみオンです。EXCP と EXCPVR でサポートされます。フォーマット 1 CCW および仮想チャンネル・ プログラムでサポートされます。現行では、VIO ではサポートされません。
.... 1...		IOBEPCIS	PCI 同期。PCI に続く次の CCW の後で (CCW+8 で)、チャンネルを同期する必 要があります。EXCP では常にサポートされます。EXCPVR では、システム が z/Architecture モードで実行され、かつ IOBEP がオンのときのみサポ ートされます。VIO ではサポートされません。
.... .1..		IOBNORWS	読み取り/書き込み同期なし。チャンネルは読み取り/書き込みの遷移時に同期し てはなりません。ユーザーは、読み取りと書き込みが異なる領域から行われ ることを保証しています。EXCP では常にサポートされます。EXCPVR では、 IOBEP がオンの場合にのみサポートされます。
.... ..1.		IOB2CSWS	2 つのチャンネル状況ワード。チャンネルの前に実行中の制御装置でエラーが発 生した場合、システムは 2 つの終了 CCW アドレスを戻します。2 番目の終了 CCW アドレスは IEDB に入ります。EXCP では常に許可されます。EXCPVR では、IOBEP がオンの場合にのみサポートされます。チャンネルの前に実行中 の制御装置でエラーがあったときにこのビットがオフの場合、システムは無効終 了 CCW アドレスをシミュレートします。
.... ...1		IOBEFMT1	チャンネル・プログラムはフォーマット 1 CCW で、IOBSTART に 31 ビッ ト・アドレスが入ります。このビットにより、システムは IOBSIOCC の代わ りに IOBESIOC を使用し、終了 CCW アドレスは IOBCMDA 内の 3 バイト の代わりに IOBCMD31 を使用します。EXCPVR に有効です。z/OS V1R6 か らは EXCP にも有効です。

表 34. IOBE 構造のマッピング (続き) :

オフセット	長さ、 または ビット・ パターン	名前	説明
7 (X'7')	1	IOBEERPM	ERP の機能の実行が許可されることを示すマスク。
	1.....	IOBEPMSG	ユーザーは、システムまたはオペレーターとの対話を必要としない、基本エラー・リカバリー手順 (ERP) リカバリーと永続エラー・メッセージを許可します。106 ページの『異なるレベルの ERP 処理の要求』を参照してください。
	.xxx		予約済み。
	xxxx		
8 (X'8')	4	IOBEUSER	ユーザーが使用するアドレス・フィールド。
	4	IOBEUPTR	ユーザーが使用する文字フィールド。
12 (X'C')	4	IOBEIEDB	ゼロ、または IEDB のアドレス (拡張エラー情報を必要とする場合)。詳しくは、105 ページの『拡張エラー情報の要求』を参照してください。
16 (X'10')	1	IOBEFLG3	フラグ・バイト 3。EXCP または EXCPVR の発行者によって設定されます。
	1... ..	IOBENSER	装置がチャンネル・プログラム・エクステント衝突検査をバイパスすることを許可するために、ユーザーによって設定されます。エクステント範囲適用 (extent range enforcement) はアクティブのままです。DASD がそれをサポートする場合にのみ意味があります。マルチシステム・アクセスまたは複数並列アクセス・ボリューム (PAV) を使用している場合、IBM 2105 Enterprise Storage Server™ のパフォーマンスが向上することがあります。
	.1.	IOBENVAL	define extent および locate record コマンド時に、装置がパラメーターの妥当性検査をバイパスすることを許可するために、ユーザーによって設定されます。エクステント適用 (extent enforcement) はアクティブのままです。DASD 装置がそれをサポートする場合にのみ意味があります。それ以外の場合は、影響を与えません。マルチシステム・アクセスまたは複数並列アクセス・ボリューム (PAV) を使用している場合、IBM 2105 Enterprise Storage Server のパフォーマンスが向上することがあります。
	..1.	IOBEDSMC	ストリーム・モード制御を使用不可にするために、ユーザーによって設定されます。
	...1	IOBEIOT	ユーザーがこれを 0 に設定した場合、要求がアクティブの間のみ、IOBETIME が要求に適用されます。これが 1 の場合、要求がキューに入れられているかアクティブの間、IOBETIME が適用されます。
 xxxx		予約済み。
17 (X'11')	1	IOBESIOC	IOBSIOCC と同じフォーマット。IOBEFMT1 がオンの場合にのみ有効です。
18 (X'12')	1	IOBETIME	制限時間。この値がゼロ以外の場合、ユーザーが要求に費やすことを許容する秒数を示します。DEBACCS フィールドで DCB が入力用にオープンされていることが示されている場合にのみ、影響を与えます。制限時間に達しても、メッセージを出したり、エラーを記録することはありません。ビット IOBEIOT は、時間の測定対象を指定します。
19 (X'13')	291		予約済み。ゼロでなければなりません。

入出力エラー・データ・ブロック・フィールド

システムは IEDB を使用して拡張エラー情報を提供します。ユーザーは IOBE にそのアドレスを設定することによって、IEDB を提供します。予約フィールドは X'00'

に設定してください。システムはセンス・バイトを IEDBSNS に移動します。使用可能なセンス・バイトが 32 バイトより少ない場合、残余データが生じることがあります。IEDB は 16 MB より下の仮想ストレージに常駐する必要があります。IEDB の使用について詳しくは、105 ページの『拡張エラー情報の要求』を参照してください。この制御ブロックの図は、図 15 を参照してください。

0			
IEDBID			
4	5	6	7
IEDBVERS	IEDBFLG1	IEDBCOD	予約済み
8-39			
IEDBSNS			
40-43			
予約済み			
44-47			
IEDB2CSW			

図 15. IEDB のフォーマット (IOSDIEDB マクロによってマップされます)

表 35. IBDB 構造のマッピング:

オフセット	長さ、または ビット・パターン	名前	説明
0 (X'0')	4	IEDBID	目印フィールド。「IEDB」でなければなりません。
4 (X'4')	1	IEDBVERS	バージョン。現在は 1 でなければなりません。マクロは、現行では IEDBVRSC をこの値として定義します。
5 (X'5')	1	IEDBFLG1	フラグ・フィールド。
	1... ..		センス・データが無効で、X'10FE' で始まっています。
	.xxx xxxx		予約済み。
6 (X'6')	1	IEDBCOD	EXCP または EXCPVR が変更する前の元の入出力完了コード。
7 (X'7')	1		予約済み。
8 (X'8')	32	IEDBSNS	センス・バイト。
	1	IEDBSNS00	センス・バイト 0。
	1	IEDBSNS01	センス・バイト 1。
(フィールド IEDBSNS02 から IEDBSNS31 は、センス・バイト 2 から 31 を定義します。)			
40(X'28')	4		予約済み。

- 41 永続入出力エラー
- 42 エクステンツ・エラー (DASD のみ)。IOBSEEK が DEB によって記述されたエクステンツの外側にあるか、またはチャンネル・プログラムが現行エクステンツの外側で実行を試みたかのいずれかです。
- 44 装置に対する前の入出力要求が完了を通知した後で、エラーが発生しました。付加ルーチンおよび (実行が許可された場合) ERP は、これは永続エラーであると判別しました。永続エラーにより、入出力要求は終了しました。CSW の内容と IOB 内のセンス・データは、試行された操作には適用されません。これらは、装置上のデータ・セットに対して試行された直前の操作に適用されます。ユーザーは、EXCP マクロ命令を再発行してチャンネル・プログラムを再始動することができます。
- 45 入出力要求の処理中に IOS でプログラム・チェックまたはマシン・チェックが発生しました。
- 48 チャンネル・プログラムはページされました。
- 4B テープ ERP によって要求されたテープ位置変更中にエラーが発生しました。
- 4F 直接アクセス・エラーの後でエラー・リカバリー・ルーチンに入りましたが、ホーム・アドレスまたはレコード 0 を読み取ることができません。
- 51 シミュレートされたエラー状況。付加ルーチンおよび (実行が許可された場合) ERP は、これは永続エラーであると判別しました。永続エラーにより、入出力要求は終了しました。このコードは、装置が永続エラー状態、ボックス処理、または非接続であることを示します。また、このコードは、未着割り込みが検出され、未着割り込みハンドラーによるリカバリー操作の結果として入出力操作が終了したことを示す場合もあります。
- 74 シミュレートされたエラー状況。このコードは、永続エラー状態、ボックス処理、または非接続の装置に対して入出力操作を開始しようとした結果として設定されます。また、このコードは、未着割り込みが検出され、未着割り込みハンドラーによるリカバリー操作の結果として入出力操作が終了した場合にも設定されます。システムは、ERP (実行が許可されている場合) および付加ルーチン呼び出しで、エラーが永続的であるか訂正可能であるかを判別するために、このコードを一時的に IOBECBCC および IEDB に設定します。エラーが永続であると判別された場合、システムは値を 51 に変更します。システムは ECB には 74 コードを設定しません。
- 7F 正常な入出力完了。ただし、ECB には表示されません。

データ・エクステント・ブロック・フィールド

DCB に対して OPEN マクロ命令が発行されると、システムによってデータ・エクステント・ブロックが構成されます。ユーザーは DEB のフィールドを変更することはできませんが、その内容を調べることはできます。DEB は IEZDEB マクロによってマップされます。EXCP および EXCPVR で使用される DEB フィールドは、427 ページの『付録 B. 制御ブロック』に示されています (すべての DEB フィールドについては、「z/OS DFSMSdfp 診断解説書」で解説しています)。

EXCP および EXCPVR の付加ルーチン

付加ルーチンとは、入出力操作に対する追加制御を提供するルーチンです。付加ルーチンを使用すると、入出力操作の状況を調べたり、さまざまな条件下で取るべきアクションを決めたりすることができます。付加ルーチンは、表 36 に示すように、制御を受け取ることができます。

表 36. EXCP 付加ルーチン

付加ルーチン	説明	呼び出される時期
ABE	異常終了	異常条件
CHE	チャンネル終了	チャンネル終了、装置例外、誤長レコード
EOE	エクステント終了	入出力ブロック内のアドレスが割り振りエクステント限界の外側にあるとき。
PCI	プログラム制御割り込み	チャンネル・プログラム内で 1 つ以上の PCI ビットがオンのとき。
PGFX	ページ固定	EXCPVR 要求のための SIO の前
SIO	入出力開始	チャンネル・プログラムの変換の直前

付加ルーチンは、監視プログラム状態で制御を取得し、下表に記述された EXCP プロセッサからポインターを受け取ります。付加ルーチンは、24 ビット・アドレスリング・モードで制御を受け取り、同じモードでそれを戻す必要があります。

レジスター 内容

- 1 要求キュー・エレメントを指します。
- 2 入出力ブロックを指します。
- 3 データ・エクステント・ブロックを指します。
- 4 データ制御ブロックを指します。
- 6 シーク・アドレス (MBBCCHHR) を指します (制御がエクステント終了 (end-of-extent) 付加ルーチンに与えられた場合。)
- 7 装置制御ブロックを指します。現行レベルのシステムでは、これは常に 16 MB 境界より下にあります。動的割り振りの nocapture オプションを使用しており、実 UCB が 16 MB 境界より上にある場合、OPEN または EOVS は、EOVS または CLOSE まで UCB をキャプチャーします。

- 13 入力レジスターまたはデータを保管するのに使用できる 16 ワード域を指します。
- 14 付加ルーチンの実行後に制御を戻すシステム内の位置を指します。システムに制御を戻す際には、レジスター 14 内の戻りアドレスからの変位を使用できます。許容される変位については、表 37 に要約し、詳しい説明は、それぞれの付加ルーチンごとに後述します。
- 15 付加ルーチンのエントリー・ポイントを指します。PGFX 付加ルーチンに入る場合、SIO エントリー・ポイントを指します。

付加ルーチンによって行われる処理には、以下の要件および制約が適用されます。

- レジスター 9 (使用される場合) は、システムに制御を戻す前に 2 進ゼロに設定しなければなりません。その他のレジスターはすべて、それを使用した場合は保管および復元する必要があります (ただし、付加ルーチンの説明の項で示されているものを除きます)。表 37 は、レジスター規則を要約しています。
- SVC 命令またはシステムの状態を変更する命令 (例えば、WTO、LPSW、または同様の特権命令) を発行することはできません。
- 入出力操作の完了をテストするループは使用できません。

以下では、付加ルーチンのタイプを示し、どのような場合に作成されるのか、システムに制御を戻す方法、および内容を保管および復元せずに使用できるレジスターについて説明します。ユーザーが特定の付加ルーチンを提供しない場合、または付加ルーチンを何も提供しない場合、システムは、その付加ルーチンがレジスター 14 からのオフセット 0 で戻ったものとして動作します。

表 37. 付加ルーチンのエントリー・ポイント、戻り、および使用可能な作業レジスター

付加ルーチン	エントリー・ポイント	戻り	使用可能な作業レジスター
EOE	レジスター 15	レジスター 14 + ABE 呼び出しス 0 レジスター 14 キップ、再試行 + 4 レジスター 14 + 8	レジスター 10、11、12、お よび 13
SIO	レジスター 15	レジスター 14 + 通常スキップ 0 レジスター 14	レジスター 10、11、および 13
PCI	レジスター 15	レジスター 14 + 通常 0	レジスター 10、11、12、お よび 13
PGFIX	レジスター 15 + 4	レジスター 14 + 通常 0	レジスター 10、11、および 13
CHE	レジスター 15	レジスター 14 + 通常スキップ、 0 レジスター 14 再 EXCP パイパ + 4 レジスター ス 14 + 8 レジスタ ー 14 + 12	レジスター 10、11、12、お よび 13

表 37. 付加ルーチンのエントリー・ポイント、戻り、および使用可能な作業レジスター (続き)

付加ルーチン	エントリー・ポイント	戻り	使用可能な作業レジスター
ABE	レジスター 15	レジスター 14 + 通常スキップ、 0 レジスター 14 再 EXCP パイパ + 4 レジスター ス 14 + 8 レジスタ ー 14 + 12	レジスター 10、11、12、お よび 13

注: 付加ルーチンからシステムにパラメーターを渡すためのレジスターの規則については、個々の付加ルーチンの説明の項で説明します。

付加ルーチンをシステムに対して使用可能にする

ここでは、付加ルーチンをシステムに対して使用可能にして、それを実行できるようにするために必要な情報を提供します。

実行する前に、付加ルーチンは SYS1.LPALIB または SYS1.SVCLIB データ・セットのメンバーになっていなければなりません。付加ルーチンを SYS1.LPALIB または SYS1.SVCLIB に入れるには、システムが構築された後で、付加ルーチンをリンク・エディットして、これらのデータ・セットに組み込みます。付加ルーチンは 8 文字のメンバー名を持っている必要があり、最初の 6 文字は IGG019 とし、最後の 2 文字は WA から Z9 の範囲の任意の文字を使用できます。ユーザーのプログラムを V=R アドレス・スペースで実行し、PCI 付加ルーチンを使用する場合は、付加ルーチンおよび PCI 付加ルーチンが参照するすべてのルーチンを、SYS1.SVCLIB または固定リンク・パック域 (LPA) に置く必要があります。ストレージに固定する必要があるプログラムのリストの提供については、「z/OS MVS 初期設定およびチューニング ガイド」を参照してください。

許可された付加ルーチン・リスト (IEAAPP00)

無許可プログラムが EXCP マクロ命令で使用される DCB をオープンする場合、その DCB に関連した付加ルーチンの名前は、SYS1.PARMLIB の IEAAPP00 メンバーにリストされている必要があります。(無許可プログラムとは、7 より大きい記憶保護キーで実行され、許可プログラム機能によって許可とマークされていないプログラムをいいます。) システムの構築後にユーザーの付加ルーチンを SYS1.LPALIB または SYS1.SVCLIB に追加した後、IEBUPDTE ユーティリティーまたは区分データ・セットを更新する別のプログラムを使用して、1 つのジョブ・ステップで、IEAAPP00 を SYS1.PARMLIB に追加し、付加ルーチンの名前をその中に入れることができます。

次の例は、IEAAPP00 を SYS1.PARMLIB に追加し、1 つの EOE 付加ルーチン、2 つの SIO 付加ルーチン、2 つの CHE 付加ルーチン、および 1 つの ABE 付加ルーチンの名前を IEAAPP00 に入れる、JCL ステートメントと IEBUPDTE 入力を示しています。

```
//          JOB          ...
//          EXEC        PGM=IEBUPDTE,PARM='NEW'
//SYSPRINT DD          SYSOUT=A
//SYSUT2   DD          DSN=SYS1.PARMLIB,DISP=SHR
//SYSIN    DD          *
```

```

./ ADD NAME=IEAAPP00
EOEAPP WA,
SIOAPP X1,X2,
CHEAPP Z3,Z4,
ABEAPP Z2
./ ENDUP
/*

```

IEBUPDTE 入力について、以下の点に注意してください。

- 付加ルーチンのタイプは、1 桁目から始まる 6 文字で識別されます。EOEAPP は EOE 付加ルーチン、SIOAPP は SIO 付加ルーチン、CHEAPP は CHE 付加ルーチン、ABEAPP は ABE 付加ルーチンを識別します。(この例では PCI 付加ルーチン名は IEAAPP00 に追加されないため、PCI 付加ルーチン ID の PCIAPP は表示されていません。)
- 付加ルーチンの名前の最後の 2 文字のみを、8 桁目から始めて指定します。
- 各ステートメントは、1 つ以上の付加ルーチン名を識別し、コンマで終了します (最後のステートメントを除いて)。

IEBUPDTE を使用して、後で付加ルーチン名を追加したり、付加ルーチン名を削除したりすることもできます。次の例は、PCI および ABE 付加ルーチンの名前を、前の例で作成された IEAAPP00 付加ルーチン・リストに追加し、SIO 付加ルーチンの名前をそのリストから削除する、JCL ステートメントと IEBUPDTE 入力を示しています。

```

//          JOB. . .
//          EXEC    PGM=IEBUPDTE,PARM='NEW'
//SYSPRINT DD      SYSOUT=A
//SYSUT2   DD      DSN=SYS1.PARMLIB,DISP=SHR
//SYSIN    DD      *
./ ADD NAME=IEAAPP00
PCIAPP Y1,
EOEAPP WA,
SIOAPP X1,
CHEAPP Z3,Z4,
ABEAPP Z2,Z4
./ ENDUP
/*

```

IEBUPDTE 入力について、以下の点に注意してください。

- IEBUPDTE に対するコマンドは ADD ですが、PARM='NEW' が指定されているため、置換が生じます。
- IEAAPP00 内に残されるすべての付加ルーチン名が繰り返されています。
- IGG019Z4 は、CHE および ABE 付加ルーチンの両方です。

入出力開始 (Start-I/O) 付加ルーチン

ERP が制御権を持っていない限り、システムは、ユーザーのチャンネル・プログラムを変換して開始する直前に、制御を SIO 付加ルーチンに渡します。この付加ルーチンは、チャンネル・プログラムが後で変換されない場合でも呼び出されます。SIO 付加ルーチンは、チャンネル・プログラムを構築できます。SIO 付加ルーチンが戻るまで、システムは IOBSTART をテストしません。

オプションの戻りベクトルは、入出力要求側に以下の選択肢を与えます。

- レジスター 14 + 0 - 通常の戻り。通常のチャンネル・プログラム変換と入出力の開始。
- レジスター 14 + 4 - 入出力操作をスキップ。チャンネル・プログラムは開始されません。チャンネル・プログラムには完了が通知されません。ユーザーは POST マクロを使用して、チャンネル・プログラムの完了を通知できます。POST マクロについての詳細は、「z/OS MVS Programming: Authorized Assembler Services Guide」および「z/OS MVS Programming: Authorized Assembler Services Reference LLA-SDU」を参照してください。

POST マクロは、以下のアクションが生じた場合、チャンネル・プログラムを完了として通知します。

- 必要なレジスターが保管される。
- レジスター 10 に完了コード設定されている。これは、ECB の 4 バイトです。
- レジスター 11 に、IOB からの ECB アドレスが設定されている。
- POST マクロが発行されている。POST (11),(10),LINKAGE=BRANCH
- 戻り時に、必要なレジスターが再確立されている。

ページ固定 (Page Fix) および EXCPVR 入出力開始 (Start I/O) 付加ルーチン

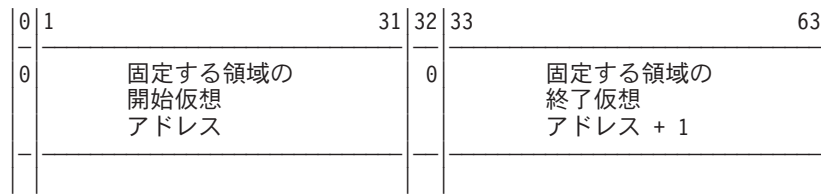
この付加ルーチンは、2 つの独立した付加ルーチンの組み合わせです。完全な付加ルーチンは、2 つのエントリー・ポイント (SIO 付加ルーチン用に 1 つと、PGFX 付加ルーチン用に 1 つ) を備えた再入可能サブルーチンです。

SIO エントリー・ポイントは、SIO サブルーチンのオフセット +0 にあります。このエントリー・ポイントから、付加ルーチン内の他の場所に分岐するための命令を使用できます。PGFX 付加ルーチンのエントリー・ポイントは、SIO サブルーチンのオフセット +4 にあります。レジスター 15 に、オフセット +0 のアドレス (SIO 付加ルーチン・エントリー・アドレス) を、PGFX 付加ルーチンのエントリー・ポイントとして設定すると、両方のエントリー・ポイントで同じエントリー・リンケージ・コードを使用できます。

PGFX 付加ルーチン

この付加ルーチンは、現行の入出力 (I/O) 要求の実行中にページング例外が生じるのを防ぐために固定する必要がある領域のアドレス・リストを作成します。この付加ルーチンには、1 つの入出力要求で複数回入ることができますが、入るたびに、固定する領域の同じリストを作成する必要があります。付加ルーチンは、レジスター 13 によって指示される 16 ワード保管域を使用できます。レジスター 10、11、および 13 は、作業用レジスターとして使用できます。

付加ルーチンによってリストに入れられる各ページ固定項目は、次のようなダブルワード・フォーマットでなければなりません。



PGFX 付加ルーチンからシステムに戻った時点で (レジスター 14 に提供される戻りアドレスを介して)、レジスター 10 は最初のページ固定項目を指し、レジスター 11 には作業域内の固定ページ項目の数が入っていなければなりません。システムは、PGFX 付加ルーチンによってリストされた領域に対応するページを固定します。関連の入出力要求が終了するまで、ページは固定されたままです。

レジスター 14 の戻りアドレス + 8 を介して、チャンネル終了 (channel end) 付加ルーチンまたは異常終了 (abnormal end) 付加ルーチンのいずれかが戻った場合、PGFX 付加ルーチンは正常に再入されません。代わりに、SIO 付加ルーチンに入り、PGFX 付加ルーチンによって作成されたページ固定リストはまだアクティブです。PURGE マクロが発行された場合 (例えば、ストレージ・スワップが発生した場合)、レジスター 14 の戻りアドレス + 8 を介して「チャンネル終了」付加ルーチンまたは「異常終了」付加ルーチンが戻った後で PGFX 付加ルーチンに入ります。入出力が復元された場合、PGFX 付加ルーチンに入ります。ページ固定リストは、ページ固定ストレージ内になければなりません。

SIO 付加ルーチン

EXCPVR を使用してチャンネル・プログラムを実行する場合、チャンネル・プログラムのオペランドの仮想アドレスを中央ストレージ・アドレスに変換してください。これは、SIO 付加ルーチン内で行う必要があります。間接データ・アドレッシングが必要な場合、SIO 付加ルーチンを使用して間接データ・アドレス・リスト (IDAL) を作成し、関連 CCW 内の IDA 標識をオンにします。

EXCPVR は VIO データ・セットに使用できますが、SIO 付加ルーチンに関して次のような考慮事項があります。

- IDAW と IDAL の使用は必要ありません。
- CSW、IDAW、または IDAL 内のアドレスは仮想でなければなりません。これは、中央ストレージ・アドレスに変換してはなりません。

UCBJBNR バイト内の UCBVRDEB ビットを調べて、データ・セットが VIO を使用して処理されているかどうかを判別できます。

仮想アドレスの変換と IDAL の構築: チャンネル・プログラム内の仮想アドレスを中央ストレージ・アドレスに変換してください。また、ユーザーの CCW にアドレスが表示される領域をチェックして、そのデータ域が特定の境界を超えているかどうかを判別してください。ビット IOBEEIDA (132 ページの図 14 で説明) は、ユーザーが 4 バイト (31 ビット) または 8 バイト (64 ビット) IDAW のいずれかを使用しているかを指定します。4 バイト IDAW を使用している場合は、2 KB 境界を考慮してください。8 バイト IDAW を使用している場合は、4 KB 境界を考慮してください。データ域が境界を超えている場合は、超えている境界ごとに IDAL に追加の IDAW を提供してください。チャンネル・サブシステムは、読み取りまたは書き込み操作中に境界を超えた場合、IDAL を使用して、読み取りまたは書き込みを

継続する領域を識別します。チャンネル・サブシステムによって処理される場合、IDAL には中央ストレージ・アドレスが入っていなければなりません。

チャンネル・プログラム内の仮想アドレスを実アドレスに変換する前に、最初にデータ域を中央ストレージ内にページを固定する必要があります。データ域のページを固定した後、LRA 命令を使用して、チャンネル・プログラム内の仮想アドレスを中央ストレージ・アドレスに変換します (中央ストレージ・アドレスが 2 GB より下の場合)。ユーザーが 24 ビットまたは 31 ビット・アドレッシング・モードのいずれであるかに関係なく、LRA 命令は 31 ビット中央ストレージ・アドレスを戻しますが、中央ストレージ・アドレスが 2 GB より上の場合は、これらのアドレッシング・モードでは失敗します。中央ストレージ・アドレスが 2 GB より上の場合は、LRAG または STRAG 命令を使用して、仮想アドレスを実アドレスに変換するか、または LRA 命令を使用する必要があります (最初に 64 ビット・アドレッシング・モードに切り替えた後で)。データ・バッファに関連した中央ストレージ・アドレスのすべてが 2 GB より下の場合は、2 KB (31 ビット) IDAW を使用して、データをアドレス指定できます。そうでない場合は、4 KB (64 ビット) IDAW を使用してデータをアドレス指定する必要があります。システム・リソースを最も効率的に使用するために、GETMAIN または STORAGE マクロを使用してストレージを取得する場合は、LOC=(ANY,64) または LOC=(BELOW,64) をコーディングします。「z/OS MVS プログラミング: アセンブラー・サービス 解説書 ABE-HSP」および 97 ページの『IDAW 要件』を参照してください。

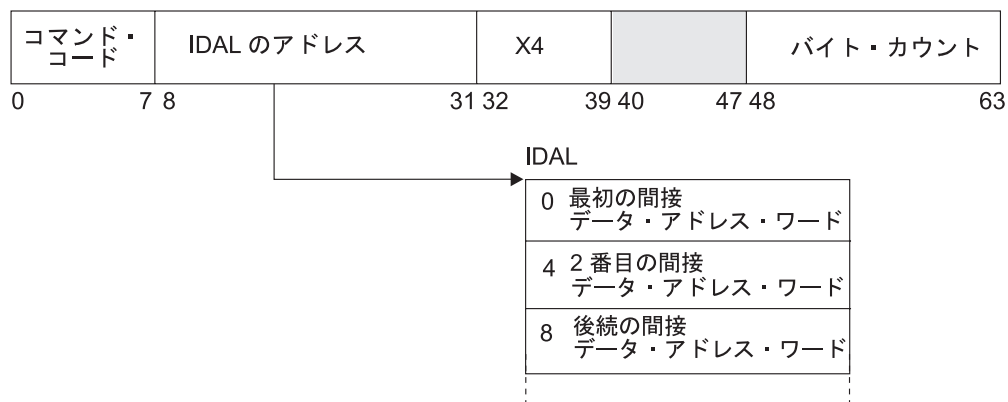
注:

1. 64 ビット IDAW は、z/Architecture モードでのみサポートされ、直接アクセス・ストレージ・デバイス (DASD) および IBM 提供のすべてのカートリッジ磁気テープ装置に対してのみサポートされます。ユーザーは、UCB 内のビット UCBEIDAW を調べて、64 ビット IDAW が装置によってサポートされているかどうかを判別することが必要です。
2. ユーザーのプログラムを 64 ビット・アドレッシング・モードに切り替える場合、変更するレジスターの高位 32 ビットを保管および復元したことを確認する必要があります。これを行わないと、ユーザーのアドレス・スペース内の他の 64 ビット・プログラムで予測不能の結果が生じることがあります。

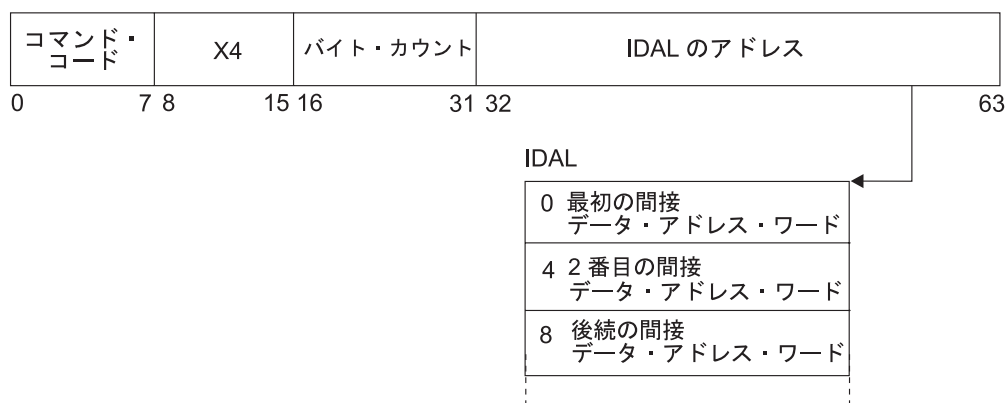
注:

次の例は、CCW と IDAW の関係を示しています。

フォーマット 0 CCW



フォーマット 1 CCW



IDAL に関する以下の情報に注意してください。

- IOBEEIDA が 1 でない場合は、最初の項目の後に、ユーザーのデータ域が 2 KB ページ境界を超えるごとに 1 項目を IDAL に入れます。IOBEEIDA が 1 の場合 (4 バイト IDAW を示す) は、最初の項目の後に、ユーザーのデータ域が 4 KB ページ境界を超えるごとに 1 項目を IDAL に入れます。
- フォーマット 0 CCW が、データ・アドレスではなく IDAL アドレスを持っている場合、ビット 37 をオンに設定して、そのことをチャンネルに通知する必要があります。これと等価のフォーマット 1 CCW ビットは、ビット 13 です。
- IDAL 内に必要な項目の最大数は、次のように CCW 内のカウントから算定されます。

IDAL 項目の数 = ((CCW バイト・カウント - 1)/P) + 1。 4 バイト IDAW を使用する場合は、この公式の P を 2048 で置き換えます。8 バイト IDAW の場合は、P を 4096 で置き換えます。(剰余がゼロでない場合は、次の高位の整数に切り上げます。)

必要な 4 バイト IDAL 項目の数は、データを超える 2 KB 境界の数によって決まります。例えば、ユーザーのデータが 800 バイトの長さで、2 KB 境界を超えない場合は、IDAL 項目は必要ありません。データが 4 KB 境界を超える場合は、2 つの IDAL 項目が必要です。データの長さが 5000 KB の場合は、少なくとも 2 つの IDAL 項目が必要です。データが 4 KB 境界を 2 つを超える場合は、4 つの IDAL 項目が必要です。

最初の間接アドレスは、データ域の最初のバイトの中央ストレージ・アドレスです。2 番目以降の間接アドレスは、データ域の 2 番目以降の 2 KB または 4 KB 境界の中央ストレージ・アドレスです。例えば、データ域の中央ストレージ・アドレスが X'707FF' で、バイト・カウントが X'1802' の場合、4 バイト IDAL には次の中央ストレージ・アドレスが入ります (中央ストレージ・アドレスは連続していると想定した場合)。

```
707FF
70800
71000
```

データ域の中央ストレージ・アドレスが X'707FF' で、バイト・カウントが X'800' の場合、IDAL には次のアドレスが入ります。

```
707FF
70800
```

プログラム制御割り込み (Program-Controlled Interruption) 付加ルーチン

チャンネルが、チャンネル・プログラム内で 1 つ以上のプログラム制御割り込み (PCI) ビットがオンになっているのを検出した場合、この付加ルーチンに入ります。PCI ビットがオンであることをチャンネルが検出するたびに (あるいは、それ以上の回数) 入ることができます。付加ルーチンに入る前に、サブチャンネル状況ワードの内容を入力ブロックのチャンネル状況ワード・フィールドに入れます。

PCI ビットがオンのチャンネル・プログラムを ERP が再試行する場合、PCI 付加ルーチンに再入します。ERP が制御を持っている場合、IOB フラグがセットされます (IOBFLAG1 = X'20')。(コマンド再試行時に生じる特殊な PCI 条件については、101 ページの『コマンド再試行に関する考慮事項』を参照してください。)

EXCP (EXCPVR ではなく) の後で PCI からチャンネル・プログラムに通知するには、ステップが ADDRSPC=VIRT (V=V) で実行されている場合、または許可プログラムが V=R で実行されている場合は、SIO 付加ルーチンの項で説明した手順を使用します。

ステップが ADDRSPC=REAL (V=R) で実行されており、許可プログラムが EXCP 要求を発行した場合、または EXCPVR 要求が発行された場合には、PCI 付加ルーチンは中央ストレージ・アドレスを使用します。PCI 付加ルーチンからチャンネル・プログラムに通知するには、次の手順を使用します。POST マクロについて詳しくは、「z/OS MVS Programming: Authorized Assembler Services Guide」および「z/OS MVS Programming: Authorized Assembler Services Reference LLA-SDU」を参照してください。

POST マクロは、次のようにコーディングします。

```
POST ecbaddr,compcode,ASCB=addr,ERRET=addr,ECBKEY=key,
LINKAGE=BRANCH,MEMREL=NO
```

ERRET ルーチン・アドレスは、BR 14 命令を指していなければなりません。この命令は、任意のアドレス・スペースからアドレス可能なストレージ内にあり (例えば、CVTBRET)、24 ビットでアドレス可能でなければなりません。

注: このキーワードを使用している場合、POST マクロから戻るときに、レジスター 9 と 14 のみが復元されます。

以下の手順で、PCI 付加ルーチンからチャンネル・プログラムに通知します。

1. POST マクロから戻るときには、レジスター 9 と 14 のみが復元されるので、必要なレジスターを保管します。
2. レジスター 0 に ECB キーを設定します。
3. レジスター 10 に 4 バイト完了コードを設定します。
4. レジスター 11 に ECB アドレスを設定します。
5. レジスター 12 にエラー・ルーチン・アドレスを設定します。これは、エラー・ルーチン・アドレスを CVTBRET のアドレスに設定し、高位バイトの高位ビット (X'80') をオンにすることによって行います。
6. レジスター 13 に ASCB アドレスを設定します。ASCB アドレスを持っていない場合は、次の手順を使用して ASCB アドレスを取得できます。
 - a. EPAR 命令を発行して、ASID を取得する。EPAR 命令については、「z/Architecture 解説書」を参照してください。
 - b. LOCASCB マクロを発行して、ASCB アドレスを取得する。LOCASCB マクロの情報は、「z/OS MVS Programming: Authorized Assembler Services Reference LLA-SDU」に記載されています。
7. POST マクロを発行します。
POST (11),(10),ASCB=(13),ERRET=(12),ECBKEY=(0),LINKAGE=BRANCH
8. 戻り時に、必要なレジスターを再設定します。

通常操作のために制御をシステムに戻すには、レジスター 14 の戻りアドレスを使用します。

エクステント終了 (End-of-Extent) 付加ルーチン

シリンダー終わり (end-of-cylinder) またはファイル保護 (file-protect) 条件が発生した場合、システムはシーク・アドレスを次の高位シリンダーまたはトラック・アドレスに更新して、要求を再試行します。新規シーク・アドレスがデータ・セットのエクステント範囲内である場合、要求は実行されます。新規シーク・アドレスがデータ・セットのエクステント範囲内でない場合、EOE 付加ルーチンに入ります。次のエクステントの要求を試行するには、新規シーク・アドレスをレジスター 6 で指示された位置に移動します。

チャンネル・プログラムに組み込まれた完全シーク (コマンド・コード = 07) によって「ファイル保護」が生じた場合、要求は永続エラーとしてフラグが付けられ、ABE 付加ルーチンに入ります。

入出力ブロックに指定されたシーク・アドレスが、データ・エクステント・ブロックに示された割り振りエクステント限界の外側である場合、エクステント終了 (EOE) 付加ルーチンに入ります。

レジスター 14 の戻りアドレスを使用して制御をシステムに戻すと、ABE 付加ルーチンに入ります。エクステント終了 (end-of-extent) エラー・コード (X'42') が入出力ブロックの ECB コード・フィールドに入れられ、後続の ECB に通知します。

オプションとして、以下の戻りアドレスを使用できます。

- レジスター 14 の内容 + 4: チャンネル・プログラムは X'42' で完了が通知されま
す。この要求のために ABE 付加ルーチンには再入しません。
- レジスター 14 の内容 + 8: 要求が再試行されます。

EOE 付加ルーチンのレジスター 10 から 13 は、その内容を保管および復元せずに
使用できます。

異常終了 (Abnormal-End) 付加ルーチン

システムが異常条件を処理するのに使用する方式を決めるには、異常終了 (ABE) 付
加ルーチンを使用します。以下では、ABE 付加ルーチンの使用法について説明しま
す。

装置例外、誤長 (wrong-length) 標識、エクステント範囲外 (out-of-extent) エラー、
代行受信条件(つまり、装置終了エラー)、装置チェック、プログラム・チェック、チ
ャネル・データ・チェック、チャンネル制御チェック、インターフェース制御チェッ
ク、チューニング・チェックなどの異常条件が生じた場合、この付加ルーチンに入
ることができます。また、以下の条件下で、すでにパージされた DCB に対して
EXCP が発行された場合にも入ることができます。

- IOBECBCC が X'41' に設定されている場合、装置例外または誤長レコード標識
(または、その両方) が原因でこの付加ルーチンに入りました。システムは前にチ
ャネル終了 (channel-end) 付加ルーチン呼び出しました (存在する場合)。これ
らの条件について詳しくは、149 ページの『チャンネル終了 (Channel-End) 付加
ルーチン』を参照してください。
- IOBECBCC が X'42' に設定されている場合、エクステント範囲外 (out-of-extent)
エラーが原因でこの付加ルーチンに入りました。システムは前に「チャンネル終
了」付加ルーチン呼び出しました (存在する場合)。
- この付加ルーチンに入り、IOBECBCC が X'4B' に設定されている場合、テー
プ・エラー・リカバリー手順 (ERP) が予期しないロード開始点を検出したか、ま
たは CSW のコマンド・アドレス・フィールドでゼロを検出したかのいずれか
です。
- IOBECBCC が X'7E' に設定されている場合、最初に代行受信条件が原因でこの付
加ルーチンに入りました。その後、エラー条件が永続的であると判別された場
合、この付加ルーチンに再入して、IOBECBCC が X'44' に設定されます。代行受
信条件は、前の要求のチャンネル終了後の装置終了時にエラーが検出されたこと
を知らせます。
- IOBECBCC が X'48' に設定されていた場合、すでにパージされた DCB に対し
て EXCP が発行されたために付加ルーチンに入りました。これは関連要求にのみ適
用されます。
- 付加ルーチンに入り、IOBECBCC が X'7F' に設定されている場合、装置チェッ
ク、プログラム・チェック、記憶保護チェック、チャンネル・データ・チェック、
チャンネル制御チェック、インターフェース制御チェック、またはチューニング・
チェックが原因と考えられます。IOBECBCC が X'7F' の場合、それは関連チャ
ネル・プログラム内で最初に検出されたエラーです。IOBIOERR フラグ
(IOBFLAG1 のビット 5) がオンの場合、IOBECBCC フィールドには X'41'、
X'42'、X'48'、X'4B'、または X'4F' (永続的な入出力エラーを表す) が入ります。

エラーが永続的かどうかを判別するには、IOB の IOBECBCC フィールドを検査します。エラーのタイプを判別するには、チャンネル状況ワード・フィールドと IOB 内のセンス情報を検査します。ただし、IOBECBCC が X'42'、X'48'、または X'4F' の場合は、これらのフィールドは適用されません。X'44' の場合、CSW は適用できますが、センス情報は装置チェック・ビットが設定されている場合にのみ有効です。

レジスター 14 の戻りアドレスを使用して制御をシステムに戻すことにより、チャンネル・プログラムに完了が通知され、その要求エレメントが使用可能にされます。オプションとして、以下の戻りアドレスを使用できます。

- レジスター 14 の内容 + 4: チャンネル・プログラムに完了は通知されませんが、その要求エレメントは使用可能にされます。ユーザーは、SIO 付加ルーチンの項で説明した呼び出しシーケンスを使用して、チャンネル・プログラムに通知できます。
- レジスター 14 の内容 + 8: チャンネル・プログラムに完了は通知されず、その要求エレメントは再試行のために要求キューに戻されます。入出力ブロックの IOBFLAG1、IOBFLAG2、および IOBFLAG3 フィールドを再初期化して、IOBERRCT フィールドをゼロにリセットしてください。追加の予防措置として、IOBSENS0、IOBSENS1、および IOBCSW フィールドもクリアしてください。
- レジスター 14 の内容 + 12: チャンネル・プログラムに完了は通知されず、その要求エレメントは使用可能にされません。(この戻りコードは、付加ルーチンが、非同期ルーチンのスケジューリングに使用するために、要求キュー・エレメントを出口エフェクターに渡した場合にのみ使用してください。)

ABE 付加ルーチンのレジスター 10 から 13 は、その内容を保管および復元せずに使用できます。

チャンネル終了 (Channel-End) 付加ルーチン

チャンネル終了 (CHE)、チャンネル終了を伴うまたは伴わない装置例外 (UE)、あるいは誤長レコード (WLR) を伴うチャンネル終了が、それ以外の異常終了条件を伴わずに発生した場合、この付加ルーチンに入ります。

レジスター 14 の戻りアドレスを使用して制御を EXCP プロセッサに戻すことにより、チャンネル・プログラムに完了が通知され、その要求エレメントが使用可能にされます。装置例外または誤長 (wrong-length) レコードの場合、チャンネル・プログラムに完了が通知される前に ERP が実行され、IOBFLAG1 内の IOBIOERR フラグ (X'04') がオンに設定されます。CSW 状況は、IOBCSW フィールドから入手できます。

付加ルーチンが誤長レコードまたは装置例外 (あるいは、その両方) を処理する場合、IOBFLAG1 内の IOBIOERR (X'04') フラグをオフにして、正常に制御を戻すことができます。その後、イベントの完了が通知されます (IOBECBCC フィールドの高位バイトから取られた、正常条件下の完了コード X'7F')。付加ルーチンが IOBIOERR フラグをゼロにリセットせずに正常に制御を戻した場合、要求は関連装置 ERP に経路指定されます。ERP がエラーを訂正できなかった場合、ABE 付加ルーチンに入り、IOBECBCC の完了コードは X'41' に設定されます。(148 ページの『異常終了 (Abnormal-End) 付加ルーチン』のステップ 1 を参照してください。)

オプションとして、以下の戻りアドレスを使用できます。

- レジスター 14 の内容 + 4: チャンネル・プログラムに完了は通知されませんが、その要求エレメントは使用可能にされます。ユーザーは、SIO 付加ルーチンの項で説明した呼び出しシーケンスを使用して、チャンネル・プログラムに通知できます。これは、入出力ブロック内の ECB 以外の ECB を通知するのに特に便利です。
- レジスター 14 の内容 + 8: チャンネル・プログラムに完了は通知されず、その要求エレメントは要求キューに戻されて、同じまたは異なるアドレスのチャンネル・プログラムを実行できるようになります。チャンネル・プログラムを正しく実行するために、入出力ブロックの IOBFLAG1、IOBFLAG2、および IOBFLAG3 フィールドを再初期化し、エラー件数フィールドをゼロに設定してください。追加の予防措置として、IOBSENS0、IOBSENS1、および IOBCSW フィールドもクリアしてください。ユーザーは、制御を戻す前に IOBSTART を変更できます。システムは、新規の EXCP または EXCPVR の場合と同様に付加ルーチンを呼び出します。
- レジスター 14 の内容 + 12: チャンネル・プログラムに完了は通知されず、その要求エレメントは使用可能にされません。(この戻りコードは、付加ルーチンが、非同期ルーチンのスケジューリングに使用するために、が要求キュー・エレメントを出口エフェクターに渡した場合にのみ使用してください。非同期出口ルーチンについての詳細は、「z/OS MVS Programming: Authorized Assembler Services Guide」を参照してください。)

CHE 付加ルーチンのレジスター 10 から 13 は、その内容を保管および復元せずに使用できます。

相対トラック・アドレスの実トラック・アドレスへの変換

24 ビットまたは 31 ビット・モードで呼び出せる常駐のシステム変換ルーチンを使用して、相対トラック・アドレスを実アドレスに変換します。

変換ルーチンには 2 つのエントリー・ポイントがあります。1 つはコミュニケーション・ベクトル・テーブル (CVT) の CVTPCNVT フィールドにあるアドレスで、65535 トラック以下のデータ・セットを対象としています。もう 1 つは、CVTPCNVT 内のアドレスからオフセット +12 にあるエントリー・ポイントで、任意のデータ・セットを対象としています。次のいずれかの方法で、変換ルーチンを呼び出してください。

- BALR 14,15 トラック変換のために +0 エントリー・ポイントを呼び出す
- BASR 14,15 トラック変換のために +0 エントリー・ポイントを呼び出す
- BAL 14,12(,15) トラック変換のために +12 エントリー・ポイントを呼び出す
- BAS 14,12(,15) トラック変換のために +12 エントリー・ポイントを呼び出す

IBM は、この変換ルーチンの呼び出しマクロを提供していません。

CVT のアドレスは、ストレージ・ロケーション 16 (マクロ IHAPSA によってマップされる、PSA データ域のフィールド FLCCVT) にあります。

変換ルーチンは、その作業のすべてを汎用レジスターで行います。レジスター 0、1、2、14、および 15 に、ルーチンへの入力をロードします。レジスターの使用法は、次のとおりです。

表 38. 相対アドレスから実アドレスに変換するためのレジスターとその使用法

レジスター	使用
0	<p>相対トラック番号の 4 バイト値をロードする必要があります。</p> <p>CVTPCNVT+0 で入った場合は、この値を TTR_n の形式にする必要があります。</p> <p>TT データ・セットの開始に対する相対トラック番号</p> <p>R そのトラック上のブロックの識別</p> <p>n 区分化された連結におけるデータ・セットの連結番号。この場合、DCB の DSORG 値は PO です。0 は 1 番目のデータ・セット、1 は 2 番目のデータ・セットを表します。データ・セットが連結されていない場合、または連結が区分化されていない場合は、ゼロを設定してください。</p> <p>CVTPCNVT+12 で入った場合は、この値を TTTR の形式にする必要があります。ここで、</p> <p>TTT データ・セットの開始に対する相対トラック番号</p> <p>R そのトラック上のブロックの識別</p> <p>このエントリー・ポイントは、任意のデータ・セットのタイプを対象としています。</p>
1	<p>データ・セットのデータ・エクステント・ブロック (DEB) のアドレスをロードする必要があります。各 DEB は 16 MB 未満でなければなりません。+0 エントリー・ポイントを呼び出した場合、呼び出されたルーチンは、このレジスターの高位バイトをクリアします。+12 エントリー・ポイントを呼び出した場合は、このバイトに 'X'00' が含まれていなければなりません。UCB アドレス・フィールドは 3 バイトであることに注意してください。</p>
2	<p>処理するブロックの実アドレスを受け取る 8 バイト域のアドレスをロードする必要があります。変換後のアドレスは、MBBCCCHHR 形式です。ここで、</p> <p>M 直接アクセス・プログラムに関連した、データ・エクステント・ブロック内のエクステント項目を示します。(0 は最初のエクステントを示し、1 は 2 番目のエクステントを示す、という具合になります。)</p> <p>BB 2 バイトのゼロ</p> <p>CC シリンダー・アドレス</p> <p>HH 実トラック・アドレス</p> <p>R ブロック番号</p>
3-8	<p>変換ルーチンによって使用されません。</p>
9-13	<p>変換ルーチンによって使用され、復元されません。+12 エントリー・ポイントを呼び出した場合、このレジスターの高位 3 バイトにゼロが含まれ、また、低位バイトは、+0 エントリー・ポイントでのレジスター 0 の低位バイトについて説明されているとおりにしている必要があります。</p>
14	<p>変換ルーチンの実行後に制御を戻す宛先アドレスをロードする必要があります。</p>

表 38. 相対アドレスから実アドレスに変換するためのレジスターとその使用法 (続き)

レジスター	使用
15	変換ルーチンによって基底レジスターとして使用され、変換ルーチンが制御を受け取る場所のアドレス (CVT のフィールド CVTPCNVT からの) をロードする必要があります。

相対変換ルーチンから実変換ルーチンへの戻りコード

制御がユーザーのプログラムに戻されたとき、レジスター 15 に以下の戻りコードのいずれかが入ります。

表 39. 相対変換ルーチンから実変換ルーチンへの戻りコード

戻りコード	説明
0 (X'00')	正常な変換。
4 (X'04')	相対トラック・アドレスは、DEB に定義されたエクステント範囲外の実トラック・アドレスに変換されました。
8 (X'08')	内部アクセス方式制御ブロックは無効です。制御ブロックが有効でなければならない場合は、IBM サービスにご連絡ください。
12 (X'0C')	内部アクセス方式制御ブロックは無効です。制御ブロックが有効でなければならない場合は、IBM サービスにご連絡ください。
16 (X'10')	受け渡された連結番号がデータ・セットには大きすぎます。
20 (X'14')	最後のエクステント内の DEBNMTRK フィールドは、入力トラック番号がこのエクステント内をポイントしていることを示しますが、計算された CCHH がこのエクステント内にありません。

実トラック・アドレスの相対トラック・アドレスへの変換

24 ビットまたは 31 ビット・モードで呼び出せる常駐のシステム変換ルーチンを使用して、実トラック・アドレスを相対トラック・アドレスに変換します。

変換ルーチンには 2 つのエントリー・ポイントがあります。1 つはコミュニケーション・ベクトル・テーブル (CVT) の CVTPCNVT フィールドにあるアドレスで、最大 65535 トラックを持つデータ・セットを対象としています。もう 1 つは、CVTPCNVT 内のアドレスからのオフセット +12 で、任意のデータ・セットを対象としています。次のいずれかの方法で、変換ルーチン呼び出します。

- BALR 14,15 トラック変換のために +0 エントリー・ポイント呼び出す
- BASR 14,15 トラック変換のために +0 エントリー・ポイント呼び出す
- BAL 14,12(,15) トラック変換のために +12 エントリー・ポイント呼び出す
- BAS 14,12(,15) トラック変換のために +12 エントリー・ポイント呼び出す

IBM は、この変換ルーチンの呼び出しマクロを提供していません。

変換ルーチンは、TTR0 (CVTPCNVT+0 で入った場合) または TTTR (CVTPCNVT+12 で入った場合) のいずれかを戻します。TTR0 および TTTR の説明は、151 ページの表 38 に記載されています。

CVT のアドレスは、ストレージ・ロケーション 16 (マクロ IHAPSA によってマップされる、PSA データ域のフィールド FLCCVT) にあります。

変換ルーチンは、その作業のすべてを汎用レジスターで行います。レジスター 0、1、2、14、および 15 に、ルーチンへの入力を読み込みます。レジスターの使用法は、次のとおりです。

表 40. 実アドレスから相対アドレスに変換するためのレジスターとその使用法

レジスター	使用
0	呼び出し側に渡す、結果の TTR0 または TTTR をロードします。これら 2 つの形式は、151 ページの表 38 のレジスター 0 に関する箇所に説明されています。
1	データ・セットのデータ・エクステント・ブロックのアドレスをロードする必要があります。各 DEB は 16 MB より下になければなりません。+0 エントリー・ポイントを呼び出した場合、呼び出されたルーチンは、このレジスターの高位バイトをクリアします。+12 エントリー・ポイントを呼び出した場合は、このバイトに 'X'00' が含まれていなければなりません。
2	TTR0 または TTTR に変換する実アドレスを含んでいる 8 バイト域のアドレスをロードする必要があります。実アドレスは、MBBCCCHHR 形式です。
3-8	変換ルーチンによって使用されません。
9-13	変換ルーチンによって使用され、復元されません。
14	変換ルーチンの実行後に制御を戻す宛先アドレスをロードする必要があります。
15	変換ルーチンによって基底レジスターとして使用され、CVT のフィールド CVTPRLTV の内容をロードする必要があります。

変換ルーチンからの戻りコード

制御がユーザーのプログラムに戻されたとき、レジスター 15 に以下の戻りコードのいずれかが入ります。

表 41. 実変換ルーチンから相対変換ルーチンへの戻りコード

戻りコード	説明
0 (X'00')	正常な変換。
4 (X'04')	CCHH はエクステント M の外にあります。戻された相対トラックは無効です。+12 エントリー・ポイントを呼び出した場合は、出力 R バイトが 'X'FE' に設定されます。
8 (X'08')	内部アクセス方式制御ブロックは無効です。制御ブロックが有効でなければならない場合は、IBM サービスにご連絡ください。
12 (X'0C')	内部アクセス方式制御ブロックは無効です。制御ブロックが有効でなければならない場合は、IBM サービスにご連絡ください。
16 (X'10')	渡されたエクステント番号 M がデータ・セットには大きすぎます。+12 エントリー・ポイントを呼び出した場合は、出力 R バイトが 'X'FE' に設定されます。

表 41. 実変換ルーチンから相対変換ルーチンへの戻りコード (続き)

戻りコード	説明
20 (X'14')	計算された相対トラック番号は、戻される番号としては大きすぎます。+0 エントリーを呼び出した場合は、トラック番号が X'FFFF' を超えたため、ルーチンが X'FFFFFFE0' を戻しました。+12 エントリー・ポイントを呼び出した場合は、トラック番号が X'FFFFFF' を超えたため、ルーチンが X'FFFFFFFE' を戻しました。この原因として考えられるのは、DEB が無効であるか、あるいは、(エントリー +0 の場合は) データ・セットがラージ・フォーマット・データ・セットであるにもかかわらず、トラック位置がデータ・セット内の X'FFFF' トラックを超えているかのいずれかです。

注: +12 エントリー・ポイントで戻りコード 4、16、および 20 が出た場合、このルーチンは渡されたものとは異なる R バイト (X'FE') の値を戻します。

RPS 装置上のブロックのセクター番号の入手

RPS 装置および非 RPS 装置の両方で使用できるプログラムの場合、UCBRPS ビット (UCB のオフセット 17 のビット 3) をテストして、装置が回転位置感知機構を備えているかどうかを判別してください。UCBRPS ビットがオフの場合、その装置に対しては set sector コマンドを指定したチャンネル・プログラムを発行しないでください。セクター変換ルーチンのエントリー・ポイントのアドレスは、CVT の CVT0SCR1 フィールドにあります。

ユーザーのプログラムは、BASR 14,15 または BAS 14,16,(15) 命令を発行することにより、変換ルーチンを呼び出すことができます。トラック・バランスをルーチンに渡す場合は、BAS 14,8(15) または BAS 14,20,(15) を使用してルーチンを呼び出してください。可変長レコードを持つモジュロ装置上でセクター値を計算する場合は、トラック・バランスをセクター変換ルーチンに渡してください。

セクター変換ルーチンは、そのエントリー・ポイントで、24 ビットまたは 31 ビット・モードで呼び出すことができます。24 ビット・モードで呼び出す場合、アドレスはすべて 16 MB 境界より下を指していなければなりません。

注:

1. セクター変換ルーチンは、PDSE または拡張フォーマットのデータ・セットをサポートしません。PDSE または拡張フォーマットのデータ・セットに対してセクター変換ルーチンを使用すると、物理レコード・フォーマットと矛盾した結果が戻されます。
2. セクター変換ルーチンは、HFS ファイルをサポートしません。HFS ファイルで変換ルーチンを使用すると、予測不能の結果を受け取ります。

RPS 装置の場合、変換ルーチンはその作業のすべてを汎用レジスターで行います。レジスター 0、2、14、および 15 に、ルーチンへの入力をロードします。レジスターの使用法は、次のとおりです。

表 42. セクター変換ルーチンのレジスターとその使用法

レジスター	使用
0	<p>固定標準ブロック、または区分データ・セット内にある固定非ブロック化レコードの場合、レジスター 0 には、XXKR 形式の 4 バイト値を含める必要があります。ここで、</p> <p>XX 物理ブロック・サイズを含んでいる 2 バイト・フィールド</p> <p>K キー長を含んでいる 1 バイト・フィールド</p> <p>R セクター値を必要とするレコード番号を含んでいる 1 バイト・フィールド</p> <p>固定長レコードを示すには、レジスター 0 の高位ビットをオフにします (0 に設定)。</p> <p>トラック・バランスの引き渡し: レジスター 0 には、必要なレコードに先行するレコードのトラック・バランスを示す 4 バイト値を含める必要があります。</p> <p>その他の場合: レジスター 0 には、BBIR 形式の 4 バイト値を含める必要があります。ここで、</p> <p>BB ターゲット・レコードまでの (ただし、これは含まない) トラック上のキーとデータ・バイトの総数。</p> <p>I 1 バイトのキー標識 (キー付きレコードの場合は 1、キーなしレコードの場合は 0)。</p> <p>R セクター値が必要なレコード番号を含んでいる 1 バイト・フィールド。可変長レコードを示すには、レジスター 0 の高位ビットをオンにします (1 に設定)。</p>
1	セクター変換ルーチンによって使用されません。
2	<p>オフセット 0 または 8 で呼び出された場合は、4 バイト・フィールドを含んでいなければなりません。ここで、</p> <p>最初のバイトは、装置の UCB 装置タイプ・コードです (UCB+19 から入手可能)。</p> <p>残りの 3 バイトは、セクター値を受け取る 1 バイト域のアドレスです。</p> <p>オフセット 16 または 20 で呼び出された場合は、セクター値を受け取る 1 バイト域のアドレスが入っていなければなりません。</p>
3-8	未使用。
9-10	変換ルーチンによって使用され、保管または復元されません。
11	<p>オフセット 0 または 8 で呼び出された場合は、未使用。</p> <p>オフセット 16 または 20 で呼び出された場合は、下位バイトに 1 バイト UCBTBYT4 が入ります。残りの 3 バイトにはゼロが入っていなければなりません。</p>
12,13	未使用。
14	セクター変換ルーチンの実行後に制御が戻されるアドレスをロードする必要があります。
15	変換ルーチンによって基底レジスターとして使用され、変換ルーチンへのエントリー・ポイントのアドレス (CVT のフィールド CVT0SCR1 からの) をロードする必要があります。

第 5 章 XDAP を使用した直接アクセス装置への読み取りおよび書き込み

この章では、XDAP マクロ命令に関する情報を提供します。他の IBM オペレーティング・システムとの互換性に関する情報も含まれています。IBM では、XDAP の代わりに VSAM などのアクセス方式を使用することをお勧めします。

直接アクセス実行プログラム (XDAP) は、直接アクセス・ボリューム上のブロックの読み取り、検査、または更新に使用されるマクロ命令です。この章では、XDAP マクロ命令の機能とその使用法について説明します。また、XDAP で使用されるマクロ命令や生成される制御ブロックについても説明します。

XDAP 仕様のほとんどはチャンネル実行プログラム (EXCP) マクロ命令の仕様と同じであるため、本書の 97 ページの『IDAW 要件』に記載されている情報を十分に理解していることが必要です。また、システム制御プログラムのアクセス方式ルーチンの使用に関する情報を提供している「*z/OS DFSMS データ・セットの使用法*」の内容も十分に理解しておく必要があります。

標準 IBM データ・アクセス方式を使用していない場合、XDAP を発行することにより、データ・セットのレコードの読み取りや更新に必要な制御情報およびチャンネル・プログラム情報を生成できます。

制約事項: XDAP は、区分データ・セット、拡張 (PDSE)、拡張フォーマット・データ・セット、UNIX ファイル、UNIX システム・サービス・データ・セット、または SYSIN および SYSOUT データ・セットをサポートしません。

XDAP を使用してデータ・セットにブロックを追加することはできませんが、既存のブロックのキーを変更することはできます。ブロック構成またはデータ・セット編成を読み取ったり、更新したりすることができます。

XDAP は、標準アクセス方式に比べてストレージ所要量が少なくてすみます。しかし、XDAP は、標準アクセス方式に含まれている制御プログラム・サービスの多くを提供しません。例えば、XDAP が発行された場合、システムはレコードのブロックまたはブロック解除を行わず、ブロック長を検査しません。

XDAP で使用する仮想ストレージ (OPEN および CLOSE パラメーター・リストを除く)、DCBE、および IOBE は、すべて 24 ビットでなければなりません。

XDAP の使用

XDAP を発行するには、処理対象のブロックを含んでいるトラックの実トラック・アドレスを提供します。ブロック識別番号またはブロック・キーも提供し、ブロックを検索する際にどちらを使用するのかを指定します。識別番号によってブロックを検索する場合、ブロックのキー部分とデータ部分の両方の読み取りまたは更新ができます。キーによってブロックを検索する場合は、データ部分のみ処理できます。

入出力操作に対する追加の制御が必要な場合は、付加ルーチンを作成して、それを LPA ライブラリーに入れる必要があります。これらのルーチンとそのコーディング仕様については、97 ページの『IDAW 要件』で説明します。

XDAP マクロ命令を使用する場合、読み取りまたは更新するデータ・セット用のデータ制御ブロック (DCB) を生成するために、ユーザーのプログラムに DCB マクロ命令をコーディングします。また、データ制御ブロックを初期化し、データ・エクステント・ブロック (DEB) を生成する OPEN マクロ命令もコーディングします。XDAP マクロ命令を実行する前に、OPEN マクロ命令を実行する必要があります。

XDAP マクロ命令をアセンブルすると、制御ブロックと後続の実行可能コードが生成されます。この制御ブロックは、論理的に 3 つのセクションに分割できます。

- イベント制御ブロック (ECB)。直接アクセス・チャンネル・プログラムが終了するたびに完了コードが提供されます。
- 入出力ブロック (IOB)。直接アクセス・チャンネル・プログラムに関する情報が入ります。
- 直接アクセス・チャンネル・プログラム。3 つか 4 つのチャンネル・コマンド・ワード (CCW) で構成されます。生成されるチャンネル・プログラムのタイプは、XDAP マクロ命令のパラメーターによって決まります。チャンネル・プログラムが実行されると、実アドレスまたはキーによってブロックを検索し、ブロックの読み取り、更新、または検査を行います。

チャンネル・プログラムが終了すると、完了コードがイベント制御ブロックに入れます。XDAP を発行して、直接アクセス・プログラムが終了した後、イベント制御ブロックのアドレスを指定するために WAIT マクロ命令を使用して、制御を再取得します。ボリュームの切り替えが必要な場合は、EOV マクロ命令を発行します。データ・セットの処理が完了した後、CLOSE マクロ命令を発行して、データ制御ブロックを復元します。

XDAP で使用されるマクロ命令

XDAP マクロ命令を使用する場合、DCB、OPEN、CLOSE、WAIT、および、場合によっては EOV マクロ命令もコーディングする必要があります。XDAP マクロ命令のパラメーターについては、160 ページの『直接アクセス・プログラムの実行』で説明します。以下では、その他の必要なマクロ命令に関する特殊要件やオプションについて説明します。パラメーターのリストは、107 ページの『EXCP での実行可能マクロ命令の使用』を参照してください。

データ制御ブロックの定義

直接アクセス・チャンネル・プログラムで読み取り、更新、または検査するデータ・セットごとに DCB マクロ命令を発行します。コーディングする必要があるマクロ命令パラメーターについては、113 ページの『データ制御ブロック (DCB) フィールド』を参照してください。

データ制御ブロックの初期化

OPEN マクロ命令は、1 つ以上のデータ制御ブロックを初期化して、関連データ・セットを処理できるようにします。直接アクセス・プログラムで使用されるすべてのデータ制御ブロックに対して OPEN を発行します。OPEN の実行時に行われる手順の一部には、以下のアクションが含まれます。

- データ・エクステント・ブロックを組み立てる
- DD ステートメントおよびデータ・セット・ラベルからの情報を DCB に転送する
- 標準ラベルを検査または作成する
- プログラマー作成の付加ルーチンをロードする

OPEN マクロ命令の 2 つのパラメーターは、初期化するデータ制御ブロックのアドレスと、データ・セットの入出力処理方式です。処理方式は INPUT、OUTPUT、UPDAT、または EXTEND を指定できます。ただし、何も指定されていない場合は、INPUT が想定されます。OPEN マクロ命令のパラメーターと各種の形式については、「z/OS DFSMS Macro Instructions for Data Sets」で説明しています。

ボリューム終了

EOV マクロ命令は、ボリューム終了 (end-of-volume (EOV)) およびデータ・セット終了 (end-of-data-set) 条件を識別します。ボリューム終了条件 (EOV) の場合、ボリュームを切り替え、標準ラベルを検査または作成します。データ・セット終了条件の場合 (別のデータ・セットが連結されている場合を除く)、EOV はユーザーのデータ・セット終了 (end-of-data-set) ルーチンに入ります。XDAP を使用している場合、EOV を発行するのは、直接アクセス・ボリュームの切り替えが必要な場合、または出力用にオープンされた直接アクセス・データ・セットに 2 次割り振りを行う必要がある場合です。EOV マクロ命令のパラメーターについては、109 ページの『ボリュームの終わり』を参照してください。

注: EOV が、同じボリュームまたは新規ボリューム上で拡張のために DADSM EXTEND を呼び出す必要がある場合 (これは DASD 出力を暗黙指定します)、EXTEND は SYSVTOC 上にエンキューを発行します。アプリケーションがすでに SYSVTOC エンキューを保持しているボリューム上のデータ・セットに対して EOV 要求が出されると、異常終了の原因になります。この場合の回避策は、2 次エクステントを必要としない十分な大きさの出力データ・セットを割り振るか、あるいは SYSVTOC エンキューを保持しているボリュームとは異なるボリューム上に出力データ・セットを置くように要求することです。

データ制御ブロックの復元

CLOSE マクロ命令は、1 つ以上のデータ制御ブロックを復元して、関連データ・セットの処理を終了できるようにします。直接アクセス・チャンネル・プログラムで使用されるすべてのデータ・セットに対して CLOSE を発行してください。CLOSE が実行されると、以下の手順のいくつかが実行されます。

- データ・エクステント・ブロックを解放する
- OPEN の実行時にデータ制御ブロック・フィールドに転送された情報を除去する
- 標準ラベルを検査または作成する
- プログラマー作成の付加ルーチンを解放する

CLOSE マクロ命令は、復元する少なくとも 1 つのデータ制御ブロックのアドレスを識別する必要があり、他のパラメーターも指定できます。CLOSE マクロ命令のパラメーターと各種の形式については、「z/OS DFSMS Macro Instructions for Data Sets」で説明しています。

コード	意味
R	ブロックを読み取ります。
W	ブロックを更新します。
V	装置がブロックの内容を読み取ることができるか検査しますが、データは転送しません。
I	識別番号によってブロックを検索します。(ブロックのキー部分 (存在する場合) およびデータ部分が、読み取り、更新、または検査されます。)
K	キーによってブロックを検索します。(ブロックのデータ部分のみが、読み取り、更新、または検査されます。) この値をコーディングする場合は、 <i>key_addr,keylength-value</i> オペランドをコーディングしてください。
<i>dcb_addr-A</i> タイプ・アドレスまたは (2-12) データ・セットのデータ制御ブロックのアドレス。	
<i>area_addr-A</i> タイプ・アドレスまたは (2-12) データ・セットのブロックの入力域または出力域のアドレス。	
<i>length_value-absexp</i> または (2-12) 入力域または出力域に転送されるバイト数。識別番号によってブロックの検索を行い、データ・セットにキーが含まれている場合には、値にキーの長さを含める必要があります。転送されるバイトの最大数は、32 767 です。	
<i>key_addr-RX</i> タイプ・アドレスまたは (2-12) キーによってブロックを検索する場合、読み取り、更新、または検査するブロックのキーを含んでいる仮想ストレージ・フィールドのアドレス。	
<i>keylength_value-absexp</i> または (2-12) キーによってブロックを検索する場合、キーの長さ。最大長は 255 バイトです。	
<i>blkref_addr-RX</i> タイプ・アドレスまたは (2-12) 検索するブロックを収容しているトラックの実トラック・アドレスが入っている、仮想ストレージ内のフィールドのアドレス。ブロックの実アドレスは、MBBCCHHR 形式です。ここで、	
M	データ・エクステント・ブロック内のエクステント項目は、直接アクセス・プログラムに関連していることを示します。
BB	ゼロでなければなりません。
CC	シリンダー・アドレス
HH	実トラック・アドレス
R	ブロック識別番号。キーによってブロックを検索する場合は、R は使用されません。

(詳しくは、150 ページの『相対トラック・アドレスの実トラック・アドレスへの変換』を参照してください。)

sector_addr-RX タイプ・アドレスまたは (2-12)
セクター値が入っている 1 バイト・フィールドのアドレス。*sector_addr* パラメーターは、回転位置感知機構 (RPS) 装置に使用されます。このパラメーターはオプションですが、これを使用するとチャンネルのパフォーマンスが向上します。このパラメーターがコーディングされている場合、*set-sector CCW* (データ・アドレス・フィールドに示されたセクター値を使用) をチャンネル・プログラムの *search-ID-equal* コマンドの前に置きます。*type* パラメ

ーターが RK、WK、または VK としてコーディングされている場合、*sector_addr* パラメーターは無視されます。実行形式のマクロでセクター・アドレスが指定されている場合、リスト形式にセクター・アドレス (必ずしも同じでなくても構いません) を指定する必要があります。実行可能形式のセクター・アドレスが使用されます。

例外: XDAP マクロが発行された場合、アドレスまたはセクター値に対する妥当性検査は行われません。ただし、セクター値が装置に対して有効でない場合、または RPS を備えていない装置に対して *sector_addr* オペランドが使用されている場合、チャンネル・プログラムの実行中に装置チェック/コマンド・リジェクト割り込みが発生します。

MF= パラメーター・リストから構成されるマクロ展開の場合は L 形式の XDAP マクロ命令を使用し、実行可能命令から構成されるマクロ展開の場合は E 形式を使用できます。

MF=E 第 1 オペランド (*ecb_symbol*) は必要であり、シンボルとしてコーディングするか、またはレジスター 2 から 12 に提供することができます。*type*、*dcb_addr*、*area_addr*、および *length_value* オペランドは、L 形式または E 形式で提供できます。*blkref_addr* オペランドは、E 形式で提供するか、あるいはプログラムによって IOB の IOBSEEK フィールドに移動することができます。*sector_addr* はオプションです。これは L 形式と E 形式のどちらでもコーディングでき、またコーディングしなくても構いません。

MF=L 最初の 2 つのオペランド (*ecb_symbol* と *type*) は必要で、シンボルとしてコーディングする必要があります。*length_value* または *keylength_value* をコーディングすることを選択した場合、これらは絶対式でなければなりません。その他のオペランド (コーディングする場合) は、A タイプ・アドレスでなければなりません。(*blkref_addr* は、コーディングされている場合、無視されます。)

XDAP マクロは、A タイプ・アドレスを含むチャンネル・プログラムを構築します。このアドレスは、L 形式のマクロ内のストレージを参照します。L 形式のマクロを作業域にコピーしてプログラムを再入可能にした場合、E 形式の XDAP マクロは A タイプ・アドレスを更新しません。その結果、無効なチャンネル・プログラムになります。

dcb_addr、*area_addr*、*blkref_addr*、および *sector_value* オペランドは、RX タイプ・アドレスとしてコーディングするか、またはレジスター 2 から 12 に提供することができます。*length_value* および *keylength_value* オペランドは、絶対式または 10 進整数として指定するか、レジスター 2 から 12 に提供することができます。

XDAP で使用される制御ブロック

XDAP マクロ命令の実行時に生成される制御ブロックは、以下のものから構成されます。

入出力ブロック

入出力ブロックは 40 バイトの長さで、イベント制御ブロックの直後に続きます。113 ページの『制御ブロック・フィールド』に、IOB の図が示されています

(128 ページの図 13)。ECB に 'X'41' が通知された場合、IOBSENS0、IOBSENS1、および IOBCSW フィールドを調べることが必要な場合があります。

イベント制御ブロック

イベント制御ブロック・フィールドについては、136 ページの『イベント制御ブロック・フィールド』を参照してください。

直接アクセス・チャンネル・プログラム

直接アクセス・チャンネル・プログラムは、24 バイトの長さ (RPS 装置に対して set sector が使用されている場合を除く) で、入出力ブロックの直後に続きます。

XDAP マクロ命令で指定された入出力操作のタイプに応じて、4 つのチャンネル・プログラムのいずれかを生成できます。4 つの可能なチャンネル・プログラムについて、それぞれの 3 つのチャンネル・コマンド・ワードを表 43 に示します。

表 43. XDAP で使用されるチャンネル・プログラム・コマンド・ワード

入出力操作のタイプ	CCW	コマンド・コード
識別番号による読み取り	1	Search ID Equal
	2	Transfer in Channel
	3	Read Key and Data
識別番号による検査	1	Search Key Equal
	2	Transfer in Channel
	3	Read Data
キーによる読み取り	1	Search ID Equal
	2	Transfer in Channel
	3	Write Key and Data
キーによる検査	1	Search Key Equal
	2	Transfer in Channel
	3	Write Data
識別番号による書き込み	1	Search ID Equal
	2	Transfer in Channel
	3	Write Key and Data
キーによる書き込み	1	Search Key Equal
	2	Transfer in Channel
	3	Write Data

注: 検査操作の場合、3 番目の CCW にフラグを立て、仮想ストレージへの情報の転送を抑制します。

RI、VI、または WI 操作でセクター・アドレスが指定されている場合、チャンネル・プログラムは 32 バイトの長さです。表 43 内の各チャンネル・プログラムの前に set sector コマンドが置かれます。

RPS 装置セクター番号

回転位置感知機構によって提供されるパフォーマンスの向上を実現するには、XDAP マクロで `sector_addr` パラメーターを指定します。RPS 装置および非 RPS 装置の両方で使用できるプログラムの場合、UCBRPS ビットをテストして、装置が回転位置感知機構を備えているかどうかを判別してください。UCBRPS ビットがオフの場合、その装置に対しては set sector コマンドを指定したチャンネル・プログラムを発行しないでください。

XDAP マクロの `sector_addr` パラメーターは、仮想ストレージ内の 1 バイト・フィールドのアドレスを指定します。このフィールドには、検索するブロックのセクター番号を保管します。ブロックのセクター番号は、常駐システムの変換ルーチンを使用して入手できます。154 ページの『RPS 装置上のブロックのセクター番号の入手』を参照してください。

第 6 章 パスワード保護データ・セットの使用

この章では、データ・セットのパスワード保護に関する情報を提供します。パスワード保護の使用はお勧めできませんが、他の IBM オペレーティング・システムとの互換性のために提供されています。代わりに、RACF 保護 (SAF を使用した) を使用してください。

この章で説明するパスワード保護は、ストレージ管理サブシステム (SMS) によって管理されるデータ・セットおよびカタログと、VSAM データ・セットに対しては適用されません。SMS はパスワードを無視します。また、PROTECT マクロと SVC は、動的として定義された装置上のボリュームはサポートしません。

SAF がアクティブの場合、パスワード保護はすべてのデータ・セットでバイパスされます。SAF が非アクティブで、アクセスされるデータ・セットが非 SMS 管理である場合にのみ、システムはパスワードの妥当性検査を実行します。システムは、システム許可機能 (SAF) インターフェースを通して、SMS 管理データ・セットおよびカタログの保護を提供します。SAF について詳しくは、「*MVS/ESA SML: Managing Data*」、*「z/OS MVS プログラミング: アセンブラー・サービス ガイド*」の『システム許可機能』、および「*z/OS MVS プログラミング: アセンブラー・サービス 解説書 ABE-HSP*」を参照してください。

VSAM データ・セット保護については、「*z/OS DFSMS データ・セットの使用法*」および「*z/OS DFSMS カタログのためのアクセス方式サービス・プログラム*」を参照してください。

パスワード保護の代わりに SAF を使用する理由として、次のものが挙げられます。

- パスワードを誰かに与えた場合、その人がそのパスワードを与える相手を制御できません。
- データ・セットはさまざまなパスワードを持つ傾向があり、それを書き留めておくことが必要になります。1 つの SAF パスワードを暗記できるならば、それに比べると安全性が劣ります。
- バッチ・ジョブ・アクセスまたは対話式の非 TSO アクセスでは、システム・オペレーターにパスワードを提供する必要があります。オペレーターへの通信は安全ではない可能性があります。ジョブの実行時にそのオペレーターが不在の場合もあります。オペレーターは各データ・セットのパスワードを別のオペレーターに与えることが必要になります。
- 各パスワードを提供する間、プログラムは一時停止します。これは、最近のシステムの自動化促進に逆行するものです。
- 特定のパスワードを誰が使用したかを知る方法がありません。
- 人の常として、特にパスワードが多い場合、パスワードを変更しない傾向があります。時間が経つにつれて、パスワードが公開される危険性が高まります。
- 少なからぬ数のデータ・セットがパスワードを持っている場合、システムが PASSWORD データ・セット項目を検索する時間が増大します。RACF を使用する

ると、増加はずっと少なく抑えることができます。RACF 総称プロファイルを使用すれば、新規データ・セットが同じプロファイルを使用している場合には、検索時間はまったく増えません。

- システム間で DASD を共用している場合、各システムのパスワード定義は独立しています。パスワードが同期しなくなる可能性があります。
- PASSWORD データ・セット項目には、データ・セット名が含まれていますが、ボリューム通し番号は含まれていません。パスワードを定義する前にデータ・セットを作成した場合、そのデータ・セット名に対して誰かがすでにパスワードを定義していることを見つける場合があります。データ・セットは、既存のパスワードをスクラッチするか、名前変更することが必要になります。
- パスワード保護は、システム管理ボリュームまたは動的装置に対してはサポートされません。

オペレーティング・システムのデータ・セット保護機能を使用するには、保護するデータ・セットの名前と各データ・セットに割り当てられたパスワードを関連付けたレコードから構成される PASSWORD データ・セットを作成し、保守します。PASSWORD データ・セットを保守するには、次の方法があります。

- ユーザー独自のルーチンを作成する
- PROTECT マクロ命令を使用する
- IEHPROGM ユーティリティー・プログラムのユーティリティー制御ステートメントを使用する
- TSO を持っている場合、TSO PROTECT コマンドを使用する

この章では、最初の 2 つの方法について説明します。最後の 2 つの方法については、以下にリストする資料で説明しています。

この章の情報を使用する前に、以下の資料の内容を十分に理解しておくことが必要です。

- 「z/OS DFSMS データ・セットの使用法」は、データ・セット保護機構について説明しています。
- 以下の資料は、データ・セット保護機構に関連したオペレーター・メッセージと応答について説明しています。
 - z/OS MVS システム・メッセージ 第 1 巻 (ABA-AOM)
 - z/OS MVS システム・メッセージ 第 2 巻 (ARC-ASA)
 - z/OS MVS システム・メッセージ 第 3 巻 (ASB-BPX)
 - z/OS MVS システム・メッセージ 第 4 巻 (CBD-DMO)
 - z/OS MVS システム・メッセージ 第 5 巻 (EDG-GFS)
 - z/OS MVS システム・メッセージ 第 6 巻 (GOS-IEA)
 - z/OS MVS システム・メッセージ 第 7 巻 (IEB-IEE)
 - z/OS MVS システム・メッセージ 第 8 巻 (IEF-IGD)
 - z/OS MVS システム・メッセージ 第 9 巻 (IGF-IWM)
 - z/OS MVS システム・メッセージ 第 10 巻 (IXC-IZP)

- 「z/OS MVS JCL 解説書」は、データ・セットをパスワード保護する必要があることを示すために使用されるデータ定義 (DD) ステートメント・パラメーターについて説明しています。これは、LABEL パラメーターのサブパラメーターです。
- 「z/OS DFSMSdfp ユーティリティー」は、IEHPROGM ユーティリティー・プログラムのユーティリティー制御ステートメントを使用して PASSWORD データ・セットを保守する方法について説明しています。
- 「z/OS TSO/E コマンド解説書」は、TSO PROTECT コマンドの使用法について説明しています。

データ・セット・セキュリティの提供

正しいデータ・セット名を使用しないとデータ・セットをオープンできないようにする通常のラベル保護に加えて、オペレーティング・システムは、機密データへの無許可アクセスを防止するデータ・セット・セキュリティ・オプションを提供しています。パスワード保護は、システム・オペレーターまたは (TSO の場合) リモート端末オペレーターが正しいパスワードを入力するまでは、データ・セットにアクセスするのを防止します。

パスワード保護データ・セットに対しては、以下のタイプのアクセスが許可されます。

- PWREAD/PWRITE-読み取りまたは書き込みアクセスのためにパスワードが必要です。
- PWREAD/NOWRITE-読み取りアクセスのためにパスワードが必要です。書き込みは許可されません。
- NOPWREAD/PWRITE-読み取りはパスワードなしで許可されます。書き込みにはパスワードが必要です。

データ・セット保護機構を使用する準備をするには、PASSWORD という名前の順次データ・セットをシステム常駐ボリュームに置きます。このデータ・セットは、保護下に置かれたデータ・セットごとに少なくとも 1 つの記録を含んでいる必要があります。各記録は、データ・セット名、そのデータ・セットのパスワード、カウンター・フィールド、保護モード標識、およびログに記録したい情報を記録するためのフィールドから構成されます。システム常駐ボリューム上で、これらの記録は、キー域 (データ・セット名とパスワード) とデータ域 (カウンター・フィールド、保護モード標識、およびロギング・フィールド) としてフォーマットされます。データ・セットは、キー域に基づいて検索されます。

- データ・セット用に割り振られる領域は、以前に PASSWORD データ・セット用に使用されてはなりません。これは、データ・セットに記録を追加する際に、予測不能の結果をもたらす原因になります。
- システム常駐ボリュームに PASSWORD データ・セットが含まれていない場合、システムはパスワード保護データ・セットへのアクセスを許可しません。これに依存して保護を行ってはいけません。システム常駐ボリューム上に PASSWORD という名前のデータ・セットを作成する人は誰でも、任意のデータ・セットのパスワードを定義できるからです。
- 磁気テープ上のデータ・セットは、標準ラベルが使用されている場合にのみ保護されます。

PASSWORD データ・セットを作成および保守するルーチンを書くことができます。PROTECT マクロ命令を使用して PASSWORD データ・セットを保守する方法については、171 ページの『PROTECT を使用した PASSWORD データ・セットの保守』を参照してください。IEHPRGM ユーティリティ・プログラムを使用して PASSWORD データ・セットを保守する方法については、「z/OS DFSMSdfp ユーティリティ」で説明しています。これらのルーチンは、ユーザー独自のライブラリーに置くこともできますし、システムのライブラリー (SYS1.LINKLIB) に入れることもできます。PASSWORD データ・セットの読み取りおよび書き込みは、データ管理アクセス方式を使用して行うことができます。

パスワード保護データ・セットは、プログラムが正しいパスワードを提供した場合にのみアクセスできます。保護データ・セットをオープンするための要求を受け取ると、オペレーティング・システムは、データ・セットがすでにこのジョブ・ステップ用にオープンされているかどうか、およびアクセス方式が前に使用された保護モードと互換性があるかどうかを検査します。どちらの条件も満たしていない場合、パスワードを要求するメッセージがオペレーター・コンソールに送られます。データをオープンしようとしているプログラムが、フォアグラウンドで TSO 下で実行されている場合、メッセージは TSO 端末オペレーターに送られます。

PASSWORD データ・セットの特性

PASSWORD データ・セットとオペレーティング・システムは同じボリューム上に常駐していなければなりません。そのボリュームは IPL ボリュームです。これは、システム常駐ボリュームと呼ばれます。PASSWORD データ・セットに割り振られるスペースは連続していなければなりません。割り振るスペースの量は、保護する必要があるデータ・セットの数によって決まります。PASSWORD データ・セットの各項目は、132 バイトのスペースを必要とします。PASSWORD データ・セットの編成は、物理順次です。レコード・フォーマットは非ブロック化固定長レコード (RECFM=F) です。データ・セットを構成する各レコードは、80 バイトの長さ (LRECL=80,BLKSIZE=80) で、前に 52 バイトのキー (KEYLEN=52) が置かれます。キー域は、完全修飾データ・セット名 (最大 44 バイトの長さ) とパスワードを含みます。1 から 8 文字の長さの英数字で表し、左揃えして、ブランクを追加して領域を埋めます。

制約事項: PASSWORD データ・セットをラージ・フォーマットまたは拡張フォーマットにすることはできません。

重要: 国際標準化機構 (ISO) 1001-1979 または米国規格協会 (ANSI) X3.27-1978 に準拠した磁気テープ上のデータ・セットは、世代別データ・セット名の一部として世代番号とバージョン番号が含まれていません。PASSWORD データ・セットに含まれている場合、これらのデータ・セットの世代番号とバージョン番号は無視されます。

プログラムが初期にデータ・セットを構築する際にパスワード・レコードを作成するか、または RACF を使用してアクセスを制御することによって、PASSWORD データ・セットを保護する必要があります。それ以降は、パスワードを入力しないと PASSWORD データ・セットをオープンできなくなります (データ・セットをスキップするオペレーティング・システム・ルーチンを除いて)。パスワード保護システム・データ・セットに問題が起きた場合、保守担当者はデータ・セットにアクセスして問題を解決するためにパスワードが必要になります。

保護データ・セットの作成

データ定義 (DD) ステートメントのパラメーター LABEL= を使用して、データ・セットをパスワード保護する必要があることを示すことができます。DASD 上のデータ・セットの場合、前に割り振られたデータ・セットに対する代替方式は、PROTECT マクロ命令、IEHPROGM ユーティリティ、または TSO PROTECT コマンドを使用することです。データ・セットを作成し、そのパスワード・レコードを PASSWORD データ・セットに入れずに、ラベルに保護標識を設定することができます。この場合、システムはそのデータ・セットに対するいかなるアクセスも許可しません。

ご使用のシステムの操作手順で、現在パスワード保護されているすべてのデータ・セットのパスワード・レコードが PASSWORD データ・セットに入っていることを確認する必要があります。独立したコンピューター・システムが共通の DASD リソースを共有している環境では、すべてのシステム上の PASSWORD データ・セットに、共有 DASD 上のすべての保護データ・セットの適切なパスワード・レコードが含まれていなければなりません。

特定の環境下では、共有 DASD 上の複数のシステムからデータ・セットが割り振りおよび割り振り解除される順序によって、保護が失われたり、データが破損したりする結果になることがあります。例えば、システム A のユーザーによってデータ・セットが割り振られてオープンされ、システム B 上の異なるユーザーによってスクラッチされた場合、最初のユーザーは割り振り解除された (フリー) 領域へのウィンドウを持っています。ウィンドウが開いている間に、いずれかのシステムのユーザーによってそのスペースにデータ・セット (保護または非保護) が割り振られた場合、新規データ・セットはウィンドウを持つユーザーからは保護されません。この問題の最も一般的な解決策は、GRS を使用することです (「z/OS MVS 計画: グローバル・リソース逐次化」を参照してください)。

割り振り処理がまだ NEW の間は、パスワード保護データ・セットはパスワードを提供せずに使用できます。データ・セットが割り振り解除された後は、データ・セットをオープンしようとする後続の試みは、パスワード・レコードが使用可能であり、正しいパスワードが提供されない限り、プログラムの終了という結果になります。保護モードが NOPWREAD で、要求が入力または逆方向読み取りのためのデータ・セットのオープンの場合は、パスワードは必要ありません。

複数のパスワード保護データ・セットを含むテープ・ボリューム

他のデータ・セットを含むテープ・ボリューム上のデータ・セットをパスワード保護するには、そのボリューム上のすべてのデータ・セットをパスワード保護してください。(標準ラベル-SL、SUL、AL、または AUL-が必要です。ラベル・タイプの定義および使用できる保護モード標識については、「z/OS DFSMS Using Magnetic Tapes」を参照してください。)

既存のパスワード保護データ・セットの後にデータ・セットを作成するために OPEN マクロ命令を発行した場合、新規データ・セットのオープン処理時に、既存のデータ・セットのパスワードが検査されます。提供されるパスワードは、PWWRITE 保護モード標識に関連付けられていなければなりません。

保護機構の操作特性

以下のトピックでは、保護機構の操作特性について説明します。

処理の終了

以下の場合、処理は終了します。

- オペレーターが 2 回の試行内に保護データの正しいパスワードを提供できない。
- オープンする保護データ・セットのパスワード・レコードが PASSWORD データ・セット内に存在しない。
- パスワード・レコード内の保護モード標識とオープン・ルーチンで指定された入出力処理方式が一致していない。例えば、読み取り専用保護モード標識に対して OUTPUT が指定された。
- ボリューム切り替え操作に関与するデータ・セットのデータ・セット名に不一致がある。これについては、以下で説明します。

ボリュームの切り替え

マルチボリューム・データ・セットのボリュームが切り替えられた場合、パスワード保護は保存されます。次の条件を満たす場合、システムは新しくマウントされたテープ・ボリュームを入力用に使用することを受け入れるか、または新しくマウントされた直接アクセス・ボリュームを受け入れます。

- データ・セット名が、データ・セットのパスワード・レコード内とジョブ・ファイル制御ブロック (JFCB) 内で同じである。(これにより、データ・セットがオープンされた以降に、問題プログラムが JFCB 内のデータ・セット名を変更しなかったことが保証されます。)
- パスワード・レコードの保護モード標識が処理モードと互換性があり、有効なパスワードが提供された。

以下のいずれの条件下でも、システムは新しくマウントされたテープ・ボリュームを出力用に使用することを受け入れます。

- HDR1 ラベルのセキュリティー標識がパスワード保護を示しており、パスワード・レコード内のデータ・セット名が JFCB 内のデータ・セット名と同じであり、保護モード標識が処理モードと互換性がある。(JFCB 内のデータ・セット名が変更された場合、オペレーターから新規パスワードが要求されます。)
- HDR1 ラベルのセキュリティー標識がパスワード保護を示していない。(パスワード保護を示すセキュリティー標識を持つ新しいラベルが書き込まれます。)
- ボリューム・ラベルのみが存在する。(パスワード保護を示すセキュリティー標識を持つ HDR1 ラベルが書き込まれます。)

連結データ・セット

データ・セットの連結に含まれている保護データ・セットごとにパスワードが要求されます。

SCRATCH および RENAME 機能

保護データ・セットを削除または名前変更するには、要求を行うジョブ・ステップがパスワードを提供できなければなりません。システムは、ジョブ・ステップが現

在、データ・セットへの書き込みを許可されているかどうかを検査します。ジョブ・ステップが許可されていない場合は、WRITE 保護モード標識に関連したパスワードを提供する必要があります。

カウンターの保守

オペレーティング・システムは、パスワードが使用されるたびに、パスワード・レコード内のカウンターを増分します。ただし、オーバーフロー標識は出しません (65535 回のオープンの後でオーバーフローします)。カウンター保守ルーチンを提供して、このカウンターをチェックし、必要であればリセットするようにしてください。

PROTECT を使用した PASSWORD データ・セットの保守

PROTECT マクロ命令を使用するには、PASSWORD データ・セットがシステム常駐ボリューム上になければなりません。PROTECT マクロを使用すると、以下のことができます。

- PASSWORD データ・セットに項目を追加する
- PASSWORD データ・セット内の項目を置き換える
- PASSWORD データ・セットから項目を削除する
- PASSWORD データ・セット内の項目に関する情報のリストを入手する。このリストには、セキュリティー・カウンター、アクセス・タイプ、および項目のデータ域内の 77 バイトのセキュリティー情報が含まれます。

PROTECT マクロは、保護された直接アクセス・データ・セットの DSCB も更新して、保護状況を反映させます。この機能により、データ・セットを保護する場合、ジョブ制御言語を使用する必要がなくなります。

PROTECT は、動的装置上のデータ・セットはサポートしません。

レコード・フォーマット

PROTECT マクロを使用する場合、PASSWORD データ・セットには、保護データ・セットごとに少なくとも 1 つのレコードが含まれていなければなりません。初めてデータ・セットを保護するときに割り当てるパスワード (キー域の最後の 8 バイト) は、制御パスワードと呼ばれます。

同じ保護データ・セットに対して、必要な数だけ 2 次レコードを作成できます。こうした追加レコードに割り当てるパスワードは、2 次パスワードと呼ばれます。この機能により、同じ保護データ・セットへのアクセスを複数のユーザーに許可すると同時に、その使用法を管理することが可能になります。例えば、あるユーザーには読み取り/書き込み許可を与え、別のユーザーには読み取りアクセスのみを許可するパスワードを与えることができます。

保護モード標識

パスワード・レコード内の保護モード標識 (3 番目のデータ・バイト) は、以下のいずれかの値に設定できます。

- X'00' は、パスワードは 2 次パスワードであり、保護データ・セットは読み取り専用 (PWREAD) であることを示します。

パスワード保護

- X'80' は、パスワードは制御パスワードであり、保護データ・セットは読み取り専用 (PWREAD) であることを示します。
- X'01' は、パスワードは 2 次パスワードであり、保護データ・セットは読み取りと書き込み (PWREAD/PWRITE) であることを示します。
- X'81' は、パスワードは制御パスワードであり、保護データ・セットは読み取りと書き込み (PWREAD/PWRITE) であることを示します。

データ・セットの保護状況の検査シーケンスは、次のようなデフォルトを生成します。

制御パスワード: 2 次パスワード:

- | | |
|--|-------------------------------------|
| 1. PWREAD/PWRITE または
PWREAD/NOWRITE | PWREAD/PWRITE または
PWREAD/NOWRITE |
| 2. NOPWREAD/PWRITE | NOPWREAD/PWRITE |

制御パスワードが項目 1 の設定値のいずれかに設定されている場合、2 次パスワードを NOPWREAD/PWRITE に設定しようとする、2 次パスワードは PWREAD/PWRITE に戻ります。

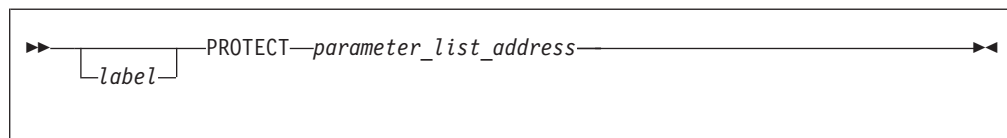
制御パスワードを項目 1 のいずれかの設定値から項目 2 の設定値に変更すると、2 次パスワードは NOPWREAD/PWRITE にリセットされます。

制御パスワードを項目 2 の設定値から項目 1 の設定値に変更すると、2 次パスワードは PWREAD/PWRITE に設定されます。

保護データ・セットの DSCB は、制御パスワードが変更された場合にのみ更新されるので、2 次パスワードの保護属性が制御パスワードの保護属性と矛盾することがあります。

PROTECT マクロ仕様

PROTECT マクロのフォーマットは、次のとおりです。



parameter list address-A タイプ・アドレス、(2-12)、または (1)

パラメーター・リストの場所を示します。パラメーター・リストは、PROTECT マクロを発行する前に作成しておく必要があります。パラメーター・リストのアドレスは、レジスター 1、レジスター 2 から 12 のいずれか、または A タイプ・アドレスとして渡すことができます。パラメーター・リストの最初のバイトを使用して、機能 (追加、置換、削除、リスト表示) を識別する必要があります。パラメーター・リストおよび機能を識別するために使用するコードについては、173 ページの図 17 から 176 ページの図 20 を参照してください。

要件: パラメーター・リストおよびリストによってアドレス指定される領域は、16 MB 境界より下の仮想ストレージになければなりません。システム常駐ボリュームの実 UCB が 16 MB 境界より上にある場合、あるいは PROTECT がデータ・セットの DSCB を更新しようとし、その UCB の 1 つが 16 MB 境界より上にある場合、PROTECT は要求に失敗します。

PROTECT マクロ・パラメーター・リスト

ADD 機能:

0 X'01'	13 制御パスワード・ポインター
1 00 00 00	16 ボリューム数
4 データ・セット名の長さ	17 ボリューム・リスト・ポインター
5 データ・セット名ポインター	20 保護コード
8 00	21 新規パスワード・ポインター
9 00 00 00	24 スtringの長さ
10 00	25 スtring・ポインター

図 17. ADD 機能のパラメーター・リスト

0 X'01' ADD 機能を示す項目コード。

4 データ・セット名の長さ。

5 データ・セット名ポインター。

13 制御パスワード・ポインター。

制御パスワードは、データ・セットが初めて保護下に置かれるときに割り当てられます。新規パスワードが制御パスワードである場合、ポインターは 3 バイトの 2 進ゼロに設定できます。

16 ボリューム数。

データ・セットがカタログされていない場合、保護としてフラグを立てるには、このフィールドにボリューム数を指定します。ゼロは、カタログ情報を使用することを要求します。

17 ボリューム・リスト・ポインター。

データ・セットがカタログされていない場合、保護としてフラグを立てるには、このフィールドにボリューム通し番号リストのアドレスを提供します。ゼロは、カタログ情報を使用することを要求します。

20 保護コード。

保護のタイプを示す 1 バイト数。X'00' は、デフォルトの保護を示します (ADD 機能の場合、デフォルトの保護は、データ・セットの制御パスワード・レコードで指定された保護タイプです)。X'01'

は、データ・セットが読み取りおよび書き込みされることを示します。X'02' は、データ・セットは読み取り専用であることを示します。X'03' は、データ・セットはパスワードなしで読み取ることができますが、書き込みアクセスの場合はパスワードが必要であることを示します。PROTECT マクロは、パラメーター・リストで指定された保護コード値を使用して、パスワード・レコードに保護モード標識を設定します。

21 新規パスワード・ポインター。

データ・セットが初めて保護下に置かれる場合、新規パスワードは制御パスワードになります。2 次項目を追加する場合は、新規パスワードは制御パスワードとは異なります。

24 スtringの長さ。

パスワード・レコードのオプションの情報フィールドに入れる文字Stringの長さ (最大 77 バイト)。情報を追加する必要がない場合は、このフィールドをゼロに設定してください。

25 String・ポインター。

オプションの情報フィールドに入れる文字Stringのアドレス。情報を追加する必要がない場合は、このフィールドをゼロに設定してください。

REPLACE 機能:

0	X'02'	13	制御パスワード・ポインター
1	00 00 00	16	ボリューム数
4	データ・セット名の長さ	17	ボリューム・リスト・ポインター
5	データ・セット名ポインター	20	保護コード
8	00	21	新規パスワード・ポインター
9	現行パスワードへのポインター	24	Stringの長さ
12	00	25	String・ポインター

図 18. REPLACE 機能のパラメーター・リスト

0 X'02'. REPLACE 機能を示す項目コード。

4 データ・セット名の長さ。

5 データ・セット名ポインター。

9 現行パスワードへのポインター

置き換えられるパスワードのアドレス。

13 制御パスワード・ポインター。

最初に保護下に置かれたときにデータ・セットに割り当てられたパ

パスワードのアドレス。現行パスワードが制御パスワードである場合、ポインタは 3 バイトの 2 進ゼロに設定できます。

16 ボリューム数。

データ・セットがカタログされていない場合、保護としてフラグを立てるには、このフィールドにボリューム数を指定します。ゼロは、カタログ情報を使用することを要求します。

17 ボリューム・リスト・ポインタ。

データ・セットがカタログされていない場合、保護としてフラグを立てるには、このフィールドにボリューム通し番号リストのアドレスを提供します。このフィールドがゼロの場合、カタログ情報が使用されます。

20 保護コード。

保護のタイプを示す 1 バイト数。X'00' は、デフォルトの保護を示します (REPLACE 機能の場合、デフォルトは、データ・セットの制御パスワード・レコードで指定された保護タイプです)。X'01' は、データ・セットが読み取りおよび書き込みされることを示します。X'02' は、データ・セットは読み取り専用であることを示します。X'03' は、データ・セットはパスワードなしで読み取ることができますが、書き込みアクセスの場合はパスワードが必要であることを示します。

21 新規パスワード・ポインタ。

現行パスワードを置き換えるために使用されるパスワードのアドレス。

24 スtringの長さ。

パスワード・レコードのオプションの情報フィールドに入れる文字 String の長さ (最大 77 バイト)。情報を追加する必要がない場合は、このフィールドをゼロに設定してください。

25 String・ポインタ。

オプションの情報フィールドに入れる文字 String のアドレス。情報を追加する必要がない場合は、このフィールドをゼロに設定してください。

DELETE 機能:

0	X'03'	9	現行パスワード・ポインタ
1	00 00 00	12	00
4	データ・セット名の長さ	13	制御パスワード・ポインタ
5	データ・セット名ポインタ	16	ボリューム数
8	00	17	ボリューム・リスト・ポインタ

図 19. DELETE 機能のパラメーター・リスト

- 0 X'03'. DELETE 機能を示す項目コード。
- 4 データ・セット名の長さ。
- 5 データ・セット名ポインタ。
- 9 現行パスワード・ポインタ。
削除するパスワードのアドレス。制御項目または 2 次項目を削除できます。
- 13 制御パスワード・ポインタ。
初めて保護下に置かれたときにデータ・セットに割り当てられたパスワードのアドレス。現行パスワードが制御パスワードでもある場合、ポインタは 2 バイトの 2 進ゼロに設定できます。
- 16 ボリューム数。
データ・セットがカタログされていない場合、保護としてフラグを立てるには、このフィールドにボリューム数を指定します。ゼロは、カタログ情報を使用することを要求します。
- 17 ボリューム・リスト・ポインタ。
データ・セットがカタログされていない場合、保護としてフラグを立てるには、このフィールドにボリューム通し番号リストのアドレスを提供します。ゼロは、カタログ情報を使用することを要求します。

LIST 機能:

0	X'04'	5	データ・セット名ポインタ
1	80 バイト・バッファ・ポインタ	8	00
4	データ・セット名の長さ	9	現行パスワード・ポインタ

図 20. LIST 機能のパラメーター・リスト

- 0 X'04'. LIST 機能を示す項目コード。
- 1 80 バイト・バッファ・ポインタ。
マクロ命令によってプログラムに戻される情報を受け取るバッファのアドレス。
- 4 データ・セット名の長さ。
- 5 データ・セット名ポインタ。
- 9 現行パスワード・ポインタ。
リストするレコードのパスワードのアドレス。

PROTECT マクロからの戻りコード

PROTECT マクロが処理を終了すると、レジスター 15 に以下の戻りコードのいずれかが入ります。

表 44. PROTECT 戻りコード

戻りコード	説明
0 (X'00')	PASSWORD データ・セットの更新が正常に完了しました。
4 (X'04')	データ・セット名のパスワードは、すでに PASSWORD データ・セット内にありました。
8 (X'08')	データ・セット名のパスワードが PASSWORD データ・セット内にありませんでした。
12 (X'0C')	制御パスワードが必要であるか、提供されたパスワードが誤っています。
16 (X'10')	提供されたパラメーター・リストが不完全であったか、誤っていました。
20 (X'14')	PASSWORD データ・セット内で入出力エラーがあったか、またはシステム常駐ボリューム (PASSWORD データ・セットを含む) が 16 MB 境界より上にある実 UCB を持っています。
24 (X'18') ¹	PASSWORD データ・セットが満杯です。
28 (X'1C')	バッファ・アドレスの妥当性検査が失敗しました。
32 (X'20') ²	LOCATE マクロが失敗しました。LOCATE の戻りコードがレジスター 1 に入り、検索された索引の数がレジスター 0 に入ります。
36 (X'24') ²	ユーザー・データ・セットに対する OBTAIN マクロが失敗しました。OBTAIN の戻りコードはレジスター 1 に入ります。ユーザーのデータ・セットが常駐するボリュームは、16 MB 境界より上にある実 UCB を持っています。レジスター 1 には 4 が入り、これは OBTAIN 戻りコードをシミュレートしています。
40 (X'28') ²	DSCB を更新できませんでした。
44 (X'2C')	PASSWORD データ・セットが存在しません。
48 (X'30')	磁気テープ・データ・セットを保護できません。
52 (X'32')	データ・セットは使用中です。
56 (X'38')	データ・セットは仮想ストレージ・アクセス方式 (VSAM) を使用しています。

注:

- PASSWORD データ・セットが満杯であることを示すメッセージがコンソールに書き出されます。
- PASSWORD データ・セットは更新されましたが、DSCB には、データ・セットの保護状況を示すフラグは立てられていません。

パスワード保護

第 7 章 システム・マクロ命令の使用

この章では、システム・マクロ命令に関する情報を提供します。マクロは、機能別にグループ化され (該当する場合)、以下の機能を実行します。

- データ・セキュリティーを確認する (DEBCHK マクロ)
- 制御ブロックおよびシステム・テーブルから装置特性を入手する (DEVTYPE マクロ)
- JFCB を変更する (RDJFCB および OPEN TYPE=J マクロ)
- 入出力アクティビティ・キューを操作する (PURGE および RESTORE マクロ)
- トラック容量の計算を行う (TRKCALC マクロ)

マクロの一部の機能は、システムの内部論理に依存する傾向があります。

この章をお読みにする前に、資料「*High Level Assembler/MVS & VM & VSE Language Reference*」の内容を十分に理解しておく必要があります。この資料は、アセンブラ言語でプログラムをコーディングするために必要な情報を提供しています。

データ・エクステント・ブロックの妥当性検査によるデータ・セキュリティーの確認

ユーザーのデータを無許可ユーザーの不注意または悪意によるアクセスから保護するのは、データ・エクステント・ブロック (DEB) の保護に依存しています。DEB にはデータ・セットがマウントされている装置に関する情報が含まれ、直接アクセス装置ストレージ・ボリューム上のデータ・セットの場所が記述されているので、これは非常に重要な制御ブロックです。

有効なシステム提供の DEB (通常は OPEN によって作成されます) のみがデータ管理機能に渡されることを確認するために、DEBCHK 検査機能が使用されます。OPEN は、作成した DEB のアドレスを DEB テーブルに置き、これが検査機能によって使用されます。DEB を作成するルーチンをコーディングする場合、作成する DEB のアドレスを DEB テーブルに追加してください。ルーチンに渡される DEB の妥当性に依存するルーチンをコーディングする場合、ルーチンに渡される DEB が DEB テーブル内に有効な項目を持っており、ユーザーの DCB またはアクセス方式制御ブロック (ACB) を指していることを確認してください。それぞれ、マクロの TYPE=ADD および TYPE=VERIFY オペランドを使用します。

非同期ルーチンが変更または削除したり、新規の DEB を DCB に割り当てたりするのを防止するには、ローカル・ロックを保持してください。この場合、DEBCHK 検査ルーチンへの分岐入力を使用し、レジスター 1 に戻された DEB アドレスを使用してください (DCB 内の DEB アドレスではなく)。プログラムがローカル・ロックを保持している限り DCB は有効です。つまり、信用できないプログラムが実行されるのを防止します。

DEBCHK マクロを呼び出す場合、プログラムは 24 ビットまたは 31 ビット・アドレスリング・モードで実行されている必要があります。

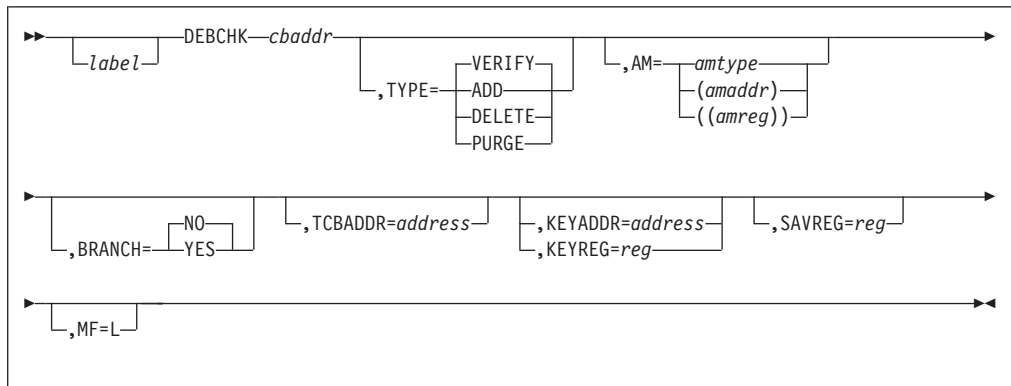
DEBCHK マクロ命令は、以下の 4 つの機能を提供します。

- DEB のアドレスを DEB テーブルに追加する。
- 特定のジョブ・ステップに関連した DEB テーブルに有効な DEB のアドレスが入っており、その DEB が DCB (または ACB) を指していることを検査する。問題プログラムはこの機能を使用して、DEB が有効かどうかを検査できます。
- DEB のアドレスを DEB テーブルから削除する。ユーザー DEB を削除するプログラムは、DEB を削除する前に、TYPE=DELETE オペランドを指定した DEBCHK マクロを発行して、DEB のアドレスを DEB テーブルから削除する必要があります。DEB 妥当性検査ルーチンが DEB テーブルからアドレスを削除しているときにエラーが発生すると、ジョブ・ステップは異常終了します。
- 前の機能と同じ方法で DEB テーブルから DEB のアドレスを削除する。ただし、この機能はジョブ・ステップを異常終了する代わりに、レジスター 15 にエラー・コードを戻すだけです。この機能は、異常終了が繰り返し発生するのを防止するために提供されています。

EXCP および EXCPVR で使用される DEB フィールドは、427 ページの『付録 B. 制御ブロック』に示されています (すべての DEB フィールドについては、『z/OS DFSMSdfp 診断解説書』で解説しています)。

DEBCHK マクロ仕様

DEBCHK マクロのフォーマットは、次のとおりです。



cbaddr 制御ブロック・アドレス。

BRANCH=NO の場合

RX タイプ・アドレス、(2-12)、または (1)

DEBCHK ルーチンに渡される制御ブロック・アドレス。MF=L がコーディングされている場合、このオペランドは無視されます。検査、追加、および削除要求の場合、*cbaddr* は、そのアドレスが DEB テーブル内にあるかどうかを検査する、DEB テーブルに追加する、または DEB テーブルから削除する DEB を指している、DCB または ACB のアドレスです。パーシ機能の場合、*cbaddr*

は、そのポインタをテーブルから消去する DEB のアドレスです。DCB または ACB は参照されません。

推奨: スプールされた DCB の DEB は、DCB を指さずに、スプールされた ACB を指します。この場合、DEBCHK は ACB に対して発行する必要があります。

BRANCH=YES の場合

検査する DEB を含んでいる DCB または ACB を指す、4 バイトの A タイプ・アドレス、またはレジスター (3-9) または (12)。

TYPE=VERIFY または ADD または DELETE または PURGE

実行される機能を示します。MF=L がコーディングされている場合、TYPE は無視されます。機能は、次のとおりです。

VERIFY TYPE オペランドがコーディングされていない場合、この機能が想定されます。制御プログラムは DEB テーブルをチェックして、DEB ポインタがテーブル内の DEB の DEBTBLOF フィールドで示された位置にあるかどうかを調べます。また、DEB をチェックして、DEBDCBAD が DEBCHK に渡された DCB (または ACB) を指しているかどうかも検査します。DEB 内の DEBAMTYP フィールドを AM オペランド値 (提供されている場合) と比較します。この 2 つは等しくなければなりません。TYPE=VERIFY は、監視プログラム状態または問題プログラム状態で発行できます。

ADD DEB アドレスを DEB テーブルに追加する前に、DEB と DCB (または ACB) は相互を指していなければなりません。DEB ポインタをテーブルに追加する前に、DEB 自体を現行 TCB DEB チェーンのキューに入れる必要があります (TCBDEB フィールドに、チェーンの最初の DEB のアドレスが入っています)。DEBCHK は、テーブル内の相対位置で DEB アドレスを DEB テーブルに追加します。DEBCHK は、DEB の DEBTBLOF フィールドに値を入れ、DEB の DEBAMTYP フィールドにアクセス方式タイプを挿入します。AM オペランドがコーディングされていない場合、DEBCHK は DEBAMTYP フィールドにゼロを入れます。TYPE=ADD は、監視プログラム状態でのみ発行できます。

DELETE DEB アドレスを DEB テーブルから削除する前に、DEB と DCB (または ACB) は相互を指していなければなりません。TYPE=DELETE は、監視プログラム状態でのみ発行できます。

PURGE DEBCHK は、DCB (または ACB) を検査せずに、

DEB ポインターを DEB テーブルから除去します。TYPE=PURGE は、監視プログラム状態でのみ発行できます。

AM アクセス方式値を指定します。各値は、特定のアクセス方式タイプに対応しています (BPAM と SAM は同じ値を持っていることに注意してください)。

値	タイプ
(X'00')	NONE
(X'01')	VSAM
(X'02')	EXCP
(X'04')	TCAM
(X'08')	GAM
(X'10')	TAM
(X'20')	BPAM
(X'20')	SAM
(X'40')	BDAM
(X'81')	SUBSYS
(X'84')	TCAMAP

オペランドは、以下の 3 通りの方法のいずれかでコーディングできますが、リスト形式 (MF=L) の命令の場合は、最初の方法のみが有効です。

amtype アクセス方式を参照します。つまり、BDAM、SAM、BPAM、TAM (これは BTAM のみを参照)、GAM、EXCP、または VSAM です。SUBSYS は、オペレーティング・システムのサブシステム (ジョブ入力サブシステムなど) を識別します。NONE は、アクセス方式またはサブシステムが指定されていないことを示します。

(amaddr) アクセス方式値の RS タイプ・アドレス。MF=L が使用されている場合、このフォーマットはコーディングできません。

((amreg)) 下位バイト (ビット位置 24 から 31) にアクセス方式値を含んでいる汎用レジスタ 1 から 14 のいずれか。高位バイトは検査されません。MF=L がコーディングされている場合、この形式は使用できません。

amaddr および *amreg* の使用は、前に MF=L 形式の DEBCHK によってアクセス方式値が生成された場合に限定する必要があります。MF=L がコーディングされていない場合、AM オペランドの重要度は TYPE によって異なります。

TYPE が ADD で、AM が指定されている場合、アクセス方式値は DEB の DEBAMTYP フィールドに挿入され、この DEB を参照する後続の DEBCHK マクロはすべて同じ AM を指定するか、または AM オペランドを省略するかのいずれかでなければなりません。TYPE=ADD で AM オペランドが省略されている場合、ヌル値

(0) が DEB に入れられ、後続の DEBCHK マクロはすべて AM オペランドを省略する必要があります。

TYPE が PURGE、DELETE、または VERIFY のときに AM オペランドが指定されている場合、アクセス方式値は DEB の DEBAMTYP フィールドの値と比較されます。AM が省略されている場合は、比較は行われません。

BRANCH=NO または **YES**

DEBCHK 検査ルーチンへの分岐入力を使用するかどうかを指定します。

NO

SAVREG

TCBADDR

KEYADDR

KEYREG

分岐入力を使用しないことを指定します。プログラムはオペランド SAVREG、TCBADDR、KEYADDR、および KEYREG を無視します。プログラムは TCB モードで実行されていなければなりません。

YES 分岐入力を使用することを指定します。TYPE=VERIFY を暗黙的または明示的に指定する必要があります。オペランド TCBADDR と KEYADDR/KEYREG が必要です。AM と MF は無視されます。BRANCH=YES の場合の注記:

- プログラムは TCB または SRB モードで実行できます。
- SAVREG= に対しては、レジスター 1、2、10、11、14、および 15 を使用してはなりません。
- *cbaddr*、TCBADDR=、または KEYADDR=/KEYREG= に対しては、レジスター 1、2、10、11、14、15、および SAVREG= に対して指定されたレジスターは使用してはなりません。
- 完了時のレジスター 10、11、および 14 の内容は予測不能です。また、SAVREG= を指定しない場合、レジスター 2 の内容も予測不能です。
- 完了時に、レジスター 1 には DEB のアドレスが入り、レジスター 15 には 0、4、または 16 が入ります (コードとその意味については、184 ページの『DEBCHK からの戻りコード』を参照してください)。
- 24 ビットまたは 31 ビット・アドレッシング・モードで操作している場合に指定できます。

TCBADDR=address-RX タイプ・アドレス、(3-9)、または (12)

DEBCHK 検査ルーチンによって使用される TCB のアドレスを含んでいるワードまたはレジスターを指定します。このオペランドは BRANCH=YES でのみ使用します。

KEYADDR=address-RX タイプ・アドレス

DCB (または ACB) にアクセスするときに使用されるキーを含んで

いるバイトの位置、またはその位置を指すレジスターを指定します。記憶保護キーは、ビット 0 から 3 です。このオペランドは **BRANCH=YES** でのみ使用します。

KEYREG=reg ビット位置 24-27 に、DCB (または ACB) にアクセスするときに使用されるキー値を含んでいるレジスターを指定します。このオペランドは **BRANCH=YES** でのみ使用します。

SAVREG=reg レジスター 2 の保管先レジスターを指定します。このオペランドは **BRANCH=YES** でのみ使用します。

MF=L リスト形式の **DEBCHK** マクロ命令を示します。MF=L がコーディングされている場合、AM キーワードに対応するアクセス方式値からなるパラメーター・リストが作成されます。この値は、AM=(amaddr) をコーディングして、別の **DEBCHK** 内で名前によって参照することもできますし、AM=((amreg)) をコーディングして、別の **DEBCHK** マクロを発行する前にレジスターの下位バイトに挿入することもできます。

DEBCHK からの戻りコード

レジスター 15 には、次のコードのいずれかが入ります。

戻りコード	意味
0 (X'00')	要求された機能は正常に完了し、レジスター 1 に DEB のアドレスが入ります。
4 (X'04')	次のいずれかです。(a) ジョブ・ステップに関連した DEB テーブルが存在しない、または (b) DEB テーブルへの指標である DEB フィールドが無効値を含んでいる、または (c) 制御ブロック・アドレスが DEB テーブル項目の内容と同じでない。
8 (X'08')	無効な TYPE が指定されました。(マクロによってではなく、分岐によって DEBCHK ルーチンに入りました。)
12 (X'0C')	プログラムが許可されておらず、 TYPE が VERIFY ではありませんでした。
16 (X'10')	DEBDCBAD は、 DEBCHK ルーチンに渡された DCB (または ACB) のアドレスを含んでいませんでした。
20 (X'14')	AM 値が DEBAMTYP フィールドの値と等しくありません。
24 (X'18')	DEB が DEB チェーン上になく、 TYPE=ADD が指定されました。
28 (X'1C')	すでに DEB テーブルに入れられた DEB に対して TYPE=ADD が指定されました。
32 (X'20')	DEB テーブルは最大サイズを超過し、 TYPE=ADD でした。

入出力装置特性の入手

DEVTYPE マクロ命令を使用して、入出力装置の特性に関連した情報を要求し、この情報を指定された領域に入れます。(チェックポイントが取られる前に実行された **DEVTYPE** マクロ命令の結果は、チェックポイント/リスタートが生じた後は有効と見なしてはなりません。) **IHADVA** マクロは、**DEVTYPE** マクロによって戻されたデータをマップします。

以下のトピックでは、**DEVTYPE** マクロ、装置特性、および特定の装置の出力について説明します。

プログラムは、24 ビットまたは 31 ビット・モードで実行されているときに DEVTYPE マクロを発行できます。プログラムが 31 ビット・モードで実行されている場合、パラメーター・リスト、情報リスト、および UCB アドレス・リストは 16 MB 境界より上に置くことができます。

200 ページの表 51 は、INFOLIST パラメーターを付けずに DEVTYPE マクロを発行した結果として得られる各装置タイプの出力を示しています。

現在サポートされる装置のすべてについて、DEVTYPE は、スペース計算を行うための十分な情報を戻すわけではありません。スペース計算は、TRKCALC を使用して行う必要があります。TRKCALC マクロの使用については、229 ページの『トラック計算の実行』を参照してください。

DEVTYPE マクロ仕様

DEVTYPE マクロ呼び出しには、4 通りの形式があります。標準形式、実行形式、リスト形式、および INFO 形式です。これらの形式について、以下で説明します。

制約事項:

- UCBLIST パラメーターで BELOW または ANY 値をコーディングしない場合は、DFSMS のどのリリースでも、本書の説明どおりに DEVTYPE をアセンブルして実行できます。
- UCBLIST パラメーターで BELOW または ANY 値をコーディングする場合、MVS/DFP™ バージョン 3 ではプログラムをアセンブルすることができません。ただし、DFSMS 上でそれをアセンブルした場合には、システムは BELOW または ANY 値を無視します。
- INFOLIST または INFO パラメーターをコーディングする場合、MVS/DFP バージョン 3 ではマクロをアセンブルまたは実行することはできません。
- INFOLIST、INFO、BELOW、ANY、および PLISTVER パラメーターを省略する場合、DFSMS 上でプログラムをアセンブルできますが、MVS/DFP バージョン 3 でそれを実行することはできません。PLISTVER=0 をコーディングする場合、DFSMS 上ではプログラムをアセンブルできますが、MVS/DFP 上ではできません。プログラムの実行は、いずれのシステム・レベルでも可能です。プログラムのアセンブリー時に、241 ページの『アセンブラー・マクロ・フェーズでの DFARELS の判別』で説明する手法を使用して、コーディングするパラメーターを選択できます。

標準形式の DEVTYPE マクロの呼び出しには、2 つのタイプがあります。

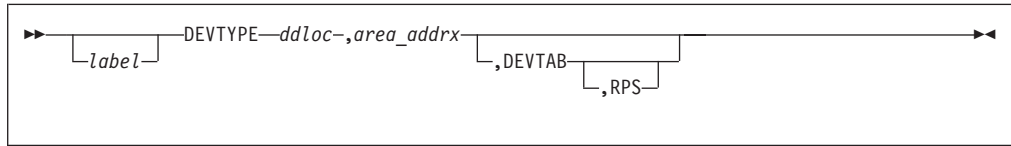
- 最小タイプの呼び出しは、装置の DD ステートメントを参照します。リスト形式または実行形式はありません。
- UCBLIST/INFOLIST タイプの呼び出しは、UCBLIST または INFOLIST パラメーターのどちらか一方または両方を指定する必要があります。

標準形式の DEVTYPE マクロの呼び出しの各種のフォーマットを、以下に示します。フォーマットの後には、パラメーターの説明を記載します。

最小タイプの呼び出し

最小タイプの呼び出しは、装置の DD ステートメントを参照します。このタイプの呼び出しは、パラメーターを汎用レジスターで受け渡しするので、リスト形式または実行形式は存在しません。ユーザーが指定した領域に装置情報を戻します。

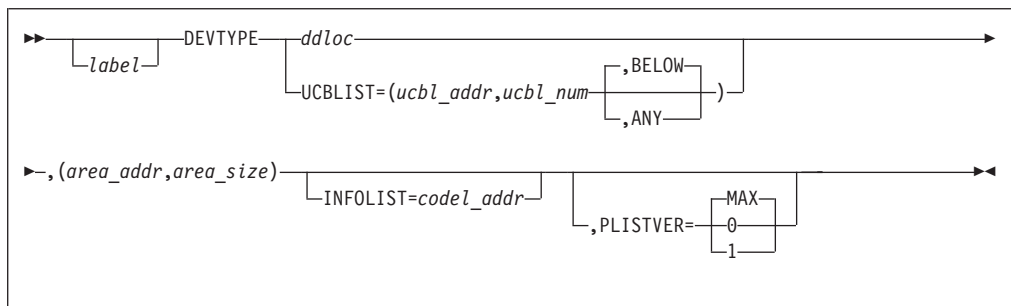
DEVTYPE マクロの最小タイプの呼び出しのフォーマットは、次のとおりです。



UCBLIST または INFOLIST タイプの呼び出し

UCBLIST または INFOLIST タイプの呼び出しは、UCBLIST または INFOLIST パラメーターのどちらか一方または両方を指定する必要があります。

UCBLIST または INFOLIST タイプの呼び出しのフォーマットは、次のとおりです。



ddloc - A タイプ・アドレス、または (1-12)

装置が割り当てられている DD ステートメントのシンボル名を含む 8 バイト・フィールドの名前。名前は 8 バイト・フィールド内に左揃えし、名前が 8 文字より少ない場合は、後を空白で埋める必要があります。各 DEVTYPE マクロの実行可能呼び出しは、割り振られたデータ・セットを識別するための *ddloc*、または 1 つ以上の装置を識別するための UCBLIST= を含んでいなければなりません。

すべてのキーワードを省略した場合、*ddloc* は (1) のみを指定できます。

area_addrx - Rx タイプ・アドレス、または (0, 2-12)

area_addr - A タイプ・アドレス、または (2-12)

装置情報を入れる領域の名前。UCBLIST または INFOLIST 機能を指定しない場合、DEVTAB および RPS オペランドを指定したかどうかに応じて、領域は 2、5、または 6 ワードの長さになります。プログラムが INFOLIST を指定せずに UCBLIST パラメーターを指定した場合、領域は UCB ごとに 6 ワードの長さが必要です。領域はワード境界上になければなりません。

INFOLIST パラメーターを指定した場合、領域の長さは、参照された DEVTYPE マクロの INFO のコーディングによって決まります。INFO の記述は、INFO リストに指定した各値に戻されるバイト数を示します。必要な領域サイズを計算するには、これらの値の合計に UCBLIST 内の *ucbl_num* を乗算します。UCBLIST (および *ddloc* の指定) を省略した場合は、乗算しないでください。

UCBLIST を指定する (*ddloc* は指定しない) 場合、(*area_addr,area_size*) は 2 番目の定位置パラメーターのままにしなければならないことに注意してください。コンマの後にそれを指定してください。

area_size - 絶対式、または (2-12)

装置情報を入れる領域のサイズ (バイト)。表 45 を参照してください。

表 45. 領域の最小サイズ

ddloc 指定	UCBLIST 指定	INFOLIST 指定	領域の最小サイズ
はい	いいえ	省略または 0	DEVTAB と RPS がコーディングされたかどうかに応じて、8、20、または 24 バイト。
はい	いいえ	はい	INFO で指定した各コードに戻されるバイト数の合計。INFO キーワードも参照。
いいえ	はい	省略または 0	UCB 当たり 24 バイト
いいえ	はい	はい	UCBLIST で指定した UCB の数と、INFO で指定した各コードに戻されるバイト数の合計との積。INFO キーワードも参照。

DEVTAB[,RPS]

DEVTAB は、直接アクセス装置に対してのみ意味を持ちます。DEVTAB が指定されている場合、ユーザーの領域には以下のワード数の情報が入ります。

- 直接アクセス装置の場合: 5 ワード
- 直接アクセス装置以外の場合: 2 ワード

DEVTAB, INFOLIST, または UCBLIST を指定しない場合、グラフィックス装置またはテレプロセシング装置を参照する場合は、1 ワードの情報が領域に入ります。その他のタイプの装置の場合は、2 ワードの情報が領域に入ります。

RPS

RPS を指定する場合、DEV TAB も指定する必要があります。
RPS パラメーターは、追加の 1 ワードの回転位置感知機構情報を DEV TAB 情報に含めます。

UCBLIST=(ucbl_addr,ucbl_num [,BELOW または ANY])

UCBLIST はリスト・サービスを提供し、これを使用して呼び出し側は 4 バイト UCB アドレスのリストを渡し、リスト内の UCB アドレス項目の数を指定します。UCBLIST パラメーターまたは *ddloc* を指定する必要があります。BELOW または ANY は、UCB パラメーターで受け渡されるアドレスに 3 バイトまたは 4 バイト UCB アドレスが含まれるかどうかを示すオプション・キーワードです。このキーワードは、AMODE 31 で実行されている呼び出し側にのみ適用されます。呼び出し側が AMODE 24 で実行されている場合、キーワード ANY は無視され、高位バイトは X'00' として扱われます。

INFOLIST を指定しない場合、装置タイプに関係なく、戻される情報は常に 6 ワード項目 (各 UCB アドレスにつき 1 項目) に戻されます。その項目の装置には適用されない情報を含んでいるワードは、変更されません。

DEVTYPE マクロは、UCBLIST = パラメーターを介して、16 MB 境界より上または下のキャプチャーされた UCB または実 UCB アドレスを受け入れます。UCB コピーの 24 ビットまたは 31 ビット・アドレスも受け入れます。UCB コピーはワード境界上になければなりませんが、16 MB 境界より上でも下でも構いません。無許可プログラムは、UCBSCAN マクロを使用し、COPY および UCBAREA キーワードを指定することにより、UCB のコピーを入手できます。詳細については、「z/OS ハードウェア構成定義(HCD) 計画」を参照してください。

ucbl_addr - A タイプ・アドレス、または (2-12)

4 バイト UCB アドレス・リストを含んでいる領域の名前。

ucbl_num - 絶対式、または (2-12)

リスト内の 4 バイト UCB アドレス項目の数。

BELOW

UCB パラメーターは、16 MB より下のストレージにある UCB のアドレス、またはキャプチャーされた UCB を含んでいます。これはデフォルトです。BELOW が指定されている場合、UCB アドレスの高位バイトは X'00' として扱われます。

ANY UCB パラメーターは、4 バイト UCB アドレスを含んでいます。31 ビット・モードで呼び出すときに ANY が指定されている場合、DEVTYPE は UCB アドレス・リスト内の各ワードを 31 ビット・アドレスとして扱います。

INFOLIST=codel_addr

DEVTYPE が戻す情報のタイプを指定する領域の名前。INFOLIST=0

をコーディングすると、省略した場合と同じ効果を持ちます。
 INFOLIST を指定する場合は、*ddloc* または UCBLIST も指定する必要があり、(*area_addr,area_size*) を指定しなければなりません。
 戻されるワードのフォーマットについては、DEVTYPE、DASD、および SUFFIX の項で説明します (195 ページを参照)。

code1_addr-A タイプ・アドレス、または (2-12)

INFO キーワードのみをコーディングした DEVTYPE マクロのインスタンスを入れる領域を指定します。

PLISTVER=0 または **1** または **MAX**

生成するマクロのパラメーター・リストのバージョンを指定します。

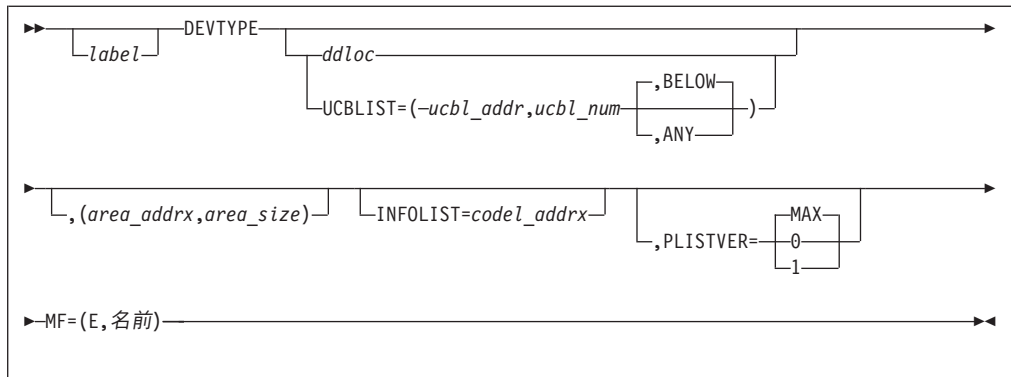
0 プログラムは、DFSMS リリース 3 より前のレベルのシステム上ではアセンブルできませんが、コーディングされたパラメーターをサポートする任意のシステム・レベルで実行できます。INFOLIST または INFO パラメーターを使用してこれをコーディングすることはできません。UCBLIST の 3 番目の値でコーディングすることができますが、MVS/DFP はその 3 番目の値を無視します。パラメーター・リストは 20 バイトの長さです。

1 プログラムは、DFSMS バージョン 1.1 以降でのみアセンブルおよび実行できることを指定します。DFSMS より前のパラメーター・リストは受け入れられません。この指定は、24 バイトのパラメーター・リストを生成します。

MAX 現行リリースでは、これは 1 をコーディングするのと同じ効果を持ちます。将来のリリースでは、この値は以前のリリースとは非互換のパラメーター・リストを許容するようになります。これはデフォルトで、24 バイトのパラメーター・リストを生成します。

DEVTYPE - 実行形式

実行形式の DEVTYPE マクロは、次のとおりです。



ddloc - RX タイプ・アドレス、または (1-12)

標準形式のマクロと同じ意味を持ちます (*ddloc* パラメーターの実行形式の要件については、下記の UCBLIST パラメーターも参照してください)。

area_addrx - RX タイプ・アドレス、または (2-12)

標準形式のマクロと同じ意味を持ちます。このパラメーターは、リスト形式および/または実行形式でコーディングする必要があります。

これは実行形式でコーディングする必要があります。

area_size - RX タイプ・アドレス、または (2-12)

この形式は、標準形式のマクロと同じ意味を持ちます。このパラメーターは、リスト形式および/または実行形式でコーディングする必要があります。

これは実行形式でコーディングする必要があります。

UCBLIST=(*ucbl_addrx*,*ucbl_num* ,BELOW** または **ANY**)**

このフォーマットは標準形式のマクロと同じ意味を持ちます。

UCBLIST または *ddloc* パラメーターのいずれかをリスト形式または実行形式で指定する必要があります。両方の形式でコーディングすることもできます。いずれの場合も、実行形式のパラメーターはリスト形式の同じパラメーターをオーバーライドします。

ucbl_addrx - RX タイプ・アドレス、または (2-12)

このフォーマットは標準形式のマクロと同じ意味を持ちます。

ucbl_num - 絶対式、または (2-12)

このフォーマットは標準形式のマクロと同じ意味を持ちますが、例外として、実行形式では絶対式の最大値は 4095 になります。より大きな値をレジスターに提供できます。

BELOW

このフォーマットは標準形式のマクロと同じ意味を持ちます。実行形式で UCBLIST に 2 つまたは 3 つの値を書き込むと、リスト形式の 3 つの値をすべて置き換え、オーバーライドします。実行形式で UCBLIST を省略すると、DEVTYPE はリスト形式の値を使用します。これはデフォルトです。

ANY このフォーマットは標準形式のマクロと同じ意味を持ちます。実行形式で UCBLIST に 2 つまたは 3 つの値を書き込むと、リスト形式の 3 つの値をすべて置き換え、オーバーライドします。実行形式で UCBLIST を省略すると、DEVTYPE はリスト形式の値を使用します。

INFOLIST=codel_addrx

このフォーマットは標準形式のマクロと同じ意味を持ちます。リスト形式内の前の INFOLIST 値を除去するには、INFOLIST=0 をコーディングします。

codel_addrx - RX タイプ・アドレス、または (2-12)

このフォーマットは標準形式のマクロと同じ意味を持ちます。

PLISTVER=0 または 1 または MAX

このフォーマットは標準形式のマクロと同じ意味を持ちます。リスト形式で PLISTVER に対してデフォルト以外の値をコーディングした場合は、実行形式も同じ値をコーディングしてください。

MF=(E,name) 実行形式の DEVTYPE を指定します。実行形式の DEVTYPE マクロは、UCBLIST または INFOLIST 機能を使用する場合にのみ有効です。

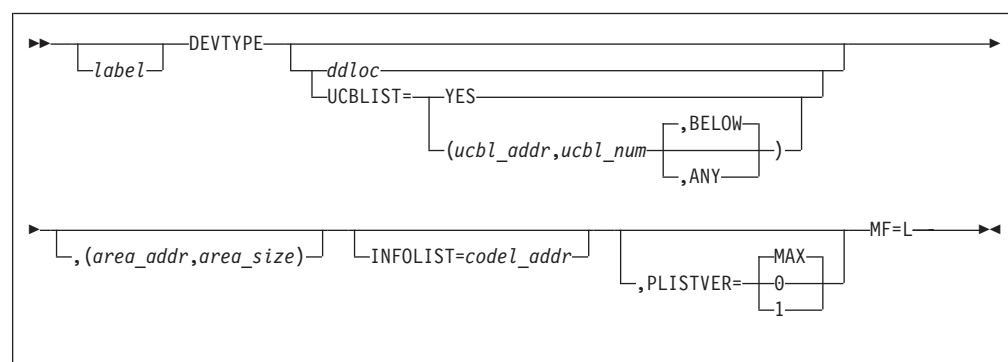
E 実行形式のマクロを指定します。実行形式を使用してパラメータ・リストを変更し、DEVTYPE 機能呼び出します。

name - RX タイプ・アドレス、または (1-12)

対応する MF=L 形式によって作成されたパラメータ・リストのラベル。

DEVTYPE - リスト形式

リスト形式の DEVTYPE マクロは、次のとおりです。



label 使用するパラメータ・リストのラベル。MF=L で *label* を指定する場合、*label* は MF=(E,name) の *name* と同じでなければなりません。

ddloc-A タイプ・アドレス

このフォーマットは標準形式のマクロと同じ意味を持ちます。

UCBLIST=YES または (*ucbl_addr,ucbl_num,BELOW* または **ANY**)

このフォーマットは標準形式のマクロと同じ意味を持ちます。これはデフォルトです。

YES MF=L がコーディングされている場合、DEVTYPE は UCBLIST=YES をプレースホルダーとして許可します。マクロ展開には影響を与えません。

ucbl_addr-A タイプ・アドレス

このフォーマットは標準形式のマクロと同じ意味を持ちます。

ucbl_num-絶対式

このフォーマットは標準形式のマクロと同じ意味を持ちます。

BELOW

このフォーマットは標準形式のマクロと同じ意味を持ちます。これはデフォルトです。

ANY このフォーマットは標準形式のマクロと同じ意味を持ちます。

UCBLIST=YES のコーディングは、マクロ展開には影響を与えません。同じマクロの DD 名パラメーターでは、UCBLIST=YES をコーディングしないでください。

area_addr-A タイプ・アドレス

このフォーマットは標準形式のマクロと同じ意味を持ちます。このパラメーターは、リスト形式および/または実行形式でコーディングする必要があります。

area_size-絶対式

このフォーマットは標準形式のマクロと同じ意味を持ちます。このパラメーターは、リスト形式および/または実行形式でコーディングする必要があります。

INFOLIST=codel_addr

このフォーマットは標準形式のマクロと同じ意味を持ちます。

codel_addr-A タイプ・アドレス、または (2-12)

このフォーマットは標準形式のマクロと同じ意味を持ちます。

PLISTVER=0 または **1** または **MAX**

このフォーマットは標準形式のマクロと同じ意味を持ちます。

リスト形式で PLISTVER に対してデフォルト以外の値をコーディングした場合は、実行形式も同じ値をコーディングしてください。

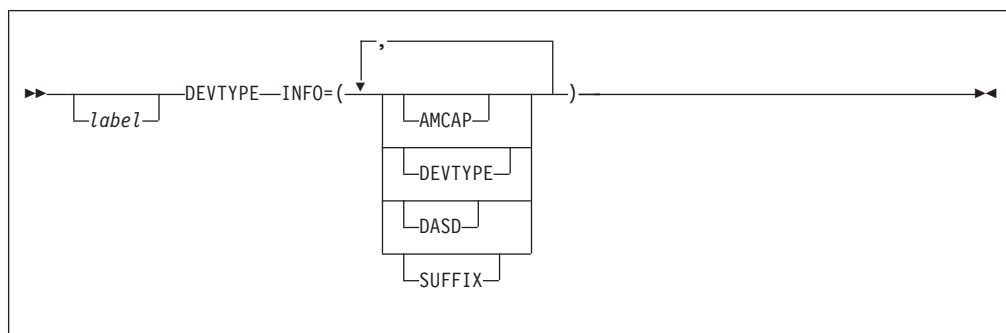
MF=L

リスト形式の DEVTYPE を指定します。リスト形式の DEVTYPE マクロは、UCBLIST または INFOLIST 機能を使用する場合にのみ有効です。MF=L を指定してパラメーター・リストを作成し、後で実行形式のマクロを指定して値を提供できます。

DEVTYPE - INFO 形式

INFO 形式の DEVTYPE マクロは、INFOLIST パラメーターのコード・リストを生成するのに使用できます。INFO 形式は、実行可能ではありません。リスト形式または実行形式はありません。DEVTYPE の INFOLIST パラメーターを指定した場合、INFO パラメーターのみを持つ DEVTYPE の展開を参照します。

INFO 形式の DEVTYPE マクロは、次のとおりです。



INFO=(AMCAP または DEVTYPE または DASD または SUFFIX)

検索する情報のタイプを指定します。示されている値の任意の組み合わせを任意の順序で指定してください。リストには、同じ値を複数回指定できます。INFO 値を囲む括弧は、値が 1 つだけのときは省略できます。

INFO キーワードを指定すると、マクロ展開はリストになり、実行可能でなくなります。INFO キーワードを指定する場合は、MF キーワードを省略するか、または MF=L を指定してください。E 形式はこれらの値を更新しないことに注意してください。このことは、INFOLIST を指定して、再入可能 CSECT でアセンブルされた DEVTYPE 展開を参照できることを意味しています。

DEVTYPE は、ユーザーが *area_addrx* パラメーターで提供した領域に情報を戻します。ユーザーが要求する各タイプの情報は、固定長です (下記を参照)。実行時に、DEVTYPE マクロは領域の長さが十分かどうかを検査します。情報は、ユーザーが INFO= キーワードに値を指定した順序で戻されます。要求された情報が装置またはデータ・セットにとって意味を持たない場合、DEVTYPE は該当するストレージをクリアします。これは、情報の各部分が、予測可能なオフセットで領域内に戻されることを意味します。

IBM では、システムの将来のレベルで DEVTYPE に対して新規 INFO= コードのサポートを追加できます。現行レベルの DEVTYPE は、そうした将来のレベルでアセンブルされたオブジェクト・コードを許容できることが予想されます。現行レベルの DEVTYPE は、適切なバイト数を戻すものと思われます。これは、2 進ゼロ、または右側に 2 進ゼロが埋め込まれた部分情報になります。いずれの場合も、DEVTYPE はこの条件を示すために、戻りコード=0 と理由コード=4 も設定します。将来のコードによって戻された情報が有効

にゼロに設定されない場合は、ゼロをテストすると、サポートされないフィールドを判別できます。DFA をテストすれば、システムのレベルも判別できます。

DEVTYPE は、ソース・コードの下位互換性をサポートするようには設計されていません。つまり、現行レベルのシステム上で、将来の INFO 値を含むソース・コードをアセンブルすることはできません。

AMCAP アクセス方式容量の 32 バイトを戻します。下の表を参照してください。

表 46. INFO=AMCAP 32 バイトの戻りデータ

オフセット	バイト	説明
0(0)	1	フラグ。
0(0)	1... ..	BSAM、QSAM、および (DASD の場合) BPAM は、大容量ブロック・インターフェースをサポートし、ブロック・サイズ限度が、次のダブルワードに入っています。
1(1)	7	予約済み。現在はゼロに設定。
8(8)	8	サポートされる最大ブロック・サイズ。データ・セットの連結のために DEVTYPE に対して DD 名を指定する場合、この値はすべての DD の中の最大の値になります。磁気テープまたはダミー・データ・セットの場合、この値が 32760 を超えることがあるため、EXCP またはアクセス方式大容量ブロック・インターフェースが必要になります。EXCP を使用する場合を除いて、出力時に OPEN はこの値を超えるブロック・サイズを許可しません。ある種のカートリッジ磁気テープ・ドライブでは、この限度を超えると、ハードウェア・バッファリングをバイパスする原因になります。IBM では、今後、他の装置タイプに対しても 32760 を超える値をサポートする考えです。
16(10)	8	推奨最大ブロック・サイズ。これは、サポートされる最大ブロック・サイズ以下の値です。この長さを超えると、装置は効率が低下するか、または信頼性が低下します。データ・セットの連結のために DEVTYPE に対して DD 名を指定する場合、この値はすべての DD の中の最大の値になります (表 47 を参照)。詳しくは、ハードウェアの資料を参照してください。
24(18)	8	BSAM、QSAM、または BPAM によってサポートされる最大非スパン論理レコード長。装置上のさまざまなタイプのデータ・セットは、さまざまな最大レコード長を持っている可能性があります。そのため、DEVTYPE で UCBLIST をコーディングし、DD 名をコーディングしなかった場合、この値は BSAM、QSAM、および BPAM の可能なデータ・セット・タイプの最小の値になります。

表 47. EXCP またはアクセス方式の大容量ブロック・インターフェースを使用している場合にサポートされる最適および最大ブロック・サイズ:

装置タイプ	最適値	最大値
DASD	ハーフトラック	32 760
リール・テープ	32 760	32 760
3480, 3490	65 535	65 535
3590	262 144 (256 KB)。一部の古いモデルは例外で、その場合は 229 376 (224 KB)。	262 144 (256 KB)
DUMMY	16	5 000 000

DEVTYPE UCB の 4 バイト UCBTYP フィールドを戻します。

シミュレートされた UCBTYP: 196 ページの表 48 の以下のタイプのいずれかを参照すると、表示されている情報が出力域に入ります。

表 48. シミュレートされた装置 UCBTYP 情報

データ・セット・タイプ	出力 (16 進数)
DUMMY アプリケーション・プロセス・キュー	0000 0000
TSO 端末	0000 0101
SYSIN、SYSOUT、またはサブシステム (SUBSYS=)	0000 0102
UNIX システム・サービス (おそらく、HFS) ファイル	0000 0103

DASD 以下のように、16 バイトを戻します。

バイト 0-3 装置上のシリンダーの数 (代替を除く)。VIO データ・セットの場合、この数はデータ・セットを収容するために必要なシミュレートされたシリンダーの数です。

バイト 4-7 シリンダー当たりのトラック数。

バイト 8-9 フラグ (2 バイト)

1... ECKD がサポートされます。Define Extent、Locate Record、および関連のデータ転送コマンドがサポートされます。VIO データ・セットの場合、これらのコマンドは常にサポートされるため、このビットはオンです。

.1.. Locate Record Extended CCW がサポートされます。VIO データ・セットの場合、これらのコマンドは常にサポートされるため、このビットはオンです。

..1. コントローラー・キャッシュがサポートされます。

...x xxxx 予約済み。

バイト 9 予約済み。

バイト 10-11 予約済み。

バイト 12 トラック・セット・サイズ。装置が

read-any または write-any をサポートしない場合は、ゼロです。詳しくは、「3990 Reference」を参照してください。

バイト 13-15 予約済み。

DEVTYPE は、現在はゼロを戻しますが、将来のリリースでは異なる値を戻す可能性があります。

SUFFIX

システムが装置に保管できる拡張フォーマット・データ・セットの各ブロックに追加する接尾部の長さを 2 バイトで戻します。この情報は、TRKCALC マクロを使用したスペース計算や、最適のブロック・サイズを判別するのに使用します。拡張フォーマット・データ・セットをサポートする装置タイプの場合、DEVTYPE は 32 を戻します。非ゼロ値は、データ・セットが実際に拡張フォーマットであること、あるいは装置が拡張フォーマット・データ・セットをサポートすることを意味するものではありません。すべてのストレージ・コントローラーが拡張フォーマット・データ・セットをサポートできるわけではありません。将来のリリースで、この値は変更される可能性があります。

装置特性情報

INFOLIST パラメーターをコーディングしない場合、DEVTYPE マクロを発行した結果として、以下の情報がユーザーの領域に入られます。

シミュレートされた装置特性: 表 49 に示されたタイプのいずれかを参照すると、表示されている情報が出力域のワード 0 とワード 1 に入ります。

表 49. シミュレートされた装置特性情報

データ・セット・タイプ	ワード 0 (16 進数)	ワード 1 (16 進数)
DUMMY アプリケーション・プロセス・キュー	0000 0000	0000 0000
TSO 端末	0000 0101	0000 7FF8
SYSIN、SYSOUT、またはサブシステム (SUBSYS=)	0000 0102	0000 7FF8
UNIX システム・サービス (おそらく、HFS) ファイル	0000 0103	0000 7FF8

ワード 0

UCB の UCBTYP フィールドの定義通りに装置を記述します。

ワード 1

アクセス方式の大容量ブロック・インターフェースを使用しない最大ブロック・サイズ。最大値は 32760 バイトです。直接アクセス装置の場合、この値は、キーなしブロックの最大サイズ、またはオペレーティング・システムによって許容される最大ブロック・サイズのいずれか小さい方の値です。磁気テープ装置の場合、この値は、アクセス方式によって許容される最大ブロック・サイズです。これらの装置タイプおよびその他の装置タイプについては、200 ページの表 50 を参照してください。

プログラムが INFOLIST を使用せずに DEVTAB または UCBLIST のいずれかを指定している場合、次の 3 ワードには、直接アクセス装置に関する以下の情報が入ります。

ワード 2

バイト 0-1 装置上の物理シリンダーの数 (代替を含む)。これは、符号なし 16 ビット数値として扱ってください。

推奨: バイト 0 と 1 を使用する前に、ワード 4、バイト 1、ビット 0 の説明をお読みください。VIO データ・セットの場合、このビットはゼロで、シリンダーの数は、シミュレートされたデータ・セットを収容するために必要な数です。これは、シミュレートされた実際の装置の数とは異なる場合があります。

バイト 2-3 シリンダー当たりのトラック数。

ワード 3

バイト 0-1 最大トラック長。この値はワード 1 の値 (最大ブロック・サイズ) に等しくないことに注意してください。

バイト 2 ブロック・オーバーヘッド、キー付きブロック-トラック上の最後のブロックを除く各キー付きブロックのギャップと検査ビットに必要なバイト数。

推奨: バイト 2 と 3 を使用する前に、ワード 4 の説明をお読みください。

バイト 3 ブロック・オーバーヘッド-トラック上の最後のブロックであるキー付きブロックのギャップと検査ビットに必要なバイト数。

バイト 2-3 ブロック・オーバーヘッド-トラック上の最後のブロックを含めた任意のキー付きブロックのギャップと検査ビットに必要なバイト数。この形式の使用は、ワード 4 のバイト 1 ビット 4 の 1 によって示されます。

基本オーバーヘッド-カウント・フィールドに必要なバイト数。この形式の使用は、ワード 4 のバイト 1 ビット 3 の 1 によって示されます。

ワード 4

バイト 0 ブロック・オーバーヘッド、キーなしブロック-ワード 3、バイト 2 または 3 (あるいは、ブロックがキーなしの場合は、バイト 2 と 3) から減算されるバイト数。

ワード 4 のバイト 1 ビット 3 が 1 の場合、このバイトには、モジュロ装置のモジュロ係数が入ります。

バイト 1

- ビット 0 オンの場合、ワード 2、バイト 0 に示されたシリ
ンダー数を示し、1 は無効です。
- ビット 1 オンの場合、ECKD がサポートされます。Define
Extent、Locate Record、および関連データ転送コマ
ンドが、この装置用にインプリメントされていま
す。VIO データ・セットの場合、これらのコマンド
は常にサポートされるため、このビットはオンで
す。
- ビット 2-3 両方ともオンの場合、装置はキャッシュ・ストレ
ージ制御に接続されていることを示します。
- ビット 3 オンの場合、モジュロ装置 (3380、3390 など) を示
します。
- ビット 4 オンの場合、ワード 3 のバイト 2 と 3 には、最
後のブロックを含めたトラック上の任意のトラッ
クのブロック・オーバーヘッドを示すハーフワードが
入ります。
- ビット 5 オンの場合、装置はページング CCW をサポートし
ます。
- ビット 6 オンの場合、装置は代替シリンドラーを持っていま
せん。
- ビット 7 オンの場合、トラックの最後のブロックを除くす
べてのブロックに許容度係数を適用する必要があります。

バイト 2-3 許容度係数-この係数は、ブロックの有効長を計算するのに使用され
ます。計算は、以下の順序で行います。

- ステップ 1 ブロックのキー長をブロックのデータ長に追加す
る。
- ステップ 2 ワード 4 のバイト 1 のビット 7 をテストする。
ビット 7 が 0 の場合、ステップ 3 を実行しま
す。ビット 7 が 1 の場合、ステップ 1 で計算さ
れた合計に許容度計数を乗算します。乗算の結果を
9 ビット右にシフトします。
- ステップ 3 上記で得られた値に、該当するブロック・オーバ
ーヘッドを追加する。

ワード 4 のバイト 1、ビット 3 が 1 の場合、バイト 2 と 3 に
は、データまたはキー・フィールドのオーバーヘッドが入ります。

プログラムが DEVTAB と RPS を指定している場合、または INFOLIST を指定せ
ずに UCBLIST を指定している場合、次のワードには以下の情報が入ります。

ワード 5

- バイト 0-1 セクター計算用の R0 オーバーヘッド
- バイト 2 各トラックのセクター数
- バイト 3 各トラックのデータ・セクター数

表 50 と 表 51 は、DEVTYPE マクロを発行した結果としての各装置タイプの出力を示しています。

プログラムが UCBLIST を指定し、INFOLIST を指定していない場合、出力は UCB リストに含まれる各 UCB アドレスごとに 1 つの 6 ワード項目から構成されます。

表 50. DEVTYPE マクロからの出力

IBM 装置 ¹	最大アクセス方式レコード・サイズ (大容量ブロック・インターフェースを使用していない場合)
2540 読取装置	80
2540 パンチ	80
2501 読取装置	80
3890 書類読取処理装置	80
3505 読取装置	80
3525 パンチ	80
1403 プリンター	120 ¹
3203 モデル 5 プリンター	132
3211 プリンター	132 ¹
3262 モデル 5 プリンター	132
4245 プリンター	132
4248 プリンター	132 ²
3800 または 3900 印刷サブシステム	136 ³
3410、3420、3422、3424 ⁴	32760
3430、3480、3490、3590 磁気テープ装置	

注:

1. ある種のモデルはより大きい行サイズを使用できますが、最小行サイズを想定していません。
2. IBM 4248 プリンターは、168 印刷位置機能が装置にインストールされている場合でも、132 文字を戻します。
3. IBM 3800 印刷サブシステムは、10 ピッチで 136 文字/行、12 ピッチで 163 文字/行、15 ピッチで 204 文字/行を印刷できます。マシンのデフォルトは、10 ピッチで 136 文字/行です。
4. 3424 磁気テープ装置は、Brazil, S.A. でのみ使用可能です。

表 51. DEVTYPE マクロからの出力 - DASD 装置

IBM 装置	最大レコード・サイズ (ワード 1、10 進数)	DEVTAB (ワード 2、3、4、16 進数)	RPS (ワード 5、16 進数)
3380 モデル AD4、AJ4、BD4、BJ4、CJ2 ディスク・ストレージ	32 760	0376 000F BB60 0100 2010 010B	04E0 DED6

表 51. DEVTYPE マクロからの出力 - DASD 装置 (続き)

IBM 装置	最大レコード・サイズ (ワード 1、10 進数)	DEVTAB (ワード 2、3、4、16 進数)	RPS (ワード 5、16 進数)
3380 モデル AD4、AJ4、BD4、BJ4 ディスク・ストレージ (キャッシュ・ストレージ制御に接続)	32 760	0376 000F BB60 0100 2030 010B	04E0 DED6
3380 モデル AE4 および BE4 ディスク・ストレージ	32 760	06EB 000F BB60 0100 2010 010B	04E0 DED6
3380 モデル AE4 および BE4 ディスク・ストレージ (キャッシュ・ストレージ制御に接続)	32 760	06EB 000F BB60 0100 2030 010B	04E0 DED6
3380 モデル AK4 および BK4 ディスク・ストレージ	32 760	0A60 000F BB60 0100 2010 010B	04E0 DED6
3380 モデル AK4 および BK4 ディスク・ストレージ (キャッシュ・ストレージ制御に接続)	32 760	0A60 000F BB60 0100 2030 010B	04E0 DED6
3390 モデル 1 (3990 モデル 2 に接続)	32 760	0459 000F E5A2 0000 0052 0000	0594 E000
3390 モデル 1 (3990 モデル 3 に接続)	32 760	0459 000F E5A2 0000 0072 0000	0594 E000
3390 モデル 2 (3990 モデル 2 に接続)	32 760	08B2 000F E5A2 0000 0052 0000	0594 E000
3390 モデル 2 (3990 モデル 3 に接続)	32 760	08B2 000F E5A2 0000 0072 0000	0594 E000
3390 モデル 3 (3990 モデル 2 に接続)	32 760	0D0B 000F E5A2 0000 0052 0000	0594 E000
3390 モデル 3 (3990 モデル 3 に接続)	32 760	0D0B 000F E5A2 0000 0072 0000	0594 E000
3390 モデル 3 (3990 モデル 6 に接続)	32 760	0D0B 000F E5A2 0000 0072 0000	0594 E000
3390 モデル 9 (3990 モデル 2 に接続)	32 760	2721 000F E5A2 0000 0052 0000	0594 E000
3390 モデル 9 (3990 モデル 3 に接続)	32 760	2721 000F E5A2 0000 0052 0000	0594 E000
3390 モデル 9 (3990 モデル 6 に接続)	32 760	2721 000F E5A2 0000 0052 0000	0594 E000
9345 モデル 1	32 760	05A0 000F BC98 0000 0052 0000	04A0 D500

推奨: 現在サポートされる装置のすべてについて、DEVTYPE は、スペース計算を行うための十分な情報を戻すわけではありません。スペース計算を行うには、TRKCALC マクロとセクター変換ルーチンを使用してください。TRKCALC マクロの使用については、229 ページの『トラック計算の実行』を参照してください。セクター変換ルーチンについては、154 ページの『RPS 装置上のブロックのセクター番号の入手』を参照してください。

DEVTYPE - 戻りコードと理由コード

制御は、プログラムの DEVTYPE マクロ命令の後の次の実行可能命令に戻されません。レジスター 15 には DEVTYPE マクロからの戻りコードが入り、レジスター 0 には理由コードが入ります。レジスター 2 から 14 の内容は未変更です。レジスター 1 の内容は予測不能です。戻りコードとその意味は、次のとおりです。

戻りコード	意味
0 (X'00')	情報はユーザーの作業域に正常に保管されました。
	理由コード 意味
	0 (X'00') すべての情報が使用可能です。
	4 (X'04') DEVTYPE は 1 つ以上の INFO パラメーター・コードを認識しませんでした。 DEVTYPE は、該当する量の戻り域をクリアし、残りの INFO コードを処理しました。
4 (X'04')	呼び出しエラー。
	理由コード
	意味
	4 (X'04')
	DD 名が定義されていません。
	8 (X'08')
	パラメーター・リストが無効です。エラーは、バージョン・コード、長さ、ゼロ・フィールド、または戻り域の大きさが不十分のいずれかが考えられます。現行リリースでは、DEVTYPE は完全な UCB 検査を行いません。UCBLIST パラメーターの 3 番目の値に正しい値がコーディングされていないために、UCB アドレスが無効である可能性があります。
8 (X'08')	サポートされない装置クラス。
	理由コード
	意味
	12 (X'0C')
	DEVTYPE は装置クラスをサポートしていません。これは DASD、テープ、サブシステム (スプールを含む)、ユニット・レコード、TSO 端末、ダミー、通信、グラフィックス、またはチャンネル間アダプターでなければなりません。UCBLIST がコーディングされていた場合、DEVTYPE はリストの残りを無視しました。

DEVTYPE - 例 1 - DD ステートメントを参照

```

DEVTYPE MYDD,DEVINFO,DEVTAB
.
.
.
MYDD   DC      CL8'DATATAB'
DEVINFO DC      5F'0'
```

例 1 の DEVTYPE は、DATATAB が DASD データ・セットの場合、20 バイトの装置情報を戻します。そうでない場合は、8 バイトを戻します。

DEVTYPE - 例 2 - パラメーター・リストの作成を含む

```

MVC DTLIST,KDTLIST BUILD PARAMETER LIST IN DYNAMIC STORAGE
*****
* RETRIEVE FOUR BYTE UCBTYP FOR SYSUT1 DEVICE
*****
DEVTYPE MF=(E,DTLIST),,(AREA,L'AREA)
.
.
*****
* RETRIEVE 20 BYTES (DASD INFO AND UCBTYP) FOR THE UNIT DESCRIBED BY THE
* UCB THAT UCBAD POINTS TO. THE AREA ADDRESS AND LENGTH ARE STILL
* IN THE PARAMETER LIST FROM THE DEVTYPE EXECUTION PERFORMED ABOVE
*****
DEVTYPE UCBLIST=(UCBAD,1),INFOLIST=ILIST2,MF=(E,DTLIST)
.
.
KDTLIST DEVTYPE FIRSTDD,MF=L,INFOLIST=ILIST1 NON-MODIFIABLE PARAMETER
* LIST
LDTLIST EQU *-KDTLIST
FIRSTDD DC CL8'SYSUT1'
ILIST1 DEVTYPE INFO=DEVTYPE
ILIST2 DEVTYPE INFO=(DASD,DEVTYPE) REQUEST DATA AT AREA
DYNAMIC DSECT
UCBAD DS A ADDRESS OF UCB
DTLIST DS CL(LDTLIST) DEVTYPE PARAMETER LIST(MODIFIABLE)
DS 0F ALIGNMENT FOR TRKCYL
AREA DS 0CL20 INFORMATION FROM
* DEVTYPE INFO=(DASD,DEVTYPE)
TYP1 DS 0CL4 INFORMATION FROM
* DEVTYPE INFO=DEVTYPE (UCBTYP)
* FOR FIRSTDD
NUMCYL DS F NUMBER OF CYLINDERS ON VOLUME
TRKCYL DS F NUMBER OF TRACKS PER CYLINDER
DS CL8 MISCELLANEOUS
TYP2 DS CL4 UCBTYP FROM UCB POINTED TO
* FROM UCBAD

```

例 2 の DEVTYPE は、動的に獲得したストレージにパラメーター・リストを作成し、プログラムを再入可能にします。次に、最初の実行形式で追加パラメーターを提供し、2 番目の実行形式でその一部をオーバーライドします。実際には、DEVTYPE の最初の指定は、次のようになります。

```
DEVTYPE FIRSTDD,(AREA,L'AREA),INFOLIST=ILIST1
```

ILIST1 は、戻す必要がある 4 バイトを記述します。これは 20 バイト域の先頭になります。リスト形式の KDTLIST は、最初の実行形式によってオーバーライドされないパラメーターを指定します。実行形式は、実行時に決められるパラメーターを指定します。実際には、DEVTYPE の 2 番目の指定は、次のようになります。

```
DEVTYPE ,(AREA,L'AREA),INFOLIST=ILIST2,UCBLIST=(UCBAD,1)
```

INFOLIST は、戻す必要がある 20 バイトを記述します。

最初の実行形式は、2 つの定位置パラメーターの前にキーワード・パラメーター (MF) をコーディングするという異例の手法を取っています。最初の定位置の値はヌルで、2 番目の定位置は (AREA,L'AREA) です。これは混乱を招くため、一般的には良い手法とはいえません。ここでは、アセンブラー H および高水準アセンブラーが持つ柔軟性を示す目的でのみ使用されています。

DEVTYPE の別の例は、242 ページの図 27 を参照してください。

ジョブ・ファイル制御ブロックの読み取りおよび変更 (RDJFCB マクロ)

OPEN マクロ命令の結果として実行される機能を達成するために、オープン・ルーチンは、ユーザーがデータ定義 (DD) ステートメントで提供した情報にアクセスすることが必要です。この情報は、システムによってジョブ・ファイル制御ブロック (JFCB) に保管されます。

ある種のアプリケーションでは、データ・セットに対して OPEN マクロ命令を発行する前に、JFCB の内容 (以前に割り振りパラメーターで指定した) を変更することが必要になる場合があります。例えば、順次データ・セットの最後にレコードを追加するとします。現在割り振られているスペースが使い尽くされた場合、既存のデータ・セットを拡張できるように、2 次割り振り数量を追加することが必要になります。ユーザーを支援するために、システムは RDJFCB マクロ命令を提供しています。このマクロ命令は、出口リストで指定された領域に JFCB を移動します。出口リストでの RDJFCB マクロ命令の使用については、208 ページの『例』で説明します。後でユーザーが OPEN マクロ命令を発行する場合、TYPE=J オペランドを指定して、ユーザーが指定した領域内の JFCB を使用してデータ・セットをオープンできます。

RDJFCB と DCB を使用すると、データ・セット名、AMP パラメーター、および VSAM データ・セットのボリューム通し番号を調べることができます。DCB では MACRF と DSORG の任意の有効な組み合わせを使用できます。最も単純な形は、DSORG=PS,MACRF=R です。OPEN TYPE=J を指定して、JFCB を使用して VSAM データ・セットをオープンすることはできません。

RDJFCB マクロを使用すると、連結内のデータ・セットの割り振り情報も検索できます。データ・セットを選択することもできますし、デフォルトで、連結内のすべてのデータ・セットの情報を検索することもできます。

以下の項目を検索できます。

- すべての JFCB
- すべてのボリューム通し番号
- ブロック・サイズ限度
- DD に関連したパス名 (PATHS)。この場合、JFCB 内のデータ・セット名はダミー値です。

RDJFCB を使用してこの情報を検索する方法については、209 ページの『タイプ 07 JFCB 出口リスト項目』で説明しています。

ヒント: OPEN マクロ命令を発行する前に、JFCBTSDM フィールドのビット

JFCNWRIT を 1 に設定すると、オープン処理の終了時に JFCB は書き戻されません。OPEN TYPE=J は通常、ユーザーのプログラムの JFCB の変更コピーを移動して、システム・コピーと置き換えます。この移動が確実に行われるようにするために、プログラムは JFCBMASK+4 フィールドのビット・ゼロを 1 に設定する必要があります。IBM では JFCNWRIT で設定することはお勧めしません。ユーザーの JFCB (システムがデータ・セットをオープンするのに使用した) が書き戻されない場合、EOV、CLOSE、またはジョブ/ステップの終了処理時にエラーが発生することがあります。OPEN はユーザーの JFCB 内の情報を更新した可能性があり、それがシス

テム・コピーの JFCB に反映されないからです。例えば、非特定テープ・データ・セットがオープンされた場合、OPEN はユーザー提供の JFCB を、選択されたテープのボリューム通し番号で更新します。しかし、システム・コピーの JFCB には、このボリューム通し番号が反映されません。これは EOVS、CLOSE、またはジョブ/ステップの終了処理時のエラーの原因になります (例えば、ジョブで要求されても、データ・セットがカタログされないことがあります)。

JFCB に対して行える変更には、次のものが含まれます。

- DSCB の作成日および有効期限フィールドを JFCB に移動する。
- JFCB のボリューム数 (number-of-volumes) フィールドを変更する。

ボリューム数 (number-of-volumes) フィールドは、JFCB と JFCBX (拡張) 内の使用可能なボリューム通し番号の総数を超えない値にのみ変更できます。JFCB は、5 個のボリューム通し番号を持つことができます。各 JFCBX は、15 個のボリューム通し番号を持つことができます。JFCBX が必要かどうかと、必要な JFCBX の個数は、データ・セットの割り振り時に決めます。JFCBX は割り振り後に動的に作成することはできません。従って、ボリューム数 (number-of-volumes) フィールドの最大値は、JFCB と存在する JFCBX の個数に基づきます。ボリューム数 (number-of-volumes) フィールドをその最大値を超える値に設定しても、JFCBX は動的に作成されません。

- DCB フィールドを DSCB から JFCB に移動する。
- ボリューム通し番号を JFCB に追加する (210 ページの『RDJFCB セキュリティー』を参照してください)。

5 個を超えたボリューム通し番号は、JFCBX (拡張) に書き込まれます。JFCBX は、ユーザー・プログラムによって変更することはできません。ストリーム・モード

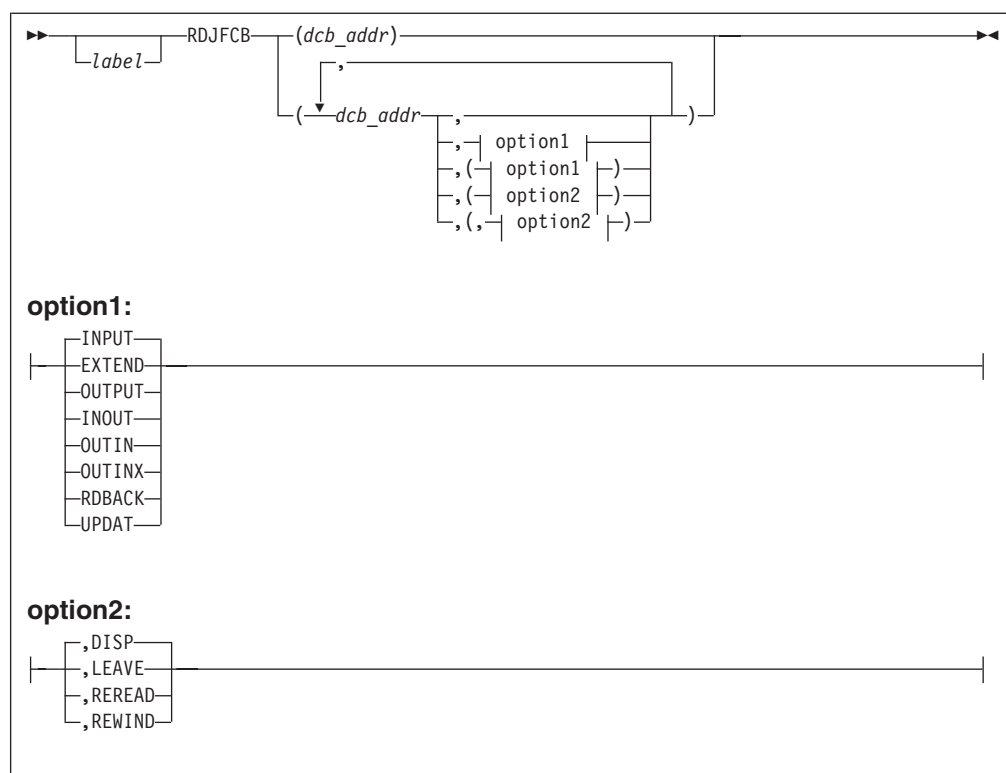
- JFCB 内のデータ・セットのシーケンス番号フィールドを変更する。これは、後続の OPEN TYPE=J が処理するファイルを指定します。この手法を使用して、テープ・ボリュームまたはボリューム・セット上に多数のデータ・セットを書き込むことができます。RDJFCB を使用して、それぞれの OPEN TYPE=J の前に、データ・セットのシーケンス番号を 1 だけ増やしてください。装置がバッファ・テープ・マークをサポートしている場合、DCBE マクロで SYNC=NONE オプションを要求すると、パフォーマンスが著しく向上します。これにより、装置は「ストリーム・モード」に入ることができ、はるかに高速でテープ・マークを書き込めます。副次作用の 1 つは、入出力エラーが予測不能の時間に反映されることです。これは、ユーザー・データ・ブロックの OPEN または CLOSE での ABEND につながります。また、装置に誤動作が生じた場合、アプリケーションが前にクローズした複数のデータ・セットが失われる結果にもなります。次のデータ・セット以外のテープ・データ・セットを高速で配置するには、221 ページの『高速カートリッジ・テープの位置決め』で説明する手法を使用できます。
- JFCB 内のデータ・セット名フィールドまたはメンバー・フィールドを変更する。210 ページの『RDJFCB セキュリティー』および 210 ページの『許可プログラムによる RDJFCB の使用』を参照してください。

- フィールド JFCBFLG3 のビット JFCDQDSP を設定して、デマウント機能でテープ・ボリューム DEQ を呼び出す (219 ページの『テープ・ボリュームのデマウント機能での DEQ』を参照してください)。
 - JFCB の JFCRBIDO フィールドを変更して、カートリッジ・テープをサポートする装置上のテープ・ボリュームの特定データ・ブロックに高速で位置決めできるようにする (221 ページの『高速カートリッジ・テープの位置決め』を参照してください)。
 - ビット JFCNDSCB を設定して、OPEN がデータ・セット・ラベルからのフィールドを JFCB にマージするのを防止する。これは、JFCB や DCB など他のソースから情報を提供する必要があることを意味します。データ・セット・ラベルは、DSCB または標準磁気テープ・ラベルです。このビットは、すべてのフィールドを制御するわけではありません。例えば、作成日や有効期限は制御しません。このビットをオンに設定すると、DISP=MOD あるいは OPEN マクロで EXTEND オプションを使用している場合、自動ステップ再始動の後でデータ・セットが誤って位置決めされる原因になるので注意が必要です。このビットをオンに設定した場合、システムが JFCBDSCB (最初のボリュームの DSCB の TTR) を訂正するのを妨げることがあります。このビットをオンに設定すると、内部システム論理に依存することになるため、良いプログラミング手法とはいえません。
 - ビット JFCNDCB を設定して、OPEN が DCB からのフィールドを JFCB にマージするのを防止する。これは、データ・セット・ラベルに正しい情報が設定されるのを妨げ、また OPEN が特定のチェックを行うのを妨げます。このビットをオンに設定すると、内部システム論理に依存することになるため、良いプログラミング手法とはいえません。
- 2 次割り振り数量は、JFCNWRIT または JFCNDSCB の設定によって防止されない限り、DSCB から JFCB に移動されます。

RDJFCB マクロ仕様

RDJFCB マクロ命令は、ジョブ・ファイル制御ブロック (JFCB) をユーザーが選択した領域に移動します。この領域は、指定するデータ制御ブロックごとに DCB の EXLST パラメーターで識別します。

RDJFCB マクロのフォーマットは、次のとおりです。



ヒント: オプション付き、またはなしで複数の DCB を指定する必要がある場合は、各 DCB (および、オプション) を、図の表示どおりにコーディングし、それぞれの追加 DCB の前にコンマを付けてください。標準形式の RDJFCB マクロの例を以下に示します。

```
RDJFCB (DCB1)
RDJFCB (DCB1,INPUT)
RDJFCB (DCB1,(INPUT))
RDJFCB (DCB1,(INPUT,REREAD))
RDJFCB (DCB1,,DCB2)
RDJFCB (DCB1,,DCB2,(INPUT,REREAD),DCB3,INPUT)
```

図 21. 標準形式の RDJFCB マクロの例

dcb_address, または (*options*)

(OPEN マクロ命令の *dcb_address*、*option1*、および *option2* オペランドと同じですが (「z/OS DFSMS Macro Instructions for Data Sets」を参照))、MODE オペランドは例外で、これは RDJFCB マクロでは無効です。

option オペランドは、RDJFCB 処理には影響を与えません。ただし、リスト形式の RDJFCB マクロでこれを指定し、生成されたパラメーター・リストを実行形式のマクロで参照することができます。

- RDJFCB マクロでは MF パラメーターも使用できます。その構文、使用、および影響は、「z/OS DFSMS Macro Instructions for Data Sets」に記載された OPEN

マクロの説明と同じです。MODE=31 を持たないリスト形式の OPEN マクロを参照する実行形式の RDJFCB マクロをコーディングすることもできます。

- RDJFCB パラメーター・リスト、DCB、出口リストで指定された JFCB 領域、および出口リスト自体も、16 MB より下に常駐する必要がありますが、呼び出し側プログラムは 16 MB より上にあっても構いません。

例

図 22 では、EX1 のマクロ命令は、2 つのデータ制御ブロック (INVEN と MASTER) のパラメーター・リストを作成します。リストを作成する際に、両方のデータ制御ブロックは入力のためにオープンされるものと想定しています。また、両方のブロックの option2 は DISP を想定しています。EX2 のマクロ命令は、システムが作成した INVEN と MASTER 用の JFCB をユーザーが指定した領域に移動し、問題プログラムが変更のために JFCB を使用できるようにします。EX3 のマクロ命令は、INVEN という名前のデータ制御ブロックのパラメーター・リスト項目を変更し、TYPE=J オペランドを通して、問題プログラムがシステム用に JFCB を提供することを示します。

```

EX1      RDJFCB (INVEN,,MASTER),MF=L
        .
        .
EX2      RDJFCB MF=(E,EX1)
        .
        .
EX3      OPEN (,(RDBACK,LEAVE)),TYPE=J,MF=(E,EX1)
        .
        .
INVEN    DCB      EXLST=LSTA,...
MASTER   DCB      EXLST=LSTB,...
LSTA     DS       0F
         DC       AL1(EXLLASTE+EXLRJFCB)
         DC       AL3(JFCBAREA)
        .
        .
JFCBAREA DS       0F,176C
        .
        .
LSTB     DS       0F
        .
        .
        IHAEXLST ,      DCB exit list mapping
    
```

図 22. RDJFCB マクロを使用したコード例

RDJFCB マクロ命令では、複数のデータ制御ブロック・アドレスと関連オプションを指定できます。この機能により、複数のジョブ・ファイル制御ブロックを並列で読み取ることが可能になります。

出口リスト・アドレスは、RDJFCB マクロ命令で指定される各 DCB に提供する必要があります。各出口リストには、RDJFCB がサポートする一方または両方のタイプのアクティブ項目を含めることが必要です。

RDJFCB は、出口リストの項目の 2 つのタイプについて、それぞれのタイプの最初の項目を処理します。例えば、タイプ 07、07、および 13 を含んでいる 3 項目のリストでは、RDJFCB は最初と最後の項目を処理し、2 番目の項目は無視します。無視された項目は、RDJFCB 戻りコードには影響を与えません。

以下では、各項目について簡単に説明します。出口リストの詳しい説明は、「z/OS DFSMS データ・セットの使用法」に記載されています。

RDJFCB が実行された後、レジスター 15 に以下のコードのいずれかが入ります。

表 52. RDJFCB マクロからの戻りコード

戻りコード	意味
0 (X'00')	RDJFCB 機能は正常に完了しました。
4 (X'04')	1 つ以上の DCB で以下の条件のいずれかが検出され、スキップされました。スキップされなかった DCB は正常に処理されました。 <ul style="list-style-type: none"> • DCB は、Open/Close/EOV または同様の機能によって処理中であった。 • DCB 内の DDNAME を持つデータ・セットが割り振られていない。 • DCB はオープンされておらず、その DDNAME はブランクである。
8 (X'08')	1 つ以上の DCB に、処理できない ARL がありました。各 ARL にその状況を説明した理由コードが入っています。 <p>1 つ以上の DCB が、戻りコード 4 の項で説明した条件を検出した可能性があります。このタイプの ARL には、理由コードは含まれません。</p>

タイプ 07 JFCB 出口リスト項目

タイプ 07 JFCB 出口リスト項目を使用して、以下で説明するように、さまざまなタスクを実行できます。

タイプ 07 JFCB 出口リスト項目のフォーマットは、次のとおりです。

16 進コード (高位バイト)	出口リスト項目の内容 (下位 3 バイト)
07	RDJFCB または OPEN (TYPE=J) マクロ命令を使用する場合に必要な 176 バイト域のアドレス。

JFCB が読み取った情報を入れる仮想ストレージ域は、次の条件を満たしていなければなりません。

- ユーザーの領域内にある。
- ワード境界上にある。
- 少なくとも 176 バイトの長さである。

要件: 31 ビット・アドレッシング・モードで実行しているユーザーは、この領域が 16 MB より下の仮想ストレージに配置されていることを確認する必要があります。各出口リスト項目は 4 バイトの長さでなければなりません。システムは、最初に現れた出口リスト項目コードのみを認識します。出口リストの終了は、項目コード・バイトの高位ビットを 1 に設定することによって示します。

このマクロ命令の実行時には、DCB はオープンでもクローズでも構いません。連結順次データ・セットにアクセスし、DCB がオープンしている場合、RDJFCB ルーチンは、現行データ・セットの JFCB を読み取ります。その他の場合はすべて、RDJFCB ルーチンは最初の (または、唯一の) データ・セットを読み取ります。

RDJFCB 要求に関連した DCB の処理中に RDJFCB ルーチンに障害が起きた場合、またはユーザーが出口リスト項目の下位 3 バイトに仮想ストレージ・アドレスを提供していない場合、タスクは異常終了します。DCB ABEND 出口を通して使用可能なオプション (「z/OS DFSMS データ・セットの使用法」で説明) は、RDJFCB マクロ命令が発行された場合は利用不可です。

RDJFCB セキュリティー: OPEN TYPE=J は、ユーザー提供の JFCB で指定されたボリューム通し番号と、システム・コピーの JFCB 内のボリューム通し番号を比較します。異なるボリューム通し番号は、それぞれ排他的にエンキューされます。CLOSE ルーチンはボリュームをデキューしないため、ジョブ・ステップが終了するまでボリュームはエンキューされたままです。ジョブ・ステップがすでにボリュームをオープンしている場合、OPEN TYPE=J は継続します。ボリュームが別のジョブ・ステップによってエンキューされている場合は、ABEND 413 が発生し、戻りコード X'04' が戻されます。

一部の JFCB 変更は、既存のパスワード保護または RACF 保護データ・セットのセキュリティを侵害することがあります。特に以下の変更は、変更を行うプログラムが許可プログラムでない限り (許可プログラムとは、監視プログラム状態にあるか、システム記憶保護キー (キー 0 から 7) の 1 つで実行されているか、または許可プログラム機能のもとで許可されたプログラムをいいます)、あるいはパスワードを提供できない限り、許可されません。

- パスワード保護データ・セットの処理を OLD または MOD から NEW に変更する。
- 処理が NEW のときに、データ・セット名または 1 つ以上ボリューム通し番号を変更する。
- ラベル処理をバイパスするために、ラベル処理の指定を変更する。

JFCB 内のデータ・セット名を変更しても、データ・セットの名前には影響を与えません。異なるデータ・セットを参照するだけです。

許可プログラムによる RDJFCB の使用: VTOC をオープンする場合を除いて、ユーザーが JFCB 内でデータ・セット名を変更し、ジョブ・ステップが APF 許可の場合には、置換されたデータ・セット名の分類名である「SYSDSN」に基づいてシステムのエンキューを行う必要があります。分類名 SYSDSN に対して ENQ マクロを発行するには、許可が必要です。

プログラムがデータ・セット名またはボリューム通し番号を変更し、プログラムが許可されていない場合、OPEN TYPE=J は新しい名前のために動的割り振りを呼び出します。CLOSE は自動的にデータ・セットの割り振りを解除します。

他のシステム機能 (例えば、部分リリース) との正しいインターフェースを使用するために、ENQ マクロには、イニシエーターの TCB とデータ・セット名の長さ (末尾空白を含まない) を含めることが必要です。データ・セットの処理が完了したら、DEQ マクロを使用して、リソースを解放する必要があります。プログラムが許可されていない場合、OPEN TYPE=J マクロを発行して、置換されたデータ・セット名がすでに別のジョブ・ステップによってエンキューされている場合には、ABEND 913 が発生し、戻りコード X'1C' が戻されます。

内容を変更するために VTOC データ・セットをオープンする (つまり、OUTPUT、OUTIN、INOUT、UPDAT、OUTINX、または EXTEND のためにオープンする) には、プログラムは許可プログラム機能 (APF) のもとで許可されていることが必要です。APF は、データ・セットとプログラムに対してセキュリティーと保全性を提供します。プログラムを許可する方法についての詳細は、「z/OS MVS プログラミング: アセンブラー・サービス ガイド」および「z/OS MVS プログラミング: アセンブラー・サービス 解説書 ABE-HSP」に記載されています。

この方法で VTOC を展開することはできません。

制約事項: データ・セット名を NULLFILE (ダミー・データ・セットを意味する) に変更しないでください。名前を NULLFILE に変更すると、DD ステートメントで指定されたデータ・セットに割り振られた装置が、ジョブ/ステップの終了時に割り振り解除されるのを妨げることがあります。

マルチボリューム・データ・セットへのランダム入出力のための BSAM または EXCP の使用: マルチボリューム・データ・セットに対して BDAM DCB をオープンすると、OPEN はプログラムをすべてのボリュームに同時にリンクして、プログラムがボリューム境界を無視してデータ・セットのすべてのボリュームを 1 つのエンティティーとして扱えるようにします。マルチボリューム・データ・セットに対して BSAM または EXCP DCB をオープンすると、OPEN はプログラムが一度に 1 つのボリュームにのみアクセスするようにします。これはディスクとテープの両方に当てはまります。別のボリュームに切り替えるには、BSAM の場合は CHECK または FEOV マクロ、EXCP の場合は EOVS マクロを発行します。前ボリュームに戻るには、データ・セットをクローズしてから再オープンする必要があり、操作が低速になります。

ユーザー・プログラムは、1 つのボリュームにつき 1 つの DCB を使用し、RDJFCB と OPEN TYPE=J を使用して、すべてのボリュームを並列で処理できます。プログラムは、各ボリュームの DCB を把握している必要があります。プログラムは RDJFCB マクロを使用して JFCB を読み取り、TYPE=J を指定した OPEN を使用して、データ・セットの各ボリュームをオープンします。213 ページの図 23 のコーディング例は、EXCP DCB を使用したプロシーチャーを示しています。

この手法は、ストライプ・データ・セットには使用できません。OPEN は常にストライプ・データ・セットのすべてのボリュームを並列でオープンするからです (BDAM と同様に)。

BSAM を使用して非ストライプ・ボリュームを並列で読み取る場合、CHECK マクロの使用は避ける必要があります。これは現行ボリュームの終わりに達すると、自動的に次のボリュームに移動できるからです。CHECK の代わりに WAIT または EVENTS を使用してください。WAIT または EVENTS については、「z/OS DFSMS データ・セットの使用法」および「z/OS DFSMS Macro Instructions for Data Sets」を参照してください。DCBE マクロの MULTACC パラメーターを使用して入出力を最適化する場合、TRUNC マクロも発行する必要があります。

テープを使用する場合、割り振られた各装置で複数の DCB をオープンすることはできません。割り振られた装置の数は、TIOT 項目の長さから、または IEFDDSRV マクロを発行することによって、計算できます。IEFDDSRV は DVAR_NUM_DVENT 内の装置の数を戻します。「z/OS MVS プログラミング: アセンブラー・サービス 解説書 ABE-HSP」を参照してください。

プログラムが以下のいずれかを行うと、データ・セットを損傷します。

- BSAM を使用し、OPEN オプションが UPDAT ではないときに、WRITE マクロを発行した場合、データ・セットは新規トラックまたは次のボリュームに拡張される可能性があります。後者の場合、1 つのボリュームに対して 2 つの DCB がオープンされていることになり、相互に干渉することがあります。
- BSAM を使用しており、OPEN のオプションが UPDAT ではなく、CLOSE の前の最後の操作が WRITE (および、CHECK、WAIT、または EVENTS) の場合、CLOSE はそのボリュームをデータ・セットの最後のボリュームとしてマークを付けます。その結果、どのボリュームにも最後のボリュームとしてのマークが付けられることとなります。このことは、EXCP を使用しており、CLOSE で DCBOFLGS のビット 0 がオンであることが検出された場合にも当てはまりません。EXCP については、110 ページの表 31、および 121 ページの『装置依存のパラメーター』を参照してください。後にプログラムがデータ・セットのボリュームを順次に読み取ろうとした場合 (プログラムがバックアップを取る場合など)、プログラムはこのボリュームの終わりより先は読み取りません。また、後にプログラムが EXTEND または OUTINX オプションを使用して、あるいは DISP=MOD で OUTPUT または OUTIN オプションを使用してオープンし、ボリュームの終わりにレコードを追加しようとした場合、間違ったボリュームに追加する可能性があります。

DASD 上でデータ・セットを新たに割り振る場合、SMS の保証スペース・オプションを使用していない限り、スペースは最初のボリュームにのみ割り振られます。DASD とテープの両方とも、データ・セットがまだボリュームに書き込まれていない場合、OPEN は失敗します。

```

ALLVOLS RMODE 24          Because of DCB exit list & JFCB
                RDJFCB DCB1      Read in the JFCB
                LTR  R15,R15     Branch if the DD name
                BNZ  NODD        is not defined
                * Calculate amount of storage for one DCB per volume and get storage.
                SR  R0,R0        Prepare for IC
                IC  R0,JFCBNVOL   Get number of volumes
                LR  R3,R0        Save number of volumes
                MH  R0,=Y(DCBLNGXE) Mult by EXCP DCB length without append.
                STORAGE OBTAIN,LENGTH=(0),LOC=(BELOW,ANY),ADDR=DCBAddrL
                LR  R4,R1        Point to area for first DCB
                LA  R5,1         Set first volume sequence number
OpenLoop  STH  R5,JFCBVLSQ     Tell OPEN which volume to open
                MVC  0(DCBLNGXE,R4),DCB1 Build a DCB
                OPEN ((R4),UPDAT),TYPE=J Use TYPE=J for one volume
                LTR  R15,R15     Branch in the unlikely event
                BNZ  OpenFail    that OPEN failed
                LA  R4,DCBLNGXE(,R4) Point to place for next DCB
                LA  R5,1(R5)     Increment the volume counter
                BCT  R3,OpenLoop Loop until all volumes are open
                .
                .
                .
DCB1      DCB  DDNAME=SYSUT1,MACRF=(E),EXLST=ExitL,DSORG=PS
* The following is a DCB exit list.
ExitL    DC  0F'0',AL1(EXLLASTE+EXLRJFCB) Last entry, for JFCB
                DC  AL3(JFCB)    Address of JFCB area
DCBAddrL DS  A                  Address of DCB list
JFCB     DS  CL176              JFCB READ IN HERE
                ORG  JFCB+70     Go back to remap
JFCBVLSQ DS  H                  Volume sequence number
                ORG  JFCB+117
JFCBNVOL DS  FL1                Number of volumes allocated
                ORG  ,
* Mapping macro IEFJFCBN might be used instead.
                DCBD DSORG=XE,DEV=(DA,TA) Map an EXCP DCB
                IHAEXLST ,      DCB exit list mapping

```

図 23. EXCP によるマルチボリューム・データ・セットの処理

タイプ 13 JFCB 出口リスト項目

タイプ 13 JFCB 出口リスト項目を使用すると、以下で説明するように、選択された割り振り情報を検索できます。システムは、タイプ X'07' および X'13' 出口リスト項目の両方をサポートします。RDJFCB は、それぞれのタイプの最初の項目を使用します。

タイプ 13 JFCB 出口リスト項目のフォーマットは、次のとおりです。

16 進コード (高位バイト)	出口リスト項目の内容 (下位 3 バイト)
13	割り振り検索リストのアドレス。

RDJFCB を使用した割り振り情報の検索: RDJFCB は、DCB 出口リスト項目タイプ 13 を使用して、連結されている可能性のあるデータ・セットの割り振り情報 (JFCB およびボリューム通し番号) を検索します。出口リスト項目コードは X'13' で、「割り振り情報検索」として定義されます。この項目の 2 番目から 4 番目の

表 53. 割り振り検索リストのフォーマット (IHAARL マクロによってマップ) (続き)

オフセット	バイト	名前	説明
2 (X'02')	2	ARLIDENT	EBCDIC 'AR'
4 (X'04')	1	ARLOPT1	オプション・バイト。
	0	ARLLANY	ARA は 16 MB より下になければなりません。
	1...		ARA は 16 MB より上に置くことができます。
ARLUSS	.1..		要求 ARA には、PATH がコード化された項目ごとにパス名があります。
	. .xx xxxx		予約済み。ゼロでなければなりません。
5 (X'05')	7	ARLRSVD1	予約済み。ゼロでなければなりません。
12 (X'0C')	2	ARLRETRV	情報を検索するデータ・セットの数。0 の場合、連結内のすべてを検索します。
14 (X'0E')	2	ARLFIRST	情報を検索する連結内の最初のデータ・セットの番号。0 または 1 は、連結内の最初のデータ・セットから情報検索を開始することを指定します。
以下のフィールドは、RDJFCB によって設定されます。			
16 (X'10')	4	ARLAREA	ARA のアドレス。 216 ページの表 54 を参照。
20 (X'14')	1	ARLPOOL	ARA が入っているストレージ・サブプール。
21 (X'15')	3	ARLRLEN	ARA の長さ。
24 (X'18')	2	ARLRTRVD	JFCB が検索された連結データ・セットの数。
26 (X'1A')	2	ARLCONC	連結データ・セットの数。連結していない場合、この値は 1 です。
28 (X'1C')	1	ARLRCODE	理由コード 0 = 要求情報が読み取られました。
			以下の理由コードは、戻りコード 8 に関連しています。 4 = ARLFIRST が ARLCONC より大きくなっています。 8 = 情報を読み取るにはストレージが不十分です。ARLPOOL と ARLRLEN は、RDJFCB が必要とする情報を記述します。
29 (X'1D')	7	ARLRSVD2	予約済み。RDJFCB によって使用されません。

システム・マクロ

表 54. 割り振り検索域のフォーマット (IHAARA マクロによってマップ)

オフセット	バイト	名前	説明
0 (X'00')	2	ARALEN	このデータ・セットの情報の長さ (このフィールドを含む)。ARA の開始アドレスに長さフィールドの値を加算して、連結内の次のデータ・セットの ARA のアドレスを指定します (要求された場合)。長さフィールドは、ボリューム数の計算には使用しないでください。
2 (X'02')	1	ARAFLG	フラグ。
	1... ..	ARAXINF	拡張情報セグメントが存在します。
	.xxx xxxx		予約済み。ゼロでなければなりません。
3 (X'03')	1	ARAXINOF	現行データ・セットの割り振り検索域の開始から拡張情報セグメントまでのオフセット (ダブルワード単位)。
4 (X'04')	176(Dec)	ARAJFCB	JFCB
180 (X'B4')	可変	*	6 番目以降のボリューム通し番号。JFCBNVOL 内の値によって判別されます。ボリューム通し番号の個数が、指定されたボリューム数より少ない場合、リストの最後の項目にはすべてブランクが入っている可能性があります。項目の最初のバイトが X'FF' の場合、JCL で指定された VOL=REF およびボリュームを判別できませんでした。

拡張情報セグメント。DSECT 名は ARAXINFO。

0 (X'00')	2	ARAXINLN	拡張情報セグメントの長さ (このフィールドを含む)。
2 (X'02')	2	ARAPATHO	ARAPATHO は、ゼロ以外の場合、パス名 (ARAPATHNAME) に拡張される情報セグメントの先頭からのオフセットです。この場合、
			JFCB 内のデータ・セット名に意味はありません。ゼロの場合、この項目にパス名は入りません。
			予約済み。0 でなければなりません。
4 (X'04')	4		
8 (X'08')	8	ARAXLBKS	このデータ・セットのブロック・サイズ、または 0。0 の場合、ブロック・サイズは ARAJFCB 内の JFCB にあります。

表 54. 割り振り検索域のフォーマット (IHAARA マクロによってマップ) (続き)

オフセット	バイト	名前	説明
16 (X'10')	8	ARABKSLM	システムが決定するブロック・サイズの最大許容値 (DFA 内の BLKSZLIM または DFABLKSZ 値)。この整数値は、ブロック・サイズ (BLKSIZE) がすべてのソースから省略されており、アプリケーションが出力のためにプログラムをオープンした場合にのみ意味を持ちます。これは、以下のソースから使用可能な最初の値です。 <ol style="list-style-type: none"> DD ステートメントまたは動的割り振りの BLKSZLIM キーワード。 データ・クラス内のブロック・サイズ限度。ストレージ管理者によって設定されず。データ・セットが SMS 管理でない場合でも使用可能です。 システム・プログラマーによって SYS1.PARMLIB 内の DEVSUPxx メンバーの TAPEBLKSZLIM キーワードで設定されたシステム・デフォルト。DFA でも使用可能です。 32760 <p>現行レベルのオペレーティング・システムの最小値は、32760 です。</p>
24 (X'16')	16	*	予約済み。
以下のフィールドは、ARAPATHO がゼロ以外の場合に限り存在します。			
x	2	ARAPATHNAME	DSECT
		ARAPATHLEN	パス名の長さ (末尾ブランクを除く)。
			ARAXINLN のアドレスに ARAPATHO の 2 バイトの値を加算して、x の値を計算します。
x+2	255	ARAPATHNAM	パス名。
	257	ARAPNAMLEN	最大パス名セクションのシンボルの長さ。

検索域からの情報の使用が終了したら、FREEMAIN または STORAGE マクロを発行して、GETMAIN を通して獲得した領域をすべて解放する必要があります (RDJFCB によって獲得された ARA を含む)。FREEMAIN または STORAGE マクロをコーディングする場合、それぞれ ARLRLEN、ARLPPOOL、および ARLAREA フィールドからの *length*、*subpool*、および *address* 値を指定してください。DSECT 名は ARAXINFO です。

FREEMAIN マクロは、次のようにコーディングします。

```
FREEMAIN RU,LV=length,SP=subpool,A=address
```

RDJFCB が ARL フィールドに正常に記入した場合、レジスター 15 はゼロに設定されます。それ以外の場合は、209 ページの表 52 を参照してください。

例: 218 ページの図 24 では、ALLOCINF のマクロ命令は、入力用のオープンと想定し、1 つの DCB (INDCB) のパラメーター・リストを作成します。DDNAME SYSLIB に割り振られたすべてのデータ・セットの JFCB とボリューム通し番号が

検索されます。

```

***JCL FOR FOLLOWING INVOCATION OF RDJFCB:
//SYSLIB DD    DISP=SHR,DSN=DEPT61.MACLIB
//        DD    DISP=SHR,DSN=CORPORAT.MACLIB
//        DD    PATH='/projects/sasp/maclib',PATHOPTS=ORDONLY
//        DD    DISP=SHR,DSN=SYS1.MACLIB
***EXAMPLE CODE TO INVOKE RDJFCB ALLOCATION INFORMATION RETRIEVAL:
*   GET A COPY OF THE JFCB FOR THE FIRST OR ONLY DATA SET ALLOCATED
*   TO SYSLIB AND TRY TO READ THE JFCBS VOLUME SERIAL NUMBERS
*   AND PATH NAMES FOR ALL DATA SETS ALLOCATED TO SYSLIB.
*
ALLOCINF RDJFCB (INDCB)
      LTR  R15,R15          TEST RDJFCB RETURN CODE
      BNZ  NOJFCB          BRANCH IF INFORMATION NOT AVAILABLE
      ICM  R1,X'F',SLBAREA GET AND TEST ADDRESS OF ARL
      BZ   OLDSYSTEM      GO IF SYSTEM DOES NOT SUPPORT ARL
      CLI  SLBRCODE,0     TEST RDJFCB REASON CODE
      BNE  NOJFCB          BRANCH IF INFORMATION NOT AVAILABLE
*
*   LOOP THROUGH THE JFCBS IN THE AREA TO WHICH SLBAREA POINTS.
*   CODE CAN BE INSERTED HERE TO PRINT THE DATA SET NAMES, VOLUME SERIAL NUMBERS
*   AND PATH NAMES.
      L    R9,SLBRTRVD     GET NUMBER OF JFCB'S RETRIEVED
      L    R2,SLBAREA      POINT TO ARA
      USING ARA,R2
LOOPARA TM  ARAFLG,ARAXINF BRANCH IF NO EXTENDED
      BZ   USEJFCB         INFORMATION SEGMENT
      SR   R3,3            PREPARE FOR IC
      IC   R3,ARAXINOF    GET DOUBLEWORD OFFSET
      SLL  R3,3            GET BYTE OFFSET
      AR   R3,R2          POINT TO EXTENDED INFO SEGMENT
      USING ARAXINLN,R3   EXTENDED INFORMATION SEGMENT
      SR   R4,R4          PREPARE FOR ICM
      ICM  R4,B'0011',ARAPATHO BRANCH IF NO PATH
      BZ   USEJFCB         NAME
      USING ARAPATHNAME,R4
*   PRINT PATH NAME
      .
      .
      B    NEXTARA
*   PRINT DATA SET NAME IN JFCB.
USEJFCB ...
      .
      .
NEXTARA AH  R2,ARALEN     POINT TO NEXT ARA ENTRY
      BCT  R9,LOOPARA     DECREMENT JFCB COUNTER, LOOP IF MORE
      .
      .
      SR   R2,R2
      IC   R2,SLBPOOL
      SR   R0,R0
      ICM  R0,B'0111',SLBRLN
      FREEMAIN RU,LV=(0),SP=(R2),A=SLBAREA
      .
      .

```

図 24. 割り振り情報検索のコード例 (1/2)

```

OLDSYSTM DS    0H          ROUTINE TO HANDLE JUST LIBJFCB
      .
      .
*
NOJFCB  DS    0H          ROUTINE TO HANDLE INABILITY TO GET THE
*                               JFCB. THE DATA SET MAY NOT BE ALLOCATED.
      .
      .
*

SLBOPNX DS    0H          DCB OPEN EXIT ROUTINE FOR SYSLIB.
*                               HANDLES RECFM, LRECL, AND BLKSIZE.
      .
      .
INDCB   DCB    DSORG=P0,DDNAME=SYSLIB,MACRF=R,SYNAD=INEROR,          X
      .
INEXLST DC    0F'0',AL1(EXLDCBEX)  ENTRY CODE FOR OPEN EXIT ROUTINE
DC      AL3(SLBOPNX)  ADDR OF DCB OPEN EXIT ROUTINE
DC      AL1(EXLARL)   ENTRY CODE TO RETRIEVE
*                               ALLOCATION INFORMATION
      .
DC      AL3(SLBSTRT)  ADDR OF ALLOCATION RETRIEVAL LIST
DC      AL1(EXLLASTE+EXLRJFCB) ENTRY CODE TO RETRIEVE FIRST JFCB
*                               AND INDICATE LAST ENTRY IN LIST
*
DC      AL3(LIBJFCB)  ADDR OF JFCB FOR FIRST DATA SET
*
* AN ALLOCATION RETRIEVAL LIST FOLLOWS, POINTED TO BY DCB EXIT LIST.
*
SLBSTRT IHAARL DSECT=NO,PREFIX=SLB
      DC    0F'0'
LIBJFCB DC    CL176' '          FIRST JFCB
      .                               IHAARA ,
      .                               IHAEXLST ,          DCB exit list mapping

```

図 24. 割り振り情報検索のコード例 (2/2)

テープ・ボリュームのデマウント機能での DEQ

この機能は、無制限の長さのテープ・データ・セット (ログ・テープなど) を作成する、長時間実行されるプログラムでの使用を目的としています。そうしたプログラムでこの機能を使用すると、処理済みのボリュームを別のジョブに割り振って処理すること (データ削減など) が可能になります。無制限の長さのデータ・セットをクローズして、動的に割り振り解除しない限り、この処理を他の目的に使用することは禁止されています。

この機能は、オフセット 163 (X'A3') のフィールド JFCBFLG3 のビット JFCDQDSP (ビット 0) を 1 に設定して、RDJFCB/OPEN TYPE=J インターフェースを通してのみ呼び出すことができます。以下の条件をすべて満たす場合、OPEN または EOVS によってボリュームがデマウントされるときに、テープのボリューム通し番号がデキューされ、メッセージ IEC502E が出されます。

- テープ・ボリュームは OPEN または EOVS で使用するために検査済みである (テープは、ファイル保護、ラベル・タイプ、および密度競合が解決された後、検査済みと見なされます。検査済みについて詳しくは、220 ページを参照してください)。
- JFCDQDSP が 1 に設定されている。

- プログラムが APF で許可されている (記憶保護キーおよび監視/問題プログラム状態は関係ありません)。
- 出力のためにテープ・ボリュームを即時に処理する必要がある。つまり、OPEN がボリュームを検査し、OPEN オプションが OUTPUT、OUTIN、または OUTINX であるか、あるいは EOV がボリュームを検査し、DCB が OUTPUT、OUTIN、INOUT、または EXTEND 用にオープンされ、データ・セットに対する最後の操作が出力操作であったか (DCBOFLWR が 1 に設定されている) のいずれかです。

EOV で JFCDQDSP が 1 に設定されているのを検出するには、プログラムは、JFCBTSDM のビット 4 を 1 に設定して、JFCB の再書き込みを禁止しないようにする必要があります。

テープ・ボリュームは、ファイル保護、ラベル・タイプ、および密度競合が解決された後、検査済みと見なされます。この検査の後、さらに OPEN または EOV 処理が進んで、有効期限、セキュリティー保護、チェックポイント・データ・セット保護、または入出力エラーのためにボリュームがリジェクトされたとしても、ボリュームはデマウント時にデキューされます。

ボリューム通し番号がデキューされると、ボリュームは別のジョブへの割り振りに使用できるようになります。ただし、ボリュームの DEQ はボリュームを割り振り解除せずに行われるので、許可プログラムとシステムの両方とも、デマウント機能での DEQ を誤用しないように十分に注意する必要があります。そうした誤用について、以下で説明します。

- 許可プログラムは、デマウント機能でのテープ・ボリューム DEQ を使用して、データ・セットをクローズし、再オープンしてはなりません。これを行うと、以下のいずれかが起こります。
 - デキューされたボリュームはマウントすることができ、現在、別のジョブで使用されている。許可プログラムのためにボリュームのマウントが要求された場合、オペレーターはマウントを満たすことができません。そのため、オペレーターは、要求しているジョブを取り消すか、そのボリュームを使用しているジョブを取り消すか、要求しているジョブがタイムアウトになるのを待つか、またはそのボリュームを使用しているジョブが終了するのを待つかのいずれかが必要になります。
 - デキューされたボリュームは、別のジョブに割り振ることができるが、まだ使用されていない。オペレーターは、許可ジョブのマウント要求を満たすためにボリュームをマウントします。別のジョブによってボリュームのマウントが要求された場合、オペレーターはそのマウント要求を満たすことができず、前の項目と同じ選択に直面します。
 - デキューされたボリュームはまだ別のジョブに割り振ることはできず、ボリュームは許可ジョブのマウント要求を満たすためにマウントされている。別のジョブがボリュームを割り振ることができ、ボリュームのマウントを要求した場合、前の項目と同じ状況になります。

APF 許可を持つプログラムの実行を許可し、プログラムがデマウント機能での DEQ を使用してデータ・セットをクローズし、再オープンしないようにするのは、インストール・システムの責任です。

- 許可プログラムがデマウント機能での DEQ を使用しているが (データ・セット 1)、別のテープ・データ・セット (データ・セット 2) も処理する場合には、注意が必要です。DD ステートメントでデータ・セット 1 とデータ・セット 2 に対して同じボリューム通し番号がコーディングされたと想定してください。データ・セット 1 のボリュームはデマウントされたので、そのボリュームがまだデータ・セット 2 のために要求される可能性があっても、ボリュームはデキューされます。データ・セット 2 と別のジョブはデキューされたボリュームを争奪するため、前述のすべての問題が生じることになります。

デマウント機能での DEQ の用途 (つまり、無制限の長さのテープ・データ・セットを作成する、長時間実行されるアプリケーション) を考えると、この問題は起きてはなりません。このタイプのアプリケーションは通常、ユーザー作成の DD ステートメントを使用してバッチの実行を通して呼び出されることはありません。

- デマウント機能での DEQ によりボリュームがデマウントおよびデキューされた後、特定または非特定のマウント要求に応答してボリュームがマウントされた場合、ボリュームは制御プログラムによって自動的にリジェクトされません。この機能を使用していない場合は、制御プログラムはボリュームが使用中であることを認識し (ENQ によって)、ボリュームをリジェクトします。したがって、デマウント機能での DEQ を使用する場合、こうしたエラーに対する制御プログラムの保護の低下を考慮して、操作手順 (実際には、間違っただボリュームがマウントされるのを防止すること) を検討する必要があります。特に、許可プログラムにボリュームが再マウントされ、ボリュームが以前にその許可プログラムによって使用されていた場合、JFCB 内に重複するボリューム通し番号が存在し、制御プログラムは EOV 処理時にボリュームを解放できなくなります。
- チェックポイント/リスタートに関する考慮事項については、「z/OS DFSMSdfp Checkpoint/Restart」で説明しています。

高速カートリッジ・テープの位置決め

カートリッジ・テープの高速位置決めは、IBM 標準ラベル付きテープ上のデータ・セットをいずれかの EXTEND (OUTINX、EXTEND、または DISP=MOD) でオープンする場合に使用できます。高速位置決めは、そのようなデータ・セットの開始でオープンする場合にも使用できます。高速位置決めを呼び出すには、プログラムは JFCB の特定のフィールドを変更し、OPEN TYPE=J を使用してデータ・セットをオープンする必要があります。正しいハードウェア・レベルの IBM 3590 モデル A60 で書き込みや読み取りを行う場合は、ここで説明する手順を使用することはそれほど重要ではありません。磁気テープ・サブシステムは、以下のようなパフォーマンス向上を自動的に達成します。この手順は、パフォーマンスを低下させることはありません。

ヒント: IBM 3480、IBM 3490、または旧型の IBM 3590 では、この手法は、データ・セット・シーケンス番号を設定するより、はるかに大きなパフォーマンス向上を実現できます。加えて、DFSMSrmm を備えたシステムは、すべてのカートリッジ・テープに対して自動的にこの高速手法を使用します。IBM 3590 モデル A60 の場合は、どちらの手法でもハイパフォーマンスが得られます。

JFCB の変更は、以下の手順で行います。

1. RDJFCB マクロを発行して、システムに JFCB をユーザーの作業域に移動させます。
2. JFCBFLG3 フラグ・バイトに JFCPOSID フラグを設定して、高速検索用のブロック ID を提供していることを示します。
3. ブロック ID を JFCB の JFCRBIDO フィールドに移動します。データ・セットの開始でオープンする場合は、そのデータ・セットの最初のヘッダー・ラベル・レコードのブロック ID を使用します。データ・セットの最後でオープンする場合 (例えば、それを拡張する場合) は、そのデータ・セットのユーザー・データの最後のブロックの直後にあるテープ・マークのブロック ID を使用します。
4. OPEN TYPE=J マクロを発行して、ユーザーが変更した JFCB をシステムに使用させます。

テープが位置決めされた後、OPEN は拡張するデータ・セットのトレーラー・ラベルを処理します。

JFCPOSID フラグをオフに設定すると、OPEN は、高速位置決め機能がアクティブでない場合と同様に、通常どおりにボリュームを位置決めします。

JFCPOSID フラグをオンに設定したが、JFCRBIDO フィールドにブロック ID を提供しなかった場合、OPEN は通常どおりにボリュームを位置決めし、以下のいずれかを行います。

- データ・セットの開始でオープンする場合、OPEN はそのデータ・セットの最初のヘッダー・ラベル・レコードのブロック ID を JFCRBIDO フィールドに挿入します。
- データ・セットの最後でオープンする場合、OPEN はそのデータ・セットのユーザー・データの最後のブロックの直後にあるテープ・マークのブロック ID を JFCRBIDO フィールドに挿入します。

OPEN は、ユーザー・コピーの JFCB を更新しません。システム・コピーの JFCB 内の新しい値を検索するには、OPEN の後で RDJFCB を発行します。

CLOSE 処理時に JFCPOSID フラグがオンの場合 (ユーザーが OPEN の前にオンに設定した場合)、CLOSE は次のデータ・セット (存在しない場合もあります) の最初のヘッダー・ラベル・レコードのブロック ID を JFCRBIDC フィールドに挿入します。そのため、ユーザーがカートリッジ磁気テープ装置の割り振りを解除して、現行ブロック ID を後続の処理のために使用したい場合は、データ・セットをクローズする前にブロック ID を保管してください。

以下の条件のいずれかが存在する場合、OPEN は JFCPOSID フラグをリセットします。

- プログラムが TYPE=J でない OPEN を発行する。
- 要求されたテープ・ボリュームが IBM 標準ラベル付きテープでない。
- 要求された装置がバッファ磁気テープ装置でない。

例外:

1. 動的割り振り解除を指定した場合 (SVC99、DD ステートメントの FREE=CLOSE、または CLOSE マクロの FREE オプションを使用して)、次のデータ・セットのブロック ID はプログラムに対して使用可能になりません。

2. 高速位置決めを使用している場合、データ・セット・シーケンス番号は、DD ステートメントの LABEL=(seqno,SL) で明示的に指定するか、デフォルトを使用して、通常どおりに指定してください。

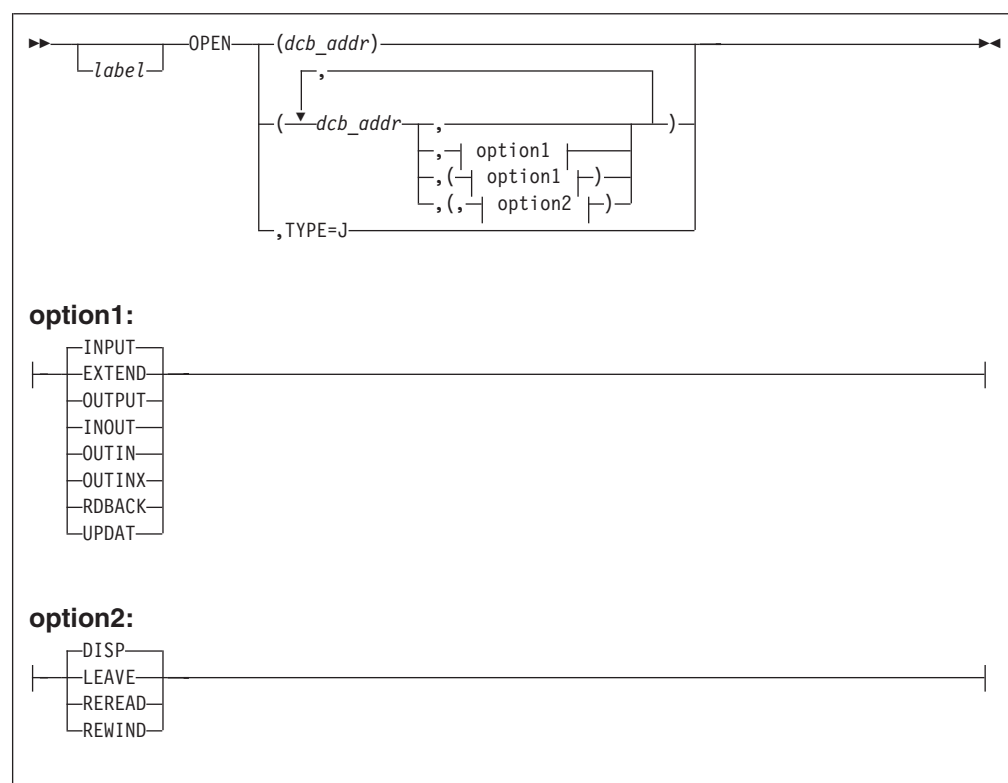
システム・ルーチンは、高速位置決めのために JFCRBIDO フィールドを使用した後、システム・コピーの JFCB 内の JFCRBIDO をクリアして、後続の OPEN で誤って解釈されるのを防止します。

OPEN - JFCB 処理のためのデータ制御ブロックの初期化

OPEN マクロ命令は、1 つ以上のデータ制御ブロック (DCB) を初期化して、関連データ・セットを処理できるようにします。

OPEN マクロ命令のオペランドの詳細な説明は、「z/OS DFSMS Macro Instructions for Data Sets」資料に記載されています。TYPE=J オプションは、JFCB の変更と合わせて使用されるので、システム・プログラマーのみ、またはシステム・プログラマーの監視下でのみ使用する必要があります。

OPEN TYPE=J マクロのフォーマットは、次のとおりです。



ヒント: オプション付き、またはなしで複数の DCB を指定する必要がある場合は、各 DCB (および、オプション) を、図の表示どおりにコーディングし、それぞれの追加 DCB の前にコンマを付けてください。

```

OPEN (DCB1),TYPE=J
OPEN (DCB1,INPUT),TYPE=J
OPEN (DCB1,(INPUT)),TYPE=J
OPEN (DCB1,(INPUT,REREAD)),TYPE=J
OPEN (DCB1,,DCB2),TYPE=J
OPEN (DCB1,,DCB2,(INPUT,REREAD),DCB3,INPUT),TYPE=J

```

図 25. 標準形式の OPEN TYPE=J マクロの例

TYPE=J

参照される DCB ごとに、初期化時に使用する必要があるジョブ・ファイル制御ブロック (JFCB) をユーザーが提供したことを指定します。JFCB は、DD ステートメントの内部表現です。

データ制御ブロックの初期化時に、DCB または既存のデータ・セット・ラベルからの情報、あるいはシステム制御情報を使用して、関連 JFCB を変更できます。

TYPE=J オペランドを指定した場合は、DD ステートメントも提供する必要があります。ただし、ユーザーはシステム作成 JFCB の多くのフィールドを変更できるため、DD ステートメントで提供する情報量は、ユーザーの判断に任されます。DD ステートメントで DUMMY を指定すると、オープン・ルーチンは JFCB DSNAME を無視し、データ・セットをダミーとしてオープンします。(システム作成 JFCB を変更するコーディングの例は、208 ページの図 22 を参照してください。) DD ステートメントでは、少なくとも装置割り振りと (共用状況を防止する方法については、「z/OS MVS JCL ユーザーズ・ガイド」を参照してください)、関連の DCB DCBDDNAM フィールドに対応する DD 名を指定する必要があります。

MODE オペランドは、OPEN マクロ命令の TYPE=J オペランドとは一緒に使用できないので、ここには示されていません。

TYPE=J を指定した OPEN は、呼び出し側からの JFCBX を受け入れないので、最初の 5 つのボリュームの後のボリューム通し番号は変更できません。

OPEN TYPE=J は、割り振り時にボリュームに割り当てられたボリューム属性 (PRIVATE、PUBLIC、または STORAGE) を変更しません。例えば、ボリューム状況の PRIVATE が必要であるが、割り振りでは PUBLIC 状況が割り当てられてしまう場合、VOL=PRIVATE を DD ステートメントで指定する必要があります。

入出力要求のページと復元

システムのページ・ルーチンは、停止操作または静止操作のいずれかを行います。停止操作では、ページ・ルーチンは、EXCP または EXCPVR マクロ命令で開始された指定の入出力要求の処理を停止します。静止操作では、ページ・ルーチンには以下の手順が含まれます。

- 実行のためにシステムに渡され、現在実行中の入出力要求 (EXCP または EXCPVR マクロ命令によって開始された) を完了させます。
- まだ開始されていない、またはシステムに渡されていない要求の処理を停止しますが、要求の IOB を保管して、後で再処理 (復元) できるようにします。

システムの復元ルーチンは、静止された入出力要求を再処理することを可能にします。

制約事項: EXCP または EXCPVR マクロによって開始されていない入出力要求に対して行われるページおよび復元処理は、ここでは対象外です。区分データ・セット (PDS) に対して順次アクセス方式 (SAM) を指定した PURGE および RESTORE マクロを使用するユーザー・アプリケーション (例えば、入出力を同期化するための) は、PDSE、順次拡張フォーマット・データ・セット、または HFS ファイルに対しては適用できません。SAM はこれらのタイプのデータへのアクセスには EXCP または EXCPVR を使用しないからです。

ページおよび復元ルーチンに制御を渡すには、パラメーター・リストを作成し、そのアドレスをレジスター 1 に入れてから、マクロ命令を発行します。

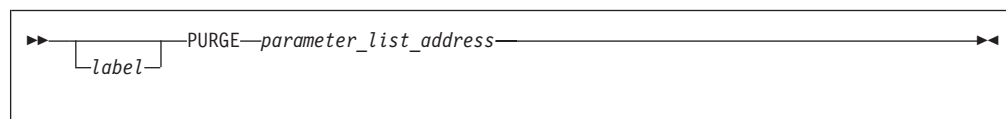
PURGE または RESTORE マクロ (および、パラメーター・リスト) には、24 ビットまたは 31 ビット・アドレッシング・モードを使用できます。

PURGE マクロ仕様

PURGE マクロは、入出力要求を停止または終了するのに使用します。

PURGE 機能によって提供される 31 ビット・インターフェースの使用については、「z/OS MVS Programming: Authorized Assembler Services Reference ALE-DYN」内の『General-Use Mapping Macros』のセクションを参照してください。

PURGE マクロのフォーマットは、次のとおりです。



parameter_list_address-RX タイプ・アドレス、(2-12)、または (1)

ユーザーがストレージのワード境界に作成した、12 または 16 バイトの長さのパラメーター・リストのアドレス。パラメーター・リスト・アドレスは、RX タイプのアドレスとして指定するか、あるいはレジスター 2 から 12、またはレジスター 1 に指定できます。マッピング・マクロの名前は IECDPPL です。

パラメーター・リストのフォーマットと内容を以下に示します。

バイト	内容
0	ページ・ルーチンのアクションを指定する 1 バイト。ビットの設定とその意味は、次のとおりです。
1... ..	単一データ・セットに対する入出力要求をページします。このビットの設定は、バイト 12 のビット 2 が 0、およびバイト 0 のビット 6 が 0 のときのみ効力を持ちます。
0... ..	TCB またはアドレス・スペースに関連した入出力要求をページするか、または複数のデータ・セットに対する入出力要求をページします。バイト 12 のビット 2 が 1 の場合、要求はアドレス・スペース

- に関連した入出力をパージすることです。バイト 12 のビット 2 が 0 で、バイト 0 のビット 6 が 1 の場合、要求は TCB に関連した入出力をパージすることです。バイト 12 のビット 2 が 0 で、バイト 0 のビット 6 が 0 の場合、要求は複数のデータ・セットに対する入出力をパージすることです。
- .1.. パージされた入出力要求に関連した ECB を通知します。
 - ..1. 入出力要求処理を停止します。(0 の場合、入出力要求処理を静止します。)
 - ...1 関連要求をパージします。(データ・セット・パージが要求された場合にのみ有効です。)
 - 0... 予約済み-ゼロでなければなりません。
 -1.. 非同期でスケジュールされた処理の TCB 要求ブロック・チェーンをパージしません。
 -1. TCB に関連した入出力要求をパージします。このビットの設定は、バイト 12 のビット 2 が 0 の場合にのみ影響を与えます。
 -1 これは、16 バイトのパラメーター・リストです。追加のパージ・オプションは、バイト 12 から 15 に指定されます。(このビットがオフの場合、リストは 12 バイトの長さで、パージ・ルーチンはこのリストのバイト 4 またはレジスター 15 に戻りコードを入れません。)
- 1,2,3 単一データ・セットに対する入出力要求をパージする場合、DEB のアドレス。複数のデータ・セットに対する入出力要求をパージする場合、DEB チェーン内の最初の DEB のアドレス。(各 DEB の next-to-the-last ワードは、チェーン内の次の DEB を指していなければなりません。最後の DEB の 2 番目のワードにはゼロが入っていなければなりません。)
- 4 1 バイトのゼロ。(バイト 0 のビット 7 がオンの場合、パージ・ルーチンはバイト内にコードを入れます。パージ操作が正常に行われた場合は X'7F'。正常に行われなかった場合は X'40' です。バイト 0 のビット 7 がオフの場合は、このバイトに X'7F' が入ります。)
- 5,6,7 バイト 0 のビット 6 をオンにした場合、パージする必要がある入出力要求に関連した TCB のアドレス。その TCB 下で実行中の場合は、ゼロです。
- 8 X'00' または X'02' の値は、EXCP が所有者であることを示します。
- 9,10,11 ストレージ内のワードのアドレスまたは DEBUSPRG フィールドのアドレス (このパラメーター・リスト内の DEB アドレスより X'11' バイト上回ります)。パージ・ルーチンは、ユーザーが指定したアドレスに、パージされた入出力復元リストを保管します。その復元リ

ストには IOB チェーンの最初の IOB へのポインターが入っています。ポインターの位置とチェーンのフォーマットは、229 ページの図 26 に示されています。

注: このフィールドは、静止オプションにのみ関係があります。

- 12 追加ページ・オプションを指定できるバイト。ビットの設定とその意味は、次のとおりです。

注: 以下は、バイト 0 のビット 7 が 1 に設定されている場合にのみ適用されます。

..1.	アドレス・スペースに関連した入出力要求をページします。(プログラムは、監視プログラム状態でなければなりません。) このビットの設定は、バイト 0 のビット 6 およびバイト 0 のビット 0 の設定に関係なく影響を与えます。
...1	これがデータ・セット・ページの場合、ページ操作に関連したすべての DEB の妥当性を検査します。ページ操作のタイプに関係なく、このパラメーター・リストの妥当性検査をして、ページ操作の選択に矛盾がないことを確認してください。(プログラムが問題プログラム状態の場合には、ビットの設定に関係なく、このアクションが取られます。)
.... 1...	入出力要求が元の TCB のもとで再処理 (復元) されることを保証します。(これがゼロで、バイト 0 のビット 7 がオンの場合、入出力要求は、復元要求をしているプログラムの TCB のもとで再処理されます。)
.... .0..	ゼロでなければなりません。

- 13 1 バイトのゼロ。

- 14,15 バイト 12 のビット 2 がオンの場合、ページする入出力要求に関連したアドレス・スペースの 2 バイト ID。

制御は、プログラムの PURGE マクロ命令の後続の命令に戻されます。

PURGE からの戻りコード

ページ操作は正常に行われました。レジスター 15 にゼロが入ります。そうでない場合、レジスター 15 には、次の戻りコードのいずれかが入ります。

戻りコード	意味
4 (X'04')	指定の TCB に関連した入出力要求をページする要求は、TCB がジョブ・ステップ TCB を指しておらず、要求側は問題プログラム状態であるために、受け入れられませんでした。

戻りコード	意味
8 (X'08')	アドレス・スペース・ページ操作を要求したが監視プログラム状態でなかったか、あるいはデータ・セット・ページ操作を要求したが、ページ・パラメーター・リストのバイト 1、2、および 3 で DEB アドレスを提供しなかったかのいずれかです。
20 (X'14')	別のページ要求がユーザーの要求より優先使用されました。時間制御ループ内でページ要求を再発行することができません。

例外: パラメーター・リストのバイト 0 のビット 7 をゼロに設定した場合、ページ操作の結果に関係なく、レジスター 15 にはゼロが入ります。

IOB チェーンの変更

この手順は推奨できません。ただし、ページされた入出力要求を復元する順序を変更したり、ページされた要求が復元されるのを防止する場合は、IOB チェーン内の IOB ブロックの順序を変更したり、チェーンから IOB を除去することができます。IOB チェーンのアドレスは、ページ入出力復元リストから入手できます (229 ページの図 26 を参照)。(パッケージ入出力復元リストのアドレスは、ページ・パラメーター・リストのバイト 9 から 11 に示されます。)一部の IOB は異なる記憶保護キー内に存在することがあるので注意してください。

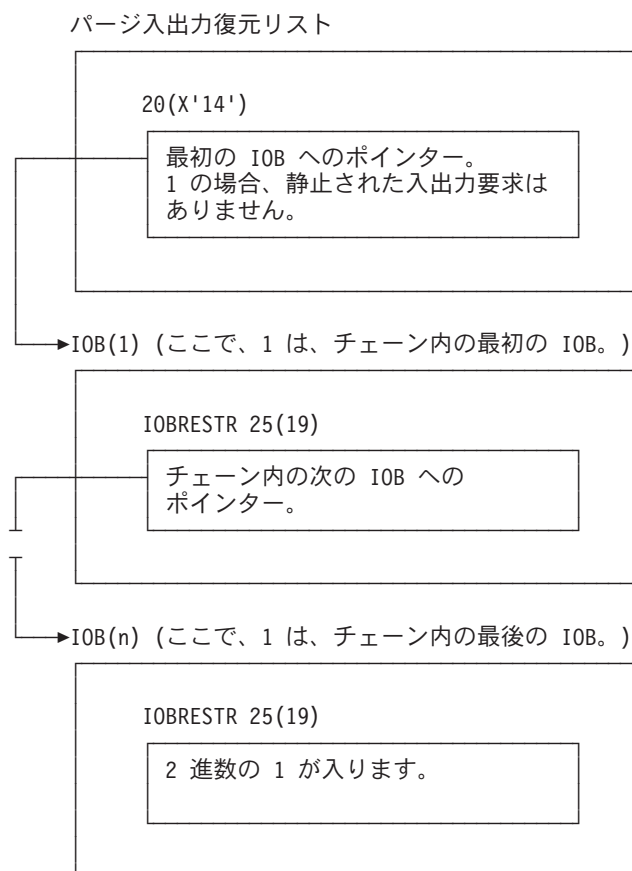
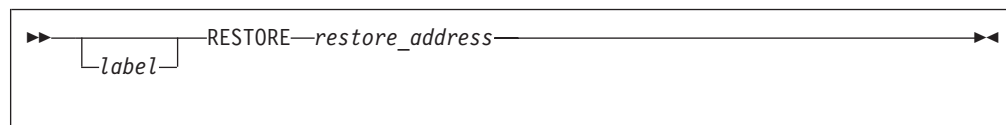


図 26. IOB チェーン

RESTORE マクロ仕様

RESTORE マクロは、入出力要求を再処理するために使用されます。

RESTORE マクロのフォーマットは、次のとおりです。



restore_address-RX タイプ・アドレス、(2-12) または (1)

ページ・パラメーター・リストのバイト 9 に指定したアドレス。バイト 9 については、226 ページを参照してください。

トラック計算の実行

TRKCALC マクロは、DASD トラック容量を計算します。このマクロは、EXCP アプリケーションおよびその他の高機能アプリケーションを対象としています。

TRKCALC を使用して、以下のことを判別できます。

- トラックに書き込める等長レコードの数。

- 合計トラック容量。
- ラック上の残りスペースにレコードを書き込めるかどうかを判別し、新しいトラック・バランスを戻す。
- 最後のレコードをトラックから除去した場合、トラック・バランスはどうか。
- トラックに書き込める可能な最長レコードの長さ。

TRKCALC ルーチンは、SVC 命令や入出力は発行しません。TRKCALC は、SRB ルーチンまたは TCB モードで呼び出すことができます。24 ビットまたは 31 ビット・アドレッシング・モード、および監視プログラム状態または問題プログラム状態で呼び出せます。

TRKCALC の使用

ここでは、TRKCALC を使用してさまざまなタスクを達成する方法について概説します。TRKCALC パラメーターのコーディング方法および出力を戻す方法についての詳細は、231 ページの『TRKCALC マクロ仕様』を参照してください。

トラックに書き込める等長レコードの数を判別する

トラックに書き込める等長レコードの数を判別するには、FUNCTN=TRKCAP を指定して TRKCALC をコーディングします。トラック上の既存のレコードの数、および新規レコードのキーとデータの長さを指定する必要があります (R、K、および DD キーワード、または RKDD パラメーターの R、K、および DD バイトを使用)。

トラックが空であると見なす場合は、R 値を 1 に指定します。そうでない場合は、R 値をトラック上の既存のレコード数より 1 だけ大きい値に指定します。

既存のレコードの長さが新規レコードの長さ (DD 値で指定) と異なる場合、BALANCE パラメーターをコーディングします。そうでない場合は、BALANCE パラメーターは省略してください。

合計トラック容量を判別する

合計トラック容量を判別するには、次のようにコーディングして、

```
FUNCTN=TRKBAL,REMOVE=YES
```

R=1 または RKDD の R バイトを 1 に設定します。

注: この値は、空のトラックを表すために、後で TRKCALC を呼び出すときの BALANCE パラメーターの入力としてのみ役立ちます。このサイズのレコードを書き込むことはできません。

ラック上の残りスペースにレコードを書き込めるかどうかを判別し、新しいトラック・バランスを戻す

ラック上の残りスペースにレコードを書き込めるかどうかを判別し、新しいトラック・バランスを戻すには、次のようにコーディングして、

```
FUNCTN=TRKBAL,REMOVE=NO
```

BALANCE パラメーターをコーディングします。後に TRKCALC への呼び出しで BALANCE パラメーターを使用して、この新しいトラック・バランスを提供できます。

最後のレコードをトラックから除去した場合のトラック・バランスを判別する

最後のレコードをトラックから除去した場合のトラック・バランスを判別するには、FUNCTN=TRKBAL と REMOVE=YES をコーディングします。R、K、および DD パラメーターまたは RKDD パラメーターを使用して、除去するレコードを識別します。これはトラックの最後のレコードでなければなりません。

トラックに書き込める可能な最長レコードの長さを判別する

トラックに書き込める可能な最長レコードの長さを判別するには、FUNCTN=TRKBAL, REMOVE=NO, MAXSIZE=YES をコーディングします。トラック上の既存のレコードの数 (R キーワードまたは RKDD パラメーターを使用) および X'FFFF' のデータ長 (DD キーワードまたは RKDD パラメーターを使用) を指定する必要があります。X'FFFF' の DD 値は、ディスク上でサポートされる値より大きくします。戻りコード 8 (レコードが収まらないことを意味する) を除いて、TRKCALC は最大可能なレコードのサイズを戻します。

トラックが空であると見なす場合は、R 値を 1 に指定します。

トラックが空でない場合、R 値は、トラック上の既存のレコード数より 1 だけ大きい値に指定し、BALANCE パラメーターをコーディングします。この場合、TRKCALC は戻りコード 8 およびトラックの残りに収まる最長可能レコードの長さを戻します。

注: 戻される値は、EXCP 以外のアクセス方式によってサポートされる値よりも大きい場合があります。

制約事項

PDSE の構造とフォーマットは固有であるため、非 EXCP ユーザー・アプリケーションは、PDSE に対する TRKCALC から一貫性のある情報を得ることは期待できません。ただし、処理はエラー表示を出さずに完了します。

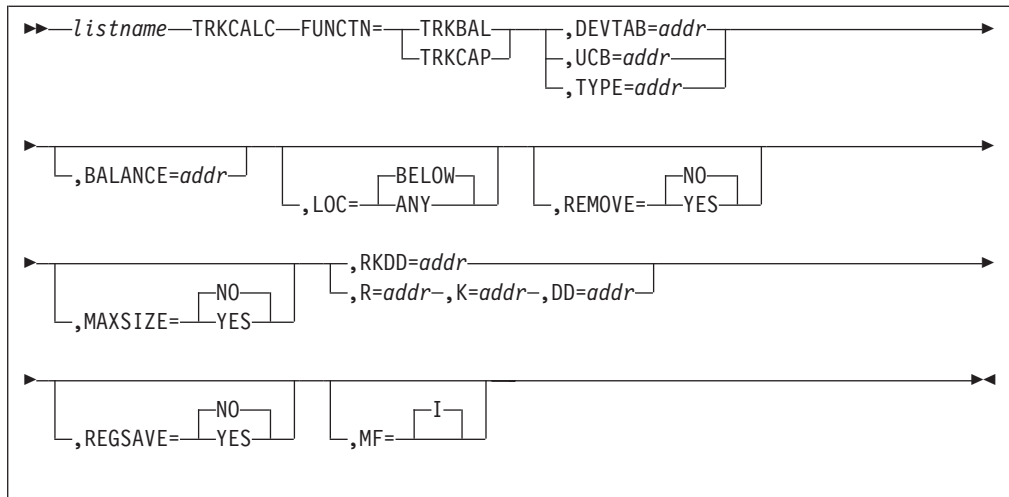
TRKCALC は HFS ファイルをサポートしません。HFS ファイルに対して TRKCALC を使用すると、予測不能の結果を受け取ります。

TRKCALC マクロ仕様

標準形式、リスト形式、実行形式、および DSECT 形式のマクロについて説明します。マクロの説明の後に、TRKCALC マクロの例を示します。

TRKCALC - 標準形式

TRKCALC マクロのフォーマットは、次のとおりです。

**FUNCTN=TRKBAL または TRKCAP**

実行する機能を指定します。3つのキーワード (DEVTAB、UCB、または TYPE) のいずれかを指定して、マクロの情報源を提供します。

TRKBAL

REMOVE=NO が指定されている場合、TRKBAL は追加レコードがトラックに収まるかどうかを計算し、レコードが追加された場合の新しいトラック・バランスを算定します。REMOVE=YES が指定されている場合、TRKBAL はトラックからレコードを除去した場合、トラック・バランスはどうなるかを計算します。トラックに追加または除去されるレコードは、RKDD パラメーター、または R、K、および DD パラメーターによって定義します。

R が 1 に等しく (または、RKDD パラメーターの R 値が 1)、REMOVE=NO が指定されている場合、TRKCALC はレコード 1 が空のトラックに追加されるものとして扱います。R が 1 に等しく、REMOVE=YES が指定されている場合、TRKCALC はレコード 1 がトラックから削除されて、空のトラックが残るものとして扱います。

R が 1 に等しくない場合、指定されたレコードがトラックに追加または削除されます。入力トラック・バランスが BALANCE パラメーターを通して提供されていない場合、トラックは RKDD パラメーター (または R、K、および DD パラメーター) で指定された等サイズのレコードを含んでいるものと想定します。

REMOVE=NO が指定されている場合、以下のいずれかが生じます。

- レコードがトラックに収まる場合、レジスタ 0 に新しいトラック・バランスが入ります。
- レコードがトラックに収まらず、MAXSIZE=NO が指定されている場合、レコードが収まらないという戻りコードがレジスタ 15 に入れられます。

- レコードが収まらず、MAXSIZE=YES が指定されている場合、次のいずれかが起こります。
 - 残りのスペースに収まる最大レコードのデータ長が、レジスター 0 に戻される。
 - 残りのスペースにはどのレコードも収まらないことを示すコードが戻される。

REMOVE=YES が指定されている場合、以下のいずれかが生じます。

- R が 1 に等しい場合、レジスター 0 にトラック容量が入ります。
- R が 1 に等しくない場合、レジスター 0 には、入力レコードによって使用されたトラック・バランスだけ増分された入力トラック・バランス (BALANCE パラメーターを通して提供) が入ります。入力バランスが提供されない場合、レジスター 0 には、R-1 個のレコードがトラックに書き込まれた後に残ったトラック容量が入ります。

TRKCAP

完全なトラック (R が 1 に等しい) または部分的に充てんされたトラック (R が 1 に等しくない) に書き込める固定長レコードの数を計算し、レジスター 0 に戻します。レコードは、RKDD パラメーターの K および DD 値、または K および DD パラメーターによって定義されます。

R の値に応じて、以下のいずれかが生じます。

- R が 1 に等しい場合、TRKCALC は BALANCE パラメーターを無視して、トラックが空であるものとして計算します。
- R が 1 に等しくなく、BALANCE パラメーターが省略されている場合、すでに K および DD 値によって定義された長さの R-1 個のレコードを含んでいるトラックについて計算します。
- R が 1 に等しくなく、BALANCE パラメーターが提供されている場合、残りのトラック・バランスが BALANCE パラメーターの値であるトラックについて計算します。

DEVTAB=addr-RX タイプ・アドレス、(2-12)、(0)、(14)

addr は、装置特性テーブル項目 (DCTE) のアドレスを含んでいるワードを指定します。レジスターを指定した場合、DCTE の実アドレスが入っています。DCTE のアドレスは、オープンされた DCB の DCBDVTBL フィールドで開始するワードに入っています。

UCB=addr-RX タイプ・アドレス、(2-12)、(0)、(14)

addr は、UCB のアドレスを含んでいるワードを指定します。レジスターを指定した場合、UCB の実アドレスが入っています。

TRKCALC マクロは、UCB または UCB コピーのアドレスを受け入れます。無許可プログラムは、UCBSCAN マクロを使用し、COPY および UCBAREA キーワードを指定することにより、UCB

のコピーを入手できます。詳しくは、「z/OS ハードウェア構成定義 (HCD) 計画」を参照してください。

TYPE=addr-RX タイプ・アドレス、(2-12)、(0)、(14)

UCB 装置タイプ (UCBTBYT4) のアドレスを指定するか、またはレジスタの下位バイトで 1 バイト UCB 装置タイプを指定することができます。

LOC=BELOW または **ANY**

UCB パラメーターで受け渡される値が 4 バイト・アドレスか、3 バイト・アドレスかを示すオプション・パラメーター。このパラメーターは、AMODE 31 で実行されている呼び出し側にのみ適用されます。呼び出し側が AMODE 24 で実行されている場合、このパラメーターは無視され、高位バイトは X'00' として扱われます。

BELOW

UCB パラメーターは、16 MB より下のストレージにある UCB の UCB アドレス、またはキャプチャーされた UCB を含んでいます。これはデフォルトです。

LOC=BELOW が指定されている場合、UCB アドレスの高位バイトは X'00' として扱われます。

ANY UCB パラメーターで受け渡されるアドレスには、3 バイトまたは 4 バイト UCB アドレスが入っています。

31 ビット・モードで呼び出すときに LOC=ANY が指定されている場合、TRKCALC は UCB アドレスを 31 ビット・アドレスとして扱います。

BALANCE=addr-RX タイプ・アドレス、(2-12)、(0)、(14)

トラック・バランスを含んでいるハーフワードのアドレスを指定するか、またはレジスタの下位 2 バイト内のバランスを指定できません。提供される値は、ユーザーが最後に TRKCALC を発行したときに戻された値である可能性があります。R が 1 に等しい場合、バランスはトラック容量にリセットされ、ユーザーが提供した値は無視されます。これは入力値であり、TRKCALC マクロによって変更されません。結果のトラック・バランスは、レジスタ 0 および TRKCALC パラメーター・リストのフィールド STARBAL に戻されます。このパラメーターにユーザーが提供する値は、使用中の装置タイプに対して有効な値でなければなりません。

REMOVE=YES または **NO**

トラックからレコードを削除するかどうかを示します。

YES レコード番号 (R キーワードで指定) によって識別されたレコードをトラックから削除することを指定します。トラック・バランスは、減分ではなく、増分されます。

YES は、FUNCTN=TRKBAL 呼び出しでのみ有効です。

NO トラックからレコードを削除しないことを指定します。NO がデフォルトです。

MAXSIZE=YES または **NO**

YES 指定されたレコードが収まらない場合、収まる、指定のキー長を持つレコードの最大長が戻されます (レジスター 0)。

YES は、FUNCTN=TRKBAL 呼び出しでのみ有効です。

NO 最大サイズは戻されません。NO がデフォルトです。

RKDD=addr-RX タイプ・アドレス、(2-12)、(0)、(14)

addr は、レコード番号 (1 バイト)、キー長 (1 バイト)、およびデータ長 (2 バイト) (それぞれ、バイト 0、1、および 2 と 3) を含んでいるワード、またはレコード番号、キー長、およびデータ長を含んでいるレジスターを指定します。R、K、および DD をこのキーワードで指定するか、または代わりに以下の 3 つのキーワードを使用できます。

R=addr-RX タイプ・アドレス、(2-12)、(0)、(14)、または *n*

レコード番号のアドレスを指定するか、またはレジスターの下位バイトまたは即値データ (*n*) を使用してレコード番号を指定できます。*n* (即値データ) の場合は、10 進数字を指定してください。

K=addr-RX タイプ・アドレス、(2-12)、(0)、(14)、または *n*

レコードのキー長の 16 進値を含んでいるフィールドのアドレスを指定するか、またはレジスターの下位バイトまたは即値データ (*n*) を使用してレコードのキー長を指定できます。*n* (即値データ) の場合は、10 進数字を指定してください。

DDaddr-RX タイプ・アドレス、(2-12)、(0)、(14)、または *n*

レコードのデータ長の 16 進値を含んでいるフィールドのアドレスを指定するか、またはレジスターの下位 2 バイトまたは即値データ (*n*) を使用してレコードのデータ長を指定できます。*n* (即値データ) の場合は、10 進数字を指定してください。

REGSAVE=YES または **NO**

レジスターを保管する必要があるかどうかを指定します。

YES TRKCALC 呼び出しの後、レジスター 1 から 14 を呼び出し側提供の保管域 (レジスター 13 によって指示) に保管し、復元することを指定します。そうしない場合、レジスター 1、9、10、11、および 14 は変更されます。レジスター 0 と 15 は、常に TRKCALC 呼び出しによって変更されます。

NO TRKCALC 呼び出しの後、レジスターを保管しないことを指定します。NO がデフォルトです。

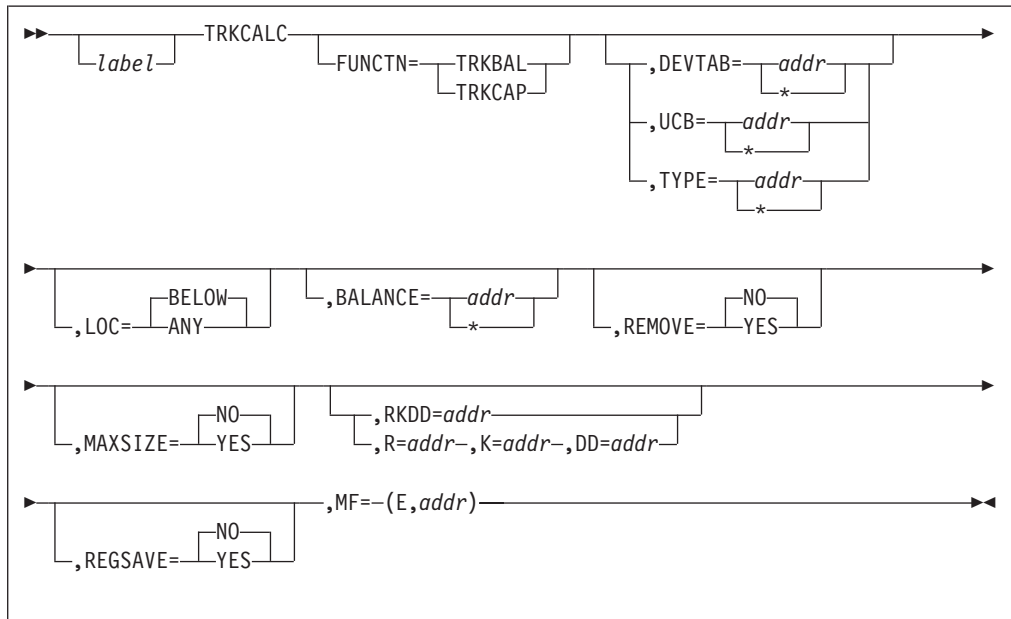
MF=I

指定のキーワードを使用して、TRKCALC パラメーター・リストおよびパラメーター・リスト初期化のストレージ定義を指定し、TRKCALC 機能呼び出しします。MF=I はデフォルトです。

TRKCALC - 実行形式

リモート・パラメーター・リストは、実行形式の TRKCALC マクロによって参照し、変更することができます。TRKCALC ルーチンが呼び出されます。オペランドの機能は、標準形式の場合と同じです。

実行形式の TRKCALC マクロのフォーマットは、次のとおりです。



FUNCTN=TRKBAL または TRKCAP

標準形式と同様にコーディングします。このキーワードが省略されている場合、REMOVE、MAXSIZE、LAST、および RX 形式の BALANCE の指定は無視されます。また、UCB がコーディングされていて、障害が起きた場合、または TYPE が指定されている場合は、DEVTAB が想定されます。FUNCTN を使用する場合、情報源を提供するために、いずれかのキーワード (DEVTAB、UCB、または TYPE) を指定する必要があります。

DEVTAB=*addr* または *-RX タイプ・アドレス、(2-12)、(0)、(14)

* サブパラメーターを除いて、標準形式と同様にコーディングします。パラメーター・リストに装置特性テーブル項目 (DCTE) のアドレスを挿入した場合は、* を指定します。

UCB=*addr* または *-RX タイプ・アドレス、(2-12)、(0)、(14)

* サブパラメーターを除いて、標準形式と同様にコーディングします。パラメーター・リストに UCB のアドレスを挿入した場合は、* を指定します。

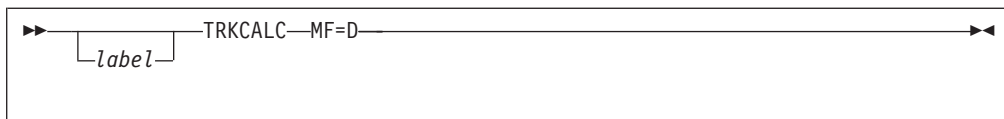
TRKCALC マクロは、UCB または UCB コピーのアドレスを受け入れません。無許可プログラムは、UCBSCAN マクロを使用し、COPY および UCBAREA キーワードを指定することにより、UCB のコピーを入手できます。詳しくは、「z/OS ハードウェア構成定義(HCD) 計画」を参照してください。

TYPE=*addr* または *-RX タイプ・アドレス、(2-12)、(0)、(14)

* サブパラメーターを除いて、標準形式と同様にコーディングします。パラメーター・リストに UCB タイプ (UCBTYP) のアドレスを挿入した場合は、* を指定します。

LOC=BELOW または ANY

標準形式と同様にコーディングします。



すべての形式の MF の入力レジスタの使用法

レジスタ 使用

- 0、2-12、14 キーワードの入力を提供するために使用できます。
- 1 MF=E 呼び出しのパラメーター・リストのアドレスのみを提供します。
- 13 REGSAVE=YES が指定されていない場合、キーワードの入力。
- 15 MF=E 呼び出しの TRKCALC パラメーター・リストを作成するための作業用レジスタ。入力レジスタとしては使用できません。

TRKCALC からの出力

FUNCTN=TRKBAL

出力 意味

- R15=X'00' レコードはトラックに収まります。レジスタ 0 と STARBAL に、新しいトラック・バランスが入ります。
- R15=X'04' レコードはトラックに収まりません。MAXSIZE=YES が指定されている場合、部分レコードも収まりません。レジスタ 0 と STARBAL はゼロに設定されます。
- R15=X'08' レコードはトラックに収まりません。MAXSIZE=YES が指定されており、部分レコードは収まります。レジスタ 0 と STARBAL は、トラックの残りの部分に収まる、指定のキー長を持つデータ・バイトの最大数に設定されます。
キー長は、最大データ・バイトのカウントから除外されます。
- R15=X'0C' ユーザーは装置タイプを提供しましたが、装置特性テーブルは、そのタイプの装置はシステム上に生成されていないことを示しています。レジスタ 0 はゼロに設定されます。
- STARBAL これは TRKCALC パラメーター・リストのトラック・バランス・フィールドです。このフィールドは最初に、R が 1 に等しい場合はトラック容量、R が 1 に等しくない場合は提供された BALANCE 値、また、R が 1 に等しくなく、BALANCE が省略されている場合は、計算されたバランス値に設定されます。STARBAL は、レコードが収まる場合は、新しいトラック・バランスに更新されます。そうでない場合、STARBAL は入力トラック・バランス値のままです。

FUNCTN=TRKCAP

出力 意味

- R15=X'00' レジスタ 0 には、R が 1 に等しい場合は、トラ

ックに収まるレコードの数、R が 1 に等しくない場合は、トラックの残りに収まるレコードの数が入ります。

R15=X'04'

指定された長さのレコードは、全トラック (R が 1 に等しい場合) または部分トラック (R が 1 に等しくない場合) に収まりません。レジスター 0 はゼロに設定されます。

R15=X'0C'

ユーザーは装置タイプを提供しましたが、装置特性テーブルは、そのタイプの装置はシステム上に生成されていないことを示しています。レジスター 0 はゼロに設定されます。

STARBAL

これは TRKCALC パラメーター・リストのトラック・バランス・フィールドです。このフィールドは最初に、R が 1 に等しい場合はトラック容量、R が 1 に等しくない場合は提供された BALANCE 値、また、R が 1 に等しくなく、BALANCE が省略されている場合は計算されたバランス値に設定されます。

TRKCALC からの戻りコード

TRKCALC マクロは、レジスター 15 に戻りコードを渡します。戻りコードとその意味は、次のとおりです。

戻りコード	意味
0 (X'00')	レジスター 0 に新しいトラック・バランスが入っていることを示します。
4 (X'04')	レコードが収まらなかったことを示します (レジスター 0 = 0)。
8 (X'08')	レコードが収まらなかったことを示します。(レジスター 0 には、収まる最大データ長が入ります。)
12 (X'0C')	システムはユーザーが指定した装置タイプの属性に一致する項目を装置特性テーブルで見付けられませんでした。レジスター 0 はゼロに設定されます。

TRKCALC マクロの例

この例では、10 バイトのキーを持つ指定サイズのレコードがいくつ IBM 3380 トラックに収まるかを判別するために、TRKCALC がコーディングされています。マクロを発行した後、レコード数は NUMREC に保管されます。

```

TRKCALC  FUNCTN=TRKCAP,TYPE=UTYPE,R=1,K=10,DD=DL,          X
          MF=(E,(1))
.
.
ST      0,NUMREC  SAVE NUMBER OF RECORDS
.
.
DL      DC  H'xxxx'  DATA LENGTH
UTYPE   DC  X'0E'
NUMREC  DS  F        MAX # OF RECORDS

```

この例では、指定のトラック・バランスで、3380 のトラックに別のレコードが収まるかどうかを判別するために、TRKCALC がコーディングされています。

```
TRKCALC  FUNCTN=TRKBAL,TYPE=UTYPE,R=REC,K=KL,DD=DD,          X
          BALANCE=BAL,MAXSIZE=YES,MF=(E,(1))
.
.
UTYPE   DC   X'0E'
REC     DC   X'xx'
KL      DC   X'xx'
DD      DC   H'xxxx'
BAL     DC   H'xxxx'
```

DFSMS のレベルと名前の判別

IHADFA マッピング・マクロを使用して、DFSMS のレベルと名前を判別できます。これはデータ機能域をマップします。CVT マッピング・マクロを使用して、DFA を指すシンボル CVTDFA を定義します。DFA の DFARELS フィールドは、製品レベルを指定する 4 バイトです。

DFSMS のバージョン、リリース、および修正レベルの判別

DFARELS の最初のバイトには、プログラムを実行している DFSMS のレベルを示す 2 進値が入っています。

値 意味

- 3 プログラムは、z/OS バージョン 1 リリース 3 以上に専用の DFSMS レベルで実行されています。後続の 3 バイトは、その DFSMS が対象とする z/OS のバージョン、リリース、および修正レベルを指定します。他の 3 バイトの値は、X'010300' 以上です。インストールされている z/OS のレベルによって異なります。
- 2 プログラムは、OS/390 または z/OS の最初の 2 つのリリースに専用の DFSMS レベルで実行されています。DFARELS の X'02020A00' の値は、DFSMS for OS/390 バージョン 2 リリース 10、修正レベル 0 を指定します。DFSMS は、z/OS の最初の 2 つのリリースでは変更されなかったため、これらのリリースも X'02020A00' の値を持っています。
- 1 プログラムは DFSMS/MVS で実行されており、後続の 3 バイトは、DFSMS/MVS のバージョン、リリース、および修正レベルを指定します。DFARELS の X'01010200' の値は、DFSMS/MVS バージョン 1、リリース 2、修正レベル 0 を指定します。
- 0 プログラムは DFSMS 上で実行されていません。MVS/XA DFP バージョン 2 または MVS/DFP バージョン 3 で実行されており、後続の 3 バイトにもゼロが入ります。この 2 つの製品では、2 バイト・フィールド DFAREL を調べることによってリリース・レベルを判別できます。DFAREL は、IHADFA のコメントに記述されています。

IBM では、将来の DFA レベルでは、4 バイト DFARELS には前の値より小さい値は含めないようにする意向です。DFARELS のテストの目的が、DFSMS の特定の機能が使用可能かどうかを判別することである場合、プログラムは DFARELS の 4 バイトすべてをテストすることをお勧めします。IBM では、DFARELS の下位 3

バイトの 1 つが前のリリースの対応するバイトより小さい値が入る場合、高位バイトにより大きい値を入れるようにする意向です。

MVS/XA DFP バージョン 2 または MVS/DFP バージョン 3 で実行するように設計されたプログラムとの互換性のために、DFSMS は DFAREL を値 X'3321' に設定します。これは、MVS/DFP バージョン 3、リリース 3、修正レベル 2 を指定します。最後の桁は、システムは実際には DFP 3.3.2 より高いレベルであることを示します。

DFSMS のレベルを判別する代替方法については、263 ページの『DFSMS レベル判別の呼び出し』も参照してください。

データ機能域 (DFA) 制御ブロックのフィールドのレイアウトは、435 ページの『データ機能域 (DFA) フィールド』を参照してください。

DFSMS の名前の判別

DFARELS の値が X'03010300' 以上の場合、システムは z/OS バージョン 1 リリース 3.0 以降であることを示します。これは、フィールド DFAELNMP が DFSMS の名前を含んでいる構造後を指していることを意味します。435 ページの『データ機能域 (DFA) フィールド』の DFAELNM を参照してください。

アセンブラ・マクロ・フェーズでの DFARELS の判別

前に説明したように、プログラムは実行時に DFARELS フィールドをテストできます。これは、特定レベルのシステム上でのみ使用可能な新規マクロ・パラメーターをオプションで使用するプログラムをアセンブルすることを許可しません。プログラムが古いレベルのシステム上でアセンブルされている場合、構文エラーを受け取ります。

この解決策の 1 つは、IHADFA マクロによって設定されるマクロ変数シンボルをテストすることです。シンボルの名前は &IHADFARELS で、文字タイプのグローバル変数シンボルです。プログラムは、IHADFA 呼び出しの後で、その値をテストする必要があります。

プログラムをアセンブルしたとは異なるリリース上で実行できるかどうかは、他のシステム機能によって決まります。一部の新機能については、古いリリースは新機能を見捨てます。その他の新機能は、古いリリース上では失敗します。

IHADFA マクロは、変数シンボル &IHADFARELS を 8 文字値に設定します。値の中の各対の文字は、DFARELS の 1 バイトの 10 進値を表します。EBCDIC 値の「A」から「F」は、数字の正しい照合シーケンスではないため、これは 16 進数字ではありません。

以下は、&IHADFARELS を使用するプログラムの例です。

```

xxxx    CSECT
      .
      .
      .
GBLCL  &IHADFARELS  Set by IHADFA macro to be system level
      IHADFA ,      Set &IHADFARELS and define DFARELS
xxxx    CSECT      Reset CSECT
      .
      .
      .
* Expand one of two macro invocations.  Either works on any DFSMS
* release.  If in 31-bit mode on 1.3 or later, then ANY means a UCB
* may be above the line.  Neither works on DFP Version 3 when assembled
* on DFSMS.
      AIF ('&IHADFARELS' LT '01010300').OLD
* If executing in 31-bit mode on 1.3 or later, this requires that each
* UCB address be 31-bit.  They may point below the line.  On an older
* level of DFSMS, the ANY has no effect.  DFP 3.x will reject it.
      DEVTTYPE UCBLIST=(MYLIST,1,ANY),MF=(E,DEVTLIST)
      AGO .CONT
.OLD   DEVTTYPE UCBLIST=(MYLIST,1),MF=(E,DEVTLIST)
.CONT  ANOP
      .
MYLIST DC    A(0)
DEVTLIST DEVTTYPE ,(DEVINFO,24),MF=L

```

図 27. サンプル &IHADFARELS プログラム

DFSMS/MVS V1R3 より前に出荷された IHADFA マクロは、&IHADFARELS を設定しません。IHADFA が &IHADFARELS を設定しない場合でも、例に示されている手法を使用できます。

この IHADFA を使用して別のマクロ呼び出しを決定する手法は、IHADFA が他のマクロと同じリリースの全マクロ・ライブラリー内にあることを前提としています。異なるリリースまたは製品からのマクロでは正しく働かない場合があります。

以下は、マッピング・マクロがアセンブリー時に必要なシンボルを定義しているかどうかを判別する例です。実行時にプログラムは DFARELS をテストして、実行方法を決めます。

```

      GBLCL  &IHADFARELS  Set by IHADFA macro to be system level
      IHADFA ,      Learn release of assembly & execution
TRKLIST DSECT
      TRKCALC MF=D      DSECT for TRKCALC parameter list
      SPACE 2
* If global symbol &IHADFARELS has a null value or is less than
* 01010300, then TRKCALC did not define a certain symbol.  Since
* other parts of this program use it, it must be defined.
      AIF ('&IHADFARELS' GE '01010300').GOTBIT  Go if newer
STARLOC EQU  X'01'      LOC=ANY.  DEVTAB or UCB may be above line
.GOTBIT ANOP

```

図 28. シンボル定義を判別する例

第 8 章 カートリッジ磁気テープ・サブシステム上のメッセージの表示

この章では、MSGDISP マクロを使用して、ディスプレイを備えた磁気テープ装置上にメッセージを表示する方法について説明します。MSGDISP を使用すると、表示するメッセージとその表示方法 (例えば、定常、明滅) を指定できます。ここでは、標準形式、実行形式、およびリスト形式のマクロについて説明します。マクロの 6 つの主要なパラメーターとその機能は、以下のとおりです。

値	意味
MOUNT	マウント要求の間、ボリュームがロードされて作動可能になるまで、ディスプレイの位置 1 に 'M' を表示します。'M' の後に、ボリューム通し番号とラベル・タイプが続きます。
VERIFY	位置 2 から 8 にボリューム通し番号とラベル・タイプを表示して、ボリュームが受け入れられたことを示します。
RDY	データ・セットがオープンされている間、位置 2 から 7 にテキストを表示します。
DEMOUNT	ボリュームがデマウントされるまで、位置 1 にボリューム処理標識を表示します。
RESET	表示域をクリアします。
GEN	2 つのメッセージを交互に表示するオプションを含めて、全範囲の表示オプションを提供します。

RDY パラメーターを除くすべてのパラメーターは、呼び出し側が監視プログラム状態であるか、ストレージ記憶保護キー 0 から 7 を持っているか、または許可プログラム機能によって許可されていることが必要です。

MSGDISP マクロは、24 ビットまたは 31 ビット・アドレッシング・モードで発行できます。31 ビット・アドレッシング・モードを使用する場合は、すべてのアドレスが有効な 31 ビット・アドレスでなければなりません。

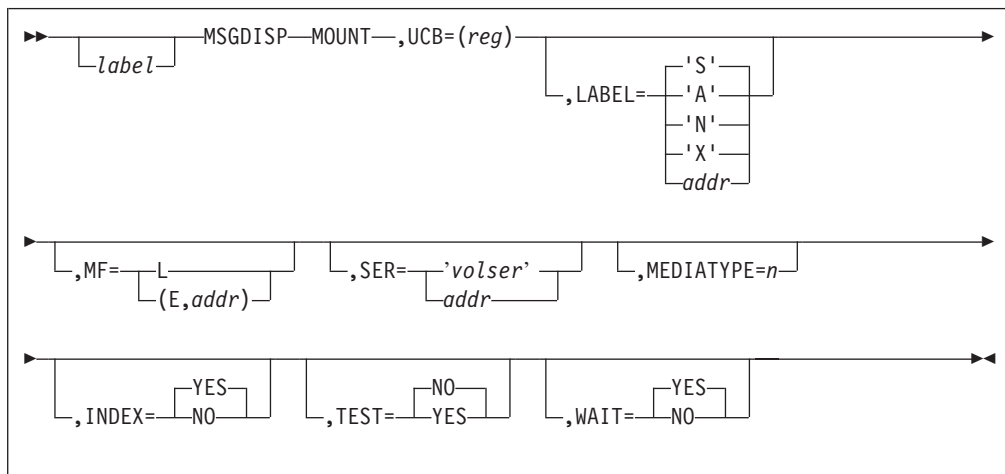
MSGDISP マクロは、メッセージ表示サービス・ルーチンへの入力として、パラメーター・リストを生成します。IGXMSGEX という名前のインストール・システム出口ルーチンをコーディングすることができ、これは MSGDISP が MOUNT、DEMOUNT、VERIFY、または GEN 要求を処理しているときに制御を取得します。出口ルーチンは、表示されるメッセージ・テキスト (2 つの 8 バイト・ストリング) と、フォーマット制御バイトの 1 ビットを変更できます。詳細については、資料「z/OS DFSMS Installation Exits」を参照してください。

MSGDISP マクロ仕様

6 つの主要パラメーターを使用して MSGDISP を指定するためのフォーマット、および MSGDISP によって生成される戻りコードについて、以下で説明します。

MSGDISP - マウント・メッセージの表示

MOUNT パラメーターを持つ MSGDISP マクロのフォーマットは、次のとおりです。



MOUNT マウント要求の間、ディスプレイの位置 1 に 'M' を表示します。'M' の後に、ボリューム通し番号とラベル・タイプが続きます。ボリュームがロードされて作動可能になるまで、ディスプレイはオン・オフで明滅します。マウント要求時に装置が作動可能である場合には、'M' は表示されません。

UCB=(reg)-(2-12)

装置の UCB アドレスが入っているレジスターを指定します。UCB コピーではなく UCB のアドレスを使用してください。

LABEL='A' または 'N' または 'S' または 'X' または addr

マウントされたボリュームのラベル・タイプを位置 8 に表示します。ブランク以外の不明のラベル・タイプを指定した場合は「?」が表示されます。

'A'

ISO/ANSI (AL) またはユーザー・ラベル付き ISO/ANSI (AUL) を指定します。アポストロフィが必要です。

'N'

ラベルなし (NL)、LTM (VSE)、またはラベル迂回処理 (BLP) を指定します。アポストロフィが必要です。

'S' IBM 標準 (SL) またはユーザー・ラベル付き IBM 標準 (SUL) を指定します。アポストロフィが必要です。

'X'

標準外ラベル (NSL) を指定します。アポストロフィが必要です。

addr-RX タイプ・アドレス、A タイプ・アドレス、または (2-12)

「A」、「N」、「S」、または「X」を含んでいる領域のアドレスを指定します。(これらの文字については、下記の説明を参照してください。) MF=L の場合は、A タイプ・アドレスのみを指定できます。

MF=L または **(E,addr)**

実行形式またはリスト形式の MSGDISP を指定します。このパラメーターを指定しない場合、標準形式のマクロが使用されます。

L リスト形式の MSGDISP を指定します。これは、実行形式への入力として使用できるパラメーター・リストを生成します。実行形式は、そのパラメーター・リストを変更できません。

(E,addr)

実行形式のマクロと既存のパラメーター・リストを使用することを指定します。

addr-RX タイプ・アドレス、(1)、または (2-12)

パラメーター・リストのアドレスを指定します。

SER='volser' または *addr*

マウントするボリュームの通し番号を指定します。通し番号は、位置 2 から 7 に表示されます。ユーザーが **SER** を指定しない場合、システムがボリューム通し番号を提供します。通し番号が利用不可の場合、ボリューム使用属性がデフォルトの「PRIVAT」を示していない限り、スクラッチ・ボリュームが使用されます。

'volser' ボリューム通し番号をリテラルとして指定します。アポストロフィで囲んで指定してください。

addr-RX タイプ・アドレス、A タイプ・アドレス、または (2-12)

ボリューム通し番号のアドレスを指定します。MF=L の場合は、A タイプ・アドレスのみを指定できます。

MEDIATYPE=n

SCRTCH または PRIVAT マウントにマウントするメディア・タイプを指定します。MEDIATYPE キーワードは、手動テープ・ライブラリー (MTL) にある装置にボリュームをマウントする場合のみ適用されます。手動テープ・ライブラリー以外にある装置に対して MEDIATYPE を指定しても、無視されます。値は、リテラル、値を含む 1 バイト・フィールドのアドレス、または値を含むアドレス可能フィールドの名前として指定できます。MEDIATYPE の有効な値は、数値 1 から 8 です。

TEST=NO または **YES**

マクロ展開に、UCB をテストしてメッセージ表示がサポートされているかどうかを判別するコードを含めるかどうかを指定します。テストの結果、メッセージ表示がサポートされていない場合、SVC は呼び出されません。

NO マクロ展開には、UCB をテストして装置がメッセージ表示がサポートされているかどうかを判別するコードを含めないこと指定します。

YES メッセージ表示サービス・ルーチンを呼び出す前に、MSGDISP マクロによって UCB をテストすることを指定します。

要件: TEST=YES の場合、UCB マッピング・マクロ (IEFUCBOB) をソース・コードに含める必要があります。

制約事項: TEST=YES を使用する場合は制約があります。AMODE 24 モードで実行されている、TEST=YES パラメーターを使用して MSGDISP マクロを呼び出すプログラムは、16 MB 境界より上にある UCB の実アドレスを渡すことはできません。このようなプログラムは、キャプチャーされた UCB アドレスを渡すか、あるいは、実アドレスを渡す場合は、UCB が 16 MB 境界より下になければなりません。

INDEX=NO または YES

スクラッチ・マウント要求を満たすために、カートリッジ自動挿入機構 (ACL) に指標付けする必要があるかどうかを指定します。

NO ACL の状態に関係なく、指標付けしないことを指定します。

YES 以下の場合、指標付けする必要があることを指定します。

- ACL が存在し、ロードされている。かつ
- 要求が SCRTCH または PRIVAT である。

WAIT=NO または YES

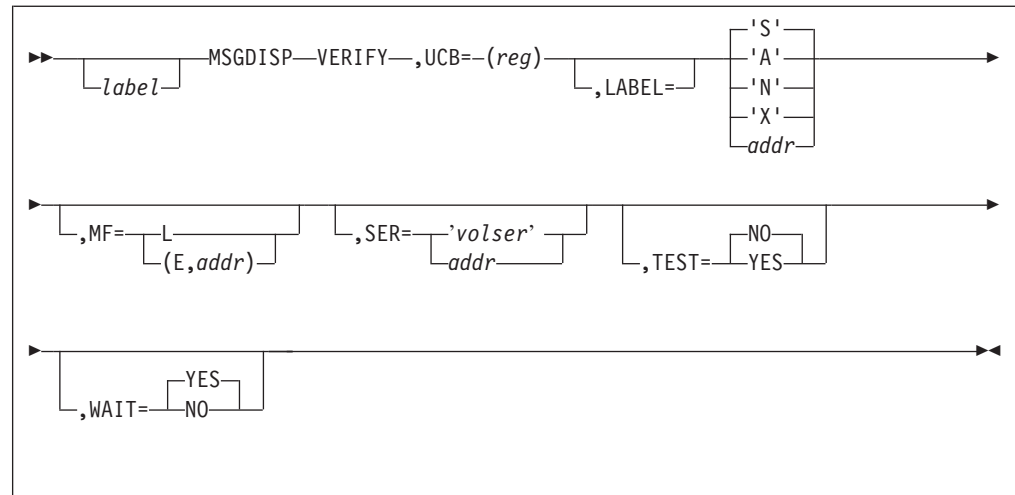
制御をユーザーに戻す時期を指定します。

NO MSGDISP 機能は、呼び出し側のために開始された入出力が完了するのを待たないことを指定します。MSGDISP が戻ったときに、入出力要求はまだ実行中である可能性があります。入出力戻りコードは戻されず、入出力エラーは、エラー・リカバリー手順によって永続エラーと同じ方法で記録されます。

YES 入出力が完了した時点で制御を戻すことを指定します。

MSGDISP - 検査メッセージの表示

VERIFY パラメーターを持つ MSGDISP マクロのフォーマットは、次のとおりです。

**VERIFY**

処理のために受け入れられたボリュームの通し番号とラベル・タイプを表示します。通し番号は、ポッド上の位置 2 から 7 に表示され、1 文字のラベル・タイプは、位置 8 に表示されます。位置 1 はブランクのままです。表示は、次の表示要求が実行されるまで続きます。

UCB=(reg)-(2-12)

装置の UCB アドレスが入っているレジスターを指定します。UCB コピーではなく UCB のアドレスを使用してください。

LABEL='A' または 'N' または 'S' または 'X' または addr

マウントされたボリュームのラベル・タイプをディスプレイの位置 8 に指定します。ブランク以外の不明ラベル・タイプが指定された場合は「?」が表示されます。

'A'

ISO/ANSI (AL) またはユーザー・ラベル付き ISO/ANSI (AUL) を指定します。アポストロフィが必要です。

'N'

ラベルなし (NL)、LTM (VSE)、またはラベル迂回処理 (BLP) を指定します。アポストロフィが必要です。

'S'

IBM 標準 (SL) またはユーザー・ラベル付き IBM 標準 (SUL) ラベルを指定します。アポストロフィが必要です。

'X'

標準外ラベル (NSL) を指定します。アポストロフィが必要です。

addr-RX タイプ・アドレス、A タイプ・アドレス、または (2-12)

「A」、「N」、「S」、または「X」を含んでいる領域のアドレスを指定します (これらの文字の説明は、下記を参照)。MF=L の場合は、A タイプ・アドレスのみを指定できます。

MF=L または **(E,addr)**

実行形式またはリスト形式の MSGDISP を指定します。このパラメーターを指定しない場合、標準形式のマクロが使用されます。

L リスト形式の MSGDISP を指定します。これは、実行形式への入力とし

て使用できるパラメーター・リストを生成します。実行形式は、そのパラメーター・リストを変更できます。

(E,addr)

実行形式のマクロと既存のパラメーター・リストを使用することを指定します。

addr-RX タイプ・アドレス、(1)、または (2-12)
パラメーター・リストのアドレスを指定します。

SER='volser' または *addr*

検査されたボリュームの通し番号を指定します。通し番号は、位置 2 から 7 に表示されます。ユーザーが **SER** を指定しない場合、システムがボリューム通し番号を提供します。通し番号が利用不可の場合、ボリューム使用属性がデフォルトの「PRIVAT」を示していない限り、スクラッチ・ボリュームが使用されます。

'volser' ボリューム通し番号をリテラルとして指定します。アポストロフィで囲んで表現してください。

addr-RX タイプ・アドレス、**A** タイプ・アドレス、または (2-12)
ボリューム通し番号のアドレスを指定します。MF=L の場合は、**A** タイプ・アドレスのみを指定できます。

TEST=NO または **YES**

マクロ展開に、**UCB** をテストしてメッセージ表示がサポートされているかどうかを判別するコードを含めるかどうかを指定します。テストの結果、メッセージ表示がサポートされていない場合、**SVC** は呼び出されません。

NO マクロ展開には、**UCB** をテストして装置がメッセージ表示がサポートされているかどうかを判別するコードを含めないこと指定します。

YES メッセージ表示サービス・ルーチンを呼び出す前に、**MSGDISP** マクロによって **UCB** をテストすることを指定します。

要件: **TEST=YES** の場合、**UCB** マッピング・マクロ (**IEFUCBOB**) をソース・コードに含めることが必要です。この要件に従わない場合、展開コードでプログラム・チェックが生じることがあります。**AMODE 24** モードで実行されている、**TEST=YES** パラメーターを使用して **MSGDISP** マクロを呼び出すプログラムは、16 MB 境界より上にある **UCB** の実アドレスを渡すことはできません。このようなプログラムは、キャプチャーされた **UCB** アドレスを渡すか、あるいは、実アドレスを渡す場合は、**UCB** が 16 MB 境界より下になければなりません。

WAIT=NO または **YES**

制御をユーザーに戻す時期を指定し、**MSGDISP** 機能は、呼び出し側のために開始された入出力が完了するのを待たないことを指定します。**MSGDISP** が戻ったときに、入出力要求はまだ実行中である可能性があります。

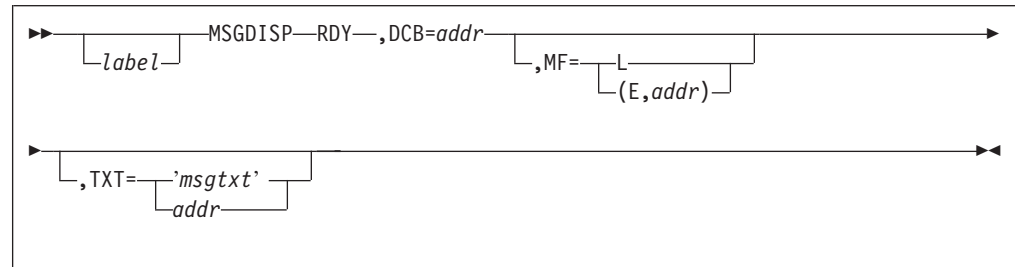
NO **MSGDISP** 機能は、呼び出し側のために開始された入出力が完了するのを待たないことを指定します。**MSGDISP** が戻ったときに、入出力要求はまだ実行中である可能性があります。入出力戻りコード

は戻されず、入出力エラーは、エラー・リカバリー手順によって永続エラーと同じ方法で記録されます。

YES 入出力が完了した時点で制御を戻すことを指定します。

MSGDISP - 作動可能メッセージの表示

RDY パラメーターを持つ MSGDISP マクロのフォーマットは、次のとおりです。



RDY データ・セットがオープンされている間、TXT パラメーターで提供されたテキストを、位置 2 から 7 に表示します。表示は定常状態で (明滅しない)、括弧で囲われています。表示はテープ・プール・コンソールにも書き出されます (宛先コード 3、記述子コード 7)。

DCB=addr マウントされたボリューム上のデータ・セットに対してオープンされた DCB のアドレスを指定します。複数の装置が割り振られている場合、メッセージ表示は、現在使用中のボリュームを含んでいる装置に送られます。

ヒント: 複数の装置または複数のボリュームが割り振られている場合、DCB 出口リストに指定された EOVS ユーザー出口を使用して、ボリューム終了 (end-of-volume) 条件の後で、メッセージ表示を更新できます。非類似特性を持つ連結データ・セットの場合、DCB OPEN 出口を使用して表示を更新することもできます。

addr-RX タイプ・アドレス、A タイプ・アドレス、または (2-12) オープンされた DCB のアドレスを指定します。MF=L の場合は、A タイプ・アドレスのみを指定できます。

MF=L または **(E,addr)**

実行形式またはリスト形式の MSGDISP を指定します。このパラメーターが指定されていない場合、標準形式のマクロが使用されます。

L リスト形式の MSGDISP を指定します。これは、実行形式への入力として使用できるパラメーター・リストを生成します。実行形式は、そのパラメーター・リストを変更できません。

(E,addr)

実行形式のマクロと既存のパラメーター・リストを使用することを指定します。

addr-RX タイプ・アドレス、(1)、または (2-12) パラメーター・リストのアドレスを指定します。

TXT='msgtxt' または *addr*

ディスプレイの位置 2 から 7 に表示する最大 6 文字を指定します。TXT を指定しない場合、ブランクが表示されます。

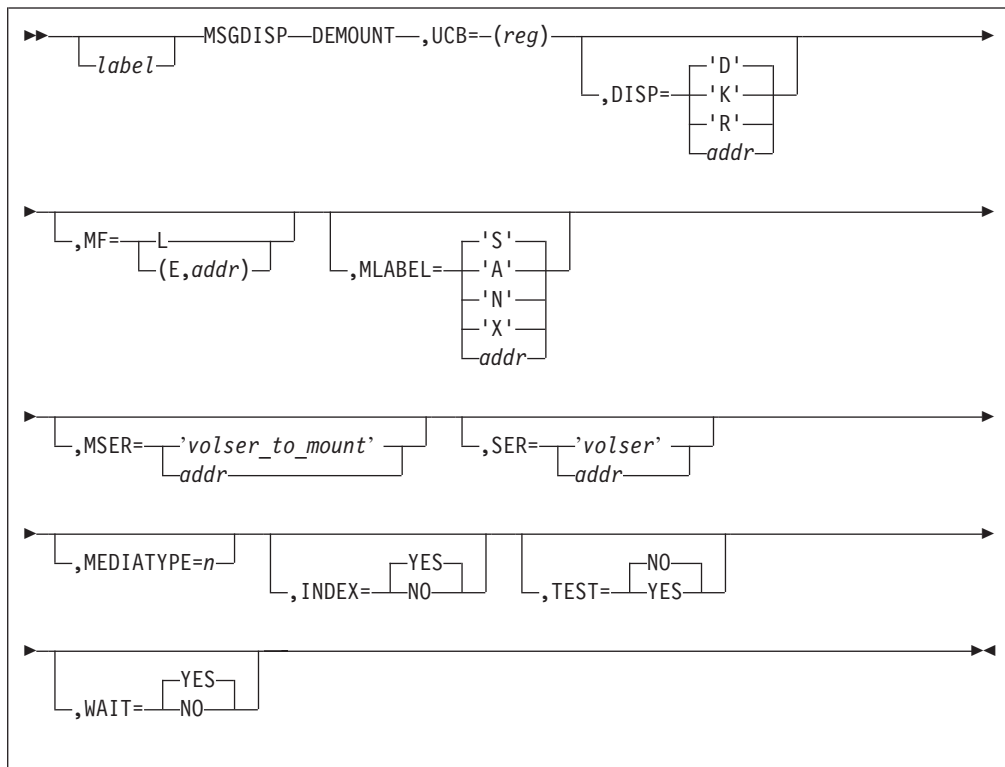
'msgtxt'

テキストをリテラルとして指定します。アポストロフィで囲んで表現してください。

addr-RX タイプ・アドレス、A タイプ・アドレス、または (2-12) 表示するテキストを含んでいる領域のアドレスを指定します。MF=L の場合は、A タイプ・アドレスのみを指定できます。

MSGDISP - デマウント・メッセージの表示

DEMOUNT パラメーターを持つ MSGDISP マクロのフォーマットは、次のとおりです。



DEMOUNT

ボリュームがデマウントされるまで、位置 1 にボリューム処理標識を表示します。オプションで、デマウントするボリュームの通し番号を同時に表示できます。表示はオン/オフで明滅します。表示要求が実行されたときにボリュームが装置にマウントされていない場合、ブランクが表示されます。

MSER パラメーターを指定すると、デマウント・メッセージと次のボリュームのマウント・メッセージを交互に (明滅) 表示できます。

UCB=(reg)-(2-12)

装置の UCB アドレスが入っているレジスターを指定します。UCB コピーではなく UCB のアドレスを使用してください。

DISP='D' または **'K'** または **'R'** または *addr*

ポッドの位置 1 に表示する文字 (ボリューム処理を表す) を指定します。

'D'

公用ボリュームをデマウントすることを指定します。アポストロフィが必要です。「D」は、無効文字を指定した場合、またはボリューム使用属性が不明の場合 (ラベルの読み取り時の自動ボリューム認識 (AVR) エラーなど) にも表示されます。

'K'

専用ボリュームを保持し、それをライブラリーに戻すことを指定します。アポストロフィが必要です。

'R'

専用ボリュームは将来使用するために装置の近くに保持することを指定します。アポストロフィが必要です。

addr-RX タイプ・アドレス、A タイプ・アドレス、または (2-12)

「D」、「K」、または「R」を含んでいる領域のアドレスを指定します。

MF=L の場合は、A タイプ・アドレスのみを指定できます。

MF=L または **(E,*addr*)**

実行形式またはリスト形式の **MSGDISP** を指定します。このパラメーターを指定しない場合、標準形式のマクロが使用されます。

L リスト形式の **MSGDISP** を指定します。これは、実行形式への入力として使用できるパラメーター・リストを生成します。実行形式は、そのパラメーター・リストを変更できます。

(E,*addr*)

実行形式のマクロと既存のパラメーター・リストを使用することを指定します。

addr-RX タイプ・アドレス、(1)、または (2-12)

パラメーター・リストのアドレスを指定します。

MLABEL='A' または **'N'** または **'S'** または **'X'** または *addr*

デマウントの後でロードして作動可能にするボリュームのラベル・タイプを、位置 8 に表示します。ユーザーがブランク以外の不明のラベル・タイプを指定した場合は「?」が表示されます。このパラメーターは、**MSER** パラメーターも指定した場合にのみを指定できます。

'A'

ISO/ANSI (AL) またはユーザー・ラベル付き ISO/ANSI (AUL) を指定します。アポストロフィが必要です。

'N'

ラベルなし (NL)、LTM (VSE によって作成される、先行テープ・マーク)、またはラベル迂回処理 (BLP) を指定します。アポストロフィが必要です。

'S'

IBM 標準 (SL) またはユーザー・ラベル付き IBM 標準 (SUL) ラベルを指定します。アポストロフィが必要です。

'X'

標準外ラベル (NSL) を指定します。アポストロフィが必要です。

addr-RX タイプ・アドレス、*A* タイプ・アドレス、または (2-12)

「A」、「N」、「S」、または「X」を含んでいる領域のアドレスを指定します (これらの文字の説明は、下記を参照)。MF=L の場合は、*A* タイプ・アドレスのみを指定できます。

MSER='volser-to-mount' または *addr*

次のボリュームのマウント・メッセージをデマウント・メッセージと交互に (明滅) 表示します。現行ボリュームをデマウントするまで、表示は続きます。デマウントされると、ユーザーがボリュームをロードして装置を作動可能にするまで、マウント・メッセージが表示 (明滅) されます。デマウントされた時点でボリュームがマウントされず、マウント・メッセージが実行されている場合、ボリュームがロードされて作動可能になるまで、マウント・メッセージのみが表示 (明滅) されます。

'volser-to-mount'

マウントするボリュームのボリューム通し番号をリテラルとして指定します。アポストロフィが必要です。

addr-RX タイプ・アドレス、*A* タイプ・アドレス、または (2-12)

マウントするボリュームのボリューム通し番号のアドレスを指定します。MF=L の場合は、*A* タイプ・アドレスのみを指定できます。

SER='volser' または *addr*

デマウントするボリュームの通し番号を指定します。通し番号は、位置 2 から 7 に表示されます。ユーザーが **SER** を指定しない場合、システムがボリューム通し番号を提供します。通し番号が利用不可の場合、ボリューム使用属性がデフォルトの「PRIVAT」を示していない限り、スクラッチ・ボリュームが使用されます。

'volser'

ボリューム通し番号をリテラルとして指定します。アポストロフィで囲んで指定してください。

addr-RX タイプ・アドレス、*A* タイプ・アドレス、または (2-12)

ボリューム通し番号のアドレスを指定します。このパラメーターは、MF=L 形式に対しては無効です。MF=L の場合は、*A* タイプ・アドレスのみを指定できます。

MEDIATYPE=n

SCRATCH または PRIVAT デマウントで取り外すメディア・タイプを指定します。MEDIATYPE キーワードは、手動テープ・ライブラリー (MTL) にある装置でボリュームを取り外す場合のみ適用されます。手動テープ・ライブラリー以外にある装置に対して MEDIATYPE を指定しても、無視されます。値は、リテラル、値を含む 1 バイト・フィールドのアドレス、または値を含むアドレス可能フィールドの名前として指定できます。MEDIATYPE の有効な値は、数値 1 から 8 です。

INDEX=NO または **YES**

スクラッチ・マウント要求を満たすために、ACL に指標付けする必要があるかどうかを指定します。

NO ACL の状態に関係なく、指標付けしないことを指定します。

YES 以下の場合、指標付けする必要があることを指定します。

- ACL が存在し、ロードされている。かつ
- 要求が SCRTCH または PRIVAT である。

TEST=NO または YES

マクロ展開に、UCB をテストしてメッセージ表示がサポートされているかどうかを判別するコードを含めるかどうかを指定します。テストの結果、メッセージ表示がサポートされていない場合、SVC は呼び出されません。

NO マクロ展開には、UCB をテストして装置がメッセージ表示がサポートされているかどうかを判別するコードを含めないこと指定します。

YES メッセージ表示サービス・ルーチンを呼び出す前に、MSGDISP マクロによって UCB をテストすることを指定します。

要件: TEST=YES の場合、UCB マッピング・マクロ (IEFUCBOB) をソース・コードに含めることが必要です。この要件に従わない場合、展開コードでプログラム・チェックが生じることがあります。AMODE 24 モードで実行されている、TEST=YES パラメータを使用して MSGDISP マクロを呼び出すプログラムは、16 MB 境界より上にある UCB の実アドレスを渡すことはできません。このようなプログラムは、キャプチャーされた UCB アドレスを渡すか、あるいは、実アドレスを渡す場合は、UCB が 16 MB 境界より下になければなりません。

WAIT=NO または YES

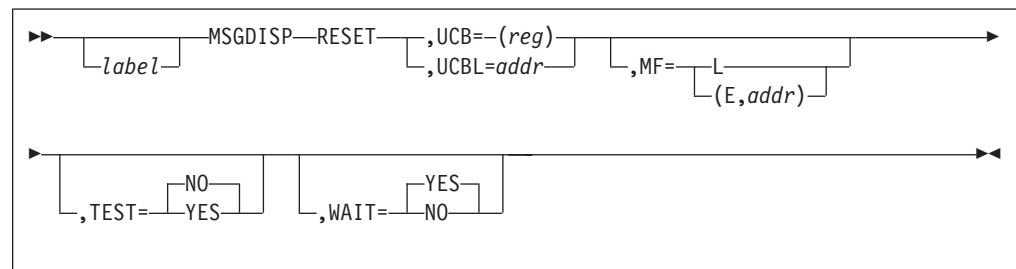
制御をユーザーに戻す時期を指定します。

NO MSGDISP 機能は、呼び出し側のために開始された入出力が完了するのを待たないことを指定します。MSGDISP が戻ったときに、入出力要求はまだ実行中である可能性があります。入出力戻りコードは戻されず、入出力エラーは、エラー・リカバリー手順によって永続エラーと同じ方法で記録されます。

YES 入出力が完了した時点で制御を戻すことを指定します。

MSGDISP - メッセージ表示のリセット

RESET パラメータを持つ MSGDISP マクロのフォーマットは、次のとおりです。



RESET

ディスプレイ上の既存のデータをすべてクリアします。WAIT=NO を指定し、要求した最後のサービスがデマウントの場合、ディスプレイはクリアされません。

クリアされた後、ディスプレイは装置の内部状況 (例えば、装置が作動可能であることを示すメッセージ) を表示します。

UCB=(reg)-(2-12)

装置の UCB アドレスが入っているレジスターを指定します。UCB コピーではなく UCB のアドレスを使用してください。

UCBL=addr-RX タイプ・アドレス、A タイプ・アドレス、(0)、または (2-12)

最大 64 ワードを含むリストのアドレスを指定します。リスト内の各ワードには、そのディスプレイをリセットする装置を表す UCB のアドレスが入ります。リストの終了は、リストの最後のアドレスの高位ビット内の 'I' によって示されます。リストの処理中にエラーが検出された場合、制御を再取得したときにレジスター 1 が関連の UCB を指しています。

TEST=YES および WAIT=NO と一緒に UCBL を指定することはできません。

MF=L または **(E,addr)**

実行形式またはリスト形式の MSGDISP を指定します。このパラメーターを指定しない場合、標準形式のマクロが使用されます。

L リスト形式の MSGDISP を指定します。これは、実行形式への入力として使用できるパラメーター・リストを生成します。実行形式は、そのパラメーター・リストを変更できます。

(E,addr) 実行形式のマクロと既存のパラメーター・リストを使用することを指定します。

addr-RX タイプ・アドレス、(1)、または (2-12)
パラメーター・リストのアドレスを指定します。

TEST=NO または **YES**

マクロ展開に、UCB をテストしてメッセージ表示がサポートされているかどうかを判別するコードを含めるかどうかを指定します。テストの結果、メッセージ表示がサポートされていない場合、SVC は呼び出されません。

NO マクロ展開には、UCB をテストして装置がメッセージ表示がサポートされているかどうかを判別するコードを含めないこと指定します。NO がデフォルトです。

YES メッセージ表示サービス・ルーチンを呼び出す前に、MSGDISP マクロによって UCB をテストすることを指定します。UCBL パラメーターを指定する場合、TEST=YES は指定できません。

要件: TEST=YES の場合、UCB マッピング・マクロ (IEFUCBOB) をソース・コードに含めることが必要です。この要件に従わない場合、展開コードでプログラム・チェックが生じることがあります。AMODE 24 モードで実行されている、TEST=YES パラメーターを使用して MSGDISP マクロを呼び出すプログラムは、16 MB 境界より上にある UCB の実アドレスを渡すことはできません。このようなプログラムは、キ

ャプチャーされた UCB アドレスを渡すか、あるいは、実アドレスを渡す場合は、UCB が 16 MB 境界より下になければなりません。

WAIT=NO または YES

制御をユーザーに戻す時期を指定します。

NO MSGDISP 機能は、呼び出し側のために開始された入出力が完了するのを待たないことを指定します。MSGDISP が戻ったときに、入出力要求はまだ実行中である可能性があります。入出力戻りコードは戻されず、入出力エラーは、エラー・リカバリー手順によって永続エラーと同じ方法で記録されます。

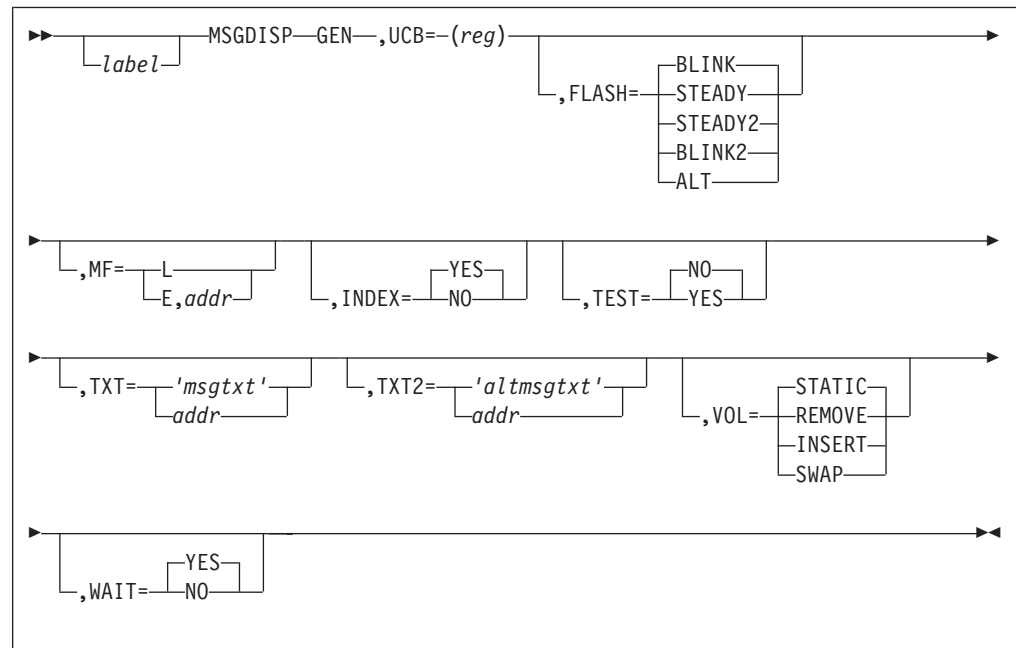
UCBL パラメーターを指定する場合、WAIT=NO は指定できません。

YES 入出力が完了した時点で制御を戻すことを指定します。

WAIT=YES が指定されている場合にのみ、デマウント・メッセージをリセットできます。

MSGDISP - 全範囲の表示オプションの提供

GEN パラメーターを持つ MSGDISP マクロのフォーマットは、次のとおりです。



カートリッジ・メッセージ

GEN 全範囲の表示オプションを指定します。

UCB=(reg)-(2-12)

装置の UCB アドレスが入っているレジスターを指定します。UCB コピーではなく UCB のアドレスを使用してください。

FLASH=STEADY または **STEADY2** または **BLINK** または **BLINK2** または **ALT** メッセージ表示モードを指定します。

ヒント: **VOL=SWAP** を指定すると、**FLASH=ALT** が指定された場合と同様に、メッセージは常に表示されます。

STEADY 1 次メッセージ (TXT) を明滅せずに表示することを指定します。

STEADY2 代替メッセージ (TXT2) を明滅せずに表示することを指定します。

BLINK 1 次メッセージ (TXT) を、約 2 秒間オン、1/2 秒間オフの速度で、明滅して表示することを指定します。

BLINK2 代替メッセージ (TXT2) を、約 2 秒間オン、1/2 秒間オフの速度で、明滅して表示することを指定します。

ALT 1 次メッセージと代替メッセージ (TXT と TXT2) を、約 2 秒間オン、1/2 秒間オフの速度で、明滅して表示することを指定します。

MF=L または **(E,addr)**

実行形式またはリスト形式の **MSGDISP** を指定します。このパラメーターを指定しない場合、標準形式のマクロが使用されます。

L リスト形式の **MSGDISP** を指定します。これは、実行形式への入力として使用できるパラメーター・リストを生成します。実行形式は、そのパラメーター・リストを変更できます。

(E,addr) 実行形式のマクロと既存のパラメーター・リストを使用することを指定します。

addr

パラメーター・リストのアドレスを指定します。RX タイプのアドレス、または 2 から 12 の範囲のレジスターを指定します。

INDEX=NO または **YES**

スクラッチ・マウント要求を満たすために、**ACL** に指標付けする必要があるかどうかを指定します。

NO **ACL** の状態に関係なく、指標付けしないことを指定します。

YES 以下の場合、指標付けする必要があることを指定します。

- **ACL** が存在し、ロードされている。かつ
- 要求が **SCRATCH** または **PRIVATE** である。

TEST=NO または **YES**

装置がメッセージを表示できるかどうかを判別するために UCB をテストするかどうかを指定します。

NO マクロ展開には、UCB をテストして装置がメッセージ表示がサポートされているかどうかを判別するコードを含めないこと指定します。

YES メッセージ表示サービス・ルーチンを呼び出す前に、MSGDISP マクロによって UCB をテストすることを指定します。

要件: TEST=YES の場合、UCB マッピング・マクロ (IEFUCBOB) をソース・コードに含めることが必要です。この要件に従わない場合、展開コードでプログラム・チェックが生じることがあります。AMODE 24 モードで実行されている、TEST=YES パラメーターを使用して MSGDISP マクロを呼び出すプログラムは、16 MB 境界より上にある UCB の実アドレスを渡すことはできません。このようなプログラムは、キャプチャーされた UCB アドレスを渡すか、あるいは、実アドレスを渡す場合は、UCB が 16 MB 境界より下になければなりません。

TXT='msgtxt' または *addr*

ディスプレイの位置 1 から 8 に表示する 8 文字を指定します。TXT を指定しない場合、ブランクが表示されます。

'msgtxt'

8 文字をリテラルとして指定します。アポストロフィが必要です。

addr-RX タイプ・アドレス、A タイプ・アドレス、または (2-12) 8 文字を含む領域のアドレスを指定します。MF=L の場合は、A タイプ・アドレスのみを指定できます。

TXT2='altmsgtxt' または *addr*

ディスプレイの位置 1 から 8 に表示する代替の 8 文字を指定します。TXT2 を指定しない場合、ブランクが表示されます。

'altmsgtxt'

8 文字をリテラルとして指定します。アポストロフィが必要です。

addr-RX タイプ・アドレス、A タイプ・アドレス、または (2-12) 8 文字を含む領域のアドレスを指定します。MF=L の場合は、A タイプ・アドレスのみを指定できます。

VOL=STATIC または **REMOVE** または **INSERT** または **SWAP**

ボリューム状況に基づいて、メッセージ表示モードを指定します。

STATIC ボリューム状況に関係なく、次のメッセージ要求が実行されるまで、または次のコマンドがボリューム移動を開始するまで、メッセージを表示することを指定します。

- REMOVE** 現行ボリュームがデマウントされるまで、メッセージを表示することを指定します。要求が実行されるときにボリュームがマウントされていない場合、このパラメーターは無視されます。
- INSERT** ボリュームが存在し、テープがスレッド化され、アクティブ/非アクティブ・スイッチがアクティブ位置になるまで、メッセージを表示することを指定します。要求が実行されるときにボリュームがロードされて作動可能状態の場合、このパラメーターは無視されます。
- SWAP** FLASH=ALT が指定された場合と同様に、メッセージを常に表示することを指定します。現行ボリュームがデマウントされるまで、TXT および TXT2 デイスプレーからのデータが交互に (明滅) 表示されます。デマウントされると、新規ボリュームがロードされて作動可能になるまで、TXT2 データのみを表示 (明滅) します。このパラメーターが指定されているときにボリュームがマウントされていない場合、新規ボリュームがロードされて作動可能になるまで、TXT2 データのみを表示 (明滅) します。

WAIT=NO または YES

制御をユーザーに戻す時期を指定します。

- NO** MSGDISP 機能は、呼び出し側のために開始された入出力が完了するのを待たないことを指定します。MSGDISP が戻ったときに、入出力要求はまだ実行中である可能性があります。入出力戻りコードは戻されず、入出力エラーは、エラー・リカバリー手順によって永続エラーと同じ方法で記録されます。
- YES** 入出力が完了した時点で制御を戻すことを指定します。これはデフォルトです。

MSGDISP からの戻りコード

システムが制御を問題プログラムに戻すときに、レジスター 15 の下位バイトに戻りコードが入ります。レジスター 0 の下位バイトに理由コードが入ることがあります。これらのコードの説明を、下の表に示します。

戻りコード	理由コード	意味
0 (X'00')		正常終了。
4 (X'04')		装置は MSGDISP をサポートしない。
8 (X'08')	1 (X'01')	無効な入力パラメーター。
	2 (X'02')	無効な DCB または DEBCHK エラー。
	3 (X'03')	環境エラー。
	4 (X'04')	許可 (TESTAUTH) 違反。
	5 (X'05')	無効な UCB。UCB コピーではなく UCB のアドレスが必要です。
	6 (X'06')	無効な要求。
	11 (X'0B')	ESTAE マクロ呼び出しの失敗。

戻りコード	理由コード	意味
12 (X'0C')	12 (X'0C')	GETMAIN 要求の失敗。 入出力エラー (システムは要求にエラーを通知しました)。

ドライブ・ディスプレイにハードウェア障害がある場合、ディスプレイのロード時に入出力エラーが発生します。

RESET UCBL 操作で戻りコード X'04' または X'0C' を受け取った場合、制御を再取得したときに、レジスター 1 はそのエラーに関連した UCB を指しています。

第 9 章 DFSMSdfp™ 呼び出し可能サービスの使用

この章では、DFSMSdfp 関連の呼び出し可能サービスについて説明します。呼び出し可能サービスは、SYS1.CSSLIB (呼び出し可能システム・サービス・ライブラリー) に常駐しています。これを呼び出すには、ユーザーのプログラムから CALL ステートメントを発行し、必要なサービスおよびサービス固有の引数とストレージ域のリストを識別するパラメーターを指定します。この章で説明するルーチンユーザーのプログラムとリンク・エディットした場合、ルーチンは将来のレベルのオペレーティング・システムでも引き続き正しく実行できるようにする意向です。

DFSMSdfp 呼び出し可能サービスは、14 の呼び出し可能システム・サービスをサポートします。これは、LE/370 によってサポートされる高水準言語およびアセンブラ言語の呼び出し側プログラムによって呼び出すことができます。

注: IGWAQSMS は、これらのサービスによって置き換えられ、使用可能でなくなりました。他の 9 つの機能は、Character Data Representation Architecture (CDRA) ID の使用に関連したものです。これらは、図形文字データを一貫して正しく処理するために必要なアプリケーション・プログラム・インターフェース (API) です。ここでは、そのリストを示しますが (262 ページの『Character Data Representation Architecture (CDRA) API』を参照)、使用法についての詳しい情報は、「*Character Data Representation Architecture Reference and Registry*」を参照してください。

これらのサービスを使用すると、アセンブラ言語または高水準言語で書かれたプログラムは、以下のことができます。

- IGWASYS を使用して、システム上の DFSMSdfp のバージョン、リリース、および修正レベル、ならびに SMS サブシステムの状況を判別する。
- IGWASMS を使用して、SMS 管理データ・セットの特定のデータ・セット属性を戻す。データ・セット属性には、次のものが含まれます。
 - SMSDATA-SMS クラス名、つまり、ストレージ・クラス、管理クラス、およびデータ・クラス。
 - DSTYPE-現行では、データ・セットが PDSE タイプ・データ・セットか、HFS か、あるいはそのどちらでもないかを示します。

注: 非 SMS 管理データ・セットに対して IGWASMS が呼び出された場合、DSTYPE 属性はゼロが戻されます。

- IGWABWO を使用して、DFSMSdfp と通信し、さまざまなデータ・セット関連の標識を検索または設定する。この標識を使用して、ユーザーのプログラムは、更新用にオープンされているデータ・セットがバックアップに適格であるかどうかを判別し、適格である場合、実行できるまたは実行する必要があるアクションを判別できます。標識の説明については、274 ページの表 55 を参照してください。
- IGWLSHR を使用して、現行システム上で使用中の DFSMSdfp 共用属性を判別する。

呼び出し可能サービス

- IGWARLS を使用して、カタログ内のレコード・レベル共用 (RLS) に関する情報を照会する。これには、LOG、BWO、および LOGSTREAMID パラメーターの値、VSAM QUIESCED 標識、RLS_RECOVERY_TIMESTAMP フィールド、および スフィアが順方向リカバリーを必要とするかどうかが含まれます。

ここでは、アセンブラー言語で書かれた再入不可プログラムからサービスを呼び出す方法について説明します。再入可能プログラムの CALL の用法については、「z/OS MVS プログラミング: アセンブラー・サービス 解説書 ABE-HSP」を参照してください。高水準言語で書かれたプログラムの CALL の用法については、該当する言語の資料を参照してください。

プログラムは、DFSMSdfp 呼び出し可能サービスを 24 ビットまたは 31 ビット AMODE で呼び出すことができます。プログラムは、任意の記憶保護キーで実行でき、監視プログラム状態または問題プログラム状態のいずれでも構いません。呼び出し可能サービスを呼び出す場合、プログラムは、標準 18 ワード保管域のアドレスをレジスター 13 に提供する必要があります。ここに示す構文図は、アセンブラー言語の CALL ステートメントを示しています。呼び出し可能サービスは、次の 2 とおりの方法で呼び出すことができます。

- 呼び出しアプリケーションに CALL ステートメントをコーディングする。呼び出し可能サービスの IGWASYS、IGWASMS、IGWABWO、IGWLSHR、または IGWARLS は、その他のものと共に、SYS1.CSSLIB 内にあります。呼び出しアプリケーションをリンク・エディットする際に、ライブラリー連結に SYS1.CSSLIB を指定してください。
- 呼び出しアプリケーションで必要なサービス (IGWASYS、IGWASMS、IGWABWO、IGWLSHR、または IGWARLS) に対する LINK または LOAD/CALL を発行する。LINK マクロの使用例は、268 ページを参照してください。CALL および LOAD/CALL マクロの使用例は、266 および 271 ページを参照してください。

呼び出し可能サービスを呼び出すには、呼び出し時に一連の引数を特定の順序で指定します。各呼び出し可能サービスに関連した引数の数は固定され、引数のタイプは 32 ビット 2 進整数 (以後、これを整数と呼びます) および固定長 EBCDIC 文字ストリングに制限されます。CALL ステートメントのフォーマットについては、後節で説明します。

Character Data Representation Architecture (CDRA) API

以下の CDRA API は、DFSMS 製品ライブラリーに含まれています。API とその使用に関する詳しい説明は、「*Character Data Representation Architecture Reference and Registry (SC09-2190)*」を参照してください。

API	説明
CDRGESP	エンコード・スキーム、文字セット、およびコード・ページ・エレメントの取得
CDRSCSP	指定の ES (CS、CP) から短形式 (CCSID) の取得
CDRGESE	エンコード・スキーム・エレメントとそのサブエレメントの取得
CDRGCTL	制御機能定義の取得
CDRSMXC	指定の ES、CP の最大 CS をもつ短形式 (CCSID) の取得
CDRMSCI	複数ステップ変換の初期化

CDRMSCP	複数ステップ変換の実行
CDRMSCC	複数ステップ変換のクリーンアップ
CDRXSRF	フィードバック・コードから状況コードと理由コードの抽出

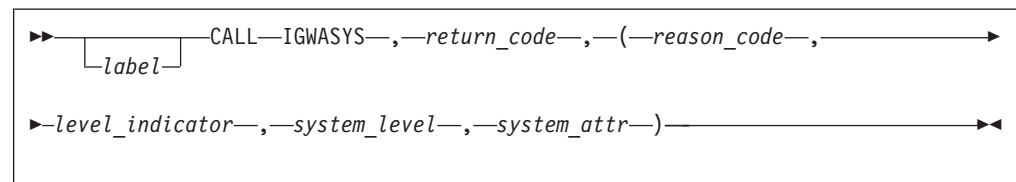
DFSMS レベル判別の呼び出し

DFSMS レベル判別呼び出し (IGWASYS) は、DFSMS のバージョン、リリース、および修正レベルを戻します。DFSMS を含む製品の名前を表すコード番号も戻します。この 4 つの数字は、DFSMS を実行する設計環境を表します。このレベルは、オペレーティング・システムの残りの部分のリリースより以前である可能性があります。

システムのレベルを判別するための代替手法が、240 ページの『DFSMS のレベルと名前の判別』に説明されています。

フォーマット

システム属性 IGWASYS 呼び出しのフォーマットは、次のとおりです。



パラメーター

return_code

IGWASYS からの戻りコード。戻りコードはレジスター 15 にも戻されません。説明は、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』を参照してください。これは、整数として定義する必要がある、出力引数です。

reason_code

IGWASYS サービスからの理由コード。理由コードはレジスター 0 にも戻されません。説明は、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』を参照してください。これは、整数として定義する必要がある、出力引数です。

level_indicator

レベル情報が要求されている製品。これは、入力引数です。level_indicator は整数として定義してください。MVS/XA DFP バージョン 2 または MVS/DFP バージョン 3 のレベルを要求するには、1 の値をコーディングし、DFSMS/MVS 以降のレベルを要求するには、2 の値をコーディングします。この値は、システムが system_level に戻す値 (下記) に影響を与えません。

system_level

サービスを要求しているシステムにインストールされている製品レベル。これは、出力引数で、4 エlement配列の整数です。配列エlementは、以下の整数から構成されます。

- バージョン番号
- リリース番号
- 修正レベル
- 特殊標識

level_indicator で 1 の値を渡した場合、DFSMS は *system_level* 値として 3、3、2、および 1 を戻します。最初の 3 つの値は、DFSMS が MVS/DFP バージョン 3 リリース 3 修正レベル 2 の機能を含んでいることを反映しています。システムは 4 番目の値を 1 に設定して、システムは実際には 3.3.2 より高いレベルであることを示します。プログラムが MVS/DFP 3.3.2 で実行されている場合には、この 4 つの値は 3、3、2、および 0 として戻されることになります。

level_indicator で 2 の値を渡した場合、IGWASYS は 4 番目の *system_level* ワードを、IGWASYS が含まれている製品名を表すコードに設定します。どのレベルのシステムも、このワードには前のリリースより小さい値は入りません。次の値のいずれかが戻されます。

- 1 「DFSMS/MVS」。MVS/ESA SP バージョン 5 では、これが MVS/ESA システムを構成していました。OS/390 MVS では、これは OS/390 バージョン 1 またはバージョン 2 の一部でした。最初の 3 つの *system_level* ワードは、DFSMS/MVS を表します。
- 2 DFSMS は OS/390 の一部であり、独立した製品ではありません。最初の 3 つの *system_level* ワードは、OS/390 バージョン 2、リリース 10、修正レベル 0 以降を示します。これらのワードは、DFSMS がサポートするように設計された OS/390 レベルを表します。DFSMS は OS/390 2.10 から変更されていないため、z/OS DFSMS バージョン 1 リリース 1 および 2 も、この値を表示します。
- >3 DFSMS は z/OS の一部であり、独立した製品ではありません。最初の 3 つの *system-level* ワードは、z/OS バージョン 1、リリース 3、修正レベル 0 以降を表します。これらのワードは、DFSMS がサポートするように設計された z/OS レベルを表します。

システムの次のリリースに新規レベルの DFSMS が含まれていない場合、IGWASYS は引き続き元の DFSMS リリースを戻します。これは、構成問題の診断やサービスに役立ちます。

その意図は、システムの将来のリリースで、プログラムが特定のレベルで実行されているのか、あるいはそれ以降のレベルで実行されているのかを判別できるようにすることです。そのようなプログラムにしたい場合は、プログラムでシステム・レベルの 4 つのワードのすべてをテストすることをお勧めします。最初に 4 番目のワードをテストして、1 より大きい値を持っていることを確認します。次に他の 3 つのワードをテストして、それが適切なバージョン、リリース、および修正レベルを指定しているかどうかを調べます。下の図の最初の列の値の 1 つが、前のリリースで戻された対応する値より小さい場合、2 番目の列の値の 1 つに、前のリリースで戻された値より大きい値が入るようにするというのが IBM の意向です。

Third word (modification level)	Fourth, first or second word
Second word (release)	Fourth or first word
First word (version)	Fourth word

system_attr

システム属性が *system_attr* 配列に戻されます。これは、出力引数です。配列エレメントの定義は、次のとおりです。

- 1 ストレージ管理サブシステムの状況。0 の値は、非アクティブを示します。1 は、アクティブを示します。
- 2-4 予約済みエレメント。0 が戻されます。
- 4 エレメント配列の整数として定義してください。

戻りコード

IGWASYS の戻りコードおよび理由コードは、277 ページの表 56 を参照してください。

例

次の例は、CALL ステートメントを使用したシステム属性呼び出しを示しています。

```

      .
      .
CALL  IGWASYS,(RC1,RS1,CODE1,LEVEL,ATTR)  Test pre-DFSMS/MVS
LTR   R15,R15          Test return code
BNZ   BADSYS
CLC   LEVEL,=F'2'      Test for MVS/XA DFP Version 2
BE    OLDSYS
MVC   SYSNAME,UNKNAME  Assume name is unknown
BL    BADSYS           Branch if unknown system
MVC   SYSNAME,=CL12'MVS/DFP'  Show we are on MVS/DFP
CLC   LEVEL+12(4),=F'1'  See if after MVS/DFP
BL    SHOWSYS         Branch if before DFSMS/MVS
CALL  IGWASYS,(RC1,RS1,CODE2,LEVEL,ATTR)
LTR   R15,R15          Branch if environment or
BNZ   BADSYS           system error
MVC   SYSNAME,UNKNAME  Assume unknown product name
CLC   LEVEL+12(4),=F'1'  Test for DFSMS/MVS code
BL    OLDSYS           Branch if unexpected code
MVC   SYSNAME,=CL12'DFSMS/MVS'  Show product name
BE    SHOWSYS         Branch if DFSMS/MVS
MVC   SYSNAME,=CL12'OS/390 DFSMS'  Set assumed new name
CLC   LEVEL+12(4),=F'3'  Branch if OS/390
BL    SHOWSYS
MVC   SYSNAME,=CL12'z/OS DFSMS  Branch if z/OS
BE    SHOWSYS
MVC   SYSNAME,=CL12'AFTER z/OS'
SHOWSYS EQU
      .
      .
RC1   DC   F'0'        Return code
RS1   DC   F'0'        Reason code
CODE1 DC   F'1'        Ask for pre-DFSMS DFP level
CODE2 DC   F'2'        Ask for level of DFSMS, or later
LEVEL DC   4F'0'       Version, release, Modification Level and code
ATTR  DC   4F'0'       SMS attributes
SYSNAME DC CL12'MVS/XA DFP'  Name of product
UNKNAME DC (L'SYSNAME)C'?'  Constant for unknown name

```

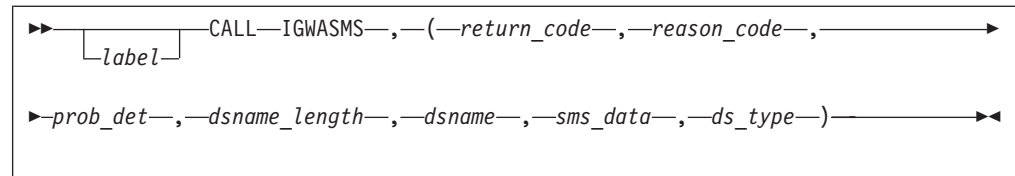
図 29. IGWASYS CALL ステートメントの例

データ・セット属性検索の呼び出し

データ・セット属性検索呼び出し (IGWASMS) は、データ・セットの関連 SMS クラスと、それが PDSE であるか、HFS データ・セットであるか、あるいはそのどちらでもないかを戻します。

フォーマット

データ・セット属性 IGWASMS 呼び出しのフォーマットは、次のとおりです。



パラメーター

return_code

IGWASMS からの戻りコード。戻りコードはレジスタ 15 にも戻されます。戻りコードについては、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』で説明しています。これは、整数として定義する必要がある、出力引数です。

reason_code

IGWASMS からの理由コード。理由コードはレジスタ 0 にも戻されます。理由コードについては、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』で説明しています。これは、整数として定義する必要がある、出力引数です。

prob_det

問題判別データ。問題判別データについての詳細は、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』を参照してください。これは、2 エレメント配列の整数として定義する必要がある、出力引数です。

dsname_length

dsname で呼び出し側によって提供されたデータ・セット名の長さ (バイト)。値は 1 から 44 の数値です。これは、整数として定義する必要がある、必須の入力引数です。

dsname

IGWASMS がサービスするデータ・セットの名前。VSAM データ・セットの場合は、クラスター名を指定する必要があります。これは、長さ *dsname_length* の EBCDIC 文字データとして定義する必要がある、必須の入力引数です。

sms_data

指定のデータ・セットに関連した SMS クラス名。左寄せされ、右側に空白を埋め込んで戻されます。配列エレメントは、以下の環境で戻されません。

- ストレージ・クラス名、またはデータ・セットが SMS データ・セットでない場合は、空白。
- 管理クラス名、またはデータ・セットが関連管理クラスを持っていない場合は、空白。
- データ・クラス名、またはデータ・セットが関連データ・クラスを持っていない場合は、空白。

これは、3 エレメント配列 (各項目は 30 (バイト) 文字 EBCDIC スtring) として定義する必要がある、出力引数です。

ds_type

データ・セット *dsname* のタイプが戻されます。1 の値は、データ・セットが PDSE タイプのデータ・セットであることを示します。2 の値は、データ・セットが HFS タイプのデータ・セットであることを示します。HFS データ・セットはファイル・システムを定義し、ファイル・システム内のファイルではありません。0 の値は、そのどちらでもないことを示します。その他の値は、現行では定義されていません。これは、整数として定義する必要がある、出力引数です。

戻りコード

IGWASMS の戻りコードおよび理由コードは、277 ページの表 56 を参照してください。

例

次の例は、LINK ステートメントを使用したデータ・セット属性呼び出しのコーディング例を示しています。

```

      .
      .
      LINK EP=IGWASMS,MF=(E,ASMSLIST)
      .
      .
RC2      DC    F'0'
RS2      DC    F'0'
      PROB1  DC    2F'0'
      DSNLEN1 DC  A(L'DSN1)
DSN1     DC    CL12'THIS.DATASET'
SMSDATA  DC    3CL30' '
DSTYPE   DC    F'0'
ASMSLIST DC    A(RC2)
          DC    A(RS2)
          DC    A(PROB1)
          DC    A(DSNLEN1)
          DC    A(DSN1)
          DC    A(SMSDATA)
          DC    A(DSTYPE)
    
```

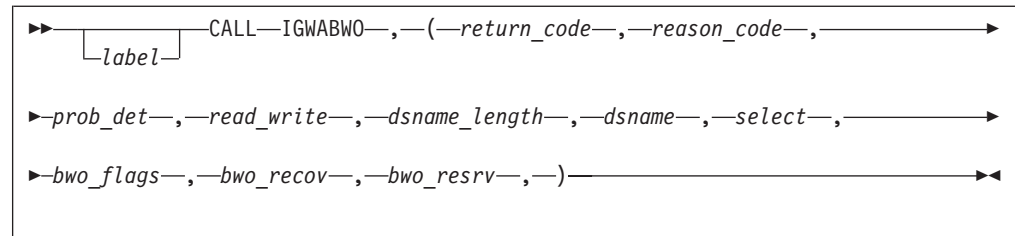
図 30. IGWASMS LINK ステートメント呼び出しの例

データ・セット Backup-While-Open サポートの呼び出し

データ・セット backup-while-open サポート呼び出し (IGWABWO) は、DFSMSdfp と通信して、更新のためにオープンされている間にデータ・セットのバックアップを取ることに関連した標識を検索または設定します。

フォーマット

IGWABWO 呼び出し可能サービスのフォーマットは、次のとおりです。



パラメーター

- return_code* IGWABWO からの戻りコード。戻りコードはレジスタ 15 にも戻されます。戻りコードについては、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』で説明しています。これは、整数として定義する必要がある、出力引数です。
- reason_code* IGWABWO からの理由コード。理由コードはレジスタ 0 にも戻されます。理由コードについては、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』で説明しています。これは、整数として定義する必要がある、出力引数です。
- prob_det* 問題判別データ。問題判別データについての詳細は、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』を参照してください。これは、2 エレメント配列の整数として定義する必要がある、出力引数です。
- read_write* このサービスの機能標識。呼び出し側が提供する 0 の値は、指定のデータ・セットの backup-while-open (BWO) データに対する READ タイプの要求であることを示します。1 の値は、WRITE タイプの要求であること、つまり、提供された引数 *bwo_flags*、*bwo_recov*、および *bwo_resrv* を使用して、指定のデータ・セットの BWO データを初期化または更新することを示します。選択引数は、処理する引数を指示します。これは、整数として定義する必要がある、必須の入力引数です。
- dsname_length* *dsname* で呼び出し側によって提供されたデータ・セット名の長さ (バイト)。値は 1 から 44 の数値です。これは、整数として定義する必要がある、必須の入力引数です。
- dsname* IGWABWO サービスが操作するデータ・セットの名前。システム管理 VSAM タイプ・データ・セットのみが、更新のためにオープンされている間にバックアップを取るのに適格です。指定された *dsname* は、VSAM データ・セットの基本クラスター名でなければなりません。これは、長さ *dsname_length* の EBCDIC 文字データとして定義する必要がある、必須の入力引数です。
- select* 以下の引数のどれを処理するのかを示します。引数は、該当する値を選択することによって指定します。これは、必須の入力引数です。
- 1 は、*bwo_flags* を処理する場合。
 - 2 は、*bwo_recov* を処理する場合。

3 は、*bwo_flags* と *bwo_recov* を処理する場合。

いくつかの引数を処理するかに関係なく、3 つのフィールド (*bwo_flags*、*bwo_recov*、および予約済み *bwo_resrv* フィールド) をすべてプログラムに定義し、呼び出しに含める必要があります。READ タイプの要求では、選択されなかった引数は値を受け取りません。WRITE タイプの要求では、選択されなかった引数の値は無視されます。

bwo_flags

この引数は、3 エLEMENT配列で、そのELEMENTは SMS データ・セットに関連した 3 つのフラグに対応しています。*bwo_flags* は、*read_write=0* タイプの要求の出力引数であり、*read_write=1* タイプの要求の必須の入力引数です。

1. 最初のELEMENTはフラグ BWO1 に関連しています。1 はオンで、0 はオフです。
2. このELEMENTはフラグ BWO2 に対応しています。1 はオンで、0 はオフです。
3. このELEMENTはフラグ BWO3 に対応しています。1 はオンで、0 はオフです。

3 ELEMENT配列の整数として定義してください。

bwo_recov

この引数は、BWO に適格なデータ・セットに関連したリカバリー・タイム・スタンプを含んでいる 8 バイトストレージ域です。*bwo_recov* は、*read_write=0* タイプの要求の出力引数であり、*read_write=1* タイプの要求の必須の入力引数です。CICS VSAM データ・セットのタイム・スタンプのフォーマットは、次のとおりです。

- 最初のワードには、パック 10 進数フォーマット 0CYDDDDF の日付が入ります。ここで、
 - 0C** 世紀 - 00 は 19YY を表し、01 は 20YY を表します。
 - YY** 年の最後の 2 桁
 - DDD** 年間通算日 (ユリウス日付)
 - F** 符号 (F は正数)
- 2 番目のワードには、パック 10 進数フォーマット HHMSSTF の時刻が入ります。ここで、
 - HH** 時、24 時間クロックに基づきます。
 - MM** 分
 - SS** 秒
 - T** 10 分の 1 秒
 - F** 符号 (F は正数)

bwo_resrv

この引数は、将来の利用のために予約されています。*bwo_resrv* 引数は書き込みや読み取りはできませんが、プログラムに定義し、呼び出しに含める必要があります。16 バイトの長さの EBCDIC 文字データとして定義してください。

戻りコード

IGWABWO の戻りコードおよび理由コードは、277 ページの表 56 を参照してください。

例

次の例は、LOAD および CALL ステートメントを使用した、WRITE および READ タイプの backup-while-open 呼び出しのコーディング例です。

```

      .
      .
      LOAD EP=IGWABWO
      LR R2,R0
      CALL (R2),(RC1,RS1,PROB1,RW1,DSNLEN1,DSN1,SEL1,BWOF1,BWOR1,BWRE)
      CALL (R2),(RC2,RS2,PROB2,RW2,DSNLEN2,DSN2,SEL2,BWOF2,BWOR2,BWRE)
      .
      .
* ARGUMENTS FOR WRITING
      RC1      DC      F'0'
      RS1      DC      F'0'
      PROB1    DC      2F'0'
      RW1      DC      F'1'          WRITE
      DSNLEN1  DC      F'11'
      DSN1     DC      CL44'THAT.VSAM01'
      SEL1     DC      F'3'          WRITE BWO_FLAGS AND BWO_RECOV
      BWOF1    DC      F'0'
      DC       DC      F'1'
      DC       DC      F'0'
      BWOR1    DS      0F
      DC       DC      X'0096137F'   INPUT DATE IN 0CYDDDF FORMAT
      DC       DC      X'1045301F'   INPUT TIME IN HHMSSTF FORMAT
      BWRE1    DC      CL16' '
* ARGUMENTS FOR READING
      RC2      DC      F'0'
      RS2      DC      F'0'
      PROB2    DC      2F'0'
      RW2      DC      F'0'          READ
      DSNLEN2  DC      F'11'
      DSN2     DC      CL44'THAT.VSAM01'
      SEL2     DC      F'3'          READ BWO_FLAGS AND BWO_RECOV
      BWOF2    DC      3F'0'
      BWOR2    DC      CL8' '
      BWRE2    DC      CL16' '
*
```

図 31. LOAD および CALL ステートメントを使用した IGWABWO の例

Backup-While-Open 機能の使用

以下では、backup-while-open (BWO) 機能の使用法について説明します。268 ページの『データ・セット Backup-While-Open サポートの呼び出し』で説明した IGWABWO サービスを使用して、BWO フラグおよび BWO リカバリー・フィールドを検索または更新できます。BWO 標識については、274 ページの表 55 で説明します。

高可用性を必要とする環境では、アプリケーション・データ・セットの整合したバックアップ・コピーを作成するために、アプリケーションを停止または静止するのは不可能であったり、望ましくない場合があります。

呼び出し可能サービス

こうした環境のために、DFSMSdfp は、出力用にオープンされた SMS 管理 VSAM データ・セットのバックアップを取ることを許可するサポートを提供しています。このサポートは、復元されたデータベースを整合点までリカバリーできるアプリケーション (データベース・システムなど) にのみ役立ちます。これは通常、アプリケーションによって保守されるログから行われ (順方向リカバリー)、このログは変更 (追加、削除、または更新) されたすべてのレコードのレコード・イメージを含んでいます。これらのイメージをデータベースのバックアップ・コピーに再適用し、特定の時点のデータベースの状況を論理的に再作成します。

BWO によって提供されるサポートは、バックアップの進行中にデータ・セットに対する出力または更新活動が行われないようにするためにデータベースのデータ・セットを静止できるオンライン・アプリケーションには必要がない可能性があります。ここでのデータ・セットの静止とは、データ・セットをクローズして割り振り解除することを意味します。

このサポートの操作の説明では、以下の用語を使用します。

- データベース・マネージャー: 処理するデータ・セットへのアクセスを制御するアプリケーション。BWO サポートを有効にするには、データベース・マネージャーは、データベースのポイント・イン・タイム (特定時点) 再構成を可能にする何らかのロギング機能を備えている必要があります。
- バックアップ・マネージャー: バックアップおよび復元機能を実行するアプリケーションまたは製品 (DFSMSHsm、DFSMSdss など)。
- リカバリー・マネージャー: リカバリー・ログのインベントリを管理し、復元されたデータ・セットに適切なログを適用するコンポーネント。

以下では、BWO サポートと BWO のユーザーの関係について説明します。説明に出てくる BWO フラグのさまざまな状態については、274 ページの表 55 を参照してください。BWO 標識はカタログに保存されています。BWO フラグの状態は、DFSMSdfp 呼び出し可能システム・サービスの IGWABWO を使用して設定およびリセットされます。

- 初期割り振りでは (IDCAMS DEFINE、IDCAMS または TSO ALLOCATE、JCL および動的割り振り)、データ・セットの BWO は使用可能にされません (デフォルトでは、BWO フラグ状態は 000 になります)。
- データベース・マネージャーは、データ・セットをオープンする前に BWO フラグを検査して、それがダウン・レベル (BWO フラグの状態が 101 または 001) でないことを確認する必要があります。データ・セットがダウン・レベルである場合、リカバリー・マネージャーを使用して、ログ変更をデータ・セットに適用する必要があります。
- データベース・マネージャーは、出力用にオープンされている間にバックアップを取ることが許されている各データ・セットのフラグ BWO1 (BWO フラグ状態 100) をオンに設定する必要があります。これは、アプリケーションによってアクセスされているかどうかに関係なく、バックアップ・マネージャーがデータ・セットを逐次化せずにバックアップを開始することを許可します。
- バックアップ・マネージャーは、バックアップを開始する前に、BWO フラグを検索する必要があります。BWO1 がオンの場合、逐次化せずにバックアップを取ることができます。そうでない場合、バックアップ・マネージャーは通常のデータ・セット逐次化を行う必要があります。

バックアップが完了した後、バックアップ・マネージャーは再度 BWO フラグを検索することが必要です。BWO フラグの状態が変更された場合は、バックアップ中のいずれかの時点で有効なバックアップ作成を妨げるアクションが発生しました。バックアップ・マネージャーは、いま作成されたバックアップを廃棄することが必要です。

- データ・セットをリカバリーする必要がある場合、最初にバックアップ・マネージャーを使用して復元します。復元の期間中、データ・セットは逐次化され、アプリケーションがそれにアクセスするのを防止します。バックアップ・マネージャーは、復元が完了した時点で BWO フラグを設定して、逐次化してまたは逐次化せずに作成されたバックアップ・コピーを使用して復元が行われたのかどうかを示す必要があります。
 - 逐次化せずにバックアップが取られた場合、BWO フラグを 101 に設定する必要があります。
 - 逐次化してバックアップが取られた場合、BWO フラグを 000 に設定する必要があります。

いずれの場合も、アプリケーション管理者が、リカバリー・ログを適用するかどうかを決めます。

リカバリー・マネージャーが処理を完了するまで、データベース・マネージャーはデータ・セットへのアクセスを許可してはなりません。

- リカバリー・マネージャーは、データ・セットをオープンする前に BWO フラグを 001 に変更し、ログを適用し、フラグを 000 に設定して、データ・セットが処理されて整合した状態になったことを示す必要があります。その後、データベース・マネージャーはデータ・セットへの通常のアクセスを再確立できます (これにより、データ・セットをオープンして使用できるようになります)。
- BWO が使用可能にされているデータ・セットの場合、ある種の条件下では、システムが BWO フラグを 010 に設定して、逐次化せずにバックアップ・コピーを開始することを防止することがあります (例えば、CI/CA 分割中)。これは、逐次化せずにバックアップを開始してはならないこと (BWO1 オフ)、および現在進行中のバックアップは無効と見なす必要があることを示します。バックアップの開始を防止していた条件が終了すると、システムは BWO フラグを 110 にリセットします。これは、逐次化せずにバックアップを開始できるようになったこと、および進行中のバックアップは廃棄する必要があることを示します。
- データベースのデータ・セットが、順方向リカバリー・ログを作成しないバッチ・プログラムによってアクセスされる場合 (データベース・マネージャーがデータ・セットにアクセスしていないときに)、データベース・マネージャーはクローズ時 (つまり、OX1) に、BWO1 フラグをクリアして BWO3 フラグを設定する必要があります。BWO2 フラグの設定は変更してはなりません。バックアップ・マネージャーが逐次化しないバックアップの終了時にこの BWO 状態を検出した場合、バックアップは有効ではなく、廃棄する必要があります。BWO フラグ状態が 011 の場合、バックアップ・マネージャーは逐次化してバックアップを開始できますが、フラグは 000 にリセットする必要があります。

注:

1. バックアップは大量の入出力活動を生じる場合があるので、アプリケーションの応答時間への影響を避けるために、データ・セットに対する活動が最も少ない時間帯にバックアップを取ることが必要です。

呼び出し可能サービス

2. BWO フラグは、IGWABWO を使用してそれを読み取り、設定する間、ロックされません。フラグの設定が変更される場合、逐次化を提供するのはアプリケーションの責任です。
3. KSDS データ・セットは特別な考慮事項が必要です。挿入および更新は、制御インターバル (CI) および制御域 (CA) の分割を伴う場合があります。CI および CA の分割中に逐次化せずに取られたバックアップは、バックアップ・コピー内のレコードの脱落や重複を防ぐために、IBM 製品によって廃棄されます。

CI および CA 分割の頻度は、挿入活動、レコードの長さを増やす更新活動、および CI と CA 内のフリー・スペースの量によって異なります。BWO が使用可能にされた KSDS データ・セットの場合、挿入や更新が少ない期間にバックアップを実行するか、または制御インターバルと制御域に適切なフリー・スペースがあることを確認してください。

4. 逐次化せずにバックアップを取ることができるのは、次の場合です (共用オプション 1 または 2 に定義されたデータ・セット上)。
 - データベースのアップグレード・セット内に代替索引が含まれている場合。
 - データベースがパス名によって、または *ddname* または *dsname* 共用を使用してアクセスされる場合。
5. データ・セットは、BWO フラグの設定に関係なくオープンできます。データ・セットの内容が整合しているかどうかを判別するのはデータベース・マネージャーの責任です。
6. データベース・マネージャーは、BWO リカバリー・フィールドを使用して、リカバリー・マネージャーが適切なリカバリー・ログを見つけるために使用する情報 (ログ RBA やログ・タイム・スタンプなど) を保管できます。
7. BWO 概念は、論理データ・セット・バックアップ/復元にも適用されます。物理データ・セット・バックアップ/復元または全ボリューム・ダンプ/復元には適用されません。

表 55 は、それぞれの BWO (BWO1、BWO2、BWO3) 標識の意味を示しています。

表 55. Backup-While-Open 標識

BWO 設定	説明
000	データ・セットは、逐次化しないバックアップをサポートしません。
001	順方向リカバリーが進行中です。リカバリー後に 000 にリセットしてください。
010	BWO データ・セットの CI/CA 分割が進行中です。バックアップを開始しないでください。バックアップが進行中の場合は、終了時に廃棄してください。データベース・マネージャーが分割中に異常終了した場合、オープン時にこの状態が存在する可能性があります。データベースのアクションは、データベース・マネージャーによって異なります。データ・セットを復元して、順方向リカバリーするか、あるいは AIX が存在する場合、変更をバックアウトすることが必要になります。
011	データベース・マネージャーは BWO データ・セットをクローズし、前にそれをオープンしたときに CI/CA 分割が発生していました。バックアップ・マネージャーはこれを 000 にリセットし、バックアップのために逐次化する必要があります (BWO を適用せずに)。データベース・マネージャーは、オープン時にこの状態を 110 に変更する必要があります。

表 55. Backup-While-Open 標識 (続き)

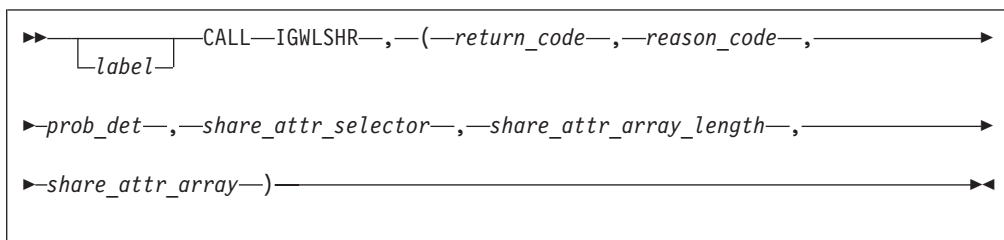
BWO 設定	説明
100	BWO データ・セットがデータベース・マネージャーによってオープンされました。逐次化せずにバックアップを取ってください。
101	データ・セットが復元され、それを使用する前に順方向リカバリーが必要です。順方向リカバリーの前に、これを 001 にリセットしてください。
110	BWO データ・セット上で CI/CA 分割が発生し、完了しました。データベース・マネージャーが異常終了した場合、オープン時にこの状態が存在する可能性があります。逐次化せずにバックアップを取ってください。バックアップの前に、これを 100 にリセットしてください。
111	無効な状態。

DFSMSdfp 共用属性の呼び出し

DFSMSdfp 共用属性呼び出し (IGWLSHR) は、現在使用中の DFSMSdfp 共用属性を戻します。

フォーマット

IGWLSHR 呼び出し可能サービスのフォーマットは、次のとおりです。



パラメーター

return_code

IGWLSHR からの戻りコード。戻りコードはレジスター 15 にも戻されます。戻りコードについては、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』で説明しています。これは、整数として定義する必要がある、出力引数です。

reason_code

IGWLSHR からの理由コード。理由コードはレジスター 0 にも戻されます。理由コードについては、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』で説明しています。これは、整数として定義する必要がある、出力引数です。

prob_det

問題判別データ。問題判別データについての詳細は、277 ページの『IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード』を参照してください。これは、2 エレメント配列の整数として定義する必要がある、出力引数です。

share_attr_selector

要求する DFSMSdfp 共用属性を指定するために使用します。PDSE 共用プロトコル属性を要求する場合は、1 の値をコーディングします。これは、整数として定義する必要がある、必須の入力引数です。

share_attr_array_length

共用属性配列のサイズを指定するために使用します。これは、share_attr_array 内の配列エレメント数を表す長さです。選択された share_attr_selector 値に必要な値をコーディングしてください。share_attr_selector が 1 の場合、share_attr_array_length は最小で 1 でなければなりません。これは、整数として定義する必要がある、必須の入力引数です。

share_attr_array

DFSMSdfp 共用属性を戻します。これは、出力引数です。長さ share_attr_array_length の整数の配列として定義してください。ここで、

share_attr_array_length は、選択された *share_attr_selector* 値に必要な長さです。*share_attr_selector* 値の配列エレメントは、次のように戻されます。

- 1 PDSE 共用プロトコルの状況。0 の値は、PDSE サポートが利用不可であることを示します。つまり、システムは呼び出しをサポートしますが、SMS PDSE サポートがアクティブではありません。システムがこの呼び出しをサポートしない場合には (前のリリースを使用している場合と同様に)、戻りコード 36 が戻されます。

1 の値は、通常の PDSE 共用プロトコルが使用されていることを示します。2 の値は、拡張 PDSE 共用プロトコルが使用されていることを示します。

- 2-4 予約済みエレメント。0 が戻されます。

戻りコード

IGWLSHR の戻りコードおよび理由コードは、表 56 を参照してください。

IGWASYS、IGWASMS、IGWABWO、IGWLSHR 戻りコードと理由コード

IGWASYS、IGWASMS、IGWABWO、または IGWLSHR は、呼び出し側プログラムに制御を戻すときに、戻りコードと理由コードを提供します。IGWASMS と IGWABWO は、問題判別に役立つ追加データを *prob_det* 配列に戻すことができます。IGWLSHR は、*share_attr_selector* および *share_attr_array_length* 引数に関する追加データを戻すことができます。下の表は、戻りコードと理由コードの組み合わせを識別し、それぞれの意味を説明し、追加の問題判別データが戻される状況とその内容を説明し、取る必要がある処置を推奨しています。

表 56. IGWASYS、IGWASMS、IGWABWO、IGWLSHR の戻りコードと理由コード

戻りコード、 10 進数 (16 進数)	理由コード 10 進数 (16 進数)	説明
0 (0)	0 (0)	操作は正常に終了しました。
4 (4)	4 (4)	操作は正常に行われましたが、 <i>bwo_recov</i> 引数が <i>dsname</i> で指定されたデータ・セットに対して有効な値ではありませんでした。その理由は、これが DFP 3.1.0 のもとで作成されたために、 <i>bwo_recov</i> がデータ・セットに追加されなかったことにあります。必要に応じて、 <i>bwo_recov</i> をデータ・セットに追加してください。
8 (8)	4 (4)	無効な <i>dsname_length</i> または <i>share_attr_selector</i> が指定されました。引数を訂正して、要求を再試行してください。
8 (8)	8 (8)	無効なブランクの <i>dsname</i> または無効な <i>share_attr_array_length</i> が指定されました。引数を訂正して、要求を再試行してください。
8 (8)	12 (C)	無効な <i>read_write</i> が指定されました。値 0 または 1 を指定する必要があります。 <i>read_write</i> 引数を訂正して、要求を再試行してください。
8 (8)	16 (10)	<i>bwo_flags</i> に提供された値が無効です。BWO1、BWO2、および BWO3 は、0 または 1 の値を持つ必要があります。 <i>bwo_flags</i> 引数を訂正して、要求を再試行してください。
8 (8)	20 (14)	BWO は、VSAM タイプのデータ・セットに対してのみサポートされます。指定された名前は VSAM クラスター名ではありませんでした。 <i>dsname</i> 引数に VSAM クラスター名を指定して、要求を再試行してください。

呼び出し可能サービス

表 56. IGWASYS、IGWASMS、IGWABWO、IGWLSHR の戻りコードと理由コード (続き)

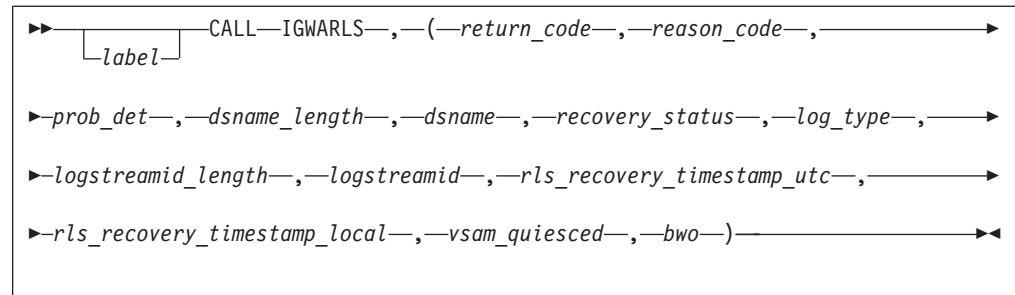
戻りコード、 10 進数 (16 進数)	理由コード 10 進数 (16 進数)	説明
8 (8)	24 (18)	無効な <i>level_indicator</i> が指定されました。 <i>level_indicator</i> 引数を訂正して、要求を再試行してください。
8 (8)	28 (1C)	無効な <i>select</i> 引数が指定されました。1 から 3 の間の値を指定する必要があります。 <i>select</i> 引数を訂正して、要求を再試行してください。
8 (8)	32 (20)	<i>dsname</i> に指定されたデータ・セットは、SMS 管理データ・セットではありません。 <i>dsname</i> 引数を訂正して、要求を再試行してください。
12 (C)	8 (8)	要求を処理するための仮想ストレージが不十分です。一部の仮想ストレージを解放して、要求を再試行してください。状態が続く場合は、IBM に連絡してプログラミングの支援を依頼してください。
12 (C)	12 (C)	<i>dsname</i> に指定されたデータ・セットが見つかりませんでした。データ・セットが存在し、 <i>dsname</i> に正しく指定したことを確認してください。
16 (10)	4 (4)	カタログ管理への呼び出しでエラーが発生しました。カタログ戻りコードが <i>prob_det</i> の最初のエレメントに入り、カタログ理由コードが <i>prob_det</i> の 2 番目のエレメントに入ります。カタログ戻りコードと理由コードの説明は、メッセージ IDC3009I を参照してください。カタログ管理戻りコード 8 は、指定されたデータ・セットが見つからなかったことを示します。この戻りコードを受け取った場合は、 <i>dsname</i> を訂正して、要求を再試行してください。
20 (14)	4 (4)	IGWASYS/SMS/BWO または IGWLSHR の処理中にシステム・エラーが発生しました。 <i>prob_det</i> のエレメントに追加の診断データが入っています。IBM に連絡してプログラミングの支援を依頼し、IGWASYS/SMS/BWO <i>return_code</i> 、 <i>reason_code</i> 、および <i>prob_det</i> の値を提供してください。
36 (24)	4 (4)	IGWLSHR サービス・モジュールまたは IGWASYS/SMS/BWO サービス・モジュールの IGWAMCS1 と IGWAMCS2 へのリンケージを確立できません。間違ったレベルのオペレーティング・システムが使用されているか、または呼び出し可能システム・サービス・ライブラリー SYS1.CSSLIB に、必要なサービスが欠落しているかのいずれかです。システム・プログラマーに連絡して支援を依頼してください。

レコード・レベル共用照会の呼び出し

IGWARLS は、カタログ内でレコード・レベル共用に関する情報を照会するのに使用します。

フォーマット

IGWARLS 呼び出しステートメントのフォーマットは、次のとおりです。



パラメーター

return_code

IGWARLS からの戻りコード。戻りコードはレジスター 15 にも戻されます。戻りコードについては、280 ページの『戻りコード』で説明しています。これは、出力引数です。*return_code* は整数として定義してください。

reason_code

IGWARLS からの理由コード。理由コードはレジスター 0 にも戻されます。理由コードについては、280 ページの『戻りコード』で説明しています。これは、出力引数です。*reason_code* は整数として定義してください。

prob_det

問題判別データ。問題判別データについての詳細は、280 ページの『戻りコード』を参照してください。これは、出力引数です。*prob_det* は 2 エレメント配列の整数として定義してください。

dsname_length

dsname で呼び出し側によって提供されたデータ・セット名の長さ (バイト)。値は 1 から 44 の数値です。これは、必須の入力引数です。*dsname_length* は整数として定義してください。

dsname

IGWARLS サービスが操作する基本クラスターの名前。これは、必須の入力引数です。*dsname* は、長さ *dsname_length* の EBCDIC 文字データとして定義してください。

recovery_status

スフィアは順方向リカバリーが必要としてマークされているかどうかに関する標識を戻します。

- 0 - VSAM スフィアに必要な RLS リカバリーが保留中ではありません。
- 1 - 必要な RLS リカバリーが保留中です。

recovery_status は、整数として定義される出力引数です。

log_type

DEFINE CLUSTER の LOG= パラメーターの指定が戻されます。

- 1 - LOG パラメーターは未定義
- 2 - LOG=NONE
- 3 - LOG=UNDO
- 4 - LOG=ALL

log_type は、整数として定義される出力引数です。

呼び出し可能サービス

logstreamid_length

LOGSTREAMID フィールドの長さ (バイト)。これは、必須の入力パラメーターです。 *logstreamid_length* は整数として定義してください。
logstreamid_length の値は、少なくとも 26 バイトの長さでなければなりません。

logstreamid

DEFINE CLUSTER の LOGSTREAMID の指定が戻されます。パラメーターが未定義の場合、ブランクが戻されます。呼び出し側は、戻された LOGSTREAMID フィールドを右から左にスキャンして非ブランク文字を見つけることによって、またはフィールド全体をスキャンすることによって、そのサイズを判別できます。 *logstreamid* は、*logstreamid_length* の長さの EBCDIC 文字データ・フィールドとして定義される出力引数です。

rls_recovery_timestamp_utc

STCK フォーマットのダンプ/コピーの UTC 時間 (以前は GMT と呼ばれていた) を表す出力引数。このフィールドは、8 バイト符号なし整数として定義されます。プログラムはこのフィールドを整数または文字ストリングと見なします。

rls_recovery_timestamp_local

STCK フォーマットのダンプ/コピーの地方時を表す出力引数。このフィールドは、8 バイト符号なし整数として定義されます。プログラムはこのフィールドを整数または文字ストリングと見なします。

vsam_quiesced

スフィアが VSAM_QUIESCED としてマークされているかどうかを示す出力引数。

- 0 - スフィアは VSAM_QUIESCED としてマークされていません。
- 1 - スフィア VSAM_QUIESCED としてマークされています。

vsam_quiesced は、整数として定義される出力引数です。

bwo

クラスター定義の BWO パラメーターの値を示す出力引数。

- 1 - BWO パラメーターは未定義。
- 2 - BWO = TYPECICS 処理が許可される。
- 3 - BWO = NO。BWO 処理は許可されない。
- 4 - BWO = TYPEIMS 処理が許可される。
- 5 - BWO = TYPEOTHER 処理が許可される。

戻りコード

IGWARLS は、呼び出し側プログラムに制御を戻すときに、戻りコードと理由コードを提供します。IGWARLS は、問題判別に役立つ追加データを *prob_det* 配列に戻すことができます。表 57 は、戻りコードと理由コードの組み合わせを識別し、それぞれの意味を説明し、追加の問題判別データが戻される状況とその内容を説明し、取る必要がある処置を推奨しています。

表 57. IGWARLS の戻りコードと理由コード

戻りコード、 10 進数 (16 進数)	理由コード、 10 進数 (16 進数)	説明
0 (0)	0 (0)	操作は正常に終了しました。

表 57. IGWARLS の戻りコードと理由コード (続き)

戻りコード、 10 進数 (16 進数)	理由コード、 10 進数 (16 進数)	説明
8 (8)	4 (4)	無効な <i>dsname_length</i> が指定されました。 <i>dsname_length</i> 引数を訂正して、要求を再試行してください。
8 (8)	8 (8)	ブランクの無効な <i>dsname</i> が指定されました。 <i>dsname</i> 引数を訂正して、要求を再試行してください。
8 (8)	20 (14)	IGWARLS は、VSAM データ・セットに対してのみサポートされます。指定された名前は、基本クラスターの名前ではありませんでした。 <i>dsname</i> 引数に基本クラスター名を指定して、要求を再試行してください。
8 (8)	32 (20)	<i>dsname</i> に指定されたデータ・セットは、SMS 管理データ・セットではありません。 <i>dsname</i> 引数を訂正して、要求を再試行してください。
8 (8)	40 (28)	IGWARLS の場合、指定された <i>logstreamid_length</i> が無効であったか (≤ 0)、または要求された <i>logstreamid</i> を戻すために十分な大きさではありませんでした。 <i>logstreamid_length</i> 引数を訂正して、要求を再試行してください。
12 (C)	8 (8)	要求を処理するための仮想ストレージが不十分です。要求を再試行してください。状態が続く場合は、IBM に連絡してプログラミングの支援を依頼してください。
12 (C)	12 (C)	<i>dsname</i> に指定されたデータ・セットが見つかりませんでした。データ・セットが存在し、 <i>dsname</i> に正しく指定したことを確認してください。
12 (C)	14 (E)	<i>dsname</i> に指定されたデータはカタログ内で見つかりましたが、その属性が利用不可です。データ・セットを <i>dsname</i> に正しく指定したことを確認してください。
12 (C)	16 (10)	<i>dsname</i> に指定されたデータ・セットにアクセスできません。データ・セットは HSM にマイグレーション済みです。データ・セットを HRECALL して、要求を再試行してください。
16 (10)	4 (4)	カタログ管理への呼び出しでエラーが発生しました。カタログ戻りコードが <i>prob_det</i> の最初のエレメントに入り、カタログ理由コードが <i>prob_det</i> の 2 番目のエレメントに入ります。カタログ戻りコードと理由コードの説明は、メッセージ IDC3009I を参照してください。カタログ管理戻りコード 8 は、指定されたデータ・セットが見つからなかったことを示します。この戻りコードを受け取った場合は、 <i>dsname</i> を訂正して、要求を再試行してください。
20 (14)	4 (4)	IGWARLS 処理中にシステム・エラーが発生しました。 <i>prob_det</i> のエレメントに追加の診断データが入っています。IBM に連絡してプログラミングの支援を依頼し、IGWARLS <i>return_code</i> 、 <i>reason_code</i> 、および <i>prob_det</i> の値を提供してください。
36 (24)	4 (4)	IGWRLS サービス・モジュール IGWAMCS4 へのリンケージを確立できません。間違ったレベルのオペレーティング・システムが使用されているか、または呼び出し可能システム・サービス・ライブラリー SYS1.CSSLIB に、必要なサービスが欠落しているかのいずれかです。システム・プログラマーに連絡して支援を依頼してください。

例

以下の例は、LOAD および CALL ステートメントを使用した RLS 照会呼び出しを示しています。

```

      .
      .
      LOAD EP=IGWARLS
      LR   R9,R0
      CALL (R9),(RC1,RS1,PROB1,DSNLEN1,DSN1,RECSTAT,LOGTYPE,      X
            LOGSTRML,LOGSTRM,RCVTMG,RCVTML,VSAMQUIS,BWO)

      RC1   DC   F'0'
      RS1   DC   F'0'
      PROB1 DC   2F'0'
      DSNLEN1 DC A(L'DSN1)
      DSN1   DC   CL12'BASE.CLUSTER'
      RECSTAT DC   F'0'
      LOGTYPE DC   F'0'
      LOGSTRML DC A(L'LOGSTRM)
      LOGSTRM DC   CL26' '
      RCVTMG DC   XL8'00'
      RCVTML  DC   XL8'00'
      VSAMQUIS DC   F'0'
      BWO     DC   F'0'
    
```

図 32. LOAD および CALL ステートメントを使用した IGWARLS 照会呼び出しの例

第 10 章 DESERV 出口の使用

DESERV 機能についての詳しい説明は、「z/OS DFSMS データ・セットの使用法」を参照してください。そのマクロについては、「z/OS DFSMS Macro Instructions for Data Sets」を参照してください。

BLDL および STOW 機能の置き換えとして、インターフェース DESERV (GET、PUT、DELETE、RENAME、および UPDATE) が導入されたのに伴い、先進的アプリケーションおよびベンダー製品によって使用される 2 つの機能が使用不可になりました。これらの機能は、SVC スクリーニングおよび SVCUPDTE と呼ばれます (BLDL および STOW マクロに適用されるため)。DESERV 出口を SVC スクリーニングおよび SVCUPDTE 機能と併用すると、PDS および PDSE ディレクトリーへのアクセスをモニターしたり、更新したりできます。以下の状態が生じると、システムは DESERV GET 機能呼び出します。

- バインダーを使用してプログラム・オブジェクトまたはロード・モジュールをバインドしており、含めるメンバー名を検索している場合。(リンケージ・エディターは DESERV を使用しないので注意してください。)
- システムが ATTACH、LINK、LOAD、または XCTL 機能の処理中に、ストレージにロードするモジュールを検索している場合。

このような状態では、システムは BLDL を発行する代わりに DESERV GET を使用します。ただし、状態によっては、DESERV GET が BLDL を発行してディレクトリー検索を行うことがあります (BLDL 機能は DESERV 呼び出しを発行しません)。以下の状態が生じると、システムは DESERV PUT 機能呼び出します。

- バインダーが PDSE にプログラム・オブジェクトを作成する場合 (リンケージ・エディターは DESERV を使用せず、バインダーも PDS にロード・モジュールを作成する際に DESERV PUT を使用しないことに注意してください)。
- IEBCOPY ジョブが IEBCOPY アンロード・データ・セットからプログラム・オブジェクトをロードする場合。
- IEBCOPY ジョブが PDSE にメンバーをコピーし、メンバー名の 1 つが 63 バイトの長さを超えている場合。

このような状態では、システムは STOW を発行する代わりに DESERV PUT を使用します。DESERV PUT と STOW コードは対話しません。STOW は DESERV PUT を発行せず、DESERV PUT も STOW を発行しません。

現行では、システムは RENAME または UPDATE 機能を使用しません。SMP/E が DELETE 機能の唯一の既知のユーザーです。名前変更、更新、および削除機能と STOW コードは対話しません。

SVC スクリーニングを使用して、SVC スクリーン・テーブルがタスク制御ブロック (TCB) に関連付けられます。テーブルは、特定の SVC 番号を無効としてマークを付けます。また、テーブルは、無効な SVC が発行された場合に制御を取得するルーチンのアドレスも定義します。これにより、スクリーン・ルーチンは、機能を禁止したり、自身で機能を実行したり、あるいは SVC スクリーニングを一時的に

使用不可にして SVC を再発行したりすることが可能になります。この手法は、SVC ルーチンに対してフロントエンドおよびバックエンド・メカニズムを提供しません。

SVCUPDTE を使用して、アプリケーションはシステムの SVC テーブル項目を動的に置換または削除したり、指定のエントリー・ポイントのルーチンの SVC 番号を入手したりできます。SVCUPDTE の置換機能の具体的な使用例の 1 つとして、以下のシナリオを使用して IBM 提供の SVC ルーチンを置き換えます。

1. SVC テーブルから SVC 18 (BLDL) の SVC 項目を抽出する。
2. SVCUPDTE を発行して、ベンダーのバージョンの BLDL 機能をインストールする。
3. SVC 18 が発行されると、ベンダーの BLDL モジュールが制御を取得する。
4. ベンダーの BLDL は、機能を実行して呼び出し側に制御を戻すか、または分岐して、前に SVC テーブルからアドレスを取得した IBM 提供の BLDL コードに入る。

DESERV は、SVC ルーチン BLDL および STOW の SVC スクリーニングと同様のタスク・レベル出口をインターフェースに提供します。DESERV は、SVCUPDTE 置換オプションと同様のグローバル出口もインターフェースに提供します。SVC スクリーニングおよび SVCUPDTE の使用について詳しくは、「z/OS MVS Programming: Authorized Assembler Services Reference SET-WTO」および「z/OS MVS Programming: Authorized Assembler Services Guide」を参照してください。

タスク出口とグローバル出口の説明

タスク・レベル出口

LPA の初期化後に、任意の TCB に対してタスク・レベル DESERV 出口を確立できます。タスクに対して確立された後、DESERV 出口には、TCB 下で発行される DESERV 機能のために制御が与えられます。GET、PUT、RENAME、DELETE、または UPDATE 呼び出しごとに 2 回、出口エントリー・ポイントに制御が与えられます。出口に渡されるパラメータは、この呼び出しが DESERV 機能の実行前に行われるのか、実行後に行われるのかを示します。

連結内に PDSE がないときに DESERV FUNC=GET 呼び出しが行われた場合、DESERV コードは BLDL を発行して、要求された名前を検索します。この場合、タスク・レベル出口が呼び出されます。ただし、SVC スクリーニングもアクティブの場合、出口は次のいずれかのタスクを実行します。

- この DESERV 呼び出しを処理しない (つまり、BLDL 要求の処理は SVC スクリーン・ルーチンに任せる)。
- 前処理および後処理出口を使用してこの DESERV 呼び出しを処理するが、同時に、前処理出口で BLDL に対する SVC スクリーニングを使用不可にし、後処理出口で BLDL スクリーニングを使用可能にする。出口に渡されるパラメータは、BLDL が発行前であるのか、発行済みであるのかを示します。

SVC スクリーニング機能と同様に、DESERV タスク・レベル出口機能では、出口を後続の接続タスクに伝搬する必要があるかどうかをユーザーが指示できます。

タスクの最初の DESERV 呼び出しは、TCB チェーンを検索して、伝搬が指定された DESERV タスク・レベル出口ルーチンを見つけます。DESERV は、発信 TCB ポインター (TCBOTC) の後の TCB チェーンを検索します。存在しない場合、タスクには DESERV タスク・レベル出口が存在しないことを示すマークが付けられます。そのため、伝搬オプションを機能させるには、ATTACH マクロを発行する前に、出口ルーチンを確立しておく必要があります。これは、SVC スクリーニングをインプリメントするのとはほぼ同じです。相違点は、SVC スクリーニングは ATTACH の時点で伝搬するのに対して、DESERV 出口は ATTACH 時には伝搬しない場合があることです (例えば、接続プログラムがジョブ・パック・キュー内で見つかった場合、モジュールを見つけるためのディレクトリー検索 (つまり、DESERV GET) は行われません)。

グローバル出口

LPA の初期化後に、システムに対してグローバル DESERV 出口を確立できます。グローバル出口を確立した後、共通ストレージ内に DST (DESERV スクリーン・テーブル) ストレージを取得します。DST は、DESERV 出口を識別するのに使用されます。確立後、DESERV 出口は DESERV GET、PUT、RENAME、DELETE、または UPDATE 機能が呼び出されるたびに制御を受け取ります。出口エントリー・ポイントには、呼び出しごとに 2 回制御が与えられます。DESERV が実行される前に 1 回と、DESERV 機能から戻った直後に 1 回です。出口への呼び出しは、この呼び出しが DESERV 実行の前であるか後であることを示します。グローバル出口ルーチンは、共通のアドレス可能ストレージに常駐する必要があります。

連結内に PDSE がないときに DESERV FUNC=GET 呼び出しが行われた場合、DESERV コードは BLDL を発行して、要求された名前を検索します。この場合、グローバル出口が呼び出されます。SVCUPDTE 機能とグローバル出口の両方を使用する場合は、グローバル DESERV 出口をインプリメントする前に、DESERV グローバル出口と、SVC が発行されたときに制御が渡されるルーチンとの相互作用を考慮してください。

タスク・レベル出口とグローバル出口の相互作用

タスク・レベル DESERV 出口とグローバル DESERV 出口が両方とも定義されている場合、規定の呼び出しシーケンスがあります。タスク・レベル出口が最初に呼び出されます。タスク・レベル出口が、DESERV 機能を終了する必要があることを示している場合 (出口からの戻りコード 4 によって)、DESERV は即時に呼び出し側に戻ります。ただし、タスク・レベル出口が戻りコード 0 で戻った場合は、グローバル出口に制御が渡されます。グローバル出口は、即時に制御を DESERV の呼び出し側に戻す必要があることを示すことができます (戻りコード 4 によって)。この場合、DESERV の呼び出し側に戻る前に、タスク・レベル出口に制御が渡されて、DESERV 機能が完了したことが示されます。タスク・レベル出口から戻った後で、DESERV は呼び出し側に戻ります。

グローバル出口が戻りコード 0 で戻った場合、DESERV 機能が実行され、後処理出口を呼び出します。後処理出口の呼び出しは、最初にグローバル出口に対して行われ、2 番目にタスク・レベル処理出口に対して行われます。このシーケンス (前処理出口シーケンスと逆の順序) は、SVC スクリーン・ルーチンと更新された SVC ルーチンの両方が残っていた場合に見られる戻りシーケンスをシミュレートするた

DESERV 出口

めに選択されたものです。下の図は、出口ルーチン呼び出しシーケンスを示しています。

```
DESERV GET, PUT, RENAME, DELETE or UPDATE is issued
enter DESERV GET, PUT, RENAME, DELETE, or UPDATE
  call task exit for pre-processing
  if return_code = 0 then
    call global exit for preprocessing
    if return_code = 0 then
      process GET, PUT, RENAME, DELETE, or UPDATE
      GOTO post_process_global
    else if return_code = 4 then
      GOTO post_process_task
  else if return_code = 4 then
    return to DESERV caller

Post_process_global:
  call global exit for post-processing

Post_process_task:
  call task exit for post-processing
  return to DESERV caller
```

図 33. 出口ルーチン呼び出しシーケンス

複数のタスク・レベル出口または複数のグローバル出口の確立

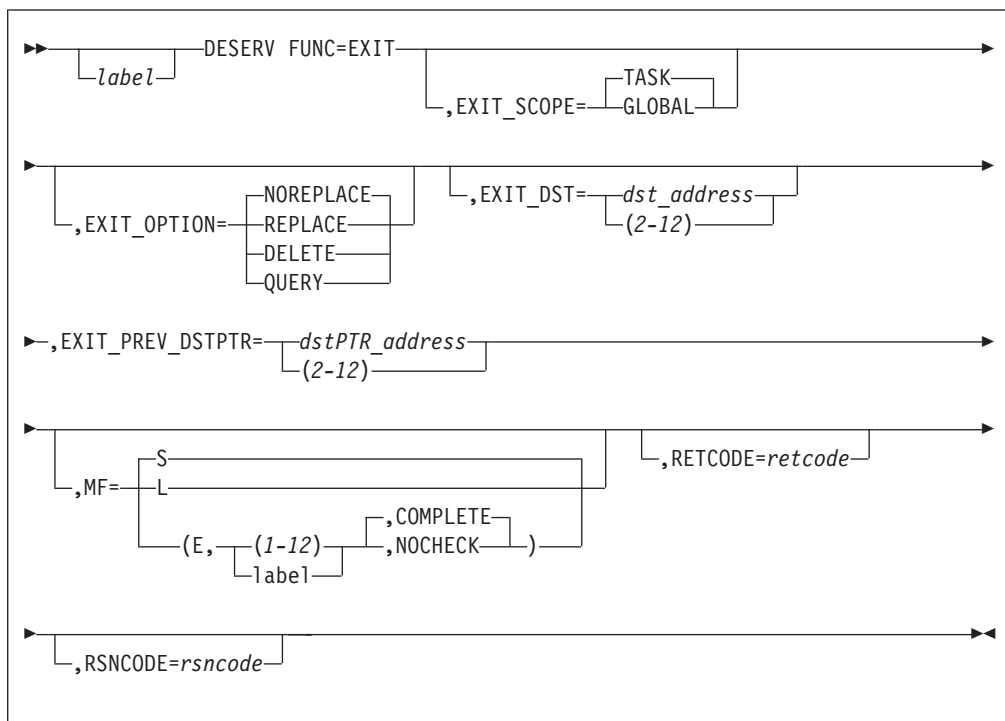
システムは DST (DESERV スクリーン・テーブル) によって DESERV 出口を識別します。システムは、タスクごとに多くても 1 つのタスク・レベル DST を保守し、システム用に多くても 1 つの DST を保守します (これはグローバル出口を表します)。ただし、複数の DESERV 出口を確立することも可能です。指定のタイプ (タスクまたはグローバル) の複数の出口をサポートするには、EXIT_OPTION=REPLACE を指定して DESERV FUNC=EXIT を発行すると、DESERV は、置換された DST のアドレス (または、DST が置換されなかった場合は、ゼロ) を戻します。この場合、前に定義された出口に制御を渡すのは、新しく定義された出口の責任です。

DESERV FUNC=EXIT の発行 (呼び出し環境)

割り込み:	使用可能
状態およびキー:	監視プログラム状態、またはシステム・キー (0-7)
ASC モード:	P=H=S
AMODE、RMODE:	制限なし
ロック:	保持しない
レジスター:	<ul style="list-style-type: none"> レジスター 15 と 0 以外のすべてのレジスターの内容は、戻るときに復元されます。 レジスター 15 に戻りコードが入り、レジスター 0 に理由コードが入ります。 保管域は必要ありません。
逐次化の要件:	なし。DESERV が逐次化を保守します。そのため、呼び出し側は ENQ のような逐次化を提供する必要はありません。

呼び出し構文

以下の図は、DESERV EXIT 機能の構文を示しています。

**FUNC=EXIT**

DESERV 出口で操作する DESERV 機能を要求します。MF=E が「NOCHECK」を指定し、他のキーワードを付けずにコーディングされている場合、または MF=L が他のキーワードを付けずにコーディングされている場合を除いて、このキーワードは常に必要です。MF=E の場合、MF=L DESERV マクロ呼び出しで指定された FUNC キーワードおよびその他のキーワードが完全にコーディングされたものと想定されます。

EXIT_SCOPE=GLOBAL または TASK

指定された出口は TASK レベルであるか、GLOBAL レベルであるかを指定します。

EXIT_OPTION=REPLACE または NOREPLACE または DELETE または QUERY

EXIT_OPTION=REPLACE または NOREPLACE がコーディングされている場合、この DESERV FUNC=EXIT の呼び出しは既存の DESERV EXIT (EXIT_SCOPE パラメーターで指定された TASK または GLOBAL) を置き換える必要があるかどうかを指定します。EXIT_PREV_DST は、既存の出口を戻すか、または存在しない場合はゼロに設定されます。

EXIT_OPTION=DELETE がコーディングされている場合、現行出口を削除すること (EXIT_OPTION=DELETE) を指示します。この場合、EXIT_DST パラメーターは、削除する DST のアドレスを指定します。このアドレスは、比較交換操作で比較値として使用され、現在アクティブの出口のみが削除されます。

EXIT_PREV_DSTPTR パラメーターで指定された DST アドレスは、交換値として使用されます。

EXIT_OPTION=QUERY がコーディングされている場合、EXIT_PREV_DSTPTR パラメーターを介して現行出口 DST アドレスを戻すことを指示します。

EXIT_DST=deserv_exit_screen_table RX タイプ・アドレス、または (2-12)

DESERV スクリーン・テーブル (DST) のアドレスを指定します。スクリーン・テーブルは、IGWDES マッピング・マクロの DST DSECT によってマップされます。EXIT_OPTION が NOREPLACE または REPLACE の場合、このパラメーターは DST のアドレスを定義し、操作が成功した場合、現行のアクティブ出口になる出口ルーチン・アドレスを定義します。EXIT_OPTION が DELETE の場合、このパラメーターは、現行 DST を削除する際に比較交換操作の比較値として使用するアドレスを持つ DST を定義します (入力 DST のアドレスを比較値として使用)。比較が失敗した場合、DESERV は戻りコードと理由コードを戻します。EXIT_OPTION=QUERY がコーディングされている場合、このパラメーターは必要ありません。

EXIT_PREV_DSTPTR=addr_of_deserv_exit_screen_table RX タイプ・アドレス、または (2-12)

EXIT_PREV_DSTPTR は、EXIT_OPTION が NOREPLACE、REPLACE、または QUERY に指定されている場合は、出力パラメーター、EXIT_OPTION が DELETE に指定されている場合は、入力パラメーターです。

EXIT_OPTION=NOREPLACE または QUERY の場合、このパラメーターは、DESERV が現行 DST のアドレス (DST が存在しない場合は、ゼロ) を戻す 4 バイト・フィールドを指定します。

EXIT_OPTION=REPLACE の場合、このパラメーターは、DESERV が正常に置換された DST のアドレス (前の DST が存在しない場合は、ゼロ) を戻す 4 バイト・フィールドを指定します。

EXIT_OPTION=DELETE の場合、このパラメーターは、DESERV が現行 DST として復元する DST を指す 4 バイト・フィールドを指定します。このアドレスは、比較交換操作の交換値です。

MF=S または L または {E,{(1-12) または label}{COMPLETE または NOCHECK}} RX タイプ・アドレス、または (1-12) - MF=E の 2 番目の引数の場合。 デフォルト=S - MF キーワードが指定されていない場合。デフォルト=COMPLETE - MF=E が 3 番目の引数 (COMPLETE または NOCHECK) なしで指定されている場合。

マクロ展開のフォーマットを指定します。

標準形式 S は、必要なキーワードおよび無効なキーワードをすべて検査します。この形式は、ディレクトリー項目サービス・ルーチン呼び出しのためのパラメーター・リストとコードの完全なインライン展開を生成します。標準形式は、再入可能でないプログラム用、あるいはパラメーター・リストの値を変更しないプログラム用です。

L は、リスト形式のマクロを指定します。この形式は、リモート・パラメーター・リストを生成します。引数タイプ KEY または SYM のキーワードのみをコーディングできます。コードは生成されない (adcon が生成される) ため、レジスターは無効です。無効キーワード検査が行われます。

MF=L 呼び出しで設定されるデフォルトを持つキーワードは、MF=E 呼び出し時にデフォルトにリセットされません。

E は、実行形式のマクロを指定します。この形式は、リモート・パラメーター・リストを更新し (MF=L)、制御権を DESERV ルーチンに移動します。

MF=E 形式の 2 番目のパラメーターは、MF=L DESERV 呼び出しによって作成されたパラメーター・リストのアドレスです。このパラメーターは、RX タイプのアドレス (例えば、MF=L マクロ呼び出しからのラベル) または括弧で囲んだレジスターとして指定する必要があります。

3 番目のパラメーター (COMPLETE または NOCHECK) はオプションです。デフォルトは COMPLETE です。この引数は、必要なキーワードの検査を行うかどうかを指定します。MF=E が NOCHECK 引数を使用してコーディングされている場合、許可されるキーワードを何も指定しない、その一部を指定する、または全部を指定することができ、欠落しているキーワードは MF=L マクロ呼び出しでコーディングされたものと想定されます。MF=E が COMPLETE 引数を使用してコーディングされているか、デフォルトを取ることができる場合、パラメーター・リストはゼロに設定されます (パラメーター・リスト・ヘッダーを除く)。DESERV マクロのデフォルトは 0 なので、これはすべてをデフォルトに設定します。必要なキーワードをすべて指定しなければなりません。

RETCODE=retcode RX タイプ・アドレス、または (2-12)

DESERV によって戻される戻りコードを保管する場所のアドレスを指定します。MF=L マクロ形式では指定できません。デフォルトは、戻りコードを仮想ストレージに保管しないことです。RETCODE がコーディングされているかどうかに関係なく、戻りコードは常にレジスター 15 に戻されます。

RSNCODE=rsncode RX タイプ・アドレス、または (2-12)

DESERV によって戻される理由コードを保管する場所のアドレスを指定します。MF=L マクロ形式では指定できません。デフォルトは、戻りコードを仮想ストレージに保管しないことです。RETCODE がコーディングされているかどうかに関係なく、戻りコードは常にレジスター 0 に戻されます。

DESERV コードは、DFSMSdfp バージョン 1.1 からシステムで使用可能です。ただし、EXIT 機能呼び出すには、適切な PTF をシステムに適用して、サポートを使用可能にする必要があります。ご使用のプログラムをテストして、適切なレベルの DFSMS™ または PTF がインストールされているかどうかを調べることができます。

DFSMSdfp バージョン 1.1 または 1.2 がインストールされている場合、DFA ビット DFADEEXT を検査すると、適切な DESERV EXIT サポート・コードがインストールされているかどうかを判別できます。DFADEEXT がオンの場合、DESERV EXIT サポート・コードは使用可能です。

DFSMSdfp バージョン 1.3 以上がインストールされている場合、DESERV EXIT サポート・コードは使用可能です。DFARELS の DFA フィールドに DFSMS のリリース番号が入っています。ユーザー出口は、DFARELS を調べることにより、DFSMS リリースをテストできます。DFADEEXT は常にオンです。

サポート・コードが使用可能でない DFSMS で DESERV FUNC=EXIT インターフェイスが呼び出された場合、DESERV はエラー戻りコードと理由コードを戻します。

DESERV 出口のインストールまたは置換

プログラムは、タスク・モードで操作している場合、FUNC=EXIT と適切なパラメータを指定して DESERV マクロを発行することにより、DESERV 出口を確立できます。

EXIT_OPTION =	操作	パラメーター
REPLACE	新規の出口定義が、最新に定義された出口と置き換えられます。前に定義された出口の DST (DESERV スクリーン・テーブル) アドレスが、呼び出し側に戻されます。	EXIT_DST と EXIT_PREV_DSTPTR を指定する必要があります。
NOREPLACE (現在、出口が存在しない場合)	出口定義がアクティブにされます。前の DST アドレスとして、ゼロの値が戻されます。	上記を参照。
NOREPLACE (現在、出口が存在する場合)	現在定義されている出口がアクティブ出口のままです。前の DST のアドレスとして、現在定義されている DST のアドレスが戻されます。	上記を参照。

注: EXIT_OPTION=REPLACE が指定されている場合、呼び出し側は、前の DST アドレスが戻されることを予期する必要があります。最終的に新規出口を使用不可にする場合、古い DST アドレスに置き換えるのは、呼び出し側の責任です。出口ルーチンも、前に定義された出口が存在したことを認識し、前に定義された出口を呼び出すことを選択することが必要になる場合があります。

EXIT_DST は、DST アドレスを定義します。DESERV FUNC=EXIT の呼び出し側は、DST 用のストレージを所有および管理します。DST は、他のアプリケーションの管理下の制御ブロック内にカプセル化することもできます。これは、出口ルーチンを呼び出したアプリケーションの内容を判別するのに役立ちます。タスク・レベル出口で DST が使用されている場合、DST_FLAGS_PROP ビットをオンに設定して、この DST をこのタスクによって接続されるタスクに伝搬する必要があることを示すことができます。伝搬機能は、グローバル出口に対してはサポートされません。DST のフォーマットを表 58 に示します。

表 58. DESERV スクリーン・テーブル構造

オフセット	長さ、または ビット・パターン	名前	説明
0 (X'0')	20	DST	(構造)
0 (X'0')	16	DST_HEADER	(文字)
0 (X'0')	8	DST_ID	目印 'IGWDST' (文字)
08 (X'08')	4	DST_LEN	DST の長さ
	X'14'	DST_LEN_IV	DST_LEN で使用される定数
12 (X'0C')	1	DST_LEV	制御ブロック・レベル (符号なし)
	X'01'	DST_LEV_IV	DST_LEV で使用される定数

DESERV 出口

表 58. DESERV スクリーン・テーブル構造 (続き)

オフセット	長さ、または ビット・パターン	名前	説明
13 (X'0D')	1	DST_FLAGS	DST フラグ (符号なし)
	xxxx xxx.	-	予約済み
1	DST_FLAGS_PROP	この DST を下位レベル・タスクに 搬送する
14 (X'0E')	2	DST_RES	予約済み
16 (X'10')	4	DST_EXIT	出口ルーチン・スクリーン・テー ブルのアドレス (アドレス)

EXIT_PREV_DSTPTR は、この DESERV FUNC=EXIT 呼び出しの前に定義された DST のアドレスを戻します。

DESERV 出口の削除

DESERV 出口を確立したアプリケーションは、FUNC=EXIT を指定して DESERV マクロを発行することにより、現在アクティブの出口を削除できます。以下を指定できます。

表 59. DESERV 出口の削除

パラメーター	指定
EXIT_SCOPE	(アプリケーションに依存)
EXIT_OPTION	DELETE
EXIT_DST	削除する現在アクティブの DST
EXIT_PREV_DSTPTR	アクティブになる DST

注: DST のアドレスは、現在定義されている DST のアドレスと一致していなければなりません。アドレスが一致していない場合、DESERV はエラー戻りコードと理由コードを戻して、DELETE 操作が失敗したことを示し、アクティブの DST は影響を受けずに残ります。

タスク関連の出口の場合、タスクが終了すると、DESERV 出口は暗黙的に削除されます。グローバル出口は、DESERV FUNC=EXIT 呼び出しを発行して明示的にのみ削除できます。

DESERV 出口がアクティブかどうかの判別

DESERV 出口がアクティブかどうかを判別するには、FUNC=EXIT を指定して DESERV マクロを発行し、次のオプションを選択します。

表 60. DESERV 出口がアクティブかどうかの判別

パラメーター	選択
EXIT_OPTION	QUERY
EXIT_SCOPE	(アプリケーションに依存)

表 60. DESERV 出口がアクティブかどうかの判別 (続き)

パラメーター	選択
EXIT_PREV_DSTPTR	現在アクティブの DST アドレス (または、現在定義されている DST アドレスがない場合は、ゼロ) が戻される 4 バイト域のアドレス。

DESERV 出口の作成

出口処理

DESERV 出口は、DESERV GET、PUT、RENAME、UPDATE、または DELETE 機能処理の前に 1 回と、DESERV が呼び出し側に戻る直前に 1 回、制御を取得します。DESERV 出口は、キー 0 および監視プログラム状態で制御を受け取ります。レジスター 13 は、18 ワードのキー 0 レジスター保管域を指します。

DESX DSECT は、入力を出口ルーチンにマップし、IGWDES マクロで定義されます。レジスター 1 は、出口ルーチンに入るときの DESX を指します。DESX 構造を表 61 に示します。

表 61. DESX 構造マッピング DESERV 出口パラメーター・リスト

オフセット	長さ、または ビット・パターン	名前	説明
0 (X'0')	36	DESX	(構造)
0 (X'0')	16	DESX_HEADER	(文字)
0 (X'0')	8	DESX_ID	目印 - IGWDESX (文字)
08 (X'08')	4	DESX_LEN	DESX の長さ (符号付き)
	X'24'	DESX_LEN_IV	DESX_LEN で使用される定数
12 (X'0C')	1	DESX_LEV	制御ブロック・レベル (文字)
	X'01'	DESX_LEV_IV	DESX_LEV で使用される定数
13 (X'0D')	3	-	予約済み
16 (X'10')	4	DESX_DESP_PTR	呼び出し側の DESP のアドレス (アドレス)
20 (X'14')	4	DESX_DST	DESERV スクリーン・テーブルのアドレス (アドレス)
24 (X'18')	1	DESX_CALLER_KEY	ビット 0-3 の DESERV 呼び出し側キー (符号なし)
25 (X'19')	1	DESX_FLAGS	(ビット・ストリング)
	1... ..	DESX_BLDL_BIT	DESERV は、この GET 要求を処理するために BLDL を発行します。
	.1..	DESX_PREV_BIT	EXIT は DESERV PUT または GET 機能の前に呼び出されました。
	..1.	DESX_POST_BIT	EXIT は DESERV PUT または GET 機能の後に呼び出されました。
26 (X'1A')	2	-	予約済み

表 61. DESX 構造マッピング DESERV 出口パラメーター・リスト (続き)

オフセット	長さ、または ビット・パターン	名前	説明
28 (X'1C')	4	DESX_RETURN_CODE	DESERV 呼び出し側に戻される戻りコード (符号なし)
32 (X'20')	4	DESX_REASON_CODE	DESERV 呼び出し側に戻される理由コード (符号なし)

DESERV 戻りコードと理由コードは、PUT コードを除いて、「z/OS DFSMS データ・セットの使用法」に記載されています。PUT の戻りコードと理由コードは、310 ページの図 34 を参照してください。

GET 機能に関連したパラメーター

DESERV 出口が DESERV GET 機能呼び出しのために制御を取得した場合、DESX_DESP_PTR は DESERV パラメーター・リストを指します。DESP フィールドが DESP_FUNC=X'01' (DESP_FUNC_GET) の場合、これは GET 機能のパラメーター・リストを示します。DESERV GET 呼び出しに関連したフィールドの DESP 構造は、295 ページの表 62 を参照してください。

DESERV GET は、選択されたメンバーに関する情報を戻します。この情報は、DESB 構造に戻されます。DESB は、IGWDES マクロの DESB DSECT によってマップされます。DESB 用のストレージが DESERV GET によって提供される場合、DESP_AREA_PTR フィールドには、このストレージのアドレスが入ります。代わりに、呼び出し側は、DESERV GET が DESB 用のストレージを取得するように要求する場合があります。この場合、DESP_AREAPTR_PTR フィールドには、DESERV GET が DESB のアドレスを戻す 4 バイト域のアドレスが入ります。DESB は、DESP_SUBPOOL によって識別されたサブプール (または、デフォルトのサブプール・ゼロ) で取得します。フラグ DESP_SUBPOOL_FLG は、サブプールが DESERV GET の呼び出し側によって明示的に指定されたかどうかを示します。

DESERV GET 呼び出しは、検索するメンバーを、名前リスト、PDS 形式のディレクトリー項目、または SMDE によって識別します。DESP フィールド DESP_GETTYPE は、取得タイプを定義します。取得タイプが PDSDE の場合 (検索されるメンバーは PDS 形式のディレクトリー項目によって定義されます)、DESP_PDSDE_PTR は BLDL マクロによって戻されるディレクトリー項目を指します。IHAPDS マクロの PDS2 DSECT が、この構造をマップします。PDSDE 取得タイプの DESERV GET の機能は、PDS スタイル・ディレクトリー項目の連結番号で識別されるライブラリー・タイプによって異なります。連結番号が PDS を示している場合、GET 機能は単に PDS スタイルのディレクトリー項目を SMDE に変換するだけです。連結番号が PDSE を示している場合、GET 機能は、PDS2TTRP フィールドで示されたメンバーに接続し、該当する SMDE を戻します。いずれの場合も、SMDE は出力バッファのデータ部分 (DESB、以下にマップ) に戻されます (PDSE の場合は、メンバーが実際に存在する場合)。

取得タイプが名前リストの場合、DESL 域は検索する名前を指します。DESP_NAME_LIST_PTR は DESL を指し、IGWDES マクロの DESL DSECT がそ

れをマップします。DESL は、DESP フィールド DESP_NAME_LIST2 で定義された項目数からなる配列です。DESL パラメーター・リストは、297 ページの表 63 に示します。

取得タイプが SMDE の場合、DESP_SMDE_PTR は DESERV GET によって戻されたシステム管理ディレクトリー項目 (SMDE) を指します。DESERV GET は、SMDE によって示されたメンバーに接続します。DESERV GET は、SMDE のコピーを出力 DESB に戻します。戻される SMDE は勿論、更新済みの接続トークンおよび接続 ID フィールドを持っています。

これらは、検索する PDS または PDSE の DESERV GET の表示を制御する 2 つの入力フラグです。DESP_C370LIB フラグがオンの場合、PDS は C370LIB として表示されることがあります。これは、PDS が特殊メンバー名 @@DC370\$ を持っている場合、このメンバーは PDS の「実」ディレクトリーとして扱われることを意味します。DESP_SYSTEM_DCB がオンの場合、これは、呼び出し側 (許可プログラムであることが必要) が、この DCB は「システムが所有」しており、どの DEB チェーン上にも存在しないので、DEBCHK を行ってはならないと指示したことを示します。

ヒント: 特殊メンバー名 @@DC370\$ は、ご使用の画面またはプリンターに正しく表示されないことがあります。最初の 2 文字は X'7C' で、最後の 2 文字は X'5B' です。

表 62. DESERV GET 呼び出しの DESP の構造

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	104	DESP	DE サービス・パラメーター・リスト (構造)
00 (X'00')	16	DESP_HEADER	(文字) 'IGWDESP'
00 (X'00')	8	DESP_ID	目印 IGWDESP (文字)
08 (X'08')	24	DESP_LEN	DESP の長さ (符号付き)
	X'04'	DESP_LEN_IV	DESP_LEN で使用される定数
12 (X'0C')	1	DESP_LEV	制御ブロック・レベル (文字)
	X'01'	DESP_LEV_IV	DESP_LEV で使用される定数
13 (X'0D')	3	-	予約済み
16 (X'10')	1	DESP_FUNC	機能タイプ (符号なし)
	X'07'	DESP_FUNC_DELETE	機能は DELETE
	X'08'	DESP_FUNC_RENAME	機能は RENAME
	X'09'	DESP_FUNC_UPDATE	機能は UPDATE
	X'04'	DESP_FUNC_PUT	機能は PUT
	X'01'	DESP_FUNC_GET	機能は GET
	X'00'	DESP_FUNC_OMITTED	機能は省略
17 (X'11')	3	-	予約済み
20 (X'14')	4	-	予約済み
24 (X'18')	12	DESP_DATA	機能データ (文字)

DESERV 出口

表 62. DESERV GET 呼び出しの DESP の構造 (続き)

オフセット	長さ、または ビット・パターン	名前	説明
24 (X'18')	2	DESP_FLAGS	フラグ (ビット・ストリング)
	1... ..	DESP_BYPASS_LLA	0=USE LLA, 1=BYPASS LLA
	.x.. ..	-	予約済み
	..1.	DESP_SUBPOOL_FLG	0=SUBPOOL は指定されない、1=SUBPOOL を指定
	...1	DESP_C370LIB	1= PDS を C370LIB として扱います (@@DC370\$ メンバーが存在する場合)
 xx..	-	予約済み
1.	DESP_SYSTEM_DCB	1= DCB をシステム DCB として扱います。
26 (X'1A')	1	-	予約済み
27 (X'1B')	1	-	予約済み
28 (X'1C')	1	DESP_LIBTYPE	DCB または DEB 入力 =X'02' か、DEB 入力 =X'01' かを示します。(符号なし)
	X'02'	DESP_LIBTYPE_DCB	DESP_LIBTYPE で使用される定数
	X'01'	DESP_LIBTYPE_DEB	DESP_LIBTYPE で使用される定数
	X'00'	DESP_LIBTYPE_OMITTED	DESP_LIBTYPE で使用される定数
29 (X'1D')	1	DESP_GETTYPE	入力が名前リストか、PDSDE かを示します。(NAME_LIST 入力=(X'01'、PDSDE 入力 =X'02') (符号なし)
	X'03'	DESP_GETTYPE_SMDE	DESP_GETTYPE で使用される定数
	X'02'	DESP_GETTYPE_PDSDE	DESP_GETTYPE で使用される定数
	X'01'	DESP_GETTYPE_NAME_LIST	DESP_GETTYPE で使用される定数
	X'00'	DESP_GETTYPE_OMITTED	DESP_GETTYPE で使用される定数
30 (X'1E')	1	-	予約済み
31 (X'1F')	1	-	予約済み
32 (X'20')	1	-	予約済み
33 (X'21')	1	DESP_SUBPOOL	DESB を取得するためのサブプール番号
34 (X'22')	1	DESP_CONN_INTENT	接続意図 (符号なし)
	X'03'	DESP_CONN_INTENT_INPUT	INPUT
	X'02'	DESP_CONN_INTENT_EXEC	EXEC
	X'01'	DESP_CONN_INTENT_HOLD	HOLD
	X'00'	DESP_CONN_INTENT_NONE	なし
35 (X'23')	1	-	予約済み
36 (X'24')	4	DESP_DCB_PTR	DCB アドレス、DESP_LIBTYPE=X'02' の場合に有効 (アドレス)
40 (X'28')	4	DESP_DEB_PTR	DEB アドレス、DESP_LIBTYPE=X'01' の場合に有効 (アドレス)
44 (X'2C')	4	DESP_CONN_ID_PTR	接続 ID アドレス
48 (X'30')	4	DESP_AREAPTR_PTR	バッファ・アドレスの出力フィールドのアドレス

表 62. DESERV GET 呼び出しの DESP の構造 (続き)

オフセット	長さ、または ビット・パターン	名前	説明
52 (X'34')	4	DESP_AREA_PTR	バッファ・アドレス
56 (X'38')	4	DESP_AREA2	バッファ長 (符号なし)
60 (X'3C')	4	-	予約済み
64 (X'40')	4	-	予約済み
68 (X'44')	4	DESP_ENTRY_GAP	項目ギャップ・サイズ (符号付き)
72 (X'48')	4	-	予約済み
76 (X'4C')	4	-	予約済み
80 (X'50')	4	DESP_NAME_LIST_PTR	名前リスト・アドレス、 DESP_GETTYPE=X'01' の場合に有効 (アドレス)
84 (X'54')	4	DESP_NAME_LIST2	入力リストの項目数、DESP_GETTYPE=X'01' の場合に有効 (符号なし)
88 (X'58')	4	-	予約済み
92 (X'5C')	4	DESP_PDSDE_PTR	BLDL ディレクトリー項目アドレス、 DESP_GETTYPE=X'02' の場合に有効 (アドレス)
92 (X'5C')	4	DESP_SMDE_PTR	SMDE ディレクトリー項目アドレス、 DESP_GETTYPE=X'03' の場合に有効 (アドレス)
96 (X'60')	4	-	予約済み
100 (X'64')	4	-	予約済み

表 63. DESL 構造

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	16	DESL	名前リスト (構造)
00 (X'00')	16	DESL_ENTRY	名前リスト項目 (文字)
00 (X'00')	1	DESL_FLAGS	フラグ (符号なし)
	1... ..	DESL_MODULE_BUFFERED_ LLA	モジュールは LLA によってステージ
1 (X'1')	1	DESL_CODE	結果コード (新しい名前が存在=X'03'、エラー=X'02'、未検出または未処理=X'01'、検出=X'00') (符号なし)
	X'03'	DESL_CODE_NEWNAME_ EXISTS	func=rename の場合、新しい名前がすでに PDSE に存在したことを示します。
	X'02'	DESL_CODE_ERROR	予期しないエラーが発生しました。 DESL_ERRCODE フィールドは DESRF 値 に設定されます。
	X'01'	DESL_CODE_NOTFOUND	項目が見つからないか、項目が未処理で す。func=rename の場合、古い名前が見つ かりませんでした。
	X'00'	DESL_CODE_SUCC	項目は正常に処理されました。

DESERV 出口

表 63. DESL 構造 (続き)

オフセット	長さ、または ビット・パターン	名前	説明
2 (X'2')	2	DESL_ERRCODE	エラー理由コード (エラーの場合、 DESERV 理由コードの下位ハーフワード) (符号なし)
4 (X'4')	4	-	予約済み
08 (X'08')	4	DESL_SMDE_PTR	DESB 内の SMDE へのポインター。GET 機能の出力、UPDATE 機能の入力 (アドレ ス)
08 (X'08')	4	DESL_NEW_NAME_PTR	RENAME 機能の新しい名前 (DESN) 記述 子へのポインター (アドレス)
12 (X'0C')	4	DESL_NAME_PTR	GET および DELETE 機能の名前 (DESN) 記述子へのポインター (アドレス)
12 (X'0C')	4	DESL_OLD_NAME_PTR	RENAME 機能の古い名前 (DESN) 記述子 へのポインター (アドレス)

DESERV GET 呼び出し側は、可変長の名前を指す DESL を作成します。DESN DSECT は、IGWDES マクロでこれらの名前をマップします。DESN パラメータ
ー・リストは、表 64 を参照してください。

表 64. DESN パラメーター・リスト

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	可変	DESN	名前レコード (構造)
00 (X'00')	2	DESN_LEN	後続の名前の長さ (符号なし)
2 (X'02')	可変	DESN_VAL	名前データ (文字)

DESP の DESP_CONN_INTENT フィールドは、呼び出し側によって要求された接
続意図を示します。接続意図は、PDSE 内で名前が見つかった場合にのみ効力を持
ちます。接続意図が DESP_CONN_INTENT_HOLD (X'01') の場合、効力は BLDL
呼び出しと同様です (メンバーは HOLD 用に接続されており、これはメンバーを読
み取るには不十分だからです)。接続意図が DESP_CONN_EXEC (X'02') または
DESP_CONN_INTENT_INPUT (X'03') の場合、効力は FIND 呼び出しと同様です
(メンバーは接続され、十分な制御ブロックが構築されており、メンバーを読み取れ
るからです)。GET 機能は、現行では NONE の接続意図をサポートしていません。

DESERV GET からの出力は、DESL 内のフラグとエラー・コード (取得タイプが名
前リストの場合) および SMDE (システム管理ディレクトリー項目) ポインターで
す。取得タイプ (gettype) が名前リストの場合、SMDE は DESL_SMDE_PTR フィ
ールドによって指示されます。取得タイプが PDSDE の場合、SMDE は DESB 内
のラベル DESB_DATA にあります。SMDE は IGWSMDE マクロの SMDE
DSECT および IEWPMAR マクロの PMAR DSECT によってマップされます。
SMDE は、DESERV GET の呼び出し側によって提供される出力バッファに常駐
します。出力バッファは、IGWDES マクロの DESP DSECT によってマップされ
ます。DESB 構造を 299 ページの表 65 に示します。

基本 SMDE フォーマットは、表 66 に示します。

表 65. DESB パラメーター・リスト

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	可変	DESB	DEServ バッファ・ヘッダー (構造)
00 (X'00')	40	DESB_FIXED	(文字)
00 (X'00')	16	DESB_HEADER	(文字)
00 (X'00')	8	DESB_ID	目印 - IGWDESB (文字)
08 (X'08')	4	DESB_LEN	バッファの長さ (符号付き)
12 (X'0C')	1	DESB_LEV	制御ブロック・レベル (文字)
	X'01'	DESB_LEV_IV	DESB_LEV で使用される定数
13 (X'0D')	3	-	予約済み
16 (X'10')	4	DESB_NEXT	次のバッファのポインター (アドレス)
20 (X'14')	4	-	予約済み
24 (X'18')	4	DESB_COUNT	このバッファ内の項目数 (符号なし)
28 (X'1C')	4	DESB_AVAIL	バッファ内のフリー・スペースの開始 (アドレス)
32 (X'20')	1	-	予約済み
33 (X'21')	1	DESB_SUBPOOL	サブプール番号 (符号なし)
34 (X'22')	2	DESB_GAP_LEN	ユーザーが要求したギャップの長さ (符号なし)
36 (X'24')	4	-	予約済み
40 (X'28')	可変	DESB_DATA	データ域の開始 (文字)

表 66. SMDE フォーマット

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	可変	SMDE	メンバー・ディレクトリー項目 (構造)
00 (X'00')	44	SMDE_BASIC	基本セクションの開始 (文字)
00 (X'00')	16	SMDE_HDR	ヘッダー (文字)
00 (X'00')	8	SMDE_ID	目印 (文字)
08 (X'08')	4	SMDE_LEN	制御ブロックの長さ。これは、SMDE セクションのサイズとユーザー・データのサイズの合計。(符号なし)
12 (X'0C')	1	SMDE_LVL	SMDE バージョン番号 (符号なし)
	X'01'	SMDE_LVL_VAL	SMDE_LVL で使用される定数
13 (X'0D')	3	-	予約済み

DESERV 出口

表 66. SMDE フォーマット (続き)

オフセット	長さ、または ビット・パターン	名前	説明
16 (X'10')	1	SMDE_LIBTYPE	ソース・ライブラリー・タイプ。可能な値は、SMDE_LIBTYPE_XXX のような名前を使用して、下に宣言されています。(符号なし)
	X'03'	SMDE_C370LIB	SMDE_LIBTYPE で使用される定数
	X'02'	SMDE_LIBTYPE_HFS	SMDE_LIBTYPE で使用される定数
	X'01'	SMDE_LIBTYPE_PDSE	SMDE_LIBTYPE で使用される定数
	X'00'	SMDE_LIBTYPE_PDS	SMDE_LIBTYPE で使用される定数
17 (X'11')	1	SMDE_FLAG	フラグ・バイト (ビット・ストリング)
	1...	SMDE_FLAG_ALIAS	項目は別名です。
	.1..	SMDE_FLAG LMOD	メンバーはプログラムです。
	..xx xxxx	*	予約済み
18 (X'12')	2	-	予約済み、ゼロでなければなりません。
20 (X'14')	5	-	拡張 MLTK (文字)
20 (X'14')	1	-	予約済み、ゼロでなければなりません。
21 (X'15')	4	SMDE_MLTK	MLT と連結番号 (文字)
21 (X'15')	3	SMDE_MLT	メンバーの MLT - HFS の場合はゼロ (文字)
24 (X'18')	1	SMDE_CNCT	連結番号 (符号なし)
25 (X'19')	1	SMDE_LIBF	ライブラリー・フラグ - Z バイト (符号なし)
	X'02'	SMDE_LIBF_TASKLIB	SMDE_LIBF で使用される定数
	X'01'	SMDE_LIBF_LINKLIB	SMDE_LIBF で使用される定数
	X'00'	SMDE_LIBF_PRIVATE	SMDE_LIBF で使用される定数
26 (X'1A')	2	SMDE_NAME_OFF	名前オフセット (符号付き)
28 (X'1C')	2	SMDE_USRD_LEN	ユーザー・データ長 (符号付き)
28 (X'1C')	2	SMDE_PMAR_LEN	プログラム管理属性レコード・セクション (PMAR、PMARR、PMARL) の長さの合計 (符号付き)
30 (X'1E')	2	SMDE_USERD_OFF	ユーザー・データ・オフセット (符号付き)
30 (X'1E')	2	SMDE_PMAR_OFF	プログラム管理属性レコード・オフセット (符号付き)
32 (X'20')	2	SMDE_TOKEN_LEN	トークン長さ (符号付き)
34 (X'22')	2	SMDE_TOKEN_OFF	トークン・データ・オフセット (符号付き)
36 (X'24')	2	SMDE_PNAME_OFF	基本名オフセット、非別名 SMDES の場合、またはライブラリー・タイプが PDS で、これがプログラムでない場合は、ゼロ。(符号付き)
38 (X'26')	2	SMDE_NLST_CNT	ユーザー・データ・フィールドの開始に存在する注釈リスト項目の数。非 PDS メンバーの場合は、常にゼロ。(符号付き)
40 (X'28')	4	-	予約済み
44 (X'2C')	可変	SMDE_SECTIONS	項目セクションの開始 (文字)

表 67 から表 70 は、オプション SMDE_SECTIONS、または SMDE の拡張を示しています。

表 67. ディレクトリー項目名セクション

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	可変	SMDE_NAME	名前記述子 (構造)
00 (X'00')	2	SMDE_NAME_LEN	項目名の長さ (符号付き)
2 (X'02')	可変	SMDE_NAME_VAL	項目名 (文字)

表 68. ディレクトリー項目注釈リスト・セクション (PDS のみ)

オフセット	長さ、またはビット・ パターン	名前	説明
00 (X'00')	可変	SMDE_NLST	拡張注釈リスト (構造)
00 (X'00')	4	SMDE_NLST_ENTRY	注釈リスト項目 (文字)
00 (X'00')	3	SMDE_NLST_RLT	注釈リスト・レコード位置トークン (文字)
3 (X'03')	1	SMDE_NLST_NUM	この注釈リスト・ブロックによって記述された RLT の数。0 の場合、これは注釈リストではなく、データ・ブロックです。(符号なし)

表 69. ディレクトリー項目トークン・セクション

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	32	SMDE_TOKEN	(構造)
00 (X'00')	4	SMDE_TOKEN_CONNID	CONNECT_IDENTIFIER (符号なし)
4 (X'04')	4	SMDE_TOKEN_ITEMNO	項目番号 (符号なし)
08 (X'08')	24	SMDE_TOKEN_FT	ファイル・トークン (文字)

表 70. ディレクトリー項目基本名セクション

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	可変	SMDE_PNAME	基本名記述子 (構造)
00 (X'00')	2	SMDE_PNAME_LEN	基本名の長さ (符号付き)
2 (X'02')	可変	SMDE_PNAME_VAL	基本名 (文字)

SMDE がプログラム (ロード・モジュールまたはプログラム・オブジェクト) のディレクトリー項目を表す場合、プログラムの属性は PMAR 構造によって定義されます。PMAR は SMDE のサブフィールドで、そのオフセットはフィールド SMDE_PMAR_OFF によって定義されます。302 ページの表 71 は、基本 PMAR 定義を示しています。304 ページの表 72 と 306 ページの表 73 は、それぞれ、プログラム・オブジェクト (PMARL) とロード・モジュール (PMARR) の拡張 PMAR を示しています。

DESERV 出口

SMDE が PDS または PDSE のデータ・メンバーを表す場合、SMDE_USRD_OFF フィールドは、ディレクトリー項目のユーザー・データの SMDE 内のオフセットを示します。

表 71. ディレクトリー項目名前セクション： データは常に SMDE 内のオフセット SMDE_PMAR_OFF に存在しません。

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	30	PMAR	プログラム・ユーザー・データの基本セクション (構造)
00 (X'00')	30	PMAR_ENTRY	PMAR セクションの代替名 (文字)
00 (X'00')	2	PMAR_SLEN	セクションの長さ (符号なし)
2 (X'02')	1	PMAR_LVL	PMAR フォーマット・レベル (符号なし)
	X'02'	PMAR_LVL_VAL	PMAR で使用される定数
	X'01'	PMAR_PM1_VAL	PMAR で使用される定数
	X'02'	PMAR_PM2_VAL	PMAR で使用される定数
3 (X'03')	1	PMAR_PLVL	バインド・プロセッサ作成オブジェクト 1 - E レベル・リンケージ・エディター、 2 - F レベル・リンケージ・エディター、 3 - (VS1/VS2) リンケージ・エディター、 4 - XA リンケージ・エディター、 5 - バインダー・バージョン 1。(符号なし)
	X'01'	PMAR_PLVL_E_VAL	PMAR_PLVL で使用される定数
	X'02'	PMAR_PLVL_F_VAL	PMAR_PLVL で使用される定数
	X'03'	PMAR_PLVL_AOS_VAL	PMAR_PLVL で使用される定数
	X'04'	PMAR_PLVL_XA_VAL	PMAR_PLVL で使用される定数
	X'05'	PMAR_PLVL_B1_VAL	PMAR_PLVL で使用される定数
	X'06'	PMAR_PLVL_B2_VAL	PMAR_PLVL で使用される定数
4 (X'04')	4	PMAR_ATR	属性バイト (文字)
4 (X'04')	1	PMAR_ATR1	最初の属性バイト。このフラグは、マクロ IHAPDS によって宣言された PDS2ATR1 内の対応するフラグと同じオフセットになければなりません。(ビット・ストリング)
	1...	PMAR_RENT	再入可能
	.1.	PMAR_REUS	再使用可能
	..1.	PMAR_OVLY	オーバーレイ構造
	...1	PMAR_TEST	テストされるモジュール - TSO/E TEST
 1...	PMAR_LOAD	ロード可能のみ
1..	PMAR_SCTR	分散形式
1.	PMAR_EXEC	実行可能
1	PMAR_1BLK	ロード・モジュールは、1 ブロックのテキスト・データのみを含み、RLD データは持っていません。

表 71. ディレクトリー項目名前セクション (続き): データは常に SMDE 内のオフセット SMDE_PMAR_OFF に存在します。

オフセット	長さ、または ビット・パターン	名前	説明
5 (X'05')	1	PMAR_ATR2	2 番目の属性バイト。このフラグは、マクロ IHAPDS によって宣言された PDS2ATR2 内の対応するフラグと同じオフセットになければなりません。(ビット・ストリング)
	1..	PMAR_FLVL	オンの場合、プログラムは E レベル・リンケージ・エディターによって処理することはできません。オフの場合、プログラムは任意のレベルのリンケージ・エディターまたはバインダーによって処理できます。
	.1..	PMAR_ORGO	リンケージ・エディターが割り当てたテキストの最初のブロックの起点はゼロです。
	..x.	-	予約済み
	...1	PMAR_NRLD	プログラムは RLD 項目を含んでいません。
 1..	PMAR_NREP	モジュールはリンケージ・エディターによって再処理できません。
1..	PMAR_TSTN	モジュールは TSO/E TEST シンボル・レコードを含んでいます。
x.	-	予約済み
1	PMAR_REFR	リフレッシュ可能プログラム
6 (X'06')	1	PMAR_ATR3	3 番目の属性バイト。(ビット・ストリング)
6 (X'06')	1	PMAR_FTB1	フラグ・バイトの代替名。このフラグは、マクロ IHAPDS によって宣言された PDS2FTB1 内の対応するフラグと同じオフセットになければなりません。(ビット・ストリング)
	x...	-	予約済み
	.1..	PMAR_BIG	このプログラムは 16 MB 以上の仮想ストレージを必要とします。
	..1.	PMAR_PAGA	ページ位置合わせが必要です。
	...1	PMAR_XSSI	SSI 情報が存在します。
 1..	PMAR_XAPF	APF 情報が存在します。
1..	PMAR_LFMT	PMARL が PMAR の後に続きます。
xx	-	予約済み
7 (X'07')	1	PMAR_ATR4	4 番目の属性バイト (ビット・ストリング)

DESERV 出口

表 71. ディレクトリー項目名前セクション (続き): データは常に SMDE 内のオフセット SMDE_PMAR_OFF に存在します。

オフセット	長さ、または ビット・パターン	名前	説明
7 (X'07')	1	PMAR_FTB2	フラグ・バイトの代替名。このフラグは、マクロ IHAPDS によって宣言された PDS2FTB2 内の対応するフラグと同じオフセットになければなりません。(ビット・ストリング)
	1...	PMAR_ALTP	代替基本フラグ。基本名に対してオンの場合、基本名がバインダーによって生成されたことを示します。別名に対してオンの場合、バインド時に基本名として長い別名が指定されたことを示します。
	.xx.	-	予約済み
	...1 ...	PMAR_RMOD	RMODE は ANY です。
 xx..	PMAR_AAMD	別名エンタリー・ポイント・アドレッシング・モード。B'00' の場合、AMODE は 24 です。B'10' の場合、AMODE は 31 です。B'11' の場合、AMODE は ANY です。
xx	PMAR_MAMD	メイン・エンタリー・ポイント・アドレッシング・モード。B'00' の場合、AMODE は 24 です。B'10' の場合、AMODE は 31 です。B'11' の場合、AMODE は ANY です。
08 (X'08')	1	-	予約済み
9 (X'09')	1	PMAR_AC	APF 許可コード (符号なし)
10 (X'0A')	4	PMAR_STOR	必要な仮想ストレージ (符号なし)
14 (X'0E')	4	PMAR_EPM	メイン・エンタリー・ポイント・オフセット (符号なし)
18 (X'12')	4	PMAR_EPA	このエンタリー・ポイントのオフセット (符号なし)
22 (X'16')	4	PMAR_SSI	SSI 情報 (ビット・ストリング)
22 (X'16')	1	PMAR_CHLV	メンバーの変更レベル (符号なし)
23 (X'17')	1	PMAR_SSFb	SSI フラグ・バイト (ビット・ストリング)
24 (X'18')	2	PMAR_MSER	メンバー・シリアル番号 (予約済み)
26 (X'1A')	4	-	予約済み
30 (X'1E')	可変	PMAR_END	基本セクションの終了 (文字)

表 72. プログラム・オブジェクトに固有の LLoader 属性: PMAR_LFMT=ON の場合、このセクションが PMAR 基本セクションの後に続きます。

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	50	PMARL	プログラム・オブジェクトの LLoader セクション (構造)
00 (X'00')	2	PMARL_SLEN	セクションの長さ (符号なし)
2 (X'02')	48	PMARL_DATA	セクション・データ (文字)
2 (X'02')	4	PMARL_ATR	属性バイト (文字)

表 72. プログラム・オブジェクトに固有の *LSLoader* 属性 (続き): *PMAR_LFMT=ON* の場合、このセクションが *PMAR* 基本セクションの後に続きます。

オフセット	長さ、または ビット・パターン	名前	説明
2 (X'02')	1	PMARL_ATR1	5 番目の属性バイト (ビット・ストリング)
	1...	PMARL_NMIG	このプログラム・オブジェクトは直接 PDS ロード・モジュール・フォーマットに変換できません。
	.1.	PMARL_PRIM	FETCHOPT PRIME オプション
	.1.	PMARL_PACK	FETCHOPT PACK オプション
	...x xxxxx	-	予約済み
3 (X'03')	1	PMARL_ATR2	5 番目の属性バイト (ビット・ストリング)
	1...	PMARL_CMPR	圧縮フォーマット・モジュール
	.1.	PMARL_1RMOD	最初のセグメントは RMODE Any。PM2 レベル PO に対してのみ設定。
	.1.	PMARL_2RMOD	2 番目のセグメントは RMODE Any。少なくとも 2 つのセグメントがある場合、PM2 レベル PO に対して設定。
	...1	PMARL_SEGM	ローダー・データには、>1 のロード可能項目を含むセグメント・テーブルまたは Gas テーブルが含まれています。PM2 レベル PO に対してのみ設定。
 1..	PMARL_1ALIN	最初のセグメントはページ位置合わせされません。PM2 レベル PO に対してのみ設定。
1..	PMARL_2ALIN	2 番目のセグメントはページ位置合わせされません。少なくとも 2 セグメントある場合、PM2 レベル PO に対して設定。
1.	PMARL_FILL	FILL オプションが指定されました。PM2 レベル PO に対してのみ設定。
.... ...x	-	予約済み	
4 (X'04')	1	PMARL_FILLVAL	充てん文字文字値。PM2 レベル PO に対してのみ設定。
5 (X'05')	1	-	予約済み

以下の注記は、使用する目的で示すものではありません。情報としてのみ、ここに記載します。

6 (X'06')	4	PMARL_MPGS	DASD 上のプログラムの全長 (ページ数) (符号なし)
10 (X'0A')	40	PMARL_MDAT	DASD プログラム記述子 (文字)
10 (X'0A')	4	PMARL_TXTL	テキストの長さ (符号なし)
14 (X'0E')	4	PMARL_TXTO	テキストへのオフセット (アドレス)
18 (X'12')	4	PMARL_BDRL	バインダー指標の長さ (符号なし)
22 (X'16')	4	PMARL_BDRO	バインダー指標へのオフセット (アドレス)
26 (X'1A')	4	PMARL_RDTL	PRDT の長さ (符号なし)
30 (X'1E')	4	PMARL_RDTO	PRDT へのオフセット (アドレス)

DESERV 出口

34 (X'22')	4	PMARL_RATL	PRAT の長さ (符号なし)
38 (X'26')	4	PMARL_RATIO	PRAT へのオフセット (アドレス)
42 (X'2A')	4	PMARL_NVSPGS	プログラム・オブジェクトを含む仮想ストレージ・ページの数、PM2 レベル PO の場合。
42 (X'2A')	4	PMARL_LMDL	LSLoader データの長さ (符号なし)、PM1 レベル PO の場合。
46 (X'2E')	4	PMARL_LMDO	LSLoader データへのオフセット (アドレス)
50 (X'32')	2	PMARL_NSEG	ロード可能セグメントの数
52 (X'34')	2	PMARL_NGAS	Gas テーブルの項目数
54 (X'36')	4	PMARL_1STOR	最初のロード可能セグメントに必要な仮想ストレージ、PMARL_NSEG > 1 のときに有効。
58 (X'3A')	4	PMARL_2STOR	2 番目のロード可能セグメントに必要な仮想ストレージ、PMARL_NSEG > 1 のときに有効。
62 (X'3E')	4	PMARL_2TXTO	gas を含んでいる 2 番目のセグメントへのオフセット、PMARL_NSEG > 1 のときに有効。

終了情報専用フィールド

66 (X'42')	16	PMARL_TRACE	AUDIT トレース・データ
66 (X'42')	4	PMARL_DATE	日付を保管しました。
70 (X'46')	4	PMARL_TIME	時刻を保管しました。
74 (X'4A')	8	PMARL_USER	ユーザーまたはジョブ識別
82 (X'52')	可変	PMARL_END	LSLoader セクションの終了 (文字)

表 73. ロード・モジュールに固有の属性 (PDS のみ) : PMAR_LFMT=OFF の場合、このセクションが PMAR 基本セクションの後に続きます。

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	23	PMARR	ロード・モジュール (PDS) 属性セクション (構造)
00 (X'00')	2	PMARR_SLEN	セクションの長さ (符号なし)
2 (X'02')	21	PMARR_DATA	セクション・データ (文字)
2 (X'02')	8	PMARR_TTRS	TTR フィールド (文字)
2 (X'02')	3	PMARR_TTRT	テキストの最初のブロックの TTR (文字)
5 (X'05')	1	PMARR_ZERO	ゼロ (文字)
6 (X'06')	3	PMARR_TTRN	注釈リストまたは分散変換テーブルの TTR。分散ロード形式またはオーバーレイ構造のモジュールにのみ使用されます。(文字)

表 73. ロード・モジュールに固有の属性 (PDS のみ) (続き): PMAR_LFMT=OFF の場合、このセクションが PMAR 基本セクションの後に続きます。

オフセット	長さ、または ビット・パターン	名前	説明
9 (X'09')	1	PMARR_NL	分散形式のモジュールおよびオーバーレイ構造のモジュールの注釈リストの項目数、あるいはゼロ。(アドレス)
10 (X'0A')	2	PMARR_FTBL	テキストの最初のブロックの長さ (符号付き)
12 (X'0C')	3	PMARR_ORG	0 の場合、ロード・モジュールの起点 (符号なし)
12 (X'0C')	2	-	予約済み
14 (X'0E')	1	PMARR_RLDS	最初のテキスト・レコードに続く RLD/CTL レコードの数。
15 (X'F')	8	PMARR_SCAT	分散ロード情報 (文字)
15 (X'F')	2	PMARR_SLSZ	分散リストの長さ (符号なし)
17 (X'11')	2	PMARR_TTSZ	変換テーブルの長さ (符号なし)
19 (X'13')	2	PMARR_ESDT	最初のテキスト・ブロックの ESDID (文字)
21 (X'15')	2	PMARR_ESDC	EP 制御セクションの ESDID (文字)
23 (X'17')	可変	PMARR_END	ロード・モジュール属性の終了 (文字)

表 74. 不定形式の別名: PUT 機能への入力としてのみ使用。

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	7	PMARA	PMAR 別名項目セクション (構造)
00 (X'00')	2	PMARA_LEN	セクションの長さ (符号なし)
2 (X'02')	5	PMARA_DATA	セクション・データ (文字)
2 (X'02')	4	PMARA_EPA	エントリー・ポイント・オフセット (符号なし)
6 (X'06')	1	PMARA_ATR	属性バイト (文字)
6 (X'06')	1	PMARA_ATR1	最初の属性バイト (ビット・ストリング)
6 (X'06')	1	PMARA_FT2	フラグ・バイトの代替名。このフラグは、マクロ IHAPDS によって宣言された PDS2FTB2 内の対応するフラグと同じオフセットになければなりません。(ビット・ストリング)
	xxxx	-	予約済み
 11..	PMARA_AMD	別名項目アドレッシング・モード。B'00' の場合、AMODE は 24 です。B'10' の場合、AMODE は 31 です。B'11' の場合、AMODE は ANY です。
xx	-	予約済み
7 (X'07')	可変	PMARA_END	別名項目セクションの終了 (文字)

PUT 機能に関連したパラメーター

DESERV 出口が DESERV PUT 機能呼び出しのために制御を取得した場合、DESX_DESP_PTR は DESERV パラメーター・リストを指します。DESP フィールドが DESP_FUNC=X'04' (DESP_FUNC_PUT) の場合、これは PUT 機能のパラメーター・リストを示します。表 75 は、DESERV PUT 呼び出しに関連した DESP のフィールドを示します。

表 75. DESERV PUT DESP フィールド

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	104	DESP	DE サービス・パラメーター・リスト (構造)
00 (X'00')	16	DESP_HEADER	標準ヘッダー (文字)
00 (X'00')	8	DESP_ID	目印 'IGWDESP' (文字)
08 (X'08')	4	DESP_LEN	DESP の長さ (符号付き)
	4	DESP_LEN_IV	DESP_LEN に使用される定数
12 (X'0C')	1	DESP_LEV	制御ブロック・レベル (文字)
	4	DESP_LEV_IV	DESP_LEV に使用される定数
13 (X'0D')	3	-	予約済み
16 (X'10')	1	DESP_FUNC	機能タイプ (GET=X'01'、 PUT=X'04'、 DELETE=X'07'、 RENAME=X'08'、 UPDATE=X'09') (符号なし)
	X'07'	DESP_FUNC_DELETE	DESP_FUNC に使用される定数
	X'08'	DESP_FUNC_RENAME	DESP_FUNC に使用される定数
	X'09'	DESP_FUNC_UPDATE	DESP_FUNC に使用される定数
	X'04'	DESP_FUNC_PUT	DESP_FUNC に使用される定数
	X'01'	DESP_FUNC_GET	DESP_FUNC に使用される定数
	X'00'	DESP_FUNC_OMITTED	DESP_FUNC に使用される定数
17 (X'11')	3	-	予約済み
20 (X'14')	4	-	予約済み
24 (X'18')	12	DESP_DATA	機能データ (文字)
24 (X'18')	2	-	予約済み
26 (X'1A')	1	-	予約済み
27 (X'1B')	1	-	予約済み
28 (X'1C')	1	DESP_LIBTYPE	入力 DCB であるか DEB であるかを示し ます。(符号なし)
	X'02'	DESP_LIBTYPE_DCB	DCB 入力
	X'01'	DESP_LIBTYPE_DEB	DEB 入力
	X'00'	DESP_LIBTYPE_OMITTED	省略
29 (X'1D')	1	-	予約済み
30 (X'1E')	1	-	予約済み
31 (X'1F')	1	-	予約済み

表 75. DESERV PUT DESP フィールド (続き)

オフセット	長さ、または ビット・パターン	名前	説明
32 (X'20')	1	DESP_OPTION	REPLACE オプション (REPLACE=X'01'、 NOREPLACE=X'00') (符号なし)
	X'02'	DESP_OPTION_REPLACE_ALIAS	DESP_OPTION で使用される定数
	X'01'	DESP_OPTION_REPLACE	DESP_OPTION で使用される定数
	X'00'	DESP_OPTION_NOREPLACE	DESP_OPTION で使用される定数
33 (X'21')	1	-	予約済み
34 (X'22')	1	-	予約済み
35 (X'23')	1	-	予約済み
36 (X'24')	4	DESP_DCB_PTR	DCB アドレス
40 (X'28')	4	DESP_DEB_PTR	DEB アドレス
44 (X'2C')	4	-	予約済み
48 (X'30')	4	-	予約済み
52 (X'34')	4	-	予約済み
56 (X'38')	4	-	予約済み
60 (X'3C')	4	-	予約済み
64 (X'40')	4	-	予約済み
68 (X'44')	4	-	予約済み
72 (X'48')	4	DESP_MEM_DATA_PTR	MEM_DATA_ADDRESS
76 (X'4C')	4	DESP_MEM_DATA2	MEM_DATA 項目カウント (符号なし)
80 (X'50')	4	-	予約済み
84 (X'54')	4	-	予約済み
88 (X'58')	4	-	予約済み
92 (X'5C')	4	-	予約済み
96 (X'60')	4	-	予約済み
100 (X'64')	4	-	予約済み

PUT の戻りコードと理由コード

310 ページの図 34 は、PUT 機能の戻りコードと理由コードを説明しています。

DESERV 出口

戻りコード	説明
DESRC_SUCC	X'00' 正常な処理
	理由コード 説明 DESR_SUCC X'00' 正常な処理
DESRC_INFO 4	説明 完全に成功したわけではありません。
DESRC_WARN 8	結果に疑問があります。
	理由コード 説明 DESR_S DST_ALREADY_EXISTS X'44B' EXIT が存在し、NOREPLACE を指定した DESERV FUNC EXIT が発行されました。現行の出口は置き換えられません。 DESR_S DST_COMP_SWAP_FAILED X'451' EXIT_OPTION=DELETE が指定した DST アドレスは現行ではありません。比較交換は失敗しました。
DESRC_PARM 12	脱落/無効パラメーター
	理由コード 説明 DESR_S EXIT_OPTION_INVALID X'44F' 指定された EXIT_OPTION はサポートされません。 DESR_S EXIT_SCOPT_INVALID X'45' 指定された EXIT_SCOPE はサポートされません。 DESR_S EXIT_DST_PTR_ZERO X'44C' DESERV FUNC=EXIT の呼び出し側は EXIT_DST パラメーターを介してゼロの DST アドレスを提供しました。 DESR_S INVAL_DST_HEADER X'44D' DST ヘッダーが正しくありません。 DESR_S INVAL_PREVDST_HEADER X'453' DST ヘッダーは、DESP_PREV_DSTPTR_PTR が指している DST では無効です。これは EXIT_OPTION=DELETE のために検査されます。 DESR_S PREV_DSTPTR_PTR_ZERO X'452' 前の DST へのポインターがゼロです。これは EXIT_OPTION=DELETE のために検査されます。 DESR_S INVALID_PARM_LIST_HEADER X'411' DESP の ID、長さ、またはレベルが無効です。 DESR_S UNSUPPORTED_FUNC X'424' FUNC 値が誤りです。 DESR_S DEB_REQUIRES_AUTH X'423' DEB を渡すには、呼び出し側は、監視プログラム状態であるか、特権キーを持っていないければなりません。 DESR_S INVALID_DCB_PTR X'422' DCB のアドレスが 0 です。 DESR_S DCB_NOT_OPEN X'421' 渡された DCB がオープンされていません。 DESR_S INVALID_DEB_PTR X'41E' DEB のアドレスが 0 であるか、または DEB は入力でしたが、DEB によって指示された DCB が DEB を指していませんでした。
DESRC_CALR 16	呼び出し側に問題があります。
	理由コード 説明 DESR_S DEBCHK_FAILED X'41D' DEBCHK マクロが失敗しました。DCB または DEB が無効でした。 DESR_S AUTH_ERROR X'449' 呼び出し側が監視プログラム状態またはシステム・キーではありません。
DESRC_ENVR 20	リソースが利用不可です。
DESRC_IOER 24	入出力エラー
DESRC_MEDE 28	メディア・エラー
DESRC_DSLE 32	データ・セット論理エラー
DESRC_SEVE 36	重大エラー
	理由コード 説明 DESR_S UNKNOWN X'447' 不明の理由 (つまり、プログラム・チェック) で入ったときに、DESERV リカバリー・ルーチンによって発行されました。 DESR_S ADD_STACK_FAILED X'437' IGWFESTK 要求からのゼロ以外の戻りコード。 DESR_S SETLOOK_ERR X'407' SETLOCK からの不良戻りコード DESR_S EXTRACT_ERROR X'406' IGWFTOKM EXTRACT が失敗しました。 DESR_S SET_ERROR X'405' IGWFTOKM SET が失敗しました。

DA6S3027

図 34. PUT の戻りコードと理由コード

DESD (DESERV メンバー・データ記述子) は、DESERV PUT 機能への入力です。
DESD は、DESP フィールド DESP_MEM_DATA2 で定義された項目数からなる配

列です。DESP_MEM_DATA_PTR は DESD を指し、IGWDES マクロの DESD CSECT がそれをマップします。DESD 構造を表 76 に示します。

表 76. DESD パラメーター・リスト

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	16	DESD	メンバー・データ記述子 (構造)
00 (X'00')	16	DESD_ENTRY	項目記述子 (文字)
00 (X'00')	1	DESD_FLAG	フラグ (ビット・ストリング)
	1... ..	DESD_FLAG_ALIAS	別名項目
1 (X'01')	1	DESD_CODE	処理コード (エラー=X'02', 未処理=X'01', 成功=X'00')(符号なし)
	X'02'	DESD_CODE_ERROR	DESD_CODE で使用される定数
	X'01'	DESD_CODE_NOGO	DESD_CODE で使用される定数
	X'00'	DESD_CODE_SUCC	DESD_CODE で使用される定数
2 (X'02')	2	DESD_ERRCODE	エラー・コード (エラーの場合、DESERV 理由コードの下位ハーフワード) (符号なし)
4 (X'04')	2	-	予約済み
6 (X'06')	2	DESD_DATA_LEN	データ域の長さ (符号なし)
08 (X'08')	4	DESD_DATA_PTR	データのアドレス (アドレス)
12 (X'0C')	4	DESD_NAME_PTR	可変長名のアドレス (アドレス)

上で定義されているように、DESD_NAME_PTR は DESN 構造を指します。DESD_DATA_PTR は、PDSE ディレクトリーに保管されるプログラム・オブジェクトのディレクトリー項目を指します。ディレクトリー項目のフォーマットは、DESD 項目が基本名を表すか、別名を表すかによって異なります。基本名の場合、DESD_DATA_PTR は IEWPMAR によってマップされた CSECT PMAR を指します。別名の場合、DESD_DATA_PTR はマクロ IEWPMAR によってマップされた CSECT PMARA を指します。307 ページの表 74 は PMARA 構造を示しています。DESD_FLAG_ALIAS は、DESD 内の項目を基本名または別名として識別します。

DESERV 出口は、DESERV の呼び出し側に戻す必要がある現行戻りコードおよび理由コードに渡されます。出口 (前処理または後処理) は、レジスター 15 に戻りコード 4 を戻すと、DESERV は異なる戻りコードと理由コードを DESERV 呼び出し側に戻すことがあります。出口がレジスター 15 に戻りコード 4 を戻して DESERV に制御を戻した場合、DESX_RETURN_CODE および DESX_REASON_CODE の値 (多分、変更されている) が、呼び出し側 DESERV に戻されます。前処理出口がレジスター 15 に戻りコード 0 を戻した場合、処理は継続されます。一方、後処理出口がレジスター 15 に戻りコード 0 を戻した場合は、DESX_RETURN_CODE および DESX_REASON_CODE の元の値 (出口への入力として渡された値) が DESERV の呼び出し側に戻されます。

出口処理には、DESERV パラメーター・リスト (DESP) の調査、あるいは他のインターフェース構造やバッファの調査または変更を組み込むことができます。DESP は、DESERV 出口によって変更されません。出口には呼び出し側キー DESP が渡されますが、DESERV はすでに DESP のキー 5 コピーを作成しており、それが

DESERV 処理に使用されます。したがって、呼び出し側キー DESP に対する変更は、DESERV 処理には影響を与えません。DESERV インターフェース構造は、マクロ IGWDES によって定義されます。マクロ IGWSMDE および IEWPMAR は、ディレクトリー項目のフォーマットを定義します。

DELETE 機能に関連したパラメーター

DESERV が DESERV DELETE 機能呼び出しの制御を取得した場合、DESX_DESP_PTR は DESERV パラメーター・リストを指します。DESP フィールドが DESP_FUNC=X'07' (DESP_FUNC_DELETE) の場合、これは DELETE 機能のパラメーター・リストを示します。表 77 は、DESERV DELETE 呼び出しに関連した DESP フィールドを示しています。

表 77. DESERV DELETE DESP フィールド

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	104	DESP	DE サービス・パラメーター・リスト (構造)
00 (X'00')	16	DESP_HEADER	標準ヘッダー (文字)
00 (X'00')	8	DESP_ID	目印 IGWDESP (文字)
08 (X'08')	4	DESP_LEN	DESP の長さ (符号付き)
	X'04'	DESP_LEN_IV	DESP_LEN で使用される定数
12 (X'0C')	1	DESP_LEV	制御ブロック・レベル (文字)
	X'04'	DESP_LEV_IV	DESP_LEV で使用される定数
13 (X'0D')	3	-	予約済み
16 (X'10')	1	DESP_FUNC	機能タイプ (GET=X'01'、 PUT=X'04'、 DELETE=X'07'、 RENAME=X'08'、 UPDATE=X'09')
	X'09'	DESP_FUNC_UPDATE	DESP_FUNC で使用される定数
	X'08'	DESP_FUNC_RENAME	DESP_FUNC で使用される定数
	X'07'	DESP_FUNC_DELETE	DESP_FUNC で使用される定数
	X'04'	DESP_FUNC_PUT	DESP_FUNC で使用される定数
	X'01'	DESP_FUNC_GET	DESP_FUNC で使用される定数
	X'00'	DESP_FUNC_OMITTED	DESP_FUNC で使用される定数
	-	-	-
17 (X'11')	3	-	予約済み
20 (X'14')	4	-	予約済み
24 (X'18')	12	DESP_DATA	機能データ (文字)
24 (X'18')	2	-	予約済み
26 (X'1A')	1	-	予約済み
27 (X'1B')	1	-	予約済み
28 (X'1C')	1	DESP_LIBTYPE	入力が DCB であるか DEB であるかを示し ます。(DCB 入力 X'02'、DEB 入力 X'01') (符 号なし)
	X'02'	DESP_LIBTYPE_DCB	DESP_LIBTYPE で使用される定数
	X'01'	DESP_LIBTYPE_DEB	DESP_LIBTYPE で使用される定数
	X'00'	DESP_LIBTYPE_OMITTED	DESP_LIBTYPE で使用される定数

表 77. DESERV DELETE DESP フィールド (続き)

オフセット	長さ、または ビット・パターン	名前	説明
29 (X'1D')	1	-	予約済み
30 (X'1E')	1	-	予約済み
31 (X'1F')	1	-	予約済み
32 (X'20')	1	-	予約済み
33 (X'21')	1	-	予約済み
34 (X'22')	1	-	予約済み
35 (X'23')	1	-	予約済み
36 (X'24')	4	DESP_DCB_PTR	DCB アドレス
40 (X'28')	4	DESP_DEB_PTR	DEB アドレス
44 (X'2C')	4	-	予約済み
48 (X'30')	4	-	予約済み
52 (X'34')	4	-	予約済み
56 (X'38')	4	-	予約済み
60 (X'3C')	4	-	予約済み
64 (X'40')	4	-	予約済み
68 (X'44')	4	-	予約済み
72 (X'48')	4	-	予約済み
76 (X'4C')	4	-	予約済み
80 (X'50')	4	DESP_NAME_LIST	削除する名前のリスト (DESL) (アドレス)
84 (X'54')	4	DESP_NAME_LIST2	DESL 名前リストの項目数 (符号なし)
88 (X'58')	4	-	予約済み
92 (X'5C')	4	-	予約済み
96 (X'60')	4	-	予約済み
100 (X'64')	4	-	予約済み

DESL は、削除する名前を指します。DESP_NAME_LIST_PTR は DESL を指し、IGWDES マクロの DESL DSECT がそれをマップします。DESL は、DESP フィールド DESP_NAME_LIST2 が定義する項目数からなる配列です。DESL 構造は、297 ページの表 63 に示されています。

RENAME 機能に関連したパラメーター

DESERV 出口が DESERV RENAME 機能呼び出しのために制御を取得した場合、DESX_DESP_PTR は DESERV パラメーター・リストを指します。DESP フィールドが DESP_FUNC=X'08' (DESP_FUNC_RENAME) の場合、これは RENAME 機能のパラメーター・リストを示します。表 78 は、DESERV RENAME 呼び出しに関連した DESP のフィールドを示します。

表 78. DESERV RENAME DESP フィールド

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	104	DESP	DE サービス・パラメーター・リスト (構造)

DESERV 出口

表 78. DESERV RENAME DESP フィールド (続き)

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	16	DESP_HEADER	標準ヘッダー (文字)
00 (X'00')	8	DESP_ID	目印 IGWDESP (文字)
08 (X'08')	4	DESP_LEN	DESP の長さ (符号付き)
	X'04'	DESP_LEN_IV	DESP_LEN で使用される定数
12 (X'0C')	1	DESP_LEV	制御ブロック・レベル (文字)
	X'04'	DESP_LEV_IV	DESP_LEV で使用される定数
13 (X'0D')	3	-	予約済み
16 (X'10')	1	DESP_FUNC	機能タイプ (GET=X'01'、 PUT=X'04'、 DELETE=X'07'、 RENAME=X'08'、 UPDATE=X'09')
	X'09'	DESP_FUNC_UPDATE	DESP_FUNC で使用される定数
	X'08'	DESP_FUNC_RENAME	DESP_FUNC で使用される定数
	X'07'	DESP_FUNC_DELETE	DESP_FUNC で使用される定数
	X'04'	DESP_FUNC_PUT	DESP_FUNC で使用される定数
	X'01'	DESP_FUNC_GET	DESP_FUNC で使用される定数
	X'00'	DESP_FUNC_OMITTED	DESP_FUNC で使用される定数
17 (X'11')	3	-	予約済み
20 (X'14')	4	-	予約済み
24 (X'18')	12	DESP_DATA	機能データ (文字)
24 (X'18')	2	-	予約済み
26 (X'1A')	1	-	予約済み
27 (X'1B')	1	-	予約済み
28 (X'1C')	1	DESP_LIBTYPE	入力が DCB であるか DEB であることを示します。 (DCB,DEB) (DCB 入力 X'02'、 DEB 入力 X'01') (符号なし)
	X'02'	DESP_LIBTYPE_DCB	DESP_LIBTYPE で使用される定数
	X'01'	DESP_LIBTYPE_DEB	DESP_LIBTYPE で使用される定数
	X'00'	DESP_LIBTYPE_OMITTED	DESP_LIBTYPE で使用される定数
29 (X'1D')	1	-	予約済み
30 (X'1E')	1	-	予約済み
31 (X'1F')	1	-	予約済み
32 (X'20')	1	-	予約済み
33 (X'21')	1	-	予約済み
34 (X'22')	1	-	予約済み
35 (X'23')	1	-	予約済み
36 (X'24')	4	DESP_DCB_PTR	DCB アドレス
40 (X'28')	4	DESP_DEB_PTR	DEB アドレス
44 (X'2C')	4	-	予約済み
48 (X'30')	4	-	予約済み
52 (X'34')	4	-	予約済み
56 (X'38')	4	-	予約済み
60 (X'3C')	4	-	予約済み
64 (X'40')	4	-	予約済み
68 (X'44')	4	-	予約済み
72 (X'48')	4	-	予約済み

表 78. DESERV RENAME DESP フィールド (続き)

オフセット	長さ、または ビット・パターン	名前	説明
76 (X'4C')	4	-	予約済み
80 (X'50')	4	DESP_NAME_LIST	名前変更操作の対の名前のリスト (DESL) (アドレス)
84 (X'54')	4	DESP_NAME_LIST2	DESL 名前リストの項目数 (符号なし)
88 (X'58')	4	-	予約済み
92 (X'5C')	4	-	予約済み
96 (X'60')	4	-	予約済み
100 (X'64')	4	-	予約済み

DESL は、名前変更する名前と新しい名前を指します。DESP_NAME_LIST_PTR は DESL を指し、IGWDES マクロの DESL DSECT がそれをマップします。DESL は、DESP フィールド DESP_NAME_LIST2 が定義する項目数からなる配列です。DESL 構造は、297 ページの表 63 に示されています。

UPDATE 機能に関連したパラメーター

DESERV が DESERV UPDATE 機能呼び出しの制御を取得した場合、DESX_DESP_PTR は DESERV パラメーター・リストを指します。DESP フィールドが DESP_FUNC=X'09' (DESP_FUNC_UPDATE) の場合、これは UPDATE 機能のパラメーター・リストを示しています。表 79 は、DESERV UPDATE 呼び出しに関連した DESP フィールドを示しています。

表 79. DESERV UPDATE DESP フィールド

オフセット	長さ、または ビット・パターン	名前	説明
00 (X'00')	104	DESP	DE サービス・パラメーター・リスト (構造)
00 (X'00')	16	DESP_HEADER	標準ヘッダー (文字)
00 (X'00')	8	DESP_ID	目印 'IGWDESP' (文字)
08 (X'08')	4	DESP_LEN	DESP の長さ (符号付き)
	X'04'	DESP_LEN_IV	DESP_LEN で使用される定数
12 (X'0C')	1	DESP_LEV	制御ブロック・レベル (文字)
	X'04'	DESP_LEV_IV	DESP_LEV で使用される定数
13 (X'0D')	3	-	予約済み
16 (X'10')	1	DESP_FUNC	機能タイプ (GET=X'01'、PUT=X'04'、DELETE=X'07'、RENAME=X'08'、UPDATE=X'09')
	X'09'	DESP_FUNC_UPDATE	DESP_FUNC で使用される定数
	X'08'	DESP_FUNC_RENAME	DESP_FUNC で使用される定数
	X'07'	DESP_FUNC_DELETE	DESP_FUNC で使用される定数
	X'04'	DESP_FUNC_PUT	DESP_FUNC で使用される定数
	X'01'	DESP_FUNC_GET	DESP_FUNC で使用される定数
	X'00'	DESP_FUNC_OMITTED	DESP_FUNC で使用される定数
	-	-	-
17 (X'11')	3	-	予約済み

DESERV 出口

表 79. DESERV UPDATE DESP フィールド (続き)

オフセット	長さ、または ビット・パターン	名前	説明
20 (X'14')	4	-	予約済み
24 (X'18')	12	DESP_DATA	機能データ (文字)
24 (X'18')	2	-	予約済み
26 (X'1A')	1	-	予約済み
27 (X'1B')	1	-	予約済み
28 (X'1C')	1	DESP_LIBTYPE	機能サブタイプ (DCB,DEB) (DCB 入力 X'02'、 DEB 入力 X'01') (符号なし)
	X'02'	DESP_LIBTYPE_DCB	DESP_LIBTYPE で使用される定数
	X'01'	DESP_LIBTYPE_DEB	DESP_LIBTYPE で使用される定数
	X'00'	DESP_LIBTYPE_OMITTED	DESP_LIBTYPE で使用される定数
29 (X'1D')	1	-	予約済み
30 (X'1E')	1	-	予約済み
31 (X'1F')	1	-	予約済み
32 (X'20')	1	-	予約済み
33 (X'21')	1	-	予約済み
34 (X'22')	1	-	予約済み
35 (X'23')	1	-	予約済み
36 (X'24')	4	DESP_DCB_PTR	DCB アドレス
40 (X'28')	4	DESP_DEB_PTR	DEB アドレス
44 (X'2C')	4	-	予約済み
48 (X'30')	4	-	予約済み
52 (X'34')	4	-	予約済み
56 (X'38')	4	-	予約済み
60 (X'3C')	4	-	予約済み
64 (X'40')	4	-	予約済み
68 (X'44')	4	-	予約済み
72 (X'48')	4	-	予約済み
76 (X'4C')	4	-	予約済み
80 (X'50')	4	DESP_NAME_LIST	ディレクトリー情報を更新するための組み込み PMAR (および PMARL) を含む SMDE のリスト(DES�) (アドレス)
84 (X'54')	4	DESP_NAME_LIST2	DES� 名前リストの項目数 (符号なし)
88 (X'58')	4	-	予約済み
92 (X'5C')	4	-	予約済み
96 (X'60')	4	-	予約済み
100 (X'64')	4	-	予約済み

DES� は、更新する SMDE を指します。DESP_NAME_LIST_PTR は DES� を指し、IGWDES マクロの DES� DSECT がそれをマップします。DES� は、DESP フ

フィールド DESP_NAME_LIST2 が定義する項目数からなる配列です。DESL 構造は、297 ページの表 63 に示されています。

出口ルーチンの入り口環境

割り込み:	使用可能
状態およびキー:	監視プログラム状態およびキー 0
ASC モード:	P=H=S
AMODE、RMODE:	AMODE=31、RMODE=ANY
ロック:	ロックを保持しない
レジスター:	
	0 予測不能
	1 IGWDES 内の DSECT DESX によってマップされた DESERV EXIT パラメーター・リストのアドレス。
	2-12 予測不能
	13 18 ワード保管域 (キー 0)。
	14 戻りアドレス
	15 出口ルーチンのエントリー・ポイント・アドレス

出口は TASK モードに入り、DESERV リカバリー環境は ESTAE です。したがって、出口ルーチンは、必要であれば SVC を発行できます。

出口ルーチンの出口環境

割り込み:	使用可能
状態およびキー:	監視プログラム状態およびキー 0
ASC モード:	P=H=S
AMODE、RMODE:	AMODE=31、RMODE=ANY
ロック:	ロックを保持しない
レジスター:	
	特に指示がない限り、レジスターはすべて、入ったときの内容に復元する必要があります。
	15 戻りコード
	前処理出口の場合:
	R15 = 0 この DESERV 呼び出しの処理を継続します。
	R15 = 4 この DESERV 呼び出しの処理を中断します。制御は即時に DESERV の呼び出し側に戻され、出口によって戻りコードと理由コードが DESX フィールド

の DESX_RETURN_CODE と DESX_REASON_CODE に設定されます。

後処理出口の場合:

R15 = 0 この DESERV 呼び出しの処理を継続します (つまり、DESERV の呼び出し側に戻ります)。

R15 = 4 制御は DESERV の呼び出し側に戻され、出口によって戻りコードと理由コードが DESX フィールドの DESX_RETURN_CODE と DESX_REASON_CODE に設定されます。

DESX_RETURN_CODE と DESX_REASON_CODE の値は、マクロ IGWDES に記述されています。理由コードの構造は、最初の 2 バイトは、システム・コンポーネント ID とモジュール ID (つまり、システム診断情報) です。下位 2 バイトには、マクロ IGWDES に示されている実際の理由コードが入っています。GET、RENAME、DELETE、および UPDATE の追加の戻りコードおよび理由コードについては、「z/OS DFSMS Macro Instructions for Data Sets」を参照してください。

制約事項: 出口からこれ以外の戻りコードが戻された場合は、DESERV コードによって SVC ダンプが取られます。

DESERV 出口に入るときのレジスター

DESERV 出口が制御を取得した時点での汎用レジスターの内容は、次のとおりです。

レジスター	内容
0	適用外
1	DESX のアドレス
2-12	適用外
13	レジスター保管域のアドレス
14	戻りアドレス
15	DESERV 出口エンタリー・ポイントのアドレス

DESERV 出口から戻るときのレジスター

制御を DESERV に戻した時点で、レジスターの内容は、以下のようにセットアップされていなければなりません。

レジスター	内容
0-14	入ったときの内容に復元されます。
15	戻りコード

DESERV 出口の戻りコードと理由コード

戻りコード	説明
00 (X'00')	DESERV 機能を継続します。
04 (X'04')	即時に DESERV の呼び出し側に戻り、それぞれ DESX_RETURN_CODE と DESX_REASON_CODE によって定義された戻りコードと理由コードを戻します。

DESERV FUNC=EXIT 戻りコードと理由コード

戻りコードと理由コードのフォーマットは、次のとおりです。

オフセット/長さ	説明
00 (X'00') 1 バイト	SMS コンポーネント・コード (X'27') は、共通アダプターを示します (DESERV はその一部)
01 (X'01') 1 バイト	モジュール ID - 問題診断に使用されます。
02 (X'02') 2 バイト	理由コード - エラーを識別します。DESERV 理由コードをテストするプログラムは、最後の 2 バイトのみを調べる必要があります。コンポーネント ID とモジュール ID はテストしてはなりません。これらは、診断の目的でのみ報告されます。

以下は、DESERV FUNC=EXIT が戻す理由コードの下位 2 バイト値です (戻りコード別にソート)。

DESERV 出口

戻りコード	説明
DESRC_SUCC 0	成功。 理由コード DESRS_SUCC X'00' 成功
DESRC_INFO 4	完全に成功したわけではありません。 理由コード DESRS_NAME_NOT_DEFINED X'400' 置換される名前が前に存在していませんでした。
DESRC_WARN 8	結果に疑問があります。
DESRC_PARM 12	脱落/無効パラメーター 理由コード DESRS_PDS_NOT_SUPPORTED X'415' この機能は PDSE データ・セットを必要とします。 DESRS_INVALID_PUT_OPTION X'429' PUT 機能では、OPTION フィールドを指定する必要があります。 DESRS_INVALID_MEM_DATA_PTR X'428' MEM_DATA ブロックのアドレスが 0 です。 DESRS_INVALID_MEM_DATA_CNT X'427' MEM_DATA ブロック内の項目数が 0 です。 DESRS_DCB_NOT_OPEN_OUTPUT X'41F' 機能 PUT を使用するには、DCB は出力用にオープンされていなければなりません。 DESRS_INVALID_PARM_LIST_HEADER X'411' DESP の ID、長さ、またはレベルが無効です。 DESRS_UNSUPPORTED_FUNC X'424' FUNC 値が誤りです。 DESRS_DEB_REQUIRES_AUTH X'423' DEB を渡すには、呼び出し側は、監視プログラム状態であるか、特権キーを持っていなければなりません。 DESRS_INVALID_DCB_PTR X'422' DCB のアドレスが 0 です。 DESRS_DCB_NOT_OPEN X'421' 渡された DCB がオープンされていません。 DESRS_INVALID_DEB_PTR X'41E' DEB のアドレスが 0 であるか、または DEB は入力でしたが、DEB によって指示された DCB が DEB を指していませんでした。
DESRC_CALR 16	呼び出し側に問題があります。 理由コード DESRS_INVALID_CT X'414' 接続トークンが無効です。 DESRS_INVALID_MLT X'413' MLT が無効です。 DESRS_MORE_THAN_1_PRIMARY X'412' MEM_DATA は 1 つのメンバーのみを基本として指定する必要があります。 DESRS_INVALID_NAME_LENGTH X'425' 別名の長さが 0 または DESRS_INVALID_NAME_PREFIX X'410' 名前の最初の 8 バイトが X'FF' でした。 DESRS_DATA_LENGTH_ERROR X'43A' DESD データ長が無効です。データ長は 0 より大きく、108 バイトより小さくしなければなりません。 DESRS_NP_PRIMARY_NAME X'40F' MEM_DATA は 1 つのメンバーを基本として指定する必要があります。 DESRS_NAME_ALREADY_EXISTS X'40E' 名前の競合により PUT が失敗しました。 DESRS_NAME_IS_PRIMARY_NAME X'400' 置換される名前が前に存在していませんでした。 DESRS_PRI_NM_THIS_FILE X'43C' 指定された別名は基本名であり、オプションは基本名の削除を許可しませんでした。 DESRS_DEBCHK_FAILED X'3FD' このメンバーの別名は、基本名と同じです。 X'41D' DEBCHK マクロが失敗しました。DCB または DEB が無効でした。
DESRC_ENVR 20	リソースが利用不可です。
DESRC_IOER 24	入出力エラー
DESRC_MEDE 28	メディア・エラー
DESRC_DSLE 32	データ・セット論理エラー
DESRC_SEVE 36	重大エラー 理由コード DESRS_UNKNOWN X'447' 不明の理由 (つまり、プログラム・チェック) で入ったときに、DESERV リカバリー・ルーチンによって発行されました。 DESRS_ADD_STACK_FAILED X'437' IGWFESTK 要求からのゼロ以外の戻りコード。 DESRS_SETLOOK_ERR X'407' SETLOCK からの不良戻りコード

DA6S3026

図 35. 出口 DESERV 機能の戻りコードと理由コード

追加の戻りコードと理由コード: GET、RENAME、DELETE、および UPDATE の戻りコードと理由コードについては、「z/OS DFSMS Macro Instructions for Data Sets」を参照してください。

DESERV 出口の例

以下のプログラム・セグメントは、タスク・モードの DESERV 出口を確立および削除し、これによって前のタスクを復元します。サンプルは汎用で、グローバル出口またはタスク・レベル出口のいずれにも十分適用できますが、ここではタスク・レベル・サポートを示しています。グローバル出口を確立する場合は、共通ストレージ内に DST ストレージを取得してください。


```

*
*   HERE THE APPLICATION WOULD DO SOMETHING TO
*   CAUSE A DESERV CALL TO BE DONE. AN EXAMPLE OF THIS TYPE OF THING
*   WOULD BE ATTACHING THE BINDER AS A SUBTASK (THE BINDER USES
*   DESERV.)
*
*   .
*   .
*   .
*
* DELETE THE TASK LEVEL EXIT
*
*     DESERV FUNC=EXIT,
*           EXIT_SCOPE=TASK,
*           EXIT_OPTION=DELETE,
*           EXIT_DST=MY_DST,
*           EXIT_PREV_DSTPTR=MY_PREV_DSTPTR,
*           MF=S
*
*   .
*   .
*   .
*
* RESTORE REGISTERS AND RETURN
*   L      13,SAVE+4
*   LM     14,12,12(13)
*   SR     15,15      SET RETURN CODE
*   BR     14          RETURN TO CALLER
*
* THE FOLLOWING IS A BLOCK USED BY THIS APPLICATION. THIS BLOCK CAN
* BE USED BY THE EXIT. THE EXIT ALSO (SMARTLY) FINDS THE
* PREV_DST_PTR FIELD IN MY_BLOCK WHEN IT GIVES CONTROL TO THE
* PREVIOUSLY ESTABLISHED TASK LEVEL EXIT.
MY_BLOCK      DS  0D          MY APPLICATION BLOCK
MY_APPLICATION_STUFF DS  CL8      MY APPLICATION STUFF
MY_DST        DS  CL(DST_LEN_IV) MY DST IMBEDDED IN MY BLOCK
MY_PREV_DSTPTR DS  F           ADDRESS OF PREVIOUS TASK LEVEL
*
*   EXTRN  DESEXIT
ADDR_DESEXIT  DC  A(DESEXIT)    ADDRESS OF MY TASK LEVEL EXIT
DST_ID_CONST  DC  CL8'IGWDST '  DST EYECATCHER
*
*   DST
CURRENT_DSTPTR DS  F           ADDRESS OF CURRENT TASK LEVEL
*
*   DST
SAVE          DS  18F          REGISTER SAVE AREA
*
*   END

```

図 36. タスク・レベル DESERV 出口の確立と削除 (2/2)

以下のプログラム・セグメントは、サンプル DESERV 出口を示しています。これは、前のコード・セグメントによって確立された出口です。このサンプルは、制御を前の出口に渡す方法を示しています。このサンプル出口は再入可能ではないので、サブタスクが 1 つだけであることが前提です。

```

DESEXIT CSECT
DESEXIT AMODE 31          Must be AMODE 31
DESEXIT RMODE ANY       Could be RMODE 24 if required
*
* entry code to save registers and establish base register.
  USING *,12
  STM  14,12,12(13)      SAVE REGISTERS
  LR   12,15             ESTABLISH BASE REGISTER
  LA   2,SAVE            ADDRESS REGISTER SAVE AREA
  ST   2,8(13)          FORWARD CHAIN SAVE AREA
  ST   13,SAVE+4        BACKWARD CHAIN SAVE AREA
  LR   13,2             ESTABLISH SAVE AREA
*
* Assume this exit is only interested in the output from GET functions.
* Therefore ignore entry for all other functions and only
* process the get function invocations where get processing is complete
*
  SLR  2,2              clear reg
  ST   2,EXIT_RC        initialize return code
  LR   2,1              get the DESX address in reg 2
  USING DESX,2          map the DESX
  L    3,DESX_DST_PTR   get address of DST
  LR   4,3              get address of DST
  LA   5,MY_DST-MY_BLOCK(,0) get offset to DST within MY_BLOCK
  SR   4,5              get the address of MY_BLOCK
  USING MY_BLOCK,4      map MY_BLOCK
*
* First give control to any previously established DESERV
* exits. If the value in PREV_DST_PTR is zero, then there was no
* previous DST (that is, no previous DESERV exit). PREV_DST_PTR was
* saved in MY_BLOCK by the SAMPLE CSECT that enabled this
* exit.
  SR   15,15            simulate previous exit's return code
  ICM  5,15,PREV_DST_PTR get previous DST address
  BZ   NOPREVDST        branch if zero, no previous DST
* There was a previous exit, to which you transfer control.
* First build a DESX then branch to the previous exit.
*
*
* Getmain dynamic storage for interfacing with the other exit
*
*
  GETMAIN RU,LV=DESX_LEN_IV,SP=230,KEY=0,LOC=(ANY)
  ST   1,MY_DESX_STG_PTR
  MVC  0(L'DESX,1),DESX copy the DESX that was input to this
*                               routine.
  ST   5,DESX_DST_PTR-DESX(,1) Set previous DST address in DESX
  L    15,DST_EXIT_PTR-DST(,5) get address of previous exit
  BALR 14,15            call the previous exit in 31 bit amode

```

図 37. サンプル DESERV 出口ルーチン (1/3)

```

*
* CONTROL INFORMATION
*
SAVE          DS 18F          REGISTER SAVE AREA
ZERO          DC F'0'        constant of zero
EXIT_RC       DC F'0'        Return code to pass back to DESERV
DESX_ID_CONST DC CL8'IGWDESX ' EYECATCHER FOR DESX
MY_DESX_STG_PTR DC F'0'      ADDRESS OF GETMAINED STORAGE FOR
*                                DESX TO PASS TO PREVIOUS EXIT.
MY_BLOCK      DSECT          MY APPLICATION BLOCK
MY_APPLICATION_STUFF DS CL8    MY APPLICATION STUFF
MY_DST        DS CL(DST_LEN_IV) MY DST IMBEDED IN MY BLOCK
PREV_DST_PTR  DS F           ADDRESS OF PREVIOUS TASK LEVEL EXIT
                                DESERV MAPPINGS
                                IGWDES
                                IGWSMDE
                                IEWPMAR
                                DIRECTORY ENTRY
                                PROGRAM MANAGEMENT ATTRIBUTE RECORD
*                                THE PMAR IS A SUB RECORD OF THE SMDE
                                END

```

図 37. サンプル *DESERV* 出口ルーチン (3/3)

第 11 章 システム・イメージ・ライブラリーの保守

この章では、システム・イメージ・ライブラリー SYS1.IMAGELIB を保守する方法について説明します。SYS1.IMAGELIB は、汎用文字セット (UCS)、用紙制御バッファ (FCB)、および DFSMSdfp がサポートする IBM プリンターのプリンター制御情報を含んでいる区分データ・セットです (PDSE はサポートされません)。プリンター制御情報は、プリンターのタイプに応じて、以下の形で組み込まれています。

- UCS イメージ (モジュールとも呼ばれます)
- UCS イメージ・テーブル
- FCB イメージ (モジュールとも呼ばれます)
- 制御モジュール

システムは UCS イメージとイメージ・テーブルを使用して、ユーザーが要求する UCS を、対応する印刷バンド/トレーンに関連付けます。システム・インストール時に、ほとんどの IBM 標準 UCS イメージが SYS1.IMAGELIB に組み込まれます。新規リリースの DFSMS をインストールすると、IBM 提供の名前を持っていたイメージを置き換えるので注意が必要です。イメージを変更または置換した場合は、そのコピーを別に保守することが必要になる場合があります。下の表は、IBM 1403、3203、および 3211 プリンターの標準文字セット・イメージを示しています。

プリンター	イメージ
1403 または 3203	AN、HN、PCAN、PCHN、PN、QN、QNC、RN、SN、TN、XN、YN
3211	A11、G11、H11、P11、T11

詳細なプリンター固有の情報や使用可能な印刷バンド/トレーンを調べるには、下の表にリストされている資料を参照してください。

資料名	内容
<i>IBM 2821 Control Unit Component Description</i>	1403 プリンターのユーザー設計チェーン/トレーンの作成に関する情報。
<i>IBM 3203 Printer Component Description and Operator's Guide</i>	3203 プリンターのユーザー設計トレーンの作成に関する情報。
<i>IBM 3211 Printer, 3216 Interchangeable Train Cartridge, and 3811 Printer Control Unit Component Description and Operator's Guide</i>	3211 プリンターのユーザー設計トレーンの作成に関する情報。
<i>IBM 3262 Model 5 Printer Product Description</i>	3262 モデル 5 プリンターのバンド ID に関する情報。
<i>IBM 3800 Printing Subsystem Programmer's Guide</i>	IBM 3800 モデル 1 印刷サブシステムの計画、変換、および使用に関する情報。
<i>IBM 3800 Printing Subsystem Model 3 Programmer's Guide: Compatibility</i>	IBM 3800 モデル 3、6、および 8 印刷サブシステムの計画、変換、および使用に関する情報。

資料名	内容
<i>IBM 4245 Printer Model 1 Component Description and Operator's Guide</i>	4245 プリンターのバンド ID に関する情報。
<i>IBM 4248 Printer Model 1 Description</i>	4248 プリンターのバンド ID に関する情報。
<i>IBM 6262 Printer Print Band Manual</i>	6262 プリンターのバンド ID に関する情報。
<i>IBM 6262 Printer Model 014 User's Guide</i>	6262 モデル 14 プリンターの操作と保守に関する情報。

この章の情報を使用するには、以下の表にリストした資料の内容を十分に理解している必要があります。

資料名	内容
<i>z/OS DFSMS Macro Instructions for Data Sets</i>	SETPRT マクロについて説明しています。必要なイメージ (モジュール) を指定するのに使用できます。
<i>z/OS DFSMSdfp ユーティリティ</i>	IEBIMAGE ユーティリティ・プログラムについて説明しています。
<i>z/OS MVS JCL 解説書</i>	OPEN 時に処理される DD ステートメントの CHARS、MODIFY、UCB、および FCB パラメーターについて説明しています。
<i>z/OS JES2 初期設定およびチューニング ガイド</i>	JES2 参照情報が記載されています。
<i>z/OS JES3 初期設定およびチューニング ガイド</i>	JES3 参照情報が記載されています。

表 80 は、SYS1.IMAGELIB のプリンター固有の内容についての情報を提供します。

表 80. SYS1.IMAGELIB の内容

プリンター・タイプ	UCS イメージ	UCS イメージ・テーブル ⁸	FCB イメージ	制御モジュール
1403	X			
3203 ¹	X		X	
3211	X		X	
3262 モデル 5 ^{2,4,5}		X	X	
4245 ³		X ⁶	X	
4248 ³		X ⁶	X	
6262 モデル 14 ^{2,4,5}		X	X	
3800				X ⁷

表 80 の注:

1. IBM 3203 モデル 5 プリンターは、JES によって 3211 プリンターと同じに扱われます。ただし、3203 モデル 5 は 3211 指標付け機能をサポートせず、JES

からの指標付けコマンドを無視します。3203 モデル 5 は、独自の固有 UCS イメージを使用しますが、3211 FCB イメージを使用します。

2. このプリンターは、4248 ネイティブ・モードで操作している 4248 と同様に操作できます。この場合は、本章の 4248 関連の UCS セクションを使用してください。UCS 情報はイメージ・テーブルに含まれています。
3. このプリンターはネイティブ・モードで操作できます。この場合、UCS 情報はイメージ・テーブルに含まれています。このプリンターは 3211 互換モードで操作することもできます。この場合、UCS 情報は UCS イメージに含まれていません。その場合には、本章の 3211 関連の UCS セクションを使用する必要があります。
4. このプリンターは、4248 プリンターと同じイメージ・テーブルを使用します。ただし、可変印刷速度または水平コピー機能および 4248 プリンターのホスト・スタッカー制御はサポートしません。
5. このプリンターは、4248 プリンターと同じ FCB イメージを使用します。
6. イメージ・テーブルは IBM によって提供されます。4245 イメージ・テーブルの内容を、336 ページの表 81 に示します。4248 イメージの内容は、337 ページの表 82 に示します。
7. 以下の 3800 印刷サブシステム制御モジュールが存在します。
 - 文字配列テーブル
 - 図形文字変更テーブル
 - コピー変更テーブル
 - ライブラリー文字セット
 - FCB イメージ

IEBIMAGE ユーティリティ・プログラムを使用して、これらの制御モジュールを作成および保守できます。IEBIMAGE についての詳細は、「z/OS DFSMSdfp ユーティリティ」を参照してください。プリンターをページ・モードで実行しているときは、IEBIMAGE や SYS1.IMAGELIB は使用しません。その場合、PSF for z/O は、別のライブラリーから別のタイプのモジュールを使用します。詳しくは、「3900 製品説明」を参照してください。

8. 336 ページの図 41 は、UCS イメージ・テーブル項目の構造を定義し、説明しています。

SYS1.IMAGELIB 内の UCS イメージ

このセクションは、IBM 1403、3203、および 3211 プリンターに適用されます。SYS1.IMAGELIB には、これらのプリンターの UCS イメージが含まれています。アセンブラーまたはリンケージ・エディターを使用して、UCS イメージを SYS1.IMAGELIB 常駐のイメージに追加することができます。アセンブラーは、実行可能コードは生成しません。これは単に DC ステートメントを準備するだけで、リンケージ・エディターがそれを SYS1.IMAGELIB に入れます。新規の UCS イメージを作成する場合は、以下の規則を守ってください。

1. メンバー名は 5 文字から 8 文字の長さでなければなりません。最初の 4 文字には、以下の該当する UCS 接頭部を使用する必要があります。

接頭部	意味
UCS1	1403 プリンター

接頭部	意味
UCS2	3211 プリンター (または 3211 互換プリンター)
UCS3	3203 プリンター

最初の 4 文字の後に、1 文字から 4 文字の長さの文字セット・コードを続ける必要があります。アセンブラー言語規則に基づく文字と数字の有効な組み合わせが受け入れられます。ただし、単一文字の U または C は使用しないでください。これらはシステムによって認識される特殊条件のシンボルであるからです。イメージを UCS バッファにロードするための DD ステートメントまたは SETPRT マクロで、割り当てられた文字セット・コードを指定してください。

ALIAS ステートメントを使用すると、新規のイメージの別名を提供できます。(ALIAS ステートメントについて詳しくは、「z/OS MVS プログラム管理: ユーザーズ・ガイドおよび解説書」を参照してください。)

- 文字セット・イメージ・ロード・モジュールの最初のバイトは、イメージがデフォルトであるかどうかを指定します。プログラムがプリンターに対する OPEN マクロを発行し、その中の UCS バッファにデフォルト・イメージがロードされており、JCL が UCS 名を指定していない場合、システムはバッファ内のイメージを使用します。バッファにデフォルト・イメージがロードされていない場合、システムはオペレーターにアクションを取るよう指示します。

JES2 の最初のバイトで、以下を指定してください。

値	意味
X'00'	イメージをデフォルトとして使用しないことを示します。
X'40'	出力を大文字変換する必要があることを示します。
X'80'	デフォルト・イメージを示します。
X'C0'	デフォルト・イメージと大文字への変換を示します。

非 JES2 の場合、以下を指定します。

値	意味
X'00'	イメージをデフォルトとして使用しないことを示します。
X'80'	デフォルト・イメージを示します。

- ロード・モジュールの 2 番目のバイトは、イメージ検査のために印刷する必要がある行数 (n) を示します。イメージ検査について詳しくは、343 ページの『UCS イメージの検査』を参照してください。
- 次の n バイトの各バイトは、各検査行に印刷する文字数を示します。3211 プリンターの場合、1 行に印刷される文字の最大数は 48 です。連想ビットのバイト (下記の規則 5 を参照) は、検査時には印刷されません。
- UCS イメージ自体を、前述のフィールドの後に続ける必要があります。イメージは、プリンターによって必要とされるバイト数を充てんする必要があります。イメージの長さについては、下表を参照してください。アセンブラー言語構文では、文字セット・イメージ内で、それぞれ単一アポストロフィまたは単一アンパサンド (& 記号) を表すために、2 つのアポストロフィまたは 2 つのアンパサンドをコーディングする必要があることに注意してください。

プリンター	イメージの長さ
1403	240 バイト
3203	304 バイト (240 文字の後に 64 バイトの連想ビット)
3211	512 バイト (432 文字の後に 15 バイトの X'00', 64 バイトの連想ビット、および予約済み 1 バイト X'00')

UCS イメージを SYS1.IMAGELIB に追加する場合、データ・チェックを防止するために、連想ビットをコーディングする必要があります。連想ビットのコーディングについて詳しくは、該当のプリンターの資料を参照してください。

UCS イメージのコーディング例

図 38 は、1403 UCS イメージ YN を SYS1.IMAGELIB に追加するための JCL の例を示しています。すべての例に適用される注を、333 ページの図 40 の後に記載します。

```
//ADDYN      JOB  MSGLEVEL=1
//STEP       EXEC PROC=ASMHCL,PARM.ASM='NODECK,LOAD',
//           PARM.LKED='LIST,OL,REFR,RENT,XREF'
//ASM.SYSIN DD *
UCS1YN      CSECT
             DC  X'80'          (THIS IS A DEFAULT IMAGE)
             DC  AL1(6)         (NUMBER OF LINES TO BE PRINTED)
             DC  AL1(39)        (39 CHARACTERS TO BE PRINTED ON LINE 1)
             DC  AL1(42)        (42 CHARACTERS TO BE PRINTED ON LINE 2)
             DC  AL1(39)        (39 CHARACTERS TO BE PRINTED ON LINE 3)
             DC  AL1(39)        (39 CHARACTERS TO BE PRINTED ON LINE 4)
             DC  AL1(42)        (42 CHARACTERS TO BE PRINTED ON LINE 5)
             DC  AL1(39)        (39 CHARACTERS TO BE PRINTED ON LINE 6)
*           THE FOLLOWING SIX LINES REPRESENT THE TRAIN IMAGE
             DC  C'1234567890STABCEFGHIJKLMNOPQRUVWXYZ*,.'
             DC  C'1234567890STABCEFGHIJKLMNOPQRUVWXYZ*,.##-$'
             DC  C'1234567890STABCEFGHIJKLMNOPQRUVWXYZ*,.'
             DC  C'1234567890STABCEFGHIJKLMNOPQRUVWXYZ*,.'
             DC  C'1234567890STABCEFGHIJKLMNOPQRUVWXYZ*,.##-$'
             DC  C'1234567890STABCEFGHIJKLMNOPQRUVWXYZ*,.'
             END
/*
//LKED.SYSLMOD DD DSNAME=SYS1.IMAGELIB(UCS1YN),DISP=OLD,
//           SPACE=          (OVERRIDE SECONDARY ALLOCATION)
```

図 38. 1403 UCS イメージを SYS1.IMAGELIB に追加するコード

333 ページの図 40 で、この図の注を参照してください。

図 39 は、3203 UCS イメージ YN を SYS1.IMAGELIB に追加するための JCL の例を示しています。この図の注は、333 ページの図 40 の後にあります。

```

//ADYN3203 JOB MSGLEVEL=1
//STEP EXEC PROC=ASMHCL,PARM.ASM='NODECK,LOAD',
// PARM.LKED='LIST,OL,REFR,RENT,XREF'
//ASM.SYSIN DD *
UCS3YN CSECT
        DC X'80' (THIS IS A DEFAULT IMAGE)
        DC AL1(6) (NUMBER OF LINES TO BE PRINTED)
        DC AL1(39) (39 CHARACTERS TO BE PRINTED ON LINE 1)
        DC AL1(42) (42 CHARACTERS TO BE PRINTED ON LINE 2)
        DC AL1(39) (39 CHARACTERS TO BE PRINTED ON LINE 3)
        DC AL1(39) (39 CHARACTERS TO BE PRINTED ON LINE 4)
        DC AL1(42) (42 CHARACTERS TO BE PRINTED ON LINE 5)
        DC AL1(39) (39 CHARACTERS TO BE PRINTED ON LINE 6)
* THE FOLLOWING SIX LINES REPRESENT THE TRAIN IMAGE
        DC C'1234567890STABCDEFGHIJKLMNOPQRUVWXYZ*,.'
        DC C'1234567890STABCDEFGHIJKLMNOPQRUVWXYZ*,.##-$'
        DC C'1234567890STABCDEFGHIJKLMNOPQRUVWXYZ*,.'
        DC C'1234567890STABCDEFGHIJKLMNOPQRUVWXYZ*,.'
        DC C'1234567890STABCDEFGHIJKLMNOPQRUVWXYZ*,.##-$'
        DC C'1234567890STABCDEFGHIJKLMNOPQRUVWXYZ*,.'
* THE FOLLOWING FOUR DC INSTRUCTIONS DEFINE THE ASSOCIATIVE BITS,
* UCSB BYTE POSITIONS 241-304
        DC X'C0101010101010101010004000000000010'
        DC X'10101010101010101000404000000040001010'
        DC X'101010101010004000000000101010101010'
        DC X'10101010004000000000'
        END
/*
//LKED.SYSLMOD DD DSNAME=SYS1.IMAGELIB(UCS3YN),DISP=OLD,
// SPACE= (OVERRIDE SECONDARY ALLOCATION)

```

図 39. 3203 UCS イメージを SYS1.IMAGELIB に追加するコード

注

333 ページの図 40 で、この図の注を参照してください。

図 40 は、3211 UCS イメージ A11 を SYS1.IMAGELIB に追加するための JCL の例を示しています。

```
//ADDA11 JOB MSGLEVEL=1
//STEP EXEC PROC=ASMHCL,PARM.ASM='NODECK,LOAD',
// PARM.LKED='LIST,OL,REFR,RENT,XREF'
//ASM.SYSIN DD *
UCS2A11 CSECT
DC X'80' (THIS IS A DEFAULT IMAGE)
DC AL1(9) (NUMBER OF LINES TO BE PRINTED)
DC AL1(48) (48 CHARACTERS TO BE PRINTED ON LINE 1)
DC AL1(48) (48 CHARACTERS TO BE PRINTED ON LINE 2)
DC AL1(48) (48 CHARACTERS TO BE PRINTED ON LINE 3)
DC AL1(48) (48 CHARACTERS TO BE PRINTED ON LINE 4)
DC AL1(48) (48 CHARACTERS TO BE PRINTED ON LINE 5)
DC AL1(48) (48 CHARACTERS TO BE PRINTED ON LINE 6)
DC AL1(48) (48 CHARACTERS TO BE PRINTED ON LINE 7)
DC AL1(48) (48 CHARACTERS TO BE PRINTED ON LINE 8)
DC AL1(48) (48 CHARACTERS TO BE PRINTED ON LINE 9)
* THE FOLLOWING NINE LINES REPRESENT THE TRAIN IMAGE
* NOTE 2 AMPERSANDS MUST BE CODED TO GET 1 IN ASSEMBLER SYNTAX
DC C'1<.+IHGFEDCBA*$-RQPONMLKJ%,&&ZYXWVUTS/@#098765432'
DC C'1<.+IHGFEDCBA*$-RQPONMLKJ%,&&ZYXWVUTS/@#098765432'
DC C'1<.+IHGFEDCBA*$-RQPONMLKJ%,&&ZYXWVUTS/@#098765432'
DC C'1<.+IHGFEDCBA*$-RQPONMLKJ%,&&ZYXWVUTS/@#098765432'
DC C'1<.+IHGFEDCBA*$-RQPONMLKJ%,&&ZYXWVUTS/@#098765432'
DC C'1<.+IHGFEDCBA*$-RQPONMLKJ%,&&ZYXWVUTS/@#098765432'
DC C'1<.+IHGFEDCBA*$-RQPONMLKJ%,&&ZYXWVUTS/@#098765432'
DC C'1<.+IHGFEDCBA*$-RQPONMLKJ%,&&ZYXWVUTS/@#098765432'
DC C'1<.+IHGFEDCBA*$-RQPONMLKJ%,&&ZYXWVUTS/@#098765432'
DC 15X'00' (RESERVED FIELD, BYTES 433-447)
* THE FOLLOWING FOUR DC INSTRUCTIONS DEFINE THE ASSOCIATIVE BITS,
* UCSB BYTE POSITIONS 448-5111
DC X'C010101010101010100040404240004010'
DC X'10101010101010101000404041000040401010'
DC X'101010101010004040000000101010101010'
DC X'10101010004040444800'
DC X'00' (RESERVED FIELD, BYTE 512)
END
/*
//LKED.SYSLMOD DD DSN=SYS1.IMAGELIB(UCS2A11)DISP=OLD,
// SPACE= (OVERRIDE SECONDARY ALLOCATION)
```

図 40. 3211 UCS イメージを SYS1.IMAGELIB に追加するサンプル・コード

331 ページの図 38、332 ページの図 39、および 図 40 のサンプル・コードでは、以下の事項に注意してください。

- RENT リンケージ・エディター属性が必要です。
- 3203 および 3211 プリンターの場合、データ・チェックを避けるために、64 バイトの連想ビットをコーディングしてください。特定のイメージについて、これらのビットをコーディングする方法を調べるには、「*IBM 3203 Printer Component Description and Operator's Guide*」または「*IBM 3211 Printer, 3216 Interchangeable Train Cartridge, and 3811 Printer Control Unit Component Description and Operator's Guide*」を参照してください。

- ASMHCL プロシージャーを実行しても、実行可能コードを生成しません。アセンブラー/リンケージ・エディターは、単に UCS イメージを SYS1.IMAGELIB に入れるだけです。
- ここでは、ASMHCCL カタログ式プロシージャーは 2 次割り振りを指定しているため、SPACE パラメーターはオーバーライドされます。オーバーライドを削除することにより、元の 2 次割り振り量を使用することを指定できます。

UCS イメージ別名

IBM 提供の印刷バンドおよびトレーンの多くは、別名が提供されています。例えば、データ・セットが 3211 で印刷された場合、1403 TN トレーンを要求すると、T11 トレーンが割り当てられます。現在 SYS1.IMAGELIB で使用されている命名規則に従って割り当てられた別名は、次のとおりです。

イメージ	別名
UCS1AN	UCS1A11
UCS1HN	UCS1H11
UCS1PN	UCS1P11
UCS1TN	UCS1T11
UCS2A11	UCS2AN
UCS2H11	UCS2HN
UCS2P11	UCS2PN、UCS2RN、UCS2QN
UCS2T11	UCS2TN

イメージと別名は、システムのインストール時に SYS1.IMAGELIB に組み込まれます。

一部のバンド/トレーン (SN および G11 など) は、他のプリンター上に等価バンド/トレーンを持っていないため、別名を持っていません。システムは別名を割り当てることができます (選択した場合)。(ALIAS ステートメントについての詳細は、「z/OS MVS プログラム管理: ユーザーズ・ガイドおよび解説書」を参照してください。) 名前 (または、別名) を指定しない場合は、システム定義の SYSOUT クラスまたはプリンター宛先コードを指定して、データ・セットを正しいプリンターに割り当てる必要があります。JES がプリンターでデータ・セットを印刷するように指示され、指定されたイメージが存在しない場合、JES はオペレーターに通知します。オペレーターは、有効なバンド/トレーンを使用してデータ・セットを印刷するか、またはデータ・セットを正しいプリンターに宛先変更することができます。システムが新規のブランド/トレーンを定義する場合、イメージを SYS1.IMAGELIB に組み込むときに、ALIAS ステートメントを通してその別名を提供できます。

SYS1.IMAGELIB 内の UCS イメージ・テーブル

このセクションは、IBM 3262 モデル 5、4245、4248、および 6262 モデル 14 プリンターにのみ適用されます。SYS1.IMAGELIB は、これらのプリンターの UCS イメージを含んでいませんが、代わりに、イメージ・テーブルが入っています。プリンターを 3211 互換モードで実行している場合、UCS 情報はイメージ・テーブルに含まれています。この章の 3211 関連のセクションを参照してください。各バンドの UCS イメージはプリンター内部に保管されており、ユーザーがマシンの電源をオンにして、新規のバンドをインストールすると、自動的に UCS バッファーに

ロードされます。イメージ・テーブル項目のフォーマットについては、336 ページの図 41 を参照してください。イメージ・テーブル項目を追加または変更する方法については、339 ページの『UCS イメージ・テーブル項目の追加または変更』を参照してください。

SYS1.IMAGELIB は、イメージ・テーブルをサポートする各タイプのプリンターごとに 1 つの UCS イメージ・テーブルを含んでいます。イメージ・テーブルには、ほとんどのシステムの標準 IBM 提供バンドの項目が含まれています。4245 イメージ・テーブルは UCS5 という名前です。共用 4248、3262 モデル 5、および 6262 モデル 14 イメージ・テーブルは、UCS6 という名前です。

UCS イメージ・テーブル内の別名

イメージ・テーブルは、IBM 4245 および 4248 プリンターで使用されるほとんどのインストール標準印刷バンドの別名も定義します。IBM 提供のイメージ・テーブルは、IBM 3262 モデル 5 または 6262 モデル 14 プリンターの別名は提供しません。

一部の印刷バンド (SN および KA22 など) は、他のプリンター上に等価バンドが存在しないため、別名を持っていません。該当する UCS イメージ・テーブルの項目を追加または変更することにより、別名を追加できます。339 ページの『UCS イメージ・テーブル項目の追加または変更』を参照してください。標準的な UCS イメージ・テーブル項目を、336 ページの図 41 に示します。

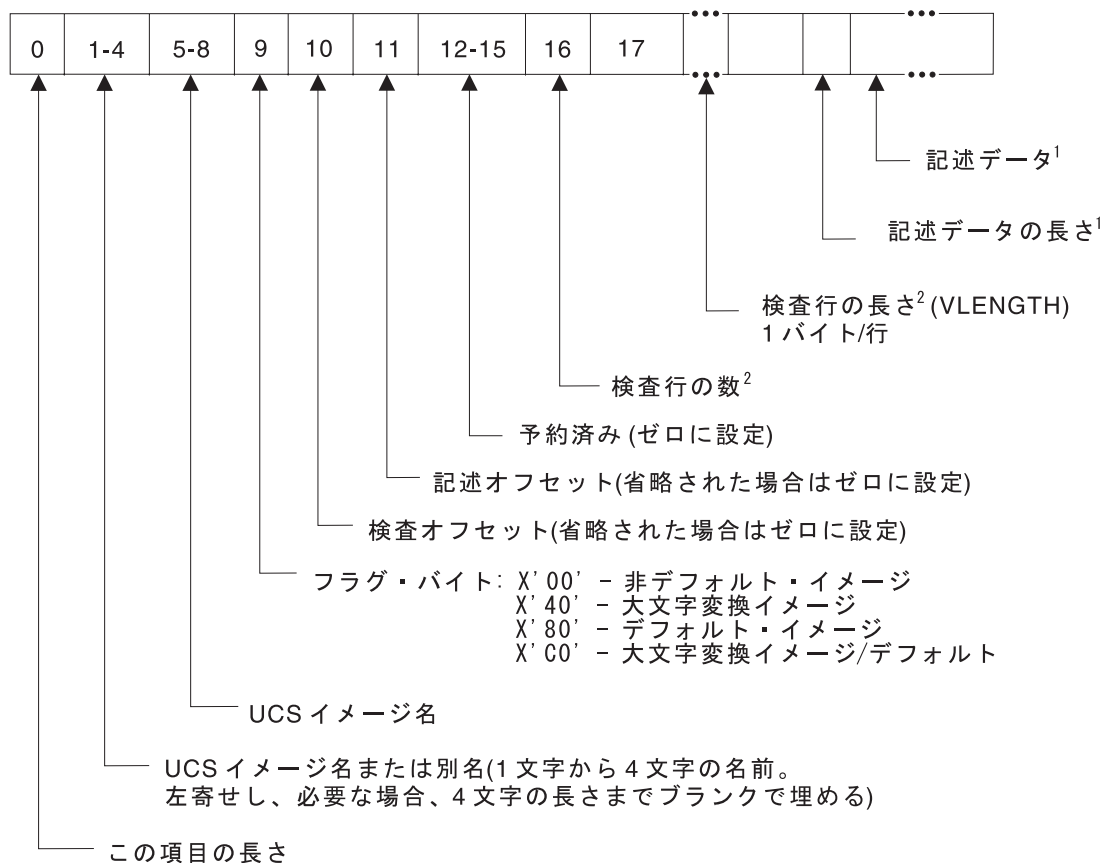


図 41. UCS イメージ・テーブル項目フォーマット

図 41 の注:

1. このフィールドはオプションです。データ・フィールドの記述は可変長で、最大 32 バイトです。
2. 4245 プリンターの場合、このフィールドはオプションです。3262 モデル 5、4248、および 6262 モデル 14 の場合、このフィールドは適用されず、X'00' に設定されます。

4245 プリンターの UCS イメージ・テーブル UCS5 の内容 (IGGUCS5 マクロ) を、表 81 に示します。

表 81. UCS5 イメージ・テーブルの内容

名前	別名	デフォルト	説明
AN21	AN21	はい	デフォルト UCS イメージ
AN21	AN	いいえ	1403/3203 AN イメージ
AN21	A11	いいえ	3211 A11 イメージ
AN21	40E1	いいえ	4248 40E1 イメージ
HN21	HN21	いいえ	非デフォルト UCS イメージ
HN21	HN	いいえ	1403/3203 HN イメージ
HN21	H11	いいえ	3211 H11 イメージ
HN21	4101	いいえ	4248 4101 イメージ

表 81. UCS5 イメージ・テーブルの内容 (続き)

名前	別名	デフォルト	説明
PL21	PL21	いいえ	非デフォルト UCS イメージ
PL21	PN	いいえ	1403/3203 PN イメージ
PL21	P11	いいえ	3211 P11 イメージ
PL21	4121	いいえ	4248 4121 イメージ
SN21	SN21	いいえ	非デフォルト UCS イメージ
SN21	4201	いいえ	4248 4201 イメージ
TN21	TN21	いいえ	非デフォルト UCS イメージ
TN21	TN	いいえ	1403/3203 TN イメージ
TN21	T11	いいえ	3211 T11 イメージ
TN21	4181	いいえ	4248 4181 イメージ
GN21	GN21	いいえ	非デフォルト UCS イメージ
GN21	G11	いいえ	3211 G11 イメージ
GN21	41C1	いいえ	4248 41C1 イメージ
RN21	RN21	いいえ	非デフォルト UCS イメージ
RN21	RN	いいえ	1403/3203 RN イメージ
KA21	KA21	いいえ	非デフォルト UCS イメージ
KA21	4041	いいえ	4248 4041 イメージ
KA22	KA22	いいえ	非デフォルト UCS イメージ
FC21	FC21	いいえ	非デフォルト UCS イメージ
FC21	4161	いいえ	4248 4161 イメージ

4248 プリンターの UCS イメージ・テーブル UCS6 の内容 (IGGUCS6 マクロ) を、表 82 に示します。

表 82. UCS6 イメージ・テーブルの内容

名前	別名	デフォルト	説明
40E1	40E1	はい	デフォルト UCS イメージ
40E1	AN21	いいえ	4245 AN21 イメージ
40E1	AN	いいえ	1403/3203 AN イメージ
40E1	A11	いいえ	3211 A11 イメージ
4101	4101	いいえ	非デフォルト UCS イメージ
4101	HN21	いいえ	4245 HN21 イメージ
4101	HN	いいえ	1403/3203 HN イメージ
4101	H11	いいえ	3211 H11 イメージ
41C1	41C1	いいえ	非デフォルト UCS イメージ
41C1	GN21	いいえ	4245 GN21 イメージ
41C1	G11	いいえ	3211 G11 イメージ
4121	4121	いいえ	非デフォルト UCS イメージ
4121	PL21	いいえ	4245 PL21 イメージ
4121	PN	いいえ	1403/3203 PN イメージ

表 82. UCS6 イメージ・テーブルの内容 (続き)

名前	別名	デフォルト	説明
4121	P11	いいえ	3211 P11 イメージ
4181	4181	いいえ	非デフォルト UCS イメージ
4181	TN21	いいえ	4245 TN21 イメージ
4181	TN	いいえ	1403/3203 TN イメージ
4181	T11	いいえ	3211 T11 イメージ
4061	4061	いいえ	非デフォルト UCS イメージ
40C1	40C1	いいえ	非デフォルト UCS イメージ
4161	4161	いいえ	非デフォルト UCS イメージ
4161	FC21	いいえ	4245 FC21 イメージ
4201	4201	いいえ	非デフォルト UCS イメージ
4201	SN21	いいえ	4245 SN21 イメージ
4041	4041	いいえ	非デフォルト UCS イメージ
4041	KA21	いいえ	4245 KA21 イメージ

ヒント: 4245 および 4248 プリンターのイメージ・テーブルには、USA とカナダのバンド ID のみが含まれています。他の国のバンド ID をサポートするには、UCS イメージ・テーブルを変更してください。表 83 および 339 ページの『UCS イメージ・テーブル項目の追加または変更』を参照してください。

3262 モデル 5 および 6262 モデル 14 プリンターは、4248 UCS イメージ・テーブル UCS6 を使用します。ただし、IBM では 3262 モデル 5 または 6262 モデル 14 プリンターのバンド名または別名を提供していません。3262 モデル 5 または 6262 モデル 14 UCS イメージを使用するには、名前と別名を UCS6 に追加してください。339 ページの『UCS イメージ・テーブル項目の追加または変更』で、UCS イメージ・テーブルに項目を追加する方法を説明しています。3262 モデル 5 に使用可能なバンドのリストは、表 83 を参照してください。6262 モデル 14 に使用可能なバンドのリストは、「*IBM 6262 Printer Print Band Manual*」および「*IBM 6262 Printer Model 014 User's Guide*」を参照してください。

表 83 では、以下の事項に注意してください。

- UCS6 バンド名の別名として、任意の 1 文字から 4 文字の名前を定義できます。
- *xx* は、スイッチ設定を示しますが、これは無視されます。特別注文 (RPQ) バンドを使用する場合は、スイッチ番号 3 をオンにする必要があります。
- ここでの *bb* はスペースを表すことに注意してください。

表 83. 3262 モデル 5 印刷バンド

文字セット名	UCS6 バンド名	3262 モデル 5 バンド・ イメージ選択スイッチ設定 (位置 1-8)
U.S./インターナショナル 48 char EBCDIC	00 <i>bb</i>	<i>xx</i> 00 0000

表 83. 3262 モデル 5 印刷バンド (続き)

文字セット名	UCS6 バンド名	3262 モデル 5 バンド・ イメージ選択スイッチ設定 (位置 1-8)
63 char EBCDIC	01bb	xx00 0001
64 char EBCDIC	02bb	xx00 0010
96 char EBCDIC	03bb	xx00 0011
48 char AON OCR	04bb	xx00 0100
48 char BON OCR	05bb	xx00 0101
オーストリア/ドイツ		
52 char EBCDIC	06bb	xx00 0110
63 char EBCDIC	07bb	xx00 0111
64 char EBCDIC	08bb	xx00 1000
96 char EBCDIC	09bb	xx00 1001
52 char AON OCR	0Abb	xx00 1010
52 char BON OCR	0Bbb	xx00 1011
カナダ/フランス		
116 char EBCDIC	0Cbb	xx00 1100
カタカナ		
96 char EBCDIC	0Dbb	xx00 1101
128 char EBCDIC	0Ebb	xx00 1110
UINN		
128 char U.S. text	0Fbb	xx00 1111
WTNN		
128 char World Trade text	10bb	xx01 0000

UCS イメージ・テーブル項目の追加または変更

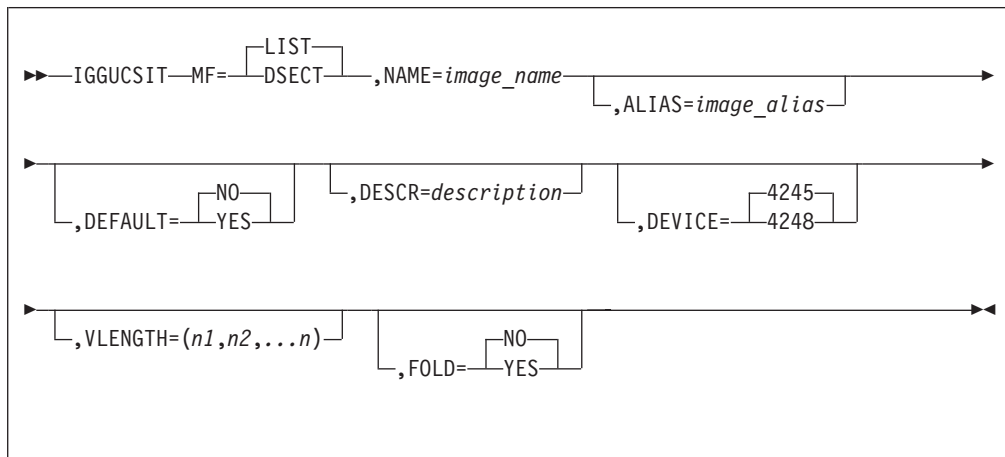
3262 モデル 5、4245、4248、または 6262 モデル 14 プリンターで新規の UCS イメージ名/別名を使用するには、そのイメージ名/別名の項目を UCS イメージ・テーブルに追加します。以下の手順に示すように、アセンブラーを使用してイメージ・テーブル・オブジェクト・モジュールを作成し、次にオブジェクト・モジュールを SYS1.IMAGELIB にリンク・エディットします。同様に、他のイメージをデフォルトとして指定したり、古いイメージの記述を変更する場合も、イメージ・テーブルを変更します。

新規の UCS テーブル項目を作成したり、古い項目のフォーマットを変更するには、次の手順を使用してください。IGGUCSIT マクロのコーディングの例は、342 ページの図 42 および 343 ページの図 43 を参照してください。

1. 新規の UCS イメージ・テーブル項目を作成するには、下記のように、IGGUCSIT マクロを発行します。以下の例に示すようにイメージ・テーブルを更新する場合、ユーザーが既存の項目を置換する意図であっても、リンケージ・エディターはテーブルの開始に新しい項目を作成します。後でシステムがテーブルを使用するときに、システムは最初に新しい項目を検出し、これにより古い項目は事実上、置換されます。
2. IGGUCS5 または IGGUCS6 マクロ (両方とも、SYS1.MODGEN に常駐) を使用して、UCS イメージ・テーブル・ソースを組み込みます。
3. イメージ・テーブル・モジュール (UCS5 または UCS6) をアセンブルします。
4. アセンブルされたモジュールを SYS1.IMAGELIB にリンク・エディットします。

要件: RENT リンケージ・エディター属性が必要です。

IGGUCSIT マクロのフォーマットは、次のとおりです。



MF=LIST または DSECT

マクロ命令の形式を指定します。

LIST 別の IGGUCSIT パラメーターで提供された情報に基づいて UCS イメージ・テーブル項目を生成します。LIST が選択されているか、デフォルトになる場合、NAME パラメーターもコーディングする必要があります。

DSECT 単一 UCS イメージ・テーブル項目の DSECT を生成します (336 ページの図 41 に示されたサンプル項目と同様の)。DSECT をコーディングした場合、IGGUCSIT の他のパラメーターはすべて無視されます。

LIST がデフォルトです。

NAME=*image_name*

1 文字から 4 文字の UCS イメージ名を指定します。

ALIAS=*image_alias*

1 文字から 4 文字の UCS イメージの別名を指定します。ALIAS が指定されていない場合、NAME パラメーターでコーディングされたイメージ名が UCS イメージ・テーブルに入れられます。

例外: 3262、4248、および 6262 プリンターは、共通のバンド ID を持っていません。これらの装置はすべて UCS6 イメージ・テーブルを使用し、プリンター・タイプごとに固有の別名を選択するからです。別名はイメージ・テーブルに複数回表示されてはなりません。

DEFAULT=YES または NO

新規の UCS イメージをデフォルト値として使用するかどうかを指定します。

YES この UCS イメージがデフォルトであることを示します。デフォルト・イメージは、特定のイメージを要求しないジョブのために、システムによって使用されます。

NO この UCS イメージはデフォルトとして使用してはならないことを示します。

DEFAULT パラメーターが指定されていない場合、新規の UCS イメージはデフォルトとして使用されません。

DESCR=description

新規の UCS イメージに関する記述情報を指定します。*description* は、最大 32 文字の長さの EBCDIC または 16 進文字です。EBCDIC と 16 進文字を組み合わせることはできません。

記述情報は、検査表示のヘッダー行の実際の UCS イメージ名の後に置かれません。DESCR パラメーターを省略した場合、記述は表示されません。検査表示について詳しくは、343 ページの『UCS イメージの検査』を参照してください。

4245 プリンターの場合、VLENGTH が指定されていない場合、DESCR パラメーターは無視されます。

DEVICE=4245 または 4248

イメージ・テーブル項目を作成する装置のタイプを指定します。

IGGUCSIT マクロの最初の呼び出しで MF=LIST を指定した場合、DEVICE はデフォルトの 4245 になります。後続の呼び出しのデフォルトは、最初の呼び出しで指定された (または、デフォルトの) プリンター・タイプになります。異なる DEVICE 仕様をもつテーブル項目は許可されません。

3262 モデル 5 または 6262 モデル 14 プリンターの場合、適切な形式のイメージ・テーブル項目を作成するには、DEVICE=4248 を指定する必要があります。

VLENGTH=(n1,n2,.. . n)

UCS 検査表示の各行の長さを指定します。すべての行が同じ長さであっても、それぞれの行の長さを個別に指定する必要があります。

n1 は、印刷行 1 の長さです。*n2* は、印刷行 2 の長さです。*n* は、最後の印刷行の長さです。完全なイメージを表示するには、検査行の長さの合計が 350 に等しくなることが必要です。

検査レポートについての詳細は、343 ページの『UCS イメージの検査』を参照してください。

VLENGTH パラメーターは、3262 モデル 5、4248、または 6262 モデル 14 プリンターに対しては無効です。

FOLD=YES または NO

UCS イメージを大文字変換する必要があるかどうかを示します。

YES UCS イメージを大文字変換することを示します。大文字または小文字のデータ・コードから大文字のみで印刷できます。大文字への変換は、UNFOLD コマンドを受け取るまで続きます。

NO UCS イメージを大文字変換しないことを示します。これはデフォルトです。

UCS イメージ・テーブルへの追加

342 ページの図 42 では、記述『RPQ BAND』を持つバンド名 RPQ1 が UCS5 に追加されます。UCS 検査表示には、各行が 50 文字で 7 行が印刷されます。マ

クロ IGGUCS5 により、UCS イメージ・テーブル・ソース (IBM が配布) がテーブル項目に組み込まれます。

```

                                                                    72
//UCS5  JOB . . .
//      EXEC ASMHCL,
//          PARM.ASM='NODECK,LOAD',
//          PARM.LKED='OL,RENT,REUS'
//SYSPRINT DD SYSOUT=A
//ASM.SYSIN DD *
          TITLE 'UPDATED UCS5 IMAGE TABLE'
UCS5     CSECT
          IGGUCSIT NAME=RPQ1,
          VLENGTH=(50,50,50,50,50,50,50),
          DESCR='RPQ BAND'
          IGGUCS5
          END
/*
//LKED.SYSLMOD DD DSN=SYS1.IMAGELIB(UCS5),DISP=OLD,
//              SPACE= (OVERRIDE SECONDARY ALLOCATION)

```

図 42. 4245 UCS イメージ・テーブル (UCS5) への新規バンド ID の追加

図 42 に示すように、新規バンド ID を 4245 UCS イメージ・テーブルに追加する場合、以下の事項に注意してください。

- RENT リンケージ・エディター属性が必要です。
- ASMHCL プロシーチャーを実行しても、実行可能コードを生成しません。アセンブラー/リンケージ・エディターは、更新された UCS イメージ・テーブルを SYS1.IMAGELIB に入れます。
- ここでは、ASMHCCL カタログ式プロシーチャーは 2 次割り振りを指定しているため、SPACE パラメーターはオーバーライドされます。オーバーライドを除去すると、元の 2 次割り振り量が使用されます。

343 ページの図 43 では、バンド名 40E1 DEFAULT BAND が UCS6 に追加され、デフォルト・バンドとして定義されました。バンド 40E1 に対して、別名 HN21 も定義されました。マクロ IGGUCS6 により、UCS イメージ・テーブル・ソース (IBM が配布) がテーブル項目に組み込まれます。

```

//UCS6 JOB . . .
// EXEC ASMHCL,
// PARM.ASM='NODECK,LOAD',
// PARM.LKED='OL,RENT,REUS'
//SYSLIB DD
// DD DSN=SYS1.AMODGEN,DISP=SHR
//SYSPRINT DD SYSOUT=A
//ASM.SYSIN DD *
TITLE 'UPDATED UCS6 IMAGE TABLE'
UCS6 CSECT
      IGGUCSIT NAME=40E1, X
      DEVICE=4248, X
      ALIAS=HN21, X
      DEFAULT=YES, X
      DESCR='40E1 DEFAULT BAND'
      IGGUCS6
      END
/*
//LKED.SYSLMOD DD DSN=SYS1.IMAGELIB(UCS6),DISP=OLD,
// SPACE= (OVERRIDE SECONDARY ALLOCATION)

```

図 43. 4248 UCS イメージ・テーブル (UCS6) への新規デフォルト項目の追加

図 43 では、以下の事項に注意してください。

- この方法は、40E1 の重複項目を作成し、これがテーブルの最初の項目になります。テーブルは順次に検索されるため、新しい項目は常に古い項目より前に見つかり、これによって古い項目を置き換えます。
- RENT リンケージ・エディター属性が必要です。
- ASMHCL プロシージャーを実行しても、実行可能コードを生成しません。アセンブラー/リンケージ・エディターは、更新された UCS イメージ・テーブルを SYS1.IMAGELIB に入れます。
- ASMHCL カタログ式プロシージャーは 2 次割り振りを指定しているため、SPACE パラメーターはオーバーライドされます。オーバーライドを除去すると、元の 2 次割り振り量が使用されます。

UCS イメージの検査

1403 (UCS 機能を持つ)、3203、3211、3262 モデル 5、4245、4248、および 6262 モデル 14 プリンターの場合、次のパラメーターのいずれかを使用して UCS イメージを印刷し、視覚的に検査することができます。

- JCL: UCS=(*character set code*,,**VERIFY**)
- SETPRT マクロ: UCS=(*character set code*,,**V**).

これらのパラメーターは、SYSOUT データ・セットには無効です。

これらのパラメーターは、3262 モデル 5、4248、および 6262 モデル 14 プリンターに対しても使用できます。ただし、UCS イメージを 3262 モデル 5、4248、または 6262 モデル 14 から直接読み取ることができないため、ヘッダー情報のみが印刷されます。プリンター上に表示される検査表示ヘッダーは、次のようになります。

```
UCS IMAGE VERIFICATION image_id [,FOLD] [description]
```

image_id 1 文字から 4 文字の UCS イメージ名。

description UCS イメージ・テーブル内のこの UCS イメージに対して提供される記述情報。

UCS VERIFY パラメーターについて詳しくは、「*z/OS MVS JCL 解説書*」および「*z/OS DFSMS Macro Instructions for Data Sets*」を参照してください。

SYS1.IMAGELIB 内の FCB イメージ

システムのインストール時に、2 つの標準 FCB イメージ (STD1 と STD2) が SYS1.IMAGELIB に組み込まれます。この標準イメージの名前は、文字「FCB2」で始まります。4248、3262 モデル 5、および 6262 モデル 14 プリンターの場合、名前が「FCB4」で始まる FCB イメージを定義できます。これらのプリンターは、「FCB2」接頭部をもつイメージ (3211 形式 FCB と呼ばれます) と「FCB4」接頭部をもつイメージの両方を使用できます。3800 を除いて、他のプリンターはすべて「FCB2」接頭部をもつイメージのみを使用できます。

ここでは「FCB2」接頭部をもつイメージを作成または置換する方法について説明します。「*z/OS DFSMSdfp ユーティリティ*」は、IEBIMAGE ユーティリティ・プログラムを使用して、3262 モデル 5、4248、または 6262 モデル 14 プリンター用の「FCB4」接頭部をもつイメージを作成、更新、または置換する方法について説明しています。「FCB4」接頭部をもつイメージについての詳細は、「*IBM 6262 Printer Model 014 Product Description*」または「*IBM 4248 Printer Model 1 Description*」を参照してください。

3262 モデル 5、4245、4248、および 6262 モデル 14 プリンターは、パワーオン時に、それぞれデフォルト FCB イメージをバッファにロードします。3262 モデル 5 のデフォルト FCB イメージは、11 インチの用紙、6 行/インチで、チャンネル 1 は 3 行目の印刷行、チャンネル 12 は行 64 になります。4245 デフォルト FCB イメージは、11 インチの用紙、6 行/インチで、チャンネル 1 は最初の印刷行です。4248 デフォルト FCB イメージは、ロードされた最後の FCB イメージです。6262 モデル 14 のデフォルト FCB イメージは、ロードされて保管された最後の FCB イメージまたはプリンターと一緒に出荷されたデフォルトのいずれかです。

STD1 は、8½ インチ用紙で 6 行/インチの行送りを設定します。STD2 は、デフォルト FCB イメージで、11 インチ用紙で 6 行/インチの行送りを設定します。両方のイメージとも、チャンネルは均等の間隔で、チャンネル 1 が 4 行目、チャンネル 9 が最終行になります。標準 STD1 および STD2 イメージのフォーマットは、345 ページの図 44 と 346 ページの図 45 を参照してください。

3800 印刷サブシステム STD3 の標準 FCB イメージは、システムのインストール時に SYS1.IMAGELIB に組み込まれます。3800 のモデルはすべて、名前が「FCB3」で始まる FCB イメージを使用します。IEBIMAGE ユーティリティを使用して、3800 印刷サブシステムの FCB モジュールの作成および変更を行います。

```
FCB2STD1  CSECT
DC      X'80'          DEFAULT
DC      AL1(48)       FCB IMAGE LENGTH = 48
DC      X'000000'     LINE 1, 2, 3
DC      X'01'         LINE 4, CHANNEL 1
DC      X'000000'     LINE 5, 6, 7
DC      X'02'         LINE 8, CHANNEL 2
DC      X'000000'     LINE 9, 10, 11
DC      X'03'         LINE 12, CHANNEL 3
DC      X'000000'     LINE 13, 14, 15
DC      X'04'         LINE 16, CHANNEL 4
DC      X'000000'     LINE 17, 18, 19
DC      X'05'         LINE 20, CHANNEL 5
DC      X'000000'     LINE 21, 22, 23
DC      X'06'         LINE 24, CHANNEL 6
DC      X'000000'     LINE 25, 26, 27
DC      X'07'         LINE 28, CHANNEL 7
DC      X'000000'     LINE 29, 30, 31
DC      X'08'         LINE 32, CHANNEL 8
DC      X'000000'     LINE 33, 34, 35
DC      X'0A'         LINE 36, CHANNEL 10
DC      X'000000'     LINE 37, 38, 39
DC      X'0B'         LINE 40, CHANNEL 11
DC      X'000000'     LINE 41, 42, 43
DC      X'0C'         LINE 44, CHANNEL 12
DC      X'000000'     LINE 45, 46, 47
DC      X'19'         LINE 48, CHANNEL 9-END OF FCB IMAGE
END
```

図 44. 標準 STD1 FCB イメージのフォーマット

```

FCB2STD2  CSECT
DC      X'80'          DEFAULT
DC      AL1(66)       FCB IMAGE LENGTH = 66
DC      X'000000'     LINE 1, 2, 3
DC      X'01'         LINE 4, CHANNEL 1
DC      X'0000000000' LINE 5, 6, 7, 8, 9
DC      X'02'         LINE 10, CHANNEL 2
DC      X'0000000000' LINE 11, 12, 13, 14, 15
DC      X'03'         LINE 16, CHANNEL 3
DC      X'0000000000' LINE 17, 18, 19, 20, 21
DC      X'04'         LINE 22, CHANNEL 4
DC      X'0000000000' LINE 23, 24, 25, 26, 27
DC      X'05'         LINE 28, CHANNEL 5
DC      X'0000000000' LINE 29, 30, 31, 32, 33
DC      X'06'         LINE 34, CHANNEL 6
DC      X'0000000000' LINE 35, 36, 37, 38, 39
DC      X'07'         LINE 40, CHANNEL 7
DC      X'0000000000' LINE 41, 42, 43, 44, 45
DC      X'08'         LINE 46, CHANNEL 8
DC      X'0000000000' LINE 47, 48, 49, 50, 51
DC      X'0A'         LINE 52, CHANNEL 10
DC      X'0000000000' LINE 53, 54, 55, 56, 57
DC      X'0B'         LINE 58, CHANNEL 11
DC      X'0000000000' LINE 59, 60, 61, 62, 63
DC      X'0C'         LINE 64, CHANNEL 12
DC      X'00'         LINE 65
DC      X'19'         LINE 66, CHANNEL 9-END OF FCB IMAGE
END
    
```

図 45. 標準 STD2 FCB イメージのフォーマット

イメージ・ライブラリーへの FCB イメージの追加

アセンブラーおよびリンケージ・エディターを使用して、3211 フォーマット FCB イメージを SYS1.IMAGELIB 内に常駐するイメージに追加することができます。実行可能コードは生成されません。アセンブラーは DC を準備し、リンケージ・エディターがそれを SYS1.IMAGELIB にリンクします。新規の FCB イメージは、以下の規則に従って構成する必要があります。

1. メンバー名は 8 バイトを超えることはできず、接頭部 FCB2 で始まらなければなりません。後続の文字は FCB イメージを識別し、イメージ ID と呼ばれます。有効なアセンブラー言語文字の任意の組み合わせを使用できますが、単一文字の 'C' または 'U' は例外で、これらは特殊条件を認識するためにシステムによって使用されます。イメージを FCB バッファにロードするには、DD ステートメントの FCB キーワードまたは SETPRT マクロで、イメージ ID を指定する必要があります。
2. FCB ロード・モジュールの最初のバイトは、そのイメージがデフォルトであるかどうかを指定します。(デフォルトのイメージは、特定のイメージを要求しないジョブのために、システムによって使用されます。) 最初のバイトには、以下を指定してください。

値	意味
X'80'	デフォルト・イメージを示します。
X'00'	非デフォルト・イメージを示します。

3. ロード・モジュールの 2 番目のバイトは、FCB イメージにロードするために制御装置に転送する必要があるバイトの数を示します。このカウントには、印刷位置指標付け機能のバイトも含まれます (使用される場合)。
4. ロード・モジュールの 3 番目のバイト (FCB イメージの最初のバイト) は、印刷位置指標付けバイトまたは行/インチ (lines-per-inch) バイトです。印刷位置指標付けバイトはオプションで、使用する場合は、lines-per-inch バイトの前に置きます。3203 モデル 5、3262 モデル 5、4245、4248、および 6262 モデル 14 プリンターは、指標付け機能をサポートしないため、指標付けバイトを受け入れて廃棄します (存在する場合)。印刷位置指標付け機能とその使用については、資料「*IBM 3211 Printer, 3216 Interchangeable Train Cartridge, and 3811 Printer Control Unit Component Description and Operator's Guide*」を参照してください。

3 番目のバイトの特殊指標フラグには、X'80' と、1 から 32 (デフォルトは 1) の 2 進数の指標値が入ります。この指標値は、左マージンを設定します。1 は、左揃えを示します。その他の値は、指定されたスペースの数だけ字下げされた行を示します。

用紙イメージは、行/インチ (lines-per-inch) (LPI) バイトで始まります。LPI バイトは、1 インチ当たりの行数 (6 または 8) を定義し、ページの最初の行も表します。

要件: JES2 によって制御されるプリンターでは、ここにチャンネル 1 ID を指定する必要があります。

通常、FCB イメージの長さは、それが表す用紙の長さに整合しています。例えば、6 LPI で印刷される 8½ インチ用紙は、51 バイトの長さ (8½ インチに 6 LPI を乗算) の FCB イメージを持っています。

LPI バイトは、次のように表示されます。

値	意味
X'ln'	8 LPI を設定します。
X'on'	6 LPI を設定します。

5. 残りのバイト (行) にはすべて X'0n' を入れる必要があります。ただし、最後のバイトは例外で、これは X'ln' でなければなりません。文字 n は、チャンネル (1 から 12) を表す 16 進値 1 から C、あるいは 0 (チャンネルを示さないことを意味する) です。

348 ページの図 46 では、FCB ロード・モジュールがアSEMBルされ、SYS1.IMAGELIB に追加されます。イメージは、印刷濃度 8 行/インチの 11 インチ用紙、15 改行文字位置 (1½ インチ) の右シフトを定義しています。

```

//ADDFCB JOB MSGLEVEL=1
//STEP EXEC PROC=ASMHCL,PARM.ASM='NODECK,LOAD',
// PARM.LKED='LIST,OL,REFR,RENT,XREF'
//ASM.SYSIN DD *
FCB2ID1 CSECT
*THIS EXAMPLE IS FOR A FORM LENGTH OF 11 INCHES WITH 8 LPI (88 LINES)
DC X'80' THIS IS A DEFAULT IMAGE
DC AL1(89) LENGTH OF FCB IMAGE AND INDEXING BYTE
DC X'8F' OFFSET 15 CHARACTERS TO THE RIGHT
DC X'10' 8 LINES PER INCH-NO CHANNEL FOR LINE 1
DC XL4'0' 4 LINES NO CHANNEL
DC X'01' CHANNEL 1 IN LINE 6
DC XL6'0' 6 LINES NO CHANNEL
DC X'02' CHANNEL 2 IN LINE 13
DC XL6'0' 6 LINES NO CHANNEL
DC X'03' CHANNEL 3 IN LINE 20
DC XL6'0' 6 LINES NO CHANNEL
DC X'04' CHANNEL 4 IN LINE 27
DC XL6'0' 6 LINES NO CHANNEL
DC X'05' CHANNEL 5 IN LINE 34
DC XL6'0' 6 LINES NO CHANNEL
DC X'06' CHANNEL 6 IN LINE 41
DC XL6'0' 6 LINES NO CHANNEL
DC X'07' CHANNEL 7 IN LINE 48
DC XL6'0' 6 LINES NO CHANNEL
DC X'08' CHANNEL 8 IN LINE 55
DC XL6'0' 6 LINES NO CHANNEL
DC X'09' CHANNEL 9 IN LINE 62
DC XL6'0' 6 LINES NO CHANNEL
DC X'0A' CHANNEL 10 IN LINE 69
DC XL6'0' 6 LINES NO CHANNEL
DC X'0B' CHANNEL 11 IN LINE 76
DC XL6'0' 6 LINES NO CHANNEL
DC X'0C' CHANNEL 12 IN LINE 83
DC XL4'0' 4 LINES NO CHANNEL
DC X'10' POSITION 88 LAST LINE IN IMAGE
END
/*
//LKED.SYSLMOD DD DSNAME=SYS1.IMAGELIB(FCB2ID1),DISP=OLD,
// SPACE= (OVERRIDE SECONDARY ALLOCATION)

```

図 46. FCB ロード・モジュールをアセンブルして *SYS1.IMAGELIB* に追加するためのサンプル・コード

図 46 では、以下の事項に注意してください。

- RENT リンケージ・エディター属性が必要です。
- ASMHCL プロシージャーを実行しても、実行可能コードを生成しません。アセンブラー/リンケージ・エディターを使用して、FCB イメージを *SYS1.IMAGELIB* に入れます。
- ここでは、ASMHCL カタログ式プロシージャーは 2 次割り振りを指定しているため、SPACE パラメーターはオーバーライドされます。オーバーライドを除去すると、元の 2 次割り振り量が使用されます。

FCB イメージの変更

用紙制御バッファーにロードする前に、仮想ストレージ内の FCB イメージを変更するには、以下の順序でマクロ命令を使用して、FCB イメージを仮想ストレージに読み取ります。

1. IMGLIB マクロ命令を使用して、OPEN パラメーターを指定する。
2. BLDL マクロ命令を使用して、FCB イメージがイメージ・ライブラリー内に存在するかどうかを判別する。
3. LOAD マクロ命令を使用して、イメージを仮想ストレージにロードする。
4. イメージを読み取った後、CLOSE パラメーターと最初の IMGLIB マクロで作成された DCB のアドレスを指定して、IMGLIB マクロ命令を発行する。

SETPRT マクロ命令を使用すると、用紙制御バッファーに変更イメージをロードできます。3800 以外のプリンターは、出口リスト内の FCB 項目を使用する必要があります (これについては、「z/OS DFSMS データ・セットの使用法」で説明しています)。

BLDL および SETPRT マクロのフォーマットは、「z/OS DFSMS Macro Instructions for Data Sets」に記載されています。LOAD マクロのフォーマットは、「z/OS MVS プログラミング：アセンブラー・サービス ガイド」および「z/OS MVS プログラミング：アセンブラー・サービス 解説書 ABE-HSP」に記載されています。

IMGLIB マクロのフォーマットは、次のとおりです。



- OPEN** SYS1.IMAGELIB をオープンし、その DCB を作成することを指定します。DCB のアドレスは、レジスター 1 に戻されます。
- CLOSE** SYS1.IMAGELIB をクローズします。
- addr* DCB を指すワードの RX タイプ・アドレスを指定します。(reg) 形式でコーディングした場合、括弧内のレジスターには、ワードのアドレスではなく、DCB のアドレスが入ります。

IMGLIB からの戻りコード

OPEN パラメーターを指定した IMGLIB マクロからの戻りコードは、以下のとおりです。

戻りコード	意味
0 (X'00')	操作が正常に行われました。
4 (X'04')	SYS1.IMAGELIB を含むボリュームがマウントされていないか、必要なカタログ・ボリュームがマウントされていないかのいずれかです。
8 (X'08')	SYS1.IMAGELIB がカタログの指すボリューム上に存在しないか、または SYS1.IMAGELIB がカタログされていません。
12 (X'0C')	カタログまたは VTOC の読み取り中にエラーが発生しました。

IMGLIB マクロによって作成された DCB を参照できるのは、BLDL マクロと LOAD マクロだけです。

3211 指標付け機能に対する JES サポート

JES2 は、3211 指標付け機能を 2 とおりの方法でサポートしています。

- /*OUTPUT カードでの INDEX パラメーターの指定
- 拡張 FCB イメージ

システムは 2 つの特殊 FCB を提供します。6 行/インチ用の FCB26 と 8 行/インチ用の FCB28 です (それぞれ、FCB=6 および FCB=8 として指定します)。これらの FCB は、チャンネル 1 標識がイメージの位置 1、特殊標識フラグが 3 番目のバイト、1 インチ当たりの行数が 4 番目のバイトに入ります。

FCB26 と FCB28 の 3 番目のバイトの特殊指標フラグには、X'80' と、1 から 32 の範囲 (デフォルト=1) の 2 進数の指標値が入ります。指標値は左マージンを設定します (1 は、左揃えを示します。その他の値は、印刷行を N-1 位置だけ字下げします)。

JES2 が他の FCB イメージを使用する場合、イメージはチャンネル 1 を位置 1 に指定する必要があります。そうしないと、JES2 は用紙をプリンターに正しく置きません。(STD1 と STD2 はチャンネル 1 を位置 1 に指定しないため、JES2 用に変更する必要があります。)

他の FCB イメージの 3 番目のバイト (1 インチ当たりの行数を指定) に X'80' 以外のデータ文字が入っている場合、JES2 はその仕様を使用し、指標値 1 を提供します。

JES3 は 3211 指標付け機能をサポートせず、JES3 からの指標付けコマンドは 3203 モデル 5 によって無視されます。

第 12 章 階層ファイル・システム・データ・セット

階層ファイル・システム (HFS) データ・セットは、POSIX 準拠のファイル・システムを含むデータ・セットです。これは階層構造で編成されたファイルおよびディレクトリーの集合で、z/OS UNIX システム・サービスを使用してアクセスできます。多くの標準 BSAM、QSAM、BPAM、および VSAM インターフェースを使用して、z/OS UNIX HFS ファイル内のデータにアクセスできます。これらのアクセス方式を使用するほとんどのアプリケーションは、再アセンブリーまたは再コンパイルせずに HFS ファイルにアクセスできます。

HFS データ・セットの内容は、1 つのルート・ディレクトリーとさまざまなサブディレクトリーを基にしたツリーのような構造になっています。HFS データ・セット内のファイルは、パスとファイル名によって識別されます。

DFSMSHsm は、DFSMSdss をデータ移動プログラムとして使用している場合は HFS データ・セットを自動的にバックアップできますが、HFS データ・セット内の個々のファイルをバックアップすることはできません。

z/OS UNIX を評価し、企業にインストールすることに決めた後、以下の計画作業を進めることができます。

- HFS データ・セットを格納する DASD 装置を決める。
- それに対するアクセスを制御する方法を決める。
- ファイル・システムを構造化する。
- バックアップ、復元、および有効期限に関する方針を決定する。
- HFS の命名方針を決定する (ファイル名は最大 255 文字の長さ、パス名は最大 1023 文字の長さにできます)
- データ・クラス、管理クラスを定義し、HFS データ・セットの ACS ルーチンおよび JCL ステートメントをコーディングする。

階層ファイル・システム・データ・セットの作成

DSNTYPE パラメーターで HFS を指定して、HFS データ・セットを割り振ります。HFS データ・セットのデータ・クラスも定義できます。カタログおよびアンカカタログ HFS データ・セットの両方が単一の非 SMS 管理ボリューム上に常駐できます。カタログ HFS データ・セットは、ストレージ管理サブシステム管理ボリューム上にある場合は、マルチボリュームにすることもできます。このタイプのデータ・セットは、複数のボリューム上の 255 エクステンツの直接アクセス・ストレージ・デバイス (DASD) スペースに拡張できます (最大 59 ボリューム、ボリューム当たり 123 エクステンツ)。

z/OS UNIX または HFS データ・セットを使用するには、RACF または同等のセキュリティ製品がシステムにインストールされ、アクティブであることが必要です。z/OS UNIX は、ユーザー ID とファイル・アクセス制御情報を検査することにより、システム・セキュリティを保守します。

ルート・ファイル・システムの定義

z/OS UNIX のインストール時に、システム・プログラマーまたはストレージ管理者は、SYS1.PARMLIB の BPXPRMxx メンバーに ROOT ステートメントをコーディングします。これは、z/OS UNIX システム・サービスを開始するときにシステムが論理的にマウントするルート・ファイル・システムを含んでいる HFS データ・セットを識別します。

```
ROOT    FILESYSTEM('OMVS.ROOT')
        TYPE(HFS)
        MODE(RDWR)
```

ルート・ファイル・システムは、ファイル構造全体の開始点です。これは、ルート・ディレクトリーと関連の HFS ファイルまたはサブディレクトリーから構成されます。インストールの完了後、MVS システムの実行中に、ルート・ファイル・システムを入れる HFS データ・セットを作成する (割り振る) ことができます。

メンバー BPXPRMxx については、「z/OS MVS 初期設定およびチューニング 解説書」を参照してください。

ルート・ファイル・システムの作成とマウント

データ・セットを割り振るジョブを実行して、ルート・ファイル・システムを格納する HFS データ・セットを作成します。割り振り時に、z/OS UNIX は基本ルート・ディレクトリーを作成しますが、ユーザーの特定の要件に合わせてそれを変更できます。HFS データ・セットを指定するには、DSNTYPE=HFS を指定します。

```
//STEP1 EXEC PGM=IEFB14
//MKFS   DD  DSNAME=OMVS.ROOT,DISP=(NEW,CATLG),
//        SPACE=(CYL,1)
//        DSNTYPE=HFS,SPACE=(CYL,(1,1,1))
```

ヒント: HFS データ・セットを SMS 管理にするには、ACS ルーチンを使用するか、または JCL で STORCLAS パラメーターを指定します。HFS データ・セットは SMS 管理である必要はありません。

書き込み操作は、ファイル・システムを損傷する危険性が最大です。この理由から、ルート・ファイル・システムは小さく保ち、それに対する書き込み活動を最小化して、損傷の危険を最小限にとどめることが必要です。

また、ユーザーのファイルはすべてファイル・システムに入れて、それをルート・ファイル・システム上にマウントするようにして (ルート自体の一部として定義するのではなく)、ユーザーのルートへの書き込みアクセスを拒否すれば、ルートを不注意による損傷からさらに保護できます。z/OS UNIX システム・サービスに障害を起こさずに、損傷したユーザー・ディレクトリーまたはファイルをアンマウントして交換することができます。

追加のファイル・システムおよびディレクトリーの作成

ルート・ファイル・システムに対して HFS データ・セットを割り振り、TSO/E ユーザーとしてログオンした後、MKDIR コマンドを使用して、ルート・ファイル・システム内に追加ディレクトリーを定義できます。例えば、/u/joe ディレクトリーを作成するには、以下を発行します。

```
MKDIR '/u'
MKDIR '/u/joe'
```


このディレクトリーは、追加のマウント可能ファイル・システムのマウント・ポイントとして使用できます。IBM 提供のプログラムを使用すると、ディレクトリー、疑似 TTY ペア、および装置ファイルを作成できます。その後で、対話式のユーザー・プログラムやアプリケーション・プログラムを使用して、これらの追加ファイル・システムにファイルを追加できます。

MKDIR コマンドは、作成するディレクトリーの親ディレクトリーに対する書き込み許可を必要とします。

ルート・ファイル・システムへのファイル・システムの追加とマウント

ユーザーが適切な権限を持っている場合、専用のディレクトリーとデータ・ファイル構造を持つ別のマウント可能ファイル・システムを作成し、それをルート・ファイル・システム内または別のファイル・システム内のディレクトリーに置くことができます。各ファイル・システムは、TSO/E MOUNT コマンドを使用して、別のファイル・システム内のディレクトリー (マウント・ポイント) に論理的にマウントできます。ファイル・システムをアンマウントするには、UNMOUNT コマンドを使用します。追加ファイル・システムを作成してマウントするには、次の手順で行います。

1. TSO/E ALLOCATE コマンド (下記の例を参照) または JCL DD ステートメント (352 ページの『ルート・ファイル・システムの作成とマウント』を参照) を使用して、追加の HFS データ・セットを割り振ります。

```
ALLOCATE DSNAME('OMVS.USER.JOE') NEW DSNTYPE(HFS) BLKSIZE(0)
LRECL(0) RECFM(U) DSORG(PO) SPACE(1,1) CYLINDERS
```

空のルート・ディレクトリーを持つ新規データ・セットが割り振られます。

2. 許可ユーザーが TSO/E MOUNT コマンドを入力して、新規ファイル・システムを既存のファイル・システム内のディレクトリーに論理的にマウントします。

```
MOUNT FILESYSTEM('OMVS.USER.JOE') TYPE(HFS) MOUNTPOINT('/u/joe')
```

ユーザーは、SYS1.PARMLIB の BPXPRMxx メンバーに MOUNT コマンドを追加することにより、z/OS UNIX を開始するたびに自動的に追加ファイル・システムを論理的にマウントすることを指定できます。ファイル・システムのマウントには、以下の制約事項が適用されます。

- マウント・ポイントはディレクトリーでなければならない。
- ファイル・システムがマウントされている期間中は、ディレクトリー内のファイルはアクセス不能である。
- マウント・ポイントでは同時に 1 つのマウントのみをアクティブにできる。
- ファイル・システムは同時に 1 つのディレクトリーにのみマウントできる。

以下のタスクを実行するための特殊 HFS ファイルも作成できます。

- ハードウェア装置を表す (文字特殊ファイル)。
- HFS ファイルに対して別名の使用を許可する (シンボリック・リンク)。
- 1 つのプロセスから別のプロセスにデータを送信し、受信側プロセスは先入れ先出し法でデータを読み取る (FIFO 特殊ファイル、名前付きパイプとも呼ばれます)。ここで使用されているプロセスという用語は、fork 機能によって作成されるプログラム、または z/OS UNIX システム・サービスを要求するプログラムのいずれかとして定義されます。

ファイル・システム・サイズの管理

ユーザーがファイルを追加したり、既存のファイルを拡張したりするにつれて、ファイル・システムのサイズが大きくなります。最終的に、ファイル・システムはボリューム上のスペースを超えてしまうことがあります。この場合、HFS データ・セットに責任を持つストレージ管理者またはシステム・プログラマーは、個々の HFS ファイルを、スペースが使用可能な別のファイル・システムに移動して、ボリューム上でより多くのスペースを使用できるようにするか、あるいは以下のいずれかを行うことができます。

- ファイル・システム全体を別のボリューム・セットに移動する。これは次の手順で行います。
 1. 許可ユーザーが TSO/E UNMOUNT コマンドを入力して、ファイル・システムを論理的にアンマウントする。
 2. 異なるデータ・セット名を持つ HFS データ・セットを、適切なスペースが使用可能なボリュームまたはボリューム・セットに割り振る。
 3. DFSMSdss DUMP 機能を使用して、古いファイル・システムを論理的にダンプする。
 4. DFSMSdss RESTORE 機能を使用して、ダンプされたファイル・システムを、十分なスペースを持つボリュームまたはボリューム・セットに、新しい名前で復元する。元のファイル・システム名を維持したい場合は、最初に既存のファイル・システムを削除してから、DFSMSdss を使用して、名前変更せずにそれを復元します。
- 削除するか、別のファイル・システムに移動するかのいずれかの方法で、ファイル・システムからファイルを除去する。選択されたファイルをファイル・システム内の特定のディレクトリーから除去することが不可能な場合、同じファイル・システム内の異なるディレクトリーから別のファイルを除去することは可能なことがあります。目的は、ファイル・システムのサイズを削減することです。
- 別のボリュームまたはボリューム・セット上に新規ファイル・システムを作成し、満杯のファイル・システムから一部のファイルを新規ファイル・システムに移動する。この方法の結果として生じる可能性がある問題を避けるために、元の名前を使用してシンボリック・リンクを定義してください。
- IDCAMS ALTER ADDVOLUMES コマンドを使用して、ファイル・システム候補ボリューム・リストに別のボリュームを追加する。追加ボリュームを HFS が使用できるようにするには、ファイル・システムをアンマウントし、再マウントする必要があります。
- パラメーター FSFULL を指定して HFS をマウントすると、ストレージ管理者またはシステム・プログラマーは、ファイル・システム内のスペースをモニターできます。例えば、mount parm('FSFULL(70,10)') を指定すると、HFS はファイル・システムが 70 % 満杯になった時点でメッセージ IGW023A を発行し、ファイル・システムが 80 % および 90 % 満杯になった時点で追加の IGW023I メッセージを発行します。

BPX1PCT 呼び出し可能サービスを使用して、ファイル・システムを拡張できます (359 ページの『HFS に対する pfscpl (BPX1PCT) 物理ファイル・システム制御の使用』を参照してください)。

ファイル・システム・アクティビティの管理

ファイル・システムのアクティビティが過剰になってアクセスが低速になった場合、次の作業のいずれかを行うことができます。

- ファイル・システムをより高速で入出力処理できるボリューム (例えば、より高速のチャンネルまたはキャッシュ・ストレージ制御を備えたボリューム) に移動する。
- サブツリーを満杯のファイル・システムから別のボリューム上の新規ファイル・システムに移動する。サブツリーのヘッドにあった空になったディレクトリーに、新規ファイル・システムをマウントします。これにより、入出力アクティビティが 2 つのボリューム間に分割されます。この処置による障害を避けるために、元の名前を使用してシンボリック・リンクを定義してください。

HFS データ・セット属性へのアクセス

HFS ファイル・システムのサイズ情報を要求するアプリケーション (例えば、対話式システム生産性向上機能 (ISPF) オプション 3.4I または DCOLLECT) は、DFSMS/MVS 1.5 現在、その要求にサービスするには OMVS が必要です。これは、ユーザー ID が OMVS に対して定義され、データ・セットにアクセスするためのリソース・アクセス管理機能 (RACF) 権限を持っている必要があることを意味します。HFS ファイル・システムは、このシステムにマウント済みであるか、またはマウント可能であること (つまり、すでに別のシステムに READ/WRITE マウントされていないこと) が必要です。

ファイル・システムのトランスポート

物理的に別の場所に移送できるストレージ・メディアにデータ・セットをコピーすることが必要な場合があります。それを行うには、以下の作業のいずれかを行います。

- PAX、CPIO、または TAR シェル・コマンドを使用して、テープ・アーカイブ (TAR) フォーマットでファイル・システムをコピーする。
- 許可ユーザーがファイル・システムを論理的にアンマウントし、異なるデータ・セット名を持つ HFS データ・セットを割り振り、DFSMSdss ダンプ・ユーティリティーを使用して、古いファイル・システムを新規データ・セットにコピーする。

ファイル・システムの除去 (削除)

ファイル・システム (HFS データ・セット) を別のファイル・システムにマウントする場合、またはそれを削除する場合、最初にそれを含んでいる HFS データ・セットに対して TSO/E UNMOUNT コマンドを使用して、論理的にアンマウントします。その後で、MOUNT コマンドの一方または両方を使用できます (353 ページの『ルート・ファイル・システムへのファイル・システムの追加とマウント』を参照してください)。

以下のいずれかの手法を使用して、ファイル・システムを除去できます。

- DELETE コマンドを使用する (IDCAMS または TSO)。
- HFS データ・セットに対して DISP=(OLD,DELETE) を指定して IEFBR14 ジョブを実行する。
- ISPF オプション 3.x。
- IEHPROGM の SCRATCH および UNCATLG コマンドを使用する。

ファイル・システムのマイグレーション

ファイル・システムがアンマウントされ、事前定義された時間その状態のままである場合、システムはそれを優先順位の低いストレージ・メディアにマイグレーションすることができます。ファイル・システムに対してマウント・コマンドが発行された場合、システムは自動的にマイグレーション・ストレージからマイグレーション済みファイルを再呼び出しします。

HFS データ・セットのマイグレーションを計画する場合、レベル 1 (DASD) ストレージにのみマイグレーションすることを考慮してください。テープにマイグレーションされたデータ・セットを再呼び出しすると、テープ・ボリュームを物理的にマウントする時間が必要になるため、パフォーマンスに悪影響を与えることがあります。

テープが自動ライブラリー内であれば、システムがオペレーターにマウントを要求する場合に比べて、再呼び出しがはるかに高速になります。

ファイル・システムのバックアップ

DFSMSHsm は、HFS データ・セットの自動バックアップ機能を備えています。定期的に HFS データ・セットのバックアップを取ることによって、その中に含まれているマウント可能ファイル・システムをバックアップすることができます。必要な場合、データ・セットを復元できます。DFSMSHsm は、アンマウントされたファイル・システムのマイグレーションおよび復元にも使用されます。

DFSMSdss データ・セット・ダンプを使用して、定期的にマウント可能ファイル・システム (ルート・ファイル・システムを含む) のバックアップを手動で取ることができます。これを行うには、DFSMSdss DUMP コマンドを発行します。これは、指定の HFS データ・セットに対するアクティビティを静止した後、バックアップ処理を呼び出します。バックアップが完了すると、ファイル・システムは静止解除され、ユーザーのアクティビティを再開できます。

プログラムに障害が起きてファイルやディレクトリーが損傷した場合に備えて、ファイル・システムの DFSMSdss データ・セット・ダンプを定期的に保存してください。このバックアップは、コンピューター・サイトが物理的損傷を受けた場合を考慮して、別の区域または異なる建物内に保管してください。

HFS データ・セット内の個別ファイルのバックアップを自動的に取る機能はありません。PAX、CPIO、および TAR コマンドを使用して、手動でファイルのバックアップを取ることができます。

DFSMSdss の使用について詳しくは、「z/OS DFSMSdss ストレージ管理リファレンス」を参照してください。

バックアップ・ファイル・システムのリカバリー

ファイル・システムが損傷した場合、以前のバックアップから作成された保管バージョンで置き換えることによってリカバリーできます。バックアップ・ファイル・システムをリカバリーするには、以下の作業を行う必要があります。

1. すべてのユーザーに対して、損傷したファイル・システム上のすべてのアクティビティーを停止するように通知する。
2. 許可ユーザーが IMMEDIATE オプションを指定して TSO/E UNMOUNT コマンドを入力し、損傷したファイル・システムを論理的にアンマウントする。アンマウントに失敗した場合は、FORCE オプションを使用して UNMOUNT コマンドを再入力します。
3. DFSMSdss ダンプ・ユーティリティーを使用して、バックアップ・ファイル・システムを置換ファイル・システム (HFS データ・セット) に復元する。これは前のステップと同時に行うことができます。
4. 許可ユーザーが TSO/E MOUNT コマンドを入力して、置換ファイル・システムを論理的にマウントする。SYS1.PARMLIB の BPXPRMxx メンバー内の MOUNT コマンドが、この MOUNT コマンドと整合していることを確認してください。
5. すべてのユーザーにブロードキャスト・メッセージを発行して、またはシェルを呼び出すときにすべての z/OS UNIX ユーザーにメッセージを発行して、バックレベルのファイル・システムをマウントしたことを知らせ、そのマウント・ポイントを通知する。ユーザーは、ファイル・システムのバックアップが取られた以降に追加されたファイルを再作成または追加する必要があります。

ルート・ファイル・システムのバックアップ・ファイル・システムをリカバリーする場合は、より破壊的な方法を選択できる場合があります。シャットダウンの実行方法について詳しくは、「z/OS UNIX システム・サービス 計画」の『z/OS UNIX のシャットダウン』を参照してください。

HFS 据え置きファイル・システム同期

通常の HFS ディスク強化は、定期的に稼働する同期デーモンのもとで実行され、同期デーモンが最後に実行された以降に発生したファイルおよびメタデータの変更をすべて書き出します。ファイル・システムの同期インターバル間の時間帯に多数のファイルが作成、変更、または削除されている場合には、次のファイル・システム同期の際に、これらのファイル変更を集めてディスク・バージョンのファイ

ル・システムを同期することにより、メディア・マネージャーの呼び出し回数を大幅に削減します。これによって、実際の入出力アクティビティが削減されます。

デフォルトの同期デーモン・インターバルは 1 分間であるため、入出力の削減はパフォーマンスに著しい効果をもたらします。同期デーモンは OMVS アドレス・スペースで実行され、ファイル・システムに対するユーザー要求からは独立しています。

同期デーモンを実行すると、すべての HFS 変更をまとめて 1 つの大きい入出力要求のバッチとし、それをメディア・マネージャーに渡します。その意図は、ディスク上の HFS に対して 1 つの長い入出力操作を実行することにあります。ただし、HFS が 1 つのボリューム上にあり、単一のエクステント内に格納されているという最適条件の場合でも、ディスク強化のための入出力操作では、メディア・マネージャーを数回呼び出す必要があります。HFS が複数のエクステントを持っているか、複数のボリューム上に常駐する場合、単一の入出力操作を複数のエクステントに及ぼすことはできないので、複数のチャンネル・プログラムを作成することが必要になります。

同期デーモン・インターバルに加えて、HFS は個別ファイルの同期 (fsync) もサポートします。fsync は、任意の時点にアプリケーション・プログラムによって実行できます。ただし、fsync は結果的にファイル・システム全体を同期することを知っておいてください。HFS ファイル・システムの同期操作中は現在 HFS にアクセスしているユーザーから独立して同期タスクが実行されることに注意することが非常に重要です。同期タスクを実行する場合、同期するファイル・システムに対して排他ラッチが取得されます。このラッチは、ファイル・システムの同期操作の期間中、保持されます。このラッチが保持されている間は、HFS ファイル・システムの UNIX システム・サービス・ユーザーは HFS にアクセスできません。同期タスクが実行された場合、HFS は以下を行う必要があります。

1. 新規または変更されたファイルのメタデータをすべて更新する。
2. 除去されたファイルの構造を削除する。
3. HFS 属性ディレクトリー内の内部 HFS ストレージ・マップを更新する。
4. メディア・マネージャーを呼び出して、ディスク・バージョンの HFS データ・セットを変更する入出力操作を実行する。

入出力操作は同期です。これは、メディア・マネージャー呼び出しの期間中、HFS 同期タスクが中断されることを意味します。通常は、このファイル・システム同期ロックアウト状態は短く、わずか数秒続くに過ぎないため、エンド・ユーザーが気付くことはめったにありません。ただし、状況によっては、ファイル・システム同期ロックアウトがより長く続くことがあります。ロックアウトの所要時間は、同期インターバル、SYNC タスクが実行する必要がある作業量、および入出力サブシステムのパフォーマンスによって異なります。

完了までに長時間かかるファイル・システム同期操作がアプリケーションに与える影響を最小化するために、ユーザーは以下の 1 つ以上を行うことができます。

1. 影響を受ける HFS は、より短い同期インターバルを指定してマウントする。これにより全体的な HFS パフォーマンスは低下しますが、同期ロックアウトの長さは短縮されます。
2. HFS を複数の小さい HFS データ・セットに分割する。

3. 大きいディレクトリーに対して「rm -R」のような影響の大きいコマンドの使用を避ける。

SYNC 値の指定方法

デフォルトの HFS SYNC デモン・インターバル値は 60 秒です。

SYS1.PARMLIB の BPXPRM_{xx} メンバー内の FILESYSTYPE パラメーターの SYNCDEFAULT オプションを変更することにより、このデフォルトをオーバーライドできます。個々の HFS データ・セットがオーバーライド SYNC 値を持つことができます。これは、ROOT パラメーターで SYNC 値を指定するか、または SYS1.PARMLIB の BPXPRM_{xx} メンバー内の MOUNT パラメーターで SYNC 値を指定することによって達成できます。

FILESYSTYPE、ROOT、および MOUNT パラメーターの指定方法については詳しくは、「*z/OS MVS 初期設定およびチューニング 解説書, SA88-8564*」を参照してください。また、SYNC 値を指定した MOUNT コマンドを使用して、個々の HFS データ・セットをユーザーがマウントすることもできます。

HFS に対する pfsctl (BPX1PCT) 物理ファイル・システム制御の使用

以下の情報は、HFS に対する pfsctl (BPX1PCT) 呼び出し可能サービスの使用について説明しています。

pfsctl 呼び出し可能サービス (BPX1PCT) は、コマンドと引数を、物理ファイル・システムに伝えます。コマンドと引数の意味は、物理ファイル・システムに固有であり、物理ファイル・システムによって定義されます。

フォーマット

```
CALL BPX1PCT,(File_System_type,
              Command,
              Argument_length,
              Argument,
              Return_value,
              Return_code,
              Reason_code)
```

File_System_type:

Type:	Character string
Length:	8 bytes
Character string:	HFS

Command:

Type:	Integer
Length:	Fullword

物理ファイル・システムが HFS データ・セットの場合、以下の値のいずれかをコーディングできます。

X'40000001'	DisplayBufferLimits
X'00000002'	ChangeBufferLimits
X'40000003'	DisplayGlobalStats
X'40000004'	DisplayFSStats
X'00000005'	ExtendFS

これらの HFS 定義コマンドとそれぞれのデータ域は、SYS1.MODGEN 内にある GFUMPCTL で定義されています。

Argument_Length: 4 バイト整数。有効な値は、以下を参照してください。

Argument: 以下で説明する構造。365 ページの表 85、365 ページの表 86、366 ページの表 87、または 367 ページの表 88 に示されています。

Return Value: フルワード

Return_Code: BPX1PCT が 363 ページの表 84 で説明している値のいずれかに設定する 4 バイト整数。

Reason_Code: BPX1PCT が 363 ページの表 84 で説明している値のいずれかに設定する 4 バイト整数。

物理ファイル・システムが HFS データ・セットの場合、以下のセクションが適用されます。

DisplayBufferLimits コマンド

機能: DisplayBufferLimits コマンドは、データ域のストレージ限度 (具体的には、現在有効な HFS バッファの VMAX および FMIN 値) を戻します。

コマンド: X'40000001'

引数の長さ: 引数の長さは、データ域の長さです。データ域は、少なくとも PCTL_BFRLIMITS_TYPE の長さでなければなりません。365 ページの表 85 を参照してください。

引数: PCTL_BFRLIMITS_TYPE

使用上の注意:

1. 呼び出し側は許可を得ている必要はありません。
2. VMAX 値と FMIN 値は、両方ともメガバイト (MB) 単位です。
3. このコマンドは、confighfs シェル・コマンドと等価です。

```
confighfs -l
```

ChangeBufferLimits コマンド

機能: ChangeBufferLimits コマンドは、データ域に要求された HFS バッファのストレージ限度 (具体的には、VMAX 値と FMIN 値) を変更します。

コマンド: X'00000002'

引数の長さ: 引数の長さは、データ域の長さです。データ域は、少なくとも PCTL_BFRLIMITS_TYPE の長さでなければなりません。365 ページの表 85 を参照してください。

引数: PCTL_BFRLIMITS_TYPE

使用上の注意:

- 以下のフィールドは、この構造内で設定する必要があります。
 - PCTL_BL_ACTION フラグを設定する必要があります。
 - PCTL_BL_VMAX フラグを設定し、フィールド PCTL_BL_VMAX_VAL に新規 VMAX 値を指定して、VMAX を変更します。
 - PCTL_BL_FMIN フラグを設定し、フィールド PCTL_BL_FMIN_VAL に新規 FMIN 値を指定して、FMIN を変更します。
- 呼び出し側はスーパーユーザーでなければなりません。
- VMAX と FMIN の両方を変更する要求は、1 回の呼び出しで達成できます。
- VMAX 値と FMIN 値はメガバイト (MB) 単位です。
- このコマンドは、confighfs シェル・コマンドと等価です。

```
confighfs -v n
```

上記のコマンドは、最大仮想ストレージ (VMAX) を n に設定します。ここで、n は MB 単位です。

```
confighfs -f n
```

上記のコマンドは、最小仮想ストレージ (FMIN) を n に設定します。ここで、n は MB 単位です。

DisplayGlobalStats コマンド

機能: DisplayGlobalStats コマンドは、データ域の HFS グローバル・システム・レベル統計を戻します。

コマンド: X'40000003'

引数の長さ: 引数の長さは、データ域の長さです。データ域は、少なくとも PCTL_GLOBALSTATS_TYPE の長さでなければなりません。365 ページの表 86 を参照してください。

引数: PCTL_GLOBALSTATS_TYPE

使用上の注意:

- 呼び出し側は許可を得ている必要はありません。
- このコマンドは、次のコマンドと等価です。

```
confighfs -q
```

DisplayFSStats コマンド

機能: DisplayFSStats コマンドは、データ域の HFS ファイル・システム・レベル統計を戻します。ファイ

ル・システム・レベル統計で戻される値は、指定されたファイル・システムの現行マウントに関連しています。

コマンド: X'40000004'

引数の長さ: 引数の長さは、データ域の長さです。データ域は PCTL_FSSTATS_TYPE の長さでなければなりません。366 ページの表 87 を参照してください。

引数: PCTL_FSSTATS_TYPE

使用上の注意:

1. ファイル・システム名は、PCTL_FSSTATS_TYPE 構造内のフィールド PCTL_FS_NAME に入力として提供する必要があります。
2. 呼び出し側は許可を得ている必要はありません。
3. このコマンドは、confighfs シェル・コマンドと等価です。

confighfs pathname

ここで、pathname は、HFS を識別する絶対または相対パス名です。

ExtendFS コマンド

機能: ExtendFS コマンドは、データ域の指定に従って、指定されたファイル・システムを要求された量だけ拡張する (追加のディスク・スペースを割り振る) ことを試みます。

コマンド: X'00000005'

引数の長さ: 引数の長さは、データ域の長さです。データ域は、少なくとも PCTL_EXTENDFS_TYPE の長さでなければなりません。367 ページの表 88 を参照してください。

引数: PCTL_EXTENDFS_TYPE

使用上の注意:

- 以下のフィールドは、この構造内で設定する必要があります。
 - PCTL_EXT_FSNAME は、拡張するファイル・システムの名前に設定します。
 - 要求が新規ボリューム上にスペースを割り振ることである場合は、PCTL_EXT_NEW_VOL フラグをオンに設定します。
 - PCTL_EXT_UNIT は、以下の拡張単位のいずれかに設定します。
 1. ExtendFS_UNIT_MB は、拡張する量はメガバイト (MB) 単位であることを示します。
 2. ExtendFS_UNIT_TRK は、拡張する量はトラック数であることを示します。
 3. ExtendFS_UNIT_CYL は、拡張する量はシリンダー数であることを示します。
 - PCTL_EXT_AMT は、拡張する量に設定します。
 - 呼び出し側はスーパーユーザーでなければなりません。

- フラグ PCTL_EXT_NEW_VOL = オフの場合、追加スペースを取得する試みは、HFS の最後のボリューム上でのみ行われます。HFS は新規ボリュームには拡張されません。
- フラグ PCTL_EXT_NEW_VOL = オンの場合、追加スペースを取得する試みは、新規ボリューム上でのみ行われます。この場合、HFS 用の候補ボリュームが存在する必要があるため、その候補ボリュームは HFS の現行マウント時に存在していなければなりません。
- HFS 用の 2 次スペース量が存在する場合、それは無視されます。
- 要求された全体量を取得できない場合、操作は失敗します。
- このコマンドは、confighfs シェル・コマンドと等価です。

confighfs -x size pathname	同一ボリューム上で拡張する場合
confighfs -xn size pathname	新規のボリュームに拡張する場合

ここで、size は、拡張単位 M (メガバイト)、T (トラック数)、または C (シリンダー数) の接尾部を持つ拡張量を示し、pathname は、HFS を識別する絶対または相対パス名を示します。

HFS に対して BPX1PCT を使用すると、戻り値が -1 の場合にのみ、非ゼロの戻りコードと理由コードを戻します。表 84 は、HFS DisplayBufferLimits、ChangeBufferLimits、DisplayGlobalStats、DisplayFSStats、および ExtendFS コマンドの BPX1PCT によって戻される戻りコードと理由コードのリストを示しています。

表 84. BPX1PCT - 戻りコードと理由コード

戻りコード	理由コード	説明	コマンド
0	0	正常終了	
EINVAL	5B360106	無効な PCTL_EXT_UNIT パラメーターを指定して、HFS pfscctl サービス ExtendFS 機能が呼び出されました。	ExtendFS のみ
	5B360108	ゼロの拡張値を指定して、HFS pfscctl ExtendFS 機能が呼び出されました。	ExtendFS のみ
EFAULT	5B360102	サポートされないコマンド・コードを使用して、HFS pfscctl サービスが呼び出されました。	
	5B360103	HFS pfscctl コマンドで提供されたデータ域のアドレスがゼロです。	
	5B360107	無許可の呼び出し側によって、HFS pfscctl サービスの ChangeBufferLimits または ExtendFS 機能が呼び出されました。	

表 84. BPX1PCT - 戻りコードと理由コード (続き)

戻りコード	理由コード	説明	コマンド
EMVSERR	5B360101	ローカル・リカバリー・ルーチン追加機能の実行中にエラーが発生しました。	
ENOBUFS	5B360104	HFS pfscctl コマンドで提供されたデータ域の長さが、要求を完了させるために必要な最小サイズより小さくなっていました。	
ENOENT	5B360105	アンマウントされた HFS.unted を指定して、HFS pfscctl サービスが呼び出されました。	DisplayFSStats および ExtendFS のみ。
ENOSPC	5B27C005	DADSM エラーが発生しました。HFS データ・セットを拡張するために使用可能なスペースがありません。	ExtendFS のみ
	>80000000	ファイル・システムが HFS スペース不足エラー状態にあり、拡張量が同期シャドー書き込みを完了させるための要件を満たすには不十分でした。理由コードに、必要な追加トラック数の 2 の補数が入っています。	ExtendFS のみ
EIO	5B27C00A	DADSM エラーが発生しました。入出力エラー VTOC。	ExtendFS のみ
EMVSPFSPERM	5B27xxxx	予期しないエラーが発生しました。	ExtendFS のみ
	5BB3xxxx	予期しないエラーが発生しました。	ChangeBufferLimits のみ
EMVSPARM	5B36000E	VMAX が最小を下回っています (VMAX は最小に設定されます)	ChangeBufferLimits のみ
	5B36000F	FMIN が VMAX を超えています (FMIN は VMAX に設定されます)	ChangeBufferLimits のみ
	5B360010	FMIN は使用可能な最大です (FMIN は最大に設定されます)	ChangeBufferLimits のみ
	5B360014	VMAX が FMIN を下回っています (VMAX は FMIN に設定されます)	ChangeBufferLimits のみ

365 ページの表 85 は、DisplayBufferLimits および ChangeBufferLimits コマンドの GFUMPCTL 構造を示しています。

表 85. *DisplayBufferLimits* および *ChangeBufferLimits* コマンドの構造 (GFUMPCTL)

オフセット	タイプ	長さ	名前	説明
0 (0)	STRUCTURE	28	PCTL_BFRLIMITS_TYPE	
0 (0)	CHARACTER	1	PCTL_BL_FLGS	フラグ
	1... ..	1	PCTL_BL_ACTION	オフ: ディスプレイ・バッファー限度
				オン: 変更バッファー限度
1 (1)	CHARACTER	1	PCTL_BL_LIMIT	変更する限度
	1... ..		PCTL_BL_VMAX	VMAX 変更要求
	.1.. ..		PCTL_BL_FMIN	FMIN 変更要求
2 (2)	CHARACTER	1	*	予約済み
4 (4)	SIGNED	4	PCTL_BL_VMAX_VAL	VMAX 変更値
8 (8)	SIGNED	4	PCTL_BL_FMIN_VAL	FMIN 変更値
12 (C)	CHARACTER	16	*	予約済み
28 (1C)	CHARACTER		PCTL_BL_END	

表 86 は、*DisplayGlobalStats* コマンドの GFUMPCTL 構造を示しています。

表 86. *DisplayGlobalStats* コマンドの構造 (GFUMPCTL)

オフセット	タイプ	長さ	名前	説明
0 (0)	(0) STRUCTURE	200	PCTL_GLOBALSTATS_TYPE	
0 (0)	CHARACTER	8	PCTL_GS_CURR_TIME	現在のタイム・スタンプ
8 (8)	CHARACTER	8	*	予約済み
16 (10)	SIGNED	4	PCTL_GS_TOTVIRT	すべてのバッファー・プールによって使用中のものを合わせた合計仮想 (ページ数)
20 (14)	SIGNED	4	PCTL_GS_TOTFIX	すべてのバッファー・プールによって使用中のものを合わせた合計固定 (ページ数)
24 (18)	CHARACTER	28	PCTL_GS_BP1	バッファー・プール 1 の統計
24 (18)	UNSIGNED	2	PCTL_GS_BP1_BFRSIZE	各バッファーのサイズ (ページ数)
26 (1A)	UNSIGNED	2	PCTL_GS_BP1_DSCNT	プール内のデータ・スペースの数
28 (1C)	SIGNED	4	PCTL_GS_BP1_TOTVIRT	このプールによって使用された合計仮想
32 (20)	SIGNED	4	PCTL_GS_BP1_TOTFIX	このプールによって使用された合計固定
36 (24)	UNSIGNED	8	PCTL_GS_BP1_FIXD_Y	入出力要求の前にすでにバッファーが固定されていた回数
44 (2C)	UNSIGNED	8	PCTL_GS_BP1_FIXD_N	入出力要求の前にすでにバッファーが固定されていなかった回数
52 (34)	CHARACTER	28	PCTL_GS_BP2	バッファー・プール 2 の統計
52 (34)	UNSIGNED	2	PCTL_GS_BP2_BPRSIZE	各バッファーのサイズ (ページ数)
54 (36)	UNSIGNED	2	PCTL_GS_BP2_DSCNT	プール内のデータ・スペースの数
56 (38)	SIGNED	4	PCTL_GS_BP2_TOTVIRT	このプールによって使用された合計仮想
60 (3C)	SIGNED	4	PCTL_GS_BP2_TOTFIX	このプールによって使用された合計固定
64 (40)	UNSIGNED	8	PCTL_GS_BP2_FIXD_Y	入出力要求の前にすでにバッファーが固定されていた回数
72 (48)	UNSIGNED	8	PCTL_GS_BP2_FIXD_N	入出力要求の前にすでにバッファーが固定されていなかった回数
80 (50)	CHARACTER	28	PCTL_GS_BP3	バッファー・プール 3 の統計
80 (50)	UNSIGNED	2	PCTL_GS_BP3_BFRSIZE	各バッファーのサイズ (ページ数)

HFS の計画

表 86. *DisplayGlobalStats* コマンドの構造 (*GFUMPCTL*) (続き)

オフセット	タイプ	長さ	名前	説明
82 (52)	UNSIGNED	2	PCTL_GS_BP3_DSCNT	プール内のデータ・スペースの数
84 (54)	SIGNED	4	PCTL_GS_BP3_TOTVIRT	このプールによって使用された合計仮想
88 (58)	SIGNED	4	PCTL_GS_BP3_TOTFIX	このプールによって使用された合計固定
92 (5C)	UNSIGNED	8	PCTL_GS_BP3_FIXD_Y	入出力要求の前にすでにバッファが固定されていた回数
100 (64)	UNSIGNED	8	PCTL_GS_BP3_FIXD_N	入出力要求の前にすでにバッファが固定されていなかった回数
108 (6C)	CHARACTER	28	PCTL_GS_BP4	バッファ・プール 4 の統計
108 (6C)	UNSIGNED	2	PCTL_GS_BP4_BFRSIZE	各バッファのサイズ (ページ数)
110 (6E)	UNSIGNED	2	PCTL_GS_BP4_DSCNT	プール内のデータ・スペースの数
112 (70)	SIGNED	4	PCTL_GS_BP4_TOTVIRT	このプールによって使用された合計仮想
116 (74)	SIGNED	4	PCTL_GS_BP4_TOTFIX	このプールによって使用された合計固定
120 (78)	CHARACTER	8	PCTL-GS_BP4_FIXD_Y	入出力要求の前にすでにバッファが固定されていた回数
128 (80)	CHARACTER	8	PCTL_GS_BP4_FIXD_N	入出力要求の前にすでにバッファが固定されていなかった回数
136 (88)	CHARACTER	8	PCTL_GS_META_Y	検索時にメタデータがキャッシュ内にあった回数
144 (90)	CHARACTER	8	PCTL_GS_META_N	検索時にメタデータがキャッシュ内になかった回数
152 (98)	CHARACTER	8	PCTL_GS_RPN0_Y	読み取り/書き込み時に RPN0 がキャッシュ内にあった回数
160 (A0)	CHARACTER	8	PCTL_GS_RPN0_N	読み取り/書き込み時に RPN0 がキャッシュ内になかった回数
168 (A8)	CHARACTER	32	*	予約済み
200 (C8)	CHARACTER		PCTL_GS_END	

表 87 は、*DisplayFSStats* コマンドの *GFUMPCTL* 構造を示しています。

表 87. *DisplayFSStats* コマンドの構造 (*GFUMPCTL*)

オフセット	タイプ	長さ	名前	説明
0 (0)	STRUCTURE	228	PCTL_FSSTATS_TYPE	
0 (0)	CHARACTER	44	PCTL_FS_NAME	ファイル・システム名
44 (2C)	BITSTRING	4	PCTL_FS_FLAGS	RFS (内部 HFS 制御ブロック) からのフラグ
44 (2C)	BITSTRING	1	PCTL_FS_RFS_MTAB_FLAGS	MTAB フラグ
45 (2D)	BITSTRING	1	PCTL_FS_RFS_FLAGS	RFS フラグ
46 (2E)	BITSTRING	1	PCTL_FS_RFS_SYNC_ERR	RFS エラー・フラグ
47 (2F)	BITSTRING	1	*	予約済み
48 (30)	CHARACTER	8	PCTL_FS_CURR_TIME	現在のタイム・スタンプ
56 (38)	CHARACTER	8	PCTL_FS_MOUNT_TIME	マウント・タイム・スタンプ
64 (40)	UNSIGNED	2	PCTL_FS_SYNC	SYNC インターバル (秒数)
66 (42)	CHARACTER	2	*	予約済み
68 (44)	SIGNED	4	PCTL_FS_SIZE	ファイル・システムのサイズ (ページ数)
72 (48)	SIGNED	4	PCTL_FS_USED	FS で使用されたページの数
76 (4C)	SIGNED	4	PCTL_FS_AD_ALLOC	FS で AD に割り振られたページの数

表 87. *DisplayFSStats* コマンドの構造 (*GFUMPCTL*) (続き)

オフセット	タイプ	長さ	名前	説明
80 (50)	UNSIGNED	8	PCTL_FS_SEQ_IO	データの順次入出力要求の数
88 (58)	UNSIGNED	8	PCTL_FS_RANDOM_IO	データのランダム入出力要求の数
96 (60)	UNSIGNED	8	PCTL_FS_META_Y	検索時にメタデータがキャッシュ内にあった回数
104 (68)	UNSIGNED	8	PCTL_FS_META_N	検索時にメタデータがキャッシュ内になかった回数
112 (70)	UNSIGNED	8	PCTL_FS_RPN0_Y	読み取り/書き込み時に RPN0 がキャッシュ内にあった回数
120 (78)	UNSIGNED	8	PCTL_FS_RPN0_N	読み取り/書き込み時に RPN0 がキャッシュ内になかった回数
128 (80)	UNSIGNED	8	PCTL_FS_IX_TOPS	索引新規先頭 (index new tops) の数
136 (88)	UNSIGNED	8	PCTL_FS_IX_SPLITS	索引分割の数
144 (90)	UNSIGNED	8	PCTL_FS_IX_JOINS	索引結合の数
152 (98)	UNSIGNED	8	PCTL_FS_IX_RDHIT	索引ページ読み取りヒットの数
160 (A0)	UNSIGNED	8	PCTL_FS_IX_RDMISS	索引ページ読み取りミスの数
168 (A8)	UNSIGNED	8	PCTL_FS_IX_WRHIT	索引ページ書き込みヒットの数
176 (B0)	UNSIGNED	8	PCTL_FS_IX_WRMIS	索引ページ書き込みミスの数
184 (B8)	UNSIGNED	4	PCTL_FS_PGS_CACHED	このファイル・システムによってキャッシュされたデータ・バッファ・ページの数
188 (BC)	CHARACTER	4	PCTL_FS_HFRFN	DMIB からの HFRFN
192 (C0)	UNSIGNED	4	PCTL_FS_MEM_CNT	RFS からのメンバー・カウント
196 (C4)	CHARACTER	32	*	予約済み
228 (E4)	CHARACTER		PCTL_FS_END	

表 88 は、ExtendFS コマンドの *GFUMPCTL* 構造を示しています。

表 88. *ExtendFS* コマンドの構造 (*GFUMPCTL*)

オフセット	タイプ	長さ	名前	説明
0 (0)	STRUCTURE	68	PCTL_EXTENDFS_TYPE	
0 (0)	CHARACTER	44	PCTL_EXT_FSNAME	ファイル・システム名
44 (2C)	CHARACTER	1	PCTL_EXT_FLGS	フラグ
	1... ..		PCTL_EXT_NEW_VOL	新規ボリュームへ拡張
45 (2D)	CHARACTER	1	PCTL_EXT_UNIT	拡張する量の単位 ('M'、'T'、または 'C')
46 (2E)	CHARACTER	2	*	予約済み
48 (30)	SIGNED	4	PCTL_EXT_AMT	拡張する量
52 (34)	CHARACTER	16	*	予約済み
68 (44)	CHARACTER		PCTL_EXT_END	

表 89. サポートされる拡張単位の定数

タイプ	値	名前	説明
CHARACTER	1	EXTENDFS_UNIT_MB	pctl_ext_unit の定数 ('M'、メガバイト (MB))
CHARACTER	1	EXTENDFS_UNIT_TRK	pctl_ext_unit の定数 ('T'、トラック数)
CHARACTER	1	EXTENDFS_UNIT_CYL	pctl_ext_unit の定数 ('C'、シリンダー数)

HFS の計画

第 13 章 サブシステムの統計、状況、およびカウント情報へのユーザー・アクセス

この章では、IDCAMS によって提供されるプログラミング・インターフェースについて説明します。

ユーザー・アクセスは、非アクセス方式サービスの呼び出し側 (例えば、Cache RMF Reporter) 専用提供されています。その目的は、アクセス方式サービス・プログラム LISTDATA ルーチン (IDCSS01) を呼び出して、サブシステムの統計、状況、またはカウント情報を入手することです。この情報を入手するには、呼び出し側は IDCSS01 (SYS1.LINKLIB に格納されている) にリンクする前に許可ライブラリー内になければなりません。呼び出し側は、レジスター 1 が指す 3 ワードのパラメーター・リストを IDCSS01 に渡します。呼び出し側が許可なく IDCSS01 へのリンクを試みると、システムはエラー・コード 047 で異常終了します。

レジスター 1 パラメーター・リスト

Word 1

ゼロでなければなりません。

Word 2

IDCSS01 が必要とする引数リスト SSGARGL (以下に詳述) へのポインターのアドレスが入ります。この引数リスト内に SSGOADR という名前のフィールドがあり、これは IDCSS01 がサブシステムの統計、状況、またはカウント情報を戻すバッファー域を指します。バッファー域は、呼び出し側が取得することも、IDCSS01 に取得を任せることもできます。

Word 3

2 バイトの 2 進値戻りコードを受け取る領域を指します。可能な戻りコードは、このセクションの最後にリストします。

渡される引数リスト -- SSGARGL

以下では、SSGARGL について説明します。SSGARGL は、Word 2 が指すワードによって指示される領域です。Word 2 は、IDCSS01 に渡されるパラメーター・リストの一部です。一部のフィールドは、呼び出し側が設定する必要があります。その他のフィールドは、IDCSS01 が設定します。呼び出し側はオプション・フラグを設定して、要求される情報が、状況 (SSGRSS) であるか、カウント (SSGRPD) 情報であるかを示す必要があります。カウント情報を要求する場合、呼び出し側は、それを適用するのは、すべてのサブシステムか (SSGALL)、特定のサブシステムか (SSG1SS)、または特定の装置か (SSGDEV) を示すことが必要です。さらに、呼び出し側は、キャッシング・サブシステム・ボリュームを割り振る DD ステートメントの ddname を渡すか (SSGADDN を通して)、または情報を要求するキャッシング・サブシステム・ボリュームのボリュームと装置を識別するか (SSGAVOL と SSGUNIT を通して) のいずれかが必要です。呼び出し側が SSGADDN を渡した場

合、IDCSS01 が、SSGAVOL と SSGUNIT を設定します。以下のリストには、IDCSS01 が設定するその他のフィールドが含まれています。

OFFSET DECIMAL	OFFSET HEX	TYPE	LENGTH	NAME (DIM)	DESCRIPTION
0	(0)	STRUCTURE	96	SSGARGL	
0	(0)	CHARACTER	8	SSGHEAD	SSGARGL IDENTIFIER
8	(8)	ADDRESS	4	SSGADDN	DDNAME ADDRESS
12	(C)	ADDRESS	4	SSGAVOL	VOLUME ADDRESS
16	(10)	BITSTRING	4	SSGUNIT	DEVICE TYPE
16	(10)	BITSTRING	1	SSGUNIT1	GENERAL FLAGS
17	(11)	BITSTRING	1	SSGUNIT2	GENERAL FLAGS
18	(12)	BITSTRING	1	SSGUNIT3	DEVICE CLASS
19	(13)	BITSTRING	1	SSGUNIT4	DEVICE TYPE WITHIN THE CLASS
20	(14)	SIGNED	4	SSGOLN	LENGTH OF OUTPUT BUFFER
24	(18)	ADDRESS	4	SSGOADR	ADDRESS OF OUTPUT BUFFER
28	(1C)	BITSTRING	2	SSGOPT	OPTIONS BYTE @H17C
		1... ..		SSGRPD	SENSE SUBSYSTEM COUNTS
		.1.. ..		SSGRSS	SENSE SUBSYSTEM STATUS
		..1.		SSGCACHE	ON=CACHING,OFF=PAGING
		...1		SSGSDS	ON=1 SD, OFF= 2 SD'S
	 1..		SSGALL	SENSE FOR ALL SUBSYSTEMS
	1..		SSG1SS	SENSE FOR SPECIFIED SUBSYSTEM
	1.		SSGDEV	SENSE FOR SPECIFIED DEVICE
	1		SSGAMD	PTR TO 3880 MODEL PASSED INSTEAD OF PTR TO VOLUME
29	(1D)	1... ..		SSG2SD	PRINT 2 SD - H7A @H14N
		.1.. ..		SSGACD	ACCESSCODE @H18A
		..1.		SSGOFFL	SSGAVOL POINTS TO DEVICE ID OF AN OFFLINE DEVICE @H19A
		...1		SSGARGL2	this flag indicates that the parameter list is extended by the 16 bytes defined as SSGARGLX
	 1..		SSGLINKP	Link performance statistics supported by caller and request return of statistics when request for Subsystem counts are made (SSGRPD = ON) for one or more Subsystems (SSG1SS=ON or SSGALL = ON). Results passed back in area pointed to by SSGLPOAR
	1..		SSGARGL3	this flag indicates that the parameter list is extended with area defined as SSGARLGY
30	(1E)	BITSTRING	1	SSGMDLID	CU MODEL IDENTIFIER @H17A
31	(1F)	CHARACTER	1	SSGRCIOS	'old' parameter list end

Parameter List Extension.

Note: IDCSS01 will not reference fields contained in SSGARGLX unless the bit defined as SSGARGL2 is set to '1'.

32	(20)	CHARACTER	16	SSGARGLX	parm list extension
32	(20)	UNSIGNED	1	SSGATIME	I/O timeout value in seconds.@01A When SSGATIME is not zero, it will be stored into IOSXTIME in the IOSB. If the I/O is active or queued longer than SSGATIME, the I/O will be terminated and IDCSS01 will return a return code of 60. Hex values for SSGATIME support seconds from 1 to 255.
33	(21)	CHARACTER	2	SSGADEVN	Binary device number that received an IO (RC8) or

35	(23)	UNSIGNED	1	SSGLPRET	timeout (RC60) from IDCSS01 Reason code from link performance statistics processing
36	(24)	ADDRESS	4	SSGLPOAR	Address of link performance Statistics output buffer queue. When zero on return statistics not available for devices found on counts request or statistics not not requested by the caller
40	(28)	CHARACTER	8	*	reserved for future use

Parameter List Extension number 3

Note: IDCSS01 will not reference fields contained in SSGARGLY
unless the bit defined as SSGARGL3 is set to '1'.

Entire section added

```
=====
```

48	(30)	CHARACTER	48	SSGARGLY	Parm list extension 3
48	(30)	BITSTRING	4	SSGOPTY	Extended options
		1...		SSGRANKP	RANK performance statistics are supported by caller and request return of statistics when request for Subsystem counts are made (SSGRPD = ON) for one or more Subsystems (SSG1SS=ON or SSGALL = ON). Results passed back in area pointed to by SSGRSOAR
		.1..		SSGSEGMP	Segment pool statistics are supported by caller and request return of statistics when request for Subsystem counts are made (SSGRPD = ON) for one or more Subsystems (SSG1SS=ON or SSGALL = ON). Results passed back in area pointed to by SSGSPOAR
52	(34)	CHARACTER	8	SSGRETY	Extended return codes
52	(34)	UNSIGNED	1	SSGRSRET	Reason code from Rank Performance Statistics processing
53	(35)	UNSIGNED	1	SSGSGRET	Reason code from Segm Performance Statistics processing
60	(3C)	ADDRESS	4	SSGRSOAR	Address of Rank performance Statistics output buffer queue. When zero on return statistics not available for devices found on counts request or statistics not not requested by the caller
64	(40)	ADDRESS	4	SSGSPOAR	Address of segment pool performance Statistics output buffer queue. When zero on return statistics not available for devices found on counts request or statistics not not requested by the caller
68	(44)	CHARACTER	28	*	reserved for future use

MAPPING FOR THE SUBSYSTEM GET OUTPUT BUFFER

```
=====
```

OFFSET DECIMAL	OFFSET HEX	TYPE	LENGTH	NAME (DIM)	DESCRIPTION
-------------------	---------------	------	--------	------------	-------------

```

=====
0      (0) STRUCTURE      * SSGBUFR
0      (0) CHARACTER      6 SSGDAVOL      VOLUME SERIAL
6      (6) BITSTRING      1 *
      1... ....          SSG_UA_FLAG 'ONE' UNIT ADDR FORMAT
      .1.. ....          SSG_FND_STG2 2ND STG DIR PATH FND SS01 TO
                               LA01 IDCSS01 PASSED TO IDCLA01
      ..11 1111          *
7      (7) CHARACTER      1 SSGDCUID      Real CUID for data in
                               SSGDADA@02C
8      (8) CHARACTER      4 SSGDAUA1      FIRST UNIT ADDRESS
12     (C) CHARACTER      2 SSGDAUB1      FIRST UNIT ADDRESS (BINARY)
14     (E) SIGNED         2 SSGDALN      DATA LENGTH
16     (10) CHARACTER     * SSGDADA      DATA AREA
16     (10) CHARACTER     192 SSGDAXPF     CACHING SUBSYSTEM PERFORMANCE
                               STATISTICS m01C
16     (10) CHARACTER     160 SSGDASPF    CACHING SUBSYSTEM COUNTS
16     (10) CHARACTER     80 SSGDA2SD     STATUS FOR 2 SD @H20C
16     (10) CHARACTER     80 SSGDAIPF    PAGING SUBSYSTEM COUNTS
16     (10) CHARACTER     44 SSGDASS     SUBSYSTEM STATUS (LNGTHND)
                               @H20C
16     (10) CHARACTER     40 SSGDASS_40
                               STATUS FOR 3380
16     (10) CHARACTER     24 SSGDAACD
                               ACCESSCODES
16     (10) CHARACTER     12 SSGDAAC0
                               STORAGE CLUSTER 0 WORD
28     (1C) CHARACTER     12 SSGDAAC1
                               STORAGE CLUSTER 1 WORD
=====

```

MAPPING FOR THE SUBSYSTEM GET OUTPUT BUFFER
FOR LINK PERFORMANCE STATISTICS

Output buffer is a chain, anchored by SSSLPOAR, of link performance statistics tables. Each table entry represents data read from a single storage subsystem box. The statistics from a single box may be described with more than one table entry. The chain is a single threaded queue. Each chain element needs to be freed by the caller of this interface. Output buffer is built for storage boxes that support link performance statistics and are returned only if requested by caller (SSGLINKP) on a request for performance data (SSGRPD) for requests with scope of subsystem (SSGLSS) or all subsystems (SSGALL). The parameter list extension must also be passed as indicated with SSGARLG2.

Entire structure added

```

=====
OFFSET  OFFSET
DECIMAL  HEX  TYPE      LENGTH  NAME (DIM)  DESCRIPTION
=====  =====
0      (0) STRUCTURE      * SSSLBUFR
0      (0) CHARACTER      56 SSSLHDR     Queue element header
0      (0) UNSIGNED      4 SSSLQDAL    Length of this queue element,
                               header plus data
4      (4) UNSIGNED      4 SSSLQDAO    Offset to the start of the
                               statistics sets
8      (8) ADDRESS       4 SSSLFWDP    Pointer to next queue element,
                               when zero end of queue
=====

```

Orientation to subsystem that stats pertain

```

=====
12     (C) CHARACTER      6 SSSLVOL     Volume serial of device that
                               stats were read from
18     (12) CHARACTER     2 SSSLDEVN    Device number of device that
                               Stats were read from
20     (14) CHARACTER     8 *          Reserved
=====

```

28	(1C)	CHARACTER	28	SSGLLHDR	Link stats info
28	(1C)	UNSIGNED	2	SSGLNSET	Num of link stats sets read
30	(1E)	UNSIGNED	2	SSGLLSET	Size of each set
32	(20)	CHARACTER	6	SSGLCUT	Control unit type
38	(26)	CHARACTER	3	SSGLCUM	Control unit model
41	(29)	CHARACTER	10	SSGLSEQ	Control unit sequence number
51	(33)	CHARACTER	1	SSGLVER	Version of link statistics
52	(34)	CHARACTER	4	*	Not used
56	(38)	CHARACTER	*	SSGLDADA	Link performance stats read

Added mapping of link statistics

56	(38)	CHARACTER	96	SSGLLSTA(*)	Stats entry
OFFSET DECIMAL	OFFSET HEX	TYPE	LENGTH	NAME (DIM)	DESCRIPTION
56	(38)	UNSIGNED	1	SSGLLTYP	Link type 01 - ESCON 02 - Fibre Channel 1 Gb/s 03 - Fibre Channel 2 Gb/s 04 - Fibre Channel 4 Gb/s 05 - Fibre Channel 10 Gb/s
57	(39)	UNSIGNED	1	*	Unused
58	(3A)	UNSIGNED	1	SSGLLBYI	Byte increment 01 - 128K bytes
59	(3B)	UNSIGNED	1	SSGLLTM1	Time increment 01 - 16 MS
60	(3C)	CHARACTER	2	*	Unused
62	(3E)	CHARACTER	2	SSGLLIID	Interface id
64	(40)	CHARACTER	88	SSGLLSAT	Link statistics
64	(40)	UNSIGNED	4	SSGLERDB	Eckd read bytes in byte increment
68	(44)	UNSIGNED	4	SSGLEWRB	Eckd write bytes in byte increment
72	(48)	UNSIGNED	4	SSGLERDO	Eckd read operations. for escon ports, one count per chain which transfers data to the host. for ficon ports, one count per command which transfer data to the host
76	(4C)	UNSIGNED	4	SSGLEWRO	Eckd write operations. for escon ports, one count per chain which transfers data to the host. for ficon ports, one count per command which transfer data to the host
80	(50)	UNSIGNED	4	SSGLERDT	Eckd read accumulated time on channel. the active processing time for each command is accumulated based on increment value.
84	(54)	UNSIGNED	4	SSGLEWRT	Eckd write accumulated time on channel. the active processing time for each command is accumulated based on increment value.
88	(58)	UNSIGNED	4	SSGLPRDB	Pprc send bytes in byte increment
92	(5C)	UNSIGNED	4	SSGLPWRB	Pprc received bytes in byte increment
96	(60)	UNSIGNED	4	SSGLPRDO	Pprc send operations. each pprc write command sent by the pprc primary
100	(64)	UNSIGNED	4	SSGLPWRO	Pprc received operations. each pprc write command received by the pprc secondary.
104	(68)	UNSIGNED	4	SSGLPRDT	Pprc send accumulated time based on increment value
108	(6C)	UNSIGNED	4	SSGLPWRT	Pprc received accumulated time

112	(70)	UNSIGNED	4	SSGLSRDB	based on increment value Scsi read bytes in byte increment
116	(74)	UNSIGNED	4	SSGLSWRB	Scsi write bytes in byte increment
120	(78)	UNSIGNED	4	SSGLSRDO	Scsi read operations. each scsi read is counted
124	(7C)	UNSIGNED	4	SSGLSWRO	Scsi write operations. each scsi write is counted
128	(80)	UNSIGNED	4	SSGLSRDT	Scsi read accumulated time based on increment value
132	(84)	UNSIGNED	4	SSGLSWRT	Scsi write accumulated time based on increment value
136	(88)	CHARACTER	16	*	Reserved

MAPPING FOR THE SUBSYSTEM GET OUTPUT BUFFER
FOR RANK PERFORMANCE STATISTICS

Output buffer is a queue, anchored by SSGRSOAR, of rank performance statistics. Each queue element represents data read from a single storage subsystem box that describes sets of statistics. The queue is a single threaded queue. Each queue element needs to be freed by the caller of this interface. Output buffer built for storage subsystem boxes that support rank performance statistics and are returned only if requested by caller (SSGRANKP) on a request for performance data (SSGRPD) for requests with scope of subsystem (SSG1SS) or all subsystems (SSGALL).

Entire section added

=====

OFFSET DECIMAL	OFFSET HEX	TYPE	LENGTH	NAME (DIM)	DESCRIPTION
0	(0)	STRUCTURE	*	SSGRBUFR	
0	(0)	CHARACTER	56	SSGRHDR	Queue element header
0	(0)	UNSIGNED	4	SSGRQDAL	Length of this queue element, header plus data
4	(4)	UNSIGNED	4	SSGRQDAO	Offset to the start of the statistics sets
8	(8)	ADDRESS	4	SSGRFWDP	Pointer to next queue element, when zero end of queue

Orientation to subsystem that stats pertain

=====

12	(C)	CHARACTER	6	SSGRVOL	Volume serial of device that stats where read from
18	(12)	CHARACTER	2	SSGRDEVN	Device number of device that Stats where read from
20	(14)	CHARACTER	8	*	Reserved
28	(1C)	CHARACTER	28	SSGRRHDR	Rank stats info
28	(1C)	UNSIGNED	2	SSGRNSET	Num of rank stat sets available to be returned on the ESS starting from first rank ID SSGRRID. Can be used to determine buffer requirements
30	(1E)	UNSIGNED	2	SSGRLSET	Size of each rank stats set excluding size of array info SSGRRAR
32	(20)	CHARACTER	6	SSGRUCUT	Control unit type
38	(26)	CHARACTER	3	SSGRUCUM	Control unit model
41	(29)	CHARACTER	10	SSGRSEQ	Control unit sequence number
51	(33)	CHARACTER	1	SSGRVER	Version of rank statistics
52	(34)	UNSIGNED	2	SSGRARNM	Num of array info sets avail to be returned on the ESS starting from first rank id SSGRRID. Can be used to

54	(36)	UNSIGNED	2	SSGRARSZ	determine buffer requirements Size of each array information set SSGRRAR
56	(38)	CHARACTER	*	SSGRDADA	Rank performance stats read
56	(38)	CHARACTER	56	SSGRRSTA(*)	Rank statistics entry
56	(38)	UNSIGNED	2	SSGRRKID	Rank Identifier

The returned data consists of rank sets with each rank set containing one or more array information entries.

```
=====
```

OFFSET DECIMAL	OFFSET HEX	TYPE	LENGTH	NAME (DIM)	DESCRIPTION
0	(0)	STRUCTURE	*	SSGRRSTT	Rank stats entry mapping includes one or more entries for array information
0	(0)	UNSIGNED	2	SSGRRID	Rank identifier
2	(2)	UNSIGNED	2	SSGRRPNM	Segment pool number
4	(4)	UNSIGNED	1	SSGRRCNT	Count of arrays in rank
5	(5)	CHARACTER	3	*	Not used

In the statistics below bytes are accumulated in units of 128 KB, and time is accumulated in units of 16 milliseconds

```
=====
```

8	(8)	UNSIGNED	4	SSGRRBYR	Rank bytes read
12	(C)	UNSIGNED	4	SSGRRBYW	Rank bytes written
16	(10)	UNSIGNED	4	SSGRRROP	Rank read operations
20	(14)	UNSIGNED	4	SSGRRWOP	Rank write operations
24	(18)	UNSIGNED	4	SSGRRKRT	Rank read response time
28	(1C)	UNSIGNED	4	SSGRRKWT	Rank write response time
32	(20)	CHARACTER	24	SSGRRAR(*)	Array information mapping
32	(20)	UNSIGNED	2	SSGRRRID	Rank array id
34	(22)	CHARACTER	16	SSGRRREBC	Array type in ebcdic
50	(32)	UNSIGNED	1	SSGRRRTP	Array type

01 = RAID-5 02 = RAID-10 03 = RAID-6 04-FF NOT USED

```
=====
```

51	(33)	UNSIGNED	1	SSGRRRAW	Array width
52	(34)	UNSIGNED	1	SSGRRASP	Array speed (1000 RPM)
53	(35)	CHARACTER	2	SSGRRACP	Array capacity (GB)
55	(37)	CHARACTER	1	*	Not used

MAPPING FOR THE SUBSYSTEM GET OUTPUT BUFFER FOR SEGMENT POOL PERFORMANCE STATISTICS

Output buffer is a queue, anchored by SSGSPOAR, of segment pool perf statistics. Each queue element represents data read from a single storage subsystem box that describes sets of statistics. The queue is a single threaded queue. Each queue element needs to be freed by the caller of this interface. Output buffer built for storage subsystem boxes that support rank performance statistics and are returned only if requested by caller (SSGSEGMP) on a request for performance data (SSGRPD) for requests with scope of subsystem (SSG1SS) or all subsystems (SSGALL).

Entire section added

```
=====
```

OFFSET DECIMAL	OFFSET HEX	TYPE	LENGTH	NAME (DIM)	DESCRIPTION
0	(0)	STRUCTURE	*	SSGSBUFR	
0	(0)	CHARACTER	56	SSGSHDR	Queue element header
0	(0)	UNSIGNED	4	SSGSQDAL	Length of this queue element, header plus data
4	(4)	UNSIGNED	4	SSGSQDAO	Offset to the start of the

8	(8) ADDRESS	4	SSGSFWDP	statistics sets Pointer to next queue element, when zero end of queue
---	-------------	---	----------	---

Orientation to subsystem that stats pertain

```

=====
12      (C) CHARACTER      6      SSGSVOL      Volume serial of device that
18      (12) CHARACTER     2      SSGSDEVN     Device number of device that
20      (14) CHARACTER     8      *            Stats where read from
28      (1C) CHARACTER    28      SSGSSHDR     Reserved
28      (1C) UNSIGNED      2      SSGSNSSET    Segment pool stats info
30      (1E) UNSIGNED      2      SSGSLSET     Num of pool stats sets read
32      (20) CHARACTER     6      SSGSCUT      Size of each set
38      (26) CHARACTER     3      SSGSCUM      Control unit type
41      (29) CHARACTER    10     SSGSSEQ      Control unit model
51      (33) CHARACTER     1      SSGSVER      Control unit sequence number
52      (34) CHARACTER     4      *            Version of pool statistics
56      (38) CHARACTER     *      SSGSDADA     Not used
56      (38) CHARACTER    52     SSGSSSTA(*)  Segment pool perf stats read
56      (38) UNSIGNED      2      SSGSSGID     Segment pool stats entry
                                           Segment pool
                                           identifier
=====

```

OFFSET DECIMAL	OFFSET HEX	TYPE	LENGTH	NAME (DIM)	DESCRIPTION
0	(0)	STRUCTURE	52	SSGSSSTT	seg pool statistics entry includes one or more entry
0	(0)	UNSIGNED	2	SSGSSPID	Seg pool identifier
2	(2)	UNSIGNED	1	SSGSSPLT	Segment pool type

00-03 = reserved 04 = FB 1GB 05-131 = reserved 132 = CKD 1GB
133-255 = reserved

OFFSET DECIMAL	OFFSET HEX	TYPE	LENGTH	NAME (DIM)	DESCRIPTION
3	(3)	CHARACTER	1	*	Not used
4	(4)	UNSIGNED	4	SSGSSCAP	Real seg pool capacity (GB)
8	(8)	UNSIGNED	4	SSGSSNMS	Num real segments in seg pool
12	(C)	UNSIGNED	4	SSGSSNMA	Num real allocated segments in in segment pool
16	(10)	UNSIGNED	4	SSGSSRSC	Real segment conversions
20	(14)	UNSIGNED	4	SSGSSVCP	Virtual seg pool capacity(GB)
24	(18)	UNSIGNED	4	SSGSSNMV	Number of virtual segments in segment pool
28	(1C)	UNSIGNED	4	SSGSSVSC	Virtual segment conversions
32	(20)	UNSIGNED	4	SSGSSSDY	Number of segments that were sources of dynamic segment relocations
36	(24)	UNSIGNED	4	SSGSSTDY	Number of segments that were targets of dynamic segment relocations
40	(28)	CHARACTER	12	*	Not used

RETURN CODES FROM IDCSS01

```

=====
LEN  TYPE      VALUE      NAME      DESCRIPTION
=====
2    NUMB HEX    0000      SS01CC00  SUCCESSFUL COMPLETION
2    NUMB HEX    0004      SS01CC04  GETMAIN FAILED
2    NUMB HEX    0008      SS01CC08  I/O ERROR OCCURRED
2    NUMB HEX    000C      SS01CC12  VOLUME NOT FOUND
=====

```


2	NUMB HEX	0010	SS01CC16	ESTAE FAILED
2	NUMB HEX	0014	SS01CC20	LRA INSTRUCTION FAILED
2	NUMB HEX	0018	SS01CC24	BUFFER NOT LARGE ENOUGH
2	NUMB HEX	001C	SS01CC28	FILE NOT IN TIOT @H19A
2	NUMB HEX	0020	SS01CC32	VOLUME NOT FOUND ALL
2	NUMB HEX	0024	SS01CC36	NO PATH(S) TO SD(S)
2	NUMB HEX	0028	SS01CC40	NO ONLINE PATH FROM IOS
2	NUMB HEX	002C	SS01CC44	REQUEST NOT SUPPORTED
2	NUMB HEX	0030	SS01CC48	WARNING MSG ISSUED
2	NUMB HEX	0034	SS01CC52	ERROR MSG ISSUED @H19A
2	NUMB HEX	0038	SS01CC56	SERIOUS ERROR MSG @H19A
2	NUMB HEX	003C	SS01CC60	I/O timeout occurred
2	NUMB HEX	0040	SS01CC64	Reserved
2	NUMB HEX	0044	SS01CC68	RESERVED
2	NUMB HEX	0063	SS01CC99	RESERVED

REASON CODES FROM ESS (BOX) PERFORMANCE STATISTICS
(SSGLPRET, SSGRSRET, SSGSGRET).

LEN	TYPE	VALUE	NAME	DESCRIPTION
1	NUMB HEX	01	SS01LP01	Stats requested, all stats returned on supporting subsystem boxes
1	NUMB HEX	02	SS01LP02	Stats not requested by the caller
1	NUMB HEX	03	SS01LP03	Stats requested, but no supporting subsystem boxes found to support them
1	NUMB HEX	04	SS01LP04	Stats requested, but stats from one or more supporting subsystem boxes could not be read

付録 A. CVAF VTOC アクセス・マクロ

この付録には、内部システム論理に依存する VTOC 索引情報が含まれています。この情報は、ユーザーが CVAF マクロ命令を使用して VTOC を変更するのを支援する目的で提供されています。

CVAF マクロ命令のコーディング

ここでは、CVAF マクロ命令をコーディングし、使用可能なパラメーター・オプションから選択して必要なサービスを指定する方法について説明します。ユーザーがマクロをインプリメントする際に役立つコーディングの例も示します。

CVAFDIR マクロの概要および仕様

索引付き VTOC または索引なし VTOC の場合、CVAFDIR マクロを使用して、以下の機能を実行できます。

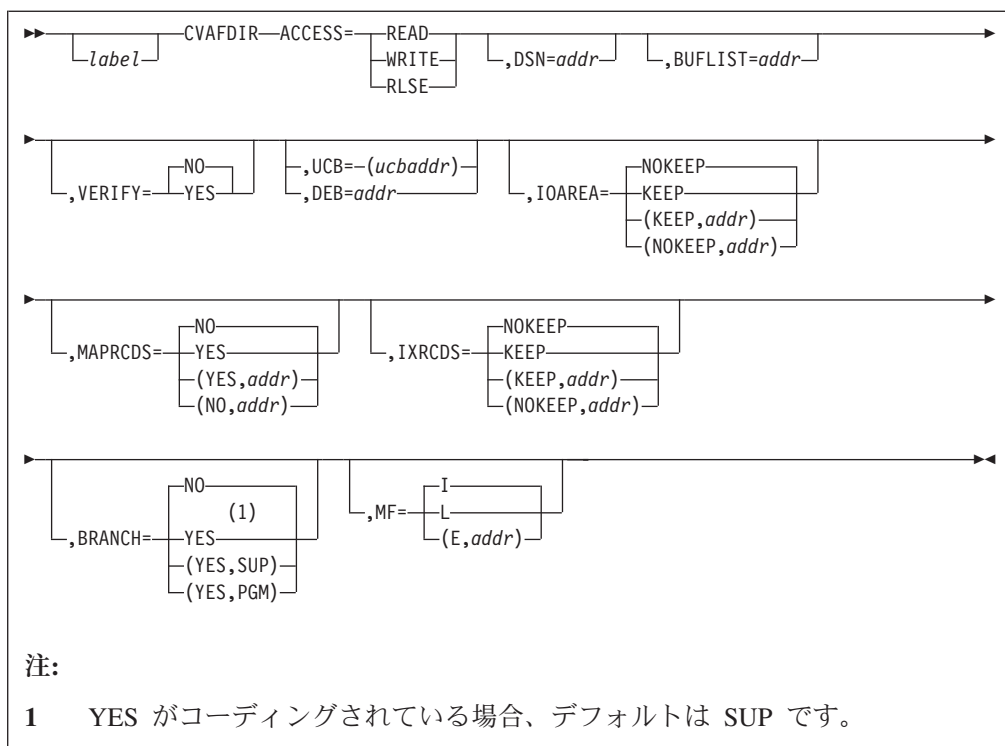
- データ・セット名を指定して、DSCB の読み取りまたは書き込みをする。
- アドレスを指定して、DSCB の読み取りまたは書き込みをする。

さらに、索引付き VTOC の場合、CVAFDIR マクロを使用して、以下の機能を実行できます。

- VTOC 索引レコードの読み取りまたは書き込み。(これにより、呼び出し側プログラムは VTOC 索引を変更できます。)
- 最初の高位 VIER および見出し検索で使用された VIER を読み取り、仮想ストレージに保存する。
- スペース・マップ VIR を読み取り、仮想ストレージに保存する。
- 仮想ストレージに保存されていた VIR を解放する。

追加情報については、50 ページの『DSCB への直接アクセス』を参照してください。

CVAFDIR マクロのフォーマットは、次のとおりです。



ACCESS: DSCB または VIR の読み取りまたは書き込み、またはバッファ・リストの解放

ACCESS が READ または WRITE の場合、索引付きまたは索引なし VTOC の単一の DSCB がアクセスされるか、または索引付き VTOC の 1 つ以上の VIR がアクセスされます。

ACCESS=READ

単一の DSCB または 1 つ以上の VIR を、バッファ・リスト内にアドレスがあるバッファに読み取ることを指定します。

DSCB 用のバッファ・リストの場合、バッファ・リスト内の 1 つの項目のみが使用されます。スキップ・ビットがゼロに設定され、非ゼロのバッファ・アドレスを持つ、最初の項目が使用されます。

スキップ・ビットがオフに設定されたバッファ・リスト項目を持つすべての VIR が、バッファに読み取られます。

DSCB バッファ・リストに対する ACCESS=READ の場合、DSN と BUFLIST が必要です。

ACCESS=WRITE

単一の DSCB または 1 つ以上の VIR を、バッファ・リスト内にアドレスがあるバッファから書き込むことを指定します。

ACCESS=WRITE は、呼び出し側が APF 許可されている場合、BRANCH=NO でのみ許可されます。

DSCB バッファ・リストに対する ACCESS=WRITE の場合、DSN と BUFLIST が必要です。

バッファ・リスト項目の変更ビットが設定されている場合、変更ビットが設定された項目のみが書き込まれます。どの変更ビットもオンになっていない場合、すべての VIR が書き込まれます。

ACCESS=RLSE

VIR バッファ・リストにのみ適用されます。BUFLIST キーワードで識別された VIR バッファ・リスト・チェーン内の 1 つ以上のバッファの解放と、すべてのバッファが解放される各バッファ・リストの解放を要求します。

ACCESS=RLSE の場合、DSN と BUFLIST は必要ありません。

スキップ・ビットがゼロに設定され、非ゼロのバッファ・アドレスを持つ、バッファ・リスト内のバッファのみが解放されます。スキップ・ビットが 1 に設定されている項目を持つバッファ・リストは解放されません。

索引付き VTOC では、ACCESS=RLSE がコーディングされている場合、BUFLIST キーワードで指示されたバッファ・リストとバッファが、CVAF パラメータ・リストの CVMRCDS および CVIRCDS フィールドで提供されたバッファ・リストとともに解放されます。BUFLIST フィールドに CVMRCDS または CVIRCDS バッファが提供されている場合 (直接に、またはチェーンングを通して間接的に)、CVAF がバッファを複数回解放するのを防止するために、キーワード MAPRCDS=YES、IXRCDS=KEEP または MAPRCDS=(NO,0)、IXRCDS=(NOKEEP,0) をコーディングする必要があります。バッファが解放されると、バッファ・リストを指している CVAF パラメータ・リスト・フィールドが更新されます。

DSN: DSCB の名前の指定

DSN=addr アクセスする DSCB の 44 バイトのデータ・セット名のアドレスを提供します。

ACCESS=READ または WRITE で、要求が DSCB の読み取りまたは書き込みである場合、DSN が必要です。140 バイト DSCB が指定されている場合、次のようになります。

- CVAF はストレージ・ロケーションの妥当性検査を行いますが、その場所の内容は無視します。
- VTOC 内部のエクステンントを指す引数を指定します。

BUFLIST: 1 つ以上のバッファ・リストの指定

BUFLIST=addr DSCB または VIR の読み取りまたは書き込みに使用されるバッファ・リストのアドレスを提供します。

VERIFY: DSCB が Format-0 DSCB であるかどうかの検査

VERIFY=YES CVAF は、DSCB を書き込む前に、DSCB が format-0 DSCB であるかどうかを検査します。キーの最初の 4 バイトを 2 進ゼロと比

較します。キーが 4 バイトのゼロで始まっていない場合、DSCB は書き込まれず、エラー・コードが戻されます。これはデフォルトです。

VERIFY=NO CVAF は、DSCB のキーをテストしません。

制約事項: VERIFY は、140 バイト DSCB を書き込む場合にのみ適用されます。

VIR が書き込まれる場合は、VERIFY は無視されます。

UCB または DEB: アクセスする VTOC の指定

UCB= *rs* タイプまたは (2-12) 標準形式 **UCB=** *rx* タイプまたは (2-12) 実行形式
 アクセスする VTOC の UCB のアドレスを指定します。UCB アドレスは、キャプチャーされた UCB のアドレスでも、16 MB 境界より上または下の実 UCB のアドレスでも構いません。UCB コピーではなく、UCB のアドレスを使用してください。無許可の呼び出し側は、このパラメーターを使用してはなりません。プログラムが 31 ビット・モードの場合、このアドレスは 31 ビット・アドレスでなければなりません。高位バイトはアドレスの一部です。MF=L と一緒に UCB パラメーターをコーディングしてはなりません。

推奨: UCB パラメーターのアドレスは、レジスター (2-12) としてコーディングしてください。RX タイプのアドレスをコーディングすると、予測不能の結果になります。

注: 呼び出し側の AMODE に一致する UCB アドレスを提供する必要があります。つまり、AMODE=24 は 24 ビット UCB アドレスを必要とし、一方、AMODE=31 は 31 ビット UCB アドレスを必要とします。

DEB=addr アクセスするボリューム目録 (VTOC) に対してオープンされた DEB のアドレスを提供します。DEB サブパラメーターを指定した場合、CVAF は VTOC または VTOC 索引に対する出力要求を許可しません。CVAF は CVAF 呼び出しの期間中 DEB テーブルからその DEB を除去するので、ユーザーが許可されていない場合 (APF もシステム・キーもない場合)、ユーザーはその DEB によって表されるデータ・セットに対して非同期アクティビティ (EXCP、CLOSE、EOV など) を実行することはできません。ユーザーが許可されていない場合 (APF 許可もシステム・キーもない場合)、CVAFDIR に対して DEB アドレス (UCB ではなく) を指定してください。詳しくは、44 ページの『ボリュームの識別』を参照してください。

IOAREA キーワードを通して、前に入手した入出力域を提供する場合は、UCB も DEB も提供する必要はありません。そうでない場合は、UCB または DEB のいずれかを提供してください。UCB アドレスを提供した場合、CVPL 内の UCB は入出力域内の UCB アドレスによってオーバーレイされます。CVPL で DEB と UCB アドレスの両方を提供した場合、DEB アドレスが使用され、CVPL 内の UCB アドレスは、DEB 内の UCB アドレスによってオーバーレイされます。

IOAREA: 入出力作業域の保持または解放

IOAREA=KEEP

CVAF 要求の完了時に、CVAF パラメーター・リストに関連した CVAF 入出力域を保持する必要があることを指定します。呼び出し側が許可されている場合 (APF またはシステム・キー)、BRANCH=NO のみ IOAREA=KEEP をコーディングできます。

IOAREA=KEEP がコーディングされている場合、呼び出し側は、さらに VTOC へのアクセスが必要かどうかに関係なく、将来の何らかの時点で、IOAREA=NOKEEP を指定して CVAF を呼び出す必要があります。そのような呼び出し側の例としては、CVAF を呼び出すルーチンのリカバリー・ルーチンがあります。

IOAREA=KEEP をコーディングした場合、プログラムは特定の初期化機能をバイパスできるため、後続の CVAF 要求がより効率的になります。前に取得した CVAF 入出力域が提供されている場合は、DEB または UCB を指定する必要はありません。また、これらの値を変更することもできません。

IOAREA=KEEP が最初に発行されたときに、CVAF は、CVAF パラメーター・リストに CVAF 入出力域を戻します (CVIOAR)。後続の CVAF 呼び出しは、その同じパラメーター・リストを使用することができ、CVAF は入出力域を CVIOAR から取得します。

現行ボリューム上の処理が終了したら、保持していたすべての領域を解放してください。

IOAREA=(KEEP,addr)

前に取得した入出力域のアドレスを提供します。異なる CVAF パラメーター・リストが使用されている場合、アドレスを IOAREA キーワードの 2 番目のパラメーターとしてコーディングすることにより、前に取得した入出力域を CVAF に渡すことができます。

IOAREA=NOKEEP

CVAF 要求の完了時に、作業域が解放されます。

IOAREA=(NOKEEP,addr)

CVAF 要求の完了時に、前に取得された作業域が解放されます。

MAPRCDS: MAPRCDS バッファースト・リストとバッファーストの保持または解放

このキーワードは、索引付き VTOC にのみ適用され、MAPRCDS バッファースト・リストとバッファーストの処理を指定します。

MAPRCDS=YES

処理の終了時にバッファースト・リストとバッファーストを保存することを指定します。

CVAF パラメーター・リスト内にバッファースト・リスト・アドレスがない場合、CVAF は MAP VIR を読み取って、それが取得したバッファーストに入れます。処理後に CVAF パラメーター・リスト・フィールド CVMRCDS から、アドレスと VIR の RBA を含んでいるバッファースト・リストにアクセス

できます。バッファ・リストとバッファは、ユーザーの保護キー内にあります (ユーザーが許可されていない場合は、サブプール 0、許可されている場合は、サブプール 229)。

現行ボリューム上での処理が終了したら、保持されていたすべての領域を解放してください。

MAPRCDS=(YES,addr)

MAPRCDS=YES がコーディングされ、バッファ・リスト・アドレス (CVAF パラメーター・リストの CVMRCDS) が提供されている場合、VIR は読み取られません。

CVAFDIR マクロで使用された CVMRCDS バッファ・リストを、MAPRCDS キーワードを通して、別の CVAF マクロ呼び出しに渡すことができます。

索引なし VTOC に対して MAPRCDS=YES がコーディングされている場合、機能は実行されますが、エラー・コードが戻されます。

MAPRCDS=NO

MAPRCDS=NO がコーディングされている場合、CVPL の CVMRCDS フィールドにアドレスが指定されているバッファ・リスト内の、スキップ・ビットがオンでないすべてのバッファが解放されます。バッファがすべて解放された場合、バッファ・リストも解放されます。

MAPRCDS=(NO,addr)

CVAF によって前に取得されたバッファ・リストとバッファを解放します。

ユーザーは、CVAF によって取得されたバッファ・リストとバッファを解放する必要があります。これは 3 通りの方法のいずれかで行うことができます。

- バッファを取得した CVAFDIR マクロで MAPRCDS=NO をコーディングする。
- 後続の CVAF マクロで MAPRCDS=NO をコーディングする。
- CVAFDIR ACCESS=RLSE をコーディングし、BUFLIST キーワードでバッファ・リストのアドレスを提供する。

要件: 読み取られた MAP レコードの保全性を維持するために、VTOC をエンキューし、装置を予約する必要があります。

IXRCDS: 仮想ストレージに VIERS を保存

このキーワードは、索引付き VTOC にのみ適用されます。

IXRCDS=KEEP

ストレージに読み取られた VIER を仮想ストレージに保持することを指定します。VIER は、処理を正常に完了できなくても保存されます。CVAF パラメーター・リストのフィールド CVIRCDS には、読み取られた VIR バッファ・アドレスと VIER の RBA を含んでいるバッファ・リストのアドレスが入ります。

見出し検索機能は、バッファ・リストを動的に更新し、必要な場合は、追加のバッファ・リストを取得して、相互をチェーニングします。

IXRCDS=KEEP が指定され、CVPL の CVAF にバッファ・リストが提供されていない場合、CVAF はバッファ・リストとバッファを取得し、最初の高位 VIER を読み取ります。バッファ・リストのアドレスは、CVPL の CVMICDS フィールドに入れます。

バッファ・リストとバッファは、ユーザーの保護キー内にあります。ユーザーが許可されていない場合は、サブプール 0、許可されている場合は、サブプール 229 です。

索引なし VTOC に対して IXRCDS=KEEP がコーディングされている場合、DSCB の読み取りまたは書き込み要求は実行されますが、エラー・コードが戻されます。

現行ボリューム上での処理が終了したら、保持されていたすべての領域を解放してください。

IXRCDS=(KEEP,addr)

アドレスを IXRCDS キーワードの 2 番目のパラメーターとして指定することにより、1 つの CVAF 要求からの索引レコード・バッファ・リスト・アドレスが、この CVAF パラメーター・リストに渡されます。

IXRCDS=NOKEEP

IXRCDS=NOKEEP がコーディングされている場合、アクセスされた VIER (ある場合) は保存されません。さらに、CVAF パラメーター・リストの CVIRCDS フィールドに提供されたバッファ・リストは解放され、バッファ・リスト内で見つかったすべてのバッファも同様に解放されます。バッファ・リスト内のいずれかの項目にスキップ・ビットが設定されている場合、バッファおよびバッファ・リストは解放されません。これはデフォルトです。

IXRCDS=(NOKEEP,addr)

前にアクセスした VIER を保存しないことを指定します。

ユーザーは、CVAF によって取得されたバッファ・リストとバッファを解放する必要があります。これは 3 通りの方法のいずれかで行うことができます。

- バッファを取得した CVAFDIR マクロで IXRCDS=NOKEEP をコーディングする。
- 後続の CVAF マクロで IXRCDS=NOKEEP をコーディングする。
- CVAFDIR ACCESS=RLSE をコーディングし、BUFLIST キーワードでバッファ・リストのアドレスを提供する。

要件: 読み取られた VIER の保全性を維持するために、VTOC をエンキューし、装置を予約する必要があります。

BRANCH: マクロの入り口の指定

BRANCH=(YES,SUP)

分岐入力を要求します。プログラムは監視プログラム状態でなければなりません。保護キー検査はバイパスされます。

BRANCH=YES がコーディングされている場合、18 ワードの保管域を提供する必要があります。CVAF の入り口では、ロックは保持できません。SRB モードは許可されません。

BRANCH=YES

YES がコーディングされている場合、SUP がデフォルトなので、BRANCH=(YES,SUP) と等価です。保護キー検査はバイパスされます。

BRANCH=(YES,PGM)

分岐入力を要求します。プログラムは APF によって許可されているか、問題プログラム状態でなければなりません。保護キー検査はバイパスされません。

BRANCH=NO

SVC 入力を要求します。出力操作を要求する場合、プログラムは APF によって許可されている必要があります。保護キー検査が実行されます。これはデフォルトです。

MF: マクロの形式の指定

このキーワードは、リスト形式、実行形式、通常形式のマクロのいずれを要求するのかを指定します。

MF=I I がコーディングされている場合、あるいは L も E もコーディングされていない場合、CVAF パラメーター・リストが生成され、CVAF が呼び出されます。これは、通常形式のマクロです。

MF=L リスト形式のマクロを示します。パラメーター・リストが生成されますが、CVAF は呼び出されません。

MF=(E,addr)

実行形式のマクロを示します。アドレスが *addr* で指定されている CVAF パラメーター・リストは、この形式のマクロによって変更できます。

CVAFDIR からの戻りコード

CVAF から戻ると、レジスター 1 に CVPL (CVAF パラメーター・リスト) のアドレスが入り、レジスター 15 に、以下の戻りコードのいずれかが入ります。

戻りコード	意味
0 (X'00')	要求は、正常に実行されました。ただし、CVAFDIR 要求が DSCB の読み取りまたは書き込みで、VTOC 索引構造エラーが検出された場合、CVSTAT フィールドは、検出された構造エラーを示します。(CVSTAT コードの説明は、425 ページの『VTOC 索引エラー・メッセージと関連コード』にあります。)
4 (X'04')	エラーが発生しました。CVPL の CVSTAT フィールドに、エラーの原因が示されます。(CVSTAT コードの説明は、425 ページの『VTOC 索引エラー・メッセージと関連コード』にあります。)
8 (X'08')	VTOC 索引レコードの読み取りまたは書き込み要求の処理中の無効な VTOC 構造。CVPL の CVSTAT フィールドに、エラーの原因が示されます。(CVSTAT コードの説明は、425 ページの『VTOC 索引エラー・メッセージと関連コード』にあります。)
12 (X'0C')	CVAF パラメーター・リストが保護キー内に存在しないか、または無効です (ID が無効であるか、長さフィールドが誤っているか、または CVFCTN (機能コード) フィールドが無効またはこのリリースでサポートされていません)。CVPL は変更されませんでした。
16 (X'10')	入出力エラーが発生しました。

VTOC での CVAFDIR マクロの使用例

この例は、CVAFDIR マクロを使用して、指定されたデータ・セット名の DSCB を読み取り、DSCB が区分データ・セット用であるかどうかを判別します。44 バイトのデータ・セット名のアドレスが、レジスター 5 (例のラベル RDSN) でプログラムに提供されます。VTOC に対してオープンされた DEB のアドレスは、レジスター 4 (例のラベル RDEB) でプログラムに提供されます。

バッファ・リストはプログラム内にあり、ICVAFBFL マクロによって生成されます。DSCB バッファはプログラム内にあり、IECSDSL1 マクロによって生成されます。

CVAF マクロ

```

DIRXMP1 CSECT
        STM 14,12,12(RSAVE)
        BALR 12,0
        USING *,12
        ST RSAVE,SAVEAREA+4
        LA RWORK,SAVEAREA
        ST RWORK,8(,RSAVE)
        LR RSAVE,RWORK
*****
*
*          REGISTERS
*
*****
REG1    EQU 1          REGISTER 1
RWORK   EQU 3          WORK REGISTER
RDEB    EQU 4          DEB ADDRESS
RDSN    EQU 5          ADDRESS OF DATA SET NAME
RSAVE   EQU 13         SAVE AREA ADDRESS
REG15   EQU 15         RETURN CODE REGISTER 15
*****
*
*          RETURN CODES
*
*****
PDSRTN  EQU 0          DATA SET A PDS RETURN CODE
NOTFND  EQU 4          DATA SET NOT FOUND RETURN CODE
NOTPDS  EQU 8          DATA SET NOT A PDS RETURN CODE
UNEXPECD EQU 12        UNEXPECTED ERROR RETURN CODE
*****
*
*          READ DSCB INTO DS1FMTID.
*          DATA SET NAME ADDRESS SUPPLIED IN RDSN.
*          ADDRESS OF DEB OPEN TO VTOC SUPPLIED IN RDEB.
*          DETERMINE IF DATA SET IS A PARTITIONED DATA SET.
*          THIS PROGRAM IS NEITHER REENTRANT NOR REUSABLE.
*
*****
        XC BUFLIST(BFLHLN+BFLELN),BUFLIST ZERO BUFFER LIST
        OI BFLHFL,BFLHDSCB    DSCBS TO BE READ WITH BUFFER LIST
        MVI BFLHNOE,1        ONE BUFFER LIST ENTRY
        LA RWORK,DS1FMTID    ADDRESS OF DSCB BUFFER
        ST RWORK,BFLEBUF     PLACE IN BUFFER LIST
        OI BFLEFL,BFLECHR    CCHHR OF DSCB RETURNED BY CVAF
        MVI BFLELTH,DSCBLTH  DATA PORTION OF DSCB READ - DSN *
                               SUPPLIED IN CVPL
        MVC DS1DSNAM,0(RDSN)  MOVE IN DATA SET NAME TO WORKAREA
        CVAFDIR ACCESS=READ,DSN=DS1DSNAM,BUFLIST=BUFLIST,DEB=(RDEB)
        USING CVPL,REG1      ADDRESSABILITY TO CVPL
        LTR REG15,REG15      ANY ERROR
        BZ NOERROR           BRANCH IF NOT

```

図 47. VTOC での CVAFDIR マクロの例 (1/2)

```

*****
*
*       DETERMINE WHAT ERROR IS
*
*****
C     REG15,ERROR4      IS RETURN CODE 4
BNE  OTHERERR         BRANCH IF NOT 4
CLI  CVSTAT,STAT001   IS IT DATA SET NAME NOT FOUND?
BNE  OTHERERR         BRANCH IF NOT
DROP REG1             ADDRESSABILITY TO CVPL NOT NEEDED
*****
*
*       DATA SET NAME NOT FOUND
*
*****
L     RSAVE,4(,RSAVE)
RETURN (14,12),RC=NOTFND SET UP DATA SET NOT FOUND ERROR
NOERROR EQU *          DSCB READ
MVC  F1CCHHR,BFLEARG  MOVE CCHHR OF FORMAT 1/4 DSCB TO   *
                          WORKAREA
CLI  DS1FMTID,C'4'    IS DSCB A FORMAT 4 DSCB
BE   NOTF1            BRANCH IF YES. NOT A FORMAT 1
TM   DS1DSORG,DS1DSGPO IS FORMAT 1 DSCB FOR PARTITIONED *
                          DATA SET
BO   PDS              BRANCH IF PDS
NOTF1 EQU *          DSCB IS NOT A PDS
L     RSAVE,4(,RSAVE)
RETURN (14,12),RC=NOTPDS SET UP NOT PDS RETURN CODE
PDS   EQU *          DATA SET IS PARTITIONED
L     RSAVE,4(,RSAVE)
RETURN (14,12),RC=PDSRTN SET UP PDS RETURN CODE
OTHERERR EQU *       UNEXPECTED ERROR
L     RSAVE,4(,RSAVE)
RETURN (14,12),RC=UNEXPECD

ERROR4 DC  F'4'      ERROR RETURN CODE 4
BUFLIST ICVAFBFL DSECT=NO BUFFER LIST
        IECSDSL1 (1)  FORMAT 1 DSCB DATA SET NAME AND   *
                          BUFFER
DSCBLTH EQU *-IECSDSL1-L'DS1DSNAM LENGTH OF DATA PORTION OF DSCB
F1CCHHR DS  XL5      CCHHR OF DSCB
SAVEAREA DS  18F     SAVE AREA
CVPL     ICVAFPL ,    CVPL MAPPING MACRO

END

```

図 47. VTOC での CVAFDIR マクロの例 (2/2)

索引付き VTOC での CVAFDIR マクロの使用例

この例は、CVAFDIR マクロを使用して、VTOC から 1 つ以上の DSCB を読み取ります。UCB は、レジスター 4 (ラベル RUCB) でプログラムに提供されます。各 DSCB の TTR が呼び出し側に戻されます。このプログラムは APF 許可が必要です。

パラメーター・リストのアドレスは、レジスター 5 (ラベル RLIST) でプログラムに提供されます。パラメーター・リストは、1 つ以上の 3 ワード項目を含んでいます。3 ワード項目のそれぞれのフォーマットは、LISTMAP DSECT によってマップされます。最初のワードには、読み取る DSCB のデータ・セット名のアドレスが

入ります。2 番目のワードには、読み取った DSCB を入れる 96 バイト・バッファ
ーのアドレスが入ります。2 番目のワードには、読み取られた DSCB の 3 バイト
TTR のアドレスが入ります。

CVPL は、ラベル CVPL でリスト形式の CVAFDIR マクロによって生成されま
す。BUFLIST、IXRCDS、IOAREA、および BRANCH キーワードは、リスト形式
のマクロでコーディングされています。複数の DSCB が読み取られる場合のオーバ
ーヘッドを避けるために、IXRCDS=KEEP と IOAREA=KEEP がコーディングされ
ています。リスト形式の CVAFDIR マクロで BRANCH=(YES,PGM) をコーディン
グして、CVPL が CVIPGM ビットを 1 に設定するようにします。これは、呼び出
し側が APF によって許可され、監視プログラム状態にないことを CVAF に示しま
す。次に、実行形式の CVAFDIR マクロで BRANCH=YES (BRANCH=(YES,PGM)
ではなく) を指定します。CVIPGM ビットはリスト形式のマクロで設定されるから
です。

プログラムが終了して CVAF 入出力域と索引レコード・バッファ・リストを解放
する前に、ACCESS=RLSE を指定した CVAFDIR マクロがコーディングされていま
す。ユーザー提供のバッファ・リストは解放する必要がないため、BUFLIST=0 が
コーディングされています。BUFLIST はリスト形式の CVAFDIR マクロでコーデ
ィングされたので、CVPL の CVBUFL フィールドにあります。解放機能の場合、
このフィールドはゼロに設定されていなければなりません。

```

DIRXMP2  CSECT
          STM 14,12,12(13)
          BALR 12,0
          USING *,12
          ST 13,SAVEAREA+4
          LA RWORK,SAVEAREA
          ST RWORK,8(,13)
          LR 13,RWORK
*****
*
*          REGISTERS
*
*****
RWORK    EQU 3          WORK REGISTER
RUCB     EQU 4          UCB ADDRESS SUPPLIED BY CALLER
RLIST    EQU 5          ADDRESS OF PARAMETER LIST
RDSN     EQU 6          ADDRESS OF DATA SET NAME
RTTR     EQU 7          ADDRESS OF TTR
REG15    EQU 15        RETURN CODE REGISTER 15
*****
*
*          READ DSCB OF DATA SET NAME SUPPLIED. RETURN TTR OF DSCB.
*          UCB ADDRESS SUPPLIED IN RUCB.
*          ADDRESS OF PARAMETER LIST IN RLIST.
*          WORD 1 OF PARAMETER LIST = ADDRESS OF DATA SET NAME
*          WORD 2 OF PARAMETER LIST = ADDRESS OF DSCB TO BE RETURNED
*          WORD 3 OF PARAMETER LIST = ADDRESS OF TTR TO BE RETURNED
*          WORDS 1-3 DUPLICATED WITH THE HIGH ORDER BIT OF
*          WORD 3 SET TO ONE FOR LAST ENTRY.
*
*****
          USING LISTMAP,RLIST    ADDRESSABILITY TO PARMLIST
TOPLOOP  EQU *                  LOOP FOR EACH DSCB
          XC BUFLIST(BFLHLN+BFLELN),BUFLIST ZERO BUFFER LIST
          OI BFLHFL,BFLHDSCB    DSCBS TO BE READ WITH BUFFER LIST
          MVI BFLHNOE,1         ONE BUFFER LIST ENTRY
          LA RWORK,LISTDSCB     ADDRESS OF DSCB BUFFER
          ST RWORK,BFLEBUF      PLACE IN BUFFER LIST
          OI BFLEFL,BFLETTR     TTR OF DSCB RETURNED BY CVAF
          MVI BFLELTH,DSCBLTH   DATA PORTION OF DSCB READ - DSN *
                                  SUPPLIED IN CVPL
          L RDSN,LISTDSN        ADDRESS OF DATA SET NAME
          CVAFDIR DSN=(RDSN),UCB=(RUCB),MF=(E,CVPL),BRANCH=YES
          L RTTR,LISTTTR        ADDRESS OF TTR TO BE RETURNED
          USING TTRMAP,RTTR     MAP OF TTR
          LTR REG15,REG15       ANY ERROR
          BZ NOERROR            BRANCH IF NOT
          XC TTR,TTR            ZERO TTR INDICATING NO DSCB
          B RELOOP              GET NEXT ENTRY
NOERROR  EQU *                  DSCB READ
          MVC TTR,BFLEARG       RETURN TTR OF DSCB

```

図 48. 索引付き VTOC での CVAFDIR マクロの使用例 (1/2)

```

RELOOP EQU *
TM LASTLIST, LASTBIT
LA RLIST, NEXTLIST
BZ TOPLOOP
CVAFDIR ACCESS=RLSE,
         IOAREA=NOKEEP,
         IXRCDS=NOKEEP,
         BUFLIST=0,
         BRANCH=YES,
         MF=(E, CVPL)
L 13, SAVEAREA+4
RETURN (14, 12)

BUFLIST ICVAFBFL DSECT=NO          BUFFER LIST

SAVEAREA DS 18F                    REGISTER SAVE AREA
LISTMAP DSECT
LISTDSN DS F                        ADDRESS OF DATA SET NAME
LISTDSCB DS F                       ADDRESS OF BUFFER FOR DSCB TO BE
                                         RETURNED
LISTTTR DS 0F                       ADDRESS OF TTR OF DSCB TO BE
                                         RETURNED
LASTLIST DS X                        FIRST BYTE
LASTBIT EQU X'80'                   LAST ENTRY IN LIST
DS XL3                               REMAINDER OF TTR ADDRESS
NEXTLIST EQU *                       NEXT LIST
DSCB DSECT
IECSDSL1 (1)
DSCBLTH EQU *-DSCB-L'DS1DSNAM       LENGTH OF DATA PORTION OF DSCB
TTRMAP DSECT
TTR DS XL3                           TTR TO BE RETURNED
DIRXMP2 CSECT
CVPL CVAFDIR ACCESS=READ, BUFLIST=BUFLIST, MF=L,
      IOAREA=KEEP,
      IXRCDS=KEEP                    KEEP IOAREA TO AVOID OVERHEAD
                                         KEEP VIERS FOR 2ND AND SUBSEQUENT CALLS*
                                         CALLED IN PROGRAM STATE BUT APF
                                         AUTHORIZED SO UCB IS SUPPLIED
                                         OVERLAY CVPL WITH EXPANSION OF MAP
      ORG CVPL
CVPLMAP ICVAFPL DSECT=NO

END

```

図 48. 索引付き VTOC での CVAFDIR マクロの使用例 (2/2)

CVAFDSM マクロの概要および仕様

CVAFDSM マクロは、索引付きまたは索引なし VTOC のボリューム情報を入手するのに使用されます。

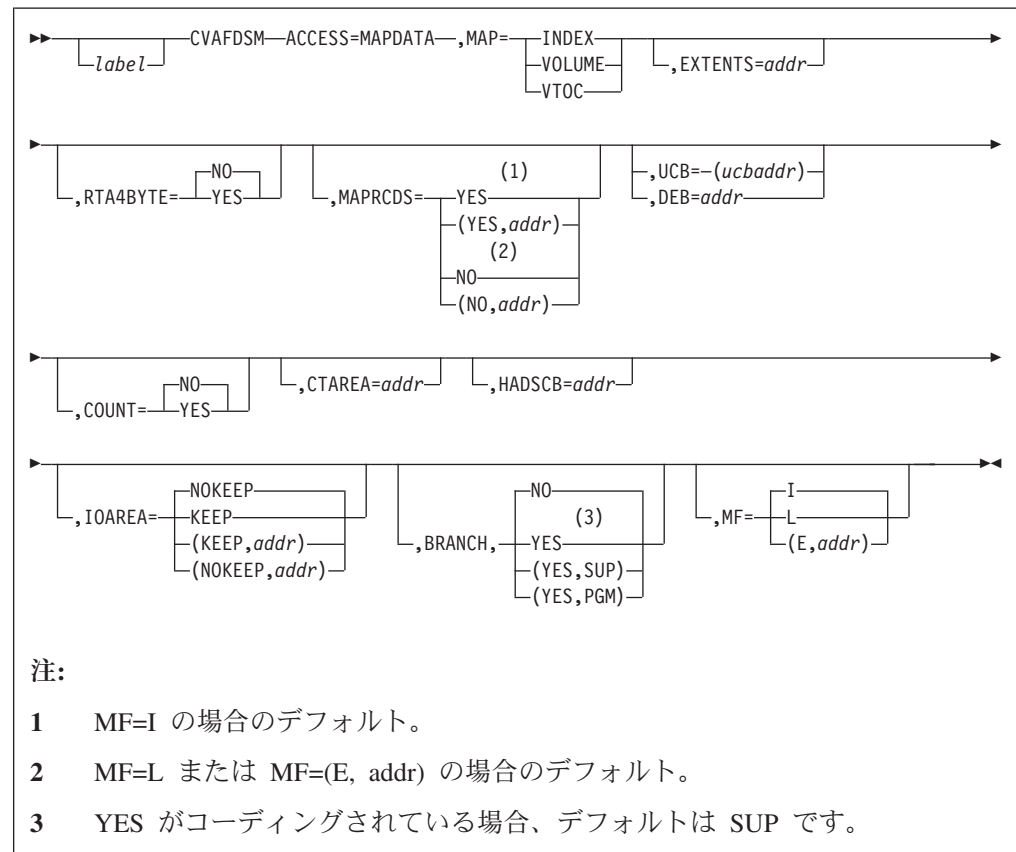
CVAFDSM マクロを索引付き VTOC に対して使用すると、以下の情報を入手できます。

- ボリューム上の未割り振りスペースを記述する 1 つ以上のエクステント。
- VTOC 上のフリー DSCB の数
- VTOC 索引内のフリー VTOC 索引レコードの数
- VTOC INDEX によって決定される、高位割り振り DSCB を示す値

CVAFDSM マクロを索引なし VTOC に対して使用すると、以下の情報を入手できます。

- ボリューム上の未割り振りスペースを記述する 1 つ以上のエクステント。

CVAFDSM マクロのフォーマットは、次のとおりです。



ACCESS: 索引スペース・マップまたは VTOC からの情報の要求

ACCESS=MAPDATA

索引スペース・マップまたはフリー・スペース DSCB からデータを取得します。

索引スペース・マップからは、以下のデータが得られます。

- format-0 DSCB の数 (データは DSCB の VTOC マップから取得されます)。
- 索引内の未割り振り VIR の数 (データは VTOC 索引マップから取得されます)。
- 未割り振りパック・スペースのエクステンツの数 (および位置) (データは VTOC パック・スペース・マップから取得されます)。

索引なし VTOC からは、以下のデータが得られます。

- 未割り振りパック・スペースのエクステンツの数 (および位置)。

MAP: アクセスするマップの識別

MAP=INDEX

VTOC 索引マップ (VIXM) にアクセスして、未割り振り VIR の数を戻すことを指定します。COUNT=YES もコーディングする必要があります。

MAP=VOLUME

索引付き VTOC の場合、VTOC パック・スペース・マップ (VPSM) にアクセスし、パック・スペースの未割り振りエクステントに関する情報を戻すことを指定します。

索引なし VTOC の場合、フリー・スペース DSCB から未割り振りエクステントに関する情報を戻すことを指定します。

索引付きおよび索引なし VTOC の場合、EXTENTS=*addr* および COUNT=NO もコーディングする必要があります。

MAP=VTOC

DSCB の VTOC マップ (VMDS) にアクセスして、format-0 DSCB の数を戻すことを指定します。COUNT=YES もコーディングする必要があります。

EXTENTS: エクステントが戻されるストレージ域**EXTENTS=*addr***

エクステント情報を戻す必要がある場所のアドレスを指定します。このパラメーターは、MAP=VOLUME も指定してボリュームから未割り振りスペース情報を戻すように要求した場合にのみ指定します。

RTA4BYTE=YES が指定されている場合、次のようになります。

- フリー・エクステントに関する情報は、マッピング・マクロ ICVEDT02 を使用します (ICVEDT02 のフォーマットは、49 ページの『マクロ ICVEDT02 を使用したエクステント領域のマップ』を参照してください)。
- EXTENTS= は、未割り振りスペースの 4 バイト相対トラック・アドレスを渡すために使用される制御ブロック (ICVEDT02) のアドレスを指定します。ユーザーはこのストレージを提供し、このマッピング・マクロを使用して、それを初期化する必要があります。

CVAF を呼び出す前に、以下のようにして、ICVEDT02 制御ブロックを初期化してください。

- 最初の 8 バイトに「ICVEDT02」を入れます (DT2X7EYE)。
- 合計領域長さ (バイト) を DT2X7LEN に入れます。この値は、36 + (8 * DT2X7ENT 内の値です)。
- 制御ブロック・レベル番号を「1」に設定します (DT2X7LEV)。(IBM が制御ブロックに影響を与える変更を行った場合、この値は将来のリリースで異なるものになることがあります。)
- 初めて CVAF を呼び出す前に、制御ブロック静的領域 DT2X7FLG から DT2X7RE2 の残りのフィールドをゼロに設定します。追加のエクステントのための呼び出し時には、これらのフィールドは、前の呼び出しが戻した状態のまま変更しないでください。
- DT2X7ENT を、ユーザーが提供するストレージに収まるエクステント記述子項目の合計数に設定します。
- CVAF が検索を開始する相対トラック・アドレス (RTA) を、最初のエクステント域 DT2RTAST(1) の最初の 4 バイトに入れます。

CVAF は、最初のエクステント項目を、提供された開始 RTA より高い値を持つ、次に検出されたフリー・エクステントに関する情報で更新します。後続の各エクステント項目には、追加のフリー・スペース・エクステントに関する情報が記入されます (RTA の昇順で)。

どの呼び出しの場合も、提供されたストレージ域が満杯になる前にボリュームからすべての未割り振りエクステントが戻された場合、残りの項目はゼロに設定されます。この場合、CVAF はレジスター 15 に戻りコード 4 を入れ、CVSTAT フィールドを X'20' に設定して、データ終わりを示します。

提供されたストレージ域が未割り振りエクステントで埋められる前にデータの終わりに達した場合、CVAF はレジスター 15 に戻りコード 4 を設定します。戻りコード 0 がレジスター 15 に設定されている場合、ユーザーは再度 CVAF を呼び出して、残りの未割り振りスペース情報があれば、それを入手できます。CVAF が内部で使用する再始動情報を保管している可能性があるため、ICVEDT02 内のヘッダー情報を変更してはなりません。代わりに、前の CVAF 検索から戻された最後の終了 RTA+1 (DT2RTAED) を DT2RTAST にコピーしてください。

RTA4BYTE=NO が指定されているか、デフォルトである場合、次のようになります。

- フリー・エクステントに関する情報は、XXYYZ 形式です (XXYYZ 形式については、396 ページの『RTA4BYTE: 使用されるエクステント域のタイプの指定』を参照してください)。
- EXTENTS= は、後続の 5 バイト項目の数が入っている 1 バイト・カウント・フィールドのアドレスです。このストレージ域はユーザーが提供します。この領域の長さ (バイト) は、 $1 + (count * 5)$ です。ここで、*count* は、領域の最初のバイトの値です。最初の 5 バイト・エクステント域項目の最初の 2 バイト (「XX」) は、CVAF が検索を開始する地点の相対トラック・アドレス (RTA) です。CVAF は、最初の項目を、提供された開始 RTA より高い値を持つ、次に検出されたフリー・エクステントに関する情報で更新します。後続の各項目には、追加のフリー・スペース・エクステントに関する情報が記入されます (トラック・アドレスの昇順で)。
- CVAF を必要に応じて複数回呼び出すことにより、単一呼び出しで領域が保持できる量を超えるエクステントを検索できます。戻される最初のエクステントは、領域内の最初のエクステント (XXYYZ) に記録された相対トラック・アドレス (「XX」) の後の最初のフリー・エクステントです。
- ボリューム上の最初のフリー・エクステントを検索するには、最初の項目の「XX」をゼロに設定します。追加回数の呼び出し時には、最初の項目の「XX」を、前の呼び出しで戻された最後の相対トラック・アドレスに設定します。

推奨: 大きいボリュームを使用する場合は、エクステント情報を要求するときに RTA4BYTE=YES を指定してください。プログラムで RTA4BYTE=NO

が指定されているか、デフォルトである場合、エクステントが 64x1024 トラック境界を超えると、CVAF 要求は STAT075 の CVSTAT で失敗します。

RTA4BYTE: 使用されるエクステント域のタイプの指定

RTA4BYTE=YES

エクステント域には RTA RTA+1 形式の対のアドレスが入ることを指定します。ここで、RTA は、エクステントの最初のトラックの 4 バイト相対アドレスを表し、RTA+1 は、エクステントの最後のトラックの 4 バイト相対アドレス、プラス 1 を表します。

RTA4BYTE=YES を指定した場合、マクロ ICVEDT02 を使用して、エクステント域をマップする必要があります。説明は、49 ページの表 18 を参照してください。

RTA4BYTE=YES は、索引なし VTOC の場合は必要パラメーターであり、索引付き VTOC の場合はオプション・パラメーターです。

RTA4BYTE=NO

エクステント域は XXYYZ 形式であることを指定します。ここで、

XX	エクステントの最初のトラックの相対トラック・アドレス。
YY	エクステント内の完全なシリンダーの数。
Z	エクステント内の追加トラックの数。

RTA4BYTE パラメーターを指定しない場合、デフォルトは RTA4BYTE=NO です。

MAPRCDS: MAPRCDS バッファ・リストとバッファの保持または解放

MAPRCDS=YES

機能の終了時にバッファ・リストとバッファを保存することを指定します。

MAPRCDS=YES が指定され、バッファ・リストが CVAF パラメーター・リストを通して提供されていない場合、CVAF は VIR を読み取って、CVAF が取得したバッファに入れます。CVAF 呼び出し後に、CVAF パラメーター・リスト・フィールドから、アドレスと VIR の RBA を含んでいるバッファ・リストにアクセスできます。バッファ・リストとバッファは、呼び出し側の保護キー内にあります (呼び出し側が許可されていない場合は、サブプール 0、許可されている場合は、サブプール 229)。

MF=I が指定されているか、デフォルトである場合、MAPRCDS=YES がデフォルトです。

現行ボリューム上の処理が終了したら、保持していたすべての領域を解放してください。

MAPRCDS=(YES,addr)

MAPRCDS=YES がコーディングされているが、バッファ・リスト・アドレス (CVAF パラメーター・リストの CVMRCDS) が提供されている場合、VIR は読み取られません。

1 つの CVAF マクロ呼び出しからの CVMRCDS バッファ・リストを、MAPRCDS キーワードを通して、別の CVAF マクロ呼び出しに渡すことができます。

MAPRCDS=NO

MAP レコード・バッファとバッファ・リストは、CVAFDSM 機能の完了時に解放されることを指定します。

MF=L が指定されている場合、NO がデフォルトです。

MAPRCDS=(NO,addr)

CVAF によって前に取得されたバッファ・リストとバッファを解放します。

CVAF によって取得されたバッファ・リストとバッファは、呼び出し側が解放する必要があります。これは次のいずれかの方法で行うことができます。

- バッファを取得した呼び出しで MAPRCDS=NO をコーディングする。
- 後続の CVAF 呼び出しで MAPRCDS=NO をコーディングする。
- CVAFDIR ACCESS=RLSE をコーディングし、BUFLIST キーワードでバッファ・リストを提供する。

MF=(E,addr) がコーディングされ、MAPRCDS がコーディングされていない場合、MAPRCDS のパラメータ・リスト値は変更されません。

要件: 読み取られた MAP レコードの保全性を維持するために、VTOC をエンキューし、装置を予約する必要があります。

UCB または DEB: アクセスする VTOC の指定

UCB= *rs* タイプまたは (2-12) 標準形式 **UCB=** *rx* タイプまたは (2-12) 実行形式
アクセスする VTOC の UCB のアドレスを指定します。UCB アドレスは、キャプチャーされた UCB のアドレスでも、16 MB 境界より上または下の実 UCB のアドレスでも構いません。UCB コピーではなく、UCB のアドレスを使用してください。無許可の呼び出し側は、このパラメータを使用できません。プログラムが 31 ビット・モードの場合、このアドレスは 31 ビット・アドレスでなければなりません。高位バイトはアドレスの一部です。MF=L と一緒に UCB パラメータをコーディングしてはなりません。

DEB=addr

アクセスする VTOC に対してオープンされた DEB のアドレスを提供します。DEB サブパラメータを指定した場合、CVAF は VTOC または VTOC 索引に対する出力要求を許可しません。CVAF は CVAF 呼び出しの期間中 DEB テーブルからその DEB を除去するので、許可がない場合、ユーザーはその DEB によって表されるデータ・セットに対して非同期アクティビティ (EXCP、CLOSE、EOV など) を実行することはできません。ユーザーが許可されていない場合 (APF 許可もシステム・キーもない場合)、CVAFDSM に対して DEB アドレス (UCB ではなく) を指定してください。詳しくは、44 ページの『ポリシーの識別』を参照してください。

IOAREA キーワードを通して、前に入手した入出力域を提供する場合は、UCB も DEB も提供する必要はありません。そうでない場合は、UCB または DEB のいずれかを提供してください。UCB アドレスを提供した場合、CVPL 内の UCB は入

出力域内の UCB アドレスによってオーバーレイされます。CVPL で UCB と DEB アドレスの両方を提供した場合、DEB アドレスが使用され、CVPL 内の UCB アドレスは、DEB 内の UCB アドレスによってオーバーレイされます。

COUNT: 未割り振り DSCB または VIR 数の取得

COUNT=YES 指定のスペース・マップ内の未割り振り DSCB または VIR の数を要求することを指定します。COUNT=YES がコーディングされている場合、MAP=VTOC または MAP=INDEX を指定する必要があります。

COUNT=NO 未割り振り DSCB または VIR の数は必要ないが、バック上のフリー・スペースに関する情報が必要であることを示します。COUNT=NO がコーディングされているか、デフォルトである場合、MAP=VOLUME をコーディングする必要があります。

CTAREA: Format-0 DSCB の数を入れるフィールドの提供

CTAREA=addr

COUNT=YES、MAP=VTOC が指定されている場合に format-0 DSCB の数を入れる、あるいは COUNT=YES、MAP=INDEX が指定されている場合に VTOC 索引内の未割り振り VIR の数を入れる、4 バイト・フィールドのアドレスを指定します。

HADSCB: 高位割り振り DSCB の CCHHR を入れるフィールドの提供

HADSCB=addr COUNT=YES と MAP=VTOC を指定する場合、VTOC の高位割り振り DSCB の CCHHR を入れる 5 バイト・フィールドのアドレスを指定します。

IOAREA: 入出力作業域の保持または解放

IOAREA=KEEP

CVAF 要求の完了時に、CVAF パラメーター・リストに関連した CVAF 入出力域を保持する必要があることを指定します。呼び出し側が許可されている場合 (APF またはシステム・キー)、BRANCH=NO のみ IOAREA=KEEP をコーディングできます。

IOAREA=KEEP がコーディングされている場合、呼び出し側は、さらに VTOC へのアクセスが必要かどうかに関係なく、将来の何らかの時点で、IOAREA=NOKEEP を指定して CVAF を呼び出す必要があります。そのような呼び出し側の例として、CVAF の呼び出し側のリカバリー・ルーチンがあります。

IOAREA=KEEP をコーディングすると、特定の初期化機能をバイパスできるので、後続の CVAF 要求がより効率的になります。前に取得された CVAF 入出力域が提供される場合、DEB も UCB も指定する必要はありません。どちらも変更できません。

IOAREA=KEEP が最初に発行されたときに、CVAF は、CVAF パラメーター・リストに CVAF 入出力域を戻します (CVIOAR)。後続の CVAF 呼び出しは、その同じパラメーター・リストを使用することができ、CVAF は入出力域を CVIOAR から取得します。

現行ボリューム上の処理が終了したら、保持していたすべての領域を解放してください。

IOAREA=(KEEP,addr)

前に取得した入出力域のアドレスを提供します。異なる CVAF パラメーター・リストが使用されている場合、アドレスを IOAREA キーワードの 2 番目のパラメーターとしてコーディングすることにより、前に取得した CVAF 入出力域を CVAF に渡すことができます。

IOAREA=NOKEEP

CVAF 要求の完了時に、作業域が解放されます。これはデフォルトです。

IOAREA=(NOKEEP,addr)

CVAF 要求の完了時に、前に取得された作業域が解放されます。

BRANCH: マクロの入り口の指定

BRANCH=(YES,SUP)

分岐入力を要求します。呼び出し側は監視プログラム状態でなければなりません。保護キー検査はバイパスされます。

BRANCH=YES がコーディングされている場合、18 ワードの保管域を提供する必要があります。CVAF の入り口では、ロックは保持できません。SRB モードは許可されません。

BRANCH=YES

YES がコーディングされている場合、SUP がデフォルトなので、BRANCH=(YES,SUP) と等価です。保護キー検査はバイパスされます。

BRANCH=(YES,PGM)

分岐入力を要求します。呼び出し側は APF 許可され、問題プログラム状態でなければなりません。保護キー検査はバイパスされます。

BRANCH=NO

SVC 入力を要求します。出力操作を要求する場合、呼び出し側は APF 許可を得ていなければなりません。保護キー検査が実行されます。これはデフォルトです。

MF: マクロの形式の指定

このキーワードは、リスト形式、実行形式、通常形式のマクロのいずれを要求するのかを指定します。

MF=I I がコーディングされている場合、あるいは L も E もコーディングされていない場合、CVAF パラメーター・リストが生成され、CVAF を呼び出すためのコードも生成されます。これはデフォルトです。

MF=L リスト形式のマクロを示します。パラメーター・リストが生成されますが、CVAF を呼び出すためのコードは生成されません。

MF=(E,addr) 実行形式のマクロを示します。addr として提供されたリモート CVAF パラメーター・リストが使用され、実行形式のマクロによって変更できます。

CVAFDSM からの戻りコード

CVAF から戻るときに、レジスター 1 に CVPL (CVAF パラメーター・リスト) のアドレスが入り、レジスター 15 に、以下の戻りコードのいずれかが入ります。

戻りコード	意味
0 (X'00')	要求は、正常に実行されました。
4 (X'04')	データ終わり (CVSTAT は 10 進数 32 に設定)、またはエラーが検出されました。CVPL の CVSTAT フィールドに、エラーの原因が示されます。(CVSTAT コードの説明は、425 ページの『VTOC 索引エラー・メッセージと関連コード』にあります。)
8 (X'08')	無効な VTOC 索引構造。CVSTAT にエラーの原因が示されます。(CVSTAT コードの説明は、425 ページの『VTOC 索引エラー・メッセージと関連コード』にあります。)
12 (X'0C')	以下のいずれかです。 <ul style="list-style-type: none"> • CVAF パラメーター・リストがユーザーの保護キー内に存在しない。 • CVAF パラメーター・リストの保護キーが無効である。 • CVAF パラメーター・リスト ID が無効である。 • CVAF パラメーター・リストの長さが誤っている。 • CVAF パラメーター・リストが変更されなかった。 • 機能コード (CVFCTN) フィールドが無効である。 • 機能コード (CVFCTN) フィールドが、このリリースによってサポートされない。
16 (X'10')	入出力エラーが発生しました。

CVAFFILT マクロの概要および仕様

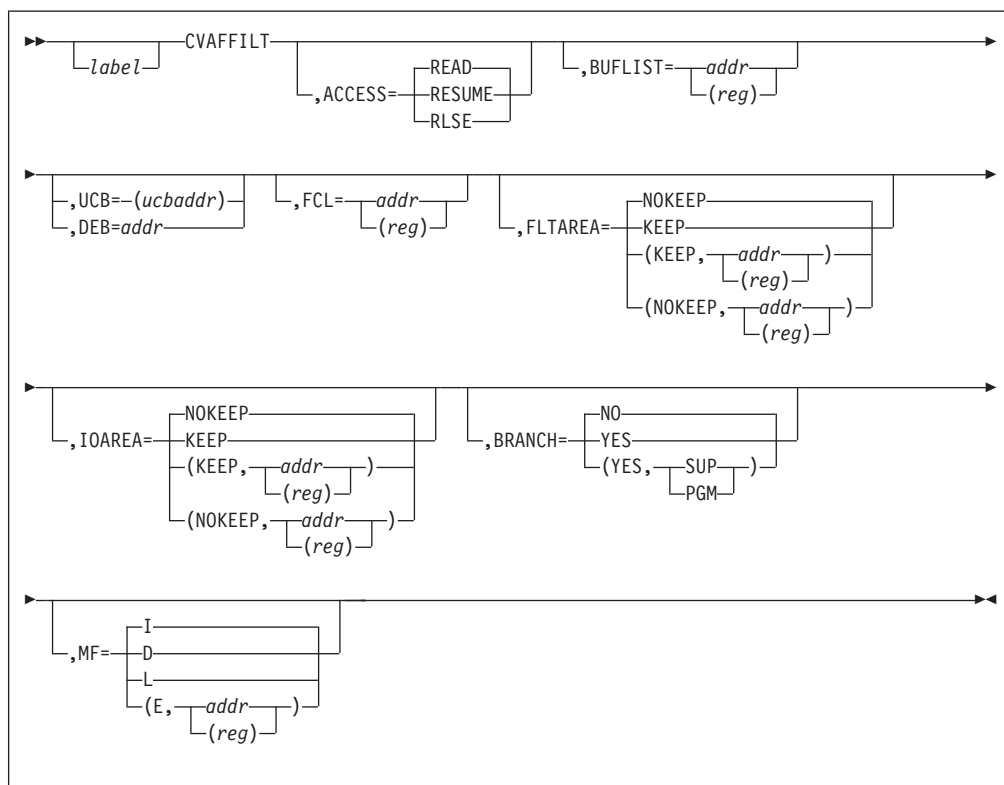
CVAFFILT マクロを使用して、CVAF フィルター・サービスを呼び出すことができます。また、これを使用して、CVAF パラメーター・リスト (CVPL) をマップしたり、初期化したりすることもできます。CVAF フィルターは、索引付きまたは索引なし VTOC からデータ・セット DSCB チェーンを検索し、それをユーザーが提供するバッファーに入れます。単一の部分修飾データ・セット名または完全修飾データ・セット名のリストの DSCB を要求できます。

特定の DASD 装置を識別し、要求を定義するフィルター基準リスト (FCL) と、DSCB を戻すための CVAF バッファー・リスト (バッファーを含む) の両方を提供します。FCL の 2 つのエレメントのフォーマットは、56 ページの表 19 と 58 ページの表 20 に示されています。バッファー・リストのフォーマットは、47 ページの『バッファー・リストの使用』に示されています。CVAFFILT は、DSCB の完全セットを、VTOC にチェーニングされている順序 (format-1、format-2、format-3 の順) で戻します。

リスト形式のマクロでコーディングされたキーワードは、実行形式ではコーディングする必要はありません。同じ CVAFFILT パラメーター・リストを使用する場合、1 つの CVAFFILT 呼び出しでコーディングされたキーワードは、オーバーライドされない限り、後続の呼び出しでも有効です。

追加情報については、55 ページの『CVAF フィルターを使用した DSCB セットの読み取り』を参照してください。

CVAFFILT マクロのフォーマットは、次のとおりです。



制約事項: CVAFFILT の後の第 1 オペランドには、先行コンマをコーディングしないでください。

制御ブロック・アドレス解決:キーワード=*addr* または (*reg*)

ユーザーは、呼び出し側として、CVAF フィルターに必要な制御ブロックを定義または参照します。呼び出し側が定義する制御ブロックは、BUFLIST、CVPL、および FCL です。呼び出し側が参照する制御ブロックは、DEB、FLTAREA、IOAREA、および UCB です。CVAFFILT マクロは、ユーザーが定義制御ブロックを指定するか、参照制御ブロックを指定するかに応じて、**keyword=addr** および **keyword=(reg)** に対して異なる命令を生成します。

- ユーザーが制御ブロックのアドレスを (*reg*) として指定した場合、CVAFFILT マクロは、指定されたレジスターにそのアドレスが入っているものと想定します。
- ユーザーが「定義」制御ブロックのアドレスを *addr* として定義した場合、CVAFFILT マクロは、指定された場所は制御ブロック自体の場所であると想定します。マクロはロード・アドレス命令 (LA) を生成して、制御ブロックのアドレスを取得します。
- ユーザーが「参照」制御ブロックのアドレスを *addr* として定義した場合、CVAFFILT マクロは、指定された場所は制御ブロックのアドレスを含んでいるワードの場所であると想定します。マクロはロード命令 (L) を生成して、制御ブロックのアドレスを取得します。

ACCESS: DSCB の検索、または FLTAREA およびまたは IOAREA の解放

ACCESS=READ

フィルター基準リスト (FCL) に指定されたデータ・セット名に関連したすべての DSCB を検索し、それをユーザーのバッファーに入れます。ユーザーは、1 つ以上の完全修飾名のリスト、または単一の部分修飾名 (単一または 2 重アスタリスク表記を使用) のいずれかを提供することにより、検索された DSCB を選択する (フィルターに掛ける) ことができます。(部分修飾名の例は、405 ページの『CVAFFILT の部分修飾名』を参照してください。)

バッファーの数が、要求された DSCB をすべて保持するには不十分である場合、CVAFFILT はそのことを、CVAF パラメーター・リスト (CVPL) の CVSTAT 状況バイトで示します。ACCESS=RESUME を指定して呼び出しを発行すると、READ 機能を再開できます。426 ページの『CVSTAT フィールドに入るコード』を参照してください。

部分修飾名で DSCB を選択する場合、CVAFFILT は FCL リスト内の最初のデータ・セット名のみを使用します。FCL 内の FCLCOUNT カウント・フィールドを 1 に設定してください。そうしないと、CVAFFILT は CVPL の CVSTAT 状況バイトにエラー・コード 63 を戻します。CVAFFILT によって戻される DSCB は、データ・セット名の順序どおりでない場合があります。ただし、各データ・セットの DSCB は、常に順序どおり (format-1、format-2、format-3) になります。

完全修飾名で DSCB を選択する場合、ユーザーは CVAF フィルターが選択されたデータ・セット名の DSCB を、FCL によって暗黙指定されたデータ・セット名の順序で戻すことを要求できます。56 ページの表 19 の FCL1ORDR フラグを参照してください。

必ず、FCL リスト内の各データ・セット名の状況バイトをテストして、正常終了を確認してください。(一部のエラー条件では、データ・セットの DSCB が戻されません。) 58 ページの表 20 の FCLDSNST バイトを参照してください。

ACCESS=RESUME

要求された DSCB をすべて収容できる十分なバッファーが提供されないために終了した、前に開始された READ または RESUME 機能を再開します。

RESUME 機能を正しく実行するには、前の READ および RESUME 機能呼び出しのそれぞれに、キーワード FLTAREA=KEEP がコーディングされていることが必要です。

ACCESS=RLSE

前に保持されたフィルター保管域 (FLTAREA) およびまたは CVAF 入出力作業域 (IOAREA) を解放します。

BUFLIST: バッファー・リストの指定

BUFLIST=addr または (reg)

DSCB を読み取るために使用されるバッファー・リストのアドレスを指定します。ACCESS=RLSE を指定する場合、標準形式のマクロでは BUFLIST キーワ

ードが必要です。バッファ・リスト・ヘッダーおよびバッファ・リスト項目のフォーマットは、それぞれ 48 ページの表 16 および 49 ページの表 17 を参照してください。

UCB または DEB: アクセスする VTOC の指定

UCB= *rs* タイプまたは (2-12) 標準形式 **UCB=** *rx* タイプまたは (2-12) 実行形式
 アクセスする VTOC の UCB のアドレスを指定します。UCB アドレスは、キャプチャーされた UCB のアドレスでも、16 MB 境界より上または下の実 UCB のアドレスでも構いません。UCB コピーではなく、UCB のアドレスを使用してください。無許可の呼び出し側は、このパラメーターを使用してはなりません。プログラムが 31 ビット・モードの場合、このアドレスは 31 ビット・アドレスでなければなりません。高位バイトはアドレスの一部です。MF=L と一緒に UCB パラメーターをコーディングしてはなりません。

DEB=*addr* または (*reg*)

アクセスする VTOC に対してオープンされた DEB のアドレスを提供します。ユーザーが許可されていない場合、CVAFFILT に対して DEB アドレス (UCB ではなく) を指定してください。また、CVAF は CVAF 呼び出しの期間中 DEB テーブルからその DEB を除去するので、許可されていない場合、ユーザーはその DEB によって表されるデータ・セットに対して非同期アクティビティ (EXCP、CLOSE、EOV など) を実行することはできません。詳しくは、44 ページの『ボリュームの識別』を参照してください。

IOAREA キーワードを通して、前に入手した入出力域を提供する場合は、UCB も DEB も必要ありません。そうでない場合は、UCB または DEB のいずれかを提供してください。UCB アドレスを提供した場合、CVPL 内の UCB は入出力域内の UCB アドレスによってオーバーレイされます。CVPL で UCB と DEB アドレスの両方を提供した場合、DEB アドレスが使用され、CVPL 内の UCB アドレスは、DEB 内の UCB アドレスによってオーバーレイされます。

FCL: フィルター基準リストの指定

FCL=*addr* または (*reg*)

フィルター基準リストのアドレスを提供します。これは、標準形式のマクロで ACCESS=READ が指定されている場合に必要です。フィルター基準リストの 2 つの要素のフォーマットは、56 ページの表 19 と 58 ページの表 20 に示されています。

FLTAREA: フィルター保管域の保持または解放

FLTAREA=KEEP

フィルター保管域を保持することを指定します。このオペランドは、後で RESUME 機能が呼び出される可能性がある場合にコーディングしてください (呼び出し側が提供するバッファの数が、戻される DSCB のすべてを収容するには不十分であるために早期に終了した処理を再開するために)。

CVAFFILT は、保持されたフィルター保管域のアドレスを CVAFFILT パラメーター・リストに戻します (CVFSA フィールド)。後続の RESUME 呼び出しで同じパラメーター・リストを指定すると、CVAFFILT は同じフィルター保管域を再使用します。

ヒント: このオペランドをコーディングした場合、後で ACCESS=RLSE を指定した CVAFFILT を発行して、フィルター保管域を解放する必要があります。

FLTAREA=(KEEP,addr または (reg))

前に取得したフィルター保管域のアドレスを提供します。追加の注意事項については、FLTAREA=KEEP オペランドの説明を参照してください。

FLTAREA=NOKEEP

CVAF 要求の完了時に、フィルター保管域を解放します。

FLTAREA=(NOKEEP,addr または (reg))

CVAF 要求の完了時に、前に取得したフィルター保管域を解放します。

IOAREA: 入出力作業域の保持または解放

IOAREA=KEEP

CVAF 入出力作業域を保持することを指定します。許可された呼び出し側の場合、CVAFFILT は、保持された入出力作業域のアドレスを CVAFFILT パラメーター・リストに戻します (CVIOAR)。後続の呼び出しで同じパラメーター・リストを指定すると、CVAFFILT は同じ入出力作業域を再使用します。

ヒント: このオペランドをコーディングした場合、後で ACCESS=RLSE を指定した CVAFFILT を発行して、入出力作業域を解放する必要があります。

IOAREA=(KEEP,addr または (reg))

前に取得したフィルター保管域のアドレスを提供します。追加の注意事項については、IOAREA=KEEP オペランドの説明を参照してください。

IOAREA=NOKEEP

CVAF 要求の完了時に、フィルター保管域を解放します。

IOAREA=(NOKEEP,addr または (reg))

CVAF 要求の完了時に、前に取得した CVAF 入出力作業域を解放します。

BRANCH: マクロの入り口の指定

BRANCH=NO

SVC (デフォルト) 入力を指定します。保護キー検査が実行されます。

BRANCH=YES

YES をコーディングした場合、SUP がデフォルトなので、BRANCH=(YES,SUP) と等価です。ユーザーは監視プログラム状態でなければなりません。保護キー検査はバイパスされます。

BRANCH=(YES,SUP)

分岐入力を要求します。ユーザーは監視プログラム状態でなければなりません。保護キー検査はバイパスされます。BRANCH=YES を指定した場合は、18 ワードの保管域を提供してください。CVAF の入り口では、ロックを保持できません。SRB モードを使用することはできません。

BRANCH=(YES,PGM)

分岐入力を要求します。ユーザーは APF 許可され、問題プログラム状態でなければなりません。保護キー検査はバイパスされます。

MF: マクロの形式の指定

DSECT 形式、リスト形式、実行形式、通常形式のマクロのいずれを要求するのかを指定します。ユーザーは、24 ビットまたは 31 ビット・アドレッシング・モードのいずれでも構いません。ユーザーが許可されていない場合、OPEN によって作成された DEB のアドレスを渡さなければなりません。ユーザーが許可されている場合は、DEB アドレスまたは UCB アドレスのいずれかを渡すことができます。ボリュームが割り振られ、マウントされたままであること (例えば、動的割り振りによって) を確認する必要があります。

MF=I 標準形式のマクロを指定します。CVAF パラメーター・リストが生成され、CVAF が呼び出されます。デフォルトは MF=I です。

MF=D DSECT 形式のマクロを指定します。このマクロは、固有の CVAF フィルター CVPL (標準 CVPL より 4 バイト長い) をマップするために、ICVAFPL マクロに対する要求を生成します。

MF=L リスト形式のマクロを指定します。CVAF パラメーター・リストが生成されますが、CVAF は呼び出されません。

MF=(E,addr または (reg))

実行形式のマクロを指定します。アドレスが *addr* または *reg* に指定されている CVAF パラメーター・リストが使用されます。この形式のマクロを使用して、パラメーター・リストを変更できます。

CVAFFILT からの戻りコード

CVAF フィルター・サービスは、メッセージを発行しません。CVAF から戻るときに、レジスター 1 に CVAF パラメーター・リストのアドレスが入り、レジスター 15 に、以下の戻りコードのいずれかが入ります。

戻りコード	意味
0 (X'00')	要求は、正常に実行されました。
4 (X'04')	論理エラー。CVSTAT に状況情報が入ります。
8 (X'08')	無効な VTOC 構造。
12 (X'0C')	CVAFFILT パラメーター・リストのキーが誤っているか、無効です。
16 (X'10')	入出力エラー。

CVAF パラメーター・リストの CVSTAT に状況コードが入ります。状況コードのリストは、426 ページの『CVSTAT フィールドに入るコード』を参照してください。

CVAFFILT の部分修飾名

CVAFFILT は、以下の説明文の中に示すように、単一または 2 重アスタリスク表記および % 記号を使用した部分修飾データ・セット名をサポートします。

- 単一アスタリスクを使用して、単一修飾子を表すことができます。例えば、SYS1.*.LOAD は、3 つの修飾子 (最初が SYS1、2 番目は任意の修飾子、3 番目は LOAD) を持つデータ・セットを指定します。
- 単一アスタリスクを使用して、ゼロ個以上の指定されていない文字を表すこともできます。例えば、LOAD.*LIB は、2 つだけの修飾子 (LOAD が最初で、2 番目は文字ストリング LIB で終わる修飾子 (例: LINKLIB)) を持つデータ・セット

を指定します。アスタリスクは、修飾子の中のどの位置にでも使用できます。2つの単一アスタリスクを `LOAD.A*B*.LIB` のように使用することもできます。`CVAFFILT` は、単一修飾子内で他の文字と一緒に複数の単一アスタリスクを使用することはサポートしません (例えば、`LOAD.B**.LIB` は無効です)。

- 2重アスタリスクは、ゼロ個以上の修飾子のプレースホルダーを表します。例えば、`SYS1.**` は、`SYS1` を最初または唯一の修飾子として持つデータ・セットを指定します。
- パーセント記号 (%) を部分修飾名の一部として指定できます。% 記号は、% 記号によって示された位置にある単一文字を除いて、その名前が部分修飾名に一致するデータ・セットを指定します。例えば、`SYS%.*.LOAD%` は、3つの修飾子 (最初は `SYS` で始まる4文字の修飾子、2番目は任意の修飾子、3番目は `LOAD` で始まる5文字の修飾子) を持つデータ・セットを指定します。

CVAFFILT マクロの使用例

この例は、`CVAFFILT` マクロを使用して、指定の VTOC からすべての `format-1`、`format-2`、および `format-3` DSCB を読み取り、フォーマット・タイプ別に DSCB の総数を計算し、総数を呼び出し側プログラム (`CVAF` の呼び出し側ではなく、この例のプログラムの呼び出し側) に戻します。VTOC に対してオープンされた `DEB` のアドレスは、レジスター 1 (例のラベル `R1`) でプログラムに提供されます。

バッファー・リストとフィルター基準リストは、プログラムで定義されています。`ICVAFBFL` マクロはバッファー・リストを生成し、`ICVFCL` マクロはフィルター基準リストを生成します。

```

FLTXMP  CSECT
*
* INPUT ... REGISTER 1 - ADDRESS OF A DEB OPENED TO THE VTOC.
*           REGISTER 13 - ADDRESS OF A STANDARD REGISTER SAVE AREA
*           REGISTER 14 - ADDRESS OF THE RETURN POINT WITHIN CALLER
*
* PROCESS . USE THE CVAFFILT MACRO (ACCESS=READ, ACCESS=RESUME, AN D
*           ACCESS=RLSE) TO READ ALL FORMAT 1, 2, AND 3 DSCBS FROM A
*           GIVEN VTOC. IF FILTER SERVICE DETECTS AN ERROR CONDITION,
*           IT RETURNS DIAGNOSTIC INFORMATION FOR DEBUGGING ANALYSIS.
*
* OUTPUT .. REGISTER 1 = ADDRESS OF THE DATA RETURN AREA (SEE LABEL
*                RET_AREA AT THE END OF THIS LISTING).
*           REGISTER 15 = ZERO IF NO ERRORS WERE ENCOUNTERED.
*                OTHERWISE, ERROR INFORMATION IS PROVIDED
*                IN THE DATA RETURN AREA (SEE LABEL RET_ERR).
*
* THE CVAF PARAMETER LIST (CVPL), FILTER CRITERIA LIST (FCL), BUFFER
* LIST, AND DSCB BUFFERS ARE DEFINED WITHIN THIS CSECT.
*
*****
*           EQUATES FOR ASSEMBLY CONSTANTS AND REGISTERS
*****
BFLE_N  EQU 11  NUMBER OF BUFFER LIST ELEMENTS AND BUFFERS DESIRED
R0      EQU 0
R1      EQU 1
R2      EQU 2
R3      EQU 3
R4      EQU 4
R5      EQU 5
R11     EQU 11
R12     EQU 12
R13     EQU 13
R14     EQU 14
R15     EQU 15
*
*****
*           SAVE CALLER'S REGISTERS AND ESTABLISH A NEW REGISTER SAVE AREA
*****
          STM  R14,R12,12(R13)      SAVE CALLER'S REGISTERS
          BALR R12,0                ESTABLISH THIS PROGRAM'S
          USING *,R12              BASE REGISTER
          ST   R13,SAVEAREA+4      SAVE ADDRESS OF CALLER'S SAVE AREA
          LA   R15,SAVEAREA        GET ADDRESS OF THE NEW SAVE AREA
          ST   R15,8(,R13)        CHAIN CALLER'S AREA TO OURS
          LR   R13,R15            ESTABLISH THE NEW SAVE AREA
*

```

図 49. CVAFFILT マクロの例 (1/5)

```

*****
* ESTABLISH ADDRESSABILITY TO THE CVPL. PLACE GIVEN DEB ADDRESS
* IN THE CVPL, INITIALIZE THE FLAG BYTE AND THE RETURN DATA AREA.
*****
        LA   R11,CVPL_DEF           ESTABLISH ADDRESSABILITY
        USING CVPL_MAP,R11         TO THE CVPL
        ST   R1,CVDEB              PLACE GIVEN ADDR(DEB) IN CVPL
        MVI  FLAGS,F_RSET          RESET THE LOCAL FLAG BYTE
        XC   RET_AREA,RET_AREA     INIT. DATA RETURN AREA TO ZERO
*
*****
* INITIALIZE THE BUFFER LIST HEADER (BFLH) AND ELEMENTS (BFLE)
*****
        XC   BFLH_DEF(BFL_SIZE),BFLH_DEF SET BUFR LIST AREA TO ZERO
        LA   R1,BFLH_DEF           R1 -> BUFFER LIST HEADER
        USING BFL_MAP,R1          ESTABLISH ADDRESSABILITY
        MVI  BFLHNOE,BFLE_N        SET NUMBER OF BUFFER ELEMENTS
        OI   BFLHFL,BFLHDSCB       IDENTIFY AS DSCB BUFR ELEMNT LIST
        LA   R2,BFLH_DEF+BFLHLN    R2 -> FIRST BUFFER LIST ELEMENT
        USING BFLE,R2             ESTABLISH ADDRESSABILITY
        LA   R3,DSCB_DEF           R3 -> FIRST DSCB BUFFER
        LA   R4,BFLE_N             R4 = NUMBER OF ELEMENTS AND BUFRS
BFLE_INT OI   BFLEFL,BFLECHR       REQUEST CCHHR ON RETURN
        MVI  BFLELTH,DSCB_SIZ     SET BUFR LNGTH TO FULL DSCB SIZE
        ST   R3,BFLEBUF           SET ADDR(DSCB BUFFER)
        LA   R2,BFLELN(R2)        R2 -> NEXT BUFFER LIST ELEMENT
        LA   R3,DSCB_SIZ(R3)      R3 -> NEXT DSCB BUFFER
        BCT  R4,BFLE_INT          LOOP THROUGH ALL ELEMENTS
        DROP R1,R2               DROP TEMP USING
*
*****
* INITIALIZE THE FILTER CRITERIA LIST (FCL) HEADER AND ELEMENT
*****
        XC   FCL_DEF(FCL_SIZE),FCL_DEF SET FCL AREA TO ZERO
        LA   R1,FCL_DEF           R1 -> FCL HEADER
        USING FCL_MAP,R1          ESTABLISH ADDRESSABILITY
        MVC  FCLID,FCL_ID         SET THE EYECATCHER 'FCL '
        MVC  FCLCOUNT,=H'1'     SET NUMBER OF FCL ELEMENTS
        LA   R2,FCLHDEND          R2 -> FIRST (ONLY) FCL ELEMENT
        USING FCLDSN,R2          ESTABLISH ADDRESSABILITY
        MVI  FCLDSNLG,X'02'      SET LENGTH(DSN PATTERN)
        LA   R3,=C'***'          R3 -> C'***'
        ST   R3,FCLDSNA          SET ADDR(DSN PATTERN)
        DROP R1,R2               DROP TEMP USING
*

```

図 49. CVAFFILT マクロの例 (2/5)


```

*****
* ISSUE CVAFFILT ACCESS=READ REQUEST
*****
      MVI  RET_FTN,RET_READ      IDENTIFY THE CURRENT FUNCTION
      CVAFFILT ACCESS=READ,FCL=FCL_DEF,BUFLIST=BFLH_DEF,      X
      MF=(E,CVPL_DEF)

*
*****
* TEST THE RETURN FROM CVAFFILT ACCESS=READ OR ACCESS=RESUME
*****
TEST_RET LTR  R15,R15          IF DSCB RETURN IS COMPLETE
          BZ   COUNTLST        GO COUNT LAST SET OF DSCBS
          CH   R15,=H'4'       IF RETURN CODE OTHER THAN FOUR
          BNE  ERR_RET          GO RETURN THE ERROR CONDITION
          CLI  CVSTAT,STAT064   IF OTHER THAN RESUME RECOMMENDED
          BNE  ERR_RET          GO RETURN THE ERROR CONDITION
          B    COUNTCUR        ELSE GO COUNT CURRENT DSCB SET

*
*****
* COUNT THE NUMBER OF FORMAT 1, 2, AND 3 DSCBS RETURNED
*****
COUNTLST OI  FLAGS,F_LAST     INDICATE LAST SET OF DSCBS
COUNTCUR LM  R1,R3,RET_F1     GET PRIOR DSCB COUNTS
          LA   R4,FCL_DEF       R4 -> FCL HEADER
          USING FCL_MAP,R4      ESTABLISH ADDRESSABILITY
          SLR  R5,R5            R5 = ZERO FOR FOLLOWING ICM
          ICM  R5,B'0011',FCLDSCBR R5 = NUMBER OF DSCBS RETURNED
          BZ   TST_RSUM         IF ZERO, GO TEST FOR RESUME
          LA   R4,DSCB_DEF      R4 -> FIRST DSCB BUFFER
          USING DSCB_MAP,R4     ESTABLISH ADDRESSABILITY
COUNTNXT CLI  DS1FMTID,X'F1'   IF FORMAT 1 DSCB
          BE   COUNT_F1         GO INCREMENT ITS COUNTER
          CLI  DS1FMTID,X'F2'   IF FORMAT 2 DSCB
          BE   COUNT_F2         GO INCREMENT ITS COUNTER
          LA   R3,1(R3)         ELSE INCREMENT FORMAT 3 DSCB COUNT
          B    COUNTTST        GO TEST FOR MORE DSCBS

*
COUNT_F2 LA   R2,1(R2)         INCREMENT COUNT OF FORMAT 2 DSCBS
          B    COUNTTST        GO TEST FOR MORE DSCBS

*
COUNT_F1 LA   R1,1(R1)         INCREMENT COUNT OF FORMAT 1 DSCBS
COUNTTST LA   R4,DSCB_SIZ(R4)  R4 -> NEXT DSCB BUFFER
          BCT  R5,COUNTNXT      LOOP THROUGH ALL RETURNED DSCBS
          STM  R1,R3,RET_F1     SAVE UPDATED DSCB COUNTS
          DROP R4               FINISHED COUNTING CURR DSCB SET

*

```

図 49. CVAFFILT マクロの例 (3/5)

CVAF マクロ

```

*****
* CONDITIONALLY ISSUE CVAFFILT ACCESS=RESUME REQUEST
*****
TST_RSUM  TM   FLAGS,F_LAST      IF LAST SET OF DSCBS COUNTED
          BO   RLSE_REQ          GO REQUEST A RELEASE
          MVI  RET_FTN,RET_SUM    IDENTIFY THE CURRENT FUNCTION
          CVAFFILT ACCESS=RESUME,MF=(E,CVPL_DEF) RESUME REQUEST
          B    TEST_RET          GO TEST THE RETURN CODES
*
*****
* ISSUE CVAFFILT ACCESS=RLSE REQUEST
*****
RLSE_REQ  MVI  RET_FTN,RET_RLSE  IDENTIFY THE CURRENT FUNCTION
          CVAFFILT ACCESS=RLSE,FLC=0,BUFLIST=0,FLTAREA=NOKEEP,
          MF=(E,CVPL_DEF)
          LTR  R15,R15            IF NO ERROR ON RELEASE REQUEST
          BZ   RETURN            GO RETURN TO CALLER
          LA   R1,RET_RLSE       ELSE INDICATE ERROR IN RELEASE
*
*****
* PLACE ERROR INFORMATION IN THE OUTPUT DATA RETURN AREA
*****
ERR_RET   MVC  RET_STAT,CVSTAT   COPY CVSTAT TO RETURN AREA
          ST   R15,RET_RC        COPY CVAFFILT'S RETURN CODE
*
*****
* ASSUME R15 = DESIRED RETURN CODE, SET R1 -> DATA RETURN AREA,
* RESTORE CALLER'S REGISTERS 0, 2-14, AND RETURN TO CALLER
*****
RETURN    LA   R1,RET_AREA       R1 -> OUTPUT DATA RETURN AREA
          L    R13,SAVEAREA+4    R13 -> CALLER'S REG SAVE AREA
          L    R14,12(R13)       RESTORE CALLER'S REGISTER 14
          L    R0,20(R13)        RESTORE CALLER'S REGISTER 0
          LM   R2,R12,28(R13)    RESTORE CALLER'S REGISTERS 2-12
          BR   R14               RETURN TO CALLER
          DC   X'00'             LOCAL FLAG BYTE
          F_RSET EQU X'00'       FLAG RESET VALUE
          F_LAST EQU X'80'       LAST DSCB SET ENCOUNTERED
*
          LTORG
*
          SAVEAREA DC 18F'0'     REGISTER SAVE AREA
*

```

図 49. CVAFFILT マクロの例 (4/5)

```

*****
* MAPPING MACROS
*****
CVPL_MAP ICVAFPL CVPLFSA=YES
FCL_MAP ICVFCL
BFL_MAP ICVAFBFL
      PUSH PRINT
      PRINT NOGEN
DSCB_MAP DSECT
      IECSDSL1 (1)      USE FORMAT 1 DSCB MAPPING TO GET BUFFER SIZE
DSCB_SIZ EQU *-IECSDSL1      LENGTH OF FULL DSCB
      POP PRINT
*****
*      SPACE ALLOCATION FOR CVPL, FCL, BFL, AND DSCB BUFFERS
*****
FLTAMP CSECT , CONTINUATION OF CSECT
CVPL_DEF CVAFFILT MF=L,BRANCH=NO,FLTAREA=KEEP
FCL_ID DC CL4'FCL '
FCL_DEF DS (FCLHDLEN+FCLDSNEL)X FCL HEADER AND ONE FCL ELEMENT
FCL_SIZE EQU *-FCL_DEF
* DEFINE A CVAF BUFFER LIST WITH N BUFFER LIST ELEMENTS
BFLH_DEF DS (BFLHLN)X BUFFER LIST HEADER
BFLE_DEF DS (BFLE_N*BFLELN)X N BUFFER LIST ELEMENTS
BFL_SIZE EQU *-BFLH_DEF
* DEFINE N FULL DSCB BUFFERS
DSCB_DEF DS (BFLE_N*DSCB_SIZ)X
*****
*      OUTPUT DATA RETURN AREA
*****
      DS OF
RET_AREA DS 0XL20 OUTPUT DATA RETURN AREA
RET_F1 DC F'0' COUNT OF FORMAT 1 DSCBS RETURNED BY CVAFFILT
RET_F2 DC F'0' COUNT OF FORMAT 2 DSCBS RETURNED BY CVAFFILT
RET_F3 DC F'0' COUNT OF FORMAT 3 DSCBS RETURNED BY CVAFFILT
*
RET_ERR DS 0XL8 IF NO ERROR CONDITION WAS ENCOUNTERED THEN
*
* REGISTER 15 = ZERO ON RETURN
*
* OTHERWISE THE FOLLOWING INFORMATION IS PROVIDED
RET_FTN DC X'00' IDENTIFICATION OF CVAFFILT SUBFUNCTION
RET_READ EQU X'01' CVAFFILT ACCESS=READ
RET_RSUM EQU X'02' CVAFFILT ACCESS=RESUME
RET_RLSE EQU X'03' CVAFFILT ACCESS=RLSE
      DC XL2'00' UNUSED
RET_STAT DC X'00' COPY OF CVSTAT FROM CVAFFILT
RET_RC DC F'0' COPY OF THE RETURN CODE FROM CVAFFILT
      END

```

図 49. CVAFFILT マクロの例 (5/5)

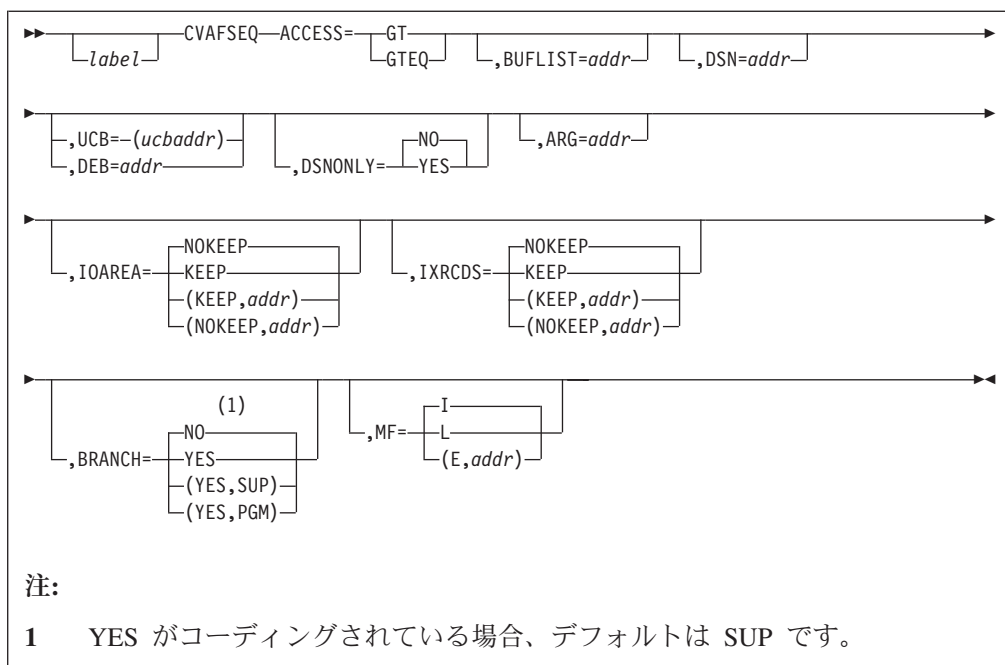
CVAFSEQ マクロの概要および仕様

CVAFSEQ マクロを使用して、以下のことができます。

- 索引付き VTOC をデータ・セット名 (DSN) 順に順次に読み取る。
- 索引付き VTOC または索引なし VTOC を物理順に読み取る。

追加情報については、53 ページの『DSN または DSCB への順次アクセス』を参照してください。

CVAFSEQ マクロのフォーマットは、次のとおりです。



ACCESS: 提供された DSN と戻される DSN の関係の指定

ACCESS=GT DSN または引数値を使用して、DSN または引数が提供された値より大きい DSCB を戻すことを指定します。

ACCESS=GTEQ

DSN または引数値を使用して、DSN または引数が提供された値以上である DSCB を戻すことを指定します。

推奨: ACCESS=GTEQ を指定した CVAF 呼び出しの後に ACCESS=GT 要求を指定する必要があります。そうしないと、同じ DSCB または名前が戻されます。

BUFLIST: 1 つ以上のバッファ・リストの指定

BUFLIST=addr

DSCB または VIR の読み取りまたは書き込みに使用されるバッファ・リストのアドレスを提供します。

DSN: DSN 順または物理順でのアクセスの指定

DSN=addr ゼロまたはデータ・セット名を含んでいる 44 バイト域のアドレスを提供します。DSN キーワードを指定すると、索引付き VTOC に DSN 順でアクセスします。DSNONLY=NO がコーディングされているか、デフォルトである場合、BUFLIST が必要です。

DSN 省略 DSN キーワードを省略した場合、索引付きまたは索引なし VTOC のアクセスは物理順になります。BUFLIST が必要です。

順序が物理順の場合、最初のバッファ・リスト項目の引数フィールドを、ゼロまたは DSCB の引数に初期化します。引数がゼロの場合 (BFLEARG=00)、読み取りは VTOC の開始から始まります。CVAFSEQ マクロの単一の呼び出しで複数の

DSCB を読み取るには、ユーザーは許可 (APF またはシステム・キー) が必要です。詳しくは、54 ページの『物理順次アクセスの開始』を参照してください。

UCB または DEB: アクセスする VTOC の指定

UCB= *rs* タイプまたは (2-12) 標準形式 **UCB=** *rx* タイプまたは (2-12) 実行形式
 アクセスする VTOC の UCB のアドレスを指定します。UCB アドレスは、キャプチャーされた UCB のアドレスでも、16 MB 境界より上または下の実 UCB のアドレスでも構いません。UCB コピーではなく、UCB のアドレスを使用してください。無許可の呼び出し側は、このパラメーターを使用してはなりません。プログラムが 31 ビット・モードの場合、このアドレスは 31 ビット・アドレスでなければなりません。高位バイトはアドレスの一部です。MF=L と一緒に UCB パラメーターをコーディングしてはなりません。

推奨:

- UCB パラメーターのアドレスは、レジスター (2-12) としてのみコーディングしてください。RX タイプのアドレスをコーディングすると、予測不能の結果になります。
- 前の CVAFSEQ 要求から CVPL に戻された UCB アドレスは使用しないでください。特に AMODE=24 の場合、無効場合があります (これは、キャプチャーされた後、キャプチャー解除されたアドレスであるため)。推奨は、IOAREA=KEEP を使用することです。

DEB=*addr*

アクセスする VTOC に対してオープンされた DEB のアドレスを提供します。DEB サブパラメーターを指定した場合、CVAF は VTOC または VTOC 索引に対する出力要求を許可しません。CVAF は CVAF 呼び出しの期間中 DEB テーブルからその DEB を除去するので、ユーザーが許可されていない場合、ユーザーはその DEB によって表されるデータ・セットに対して非同期アクティビティ (EXCP、CLOSE、EOV など) を実行することはできません。ユーザーが許可されていない場合 (APF 許可もシステム・キーもない場合)、CVAFSEQ に対して DEB アドレス (UCB ではなく) を指定してください。詳しくは、44 ページの『ボリュームの識別』を参照してください。

IOAREA キーワードを通して、前に入手した入出力域を提供する場合は、UCB も DEB も提供する必要はありません。そうでない場合は、UCB または DEB のいずれかを提供してください。UCB アドレスを提供した場合、CVPL 内の UCB は入出力域内の UCB アドレスによってオーバーレイされます。CVPL で UCB と DEB アドレスの両方を提供した場合、DEB アドレスが使用され、CVPL 内の UCB アドレスは、DEB 内の UCB アドレスによってオーバーレイされます。

DSNONLY: データ・セット名の読み取りのみを指定

このキーワードは、索引付き VTOC に DSN 順でアクセスする場合にのみ適用されます。

DSNONLY=NO

VTOC 索引からデータ・セット名を取得し、DSCB を読み取って、BUFLIST キーワードを通して提供されたバッファーに入れることを指定します。BUFLIST が必要です。

DSNONLY=YES

VTOC 索引からデータ・セット名を取得することのみを要求します。 ARG キーワードがコーディングされている場合、DSCB の引数が戻されます。

ARG: DSCB の引数を戻す場所の指定

このキーワードは、DSNONLY=YES をコーディングして、索引付き VTOC に DSN 順でアクセスする場合にのみ適用されます。

ARG=addr DSNONLY=YES がコーディングされている場合、VTOC 索引内の各データ・セット名の CCHHR を戻す 5 バイト域のアドレスを提供します。

IOAREA: 入出力作業域の保持または解放**IOAREA=KEEP**

CVAF 要求の完了時に、CVAF パラメーター・リストに関連した CVAF 入出力域を保持する必要があることを指定します。呼び出し側が許可されている場合 (APF またはシステム・キー)、BRANCH=NO でのみ IOAREA=KEEP をコーディングできます。

IOAREA=KEEP がコーディングされている場合、呼び出し側は、さらに VTOC へのアクセスが必要かどうかに関係なく、将来の何らかの時点で、IOAREA=NOKEEP を指定して CVAF を呼び出す必要があります (例えば、CVAF の呼び出し側のリカバリー・ルーチン)。

IOAREA=KEEP をコーディングすると、特定の初期化機能をバイパスできるので、後続の CVAF 要求がより効率的になります。前に取得された CVAF 入出力域が提供される場合、DEB も UCB も指定する必要はありません。また、どちらも変更できません。

IOAREA=KEEP が最初に発行されたときに、CVAF は、CVAF パラメーター・リストに CVAF 入出力域を戻します (CVIOAR)。後続の CVAF 呼び出しは、その同じパラメーター・リストを使用することができ、CVAF は入出力域を CVIOAR から取得します。

現行ボリューム上の処理が終了したら、保持していたすべての領域を解放してください。

注: IOAREA=KEEP がアクティブの間は、連続した CVAFSEQ 要求の中で AMODE=24 と AMODE=31 の間を前後に切り替えてはなりません。これは、CVAF での ABENDS のような問題の原因になります。

IOAREA=(KEEP,addr)

前に取得した入出力域のアドレスを提供します。異なる CVAF パラメーター・リストが使用されている場合、アドレスを IOAREA キーワードの 2 番目のパラメーターとしてコーディングすることにより、前に取得した CVAF 入出力域を CVAF に渡すことができます。

IOAREA=NOKEEP

CVAF 要求の完了時に、作業域が解放されます。

IOAREA=(NOKEEP,addr)

CVAF 要求の完了時に、前に取得された作業域が解放されます。

IXRCDS: VIER を仮想ストレージに保持

このキーワードは、索引付き VTOC にのみ適用されます。

IXRCDS=KEEP

CVAF 機能でストレージに読み取られた VIER を仮想ストレージに保持することを指定します。VIER は、索引機能が失敗しても保存されます。VIER は、CVAF パラメーター・リストからアクセスされます (CVIRCDS)。CVIRCDS は、読み取られた VIR バッファ・アドレスと VIER の RBA を含んでいるバッファ・リストのアドレスです。

見出し検索機能は、バッファ・リストを動的に更新し、必要な場合は、追加のバッファ・リストを取得して、相互をチェーニングします。

IXRCDS=KEEP が指定され、CVPL の CVAF にバッファ・リストが提供されていない場合、CVAF はバッファ・リストとバッファを取得し、最初の高位 VIER を読み取ります。バッファ・リストのアドレスは、CVPL の CVIRCDS フィールドに入れられます。

バッファ・リストとバッファは、呼び出し側の保護キー内にあります。呼び出し側が許可されていない場合、サブプールは 0 です。呼び出し側が許可されている場合は、サブプール 229 です。

索引なし VTOC に対して IXRCDS=KEEP の場合、DSCB を読み取る要求は実行されますが、エラー・コードが戻されます。

現行ボリューム上の処理が終了したら、保持していたすべての領域を解放してください。

IXRCDS=(KEEP,addr)

アドレスを IXRCDS キーワードの 2 番目のパラメーターとして指定することにより、1 つの CVAF 呼び出しからの CVIRCDS を別の CVAF パラメーター・リストに渡すことができます。

IXRCDS=NOKEEP

IXRCDS=NOKEEP がコーディングされている場合、アクセスされた VIER (ある場合) は保存されません。さらに、CVAF パラメーター・リストの CVIRCDS フィールドに提供されたバッファ・リストは解放され、バッファ・リスト内で見つかったすべてのバッファも同様に解放されます。バッファ・リスト内のいずれかの項目にスキップ・ビットが設定されている場合、バッファおよびバッファ・リストは解放されません。

IXRCDS=(NOKEEP,addr)

前にアクセスした VIER を保存しないことを指定します。

ユーザーは、CVAF によって取得されたバッファ・リストとバッファを解放する必要があります。これは 3 通りの方法のいずれかで行うことができます。

- バッファを取得した CVAFSEQ マクロで IXRCDS=NOKEEP をコーディングする。
- 後続の CVAF マクロで IXRCDS=NOKEEP をコーディングする。
- CVAFDIR ACCESS=RLSE をコーディングし、BUFLIST キーワードでバッファ・リストのアドレスを提供する。

要件: 読み取られた VIER の保全性を維持するために、VTOC をエンキューし、装置を予約する必要があります。

BRANCH: マクロの入りの指定

BRANCH=(YES,SUP)

分岐入力を要求します。呼び出し側は監視プログラム状態でなければなりません。保護キー検査はバイパスされます。

BRANCH=YES がコーディングされている場合、18 ワードの保管域を提供する必要があります。CVAF の入り口では、ロックは保持できません。SRB モードは許可されません。

BRANCH=YES

YES がコーディングされている場合、SUP がデフォルトなので、BRANCH=(YES,SUP) と等価です。保護キー検査はバイパスされます。

BRANCH=(YES,PGM)

分岐入力を要求します。呼び出し側は APF 許可され、問題プログラム状態でなければなりません。保護キー検査はバイパスされます。

BRANCH=NO

SVC 入力を要求します。出力操作を要求する場合、呼び出し側は APF 許可を得ていなければなりません。保護キー検査が実行されます。これはデフォルトです。

MF: マクロの形式の指定

このキーワードは、リスト形式、実行形式、通常形式のマクロのいずれを要求するのかを指定します。

MF=I I がコーディングされている場合、あるいは L も E もコーディングされていない場合、CVAF パラメーター・リストが生成され、CVAF を呼び出すためのコードも生成されます。これは、通常形式のマクロです。

MF=L リスト形式のマクロを示します。パラメーター・リストが生成されますが、CVAF を呼び出すためのコードは生成されません。

MF=(E,addr) 実行形式のマクロを示します。addr として提供されたリモート CVAF パラメーター・リストが使用され、実行形式のマクロによって変更できます。

CVAFSEQ からの戻りコード

CVAF から戻るときに、レジスター 1 に CVPL (CVAF パラメーター・リスト) のアドレスが入り、レジスター 15 に、以下の戻りコードのいずれかが入ります。

戻りコード	意味
0 (X'00')	要求は、正常に実行されました。
4 (X'04')	データ終わり (CVSTAT は 10 進数 32 に設定)、またはエラーが検出されました。CVPL の CVSTAT フィールドに、エラーの原因が示されます。エラーの説明は、425 ページの『VTOC 索引エラー・メッセージと関連コード』にあります。

戻りコード	意味
8 (X'08')	無効な VTOC 索引構造。CVSTAT にエラーの原因が示されます。エラーの説明は、425 ページの『VTOC 索引エラー・メッセージと関連コード』にあります。
12 (X'0C')	CVPL (CVAF パラメーター・リスト) が保護キー内に存在しないか、または無効です (ID が無効であるか、長さフィールドが誤っているか、または CVFCTN フィールドが無効です)。CVPL は変更されませんでした。
16 (X'10')	入出力エラーが発生しました。

索引付き VTOC での CVAFSEQ マクロの使用例

この例は、CVAFSEQ マクロを使用して、提供された 2 つのデータ・セット名によって定義される範囲内にあるデータ・セット名を持つ VSAM データ・セットの数を数えます。2 つのデータ・セット名のアドレスは、それぞれレジスター 6 と 7 (ラベル RDSN1 と RDSN2) でプログラムに提供されます。VTOC に対してオープンされた DEB のアドレスは、レジスター 4 (ラベル RDEB) でプログラムに提供されます。

CVAF パラメーター・リストは、リスト形式の CVAFSEQ マクロによって展開されます。リスト形式のマクロで ACCESS=GTEQ が指定されているため、CVPL の最初の実行ではコーディングされていません。CVPL の後続の実行 (ラベル RELOOP) では、ACCESS=GT を指定しています。

データ終わりは、CVSTAT フィールドと STAT032 の値 (ICVAFPL マッピング・マクロによって等価にされている) を比較することによってテストされます。

データ・セット名の基準に一致する VSAM DSCB の数が、レジスター 15 に戻されます。ただし、エラーが検出された場合は、負の 1 がレジスター 15 に戻されます。

```

SEQXMP1  CSECT
          STM 14,12,12(13)
          BALR 12,0
          USING *,12
          ST 13,SAVEAREA+4
          LA RWORK,SAVEAREA
          ST RWORK,8(,13)
          LR 13,RWORK
*****
*
*          REGISTERS
*
*****
REG1     EQU 1          REGISTER 1
RWORK    EQU 3          WORK REGISTER
RDEB     EQU 4          DEB ADDRESS
RDSN1    EQU 6          ADDRESS OF DATA SET NAME 1
RDSN2    EQU 7          ADDRESS OF DATA SET NAME 2
REG15    EQU 15         RETURN CODE REGISTER 15
*****
*
*          COUNT THE NUMBER OF VSAM DATA SETS WHOSE DATA SET NAMES ARE
*          BETWEEN DSN1 AND DSN2 INCLUSIVELY.
*          RDSN1 CONTAINS ADDRESS OF DSN1.
*          RDSN2 CONTAINS ADDRESS OF DSN2.
*          ADDRESS OF DEB OPEN TO VTOC SUPPLIED IN RDEB.
*
*****
          XC BUFLIST(BFLHLN+BFLELN),BUFLIST ZERO BUFFER LIST
          OI BFLHFL,BFLHDSCB DSCBS TO BE READ WITH BUFFER LIST
          MVI BFLHNOE,1 ONE BUFFER LIST ENTRY
          LA RWORK,DS1FMTID ADDRESS OF DSCB BUFFER
          ST RWORK,BFLEBUF PLACE IN BUFFER LIST
          MVI BFLELTH,DSCBLTH DATA PORTION OF DSCB READ - DSN *
          SUPPLIED IN CVPL
          MVC DS1DSNAM,0(RDSN1) MOVE IN STARTING DATA SET NAME TO *
          WORKAREA
          XR RWORK,RWORK ZERO COUNT
          CVAFSEQ DEB=(RDEB), FIND FIRST DATA SET WHOSE DATA SET *
          BUFLIST=BUFLIST, NAME IS GREATER THAN OR EQUAL TO *
          MF=(E,CVPL) THAT OF DSN1
LOOP     EQU * LOOP UNTIL END OF DATA OR DATA SET *
          NAME GREATER THAN DSN2
          USING CVPL,REG1 ADDRESSABILITY TO CVPL
          LTR REG15,REG15 ANY ERROR
          BZ TESTDSN BRANCH IF NOT-CHECK DSN LIMIT

```

図 50. 索引付き VTOC での CVAFSEQ マクロの使用例 (1/3)


```

*****
*
*          UNEXPECTED ERROR PROCESSING
*
*****
LA  RWORK,1(0,0)          ONE IN RWORK
LNR RWORK,RWORK          SET NEGATIVE COUNT INDICATING ERROR
RELEASE CVAFDIR ACCESS=RLSE,  RELEASE CVAF BUFFERS/IOAREA      *
        BUFLIST=0,          DO NOT RELEASE USER BUFFER LIST    *
        IXRCDS=NOKEEP,      RELEASE CVAF VIER BUFFERS          *
        MF=(E,CVPL)         RELEASE CVAF I/O AREA
LR  REG15,RWORK          CURRENT COUNT IS RETURN CODE
L   13,SAVEAREA+4
RETURN (14,12),RC=(15)   RETURN CURRENT COUNT
ERROR4 DC  F'4'          ERROR RETURN CODE 4
BUFLIST ICVAFBFL DSECT=NO  BUFFER LIST
        IECSDSL1 (1)       FORMAT 1 DSCB DATASET NAME AND      *
                           BUFFER
DSCBLTH EQU *-IECSDSL1-L'DS1DSNAM LENGTH OF DATA PORTION OF DSCB
SAVEAREA DS 18F          SAVE AREA
CVPL  CVAFSEQ ACCESS=GTEQ,  READ DSCB WITH DSN >= SUPPLIED DSN *
        IXRCDS=KEEP,        KEEP VIERS IN STORAGE DURING CALLS *
        DSN=DS1DSNAM,       SUPPLIED DATA SET NAME              *
        BUFLIST=BUFLIST,    *
        MF=L
ORG  CVPL                EXPAND MAP OVER LIST
CVPLMAP ICVAFPL DSECT=NO  CVPL MAP
END

```

図 50. 索引付き VTOC での CVAFSEQ マクロの使用例 (3/3)

索引なし VTOC での CVAFSEQ マクロの使用例

この例は、物理順に 5 つまでの DSCB を読み取ります。UCB のアドレスは、レジスター 5 (ラベル RUCB) でプログラムに提供されます。パラメーター・リストのアドレスは、レジスター 4 (ラベル RLIST) に提供されます。パラメーター・リストの最初のワードには、5 バイト・フィールドのアドレスが入ります。プログラムの入り口では、このフィールドは、前に DSCB が読み取られていない場合は、ゼロに設定されています。そうでない場合、このフィールドは、読み取られた最後の DSCB の CCHHR に設定されています。この 5 バイト・フィールドは、このプログラムの呼び出し側によって提供され、このプログラムで変更することはできません。

パラメーター・リストの残りは 1 つ以上の 2 ワード項目からなり、最大で 5 つの 2 ワード項目を含みます。各項目の最初のワードには、140 バイト DSCB バッファァーが入ります。2 番目のワードには、DSCB の CCHHR を入れる 5 バイト・フィールドのアドレスが入ります。

5 つのバッファァー・リスト項目を含むバッファァー・リストが、プログラムに格納されています。ICVAFBFL マクロは、バッファァー・リスト・ヘッダーと 1 つのバッファァー・リスト項目を生成します。ICVAFBFL マクロの後、残りのバッファァー・リスト項目が生成されます。

CVAFSEQ マクロは、プログラムの中で 1 回だけ使用され、パラメーター・リストに存在する 2 ワード項目の数だけの DSCB を読み取ります。バッファァー・リス

ト・ヘッダー・フィールド BFLHNOE は、CVAFSEQ が処理するバッファー・リスト項目数に初期化されます。この数は、このプログラムに提供されたパラメーター・リスト内の 2 ワード項目の数に一致します。

CVAFSEQ 呼び出しの後、読み取られた各 DSCB の CCHHR が、バッファー・リスト項目フィールド BFLEARG から、プログラムの呼び出し側によってアドレスが提供されたフィールドに移動されます。BFLEARG フィールドがゼロの場合、前に読み取られた DSCB は VTOC 内の最後の DSCB だったことになります。

最初のバッファー・リスト項目の BFLEARG は、呼び出し側によって提供された CCHHR に初期化されます。そのアドレスは、パラメーター・リストの 3 番目のワードです。この CCHHR は、CVAFSEQ 呼び出しの開始場所としての役目を果たします。提供された CCHHR より大きい CCHHR を持つ DSCB が読み取られます。

このプログラムは APF 許可が必要です。

```

SEQXMP2 CSECT          STM  14,12,12(13)
        BALR  12,0
        USING *,12
        ST   13,SAVEAREA+4
        LA   RWORK,SAVEAREA
        ST   RWORK,8(,13)
        LR   13,RWORK
*****
*
*       REGISTERS
*
*****
REG1    EQU   1          REGISTER 1
RWORK  EQU   3          WORK REGISTER
RLIST  EQU   4          ADDRESS OF PARM LIST
RUCB   EQU   5          UCB ADDRESS
RCURRENT EQU  6          CURRENT ENTRY IN PARM LIST
RBLE   EQU   7          CURRENT BUFFER LIST ENTRY
RCOUNT EQU   8          COUNT OF ENTRIES IN BUFFER LIST
REG15  EQU  15          RETURN CODE REGISTER 15
*****
*
*       READ UP TO 5 DSCBS.
*       RUCB CONTAINS ADDRESS OF UCB.
*       RLIST CONTAINS ADDRESS OF PARAMETER LIST.
*       WORD 0 = ADDRESS OF CCHHR OF LAST DSCB READ. THIS DSCB IS
*             NOT TO BE READ
*       WORD 1 = ADDRESS OF DSCB BUFFER.
*       WORD 2 = ADDRESS OF CCHHR OF DSCB READ.
*       WORD1 AND WORD2 REPEATED UP TO 4 TIMES.
*       HIGH ORDER BIT OF WORD 2 SET TO ONE FOR LAST ENTRY.
*
*****

```

図 51. 索引なし VTOC での CVAFSEQ マクロの使用例 (1/4)

```

        USING LIST,RLIST          ADDRESSABILITY TO PARM LIST
XC   BFLHDR(BFLHLN+5*BFLLEN),BFLHDR ZERO BUFFER LIST WITH      *
        5 BUFFER LIST ENTRIES
OI   BFLHFL,BFLHDSCB           DSCBS TO BE READ WITH BUFFER LIST
LA   RCURRENT,LISTPRMS         FIRST DOUBLEWORD ENTRY IN PARM LIST
USING LISTPRMS,RCURRENT        USING ON DOUBLEWORDS
LA   RBLE,BFLE                 FIRST BUFFER LIST ENTRY
USING BFLE,RBLE
L    RWORK,LISTSTRT            ADDRESS OF STARTING CCHHR
MVC  BFLEARG,0(RWORK)          MOVE STARTING CCHHR INTO FIRST      *
        BUFFER LIST ENTRY
XR   RCOUNT,RCOUNT            ZERO COUNT
BUFLOOP EQU *                  PUT BUFFER ADDRESSES IN BUFFER LIST *
        ENTRIES
LA   RCOUNT,1(,RCOUNT)        INCREMENT COUNT
L    RWORK,LISTBUF             ADDRESS OF DSCB BUFFER
ST   RWORK,BFLEBUF-BFLE(,RBLE) PLACE IN BUFFER LIST
MVI  BFLELTH-BFLE(RBLE),DSCBLTH FULL DSCB READ
TM   LISTLAST,LASTBIT         IS IT LAST ENTRY IN LIST
LA   RCURRENT,LISTNEXT        INCREMENT TO NEXT ENTRY IN LIST
LA   RBLE,BFLELN(,RBLE)       INCREMENT TO NEXT BUFFER LIST ENTRY
BZ   BUFLOOP                  LOOP TO PUT NEXT BUFFER IN BFLE
STC  RCOUNT,BFLHNOE          SET NUMBER OF ENTRIES IN BUFFER      *
        LIST HEADER
DROP RCURRENT,RBLE
*****
*
*   READ UP TO 5 DSCBS WHOSE CCHHR IS GREATER THAN THE CCHHR IN
*   THE FIRST BUFFER LIST ENTRY
*
*****
CVAFSEQ UCB=(RUCB),           CALL CVAF                      *
        BRANCH=YES,           BRANCH ENTER                      *
        MF=(E,CVPL)
USING CVPL,REG1              ADDRESSABILITY TO CVPL
LTR  REG15,REG15             ANY ERROR
BZ   MOVECHR                 BRANCH IF MOVE IN CCHHRS
*****
*
*   DETERMINE WHAT ERROR IS
*
*****
C    REG15,ERROR4            IS RETURN CODE 4
BNE  OTHERERR                BRANCH IF NOT 4
CLI  CVSTAT,STAT032          IS IT END OF DATA?
BNE  OTHERERR                BRANCH IF NOT
DROP REG1                    ADDRESSABILITY TO CVPL NOT NEEDED

```

図 51. 索引なし VTOC での CVAFSEQ マクロの使用例 (2/4)

```

*****
*
*       DETERMINE IF ANY DSCBS HAVE BEEN READ. BFLEARG IS NON-ZERO
*       IN EACH BUFFER LIST ENTRY FOR WHICH A DSCB HAS BEEN READ
*
*****
MOVECHR EQU *                IS DATA SET NAME GREATER THAN DSN2
        LA  RCURRENT,LISTPRMS  FIRST ENTRY IN PARM LIST
        USING LISTPRMS,RCURRENT
        LA  RBLE,BFLE          FIRST BUFFER LIST ENTRY
        USING BFLE,RBLE
CHRLOOP EQU *                MOVE CCHHR ARGUMENT TO CALLER AREA
        L   RWORK,LISTCHR      ADDRESS OF CCHHR OF CALLER
        XC  0(L'BFLEARG,RWORK),0(RWORK) ZERO CALLER CCHHR AREA
        NC  BFLEARG,BFLEARG     IS CCHHR ZERO
        BZ  EXIT                BRANCH IF YES-NO MORE DSCBS
        MVC 0(L'BFLEARG,RWORK),BFLEARG MOVE CCHHR TO CALLER AREA
        TM  LISTLAST,LASTBIT    LAST ENTRY IN PARM LIST?
        BO  EXIT                BRANCH IF YES
        LA  RCURRENT,LISTNEXT   NEXT ENTRY IN LIST
        LA  RBLE,BFLELN(,RBLE)  NEXT BUFFER LIST ENTRY
        B   CHRLOOP            TEST NEXT BFLE
EXIT    EQU *                RETURN TO CALLER
        L   13,SAVEAREA+4
        RETURN (14,12)
OTHERERR EQU *              ERROR PROCESSING
*
*
*
ERROR4  B   EXIT              RETURN
        DC  F'4'              RETURN CODE 4
        ICVAFBFL DSECT=NO     BUFFER LIST WITH ONE BUFFER LIST *
                                ENTRY
        DS  CL(4*BFLELN)      FOUR BUFFER LIST ENTRIES
SAVEAREA DS 18F              SAVE AREA
DSCB     DSECT
        IECSDSL1 (1)          FORMAT 1 DSCB DATASET NAME AND *
                                DATA
DSCBLTH EQU *-IECSDSL1      LENGTH OF DSCB
LIST     DSECT              PARAMETER LIST
LISTSTRT DS F              ADDRESS OF CCHHR TO START SEARCH
LISTPRMS EQU *
LISTBUF  DS F              BUFFER ADDRESS
LISTCHR  DS 0F             ADDRESS OF CCHHR FIELD
LISTLAST DS X              BYTE
LASTBIT  EQU X'80'         LAST DOUBLE WORD
        DS AL3              3 BYTE ADDRESS OF CCHHR
LISTNEXT EQU *              NEXT DOUBLEWORD
SEQXMP2  CSECT

```

図 51. 索引なし VTOC での CVAFSEQ マクロの使用例 (3/4)

```

*****
*
*      READ DSCBS WITH CCHHR GREATER THAN THE CCHHR IN THE FIRST
*      BUFFER LIST ENTRY.
*
*****
CVPL      CVAFSEQ ACCESS=GT,                                *
          BUFLIST=BFLHDR,      ADDRESS OF BUFFER LIST      *
          MF=L
          ORG CVPL                                EXPAND MAP OVER LIST
CVPLMAP  ICVAFPL DSECT=NO      CVPL MAP
          END
    
```

図 51. 索引なし VTOC での CVAFSEQ マクロの使用例 (4/4)

CVAFTST マクロの概要および仕様

CVAFTST マクロは、システムが索引付き VTOC をサポートするかどうかを判別し、サポートする場合、UCB が提供されている装置上の VTOC が索引付きか、索引なしかを判別します。

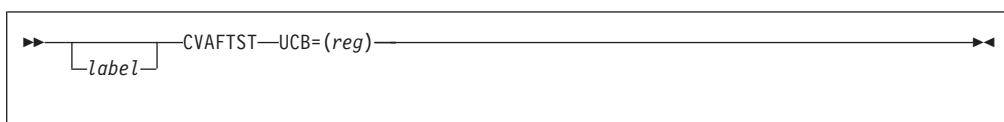
CVAFTST を発行する場合、レジスター 13 に標準 18 ワード保管域のアドレスが入っていない必要があります。

CVAFTST が装置のボリューム上の VTOC は索引付き、索引なしかを判別できない場合、戻りコード 12 を受け取ります。ボリューム上のデータ・セット (VTOC を含む) をオープン済みの場合には、CVAFTST から 12 の戻りコードは受け取らないはずです。

CVAFTST マクロを発行するためには、許可は必要ありません。

CVAFDSM マクロでの CVAFTST マクロの使用例は、392 ページの『CVAFDSM マクロの概要および仕様』を参照してください。

CVAFTST マクロのフォーマットは、次のとおりです。



UCB: テストする VTOC の指定

UCB=(reg)

VTOC をテストするボリュームの UCB のアドレスを提供します。UCB アドレスは、キャプチャーされた UCB のアドレスでも、16 MB 境界より上または下の実 UCB のアドレスでも構いません。プログラムが 31 ビット・モードの場合、このアドレスは 31 ビット・アドレスでなければなりません。高位バイトはアドレスの一部です。

推奨: UCB パラメーターのアドレスは、レジスター (2-12) としてコーディングしてください。RX タイプのアドレスをコーディングすると、予測不能の結果になります。

CVAFST マクロは、UCB または UCB コピーのアドレスを受け入れます。無許可プログラムは、UCBSCAN マクロを使用して COPY、UCBAREA、CMXTAREA、および DCEAREA キーワードを指定することにより、UCB のコピーを入手できます。UCB コピーおよび共通拡張コピーは、16 MB 境界より下のワード境界上になければなりません。DCEAREA でアクセスされるデータは、16 MB 境界より上にあっても構いません。詳細については、「z/OS ハードウェア構成定義(HCD) 計画」を参照してください。

CVAFST からの戻りコード

CVAF から戻ると、レジスター 15 に、以下の戻りコードのいずれかが入ります。

戻りコード	意味
0 (X'00')	システムは索引付き VTOC をサポートしません。ボリュームは索引なし VTOC を持っているとなす必要があります。UCB の妥当性または状況を判別するための検査は行われませんでした。
4 (X'04')	システムは索引付き VTOC をサポートしますが、ボリュームは索引なし VTOC を持っています。
8 (X'08')	システムは索引付き VTOC をサポートし、ボリュームは索引付き VTOC を持っています。
12 (X'0C')	システムは索引付き VTOC をサポートしますが、ボリュームがマウントされていないか、VIB がそれ用に初期化されていません。そのため、VTOC の状況 (索引付き、または索引なし) を判別できません。
16 (X'10')	システムは索引付き VTOC をサポートしますが、装置が DASD でないか、VIO UCB を持っている、あるいは UCB アドレスが無効です。UCB コピーのアドレスは、CMXTAREA と DCEAREA がなければ無効です。

VTOC 索引エラー・メッセージと関連コード

エラー・メッセージ

CVAF は、VTOC 索引内でエラーを検出した場合、次のメッセージを発行します。

```
IEC606I VTOC INDEX DISABLED ON dev,volser,code,[rba[,secno,offset]]
```

それに加えて、CVAF は戻りコードを CVPL の CVSTAT フィールドに入れます。

説明

共通 VTOC アクセス機能 (CVAF) は、ボリューム通し番号 *volser* を持つ装置 *dev* 上で VTOC 索引エラーを検出しました。VTOC 索引エラーの種類を表す番号が *code* フィールドに提供されます。 *code* によって示された構造エラーを含む VTOC 索引内の VIR の RBA が、*rba* フィールドに提供されます。VIR が VIER である場合、VTOC 索引項目を含んでいる VIER 内のセクション番号が *secno* フィールドに提供され、その VTOC 索引項目のセクションへのオフセットが *offset* フィールドに提供されます。

システム処置

VTOC 索引は使用不可にされます。DADSM が次回にボリューム上にスペースを割り振るときに、VTOC は索引なしフォーマットに変換されます。システム・ダンプが SYS1.DUMP データ・セットに書き込まれ、SYS1.LOGREC データ・セットに項目が作成されます。後で、メッセージ IEC604I (VTOC 変換ルーチンが使用されたことを示す) が発行されます。

プログラマー応答

システム・ダンプおよび VTOC 索引の印刷を調べ、メッセージ IEC606I 内の情報を使用して、VTOC 索引構造エラーの原因を判別します。

宛先コードと記述子コード

宛先コードは、4 (直接アクセス・プール) および 10 (システム/エラー保守)、記述子コードは、4 (システム状況) です。

CVSTAT フィールドに入るコード

エラーを診断する際に、CVSTAT フィールドのコードの説明が必要な場合は、「z/OS DFSMSdfp 診断解説書」を参照してください。

付録 B. 制御ブロック

この付録では、以下の制御ブロックについて説明します。

- EXCP および EXCPVR のデータ・エクステント・ブロック (DEB) (DEB 内のフィールドについては、「z/OS DFSMSdfp 診断解説書」を参照してください)。
- データ機能域 (DFA)。

データ・エクステント・ブロック (DEB) フィールド

ここに示されているデータ・エクステント・ブロック (DEB) フィールドは、EXCP および EXCPVR でのみ 使用してください (DEB 内のすべてのフィールドについては、「z/OS DFSMSdfp 診断解説書」を参照してください)。

共通名:	データ・エクステント・ブロック
マクロ ID:	IEZDEB
DSECT 名:	DEB (AVT セクションに先行する DSECT ステートメント) DEBBASIC は、基本セクションに使用します。DEBDASD (アクセス方式セクションの DSECT 名)。DEBACSMO (アクセス方式セクションの DSECT 名)。DEBSUBNM (DSECT サブルーチン名セクションの名前)。DEBXTN (DEB 拡張の DSECT 名)。
サイズ:	可変 (装置およびアクセス方式依存セクション)
作成者:	OPEN
指示:	DCB データ域の DCBDEBAD フィールド。DEB データ域の DEBDEBAD フィールド (チェーン内の次の DEB)
機能:	DEB は、DCB 内の情報の拡張です。各 DEB は DCB に関連付けられており、この 2 つは相互を指します。これは、データ・セットの物理的特性に関する情報、および制御プログラムが使用するその他の情報を含んでいます。

表 90. DEB フィールドの部分リスト

オフセット	タイプ	長さ	名前	説明
DEBAPPAD によってポイントされる DEB の付加ベクトル・テーブル・セクション				
0(0)		20	DEBAVT(0)	付加ベクトル・テーブル
0(0)	アドレス	4	DEBEOEA(0)	エクステント終了付加ルーチンのアドレス
0(0)	ビット・ストリング	1	DEBEOEAB	フラグ・バイト
 xxxx		DEBEOENP	X'0F' - エクステント終了付加ルーチンのために固定される 2K ページの数。
1(1)	アドレス	3	DEBEOEAD	エクステント終了付加ルーチンのアドレス
4(4)	アドレス	4	DEBSIOA(0)	入出力開始付加ルーチンのアドレス

制御ブロック

表 90. DEB フィールドの部分リスト (続き)

オフセット	タイプ	長さ	名前	説明
4(4)	ビット・ストリング 1	1	DEBSIOAB	フラグ・バイト
	.1		DEBPGFX	X'80' - DEBSIOAD 内のアドレスを使用し、DEBSIOAD 内のアドレスに 4 を加算することによって、ページ固定 (PGFX) 付加ルーチンへのエントリ・ポイントを判別できます。
	. .1		DEBSIOX	X'40' - ゼロの場合、ERP がアクティブになっていると、SIO 付加ルーチンに入りません。1 の場合は、ERP がアクティブでも SIO 付加ルーチンに入ります。
	. . .1 xxxx		DEBIOVR	X'20' - 1 の場合、EXCPVR 要求は有効です。ゼロの場合、EXCPVR 要求は無効で、実行されません。現在、影響はありません。
			DEBFIX	X'10' DEB が固定されていることを示します。
			DEBSIONP	X'0F' - SIO 付加ルーチンのために固定される 2K ページの数。
5(5)	アドレス	3	DEBSIOAD	入出力開始付加ルーチンのアドレス
8(8)	アドレス	4	DEBPCIA(0)	PCI 付加ルーチンのアドレス
8(8)	ビット・ストリング xxxx	1	DEBPCIAB	フラグ・バイト
			DEBPCINP	X'0F' - PCI 付加ルーチンのために固定される 2K ページの数。
9(9)	アドレス	3	DEBPCIAD	プログラム制御割り込み (PCI) 付加ルーチンのアドレス
12(C)	アドレス	4	DEBCEA(0)	チャンネル終了付加ルーチンのアドレス
12(C)	ビット・ストリング 1 xxxx	1	DEBCEAB	フラグ・バイト
			DEBESMVR	X'80',C'X' EXCPVR 呼び出し側の妥当性検査
			DEBCENP	X'0F' - チャンネル終了付加ルーチンのために固定される 2K ページの数。
13(D)	アドレス	3	DEBCEAD	チャンネル終了付加ルーチンのアドレス
16(10)	アドレス	4	DEBXCEA(0)	異常終了付加ルーチンのアドレス
16(10)	ビット・ストリング xxxx	1	DEBXCEAB	フラグ・バイト
			DEBXCENP	X'0F' - 異常終了付加ルーチンのために固定される 2K ページの数
17(11)	アドレス X"	3	DEBXCEAD	異常終了付加ルーチンのアドレス
			DEBAVTE	"*" 付加ベクトル・テーブルの終わり

DEB 接頭部テーブル。DEB 基本セクションから負のオフセットとしてアドレス可能です。必ずしも DEB 付加ベクトル・テーブルと隣接していません。

-16(10)		16	DEBPREFIX(0)	DEB 接頭部テーブル
-16(10)	ビット・ストリング	1	DEBWKARA	O/C/E 作業域 (直接アクセス)
-15(F)	BITSTRING	7	DEBDSCBA	O/C/E (直接アクセス) によって使用される DSCB アドレス (BBCCHHR)
-8(8)	アドレス	4	DEBXTNP(0)	DEB 拡張へのポインター
-4(4)	ビット・ストリング	1	DEBLNGTH	DEB の長さ (ダブルワード)
-3(3)	文字 X'0' X'1'	1	DEBAMTYP	アクセス方式タイプ
			DEBAMNON	"0" アクセス方式タイプが不明
			DEBAMVSM	"1" VSAM アクセス方式タイプ

表 90. DEB フィールドの部分リスト (続き)

オフセット	タイプ	長さ	名前	説明
	X'2'		DEBAMXCP	"2" EXCP アクセス方式タイプ
	X'4'			予約済み
	X'8'		DEBAMGAM	"8" グラフィックス・アクセス方式タイプ
	X'10'		DEBAMTAM	"16" BTAM アクセス方式タイプ
	X'20'		DEBAMBPM	"32" BPAM アクセス方式タイプ
	X'20'		DEBAMSAM	"32" 順次アクセス方式タイプ
	X'40'		DEBAMBDM	"64" 直接アクセス方式タイプ
	X'80'		DEBAMISM	"128" ISAM アクセス方式タイプ
	X'81'		DEBAMSUB	"129" サブシステム・アクセス方式タイプ
	X'82'		DEBAMVTM	"130" VTAM® アクセス方式タイプ
	X'84'			予約済み
	-2(2)	UNSIGNED	2	DEBTBLOF
	X'24'		DEBPREFE	"*" DEB 接頭部テーブルの終わり
DEB 基本セクション				
0(0)			DEBBASIC	"*"
0(0)	アドレス	4	DEBTCBAD(0)	この DEB 用の TCB のアドレス
0(0)	ビット・ストリング	1	DEBNMSUB	OPEN 実行プログラム・ルーチンによってロードされ、サブルーチン名セクション (DEBSUBID) に識別されているサブルーチンの数
1(1)	アドレス	3	DEBTCBB	この DEB 用の TCB のアドレス
4(4)	アドレス	4	DEBDEBAD(0)	同じタスク内の次の DEB のアドレス
4(4)	ビット・ストリング	1	DEBAMLNG	アクセス方式依存セクションのバイト数。BDAM の場合、このフィールドにはワード数で示された長さが入ります。
5(5)	アドレス	3	DEBDEBB	同じタスク内の次の DEB のアドレス
8(8)	アドレス	4	DEBIRBAD(0)	付加非同期出口で使用される IRB ストレージ・アドレス
8(8)	ビット・ストリング	1	DEBOFLGS	データ・セット状況フラグ
	xx		DEBDISP	X'C0'- データ・セット処理フラグ。ビットは、処理を設定します。
	01		DEBDSOLD	X'40' 古いデータ・セット
	10		DEBDSMOD	X'80' 変更データ・セット
	11		DEBDSNEW	X'C0' 新規データ・セット
	..1.		DEBEOF	X'20'- ファイルの終わり (EOF) が検出されました (テープ入力)。format 1 DSCB ビット 93.0 は、現行ボリュームはデータ・セットの最後のボリュームであることを示します (DASD 入力)
	...1		DEBRLSE	X'10' - 第 2 世代フォーマットを持つ、未使用の外部ストレージ (DASD) エミュレーター・テープを解放します。テープは、12 文字より短いブロックを含んでいる可能性があります。(テープ)
1.		DEBSPLIT	X'04' - 混合パリティ・レコードを含んでいる可能性がある 7トラック・エミュレーター・テープ (テープ)
1.		DEBLABEL	X'02'- 標準外ラベル
1		DEBRERR	X'01'- 短縮エラー・リカバリー・プロシージャの使用 (テープ) 連結区分編成データ・セットが BPAM を使用して処理されました (DASD)
9(9)	アドレス	3	DEBIRBB	付加非同期出口で使用される IRB ストレージ・アドレス
12(C)	ビット・ストリング	1	DEBOPATB	入出力処理と、ボリュームの終わり (EOV) 条件が発生した場合に実行される処理の両方を示すフラグ。

制御ブロック

表 90. DEB フィールドの部分リスト (続き)

オフセット	タイプ	長さ	名前	説明
	1.		DEBABEND	X'80' - ABEND によって設定され、SYSABEND または SYSUDUMP データ・セットを示します。
	.1.		DEBZERO	X'40' - 常にゼロ
	. .xx. . . .		DEBPOSIT	X'30' - データ・セット位置決めフラグ。ビットは、位置決めを設定します。
	. . 01		DEBRERED	X'10' REREAD
	. .11		DEBLEAVE	X'30' LEAVE
 xxxx		DEBACCS	X'0F' - 実行された入出力アクセスのタイプ。ビットは、アクセスを設定します。
 0000		DEBINPUT	X'0' INPUT
 1111		DEBOUTPT	X'F' OUTPUT または EXTEND
 0011		DEBINOUT	X'3' INOUT
 0111		DEBOUTIN	X'7' OUTIN または OUTINX
 0001		DEBRDBCK	X'1' RDBACK
 0100		DEBUPDAT	X'4' UPDAT
13(D)	ビット・ストリング	1	DEBQSCNT	PURGE (SVC 16) - 静止カウント。ユーザーのチャンネル・プログラムを実行している装置の数。UCBFL1 フィールドのビット 5 と 6 で示されます。
14(E)	ビット・ストリング	1	DEBFLGS1	フラグ・フィールド
	1.		DEBPWCKD	X'80' - OPEN 時にパスワードが提供されました。マルチボリューム・データ・セットのボリュームが追加されるたびに、EOV からパスワードを要求されることはありません。
	.1.		DEBEOFDF	X'40' - ファイルの終わりが検出されたので、据え置きユーザー・ラベル処理が許可されることを CLOSE に知らせるために、EOV によって設定されます。
	. . .1		DEBEXCPA	X'10' - この DEB に対して EXCP(VR) が許可されています。
1. .		DEBF1CEV	X'04' - EXCP DCB では無効です。
1.		DEBAPFIN	X'02' - オンの場合、許可プログラムをロードできます。
1		DEBXTNIN	X'01' - 1 の場合、DEB 拡張が存在します。
15(F)	ビット・ストリング	1	DEBFLGS2	フラグ・フィールド 2
1. . .		DEBDSCMP	X'08' テープ・データ・セット圧縮モード
1. .		DEBDSNCP	X'04' テープ・データ・セット非圧縮モード
16(10)	アドレス	4	DEBUSRPG(0)	ページされた入出力復元リスト (PIRL) のアドレス
16(10)	ビット・ストリング	1	DEBNMEXT	DEBBASND から開始されるエクステント記述の数。拡張フォーマットまたは PDSE データ・セットの場合、装置ごとに 1 エクステント。
17(11)	アドレス	3	DEBUSRPB	ページされた入出力復元リスト (PIRL) のアドレス
20(14)	アドレス	4	DEBRRQ(0)	関連要求キューへのポインター
20(14)	ビット・ストリング	1	DEBPRIOR	タスク所有 DEB の優先順位
24(18)	アドレス	4	DEBDCBAD(0)	この DEB に関連した DCB または ACB のアドレス
24(18)	ビット・ストリング	1	DEBPROTG(0)	高位 4 ビットのタスク記憶保護キー

表 90. DEB フィールドの部分リスト (続き)

オフセット	タイプ	長さ	名前	説明
24(18)	ビット・ストリング	1	DEBDEBID	このブロックを DEB として識別するための下位 4 ビットの 16 進数 F。
25(19)	アドレス	3	DEBDCBB	この DEB に関連した DCB または ACB のアドレス
28(1C)	アドレス	4	DEBAPPAD(0)	入出力付加ベクトル・テーブルのアドレス
28(1C)	アドレス	1	DEBEXSCL	このフィールドは、装置に依存したセクションのサイズを決定するために使用されます。2 をこの値で累乗したものが、DEBBASND での装置依存セクションの長さとなります。エクステンツ・スケールは、直接アクセス装置および装置関連データ・セットをサポートする 3525 カード・パンチの場合は 4 (16 バイト)、非直接アクセス装置および通信装置の場合は 2 (4 バイト) です。
29(1D)	アドレス	3	DEBAPPB	入出力付加ベクトル・テーブルのアドレス
32(20)			DEBBASND	基本セクションの終了
ユニット・レコード、磁気テープ、遠隔通信装置セクション				
32(20)			DEBDDS1	"*"
32(20)	アドレス	4	DEBSUCBA(0)	所定のデータ・セットに関連した UCB のアドレス。常に 16 MB 境界より下です。実 UCB が 16 MB 境界より上にある場合、割り振りは通常、UCB をキャプチャーして 24 ビット・アドレスを作成します。EXCP を使用して、動的割り振りの NOCAPTURE オプションを指定した場合、OPEN または EOVS は、後の EOVS または CLOSE まで UCB をキャプチャーします。
32(20)	ビット・ストリング	1	DEBSDVM	装置修飾子。磁気テープの場合は、MODESET 命令コードまたはモード設定機能バイト。装置レコードの場合は、予約済み。
	1101 0011		DEBMTDN4	X'D3' 9 トラック MODESET CCW コード、密度=6250BPI
	1100 0011		DEBMTDN3	X'C3' 9 トラック MODESET CCW コード、密度=1600BPI
	1100 1011		DEBMTDN2	X'CB' 9 トラック MODESET CCW コード、密度= 800BPI 7 トラック・テープ MODESET スケルトン・コード (パリティ、変換および/または型変換を備えている必要があります)
	0000 0011		DEBM7DN0	X'03' 7 トラック MODESET スケルトン、密度=200BPI
	0100 0011		DEBM7DN1	X'43' 7 トラック MODESET スケルトン、密度=556BPI
	1000 0011		DEBM7DN2	X'83' 7 トラック MODESET スケルトン、密度=800BPI 3480 テープ命令コード
	1100 0011		DEBMSTWI	X'C3' 3480 テープ即時書き込み設定 CCW コード。テープのモード設定機能バイト
	1.		DEBMTRF0	X'80' テープ記録フォーマット・ビット 0
	.1.		DEBMTRF1	X'40' テープ記録フォーマット・ビット 1
	. .1.		DEBMTWI	X'20' テープ即時書き込み (非バッファ書き込み)
	. . .1.		DEBMINHS	X'10' INHIBIT 監視プログラム・コマンド
1.		DEBMCOMP	X'08' 圧縮記録モード
1.		DEBCMPAC	「DEBMCOMP」圧縮記録モード
1.		DEBM3424	X'02' 3424 モード設定フラグ
1.		DEBMINHE	X'01' 制御装置 ERP 禁止
	X'C2'		DEBM6250	「DEBMTRF0+ DEBMTRF1+ DEBM3424」3424 設定、密度=6250BPI
	X'42'		DEBM1600	「DEBMTRF1+ DEBM3424」3424 設定、密度=1600BPI

制御ブロック

表 90. DEB フィールドの部分リスト (続き)

オフセット	タイプ	長さ	名前	説明
33(21)	アドレス	3	DEBSUCBB	指定のデータ・セットに関連した UCB のアドレス
36(24)	アドレス	4	DEBDEVED(0)	テープと装置の共通レコード・フィールドの終わり
以下のフィールドは、装置に関連したデータ・セットがサポートされている 3525 に限り存在します。				
36(24)	アドレス	4	DEBRDCB(0)	
36(24)	ビット・ストリング	1	DEBRSV06	
37(25)	アドレス	3	DEBRDCBA	
40(28)	アドレス	4	DEBPDCB(0)	
40(28)	ビット・ストリング	1	DEBRSV07	
41(29)	アドレス	3	DEBPDCBA	
44(2C)	アドレス	4	DEBWDCB(0)	
44(2C)	ビット・ストリング	1	DEBRSV08	
45(2D)	アドレス	3	DEBWDCBA	
			DEBASDSE	読み取り関連データ・セット用の DCB のアドレス
基本セクションの後に続く直接アクセス・ストレージ・デバイス・セクション。エクステントごとにこれらのセクションのいずれかがあるが、PDSE または拡張フォーマット・データ・セットの場合、装置ごとに 1 セクションあります。				
0(0)	アドレス	4	DEBUCBAD(0)	読み取り関連データ・セット用の DCB のアドレス
0(0)	ビット・ストリング	1	DEBDVMOD	パンチ関連データ・セット用の DCB のアドレス
1(1)	アドレス	3	DEBUCBA	システム使用
I 4(4)	バイナリー	1		パンチ関連データ・セット用の DCB のアドレス
I 5(5)	バイナリー	1	DEBNMTRKHI	印刷関連データ・セット用の DCB のアドレス
6(6)	文字	2	DEBSTRCC	システム使用
8(8)	文字	2	DEBSTRHH	印刷関連データ・セットの DCB のアドレス
10(A)	文字	2	DEBENDCC	"*" 装置関連 DS SEC の終わり
12(C)	文字	2	DEBENDHH	エクステント終了の読み取り/書き込みトラック・アドレス。PDSE の場合、このフィールドは予約済みです。拡張フォーマット・データ・セットの場合、このフィールドは format-1 DSCB アドレスのトラック番号を含んでいます。
14(E)	文字	2	DEBNMTRK	所定のエクステントに割り振られたトラックの数。PDSE の場合、このフィールドは、1 (X'0001') に設定されます。拡張フォーマット順次データ・セットの場合、1 番目のバイトに format-1 DSCB アドレスのレコード番号、2 番目のバイトにゼロが入ります。
			DEBDASDE	"*" DASD 装置セクションの終わり
EXCP アクセス方式、BSAM、および QSAM 依存セクション。長さは DEBAMLNG に入ります。				
0(0)	文字	2	DEBVOLSQ(0)	マルチボリューム順次データ・セットのボリューム・シーケンス番号。
0(0)	ビット・ストリング	1	DEBVOLBT	debvolsq の最初のバイト。システム用に予約済みです。

表 90. DEB フィールドの部分リスト (続き)

オフセット	タイプ	長さ	名前	説明
1(1)	SIGNED	1	DEBVLSEQ	直接アクセスの場合は、データ・セットの最初のボリュームに関連したデータ・セットのボリュームのシーケンス番号。テープの場合は、処理される最初のデータ・セットのボリュームに関連したボリュームのシーケンス番号。
2(2)	文字	2	DEBVOLNM	マルチボリューム順次データ・セットのボリューム総数。
4(4)	文字	8	DEBDSNM(0)	メンバー名。このフィールドは、メンバー名で出力データ・セットが開かれ、DSCB により区分データ・セットが指定されている場合にのみ表示されます。
4(4)	アドレス	4	DEBUTSAA(0)	ユーザー合計の保管域のアドレス
4(4)	ビット・ストリング	1	DEBRVS13	システム使用
5(5)	アドレス	3	DEBUTSAB	ユーザー合計の保管域のアドレス
8(8)	ビット・ストリング	4	DEBRVS14	システム使用 (ユーザー合計が指定された場合)
12(C)	SIGNED	2	DEBBLKSI	最大ブロック・サイズ
14(E)	SIGNED	2	DEBLRECL	論理レコード長
BPAM 依存セクション				
0(0)	文字	1	DEBEXTNM(0)	入力用に区分データ・セットが開かれた場合は、それぞれの 1 バイト・フィールドに、各データ・セット (先頭は除く) の最初のエクステント項目のエクステント番号が入ります。これは、2 つ以上のデータ・セットが連結されている場合に適用されます。フィールドのバイト数は、連結されたデータ・セット数以下となります。
0(0)	文字	8	DEBDSNAM	メンバー名の出力用に区分データ・セットが開かれる場合、このフィールドはメンバー名になります。
相対ブロック・アドレスのオプションのある固定長レコードの BDAM 依存セクション。				
0(0)	SIGNED	4	DEBDBLK(0)	装置依存セクションに記述されているエクステント当たり 1 つ 4 バイトのフィールド。
0(0)	アドレス	1	DEBDBPT	トラック当たりのブロック数
1(1)	文字	3	DEBDBPE	エクステント当たりのブロック数
サブルーチン名セクション注釈がアクセス方式依存セクションの後に続きます。アクセス方式セクションがない場合は、装置依存セクションの後になります。				
0(0)	文字	2	DEBSUBID	サブルーチン識別番号。それぞれのアクセス方式サブルーチン、付加サブルーチン、および IRB ルーチンは、8 バイトの固有の名前を持っています。各ルーチン名の下位の 2 バイトは、サブルーチンが OPEN ルーチンによりロードされた場合、このフィールドに入ります。
DEB 接頭部の DEBXTNP によってポイントされる DEB 拡張。				
0(0)	SIGNED	2	DEBXLNGH	DEB 拡張の長さ
2(2)	ビット・ストリング	1	DEBXFLG1	フラグ・バイト-システム使用
3(3)	ビット・ストリング	1	DEBXFLG2	フラグ・バイト
	.1.		DEBBYP	X'40'- オンの場合、EXCP スキャン・ルーチンは IOSBYP をオンに設定して、接頭部の作成をバイパスします。

制御ブロック

表 90. DEB フィールドの部分リスト (続き)

オフセット	タイプ	長さ	名前	説明
	. .1.		DEBCHCMP	X'20' オンの場合、EXCP スキャン・ルーチンは IOSCHCMP をオンに設定して、接頭部の作成をバイパスします。
4(4)	アドレス	4	DEBXDSAB	DSAB へのポインター
8(8)	ビット・ストリング	4		システム使用
12(C)	アドレス	4	DEBXDBPR	DEB へのポインター
16(10)	文字	24		システム使用
40(28)	ビット・ストリング	8	DEBXDEF(0)	エクステント・データ域を定義します。
40(28)	ビット・ストリング	1	DEBDEFG1	フラグ・バイト
	1.		DEBNSHED	X'80' シーク・ヘッドは許可されません。
	.1.		DEBXVDEF	X'40' DEB 拡張データ定義パラメーターが有効です。DX データを使用する場合は、オンにしてください。
41(29)	ビット・ストリング	1	DEBGATTR	グローバル属性
	xx		DEBECKD	X'C0' エクステント定義 2。B'00.....' エクステント操作を固定ブロック・アーキテクチャーの定義どおりに定義します。B'11.....' エクステント操作を CKD EXT-CCHH エクステントの定義どおりに定義します。
	1.		DEBGAEX1	X'80' エクステント定義 1
	.1.		DEBGAEX2	X'40' エクステント定義 2
	. .1.		DEBSTRTP	X'20' データ・ストレージ・タイプ - 通常/一時。 B'..1.....' 通常データ・ストレージ。 B'..0.....' 一時データ・ストレージ。
	. . .x xx . .		DEBGA345	X'1c' 属性ビット 3,4,5 B'...000..' 通常キャッシュ・アクセス。 B'...001..' キャッシュ・ロード・バイパス。 B'...010..' キャッシュ・ロード禁止。 B'...011..' 順次アクセス。 B'...100..' 予約済み。 B'...101..' 予約済み。 B'...111..' 予約済み。 B'.....XX' 予約済み。
	. . .1		DEBGA1	X'10' 属性 1
 1		DEBGA2	X'08' 属性 2
1 . .		DEBGA3	X'04' 属性 3 B'...000..' 通常キャッシュ・アクセス。 B'...001..' キャッシュ・ロード・バイパス。 B'...010..' キャッシュ・ロード禁止。 B'...011..' 順次アクセス。 B'...100..' 順次ステージング・モード。 '...101..' レコード・アクセス・モード。 B'...111..' 予約済み。 B'.....XX' 予約済み。
	. . . 0 00 . .		DEBNCACH	X'00' 通常キャッシュ・アクセス
	. . . 0 01 . .		DEBBCACH	X'04' キャッシュ・ロード・バイパス
	. . . 0 10 . .		DEBICACH	X'08' キャッシュ・ロード禁止
	. . . 0 11 . .		DEBSCACH	X'0C' 順次アクセス
1.		DEBNRD	X'02' 非保持データ・アクセス
1		DEBINHFW	X'01' 高速書き込み禁止 B'.....10' 非保持データ使用。 B'.....11' DASD 高速書き込み禁止。
42(2A)	ビット・ストリング	2	DEBBLKSZ	バイト単位のブロック・サイズ (32760 以下の場合)
44(2C)	文字	4	DEBEXTOK	ゼロ、エクステント・ロケーターまたはトークン

表 90. DEB フィールドの部分リスト (続き)

オフセット	タイプ	長さ	名前	説明
44(2C)	文字	2	DEBNRDID	サブシステム機能 ID: 非保持データ ID または並行コピー/XRC セッション ID。
46(2E)	文字	2		
48(30)	文字	4	DEBIOPID	システム使用
52(34)	文字	4	DEBBLKID	拡張ブロック数検査で、このテープ・ボリューム内のブロック数を計算するために使用されるブロック ID の値。

データ機能域 (DFA) フィールド

DFADFVAD 以外、DFA 内のすべてのフィールドは、対象とするプログラミング・インターフェースの一部です。

共通名:	データ機能域
マクロ ID:	IHADFA
DSECT 名:	DFA
目印 ID:	DFAACRON
サブプールおよびキー:	中核常駐およびキー 0; 16 MB より下に常駐。
サイズ:	112 バイト
作成者:	中核にアセンブルする。
指示:	CVT の CVTDFA フィールド。
逐次化:	該当せず
機能:	DFSMS に適用される情報を含むデータ機能域をマップします。

表 91. DFA フィールド

オフセット	タイプ/値	長さ	名前	説明
0(0)	UNSIGNED	2	DFALEN	このテーブルの長さ
				DFP のバージョン、リリース、および修正レベル情報。 DFAREL の最初の 3 桁の数字は DFP の最終レベルを表します (X'332)。 DFAREL の非ゼロの 4 桁目の数字は、この製品のレベルは指示されたレベルより高いことを示します。
2(2)	ビット・ストリング	2	DFAREL	4 桁の数字 = バージョン、リリース、修正、X

以下は機能バイトです。ビットがオンの場合、現行リリースが機能をサポートしているか、またはシステムのこの特定インスタンスが機能をサポートしていることを意味します。ソフトウェアは機能をサポートしているが機能を使用できない場合には (おそらく、相互に必要なソフトウェアまたはハードウェアが利用不可であるために)、機能ビットはオフになります。

4(4)	SIGNED	4	DFAFEATS(0)	すべての機能バイト - CS 命令によって使用される。
4(4)	ビット・ストリング	1	DFAFEATI	機能バイト 1
	1...		DFAXA	"X'80'" MVS/XA (CVT 内の CVTMVSE のコピー)
	..1.		DFALSR	"X'40'" 複数の VSAM LSR プールがサポートされます (1.1.0)
	...1.		DFAEOS	"X'20'" スクラッチ時の DASDM 消去がサポートされます (2.1.0)

制御ブロック

表 91. DFA フィールド (続き)

オフセット	タイプ/値	長さ	名前	説明
	...1 ...		DFAXRF	"X'10'" 拡張回復機能 (2.1.0)
	... 1...		DFAEXPCI	"X'08'" 制御インターバルによるエクスポート (2.1.0)
1..		DFAEOSIC	"X'04'" ICF のスクラッチ時に消去 (2.1.0)
1.		DFASMS	"X'02'" システム管理ストレージ (3.1.0)
1		DFAPDSE	"X'01'" PDSE サポートは、DFP レベルが 3.2.0 以上で、該当するレベルの SP が存在する場合、IEAVNP26 によって設定されたシステム上で使用可能です。
5(5)1 ビット・ストリング	1	DFAIPDS DFAFEAT2	"DFAPDSE" IPDS は PDSE の旧名です。 機能バイト 2
	1...		DFADLS	"X'80'" 予約済み
	.1..		DFAPML	"X'40'" 予約済み
	..1.		DFAFMS	"X'20'" ファイル管理サービスがサポートされます。
	...1 ...		DFACMPAC	"X'10'" インストール時の圧縮のデフォルト。
	... 1...		DFABPBLD	"X'08'" チャンネル・プログラム接頭部作成をバイパスします。
1..		DFASSF	"X'04'" SSF サービスが使用可能です。
1.		DFAMMEXT	"X'02'" MMS は XTIOF をサポートします。
1		DFAINDEF	"X'01'" 圧縮のデフォルトはインストール時に明示的に設定します。
6(6)	ビット・ストリング	1	DFAFEAT3	機能バイト 3
	1...		DFAVOLSN	"X'80'" センス情報から抽出された VOLSER がインストール時に受け入れられます。
	.1..		DFASAMEX	拡張フォーマット順次データ・セットがサポートされます。
	.1..		DFASMSEX	DFASAMEX の別名
	..1.		DFAKSDEX	拡張フォーマット KSDS がサポートされます。
	...1 ...		DFACMPCT	DFSMS アクセス方式圧縮がサポートされます。ビットは、SMS サブシステムの初期化によって設定されます。
	... 1...		DFARLSJ3	このシステム上の SMSVSAM サーバーが正常に初期化されました。このビットは SMS スケジューリングによって使用されます。オンになった後は、このビットは IPL の存続期間中、オンのままです。このビットは、SMSVSAM サーバーが現在操作可能であることを示すものではありません。
1..		DFARECAL	ARCHRCAL マクロを介したデータ・セット再呼び出し機能が使用可能です。
1.		DFADEEXT	DESERV 出口機能が使用可能です。
1		DFADLL	DFSMS DLL サポートが使用可能です。
7(7)	ビット・ストリング	1	DFAFEAT4	機能バイト 4
	1...		DFAFDAT	予約済み
	.1..		DFANSRV	IGGSSRV マクロを使用して、DFP NIP サービスを呼び出すことができます。
	..1.		DFADYNL	動的リンク・リストがサポートされます。
	..1.		DFACIR2	カタログ情報ルーチン IKJEHCIR は、フォーマット 2 作業域 (つまり、フルワードの長さフィールド) をサポートします。
	... 1...		DFADYLPA	動的 LPA に対する DFSMS サポートが使用可能です。
1..		DFAFORK	DFSMS ロダー FORK 出口が存在します。

表 9I. DFA フィールド (続き)

オフセット	タイプ/値	長さ	名前	説明
1.		DFASNBK	インストールされた RAMAC VIRTUAL ARRAY (RVA) のスナップショット機能を使用した「高速」バックアップを提供するソフトウェア・サポート。
1		DFASNAP	RAMAC VIRTUAL ARAY (RVA) のスナップショット機能に対する API サポートがインストールされています。
8(8)	文字	4	DFAACRON	この制御ブロックの頭字語。
12(C)	ビット・ストリング	1	DFAFEAT5	機能バイト 5
	1...		DFAUPDSE	このシステムには、非管理 PDSE サポートがインストールされています。
	.1..		DFABTSREQ	1 は、PARMLIB の IGDSMS _{xx} メンバー内の BLOCKTOKENSIZE=REQUIRE を意味します。ラージ・フォーマット・データ・セットを開く場合の制限事項です。0 は BLOCKTOKENSIZE=NOREQUIRE を意味します。
	..1.		DFABLDLS	BLDL START= および STOP= パラメーターがサポートされます。
	..xx xx..		DFAALVER	予約済み。
1.		DFAALFOR	新規テープ・ラベルを書き込むバージョン・レベル。 オフ: ANSI ラベル・バージョン 3。 オン: ANSI ラベル・バージョン 4。
1		DFAALFOR	ISO/ANSI バージョン・レベルのインストールが強制かどうかを決めます。
13(D)	ビット・ストリング	1	DFAFEAT6	機能バイト 6
	1111		DFACPSDB	PARMLIB 内の DEVSUP _{xx} の COPYSDB 値。 IEBGENER および他のコピー・プログラムの SDB オプションのシステム・レベル・デフォルト。 DFACPS** の下の VAE 定数。
	0001		DFACPSNO	COPYSDB = NO
	0010		DFACPSYE	COPYSDB = YES
	0010		DFACPSSM	COPYSDB = SMALL (COPYSDB = YES と同じ)
	0011		DFACPSIN	COPYSDB = INPUT
	0100		DFACPSLA	COPYSDB = LARGE
14(E)	ビット・ストリング	1	DFALPKG	以前はライセンス・パッケージ・バイト。現在は予約済み。
15(F)	ビット・ストリング	1	DFALPKG2	以前は 2 番目のライセンス。現在は予約済み。

バージョン、リリース、および修正レベル情報。バイト 0 は、製品コードを定義し、バイト 1-3 は、この製品のバージョン、リリース、および修正レベルを定義します。製品バイト (DFAPROD) の X'00' は、製品として DFP を示し、DFARELS のバイト 1-3 も X'00' になります。この場合には、DFAREL が DFP 製品のリリース・レベルを検査することを選択できます。DFAPROD が X'00' に等しくない場合、DFAREL は出荷された DFP 製品の最後のレベルに凍結されるので、これを検査してはなりません。製品バイト (DFAPROD) の X'01' は、製品として DFSMS を示し、DFARELS のバイト 1-3 は、DFSMS 製品のバージョン、リリース、および修正レベルを示します。製品バイト (DFAPROD) の X'02' は、OS/390 を示します。この値は、このレベルの DFSMS は OS/390 専用であることを示しません。DFSMS は OS/390 の各リリースで更新されない場合があるため、バージョン、リリース、および修正レベル・フィールドは、このレベルの DFSMS が最初に出荷された OS/390 のレベルを示します。バージョン、リリース、および修正レベル・フィールドは、2 進値です (例えば、10 は X'0A' となります)。

16(10)	ビット・ストリング	4	DFARELS	4 バイト = 製品、バージョン、リリース、修正
--------	-----------	---	---------	--------------------------

制御ブロック

表 91. DFA フィールド (続き)

オフセット	タイプ/値	長さ	名前	説明
16(10)	ビット・ストリング 0000 0000 0000 0001 0000 0002 0000 0003	1	DFAPROD DFADFP DFADFSMS DFAOS390 DFAZOS	製品バイト DFP 製品コード。 DFSMS の製品コード。 OS/390 バージョン 2 の製品コード。 z/OS の製品コード。
17(11)	ビット・ストリング	1	DFAVER	バージョン・バイト
18(12)	ビット・ストリング	1	DFARLSE	リリース・バイト
19(13)	ビット・ストリング	1	DFAMOD	修正バイト
20(14)	SIGNED	2	DFAMSMDE	8 バイトの別名を持つこのリリースの SMDE の最大長。
22(16)	UNSIGNED 1...1..	1	DFAVERBO * DFATVS	フラグ 予約済み。 オンの場合、トランザクション VSAM がインストールされています。
23(17)	UNSIGNED	1	*	予約済み。
24(18)	アドレス	4	DFACSSVT	呼び出し可能システム・サービス・ベクトル・テーブル・アドレス。
28(1C)	アドレス	4	DFADCVSO	データ変換サービス - オープン。
32(20)	アドレス	4	DFADCVSD	データ変換サービス - 変換。
36(24)	アドレス	4	DFADCVSC	データ変換サービス - クローズ。
40(28)	アドレス	4	DFAELNMP	DFSMS エlement名のアドレス。名前は DSECT DFAELNM によってマップされます。表 92 を参照。z/OS 1.3 以降でのみ有効です。
44(2C)	アドレス	4	DFADFVAD	データ機能ベクトル・テーブル・アドレス。
MVS/XA DFP バージョン 2 リリース 1 修正レベル 0 で最初に出荷された時点での DFA の終わり。このコメントより先のフィールドを参照する前に、ユーザーは DFARELS が X'02020A00' 以上であるか、または DFALEN が十分に大きいこと (DFALEN を参照) を確認する必要があります。				
48(30)	X'INTEGER'	8	DFABLKSZ	システムが決めるブロック・サイズの限度。デフォルトは 32760。DEVSUPxx PARMLIB メンバーから取得します。
56(38)	文字	56	*	RESERVED

表 92. DFA エlement名

オフセット	タイプ/値	長さ	名前	説明
0(0)	DSECT		DFAELNM	DSECT 名
0(0)	UNSIGNED	2	DFAELNML	次のフィールド内の有効な文字の長さ。現行では 10 です。
2(2)	文字	22	DFAEXTXT	EBCDIC の DFSMS エlement名。小文字を含めることができます。現行では、「z/OS DFSMS」です。

付録 C. アクセシビリティ

アクセシビリティ機能は、運動障害または視覚障害など身体に障害を持つユーザーがソフトウェア・プロダクトを快適に使用できるようにサポートします。z/OS のアクセシビリティの主要機能により、ユーザーは以下のことができるようになります。

- 画面読み上げ機能および画面拡大機能などの支援機能の使用
- キーボードのみを使用して、特定の機能または画面を使用したのと同等の機能を操作
- 色、コントラスト、フォント・サイズなど表示属性のカスタマイズ

支援機能の使用

画面読み上げ機能などの支援機能は、z/OS のユーザー・インターフェースを使用して機能します。このような製品を使用して z/OS インターフェースにアクセスする場合、その特定情報については支援テクノロジーの資料を参照してください。

ユーザー・インターフェースのキーボード・ナビゲーション

ユーザーは、TSO/E または ISPF を使用して z/OS ユーザー・インターフェースにアクセスできます。TSO/E および ISPF インターフェースへのアクセス方法については、「z/OS TSO/E 入門」、「z/OS TSO/E ユーザーズ・ガイド」、および「z/OS ISPF ユーザーズ・ガイド 第 1 巻」を参照してください。上記の資料には、キーボード・ショートカットまたはファンクション・キー (PF キー) の使用方法を含む TSO/E および ISPF の使用方法が記載されています。それぞれの資料では、PF キーのデフォルトの設定値とそれらの機能の変更方法についても説明しています。

z/OS 情報

z/OS の情報については、スクリーン・リーダーを使用して、以下のインターネット・ライブラリーにある z/OS ブックの BookServer/Library Server 版が利用できます。

www.ibm.com/servers/eserver/zseries/zos/bkserv/

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものであり、本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-0032
東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。 IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Corporation
Mail Station P300
2455 South Road
Poughkeepsie, NY 12601-5400

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

プログラミング・インターフェース情報

本書は、オペレーティング・システムのデータ管理機能を変更または拡張するのを支援することを目的としています。汎用プログラミング・インターフェースを使用して、お客様は DFSMS のサービスを利用するプログラムを書くことができます。

商標

以下は、IBM Corporation の商標です。

CICS	IBMLinks
CopyServices	IPDS
DFSMS	PSF
DFSMSdfp	RACF
DFSMSHsm	RAMAC
DFSMSrmm	RETAIN
ECKD	TotalStorage
Enterprise Storage Server	VTAM
ESCON	z/OS
FICON	
IBM	

- UNIX は、The Open Group の米国およびその他の国における登録商標です。
- 他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

用語集

以下の用語は、DFSMS ライブラリーで使用される用語を定義しています。探している用語が見つからない場合は、下記のサイトにある「IBM Software Glossary」を参照してください。

<http://www.networking.ibm.com/nsg/nsg>

この用語集は、多数の IBM ソフトウェア製品の資料に使用される技術用語を定義した文書で、絶えず変更が加えられています。

[ア行]

- | **アクセス方式サービス・プログラム (access method services)**. VSAM データ・セットと非 VSAM データ・セット、および統合カタログ機能 (ICF) を管理するための多機能サービス・プログラム。アクセス方式サービス・プログラムは、以下の機能を提供する。
- | • VSAM データ・セットおよびカタログを定義し、それらにスペースを割り振る
- | • カatalogのデータ・セット属性を変更する
- | • データ・セットを再編成する
- | • オペレーティング・システム間のデータ可搬性を容易にする
- | • データ・セットのバックアップ・コピーを作成する
- | • アクセス不能データ・セットをアクセス可能にするための補助機能
- | • データ・セットとカタログのレコードをリストする
- | • 代替索引を定義し、作成する

アクセス方式制御ブロック (ACB) (access method control block (ACB)). アプリケーション・プログラムを VSAM または VTAM プログラムにリンクする制御ブロック。

圧縮 (compress). (1) システムが同一の語句をその語句に関連するもっと短いトークンで置き換えることによって、特定のデータ・セットに必要なストレージの量を少なくすること。(2) 未使用の全スペースをデータ・セットの最後に移動し、メンバーを削除または変更して、区分データ・セット内の未使用および使用不能なスペースをレクラメーション処理すること。

圧縮フォーマット (compressed format). データ・クラス (COMPACTION) パラメーターで指定される拡張フォーマット・データ・セットの特定のタイプ。 VSAM

は、圧縮フォーマット・データ・セットの中の個々のレコードを圧縮することができる。 SAM は、圧縮フォーマット・データ・セットの個々のブロックを圧縮することができる。圧縮 (compress) を参照。

オープン時のバックアップ (BWO) (Backup-While-Open (BWO)). この機能は、データ・セットが更新のためにオープンされている間にデータ・セットのバックアップ・コピーを作成する。そのバックアップ・コピーには部分更新が含まれる。

[カ行]

拡張区分データ・セット (PDSE) (partitioned data set extended (PDSE)). 区分データ・セットのディレクトリーとメンバーによく似た、索引付きディレクトリーとメンバーが入っているシステム管理データ・セット。PDSE は区分データ・セットの代わりに使用することができる。

拡張フォーマット (extended format). EXTENDED のデータ・セット名タイプ (DSNTYPE) のデータ・セットのフォーマット。データ・セットの論理的な構成は拡張フォーマットでないデータ・セットと同じだが、物理的なフォーマットは異なる。ストライプ・データ・セット (striped data set) および圧縮フォーマット (compressed-format) も参照。

管理クラス (management class). ストレージ管理者によって定義された管理属性の集合体。割り振られているが未使用のスペースの解放の制御、データ・セットの保存/マイグレーション/バックアップの制御、集合グループの保存とバックアップの制御、およびオブジェクトの保存/バックアップ/クラス遷移の制御に使用される。

基本カタログ構造 (BCS) (basic catalog structure (BCS)). 統合カタログ機能環境におけるカタログ構造名。統合カタログ機能カタログ (integrated catalog facility catalog) も参照。

区分データ・セット (partitioned data set). それぞれにプログラム、プログラムの一部、またはデータを入れることができるメンバーと呼ばれる区画に分割された直接アクセス・ストレージのデータ・セット。

[サ行]

索引付き VTOC (indexed VTOC). データ・セット名およびフリー・スペース情報のリストを含む索引付きのボリューム目録。これによってデータ・セット検索の効率が上がる。

システム・データ (system data). z/OS またはそのサブシステムで初期化および制御のために必要なデータ・セット。

使用属性 (use attribute). (1) 新しいデータ・セットを割り振るためにボリュームを使用できる状況を制御する、DASD ボリュームに割り当てられる属性。使用属性には、*public*、*private*、*storage* がある。(2) システム管理テープ・ボリュームの場合、使用属性には *scratch* と *private* がある。

ストライピング (striping). パフォーマンスを上げるために複数のボリュームにデータ・セットを分散させるディスク・アレイの、ソフトウェアによるインプリメンテーション。

ストライプ (stripe). DFSMS において、ストライプ・データ・セットのうち、1 つのボリューム上にある部分。その部分のレコードは、必ずしも論理的に連続していない。パフォーマンスを上げるために複数のボリュームを同時に読み書きできるよう、システムはいくつかのストライプにレコードを分散させる。データ・セットがストライプされているかどうかは、アプリケーション・プログラムにはわからない。

ストライプ・データ・セット (striped data set).

DFSMS において、複数のストライプから構成される拡張フォーマット・データ・セット。SMS は、そのストレージ・クラスの SUSTAINED DATA RATE の値に基づいて、使用するストライプの数を決定する。ストライプ・データ・セットには、順次データ・ストライピング・アクセス技法を活用する。ストライプ (*stripe*) およびストライピング (*striping*) を参照。

ストレージ管理 (storage control). プロセッサ・チャンネルとストレージ・デバイスの間の対話を操作したり、チャンネル・コマンドを実行したり、ストレージ・デバイスを制御したりするストレージ・サブシステム内のコンポーネント。

ストレージ管理 (storage management). データ・セットの割り振り、配置、モニター、マイグレーション、バックアップ、リコール、リカバリー、および削除のアクティビティ。これらは、手動あるいは自動化プロセスのどちらを使用しても行える。これらのプロセスは、ストレージ管理サブシステムによって、ストレージ・リソ

ースの最適化の際に自動化される。ストレージ管理サブシステム (*Storage Management Subsystem*) も参照。

ストレージ管理サブシステム (SMS) (Storage Management Subsystem (SMS)). ストレージの管理を自動化し、集中化するために使用される DFSMS の機能。SMS を使用することによってストレージ管理者は、データ割り振り特性、パフォーマンスおよび可用性の目標、バックアップおよび保存要件、およびストレージ所要量をデータ・クラス、ストレージ・クラス、管理クラス、ストレージ・グループ、および ACS ルーチン定義を通じてシステムに記述する。

ストレージ・クラス (storage class). パフォーマンス目標と可用性要件を示すストレージ属性の集合。ストレージ管理者により定義され、このような目標と要件を満たすことができる装置を選択するのに使用される。

| **セキュリティー・サーバー (Security Server).** z/OS の
| ライセンス機能。リソース・アクセス管理機能
| (RACF)、DCE セキュリティー・サーバー、z/OS LDAP
| サーバー、z/OS ファイアウォール・テクノロジー、オ
| ープン暗号化拡張プラグイン (OCEP)、ネットワーク認
| 証サービスを含む。

装置サポート機能 (ICKDSF) (Device Support Facilities (ICKDSF)). DASD ボリュームの初期化およびトラック・リカバリーに使用されるプログラム。

[タ行]

データ・クラス (data class). ストレージ管理者により定義され、データ・セットの作成に使用される、割り振りおよびスペース属性の集合。

データ・セット制御ブロック (DSCB) (data set control block (DSCB)). 直接アクセス・ストレージ内のデータ・セットのデータ・セット・ラベル。

テープ・サブシステム (tape subsystem). 磁気テープ・サブシステムはコントローラーおよび装置から構成され、ユーザー・データをテープ・カートリッジに保管することができる。テープ・サブシステムの例として、IBM 3490 および 3490E 磁気テープ・サブシステムが含まれる。

テープ・ボリューム (tape volume). データ・セットまたはオブジェクトと使用可能なフリー・スペースを含み、ボリューム・ラベルによって識別される、テープ上のストレージ・スペース。テープ・ボリュームは、単一のテープ・カートリッジまたはリール上の記録スペース。ボリューム (*volume*) も参照。

統合カタログ機能カタログ (integrated catalog facility catalog). 基本カタログ構造 (BCS) とそれに関連するボリューム目録 (VTOC) および VSAM ボリューム・データ・セット (VVDS) から構成されるカタログ。基本カタログ構造 (basic catalog structure) および VSAM ボリューム・データ・セット (VSAM volume data set) も参照。

[ハ行]

| **バインダー (binder).** 言語翻訳プログラムおよびコンパイラーの出力を処理して実行可能プログラム (ロード・モジュールまたはプログラム・オブジェクト) にするプログラム。リンケージ・エディターおよびバッチ・ローダーに相当する。

光ディスク・ボリューム (optical volume). ボリューム・ラベルにより識別される、光ディスク上のストレージ・スペース。ボリューム (volume) も参照。

| **プログラム・オブジェクト (program object).** 仮想記憶域にロードして実行するのに適した形式のコンピューター・プログラムのすべてまたは一部。プログラム・オブジェクトは PDSE プログラム・ライブラリーまたは z/OS UNIX ファイルに保管され、ロード・モジュールよりも制約が少ない。プログラム・オブジェクトはバインダーによって作成される。

保管場所 (storage location). 災害時リカバリー、バックアップ、および重要レコード管理のためにボリュームを保管する、取り外し可能メディア・ライブラリーから物理的に離れた場所。

ボリューム (volume). DASD、テープ、または光ディスク装置上のストレージ・スペース。このストレージ・スペースは、ボリューム・ラベルで示される。DASD ボリューム (DASD volume)、光ディスク・ボリューム (optical volume)、およびテープ・ボリューム (tape volume) も参照。

ボリューム状況 (volume status). ストレージ管理サブシステムにおいて、システム管理対象のボリュームが完全に使用可能であるかどうかを示す。

- 「初期 (initial)」は、ボリュームにシステム管理に不適格なデータ・セットが含まれているために、そのボリュームはシステム管理用に準備ができていないことを示す。
- 「変換 (converted)」は、ボリューム上のすべてのデータ・セットが、関連するストレージ・クラスを持ち、統合カタログ機能カタログにカタログされていることを示す。

- 「非システム管理 (non-system-managed)」は、ボリュームにシステム管理データ・セットが含まれておらず、システム管理として初期化されていないことを示す。

ボリューム目録 (VTOC) (volume table of contents (VTOC)). (1) ボリュームの各データ・セットを記述している、直接アクセス・ボリュームのテーブル。(2) ディスクまたはディスクセット上の各ファイルとライブラリーの場所、サイズ、およびその他の特性を記述しているディスクまたはディスクセット上の領域。

[ラ行]

| **リソース・アクセス管理機能 (RACF) (Resource Access Control Facility (RACF)).** IBM ライセンス・プログラム。これは、システムに対してユーザーを識別および検査したり、保護リソースへのアクセスを許可したり、無許可でシステムに入る試みを検出してログに記録したり、保護リソースへのアクセスを検出してログ記録することにより、アクセス制御を提供する。RACF は、セキュリティ・サーバーに組み込まれている。セキュリティ・サーバー (Security Server) も参照。

D

DASD ボリューム (DASD volume). 共通ラベルにより識別され、一連の関連アドレスによりアクセスされる DASD スペース。ボリューム (volume) も参照。

| **DFSMS.** ストレージ管理、データ管理、および装置管理機能を提供する z/OS の部分。DFSMS は、DFSMSdftp、DFSMSdss、DFSMSshsm、DFSMSrmm、DFSMSstvs から構成される。

DFSMS ネットワーク・ファイル・システム (DFSMS Network File System). z/OS ネットワーク・ファイル・システム (z/OS Network File System) を参照。

DFSMSshsm. データのバックアップとリカバリー、およびストレージ階層でのボリュームのスペース管理に使用される、z/OS の DFSMS 機能コンポーネントまたは基本エレメント。

U

UTC. 協定世界時 (Coordinated Universal Time)。

V

VSAM ボリューム・データ・セット (VVDS) (VSAM volume data set (VVDS)). 指定された DASD ボリューム上にあるシステム管理データ・セットおよび VSAM の特性を記述するデータ・セット。統合カタログ機能カタログの一部。基本カタログ構造 (*basic catalog structure*) および 統合カタログ機能カタログ (*integrated catalog facility catalog*) も参照。

Z

z/OS UNIX システム・サービス (z/OS UNIX) (z/OS UNIX System Services (z/OS UNIX)). z/OS オペレーティング・システムのシェルとユーティリティー、カーネル、デバッガー、ファイル・システム、C/C++ ランタイム・ライブラリー、言語環境プログラム、その他のエレメントによって提供される一連の機能。ユーザーは、それらを使用することによって、UNIX の規格に準拠したアプリケーション・プログラムを作成および実行できる。

z/OS ネットワーク・ファイル・システム (z/OS Network File System). ワークステーション、パーソナル・コンピューター、またはネットワーク・ファイル・システム・プロトコル用のクライアント・ソフトウェアを使用している TCP/IP ネットワーク上の他のすべてのシステムから、z/OS ホスト・プロセッサ・データにリモート・アクセスできる、z/OS の基本エレメント。

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

アクセシビリティ 439
アドレス・スペース
 V=R, EXCP の使用 100
 V=V, EXCP の使用 100
イベント制御ブロック xi
イメージ
 検査 330, 343
 テーブル
 オブジェクト・モジュール 339
 項目の追加または変更 339
 長さ 330
 別名 330
 ライブラリー
 プリンター情報 328
 保守 327, 349
 FCB イメージの追加 346
 FCB イメージの変更 348
 IMAGELIB マクロ 348
 ID 346
イメージ・テーブル
 ヒント 338
インターフェース制御チェック 148
上書き, データの 67
エクステント
 終了, 付加ルーチン 118
 使用可能な 17
 制御情報 99
 トラッキング 4
 フリー・スペース 13
 割り振り 37
 VSAM データ・スペース 9
エクステント範囲外 (out-of-extent) エラー 148
エクステント領域
 ICVEDT02 マッピング・マクロ 49
エラー
 入出力リカバリー手順 (EXCP) 103
 ルーチン, 無視 118
 CVAF VTOC 索引 425
 VTOC または索引
 システムまたはユーザー・エラーからのリカバリー 62
 処理 62

オーバーヘッド, システム 120

[カ行]

カートリッジ・テープ
 高速位置決め 221
回転位置感知機構 188
カウント情報
 プログラミング・インターフェース 369
拡張データ制御ブロック (DCBE)
 EXCP で使用 125
拡張フォーマット・データ・セット
 PASSWORD データ・セットとして使用できない 168
仮想
 IDAW (間接アドレッシング・ワード) 97
カタログ
 管理
 情報検索 82
 マクロ機能 81
 項目
 非 VSAM データ・セット 88
 読み取り 82
 情報検索
 データ・セット名 82
 別名 86
 GDS 名 84
 非 VSAM データ・セット
 マクロ仕様 88
 戻りコード 93
 例 90
 保守 83
 戻りコード 94
間接
 アドレッシング・ワード (IDAW) 97
 データ・アドレス・リスト (IDAL) 143
管理クラス
 名前 266
完了コード xi
 EXCP 入出力 136
 XDAP 入出力 163
関連要求 104, 129
キーボード 439
機密データ, 消去 68
共用属性, DFSMSdfp の検索 276
許可
 RACF, DASDVOL 65, 66, 69

切り替え, ボリュームの, パスワード保護 170
区分データ・セット
 スペース
 解放 29, 37
グローバル出口 (DESERV) 285
検索, データ・セットの属性の 266
検索, DFSMSdfp 共用属性の 276
高速カートリッジ・テープの位置決め 221
構文図, 読み方 xiv
誤長標識 148
コマンド再試行 101
コミュニケーション・ベクトル・テーブル (CVT) 150, 155

[サ行]

索引付き VTOC xi
構造 16
システム
 エラー 62
 サポート 424
初期化 65
内容 10
パスワード保護 66
バッファー
 処理 383
変換 65
変更 379
保護 65
ボリューム
 更新 66
 復元 66
ボリュームの更新 66
無許可ユーザー 62
リスト表示 63
レコード 379
DSCB
 アクセス 53
 書き込み 379
 読み取り 379
索引なし VTOC
 構造 16
 使用可能なスペース 12
サブシステム状況
 プログラミング・インターフェース 369
サブシステム統計
 プログラミング・インターフェース 369

システム
 属性呼び出し 263
 マクロ命令 179
 システム常駐ボリューム 168
 システム・レベルの判別 263
 実トラック・アドレスから相対トラック・
 アドレス変換ルーチン 152
 指標付け機能、3211 の 347, 350
 順次データ・セット
 スペース、解放 29
 ショートカット・キー 439
 消去、機密データの 67
 状況コード
 RENAME マクロ 79
 SCRATCH マクロ 73
 身体障害 439
 推奨
 CVAF 412
 CVSTAT 405
 UCB パラメーター 382, 413, 425
 スクラッチ、データ・セットの 67, 74
 ストレージ・クラス。名前 266
 スペース割り振り
 未使用の解放 29
 スペース・マップ 16
 制御パスワード 171
 制御ブロック
 データ機能域 (DFA) 435
 ページ入出力復元リスト 228
 DCBE (拡張データ制御ブロック) 125
 DEB-EXCP(VR) 427
 DFA 435
 EXCP 125
 DCB 99, 113, 125
 DCBE 99
 DEB 99, 138, 427
 ECB 98, 136
 IEDB 99, 134
 IOB 98, 127, 135
 IOBE 99, 131
 EXCP で使用されるフィールド 125
 fields-EXCP(VR) 427
 XDAP 162
 制約事項
 部分修飾データ・セット名 82
 CVAFFLIT 401
 MODE 34
 MSG 20
 OBTAIN 17
 UCATDX 91
 UCB 34
 VERIFY 382
 セクター
 アドレス
 XDAP マクロ 161
 (RPS 装置) 154, 163

セクター (続き)
 番号 (RPS 装置) 154, 163
 変換ルーチン 154, 163
 世代
 名前
 絶対 84
 相対 84
 番号 84
 相対
 実トラック・アドレス変換ルーチン
 150
 世代番号 84
 装置
 依存パラメーター 121
 終了エラー・リカバリー手順 103
 制御ブロック xi
 チェック 148
 特性、入出力 184
 特性テーブル項目 (DCTE) 233
 例外 148

[夕行]

代行受信条件 148
 大容量ブロック・インターフェース
 DEVTYPE マクロ 195
 タスク・レベル出口 (DESERV) 284, 285
 妥当性検査、DEB (DEBCHK) の 179,
 184
 チェーニング・チェック 148
 逐次化、CVAF 要求の 44
 チャンネル
 制御チェック 148
 データ・チェック 148
 チャンネル・エラー
 チャンネル・プログラムの終了状況 107
 チャンネル・プログラム
 完了コード 98, 103
 コマンドのチェーニング 98
 シーク・アドレス 131
 処理要件 99
 制御情報 99
 チャンネル・エラー 107
 データ・セット特性 99
 付加ルーチン 138
 変換
 仮想アドレスから中央ストレージ・
 アドレスへの 143
 V=R アドレス・スペース 143,
 144
 V=V アドレス・スペース 101
 実ストレージ 112
 無効な終了状況 107
 問題プログラム
 実行 96
 通信 98

チャンネル・プログラム (続き)
 EXCP
 開始と実行 99, 103
 仮想アドレスから中央ストレージ・
 アドレスへの変換 150
 関連 104
 変更に関する制約事項 100
 XDAP 158, 163
 通知、完了コードの ECB への
 EXCP 入出力 136
 データ
 クラス
 名前 266
 制御ブロック xi
 領域、EXCPVR を使用して固定 112
 データ機能域 (DFA)
 制御ブロック 435
 データ・エクステンツ・ブロック (DEB
 を参照) 427
 データ・セット
 削除
 共用シリンダー 68, 70
 装置間での保管 68
 マクロ命令 67, 74
 VIO 処理済み 68
 スペース、解放 17
 スペース割り振り、解放 29
 制御ブロック xi
 セキュリティ
 アクセス・タイプ 167
 連結 170
 装置、割り当て 121
 属性検索 266
 テープ上の位置変更 117
 定義 2
 名前変更
 使用中の可能性のあるデータ・セッ
 ト 75
 パスワード保護 75
 VTOC 74
 WRITE 保護モード標識 75
 パスワード保護 165, 169
 バックアップ 268
 非 VSAM
 再カタログ 91
 編成 119
 ユーザー・ラベル・エクステンツ 37
 有効期限 68
 割り振り
 移動可能 37
 絶対 37
 DSCB フォーマット 2
 データ・セット終了 (end-of-data-set)
 条件 159
 データ・セットの名前変更
 影響を受けるボリュームの指定 74

データ・セットの名前変更 (続き)
 使用中の可能性のあるデータ・セット
 の名前変更 75
 データ・セット・セキュリティー 75
 名前変更できないデータ・セットと
 UNIX ファイル 75
 マルチボリュームについての考慮事項
 75
 SMS についての考慮事項 74
 テープ
 記録
 手法 124
 密度 124
 縮小エラー・リカバリー 118
 データ・セットの位置変更 117
 テープ・マーク 109
 反射スポット 109
 ブロック数フィールド 131
 ボリューム
 デマウント機能での DEQ 219
 パスワード保護データ・セット
 169
 ボリューム終了 109
 テープ、カートリッジ
 高速位置決め 221
 動的装置再構成 117
 トラック
 固定長レコード 233
 バランス 232, 233
 容量の計算 229, 240
 レコードの削除 234
 TRKCALC 出力 238

[ナ行]

名前
 VTOC 索引 14
 入出力
 効率、改善 112
 入出力開始 (start I/O) 付加ルーチン 142

[ハ行]

ページ入出力復元リスト 228
 ページ・ルーチン 224
 パスワード xi
 カウンターの保守 171
 制御 171
 データ・セット連結 170
 パラメーター・リスト
 ADD レコード 173
 DELETE レコード 175
 LIST レコード 176
 REPLACE レコード 174

パスワード (続き)
 保護
 データ・セット 165, 179
 テープ・データ・セット 169
 ボリュームの切り替え 170
 モード標識 171
 JFCB 変更 210
 VTOC 索引 66
 保護データの名前変更 170
 保護データ・セットの削除 170
 レコード 168
 2 次 171
 バックアップ、データ・セットの 268
 バッファ
 インストール 120
 オペレーティング・システム・オーバ
 ーヘッド 120
 解放 51, 384
 リスト
 解放 51, 384
 機能 47
 項目 48
 作成 47
 フォーマット 49
 ヘッダー・フォーマット 48
 DSCB の読み取り 402
 BUFNO のパフォーマンスの考慮事項
 120
 EXCP でのデータ転送速度 120
 判別、リリース・レベルの 240
 判別、DFSMSdfp リリース・レベルの
 263
 汎用文字セット xi
 非 VSAM
 データ・セット
 アンカタログ 90
 カタログ 88
 再カタログ 91
 ビット
 IOBEEIDA 143
 IOBEIOT 134
 S99ACUCB 107
 S99DSABA 107
 S99TIOEX 107
 UCBRPS 154
 UCBVRDEB 143
 ビットマップ、割り振り DSCB、VIR 16
 ヒント
 イメージ・テーブル 338
 CVAFFILT 404
 ファイル・システム
 作成 351
 ファイル・マーク、書き込み 123
 フィルター基準リスト xi
 フォーマット 13
 フリー VTOC レコード 4

付加ルーチン
 インストール 140
 エントリー・ポイント 139, 140
 使用可能な作業域 140
 説明 138
 プログラミングに関する制約事項 139
 命名規則 141
 戻り 139, 140
 リスト、許可 140, 148
 ABE (異常終了) 148
 CHE (チャネル終了) 149
 EOE (エクステント終了) 147
 PCI (プログラム制御割り込み) 146
 PGFX (ページ固定) 142
 SIO (入出力開始) 141, 143
 SYS1.PARMLIB リスト作成 140
 復元ルーチン 225
 物理ファイル・システム 359
 部分 DSCB 38
 部分修飾データ・セット名
 制約事項 82
 フリー VTOC レコード 4
 フリー・スペース
 再確立 62
 DASD ボリューム 17, 26
 プリンター
 イメージ xi
 イメージ・ライブラリー 328
 制御情報 327
 デフォルト・イメージ
 印刷位置指標付け機能 346
 デフォルトの指定 346
 標準 FCB イメージ 344
 バンド
 国別バンド ID 338
 特別注文 (RPQ) 338
 別名 334
 保管 334
 3262 モデル 5 338
 ページ・レイアウト 347
 文字セット・イメージ 327
 例外 340
 UCS イメージ・テーブル 329
 プログラミング・インターフェース
 カウント情報の 369
 サブシステム状況の 369
 サブシステム統計の 369
 IDCAMS によって提供 369
 プログラム・チェック 148
 ブロック ID、高速カートリッジ・テープ
 位置決め 222
 ページ
 境界、CCW データ・アドレスが超え
 ている 143
 固定
 付加ルーチン 142

ページ (続き)
 固定 (続き)
 リスト処理 142
 フォーマット 347
 ページング例外 142
 別名
 カタログ情報の検索 86
 変換
 仮想チャネル・プログラムから実チャ
 ネル・プログラムへの 112, 143
 実トラック・アドレスから相対トラッ
 ク・アドレスへの
 戻りコード 153
 レジスターの使用法 152
 セクター値、RPS 装置 154, 163
 相対トラック・アドレスの実トラッ
 ク・アドレスへの
 手順 150
 戻りコード 152
 レジスターの使用法 150
 VTOC 65
 保護
 チェック 148
 モード標識 171
 VTOC 索引 65
 ボリューム
 切り替え
 パスワード保護 170
 マルチボリューム・データ・セット
 170, 211
 EOV 中 109
 コピー (VTOC) 66
 索引付き VTOC 66
 索引なし VTOC 66
 スペース割り振り 16
 テープ、保護 169
 テープからの復元 66
 目次 xi
 ラベル 1
 リスト 82
 説明 67
 名前変更 74
 リスト項目
 スクラッチ状況コード 72, 78
 2 次状況コード 72, 78
 DDR を使用したスワッピング 117
 ボリューム、IPL 168
 ボリューム通し番号、CAMLST 内の
 名前 81

[マ行]

マクロ
 データ管理
 CATALOG 88
 LOCATE 82, 87

マクロ (続き)
 MSGDISP 243, 259
 マクロ、データ管理
 CAMLST
 RENAME オペランド 74, 81
 SCRATCH オペランド 67, 74
 SEARCH オペランド 26, 28
 SEEK オペランド 28, 29
 CLOSE
 EXCP 111
 XDAP 159
 CVAF VTOC アクセス 43, 62
 CVAFDIR 50, 379
 CVAFDSM 392
 CVAFFILT 55, 400
 CVAFSEQ 53, 411
 CVAFTST 424
 DCB 125
 EXCP 113
 XDAP 158
 DEBCHK 180, 184
 DEVTYPE 184, 203
 EOVS
 EXCP 109, 111
 XDAP 159
 EXCP 109, 112, 113, 125
 EXCPVIR 112
 LSPACE 17, 26
 OBTAIN 26, 29
 OPEN
 変更された JFCB 223
 EXCP 107, 109
 XDAP 158
 PARTREL 29, 37
 PROTECT 171, 179
 PURGE 224
 RDJFCB 204, 212
 REALLOC 37, 43
 RENAME 74, 81
 RESTORE 229
 SCRATCH 67
 TRKCALC 229, 240
 VTOC アクセス
 DADSM 17, 43
 XDAP 157, 162
 XDAP の使用 158, 159
 マクロ、マッピング
 ICVEDT02 49
 マクロ・システム
 DEVTYPE マクロ 189
 マッピング・マクロ
 IECSDSL1 387
 IHADFA 240
 マルチボリューム・データ・セット
 EXCP を使用した処理 108

メッセージ
 表示 243
 3480/3490 上の表示
 オプション 255
 作動可能 249
 パラメーター 243
 表示のリセット 253
 ボリュームの検査 246
 ボリュームのデマウント 250
 ボリュームのマウント 244
 CVAF VTOC 索引エラー 425
 IEC502E 219
 IGW023A 354
 IGW023I 354
 LSPACE マクロ 24
 MSGDISP マクロ 243
 メッセージ検索ツール、LookAt xiii
 戻りコード
 トラック・アドレス変換ルーチン
 152, 153
 CATALOG マクロ 93
 CVAF VTOC 索引エラー・メッセージ
 425
 CVAFDIR マクロ 386
 CVAFDSM マクロ 400
 CVAFFILT マクロ 405
 CVAFSEQ マクロ 416
 CVAFTST マクロ 425
 DADSM 割り振り 42
 DEBCHK マクロ 184
 DEVTYPE マクロ 201
 IGWABWO 呼び出しステートメント
 277
 IGWARLS 呼び出しステートメント
 280
 IGWASMS 呼び出しステートメント
 277
 IGWASYS 呼び出しステートメント
 277
 IGWLSHR 呼び出しステートメント
 277
 LOCATE マクロ 87
 LSPACE マクロ 24
 MSGDISP マクロ 258
 OBTAIN マクロ
 絶対装置アドレスによる VTOC か
 らの読み取り 29
 データ・セット名による VTOC か
 らの読み取り 28
 PARTREL マクロ 35
 RDJFCB マクロ 209
 REALLOC マクロ 42
 RENAME マクロ 78
 SCRATCH マクロ 72
 TRKCALC マクロ 239

[ヤ行]

有効期限、オーバーライド 68, 70
要求、拡張エラー情報の 105
要求、異なるレベルの ERP 処理の 106
要件
 PROTECT マクロ 173
 RENT 340
 SMS 管理の非 VSAM データ・セット
 69
 TEST=YES 253, 255
 UCB 18
 VTOC 384, 385, 397, 416
用紙制御バッファ・イメージ xi
呼び出し可能サービス
 再入不可プログラム 261
 システム・レベルの判別 263
 説明 261
 データ・セット backup-while-open サ
 ポート 268
 データ・セット属性検索 266
 戻りコードと理由コード 277
 呼び出し 262
 DFSMSdfp 共用属性検索 276

[ラ行]

ラージ・フォーマット・データ・セット
 213-14 ABEND 126
 BLOCKTOKENSIZE パラメーター
 126
 PASSWORD データ・セットとして使
 用できない 168
ライブラリー
 プリンター制御情報 327
 保守 348
 呼び出し可能サービス 261
 FCB イメージ 348
 SYS1.IMAGELIB データ・セット 327
ラベル、管理 109
リスト形式、DEVTYPE マクロ 191
リソース・アクセス管理機能 xi
理由コード
 DEVTYPE マクロ 201
リリース、レベルの判別 240
リリース、DFSMSdfp のレベルの判別
 263
ルート・ファイル・システム 352
例
 DEVTYPE マクロ 202
例外
 プリンター 340
例外条件 118
レコード
 現在場所 122

レジスター
 内容 138
 変換ルーチンの使用法
 相対から実への 150
 実から相対に 153
レベル
 DFSMSdfp または DFSMS 240, 263

[ワ行]

割り込み処理プロシージャ 103
割り振り
 検索域フォーマット 216
 検索リスト・フォーマット 214
 DASD スペース 13

[数字]

1403 プリンター
 UCS イメージ 329
2 次パスワード 171
31 ビット IDAW (間接アドレッシング・
 ワード) 97
3203 モデル 5 プリンター
 JES サポート 329
 UCS イメージ 329
3211 プリンター
 指標付け機能 347, 350
 JES サポート 350
 UCS イメージ 329
3262 モデル 5 プリンター
 印刷バンド 338
 デフォルト FCB イメージ 344
 別名 335
 UCS イメージ・テーブル 335, 338
3480/3490 磁気テープ・サブシステム
 テープ記録手法 124
 メッセージ、表示 243
 MSGDISP マクロ 243
3800 印刷サブシステム
 標準 FCB イメージ 345
4245 プリンター
 デフォルト FCB イメージ 344
 別名 335
 UCS イメージ・テーブル 335
 UCS イメージ・テーブルの内容 336
4248 プリンター
 イメージ・テーブルの内容 337
 デフォルト FCB イメージ 344
 別名 335
 UCS イメージ・テーブル 335
6262 モデル 14 プリンター
 デフォルト FCB イメージ 344
 別名 335
 UCS イメージ・テーブル 335, 338

64 ビット IDAW (間接アドレッシング・
 ワード) 97, 144

A

ABE 付加ルーチン
 条件 148
 XENDA オペランド 118
ALTER ADDVOLUMES コマンド 354
AMCAP
 DEVTYPE マクロ 195
AMODE 値
 UCB 値に一致するための要件 382
APF (許可プログラム機能)
 EXCP 100, 102, 108
 EXCPVR 112
 VTOC 操作 65
 CVAFDIR BRANCH 383
 CVAFDIR DEB 380
 CVAFDIR IOAREA 380
 CVAFDSM BRANCH 397
 CVAFDSM DEB 394
 CVAFFILT BRANCH 401
 CVAFFILT DEB 400
 CVAFFILT IOAREA 401
 CVAFSEQ BRANCH 414
 CVAFSEQ DEB 411
 CVAFSEQ IOAREA 414
API 261

B

BLDL 283
BLOCKTOKENSIZE パラメーター
 ラージ・フォーマット・データ・セッ
 ト 126
BPX1PCT
 呼び出し可能サービス 359
 ChangeBufferLimits 360
 DisplayBufferLimits 360
 DisplayFSStats、使用
 使用 361
 DisplayGlobalStats、使用 361
 ExtendFS、使用 362
BPXPRMxx メンバー 357
BWO (backup-while-open) 機能
 標識ビット 268

C

CAMLST マクロ
 フォーマット 27, 28
 CAT(BX) オペランド 88
 RECAT オペランド 91
 RENAME オペランド 74, 81

CAMLST マクロ (続き)
 SCRATCH オペランド 67, 74
 SEARCH オペランド 27, 28
 SEEK オペランド 28, 29
 UNCAT オペランド 90

CATALOG マクロ
 CAT(BX) オペランド 88
 RECAT オペランド 91
 UNCAT オペランド 90

CCW (チャンネル・コマンド・ワード) 100

CDRA 261, 262

ChangeBufferLimits コマンド 360

CHE 付加ルーチン
 説明 149
 CENDA オペランド 118

CLOSE マクロ
 EXCP 111
 XDAP 159

confighfs コマンド 360

CPIO コマンド 357

CSW (チャンネル状況ワード) 72, 101

CVAF (共通 VTOC アクセス機能)
 アドレッシング・モード 43
 状況コード 72, 78
 推奨 412
 逐次化 44
 トレース呼び出し 62
 バッファ 384
 解放 385
 パラメーター・リスト xi
 フィルター・サービス
 必要な制御ブロック 56
 呼び出し 400
 DSCB セットの読み取り 55
 ボリュームの識別 44
 マクロ
 コーディング 379
 使用とフォーマット 379
 VTOC アクセス 43, 62
 戻りコード
 PARTREL 35
 RENAME 78
 SCRATCH 72

CVAFDIR マクロ
 コーディング 379
 使用 50
 パラメーター 380, 386
 戻りコード 386
 例 387, 392

CVAFDSM マクロ
 使用 43, 392
 パラメーター 393, 400
 フォーマット 392
 戻りコード 400

CVAFFILT マクロ
 形式 404

CVAFFILT マクロ (続き)
 項目 404
 使用 43, 400
 制御ブロック・アドレス解決 401
 制約事項 401
 パラメーター 402, 405
 ヒント 404
 フィルター基準リスト
 項目のフォーマット 58
 ヘッダー・フォーマット 56
 フォーマット 400
 部分修飾名、例 405
 戻りコード 405
 呼び出しシーケンス 60, 62
 例 406
 DSCB、読み取り 55
 RESUME 機能 55
 RLSE 機能 62

CVAFSEQ マクロ
 使用 43, 411
 パラメーター 412, 416
 フォーマット 411
 戻りコード 416
 例 424
 DSCB または DSN アクセス 53

CVAFST マクロ
 使用 43, 424
 フォーマット 424
 戻りコード 425

CVPL (CVAF パラメーター・リスト)
 機能 45, 47
 作成 45
 初期化 400
 フォーマット 45, 47
 マッピング 400

CVPL の CVFCTN フィールド 47

CVSTAT
 コード 35, 426
 推奨 405

D

DADSM (直接アクセス装置スペース管理) 31
 定義 1
 非 VSAM データ・セット、削除 67
 戻りコード 24
 割り振りルーチン 13
 戻りコード 42

OBTAIN ルーチン
 絶対装置アドレスによる読み取り 28
 データ・セット名による読み取り 27
 戻りコード 28, 29

DADSM (直接アクセス装置スペース管理) (続き)
 RENAME マクロ
 状況コード 79
 戻りコード 78

SCRATCH マクロ
 状況コード 73
 戻りコード 72

DASD 回転位置感知機構 101

DASD (直接アクセス・ストレージ・デバイス)
 スペースの解放 13
 データ転送速度 120
 データ・セット、作成 37
 トラック容量の計算 229
 パスワード保護データ・セット 169
 バッファ割り振り 120
 ブロック・サイズ 120
 ボリューム
 初期化 66
 スペース情報 17, 26
 スペース割り振り 16
 テープからの復元 66
 フラグメント化情報 17
 VTOC 状況 17
 読み取りと書き込み、XDAP マクロを使用した 157, 162

DASDVOL
 RACF、クラス 65, 66, 69

DCB (データ制御ブロック)
 アドレス 99
 初期化 125
 生成 99
 装置依存のフォーマット 119
 装置依存パラメーター (EXCP) 121, 125
 バッファ・パラメーター 119
 パラメーター
 基礎ブロック 117
 装置依存の 121
 EXCP 113
 フィールド 114
 復元 111
 マクロ命令
 XDAP 158
 EXCP 99, 125
 OPEN インストール・システム出口 120
 OPEN 後のフォーマット (EXCP) 116

DCBD マッピング・マクロ、EXCP の 125

DCBE (拡張データ制御ブロック) 98, 125
 EXCP 99

DCBFDAD フィールド 123

DCBOFLGS フィールド 109, 112

- DCBTRBAL フィールド 123
- DDR (動的装置再構成)
テープ・データ・セットの位置変更
117
- DEB (データ・エクステン・ブロック)
取得 44
フィールド 138
レイアウト (EXCP と EXCPVR) 427
DEBCHK による妥当性検査 179, 184
EXCP 99
- DEBCHK マクロ
機能 180, 184
仕様 180, 184
戻りコード 184
リスト形式 184
レジスターの内容 183
- DELETE
機能の置き換えとして 283
- DELETE コマンド 356
- DEQ マクロ、テープ・ボリューム・デマ
ウント機能 219
- DESERV 出口
グローバル出口 285
タスク・レベル出口 284
タスク・レベル出口とグローバル出口
の間の相互作用 285
- DELETE 283
GET 283
PUT 283
RENAME 283
SVC スクリーニング 283
SVCUPDTE 283
UPDATE 283
- DEV D
パラメーター 121
DCB オペランド 124
- DEVTYPE マクロ
実行形式 189
出力 200, 201
仕様 185, 186
戻り、理由コード 201
リスト形式 186, 191
例 202
AMCAP 195
INFO 形式 193
INFOLIST タイプ 186
RPS 装置 188
UCBLIST タイプ 186
- DFA
フィールド 435
マッピング 240
- DFADEEXT 290
- DFSMS レベルの判別 240
- DFSMSdfp 共用属性検索 276
- DFSMSdfp 呼び出し可能サービス
システム属性呼び出し 263
- DFSMSdfp 呼び出し可能サービス (続き)
システム・レベルの判別 263
説明 261
データ・セット
属性検索 266
backup-while-open サポート 268
戻りコード 277
呼び出し 262
理由コード 277
DFSMSdfp 共用属性検索 276
- DFSMSdfp レベルの判別 240, 263
- DFSMSdss
ダンプ・ユーティリティ (復
元) 357
- DisplayBufferLimits コマンド 360
- DisplayFSSStats コマンド 361
- DisplayGlobalStats コマンド 361
- DSCB (データ・セット制御ブロック) 4,
13
アクセス
索引付き 53
順次 53
直接 50
物理順次 54
書き込み
アドレスによる 379
データ・セット名により直接に 50
データ・セット名による 379
DSCB 位置により直接に 51
検索 402
索引なし VTOC 12
修飾データ・セット名 400
絶対トラック・アドレス 27
チェーン 60, 400
バッファー生成 387
フォーマット・シーケンス 2
マッピング 387
読み取り
アドレスによる 379
順次 53, 420
絶対装置アドレスによる 28
セット、CVAF フィルター・サー
ビス 55
データ・セット名により直接に 50
データ・セット名による 27, 379
バッファー・リスト 402
DSCB 位置により直接に 51
VTOC 389
format-7 13
ID 4
VSAM データ・スペース・エクステン
ト 9
VTOC 内容 10
VTOC のタイプ 2
- DSN 順、DSN へのアクセス、DSCB 53
- DSSIZE 値 39
- DSTYPE データ・セット属性 266
- DUMP コマンド 356
- ## E
- ECB (イベント制御ブロック)
完了コード 72, 78
EXCP 98, 136
XDAP 163
- end-of-data-set
ルーチン 118
- EOE (エクステン・終了) 付加ルーチン
説明 147
EOEA オペランド 118
- EOV (end-of-volume)
磁気テープ 109
データの消去 67
マクロ 111
データ・セット終了
(end-of-data-set) ルーチン、ユーザ
ー 109
EXCP 109
EXCP 109, 111
- EOV (ボリューム終了)
マクロ
XDAP 159
- EXCP (チャネル実行プログラム)
インターフェース・パラメーター 117
オーバーヘッド 120
開始と実行 99, 103
間接アドレッシング・ワードの要件
97
完了処理 103
コマンドのチェーニング 98
システムによる変換 143
状況情報 136
制御ブロック 125
エラー・リカバリー 129
DCB 99, 113, 125
DCBE 99
DEB 99, 138
ECB 98, 136
IOB 98, 127, 135
装置終了エラー・リカバリー手順 103
入出力エラー処理 103
付加ルーチン 138
プログラミング 95
インターフェース 96
考慮事項 101
変更 100
マクロ
フォーマット 109
CLOSE 111
EOV 109, 111
OPEN 107, 109
WAIT 96

EXCP (チャンネル実行プログラム) (続き)
 マクロ命令 113
 マルチボリューム・データ・セットの
 要件 108
 実
 ストレージ 112
 チャンネル・プログラム 112
 戻りコード 72
 問題プログラム 96
 ABE 付加ルーチン 148
 CHE 付加ルーチン 149
 DCBE 125
 EOE 付加ルーチン 147
 PCI 付加ルーチン 146
 PDSE 95
 SIO 付加ルーチン 141
 V=R アドレス・スペース 100
 EXCPVR 97
 EXCPVR マクロ
 使用 112
 フォーマット 112
 EXCP(VR)
 ユーザー独自のチャンネル・プログラム
 の実行 95
 DEB レイアウト 427
 ExtendFS コマンド 362

F

F2DSCB パラメーター
 サポートされない xx
 FCB (用紙制御バッファー) イメージ
 標準イメージ 344
 標準イメージ STD1 345
 標準イメージ STD2 346
 JES サポート 350
 SYS1.IMAGELIB
 イメージの追加 346, 348
 イメージの変更 348, 349
 機能 344, 349
 FCL (フィルター基準リスト)
 項目のフォーマット 58
 ヘッダー・フォーマット 56
 FORCE オプション 357
 format 0-6 DSCB, 概要 2
 format-1 DSCB, VTOC からの読み取り
 27

G

GDS (世代別データ・セット)
 名前による検索 84
 GET
 DESERV 283
 GTF トレース、CVAF 処理の 62

H

HFS
 計画 351
 HFS データ・セット 2, 5, 268
 管理 354
 説明 351
 ファイル・システム・アクティビティ
 - 355
 SMS 管理 352
 HFS ファイル
 装置特性情報 197
 トランスポート 355
 バックアップ 357
 マイグレーション 356
 リカバリー 357
 DEVTYPE が戻す UCBTYP 196
 RENAME マクロ 77
 HFS ファイル・システム
 削除 356

I

ICVEDT02 マッピング・マクロ 49
 IDAL (間接アドレッシング・リスト) 97
 IDAL (間接データ・アドレス・リス
 ト) 143
 IDAW (間接アドレッシング・ワー
 ド) 97, 144
 IDCAMS
 プログラミング・インターフェース
 369
 IEAAPPO0 140
 IEBUPDTE プログラム
 SYS1.PARMLIB 付加ルーチン・リス
 ト作成 140
 IEC301A メッセージ 68, 75
 IEC502E メッセージ 219
 IEC606I メッセージ 425
 IEC614I メッセージ 72, 78
 IECDPPL マクロ 225
 IECPDSCB マクロ 38
 IECSDSL1 マクロ 3, 387
 IEDB (入出力エラー・データ・ブロック)
 状況情報 105
 EXCP 99
 VIO に関する考慮事項 106
 IEEE POSIX 標準 2
 IEHLIST プログラム 63
 IEWPMAR マクロ 312, 326
 IEZDEB
 fields-EXCP(VR) 427
 IGGUCSIT マクロ 340
 IGW023A メッセージ 354
 IGW023I メッセージ 354
 IGWABWO 呼び出しステートメント
 使用 271
 フォーマット 268
 戻りコードと理由コード 277
 IGWARLS 呼び出しステートメント
 フォーマット 278
 戻りコード 280
 IGWASMS 呼び出しステートメント
 フォーマット 266
 戻りコードと理由コード 277
 IGWASYS 呼び出しステートメント
 フォーマット 263
 戻りコードと理由コード 277
 IGWDES マクロ 289, 326
 IGWLSHR 呼び出しステートメント
 フォーマット 276
 戻りコードと理由コード 277
 IGWSMDE マクロ 312, 326
 IHAARA マクロ 216
 IHAARL マクロ 214
 IHADFA
 制御ブロック 435
 フィールド 435
 マクロ 240
 IHADVA マクロ 184
 IHAPDS マクロ 294
 IMGLIB マクロ 348
 IMMEDIATE オプション 357
 INFO 形式
 DEVTYPE マクロ 193
 INFOLIST パラメーター
 DEVTYPE マクロ 186
 IOAREA 382
 IOB (入出力ブロック)
 センス・バイト 72, 78
 EXCP 98
 EXCP で使用されるフィールド 127,
 135
 PURGE マクロ 228
 XDAP 162
 XDAP フィールド 162
 IOBE (共通拡張入出力ブロック)
 拡張エラー情報の要求 105
 EXCP 99
 IOBEEIDA ビット 143
 IOBEIOT ビット 134
 IPL ボリューム 168
 I/O
 装置、特性 184
 ページ復元リスト 228
 要求
 再処理 229
 ページ 224
 復元 224

J

- JES (ジョブ入力サブシステム)
 - 印刷出力の処理 334
- JFCB (ジョブ・ファイル制御プログラム) xi
 - 使用されるマクロ
 - IHAARL 214, 219
 - OPEN 223
 - RDJFCB 206, 212
 - タイプ 07 出口リスト項目 209
 - 変更
 - 機能 204, 224
 - 予防措置 204

K

- KEYLEN オペランド 124

L

- LOCATE マクロ
 - カタログ情報の検索 82
 - 世代名 84
 - データ・セット名 82
 - 別名 86
 - 戻りコード 87
- locate-record CCW (EXCP) 101
- LookAt メッセージ検索ツール xiii
- LSPACE マクロ 24
 - 説明 26
 - データ戻り域 26
 - フォーマット 17
 - 拡張メッセージ戻り域 25
 - データ戻り域 25
 - パラメーター・リスト 23
 - メッセージ戻り域 24
 - メッセージ戻り域 26

M

- MINAU パラメーター 39
- MKDIR コマンド 353
- MODE
 - 制約事項 34
- MOUNT コマンド 353, 357
- MSG
 - 制約事項 20
- MSGDISP マクロ
 - パラメーター
 - 作動可能メッセージ 249
 - デマウント・メッセージ 250
 - 表示オプション 255
 - ボリュームの検査 246
 - ボリュームのマウント 244

- MSGDISP マクロ (続き)
 - パラメーター (続き) 253
 - メッセージ表示のリセット 253
 - 戻りコード 258

N

- nocapture オプション 110, 431
- NOPWREAD 保護モード標識 171
- NOWRITE 保護モード標識 171

O

- OBTAIN マクロ 27, 29
 - 制約事項 17
- OPEN マクロ
 - デマウント機能での DEQ、テープ・ボリューム 219
 - 変更された JFCB 223
- EXCP
 - 実行される手順 107
 - ダミー・データ・セットの制約事項 107
- TYPE=J
 - 仕様 224
 - 呼び出し 219
 - 例 211
- XDAP 158
- OPENJ (OPEN, TYPE=J) マクロ 223
- OSF XPG/4.2 標準 2

P

- PARTREL マクロ
 - 説明 29, 37
 - 戻りコード 35
 - リスト形式 34
 - DSECT 形式 34
- PASSWORD データ・セット
 - 作成 168
 - 制限 168
 - 保守 171
 - 要件 168
- PAX コマンド 357
- PCI (プログラム制御割り込み)
 - 説明 146
 - 付加ルーチン 118, 146
 - CCW 変更 101
 - PCIA オペランド 118
- PD1SCALO フラグ・バイト 39
- PDDIRQTY (ディレクトリー・ブロック数) 38
- PDPRIQTY (トラック数による 1 次スペース要求) 38

- PDSE (拡張区分データ・セット)
 - 共用プロトコル 276
 - スペース
 - 解放 29, 37
 - 未使用の解放 17
 - 属性検索 266
 - TRKCALC マクロ 229
- pfscpl
 - 呼び出し可能サービス (BPXIPCT) 359
- PGFX 付加ルーチン 142
- POSIX 2
- PROTECT マクロ xi
 - 機能 171
 - パラメーター・リスト
 - ADD 機能 173
 - DELETE 機能 175
 - LIST 機能 176
 - REPLACE 機能 174
 - フォーマット 172
 - 保護モード標識 171
 - 戻りコード 177
 - 要件 173
 - PASSWORD データ・セット 165, 177
- PURGE マクロ
 - 仕様 225
 - パラメーター・リスト 225
 - 戻りコード 227
- PUT
 - 機能の置き換えとして 283
- PWREAD 保護モード標識 171
- PWRITE 保護モード標識 171

R

- RACF (リソース・アクセス制御機能)
 - スクラッチ、データ・セットの 68
 - データ・セットの名前変更 74
 - 戻りコード 72
 - DASDVOL クラス 65, 66, 69
 - VTOC と VTOC 索引 65
- RACF (リソース・アクセス命名機能)
 - 戻りコード 78
- RDJFCB マクロ
 - 許可プログラムによる使用 210
 - 仕様 206
 - セキュリティ 210
 - 説明 206
 - デマウントでの DEQ の呼び出し 219
 - フォーマット
 - 割り振り検索域 216
 - 割り振り検索リスト 214
 - 戻りコード 209
 - 例 208

RDJFCB マクロ (続き)
 割り振り情報の検索
 例 217
 DCB 出口リスト項目 213
 DCB 出口リスト項目
 タイプ 07 209
 タイプ '13' 213
 REALLOC マクロ
 実行形式 38
 説明 37, 43
 戻りコード 42
 リスト形式 41
 DSECT 形式 41
 REFORMAT REFVTOC コマンド
 で VTOC 索引を再作成 14
 RENAME
 機能の置き換えとして 283
 RENAME パラメーター・リスト 78
 RENAME マクロ
 仕様 74
 状況コード 79
 戻りコード 78
 例 77
 SMS 管理ボリューム 74

RENT
 要件 340
 RESTORE マクロ 229
 RPS (回転位置感知機構)
 装置セクター番号 154, 163
 パラメーター 188

S

S99ACUCB ビット 107
 S99DSABA ビット 107
 S99TIOEX ビット 107
 SCRATCH コマンド 356
 scratch パラメーター・リスト 71
 SCRATCH マクロ
 状況コード 73
 説明 67
 戻りコード 72
 例 70
 SIO (入出力開始) 付加ルーチン
 エントリー・ポイント 142
 仮想アドレスの変換 143
 EXCP (チャンネル実行プログラム) 141
 SIOA オペランド 118
 SMF (システム管理機能)
 ボリューム情報 21
 SMS 管理の非 VSAM データ・セット
 要件 69
 SMS (ストレージ管理サブシステム)
 クラス名 266
 索引付き VTOC 14
 状況 263

SMS (ストレージ管理サブシステム) (続き)
 データ・セット、削除 67
 バージョン 263
 リリース・レベル 263
 VTOC
 データ・セットの削除 68
 SSGARGL 369
 STORCLASS パラメーター 352
 STOW 283
 SUFFIX、ブロック 197
 SVC スクリーニング 283
 SVCUPDTE 283
 SVC ルーチンの置換 284
 SYS1.CSSLIB データ・セット 261
 SYS1.IMAGELIB データ・セット
 追加
 FCB イメージ 346, 348
 UCS イメージ 329
 別名 334
 保守 327, 349
 FCB イメージの変更 348, 349
 UCS イメージ・テーブル 334

T

TAR コマンド 357
 TEST=YES
 要件 253, 255
 TRKCALC マクロ
 拡張順次データ・セットの場合 197
 実行形式 235
 出力 238
 説明 229, 240
 パラメーター・リスト
 空の、インライン 237
 作成 237
 初期化 235
 シンボリック拡張 237
 ストレージ定義 235
 リモート 235
 標準形式 229
 戻りコード 239
 リスト形式 237
 例 239
 DSECT 形式 237
 PDSE との矛盾 229
 TRTCH オペランド 124
 TYPE オペランド
 DEBCHK マクロ 181
 OPEN マクロ 224
 TYPE=J (OPEN マクロ) 223

U

UCATDX
 制約事項 91
 UCB (装置制御ブロック)
 索引 131
 制約事項 34
 トラック・アドレス 233
 マッピング・マクロ 197
 要件 18
 UCB パラメーター
 推奨 382, 413, 425
 AMODE 値に一致するための要件
 382
 UCBLIST パラメーター
 DEVTYPE マクロ 186
 UCBRPS ビット 154
 UCBVRDEB ビット 143
 UCS (汎用文字セット)
 イメージ
 検査 343, 344
 作成 329
 追加、JCL 331
 別名 334
 SYS1.IMAGELIB 追加 329
 SYS1.IMAGELIB、例 331, 334
 イメージ・テーブル
 イメージ名 339
 オブジェクト・モジュール 339
 項目のフォーマット 336
 項目の変更 339
 データの追加 341
 別名 339
 UCS5 の内容 336
 UCS6 の内容 337
 システム・イメージ・ライブラリーの
 保守 327
 VERIFY パラメーター 344
 UE (装置例外) 149
 UNCATLG コマンド 356
 UNIX ファイル・システム 2
 UNMOUNT コマンド 353, 354, 356, 357
 UPDATE
 機能の置き換えとして 283

V

VCB (ボリューム制御ブロック) 83
 VERIFY
 制約事項 382
 VIER (VTOC 索引項目レコード)
 仮想ストレージに保存 384
 説明 14
 特性 16
 見出し検索 379
 読み取り 379

VIO データ・セット
 EXCP 106

VIR (VTOC 索引レコード) 14
 解放 379
 バッファ、解放 381
 読み取り 379

VIXM (VTOC 索引マップ)
 説明 16
 ビットマップ
 割り振り DSCB、VIR 16
 割り振りスペースのマップ
 VIR 16

VPSM (VTOC パック・スペース・マップ)
 説明 16
 割り振り済みシリンダーとトラック
 16

VSAM (仮想記憶アクセス方式)
 データ・スペース、定義 2

VTOC 索引
 再作成 14

VTOC (ボリューム目録)
 アクセス 50
 CVAF マクロ 62
 DADSM マクロ 17, 43
 DSCB に順次に、CVAFSEQ を使用 53
 DSCB に直接、CVAFDIR を使用 50
 DSN に順次に、CVAFSEQ を使用 53

アクセス・マクロ
 CVAFDIR 50, 379
 CVAFDSM 392
 CVAFFILT 55, 400
 CVAFSEQ 53, 411
 CVAFTST 424
 LSPACE 17
 PARTREL 29
 REALLOC 37
 SCRATCH 67

位置決め 1
 記録機能 16
 更新 44
 構造 16
 索引
 維持管理 67
 エラー・メッセージ 425
 関係 15
 構造 15
 項目レコード (VIER) 16
 再作成 14
 作成 65
 説明 14
 テスト 424
 内容 14

VTOC (ボリューム目録) (続き)
 索引 (続き)
 名前 14
 名前変更 14
 パスワード保護 66
 ボリューム 66
 マップ (VIXM) 16
 リスト表示 63
 レコード 14
 RACF 保護 65

索引なし 12
 削除
 一時 VSAM データ・セット 67
 非 VSAM データ・セット 67

システム・サポート 424
 使用 1
 初期化 65
 説明 1
 データ・セットの名前変更 74
 内容 10
 パック・スペース・マップ
 (VPSM) 16
 変換 65
 変更 67, 379
 保護 65
 ボリューム・スペース割り振り 16
 要件 384, 385, 397, 416
 読み取り
 データ・セット名による 27, 29
 物理順 411
 DSN 順 411
 レコード 379
 APF 許可 65
 DSCB
 書き込み 379
 索引なし 2
 データ・セットの定義 2
 フォーマット 2
 マップ xi
 読み取り 379
 DSCB のマップ (VMDS) 14
 V=R アドレス・スペース 100
 V=V アドレス・スペース 101

W

WAIT マクロ 96
 WLR (誤長レコード) 149

X

XDAP (直接アクセス実行プログラム)
 制御ブロック
 DCB 158
 ECB 163

XDAP (直接アクセス実行プログラム) (続き)
 制御ブロック (続き)
 IOB 162
 チャンネル・プログラム 163
 マクロ
 仕様 160, 162
 CLOSE 159
 DCB 158
 EOVS 159
 OPEN 158
 要件 158, 159

Z

zFS データ・セット 2

[特殊文字]

@@D370\$, ヒント 295



プログラム番号: 5694-A01

Printed in Japan

SD88-6462-01



日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木3-2-12