

**IMS**  
バージョン 13

データベース・ユーティリティー

**IBM**



**IMS**  
バージョン 13

データベース・ユーティリティー  
—

**IBM**

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、519 ページの『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、IMS バージョン 13 (プログラム番号 5635-A04)、IMS Database Value Unit Edition V13.1 (プログラム番号 5655-DSM)、IMS Transaction Manager Value Unit Edition V13.1 (プログラム番号 5655-TM2)、および新しい版で明記されていない限り、以降のすべてのリソースおよびモディフィケーションに適用されます。

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： SC19-3653-04

IMS

Version 13

Database Utilities

(November 7, 2017 edition)

発行： 日本アイ・ビー・エム株式会社

担当： トランスレーション・サービス・センター

© Copyright IBM Corporation 1974, 2017.

# 目次

本書について . . . . .	vii
前提知識 . . . . .	vii
本書で使用されるIMS 機能名 . . . . .	vii
新規および変更された情報の識別方法 . . . . .	vii
構文図の読み方 . . . . .	viii
IMS バージョン 13 のアクセシビリティ機能 . . . . .	x

## 第 1 部 定義ユーティリティおよび初期設定ユーティリティ . . . . . 1

第 1 章 データベース接頭部解決ユーティリティ (DFSURG10) . . . . .	3
DFSURG10 ユーティリティの例 . . . . .	12

第 2 章 データベース接頭部更新ユーティリティ (DFSURGP0) . . . . .	15
DFSURGP0 ユーティリティの制御ステートメント . . . . .	21
DFSURGP0 ユーティリティの例 . . . . .	22
DFSURGP0 ユーティリティの異常終了後の再始動 . . . . .	27

第 3 章 データベース・スキャン・ユーティリティ (DFSURGS0) . . . . .	29
DFSURGS0 ユーティリティの制御ステートメント . . . . .	35
DFSURGS0 ユーティリティの例 . . . . .	37
DFSURGS0 ユーティリティの異常終了後の再始動 . . . . .	38

第 4 章 データベース調査ユーティリティ (DFSPRSUR) . . . . .	39
DFSPRSUR ユーティリティの制御ステートメント . . . . .	43
DFSPRSUR ユーティリティの例 . . . . .	45

第 5 章 DEDB 変更ユーティリティ (DBFUDA00) . . . . .	55
DEDB 変更ユーティリティ (DBFUDA00) の制御ステートメント . . . . .	58
DBFUDA00 ユーティリティの例 . . . . .	62

第 6 章 DEDB 初期設定ユーティリティ (DBFUMIN0) . . . . .	65
DBFUMIN0 ユーティリティの制御ステートメント . . . . .	70
DBFUMIN0 ユーティリティの例 . . . . .	72

第 7 章 DEDB 順次従属削除ユーティリティ (DBFUMDL0) . . . . .	75
DBFUMDL0 ユーティリティの制御ステートメント . . . . .	77
DBFUMDL0 ユーティリティの例 . . . . .	81
DBFUMDL0 ユーティリティの処理の範囲 . . . . .	82

第 8 章 DEDB 順次従属スキャン・ユーティリティ (DBFUMSC0) . . . . .	85
DBFUMSC0 ユーティリティの制御ステートメント . . . . .	88
DBFUMSC0 ユーティリティの例 . . . . .	94
DBFUMSC0 ユーティリティのスキャンの範囲 . . . . .	95

第 9 章 HALDB 区画データ・セット初期設定ユーティリティ (DFSUPNT0) . . . . .	99
DFSUPNT0 ユーティリティの制御ステートメント . . . . .	104
DFSUPNT0 ユーティリティの例 . . . . .	104

第 10 章 HALDB 区画定義ユーティリティ (%DFSHALDB) . . . . .	107
%DFSHALDB ユーティリティの制御ステートメント . . . . .	113
%DFSHALDB ユーティリティの例 . . . . .	115
%DFSHALDB ユーティリティの実行 . . . . .	115
HALDB 区画定義ユーティリティの ISPF パネル . . . . .	116

第 11 章 MSDB 保守ユーティリティ (DBFDBMA0) . . . . .	147
DBFDBMA0 ユーティリティの制御ステートメント . . . . .	152
DBFDBMA0 ユーティリティの例 . . . . .	156
MSDB 保守ユーティリティの機能 . . . . .	158

## 第 2 部 バックアップ・ユーティリティ . . . . . 161

第 12 章 データベース・イメージ・コピー・ユーティリティ (DFSUDMP0) . . . . .	163
DFSUDMP0 ユーティリティの制御ステートメント . . . . .	173
DFSUDMP0 ユーティリティの例 . . . . .	174

第 13 章 データベース・イメージ・コピー 2 ユーティリティ (DFSUDMT0) . . . . .	177
DFSUDMT0 ユーティリティの制御ステートメント . . . . .	186
DFSUDMT0 ユーティリティの実行 . . . . .	196

第 14 章 オンライン・データベース・イメージ・コピー・ユーティリティ (DFSUICP0) . . . . .	205
DFSUICP0 ユーティリティの制御ステートメント . . . . .	211
DFSUICP0 ユーティリティの例 . . . . .	212

DFSUICP0 の障害後のイメージ・コピー・ジョブの  
再始動. . . . . 213

---

### 第 3 部 リカバリー・ユーティリテ ィー . . . . . 215

#### 第 15 章 バッチ・バックアウト・ユー ティリティー (DFSBB000) . . . . . 217

DFSBB000 ユーティリティーのユーティリティー  
制御ステートメント . . . . . 228  
DFSBB000 ユーティリティーの例 . . . . . 233  
RSR 環境における DFSBB000 ユーティリティーの  
実行 . . . . . 233

#### 第 16 章 データベース変更累積ユーテ ィリティー (DFSUCUM0) . . . . . 235

DFSUCUM0 ユーティリティーの制御ステートメン  
ト . . . . . 243  
DFSUCUM0 ユーティリティーの例 . . . . . 250

#### 第 17 章 データベース・リカバリー・ ユーティリティー (DFSURDB0) . . . . . 255

DFSURDB0 ユーティリティーの制御ステートメン  
ト . . . . . 267  
DFSURDB0 ユーティリティーの例 . . . . . 270

#### 第 18 章 DEDB エリア・データ・セッ ト比較ユーティリティー (DBFUMMH0) . 275

DBFUMMH0 ユーティリティーの制御ステートメン  
ト . . . . . 278  
DBFUMMH0 ユーティリティーの例 . . . . . 280

#### 第 19 章 DEDB エリア・データ・セッ ト作成ユーティリティー (DBFUMRI0) . 283

DBFUMRI0 ユーティリティーの制御ステートメン  
ト . . . . . 286  
DBFUMRI0 ユーティリティーの例 . . . . . 288

#### 第 20 章 HALDB 索引/ILDS 再作成ユ ーティリティー (DFSPREC0) . . . . . 291

DFSPREC0 ユーティリティーの制御ステートメン  
ト . . . . . 295  
DFSPREC0 ユーティリティーの例 . . . . . 296  
DFSPREC0 ユーティリティーの実行 . . . . . 297

#### 第 21 章 MSDB ダンプ・リカバリー・ ユーティリティー (DBFDBDR0) . . . . . 299

DBFDBDR0 ユーティリティーの制御ステートメン  
ト . . . . . 303  
DBFDBDR0 ユーティリティーの例 . . . . . 305

---

### 第 4 部 再編成ユーティリティーお よび変換ユーティリティー . . . . . 309

#### 第 22 章 データベース事前再編成ユー ティリティー (DFSURPR0) . . . . . 311

DFSURPR0 ユーティリティーの制御ステートメン  
ト . . . . . 317  
DFSURPR0 ユーティリティーの例 . . . . . 319

#### 第 23 章 HALDB マイグレーション・ エイド・ユーティリティー (DFSMAID0). 321

DFSMAID0 ユーティリティーの制御ステートメン  
ト . . . . . 325  
DFSMAID0 ユーティリティーの例 . . . . . 326  
DFSMAID0 ユーティリティーのデータ・スペース  
およびサンプリング . . . . . 327

#### 第 24 章 HD 再編成再ロード・ユーティ リティー (DFSURGL0) . . . . . 331

DFSURGL0 ユーティリティーのユーティリティー  
制御ステートメント . . . . . 339  
DFSURGL0 の出力メッセージと統計 . . . . . 342  
DFSURGL0 の例 . . . . . 344

#### 第 25 章 HD 再編成アンロード・ユーテ ィリティー (DFSURGU0) . . . . . 349

DFSURGU0 ユーティリティーの制御ステートメン  
ト . . . . . 361  
DFSURGU0 ユーティリティーの例 . . . . . 362

#### 第 26 章 高速 DEDB 直接再編成ユーテ ィリティー (DBFUHDR0) . . . . . 369

DBFUHDR0 ユーティリティーの制御ステートメン  
ト . . . . . 373  
DBFUHDR0 ユーティリティーの実行 . . . . . 376

#### 第 27 章 HISAM 再編成再ロード・ユー ティリティー (DFSURRL0) . . . . . 381

DFSURRL0 ユーティリティーの制御ステートメン  
ト . . . . . 388  
DFSURRL0 ユーティリティーの例 . . . . . 389

#### 第 28 章 HISAM 再編成アンロード・ユ ーティリティー (DFSURUL0) . . . . . 391

DFSURUL0 ユーティリティーの制御ステートメン  
ト . . . . . 401  
DFSURUL0 ユーティリティーの例 . . . . . 403

#### 第 29 章 MSDB - DEDB 間変換ユーテ ィリティー (DBFUCDB0) . . . . . 407

DBFUCDB0 ユーティリティーの制御ステートメン  
ト . . . . . 411  
DBFUCDB0 ユーティリティーの例 . . . . . 411  
MSDB の DEDB への変換 . . . . . 415  
DEDB から MSDB へのフォールバックの実行 . . 415  
DBFUCDB0 ユーティリティーの変換およびフォ  
ールバックの出口ルーチン . . . . . 416

第 30 章 データベース部分再編成ユーティリティ (DFSPRCT1 および DFSPRCT2) . . . . .	419
データベース部分再編成ユーティリティの制御ステートメント . . . . .	427
データベース部分再編成ユーティリティの例 . . . . .	429
第 31 章 ユーティリティ制御機能 (DFSUCF00) . . . . .	437
DFSUCF00 ユーティリティの制御ステートメント . . . . .	448
DFSUCF00 ユーティリティの例 . . . . .	483
DFSUCF00 ユーティリティの実行 . . . . .	484
<hr/>	
第 5 部 報告書ユーティリティおよびテスト・ユーティリティ . . . . .	495
第 32 章 データベース・モニター報告書印刷ユーティリティ (DFSUTR30) . . . . .	497
DFSUTR30 ユーティリティの例 . . . . .	499
第 33 章 プログラム分離トレース報告書ユーティリティ (DFSPIRPO) (Program-Isolation-Trace Report utility (DFSPIRPO)) . . . . .	501
DFSPIRPO ユーティリティの制御ステートメント . . . . .	505

DFSPIRPO ユーティリティの例 . . . . .	506
第 34 章 SB テスト・ユーティリティ (DFSSBHD0) . . . . .	507
DFSSBHD0 ユーティリティの制御ステートメント . . . . .	514
DFSSBHD0 ユーティリティの例 . . . . .	516
<hr/>	
第 6 部 付録 . . . . .	517
特記事項 . . . . .	519
プログラミング・インターフェース情報 . . . . .	521
商標 . . . . .	521
製品資料に関するご使用条件 . . . . .	522
IBM オンライン・プライバシー・ステートメント . . . . .	522
参考文献 . . . . .	525
索引 . . . . .	527





---

## 本書について

これらのトピックでは、データベースのマイグレーション、再編成、およびリカバリーに使用できるユーティリティーの参照情報を提供します。

この情報は、IBM® Knowledge Center で参照できます。

---

## 前提知識

本書に記載のユーティリティーを使用する前に、z/OS® オペレーティング・システムに関連した基本概念およびタスクについて理解する必要があります。さらに重要なこととして、データベース編成、アクセス方式、索引付け方式、論理関係、DEDB 区域、HALDB 区画化などの IMS™ データベースの概念を理解することが必要です。また、データベースの再編成、変換、バックアップ、およびリカバリーといったプロセスについての一般的な知識も必要です。

IMS システムおよびデータベースについてあらかじめ必要な情報の多くは、次の文書に記載されています。

- IMS V13 システム管理
- IMS V13 データベース管理

z/OS の詳細については、IBM Knowledge Center の「z/OS basic skills」トピックを参照してください。

IMS の基本概念を理解するには、「*An Introduction to IMS*」(IBM Press 出版)をお読みになると役立ちます。

IBM では、IMS の学習に役立つような講習会や自習講座を数多く提供しています。利用可能な講習の詳しいリストについては、IBM Skills Gateway にアクセスして、IMS を検索してください。

---

## 本書で使用されるIMS 機能名

本書では、「HALDB オンライン再編成」という用語は、特に断りがない限り、IMS バージョン 13 の一部として組み込まれた HALDB オンライン再編成機能を指しています。

---

## 新規および変更された情報の識別方法

IMS ライブラリーの PDF 資料のほとんどの新規および変更された情報は、左マージン内の文字 (改訂マーカ) によって示されています。「リリース計画」、ならびに「*Program Directory*」および「*Licensed Program Specifications*」の第 1 版 (-00) には、改訂マーカは含まれていません。

改訂マーカは、以下の一般的な規則に従っています。

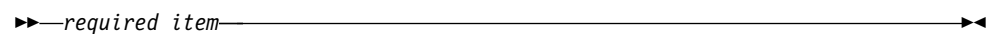
- 技術的な変更のみにマークが付けられています。形式上の変更や文法的な変更には、マークは付けられていません。
- 段落、構文図、リスト項目、操作手順、または図などの要素の一部が変更された場合、その要素の一部だけの変更であっても、要素全体に改訂マークが付けられています。
- トピックの変更が 50% を超えた場合には、そのトピック全体に改訂マークが付けられています (そのため、新規トピックではなくても、新規トピックのように見えることがあります)。

改訂マークは情報に加えられたすべての変更を示しているとは限りません。削除されたテキストとグラフィックスには、改訂マークでマークを付けることはできないためです。

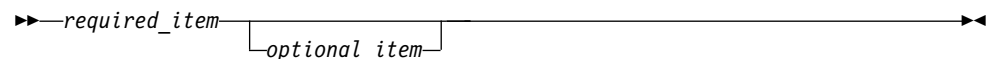
## 構文図の読み方

本書で使用されている構文図には、以下の規則が適用されています。

- 構文図は、経路を示す線に沿って、左から右、上から下に読み取ります。以下の規則が使用されます。
  - >>--- 記号は、構文図の始まりを示します。
  - ---> 記号は、構文図が次の行に続くことを示します。
  - >--- 記号は、この構文図が直前の行から続いていることを示します。
  - --->< 記号は、構文図の終わりを示します。
- 必須項目は、水平線 (メインパス) 上に表示されます。



- オプション項目は、メインパスより下に示されます。

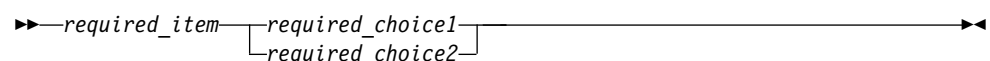


メインパスより上にオプション項目が示されている場合は、その項目が構文エレメントの実行に影響することはない、読みやすくするためのみの表記です。

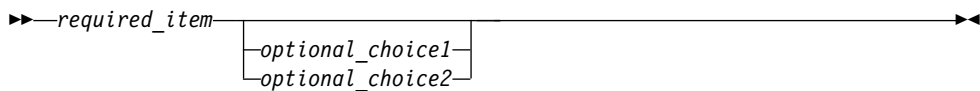


- 複数の項目から選択できる場合は、縦方向に並べて (スタック) 示されます。

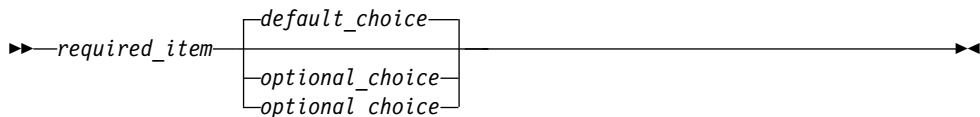
それらの項目の中から 1 つを選択する必要がある場合は、スタックの中の 1 つの項目がメインパス上に表示されます。



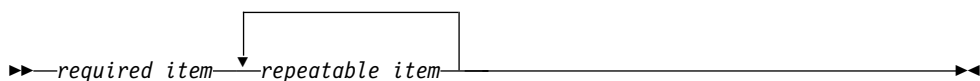
それらの項目から 1 つを選択することがオプションである場合は、スタック全体がメインパスの下に表示されます。



デフォルト項目が含まれている場合、その項目はメインパスより上に示され、他の選択項目はメインパスより下に示されます。



- メインパスの上方にある左に戻る矢印線は、項目が反復可能であることを示します。

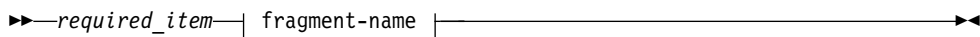


反復矢印線にコンマが含まれている場合は、反復項目をコンマで区切る必要があります。

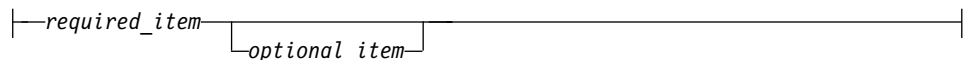


スタック上方の反復矢印線は、スタック内の項目を反復できることを示しています。

- 1 つの構文図を複数のフラグメントに分割しなければならない場合もあります。構文フラグメントはメインの構文図とは別に示されますが、フラグメントの内容は、図のメインパス上にあるものとして読む必要があります。



**fragment-name:**



- IMS では、b 記号は、該当位置に空白が 1 つあることを示します。
- キーワード、および該当する場合はキーワードの最小の省略語は、大文字で表されます。これらは、示されているとおりに入力する必要があります。変数は、すべて小文字のイタリック文字で示されます (例えば、*column-name*)。これらは、ユーザーが指定する名前または値を表します。
- キーワードとパラメーターは、構文図で間に句読点が表示されていない場合は、少なくとも 1 つのスペースで分離します。
- 句読記号、括弧、算術演算子、およびその他の記号は、構文図で示されたとおりに入力します。

- 脚注は、例えば (1) のように、数字を括弧で囲んで示してあります。

---

## IMS バージョン 13 のアクセシビリティ機能

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情報技術製品を快適に使用できるようにサポートします。

### アクセシビリティ機能

以下のリストは、IMS バージョン 13 を含む z/OS 製品の主なアクセシビリティ機能を示しています。これらの機能は、以下をサポートしています。

- キーボードのみの操作。
- スクリーン・リーダー (読み上げソフトウェア) およびスクリーン拡大鏡によって通常使用されるインターフェース。
- 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。

### キーボード・ナビゲーション

IMS バージョン 13 ISPF パネル機能には、キーボードまたはキーボード・ショートカット・キーを使用してアクセスできます。

TSO/E または ISPF を使用して IMS バージョン 13 ISPF パネルをナビゲートする詳細については、「z/OS TSO/E 入門」、「z/OS TSO/E ユーザーズ・ガイド」、および「z/OS 対話式システム生産性向上機能 (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、各インターフェースのナビゲート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。

### 関連のアクセシビリティ情報

IMS バージョン 13 のオンライン資料は、IBM Knowledge Center で参照できます。

### IBM におけるアクセシビリティ

IBM のアクセシビリティに対する取り組みについて詳しくは、*IBM Human Ability and Accessibility Center* ([www.ibm.com/able](http://www.ibm.com/able)) を参照してください。

---

## 第 1 部 定義ユーティリティーおよび初期設定ユーティリティー

定義ユーティリティーおよび初期設定ユーティリティーは、高速機能 DEDB データベースや全機能 HALDB パーティション・データベースなどの、高速機能データベースおよび全機能データベースを定義するために使用します。



---

## 第 1 章 データベース接頭部解決ユーティリティー (DFSURG10)

データベース接頭部解決ユーティリティー (DFSURG10) は、1 つ以上のデータベースのロード時、再編成時、またはその両方が行われるときに作業データ・セットに生成される情報を累積するのに使用します。

DFSURG10 ユーティリティーは、データベースについて定義された論理関係を完成するために必要な接頭部情報が入った出力データ・セットを作成します。このユーティリティーは、副次索引データベースの作成または更新に必要な情報が入った出力データ・セットも作成できます。

DFSURG10 ユーティリティーは、データベース事前再編成ユーティリティー (DFSURPR0) の実行により作成される制御データ・セットを必要とし、ユーティリティー制御ステートメントは使用しません。

HALDB データベースには接頭部解決は必要ありません。

このユーティリティーの機能は、ユーティリティー制御機能 (DFSUCF00) に実行させることもできます。

以下の図は、データベース接頭部解決ユーティリティーのフロー・ダイアグラムです。

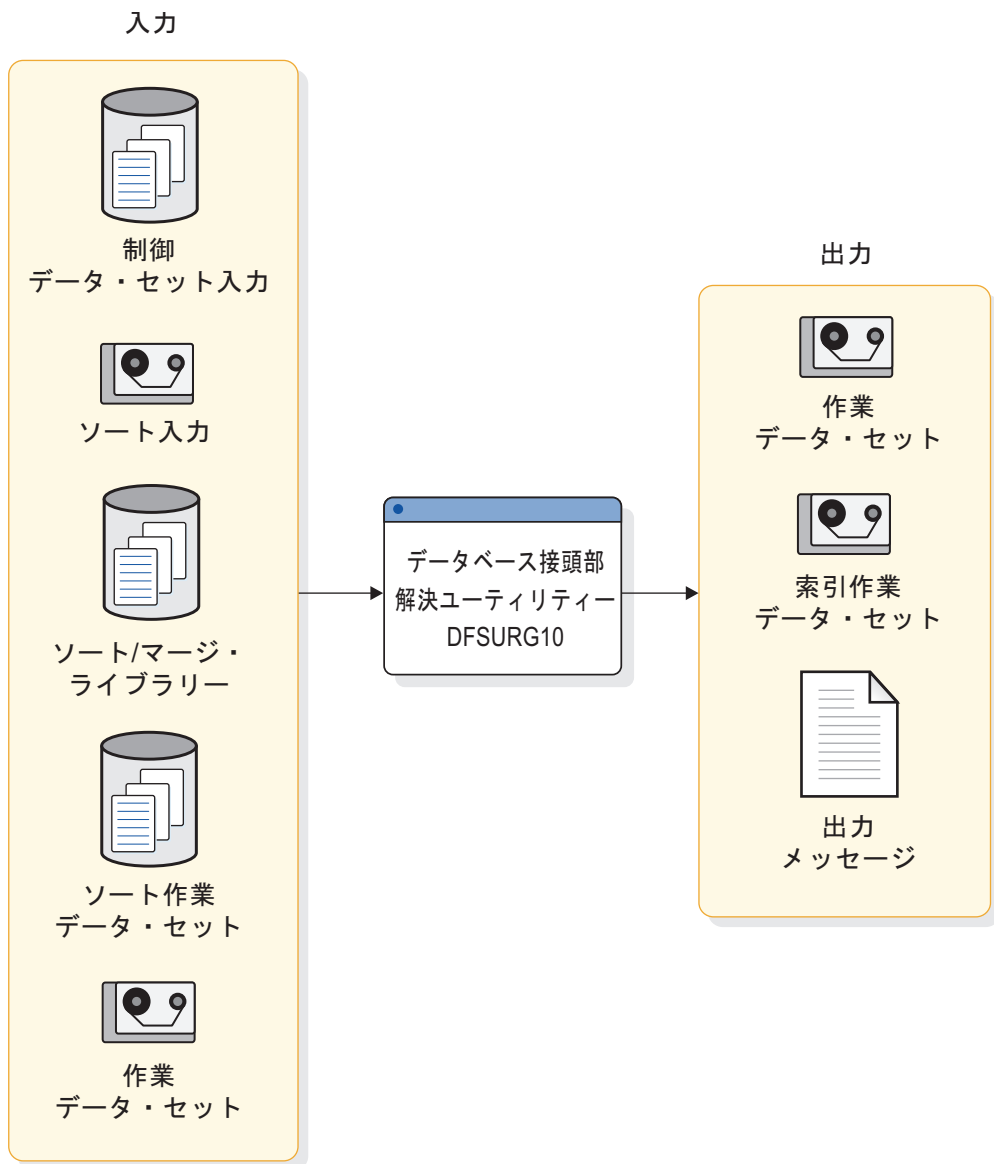


図 1. データベース接頭部解決ユーティリティ

サブセクション:

- 5 ページの『制約事項』
- 6 ページの『前提条件』
- 6 ページの『要件』
- 6 ページの『推奨事項』
- 6 ページの『入出力』
- 7 ページの『戻りコード』
- 7 ページの『JCL 指定』



## 制約事項

DFSURG10 ユーティリティは z/OS ソート/マージ・プログラムを使用します。z/OS ソート/マージ・プログラムで使用できる最大ソート・フィールドは 256 文字であり、次のような制約があります。

- どのような論理親と論理子の対の場合も、次の項目 1 と項目 2 の合計の長さは、200 文字を超えてはなりません (残りの 56 文字は、制御目的のために IMS によって使用されます)。
  1. 論理親の連結キーの長さ
  2. 論理親から見たときの論理子のシーケンス・フィールドの長さ
- 合計の計算は、論理親について 1 回、論理子について 1 回行わなければなりません。これらの合計は別々に扱われます。
- これらの数量の 1 つ以上を、以下のように合計から省くことができます。
  - 論理親が初期ロードされる場合または論理子が論理親ポインターで論理親を指していない場合は、論理親の連結キーの長さを両方の限度検査に含めなければなりません。
  - 論理子が初期ロードされる場合かつ論理子が論理兄弟チェーンを持つ場合は、その論理親から見たときの論理子のシーケンス・フィールドの長さは、論理子の限度検査に含めなければなりません。そうでない場合は、省くことができます。

論理親と論理子のどちらかで限度検査に失敗したときは、データベースの初期ロード時に論理親または論理子をロードすることを省くことができます。ロードしなかった論理親または論理子は、更新モードで稼働するアプリケーション・プログラムによって、後でデータベースに挿入することができます。データベースがロードされた後では、限度検査の 1 つ以上の要素を省くことができます。

データベース事前再編成ユーティリティは、意図したデータベースの初期ロードまたは再ロードによって影響を受ける論理親と論理子の組み合わせについて限度検査を行います。この限度検査は最悪の場合を想定した検査です。論理親と論理子の組み合わせに対する限度検査が失敗すると、メッセージ DFS885 が出力されます。

IMS では、データベースの初期ロード時にキーを持っていないセグメントや非固有キーを持つセグメントの順序は決まっておらず、z/OS のソート/マージ・プログラムでは、キー値が等しいレコードが先入れ先出しの順序になるという保証はありません。このような理由から、このプログラムを連続して実行した場合または再編成を連続して実行した場合、これらのタイプの論理子セグメントの順序が不整合になる可能性があります。

DFSURG10 ユーティリティは、全機能非 HALDB データベースのみで機能します。

データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

## 前提条件

DFSURG10 ユーティリティを実行する前に、DFSURG10 ユーティリティが入力として使用するデータ・セットを生成するために、まずほかのアプリケーション・プログラムまたはユーティリティを実行する必要があります。実行しなければならないアプリケーション・プログラムまたはユーティリティ・プログラムと、それらのプログラムが作成するデータ・セットには、以下のようなものがあります。

- データベース事前再編成ユーティリティ (DFSURPR0)。DFSURG10 ユーティリティで使用される制御データ・セットを生成します。
- ロード、再編成、またはスキャン用のアプリケーション・プログラムまたはユーティリティ。DFSURG10 ユーティリティで使用される必須 DFSURWF1 作業データ・セットを作成します。DFSURWF1 作業データ・セットを作成するために実行するアプリケーション・プログラムまたはユーティリティのタイプは、完了する必要があるタスクによって異なります。

## 要件

現在、DFSURG10 ユーティリティに対して文書化された要件はありません。

## 推奨事項

DFSURG10 ユーティリティの EXEC ステートメントでサイズを指定するときは、可能ならば見積もりのファイル・サイズを使用してください。これにより、SM1 の最適化およびパフォーマンスが向上するからです。

## 入出力

DFSURG10 ユーティリティへの 1 次入力、ロード、再編成、またはスキャン用のアプリケーション・プログラムまたはユーティリティで作成される作業データ・セット、および、データベース事前再編成ユーティリティ (DFSURPR0) で作成される制御データ・セットです。DFSURG10 ユーティリティは、他のオプションの入力および必須入力も使用します。

DFSURG10 ユーティリティにサブミットされる DFSURWF1 作業データ・セットは、連結して 1 つの SORTIN データ・セットを形成する必要があります。DCB 属性が異なるデータ・セットが複数あるときは、LRECL が最も大きいデータ・セットを連結の先頭に置いてください。

DFSURG10 ユーティリティの 1 次出力には、データベース接頭部更新ユーティリティ (DFSURGP0) で使用される作業データ・セット、およびオプションで、HISAM アンロード・プログラム (DFSURUL0) で副次索引の作成、置換、マージ、抽出に使用される作業データ・セットがあります。DFSURG10 ユーティリティは、メッセージおよび統計も作成します。

データベース接頭部解決ユーティリティの入出力は、以下の表に示すとおりです。

表 1. データベース接頭部解決ユーティリティの入出力

入力	出力
制御データ・セット	作業データ・セット
ソート入力	メッセージ
ソート/マージ・ライブラリー	統計 (STAT または SUMM がデータベース 事前再編成ユーティリティ (DFSURPR0) 制御ステートメントで指定されている場合)
ソート作業データ・セット	
作業データ・セット	
索引作業データ・セット	

## 戻りコード

プログラム終了時に次の戻りコードが返されます。

コード 意味

- 0 エラーは検出されなかった
- 4 プログラム実行中に次のメッセージのいずれかが出されたとき、あるいは次のメッセージが DB 事前再編成中に SUMM パラメーターによって先に出されたとき返されます。  
  
DFS878, DFS885, DFS961.
- 8 接頭部更新で必要になるデータが WF3 データ・セットに書き込まれなかったとき、あるいは次のメッセージの 1 つ以上がプログラム実行中に出されたとき返されます。  
  
DFS852, DFS855, DFS857, DFS876, DFS877, DFS879, DFS880, DFS881.
- 12 戻りコード 4 の箇所にリストされたメッセージの 1 つまたは 2 つおよび戻りコード 8 の箇所にリストされたメッセージの 1 つ以上が出されたとき返されます。
- 16 またはそれ以上  
z/OS ソート/マージ・プログラムから返されます。

8、12、16 のいずれかの戻りコードが接頭部解決ユーティリティ (DFSURG10) から返されたときは、接頭部更新ユーティリティ (DFSURGP0) が必要とする入力作業データ・セットが DFSURG10 によって生成されていないことがあるので、DFSURGP0 を実行しないでください。診断メッセージに示されたエラーを訂正し、データベース操作をやり直してからデータベース接頭部解決ユーティリティの再実行を試みてください。

戻りコード 4 が返されたときは、起こりうるエラーが発生している場合と、そうでない場合があります。

## JCL 指定

DFSURG10 ユーティリティは、標準の z/OS ジョブとして実行されます。DFSURG10 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステ

ートメント、および DD ステートメントが含まれます。DFSURG10 ユーティリティーはユーティリティー制御ステートメントを使用しません。

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

DFSURG10 ユーティリティーにより使用される z/OS ソート/マージ・プログラム用にオプションを指定するために、EXEC ステートメントの IMS サポート PARM フィールド・オプションを使用できます。

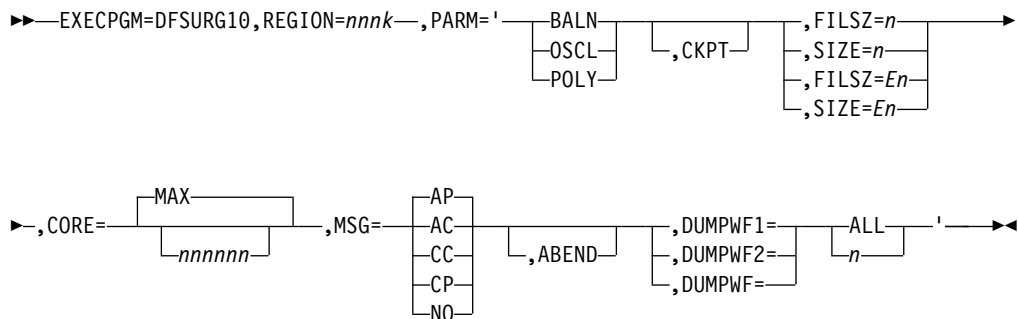
### EXEC ステートメント

EXEC ステートメントは次の例で示す形式になっていなければなりません。

```
//STEP EXEC PGM=DFSURG10,REGION=100K,PARM='CKPT,MSG=AC,FILSZ=E5000'
```

データベース接頭部解決ユーティリティーはソート・プログラムを個別に呼び出します。REGION= パラメーターに指定するサイズを大きくすると、ソート・プログラムによってはパフォーマンスが向上するものもあります。

PARM フィールドの指定は定位置の指定ではありません。以下の構文図は、すべての PARM フィールド・オプションとそのデフォルト指定を示します。



この構文図に示したキーワードだけが、接続されている z/OS ソート/マージ・プログラムに渡されます。

### ABEND

診断の目的のためにストレージ・ダンプを必要とするときだけ指定してください。ABEND を指定するときは、SYSUDUMP DD ステートメントも必要です。

### CKPT

DFSURG10 ユーティリティーが z/OS ソート/マージ・プログラムにリンク (アタッチではなく) することを指定します。

### CORE

z/OS ソート/マージ・プログラムによって使用される主記憶域の量を 6 桁の数値で指定します。ソートのタイプおよび使用するソート・デバイスに基づいて CORE= の値を計算します。

CORE= のデフォルト値は MAX です。この値は z/OS ソート/マージ・プログラムが使用する主記憶域の量に制限を設けません。

推奨事項: デフォルト指定の CORE=MAX を使用しないときに、何を指定するかが分からなければ、データ機能分類プログラム (DFSORT) で必要とされる最小限の量を使用してください。16 MB 境界より下に装置制御ブロック (UCB) を割り振る OEM ソート製品を使用する場合は、CORE=MAX を使用してください。

#### DUMPWF1、DUMPWF2、および DUMPWF

DFSURWF1 と DFSURWF2 の作業ファイル・レコードを、ソートの 1 回目または 2 回目の実行から出力されるとおりに表示する必要があるときだけ使用される診断機能です。

DUMPWF1 を使用すると、最初のソートで得られた指定レコード数 (ソート後の DFSURWF1) が SYSPRINT に出力されます。DUMPWF2 を指定すると、ソート後の DFSURWF2 にも同じサービスを要求します。DUMPWF1 要求と DUMPWF2 要求は同時に使用できますが、 $n$  の値が両方とも同じであれば、DUMPWF を指定しても同じです。 $n$  に指定する値には、9 桁まで使用できます。

#### FILSZ

FILSZ= $n$  を指定する場合、 $n$  には、ソートするデータ・セット内の正確なレコード数を指定します。この数には、出口 E15 で挿入または削除されるレコードがあればそれも含めておく必要があります。

入力データ・セット内のレコード数が  $n$  に指定した値と同じでないと、プログラムは終了し、値  $n$  がメッセージ IGH047A または IGH054I の IN フィールドで示されます。

FILSZ= $En$  を指定する場合、 $n$  には、ソートするレコードの見積もり数を指定します。 $n$  に指定する値は、入力データ・セットと出口 E15 で追加または削除されるレコードの両方を見込んだ大きさにしておかなければなりません。

例えば、総データ・セット・サイズを 5000 レコードと見積もる場合は、FILSZ=E5000 と指定してください。最大許容サイズは、E+8(Ennnnnnnnn) です。

平衡ディスク手法を用いている場合、z/OS ソート/マージ・プログラムは、次のどちらかを示すメッセージ IGH070I を印刷します。

- ファイルのサイズが指定されていなかった。
- レコード数が 10 進数で指定されていなかった。

このオペランドの指定がないと、z/OS ソート/マージ・プログラムは次のことを想定します。

- 中間ストレージがテープであれば、入力データ・セットは、ソートで 사용되는ブロック化因数で 1 つのボリューム上に置いておくことができる。
- 中間ストレージが直接アクセス装置であれば、入力データ・セットは割り振られたスペースに収容できる。

SM1 の最適化およびパフォーマンスが向上するので、可能ならば見積もりのファイル・サイズを使用してください。

## MSG

z/OS ソート/マージ・プログラムで印刷されるメッセージを制御します。デフォルト指定 AP の場合、すべての DFSORT メッセージを印刷し、重大なメッセージのみ z/OS マスター・コンソールに送信します。

## OSCL

テープ作業データ・セットにオシレーティング手法を指定します。OSCL を指定する場合は、SIZE パラメーターを指定しなければなりません。

DFSORT はこの EXEC PARM オプションを受け入れますが、処理は行いません。

## SIZE

SIZE= $n$  を指定する場合、 $n$  には、入力データ・セット内の正確なレコード数を指定します。ただし、出口 E15 で変更されるレコードは除きます。SM1 は FILSZ または SIZE を受け付けますが、FILSZ を使用すると最適化がさらに促進されるので、使用できる場合には、必ず FILSZ を使用するようにしてください。

入力データ・セット内のレコード数が  $n$  に指定した値と同じでないと、プログラムは終了し、値  $n$  がメッセージ IGH047A または IGH054I の IN フィールドで示されます。

SIZE= $En$  を指定する場合、 $n$  には、ソートするレコードの見積もり 数を指定します。 $n$  に指定する値は、入力データ・セットと出口 E15 で追加または削除されるレコードの両方を見込んだ大きさにしておかなければなりません。

例えば、総データ・セット・サイズを 5000 レコードと見積もる場合は、FILSZ=E5000 と指定してください。最大許容サイズは、E+8(Ennnnnnnn) です。

平衡ディスク手法を用いている場合、z/OS ソート/マージ・プログラムは、次のどちらかを示すメッセージ IGH070I を印刷します。

- ファイルのサイズが指定されていなかった。
- レコード数が 10 進数で指定されていなかった。

このオペランドの指定がないと、z/OS ソート/マージ・プログラムは次のことを想定します。

- 中間ストレージがテープであれば、入力データ・セットは、ソートで使用されるブロック化因数で 1 つのボリューム上に置いておくことができる。
- 中間ストレージが直接アクセス装置であれば、入力データ・セットは割り振られたスペースに収容できる。

見積もりのファイル・サイズを使用すると、大幅に SM1 の最適化が向上し、したがってパフォーマンスが向上するので、可能ならば、このサイズを与えてください。

## DD ステートメント

### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。

### SYSPRINT DD

このプログラムのためのメッセージ出力データ・セットを定義します。このデー

タ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=121 です。BLKSIZE を SYSPRINT DD ステートメントに指定する場合は、121 の倍数になっていなければなりません。

この DD ステートメントは必須です。

#### **SYSUDUMP DD**

このプログラムのためのダンプ・データ・セットを定義します。この DD ステートメントが必要になるのは、ABEND ユーティリティー制御ステートメントが含まれている場合だけです。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

#### **SYSOUT DD**

z/OS ソート/マージ・プログラムのためのメッセージ出力データ・セットを定義します。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

この DD ステートメントは必須です。

#### **SORTLIB DD**

オペレーティング・システムの z/OS ソート/マージ・プログラムのためのロード・モジュールが入っているデータ・セットを定義します。

この DD ステートメントは必須です。

#### **SORTWKnn DD**

オペレーティング・システム z/OS のソート/マージ・プログラムのための中間記憶データ・セットを定義します。

この DD ステートメントは必須です。

#### **SORTIN DD**

このプログラムのための入力データ・セットを定義します。これは、z/OS ソート/マージ・プログラムによって参照されるため、その JCL 要件に従っている必要があります。この DD ステートメントで参照されるデータ・セットは、データベースの初期ロード時、再ロード時、またはスキャン操作時に作られた DFSURWF1 データ・セットでなければなりません。これらの作業データ・セットを連結して、SORTIN データ・セットを作らなければなりません。DCB 属性が異なるデータ・セットが複数あるときは、LRECL が最も大きいデータ・セットを連結の先頭に置いてください。

このプログラムの中で指定されている DCB パラメーターは、RECFM=VB および LRECL=900 です。BLKSIZE は、データベースの初期ロード時、データベースの再ロード時、またはデータベースのスキャン時に作成された作業データ・セットに対して指定されているものと同じにしなければなりません。BLKSIZE がこれらのプログラムの DFSURWF1 DD ステートメントで指定されたものと同じであることを確かめてください。BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

この DD ステートメントは必須です。

#### **DFSURWF2 DD**

中間ソート作業データ・セットを定義します。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。このデータ・セットのサイズは、SORTIN DD ステートメントで定義された入力データ・セットのサイズとほぼ同じです。

このプログラムの中で指定されている DCB パラメーターは、RECFM=VB および LRECL=900 です。BLKSIZE は、この DD ステートメントで指定しなければなりません。BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

この DD ステートメントは必須です。

#### **DFSURWF3 DD**

このプログラムからのすべての出力データを入れておく出力作業データ・セットを定義します。このステートメントで定義された出力データ・セットは、接頭部更新ユーティリティーへの入力として渡されます。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。そのサイズは、SORTIN DD ステートメントで定義された入力データ・セットのサイズとほぼ同じです。

このプログラムの中で指定されている DCB パラメーターは、RECFM=VB および LRECL=900 です。BLKSIZE は、DFSURWF3 DD ステートメントで指定しなければなりません。BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

この DD ステートメントは必須です。

#### **DFSURCDS DD**

このプログラム用に生成されている制御データ・セットを定義します。これは、データベース事前再編成ユーティリティー (DFSURPRO) によって生成された出力制御データ・セットでなければなりません。

この DD ステートメントは必須です。

#### **DFSURIDX DD**

再編成またはロードされる DBD に副次索引が存在するとき使用される出力作業データ・セットを定義します。このデータ・セットは、副次索引 (共用または非共用) の作成、置き換え、マージ、あるいは抽出を行うときの HISAM アンロード・プログラム (DFSURUL0) への入力として使用しなければなりません。

このプログラムの中で指定されている DCB パラメーターは、RECFM=VB および LRECL=900 です。BLKSIZE は DFSURIDX DD ステートメントで指定しなければなりません。

この DD ステートメントが必要になるのは、副次索引が存在するときだけです。

関連資料:

437 ページの『第 31 章 ユーティリティー制御機能 (DFSUCF00)』

---

## **DFSURG10 ユーティリティーの例**

この例は、DFSURG10 ユーティリティーの使用法を示しています。



以下の例では、データベース接頭部解決ユーティリティを使用して、論理関係と副次索引を解決しています。3つの作業データ・セットだけが z/OS ソート/マージ・プログラムに渡され、ソート方式の選択はソート/マージ・プログラムに任せられます。SORTIN は、再ロード時または初期ロード時に、DD 名を DFSURWF1 として作成された作業データ・セットです。

DFSURWF2 は中間作業ファイルであり、このステップが終わると削除されます。

DFSURWF3 データ・セットが作成されます。これは、接頭部更新ユーティリティへの入力として使用されます。

DFSURIDX は出力データ・セットであり、ここには、HISAM アンロード/再ロード・ユーティリティを使用して副次索引を作るために必要なセグメントが書き込まれます。

図 2. データベース接頭部解決ユーティリティを使用した、論理関係および副次索引の解決

```
//PREFIXRES EXEC PGM=DFSURG10,REGION=100K,PARM='CKPT,MSG=AC,FILSZ=E5000'  
//SYSUDUMP DD SYSOUT=A  
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1200  
//SYSOUT DD SYSOUT=A  
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR,VOL=SER=SYSLIB,UNIT=SYSDA  
//SORTWK01 DD UNIT=SYSDA,SPACE=(1008,(60),,CONTIG)  
//SORTWK02 DD UNIT=SYSDA,SPACE=(1008,(60),,CONTIG)  
//SORTWK03 DD UNIT=SYSDA,SPACE=(1008,(60),,CONTIG)  
//SORTIN DD DSN=&&WF1,UNIT=SYSDA,DISP=(MOD,PASS)  
// DD DSN=&&WF1,UNIT=SYSDA,DISP=(MOD,PASS)  
//DFSURWF2 DD DSN=&&WF2,UNIT=SYSDA,SPACE=(1008,(30),,CONTIG),  
// DISP=(,PASS),DCB=(RECFM=VB,LRECL=900,BLKSIZE=1008)  
//DFSURWF3 DD DSN=&&WF3,UNIT=SYSDA,SPACE=(1008,(30),,CONTIG),  
// DISP=(,PASS),DCB=(RECFM=VB,LRECL=900,BLKSIZE=1008)  
//DFSURCDS DD DSN=*.LDJJK310.PREREORG.DFSURCDS,  
// UNIT=SYSDA,DISP=(OLD,PASS),  
// VOL=REF=*.LDJJK310.PREREORG.DFSURCDS  
//DBHVSAM1 DD DSN=DIVNTZ04.JJXS01K,DISP=SHR  
//DBHVSAM2 DD DSN=DIVNTZ04.JJXS01E,DISP=SHR  
//HIDAM DD DSN=DHONTZ04.JKXXI010,DISP=SHR  
//HIDAM2 DD DSN=DHONTZ04.JKXXI020,DISP=SHR  
//XDLBT04I DD DSN=DXINTZ04.JKXS01I,DISP=SHR  
/*
```



---

## 第 2 章 データベース接頭部更新ユーティリティ (DFSURGP0)

データベース接頭部更新ユーティリティ (DFSURGP0) は、データベースのロード、再編成、またはその両方によって接頭部情報が影響を受けた各セグメントの接頭部を更新するために使用します。

DFSURGP0 ユーティリティは、データベース接頭部解決ユーティリティ (DFSURG10) によって生成される出力を使用して、セグメント接頭部を更新します。

DFSURGP0 ユーティリティは以下の接頭部フィールドを更新します。

- 論理親ポインター・フィールド
- 論理兄弟ポインター・フィールド
- 論理子ポインター・フィールド
- 論理親と関連付けられたカウンター・フィールド

HALDB データベースはセグメント接頭部を更新する自己修復ポインター・プロセスを使用するので、ユーザーがセグメント接頭部を更新する必要はありません。

DFSURGP0 ユーティリティは異常終了から再始動できます。ただし、その異常終了がデータベースの入出力エラーによるものかどうかによって、再始動プロセスは異なります。

このユーティリティの機能は、ユーティリティ制御機能に実行させることもできます。

以下の図は、データベース接頭部更新ユーティリティのフロー・ダイアグラムです。

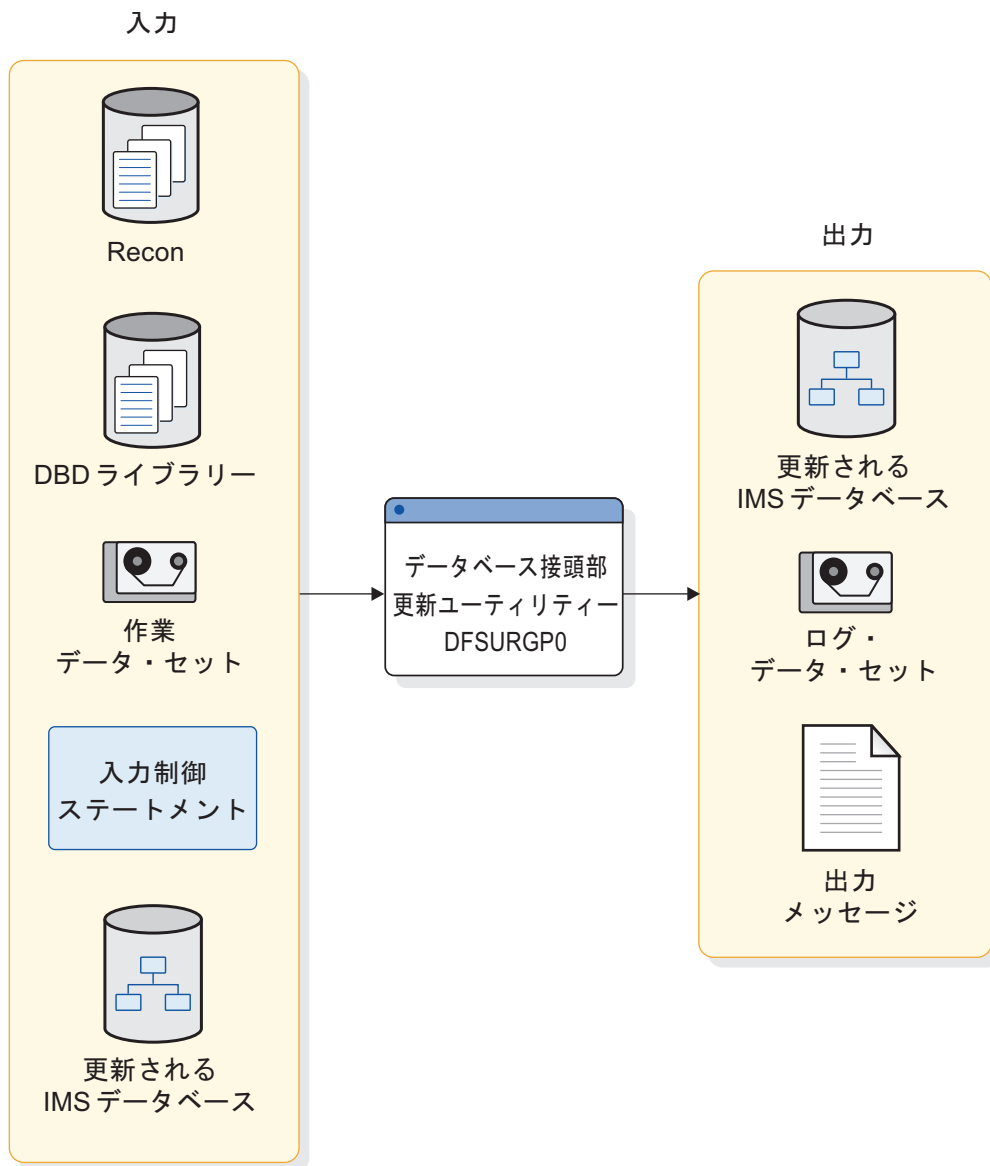


図 3. データベース接頭部更新ユーティリティ

サブセクション:

- 『制約事項』
- 17 ページの『前提条件』
- 17 ページの『要件』
- 17 ページの『推奨事項』
- 17 ページの『入出力』
- 18 ページの『戻りコード』
- 18 ページの『JCL 指定』

### 制約事項

DFSURGP0 ユーティリティは、全機能非 HALDB データベースのみで機能します。

データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。

## 前提条件

DFSURGP0 ユーティリティーを実行する前に、データベース接頭部解決ユーティリティー (DFSURG10) を実行して、DFSURGP0 ユーティリティー用の入力を生成しなければなりません。

## 要件

現在、DFSURGP0 ユーティリティーに対して文書化された要件はありません。

## 推奨事項

現在、DFSURGP0 ユーティリティーに対して文書化された推奨事項はありません。

## 入出力

DFSURGP0 ユーティリティーへの 1 次入力は、データベース接頭部解決ユーティリティー (DFSURG10) で作成される作業ファイルです。DFSURGP0 ユーティリティーの 1 次出力は、含まれるすべてのセグメント接頭部 (論理関係に使用される) が正しいデータベース・データ・セットです。

データベース接頭部更新ユーティリティーの入出力は、以下の表に示すとおりです。

表 2. データベース接頭部更新ユーティリティーの入出力

入力	出力
RECON	出力メッセージと統計
アンロードされたデータベース	再ロード出力 OSAM データ・セット
DBD ライブラリー	
入力制御ステートメント	

DFSURGP0 ユーティリティーへの入力は、論理関係接頭部情報が入っている各セグメントに適用される 1 つ以上の更新レコードです。更新レコードは、データベース接頭部解決ユーティリティーにより、データベースとセグメントの物理位置の順にソートされます。

JCL にログ DD ステートメント (IEFRDER、IEFRDER2) を含めると、DFSURGP0 ユーティリティーはオプションでログ出力データ・セットを作成できます。このログ出力を、データベースのリカバリーに使用することができます。このログ出力は、スキャンしたデータベースを更新するときに特に役立ちます。ただし、ログ出力を使用してバッチ・バックアウトを行うことはできません。DBRC が接頭部更新の実行時にアクティブであるのにログ DD ステートメントがない場合は、更新されたデータ・セットごとに DBRC NOTIFY.REORG が自動的に RECON データ・セットに記録されます。

DFSURGP0 ユーティリティーは、エラーなしで終了した場合、処理されたレコード数を示すプログラム正常終了メッセージを出します。

## 戻りコード

プログラム終了時に次の戻りコードが返されます。

コード 意味

- 0 エラーは検出されなかった
- 8 1 つ以上のエラー・メッセージが出された。メッセージはエラーを詳しく説明しており、システム出力の一部として印刷されます。

## JCL 指定

データベース接頭部更新ユーティリティーは、標準の z/OS ジョブとして実行されます。次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

**EXEC** ステートメント

EXEC ステートメントは次の形式になっていなければなりません。

```
//STEP EXEC PGM=DFSRRCO0,PARM='ULU,DFSURGP0'
```

SPIE、BUF といった通常の IMS 定位置パラメーターは、 PARM フィールドの中でプログラム名の後に置くことができます。

**DD** ステートメント

### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。STEPLIB DD ステートメントは必須です。

### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

DFSRESLB DD ステートメントは、無許可ライブラリーが IMS.SDFSRESL と連結される場合のみ必要です。

### IMS DD

ロード、再編成、またはその両方が行われたデータベースを記述する DBD が入っているライブラリーを定義します。このデータ・セットは、直接アクセス装置に置いておかなければなりません。

この DD ステートメントは必須です。

### SYSIN DD

入力制御ステートメントを入れるためのデータ・セットを定義します。このデータ・セットはテープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=80 です。 BLKSIZE は、この DD ステートメントで指定しなければなりません。 BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

DFSURGP0 ユーティリティーへの入力としてユーティリティー制御ステートメントを含める場合のみ SYSIN DD ステートメントが必要です。

#### **SYSPRINT DD**

メッセージ・データ・セットを定義します。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

この DD ステートメントは必須です。

プログラムにより与えられる DCB パラメーターは、RECFM=FB および LRECL=120 です。 BLKSIZE を指定するときは、120 の倍数になっていなければなりません。

#### **SNAPDD**

このプログラムのためのスナップ出力データ・セットを定義します。このステートメントが必要になるのは、SNAP 制御ステートメントが SYSIN データ・ストリームに指定されている場合だけです。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=120 です。 BLKSIZE はこの DD ステートメントで指定し、LRECL の倍数になっていなければなりません。 BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

#### **SYSUDUMP DD**

このプログラムのためのダンプ・データ・セットを定義します。この DD ステートメントが必要になるのは、ABEND ユーティリティー制御ステートメントが含まれている場合だけです。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

#### **DFSURWF3 DD**

このプログラムのための入力作業データ・セットを定義します。これは、接頭部解決ユーティリティーによって //DFSURWF3 DD ステートメントで生成された出力データ・セットでなければなりません。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。

この DD ステートメントは必須です。

#### **database DD**

初期ロード、再編成、あるいはスキャンが行われたデータベースを参照します。論理関係を持つデータベースのデータ・セット・グループごとに、DD ステートメントが 1 つ以上存在していなければなりません。DD 名は、DBD に指定されている DD 名と一致している必要があります。このユーティリティーで HIDAM データベースの操作を行う場合、その 1 次索引データベースについて DD ステートメントが用意されていなければなりません。

このデータ・セットは、直接アクセス装置に置いておかなければなりません。

### IEFRDER DD

接頭部更新時に作成されるシステム・ログ・データ・セットを記述します。このデータ・セットはテープに置いておくのが普通です。ただし、直接アドレス・ボリュームを使用することも可能です。この DD ステートメントはオプションです。このステートメントを使用するのは、ログ出力が必要な場合だけです。

### IEFRDER2 DD

接頭部更新時に作成される 2 次システム・ログ・データ・セットを記述します。このデータ・セットはテープに置いておくのが普通です。ただし、直接アドレス・ボリュームを使用することも可能です。この DD ステートメントはオプションです。このステートメントは、デュアル・ログ出力が必要な場合にのみ使用します。

### DFSVSAMP DD

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。

このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。

この DD ステートメントは必須です。

### DFSCTL DD

順次バッファリング (SB) の活動化を要求する SBPARM 制御ステートメントが入っているデータ・セットを記述します。

DFSCTL ファイルは順次データ・セットにすることも、PDS のメンバーにすることもできます。レコード形式は F、FB、または FBS のいずれかで、レコード長は 80 でなければなりません。このデータ・セットは、直接アクセス装置またはテープに置いておくことも、入力ストリームに入れて送ることもできます。

### RECON1 DD

最初の DBRC RECON データ・セットを定義します。

### RECON2 DD


2 番目の DBRC RECON データ・セットを定義します。

### RECON3 DD

RECON1 または RECON2 にエラーが見付かったとき使用される、オプションの DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。


関連概念:

 IMS バッファ・プール (システム定義)

関連資料:

437 ページの『第 31 章 ユーティリティー制御機能 (DFSUCF00)』

 順次バッファリング制御ステートメント (システム定義)

 DBBBATCH プロシージャ (システム定義)



## DFSURGP0 ユーティリティーの制御ステートメント

DFSURGP0 ユーティリティーでは、CHKPT、RSTRT、SNAP、および ABEND の 4 つのユーティリティー制御ステートメントを使用できます。

### CHKPT ステートメント



CHKPT ユーティリティー制御ステートメントは、このプログラムの実行中にユーティリティー・チェックポイント操作を行うことを指示します。データベース接頭部解決ユーティリティー (DFSURG10) は、DFSURGP0 ユーティリティーに対するサービスとして、DFSURWF3 データ・セット上にレコードを自動的に書き込み、これらのレコードはユーティリティー・チェックポイントとして使用されます。各ユーティリティー・チェックポイント・レコードが検出されると、z/OS システム・コンソールにチェックポイント・メッセージ (DFS867) が出されます。このメッセージは、このプログラムの名前、ユーティリティー・チェックポイント・レコードのチェックポイント番号、および処理されるボリュームのボリューム通し番号を示します。このユーティリティー・チェックポイント・メッセージは、再始動操作が必要な場合に備えて、保管しておいてください。

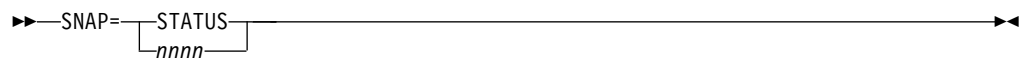
### RSTRT ステートメント



RSTRT ユーティリティー制御ステートメントは、再始動操作をこのプログラムに行わせることを指示します。nnnnn は 5 桁の 10 進数であり、volser は番号が nnnnn のチェックポイント・レコードが入っている入力ボリュームのボリューム通し番号です。nnnnn パラメーターおよび volser パラメーターは、z/OS システム・コンソールに出されたチェックポイント・メッセージから入手します。

この制御ステートメントに指定されたボリュームがマウントされていないと、正しいボリュームが使用可能になるまで FEOV が出されます。再始動が完了すると、メッセージ (DFS378) が出され、チェックポイント番号、ボリューム通し番号、およびプログラム名を示します。処理は再始動点から続行されます。

### SNAP ステートメント



SNAP オプションのユーティリティー制御ステートメントは、診断情報を得るために IMS 制御ブロックを SNAPDD データ・セットに印刷することを指示します。

STATUS パラメーターを指定すると、異常状況コードが DL/I から返されるか、あるいは異常戻りコードがバッファ・ハンドラーから返されたときに、スナップがとられます。

*nnnn* はバッファ・ハンドラーの呼び出し回数であり、*nnnn* 回の呼び出しが行われた後にスナップがとられます。 *nnnn* を指定した SNAP ステートメントは、25 個まで受け付けられます。 *nnnn* は 1 から 9999999 までの範囲で指定し、途中で空白やコンマを入れてはなりません。有効数字が必要ですが、等号と相対呼び出し番号の 1 桁目との間に空白を入れることはできません。

バッファ・ハンドラー呼び出しと DFSURWF3 データ・セットからのレコードとの間には、1 対 1 の相関関係があります。

## ABEND ステートメント

ABEND ユーティリティ制御ステートメントは、診断の目的のためにストレージ・ダンプが必要であることを示します。

▶—ABEND—▶

入力ストリームにこのステートメントを入れた場合、このユーティリティの実行中に異常条件が起これば、戻りコードがゼロより大きいときにユーザー ABEND 955 が出されます。この ABEND はプログラム実行の終了時に出力され、そのときストレージ・ダンプがとられます。この制御ステートメントを使用するときは、SYSUDUMP DD ステートメントも必要です。

---

## DFSURGP0 ユーティリティの例

ここでの例は、DFSURGP0 ユーティリティのサンプル JCL を示しています。

この節の例で、使用する DBRC で動的割り振りを行わない場合は、次の DD ステートメントをサンプル JCL に追加する必要があります。

```
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
```

サブセクション:

- 『データベースの更新』
- 23 ページの『論理関係がある 2 つのデータベースの再編成』

### データベースの更新

以下の図は、DBHVSAM1、DBHVSAM2、HIDAM、HIDAM2、および XDLBT04I の各 DD ステートメントで定義した 5 つのデータベースを更新するために DFSURGP0 を実行するときに必要な JCL を示しています。

図 4. 5 つのデータベースを更新する例

```
//PREFIXUP EXEC PGM=DFSRR00,PARM='ULU,DFSURGP0'
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
// DD DSN=IMS.DBDLIB,DISP=SHR
```

```

//IEFRDER DD DSN=DBRCIMS.LOG3,DISP=(,KEEP),VOL=SER=USER02,UNIT=SYSDA,
//        SPACE=(CYL,(1,1)),DCB=BLKSIZE=4096
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DFSURWF3 DD DSN=&&WF3,UNIT=SYSDA,DISP=(OLD,PASS)
//DBHVSAM1 DD DSN=DIVNTZ04.JJXS01K,DISP=SHR
//DBHVSAM2 DD DSN=DIVNTZ04.JJXS01E,DISP=SHR
//HIDAM DD DSN=DHONTZ04.JKXXI010,DISP=SHR
//HIDAM2 DD DSN=DHONTZ04.JKXXI020,DISP=SHR
//XDLBT04I DD DSN=DXINTZ04.JKXS01I,DISP=SHR
//DFSVSAMP DD *
2048,4
IOBF=(8192,4)
/*

```

## 論理関係がある 2 つのデータベースの再編成

以下の図は、論理関係がある 2 つのデータベースを再編成する例です。2 つのデータベースとは DIVNTZ02 (HISAM VSAM データベース) と DHVNTZ02 (HIDAM VSAM データベース) であり、これらには索引 VSAM データベース (DXVNTZ02) が付随します。

図 5. 論理関係がある 2 つのデータベースを再編成する例

```

//JOB LIB DD DSN=IMS.SDFSRESL,DISP=SHR
/*
//*****
/* -DBDNAME- -DDNAME- -DSNAME- -ACCESS-
/* DIVNTZ02 DBHVSAM1 JDSG1RC HISAM VSAM (PRIME)
/* " DBHVSAM2 JDSG1RC0 HISAM VSAM (OVERFLOW)
/* DHVNTZ02 HIDAM KDSG1RC HIDAM VSAM (GROUP1)
/* " HIDAM2 KDSG2RC HIDAM VSAM (GROUP2)
/* DXVNTZ02 XDLBT04I KINDXRC INDEX VSAM
//*****
/*
//*****
/* PREREORGANIZATION
//*****
/*
//PREREORG EXEC PGM=DFSRR00,
// PARM='ULU,DFSURPR0,,,1,,,,,,,Y,N'
//IMS DD DISP=SHR,DSN=IMS.DBDLIB
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DFSURCDS DD DSN=&&URCDS,;
// UNIT=SYSDA,
// DISP=(,PASS,DELETE),
// SPACE=(TRK,(10,10),RLSE),
// DCB=(BLKSIZE=1600)
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//SYSIN DD *
OPTIONS=(NOPUNCH,STAT,SUMM)
DBR=DIVNTZ02
DBR=DHVNTZ02
/*
//*****
/* UNLOAD THE HIDAM DATABASE - DHVNTZ02
//*****
/*
//UNLOAD1 EXEC PGM=DFSRR00,REGION=1024K,COND=(1,LT),
// PARM='ULU,DFSURGU0,DHVNTZ02,,1,,,,,,,Y,N'
//IMS DD DSN=IMS.DBDLIB, DISP=SHR

```

```

//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DFSURCDS DD DSN=&&URCDS,;
//          UNIT=SYSDA,DISP=(OLD,PASS)
//HIDAM    DD DSN=KDSG1RC,DISP=SHR
//HIDAM2   DD DSN=KDSG2RC,DISP=SHR
//XDLBT04I DD DSN=KINDXRC,DISP=SHR
//DFSURGU1 DD DSN=&&ULD1A,DISP=(NEW,PASS,DELETE),UNIT=SYSDA,
//          SPACE=(CYL,(5,1))
//RECON1   DD DSN=IMS.RECON1,DISP=SHR
//RECON2   DD DSN=IMS.RECON2,DISP=SHR
//RECON3   DD DSN=IMS.RECON3,DISP=SHR
//*****
//*        UNLOAD THE HISAM DATABASE - DIVNTZ02
//*****
//*
//UNLOAD2  EXEC PGM=DFSRR00,COND=(1,LT),
//          PARM='ULU,DFSURGU0,DIVNTZ02,,1,,,,,,Y,N'

//IMS      DD DISP=SHR,DSN=IMS.DBDLIB
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DFSURCDS DD DSN=&&URCDS,;
//          UNIT=SYSDA,DISP=(OLD,PASS)
//DBHVSAM1 DD DSN=JDSG1RC,DISP=SHR
//DBHVSAM2 DD DSN=JDSG1RC0,DISP=SHR
//HIDAM    DD DSN=KDSG1RC,DISP=SHR
//HIDAM2   DD DSN=KDSG2RC,DISP=SHR
//XDLBT04I DD DSN=KINDXRC,DISP=SHR
//DFSURGU1 DD DSN=&&ULD1B,DISP=(NEW,PASS,DELETE),UNIT=SYSDA,
//          SPACE=(CYL,(5,1))
//RECON1   DD DSN=IMS.RECON1,DISP=SHR
//RECON2   DD DSN=IMS.RECON2,DISP=SHR
//RECON3   DD DSN=IMS.RECON3,DISP=SHR
//DFSVSAMP DD *
512,10
1024,10
2048,10
4096,10
IOBF=(4096,5)
/*
/*
//*****
//*        SCRATCH AND REALLOCATE THE VSAM DATABASES
//*****
//*
//SCRATCH  EXEC PGM=IDCAMS,COND=(1,LT)
//SYSPRINT DD SYSOUT=*
//VSAMDD   DD UNIT=SYSDA,DISP=SHR,VOL=SER=RECRES
//SYSIN    DD *
DELETE JDSG1RC PURGE FILE(VSAMDD)
DELETE JDSG1RC0 PURGE FILE(VSAMDD)
DELETE KDSG1RC PURGE FILE(VSAMDD)
DELETE KDSG2RC PURGE FILE(VSAMDD)
DELETE KINDXRC PURGE FILE(VSAMDD)
DEFINE CLUSTER (NAME (JDSG1RC) -
                CYLINDERS (2,1) -
                VOL (RECRES) -
                FREESPACE (30,20) -
                SHAREOPTIONS (3,3) -
                RECSZ (200,200) -
                KEYS (5,6) -
                UNIQUE SPEED) -
                DATA (NAME(JDSG1RC1) -
                CISZ (1024)) -
                INDEX (NAME(JDSG1RC2) -
                CISZ (1024)) -
                CATALOG (VCATREC)

```

```

DEFINE CLUSTER (NAME (JDSG1RCO) -
    CYLINDERS (1,1) -
    VOL (RECRS) -
    SHAREOPTIONS (3,3) -
    RECSZ (200,200) -
    NIXD -
    UNIQUE) -
DATA (NAME(JDSG1RC3) -
    CISZ (512)) -
CATALOG (VCATREC)

DEFINE CLUSTER (NAME (KDSG1RC) -
    CYLINDERS (2,1) -
    VOL (RECRS) -
    SHAREOPTIONS (3,3) -
    RECSZ (2041,2041) -
    NIXD -
    UNIQUE) -
DATA (NAME(KDSG1RC1) -
    CISZ (2048)) -
CATALOG (VCATREC)

DEFINE CLUSTER (NAME (KDSG2RC) -
    CYLINDERS (1,1) -
    VOL (RECRS) -
    SHAREOPTIONS (3,3) -
    RECSZ (505,505) -
    NIXD -
    UNIQUE) -
DATA (NAME(KDSG2RC1) -
    CISZ (512)) -
CATALOG (VCATREC)

DEFINE CLUSTER (NAME (KINDXRC) -
    CYLINDERS (1,1) -
    VOL (RECRS) -
    FREESPACE (30,20) -
    SHAREOPTIONS (3,3) -
    KEYS (5,5) -
    RECSZ (12,12) -
    SPEED -
    UNIQUE) -
DATA (NAME(KINDXRC1) -
    CISZ (512)) -
INDEX (NAME(KINDXRC2) -
    CISZ (1024)) -
CATALOG (VCATREC)

//DFSVSAMP DD *
512,10
1024,10
2048,10
4096,10
IOBF=(4096,5)
/*
/**
/*******
/**      RELOAD THE HIDAM DATABASE - DHVNTZ02
/*******
/**
//RELOAD1 EXEC PGM=DFSRR00,
//      PARM='ULU,DFSURGL0,DHVNTZ02,,1,,,,,,,,,Y,N'
//IMS      DD DISP=SHR,DSN=IMS.DBDLIB
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DFSURCDS DD DSN=&&URCDS,UNIT=SYSDA,DISP=(OLD,PASS)
//DFSUINPT DD DSN=&&ULD1A,DISP=(OLD,DELETE),UNIT=SYSDA

//DFSURWF1 DD DSN=&&WF1A,UNIT=SYSDA,DISP=(,PASS),SPACE=(CYL,(2,1)),
//      DCB=(RECFM=VB,LRECL=900,BLKSIZE=1008)
//HIDAM      DD DSN=KDSG1RC,DISP=SHR

```

```

//HIDAM2 DD DSN=KDSG2RC,DISP=SHR
//XDLBT04I DD DSN=KINDXRC,DISP=SHR
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//*
//*****
//* RELOAD THE HISAM DATABASE - DIVNTZ02
//*****
//*
//RELOAD2 EXEC PGM=DFSRR00,
// PARM='ULU,DFSURGL0,DIVNTZ02,,1,,,,,,,,,Y,N'
//IMS DD DISP=SHR,DSN=IMS.DBDLIB
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DFSURCDS DD DSN=&&URCDS,UNIT=SYSDA,DISP=(OLD,PASS)
//DFSUINPT DD DSN=&&ULD1B,DISP=(OLD,DELETE),UNIT=SYSDA
//DFSURWF1 DD DSN=&&WF1B,UNIT=SYSDA,DISP=(,PASS),SPACE=(CYL,(2,1)),
// DCB=(RECFM=VB,LRECL=900,BLKSIZE=1008)
//DBHVSAM1 DD DSN=JDSG1RC,DISP=SHR
//DBHVSAM2 DD DSN=JDSG1RC0,DISP=SHR
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//DFSVSAMP DD *
512,10
1024,10
2048,10
4096,10
IOBF=(4096,5)
//*
//*****
//* PREFIX RESOLUTION
//*****
//*
//PREFIX EXEC PGM=DFSURG10
//IMS DD DISP=SHR,DSN=IMS.DBDLIB
//SYSPRINT DD SYSOUT=A
//SYSOUT DD SYSOUT=A
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,2,,CONTIG)
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,2,,CONTIG)
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,2,,CONTIG)
//SORTIN DD DSN=&&WF1A,UNIT=SYSDA,DISP=(OLD,DELETE)
// DD DSN=&&WF1B,UNIT=SYSDA,DISP=(OLD,DELETE)
//DFSURWF2 DD DSN=&&WF2,
// UNIT=SYSDA,
// DISP=(,DELETE),
// SPACE=(CYL,(2,2)),
// DCB=BLKSIZE=1008
//DFSURWF3 DD DSN=&&WF3,
// DISP=(,PASS),
// UNIT=SYSDA,
// SPACE=(CYL,(1,1),RLSE),
// DCB=(RECFM=VB,LRECL=900,BLKSIZE=13030,BUFNO=8)

//DFSURCDS DD DSN=&&URCDS,UNIT=SYSDA,DISP=(OLD,PASS)
//DFSURIDX DD DUMMY,DCB=BLKSIZE=1008
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//*
//*****
//* PREFIX UPDATE
//*****
//*
//UPDATE EXEC PGM=DFSRR00,REGION=1024K,
// PARM='ULU,DFSURGP0,,1,,,,,,,,,Y,N'

```

```

//IMS      DD DISP=SHR,DSN=IMS.DBDLIB
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DFSURWF3 DD DSN=&&WF3,
//          DISP=(OLD,DELETE),
//          UNIT=SYSDA
//DBHVSAM1 DD DSN=JDSG1RC,DISP=SHR
//DBHVSAM2 DD DSN=JDSG1RC,DISP=SHR
//HIDAM    DD DSN=KDSG1RC,DISP=SHR
//HIDAM2   DD DSN=KDSG2RC,DISP=SHR
//XDLBT04I DD DSN=KINDXRC,DISP=SHR
//RECON1   DD DSN=IMS.RECON1,DISP=SHR
//RECON2   DD DSN=IMS.RECON2,DISP=SHR
//RECON3   DD DSN=IMS.RECON3,DISP=SHR
//DFSVSAMP DD *
512,10
1024,10
2048,10
4096,10
IOBF=(4096,5)
/*
//DFSCCTL DD *
SBPARM ACTIV=COND,DB=SKILLDB,BUFSETS=6
/*
SB CONTROL STATEMENT
//SYSIN DD *
SNAP=STATUS
/*
//

```

---

## DFSURGP0 ユーティリティーの異常終了後の再始動

DFSURGP0 ユーティリティーが、データベースの入出力エラー以外の理由で異常終了したときは、再始動アクションを要求するか、あるいは元の入力作業データ・セットを使用してこのステップを実行し直すことによって、プログラム実行を再開することができます。

データベースの入出力エラーが原因で DFSURGP0 ユーティリティーが異常終了したときは、次の操作を実行してください。

1. 入出力エラーの原因を判別して修正する。
2. データベースをこのプログラムの実行前の状態に復元する。
3. 再始動操作を要求するか、または、元の入力作業データ・セットを使用してプログラムを再実行する。





---

## 第 3 章 データベース・スキャン・ユーティリティ (DFSURGS0)

データベース・スキャン・ユーティリティ (DFSURGS0) は、ロードまたは再編成中でない非 HALDB データベースをスキャンするために使用します。

このユーティリティの実行中に、他のデータベースのロード、再編成、またはその両方が行われると、その影響を受ける論理関係を含んでいるセグメントを識別します。影響を受けるセグメントごとに、このユーティリティは、そのセグメントが関与している関係に応じて、1 つ以上の出力レコードを生成します。

DFSURGS0 ユーティリティは DFSURWF1 出力作業データ・セットにレコードを書き込みます。DFSURWF1 作業データ・セットは、データベース接頭部解決ユーティリティ (DFSURG10) への入力として使用されます。

DFSURGS0 ユーティリティは異常終了後に再始動できます。ただし、その異常終了がデータベースの入出力エラーによるものかどうかによって、再始動プロセスは異なります。

このユーティリティの機能は、ユーティリティ制御機能に実行させることもできます。

以下の図は、データベース・スキャン・ユーティリティのフロー・ダイアグラムです。

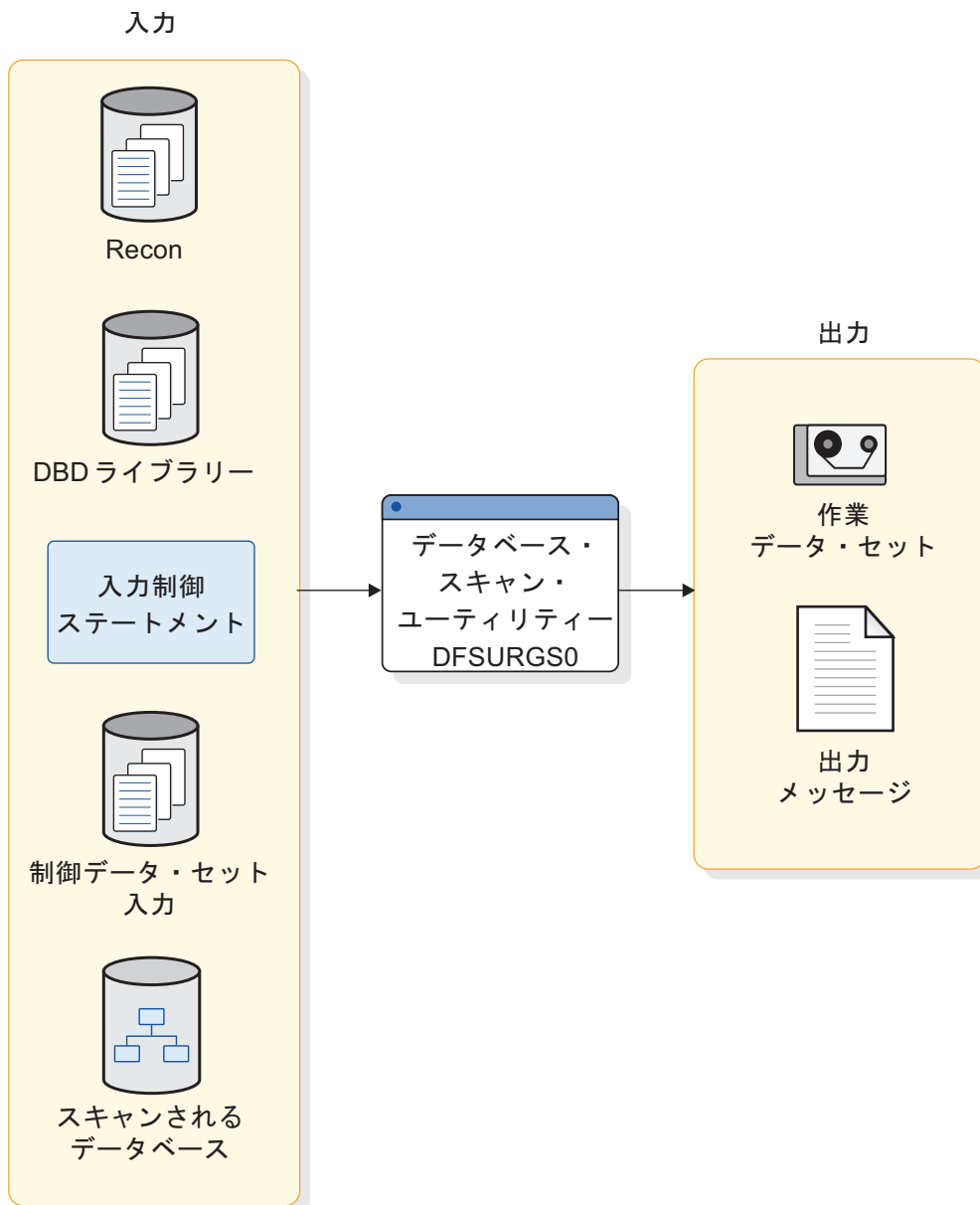


図 6. データベース・スキャン・ユーティリティのフロー・ダイアグラム

サブセクション:

- 31 ページの『制約事項』
- 31 ページの『前提条件』
- 31 ページの『要件』
- 31 ページの『推奨事項』
- 31 ページの『入出力』
- 31 ページの『戻りコード』
- 32 ページの『JCL 指定』

## 制約事項

DFSURGS0 ユーティリティは、全機能非 HALDB データベースのみで機能します。

データベースを変更するユーティリティは、データベースが静止している間に行うことはできません。

## 前提条件

現在、DFSURGS0 ユーティリティに対して文書化された前提条件はありません。

## 要件

現在、DFSURGS0 ユーティリティに対して文書化された要件はありません。

## 推奨事項

現在、DFSURGS0 ユーティリティに対して文書化された推奨事項はありません。

## 入出力

DFSURGS0 ユーティリティへの 1 次入力、スキャンされる 1 つ以上のデータベースです。DFSURGS0 ユーティリティは、他のオプション入力および必須入力を受け付けます。

DFSURGS0 ユーティリティの 1 次出力は作業データ・セットです。DFSURGS0 ユーティリティは出力メッセージも作成します。

データベース・スキャン・ユーティリティの入出力は、以下の表に示すとおりです。

表 3. データベース・スキャン・ユーティリティの入出力

入力	出力
RECON	作業データ・セット
DBD ライブラリー	出力メッセージ
入力制御ステートメント	
制御データ・セット	
スキャンされるデータベース	

## 戻りコード

プログラム終了時に次の戻りコードが返されます。

コード 意味

- 0 エラーは検出されなかった
- 8 1 つ以上のエラー・メッセージが出された。

## JCL 指定

DFSURGS0 ユーティリティは、標準の z/OS ジョブとして実行されます。DFSURGS0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含めることが必須であり可能です。

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

EXEC ステートメントは次の形式になっていなければなりません。

```
PGM=DFSRR00,PARM='ULU,DFSURGS0'
```

SPIE、BUF といった通常の IMS 定位置パラメーターは、PARM フィールドの中でプログラム名の後に置くことができます。

データベース・スキャン・ユーティリティには、このステートメントでバッファ・サイズ・パラメーターを渡すことができます。バッファ・サイズは、データベースのブロック・サイズが、2 つのブロックをストレージに置いておくのに 7 KB を超える容量を必要とするような場合に最も役立ちます。そのようにしておくと、一方のブロックで順次 GET を行っている間に、もう一方のブロックをストレージ装置から読み込むことができます。

### DD ステートメント

#### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

#### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

DFSRESLB DD ステートメントは、無許可ライブラリーが IMS.SDFSRESL と連結される場合のみ必要です。

#### IMS DD

スキャンするデータベースを記述し、そのほかに論理的に関係のあるデータベースがあれば、そのデータベースも記述している DBD が入っているライブラリーを定義します。このデータ・セットは、直接アクセス装置に置いておかなければなりません。

この DD ステートメントは必須です。

#### SYSIN DD

このプログラムのための入力データ・セットを定義します。このデータ・セットはテープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて

送ることもできます。この DD ステートメントが必要になるのは、このプログラムへの入力としてユーティリティー制御ステートメントが指定される場合だけです。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=80 です。BLKSIZE は、この DD ステートメントで指定しなければなりません。BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

#### **SYSPRINT DD**

メッセージ出力データ・セットを定義します。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=120 です。BLKSIZE を指定するときは、120 の倍数になっていなければなりません。

この DD ステートメントは必須です。

#### **SYSUDUMP DD**

このプログラムのためのダンプ・データ・セットを定義します。この DD ステートメントが必要になるのは、ABEND ユーティリティー制御ステートメントが含まれている場合だけです。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

#### **DFSURCDS DD**

このプログラムのための制御データ・セットを定義します。これは、データベース事前再編成ユーティリティーによって生成された出力制御データ・セットでなければなりません。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておかなければなりません。

この DD ステートメントは必須です。

#### **DFSURSRT DD**

再始動の目的で使用するデータ・セットを定義します。再始動が必要でなければ、このデータ・セットは必要ありません。

再始動処理を行うときは、DFSURSRT DD の連結が、前回のスキャンで得られた DFSURWF1 DD の連結と同じであることを確かめてください。ただし、RSTRT コマンドで指定したレコードが入っているデータ・セットを用いて DFSURSRT の連結を始めることもできます。前回のスキャンの実行で出された SYSPRINT に入っているチェックポイント情報を調べて、それがどのデータ・セットであるか判断してください。

元の DFSURWF1 が単一データ・セットであったときは、DFSURSRT DD ステートメントでそのデータ・セットを指定してください。

#### **database DD**

データベース事前再編成ユーティリティーの指示に従ってスキャンすべきデータベースを参照します。この DD ステートメントは、各データベースに必要です。DD 名は、DBD に指定されている DD 名と一致している必要があります。このデータ・セットは、直接アクセス装置に置いておかなければなりません。

#### DFSURWF1 DD

このプログラムのための入出力作業データ・セットを定義します。このデータ・セットは、データベース・スキャン・ユーティリティーの出力を受け入れます。また、接頭部解決ユーティリティーへの SORTIN データ・セットを構成する連結の一部となります。このデータ・セットは、接頭部解決ユーティリティーへの入力の 1 つとなり、また初期ロードとデータベース・スキャン・ユーティリティーの出力となります。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。

このプログラムの中で指定されている DCB パラメーターは、RECFM=VB および LRECL=900 です。BLKSIZE は、この DD ステートメントで指定しなければなりません。BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

#### DFSVSAMP DD

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。

このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。

この DD ステートメントは必須です。

#### DFSCTL DD

順次バッファリング (SB) の活動化を要求する SBPARM 制御ステートメントが入っているデータ・セットを記述します。

DFSCTL ファイルは順次データ・セットにすることも、PDS のメンバーにすることもできます。レコード形式は F、FB、または FBS のいずれかで、レコード長は 80 でなければなりません。このデータ・セットは、直接アクセス装置またはテープに置いておくことも、入力ストリームに入れて送ることもできます。この DD ステートメントはオプションです。

#### RECON1 DD

最初の DBRC RECON データ・セットを定義します。

#### RECON2 DD


2 番目の DBRC RECON データ・セットを定義します。

#### RECON3 DD

RECON1 または RECON2 にエラーが見つかったとき使用される、オプションの DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。


動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

関連概念:

 IMS バッファ・プール (システム定義)

関連資料:

437 ページの『第 31 章 ユーティリティー制御機能 (DFSUCF00)』

 順次バッファリング制御ステートメント (システム定義)

➡ DBBBATCH プロシージャ (システム定義)

➡ DLIBATCH プロシージャ (システム定義)

## DFSURGS0 ユーティリティーの制御ステートメント

DFSURGS0 ユーティリティーでは、DBS、CHKPT、RSTRT、および ABEND の 4 つのユーティリティー制御ステートメントを使用できます。

### DBS ステートメント

DBS ユーティリティー制御ステートメントは、データベース・スキャン・ユーティリティーにスキャンされるデータベース・セグメントを指定します。

➡—DBS=*database-name,segment-name*

,SEQ
,SEG

➡

このステートメントは、1 つ以上指定することができます。データベース名とセグメント名は、それぞれ全体の長さが 8 文字になるように、空白で埋める必要があります。ユーザーのコメントは、パラメーターの指定の後に入れることができます。

DBS 制御ステートメントが与えられていないと、データベース事前再編成ユーティリティーからの制御データ・セットに入っているスキャン情報が使用され、制御データ・セットに入っているデータベース名とセグメント名だけが受け付けられます。(スキャン・リストで指定されたデータベースはいずれも、接頭部解決ユーティリティーを実行する前にスキャンしておかなければなりません。)

DBS 制御ステートメントが与えられていないとき、データベースは次のようにスキャンされます。

- HISAM データベースのときは、非修飾 GN 呼び出しを使用して順次にスキャンされます。
- HDAM データベースまたは HIDAM データベースのときは、セグメント名で修飾された GN 呼び出しを使用して、セグメント単位でスキャンされます。

DBS 制御ステートメントがデータベース・スキャン・ユーティリティーに与えられたときは、制御データ・セットに入っているスキャン・リストは完全に無視され、作業データ・セット・レコードは、DBS ステートメントで指定されたセグメントについてだけ生成されます。これらのセグメント名は、当然、制御データ・セットに存在していなければなりません。存在しない場合は、DBS 制御ステートメントも無視されます。

スキャン・リストが DBS 制御ステートメントで与えられている場合であっても、制御データ・セットには、データベース・スキャン・ユーティリティーが必要とする他の情報が入っているので、この制御データ・セットをこのユーティリティーに与える必要があります。

データベース事前再編成ユーティリティーの SYSPUNCH 出力データ・セットに入っているスキャン・リストは、データベース・スキャン・ユーティリティーへの入

力として使用できます。SYSPUNCH 出力をこのユーティリティーへの入力として使用することの価値は、複数のデータベースをスキャンする必要があるとき、データベース・スキャン・ユーティリティーを複数実行することによって並行してデータベースのスキャンができる点にあります。これは、SYSPUNCH 出力 (DBS= ステートメント) をデータベース名で分けて、このユーティリティーの個別の実行に対して各データベースのスキャン制御ステートメントを渡すことによって行うことができます。

SEQ オプションと SEG オプションは、データベースをスキャンするとき使用する方式を指定します。SEQ オプションを指定すると、データベースは非修飾 GN 呼び出しを使用して順次にスキャンされます。SEG オプションを指定すると、データベースはセグメント名で修飾された GN (Get Next) 呼び出しを使用してスキャンされます。DBS 制御ステートメントで指定されたスキャン方式オプションは、その制御ステートメントで指定された特定のセグメントだけでなく、制御ステートメントで指定されたデータベース内のスキャン対象のすべてのセグメントに適用されます。

スキャン方式オプションが特定のデータベースについて複数の DBS 制御ステートメントで指定されたときは、そのデータベースについて最後に現れた DBS 制御ステートメントで指定された方式がデータベース全体に使用されます。どちらのオプションも指定されない場合、HISAM データベースのときは SEQ オプションが指定されたものと見なされ、HDAM または HIDAM データベースのときは SEG オプションが指定されたものと見なされます。

HD データベースのスキャン効率は、スキャンするセグメント・タイプが多数のときに SEQ オプションを指定すると向上します。逆に、HISAM データベースのスキャン効率は、スキャンするセグメントが少ないときに SEG オプションを指定すると向上します。どちらのスキャン方式が効率的であるかは、スキャンするデータベースのそれぞれの構造により決まります。最良の方式を用途別に判断しなければならない場合もあります。

## CHKPT ステートメント

CHKPT ユーティリティー制御ステートメントは、このプログラムの実行中にユーティリティー・チェックポイント操作を行うことを指示します。

▶▶CHKPT=

NO
nnnnn

▶▶

ユーティリティー・チェックポイント・レコードは、nnnnn で指定された数のレコードが生成されるたびに、DFSURWF1 DD ステートメントで指定されたデータ・セットに書き込まれます。各ユーティリティー・チェックポイント・レコードが生成されると、z/OS システム・コンソールにユーティリティー・チェックポイント・メッセージ (DFS867I) が出力されます。このメッセージは、このプログラムの名前、書き込まれたユーティリティー・チェックポイント・レコードのチェックポイント番号、およびチェックポイント・レコードが書き込まれるボリュームの出力ボリューム通し番号を示します。このユーティリティー・チェックポイント・メッセージは、再始動操作が必要な場合に備えて、保管しておいてください。



## RSTRT ステートメント

RSTRT ユーティリティー制御ステートメントは、再始動操作をこのプログラムに行わせることを指示します。

```
➡➡—RSTRT= 

|              |
|--------------|
| NO           |
| nnnnn,volser |

—————➡➡
```

*nnnnn* は 5 桁の 10 進数であり、*volser* は番号が *nnnnn* のチェックポイント・レコードが入っている入力ボリュームのボリューム通し番号です。 *nnnnn* パラメーターおよび *volser* パラメーターは、z/OS システム・コンソールに出されたチェックポイント・メッセージから入手します。

このステートメントに指定されたボリュームがマウントされていないと、正しいボリュームが使用可能になるまで FEOV が出されます。このステートメントで指定されたボリュームは、DFSURSRT DD ステートメントによって指定されていなければなりません。再始動モジュールは、この DD ステートメントで指定されたボリュームに入っているレコードを、番号が *nnnnn* のチェックポイント・レコードが現れるまで読み取っていきます。

読み取られた各レコードは、DFSURWF1 DD z/OS システム・コンソールで識別されたデータ・セットに書き込まれて、このプログラム名、チェックポイント番号、およびボリューム通し番号を示します。「RSTRT」制御ステートメントで指定されたチェックポイント・レコードは DFSURWF1 ステートメントにより指定されたデータ・セットに書き込まれているため、そのデータ・セットで使用可能な現行ボリュームが、再始動完了メッセージに示されます。

処理は再始動点から続行されます。指定されたチェックポイント・レコードが入っているボリュームと、このプログラムの前回の実行時に書き込みのあった後続のボリュームは、接頭部解決ユーティリティー (DFSURG10) に渡されるデータ・セットに含まれてはなりません。

## ABEND ステートメント

ABEND ユーティリティー制御ステートメントは、診断の目的のためにストレージ・ダンプを必要とするときに使用してください。

```
➡➡—ABEND—————➡➡
```

入力ストリームにこのユーティリティー制御ステートメントを入れた場合、このユーティリティーの実行中に異常条件が起これば、戻りコードがゼロより大きいときに ABEND U0955 が出されます。この ABEND はプログラム実行の終了時に出され、そのときストレージ・ダンプがとられます。この制御ステートメントを使用するときは、SYSUDUMP DD ステートメントも必要です。

---

## DFSURGS0 ユーティリティーの例

ここでの例は、DFSURGS0 ユーティリティーのサンプル JCL を示しています。

この節の例で、使用する DBRC で動的割り振りを行わない場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

図 7. 動的割り振りなしで DBRC を使用するための DD ステートメント

```
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
```

以下の図は、HDRELTD DD ステートメントで定義されたデータベースをスキャンするとき必要になる JCL を示しています。このデータベースは、データベース事前再編成ユーティリティーに渡された DBIL または DBR 制御ステートメントでユーザーが指定した他の 1 つ以上のデータベースに論理的に関係付けられています。制御ステートメントではなく、データベース事前再編成ユーティリティーからの制御データ・セットに入っている情報が使用されます。

図 8. HDRELTD DD ステートメントによって定義されたデータベースのスキャン

```
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSURGS0'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1200
//DFSURWF1 DD DSN=IMS.URWF1,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=TAPE11,LABEL=(,SL),
// DCB=(LRECL=300,BLKSIZE=1008,RECFM=VB)
//HDRELTD DD DSN=DATABASE.DBRELATD,DISP=OLD,
// UNIT=SYSDA,VOL=SER=DB0003,
//DFSURCDS DD DSN=IMS.RLCDS,DISP=OLD,
// UNIT=SYSDA,VOL=SER=IMSMSC
//DFSVSAMP DD input for VSAM and OSAM buffers and options
```

論理関係を持つすべてのデータベースについて DD ステートメントが必要です。

---

## DFSURGS0 ユーティリティーの異常終了後の再始動

データベース・スキャン・ユーティリティーの実行が、データベースの入出力エラー以外の理由で異常終了した場合には、プログラムの実行は再始動操作で再開できません。

このユーティリティーの実行が、データベースの入出力エラーが原因で異常終了したときは、次の操作を実行してください。

1. エラー原因を突き止め、その訂正のために必要な処置を行う。
2. データベースをこのユーティリティーの実行以前の状態に復元する。
3. 再始動操作を要求する。

---

## 第 4 章 データベース調査ユーティリティー (DFSPRSUR)

データベース調査ユーティリティー (DFSPRSUR) は、HDAM または HIDAM データベースの全体または一部をスキャンして、データベースの物理編成を記述する報告書を作成するために使用します。

この報告書は、再編成の必要があるかどうかを判断する上で役立ちます。また、データベース調査ユーティリティーは、データベース部分再編成のときに再編成されたレコードを入れることができるフリー・スペース域のサイズと位置も示します。

DFSPRSUR ユーティリティーは、オンライン・データベースに対するバッチ・メッセージ処理 (BMP) プログラムとして、またはバッチ・プログラムとして実行することができます。

サブセクション:

- 『制約事項』
- 『前提条件』
- 『要件』
- 『推奨事項』
- 40 ページの『入出力』
- 40 ページの『戻りコード』
- 40 ページの『JCL 指定』

### 制約事項

DFSPRSUR ユーティリティーは、全機能非 HALDB データベースのみで機能します。

データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。

### 前提条件

現在、DFSPRSUR ユーティリティーに対して文書化された前提条件はありません。

### 要件

DFSPRSUR ユーティリティーを、ユーティリティー・バッチ領域でバッチ・プログラムとして実行する場合、DFSVSAMP DD ステートメントが必要になります。DFSPRSUR ユーティリティーを BMP として実行する場合、IMS DLISAS 制御領域の VSAM バッファ・プール・マネージャーによってバッファが提供されます。

### 推奨事項

現在、DFSPRSUR ユーティリティーに対して文書化された推奨事項はありません。

## 入出力

分析するデータベースは、DFSPRSUR ユーティリティが BMP として実行されているとき、共用することができます。DFSPRSUR ユーティリティで調査するデータベースの特定のセクション (キー範囲またはブロック番号範囲) を指示するのは、入力ユーティリティ制御ステートメントです。

DFSPRSUR ユーティリティで使えるようにするために、調査するデータベースの PCB を入れておく PSB を定義しておかなければなりません。(この PSB には、複数のデータベース PCB を入れることができます。) データベース PCB には、対応する DBD に定義されているすべてのセグメントに対する SENSEG ステートメントが含まれていなければなりません。PSB が PROCOPT=G と指定していることを確認してください。DFSPRSUR ユーティリティを BMP として実行するときは、PSBGEN ステートメントで OLIC=YES と指定しておき、PSB を IMS DC 制御領域に対して定義しておく必要があります。

DFSPRSUR ユーティリティは、出力としてデータベース調査報告書を作成します。指定された範囲の区画ごとに、データベース調査報告書には、データベース・レコードを読み取るためにアクセスされたブロック数の分布、レコードごとにアクセスされた平均ブロック数、レコードの平均サイズ、アクセスされた全レコードの総サイズ、および実際に読み取られたレコード数といった統計が記載されています。また、この報告書から、総フリー・スペース量、サイズ別の連続フリー・スペース域の分布、あるいは最大連続フリー・スペース域の位置を知ることがもできます。

指定された範囲ごとに、調査ユーティリティは 2 次データベース内のブロック数を 10 の部分に分割し、この 10 の部分のそれぞれに含まれるセグメントのうち、特定の範囲に属するセグメントをリストします。例えば、56 個のブロックがあるとすると、調査ユーティリティはそのブロック数を 10 個の部分 (1 から 6、7 から 12、...、55 から 56) に分割し、各部分に含まれるセグメントのうち、特定の範囲に属するセグメントをリストします。

## 戻りコード

データベース調査ユーティリティは、プログラム終了時に次のような戻りコードを出します。

コード 意味

- 0 エラーは検出されなかった
- 4 警告メッセージが出された
- 8 プログラムが異常終了した

## JCL 指定

DFSPRSUR ユーティリティは、標準の z/OS ジョブとして実行されます。DFSPRSUR ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含めることが必須であり可能です。

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

#### **EXEC** ステートメント

事前に作られたブロックを使用するバッチ・プログラムとしてデータベース調査ユーティリティを実行するには、次のように指定します。

```
PGM=DFSRR00,PARM='DBB,DFSPRSUR,...'
```

事前に作られたブロックを使用しないバッチ・プログラムとしてデータベース調査ユーティリティを実行するには、次のように指定します。

```
PGM=DFSRR00,PARM='DLI,DFSPRSUR,...'
```

バッチ・メッセージ処理プログラムとしてデータベース調査ユーティリティを実行するには、次のように指定します。

```
PGM=DFSRR00,PARM='BMP,DFSPRSUR,...'
```

通常の IMS 定位置パラメーターは、 PARM フィールドの中でプログラム名の後に置くことができます。

#### **DD** ステートメント

次の DD ステートメントは、必須データ・セットとオプション・データ・セットを定義します。

##### **STEPLIB DD**

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

##### **DFSRESLB DD**

IMS SVC モジュールが入っている許可ライブラリーを指します。

##### **SYSIN DD**

このプログラムのための入力制御データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。LRECL と BLKSIZE は共に 80 でなければなりません。

##### **IMS DD**

分析するデータベースを記述する DBD と PSB が入っているライブラリーを定義します。これらのデータ・セットは、直接アクセス装置に置いておかなければなりません。このステートメントは必須であり、必ず DBD ライブラリーを定義しなければなりません。PSB ライブラリーが必要になるのは、PARM=DLI と指定されたときだけです。

##### **IMSACB DD**

分析するデータベースを記述する ACB が入っているライブラリーを定義しま

す。このデータ・セットは、直接アクセス装置に置いておかなければなりません。このステートメントが必要になるのは、PARM=DBB と指定されたときだけです。

#### **SYSPRINT DD**

メッセージおよび報告書出力データ・セットを定義します。このデータ・セットは、テープ、直接アクセス装置、またはプリンターに置いておくことも、出力ストリームに入れて送ることもできます。

このデータ・セットに関して指定されている DCB パラメーターは、RECFM=FBM と LRECL=121 です。BLKSIZE は SYSPRINT DD ステートメントで指定し、121 の倍数になっていなければなりません。

この DD ステートメントは必須です。

#### **IEFRDER DD**

IMS ログ・データ・セットを定義します。このステートメントは、調査ユーティリティーがバッチ・プログラムとして実行される時に必要ですが、DUMMY として指定することができます。

#### **database DD**

分析するデータベースを定義します。DD 名は、DBD にある DD 名と一致している必要があります。このステートメントが必要になるのは、調査ユーティリティーがバッチ・プログラムとして実行されるときだけです。(調査ユーティリティーが BMP として実行される場合は、データベース・データ・セットは制御領域 JCL に定義されている必要があります。)

#### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ・プール情報が入っているデータ・セットを記述します。

DFSPRSUR ユーティリティーを、ユーティリティー・バッチ領域でバッチ・プログラムとして実行するときは、DFSVSAMP DD ステートメントが必要になります。DFSPRSUR ユーティリティーを BMP として実行する場合、IMS DLISAS 制御領域の VSAM バッファ・プール・マネージャーによってバッファが提供されます。

#### **DFSCTL DD**

順次バッファリング (SB) の活動化を要求する SBPARM 制御ステートメントが入っているデータ・セットを記述します。

DFSCTL ファイルは順次データ・セットにすることも、PDS のメンバーにすることもできます。レコード形式は F、FB、または FBS のいずれかで、レコード長は 80 でなければなりません。このデータ・セットは、直接アクセス装置またはテープに置いておくことも、入力ストリームに入れて送ることもできます。

#### **SYSABEND DD または SYSUDUMP DD**

ダンプ・データ・セットを定義します。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

#### **RECON1 DD**

最初の DBRC RECON データ・セットを定義します。

#### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。

### RECON3 DD

RECON1 または RECON2 にエラーが見付かったとき使用される、オプションの DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

関連概念:

➡ IMS バッファ・プール (システム定義)

関連資料:

419 ページの『第 30 章 データベース部分再編成ユーティリティ (DFSPRCT1 および DFSPRCT2)』

➡ 順次バッファリング制御ステートメント (システム定義)

➡ DBBBATCH プロシージャ (システム定義)

➡ DLIBATCH プロシージャ (システム定義)

---

## DFSPRSUR ユーティリティの制御ステートメント

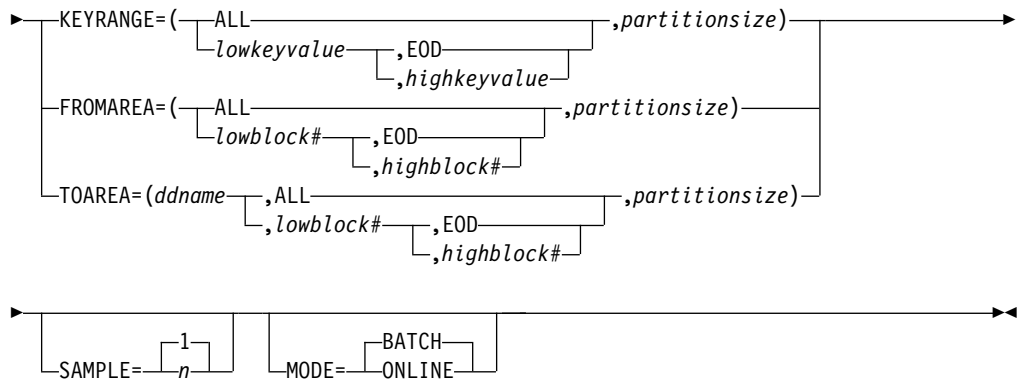
DFSPRSUR ユーティリティは、入力ステートメントを使用して処理オプションを記述します。

入力ステートメントは、以下のガイドラインに準拠している必要があります。

- ステートメントの先頭の文字は、17 桁目よりも前から始まっていなければなりません。
- 1 つのステートメントで複数のキーワードを指定することはできません。
- あるオペランドの途中で空白があると、その空白の後に続く文字は、コメントと見なされます。
- キーワードのオペランドが 1 つの入力ステートメントに入りきらないときは、継続ステートメントを使うことができます。
- ステートメントを継続するには、72 桁目に空白以外の文字が入っていなければなりません。
- 1 桁目にアスタリスクを入れると、コメントだけのステートメントが指定できます。

データベース調査ユーティリティ制御ステートメントの形式は、次のとおりです。

▶▶—DBNAME=*dbname*————▶▶



**DBNAME=**

データベースのひずみ、フリー・スペース、またはその両方を見付けるために調査するデータベースを指定します。このオペランドは、HD 編成の DBD の名前にしなければなりません。DBNAME は必須であり、一度しか使えません。

**KEYRANGE=**

データベースのひずみを見付けるために分析するキーの範囲を指定します。使用できる KEYRANGE 定義は 1 つだけです。

制約事項: KEYRANGE を指定したときは、同一ジョブ・ストリームの中の他の入力ステートメントで FROMAREA または TOAREA を指定することはできません。

データベースが HDAM のときは、KEYRANGE は無効です。オペランドはルート・セグメント・キーまたは総称キーであり、どちらも最大 255 バイトです。例えば、次の例のように値の前に X を付け、値を引用符で囲むと、キーを 16 進値で表すことができます。

```
KEYRANGE=(X'C8C5E7',X'D2C5E8',1000)
```

上限がデータベースの終わりであれば、最大キー値を「EOD」として指定できます。最小キー値を「ALL」として指定し、最大キー値を省略することができます。その場合は、範囲はデータベース全体です。

区画サイズは、その範囲の各区画に入れるデータベース・レコードの数で指定します。指定できる数は、1 から 9999 までです。データベース調査ユーティリティーは、範囲の区画別の統計と、範囲全体の統計をとります。

**FROMAREA=**

データベースのひずみを見付けるために分析するブロックの範囲を 1 つ指定します。使用できる FROMAREA 定義は 1 つだけです。

制約事項: FROMAREA を指定したときは、同一ジョブ・ストリームの中の他の入力ステートメントで KEYRANGE または TOAREA を指定することはできません。

データベースが HIDAM のときは、FROMAREA は無効です。オペランドは、ルート・アドレス可能域内のブロック番号です。上限がルート・アドレス可能域内の最後のブロックであれば、最大ブロック番号を「EOD」として指定できます。最小ブロック番号を「ALL」として指定し、最大ブロック番号を省略することができます。その場合、範囲はルート・アドレス可能域全体です。



区画サイズは、その範囲の各区画に入れるルート・アドレス可能域のブロックの数で指定します。指定できる数は、1 から 9999 までです。調査ユーティリティーは、範囲の区画別の統計と、範囲全体の統計をとります。

#### TOAREA=

フリー・スペースを見付けるために分析するデータベースの区域を 1 つ指定します。TOAREA キーワードは、データ・セット・グループ内の範囲となるように定義します。1 つのデータベースで指定できる TOAREA 定義は 10 個までです。

注: TOAREA を指定したときは、同一ジョブ・ストリームの中の他の入力ステートメントで KEYRANGE または FROMAREA を指定することはできません。DD 名 は、調査するデータベースに入っているデータ・セットを定義する DD ステートメントの名前と同じものでなければなりません。

ブロック番号は、データ・セットの範囲内ならば、どのブロック番号でも指定でき、ALL または EOD も指定できます。最小ブロック番号は、2 が最小値です。最小ブロック番号に 2 未満の値を指定すると、2 と見なされます。最小ブロック番号を ALL として指定し、最大ブロック番号を省略したときは、その範囲はデータ・セット・グループ全体です。最大ブロック番号を EOD として指定したときは、範囲の上限はデータ・セット・グループの最後のブロックになります。

区画サイズは、その範囲の各区画に入れるブロックの数で指定します。指定できる数は、1 から 9999 までです。調査ユーティリティーは、範囲の区画別の統計と、範囲全体の統計をとります。

#### SAMPLE=

ユーティリティーに各範囲の一部分だけをサンプリングさせることを指定します。このキーワードのオペランドは、1 から 1000 までの数で指定します。オペランドには、KEYRANGE と FROMAREA のときはレコード数を、TOAREA のときはブロック数を指定します。レコード (またはブロック) は  $n^{\text{th}}$  番目ごとにアクセスされ、その間に介在するレコード (またはブロック) は分析の対象となりません。例えば、 $n=10$  とすると、最初のレコード/ブロック、11 番目のレコード/ブロック、21 番目のレコード/ブロック、と以後同様にアクセスされ、データベース・ストレージ情報の報告に使用されます。SAMPLE キーワードが指定されていないと、範囲内のすべてのレコード (またはブロック) がアクセスされます。

#### MODE=

データベース調査ユーティリティーを BMP として実行するか (MODE=ONLINE)、バッチ・プログラムとして実行するか (MODE=BATCH) を指定します。BATCH がデフォルトです。

---

## DFSPRSUR ユーティリティーの例

このトピックの例では、DFSPRSUR ユーティリティーのサンプル JCL を示します。

この節にある例では、桁合わせを容易にするために、SYSIN ステートメントの上に次のようなコメント行が示されています。

```
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7----
```

このコメント行は単なる参考用です。

この節の例で、使用する DBRC で動的割り振りを行わない場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

図 9. 動的割り振りなしで DBRC を使用するための DD ステートメント

```
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
```

サブセクション:

- 『DFSPRSUR をバッチとして実行する』
- 48 ページの『DFSPRSUR をバッチ・プログラム JCL として実行する』
- 50 ページの『DFSPRSUR を BMP として実行する』

## DFSPRSUR をバッチとして実行する

この例では、DFSPRSUR をバッチ・プログラムとして実行して、FROMAREA キーワードの使用によってデータベースの一部を分析するのに必要な JCL とユーティリティー制御ステートメントを示しています。

```
//STSTN53 EXEC PGM=DFSRR00,
//          PARM='DLI,DFSPRSUR,PRPSB23P,,,,,,,,,N,N'
//IMS      DD DSN=IMS.PSBLIB,DISP=SHR
//          DD DSN=IMS.DBDLIB,DISP=SHR
//IEFRDER DD DUMMY
//SYSPRINT DD SYSOUT=A,DCB=(BLKSIZE=1210)
//SNAPDD  DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//PR23DD1 DD DSN=PR23RW00,DISP=SHR
//PR23DD2 DD DSN=PR23A,DISP=SHR
//* +---1---+---2---+---3---+---4---+---5---+---6---+---7---
//SYSIN    DD *
            MODE=BATCH
            DBNAME=PR23RW00
            FROMAREA=(001,EOD,1)
//DFSVSAMP DD input for VSAM and OSAM buffers and options
```

以下の図は、例 1 の内容を実行すると得られる FROMAREA 区画報告書のサンプルです。このような報告書が、区画ごとに 1 つ作成されます。

図 10. 調査ユーティリティーの FROMAREA 区画報告書

```
SURVEYOR FROMAREA PARTITION REPORT FOR DBD PR23RW00
PARTITION BLOCK NUMBERS 2 TO 2
TOTAL NUMBER OF ROOTS = 2
TOTAL LENGTH OF SEGMENTS = 129
AVERAGE LENGTH PER DBR = 64
SEGMENTS OF A DATABASE RECORD (DBR) MAY SPREAD ACROSS SEVERAL BLOCKS
FOR MANY REASONS. THIS TABLE SUMS THE NUMBER OF DATABASE RECORDS BY
THE NUMBER OF BLOCKS THEY OCCUPY.
# OF BLOCKS/SUM OF # DBR WHICH OCCUPY THIS # OF BLOCKS
NO. BLK 1 2 3 4 5 6-8 9-11 12-14 15-17 > 17
NO. DBR 0 2 0 0 0 0 0 0 0 0
THIS TABLE SHOWS THE NUMBER AND LENGTH OF SEGMENTS IN EACH TENTH OF
EACH DSG FOR THIS DATABASE.
DSG          BLOCK      TOTAL LENGTH NO.          AVG LENGTH  % AREA
DDNAME  BLKSIZE  LOW  HIGH  OF SEGMENTS  OF SEGMENTS  OF SEGMENTS  OCCUPIED
```

PR23DD1	4096	1-	15	53	2	26	.1
		16-	30	0	0	0	.0
		31-	45	0	0	0	.0
		46-	60	0	0	0	.0
		61-	75	0	0	0	.0
		76-	90	0	0	0	.0
		91-	105	0	0	0	.0
		106-	120	0	0	0	.0
		121-	135	0	0	0	.0
		136-	EOD	0	0	0	.0

FROMAREA 区画報告書のフィールドは次のとおりです。

**PARTITION BLOCK NUMBERS**

区画内の最小ブロック番号と最大ブロック番号

**TOTAL NUMBER OF ROOTS**

区画内のルートの総数

**TOTAL LENGTH OF SEGMENTS**

区画内の全セグメントの合計の長さ

**AVERAGE LENGTH PER DBR**

データベース・レコードの平均の長さ

**NO. BLK**

ブロック数の見出し行

**NO. DBR**

見出しに示されたブロック数にわたって広がっている DBR の数

**DSG DDNAME**

DSG の DD 名

**BLKSIZE**

DSG のブロック・サイズ

**BLOCK LOW および HIGH**

DSG の該当部分の最小ブロック番号と最大ブロック番号

**TOTAL LENGTH OF SEGMENTS**

DSG の該当部分に物理的に置かれている全セグメント (報告対象区画内で見付かったもの) の合計の長さ

**NUMBER OF SEGMENTS**

DSG の該当部分に物理的に置かれているセグメント (報告対象区画内で見付かったもの) の数

**AVERAGE LENGTH OF SEGMENTS**

DSG の該当部分に物理的に置かれているセグメント (報告対象区画内で見付かったもの) の平均の長さ

**PERCENT OF AREA OCCUPIED**

DSG の該当部分に物理的に置かれているセグメント (報告対象区画内で見付かったもの) が占めている区域の割合。報告対象範囲外のデータベース・レコードに属しているセグメントが、報告対象 DSG 区域内に物理的に置かれている可能性があります、これらは報告書には反映されません。

図 11 は、例 1 の内容を実行すると得られる FROMAREA 範囲報告書のサンプルです。このような報告書が、範囲全体を要約するために 1 つ作成されます。

図 11. 調査ユーティリティーの FROMAREA 範囲報告書

```

SURVEYOR FROMAREA RANGE REPORT FOR DBD PR23RW00
RANGE BLOCK NUMBERS      1 TO 10
TOTAL NUMBER OF ROOTS =   20
TOTAL LENGTH OF SEGMENTS =  991
AVERAGE LENGTH PER DBR =   49
SEGMENTS OF A DATABASE RECORD (DBR) MAY SPREAD ACROSS SEVERAL BLOCKS
FOR MANY REASONS. THIS TABLE SUMS THE NUMBER OF DATABASE RECORDS BY
THE NUMBER OF BLOCKS THEY OCCUPY.
# OF BLOCKS/SUM OF # DBR WHICH OCCUPY THIS # OF BLOCKS
NO. BLK  1    2    3    4    5    6-8    9-11   12-14   15-17  > 17
NO. DBR  5    15    0    0    0    0    0    0    0    0

```

FROMAREA 範囲報告書のフィールドは次のとおりです。

**RANGE BLOCK NUMBERS**

範囲内の最小ブロック番号と最大ブロック番号

**TOTAL NUMBER OF ROOTS**

範囲内のルートの総数

**TOTAL LENGTH OF SEGMENTS**

範囲内の全セグメントの合計の長さ

**AVERAGE LENGTH PER DBR**

データベース・レコードの平均の長さ

レコードの数とこれらのレコードが広がっているブロックの数との関係を示す表が次に現れます。この表のフィールドは次のとおりです。

**NO. BLK**

ブロック数の見出し行

**NO. DBR**

見出しに示されたブロック数にわたって広がっている DBR の数

**DFSPRSUR をバッチ・プログラム JCL として実行する**

この例では、DFSPRSUR をバッチ・プログラムとして実行して、TOAREA キーワードの使用によってデータベース全体を分析するのに必要な JCL とユーティリティー制御ステートメントを示しています。

```

//STSTN54 EXEC PGM=DFSRR00,
//          PARM='DLI,DFSPRSUR,PRPSB23P,,,,,,,,,N,N'
//IMS      DD DSN=IMS.PSBLIB,DISP=SHR
//          DD DSN=IMS.DBDLIB,DISP=SHR
//IEFRDER  DD DUMMY
//SYSPRINT DD SYSOUT=A,DCB=(BLKSIZE=1210)
//SNAPDD   DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//PR23DD1  DD DSN=PR23RW00,DISP=SHR
//PR23DD2  DD DSN=PR23A,DISP=SHR
// * +---1---+---2---+---3---+---4---+---5---+---6---+---7---
//SYSIN    DD *
           MODE=BATCH
           DBNAME=PR23RW00

```

```
TOAREA=(PR23DD1,ALL,1)
TOAREA=(PR23DD2,ALL,1)
//DFSVSAMP DD input for VSAM and OSAM buffers and options
```

以下の図は、例 2 の内容を実行すると得られる区画報告書のサンプルです。このような報告書が、区画ごとに 1 つ作成されます。

図 12. 調査ユーティリティの TOAREA 区画報告書

```
SURVEYOR TOAREA PARTITION REPORT FOR DBD PR23RW00 DSG PR23DD1
PARTITION BLOCK NUMBERS 149 TO 149
TOTAL NUMBER OF FREE SPACE ELEMENTS = 1
TOTAL AMOUNT OF FREE SPACE = 4085
TOTAL NUMBER OF BLOCKS ACCESSED = 1
AVERAGE AMOUNT OF FREE SPACE PER FSE = 4085
PERCENT OF FREE SPACE TO TOTAL AREA = 99.9
TEN LARGEST AREAS OF FREE SPACE
SIZE NUMBER LOCATION (FIRST TEN)
4085 1 149
```

TOAREA 区画報告書のフィールドは次のとおりです。

#### **PARTITION BLOCK NUMBERS**

区画内の最小ブロック番号と最大ブロック番号

#### **TOTAL NUMBER OF FREE SPACE ELEMENTS**

区画内で見付かったフリー・スペース・エレメントの総数。データベース調査ユーティリティは、IMS スペース管理ルーチンによってフォーマット設定されなかったブロックに残っているフリー・スペースを計算します。この計算を行うとき、これらの未フォーマット設定ブロックのおのおのには、1 つのフリー・スペース・エレメント (FSE) と共に 1 つのフリー・スペース・アンカー・ポイント (FSEAP) があるものと見なされます。

#### **TOTAL AMOUNT OF FREE SPACE**

区画内で見付かったフリー・スペースの総量

#### **TOTAL NUMBER OF BLOCKS ACCESSED**

区画内でアクセスされたブロックの総数

#### **AVERAGE AMOUNT OF FREE SPACE PER FSE**

区画内で見付かったフリー・スペース・エレメント当たりのフリー・スペースの平均量

#### **PERCENT OF FREE SPACE TO TOTAL AREA**

調査された区画ブロック内で見付かったフリー・スペースの割合

フリー・スペース域を最大のものから順に 10 個まで示す表が次に現れます。この表のフィールドは次のとおりです。

#### **SIZE**

フリー・スペース・エレメントのサイズ

#### **NUMBER**

この区画内で見付かったフリー・スペース・エレメントで SIZE に示されたサイズを持つものの数

## LOCATION (FIRST TEN)

この区画内で見付かったフリー・スペース・エレメントで SIZE に示されたサイズを持つもののうち、最初の 10 個のブロック番号

以下の図は、例 2 の内容を実行すると得られる範囲報告書のサンプルです。このような報告書が、範囲全体を要約するために 1 つ作成されます。

図 13. 調査ユーティリティの TOAREA 範囲報告書

```
SURVEYOR TOAREA RANGE REPORT FOR DBD PR23RW00
RANGE BLOCK NUMBERS      2 TO      149
TOTAL NUMBER OF FREE SPACE ELEMENTS   =      148
TOTAL AMOUNT OF FREE SPACE            =     603949
TOTAL NUMBER OF BLOCKS ACCESSED       =      148
AVERAGE AMOUNT OF FREE SPACE PER FSE =     4080
PERCENT OF FREE SPACE TO TOTAL AREA   =     99.7
```

TOAREA 範囲報告書のフィールドは次のとおりです。

### RANGE BLOCK NUMBERS

範囲内の最小ブロック番号と最大ブロック番号

### TOTAL NUMBER OF FREE SPACE ELEMENTS

範囲内で見付かったフリー・スペース・エレメントの総数。データベース調査ユーティリティは、IMS スペース管理ルーチンによってフォーマット設定されなかったブロックに残っているフリー・スペースを計算します。この計算を行うとき、これらの未フォーマット設定ブロックのおおのには、1 つのフリー・スペース・エレメント (FSE) と共に 1 つのフリー・スペース・アンカー・ポイント (FSEAP) があるものと見なされます。

### TOTAL AMOUNT OF FREE SPACE

範囲内で見付かったフリー・スペースの総量

### TOTAL NUMBER OF BLOCKS ACCESSED

範囲内でアクセスされたブロックの総数

### AVERAGE AMOUNT OF FREE SPACE PER FSE

範囲内で見付かったフリー・スペース・エレメント当たりのフリー・スペースの平均量

### PERCENT OF FREE SPACE TO TOTAL AREA

調査された範囲内で見付かったフリー・スペースの割合

## DFSPRSUR を BMP として実行する

この例では、DFSPRSUR を BMP として実行して、KEYRANGE キーワードの使用によってデータベースの一部を分析するのに必要な JCL とユーティリティ制御ステートメントを示しています。分析するデータベースの DD ステートメントは、IMS 制御領域 JCL に入れておく必要があります。

```
//STSTE01 EXEC PGM=DFSRR00,
//          PARM='DLI,DFSPRSUR,PRPSB01Y,,,,,,,,,N,N'
//IMS      DD DSN=IMS.PSBLIB,DISP=SHR
//          DD DSN=IMS.DBDLIB,DISP=SHR
//IEFRDER DD DUMMY
//SYSPRINT DD SYSOUT=A,DCB=(BLKSIZE=1210)
//SYSUDUMP DD SYSOUT=A
//SNAPDD  DD SYSOUT=A
```

```

//PR01DD DD DSN=PR01RW00,DISP=SHR
//PR01DD DD DSN=PR01I,DISP=SHR
//* +---1---+---2---+---3---+---4---+---5---+---6---+---7
//SYSIN DD *
        MODE=BATCH
        DBNAME=PR01RW00
        KEYRANGE=(000050,000100,10)
        SAMPLE=2
//DFSVSAMP DD input for VSAM and OSAM buffers and options

```

以下の図は、例 3 の JCL を実行すると得られる区画報告書のサンプルです。このような報告書が、区画ごとに 1 つ作成されます。

図 14. 調査ユーティリティの KEYRANGE 区画報告書

```

SURVEYOR KEYRANGE PARTITION REPORT FOR DBD PR01RW00
KEYRANGE = '000050'
TO =      '000090'
TOTAL NUMBER OF ROOTS =          3
TOTAL LENGTH OF SEGMENTS =       454
AVERAGE LENGTH PER DBR =        151
SEGMENTS OF A DATABASE RECORD (DBR) MAY SPREAD ACROSS SEVERAL BLOCKS
FOR MANY REASONS. THIS TABLE SUMS THE NUMBER OF DATABASE RECORDS BY
THE NUMBER OF BLOCKS THEY OCCUPY.
# OF BLOCKS/SUM OF # DBR WHICH OCCUPY THIS # OF BLOCKS
NO. BLK  1    2    3    4    5    6-8    9-11  12-14  15-17  > 17
NO. DBR  3    0    0    0    0    0    0    0    0    0
THIS TABLE SHOWS THE NUMBER AND LENGTH OF SEGMENTS IN EACH TENTH OF
EACH DSG FOR THIS DATABASE.
DSG          BLOCK      TOTAL LENGTH  NO. OF    AVG LENGTH  % AREA
DDNAME BLKSIZE  LOW  HIGH  OF SEGMENTS  SEGMENTS  OF SEGMENTS  OCCUPIED
-----
PR01DD 4096    1-    15  454          5         90          .7
          16-   30  0           0          0          .0
          31-   45  0           0          0          .0
          46-   60  0           0          0          .0
          61-   75  0           0          0          .0
          76-   90  0           0          0          .0
          91-  105  0           0          0          .0
          106- 120  0           0          0          .0
          121- 135  0           0          0          .0
          136- EOD  0           0          0          .0

```

KEYRANGE 区画報告書のフィールドは次のとおりです。

**KEYRANGE= および TO=**

区画内の最小キー値と最大キー値

**TOTAL NUMBER OF ROOTS**

区画内のルートの総数

**TOTAL LENGTH OF SEGMENTS**

区画内の全セグメントの合計の長さ

**AVERAGE LENGTH PER DBR**

データベース・レコードの平均の長さ

レコードの数とこれらのレコードが広がっているブロックの数との関係を示す表が次に現れます。この表のフィールドは次のとおりです。

**NO. BLK**

ブロック数の見出し行

**NO. DBR**

見出しに示されたブロック数にわたって広がっている DBR の数

調査しているデータベース内の各 DSG の 10 番目ごとの特性を示す表が次に現れます。この表のフィールドは次のとおりです。

**DSG DDNAME**

DSG の DD 名

**BLKSIZE**

DSG のブロック・サイズ

**BLOCK LOW および HIGH**

DSG の該当部分の最小ブロック番号と最大ブロック番号

**TOTAL LENGTH OF SEGMENTS**

DSG の該当部分に物理的に置かれている全セグメント (報告対象区画内で見付かったもの) の合計の長さ

**NUMBER OF SEGMENTS**

DSG の該当部分に物理的に置かれているセグメント (報告対象区画内で見付かったもの) の数

**AVERAGE LENGTH OF SEGMENTS**

DSG の該当部分に物理的に置かれているセグメント (報告対象区画内で見付かったもの) の平均の長さ

**PERCENT OF AREA OCCUPIED**

DSG の該当部分に物理的に置かれているセグメント (報告対象区画内で見付かったもの) が占めている区域の割合。報告対象範囲外のデータベース・レコードに属しているセグメントが、報告対象 DSG 区域内に物理的に置かれている可能性があります、これらは報告書には反映されません。

以下の図は、例 3 の内容を実行すると得られる範囲報告書のサンプルです。このような報告書が、範囲全体を要約するために 1 つ作成されます。

図 15. 調査ユーティリティの KEYRANGE 範囲報告書

```

SURVEYOR KEYRANGE RANGE REPORT FOR DBD PR01RW00
KEYRANGE = '000050'
TO = '000100'
TOTAL NUMBER OF ROOTS = 3
TOTAL LENGTH OF SEGMENTS = 454
AVERAGE LENGTH PER DBR = 151
SEGMENTS OF A DATABASE RECORD (DBR) MAY SPREAD ACROSS SEVERAL BLOCKS
FOR MANY REASONS. THIS TABLE SUMS THE NUMBER OF DATABASE RECORDS BY
THE NUMBER OF BLOCKS THEY OCCUPY.
# OF BLOCKS/SUM OF # DBR WHICH OCCUPY THIS # OF BLOCKS
NO. BLK 1 2 3 4 5 6-8 9-11 12-14 15-17 > 17
NO. DBR 3 0 0 0 0 0 0 0 0 0

```

KEYRANGE 範囲報告書のフィールドは次のとおりです。

**KEYRANGE= および TO=**

範囲内の最小ブロック番号と最大ブロック番号



**TOTAL NUMBER OF ROOTS**

範囲内のルートの総数

**TOTAL LENGTH OF SEGMENTS**

範囲内の全セグメントの合計の長さ

**AVERAGE LENGTH PER DBR**

データベース・レコードの平均の長さ

レコードの数とこれらのレコードが広がっているブロックの数との関係を示す表が次に現れます。この表のフィールドは次のとおりです。

**NO. BLK**

ブロック数の見出し行

**NO. DBR**

見出しに示されたブロック数にわたって広がっている DBR の数



## 第 5 章 DEDB 変更ユーティリティ (DBFUDA00)

DEDB エリアの変更 (ALTERAREA) または DEDB データベースの 2 ステージ・ランダムマイザーの置き換え (REPLRAND) を行うには、DEDB 変更ユーティリティ (DBFUDA00) を使用します。

DEDB 変更ユーティリティを使用して DEDB を変更するために必要な完全な手順については、以下のプロシージャを参照してください。

- DEDB 変更ユーティリティを使用したアクティブ DEDB エリアのサイズ属性の変更 (データベース管理)
- DEDB 変更ユーティリティを使用した、アクティブ DEDB エリアを変更するときのランダムマイザーの変更
- DEDB 変更ユーティリティを使用したアクティブ DEDB データベース・ランダムマイザーのオンラインでの置換 (データベース管理)

サブセクション:

- 『制約事項』
- 56 ページの『前提条件』
- 56 ページの『要件』
- 56 ページの『推奨事項』
- 56 ページの『入出力』
- 57 ページの『JCL 指定』
- 57 ページの『戻りコード』

### 制約事項

DEDB 変更ユーティリティ (DBFUDA00) を使用する際は、以下の制約事項が適用されます。

- 定義済みの順次従属 (SDEP) セグメントのある DEDB データベース域は、変更することができません。
- 複数の DEDB 変更ユーティリティを、IMS データ共用システムで同時に実行することができます。ただし、同時に実行される各ユーティリティは、異なるアクティブ DEDB データベースに対して実行する必要があります。
- ユーティリティの 1 回の実行につき、呼び出すことができる関数は 1 つのみ (ALTERAREA または REPLRAND) です。
- DEDB 変更ユーティリティで一度に変更できる DEDB データベース内の区域は 1 つのみです。DEDB 変更ユーティリティの複数のインスタンスを同時に実行できますが、各インスタンスを異なる DEDB データベースに対して実行する必要があります。
- このユーティリティは、仮想記憶オプション (VSO) モードまたは共用仮想記憶オプション (SVSO) モードの DEDB データベースをサポートしません。VSO オプションまたは SVSO オプションが指定された DEDB データベースを

| /VUNLOAD コマンドを使用せず DASD にアンロードしてから、DEDB  
| 変更ユーティリティを実行する必要があります。

- | • DEDB ALTER を 1 回実行して変更することができるのは、1 つの DEDB に  
| つき 1 つのエリアのみです。

## | 前提条件

| DEDB 変更ユーティリティでは、IBM z/OS DFSORT を DEDB 変更プロセスの  
| 一部として使用します。DFSORT では、NOMSGDD または SYSOUT カードの設  
| 定が必要です。設定されていない場合、z/OS マスター・コンソール上にエラー・  
| メッセージ ICE158A が表示される場合があります。SYSOUT カードを指定する場  
| 合、DEDB 変更が実行される IMS 制御領域の JCL で指定する必要があります。追  
| 加情報については、IBMz/OSDFSORTの資料を参照してください。

## | 要件

| DEDB 変更ユーティリティ (DBFUDA00) を使用する際は、以下の要件が適用さ  
| れます。

- | • DEDB データベースが DBRC に登録されている必要があります。
- | • ランダマイザーは、純粋な 2 ステージ・ランダムマイザーでなければならず、ま  
| た 2 ステージ・ランダムマイザーとして DBD で定義されている必要があります。
- | • DEDB 変更ユーティリティを実行するには、すべての IMS データ共用システ  
| ムが IMS バージョン 13 以降のレベルでなければなりません。RECON デー  
| タ・セットの MINVERS 値は 13.1 以降であることが必要です。

## | 推奨事項

- | • オフピークの時間帯に、DEDB 変更ユーティリティ (DBFUDA00) を実行しま  
| す。
- | • ユーティリティを実行する前に、イメージ・コピー・データ・セット  
| GENMAX の値を 1 または 2 増やして、ユーザーのイメージ・コピー・デー  
| タ・セットに対応できるようにします。後から GENMAX の元の値に復元する  
| ことができます。

## | 入出力

| 入力については、DEDB 変更ユーティリティ (DBFUDA00) は、DEDB ユーティ  
| リティ・コマンドで指定された入力パラメーターが入っている制御データ・セッ  
| トを使用します。SYSIN DD ステートメントがこの制御データ・セットを指しま  
| す。

| DEDB 変更ユーティリティ (DBFUDA00) は、以下の出力を生成することができます。

- | • PLEX のオンライン IMS システムが使用する新規 ADS。
- | • 新規 ADS にマイグレーションされるすべてのデータ。
- | • 新規の ADS 名を反映するために更新される DBRC。
- | • 新規ユーザーのイメージ・コピー・データ・セット (DBRC に登録済み)。
- | • 以前のすべてのリカバリ情報は、DBRC では無効にされています。

## JCL 指定

次の JCL ステートメントが必要です。

- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

DEDB 変更ユーティリティ (DBFUDA00) は、標準の z/OS ジョブとして実行されます。DEDB 変更ユーティリティ (DBFUDA00) の JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。DBFUDA000 ユーティリティの処理オプションは、このユーティリティに割り振った制御データ・セットの DEDB ユーティリティ・コマンドによって指定されます。

### DD ステートメント

#### STEPLIB DD

DEDB 変更ユーティリティ (DBFUDA00) が入っているライブラリーを記述します。

#### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

#### SYSIN DD

DEDB 変更ユーティリティ (DBFUDA00) の処理オプションを制御する DEDB ユーティリティ・コマンドが入っている入力制御データ・セットを記述します。

#### SYSPRINT DD

出力メッセージと統計が入っている出力データ・セットを記述します。

```
//ALTAREA JOB ....
//STEPLIB DD DSN=IMS.CRESLIB,DISP=SHR
//PROCLIB DD DSN=IMSVS.PROCLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&SOUT
//SYSUDUMP DD SYSOUT=&SOUT,...
//S0 EXEC FPUTIL,RGN=1M,DBD=DEDBJN21,REST=00,IMSID=IMS1
//SYSIN DD *
TYPE ALTER
ALTERAREA DB21AR0
PHYSICAL ADJNO
UNKEYSEG NONE
RETRY NO
TIMEOUT 30
GO
/*
```

## 戻りコード

以下の戻りコードが作成されます。

コード 意味

- 0 ユーティリティは正常に完了しました。
- 8 AREA が登録されていません  
ユーティリティは、VSO、SVSO を処理できません

AREA には SDEP があります  
 UNKEY セグメントが存在します  
 AREA が不確定です  
 専用バッファ割り振りが失敗しました  
 UOW 読み取り/書き込みの失敗です  
 同期の失敗です  
 コミットの失敗です

- 12 MINVER 13.1 が必要です  
 DBD が DEDB ではありません  
 エリアには、少なくとも 1 つの使用可能なシャドー ADS、および 1 つの使用可能なシャドー IDS が必要です
- 20 EEQE が存在します  
 エリアのオープン・エラーです  
 エリアが静止しています  
 ステージ 2 ランダマイザーではありません

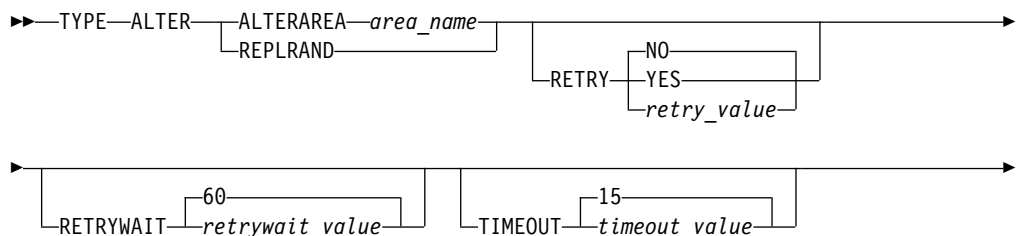
関連タスク:

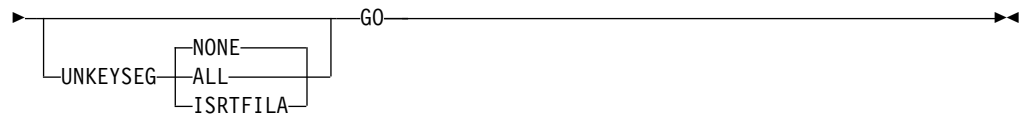
- ➡ DEDB 変更ユーティリティを使用したアクティブ DEDB エリアのサイズ属性の変更 (データベース管理)
- ➡ DEDB 変更ユーティリティおよび IMS 生成ユーティリティを使用した、アクティブ DEDB エリアを変更するときのランダマイザーの変更 (データベース管理)
- ➡ DEDB 変更ユーティリティを使用したアクティブ DEDB データベース・ランダマイザーのオンラインでの置換 (データベース管理)

## DEDB 変更ユーティリティ (DBFUDA00) の制御ステートメント

DEDB 変更ユーティリティ (DBFUDA00) は、入力ステートメントを使用して処理オプションを記述します。

以下は、DEDB 変更ユーティリティの制御ステートメントのフォーマットです。





## パラメーター

重要: DEDB 変更ユーティリティのパラメーターは、他のユーティリティに指定しないでください。

ALTERAREA または REPLRAND を指定する必要があります。

### ALTERAREA

*area\_name* で指定された DEDB データベースのアクティブな DEDB 区域の AREA ステートメント上で、SIZE、UOW、および ROOT パラメーターを変更します。DEDB 変更ユーティリティでは、ユーティリティの 1 回の実行につき 1 つの ALTERAREA 制御ステートメントがサポートされます。

制約事項:

- ALTERAREA 制御ステートメントは、REPLRAND 制御ステートメントと同時に指定することはできません。

ALTERAREA= の値は、変更するアクティブな DEDB 区域の名前です。

要件: シャドー・イメージ・コピー・データ・セットが 1 つ必要です。1 つから 6 つのシャドー・データ・セットが必要です。

**GO** ユーティリティを開始します。

### REPLRAND

ステージング ACBLIB または IMS カタログの DEDB DBD 定義から、*randomizer\_name* で指定されたランダムマイザー名を使用して、DEDB データベースのランダムマイザーを置き換えます。その他の属性は変更されません。新しいランダムマイザーの名前には、使用されているランダムマイザーとは異なる名前を使用しなければなりません。

制約事項:

- REPLRAND 制御ステートメントは、ALTERAREA 制御ステートメントと同時に指定することはできません。

REPLRAND 関数は、定義された SDEP セグメントの有無に関わらず、DEDB データベースをサポートします。

REPLRAND 関数では、シャドー・エリア・データ・セットまたはシャドー・イメージ・コピー・データ・セットは不要です。ランダムマイザーが置き換えられます。

DEDB 変更ユーティリティの REPLRAND 関数が正常終了したら、DEDB データベースの現行のランダムマイザーが新規のランダムマイザーに置き換えられます。

要件:

- REPLRAND 関数では、ステージング ACB ライブラリーまたは IMS カタログのランダムマイザーの名前には、現在使用されているランダムマイザーとは異なる名前を使用しなければなりません。
- REPLRAND 関数を使用するには、ランダムマイザーが 2 ステージ・ランダムマイザーでなければならず、また DEDB DBD 定義で 2 ステージ・ランダムマイザーとして定義されている必要があります。

#### RETRY

TIMEOUT 値の期限が切れた後、再試行するかどうかを指定するオプションの制御ステートメントです。

**NO** TIMEOUT 値の期限が切れても再試行を行わないように指定します。NO はデフォルトです。

#### YES

*retry\_value* を 5 に設定します。

#### *retry\_value*

TIMEOUT 値の期限切れ後に再試行する回数を設定します。*retry\_value* は 1 から 2 桁の数値であり、1 から 99 の値にすることができます。

#### RETRYWAIT

オプションの制御ステートメントです。TIMEOUT 値の期限が切れ、RETRY=YES または RETRY=*retry\_value* が指定されている場合に、RETRYWAIT では、タイムアウトしたプロセスを再試行する前に待機する時間を指定します。*retry\_value* は 1 から 3 桁の数値であり、1 から 999 秒の値にすることができます。デフォルト値は 60 秒です。

#### TIMEOUT

オプションの制御ステートメントです。この TIMEOUT 値は、シャドー・エリア・データ・セットをアクティブ・エリア・データ・セットと同期し、アクティブな DEDB 区域の SIZE、UOW、および ROOT の変更とアクティブ DEDB データベースの RMNAME の変更をコミットするために、DEDB 変更ユーティリティーが IMS アプリケーションを一時停止することを許可します。

TIMEOUT 値は 1 から 3 桁の数値であり、1 から 999 秒の値にすることができます。デフォルト値は 15 秒です。

#### TYPE ALTER

必須の制御ステートメントです。DEDB 変更ユーティリティーを開始します。

#### UNKEYSEG ALL|NONE|ISRTFILA

ALTERAREA 関数のこのパラメーターは、キーなしセグメントを DEDB 変更ユーティリティーが許容するかどうかを指定します。デフォルトは NONE です。

#### NONE

キーなしセグメントが DEDB 定義内で定義されている場合、ユーティリティーの実行を許可しません。

キーなしセグメントが検出されると、メッセージ DFS4657E が発行されます。

#### ISRTFILA

キーなしセグメントが DEDB 定義内に定義されている場合でも、DEDB 変更ユーティリティーの実行を許可します。ユーティリティーの処理中に、キ



キーなしセグメントに更新が適用されると、ユーティリティは完了前に強制終了されますが、1 つ例外があります。キーなしセグメントの挿入が完了し、このセグメントの挿入規則が FIRST または LAST である場合には、ユーティリティは続行されます。

ユーティリティの処理中にキーなしセグメントが検出されると、メッセージ DFS4631E または DFS4632E が発行されます。

#### ALL

ユーティリティの実行を許可し、キーなしセグメントを含む更新の処理を、以下の方法で行います。

- ターゲット・セグメントがキーを持っておらず、このセグメントへの階層パスにあるすべてのセグメントにキーがある場合、次のように処理します。
  - REPL 操作または DLET 操作の場合、ユーティリティは、更新の以前のイメージと一致するデータを持つ最初のセグメント・インスタンスをシャドウ・エリア内で見つけます。このセグメントは置き換えられるか、または削除されます。
  - 挿入規則が HERE の ISRT 操作の場合、挿入規則に FIRST が指定されている場合と同様に、挿入が実行されます。
- ターゲット・セグメントにキーがある場合、またはキーなしだが階層パス内の少なくとも 1 つのセグメントがキーなしである場合は、次のように処理します。
  - REPL 操作または DLET 操作の場合、最初のパス (ターゲット・セグメントを含む) を見つけるための検索が実行されます。検索は、以下の両方の条件が当てはまるシャドウ・エリアで、最初のインスタンスが検出されると停止します。
    - 階層パスのすべてのキーなしセグメントにあるデータが、オリジナルの更新が発生する前のこれらのセグメントに対するデータと一致する。
    - オリジナルの更新の連結キーが、返されたセグメントに対するキーと一致する。
  - 一致が検出された後、更新操作が処理されます。
  - ISRT 操作の場合、ターゲット・セグメントの親に対する最初のパスを見つけるための検索が実行されます。検索は、以下の両方の条件が当てはまるシャドウ・エリアで、最初のインスタンスが検出されると停止します。
    - 階層パスのすべてのキーなしセグメントにあるデータが、オリジナルの挿入が発生する前のこれらのセグメントに対するデータと一致する。
    - オリジナルの更新の場合の挿入セグメントの親に対する連結キーが、返された親セグメントのキーと一致する。

ターゲット・セグメントがキーなしで挿入規則が HERE である場合、挿入規則に FIRST が指定されている場合と同様に、挿入が実行されます。それ以外の場合は、定義された挿入規則に従って挿入が実行されます。

要件: 多重エリア・データ・セットの場合、2 つから 6 つのシャドー・エリア・データ・セットを定義することができます。7 つの多重エリア・データ・セットが必要な場合、DEDB 作成ユーティリティーを使用して、DEDB 変更 ALTERAREA 関数の完了後に 7 つ目のエリア・データ・セットを作成します。

## DBFUDA00 ユーティリティーの例

ここでの例は、DBFUDA00 ユーティリティーのサンプル JCL を示しています。

### DEDB 変更ユーティリティーの ALTERAREA のサンプル

DEDB 変更ユーティリティーの以下のサンプル JCL では、DEDBJN21 DEDB データベースのアクティブ・エリア DB21AR0 が変更されます。

```
//ALTAREA JOB ....
//FPUTIL PROC SOUT=A,RGN=1M,
//          DBD=,REST=00,DIRCA=002,
//          PRLD=,IMSID=,AGN=,SSM=,ALTID=
//FPU EXEC PGM=DFSRR00,REGION=&RGN,
//          PARM=(IFP,&DBD,DBF#FPU0,&REST,00,,1,
//          &DIRCA,&PLRD,0,,, &IMSID,&AGN,&SSM,,
//          &ALTID)
//STEPLIB DD DSN=IMS.CRESLIB,DISP=SHR
//PROCLIB DD DSN=IMSVS.PROCLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&SOUT
//SYSUDUMP DDSYSOUT=&SOUT,...
//SO EXEC FPUTIL,RGN=1M,DBD=DEDBJN21,REST=00,IMSID=IMS1
//SYSIN DD *
TYPE ALTER
ALTERAREA DB21AR0
RETRY NO
TIMEOUT 30
GO
/*
```

### DEDB 変更ユーティリティーの REPLRAND のサンプル

DEDB 変更ユーティリティーの以下の JCL は、アクティブな DEDB データベース DEDBJN21 のランダムマイザーを置き換えます。DEDB 変更ユーティリティーが完了すると、ACB メンバー DEDBJN21 のステージング ACBLIB または IMS カタログ内のランダムマイザーが、アクティブな DEDBJN21 DEDB データベースに使用されたランダムマイザーと置き換わります。

```
//ALTAREA JOB ....
//FPUTIL PROC SOUT=A,RGN=1M,
//          DBD=,REST=00,DIRCA=002,
//          PRLD=,IMSID=,AGN=,SSM=,ALTID=
//FPU EXEC PGM=DFSRR00,REGION=&RGN,
//          PARM=(IFP,&DBD,DBF#FPU0,&REST,00,,1,
//          &DIRCA,&PLRD,0,,, &IMSID,&AGN,&SSM,,
//          &ALTID)
//STEPLIB DD DSN=IMS.CRESLIB,DISP=SHR
//PROCLIB DD DSN=IMSVS.PROCLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&SOUT
//SYSUDUMP DD SYSOUT=&SOUT,...
//SO EXEC FPUTIL,RGN=1M,DBD=DEDBJN21,REST=00,IMSID=IMS1
//SYSIN DD *
TYPE ALTER
REPLRAND
```

```
|      RETRY      NO  
|      TIMEOUT    30  
|      GO  
|      /*  
  
|
```



---

## 第 6 章 DEDB 初期設定ユーティリティー (DBFUMIN0)

DEDB 初期設定ユーティリティー (DBFUMIN0) は、DEDB データベースの 1 つ以上の区域の 1 つ以上のアクティブ・データ・セットを初期設定するために使用します。

DEDB 初期設定ユーティリティーは、DEDB ユーティリティーが使用するシャドー・エリア・データ・セットをフォーマットするのにも使用します。シャドー・エリア・データ・セットは、アクティブ DEDB エリア・データ・セットからデータをマイグレーションするために、DEDB 変更プロセスがコピー処理のターゲット・エリア・データ・セットとして使用する DEDB エリア・データ・セットです。このエリア・データ・セットは、シャドー・エリア・データ・セットとして IMS データベース・リカバリー管理 (DBRC) に登録されており、DEDB 変更機能を除いて IMS システムにアクセスすることはできません。

アクティブ・エリア・データ・セットをフォーマットする際は、以下のことに留意してください。

- データ・セットが初期設定され、DEDB 区域が DBDGEN の指定に合わせてフォーマット設定されると、データをロードするための INSERT 呼び出しがユーザー作成プログラムから出されます。
- データのロードが終わったら、データベースのイメージ・コピーを作成します。データにアクセスする前にシステム障害が起こった場合は、リカバリーのためにイメージ・コピーが必要です。イメージ・コピーをとる前にデータにアクセスしシステム障害が起こった場合は、データ・セットを再定義し、DEDB 初期設定ユーティリティーを再実行する必要があります。

サブセクション:

- 『制約事項』
- 66 ページの『前提条件』
- 66 ページの『要件』
- 67 ページの『推奨事項』
- 67 ページの『入出力』
- 67 ページの『戻りコード』
- 67 ページの『JCL 指定』

### 制約事項

DBFUMIN0 を使用するときは、次のような制約があります。

- このユーティリティーは、停止させると、再開できません。必ず最初から実行し直してください。
- オンライン・データベース・イメージ・コピー・ユーティリティーは DEDB をサポートしないので、バッチ・イメージ・コピーが作成されるまでは、DEDB データベースをリカバリーできません。

- 初期設定ユーティリティーは、別のジョブ・ステップで実行しなければなりません。プログラムは、同じジョブ・ステップ内でユーティリティーまたはデータベースを切り替えることはできません。
- DBFUMIN0 ユーティリティーは、高速機能データベース (DEDB) のみで機能します。
- データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。
- このユーティリティーは、アクティブ・エリア・データ・セットとシャドー・データ・セットを同時にフォーマットすることはできません。別々にフォーマットする必要があります。
- 1 次スペース割り振りは複数のボリュームに分割できますが、2 次割り振りとはデータ・セット拡張はサポートされません。

DBFUMIN0 を使用してアクティブ・データ・セットとフォーマットする際には、次の制約も適用されます。

- DEDB データベースは、ACBLIB DD ステートメントにメンバーとして 1 つしか指定できないため、一度に 1 つの DEDB データベースの区域しか初期設定できません。

## 前提条件

DEDB データベースは、z/OS IDCAMS コマンドを使用して割り振る必要があります。

## 要件

DBFUMIN0 ユーティリティーを実行するには、さまざまな操作上の要件を満たす必要があります。

DEDB 初期設定ユーティリティーを使用するときは、以下のような操作上の要件があります。

- ある区域の複数エリア・データ・セットを初期設定するときは、これらのエリア・データ・セットの状況は、RECON データ・セットの中で使用不可の状況になっていなければなりません。
- ある区域のエリア・データ・セットが複数あるときは、オープンの前にそのすべてのエリア・データ・セットを DBFUMIN0 で初期設定しなければなりません。さらに別のエリア・データ・セットを追加するときは、IMS の始動後に区域をオープンする前に、データ・セット作成ユーティリティー (DBFUMRI0) を使用します。
- SMS 管理ストレージ・クラスが高速機能区域に使用され、DEFINE クラスターが複数のボリュームにわたる区域にまたがる場合には、その区域を保証スペースのストレージ・クラスに割り振る必要があります。DBFUMIN0 ユーティリティーは、候補ボリュームに常駐する区域のいかなる部分もフォーマット設定しません。

要件: DD ステートメントは、初期設定する区域および ADS ごとに必要です。

## 推奨事項

現在、DBFUMIN0 ユーティリティに対して文書化された推奨事項はありません。

## 入出力

DEDB 初期設定ユーティリティは次の入力を使用します。

- 特定の DEDB のために生成された ACB (ACBLIB の DBD メンバーへのアクセスが必要です)
- 区域が登録されていて、リカバリーが必要な状況にあるときは、DBRC RECON データ・セット
- 区域が登録されていて、エリア・データ・セットが使用不可状況にあるときは、DBRC RECON データ・セット
- どの区域を初期設定するかを指定している制御データ・セット

このユーティリティは次の出力を作成します。

- 出力メッセージと統計が入っているデータ・セット
- フォーマット設定された区域
- 区域が登録されていて、リカバリーが必要な状況にないときは、DBRC RECON データ・セット
- 区域が登録されていて、エリア・データ・セットが使用可能状況にあるときは、DBRC RECON データ・セット

## 戻りコード

DEDB 初期設定ユーティリティからは、次の戻りコードの 1 つが返されます。

コード 意味

- |    |                     |
|----|---------------------|
| 0  | 初期設定は正常に完了した        |
| 4  | パラメーターのエラー          |
| 8  | ACB または DBRC の処理エラー |
| 12 | データ・セット情報の処理エラー     |
| 16 | フォーマット設定処理時のエラー     |
| 20 | SYSPRINT エラー。       |

## JCL 指定

DBFUMIN0 ユーティリティは、標準の z/OS ジョブとして実行されます。DBFUMIN0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含める必要があります。

次の JCL ステートメントが必要です。

- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

## EXEC ステートメント

EXEC ステートメントは次の形式になっていなければなりません。

```
▶▶—EXEC—PGM=DBFUMINO, PARM=—————▶▶
|                                     |
| 'DBRC=x', 'GSGNAME=xxxxxxx' |
| 'DBRC=x'                       |
| 'GSGNAME=x'                     |
|                                     |
```

DMCB からの DBNAME は、DEDB 区域のパスワードとして使用されます。

EXEC ステートメントで DBRC=N パラメーターが指定されたときは、RECON $n$  DD ステートメントは必要ありません。

EXEC ステートメントで DBRC=Y パラメーターが指定されたときは、動的割り振りを使用している場合を除き、RECON $n$  DD ステートメントが必要です。

EXEC ステートメントの DBRC パラメーターが指定されていないときは、DBRC は、サンプルのインストール・デフォルト・メンバー DFSIDEF0 の DBRC パラメーターに従って設定されます。DFSIDEF0 メンバーにおけるバッチ・サブシステムのデフォルトは DBRC=YES です。DFSIDEF0 サンプル・モジュールを使用しない場合、またはそのモジュールが初期設定時にロードされなかった場合、IMS でデフォルトの DBRC=YES が設定されます。インストール・デフォルト・モジュール DFSIDEF0 に DBRC=FORCE が指定されている場合は、エラー・メッセージ DFS0044I DBRC REQUIRED FOR THIS EXECUTION が表示されます。

RSR が使用されている場合、GSGNAME パラメーターでは RSR グローバル・サービス・グループ名を指定します。

## DD ステートメント

### STEPCAT DD

最初に検索される専用 VSAM ユーザー・カタログを記述します。定義済み領域がユーザー・カタログに登録されている場合は、この DD ステートメントが必要です。

### ACBLIB DD

初期設定するデータベースに関する情報が入っている ACBLIB の DBD メンバーを記述します。

この DD ステートメントは必須です。

### RECON1 DD

最初の DBRC RECON データ・セットを定義します。

### RECON2 DD

2 番目の DBRC RECON データ・セットを定義します。区域が DBRC RECON データ・セットに登録されている場合は、この RECON データ・セットが必要です。

### RECON3 DD

RECON1 または RECON2 にエラーが見付かったとき使用される、オプション



の DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

#### **SYSPRINT DD**

エラー・メッセージ (処理中にエラーが起こった場合) と統計 (ユーティリティーが最後まで正常に完了した場合) を入れるデータ・セットを記述します。

#### **ddname DD (DEDB の場合) または areaname DD**

初期設定する区域ごとにデータ・セットを記述します。区域が RECON データ・セットに登録されていないときは、このステートメントの DD 名は区域名と同じものでなければなりません。

区域が RECON データ・セットに登録されているときは、このステートメントの DD 名は、RECON データ・セットの ADS リストに入っている ADS 名と同じものでなければなりません。ADS の区域は、この区域の ADS がすべて使用不可状況にあるため、リカバリーが必要なものとして設定されている必要があります。

すべてのデータ・セットは、前もって VSAM カタログに定義しておかなければなりません。

DD ステートメントがマルチボリューム・データ・セットを記述しているときは、DISP=OLD、UNIT、および VOL パラメーターを指定しなければなりません。

#### **CONTROL DD**

入力制御ステートメント・データ・セットを記述します。このデータ・セットは論理レコード長が 80 バイトであり、固定長ブロックを持っている必要があります。

### **コマンドの継続**

以下のように懸垂コンマを使用することで、オペランドの間でオペランド欄を継続することができます。

```
STARTSEQ OP='FIELD=FLD1',      (最初の行)
          VALUE=X'C4C5C2'      (継続行)
```

終了の引用符、コンマ、および再オープンの引用符を以下のように使用して、引用符付きストリングを継続します。

```
STARTROOT X'C1C2C3C4C',      (最初の行)
          '5C6C7',            (2 番目の行)
          'C8C9'              (最後の行)
```

懸垂コンマとコメントの間に 1 つのブランクを使用して、各行にコメントを含めることができます。

関連概念:

 RECON データ・セットの初期化および保守 (システム管理)

## DBFUMINO ユーティリティの制御ステートメント

DEDB 初期設定ユーティリティの AREA 制御ステートメントでは、初期設定する区域の名前を指定するか、DEDB 全体を初期設定することを指定します。このステートメントは、CONTROL DD ステートメントで定義したデータ・セットに入れておく必要があります。オプションの 2 つの制御ステートメント ACTIVE と SHADOW は、DBRC=Y を指定した DEDB 初期設定ユーティリティの実行用です。

### コマンドのフォーマット

パラメーターをフリー・フォームで指定します。

例外: フィールドは、最大文字数が 120 文字であることを除いて、何らかの特定の列に限定されません。

各ステートメントは新しい行で始める必要があります。コマンド名は、最初を非ブランク文字で開始し、1 つのブランクまたは等号で終了します。コメントを示すときは、先頭文字にアスタリスクを使用します。

コマンドでオペランドが必要な場合は、次の文字を非ブランクか、等号以外かでオペランド欄を開始します。コマンドで複数のオペランドを使用する場合は、オペランドをコンマで分離します。オペランド欄は、1 つのブランク、または行の最後で終了します。ただし、EBCDIC 値の一部である文字 (引用符で囲まれた文字ストリングとして指定) は、終了コンマまたは終了ブランクとしてカウントしません。

行のオペランド欄に続く文字は、コメントとして処理されます。

スキャン・ユーティリティおよび削除ユーティリティのオペランドは、領域に固有です。複数の領域が各実行で処理される場合、QUITCI、V5COMP、および EXCLUDE などのキーワードを領域ごとに指定する必要があります。これらの各オペランドに適用可能な値は、各 AREA コマンドに対する各 GO コマンドの前にコーディングする必要があります。例えば、ユーティリティの実行ですべての領域に対して V5COMP 関数が必要な場合、各 AREA コマンドに対する各 GO コマンドの前に V5COMP をコーディングする必要があります。

### コマンドの継続

以下のように懸垂コンマを使用することで、オペランドの間でオペランド欄を継続することができます。

```
STARTSEQ OP='FIELD=FLD1',      (最初の行)
          VALUE=X'C4C5C2'      (継続行)
```

終了の引用符、コンマ、および再オープン引用符を以下のように使用して、引用符付きストリングを継続します。

```
STARTROOT X'C1C2C3C4C',      (最初の行)
          '5C6C7',            (2 番目の行)
          'C8C9'              (最後の行)
```

懸垂コンマとコメントの間に 1 つのブランクを使用して、各行にコメントを含めることができます。

区域を指定する場合、1つの制御ステートメントで指定できるのは1つの区域のみです。ただし、複数の制御ステートメントを使用すると、複数の区域を指定することができます。

制御ステートメント **ACTIVE** および **SHADOW** は、**DBRC=Y** を指定して **DEDB** 初期設定ユーティリティ (DBFUMIN0) を実行するのに使用されます。 **ACTIVE** がデフォルトであり、これは **DEDB** 区域のエリア・データ・セットをフォーマットするのに使用します。 **SHADOW** は、**DEDB** 初期設定ユーティリティ (DBFUMIN0) がシャドー・エリア・データ・セットおよびシャドー・イメージ・コピー・データ・セットをフォーマットする場合に指定する必要があります。 **ACTIVE** または **SHADOW** は、制御ステートメント上で単独で指定します。

▶—AREA=—  
          └─areaname─┘  
          └─ALL─┘

▶—  
   └─ACTIVE─┘  
   └─SHADOW─┘

## AREA

### *areaname*

初期設定する区域の名前です。区域名は、DBD ソース内の **AREA** ステートメントの **DD1** パラメーターの定義どおり正確に入力します。

### **ALL**

**DEDB** にあるすべての区域を初期設定することを指定します。 **ALL** を指定すると、以下の制約事項が適用されます。

- **ALL** は、**CONTROL** データ・セットの最初のレコードで指定しなければなりません。
- この制御ステートメントの残りの部分とその後続くすべての制御ステートメントは無視されます。

## **ACTIVE**

オプションの制御ステートメントです。 **DBRC=Y** も指定された場合、**DEDB** 区域のエリア・データ・セットをフォーマットするように指定するか、またはそのようにデフォルトで指定されています。 **ACTIVE** を指定すると、アクティブ・エリア・データ・セットのみがフォーマットされます。シャドー・データ・セットが指定されても無視され、フォーマットされません。 **DEDB** 区域で **DBRC=N** を指定して実行した場合、**ACTIVE** 制御ステートメントは無視されません。 **ACTIVE** がデフォルト値です。

## **SHADOW**

オプションの制御ステートメントであり、**DEDB** 区域のシャドー・エリア・データ・セットおよびシャドー・イメージ・コピー・データ・セットをフォーマットするように指定します。 **SHADOW** を指定すると、シャドー・データ・セットのみがフォーマットされます。非シャドー・データ・セットが指定されても無視され、フォーマットされません。 **DEDB** 区域で **DBRC=N** を指定して実行した場合、**SHADOW** 制御ステートメントは無視されます。

## DBFUMINO ユーティリティーの例

これらの例は、DBFUMINO ユーティリティーのサンプル JCL を示しています。

この節にある例では、桁合わせを容易にするために、CONTROL DD ステートメントの上に次のようなコメント行が示されています。

```
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

このコメント行は単なる参考用です。

この節の例で、使用する DBRC で動的割り振りを行わない場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

図 16. 動的割り振りなしで DBRC を使用するための DD ステートメント

```
//RECON1 DD DSN=RECON1,DISP=SHR
//RECON2 DD DSN=RECON2,DISP=SHR
//RECON3 DD DSN=RECON3,DISP=SHR
```

この例では、DEDB を初期設定するためのサンプル JCL を示しています。

```
//*
/*      INITIALIZE DEDB DATABASE      (DATAB01)
/*
//TESTIT EXEC PGM=DBFUMINO
//ACBLIB  DD DSNAME=IMS.ACBLIB(DATAB01),DISP=SHR
//SYSPRINT DD SYSOUT=A
//DB01AR01 DD DSNAME=DB01AR01,DISP=OLD,
//          VOL=SER=VOL111
//DB01AR02 DD DSNAME=DB01AR02,DISP=OLD,
//          VOL=SER=VOL111
//DB01AR03 DD DSNAME=DB01AR03,DISP=OLD,
//          VOL=SER=VOL222
//DB01AR04 DD DSNAME=DB01AR04,DISP=OLD,
//          VOL=SER=VOL222
//DB01AR05 DD DSNAME=DB01AR05,DISP=OLD,
//          VOL=SER=VOL333
//DB01AR06 DD DSNAME=DB01AR06,DISP=OLD,
//          VOL=SER=VOL333
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7---

//CONTROL DD *
ALL          ALL AREAS INITIALIZED
/*          END OF DBFUMINO
```

この例では、シャドウ ADS をフォーマットします。

```
//DDTJ001B JOB ('A=SJ55'),'1508-LDMOORE',CLASS=K,MSGCLASS=A,
//          MSGLEVEL=(1,1),
// REGION=0M,
// USER=USRT001,PASSWORD=ALL1SDUN
/*
/******
/*
/* This job FORMATS the SHADOW ADS for use by DEDB ALTER */
/*
/* note: We must use the STAGING ACBLIB */
/*
/******
/*
/*ROUTE PRINT THISCPU/IMSSER01
//JOB LIB DD DSN=IMSTESTL.TNUC0,DISP=SHR
```

```

|          //          DD DSN=IMSB LD.I13ATSMM.CRESLIB,DISP=SHR
|          //FORMAT21 EXEC PGM=DBFUMIN0,REGION=0M,TIME=60
|          //ACBLIB DD DSN=IMSTESTL.STAGING.ACBLIB(DED BJ001),
|          //          VOL=SER=USER01,
|          //          UNIT=SYSDA,DISP=SHR
|          //SYS PRINT DD SYSOUT=A
|          //SYSUDUMP DD SYSOUT=A
|          //RECON1 DD DSN=IMSTESTL. IMS.RECON1,DISP=SHR
|          //RECON2 DD DSN=IMSTESTL. IMS.RECON2,DISP=SHR
|          //RECON3 DD DSN=IMSTESTL. IMS.RECON3,DISP=SHR
|          //DBJ1ARS1 DD DSN=IMSTESTL.DEDBJ001.DBJ1AR01.SHADOW,DISP=OLD
|          //DBJ1ARI1 DD DSN=IMSTESTL.DEDBJ001.DBJ1AR01.SHADOW.IC,DISP=OLD
|          //CONTROL DD *
|          AREA=DBJ1AR0
|          SHADOW
|          /*

```



---

## 第 7 章 DEDB 順次従属削除ユーティリティー (DBFUMDL0)

DEDB 順次従属削除ユーティリティー (DBFUMDL0) をオンラインで使用すると、DEDB 区域の特定の限度内で順次従属セグメントを論理的に削除することができます。

従属セグメントが削除されると、ユーティリティーはセグメント境界をリセットし、その区域内で解放されたスペースは再使用が可能になります。

DBFUMDL0 ユーティリティーは、IMS 高速機能領域で実行されるアプリケーションであるかのように、FP ユーティリティー・タイプ従属領域で実行されます。

サブセクション:

- 『制約事項』
- 『前提条件』
- 『要件』
- 76 ページの『推奨事項』
- 76 ページの『入出力』
- 76 ページの『戻りコード』
- 76 ページの『JCL 指定』

### 制約事項

DEDB 順次従属削除ユーティリティーは、別のジョブ・ステップで実行しなければなりません。プログラムは、同じジョブ・ステップ内でユーティリティーまたはデータベースを切り替えることはできません。

DEDB 順次従属削除ユーティリティーは再始動できません。再始動 (REST) パラメーターを使用した場合、ユーティリティーは、その再始動パラメーターで定義された区域の先頭から初期実行を試みます。

DBFUMDL0 ユーティリティーは、高速機能データベース (DEDB) のみで機能します。

データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。

### 前提条件

現在、DBFUMDL0 ユーティリティーに対して文書化された前提条件はありません。

### 要件

現在、DBFUMDL0 ユーティリティーに対して文書化された要件はありません。

## 推奨事項

現在、DBFUMDL0 ユーティリティに対して文書化された推奨事項はありません。

## 入出力

DEDB 順次従属削除ユーティリティへの入力、コマンドで指定された入力パラメーターが入っているデータ・セットから成り立っています。

DEDB 順次従属削除ユーティリティからは、次のような出力が得られます。

- 解放されたスペースを持つエリア
- 出力メッセージと統計が入っているデータ・セット

## 戻りコード

現在、DBFUMDL0 ユーティリティに対して文書化された戻りコードはありません。

## JCL 指定

DBFUMDL0 ユーティリティは、標準の z/OS ジョブとして実行されます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含める必要があります。

次の JCL ステートメントが必要です。

- JOB ステートメント
- EXEC ステートメント
- 入力および出力を指定する DD ステートメント

### EXEC ステートメント

このステートメントは、必要な JCL が入っている FPUTIL プロシーチャーを指定することができますが、次の形式にすることもできます。

```
PGM=DFSRR00
```

### DD ステートメント

#### STEPLIB DD

DEDB 順次従属削除ユーティリティが入っているライブラリーを記述します。

#### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

#### SYSIN DD

DBFUMDL0 ユーティリティの処理オプションを制御する DEDB ユーティリティ・コマンドが入っている入力制御データ・セットを記述します。

#### SYSPRINT DD

メッセージと統計が入っている出力データ・セットを記述します。

関連資料:



## DBFUMDL0 ユーティリティーの制御ステートメント

DBFUMDL0 ユーティリティーは、その処理オプションを定義するために、DEDB ユーティリティー・コマンドを使用します。DEDB ユーティリティー・コマンドは、SYSIN DD ステートメントによって DBFUMDL0 ユーティリティーに割り振られた制御データ・セットに指定されます。

### コマンドのフォーマット

パラメーターをフリー・フォームで指定します。

例外: フィールドは、最大文字数が 120 文字であることを除いて、何らかの特定の列に限定されません。

各ステートメントは新しい行で始める必要があります。コマンド名は、最初を非ブランク文字で開始し、1 つのブランクまたは等号で終了します。コメントを示すときは、先頭文字にアスタリスクを使用します。

コマンドでオペランドが必要な場合は、次の文字を非ブランクか、等号以外かでオペランド欄を開始します。コマンドで複数のオペランドを使用する場合は、オペランドをコンマで分離します。オペランド欄は、1 つのブランク、または行の最後で終了します。ただし、EBCDIC 値の一部である文字 (引用符で囲まれた文字ストリングとして指定) は、終了コンマまたは終了ブランクとしてカウントしません。

行のオペランド欄に続く文字は、コメントとして処理されます。

スキャン・ユーティリティーおよび削除ユーティリティーのオペランドは、領域に固有です。複数の領域が各実行で処理される場合、QUITCI、V5COMP、および EXCLUDE などのキーワードを領域ごとに指定する必要があります。これらの各オペランドに適用可能な値は、各 AREA コマンドに対する各 GO コマンドの前にコーディングする必要があります。例えば、ユーティリティーの実行ですべての領域に対して V5COMP 関数が必要な場合、各 AREA コマンドに対する各 GO コマンドの前に V5COMP をコーディングする必要があります。

### コマンドの継続

以下のように懸垂コンマを使用することで、オペランドの間でオペランド欄を継続することができます。

```
STARTSEQ OP='FIELD=FLD1',      (最初の行)
          VALUE=X'C4C5C2'      (継続行)
```

終了の引用符、コンマ、および再オープン引用符を以下のように使用して、引用符付きストリングを継続します。

```
STARTROOT X'C1C2C3C4C',      (最初の行)
           '5C6C7',          (2 番目の行)
           'C8C9'            (最後の行)
```

懸垂コンマとコメントの間に 1 つのブランクを使用して、各行にコメントを含めることができます。

SYSIN データ・セットには、以下の DEDB ユーティリティー・コマンドを入れることができます。

#### **ALLFMNEW (オプション)**

区域内のすべての CI が、第 6 版以降の形式を使用することを指示します。IMS が IMS 第 5 版およびそれ以前の形式の CI を検出した場合は、ユーティリティーは異常終了します。このパラメーターは、古い形式の CI がないかを確認するために使用することができます。SORTSETUP を使用した場合は、SDEP セクションに古い形式のセグメントが残されていないことを確認したときにのみ、ALLFMNEW を指定してください。

#### **AREA (必須)**

どの区域を処理するのかを、制御領域 DD ステートメント上の名前指定します。このコマンドは、各 GO コマンドまたは各 RUN コマンドの後で繰り返さなければなりません。名前は 1 文字から 8 文字までにしなければなりません。最初の文字は英字にし、残りの文字は英数字にします。英字には、@、#、および \$ が含まれます。

#### **BUFNO (オプション)**

DEDB を読み書きするために使用するバッファの数を指定します。再編成ユーティリティー以外のユーティリティーの場合は、少なくとも 7 つのバッファが必要です。BUFNO の指定がない場合は、デフォルトが使用されます。デフォルトのバッファ数は、作業単位 (UOW) 当たりの制御インターバル (CI) 数に 7 を加えたものです。

#### **ERRORACTION (オプション)**

エラーが検出され、エラー・メッセージが印刷されたときに、指定のユーティリティーにとらせるアクションを指定します。このコマンドはオプションですが、必要な回数だけ指定することができます。ERRORACTION が指定されていないと、デフォルトとして STOP オペランドが使用されます。

#### **STOP**

ユーティリティーを即時に停止させます。

#### **ABEND**

ユーティリティーが U1039 異常終了ダンプを作成することを指定します。

#### **SCAN**

ユーティリティーがエラーの有無を調べるために入力のスキャンを続けることを指定します。

#### **SCANRUN**

ユーティリティーが、次の GO コマンドが現れた後で検出されたエラーを無視することを除けば、SCAN オペランドと同じ処理を指定します。

#### **GO (オプション)**

一連の要求を区切るために使用します。2 つ以上の区域の処理、2 つ以上の出口ルーチンの使用、あるいは 2 つ以上のデータ範囲の処理を 1 つのジョブ・ステップで行う場合は、必ず指定する必要があります。

#### **LASTSCAN (オプション)**

直前の SCAN を除き、その区域で他のユーティリティーが実行されなかった場合に、停止値の置換文字として機能します。LASTSCAN は DELETE の停止点として、SCAN の停止点を使用します。

### NSQCI (オプション)

IMS パートナーが、次のコミット・インターバルの間、現行 SDEP CI およびすべての事前割り振り CI RBA を放棄することを指定します。このオプションは、すべての共用 IMS システムが同じペースで挿入を実行している場合にのみ適します。

### QUITCI (オプション)

すべての IMS パートナーが、現行 SDEP CI およびすべての事前割り振り CI RBA を即時に放棄することを指定します。このオプションは、トランザクションの処理量が少ないため、事前割り振り CI を埋めるのに長時間を要する IMS パートナーがある場合に役立ちます。QUITCI は、使用中だが一部しか埋められていない CI をすべて強制的に ADS に書き込んでから、解放します。その結果、削除ユーティリティーは CI 全体を処理し、論理開始 (DMACXVAL) を可能な限り前方に進めることができるため、SDEP が循環するときに十分なスペースが使用可能になります。QUITCI パラメーターを使用することにより、新規の SDEP が QUITCI に 1 つも挿入されないことが保証されます。したがって、次のユーティリティーの実行は、デフォルトの STOP が使用されることを想定して、最後の UHWM の後の CI から開始することができます。

### STOPHEXT (オプション)

停止時刻を 16 進形式で指定します。16 進数タイム・スタンプを使用することで、より詳細に指定できます。最新の SDEP SEGMENT の 16 進数タイム・スタンプは、POS 呼び出しを使用して入手できます。

希望するストア・クロック値に一致する 16 桁の 16 進数字を指定する必要があります。STOPHEXT の構文は次のとおりです。

```
STOPHEXT X'C5C55A5DB883FE33' /* 16 hex digits required. */  
Note: C5C55A5DB883FE33 = 2010.092 18:44:05.131.327.875
```

STARTHEXT および STOPHEXT は同じタイム・スタンプ・フォーマットを使用し、その詳細度は STARTIME および STOPTIME に比べはるかに高いものです。

### STOPRBA (オプション)

停止 RBA を指定します。規則は、STARTRBA コマンドを使用して開始 RBA を指定する場合の、上記規則と同じです。このオプションのコマンドは、スキャン・ユーティリティーまたは削除ユーティリティーで使用します。16 進数値は、X'16 進数 形式の値です。STOPRBA コマンドに指定されたアドレスは、SDEP 境界になければなりません。そうしないと、ユーティリティーが異常終了します。

### EXCLUDE

データベース・スキャン・ユーティリティーが、STOPRBA の SDEP セグメントまで (ただし SDEP セグメント自身は除く) を抽出することを指定します。このとき、データベース削除ユーティリティーは、フィールド DMACLBTS を STOPRBA+1 ではなく STOPRBA に設定します。

### STOPROOT (オプション)

ルート・キー・フィールド値を指定します。このオプションのコマンドは、スキャン・ユーティリティーまたは削除ユーティリティーで使用します。値は、16 進値、文字値、またはパック値で指定します。値は、英字 (X、C、または P)

で始め、引用符で囲んだストリングを後に続けます。例えば、X'2A1B'、C'AIN''T'、または P'-00199'。値は、長さとおよび内容の両方が評価されません。

#### **STOPSEQ (オプション)**

停止 RBA を探すときに使用される順次従属セグメントを指定します。このオプションのコマンドは、スキャン・ユーティリティーまたは削除ユーティリティーで使用します。

#### **FIELD=(name2)**

セグメント検索指数 (SSA) に指定されているフィールド名です。

名前は 1 文字から 8 文字までにしなければなりません。文字は、英字または数値を使用します。英字には、@、#、および \$ が含まれます。

#### **OP=(operator)**

SSA に指定されている比較演算子です。

#### **VALUE=(value)**

SSA に指定されているフィールド値です。

FIELD、OP、および VALUE のフィールドは、STOPROOT データと一緒に使用されて、停止 RBA を探し出します。

#### **STOPTIME (オプション)**

明示的な停止時刻を次の形式で指定します。YYYY.DDD HH:MM:SS ±hh:mm。ここで、YYYY= 年、DDD= 年間通算日、HH= 時、MM= 分、SS= 秒、および hh:mm= UTC へのオフセット (UTC に加算すると現地時間が得られるもの) です。hh:mm= オフセット・フィールドは、必ず指定する必要があります。このストリングの形式は固定しています。例えば、SS= 秒パラメーターと hh:mm= オフセット・パラメーターの間には、次のようにブランク・スペースを 1 個しか置くことができません。

```
TYPE    SCAN
AREA    <area name>
STOPTIME=C'2010.092 18:40:05 -00:00'
```

各 z/OS オペレーティング・システムは、複合システム内の他のパートナー z/OS オペレーティング・システムと同じ時刻にクロックを設定する必要があります。そうしないと、ストア・クロック形式に変換した値が、z/OS オペレーティング・システム・パートナー間で同じになりません。

#### **TYPE (必須)**

実行のタイプとして、スキャン、削除、または再編成のいずれかのユーティリティーを指定します。

**重要:** このコマンドは必須であり、最初の GO コマンドの前か、あるいは SYSIN ファイルの終わりの前に入れておかなければなりません。

プログラムは同じジョブ・ステップ内でユーティリティーを切り替えることができないため、ジョブ・ステップ内に指定できる TYPE コマンドは 1 つだけです。

#### **V5COMP**

スキャンおよび削除のすべての操作が、IMS 第 5 版の規則に従うことを指示します。それには以下の規則が含まれます。

- START および STOP を、タイム・スタンプにかかわらず、セグメント境界に置きます。
- スキャンでは、START パラメーターで指定されたセグメントから読み取りを開始します。DMACXVAL CI と START セグメントを含む CI の間にある CI は読み取りません。
- 削除では、論理終了を判別するために、STOP CI だけを読み取ります。DMACXVAL CI からは読み取りません。

V5COMP を使用すると、IMS 第 5 版のものに等しいユーティリティーから結果を作成することができます。パフォーマンスも同じようになります。ただし、スキャンおよび削除されるセグメントのセットは、CI 間にタイム・スタンプが混じっているために、共用 SDEP と同じにはなりません。

関連資料:

88 ページの『DBFUMSC0 ユーティリティーの制御ステートメント』

---

## DBFUMDL0 ユーティリティーの例

これらの例は、DBFUMDL0 ユーティリティーのサンプル JCL を示しています。

### データベース・スキャン・ユーティリティーのサンプル JCL

以下の図は、次のものを示しています。

- データベース・スキャン・ユーティリティーの JCL
- データベース削除ユーティリティーを実行するためのユーティリティー制御ステートメント。

この 2 つのユーティリティーを使用すると、データベース・スキャン・ユーティリティーはデータを取得し、データベース削除ユーティリティーはそのデータを除去します。

```
//SCAN EXEC FPUTIL,
//      DBD=DEDBJN02,REST=00
//*
//*      DBD=DBDNAME AS TARGET DATABASE FOR THIS UTILITY RUN
//*      REST=RESTART NUMBER FOR THIS RUN
//*
//SCANCOPY DD DSNAME=SCAN202,DISP=(NEW,PASS,DELETE),
//          UNIT=SYSDA,VOL=SER=IMSDCL,
//          DCB=(NCP=5,BLKSIZE=2048),
//          SPACE=(TRK,5)
//SYSIN DD *
```

```
*-----*
```

```
*          THE STOPRBA  COMMAND WILL LIMIT THE RANGE OF THE  *
*          SCAN.                                             *
```

```
*-----*
```

```
*
* COMMAND      OPERATOR      COMMENT
*
*              ERRORACTION SCANRUN      SET ERROR OPTION
*              TYPE          SCAN        ONLINE SEQ DEP SCAN UTILITY
*
*              THE TARGET DATABASE IS
*              DBDNAME=DEDBJN02
*
*              THE TARGET DEDB AREA IS
*              AREA          DB2AREA1
```

```

*                               STOP ON RBA
STOPRBA      X'29E98' LAST SEQ DEP RBA (102A03-2)
*                               THE EXIT PROGRAM NAME IS
EXIT         EXITLDM3
*

```

注: システムで使用されている DFSORT DD カードによっては、SYSOUT DD カードを組み込む必要がある場合もあります。

## データベース削除ユーティリティーのサンプル JCL

```

//DELETE EXEC FPUTIL,
//          DBD=DEDBJN02,REST=00
//*
//*          DBD=DBDNAME AS TARGET DATABASE FOR THIS UTILITY RUN
//*          REST=RESTART NUMBER FOR THIS RUN
//*
//SYSIN    DD *
*-----*
*          DELETE DEDB JN02 USING THE 'STOPROOT' TO LIMIT *
*          THE SCOPE OF THE DELETE. *
*-----*
*
* COMMAND      OPERATOR      COMMENT
*
*              SET ERROR OPTION
ERRORACTION SCANRUN
              ONLINE SEQ DEP DELETE UTILITY
TYPE          DELETE
*              THE TARGET DATABASE IS
*              DBDNAME=DEDBJN02
*              THE TARGET DEDB AREA IS
AREA          DB2AREA1
*              STOP ON ROOT SEGMENT KEY
STOPROOT=C'R211304D'
*              LAST ROOT WITH A SEQ DEP SEGMENT

```

---

## DBFUMDL0 ユーティリティーの処理の範囲

順次従属 (SDEP)は、最も古い現行 SDEP から削除していかねばなりません。そのため、開始オプション・パラメーターはありません。

STOPRBA、STOPROOT、および STOPSEQ コマンドを使用すると、削除する順次従属セグメントの限度を指定できます。従属セグメントの限度 (STOPRBA コマンド) が指定されていないと、デフォルトの終了処理が呼び出されます。最高水準点 (HWM) CI では、永続デフォルト終了タイム・スタンプ・セグメントが作成されます。削除は、区域内の最も古い順次従属 CI 境界から開始されます。削除は HWM CI のデフォルトの終了セグメントからのタイム・スタンプで終了しますが、このタイム・スタンプは削除の対象から除外されています。

デフォルトの終了について V5COMP が指定される場合、以下のシーケンスが実行されます。

- 削除範囲は、区域内の最も古い順次従属 CI 境界から開始されます。
- 論理開始は、DMACNXTS (次に割り振られる SDEP CI) に進み、事前割り振りされた SDEP CI および現行 SDEP CI はすべて廃棄されます。この組み合わせを使用すると、DEDB の SDEP 部分は空になります。

STOPRBA コマンドは、セグメントの 4 バイト RBA で、あるいはサイクル番号と RBA の 8 バイトの組み合わせで停止する場合に使用します。DEDB は、サイクル番号を RBA の接頭部として使用して、区域が存続している間に一度だけ使用される値を得ます。STOPRBA コマンドに指定されたアドレスは、SDEP 境界になればなりません。そうしないと、ユーティリティーが異常終了します。

サイクル番号が非ゼロ値で指定されると、削除ユーティリティーは、それが当該の RBA の現行サイクル番号と一致しているかどうかを確認します。サイクル番号がその RBA の現行サイクル番号と同じでないか、あるいは順次従属セグメントが現在保管されている範囲内に RBA がない場合、エラー・メッセージが印刷され、データは削除されません。サイクル番号が指定されていないか、あるいはゼロと指定されている場合は、その RBA の現行サイクル番号が使用されます。指定された RBA がセグメントの始まりのアドレスでない場合、削除は次のセグメントから開始されます。

#### STOPRBA CI 境界が指定される場合

- CI 内の最初のセグメント・タイム・スタンプが停止タイム・スタンプとして使用されます。
- V5COMP では、CI は除外され、タイム・スタンプは無視されます。

注: STOPRBA が HWM CI または事前割り振りされた SDEP CI (HWM セット内) に等しい CI 境界であり、他のユーザーも QUITCI を指定する場合は、SDEP DELETE ユーティリティーは、HWM CI を通過して論理開始を移動するように指示されます。

#### STOPRBA セグメント境界が指定される場合

- 指定されたセグメントは、タイム・スタンプ属性とロケーション属性のコミット済みマーカー・セグメントのような役割を果たします。
- V5COMP では、指定されたセグメントは停止点であり、タイム・スタンプは無視されます。

#### デフォルトの終了が使用される場合

- HWM CI のデフォルト終了セグメント・タイム・スタンプ未満のタイム・スタンプを持つすべてのセグメントが含まれます。
- V5COMP では、論理開始は DMACNXTS (次に割り振られる SDEP CI) に設定され、メッセージ DFS2637I が出されます。

STOPROOT および STOPSEQ コマンドを使用して停止 RBA を指定したときは、マーカー・セグメントが使用されるものと見なされます。マーカー・セグメントとは、固有のフィールド値を持つ特殊な順次従属セグメントです。マーカー・セグメントを維持するには、マーカー・セグメントを順次従属として挿入するアプリケーション・プログラムを実行します。

挿入されたマーカー・セグメントを持つルート・セグメントを、このユーティリティーに対して指定する必要があります。これを行うには、STOPROOT コマンドでルート・セグメントのキーを指定します。順次従属がルートに対して最後に挿入された順次従属でない場合は、必要とする従属を検索するためのセグメント検索指数 (SSA) を作成するのにユーティリティーが使用する情報を指定してください。SSA を作成するには、フィールド名、比較演算子、およびフィールド値を STOPSEQ コ

マンドに指定します。比較の規則は、通常の DEDB SSA の場合と同じです。指定するフィールド値は、DEDB 内のフィールドと同じ長さにしなればなりません。ルートが指示された区域に存在しないか、あるいは指示されたセグメントが見付からない場合は、エラー・メッセージが印刷され、データは削除されません。



---

## 第 8 章 DEDB 順次従属スキャン・ユーティリティー (DBFUMSC0)

DEDB 順次従属スキャン・ユーティリティー (DBFUMSC0) をオンラインで使用すると、順次従属セグメントをスキャンし、それを順次データ・セットにコピーしてから、ユーザー独自のプログラムを使用してオフラインでこのデータ・セットを処理することができます。

このユーティリティーは、DB ユーティリティー・タイプ従属領域で実行されます。

DBFUMSC0 ユーティリティーは出口ルーチンを使用します。IMS とともに提供されるサンプル出口ルーチンは、高速処理データベース順次従属スキャン・ユーティリティーの出口ルーチン (DBFUMSE1) です。

このユーティリティーを使用すると、次に挙げることを行うことができます。

- コピーするセグメントの範囲を指定できます。セグメントの長さが SCANCOPY DD データ・セットのブロック・サイズから 8 バイト差し引いたものを超えると、ユーティリティーは終了し、エラー・メッセージが出されます。出口ルーチンが指定されていない場合、スキャン・ユーティリティーはデフォルトを使用するため、指定された範囲のセグメント内容がそのまま渡されます。範囲の限度が指定されていないと、すべての従属セグメントがスキャンされ、コピーされます。
- セグメント内容と長さを変更できるIMS によって指定されるものを除いた出口ルーチンを指定できます。
- 次に挙げることを行うロード・モジュールを指定できます。
  - ソート基準を変更する。
  - 特定のレコード番号および平均レコード・サイズでソートする。
  - ソート用に特定のタイプのワークスペースを使用する。
  - ソート基準をまったく使用しないことを指定する。CI サブセットは、ソートなしで処理されます。

IMS の /STOP REGION コマンドを使用すると、DBFUMSC0 ユーティリティーを区域の終わりで停止させることができます。

DBFUMSC0 ユーティリティーは再始動できません。再始動 (REST) パラメーターを指定して DBFUMSC0 ユーティリティーを再始動しようとした場合、ユーティリティーは、その再始動パラメーターで定義された区域の先頭から初期実行を試みます。

サブセクション:

- 86 ページの『制約事項』
- 86 ページの『前提条件』
- 86 ページの『要件』

- 『推奨事項』
- 『入出力』
- 87 ページの『戻りコード』
- 87 ページの『JCL 指定』

## 制約事項

DEDB 順次従属スキャン・ユーティリティーは、別のジョブ・ステップで実行しなければなりません。プログラムは、同じジョブ・ステップ内でユーティリティーまたはデータベースを切り替えることはできません。

このユーティリティーは、未確定のセグメントが現れて、SYSIN 制御ステートメントで INDOUBT が指定されていないと、失敗します。

IMS 用にフォーマット設定された CI 順次セグメントを含むデータは、各 IMS パートナーに割り振られます。フォーマット設定済み CI 順次セグメントの停止位置は、特定のセグメントの終わりではなく CI の境界になります。これにより、後続のユーティリティーは同じ CI および RBA 停止点を指定して、まったく同じセグメントを処理できます。

DBFUMSC0 ユーティリティーは再始動できません。

DBFUMSC0 ユーティリティーは、高速機能データベース (DEDB) のみで機能します。

データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。

## 前提条件

現在、DBFUMSC0 ユーティリティーに対して文書化された前提条件はありません。

## 要件

現在、DBFUMSC0 ユーティリティーに対して文書化された要件はありません。

## 推奨事項

現在、DBFUMSC0 ユーティリティーに対して文書化された推奨事項はありません。

## 入出力

入力については、DBFUMSC0 ユーティリティーは、DEDB ユーティリティー・コマンドで指定された入力パラメーターが入っている制御データ・セットを使用します。SYSIN DD ステートメントがこの制御データ・セットを指します。

DBFUMSC0 ユーティリティーは、以下の出力を生成することができます。

- 順次従属セグメントのコピーが入っているデータ・セット (出口ルーチンを使用すると、このデータ・セットを特定のセグメントに制限することができます)

- 未確定セグメントが入っているデータ・セット (これらのセグメントは出口ルーチンに渡されません)
- 出力メッセージと統計が入っているデータ・セット

## 戻りコード

DBFUMSC0 ユーティリティは戻りコードを出しません。

## JCL 指定

DBFUMSC0 ユーティリティは、標準の z/OS ジョブとして実行されます。

DBFUMSC0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。DBFUMSC0 ユーティリティの処理オプションは、このユーティリティに割り振った制御データ・セットの DEDB ユーティリティ・コマンドによって指定されます。

次の JCL ステートメントが必要です。

- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

DBFUMSC0 ユーティリティには、ソート・ワークスペースを提供するためのソート作業データ定義 JCL ステートメントを入れることができます。ソート作業 DD ステートメントは、ソート・プログラムによって動的に割り振ることができます。SYSPRINT に対してソート・プログラムによって使用される SYSOUT DD ステートメントは、呼び出し元から動的に割り振り、ソート・プログラムに渡すことができます。SORTSETUP *exit\_routine\_name* パラメーター・ステートメントによって、要件を調整するための出口ルーチンを提供することができます。DBFUMSC0 ユーティリティのデフォルト設定では、順次従属セグメントをソートします。

### EXEC ステートメント

EXEC ステートメントは、DEDB 順次従属スキャン・ユーティリティを実行させます。このステートメントは、必要な JCL が入っている FPUTIL プロシーチャーを指定することができますが、次の形式にすることもできます。

```
PGM=DFSRR00
```

### DD ステートメント

#### STEPLIB DD

DBFUMSC0 ユーティリティが入っているライブラリーを記述します。

#### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

#### SCANIDT DD

INDOUBT が SYSIN 制御ステートメントで指定されていれば、未確定セグメントを SCANIDT データ・セットに書き込むことを指定します。

#### SYSIN DD

DBFUMSC0 ユーティリティの処理オプションを制御する DEDB ユーティリ

ティー・コマンドが入っている入力制御データ・セットを記述します。AREA XXXXXXXX を SYSIN DD カードとして指定して、スキャンする区域を選択します。

#### **SYSPRINT DD**

出力メッセージと統計が入っている出力データ・セットを記述します。


#### **SCANCOPY DD**

順次従属セグメントが可変長ブロック化レコード形式で入っているスキャン出力データ・セットを記述します。


制約事項: これを SYSOUT データ・セットにすることはできません。


SCANCOPY データ・セットが 1 つ作成されます。ここには、1 つの区域または複数の区域のスキャンで得られた出力が入っています。

関連概念:

 IMS 出口ルーチン作成のためのガイドライン (出口ルーチン)

関連資料:

 高速処理データベース順次従属スキャン・ユーティリティー出口ルーチン (DBFUMSE1) (出口ルーチン)

 FPUTIL プロシージャ (システム定義)

---

## **DBFUMSC0 ユーティリティーの制御ステートメント**

DBFUMSC0 ユーティリティーは、その処理オプションを指定するために、DEDB ユーティリティー・コマンドを使用します。SYSIN DD ステートメントが指す制御データ・セットにコマンドをコーディングしてください。

### **コマンドのフォーマット**

パラメーターをフリー・フォームで指定します。

例外: フィールドは、最大文字数が 120 文字であることを除いて、何らかの特定の列に限定されません。

各ステートメントは新しい行で始める必要があります。コマンド名は、最初を非空白文字で開始し、1 つの空白または等号で終了します。コメントを示すときは、先頭文字にアスタリスクを使用します。

コマンドでオペランドが必要な場合は、次の文字を非空白か、等号以外かでオペランド欄を開始します。コマンドで複数のオペランドを使用する場合は、オペランドをコンマで分離します。オペランド欄は、1 つの空白、または行の最後で終了します。ただし、EBCDIC 値の一部である文字 (引用符で囲まれた文字ストリングとして指定) は、終了コンマまたは終了空白としてカウントしません。

行のオペランド欄に続く文字は、コメントとして処理されます。

スキャン・ユーティリティーおよび削除ユーティリティーのオペランドは、領域に固有です。複数の領域が各実行で処理される場合、QUITCI、V5COMP、および EXCLUDE などのキーワードを領域ごとに指定する必要があります。これらの各オ

ペランドに適用可能な値は、各 AREA コマンドに対する各 GO コマンドの前にコーディングする必要があります。例えば、ユーティリティーの実行ですべての領域に対して V5COMP 関数が必要な場合、各 AREA コマンドに対する各 GO コマンドの前に V5COMP をコーディングする必要があります。

## コマンドの継続

以下のように懸垂コンマを使用することで、オペラントの間でオペラント欄を継続することができます。

```
STARTSEQ OP='FIELD=FLD1',      (最初の行)
          VALUE=X'C4C5C2'      (継続行)
```

終了の引用符、コンマ、および再オープン引用符を以下のように使用して、引用符付きストリングを継続します。

```
STARTROOT X'C1C2C3C4C',      (最初の行)
          '5C6C7',            (2 番目の行)
          'C8C9'              (最後の行)
```

懸垂コンマとコメントの間に 1 つのブランクを使用して、各行にコメントを含めることができます。

DBFUMSC0 が使用する必須の DEDB ユーティリティー・コマンドは、AREA および TYPE です。

### ALLFMNEW (オプション)

区域内のすべての CI が、第 6 版以降の形式を使用することを指示します。IMS が IMS 第 5 版およびそれ以前の形式の CI を検出した場合は、ユーティリティーは異常終了します。このパラメーターは、古い形式の CI がないかを確認するために使用することができます。

SORTSETUP を使用した場合は、SDEP セクションに古い形式のセグメントが残されていないことを確認したときのみ、ALLFMNEW を指定してください。

### AREA (必須)

どの区域を処理するのかを、制御領域 DD ステートメント上の名前指定します。このコマンドは、各 GO コマンドまたは各 RUN コマンドの後で繰り返さなければなりません。名前は 1 文字から 8 文字までにしなければなりません。最初の文字は英字にし、残りの文字は英数字にします。英字には、@、#、および \$ が含まれます。

### ERRORACTION (オプション)

エラーが検出され、エラー・メッセージが印刷されたときに、指定のユーティリティーにとらせるアクションを指定します。このコマンドはオプションですが、必要な回数だけ指定することができます。ERRORACTION が指定されていないと、デフォルトとして STOP オペラントが使用されます。

### STOP

ユーティリティーを即時に停止させます。

### ABEND

ユーティリティーが U1039 異常終了ダンプを作成することを指定します。

## SCAN

ユーティリティーがエラーの有無を調べるために入力のスキャンを続けることを指定します。

## SCANRUN

ユーティリティーが、次の GO コマンドが現れた後で検出されたエラーを無視することを除けば、SCAN オペランドと同じ処理を指定します。

## EXIT (オプション)

ユーザー出口ルーチンをロード・モジュール名で指定します。このコマンドはオプションですが、スキャン・ユーティリティーのときだけ使用します。名前は 1 文字から 8 文字までにしなければなりません。最初の文字は英字にし、残りの文字は英数字にします。英字には、@、#、および \$ が含まれます。

## EXPANDSEG (オプション)

セグメント拡張を行う (例えば、STEPLIB 連結で) ことを指定します。

## GO (オプション)

一連の要求を区切るために使用します。2 つ以上の区域の処理、2 つ以上の出口ルーチンの使用、あるいは 2 つ以上のデータ範囲の処理を 1 つのジョブ・ステップで行う場合は、必ず指定する必要があります。

## INDOUBT (オプション)

未確定セグメントの RBA を SYSPRINT 出力データ・セットに書き込むことを指定します。

## NOSORT (オプション)

順次従属セグメントをソートしないことを指定します。これらのセグメントはユーザー・セグメントに直接渡されます。区域の SHARELVL が現在 0 または 1 の場合は、SORT は呼び出されません。区域の SHARELVL が既に 2 または 3 になっている場合は、個別の SDEP が順不同に戻される可能性があります。NOSORT は NS または NOS に省略することができます。

## NSQCI (オプション)

IMS パートナーが、次のコミット・インターバルの間、現行 SDEP CI およびすべての事前割り振り CI RBA を放棄することを指定します。このオプションは、すべての共用 IMS システムが同じペースで挿入を実行している場合にのみ適します。

## QUITCI (オプション)

すべての IMS パートナーが、現行 SDEP CI およびすべての事前割り振り CI RBA を即時に放棄することを指定します。このオプションは、トランザクションの処理量が少ないため、事前割り振り CI を埋めるのに長時間を要する IMS パートナーがある場合に役立ちます。QUITCI は、使用中だが一部しか埋められていない CI をすべて強制的に ADS に書き込んでから、解放します。その結果、削除ユーティリティーは CI 全体を処理し、論理開始 (DMACXVAL) を可能な限り前方に進めることができるため、SDEP が循環するときに十分なスペースが使用可能になります。QUITCI パラメーターを使用することにより、新規の SDEP が QUITCI に 1 つも挿入されないことが保証されます。したがって、次のユーティリティーの実行は、デフォルトの STOP が使用されることを想定して、最後の UHWM の後の CI から開始することができます。

### **SORTSETUP (オプション)**

SORT が呼び出される前に呼び出すユーザー作成ルーチンの名前を指定して、セグメントのタイム・スタンプの代わりに各種の SORT パラメーターが渡されるようにします。DBFUMSC0 のソース・コードには、デフォルトのソート設定と、ソート入力およびソート出力出口が含まれます。

### **STARTHEXT (オプション)**

開始時刻を 16 進形式で指定します。16 進数タイム・スタンプを使用することで、より詳細に指定できます。最新の SDEP SEGMENT の 16 進数タイム・スタンプは、POS 呼び出しを使用して入手できます。

希望するストア・クロック値に一致する 16 桁の 16 進数字を指定する必要があります。STARTHEXT の構文は次のとおりです。

```
STARTHEXT X'C5C55972CA04D000' /* 16 hex digits required. */  
Note: C5C55972CA04D000 = 2010.092 18:39:58.787.661.000
```

STARTHEXT および STOPHEXT は同じタイム・スタンプ・フォーマットを使用し、その詳細度は STARTIME および STOPTIME に比べはるかに高いものです。

### **STARTIME (オプション)**

明示的な開始時刻を次の形式で指定します。YYYY.DDD HH:MM:SS ±hh:mm。ここで、YYYY= 年、DDD= 年間通算日、HH= 時、MM= 分、SS= 秒、および hh:mm= UTC へのオフセット (UTC に加算すると現地時間が得られるもの) です。この開始時刻はストア・クロック形式に変更されます。

hh:mm= オフセット・フィールドは、必ず指定する必要があります。このストリングの形式は固定しています。例えば、SS= 秒パラメーターと hh:mm= オフセット・パラメーターの間には、次のようにブランク・スペースを 1 個しか置くことができません。

```
TYPE SCAN  
AREA <area name>  
STARTIME=C'2010.092 18:39:58 -00:00'
```

STARTIME および STOPTIME は同じタイム・スタンプ・フォーマットを使用し、秒までの詳細度しかありません。

各 z/OS オペレーティング・システム・パートナーは、複合システム内の他のパートナー z/OS オペレーティング・システムと同じ時刻にクロックを設定する必要があります。そうしないと、ストア・クロックへ変換した値が、z/OS オペレーティング・システム・パートナー間で同じになりません。

### **STARTRBA (オプション)**

順次従属アドレス情報を最大 8 バイト (16 桁) で指定します。下位 4 バイトには、順次従属の処理を開始する区域内の相対バイト・アドレスを指定します。上位 4 バイトはオプションであり、サイクル番号を指定するときに使用されます。STARTRBA はオプション・コマンドですが、スキャン・ユーティリティで使われます。

### **STARTROOT (オプション)**

開始 RBA を探すときに使用されるルートのルート・キー・フィールド値を指定します。STARTROOT コマンドはオプションですが、使用するのはスキャン・ユーティリティのときです。

値は、16 進値、文字値、またはパック値で指定します。値は、英字 (X、C、または P) で始め、引用符で囲んだストリングを後に続けます。例えば、X'2A1B'、C'AIN'T'、または P'-00199'。値は、長さと内容の両方が評価されます。例えば、X'00' (1 バイト) は、X'0000' (2 バイト) と同じではありません。

#### STARTSEQ (オプション)

開始 RBA を探すときに使用される順次従属セグメントを指定します。STARTSEQ コマンドはオプションですが、使用するのはスキャン・ユーティリティーのときです。

#### FIELD=(name2)

SSA に指定されているフィールド名です。

名前は 1 文字から 8 文字までにしなければなりません。文字は、英字または数値を使用します。英字には、@、#、および \$ が含まれます。

#### OP=(operator)

SSA に指定されている比較演算子です。

#### VALUE=(value)

SSA に指定されているフィールド値です。

FIELD、OP、および VALUE のフィールドは STARTROOT データと一緒に使用されて、開始 RBA を見付けるように POS 呼び出しをセットアップします。

#### STOPHEXT (オプション)

停止時刻を 16 進形式で指定します。16 進数タイム・スタンプを使用することで、より詳細に指定できます。最新の SDEP SEGMENT の 16 進数タイム・スタンプは、POS 呼び出しを使用して入手できます。

希望するストア・クロック値に一致する 16 桁の 16 進数字を指定する必要があります。STOPHEXT の構文は次のとおりです。

```
STOPHEXT X'C5C55A5DB883FE33' /* 16 hex digits required. */  
Note: C5C55A5DB883FE33 = 2010.092 18:44:05.131.327.875
```

STARTHEXT および STOPHEXT は同じタイム・スタンプ・フォーマットを使用し、その詳細度は STARTIME および STOPTIME に比べはるかに高いものです。

#### STOPRBA (オプション)

停止 RBA を指定します。規則は、STARTRBA コマンドを使用して開始 RBA を指定する場合の、上記規則と同じです。このオプションのコマンドは、スキャン・ユーティリティーまたは削除ユーティリティーで使用します。

16 進数値は、X'16 進数 形式の値です。STOPRBA コマンドに指定されたアドレスは、SDEP 境界になければなりません。そうしないと、ユーティリティーが異常終了します。

#### EXCLUDE

DEDB スキャン・ユーティリティーが、STOPRBA の SDEP セグメントまで (ただし SDEP セグメント自身は除く) を抽出することを指定します。このとき、DEDB 削除ユーティリティーは、フィールド DMACLBTS を STOPRBA+1 ではなく STOPRBA に設定します。



### STOPROOT (オプション)

ルート・キー・フィールド値を指定します。このオプションのコマンドは、スキャン・ユーティリティまたは削除ユーティリティで使用します。

値は、16 進値、文字値、またはパック値で指定します。値は、英字 (X、C、または P) で始め、引用符で囲んだストリングを後に続けます。例えば、X'2A1B'、C'AIN''T'、または P'-00199'。値は、長さで評価され、内容は両方が評価されません。

### STOPSEQ (オプション)

停止 RBA を探すときに使用される順次従属セグメントを指定します。このオプションのコマンドは、スキャン・ユーティリティまたは削除ユーティリティで使用します。

#### FIELD=(name2)

セグメント検索索引数 (SSA) に指定されているフィールド名です。

名前は 1 文字から 8 文字までにしなければなりません。文字は、英字または数値を使用します。英字には、@、#、および \$ が含まれます。

#### OP=(operator)

SSA に指定されている比較演算子です。

#### VALUE=(value)

SSA に指定されているフィールド値です。

FIELD、OP、および VALUE の各フィールドは STOPROOT データと一緒に使用されて、STOP RBA を見付けるために使用されます。これは、START RBA を見付けるために STARTSEQ データが使用されるのと同じです。

### STOPTIME (オプション)

明示的な停止時刻を次の形式で指定します。YYYY.DDD HH:MM:SS ±hh:mm。ここで、YYYY= 年、DDD= 年間通算日、HH= 時、MM= 分、SS= 秒、および hh:mm= UTC へのオフセット (UTC に加算すると現地時間が得られるもの) です。この停止時刻はストア・クロック形式に変更されます。

hh:mm= オフセット・フィールドは、必ず指定する必要があります。このストリングの形式は固定しています。例えば、SS= 秒パラメーターと hh:mm= オフセット・パラメーターの間には、次のようにブランク・スペースを 1 個しか置くことができません。

```
TYPE    SCAN
AREA    <area name>
STOPTIME=C'2010.092 18:40:05 -00:00'
```

STARTIME および STOPTIME は同じタイム・スタンプ・フォーマットを使用し、秒までの詳細度しかありません。

### TYPE (必須)

実行のタイプとして、スキャン、削除、または再編成のいずれかのユーティリティを指定します。

重要: このコマンドは必須であり、最初の GO コマンドの前か、あるいは SYSIN ファイルの終わりの前に入れておかなければなりません。

プログラムは同じジョブ・ステップ内でユーティリティーを切り替えることができないため、ジョブ・ステップ内に指定できる TYPE コマンドは 1 つだけです。

#### V5COMP (オプション)

スキャンおよび削除のすべての操作が、IMS 第 5 版の規則に従うことを指示します。それには以下の規則が含まれます。

- START および STOP を、タイム・スタンプにかかわらず、セグメント境界に置きます。
- スキャンでは、START パラメーターで指定されたセグメントから読み取りを開始します。DMACXVAL CI と START セグメントを含む CI の間にある CI は読み取りません。
- 削除では、論理終了を判別するために、STOP CI だけを読み取ります。DMACXVAL CI からは読み取りません。
- ソートは、NOSORT が指定されていない場合に V5COMP が選択されたときに呼び出されます。

V5COMP を使用すると、IMS 第 5 版のものに等しいユーティリティーから結果を作成することができます。パフォーマンスも同じようになります。ただし、スキャンおよび削除されるセグメントのセットは、CI 間にタイム・スタンプが混じっているために、共用 SDEP と同じにはなりません。

---

## DBFUMSCO ユーティリティーの例

この例は、DEDB スキャン・ユーティリティーを実行するための JCL とユーティリティー制御ステートメントを示しています。

```
//SCAN2 EXEC FPUTIL,DBD=DEDBJN03,REST=00
//*
//* DBD=DBDNAME AS TARGET DATABASE FOR THIS UTILITY RUN
//* REST=RESTART NUMBER FOR THIS RUN
//*
//SCANCOPY DD DSNAME=SCAN203,DISP=(NEW,PASS,DELETE),
// UNIT=SYSDA,VOL=SER=IMSDCL,
// DCB=(NCP=5,BLKSIZE=2048),
// SPACE=(TRK,5)
//*
//* SCANIDT DD IS USED TO WRITING INDOUBT SEGMENTS TO
//* THE SCAN204 DATA SET
//*
//SCANIDT DD DSNAME=SCAN204,DISP=(NEW,PASS,DELETE),
// UNIT=SYSDA,VOL=SER=IMSDCL,
// DCB=(NCP=5,BLKSIZE=2048),
// SPACE=(TRK,5)
//SYSIN DD *
*-----*
* USE THE STOPROOT WITH STOPSEQ TO LIMIT THE *
* RANGE OF THE SCAN UP TO THE LAST DAY ACCUMULATION *
* OF SEGMENTS. *
*-----*
*
* COMMAND OPERATOR COMMENT
*
* ERRORACTION SCANRUN SET ERROR OPTION
* TYPE SCAN ONLINE SEQ DEP SCAN UTILITY
* THE TARGET DATABASE IS
```

```

*                               DBDNAME=DEDBJN03
*
*                               THE TARGET DEDB AREA
*   AREA          DB3AREA0
*                               THE NO. OF BUFFERS
*   STOPROOT=C'R301102A'
*                               STOP ON SEQ DEP SEGMENT - JULIAN DATE
*   STOPSEQ      FIELD=SDFLDDAT,OP='<=',VALUE=C'273'
*                               SCAN INDOUBT SEGMENTS
*   INDOUBT
*                               THE EXIT PROGRAM NAME IS
*   EXIT          EXITLDM3

```

---

## DBFUMSCO ユーティリティーのスキヤンの範囲

処理する順次従属セグメントの範囲を指定することができます。

処理する順次従属セグメントの範囲は、3 とおりの方法で指定できます。開始位置は、STARTRBA、STARTROOT、または STARTSEQ コマンドを使用して指定できます。停止位置は、STOPRBA、STOPROOT、または STOPSEQ コマンドを使用して指定できます。

範囲が指定されていない場合は、デフォルトの終了処理が呼び出されます。最高水準点 (HWM) CI では、永続デフォルト終了タイム・スタンプ・セグメントが作成されます。スキヤンは、区域内の最も古い順次従属 CI 境界から開始されます。スキヤンは HWM CI のデフォルト終了セグメントからのタイム・スタンプで終了しますが、このタイム・スタンプはスキヤンの対象に含まれていません。

STARTRBA は、セグメントの 4 バイトの RBA、またはサイクル番号と RBA の 8 バイトの組み合わせを指定するために使用します。サイクル番号は POS 呼び出しにより返され、DEDB が RBA の接頭部として使用する番号です。これから、区域が存続している間に一度だけ使用される値が得られます。

サイクル番号が非ゼロ値で指定されると、スキヤン・ユーティリティーは、それが当該の RBA の現行サイクル番号と一致しているかどうかを確認めます。サイクル番号がその RBA の現行サイクル番号と同じでないか、あるいは順次従属セグメントが現在保管されている範囲内に RBA がない場合、エラー・メッセージが印刷され、データはスキヤンされません。サイクル番号が指定されていないか、あるいはゼロと指定されている場合は、その RBA の現行サイクル番号が使用されます。開始 RBA と停止 RBA がサイクル番号なしで指定された場合、最高位の RBA から最低位の RBA へ循環して戻るため、停止 RBA が実際に開始 RBA よりも低くなる場合があります。ただし、サイクル番号は停止 RBA の方が高くなります。指定された RBA がセグメントの始まりのアドレスでない場合、スキヤンは次のセグメントから開始されます。

STARTROOT または STARTSEQ でスキヤンを開始させる場合は、マーカー・セグメントが使用されるものと見なされます。マーカー・セグメントとは、固有のフィールド値を持つ特殊な順次従属セグメントです。マーカー・セグメントを維持するには、マーカー・セグメントを順次従属として挿入するアプリケーション・プログラムを実行します。

STARTRBA CI 境界が指定される場合

- 最初のセグメント・タイム・スタンプは、選択された開始点です。

- V5COMP では、この CI でのすべてのセグメントが含まれます。

#### STARTRBA セグメント境界が指定される場合

- 指定されたセグメントは、タイム・スタンプ属性とロケーション属性のコミット済みマーカー・セグメントのような役割を果たします。
- V5COMP では、指定されたセグメントは開始点であり、タイム・スタンプは無視されます。

#### デフォルトの開始が使用される場合

- 論理開始タイム・スタンプ (LBTS) 以上のタイム・スタンプを持つすべてのセグメントが選択されます。
- V5COMP では、最初の CI にあるすべての非アポート・セグメントが選択されます。

挿入されたマーカー・セグメントを持つルート・セグメントを、このユーティリティーに対して指定する必要があります。これを行うには、STARTROOT コマンドでルート・セグメントのキーを指定します。順次従属がルートに対して最後に挿入された順次従属でない場合は、必要とするセグメントを検索するためのセグメント検索索引数 (SSA) を作成するのにユーティリティーが使用する情報を指定してください。SSA を作成するには、フィールド名、比較演算子、およびフィールド値を STARTSEQ コマンドに指定します。STARTROOT コマンドごとに 1 つのみの STARTSEQ コマンドが有効です。複数の STARTSEQ コマンドが指定される場合、最後のコマンドが使用されます。同じことが STOPROOT と STOPSEQ にも当てはまります。比較の規則は、通常の DEDB SSA の場合と同じです。指定するフィールド値は、通常の DEDB SSA の場合と同じにしなければなりません。指定するフィールド値は、DEDB 内のフィールドと同じ長さになければなりません。ルートが指示された区域に存在しないか、あるいは指示されたセグメントが見付からない場合は、エラー・メッセージが印刷され、データはスキャンされません。

以下の例は、マーカー・セグメントの使用を示しています。

- 24 時間体制でユーザーのインストール・システムを稼働していない場合は、毎日のトランザクションを 1 日の終わりに処理することがあります。特殊な順次従属セグメントをどのルート・セグメントに追加するかを選択してください。セグメントには、その日の日付が入るフィールド値があります。
- アプリケーション・プログラムを実行させて、セグメントを挿入します。これらのセグメントは、毎日の処理の終わりを示すマーカー・セグメントであり、スキャン範囲を制御するために使用できます。ユーティリティーは、マーカー・セグメントの固有フィールド値 (日付) をスキャンします。

停止 RBA は 3 とおりの方法で指定できますが、それらの方法は、開始 RBA を指定する 3 とおりの方法とまったく同じです。ただし、コマンドは、STARTRBA、STARTROOT、および STARTSEQ ではなく、STOPRBA、STOPROOT、および STOPSEQ が使用されます。最後にスキャンされるセグメントは、停止 RBA から始まるか、あるいはその前で始まる最後のセグメントです。

#### STOPRBA CI 境界が指定される場合

- CI 内の最初のセグメント・タイム・スタンプが停止タイム・スタンプとして使用されます。

- V5COMP では、CI は除外され、タイム・スタンプは無視されます。

#### STOPRBA セグメント境界が指定される場合

- 指定されたセグメントは、タイム・スタンプ属性とロケーション属性のコミット済みマーカー・セグメントのような役割を果たします。
- V5COMP では、指定されたセグメントは停止点であり、タイム・スタンプは無視されます。

#### デフォルトの終了が使用される場合

- 最高水準点 (HWM) CI のデフォルト終了セグメント・タイム・スタンプ未満のタイム・スタンプを持つすべてのセグメントが含まれます。
- V5COMP では、HWM CI までのすべての非アポート・セグメントが選択されますが、この HWM CI は選択の対象から除外されます。

停止 RBA が開始 RBA と同じである場合、セグメントは 1 つしかスキャンされません。停止 RBA が開始 RBA よりも前に現れるセグメントを参照した場合は、エラー・メッセージが印刷され、データはスキャンされません。開始 RBA と停止 RBA を、同じ方法で指定する必要はありません。



---

## 第 9 章 HALDB 区画データ・セット初期設定ユーティリティー (DFSUPNT0)

HALDB 区画を初期設定するには、HALDB 区画データ・セット初期設定ユーティリティー (DFSUPNT0) を使用してください。

DFSUPNT0 ユーティリティーは通常、「初期設定が必要」(PINIT=Y) と RECON データ・セットにフラグが立てられた区画のみ初期設定します。ただし、DFSUPNT0 ユーティリティーで、RECON データ・セットに PINIT=N と記録されている区画でも無条件に区画を初期設定することができます。初期設定した区画内のデータは無条件に失われます。

HALDB マスター・データベース名、区画名、またはその両方を指定できます。

- HALDB マスター・データベース名を指定すると、RECON データ・セット内で「区画の初期設定が必要」と指定されたデータベース内の区画のすべてが初期設定されます。
- HALDB 区画名を SYSIN ステートメントとして指定すると、その区画が無条件に初期設定され、RECON データ・セット内の区画レコードに PINIT=N と設定されます。
- DFSOVRDS DD ステートメントを指定すると、HALDB データベースのすべての区画が無条件に初期設定されます。

DFSUPNT0 ユーティリティーは、IMS による各区画の完全な検査と検証を提供するので、このユーティリティーを既存の HALDB データベースと一緒に使用することを推奨します。データベースを HALDB に変換するときに区画を初期設定するために、データベース事前再編成ユーティリティー (DFSURPRO) を使用することもできます。

IMS 領域コントローラーに実行依頼される、または SYSIN データ・セット内のステートメントとして実行依頼される EXEC パラメーター・リストに HALDB マスター・データベース名を指定できます。SYSIN ステートメントを使用すると、HALDB マスター・データベース名および区画名を複数指定できます。

区画名は、SYSIN データ・セット内のステートメントとしてのみ指定できます。複数の区画名を指定できます。

DFSUPNT0 ユーティリティーは、複数のジョブ・タスクとして実行できます。各区画を別々の DFSUPNT0 ジョブとして指定してこれらのジョブを同時に実行すると、区画初期設定処理全体を大幅に改善できます。

複数のジョブを区画名を指定しないで実行することもできますが、これは推奨できません。各タスクは、RECON 内にあるすべての必要な区画の初期設定を試み、システム・エンキューを使用して初期設定機能に沿って順次に行われます。正常にエンキューされると、初期設定を必要とする区画のリストについて DBRC が照会されます。初期設定を必要とする区画で、ジョブ・バッチ初期設定時とは異なるものは

無視されます。バッチ初期設定時に初期設定が必要だったが、もう必要としなくなった区画は無視されます。エンキュー後にまだ初期設定を必要としている区画のみが初期設定されます。

サブセクション:

- 『制約事項』
- 『前提条件』
- 『要件』
- 『推奨事項』
- 101 ページの『入出力』
- 101 ページの『戻りコード』
- 101 ページの『JCL 指定』

## 制約事項

DFSUPNT0 ユーティリティは、全機能 HALDB データベースのみで機能しません。

データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

## 前提条件

DFSUPNT0 ユーティリティを実行する前に、DFSUPNT0 ユーティリティによって動的に割り振られる z/OS に対して、すべてのデータ・セットを定義する必要があります。

## 要件

DFSUPNT0 ユーティリティを実行するには、さまざまな動作要件を満たす必要があります。

HALDB 区画データ・セット初期設定ユーティリティを使用するときは、次のような要件があります。

- データ・セットは、動的に割り振られますが、事前に定義しておく必要があります。
- DBRC が必須です。
- 動的に割り振られない場合は、RECON DD ステートメントが必須です。
- JCL DD ステートメントを区画データ・セット用にコーディングしてはなりません。区画データ・セットは、RECON からの情報に基づいて IMSによって動的に割り振る必要があります。

## 推奨事項

既存の HALDB データベースとともに使用するためには、DFSUPNT0 ユーティリティを推奨します。このユーティリティは、IMS による各区画の完全な検査と検証を提供します。データベースを HALDB に変換するときに区画を初期設定するために、データベース事前再編成ユーティリティ (DFSURPR0) を使用することもできます。



HALDB データベースの区画初期設定処理の全体的なパフォーマンスを改善するには、別々の DFSUPNT0 ジョブを使用して HALDB データベース内の各区画を初期設定した後、これらのジョブを同時に実行します。

## 入出力

DFSUPNT0 ユーティリティへの 1 次入力、初期設定を必要とする区画です。この区画データ・セットを割り振る必要はありません。

DFSUPNT0 ユーティリティの 1 次出力は初期設定済み区画です。

## 戻りコード

プログラム終了時に次の戻りコードが返されます。

コード 意味

- 0 正常終了。
- 8 処理エラーか、あるいは DFSOVRDS DD ステートメントの形式が間違っている。
- 12 入力または IMS制御ブロックが無効
- 16 環境またはユーザー・エラー
- 32 IMS の接続から異常終了が返された
- 99 内部論理エラー

## JCL 指定

DFSUPNT0 ユーティリティは、標準の z/OS ジョブとして実行されます。DFSUPNT0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含めることが必須であり可能です。

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

2 とおりの EXEC ステートメントが可能です。1 つには SYSIN があり、1 つには SYSIN がありません。EXEC ステートメントで渡すことができるのは、マスター・データベース名のみです。区画名は、SYSIN ステートメントとしてのみ使用できます。

この節の例で、動的割り振りを行わずに DBRC を使用する場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

```
//RECON1 DD DSN=RECON1,DISP=SHR
//RECON2 DD DSN=RECON2,DISP=SHR
//RECON3 DD DSN=RECON3,DISP=SHR
```

以下の例で、マスター・データベース名 HDODB1 は EXEC ステートメントに指定されています。

```
PARM=(ULU,DFSUPNT0,HDODB1,,,,,,,,,SYS3,,Y,N)
```

SYSIN がない場合、EXEC ステートメントは以下の形式でなければなりません。

```
//PINIT01 EXEC PGM=DFSRR00,REGION=2048K,
//          PARM=(ULU,DFSUPNT0,HDODB1,,,,,,,,,SYS3,,Y,N)
//STEPLIB DD DSN=IMS.SDFSRESL
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//SYSUDUMP DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
/*
```

SYSIN がある場合、EXEC ステートメントは以下の例の形式でなければなりません。

```
//PINIT02 EXEC PGM=DFSUPNT0,REGION=2048K
//STEPLIB DD DSN=IMS.SDFSRESL
//STEPCLIB DD DSN=VCATSHR,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//SYSUDUMP DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
DBVHDJ05
DBOHIDK5
/*
```

この JCL により、HALDB マスター・データベース DBVHDJ05 および DBOHIDK5 用に PINIT として RECON に記録されているすべてのデータベース区画が初期設定されました。HALDB マスター・データベースに区画名を指定する場合、区画は無条件に初期設定され、PINIT がオフの状態では RECON に記録されません。

## DD ステートメント

### STEPLIB DD

IMS.SDFSRESL を指します。ここでは、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

### IMS DD

分析するデータベースを記述する DBD と PSB が入っているライブラリーを定

義します。これらのデータ・セットは、直接アクセス装置に置いておかなければなりません。このステートメントは必須であり、必ず DBD ライブラリーを定義しなければなりません。PSB ライブラリーが必要になるのは、PARM=DLI と指定されたときだけです。

#### **RECON1 DD**

最初の DBRC RECON データ・セットを定義します。

#### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。区域が DBRC RECON データ・セットに登録されている場合は、この RECON データ・セットが必要です。

#### **RECON3 DD**

RECON1 または RECON2 にエラーが見つかったとき使用される、オプションの DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

注: 動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

#### **SYSIN DD**

このプログラムのための入力制御データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。LRECL と BLKSIZE は共に 80 でなければなりません。

#### **SYSPRINT DD**

メッセージ出力データ・セットを定義します。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=120 です。BLKSIZE を指定するときは、120 の倍数になっていなければなりません。

この DD ステートメントは必須です。

SYSPRINT データ・セットは、ユーザーにエラー・メッセージを戻すためにも使用されます。

以下の図は、SYSPRINT に対する DFS391I 出力の例です。

図 17. SYSPRINT に対する DFS391I 出力

```
DFS391I PARTITION INITIALIZATION UTILITY
DFS391I SYSIN CONTROL CARDS
DFS391I POHIDJB
DFS391I PHVNTKA
DFS391I POHIDJC
DFS391I POHIDJA
DFS391I END OF SYSIN CONTROL CARDS
```

#### **SYSUDUMP DD**

ダンプ・データ・セットを定義します。

### DFSVSAMP DD

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。

### DFS0VRDS DD

区画を無条件で初期設定するためのオプションです。RECON 内の PARTITION INIT NEEDED フラグが Y に設定されていなくても、このオーバーライド・オプションにより、HALDB データベースとそのすべての区画を初期設定できます。

DFS0VRDS DD ステートメントに対する有効な入力は、文字ストリングは INITALL または DD DUMMY のみです。INITALL または DD DUMMY 以外を入力すると、ユーティリティーが理由コード 19「THE DFS0VRDS DD CARD HAS AN INVALID INPUT」の WTO メッセージ DFS1987I を発行します。

DFS0VRDS DD ステートメントの形式を以下の図に示します。

図 18. 区画の無条件初期設定のための DD ステートメント

```
//DFS0VRDS DD *  
INITALL
```

**重要:** 従来の HALDB 区画に関する区画データは、すべて失われます。DFS0VRDS DD を指定しなかった場合と同様に、ストリング INITALL 以外の入力はユーティリティーから無視されます。

---

## DFSUPNT0 ユーティリティーの制御ステートメント

入力ステートメントは、HALDB 区画データ・セット初期設定ユーティリティーの処理オプションを記述するために使用します。

入力ステートメントは、以下に準拠している必要があります。

- SYSIN を使用して実行するときは、1 桁目に HALDB マスター・データベース名を指定します。
- SYSIN を使用しないで実行するときは、領域コントローラーに HALDB マスター・データベース名を指定します。

---

## DFSUPNT0 ユーティリティーの例

この節のすべての例で、DFSUPNT0 を実行するためのサンプル JCL を提供しています。

サブセクション:

- 『区画の初期設定』
- 105 ページの『マスター・データベース名のリストのための区画の初期設定』

### 区画の初期設定

このサンプル JCL は、PINIT または必要な区画初期設定として RECON に記録されているすべての区画を初期設定するように HALDB HDODB1 用に DFSUPNT0

を実行します。バッチ・プログラム・コントローラーに対するパラメーターは、ユーティリティー領域タイプ ULU 用のものです。区画データ・セットが事前に存在している必要がありますが、そこに含まれているデータはすべてリセットされて空になります。OSAM の空のデータ・セットは、前のジョブ・ステップで定義している必要があります。区画データ・セットは、RECON からの情報を使用して動的に割り振られます。RECON データ・セットは、IMSVS.RESLIB で見付かった MDA メンバーを使用して動的に割り振られます。

```
//STEP1 EXEC PGM=IEFBR14 /* Define OSAM data sets here */
//HDOSAM DD ... DISP=(,CATLG)
//PINIT EXEC PGM=DFSRR00,REGION=2048K,
// PARM={ULU,DFSUPNT0,HDODB1,,,,,,,,,SYS3,,Y,N}
//STEPLIB DD DSN=IMSVS.RESLIB,DISP=SHR
//DFSRESLB DD DSN=IMSVS.RESLIB,DISP=SHR
//IMS DD DSN=IMSVS.PSBLIB,DISP=SHR
// DD DSN=IMSVS.DBDLIB,DISP=SHR
//DFSVSAMP DD DSN=IMSTESTG.DFSVSAMP.DATA{VSM885FP},DISP=SH
//SYSUDUMP DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//HDOSAM DD ... DISP=OLD
/*
```

## マスター・データベース名のリストのための区画の初期設定

このサンプル JCL は、SYSIN に指定されている HALDB マスター・データベース名のリストのために PINIT として RECON に記録されているすべての区画を初期設定するように DFSUPNT0 を実行します。IMS は、静的パラメーター・リストを使用して、ユーティリティー領域タイプ ULU として接続されます。区画データ・セットは、RECON からの情報を使用して動的に割り振られます。RECON データ・セットは、IMSVS.RESLIB で見付かった MDA メンバーを使用して動的に割り振られます。

```
//STEP1 EXEC PGM=IEFBR14 /* Define OSAM data sets here */
//HDOSAM DD ... DISP=(,CATLG)
//STEP2 EXEC PGM=IDCAMS /* Define VSAM data sets here */
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
DEFINE ...
//PINIT EXEC PGM=DFSUPNT0,REGION=2048K
//STEPLIB DD DSN=IMSVS.RESLIB,DISP=SHR
//DFSRESLB DD DSN=IMSVS.RESLIB,DISP=SHR
//IMS DD DSN=IMSVS.PSBLIB,DISP=SHR
// DD DSN=IMSVS.DBDLIB,DISP=SHR
//DFSVSAMP DD DSN=IMSTESTG.DFSVSAMP.DATA{VSM885FP},DISP=SH
//SYSUDUMP DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
DBVHDJ05
DBOHIDK5
/*
```



---

## 第 10 章 HALDB 区画定義ユーティリティ (%DFSHALDB)

HALDB 区画定義ユーティリティは、DBRC への HALDB マスター・データベースの登録と、HALDB 区画の追加、変更、および削除を行うために使用します。このユーティリティは、同様の HALDB 管理機能がある DBRC コマンドの代わりに使用することができます。

HALDB 区画定義ユーティリティのパネル間をナビゲートすることによって、HALDB マスターおよび HALDB 区画に関する以下の作業を行うことができます。

- HALDB マスター・データベースを DBRC に登録する。
- 既存の HALDB データベースへの HALDB 区画の追加。
- HALDB 区画データを検索、表示、ソート、コピー、変更、削除、および印刷する。
- データ・セット・グループを作成し、変更する。
- HALDB 情報を編集する。
- HALDB 定義をエクスポートし、インポートする。
- IMS DD 名連結を表示する。
- IMS RECON/DBDLIB ライブラリーを選択する。

HALDB 区画定義ユーティリティにより、オンライン IMS システムの RECON データ・セットの競合が大幅に増加することはありません。RECON データ・セットは、DBRC 要求を処理するのにかかる時間の間のみ予約されます。このデータ・セットは、ユーティリティの実行中は保持されません。

HALDB 区画定義ユーティリティは、以下の開始コマンドを使用して ISPF 内から開始できます。

```
TSO %DFSHALDB
```

HALDB 区画定義ユーティリティは、IMS Application Menu を使用して開始することもできます。

サブセクション:

- 108 ページの『制約事項』
- 108 ページの『前提条件』
- 108 ページの『要件』
- 108 ページの『推奨事項』
- 108 ページの『入出力』
- 109 ページの『対話モードのための JCL 指定』
- 111 ページの『バッチ・インポートおよびバッチ・エクスポートのための JCL 指定』
- 113 ページの『戻りコード』

## 制約事項

HALDB 区画定義ユーティリティを使用するときは、次のような制約があります。

- HALDB 区画定義ユーティリティは、DBDLIB メンバーの HALDB データベースとして識別されたデータベースにのみ作用します。HALDB 定義が設定された後に DBD が変更された場合は、その定義を取り扱うことができません。例えば、データ・セット・グループの数を変更するには、HALDB 区画定義を削除して、再定義しなければなりません。
- 特定のデータベースの HALDB 定義を、複数のユーザーが同時に更新することはできません。RECON1 データ・セットの名前とデータベースの名前に基づいてシリアライゼーションが行われます。
- このユーティリティは、順次バッファリング (SB) をサポートしません。

## 前提条件

%DFSHALDB ユーティリティを使用して新しい HALDB マスター・データベースを DBRC に登録するには、まずデータベース記述生成 (DBDGEN) ユーティリティを使用してそのマスター・データベースの DBD を定義する必要があります。

## 要件

現在、%DFSHALDB ユーティリティに対して文書化された要件はありません。

## 推奨事項

現在、%DFSHALDB ユーティリティに対して文書化された推奨事項はありません。

## 入出力

HALDB 区画定義ユーティリティは、次のように入力を受け取ります。

- DBD 生成情報が DBDLIB から読み取られる。
- 保管された定義が RECON から取得される。
- ユーザー入力対話式パネルから請求される。
- エクスポートの結果が、次のインポート操作への入力として使用される。
- 構成情報が ISPF プロファイル・データ・セットから取得される。
- HALDB 区画が定義されているときは、キー・ストリング付きのデータ・セットが入力として使用できる。
- TSO コマンド・パラメーターとしてバッチ・インポート・パラメーターとバッチ・エクスポート・パラメーターが指定される。

HALDB 区画定義ユーティリティは、次のように入力を作成します。

- HALDB 定義が RECON データ・セットに保管される。
- 対話式メッセージが ISPF パネルに表示される。
- エクスポートされた HALDB 定義が、ユーザー指定のデータ・セットに保管される。
- いくつかのメッセージが SYSOUT ファイルに書き込まれる。



- HALDB 定義情報を ISPF リスト・ファイルに印刷することができる。

#### DSPXRUN EXPORT の出力メッセージの例

111 ページの図 22 に示されているような JCL を使用してデータベース PARTDBA をエクスポートすると、以下の出力が得られます。

図 19. エクスポートの %DFSHALDB 出力メッセージ

```
DSPM142I Start export to MEM=PARTDBA in DSN='IBMUSER.HALDB.EXPORT'
        from DBN=PARTDBA
DSPM143I The export file contains partition PAAA
DSPM143I The export file contains partition PAAB
DSPM143I The export file contains partition PAAC
DSPM143I The export file contains partition PAAD
DSPM219I Table PARTDBA was created successfully to dataset
        'IBMUSER.HALDB.EXPORT'
```

#### DSPXRUN IMPORT の出力メッセージの例

111 ページの図 22 に示されているような JCL を使用してデータベース PARTDBA をインポートすると、以下の出力が得られます。

図 20. インポートの %DFSHALDB 出力メッセージ

```
DSPM083I Start Import to DSN=PARTDBA from MEM=PARTDBA in
        DSN='IBMUSER.HALDB.EXPORT' Options=2
DSPM085I Imports start at 07/07/09 11:50
DSPM084I Import successful for partition PAAA
DSPM084I Import successful for partition PAAB
DSPM084I Import successful for partition PAAC
DSPM084I Import successful for partition PAAD
DSPM082I 4 of a total 4 partitions from table PARTDBA were
        imported to database successfully.
```

### 対話モードのための JCL 指定

以下の図は、%DFSHALDB ユーティリティを ISPF パネルでの対話モードで実行するためのサンプル TSO ログオン・プロシーチャーを示しています。

図 21. %DFSHALDB ユーティリティを対話モードで実行するための TSO ログオン・プロシーチャー

```
//HALDB01 EXEC PGM=IKJEFT01,DYNAMNBR=50,REGION=6M
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR /* IMS SDFSRESL */
//SYSPROC DD DSN=IMS.SDFSEXEC,DISP=SHR /* IMS rexx execs */
//        DD DSN=ISP.SISPCLIB,DISP=SHR
//IMS     DD DSN=your.local.DBDLIB,DISP=SHR
//ISPLIB DD DSN=IMS.SDFSPLIB,DISP=SHR /* IMS ISPF panels*/
//        DD DSN=ISP.SISPPLIB,DISP=SHR
//ISPSLIB DD DSN=IMS.SDFSLLIB,DISP=SHR /* IMS ISPF skeletons*/
//        DD DSN=ISP.SISPSLIB,DISP=SHR
//ISPMLIB DD DSN=IMS.SDFSMLIB,DISP=SHR /* IMS ISPF messages */
//        DD DSN=ISP.SISPMLIB,DISP=SHR
//ISPTLIB DD DSN=IMS.SDFSSTLIB,DISP=SHR /* IMS ISPF tables*/
//        DD DSN=ISP.SISPTLIB,DISP=SHR
//ISPLG   DD SYSOUT=*,DCB=(RECFM=VA,LRECL=125,BLKSIZE=129)
```

```

//SYSPRINT DD SYSOUT=*,DCB=(RECFM=VA,LRECL=125,BLKSIZE=129)
//SYSOUT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*,DCB=(RECFM=F,LRECL=255,BLKSIZE=255)
//SYSPRINT DD TERM=TS,SYSOUT=A
//ISPCTL0 DD UNIT=SYSDA,SPACE=(CYL,(0,1)),
// DCB=(LRECL=80,RECFM=FB,BLKSIZE=800)
//ISPCTL1 DD UNIT=SYSDA,SPACE=(CYL,(0,1)),
// DCB=(LRECL=80,RECFM=FB,BLKSIZE=800)
//ISPCTL2 DD UNIT=SYSDA,SPACE=(CYL,(0,1)),
// DCB=(LRECL=80,RECFM=FB,BLKSIZE=800)
//ISPLST1 DD UNIT=SYSDA,SPACE=(CYL,(0,1)),
// DCB=(LRECL=121,RECFM=FBA,BLKSIZE=1210)
//ISPLST2 DD UNIT=SYSDA,SPACE=(CYL,(0,1)),
// DCB=(LRECL=121,RECFM=FBA,BLKSIZE=1210)
//ISPWRK1 DD UNIT=SYSDA,SPACE=(CYL,(0,1)),
// DCB=(LRECL=256,RECFM=FB,BLKSIZE=2560)
//ISPWRK2 DD UNIT=SYSDA,SPACE=(CYL,(0,1)),
// DCB=(LRECL=256,RECFM=FB,BLKSIZE=2560)
//SYSTEM DD TERM=TS,SYSOUT=A
//SYSIN DD TERM=TS

```

TSO ログオン・プロシージャーには、次のものが含まれなければなりません。

- 実動 TSO/ISPF ログオン・プロシージャーからの DD ステートメント
- STEPLIB DD ステートメントに IMS.SDFSRESL データ・セット
- 該当の ISPF DD ステートメントに IMS ダイアログ・コンポーネント

**DD** ステートメント

#### **STEPLIB DD**

IMS.SDFSRESL を指します。ここには、IMS ロード・モジュールが入っています。

#### **IMS DD**

HALDB データベースの DBD が入っているライブラリーを定義します。

#### **SYSPROC DD**

CLIST および REXX exec が入っているデータ・セットを定義します。  
HALDB 区画定義ユーティリティーにとって、その exec がこのデータ・セット群に入っていることが必要です。

#### **ISPPLIB DD**

ISPF パネルが入っているデータ・セットを定義します。HALDB 区画定義ユーティリティーにとって、そのパネルがこのデータ・セット群に入っていることが必要です。

#### **ISPMLIB DD**

ISPF メッセージ・メンバーが入っているデータ・セットを定義します。  
HALDB 区画定義ユーティリティーにとって、そのメッセージ・メンバーがこのデータ・セット群に入っていることが必要です。

#### **ISPTLIB DD**

ISPF テーブルが入っているデータ・セットを定義します。HALDB 区画定義ユーティリティーにとって、その表がこのデータ・セット群に入っていることが必要です。この連結には、システム ISPF テーブル・ライブラリーも含めます。

#### **RECON1 DD**

この DD ステートメントはオプションですが、最初の DBRC RECON デー

タ・セットを定義します。これを指定しない場合は、RECON1 データ・セットを識別する IMS.SDFSRESL データ・セットに RECON1 メンバーがなければなりません。

#### RECON2 DD

この DD ステートメントはオプションですが、2 番目の DBRC RECON データ・セットを定義します。これを指定しない場合は、RECON2 データ・セットを識別する IMS.SDFSRESL データ・セットに RECON2 メンバーがなければなりません。

#### RECON3 DD

この DD ステートメントはオプションですが、3 番目の DBRC RECON データ・セットを定義します。これを指定しない場合は、RECON3 データ・セットを識別する IMS.SDFSRESL データ・セットに RECON3 メンバーがなければなりません。

### バッチ・インポートおよびバッチ・エクスポートのための JCL 指定

HALDB 区画定義ユーティリティのインポート機能およびエクスポート機能は、バッチ・ジョブで実行することができます。そのため、JCL に通常の ISPF DD ステートメントを入れて、ISPF を開始させる必要があります。

JCL には、以下のエレメントが含まれなければなりません。

- ISPF で使用される DD ステートメントを定義する JOB ステートメント
- STEPLIB DD ステートメントに IMS.SDFSRESL データ・セット
- 該当の ISPF DD ステートメントに IMS ダイアログ・コンポーネント

HALDB のエクスポートの結果として出力されるのは、PDS のメンバーです。HALDB に関する情報は、ISPF テーブルの形で保管されます。ISPF テーブルは、インポート処理用の入力として使用されます。エクスポートおよびインポートは、ISPF パネルまたはバッチ・ジョブから実行できます。

このバッチ・ジョブは、ISPF 環境をセットアップする標準の ISPF コマンドである ISPSTART を実行し、それから DSPXRUN コマンドを開始します。DSPXRUN コマンドは HALDB データベース、使用するインポート・ファイル、および処理オプションを識別します。

HALDB のバッチ・インポートは、以下の図に示されるジョブと同様のバッチ ISPF ジョブを実行依頼することにより、処理することができます。

バッチ・ジョブを使用してデータベースをインポートするために、以下の図に示されるジョブと同様のバッチ ISPF ジョブを実行依頼します。すべての ISPF DD 名が必要になります。ISPF はバッチで呼び出されるため、すべての ISPF DDNAME が必要です。

図 22. バッチ・インポート用のサンプル JCL

```
//*  
//DSPXRUN EXEC PGM=IKJEFT01,DYNAMNBR=50,REGION=6M  
//STEPLIB DD DISP=SHR,DSN=IMS.SDFSRESL /* IMS.SDFSRESL */  
// DD DISP=SHR,DSN=ISP.SISPLOAD  
// DD DISP=SHR,DSN=ISP.SISPPLA
```

```

//SYSPROC DD DISP=SHR,DSN=IMS.SDFSEXEC /* IMS rexx execs */
// DD DISP=SHR,DSN=ISP.SISPCLIB
//RECON1 DD DISP=SHR,DSN=IMS.RECON1
//RECON2 DD DISP=SHR,DSN=IMS.RECON2
//RECON3 DD DISP=SHR,DSN=IMS.RECON3
//IMS DD DISP=SHR,DSN=IMS.DBDLIB
//ISPPROF DD DSN=&&PROFILE, /* dummy ISPF profile */
// UNIT=SYSDA,DISP=(NEW,DELETE),
// SPACE=(3200,(30,30,1)),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)
//ISPLIB DD DISP=SHR,DSN=ISP.SISPPENU /* IMS ISPF panels */
//ISPSLIB DD DISP=SHR,DSN=ISP.SISPSLIB /* IMS ISPF skeletons */
// DD DISP=SHR,DSN=ISP.SISPSLIB
//ISPMLIB DD DISP=SHR,DSN=IMS.SDFSMLIB /* IMS ISPF messages */
// DD DISP=SHR,DSN=ISP.SISPMENU
//ISPTLIB DD DISP=SHR,DSN=ISP.SISPTENU /* IMS ISPF tables */
//ISPLOG DD SYSOUT=*,DCB=(RECFM=VA,LRECL=125,BLKSIZE=129)
//SYSPRINT DD SYSOUT=*,DCB=(RECFM=VA,LRECL=125,BLKSIZE=129)
//SYSOUT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*,DCB=(RECFM=F,LRECL=255,BLKSIZE=255)
//SYSTSIN DD *
ISPSTART CMD( +
DSPXRUN +
IMPORT +
DSN('USRTO02.ISPF.PROFILE') +
DBN(PARTDBA) +
MEM(PARTDBA))

```

区画情報をエクスポートまたはインポートするためのバッチの使用

## DD ステートメント

### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS ロード・モジュールが入っています。

### IMS DD

HALDB の DBD が入っているライブラリーを定義します。

### SYSPROC DD

CLIST および REXX exec が入っているデータ・セットを定義します。  
HALDB 区画定義ユーティリティーにとって、その exec がこのデータ・セット群に入っていることが必要です。

### ISPMLIB DD

ISPF メッセージ・メンバーが入っているデータ・セットを定義します。  
HALDB 区画定義ユーティリティーにとって、そのメッセージ・メンバーがこのデータ・セット群に入っていることが必要です。

### ISPTLIB DD

ISPF テーブルが入っているデータ・セットを定義します。HALDB 区画定義ユーティリティーにとって、その表がこのデータ・セット群に入っていることが必要です。この連結には、システム ISPF テーブル・ライブラリーも含めます。

### RECON1 DD

この DD ステートメントはオプションですが、最初の DBRC RECON データ・セットを定義します。これを指定しない場合は、RECON1 データ・セットを識別する IMS.SDFSRESL データ・セットに RECON1 メンバーがなければなりません。

### RECON2 DD

この DD ステートメントはオプションですが、2 番目の DBRC RECON データ・セットを定義します。これを指定しない場合は、RECON2 データ・セットを識別する IMS.SDFSRESL データ・セットに RECON2 メンバーがなければなりません。

### RECON3 DD

この DD ステートメントはオプションですが、3 番目の DBRC RECON データ・セットを定義します。これを指定しない場合は、RECON3 データ・セットを識別する IMS.SDFSRESL データ・セットに RECON3 メンバーがなければなりません。

### SYSTSIN DD

SYSTSIN DD ステートメントには、ISPF および HALDB 区画定義ユーティリティーの両方を開始させる TSO コマンドを入れます。次の行に継続する必要がある場合は、継続する行の最後の文字にプラス符号またはマイナス符号を使用します。

## 戻りコード


以下の戻りコードは、バッチ・インポートでのプログラム終了時に返されます。対話式ダイアログでは、終了戻りコードが使用されません。


コード 意味

- 0 正常終了。
- 4 いくつかの HALDB 区画がインポートできなかった
- 8 HALDB 区画がまったくインポートされなかった

HALDB 区画定義ユーティリティーによって表示されるいくつかのメッセージは、フィールドの値が正しくないときの標準的な ISPF メッセージ・プロンプトです。ISPF メッセージは、ISR または ISP の接頭部で始まります。

関連概念:

 HALDB オンライン再編成での IMS ユーティリティーの使用 (データベース管理)

 HALDB 区画の名前と番号 (データベース管理)

---

## %DFSHALDB ユーティリティーの制御ステートメント

HALDB 区画定義ユーティリティーは、バッチ DSPXRUN ユーティリティー制御ステートメントを提供します。DSPXRUN は、HALDB 定義のバッチ・インポートおよびバッチ・エクスポートをサポートします。DSPXRUN は、一般的な ISPF コードの実行をサポートする ISPF コマンド ISPSTART のパラメーターとして実行依頼されます。

ユーティリティー制御ステートメントは、次に示す通常の TSO コマンド構文規則と継続規則に従います。

- 行を継続させるには、その行の最後の文字として正符号 (+) を置きます。
- 各パラメーターをブランクで区切ります。

HALDB に対する DSPXRUN コマンドのコマンド構文は、次のとおりです。

▶—ISPSTART—CMD(—| *command\_string* |)——————▶

**command\_string:**

|—DSPXRUN—EXPORT—DBN(*database\_name*)—DSN(*dataset\_name*)—————▶  
|—IMPORT—|

▶—MEM(*member\_name*)—|—————|  
|—OPT(*processing\_option*)—|

**EXPORT**

バッチ・ジョブを使用してデータベース情報をエクスポートすることを選択すると、情報は、ユーザー指定の区分データ・セットに保管されます。情報は ISPF テーブルとして保管されるため、次のような ISPTLIB データ・セットの属性をもっていなければなりません: レコード・フォーマット = 固定ブロック、レコード長 = 80、データ・セット編成 = PDS (または PDS/E)。

**IMPORT**

バッチ・ジョブを使用してデータベース情報をインポートすることを選択すると、区画情報は、ユーザー指定の区分データ・セットから読み取られます。区画情報は、RECON データ・セットに対して定義されます。

**database\_name**

HALDB 区画定義ユーティリティによりエクスポート・ファイルが作成されたとき、基本パネルで指定されたデータベース名。

**data set\_name**

入力データ・セット名は、区画情報を含むデータ・セットの名前です。このデータ・セットは、区分データ・セットでなければなりません。

**member\_name**

入力メンバー名は、入力データ・セット内のメンバーの名前です。このメンバーは、HALDB 区画定義ユーティリティを使用してエクスポートされていなければなりません。

**processing\_option**

処理オプション・フィールドにより、インポートされたテーブルの区画の処理時にエラーが生じた場合に区画定義ユーティリティが行うことを決めることができます。区画定義ユーティリティは、RECON 内でインポートする各区画を記録します。エラーがある場合は、残りの区画を試行するか、処理を停止するかを選択することができます。有効な値は、下記の 1 または 2 です。

- 1 最初のエラーで停止 (前のインポート済み区画は保存されます)
- 2 全区画を試行

OPT パラメーターは、エクスポート処理時には無視されます。

関連タスク:

🔗 z/OS: オプション・テーブルおよびファイル調整 ISPF ライブラリーの割り振り

---

## %DFSHALDB ユーティリティーの例

これらの例は、%DFSHALDB ユーティリティーの使い方を示しています。

次に示す例は、111 ページの図 22 と同様の JCL が使用されることを前提としています。また、参照点として SYSTSIN ステートメントを含んでいます。

サブセクション:

- 『データベース定義のエクスポート』
- 『エラーがある場合のデータベース定義のインポートの防止』
- 『独立した HALDB のインポート』

### データベース定義のエクスポート

この例では、データベース定義をエクスポートします。

```
//SYSTSIN DD *
  ISPSTART CMD( +
    DSPXRUN EXPORT  +
    DSN('PROD.RSR.PARTS') +
    DBN(IVPDB1) +
    MEM(IVPDB1) +
  )
/*
```

### エラーがある場合のデータベース定義のインポートの防止

この例では、データベース定義をインポートし、エラーがあれば停止します。

```
//SYSTSIN DD *
  ISPSTART CMD( +
    DSPXRUN IMPORT  +
    DSN('PROD.RSR.PARTS') +
    DBN(IVPDB1) +
    MEM(IVPDB1) +
    OPT(1)  +
  )
/*
```

### 独立した HALDB のインポート

この例では、2 つの HALDB をインポートし、エラーが生じた場合でも、他の HALDB 区画に処理を続けます。それぞれのインポートが独立しています。

```
//SYSTSIN DD *
  ISPSTART CMD(DSPXRUN IMPORT DSN('PROD.RSR.HALDB') +
    DBN(IVPDB1) MEM(IVPDB1) OPT(2) )
  ISPSTART CMD(DSPXRUN IMPORT DSN('PROD.RSR.HALDB') +
    DBN(IVPDB2) MEM(IVPDB2) OPT(2) )
/*
```

---

## %DFSHALDB ユーティリティーの実行

HALDB 区画定義ユーティリティーは、主として、対話式アプリケーションとして実行されます。ただし、HALDB 区画定義ユーティリティーのインポート機能およびエクスポート機能をバッチで実行することもできます。

このユーティリティを対話式プログラムとして実行する場合は、ISPF がダイアログ・マネージャーとして使用されます。HALDB 区画定義ユーティリティを使用するには、TSO にログオンし、HALDB ダイアログ・コンポーネントを使用可能にしなければなりません。

HALDB 区画定義ユーティリティは、ISPF 内から開始されます。ISPF コマンド行に以下のコマンドを入力します。

```
TSO %DFSHALDB
```

HALDB 区画定義ユーティリティは、IMS Application Menu から開始することもできます。

---

## HALDB 区画定義ユーティリティの ISPF パネル

HALDB 区画定義ユーティリティの ISPF パネルを使用すると、IMS HALDB の区画を管理できます。

HALDB 区画定義ユーティリティにアクセスするには、以下のようになります。

1. TSO にログオンする。
2. ISPF を開始する。
3. ISPF コマンド行に、tso %dfshaldb をタイプし、Enter キーを押す。

このユーティリティは、HALDB とその区画に対してさまざまなアクションを行ういくつかのパネルとプログラムから構成されています。

**重要:** ここでは、パネル ID は、各パネル・イメージの表題の中で括弧で囲んで示してあります。それぞれのパネルの左上隅にパネル ID を表示させるには、ISPF コマンド行で panelid と入力して、Enter キーを押します。

サブセクション:

- 117 ページの『Partitioned Databases (パーティション・データベース) パネル』
- 118 ページの『ISPF メンバー・リストの表示』
- 119 ページの『HALDB 区画のオープン』
- 130 ページの『データ・セット・グループ情報の定義』
- 132 ページの『定義された区画のリストの表示』
- 140 ページの『データベース情報のオープン』
- 140 ページの『データベース情報の削除』
- 141 ページの『データベース情報のエクスポート』
- 142 ページの『データベース情報のインポート』
- 143 ページの『HALDB データ・セット連結の表示』
- 143 ページの『IMS 構成の選択』



## Partitioned Databases (パーティション・データベース) パネル

取扱対象の HALDB を、Partitioned Databases (パーティション・データベース) パネル上で定義します。実行するアクションのタイプを指定します。例えば、定義、変更、または表示です。後続のパネルが、各種の処理を通じて指示を与えてくれます。

Partitioned Databases (パーティション・データベース) パネルには ポイント・アンド・シュート方式 のテキスト・フィールドがあります (デフォルトでは青緑色)。ポイント・アンド・シュート方式フィールドを使用するためには、単にカーソルをそのテキスト上に置いて Enter キーを押すだけです。

以下の図では、ユーザーが HALDB 名を入力するスペースが与えられており、HALDB が自分自身についての情報を収集することを可能にしています。その情報は、選択するオプションや区画の現在の状態に応じて、DBDLIB か RECON データ・セットから検索することができます。

```
Help
-----
                                Partitioned Databases
Type a database name and choose an option. Then press Enter.
To select a database from a list, type a filter (*) and press F4.
Configuration . . . : DEFAULT
Database name . . . IVPDB1 +
Option . . . . . — 1. OPEN DATABASE partitions
                   2. Open database information
                   3. Delete database information
                   4. Export database definitions
                   5. Import database definitions
                   6. Show IMS DDname concatenation
                   7. Select IMS RECON / DBDLIB libraries

To exit the application, press F3.
Command ==>
F1=Help  F3=Exit  F4=Prompt
```

図 23. Partitioned Databases (パーティション・データベース) パネル (DSPXPAA)

### Configuration

「Configuration」は、一組の DBD ライブラリーおよび一組の RECON データ・セットを識別するためにユーザーが指定した名前です。既に、ログオン・プロシージャーから IMS DD ステートメントを割り振り済みで、しかも IMS.SDFSRESL を STEPLIB DD ステートメントに割り振り済みの場合は、**Configuration** (構成) オプションを使用する必要はありません。構成を定義し選択すると、これらのデータ・セットは、ログオン・プロシージャーからの割り振りをオーバーライドします。

#### データベース名

最大 8 桁までの英数字 (先頭文字は必ず英字にする) を入力します。

HALDB 名は DBDLIB データ・セットのメンバーでなければなりません。

DBDLIB データ・セットは IMS という DD 名のもとに割り振る必要があります。指定するデータベース名は、複数の ISPF セッションにわたって有効です。

アスタリスクを含めることにより、メンバー・リスト画面を要求していることを示すことができます。アスタリスクは単独で入力するか、または表示されるリストを制限するために名前の一部として入力することができます。

重要: メンバー名の一部としてアスタリスクを入れる場合、DD 名 IMS の連結は最大でも 4 つのデータ・セットまでしか含まれません。これは、ISPF の制限です。

### Option

実行する処理のタイプを示す数値。数値は、リスト内のアクションのいずれかと対応します。

### ISPF メンバー・リストの表示

Database name (データベース名) フィールドにアスタリスクを入れた場合、IMS DD 名連結のメンバーのうち、名前がそのフィルターと一致するメンバーのリストが表示されます。サンプルのメンバー・リスト画面が、以下の図に示されています。

```

File Help
-----
MEMBER LIST IMSIVP81.DBDLIB Row 00001 of 00011
Name Size TTR Alias-of AC AM RM ---- Attributes ---
. DBFSAMD1 00000158 00013B 00 24 24
. DBFSAMD2 000001A0 000143 00 24 24
. DBFSAMD3 000006E0 00014B 00 24 24
. DBFSAMD4 000002C8 000207 00 24 24
. DI21PART 00000230 000133 00 24 24
. IVPDB1 00000138 000103 00 24 24
. IVPDB1I 00000138 00010B 00 24 24
. IVPDB2 00000130 000113 00 24 24
. IVPDB3 00000188 00011B 00 24 24
. IVPDB4 00000110 000123 00 24 24
. IVPDB5 000000B0 00012B 00 24 24
**End**
Command =====> Scroll ==> CSR
F1=Help F3=Exit F12=Cancel
  
```

図 24. ISPF のメンバー・リスト (Member List) 画面 (DSPXPAM)

メンバー・リストは、IMS 連結の PDS ディレクトリーから作成されます。表示されるメンバーには、HALDB も HALDB でないものもあります。メンバー・リストは標準の ISPF リストなので、IMS 固有の情報は表示されません。

左端の桁に入力することにより、メンバー・リストから処理すべき HALDB 名を選択できます。選択した名前がパーティション・データベースの名前ではない場合、エラー・メッセージが表示されます。スラッシュ (/) 文字で HALDB 名を選択し、さらに File (ファイル) アクションで実行するアクションを選択します。

Partitioned Database パネル (DSXPAA) で示されているのと同じアクションをここで使用することができます。

「Partitioned Databases (パーティション・データベース)」パネルでオプションを指定すれば、「File」アクション・バーを使用する必要はなく、ただ Enter キーを押すだけになります。「Partitioned Databases (パーティション・データベース)」パネルの「Option」で指定したオプションを、「File」アクション・バーを使用してオーバーライドすることができます。

「Member List (メンバー・リスト)」パネルの HALDB のリストを、「File」アクション・バーを使用して取り扱うことができます。



図 25. 「File」アクション・バーの選択項目

「File」アクション・バーのオプションを使用すれば、以下のアクションを実行することができます。

- HALDB 区画の作成または変更。
- HALDB 情報の表示または変更。
- HALDB 情報の削除。
- HALDB 情報のエクスポート。
- HALDB 情報のインポート。

## HALDB 区画のオープン

HALDB 用に区画を定義する前に、DBDGEN 処理を使用して、HALDB をパーティション・データベースとして定義する必要があります。

HALDB を最初に選択する場合には、HALDB マスターの値を設定しなければなりません。継続するために Enter キーを押すと、区画用にデフォルトが設定されます。再び Enter キーを押すと、これらのデフォルトを使用して区画が定義されます。各区画について、作成時に一意的に変更することも、後で区画のリストから変更することもできます。127 ページの図 29 は、区画情報を指定するパネルの例を示します。

区画の初期セットが定義された後は (そしてその HALDB を再度選択するときはいつでも)、「Database Partitions (データベース区画)」画面を見ることになります。

**重要:** 120 ページの図 26 のパネルに最初に表示される情報のほとんどは、DBDLIB メンバーから抽出されたものです。表示された情報を変更することはできませんが、その情報は RECON データ・セットに保管され、DBDLIB メンバーには保管されません。

各 HALDB は最大 1001 区画をサポートすることができます。

```

Help
-----
                                Partitioned Database Information
Type the field values. Then press Enter to continue.
Database name . . . . . : IVPDB1
                                Master Database values
Part. selection routine . . . DFSIVD1
RSR global service group . . . BKUPGRP1
RSR tracking type . . . . . DBTRACK
Share level . . . . . 0
Database organization . . . : PHDAM
Recoverable? . . . . . はい
Number of data set groups . : 10
Online Reorganization Capable: Yes
To exit the application, press F3.
Command ==>
    F1=Help    F3=Exit    F12=Cancel

```

図 26. *Partitioned Database Information* (パーティション・データベース情報) パネル (DSPXPOA)

次に挙げるのは、「Partitioned Database Information (パーティション・データベース情報)」画面上のフィールドの説明です。

**データベース名**

- 1 文字から 8 文字までの英数字を入力します。これは、ユーザーが以前のパネルで選択した名前であり、定義中の HALDB の名前です。

**Part. selection routine**

- 1 文字から 8 文字までの英数字 (先頭文字は必ず英字にする) を入力します。ユーザーが提供した区画選択出口ルーチンの名前です。

**RSR global service group**

- 1 文字から 8 文字までの英数字 (先頭文字は必ず英字にする) を入力します。これは、HALDB の割り当て先となる RSR グローバル・サービス・グループを指定するのに使用されるオプション・パラメーターです。

**RSR tracking type**

これは、グローバル・サービス・グループに割り当てられる区画のための、RSR トラッキング (シャドーイング) のタイプを指定するのに使用されるオプション・パラメーターです。 RSR グローバル・サービス・グループが HALDB マスター用に定義されていないと、タイプ RCVTRACK または DBTRACK を指定することはできません。

- DBTRACK- HALDB 作動可能トラッキングが行われることを示します。
- RCVTRACK- リカバリー作動可能トラッキングが行われることを示します。

デフォルトは、DBTRACK です。

**Share level**

0、1、2、または 3。 Share level は、オプション・パラメーターです。共用レベルを使用して、許可サブシステムとの HALDB データ共用のレベルを指定できます。

デフォルトは、共用レベル 0 です。

## Database organization

このフィールドは HALDB 編成のタイプを示し、PSINDEX、PHIDAM、または PHDAM のいずれかを指定できます。

## Recoverable?

「Yes」は、HALDB がリカバリー可能であることを示します。「No」は、HALDB がリカバリー可能でないことを示します。デフォルトは、「Yes」です。

RSR グローバル・サービス・グループが指定されている場合、「Recoverable」フィールドは「Yes」にする必要があります。

## Number of data set groups

これは、DBDGEN で指定されたとおりの、データが入っているグループ内のデータ・セットの数です。

## Online reorganization capable

「Yes」は、この HALDB がオンライン再編成をサポートすることを指定します。No は、この HALDB がオンライン再編成をサポートしないことを指定します。これらの指定は DBRC RECON データ・セットに保管されます。

以下の図は、区画デフォルト情報を示したものです。

```
Help
-----
                                Partition Default Information
Type the field values. Then press Enter to continue.
Database name . . . . . : IVPDB1
                                Processing options
Automatic definition . . . . . No
Input dataset . . . . . 'IMS.IVPDB1.KEYS'
Use defaults for DS groups. . . . . No
                                Defaults for partitions
Partition name . . . . . IVPD101
Data set name prefix . . . . . IMS.DB01.FINANCE.YEAR2013.CURR Randomizer
Module name . . . . . DD41DUP2
Anchor . . . . . 2
High block number. . . . . 999
Bytes . . . . . 2000
Free Space
Free block freq. factor. . . 0
Free space percentage. . . 0
                                Defaults for data set groups
Block Size . . . . . 8192
DBRC options
Max. image copies. . . . . 2
Recovery period. . . . . 0
Recovery utility JCL . . . RECOVJCL
Default JCL. . . . .
Image copy JCL . . . . . ICJCL
Online image copy JCL. . . OICJCL
Receive JCL. . . . . RECVJCL
Reusable? . . . . . No
To exit the application, press F3.
Command ==>
F1=Help F3=Exit F6=Groups F12=Cancel
```

図 27. Partition Default Information (区画デフォルト情報) パネル (DSPXPCA)

重要: 「Randomizer」セクションは、HALDB が PHDAM である場合にのみ存在します。「Defaults for data set groups」セクションは、DBDGEN でただ 1 つだ

けのデータ・セット・グループが指定されている場合にのみ存在します。複数のデータ・セット・グループがある場合は、「F6=Groups」を使用して、130 ページの『データ・セット・グループ情報の定義』に説明されているダイアログを使い、すべてのデータ・セット・グループを表示してください。

次に挙げるのは、Partition Default Information (区画デフォルト情報) 画面上のフィールドについての説明です。

#### データベース名

これは、ユーザーが前出のパネルで選択した名前 (117 ページの図 23 を参照) であり、定義する HALDB の名前です。

#### Automatic definition

この値は、「Yes」または「No」です。「Yes」を指定すると、区画名 (プレースホルダーとして % 記号文字を入れる必要がある) に応じたユーザーの選択を基にして、区画が自動的に定義されます。

No (いいえ) を指定すると、各区画について固有値を指定することができます。

デフォルトは、Yes (はい) です。

#### Input data set

z/OS データ・セットの名前を与えます。PDS であれば、メンバー名を指定します。データ・セットの各行には、区画選択ストリング、または区画定義で使用されることになるハイ・キー値を入れる必要があります。

#### Use defaults for DS groups

この値は、「Yes」または「No」です。このオプションは、すべてのデータ・セット・グループを自動的に同じデフォルトに設定するか、それぞれのグループごとに値を入れるためのプロンプトを受けるか、を指定します。

「automatic definition」が「Yes」に設定されている場合は、ブランクのままにしておけます。

#### Partition name

1 文字から 7 文字までの英数字 (先頭文字は必ず英字にする) を入力します。Partition name (区画名) はそのデータ・セットの DD 名の接頭部として使用されるので、固有でなければなりません。

「automatic definitions」の場合、英数字シーケンス番号 (A から Z、0 から 9) のプレースホルダーとして、% 記号を入れる必要があります。

#### Data set name prefix

最大長 37 文字の JCL に有効な任意の英数字名。

#### Module name

1 文字から 8 文字までの英数字 (先頭文字は必ず英字にする) を入力します。これは、ランダム化モジュールの名前です。ランダム化モジュールは、ルート・セグメントを PHDAM HALDB に置いたり、PHDAM HALDB から取り出したりするための制御をします。このパラメーターは PHDAM HALDB 専用です。

#### Anchor

1 から 255 までの範囲の、1 から 3 桁の数値。PHDAM HALDB のルート・アドレス可能域にある制御インターバルまたはブロック内に要求され

る、ルート・アンカー・ポイントの数を指定します。Anchor (アンカー) オペランドは、符号なし 10 進整数でなければならず、255 を超えてはなりません。標準的な値は、1 から 5 です。このパラメーターは PHDAM HALDB 専用です。

デフォルト値は 1 です。

#### High block number

0 から  $2^{24} - 1$  までの範囲の、符号なし 10 進整数値。この値は、この HALDB のためにランダム化モジュールが生成できるようにする相対ブロック番号の最大値を指定します。このパラメーターは PHDAM HALDB 専用です。この値は、PHDAM HALDB のルート・アドレス可能域内の制御インターバルまたはブロックの数を決定します。

最大ブロック番号が 0 の場合は、ランダム化モジュールによって作成される RBN に上限チェックが実行されないことを意味します。すなわち、すべてルート・アドレス可能域です。

**Bytes** 1 から  $2^{24} - 1$  までの範囲の、符号なし 10 進整数値。この値は、別の HALDB レコードへの呼び出しによって中断されない一連の挿入呼び出しで、ルート・アドレス可能域に保管できる HALDB レコードの最大バイト数を指定します。

値 0 (ゼロ) は、すべてのバイトがアドレス可能であることを意味します。これは、DBDGEN の DBD マクロ・ステートメントの中の RMNAME キーワードから bytes パラメーターを削除することと同等です。このパラメーターは PHDAM HALDB 専用です。

#### Free block freq. factor

1 を除く、0 から 100 までの、符号なし 10 進整数値。「free block frequency factor (fbff)」は、HALDB のロードまたは再編成の際、このデータ・セット・グループ内で、 $n$  番目の制御インターバルまたはブロックごとにそれをフリー・スペースとして残すことを指定します (ここで、 $fbff=n$ )。fbff の範囲は、0 から 100 までの間の 1 を除くすべての整数値です。fbff のデフォルト値は 0 です。

#### Free space percentage

0 から 99 の範囲の、2 桁の符号なし 10 進整数。fspf はフリー・スペース・パーセント係数 (free space percentage factor) のことです。このデータ・セット・グループで、各制御インターバルまたは制御ブロックの最低何% をフリー・スペースとして残すかを指定します。

fspf のデフォルト値は 0 です。

#### ブロック・サイズ

1 から 32000 までの範囲の、符号なし偶数 10 進整数。ブロック・サイズの値を使用するのは OSAM だけです。初期値 4096 が表示されます。HALDB が OSAM でなければ、block size (ブロック・サイズ) フィールドは表示されません。

#### Max. イメージ・コピー

識別された DBDS のために DBRC が維持するイメージ・コピーの数を指定するのに使用する必要パラメーター。値は、2 から 255 までの符号なし 10 進整数にする必要があります。

### Recovery period

これは、指定された DBDS 用のイメージ・コピーのリカバリー期間を指定するのに使用されるオプション・パラメーターです。

イメージ・コピーに関する情報が RECON 内に保持される日数を示す 0 から 999 までの符号なし 10 進整数。0 を指定すると、リカバリー期間はありません。デフォルトは 0 です。

### Recovery utility JCL

1 文字から 8 文字までの英数字 (先頭文字は必ず英字にする) を入力します。これは、基幹 JCL の区分データ・セットのメンバー名を指定するのに使用されるオプション・パラメーターです。GENJCL.RECOV コマンドを発行すると、DBRC は、識別された DBDS に対してデータベース・リカバリー・ユーティリティを実行するため、このメンバーを使用して JCL を生成します。

デフォルトのメンバー名は RECOVJCL です。

### Default JCL

1 文字から 8 文字までの英数字 (先頭文字は必ず英字にする) を入力します。これは、DBDS 用に暗黙の基幹 JCL デフォルト・メンバーを指定するのに使用されるオプション・パラメーターです。指定されたメンバーは、ユーザーが定義したキーワードを解決するために GENJCL.IC、GENJCL.OIC、および GENJCL.RECOV コマンドによって使用されます。

### Image copy JCL

1 文字から 8 文字までの英数字 (先頭文字は必ず英字にする) を入力します。これは、基幹 JCL が入っている区分データ・セットのメンバー名を指定するのに使用されるオプション・パラメーターです。GENJCL.IC コマンドを発行すると、DBRC は、このメンバーを使用して JCL を生成して、識別された DBDS に対してデータベース・イメージ・コピー・ユーティリティを実行させます。

デフォルトのメンバー名は ICJCL です。

### Online image copy JCL

1 文字から 8 文字までの英数字 (先頭文字は必ず英字にする) を入力します。これは、基幹 JCL が入っている区分データ・セットのメンバー名を指定するのに使用されるオプション・パラメーターです。GENJCL.OIC コマンドを発行すると、DBRC は、このメンバーを使用して、識別された DBDS に対してオンライン・データベース・イメージ・コピー・ユーティリティを実行する JCL を生成します。

デフォルトのメンバー名は OICJCL です。

### Receive JCL

1 文字から 8 文字までの英数字 (先頭文字は必ず英字にする) を入力します。これは、GENJCL.RECEIVE コマンドに使用される基幹 JCL メンバー名を指定するのに使用するオプション・パラメーターです。

デフォルトのメンバー名は RECVJCL です。

### Reusable?

この値は、「Yes」または「No」です。データベース・イメージ・コピー・



ユーティリティーまたはオンライン・データベース・イメージ・コピー・ユーティリティーが、以前に定義したイメージ・コピー・データ・セットを再使用するかどうかを指定します。

デフォルト値は「No」です。

#### 自動区画定義

「Partition Default Information (区画デフォルト情報)」パネルでは、「*Automatic definition*」に「yes」を設定でき、その結果、介入なしに区画を定義できます。前もってデータ・セットを作成しておき、そこに区画選択ストリングを入れておく必要があります。パネルの「*input data set*」フィールドにデータ・セットの名前を指定してください。

入力データ・セットの各行には、区画選択ストリング、または区画定義で使用されることになるハイ・キー値を入れておく必要があります。このファイルでは、各行にただ 1 つの値だけを左寄せに入れておかなければなりません。ストリングの長さは、最後の非空白文字によって決められます。各レコードにはただ 1 つのストリングだけを入れる必要があります。

「Partition name」フィールドに、英数字シーケンス番号 (A から Z、0 から 9) のプレースホルダーとして、% 記号を入れてください。「partition name」として下記のように入力するとします。

```
Partition name . . . . . IVPD1%
```

区画は以下の順序で作成されます。

```
IVPD1AA
IVPD1AB
IVPD1AC
.
.
IVPD1AZ
IVPD1A0
IVPD1A1
IVPD1A2
.
.
IVPD1A9
IVPD1BA
IVPD1BB
IVPD1BC
.
.
```

Enter キーを押すと、入力データ・セットにあるキー値の数だけの区画が自動的に生成されます。

データベースの拡張時に命名順序を保持できる区画名を生成したければ、IVP1%%A のような区画名を指定することができます。この場合、区画は次の順序で作成されます。

```
IVP1AAA
IVP1ABA
.
.
IVP1AZA
IVP1A0A
IVP1A1A
```

・  
・  
IVP1A9A  
IVP1BAA  
・  
・

自動定義の処理中、(以下の図に示すように) 状況表示パネルが表示されます。この「Automatic definition status (自動定義状況)」パネルは、新規の区画が定義されるたびに更新されます。

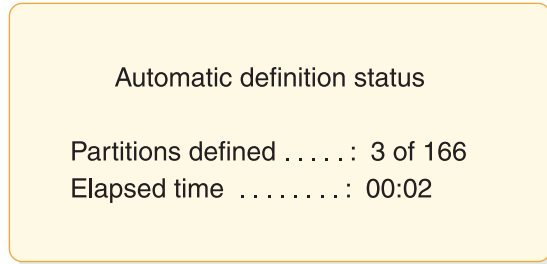


図 28. Automatic definition status (自動定義状況)

自動定義が完了すると、「Database Partitions (データベース区画)」パネルでは、区画選択ストリングにユーザーの入力データ・セットの情報が書き込まれています。

#### 手動区画定義

「Partition Default Information (区画デフォルト情報) パネルでは、**Automatic definition**」に「No」を設定でき、その結果、連続して区画を定義できます。

「**Automatic definition**」に「No」と設定した場合でも、「input data set」を使用することができます。

- 入力データ・セットを指定する場合、前もってデータ・セットを作成しておき、そこに区画選択ストリングを入れておく必要があります。「Change Partition」パネル (DSPXPPA) の「Partition selection string」フィールドは、ユーザーの入力データ・セットの内容から事前準備をしておきます。区画ごとに、「partition selection string」は、入力データ・セットのレコードから満たされます。キー値の数を超えて区画を定義しようとする場合は、入力データ・セットからの最後のキー値が「Change Partition (区画変更)」パネルに表示されるので、それを手動で変更する必要があります。
- 区画ハイ・キー値を提供するのに入力データ・セットを指定しない場合、区画ハイ・キー値は各区画ごとに手動で追加することができます。
  - 区画選択出口を指定しなかった場合は、区画ハイ・キー値が必要になります。
  - 区画選択出口を指定した場合は、区画選択ストリング値はオプションです。

デフォルトを設定し、Enter キーを押すと、「partition definition (区画定義)」画面が表示されます。フィールドを変更することができ、そして Enter キーを押して区画を定義できます。Enter キーを押すと、区画は RECON の中に定義され、partition definition (区画定義) パネルが再び表示されるのでさらに区画を定義でき

ます。区画 ID は、区画が定義されるたびに増分されます。表示されている区画が定義されないようにするには、取り消しキー (PF12) を押してください。

PF12 を押して新規区画の定義を停止すると、「Partitioned Databases (パーティション・データベース)」パネルが再び表示されます。新規区画の定義を停止するのに「F11=List」を押す選択も可能です。その場合、定義済みの区画のリストが表示されます。

```

Help
-----
                                Change Partition
Type the field values. Then press Enter.
Database name . . . . . : IVPDB1
Partition name . . . . . : IVPD101
Partition ID. . . . . : 1
Data set name prefix. . . . : IMS.DB01.FINANCE.YEAR2013.CURR
Partition Status. . . . . : _____

Partition Selection String
+00 F2F0F0F3 4BF2F2F4 40F1F77A F2F57AF0 | 2013.224 17:25:0
+10 F94BF6F3 F3F12432 00000000 00001020 | 9.6331.....
+20 A840C1A5 85404040 40E28195 40D196A2 | y Ave San Jos
+30 856B40C3 C14040F9 F5F1F4F1 00100020 | e, CA 95141....
+40 00050000 40F0F34B F0F3F440 00000100 | .... 03.034 ...
+50 F1F8F0F0 C9C2D4E2 C5D9E540 40C9C2D4 | 1800IBMSERV IBM
+60 40C39699 974B4040 F5F5F540 C2818993 | Corp. 555 Bail
+70 A840C1A5 85404040 40E28195 40D196A2 | y Ave San Jos
+80 856B40C3 C14040F9 F5F1F4F1 00403010 | e, CA 95141. ..
+90 00010500 40F0F34B F2F4F340 00324020 | .... 03.243 ..
+A0 9201913C D2FE933D 913C1F66 4360A005 | k.j.K.l.j.....-..
+B0 3233A200 D996A281 6BD785A3 85996B40 | ..s.Rosa,Peter,
+C0 000080D4 81A3A3F9 71C4C6F8 F1F4C6C2 | ...Matt9.DF814FB
+D0 9311913C F6F4F8F6 943C1F66 4360A005 | l.j.6486m.....-..
+E0 41E3453C 06000045 10110220 10416220 | .T.....
+F0 FFFF900 00004920 18007410 94000300 | ..9.....m...

Randomizer
Module name . . . . . : DD41DUP2
Anchor . . . . . : 2
High block number. . . . : 999
Bytes . . . . . : 2000

Free Space
Free block freq. factor. . . 0
Free space percentage. . . 0

Attributes for data set group A
Block Size . . . . . : 8192

DBRC options
Max. image copies. . . . 2
Recovery period. . . . 0
Recovery utility JCL . . RECOVJCL
Default JCL. . . . .
Image copy JCL . . . . ICJCL
Online image copy JCL. . OICJCL
Receive JCL. . . . . RECVJCL
Reusable? . . . . . No

Command ==>>
F1=Help F3=Exit F5=String F6=Groups F12=Cancel

```

図 29. Change Partition (区画変更) パネル (DSPXPPA)

重要: 「Randomizer」セクションは、HALDB が PHDAM である場合にのみ存在します。「data set group attributes」セクションは、DBDGEN でただ 1 つだけのデータ・セット・グループが指定されている場合にのみ存在します。複数のデー

タ・セット・グループがある場合、「F6=Groups」を使用してすべてのデータ・セット・グループを表示し、130 ページの『データ・セット・グループ情報の定義』に説明されているダイアログを使用してください。

次に挙げるのは、Change Partition (区画変更) 画面上のフィールドの説明です。

### Partition ID

1 から 32 767 までの数値。ただし、この HALDB の現行の高位区画 ID 値より小さい値です。区画定義ユーティリティーは、区画を手動でまたは自動的に作成するかどうかに関係なく、ユーザー用に区画 ID を生成します。DBRC は、この数値を RECON データ・セットの中に記録します。データ・セット名には、データ・セット名が属する区画の区画 ID が組み込まれます。

ID が区画に割り当てられた後では、その ID を変更することはできません。

### Partition Status

「Partition Status」フィールドに「disable」と入力することにより、区画を使用不可にすることができます。通常は、区画を削除前に使用不可にするだけで済みます。

使用不可にされた区画を使用可能にするためには、「Partition Status」フィールドに「enable」を入力してください。

### Partition high key

「Partition High Key」フィールドにより、区画に入れることができる最高位データベース・レコード・ルート・キーを指定することができます。区画ハイ・キーは、ご使用のシステムにより決定されます。IMS は、区画ハイ・キーを 16 進値として扱います。「Partition High Key」フィールドに値を入力する必要があります。

「Partition High Key」フィールドの長さは、DBD 定義時に FIELD ステートメントの BYTES= パラメーターを使用して指定するルート・キー長によって決められます。区画ハイ・キー長がルート・キー長より長い場合、エラー・メッセージが表示され、区画ハイ・キー長を削減する必要があります。区画ハイ・キー長が、定義されたルート・キー長より短い場合は、区画定義ユーティリティーは、ハイ・キー値を、定義されたルート・キー長まで 16 進数 'FF' で埋め込みます。区画ハイ・キー値は、HALDB 内の各区画ごとに固有である必要があります。

「Partition High Key」フィールドは 2 つのセクションから構成されます。すなわち、16 進形式で区画ハイ・キーを表示する左側の編集可能セクションと、英数字形式で区画ハイ・キーを表示する右側の表示専用セクションです。

「Partition High Key」フィールドの左側のセクションに 16 進値を直接入力することができます。区画定義ユーティリティーは、Partition High Key (区画ハイ・キー) フィールドの右側のセクションにこの値と同等の英数字の値を表示します。

ISPF エディターを使用して英数字の値を直接入力することができます。ISPF エディターにアクセスするには、F5 を押します (16 進数セクションに値をすでに入力済みの場合は、F5 を 2 回押してください)。英数字の値

が入力されると、それと同等の 16 進数値が「Partition High Key」フィールドの左側のセクションに表示されます。

英数字の値は、任意の文字情報で構成することができます。英数字の値に非表示文字が入っている場合は、16 進表記を使用してこれらの文字を示す必要があります。ISPF エディターでは、16 進文字ストリングは、単一引用符で囲み、x を接頭部か接尾部に付けます (例えば、X'c1f201ffff')。

### Partition selection string

HALDB マスター定義で区画選択ルーチンを指定した場合は、「**Change Partition (区画変更)**」パネルには、「**Partition Selection String**」フィールドのみが表示されます。区画選択ルーチンは、16 進形式の区画選択ストリングを使用して、HALDB 内の各区画にわたってレコードを分散します。

区画選択ストリングの長さは、256 バイトです。256 バイトより短い長さの区画選択ストリングを入力した場合は、区画定義ユーティリティーは、残りのバイトに X'00' を埋め込みます。

**Partition Selection String (区画選択ストリング)** フィールドは 2 つのセクションから構成されます。すなわち、16 進形式で区画選択ストリングを表示する左側の編集可能セクションと、英数字形式で区画選択ストリングを表示する右側の表示専用セクションです。

「**Partition Selection String**」フィールドの左側のセクションに 16 進値を直接入力することができます。区画定義ユーティリティーは、**Partition Selection String (区画選択ストリング)** フィールドの右側のセクションにこの値と同等の英数字の値を表示します。

ISPF エディターを使用して英数字形式で区画選択ストリングを入力することができます。ISPF エディターにアクセスするには、F5 を押します (16 進数セクションに値をすでに入力済みの場合は、F5 を 2 回押してください)。英数字ストリングを入力した後、それと同等の 16 進数値が

「**Partition Selection String**」フィールドの左側のセクションに表示されず。

英数字ストリングは、任意の文字情報で構成することができます。英数字ストリングに非表示文字が入っている場合は、16 進表記を使用してこれらの文字を示す必要があります。ISPF エディターでは、16 進文字ストリングは、単一引用符で囲み、x を接頭部か接尾部に付けます。例: X'c1f201ffff'。

### F5=String

F5 は、2 つの機能を実行します。最初に、新規データが「**Partition High Key**」フィールドまたは「**Partition Selection String**」フィールドの 16 進数セクションに入力された場合は、F5 を押すと、そのデータが区画定義ユーティリティーに渡され、16 進数ストリングと同等の英数字の値がそのフィールドの右側のセクションに表示されます。次に、16 進数セクションに非コミット・データが存在しない場合は、英数字エディターが表示されます。以下の図は、「**Partition Selection String**」フィールド用に表示されるエディター・パネルの例です。

```

EDIT      Partition Selection String
Database name . . . . : IVPDB1
Partition name . . . . : IVPD101
***** ***** Top of Data *****
=COLS>  ----+----1----+----2----+----3----+----4----+----5---

000001 '546787789af'x
***** ***** Bottom of Data *****
Command ==>
          F1=Help  F3=Exit

```

図 30. Selection String Editor (選択ストリング・エディター) (DSPXPKE)

### F6=Groups

F6 を押すと、「Data set group (データ・セット・グループ)」ダイアログが表示されます。

### F11=List

F11 を押すと、「Database Partitions (データベース区画)」パネルが表示されます。

DBDLIB からの HALDB の定義で許可されているデータ・セット・グループが 1 つのみである場合、「Attributes for data set group A」セクションが表示されます。複数グループが割り当てられている場合、グループを処理するために PF6 を押すことを促す注意が表示されます。

## データ・セット・グループ情報の定義

データ・セット・グループ情報の定義は、「Change Partition (区画変更)」パネルで F6 を押すことによって行うことができます。このセクションでは、データ・セット・グループ情報の定義方法を説明します。

HALDB に複数データ・セット・グループを定義し、自動定義を使用しない場合には、データ・セット・グループ・リストを使用してください。

データ・セット・グループ・リストから、「list」列の値に上書きすることにより、各メンバーごとの属性を変更することができます。リストには「all」行という特別な行があり、リストのすべての列を変更することができます。「all」行に値を入力して Enter キーを押すと、入力した値はそのグループのメンバーすべてに波及します。変更が行われた後、「all」列はブランクで消去されます。

**重要:** 変更を保管するには F9 を押し、その後で直前のパネルに戻るには F12 を押します。

リストには「action」列があります。許可されている唯一のアクションは、特定のグループについての情報をすべて表示することです。「Act」列にスラッシュ (/) を入力することによって、グループを選択します。以下の図は、既存のデータに上書きして Enter キーを押すことにより値を変更する方法を示しています。

```

Help
-----
                                Change Dataset Groups                Row 1 to 11 of 11
Select an item by pressing a '/' on the desired line then press Enter.
Database name . . . . . : IVPDB1
Partition name . . . . . : IVPD101
Partition ID. . . . . : 1
Data set name prefix. . . . : IMS.DB01.FINANCE.YEAR2013.CURR
Act  Group   Block   Max Image  Recovery  Recovery  Default
-----  ---  -----  -----  -----  -----  -----
   A      A      8192      2          0          RECOVJCL  _____
   B      B      8192      2          0          RECOVJCL  _____
   C      C      8192      2          0          RECOVJCL  _____
   D      D      8192      2          0          RECOVJCL  _____
   E      E      8192      2          0          RECOVJCL  _____
   F      F      8192      2          0          RECOVJCL  _____
   G      G      8192      2          0          RECOVJCL  _____
   H      H      8192      2          0          RECOVJCL  _____
   I      I      8192      2          0          RECOVJCL  _____
   J      J      8192      2          0          RECOVJCL  _____
Command ==>
F1=Help  F3=Exit  F7=Backward  F8=Forward  F9=Save  F11=Right  F12=Cancel

```

図 31. Change data set groups (データ・セット・グループ変更) パネル、その 1 (DSPXPGA)

```

Help
-----
                                Change Dataset Groups                Row 1 to 11 of 11
Select an item by pressing a '/' on the desired line then press Enter.
Database name . . . . . : IVPDB1
Partition name . . . . . : IVPD101
Partition ID. . . . . : 1
Data set name prefix. . . . : IMS.DB01.FINANCE.YEAR2013.CURR
Act  Group   Image   On. Image  Receive  Reusable?
-----  ---  -----  -----  -----  -----
   A      A      ICJCL   OICJCL   RECVJCL  No
   B      B      ICJCL   OICJCL   RECVJCL  No
   C      C      ICJCL   OICJCL   RECVJCL  No
   D      D      ICJCL   OICJCL   RECVJCL  No
   E      E      ICJCL   OICJCL   RECVJCL  No
   F      F      ICJCL   OICJCL   RECVJCL  No
   G      G      ICJCL   OICJCL   RECVJCL  No
   H      H      ICJCL   OICJCL   RECVJCL  No
   I      I      ICJCL   OICJCL   RECVJCL  No
   J      J      ICJCL   OICJCL   RECVJCL  No
Command ==>
F1=Help  F3=Exit  F7=Backward  F8=Forward  F9=Save  F11=Right  F12=Cancel

```

図 32. 「Change data set groups (データ・セット・グループ変更)」 パネル、その 2 (DSPXPGB)

```

Help
-----
                                Change a Dataset Group
Enter values, then press Enter.
Attributes for data set group B
  Block size . . . . . 8192

DBRC options
  Max. image copies . . . . 2
  Recovery period . . . . . 0
  Recovery utility JCL . . . RECOVJCL
  Default JCL . . . . . _____
  Image copy JCL . . . . . ICJCL
  Online image copy JCL . . OICJCL
  Receive JCL . . . . . RECVJCL
  Reusable? . . . . . No
Command ==>
  F1=Help  F3=Exit  F12=Cancel

```

図 33. Change a Data Set Group (データ・セット・グループ変更パネル) (DSPXPGD)

### 定義された区画のリストの表示

「Partitioned Databases (パーティション・データベース)」パネルで、「Open database partitions」を選択すると、「Database Partitions (データベース区画)」リストが表示されます。HALDB のために定義された区画がすでにある場合、リストは即時に表示されます。あるいは、もはや区画のない HALDB のために区画を定義した後、リストが表示されます。以下の図は、Database Partitions リストの例です。リストは、スクロールアップおよびスクロールダウンできる表として表示されます。

```

File Edit View Help
-----
                                Database Partitions           Row 1 to 15 of 166
Select an item by pressing a '/' on the desired line then press Enter.
Database name . . . . . : IVPDB1
Act  Name      Id      Data set name prefix           Status
---  ---      --      ---
___  IVPD101     1      IMS.DB01.FINANCE.YE2013
___  IVPD102     2      IMS.DB01.PAYROLL.YE2013
___  IVPD103     3      IMS.DB01.PAYROLL.YE2013
___  IVPD104     4      IMS.DB01.PAYROLL.YE2013
___  IVPD105     5      IMS.DB01.PAYROLL.YE2013
___  IVPD106     6      IMS.DB01.PAYROLL.YE2013
___  IVPD107     7      IMS.AB01.PAYROLL.YE2013
___  IVPD108     8      IMS.DB01.PAYROLL.YE2013
___  IVPD109     9      IMS.DB01.PAYROLL.YE2013
___  IVPD110    10      IMS.AB01.PAYROLL.YE2013      Disabled
___  IVPD111    11      IMS.DB01.PAYROLL.YE2013
___  IVPD112    12      IMS.DB01.PAYROLL.YE2013
___  IVPD113    13      IMS.TP01.PAYROLL.YE2013
___  IVPD114    14      IMS.DB01.PAYROLL.YE2013
___  IVPD115    15      IMS.DB01.FINANCE.YE2013
Command ==>
  F1=Help  F3=Exit  F7=Backward  F8=Forward  F11=Right

```

図 34. 区画 ID でソートされた、Database Partitions (データベース区画) パネル (DSPXPLA)

「Database Partitions (データベース区画)」リスト・パネルでは、先頭に HALDB 名、その下にテーブル情報があります。テーブルの各列の説明は以下にリストされています。



**Act** これは、open、copy、およびその他のコマンドを呼び出すことができる行コマンドの入力フィールドです。

**名前** Name (名前) 列には、区画の定義の間に与えられた区画名が入ります。これは最初のソート・シーケンスです。

**Id** これは、区画 ID 番号です。この番号は順次である必要はありません。

#### **Data set name prefix**

「Data set name prefix」には、区画の定義の間に与えられたデータ・セットの名前が入ります。

#### **Status**

区画を選択して「Change Partition (区画変更)」パネルの「Partition Status」フィールドに「disable」を入力することにより、区画を使用不可にすることができます。区画が使用不可にされると、「Disabled」が「Database Partitions (データベース区画)」パネル内の該当の区画の「Status」列に表示されます。使用可能な区画の場合は、その列がブランクのままになります。

「Database Partitions (データベース区画)」リスト・パネルから、個別の区画を処理することができます。「File」アクション・バーを使用するには、処理対象の区画の「act」行コマンド列にスラッシュ (/) を入力し、それからカーソルを「File」アクション・バー選択に置き、そして Enter キーを押します。実行したいアクションを、番号を入力するか、または選択項目上にカーソルを位置付けて選択し、それから再び Enter キーを押します。

PF11 キーを押すことにより、「Database Partitions (データベース区画)」パネルを呼び出して値を表示することができます。

```

File Edit View Help
-----
                                Database Partitions                Row 1 to 4 of 166
Select an item by pressing a '/' on the desired line then press Enter.
Database name . . . . . : IVPDB1
Act Name
      Partition Selection String
----- IVPD001
      +00 F2F0F0F3 4BF2F2F4 40F1F77A F2F57AF0 | 2013.224 17:25:0
      +10 F94BF6F3 F3F12432 00000000 00001020 | 9.6331.....
      +20 A840C1A5 85404040 40E28195 40D196A2 | y Ave San Jos
      +30 856B40C3 C14040F9 F5F1F4F1 00100020 | e, CA 95141....
      +40 00050000 40F0F34B F0F3F440 00000100 | ... 03.034 ...
      +50 F1F8F0F0 C9C2D4E2 C5D9E540 40C9C2D4 | 1800IBMSERV IBM
      +60 40C39699 974B4040 F5F5F540 C2818993 | Corp. 555 Bail
      +70 A840C1A5 85404040 40E28195 40D196A2 | y Ave San Jos
      +80 856B40C3 C14040F9 F5F1F4F1 00403010 | e, CA 95141. . .
      +90 00010500 40F0F34B F2F4F340 00324020 | ... 03.243 . . .
      +A0 9201913C D2FE933D 913C1F66 4360A005 | k.j.k.l.j....-.
      +B0 3233A200 D996A281 6BD785A3 85996B40 | ..s.Rosa,Peter,
      +C0 000080D4 81A3A3F9 71C4C6F8 F1F4C6C2 | ..Matt9.DF814FB
      +D0 9311913C F6F4F8F6 943C1F66 4360A005 | l.j.6486m.....
      +E0 41E3453C 06000045 10110220 10416220 | .T.....
      +F0 FFFF9000 00004920 18007410 94000300 | ..9.....m...
----- IVPD002
      +00 F2F0F0F3 4BF2F2F4 40F1F87A F1F27AF0 | 2013.224 18:12:0
      +10 F94BF6F3 F3F12432 00000000 00001020 | 9.6331.....
Command ==>
F1=Help F3=Exit F7=Backward F8=Forward F11=Right

```

図 35. キーでソートされた、Database Partitions (データベース区画) パネル (DSPXPLB)

「Database Partitions」リスト・パネルでは、先頭に HALDB 名、その下にテーブル情報があります。列には以下のものがあります。

**Act** これは、open、copy、およびその他のコマンドを呼び出すことができる行コマンドの入力フィールドです。

**名前** 「Name」列には、区画の定義の間に与えられた区画名が入ります。これは最初のソート・シーケンスです。

**Partition selection string**

「Partition selection string」は、区画選択ルーチンにより使用されます。

PF11 キーを押すことにより、「Database Partitions (データベース区画)」パネルを呼び出して「Randomizer」値を表示することができます。

```

File Edit View Help
-----
                                Database Partitions          Row 1 to 15 of 166
Select an item by pressing a '/' on the desired line then press Enter.
Database name . . . . . : IVPDB1
----- Randomizer ----- - Free Space -
Act Name      Module  Anchor High block Bytes  FBFF  FSPF
---  ---      ---    ---     ---     ---   ---
---  IVPD101  DD41DUP2  2      999     2000   0     0
---  IVPD102  DD41DUP2  2      999     2000   0     0
---  IVPD103  DD41DUP2  2      999     2000   0     0
---  IVPD104  DD41DUP2  2      999     2000   0     0
---  IVPD105  DD41DUP2  2      999     2000   0     0
---  IVPD106  DD41DUP2  2      999     2000   0     0
---  IVPD107  DD41DUP2  2      999     2000   0     0
---  IVPD108  DD41DUP2  2      999     2000   0     0
---  IVPD109  DD41DUP2  2      999     2000   0     0
---  IVPD110  DD41DUP2  2      999     2000   0     0
---  IVPD111  DD41DUP2  2      999     2000   0     0
---  IVPD112  DD41DUP2  2      999     2000   0     0
---  IVPD113  DD41DUP2  2      999     2000   0     0
---  IVPD114  DD41DUP2  2      999     2000   0     0
---  IVPD115  DD41DUP2  2      999     2000   0     0
Command ==>
F1=Help  F3=Exit  F7=Backward  F8=Forward  F11=Right

```

図 36. 「Name」でソートされた、Database Partitions (データベース区画) パネル (DSPXPLC)

**重要:** 「Randomizer」セクションは、HALDB が PHDAM である場合にのみ存在します。

「Database Partitions」リスト・パネルでは、先頭に HALDB 名、その下にテーブル情報があります。列には以下のものがあります。

**Act** これは、open、copy、およびその他のコマンドを呼び出すことができる行コマンドの入力フィールドです。

**名前** 「Name」列には、区画を定義する際に指定した区画名が入ります。これは最初のソート・シーケンスです。

**Module**

「Module」列には、ランダム化モジュールのモジュール名が入ります。

**Anchor**

「Anchor」列には、ルート・アンカー・ポイントの数が入ります。

**High block**

High block (最大ブロック) 列には、最大ブロック番号が入ります。

**Bytes** 「Bytes」フィールドについては、121 ページの図 27 で説明しています。

**FBFF** 「FBFF」列には、空きブロック頻度係数が入ります。

**FSPF** 「FSPF」列には、フリー・スペース・パーセント係数が入ります。

行コマンドを使用するには、処理対象の区画の右側にある「act」列に入力します。「Database Partitions (データベース区画)」パネルでは、複数の行コマンド (ただし、1 区画につき 1 つだけ) を入力できます。コマンドは上から順に実行されます。

区画リスト行コマンド

行コマンドを使用して、以下のアクションを実行することができます。

### Delete a partition

行コマンド・フィールドに D と入力し、Enter キーを押します。「delete confirmation (削除確認)」パネルが表示されます。削除を確認するには、オプション・フィールドに 1 を入力して Enter キーを押し、削除を取り消すには、取り消しキーを押します。

一度にいくつかの区画を削除する際、すべての削除を受け入れたい場合は、オプション・フィールドに 2 を入力できます。オプション・フィールドは、区画リストが表示されるたびにリセットされます。

### Copy a partition

選択した区画の属性を使用して新規の区画を定義するには、行コマンド・フィールドに「C」を入力します。区画名は固有でなければなりません。

「Change Partition (区画変更)」パネルを使用して、区画情報を変更します。

### Open a partition

区画をオープンするには、行コマンド・フィールドに O を入力します。そうしてから、区画情報を変更することができます。区画名と区画 ID は変更できません。コミットするために Enter キーを押すか、または変更を廃棄するために取り消しキーを押します。「Change Partition (区画変更)」パネルを使用して、区画情報を変更します。

### Print partition information

選択した区画について区画情報を印刷するには、行コマンド・フィールドに P を入力します。その情報はすぐにはプリンターに転送されません。代わりに、ISPF リスト・データ・セットに追加されます。

### 区画リスト・アクション・バー

Database Partitions パネル内の区画のリストは、行コマンドによって、または以下の図に示すような File アクション・バーの選択項目を使用することによって、取り扱うことができます。

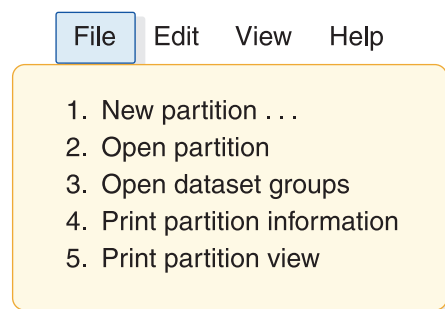


図 37. 「File」アクション・バーの選択項目

### New partition

最初に区画を作成したときに使用したのと同じパネルを使用して、新規の区画を作成することができます。

### Open partition

選択した区画をオープンして、それらを必要に応じて変更することができます。

### Open data set groups

130 ページの『データ・セット・グループ情報の定義』で説明したパネルを使用して、データ・セット・グループ・メンバーを取り扱うことができます。

### Print partition information

選択された区画についての情報が、ISPF リスト・データ・セットに書き込まれます。

### Print partition view

現在表示されているビューの情報が、ISPF リスト・データ・セットに書き込まれます。

「Database Partitions (データベース区画)」パネル上の区画のリストは、「Edit」アクション・バーを使用してさまざまな方法でソートできます。

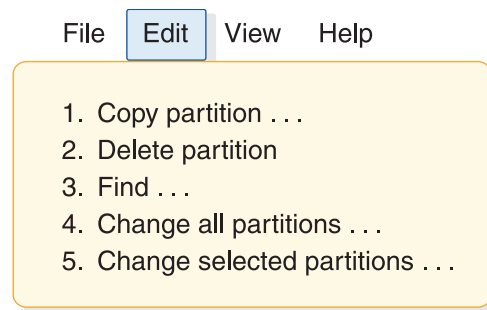


図 38. 「Edit」アクション・バーの選択項目

### Copy partition...

選択した区画の属性を使用して新規の区画を定義するには、行コマンド・フィールドにスラッシュ (/) を入力してから、「Edit - Copy partition (区画の編集/コピー・プルダウン)」パネルを使用します。区画名および区画 ID は固有でなければなりません。

この後、「Change Partition (区画変更)」パネルが表示され、新規の区画を連続的に作成できます。パネルに表示される値は、選択した区画の属性を使用して埋められたものです。

### Delete partition

区画を削除するには、行コマンド・フィールドにスラッシュ (/) を入力してから、「Edit - Delete partition (区画の編集/削除)」プルダウン・パネルを使用します。「delete confirmation (削除確認)」パネルが表示されます。削除を確認するために Enter キーを押すか、または削除を無視するために取り消しキーを押すことができます。

### Find...

選択した文字ストリングで、区画リストを検索することができます。単純な文字値のみを指定できます。カーソルは、検索値が入っている区画に位置付けられます。

検索ストリングは大文字小文字を区別しません。以下の図に示すように、「Database Partitions (データベース区画)」パネルに現在表示されているフィールドだけでなく、どのフィールドでも検索が行われます。

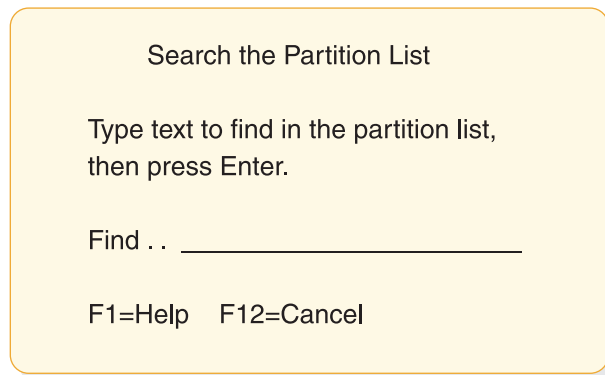


図 39. *Searching the Partition List* (区画リストの検索)

### Change all partitions...

HALDB の区画すべてについて個々のフィールドを変更するには、「Edit - Change all partitions (すべての区画の編集/変更)」プルダウン・パネルを使用します。区画名と区画 ID は変更できません。

### Change selected partitions...

選択した区画の個々のフィールドを変更するには、区画の変更のために行コマンド・フィールドにスラッシュ (/) を入力してから、「Edit - Change selected partitions (選択された区画の編集/変更)」プルダウン・パネルを使用します。区画名と区画 ID は変更できません。選択した区画のみが変更されます。

「Database Partitions (データベース区画)」パネル上の区画のリストは、「View」アクション・バー選択を使用してさまざまな方法でソートできます (以下の図に示します)。

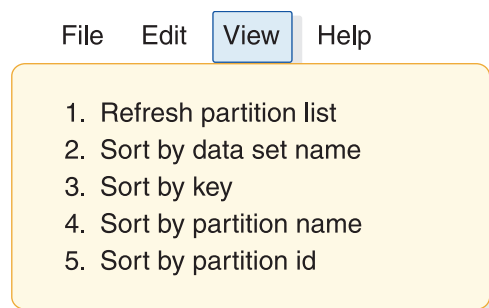


図 40. 「View」アクション・バーの選択項目

ヘルプ情報は、「Help」アクション・バーの選択肢を使用して使用することができます。

***Change all partitions***(すべての区画の変更)

「Change Partition (区画の変更)」パネル (DSPXPPB) に、「Change Partition (区画の変更)」パネルの使用例を示します。入力フィールドは空白です。変更したいフィールドのみを変更します。フィールド変更はすべての区画に適用されます。

**重要:** 同じ処理が「Change selected partitions」で使用されますが、違いは、リストからスラッシュ (/) で選択された区画にのみ変更が適用されることです。

文字フィールドを空白に変更したい場合は、1 つだけスラッシュ (/) を入力して、そのフィールド内でただ 1 つの文字になるようにします。

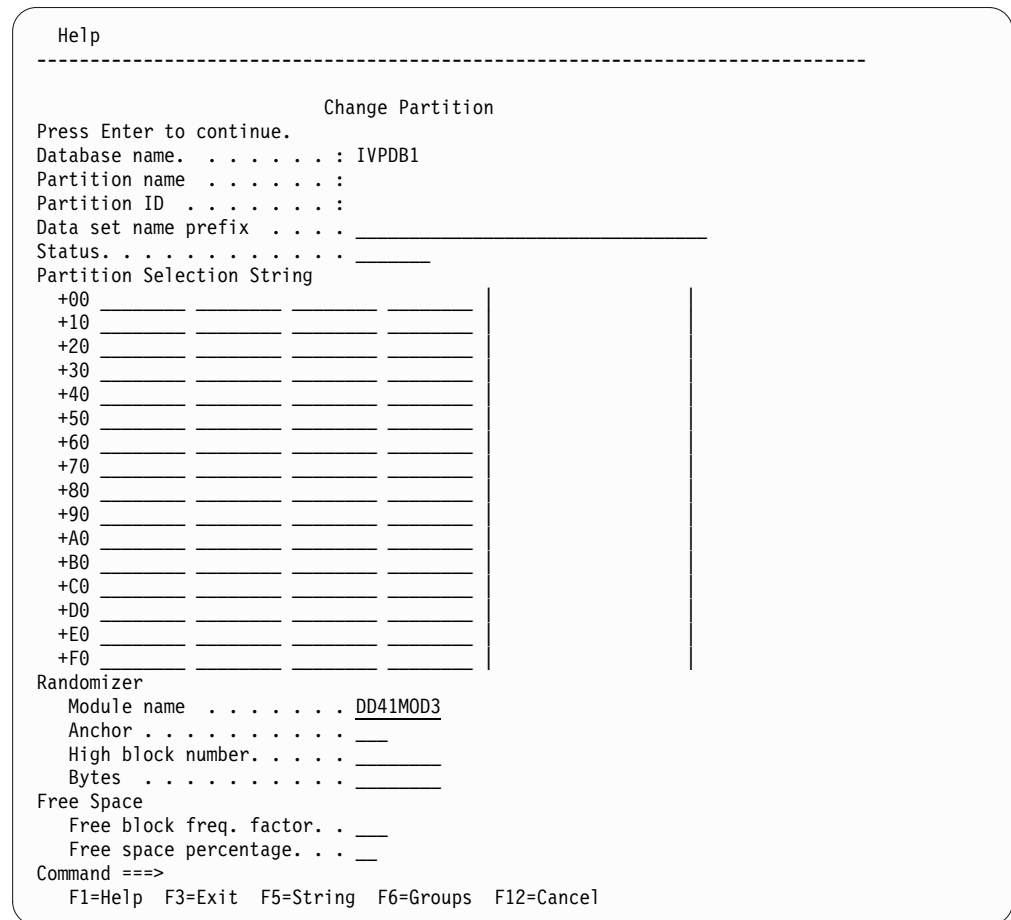


図 41. Change Partition (区画変更) パネル (DSPXPPB)

**重要:**

- 「Randomizer」セクションは、HALDB が PHDAM である場合にのみ存在します。
- 「data set groups」セクションは、DBDGEN でただ 1 つだけのデータ・セット・グループが指定されている場合にのみ存在します。複数のデータ・セット・グループがある場合、「F6=Groups」を使用してすべてのデータ・セット・グループを表示し、ダイアログを使用してください。

以下の図は、「Change Dataset Groups (データ・セット・グループ変更)」パネルを示したものです。

```

Help
-----
                                Change Dataset Groups          Row 1 to 10 of 10
Select an item by pressing a '/' on the desired line then press Enter.
Database name . . . . . : IVPDB1
Partition name . . . . . : IVPD101
Partition ID. . . . . : 1
Data set name prefix. . . . : IMS.DB01.FINANCE.YEAR2013.CURR
Act  Group  Block  Max Image  Recovery  Recovery  Default
   ___  ___  ___  ___  ___  ___  ___
   ___  All  ___  ___  ___  ___  ___
   ___  A    ___  ___  ___  ___  ___
   ___  B    ___  ___  ___  ___  ___
   ___  C    ___  ___  ___  ___  ___
   ___  D    ___  ___  ___  ___  ___
   ___  E    ___  ___  ___  ___  ___
   ___  F    ___  ___  ___  ___  ___
   ___  G    ___  ___  ___  ___  ___
   ___  H    ___  ___  ___  ___  ___
   ___  I    ___  ___  ___  ___  ___
   ___  J    ___  ___  ___  ___  ___
Command ==>
F1=Help F3=Exit F7=Backward F8=Forward F9=Save F11=Right F12=Cancel

```

図 42. 「Change data set groups (データ・セット・グループ変更)」パネル、その 1 (DSPXPGA)

## データベース情報のオープン

「Partitioned Databases (パーティション・データベース)」パネルから「Open database information」を選択すると、HALDB 用に最初に区画を定義したときに保管された、HALDB についての情報が表示されます。

```

Help
-----
                                Partitioned Database Information
Type the field values. Then press Enter to continue.
Database name . . . . . : IVPDB1
                                Master Database values
Part. selection routine . . . DFSIVD1
RSR global service group . . .
RSR tracking type . . . . .
Share level . . . . . 0
Database organization . . . : PHDAM
Recoverable? . . . . . はい
Number of data set groups . : 1
Online Reorganization Capable: Yes
To exit the application, press F3.
Command ==>
F1=Help F3=Exit F12=Cancel

```

図 43. Partitioned Database Information (パーティション・データベース情報) パネル (DSPXPOA)

フィールドを変更することができ、そして Enter キーを押して RECON の値を変更できます。取り消しキーまたは終了キーを押すと、このパネルで入力した変更はすべて廃棄されます。

## データベース情報の削除

「Partitioned Databases (パーティション・データベース)」パネルから「delete database information」を選択すると、「Delete a Database (データベースの削



除)」パネルが表示されます。削除を確認するには、スラッシュ (/) の文字を入力し、Enter キーを押します。削除を確認すると、HALDB およびその区画のすべてについての情報が RECON から削除されます。

削除を取り消す方法はありません。RECON から HALDB を削除する前に、エクスポートを実行することも可能です。

```
Help
-----
                                Delete Database Information
Type '/' to confirm the delete of the database information from RECON.
Then press Enter.
Database name . . . . . : IVPDB1
Confirm database delete . _
Command ==>
F1=Help   F3=Exit   F12=Cancel
```

図 44. Delete a Database (データベースの削除) パネル (DSPXPDA)

## データベース情報のエクスポート

「Partitioned Databases (パーティション・データベース)」パネルから「Export database information」を選択すると、情報は、ユーザー指定の区分データ・セットに保管されます。その情報は、ISPF テーブルとして保管されるため、以下の ISPTLIB データ・セット属性を持っていなければなりません。

- 固定ブロック・レコード形式
- レコード長 80
- PDS (または PDS/E) データ・セット編成

以下の図は、「Export a Database (データベースのエクスポート)」パネルを示したものです。

```
Help
-----
                                Export a Database
Type a data set name. Then press Enter.
Database Name . . . . . : IVPDB1
Output dataset name. . . . . 'TEST.RSR.PARTS'
Output member name . . . . . IVPDB1
To exit the application, press F3.
Command ==>
F1=Help   F12=Cancel
```

図 45. Export a Database (データベースのエクスポート) パネル (DSPXPEA)

フィールド

説明

データベース名

基本パネルで指定された HALDB 名。

**Output data set name**

「Output data set name」は、区画情報が入るデータ・セットの名前です。

バッチ・ジョブおよび区画定義ユーティリティ制御ステートメント DSPXRUN  
を使用して、HALDB データベース定義をエクスポートすることもできます。

## データベース情報のインポート

「Partitioned Databases (パーティション・データベース)」パネルから「Import database information」を選択するとき、情報が入っている PDS または PDS/E の名前を指定できます。

**重要:** エクスポートされた ISPF テーブルだけが、データベース定義のインポートに使用できます。HALDB のエクスポートの結果として出力されるのは、PDS のメンバーです。HALDB に関する情報は、ISPF テーブルの形式で保管されます。ISPF テーブルは、インポート処理の入力になります。

インポートは、HALDB 区画定義ユーティリティまたはバッチ・ジョブから実行されます。

Enter キーを押すと、テーブルが読み取られ、各区画が定義されます。

以下の図は、「Import a Database (データベースのインポート)」パネルを示したものです。

```
Help
-----
                                Import a Database
Type a dataset name. Then press Enter.
Database name . . . . . : IVPDB1
Input dataset name. . . . 'PROD.RSR.PARTS'
Input member name . . . . IVPDB1
Processing option . . . . — 1. Stop on first error
                           2. Try all partitions

Command ==>
F1=Help   F3=Exit  F12=Cancel
```

図 46. Import a Database (データベースのインポート) パネル (DSPXPJA)

データベース名

基本パネルで指定された HALDB 名。

### Input data set name

「Input data set name」は、区画情報を含むデータ・セットの名前です。データ・セットは区分データ・セットでなければなりません。

### Input member name

「Input member name」は、入力データ・セット内のメンバーの名前です。このメンバーは、HALDB 区画定義ユーティリティを使用してエクスポートされていなければなりません。

### Processing option

インポートされたテーブルの各区画は、RECON に定義することができます。エラーがある場合、残りの区画を試すか、または処理を停止するかを選択できます。

バッチ・ジョブおよび区画定義ユーティリティ制御ステートメント DSPXRUN  
を使用して、HALDB データベース定義をインポートすることもできます。

## HALDB データ・セット連結の表示

IMS DD 名に割り振られているデータ・セットの連結を見ることができます。ISPF 製品の一部である ISRDDN コマンドを使用して、データ・セットが表示されま  
す。STEPLIB 連結は変更されません。

```
Current Data Set Allocations                               Line 1 of 2
Volume  Disposition Act DDname  Data Set Name  List Actions: B E V F C I Q
SYS151  SHR,KEEP   >  _  IMS           IMSIVP91.DBDLIB
SYS335  SHR,KEEP   >  _  IMS           IMS91.SANJOSE.DBDLIB
----- End of Allocation List -----
Command ==>>                                         Scroll ==>> CSR
```

図 47. IMS の連結 (ISRDDN)

DFSHALDB パネルで総称データベース名を指定し、オプション 1 から 5 までを使用した場合、IMS DD 名の連結を表示するオプションは、4 つ以下の DBD データ・セットが使用されたときだけ機能します。オプション 7 を指定すると、IMS DD 名に連結されたデータ・セットは常に表示されます。

ISRDDN ユーティリティについて詳しく知るには、ISRDDN および ISPF に用意されているヘルプ (F1) 情報を使用してください。ISRDDN ユーティリティから出ると、HALDB 区画定義ユーティリティのパネルに戻ります。

## IMS 構成の選択

HALDB 区画定義ユーティリティのメニューには、IMS 構成を定義するオプションが入っています。IMS 構成は、一連の DBDLIB データ・セットおよび RECON データ・セットです。構成名を指定してください。

ログオン・プロシージャから IMS DDNAME を割り振っていて、しかも IMS.SDFSRESL ライブラリーが STEPLIB DDNAME に割り振られている場合は、構成オプションを使用しないでください。構成を定義し選択すると、これらのデータ・セットは、ログオン・プロシージャからの割り振りをオーバーライドします。

IMS がデータベースのアクセスに使用するのと同じ DBDLIB と RECON のデータ・セットを使用する必要があります。RECON1、RECON2、RECON3 に 1 つのデータ・セットと、IMS DD 名に最大 10 個の DBDLIB データ・セットを指定できます。構成の中で、RECON データ・セットを制御できます。

IMS DD 名には、DBDLIB メンバーが入っているデータ・セットが含まれます。RECON / DBDLIB Configurations (RECON / DBDLIB 構成) パネルで IMS DD 名を再割り振ります。

STEPLIB 割り振りには、実際の RECON データ・セットを示す RECON1、RECON2、および RECON3 メンバーが入っています。IMS はそれらのメンバーを使用して、どの RECON データ・セットを使用するかを決定します。STEPLIB を使用する代わりに、TSOLIB コマンドを使用して、TSO/E がコマンドやプログラムを検索するのに使用する検索順序を変更することができます。RECON / DBDLIB Configurations (RECON / DBDLIB 構成) パネルで、IMS DD 名を再割り振り

し、さらに RECON データ・セットを明示的に指定するために RECON1、RECON2、および RECON3 DD 名を割り振ります。STEPLIB 連結は変更されません。

構成だけを削除すると、その構成はリストから削除されますが、その構成内で指定されているデータ・セットは削除されません。

以下の図は、RECON/DBDLIB Configurations (RECON/DBDLIB 構成) パネルを示したものです。

```
RECON / DBDLIB Configurations Row 1 to 5 of 5
To create a new configuration, fill in the first line and press Enter.
Select a default by type '/' on the Act column then press Enter.
You can use 'O' to open or 'D' to delete a configuration.
Act  Current Name      Description
-----
*   SDFSRESL           IMS datasets
    TESTM              IMS for Matt
    TESTP              Test IMS for Peter
    TESTI              Test IMS
***** Bottom of data *****
Command ==>
F1=Help   F3=Exit   F7=Up     F8=Down   F12=Cancel
```

図 48. ユーザー構成 (DSPXPMB)

「Partitioned Databases (パーティション・データベース)」パネルで「option 7」を選択すると、構成のリストを維持することができます。リストは最初は空で、ブランク行に埋めていくことによって追加していくことができます。アクティブな構成は、「Current」列のアスタリスク (\*) によって識別されます。

行コマンド d を使用して、リストから行を削除することができます。構成のみがリストから削除されます。その構成内で指定されているデータ・セットは削除されません。

構成内で指定されているデータ・セットは、オープンのために行コマンド o を使用することによって、設定または変更されます。

以下の図は、「Configurations Details (構成詳細)」パネルを示したものです。

```

Configuration Details
Type in values in the fields and press Enter to continue.
Configuration name . . . . TEST1
Description . . . . . Test IMS RECON dataset names
RECON1 dataset . . . . 'TEST.PARTS.RECON1'
RECON2 dataset . . . . 'TEST.PARTS.RECON2'
RECON3 dataset . . . . _____
DBDLIB dataset names
DBDLIB dataset 1 . . . . 'TEST.PARTS.DBDLIB'
DBDLIB dataset 2 . . . . _____
DBDLIB dataset 3 . . . . _____
DBDLIB dataset 4 . . . . _____
DBDLIB dataset 5 . . . . _____
DBDLIB dataset 6 . . . . _____
DBDLIB dataset 7 . . . . _____
DBDLIB dataset 8 . . . . _____
DBDLIB dataset 9 . . . . _____
DBDLIB dataset 10 . . . . _____
Command ==>
F13=Help      F15=Exit      F19=Up        F20=Down      F22=Actions

```

図 49. Configuration Details (構成詳細) パネル (DSPXPMC)

RECON データ・セットは、ファイル名 RECON1、RECON2、および RECON3 に別々に割り振られます。

DBDLIB データ・セットは、ファイル名 IMS に連結されます。

**重要:** 「Partitioned Database (パーティション・データベース)」パネルで総称 HALDB 名を指定する場合、Option 6 (オプション 6) は、4 つ以下の DBD データ・セットを使用しているときのみ機能します。ただし、柔軟性を高めるために、最大 10 のデータ・セットを指定することができます。



---

## 第 11 章 MSDB 保守ユーティリティ (DBFDBMA0)

MSDB 保守ユーティリティ (DBFDBMA0) は、MSDB データベースの初期ロードのための z/OS 順次データ・セットの作成、および MSDB データベースの保守を行うために使用します。

DBFDBMA0 ユーティリティは、z/OS バッチ領域で実行されるオフライン・ユーティリティです。

この DBFDBMA0 ユーティリティは、MSDBINIT ファイルに入っている主記憶データベースの挿入、置き換え、削除、および変更を行います。これらのアクションはいずれも、このユーティリティを 1 回実行するだけで、異なる MSDB データベースに対して行うことができます。

MSDB 保守ユーティリティの INSERT 機能、REPLACE 機能、および DELETE 機能は、MSDB データベース全体に関連しています。MODIFY 機能は、MSDB データベース内のセグメントの挿入、置き換え、削除、および変更を行う場合に使用します。この機能を使用すると、指定されたセグメント内の 1 つ以上のフィールドを変更し、また、ある範囲のセグメントをキーで指定してそれらのセグメントの 1 つ以上のフィールドを変更することができます。

DBFDBMA0 ユーティリティの一般的アクションは次のとおりです。

1. 旧 MSDBINIT レコードにある MSDB 名が制御ステートメントにある名前と一致していない限り、レコードを旧 MSDBINIT ファイルから新 MSDBINIT ファイルにコピーします。
2. 旧 MSDBINIT ファイルにある MSDB 名が制御ステートメントにある名前と一致すると、制御ステートメントの中で要求されているアクションを実行します。
3. 入力ファイルのレコードがなくなるまで上記ステップ 1 と 2 を繰り返します。

以下の図は、MSDB を初期設定および変更するときに MSDB 保守ユーティリティによって使用される入力データ・セットと出力データ・セットを示しています。

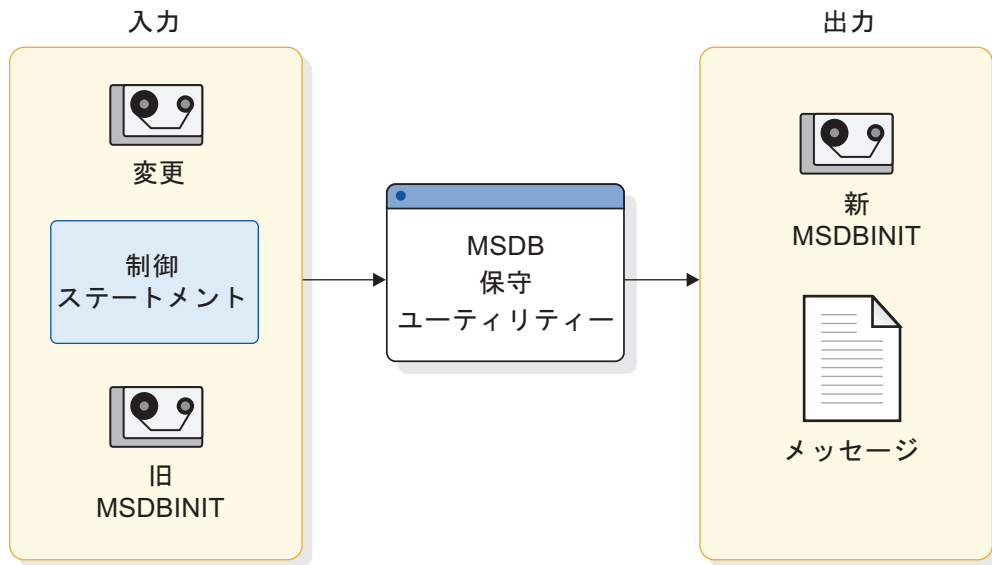


図 50. 入力データ・セットと出力データ・セットを指定する MSDB 保守ユーティリティー

サブセクション:

- 『制約事項』
- 149 ページの『前提条件』
- 149 ページの『要件』
- 149 ページの『推奨事項』
- 149 ページの『入出力』
- 151 ページの『戻りコード』
- 151 ページの『JCL 指定』

### 制約事項

ACTION 制御ステートメントに DELETE を指定して MSDB データベースを削除する場合、削除する MSDB データベースの変更レコードを含めることはできません。

DBFDBMA0 ユーティリティーおよび MSDB 変更データ・セットを使用して MSDB データベース内のレコードを変更する場合、以下のような制約があります。

- TO= キーワードを使って、ある範囲のレコードを挿入することはできません。
- MSDB 変更データ・セットでは、同一の MSDB データベースに対して 1 つのレコード(キー) を複数回指定することはできません。
- MSDBINIT ステートメントの形式を使用する MSDB 変更データ・セットを使用する場合、等号 (=) はキーワードを表す予約記号であるため、注釈の中で使用することはできません。

DBFDBMA0 ユーティリティーは、主記憶データベース (MSDB) のみで機能します。

データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。



## 前提条件

現在、DBFDBMA0 ユーティリティに対して文書化された前提条件はありません。

## 要件

DBFDBMA0 ユーティリティは少なくとも 512 KB の仮想ストレージを必要とします。

DBFDBMA0 ユーティリティは、独立したジョブ・ステップで実行しなければなりません。プログラムは、同じジョブ・ステップ内でユーティリティを切り替えることはできません。

動的 MSDB データベースは空にすることができますが、それ以外の MSDB データベースには、少なくとも 1 つのレコードがなければなりません。

## 推奨事項

現在、DBFDBMA0 ユーティリティに対して文書化された推奨事項はありません。

## 入出力

MSDB 保守ユーティリティは、実行されている特定の機能に応じて、さまざまな入力データ・セットを使用します。しかし、どの機能の場合も、そこから得られる 1 次出力は、MSDB が入っている順次データ・セットです。このデータ・セットは MSDBOLD ステートメントと MSDBNEW ステートメントで定義され、MSDB をロードする必要のあるシステム始動または再始動処理の入力とすることができます。

MSDB 保守ユーティリティの入出力は、以下の表に示すとおりです。

表 4. MSDB 保守ユーティリティが使用するデータ・セット

入力	出力
変更	新しい MSDBINIT
制御	メッセージ
旧 MSDBINIT	

MSDB 保守ユーティリティは、次の入力を使用します。

- MSDB 名、レコード・キー、データ、およびその他の情報が入っている、カード形式または MSDBINIT レコード形式の変更データ・セット。ただし、MSDB を削除するだけの場合を除きます。

新しい MSDB データベースを挿入する場合、変更データ・セットには、新しい MSDB データベースに入るようにフォーマット設定されるすべてのセグメントのデータを入れる必要があります。MSDB データベースを置き換える場合、旧 MSDBINIT データ・セットに存在する MSDB を指定し、すべてのセグメントのデータを指定する必要があります。MSDB データベースを変更する場合、変更デ

ータ・セットでは、変更するセグメントの MSDB 名およびキー・フィールド名を指定する必要があります。また、必要に応じて、そのセグメントのデータを指定します。

- RUN ステートメントと、実行される機能を指定する 1 つ以上の ACTION ステートメントが入っている制御データ・セット。

新しい MSDB データベースを挿入する場合、挿入する MSDB ごとに MODE=INSERT を指定する必要があります。MSDB データベースを置き換える場合、新しい MSDB で表示されるように置き換える MSDB ごとに MODE=REPLACE を指定する必要があります。MSDB データベースを削除する場合、削除する MSDB ごとに MODE=DELETE を指定する必要があります。MSDB データベースを変更する場合、変更する MSDB ごとに MODE=MODIFY を指定する必要があります。

- MSDB 保守ユーティリティ、ダンプ・リカバリー・ユーティリティ、または場合によってはユーザー作成ルーチンの、前回の実行によって得られたフォーマット設定済みの MSDB が入っている旧 MSDBINIT データ・セット。ただし、新しい MSDB を挿入するだけの場合を除きます。

新しい MSDB を挿入する場合、旧 MSDBINIT データ・セットがあれば、ここには、挿入する MSDB のレコードが入ってはいません。MSDB データベースを置き換える場合、旧 MSDBINIT データ・セットを指定する必要があります、そのデータ・セットには、置き換える MSDB ごとに少なくとも 1 つのレコードが入ってなければなりません。MSDB データベースを削除する場合、旧 MSDBINIT データ・セットを指定する必要があります。MSDB データベースを変更する場合、旧 MSDBINIT データ・セットを指定する必要があります、そのデータ・セットには、変更する MSDB ごとに少なくとも 1 つのレコードが入ってなければなりません。

MSDB 保守ユーティリティは、次の出力を作成します。

- MSDBINIT データ・セット (MSDBNEW)
- ユーティリティによるアクションまたはエラー・ステートメントを印刷した要約 (MSDBPRT)
- PROCLIB の更新に使用できる入力制御ステートメントが入ったデータ・セット (MSDBPUN)。このデータ・セットには、IMS.PROCLIB の DBFMSDBx メンバー内の各レコードのステートメントが入っています。このステートメントは、ライブラリー内のメンバーを更新する場合に IEBUPDTE ユーティリティで使用できます。

MSDB が高速機能アプリケーションにとって重要であり、MSDB がなくては IMS を実行できない場合は、MSDBABND= パラメーターを DBFMSDBx PROCLIB メンバーの最初のステートメントに指定することができます。MSDBABND パラメーターでは、システム初期設定時の MSDB ロード中にエラーが発生した場合に、IMS 制御領域が異常終了するように指定します。

MSDB のロード中に IMS 制御領域を異常終了させるための以下の条件を指定することができます。

- MSDB の最初のチェックポイントをとることができない。
- MSDBINIT データ・セットでのエラーが原因で MSDB をロードできない。

- MSDBINIT データ・セットでのエラーが原因で MSDB をロードできないか、少なくとも 1 つの定義済みの MSDB に対する MSDBINIT データ・セットにセグメントが存在しない。

## 戻りコード

DBFDBMA0 ユーティリティーにより、以下の戻りコードが返されます。

コード 意味

- 0 ユーティリティーが正常に実行された
- 4 エラー — エラー・メッセージが印刷された
- 8 印刷データ・セットがオープンできない

## JCL 指定

DBFDBMA0 ユーティリティーは、標準の z/OS ジョブとして実行されます。

MSDB 変更データ・セットには、ユーティリティーの処理オプションを指定するキーワードおよびパラメーターを含めることもできます。

DBFDBMA0 ユーティリティーを実行するには、次の JCL ステートメントが必要です。

- EXEC ステートメント
- 入力および出力を定義する DD ステートメント
- RUN 制御ステートメント
- ACTION 制御ステートメント

### EXEC ステートメント

このステートメントは必要な JCL を含んでいるプロシーチャーを指定することも、次のような形式にすることもできます。

```
// EXEC PGM=DBFDBMA0
```

### DD ステートメント

#### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。

#### MSDBACB DD

IMS ACB ライブラリーを定義します。このデータ・セットは、直接アクセス装置に置いておかなければなりません。

#### MSDBOLD DD

前回のユーティリティーの実行から得られた MSDB が入っている順次 MSDBINIT データ・セットを記述します。

#### MSDBNEW DD

現在のユーティリティーの実行の後の新しい MSDB が入る順次データ・セットを記述します。

**MSDBCTL DD**

制御ステートメントが入っているデータ・セットを記述します。

**MSDBCCHG DD**

ステートメント形式の変更レコードが入っているデータ・セットを記述します。  
このデータ・セットは、DBD 名とキーの順に並べなければなりません。

**MSDBLCHG DD**

MSDBINIT レコード形式の変更レコードが入っているデータ・セットを記述します。  
このデータ・セットは、DBD 名とキーの順に並べなければなりません。

**MSDBPRT DD**

印刷出力のためのメッセージ・データ・セットを記述します。

**MSDBPUN DD**

IMS.PROCLIB のメンバー DBFMSDBX を追加または置き換えるときの、  
IEBUPDTE 形式の変更ステートメントが入っている出力データ・セットを記述  
します。

## DBFDBMA0 ユーティリティーの制御ステートメント

MSDB 保守ユーティリティーには、RUN ステートメントおよび ACTION ステートメントと呼ばれる 2 種類の制御ステートメントが必要です。

これらは、1 行の 1 桁目から 71 桁目にコーディングします。ステートメントは次の行に継続できません。ステートメントには、コメントを入れることができます。

ACTION 制御ステートメントに MODIFY を指定する場合、DBFDBMA0 ユーティリティーへの入力として MSDB 変更データ・セットも含める必要があります。  
MSDB 変更データ・セットには、DBFDBMA0 ユーティリティーが MSDB データベースに対して行う変更を指定するキーワードおよびパラメーターをコーディングします。

動的 MSDB は空にすることができますが、他のタイプの MSDB には少なくとも 1 つのレコードがなければなりません。

### RUN ステートメント

以下に示すように、ユーティリティー全体の実行に RUN ステートメントを 1 つ使用します。

▶▶—PROC=*suffix*—◀◀

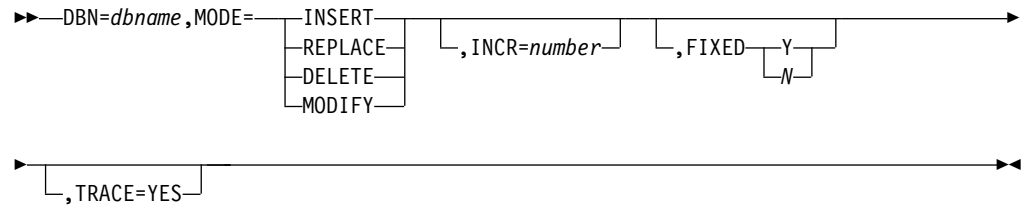
**PROC=**

MSDB 開始プロシージャー (DBFMSDBx) の 1 バイトの接尾部を指定します。

### ACTION ステートメント

このユーティリティーの実行対象となる MSDB データベースごとに、ACTION ステートメントが 1 つ必要です。入力 MSDBINIT データ・セットにあって、ACTION ステートメントで参照されない MSDB データベースはすべて、未変更のまま、新 MSDBINIT データ・セットに書き込まれます。ACTION ステートメン

トのキーワードは、コンマまたはブランクで区切ることができます。 ACTION ステートメントの形式は次のとおりです。



#### **DBN=dbname**

DBDGEN に指定されているものと同じ MSDB 名を指定します。

#### **MODE=**

以下のキーワードは、指定された MSDB に対するアクションを指定します。

##### **INSERT**

これを指定すると、MSDB 変更レコードから新しい MSDB が作成され、MSDBNEW に入れられます。INSERT を指定する場合、挿入される MSDB のすべての変更レコードにセグメント・データを入れる必要があります。また、旧 MSDBINIT ファイル (MSDBOLD) が存在する場合は、同じ MSDB 名を持つレコードが入ってはいけません。

##### **REPLACE**

これを指定すると、この MSDB のすべてのレコードが取り除かれ、変更レコード・データ・セットから新しいレコードが挿入されます。REPLACE を指定する場合は、その影響を受ける MSDB のすべての変更レコードにセグメント・データを入れる必要があります。

##### **DELETE**

これを指定すると、MSDB が削除されます。

制約事項: DELETE を指定できるのは、削除される MSDB に変更レコードが存在しない場合です。

##### **MODIFY**

これを指定すると、MSDB の中の個々のレコードが変更されます。MODIFY キーワードを指定する場合、MSDB 変更データ・セットも含める必要があります。

このユーティリティは、MSDB 変更レコード・データ・セットを旧 MSDBINIT データ・セットと比較します。データの入っている変更レコードが既存キーに提供されると、当該レコードは変更レコードで指定されたとおりに変更されます。データの入っていない変更レコードが既存キーに提供されると、当該レコードは削除されます。データの入っている変更レコードが非既存キーに提供されると、当該レコードは挿入されます。変更レコードが提供されないレコードは、いずれも変更されません。

制約事項: TO キーワードを使って、ある範囲のレコードを挿入することはできません。ある範囲のレコードを挿入することはサポートされていません。

**INCR=**

将来的な挿入に備えて動的 MSDB に予約しておく空のセグメントの数を 10 進数で指定します。動的 MSDB には、変更レコードは不要です。この数が、定義された論理端末の数を超過していないかを確認してください。

**FIXED=**

この MSDB をページ固定にするか (Y)、ページ固定にしないか (N) を指定します。Y はデフォルトです。

**TRACE=YES**

問題判別支援ツールとして使用する MSDB 保守ユーティリティ・モジュール 出入り口トレースをオンにします。

**MSDB 変更データ・セット**

MSDB 変更データ・セットは、MSDB に適用すべき変更を指定するキーワードとオペランドを含んでいます。このデータ・セットには、MSDB 内で挿入される、または古いレコードを置き換える、あるいは個々のレコードを変更する可変長レコード (セグメント) が入っています。このデータ・セットには、DELETE 操作のための変更レコードは入っていません。変更レコードは、MSDBINIT レコード形式データ・セットまたはカード形式データ・セット、あるいはその両方のデータ・セットに入れておくことができます。レコードは、MSDB 順に、各 MSDB 内ではキー順になっていなければなりません。

制約事項: MSDB 変更データ・セットでは、同一の MSDB データベースに対して 1 つのレコード(キー) を複数回指定することはできません。

**MSDBINIT** レコード形式

MODIFY 操作の場合は、次のことを行ってください。

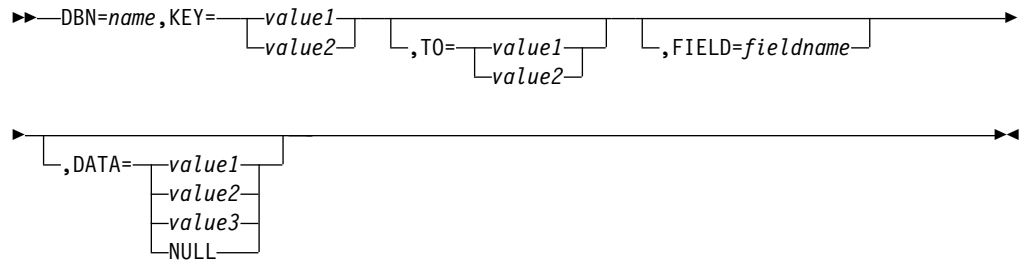
- セグメントを置き換える場合は、既存キーを KEY フィールドに入れ、必要なセグメント内容を MSDB セグメント・フィールドに入れたレコードを与えます。
- セグメントを挿入する場合は、非既存キーを KEY フィールドに入れ、必要なセグメント内容を MSDB セグメント・フィールドに入れたレコードを与えます。
- セグメントを削除する場合は、既存キーを KEY フィールドに入れ、MSDB セグメント・フィールドにはデータを入れていないレコードを与えます。

**MSDBINIT** ステートメントの形式

各ステートメントでは、80 バイト・レコードの 1 桁目から 71 桁目までにデータを入れることができます。各コメント・ステートメントの 1 桁目にアスタリスクを入れると、コメントを使用できます。コメントは、キーワードと同じステートメントに入れることができます。

制約事項: 等号 (=) はキーワードを表す予約記号であるため、コメントの中で使用することはできません。

各ステートメントには、次のキーワードをコンマまたはブランクで区切って入れることができます。



#### DBN=

MSDB 名を指定します。この名前は、DBDGEN の中の名前と同じでなければなりません。

#### KEY=

アクションの対象となるレコードのキーを指定します。DATA キーワードの項の value1 と value2 の説明を参照してください。

MSDB がセグメント・キーに LTERM 名を使用する (DBDGEN で REL=TERM、FIXED、または DYNAMIC と指定されている) 場合は、このフィールドに指定できる値は LTERM 名だけです。このオペランドの長さは、8 バイトでなければなりません。LTERM 名が 8 バイト未満であるときは、名前が 8 バイトになるように末尾ブランクを埋め込んでください。

#### TO=

これはオプション・キーワードですが、これを使用すると、1 つのステートメントの中で、ある範囲のセグメント・キーを操作することができます。TO= で参照するキーは、KEY= で指定したキーよりも値の大きいものでなければなりません。TO= が使えるのは、ACTION ステートメントが MODIFY を指定している場合だけです。DATA キーワードの項の value1 と value2 の説明を参照してください。

#### FIELD=

これはオプション・キーワードですが、どのフィールドを変更するかを指定するものです。FIELD= が使用できるのは、INSERT、REPLACE、および MODIFY のときだけです。各 FIELD= キーワードの後には、関連する DATA= キーワードが必要です。

#### DATA=

フィールドまたはセグメントの内容を指定します。FIELD= キーワードが DATA= の前に置かれている場合、これはフィールドの内容を指定します。FIELD= が DATA= の前に置かれていない場合は、これはセグメントの内容全体を指定します。

#### value1

文字フィールドを引用符で囲みます。このフィールドを継続するには、71 桁目でフィールドを中断し、ブランク以外の文字を 72 桁目に入れ、次のステートメントの 16 桁目からフィールドを続けます。

#### value2

文字フィールドの 16 進表記を引用符で囲み、その前に X を付けます。このフィールドを継続するには、71 桁目でフィールドを中断し、ブランク以外の文字を 72 桁目に入れ、次のステートメントの 16 桁目からフィールドを続けます。

### value3

算術フィールドを符号付きまたは符号なしの 10 進数で指定し、引用符ではなく括弧で囲みます。ユーティリティーは、この数をフィールド・タイプに合った形式に変換します。先行ゼロは省くことができます。

### NULL

これを指定すると、セグメント全体 (シーケンス・フィールドを除く) または指定されたフィールドがヌル値に設定されます。

文字フィールドと 16 進数フィールドはブランク (X'40') に設定されます。算術フィールドはゼロ値に設定されます。

フィールドに複数の定義が与えられている場合は、算術の定義が優先します。

---

## DBFDBMA0 ユーティリティーの例

これらの例は、MSDB 保守ユーティリティーを実行するのに必要な JCL を示しています。制御データ・セットおよび変更レコード・データ・セットは、入力として使用されます。

サブセクション:

- 『2 つの新しい MSDB の作成』
- 157 ページの『MSDB の削除、変更、および置き換え』
- 158 ページの『MSDB ファイルのマージ』

### 2 つの新しい MSDB の作成

例 1 では、2 つの新しい MSDB を作成します。

```
//STEP1 EXEC PGM=DBFDBMA0
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//MSDBPRT DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=605)
//MSDBCTL DD *
PROC=M
DBN=MSDBLM01 MODE=INSERT FIXED=YES
DBN=MSDBLM06 MODE=INSERT INCR=4 FIXED=NO
/*
//MSDBACB DD DSNAME=IMS.ACBLIB,DISP=SHR
//MSDBNEW DD DSNAME=XXXX,DISP=(NEW,CATLG),
// UNIT=SYSDA,VOL=SER=IMSDCL,
// DCB=(BLKSIZE=13030,RECFM=VBT,LRECL=13026),
// SPACE=(CYL,1)
//MSDBPUN DD SYSOUT=B
//MSDBCCHG DD *
DBN=MSDBLM01
/*
CREATE TWO NEW MSDBS
BUILD NONTERMINAL-RELATED MSDB LM01
KEY='0001'
FIELD=FIELD01 DATA='RP' CHARACTER KEY
FIELD=FIELDH01 DATA=X'9999' CHARACTER DATA
FIELD=FIELDF01 DATA=X'9FFFFFFF' FULLWORD DATA (HEX)
FIELD=FIELDP01 DATA=(1) DECIMAL DATA (ARITH)
FIELD=FIELDP02
FIELD=FIELDP02 DATA=X'9C' HEX
FIELD=FIELDX03 DATA=NULL NULL FIELD
KEY='0002'
FIELD=FIELD01 DATA='RP'
```



```

FIELD=FIELDH01 DATA=(-1)          HALFWORD TO ARITH -1
FIELD=FIELDF01 DATA=(-1)          FULLWORD TO ARITH -1
FIELD=FIELDP01 DATA=(1)           DECIMAL TO ARITH +1
/**                                LET FIELDP02 DEFAULT TO ZERO
/**                                LET FIELDP03 DEFAULT TO ZERO
/**                                LET FIELDX03 DEFAULT TO BLANKS
/**                                BUILD DYNAMIC TERMINAL RELATED MSDB LM06 WITH ONE UNUSED SEGMENT
DBN=MSDBLM06
KEY='LTERM01 '
FIELD=FIELDSEQ DATA='0001'
FIELD=FIELDH01 DATA=(0)
FIELD=FIELDX03 DATA=NULL
KEY='LTERM02 '
FIELD=FIELDSEQ DATA='0002'
FIELD=FIELDH01 DATA=(0)
FIELD=FIELDX03 DATA=NULL
KEY='LTERM03 '
FIELD=FIELDSEQ DATA='0003'
FIELD=FIELDH01 DATA=(0)
FIELD=FIELDX03 DATA=NULL
/*

```

## MSDB の削除、変更、および置き換え

例 2 では、1 つの MSDB を削除し、コメントに記述されているとおり 2 つ目を変更し、3 つ目を置き換えます。MSDBLM04 で記述のない MSDB およびセグメントはすべて、MSDBOLD から MSDBNEW にコピーされます。

```

//STEP2 EXEC PGM=DBFDBMA0
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//MSDBPRT DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=605)
//MSDBCTL DD *
DBN=MSDBLM03 MODE=DEL
DBN=MSDBLM04 MODE=MOD
DBN=MSDBLM05 MODE=REP
PROC=Y
//MSDBACB DD DSNAME=IMS.ACBLIB,DISP=SHR
//MSDBNEW DD DSNAME=XXXX,DISP=(NEW,CATLG),
//          UNIT=SYSDA,VOL=SER=IMSDCL,
//          DCB=(BLKSIZE=13030,RECFM=VBT,LRECL=13026),
//          SPACE=(CYL,1)
//MSDBOLD DD DSNAME=IMS.MSDBLM01(0),DISP=SHR
//MSDBPUN DD SYSOUT=B
//MSDBCCHG DD *
DBN=MSDBLM04
/**                                MODIFY THE MSDB BY DELETING ONE SEGMENT, CHANGING
/**                                THE TRANSACTION LIMIT IN THE DESIGNATED SEGMENTS,
/**                                AND COPYING THE REMAINDER OF THE FIELDS FROM THE
/**                                OLD FILE.
KEY='LTERM01 '                      DELETE THIS SEGMENT
KEY='LTERM02 '                      MODIFY 1 FIELD IN THIS SEGMENT.
FIELD=FIELDP03 DATA=(1500)        SET MAX TRANSACTION LIMIT
KEY='LTERM03 ',TO='LTERM09 '        MODIFY 1 FIELD IN EACH SEGMENT
/**                                WITHIN THE RANGE OF KEYS
FIELD=FIELDP03 DATA=(750)        SET MAX TRANSACTION LIMIT
DBN=MSDBLM05
/**                                INSERT 2 SEGMENTS, 1 FIELD IN EACH SEGMENT INITIALIZED
/**                                ALL OTHER FIELDS ARE SET TO NULL VALUES COPIED
KEY='LTERM03 '
FIELD=FIELDP01 DATA=(-2)          SET TIME ZONE FACTOR
KEY='LTERM05 '
FIELD=FIELDP01 DATA=(+1)          SET TIME ZONE FACTOR
/*


```

## MSDB ファイルのマージ

例 3 では、2 つの MSDB ファイルをマージします。MSDBLCHG に置かれている MSDB が、MSDBOLD に置かれている MSDB とマージされます。

```
//*          MERGE TWO MSDB FILES TO COMBINE ALL MSDBS INTO
//*          ONE INITIAL LOAD FILE.
//*
//STEP3    EXEC PGM=DBFDBMA0
//STEPLIB  DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//MSDBPRT  DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=605)
//MSDBLCHG DD DSN=MSDBUT32,DISP=SHR
//MSDBOLD  DD DSN=MSDBUT31,DISP=SHR
//MSDBCTL  DD *
           DBN=MSDBLM04 MODE=INSERT          FROM CHANGE FILE
           DBN=MSDBLM05 MODE=INSERT          FROM CHANGE FILE
//MSDBACB  DD DSN=IMS.ACBLIB,DISP=SHR
//MSDBNEW  DD DSN=MSDBUT33,DISP=(NEW,CATLG),
//          UNIT=SYSDA,VOL=SER=IMSDCL,
//          DCB=(BLKSIZE=13030,RECFM=VBT,LRECL=13026),
//          SPACE=(CYL,1)
//MSDBPUN  DD SYSOUT=B
/*
```

関連資料:

 IMS PROCLIB データ・セットの DBFMSDBx メンバー (システム定義)

---

## MSDB 保守ユーティリティーの機能

以下のサブセクションでは、DBFDBMA0 ユーティリティーを使用して MSDB データベースの挿入、置き換え、削除、および変更を行う手順について説明します。

### MSDB データベースの挿入

MSDB データベースを挿入する場合は、以下の点に注意します。

- 変更データ・セットをカード形式または MSDBINIT 形式、あるいはその両方の形式で指定する必要があります。ここでは、新しい MSDB に入るようにフォーマット設定されるすべてのセグメントのデータを入れなければなりません。
- 制御データ・セットには、挿入する MSDB ごとに MODE=INSERT を指定した ACTION ステートメントが入っていなければなりません。
- 旧 MSDBINIT データ・セットがあれば、そこには、挿入する MSDB のレコードが入ってはいけません。

MSDB は、このユーティリティーによって変更データ・セットから読み取られ、フォーマット設定され、新 MSDBINIT データ・セットに書き込まれます。

### MSDB データベースの置き換え

MSDB データベースを置き換える場合は、以下の点に注意します。

1. 変更データ・セットは、以下の要件を満たす必要があります。
  - 指定されなければならない。
  - カード形式または MSDBINIT 形式、あるいはその両方の形式でなければならない。

- 旧 MSDBINIT データ・セットに存在する MSDB を指定しなければならない。
  - すべてのセグメントのデータを指定しなければならない。
2. 制御データ・セットには、置き換えて新しい MSDB に入れる MSDB ごとに MODE=REPLACE を指定した ACTION ステートメントが入っていない限りなりません。
  3. 旧 MSDBINIT データ・セットが必要であり、ここには、置き換える MSDB ごとに少なくとも 1 つのレコードが入っていない限りなりません。

変更データ・セットに指定されている MSDB を除き、すべての MSDB が旧 MSDBINIT からコピーされ、新 MSDBINIT に書き込まれます。指定されている MSDB のいずれかが見付かると、旧 MSDBINIT からコピーされずに、変更データ・セットから読み取られフォーマット設定されます。

### MSDB データベースの削除

MSDB データベースを削除する場合は、以下の点に注意します。

- 変更データ・セットは必要ありません。
- 制御データ・セットには、削除する MSDB ごとに MODE=DELETE を指定した ACTION ステートメントが入っていない限りなりません。
- 旧 MSDBINIT データ・セットが必要です。

変更データ・セットに指定されている MSDB を除き、すべての MSDB が旧 MSDBINIT からコピーされ、新 MSDBINIT に書き込まれます。指定されている MSDB は読みとばされるだけで、無視されます。

### MSDB データベースの変更

MSDB データベースを変更する場合は、以下の点に注意します。

- 変更データ・セットはカード形式または MSDBINIT 形式、あるいはその両方の形式にすることができます。ここには、変更するセグメントの MSDB 名およびキー・フィールド名を入れなければなりません。オプションとして、そのセグメントのデータを入れる場合もあります。
- 制御ステートメント・データ・セットには、変更する MSDB ごとに MODE=MODIFY を指定した ACTION ステートメントが入っていない限りなりません。
- 旧 MSDBINIT データ・セットが必要であり、ここには、変更する MSDB ごとに少なくとも 1 つのレコードが入っていない限りなりません。

このユーティリティーは、変更レコード・データ・セットを旧 MSDBINIT と比較します。データの入っている変更レコードが既存キーに提供されると、当該レコードは変更レコードで指定されたとおりに変更されます。データの入っていない変更レコードが既存キーに提供されると、当該レコードは削除されます。データの入っている変更レコードが非既存キーに提供されると、当該レコードは挿入されます。変更レコードが提供されないレコードは、いずれも変更されません。



---

## 第 2 部 バックアップ・ユーティリティー

バックアップ・ユーティリティーは、データベース、DEDB 区域、または HALDB 区画のバックアップ・コピーを作成するために使用します。

このようなユーティリティーで作成されるバックアップ・コピーは、イメージ・コピーと呼ばれます。各トピックでは、1 つのユーティリティーを紹介し、その作動方法を説明し、その使用のための要件および制約事項を定義し、例を提供します。



---

## 第 12 章 データベース・イメージ・コピー・ユーティリティー (DFSUDMP0)

コピー中のデータ・セットのメモリー・ダンプからデータベース・データ・セットのイメージ・コピーを作成するには、データベース・イメージ・コピー・ユーティリティー (DFSUDMP0) を使用します。データベース・リカバリー・ユーティリティー (DFSURDB0) への入力として、DFSUDMP0 ユーティリティーから出力データ・セットを使用します。

DFSUDMP0 ユーティリティーを使用して、バッチまたは並行のイメージ・コピーを実行することができます。DBRC は並行イメージ・コピーに必要です。

DFSUDMP0 ユーティリティーの 1 回の実行で、複数のデータ・セットまたは区域を異種のもので混合した DASD 装置上にコピーすることができます。

DFSUDMP0 ユーティリティーでは、出力イメージ・コピーの 1 つまたは 2 つのコピーを作成できます。コピーを 2 つ指定すると、コピーの実行中に入出力エラーが起こっても、ユーティリティーは他方のコピーで最後まで続行するという利点があります。パフォーマンスがいくぶん低下しますが、実行全体をやり直す必要はありません。

共用 2 次索引のイメージ・コピーを作成する場合、データ・セット全体をコピーするために、DFSUDMP0 ユーティリティーの最初の DBD のみ指定する必要があります。

このユーティリティーの機能は、HALDB データベースのイメージ・コピーを作る場合を除き、ユーティリティー制御機能の制御下で実行できます。

以下の図は、データベース・イメージ・コピー・ユーティリティーのフロー・ダイアグラムです。

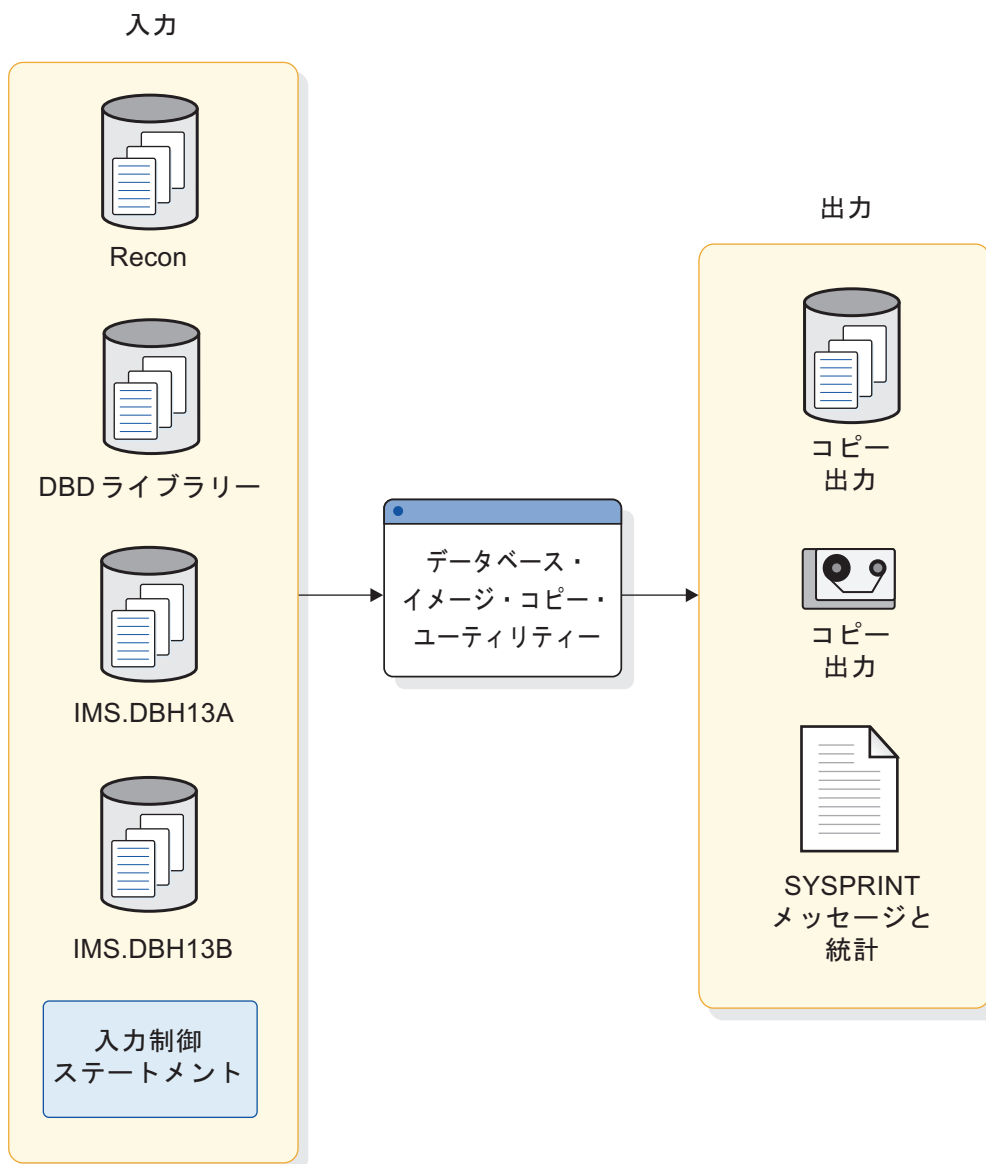


図 51. データベース・イメージ・コピー・ユーティリティ

サブセクション:

- 『制約事項』
- 165 ページの『前提条件』
- 166 ページの『要件』
- 166 ページの『推奨事項』
- 167 ページの『入出力』
- 168 ページの『戻りコード』
- 168 ページの『JCL 指定』

### 制約事項

データベース・イメージ・コピー・ユーティリティを使用するときは、次に挙げる制約があります。



- HSAM、GSAM、および MSDB データベースはこのユーティリティを使用し  
てコピーすることはできません。
- クリーン・イメージ・コピーを行っている場合、データベース・イメージ・コピ  
ー・ユーティリティの実行中は、コピー中のデータベースを更新できませ  
ん。/DBR または /DBD コマンドを、コピーするデータベースに対して発行し  
てください。メッセージ DFS0488I が出されるまで待機した後で、データベ  
ース・イメージ・コピー・ユーティリティを開始してください。EXEC ステートメ  
ントに CIC を指定した場合は、この手順は当てはまりません。(CIC パラメー  
ターは、並行イメージ・コピー処理を選択します。)代わりに、OPTION(HOLD)  
と一緒にデータベース静止コマンドを発行して、データベースを QUIESCE  
HELD 状態にすることができます。
- データベース・イメージ・コピー・ユーティリティが VSAM KSDS のコピー  
を実行中にデータベースの更新を行う場合、データベースのリカバリーのために  
イメージ・コピーを使用できません。
- 並行イメージ・コピー (ファジー・イメージ・コピー) には以下の制約がありま  
す。
  - VSAM KSDS データベース・データ・セットではサポートされません。
  - 共用レベル 0 で登録されているデータベースまたは区域ではサポートされま  
せん。
  - アクセス・レベルが EX (EXCLUSIVE) のデータベースには作成できませ  
ん。
  - リカバリー不能データベースには作成できません。リカバリー不能データベ  
ースで CIC を試みると、失敗し、メッセージ DSP0090I が出されます。
  - 統合 HALDB オンライン再編成機能が区画を再編成している間は区分デー  
タ・セットには作成できません。
- DFSUDMP0 ユーティリティは再始動できません。
- HALDB オンライン再編成を使用する IMS システムでイメージ・コピー・ユー  
ティリティを実行する場合は、さらに制約が適用されます。
- 大容量ブロック・インターフェース (LBI) は、LBI 対応テープ装置上のテープ・  
データ・セットについてのみサポートされます。データベース・イメージ・コピ  
ーは、DASD 上での出力データ・セットに対し、システムが決定したブロック・  
サイズをサポートします。データベースを変更するユーティリティは、デー  
タベースが静止している間に実行することはできません。
- RECON データ・セットに登録されていない IMS カタログ・データベースで使  
用した場合、このユーティリティでは、バッチ・イメージ・コピーのみ行うこ  
とができます。また、RECON データ・セットに登録されていないカタログ  
HALDB を識別するには、Datain DD ステートメントを指定する必要があります。

## 前提条件

DFSUDMP0 ユーティリティを使用している環境によっては、DFSUDMP0 ユー  
ティリティを実行する前に特定の前提条件を満足する必要があります。

DFSUDMP0 ユーティリティの前提条件を以下に示します。

- データベース・データ・セットまたは区域のクリーン・イメージ・コピーを作成している場合は、DFSUDMP0 ユーティリティを実行する前に、そのデータ・セットを含むデータベース、区域、または区画へのアクセスを停止する必要があります。データベース、区域、または区画に対して /DBR、/DBD、UPDATE...STOP(ACCESS) のいずれかのコマンドを発行することで、データベース、区域、または区画へのアクセスを停止できます。メッセージ DFS0488I が出されるまで待機した後で、データベース・イメージ・コピー・ユーティリティを開始してください。並行イメージ・コピーを作成している場合は、データベースまたは区画を停止する必要はありません。
- データベースで書き込みエラーが発生した場合は、データベース・イメージ・コピー・ユーティリティを実行する前にそのデータベースをリカバリーする必要があります。DBRC を使用してデータベースを登録し、DFSUDMP0 ユーティリティが DBRC を使用している場合、DBRC は DFSUDMP0 ユーティリティの実行を妨げます。

## 要件

データベース・イメージ・コピー・ユーティリティを使用するときは、次に挙げる要件があります。

- 論理レコード長とブロック化因数は実行時に計算されるので、イメージ・コピー・ユーティリティが作成するすべての出力コピーに標準テープ・ラベルを使用しなければなりません。
- HALDB 区分データ・セットのイメージ・コピーを作成している場合、DBRC がアクティブでなければなりません。そうでなければ、コピー要求は拒否されます。未登録の IMS カタログ HALDB データベースの場合、DBRC がアクティブである必要はありません。
- 複数 DEDB エリア・データ・セットのイメージ・コピーを作成している場合、制御文に指定する区域名は DBRC RECON データ・セットに登録されていなければなりません。
- データベースの停止中に DBDGEN を実行することによって 1 つ以上の区域を DEDB データベースに挿入するか DEDB データベースから削除する場合、データベースを再起動する前に、挿入または削除した区域に続くすべての区域のイメージ・コピーを行う必要があります。
- 32760 バイトを超えるブロック・サイズをコピーする場合、次の装置に大容量ブロック・インターフェースが必要です。
  - 3480 磁気テープ
  - 3490 および 3490E 磁気テープ・サブシステム
  - 3590 装置

ヒント: システム指定のブロック・サイズを使用してください。

- バッチ・イメージ・コピーを作成するために使用されるデータ・セットと一緒に、DISP=OLD を使用した場合に、データベースが QUIESCE HELD 状態のときは、DISP=SHR を指定する必要があります。

## 推奨事項

DFSUDMP0 ユーティリティには、使用と実行についていくつかの推奨事項があります。

- 該当するジョブ制御ステートメントで追加のバッファーを用意しておく、パフォーマンスが向上します。バッファーの最適数は、個々の要件によって異なります。VSAM データ・セットをコピーしている場合、AMP パラメーターを使用して追加の VSAM バッファー指定すると性能が向上します。OSAM データ・セットをコピーしている場合、DCB=BUFNO=*n* を指定することができます。ここで、*n* はトラック当たりのブロック数 / CI です。
- 特に、大きなブロック・サイズをサポートするデバイス上で 32760 バイトより大きいブロック・サイズをコピーする場合は、システム指定のブロック・サイズを使用してください。
- イメージ・コピー・ユーティリティーを一度実行して複数のデータ・セットまたは区域をコピーする場合は、データベースのすべてのデータ・セットを同時にコピーします。DEDB に対して、複数エリア・データ・セット (MADS) を DBRC RECON データ・セットに登録する場合は、イメージ・コピー・ユーティリティーへの入力として MADS を指定できます。
- ロギングしないでデータベースを更新するバッチ・ジョブを実行した直後にイメージ・コピーを呼び出してください。こうすると、リカバリーが必要な場合に、データベースの保全性が維持されます。ログ・テープを使用すると、バッチ・ジョブの開始までリカバリーできるため、そこでバッチ・ジョブを再処理することができます。ただし、その結果のデータベースは、前のバッチ実行後のデータベースと論理的には同一であっても、ビット単位では同一でない場合があります。バッチ・ジョブは反復可能でない場合があるため、そのことを前提としてください。データベースがビット単位で同一でないと、前のバッチ・ジョブの実行後に作られたログ・テープは、再処理後は有効でなくなります。したがって、リカバリー処理は、イメージ・コピーを使用して開始し、ログ・テープを適用して、ログがとられなかったバッチ実行を再処理した後に、さらにログ・テープを適用するという順序で行うことはできません。
- リモート・サイト・リカバリー (RSR) 環境では、EXEC ステートメントのパラメーター・リストにグローバル・サービス・グループ (GSG) 名を追加します。

GSG パラメーターは GSGNAME=*gsgname* です。ここで、*gsgname* は 1 文字から 8 文字までの名前です。

GSGNAME パラメーターを追加しない場合、IMS システム定義の使用中に定義された GSG 名が使用されます。

## 入出力

DFSUDMP0 ユーティリティーへの 1 次入力は、コピーするデータベース・データ・セットです。DFSUDMP0 ユーティリティーの 1 次出力は、データベース・リカバリー・ユーティリティー (DFSURDB0) への入力として使用できるデータベース・データ・セットのイメージ・コピーです。

出力コピー上の最初のレコードは、ダンプ・ヘッダー・レコードです。このヘッダー・レコードには、データ・セット ID、作成日、作成時刻といった情報が含まれています。作成日と作成時刻は、入力を検証するためにデータベース・リカバリー・ユーティリティーで必要になるものです。

以下の表は、データベース・イメージ・コピー・ユーティリティーに対する入力と出力を示します。

表 5. データベース・イメージ・コピー・ユーティリティの入出力

入力	出力
RECON	SYSPRINT メッセージと統計
DBD ライブラリー	イメージ・コピー・データ・セット
コピーする 1 つ以上のデータベース・データ・セット	
ユーティリティ制御ステートメント	

## 戻りコード

データベース・イメージ・コピー・ユーティリティから次のような戻りコードが返されます。

コード 意味

- 0 すべての操作が正常に完了した
- 4 警告メッセージが出された
- 8 1 つ以上の操作が正常に完了しなかった
- 16 すべての操作を完了する前にジョブを終了させる重大エラーが起こった

これらの戻りコードは、後続のジョブ・ステップの EXEC ステートメントで COND パラメーターを指定すると、テストすることができます。

## JCL 指定

DFSUDMP0 ユーティリティは、標準の z/OS ジョブとして実行されます。DFSUDMP0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティ制御ステートメントが必要です。

次の JCL ステートメントが必要です。

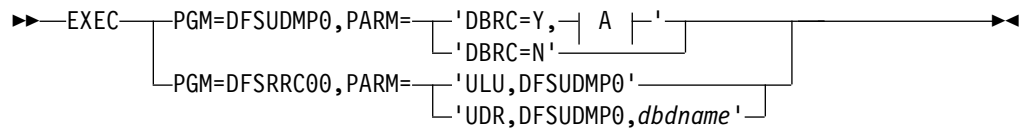
- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

1 つ以上のユーティリティ制御ステートメントを JCL ステートメントで実行依頼する必要もあります。

### EXEC ステートメント

DFSUDMP0 ユーティリティの EXEC ステートメントは、カタログ式プロシージャーを呼び出したり、特定の形式で JCL に指定したりできます。

JCL に EXEC ステートメントを指定するには以下の形式を使用します。



**A:**



以下のリストでは、DFSUDMP0 の EXEC ステートメントに組み込むことができるパラメーターについて説明しています。

**CIC**

OSAM および VSAM ESDS データベース・データ・セットの並行イメージ・コピーを作成します。

CIC を指定するときは、全機能データベースで DBRC=Y と指定しなければなりません。高速機能データベースでは、CIC は DBRC=N を使って指定できます。そのようにして得られたイメージ・コピー・データ・セットを使用して、DEDB ポインター・チェッカー (FABADA1) などのプログラムを実行することができます。これは、データベースのリカバリーに使用してはなりません。

CIC を実行するためには、DBRC 共用レベルを指定する必要があります。共用レベル 1、2、または 3 を指定できます。DBRC 共用レベルは INIT.DB コマンドで指定します。

アクセス・レベルは、システム定義でデータベースを初期設定するときに、DATABASE マクロを使用するか、CREATE DB または UPDATE DB コマンドを使用して設定します。

制約事項:

- CIC は、VSAM KSDS データベース・データ・セットではサポートされません。
- CIC は、共用レベル 0 に登録されているデータベースまたは区域には指定できません。
- CIC は、アクセス・レベルが EX (EXCLUSIVE) のデータベースに対して実行させることはできません。
- CIC は、リカバリー不能データベースに対して実行させることはできません。リカバリー不能データベースで CIC を試みると、失敗し、メッセージ DSP0090I が出されます。
- CIC は、HALDB 区画のオンライン再編成中は実行できません。

**DFSUDMP0**

DFSUDMP0 ユーティリティの通常の実行モードである IMS 領域コントローラーから独立して、DFSUDMP0 ユーティリティを実行することを指定します。DFSRRC00 とは相互に排他的です。

例えば、DFSUDMP0 ユーティリティを独立して実行することを EXEC DFSUDMP0,DBRC=Y コマンドで指定します。

## DBRC

DFSUDMP0 ユーティリティが DBRC を使用するかどうかを指定します。

PARM='DBRC=' に Y か N を指定して、デフォルトの DBRC 指定を変更することができます。DFSPBxxx メンバーまたは DFSIDEF0 インストール・デフォルト・モジュールで DBRC=Y と指定された場合は、このユーティリティの EXEC ステートメントで DBRC=N と指定変更されない限り、このユーティリティの実行時に DBRC が使用されます。DBRC=N と指定すると、このユーティリティのそのときの実行には DBRC が使用されないため、JCL を生成するために DBRC を使用しないでください。

IMS システム定義時に DBRC=N が指定された場合、このユーティリティの EXEC パラメーターで DBRC=Y と指定変更されない限り、このユーティリティの実行時に DBRC は使用されません。DBRC=Y と指定すると、このユーティリティのそのときの実行には DBRC が使用されます。

インストール・デフォルト・モジュール DFSIDEF0 に DBRC=FORCE が指定された場合、このユーティリティの EXEC ステートメントの DBRC パラメーターで DBRC を無効にすることはできません。DBRC は、このユーティリティの実行時に常に使用されます。DBRC=FORCE の指定を変更しようとする、メッセージ DFS044I が出力され、ゼロ以外の戻りコードが返されます。

**重要:** 制御ブロック生成によって DBRC パラメーターを変更したが、DFSUDMP0 ユーティリティを実行する前に IMS 中核を再リンクしていなかったときは、DFSUDMP0 パラメーターと DFSRRC00 パラメーターによって指定する 2 つの実行形式は異なります。

DFSUDMP0 ユーティリティをスタンドアロン・ユーティリティとして、(EXEC DFSUDMP0,DBRC=Y で指定された) IMS 領域コントローラーから独立して実行する場合、新しい DBRC 値は IMS.SDFSRESL の制御ブロック・モジュールからロードされます。DFSUDMP0 ユーティリティを (EXEC PGM=DFSRRC00,PARM='ULU,DFSUDMP0' で指定された) IMS 領域として実行する場合、IMS 中核内の古い DBRC 値が使用されます。

## DFSRRC00

領域コントローラー・プログラム (DFSRRC00) を使用して DFSUDMP0 ユーティリティを実行することを指定します。

DFSUDMP0 とは相互に排他的です。このパラメーターは、上位互換性を保つために、サブパラメーター ULU および UDR とともにサポートされています。

例えば、バッチ領域コントローラーの下での DFSUDMP0 の実行は、EXEC PGM=DFSRRC00,PARM='ULU,DFSUDMP0' によって指定します。

DFSUDMP0 ユーティリティの通常の実行モードは、DFSUDMP0 パラメーターで指定します。

## GSGNAME

グローバル・サービス・グループを識別する 1 から 8 文字のオプションの名前です。

## **IMSPLEX**

DBRC が結合する IMSplex を指定します。

IMSPLEX パラメーターは、DBRC を使用するすべてのジョブ・ステップで指定できます。

## **DBRCGRP**

DBRC グループが使用する RECON データ・セット内で定義する DBRC グループ ID を指定します。

## **ULU**

ロード/アンロード領域を指定します。

このパラメーターは上位互換性を保つためにサポートされており、領域コントローラー・プログラム (DFSRRCC00) を使用して DFSUDMP0 ユーティリティを実行する場合にのみ指定できます。このパラメーターを使用する場合は、通常の IMS 定位置パラメーターを使用できます。

## **UDR**

リカバリー領域を指定します。 PARM=UDR を指定するときは、有効な DB 名が必要ですが、イメージ・コピー・ユーティリティはこれを無視します。

このパラメーターは上位互換性を保つためにサポートされており、領域コントローラー・プログラム (DFSRRCC00) を使用して DFSUDMP0 ユーティリティを実行する場合にのみ指定できます。このパラメーターを使用する場合は、通常の IMS 定位置パラメーターを続けて指定できます。

## **DD** ステートメント

DFSUDMP0 ユーティリティは、必須またはオプションの DD ステートメントをいくつか使用します。

### **STEPLIB DD**

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

### **DFSRESLB DD**

IMS SVC モジュールが入っている許可ライブラリーを指します。

### **IMS DD**

ダンプするデータベースを記述する DBD が入っているライブラリーを定義します。これは DSNAME=IMS.DBDLIB であるのが普通です。このデータ・セットは、直接アクセス・ボリュームに置いておかなければなりません。

### **SYSPRINT DD**

出力メッセージ・データ・セットを定義します。このデータ・セットは、テープ、直接アクセス・ボリューム、またはプリンターに置いておくことも、出力ストリーム (SYSOUT) に入れて送ることもできます。

### **SYSIN DD**

入力制御ステートメント・データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス・ボリュームに置いておくことも、入力ストリーム (DD \* または DD DATA) に入れて送ることもできます。

### Datain DD

ダンプする入力データ・セットを定義します。このステートメントに指定される DD 名は、このデータ・セットを記述している DBD にある名前と同じでなければなりません。DD 名は、ユーティリティー制御ステートメントでも指定されていなければなりません。HALDB データ・セットでは、この DD ステートメントを省略できます。省略すると、このユーティリティーは DISP=SHR を使用してデータ・セットを動的に割り当てます。そうでない場合は、ダンプするデータ・セットごとにこの種類の DD ステートメントが 1 つ存在していなければなりません。このデータ・セットは、直接アクセス・ボリュームに置いておかなければなりません。データ・セットの最小ブロック・サイズは 69 です。これより小さいデータ・セットには、ブランクが埋め込まれます。

このステートメントは、RECON データ・セットに登録されていない IMS カタログ・データベースのイメージ・コピーを作成するのに必要です。

### Areain DD

複数 DEDB エリア・データ・セットでは、areain DD ステートメントは 7 つまで指定できます。区域が RECON データ・セットに登録されているときは、各 areain DD ステートメントで指定される DD 名は区域名ではなく、ターゲット区域の AD リストに登録されている名前と一致するものでなければなりません。区域が登録されていないときは、areain DD ステートメントで指定される DD 名は、区域名 (DBD 区域マクロの DD 名オペランド) でなければなりません。

### Dataout1 DD

ダンプされた出力データ・セットの最初のコピーを定義します。DD ステートメントは、ダンプするデータ・セットごとに 1 つ必要です。DD 名には、1 文字から 8 文字までの任意の文字ストリングを使えますが、その DD 名は関連のユーティリティー制御ステートメントにも指定されている必要があります。出力装置は直接アクセス装置またはテープのいずれかでなければなりません。標準ラベルを使用しなければなりません。使用される BLKSIZE は、最大 BLKSIZE を超えない、論理レコード長の最大倍数です。JCL に BLKSIZE が指定されていると、その BLKSIZE は最大であると見なされます。3380 以外の装置では、デフォルトの最大 BLKSIZE は、z/OS 入出力生成時にその装置の最大値として指定された BLKSIZE です。3380 では、最大 BLKSIZE は 23 KB です。論理レコードが 23 KB を超えるときは、最大値は 32 KB です。

例外: 以下の装置は、32760 バイトを超えるブロック・サイズをサポートします。

- 3480 磁気テープ
- 3490 および 3490E 磁気テープ・サブシステム
- 3590 装置

### Dataout2 DD

関連のユーティリティー制御ステートメントがダンプを 2 部コピーするよう要求している場合にのみ必要です。この名前は制御ステートメントに指定されているものでなければなりません。名前は、テープまたは直接アクセス装置の名前でなければなりません。標準ラベルを使用しなければなりません。BLKSIZE のデフォルトは、出力装置の最大容量です。



これらの 2 出力コピーのどちらかでオープン・エラーが起こるか (メッセージ DFS301A)、どちらかの出力データ・セットへの最初の PUT が失敗すると (メッセージ DFS319A)、現在の制御ステートメントは中止され、次の制御ステートメントが処理されます。

ユーティリティーが最初の PUT を行った後で、どちらかの出力データ・セットに入出力エラーが起こると、どの場合にも RC=08 が返されますが、ユーティリティーは他方の出力データ・セットへのコピーを続けます。各イメージ・コピー制御ステートメントは独立したコピーとして扱われ、最終の戻りコードは、そのジョブで返される最大値となります。

#### Recon1 DD

最初の DBRC RECON データ・セットを定義します。この Recon1 データ・セットは、制御領域が使用しているものと同じ Recon1 データ・セットでなければなりません。

#### Recon2 DD



2 番目の DBRC RECON データ・セットを定義します。この Recon2 データ・セットは、制御領域が使用しているものと同じ Recon2 データ・セットでなければなりません。

#### Recon3 DD


3 番目の DBRC RECON データ・セットを定義します。この Recon3 データ・セットは、制御領域が使用しているものと同じ Recon3 データ・セットでなければなりません。

DFSMDA マクロで動的割り振りを指定していたときは、これらの RECON データ・セット DD 名を使用しないでください。

関連概念:

-  データベースの静止 (データベース管理)
-  RECON データ・セットの初期化および保守 (システム管理)


関連タスク:


-  RSR 環境におけるイメージ・コピー (データベース管理)

関連資料:

177 ページの『第 13 章 データベース・イメージ・コピー 2 ユーティリティー (DFSUDMT0)』

205 ページの『第 14 章 オンライン・データベース・イメージ・コピー・ユーティリティー (DFSUICP0)』

-  DBBATCH プロシージャ (システム定義)

-  DLIBATCH プロシージャ (システム定義)

---

## DFSUDMP0 ユーティリティーの制御ステートメント

DFSUDMP0 ユーティリティーでは、ユーティリティー制御ステートメントを含める必要があります。

DFSUDMP0 ユーティリティー制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

**1** フィールド ID

これは文字 D でなければなりません。D は、このステートメントがデータベース・イメージ・コピー・データ・セット・ユーティリティーの制御ステートメントであることを示します。

**2** コピー数

これは、必要とするコピー数に応じて、1 か 2 にしなければなりません。

**3** ブランクまたは文字 I

I とコーディングすると、KSDS の索引のイメージ・コピーが要求されるため、13 桁目が KSDS の DD 名を参照しなければなりません。I オプションは、OSAM データ・セットでは使用できません。3 桁目をブランクにし、13 桁目が KSDS の DD 名を指定していると、KSDS のイメージ・コピーが得られます。3 桁目はブランクにしなければなりません。

KSDS の組み込み索引のイメージ・コピーとリカバリーは行えません。ただし、KSDS 全体について通常のリカバリーを行えば、組み込み索引と KSDS データ域が再作成されます。

**4-11** dbdname

これは、ダンプするデータ・セットの名前が入っている物理 DBD の名前にしなければなりません。

**13-20** 入力 of DD 名

これは、ダンプする入力データ・セットの DD 名または区域名にしなければなりません。これは、参照される DBD の中になければならず、対応する DD ステートメントが指定されている必要があります。HALDB データ・セットでは、DD ステートメントを省略できます。

**22-29** 出力 of DD 名

これは、1 次出力データ・セットの DD 名でなければなりません。対応する DD ステートメントが指定されていないなければなりません。

**31-38** コピー of DD 名

これは、ダンプされたデータ・セットの 2 番目のコピーの DD 名にしなければなりません。2 桁目が 1 のときは、このフィールドはブランクのままにしておきます。ブランクにしない場合は、対応する DD ステートメントを指定する必要があります。

**40-80** コメントは、40 桁目から 80 桁目までに入れることができます。

---

## DFSUDMP0 ユーティリティーの例

これらの例は DFSUDMP0 ユーティリティーの使用方法を示しています。

このトピックにある例では、桁合わせを容易にするために、SYSIN ステートメントの上に次のようなコメント行が示されています。

```
/* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

このコメント行は単なる参考用です。

このトピックの例で、動的割り振りを行わずに DBRC を使用する場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

図 52. 動的割り振りなしで DBRC を使用するための、またはデータベースが静止保留状況にあるときの DD ステートメント

```
//RECON1 DD DSNAME=RECON1,DISP=SHR
//RECON2 DD DSNAME=RECON2,DISP=SHR
//RECON3 DD DSNAME=RECON3,DISP=SHR
```

サブセクション:

- 『コピーと名前変更』
- 『データ・セットのコピー』
- 176 ページの『区域のコピー』

## コピーと名前変更

この例では、DD 名を DBHI3A とするデータ・セットが、DI32DB01 という名前のデータベースからコピーされます。出力データ・セットの DD 名は DBAOUT1 です。

```
/*
//STEP1 EXEC PGM=DFSUDMP0
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//IMS DD DSNAME=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//DBHI3A DD DSNAME=IMS.DBHI3A,DISP=SHR
//DBAOUT1 DD DSNAME=IMS.DBAOUT1,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=DBDMP1,LABEL=(,SL)
/* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

```
//SYSIN DD *
D1 DI32DB01 DBHI3A DBAOUT1 DUMP SINGLE DATA SET
```

## データ・セットのコピー

この例では、DD 名を DBHI3A および DBHI3B とする 2 つのデータ・セットが、DI32DB01 という名前のデータベースからコピーされます。データ・セット DBHI3A のコピーは 2 つ作成されます。

```
/*
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSUDMP0'
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//IMS DD DSNAME=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//DBHI3A DD DSNAME=IMS.DBHI3A,DISP=SHR
//DBHI3B DD DSNAME=IMS.DBHI3B,DISP=SHR
//DBAOUT1 DD DSNAME=IMS.DBAOUT1,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=DBDMP1,LABEL=(,SL)
//DBAOUT2 DD DSNAME=IMS.DBAOUT2,DISP=(NEW,KEEP),
```

```
//          UNIT=TAPE,VOL=SER=DBDMP2,LABEL=(,SL)
//DBBOUT1 DD DSNAME=IMS.DBBOUT1,DISP=(NEW,KEEP),
//          UNIT=TAPE,VOL=SER=DBDMP3,LABEL=(,SL)
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

```
//SYSIN DD *
D2 DI32DB01 DBHI3A DBAOUT1 DBAOUT2 DATA SET 1-DUMP 1+2
D1 DI32DB01 DBHI3B DBBOUT1 DATA SET 2-DUMP 1
```

## 区域のコピー

この例では、区域名を AREANAM1 とする区域が、DI32DB01 という名前のデータベースからコピーされます。この区域の ADS リストにある DD 名は DDNAME1、DDNAME2、および DDNAME3 で、データ・セット名は DSNAME1、DSNAME2、および DSNAME3 です。

出力データ・セットの DD 名は DBAOUT1 です。

```
//*
//STEP1 EXEC PGM=DFSRRRC00,PARM='ULU,DFSUDMP0'
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//IMS DD DSNAME=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//DDNAME1 DD DSNAME=DSNAME1,DISP=SHR
//DDNAME2 DD DSNAME=DSNAME2,DISP=SHR
//DDNAME3 DD DSNAME=DSNAME3,DISP=SHR
//DBAOUT1 DD DSNAME=IMS.DBAOUT1,DISP=(NEW,KEEP),
//          UNIT=TAPE,VOL=SER=DBDMP1,LABEL=(,SL)
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

```
//SYSIN DD *
D1 DI32DB01 AREANAM1 DBAOUT1 DUMP DUAL DEDB AREA
```

---

## 第 13 章 データベース・イメージ・コピー 2 ユーティリティ (DFSUDMT0)

データベース・イメージ・コピー 2 ユーティリティでは、データ機能記憶管理サブシステム (DFSMS) の並行コピー機能と高速複製機能のいずれかを使用して、IMS データベースのイメージ・コピーをとります。

DFSMS の並行コピー機能と高速複製機能は、データベースのダウン時間を最小にして、ある時点のデータベースまたはデータの集合のバックアップをとることができるハードウェアおよびソフトウェアのソリューションです。

DFSMS 並行コピー機能は、DFSMSdss コマンドの DUMP CONCURRENT を呼び出し、Enterprise Storage Server® (ESS)、3990 ストレージ制御装置、スナップショット機能を持つ RAMAC 仮想アレイ (RVA)、または同等の装置などのハードウェアを必要とします。

DFSMS 高速複製機能は、DFSMSdss コマンドの COPY FASTREP(REQ) を呼び出し、FlashCopy® 機能を持つ Enterprise Storage Server (ESS) またはスナップショット機能を持つ RAMAC 仮想アレイ (RVA) などのハードウェアを必要とします。

サブセクション:

- 『制約事項』
- 178 ページの『前提条件』
- 178 ページの『要件』
- 179 ページの『推奨事項』
- 180 ページの『入出力』
- 180 ページの『戻りコード』
- 181 ページの『JCL 指定』

### 制約事項

データベース・イメージ・コピー 2 ユーティリティに対する制約事項には、ユーザーがどの DFSMS コピー機能を使用しているかに関係なく一般に適用されるものと、特定のコピー機能を使用しているときにのみ適用されるものがあります。

#### 一般的な制約事項

データベース・イメージ・コピー2 ユーティリティには、選択するコピー機能に関係なく、以下の一般的な制約事項があります。

- HSAM、GSAM、または MSDB データベースはこのユーティリティを使用してコピーすることはできません。
- HALDB オンライン再編成でイメージ・コピー・ユーティリティを実行した場合、さらなる制約事項が適用されます。
- データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

## 並行コピー機能の制約事項

現在、データベース・イメージ 2 ユーティリティの並行コピー機能に対して、文書化された制約事項はありません。

## 高速複製機能の制約事項

以下の制約事項は、データベース・イメージ・コピー 2 ユーティリティの高速複製機能を使用している場合のみ適用されます。

- 高速複製で生成される出力コピーは 1 つのみです。
- 高速複製では、データベース・イメージ 2 ユーティリティの同一データ・セット (SAMEDS) オプションはサポートされません。
- 高速複製では、DFSMSdss COMPRESS パラメーターはサポートされません。
- 高速複製イメージ・コピー・データ・セット用に DBRC REUSE 属性はサポートされません。

## 前提条件

現在、DFSUDMT0 ユーティリティに対して文書化された前提条件はありません。

## 要件

データベース・イメージ・コピー 2 ユーティリティには、一般要件と機能に固有の操作要件があります。

### 一般要件

以下の一般要件は、データベース・イメージ・コピー 2 ユーティリティを使用している場合に、選択するコピー機能に関係なく適用されます。

- データベース・イメージ・コピー 2 ユーティリティを利用するには、データベースを DBRC に登録する必要があります。データベース・イメージ・コピー 2 ユーティリティを実行するときには DBRC をアクティブにする必要があります。
- コピーするデータベースとエリア・データ・セットは、使用中のデータベース・イメージ・コピー 2 ユーティリティの機能をサポートするハードウェア上に存在していなければなりません。
- 全機能非 HALDB データベースまたは HALDB 区画の複数の DBDS のために、複数の DBDS Select ステートメントをデータベース・イメージ・コピー 2 ユーティリティに実行依頼する場合、すべてのステートメントで同じイメージ・コピー機能 (高速複製または DFSMS 並行コピー) を指定する必要があります。
- データベース・イメージ・コピー 2 ユーティリティを使用する前に、リカバリ不能なデータベースおよび区域を停止する必要があります。更新はログに記録されないため、イメージ・コピー処理中はリカバリ不能データベースをアクティブにできません。ユーティリティ制御ステートメントでは X (排他的アクセス) を指定しなければなりません。
- DBDS または区域は、このユーティリティが処理できるようにするため、エラー (EQE または EEQE) のない状態にしておく必要があります。ある区域に複数

エリア・データ・セット (MADS) が存在する場合には、エラーのない状態のエリア・データ・セットが少なくとも 1 つは必要です。

入力データ・セットは統合カタログ機能 (ICF) に登録されている必要があり、以下のいずれかの要件を満たしている必要があります。

- SMS の管理下にある。
- 別名を使用してカタログが作成されている (すなわち、データ・セット名の高位修飾子がカタログの別名である)。
- マスター・カタログにカタログが作成されている。

VSAM キー順データ・セット (KSDS) を更新中にコピーするには、そのデータ・セットが SMS の管理下にあり、BWO(TYPEIMS) が AMS DEFINE または ALTER ステートメントに指定されている必要があります。

SMS の管理下でない VSAM KSDS、または BWO(TYPEIMS) が指定されていない VSAM KSDS をコピーするには、ユーティリティの実行前に KSDS データベース・データ・セットを停止する必要があります。

HALDB オンライン再編成でイメージ・コピー・ユーティリティを実行した場合、さらなる要約事項が適用されます。

#### 並行コピー機能の要件

以下の要件は、データベース・イメージ 2 ユーティリティの DFSMS 並行コピー機能を使用している場合のみ適用されます。

イメージ・コピーが物理的に完了するまでデータベースの更新処理が利用不可である間に (DBREL(P) 付きの SMSNOCIC)、イメージ・コピーを作成するために使用されるデータ・セットと一緒に、DISP=OLD を使用した場合に、データベースが QUIESCE HELD 状態のときは、DISP=SHR を指定する必要があります。

コピーするデータベースとエリア・データ・セットは、DFSMS 並行コピー機能をサポートする以下のハードウェア上に存在しなければなりません。

- 3990 ストレージ制御モデル 3、ライセンス内部コード付き拡張機能
- 同等の装置

#### 高速複製機能の要件

以下の要件は、データベース・イメージ 2 ユーティリティの DFSMS 高速複製機能を使用している場合のみ適用されます。

コピーする入力 DBDS またはエリア・データ・セットは、DFSMS 高速複製機能 (IBM Enterprise Storage Server ハードウェアおよびその FlashCopy 機能、または IBM RAMAC 仮想アレイ・ハードウェアおよびそのスナップショット機能など) をサポートするハードウェア上に存在しなければなりません。高速複製イメージ・コピーの出力データ・セットはカタログに登録する必要があります。

#### 推奨事項

現在、DFSUDMT0 ユーティリティに対していかなる推奨事項も示されていません。

## 入出力

ある区域に複数エリア・データ・セット (MADS) が存在する場合には、すべてのエリア・データ・セットを入力として指定する必要があります。ユーティリティーはエラーのないエリア・データ・セットを選択してコピーします。

DFSMS では、入力および出力データ・セットのいずれにも BUFNO キーワードを指定しないでください。

共用副次索引のイメージ・コピーを作成するときは、最初の DBD のみを指定してください。これにより、データ・セット全体がコピーされます。

データベース・イメージ・コピー 2 ユーティリティーからの出力は、データベース・リカバリー・ユーティリティーへの入力として使用できます。

データベース・イメージ・コピー 2 ユーティリティーの並行コピー機能によって作成された出力イメージ・コピーは、標準バッチ・イメージ・コピー形式ではなく、DFSMS ダンプ形式になります。このコピーは、イメージ・コピーがとられたときに指定されたパラメーターによって、SMSNOCIC または SMSCIC イメージ・コピーとして DBRC に登録されます。

データベース・イメージ・コピー 2 ユーティリティーの高速複製機能が作成した出力イメージ・コピーは、標準バッチ・イメージ・コピー形式または DFSMS ダンプ形式ではなく、元のデータ・セットの正確なコピーです。このコピーは、クリーン・イメージ・コピーかファジー・イメージ・コピーかによって、SMSOFFLC または SMSONLC イメージ・コピーとして DBRC に登録されます。

並行コピー機能を使用する場合は最大 4 つの出力イメージ・コピーを作成できますが、最初の 2 つだけが RECON データ・セットに登録されます。2 つ以上のコピーを指定すると、コピーの実行中に入出力エラーが起こっても、ユーティリティーは残りのコピーを最後まで続けることができるという利点があります。

高速複製機能を使用する場合、ユーティリティーは 1 つの出力イメージ・コピーのみを作成できます。

## 戻りコード

データベース・イメージ・コピー 2 ユーティリティーから次のような戻りコードが返されます。

コード 意味

- 0 処理が正常に完了した
- 4 1 つ以上のデータ・セットについて警告メッセージが出された
- 8 すべてのデータ・セットではなく一部のデータ・セットで処理が成功した
- 12 すべてのデータ・セットで処理が失敗した
- 16 すべての操作が完了する前に重大エラーによりユーティリティーが終了した

これらの戻りコードは、後続のジョブ・ステップの EXEC ステートメントで COND= パラメーターを指定すると、テストすることができます。



## JCL 指定

データベース・イメージ・コピー 2 ユーティリティは、標準の z/OS ジョブとして実行されます。次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

データベース・イメージ・コピー 2 ユーティリティの JCL 指定は、ユーティリティ制御ステートメントを含むこともできます。

### EXEC ステートメント

EXEC ステートメントは、必要なステートメントが入っているカタログ式プロシージャを呼び出すことができますが、次の形式にすることもできます。

```
▶▶ EXEC PGM=DFSRRCO0, PARM= 'ULU,DFSUDMT0'
                                └──'UDR,DFSUDMT0,dbdname'──┘
```

このステートメントは次のとおりです。

### ULU

ロード/アンロード領域を指定します。

### UDR

リカバリー領域を指定します。 PARM=UDR を指定するときは、有効な DB 名が必要ですが、イメージ・コピー 2 ユーティリティはこれを無視します。

### DD ステートメント

#### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

#### DFSMFIMS DD

入力制御ステートメント・データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス・ボリュームに置いておくことも、入力ストリーム (DD \* または DD DATA) に入れて送ることもできます。この DD ステートメントは、IMS IC2 ジョブ名またはアカウント ID のために DSS の実行に費やされた時間を関連付ける場合に使用します。DFSMFIMS DD ステートメントはオプションです。

DFSMFIMS DD ステートメントは、ユーティリティ制御ステートメント TYPE30=*jobname* または TYPE30=*'alphanumeric\_string'* のいずれかを含み、1 桁目から始めます。

#### *jobname*

ブランク・スペースが後に 1 つ続くキーワード。このブランク・スペースの後にコメントを指定できます。

### *alphanumeric\_string*

単一引用符で囲まれた 1 から 8 バイトの英数字ストリング。先頭文字は A から Z の英字です。終了引用符の後にブランク・スペースが 1 つ続きます。このブランク・スペースの後にコメントを指定できます。

単一引用符内では、先頭、途中、末尾のいずれにもブランクを指定できません。

*jobname* キーワードまたは *alphanumeric\_string* 値は、SMF タイプ 30 アカウンティング・レコードに表示されます。

DFSFMFIMS DD ステートメントは、*jobname* レコードまたは *alphanumeric\_string* 値を DFSMFDSS ヘササーバー名として渡します。name. 小文字で入力されたパラメーターは、大文字に変換されます。*jobname* キーワードまたは *alphanumeric\_string* 値は、DFSMSDSS により生成される SMF タイプ 30 アカウンティング・レコードに表示されます。

### **DFSRESLB DD**

IMS SVC モジュールが入っている許可ライブラリーを指します。

### **IMS DD**

ダンプするデータベースを記述する DBD が入っているライブラリーを定義します。これは DSNNAME=IMS.DBDLIB であるのが普通です。このデータ・セットは、直接アクセス・ボリュームに置いておかなければなりません。

### **SYSPRINT DD**

出力メッセージ・データ・セットを定義します。このデータ・セットは、テープ、直接アクセス・ボリューム、またはプリンターに置いておくことも、出力ストリーム (SYSOUT) に入れて送ることもできます。

### **SYSIN DD**

入力制御ステートメント・データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス・ボリュームに置いておくことも、入力ストリーム (DD \* または DD DATA) に入れて送ることもできます。

### *datain* DD

ダンプする入力データ・セットを定義します。このステートメントに指定される DD 名は、このデータ・セットを記述している DBD にある名前と同じでなければなりません。DD 名は、ユーティリティー制御ステートメントでも指定されていなければなりません。HALDB データ・セットでは、この DD ステートメントを省略できます。それ以外のデータ・セットでは、コピーするデータ・セットごとに 1 つずつ、このタイプの DD ステートメントが必要です。このデータ・セットは、並行コピー機能をサポートするハードウェアに置いておかなければなりません。

HALDB データ・セットでは、この DD ステートメントが存在する場合、ユーティリティーは指定された後処理に従います。そうでない場合は、DISP=SHR がデフォルトです。DD ステートメントに DD DUMMY、DSN=NULLFILE、または非アクティブ・データ・セットが含まれている場合、ユーティリティーは DD ステートメントで指定された後処理を使用して、アクティブ・データ・セットを動的に割り振ります。

複数 DEDB エリア・データ・セットでは、*datain* DD ステートメントは 7 つまで指定できます。各 *datain* DD ステートメントで指定される DD 名は、区域名ではなくて、ターゲット区域の AD リストに登録されている名前と一致するものでなければなりません。

#### ***dataout1* DD**

出力データ・セットを定義します。

この指定には以下の項目を入れることができます。

- 出力データ・セット名
- 出力データ・セット属性
- 出力データ・セット *volser*

並行コピー機能を使用して複数の出力コピーを作成する場合、*dataout1* DD ステートメントで最初の出力コピーを定義します。

高速複製機能では、出力データ・セットの HLQ 仕様ステートメントを含める場合は、*dataout1* DD ステートメントは非 SMS 管理データ・セットに対してオプションです。*dataout1* DD ステートメントを含める場合、このステートメントは非 SMS 管理出力データ・セットに対して *volser* 情報を指定するためにのみ使用されます。出力データ・セット属性は入力データ・セット属性から抽出され、出力データ・セット名は HLQ 仕様によって生成されます。SMS 管理データ・セットでは、属性は入力データ・セットから取り出されます。

DD 名には、1 文字から 8 文字までの任意の文字ストリングを使えますが、その DD 名は関連のユーティリティ制御ステートメントにも指定されている必要があります。ESS および RVA のみがフラッシュ・コピーをとるための有効な装置です。

注: 複数の DBDS をコピーするときは、複数セットの出力 DD ステートメントを用意しなければなりません。

#### ***dataout2* DD**

並行コピー機能を使用しており、関連のユーティリティ制御ステートメントがダンプを 2 部以上要求している場合のみ必要です。この名前は制御ステートメントに指定されているものでなければなりません。名前は、テープまたは直接アクセス装置の名前でなければなりません。

高速複製機能は、データ・セットの複数の出力コピーをサポートしていません。

#### ***dataout3* DD**

並行コピー機能を使用しており、関連のユーティリティ制御ステートメントがダンプを 3 部以上要求している場合のみ必要です。この名前は制御ステートメントに指定されているものでなければなりません。名前は、テープまたは直接アクセス装置の名前でなければなりません。

高速複製機能は、データ・セットの複数の出力コピーをサポートしていません。

#### ***dataout4* DD**

並行コピー機能を使用しており、関連のユーティリティ制御ステートメントがダンプを 4 部要求している場合のみ必要です。この名前は制御ステートメントに指定されているものでなければなりません。名前は、テープまたは直接アクセス装置の名前でなければなりません。

高速複製機能は、データ・セットの複数の出力コピーをサポートしていません。

### RECON1 DD

最初の DBRC RECON データ・セットを定義します。この RECON1 データ・セットは、制御領域が使用しているものと同じ RECON1 データ・セットでなければなりません。

### RECON2 DD

2 番目の DBRC RECON データ・セットを定義します。この RECON2 データ・セットは、制御領域が使用しているものと同じ RECON2 データ・セットでなければなりません。

### RECON3 DD

3 番目の DBRC RECON データ・セットを定義します。この RECON3 データ・セットは、制御領域が使用しているものと同じ RECON3 データ・セットでなければなりません。

DFSMDA マクロで動的割り振りを指定していたときは、これらの RECON データ・セット DD 名を使用しないでください。

### JCL の例

以下の図に、4 つのデータベースの DBDS をコピーするのに使用する並行コピー機能のユーティリティー制御ステートメントを示します。コピーする DBDS は、いずれも GROUPXYZ というグループのメンバーになっています。このデータ・セット全部を同じ処理オプション (XL) のもとでコピーします。すべての DBDS のイメージ・コピーが論理的に完了すると、GROUPXYZ グループ全体について 1 つの DFS3121A メッセージが出されます。この例が有効なのは、出力データ・セットがどれも、他の出力データ・セットと同じテープ・ボリューム上にない場合のみです。

図 53. 並行コピー機能によるデータ・グループの指定

```
|...+...1...+...2...+...3...+...4...+...5...+...6...+...7..
//SYSIN DD *
G GROUPXYZ XL
 2 DBNAME1 DDNAME1A ICOUT1A1 ICOUT1A2
 2 DBNAME1 DDNAME1B ICOUT1B1 ICOUT1B2
 2 DBNAME1 DDNAME1C ICOUT1C1 ICOUT1C2
 2 DBNAME1 DDNAME1D ICOUT1D1 ICOUT1D2
 2 DBNAME1 DDNAME1E ICOUT1E1 ICOUT1E2
 2 DBNAME2 DDNAME2A ICOUT2A1 ICOUT2A2
 2 DBNAME2 DDNAME2B ICOUT2B1 ICOUT2B2
 2 DBNAME3 DDNAME3A ICOUT3A1 ICOUT3A2
 2 DBNAME4 DDNAME4A ICOUT4A1 ICOUT4A2
 2 DBNAME4 DDNAME4B ICOUT4B1 ICOUT4B2
 2 DBNAME4 DDNAME4C ICOUT4C1 ICOUT4C2
/*
```

以下の図に、4 つのデータベースの DBDS をコピーするのに使用する並行コピー機能のユーティリティー制御ステートメントを示します。

図 54. 並行コピー機能を使用する場合の単一出力データ・セットへのイメージ・コピーのスタック

```
|...+...1...+...2...+...3...+...4...+...5...+...6...+...7..
//SYSIN DD *
 2 DBNAME1 DDNAME1A ICOUT101 ICOUT102 XL
```

```

S DBNAME1 DDNAME1B XL
S DBNAME1 DDNAME1C XL
S DBNAME1 DDNAME1D XL
S DBNAME1 DDNAME1E XL
2 DBNAME2 DDNAME2A ICOUT201 ICOUT202 S
S DBNAME2 DDNAME2B S
2 DBNAME3 DDNAME3A ICOUT3A1 ICOUT3A2 S C2
2 DBNAME4 DDNAME4A ICOUT4A1 ICOUT4A2 XP
2 DBNAME4 DDNAME4B ICOUT4B1 ICOUT4B2 XP
2 DBNAME4 DDNAME4C ICOUT4C1 ICOUT4C2 XP
/*

```

DBNAME1 の 5 つの DBDS のイメージ・コピーは、ICOUT101 と ICOUT102 というデータ・セットにスタックされます。DBNAME2 の 2 つの DBDS のイメージ・コピーは、ICOUT201 と ICOUT202 というデータ・セットにスタックされます。DBNAME3 の 1 つの DBDS と、DBNAME4 の 3 つの DBDS は、いずれも独自のイメージ・コピー出力データ・セットにコピーされます。出力データ・セットがどれも、他の出力データ・セットと同じテープ・ボリューム上にありません。

ここでは、グループ名を指定していませんから、DBDS ごとに異なる処理オプション (SIXL|XP) を指定できます。DBNAME1 では、すべての DBDS のイメージ・コピーが論理的に完了したときに 1 つの DFS3121A メッセージが出されます。DBNAME4 では、すべての DBDS のイメージ・コピーが物理的に完了したときに 1 つの DFS3141A メッセージが出されます。

以下の図に、並行コピー機能を使用するデータベース・イメージ・コピー 2 ユーティリティの JCL を示します。


図 55. 並行コピー機能を使用するデータベース・イメージ・コピー 2 ユーティリティの JCL


```

//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSUDMT0'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//RECON1 DD DSN=RECON1,DISP=SHR
//RECON2 DD DSN=RECON2,DISP=SHR
//RECON3 DD DSN=RECON3,DISP=SHR
//SYSPRINT DD SYSOUT=A
//DBIN DD DSN=IMS.DBIN,DISP=SHR
//OUTPUTD1 DD DSN=IMS.DBAOUT1,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=DMPN01,LABEL=(,SL)
//OUTPUTD2 DD DSN=IMS.DBAOUT2,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=DMPN02,LABEL=(,SL)
//OUTPUTD3 DD DSN=IMS.DBAOUT3,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=DMPN03,LABEL=(,SL)
//OUTPUTD4 DD DSN=IMS.DBAOUT4,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=DMPN04,LABEL=(,SL)
//SYSIN DD *
4 DBDNAMEX DBIN OUTPUTD1 OUTPUTD2 OUTPUTD3 OUTPUTD4 XL

```

関連概念:

 データベース・バックアップ・コピーの作成 (データベース管理)

 HALDB データベースの再編成 (データベース管理)

関連資料:

- ➡ z/OS: DFSMSdss - 並行コピー
- ➡ DBBBATCH プロシージャ (システム定義)
- ➡ DLIBATCH プロシージャ (システム定義)
- ➡ z/OS: MVS レコード・タイプ 30

---

## DFSUDMT0 ユーティリティーの制御ステートメント

データベース・イメージ・コピー 2 ユーティリティーには、5 つの制御ステートメントがあります。データベース・イメージ・コピー 2 ユーティリティーの実行では、ユーティリティー制御ステートメントに指定されるすべての DBDS が、並行に処理されるものとして、あるいは単一のダンプ・コマンドで処理されるものとして、DFSMSdss に対して指定されます。制御ステートメントのシリアル処理という方法はありません。

「コピー保全性」オプションと「通知・保留」オプションは、同じデータベースのすべての DBDS オカレンスで同じでなければなりません。許可は、データベースまたは区域のレベルで与えられます。

各ユーティリティー制御ステートメントは、桁位置を使用する固定形式です。グループ名ステートメントと DBDS SELECT ステートメントのどちらに構文エラーがあっても、ユーティリティーの実行は終了します。

サブセクション:

- 『並行コピー機能 DBDS 選択ステートメント』
- 189 ページの『高速複製 DBDS 選択ステートメント』
- 191 ページの『グループ名ステートメント』
- 193 ページの『高速複製機能の HLQ 仕様ステートメント』
- 195 ページの『SET PATCH 制御ステートメント』

### 並行コピー機能 DBDS 選択ステートメント

各並行コピー DBDS 選択ステートメントは、コピーする 1 つの DBDS を示します。DBDS 選択ステートメントには、出力コピー数が含まれ、各出力コピー・データ・セットに DD 名を与えます。ほかに、データ保全性オプションと通知オプションを始め、DFSMSdss に直接渡されるその他のオプションを選択できます。

58 から 61 桁目に指定されたパラメーターは、グループ名ステートメントが指定されると無視されます。

位置 説明

#### 1 ステートメント・タイプ

ブランクまたは S。ブランクなら、出力データ・セットはこの制御ステートメントで指定されています。S の場合、コピーは、S を指定していない先行する制御ステートメントで指定されたのと同じ出力データ・セットに書き込まれることとなります。S オプションで効力のあるのは、桁 4 から 20、58、および 59 だけです。

- 2 コピー数  
1 から 4。この値は、22 から 56 桁目に指定されている出力 DD 名の数以下でなければなりません。1 桁目に S があると、無視されます。
- 3 無視される
- 4-11 dbdname  
これは、ダンプするデータ・セットの名前が入っている物理 DBD の名前にしなければなりません。
- 12 無視される
- 13-20 入力の DD 名  
これは、ダンプする入力データ・セットの DD 名または区域名にしなければなりません。これは、参照される DBD の中になければならず、対応する DD ステートメントが指定されている必要があります。HALDB データ・セットでは、DD ステートメントを省略できます。
- 21 無視される
- 22-29 出力の DD 名  
これは、1 次出力データ・セットの DD 名でなければなりません。対応する DD ステートメントが指定されていないなければなりません。  
1 桁目に S が指定されると、このフィールドは無視されます。
- 30 無視される
- 31-38 OUTPUT2 DD 名  
これは、ダンプされたデータ・セットの 2 番目のコピーの DD 名にしなければなりません。2 桁目に 2、3、または 4 が含まれる場合には、この DD 名を指定して、これに対する DD ステートメントを提供する必要があります。  
1 桁目に S が指定されると、このフィールドは無視されます。
- 39 無視される
- 40-47 OUTPUT3 DD 名  
これは、ダンプされたデータ・セットの 3 番目のコピーの DD 名にしなければなりません。2 桁目に 3 または 4 が含まれる場合には、この DD 名を指定して、これに対する DD ステートメントを提供する必要があります。このデータ・セットは DBRC に登録されません。  
1 桁目に S が指定されると、このフィールドは無視されます。
- 48 無視される
- 49-56 OUTPUT4 ddname  
これは、ダンプされたデータ・セットの 4 番目のコピーの DD 名にしなければなりません。2 桁目に 4 が含まれる場合には、この DD 名を指定して、これに対する DD ステートメントを提供する必要があります。このデータ・セットは DBRC に登録されません。
- 57 無視される

## 58 DBDS のコピー保全性制御

イメージ・コピー中にデータベースを更新できるかどうかを指定します。X (排他的) は、論理コピー・フェーズ、もしくは論理コピー・フェーズと物理コピー・フェーズの両方 (どちらになるかは、59 桁目の指定で決まります) で、データベースの更新ができないことを意味します。イメージ・コピーは SMSNOCIC イメージ・コピーとして RECON に記録されます。S (共用) は、イメージ・コピー・フェーズ中にデータベースを更新できることを示します。イメージ・コピーは SMSCIC イメージ・コピーとして RECON に記録されます。デフォルトは S です。

リカバリー不能データベースの DBDS とリカバリー不能区域には、X を指定しなければならないことに注意してください。X を指定しておかないと、その DBDS または区域のイメージ・コピー処理は失敗します。

SMSNOCIC イメージ・コピーは、IMS High Performance Pointer Checker for z/OS への入力として使用できますが、SMCIC は使用できないことに注意してください。

このフィールドは、グループ名ステートメントが指定されると無視されます。

## 59 DBDS の通知・保留制御

58 桁目が X の場合に、このフィールドはデータベースをいつ更新処理に使用できるかを指定します。58 桁目が S の場合、このフィールドは無視されます。P の場合、物理コピー完了後にデータベースが更新に使用できることを示します。L は、論理コピーが完了した後でデータベースを更新に使用できることを指定します。L がデフォルトです。

RECON データ・セット内でデータベースに「イメージ・コピーが必要」のフラグが付けられていれば、DBRC は物理コピーが完了するまで更新処理を許可しません。

このフィールドは、グループ名ステートメントが指定されると無視されます。

## 60 DBDS の COMPRESS 制御

この桁に C を指定すると、DFSMS COMPRESS オプションが使用されます。この桁がブランクのままなら、DFSMSdss COMPRESS オプションは使用されません。

このフィールドは、グループ名ステートメントが指定されると無視されます。

## 61 DBDS の OPTIMIZE 制御

OPTIMIZE は、パフォーマンスを修正する DFSMS オプションです。OPTIMIZE のもとでは、実行にかかる時間と実ストレージ/仮想ストレージ割り振り量の間、またユーティリティー・パフォーマンスとアプリケーション/トランザクション処理の間に、トレードオフ関係が導入されます。

OPTIMIZE 制御には 4 つの最適化レベルが用意されていて、1 つの入出力コマンドで転送される DASD トラックの数がレベルごとに異なります。最適化レベルは番号で表され、1、2、3、4 のいずれかになります。この制御オプションを省略 (ブランクに) すると、ユーティリティーのデフォルト・



レベルが使用されます。ファジー・コピーのデフォルトは OPTIMIZE(1)、クリーン・コピーのデフォルトは OPTIMIZE(4) です。

このフィールドは、グループ名ステートメントが指定されると無視されません。

**62-72** この IMS リリースでは指定されていません。

**73-80** ユーザーのコメントを入れることができます。

グループ名ステートメントが指定されると、グループ名ステートメントの 58 から 61 桁目で指定されたパラメーターは、DBDS SELECT ステートメントで指定された対応するパラメーターに優先します。

## 高速複製 DBDS 選択ステートメント

各高速複製 DBDS 選択ステートメントは、コピーする 1 つの DBDS を示します。このステートメントには出力コピー数が含まれ、各出力コピー・データ・セットに DD 名を与えます。ほかに、データ保全性オプションと通知オプションを始め、DFSMSdss に直接渡されるその他のオプションを選択できます。

58 から 61 桁目に指定されたパラメーターは、グループ名ステートメントが指定されると無視されます。

位置 説明

**1** 高速複製ではこの桁を空白にしておきます。

**2** コピー数

高速複製は 1 つの出力コピーのみ生成するため、高速複製に対する有効な値は 『1』 または空白のみです。

1 以外の値を指定すると、イメージ・コピーが失敗し、メッセージ DFS3158A が理由コード 4 とともに発行されます。

**3** 無視される

**4-11** dbdname

これは、ダンプするデータ・セットの名前が入っている物理 DBD の名前にしなければなりません。

**12** 無視される

**13-20** 入力 DD 名

これは、ダンプする入力データ・セットの DD 名または区域名にしなければなりません。これは、参照される DBD の中になければならず、対応する DD ステートメントが指定されている必要があります。HALDB データ・セットでは、DD ステートメントを省略できます。

**21** 無視される

**22-29** 出力 DD 名

出力 DD 名

対応する DD ステートメントで識別される出力データ・セットの DD 名を指定します。

22 から 29 桁目に DD 名を指定する必要があるかどうかは、以下によって決まります。

- 出力データ・セットが SMS の管理下にあり、HLQ 仕様ステートメントでデータ・セット名に高位修飾子を指定する場合、22 から 29 桁をブランクのままにして、対応する DD ステートメントを含めないようにします。
- 出力データ・セットが SMS の管理下にあるが、HLQ 仕様ステートメントでデータ・セット名に高位修飾子を指定しない場合、DD 名を指定して、出力データ・セットを定義する対応 DD ステートメントを含める必要があります。
- 出力データ・セットが SMS の管理下になく、HLQ 仕様ステートメントでデータ・セット名に高位修飾子を指定する場合、DD 名を入力し、データ・セットを割り振る必要があるターゲット・ボリュームのみ指定する DD ステートメントを含める必要があります。この DD ステートメントにデータ・セット名を指定する必要はありません。
- 出力データ・セットが SMS の管理下になく、HLQ 仕様ステートメントでデータ・セット名に高位修飾子を指定しない場合、DD 名を指定し、データ・セット名とターゲット・ボリュームの両方を指定する DD ステートメントを指定する必要があります。

**30-57** これらの桁は、高速複製 DBDS 選択ステートメントで無視されます。

**58** DBDS のコピー保全性制御

イメージ・コピー中にデータベースを更新できるかどうかを指定します。X (排他的) はイメージ・コピー・プロセス中にデータベースを更新できないことを示します。イメージ・コピーは、RECON データ・セットに SMSOFFLC として記録されます。S (共用) はイメージ・コピー・プロセス中にデータベースを更新できることを示します。イメージ・コピーは SMSONLC イメージ・コピーとして RECON に記録されます。デフォルトは S です。

リカバリー不能データベースの DBDS とリカバリー不能区域には、X を指定しなければならないことに注意してください。X を指定しておかないと、その DBDS または区域のイメージ・コピー処理は失敗します。

SMSOFFLC イメージ・コピーは、IMS High Performance Pointer Checker for z/OS への入力として使用できますが、SMSONLC イメージ・コピーは使用できません。

このフィールドは、グループ名ステートメントが指定されると無視されます。

**59-61** これらの桁は、高速複製 DBDS 選択ステートメントで無視されます。

**62** コピー方法

この桁に 『F』 を入力し、DFSMS 高速複製機能を選択します。

**63** HLQ 標識

有効値は 『H』 またはブランクです。

このフィールドが 『H』 の場合:

- このステートメントの直後に HLQ 仕様ステートメントが続きます。HLQ 仕様ステートメントは、動的に生成されたイメージ・コピー・データ・セット名で使用される高位修飾子を指定します。
- 出力データ・セットは DFSMSdss によって動的に作成されます。
- 出力データ・セット名は DD ステートメント上で JCL で提供されません。
- DD ステートメントをボリューム通し番号情報を指定するために使用すると、22 から 28 桁にはその dataout1 DD ステートメントの名前が含まれます。

**64-72** このリリースでは指定されていません。

**73-80** ユーザーのコメントを入れることができます。

グループ名ステートメントが指定されると、グループ名ステートメントの 58 から 61 桁目で指定されたパラメーターは、DBDS SELECT ステートメントで指定された対応するパラメーターに優先します。

## グループ名ステートメント

オプションのグループ名をデータベース・イメージ・コピー 2 ユーティリティに対して指定するには、グループ名制御ステートメントを最初の制御ステートメントとして提供する必要があります。1 つのジョブ・ステップの入力ストリームで、グループ名ステートメントは 1 つだけ使用できます。

グループ名ステートメントの次の属性に注意してください。

- グループ名ステートメントには、そのグループを構成する DBDS を指定する制御ステートメントを続けなければなりません。
- 指定された名前を持つグループが存在するかどうか、ユーティリティによる確認はありません。
- グループ名ステートメントに続いて指定される DBDS が実際にそのグループのメンバーかどうか、ユーティリティによる確認はありません。

グループ名ステートメントに指定するオプションは、コピー中のすべての DBDS に適用されます。58 から 62 桁目に指定されるか 58 から 62 桁目でデフォルトになるパラメーターは、DBDS SELECT ステートメントに指定された対応パラメーターに優先します。

位置 説明

**1** ステートメント・タイプ

「G」は、この制御ステートメントでグループ名が指定されることを意味します。

**2-3** 予約済み

**4-11** グループ名

後続の制御ステートメントで指定される DBDS は、このグループのメンバーです。

**12-57** 予約済み

**58** グループ全体のコピー保全性制御

イメージ・コピー中にデータベースを更新できるかどうかを指定します。有効値は X または S です。デフォルトは S です。

X (排他的) は、イメージ・コピー・プロセスのすべてまたは一部でデータベースを更新できないことを示します。並行コピー機能の場合は、論理コピー・フェーズ中または論理および物理コピー・フェーズ中であり、これは 59 桁目の指定によって決まります。どちらの場合も、イメージ・コピーは RECON に SMSNOCIC イメージ・コピーとして記録されます。高速複製機能では、X は、イメージ・コピー・プロセスが完了するまでデータベースを更新できないことを示します。イメージ・コピーは、RECON データ・セットに SMSOFFLC イメージ・コピーとして記録されます。

S (共用) は、イメージ・コピー・プロセス中にデータベースを更新できることを示します。並行コピー・プロセスでは、イメージ・コピーは RECON に SMSCIC イメージ・コピーとして記録されます。高速複製機能では、イメージ・コピーは RECON データ・セットに SMSONLC イメージ・コピーとして記録されます。

**重要:** リカバリー不能なデータベースには、X を指定しなければなりません。S を指定すると、グループのリカバリー不能メンバーではイメージ・コピー処理が失敗します。

**要確認:** SMSNOCIC イメージ・コピーは、IMS High Performance Pointer Checker for z/OS への入力として使用できますが、SMCIC イメージ・コピーは使用できません。

59 並行コピー機能では、グループの制御を通知および保持します。

58 桁目が X の場合に、このフィールドはデータベースをいつ更新処理に使用できるかを指定します。このフィールドは以下の場合に無視されます。

- 58 桁目が S と指定されている場合
- グループ名ステートメントの 62 桁目の F によって高速複製機能が使用されている場合

P は、物理コピーが完了した後でデータベースを更新に使用できることを指定します。L は、論理コピーが完了した後でデータベースを更新に使用できることを示します。L がデフォルトです。

RECON データ・セット内でデータベースに「イメージ・コピーが必要」のフラグが付けられていれば、DBRC は物理コピーが完了するまで更新処理を許可しません。

60 グループの COMPRESS 制御

この桁に C を指定すると、DFSMSdss COMPRESS オプションが使用されます。この桁がブランクのままなら、DFSMS COMPRESS オプションは使用されません。

高速複製機能は COMPRESS オプションをサポートしません。高速複製オプションを 62 桁目の 『F』 で指定すると、このオプションは無視されます。

61 グループの OPTIMIZE 制御

OPTIMIZE は、パフォーマンスを修正する DFSMS オプションです。OPTIMIZE のもとでは、実行にかかる時間と実ストレージ/仮想ストレージ割り振り量の間、またユーティリティー・パフォーマンスとアプリケーション/トランザクション処理の間に、トレードオフ関係が導入されます。

OPTIMIZE 制御には 4 つの最適化レベルが用意されていて、1 つの入出力コマンドで転送される DASD トラックの数がレベルごとに異なります。最適化レベルは番号で表され、1、2、3、4 のいずれかになります。この制御オプションを省略すると、ユーティリティーのデフォルト・レベルが使用されます。ファジー・コピーのデフォルトは OPTIMIZE(1)、クリーン・コピーのデフォルトは OPTIMIZE(4) です。

62 コピー方法。

F またはブランク。高速複製機能を選択するには F を入力します。並行コピー機能の場合はブランクのままにします。

63-72 この IMS リリースでは指定されていません。

73-80 ユーザーのコメントを入れることができます。

## 高速複製機能の HLQ 仕様ステートメント

HLQ 仕様ステートメントは、動的に生成した出力データ・セット名に対して高位修飾子を定義し、低位修飾子としてタイム・スタンプを付加するかどうかを指定します。HLQ 仕様ステートメントは、高速複製イメージ・コピーに対してのみ有効です。

HLQ 仕様ステートメントには、以下の要件があります。

- HLQ 仕様ステートメントが適用される高速複製 DBDS 選択ステートメントの 63 桁目に H を指定する必要があります。
- このステートメントは、HLQ 仕様ステートメントが適用される DBDS 選択ステートメントの直後に指定する必要があります。

HLQ を高速複製で使用する場合、動的に生成される IC データ・セット *ichlq.dbdname.ddname.Dyyddd.Thhmmss* または *ichlq.dbdname.ddname* とすることができます。3 桁目に Y を指定しない場合、IC データ・セットの形式は *ichlq.dbdname.ddname* になります。

データ・セット名の形式は、3 桁目のタイム・スタンプ指定の追加指定によって決まります。3 桁目のタイム・スタンプ指定が N か、指定を省略すると、形式は *ichlq.dbdname.ddname* になります。3 桁目のタイム・スタンプ指定が Y の場合、形式は *ichlq.dbdname.ddname.Dyyddd.Thhmmss* になります。

出力データ・セット名を HLQ 指定を使用して動的に生成する場合、データ・セット名の形式は *ichlq.dbdname.ddname.Dyyddd.Thhmmss* になります。

- ここで、*hlq* は、HLQ 仕様ステートメントの 4 から 29 桁目で定義される高位修飾子です。
- *dbdname* はコピーするデータベースの DBD 名です (最低 1 文字)。
- *ddname* は、出力データ・セット・グループまたは区域の DD 名です (最低 1 文字)。

- `Dyyddd.Thhmmss` は、HLQ 仕様ステートメントの 3 桁目に指定されるオプションのタイム・スタンプです。

表 6. 高速複製 HLQ 仕様ステートメントの桁とパラメーターの説明

位置	説明
1	ステートメント・タイプを指定します。H が必須です。
2	HLQ 仕様ステートメントが適用される出力データ・セットを示します。  高速複製は出力コピーを 1 つだけ生成するので、有効な値は 『1』 または ブランクです。ブランクの場合のデフォルトは 1 です。
3	低位修飾子のタイム・スタンプ指定です。  有効な値は次のとおりです。  Y タイム・スタンプ・トレーラーは、低位修飾子として出力データ・セッ ト名に付加されます。例えば、出力データ・セット名の結果の形式は <code>ichlq.dbdname.ddname.Dyyddd.Thhmmss</code> となります。  N タイム・スタンプ・トレーラーは、出力データ・セット名に付加されま せん。例えば、出力データ・セット名の結果の形式は <code>ichlq.dbdname.ddname</code> となります。
4-29	高位修飾子指定。  動的に生成される出力データ・セット名で高位修飾子として使用される文字 ストリングを定義します。
30	ブランク
31-38	高速複製イメージ・コピー・データ・セットの記憶クラス  オプションで、イメージ・コピー・データ・セットの割り振りの記憶クラス を指定します。この指定は、自動クラス選択 (ACS) ルーチンへの入力とし て使用されます。ユーザーは、指定されたストレージ・クラスに対する適切 な RACF® 許可を持っている必要があります。  31 から 38 桁目がブランクの場合、記憶クラスは使用されません。  このキーワードは、データベース・イメージ・コピー 2 ユーティリティ の DFSMS 高速複製オプションに対してのみ有効です。
39	ブランク
40-47	高速複製イメージ・コピー・データ・セットの管理クラス  オプションで、イメージ・コピー・データ・セットの割り振りの管理クラス を指定します。この指定は、ACS ルーチンへの入力として使用されます。 ユーザーは、指定された管理クラスに対する適切な RACF 許可を持ってい る必要があります。  40 から 47 桁目がブランクの場合、管理クラスは使用されません。  このキーワードは、データベース・イメージ・コピー 2 ユーティリティ の DFSMS 高速複製オプションに対してのみ有効です。

## SET PATCH 制御ステートメント

SET PATCH 制御ステートメントは、データベース・イメージ・コピー 2 ユーティリティの現在の実行に対する DFSMSdss 処理オプションを設定します。この制御ステートメントでは、DFSMSdss SET PATCH のコマンド・オフセットおよび値を指定できます。

各 SET PATCH 制御ステートメントには、最大 10 の SET PATCH 指定を含めることができます。複数の SET PATCH 指定は、連続して入力する必要があります。ある桁を空白のままにすると、後続の SET PATCH 指定は無視されます。

複数の SET PATCH 制御ステートメントを含めることができます。

SET PATCH 制御ステートメントは、制御ステートメント・データ・ストリーム内に Group ステートメントが存在すれば、それに続けて指定できます。

各 SET PATCH 指定は、*xxx* をパッチ・バイト、*yy* をパッチ・バイトの設定値とした場合に、以下の形式で入力する必要があります。 *xxx=yy* SET PATCH 制御ステートメントに無効な構文が含まれていると、処理が終了して、エラー・メッセージ DFS3158A が理由コード 3 とともに発行されます。

ただし、データベース・イメージ・コピー 2 ユーティリティは、無効な DFSMSdss パッチ・バイト値を検査しません。例えば、132=FF を指定すると、データベース・イメージ・コピー 2 ユーティリティは、132 が有効なパッチ・バイトであるかどうかを検証しません。

表 7. データベース・イメージ・コピー 2 ユーティリティ SET PATCH 制御ステートメントの桁とパラメーターの説明

位置	説明
1	制御ステートメント識別子 P は必須な値です。
2	空白
3-8	最初または唯一の SET PATCH 指定。この指定は必須です。
9	空白
10-15	2 番目の SET PATCH 指定。この指定はオプションです。
16	空白
17-22	3 番目の SET PATCH 指定。この指定はオプションです。
23	空白
24-29	4 番目の SET PATCH 指定。この指定はオプションです。
30	空白
31-36	5 番目の SET PATCH 指定。この指定はオプションです。
37	空白
38-43	6 番目の SET PATCH 指定。この指定はオプションです。
44	空白
45-50	7 番目の SET PATCH 指定。この指定はオプションです。
51	空白
52-57	8 番目の SET PATCH 指定。この指定はオプションです。
58	空白

表 7. データベース・イメージ・コピー 2 ユーティリティの SET PATCH 制御ステートメントの桁とパラメーターの説明 (続き)

位置	説明
59-64	9 番目の SET PATCH 指定。この指定はオプションです。
65	ブランク
66-71	10 番目の SET PATCH 指定。この指定はオプションです。

関連資料:

 z/OS: DFSMSdss パッチ・エリア

## DFSUDMT0 ユーティリティの実行

選択するオプションによっては、DFSUDMT0 ユーティリティの実行時にいくつかの操作上の考慮事項があります。

サブセクション:

- 『並行コピー機能』
- 197 ページの『高速複製機能』
- 198 ページの『複数データベース・データ・セット入力』
- 199 ページの『グループ名の指定』
- 199 ページの『複数のイメージ・コピーの単一出力データ・セット』
- 200 ページの『高速複製イメージ・コピーの動的データ・セット名』
- 200 ページの『イメージ・コピー完了通知』
- 203 ページの『DFSMSdss SET PATCH コマンドの指定』

### 並行コピー機能

並行コピー機能ではファジー・イメージ・コピーとクリーン・イメージ・コピーの両方を行うことができます。

データベース・イメージ・コピー 2 ユーティリティの DFSMS 並行コピー・オプションを使用する場合、DFSMS がデータの並行コピー・セッションを初期設定する間、データベースを使用できません。この時間は、完全なバックアップに必要な時間のごく一部です。並行コピー・セッションが確立されると、コピーは論理的に完了します。並行コピーの初期設定後に、DFSMS がデータを読み取って、出力コピーを作成している間に更新が再開される場合があります。作成されたコピーには、更新アクティビティの結果は何も反映されません。バックアップが要求されて即時に作成されたかのようになります。

イメージ・コピーが物理的に完了したら、これは、SMSNOCIC または SMSCIC イメージ・コピーのいずれかとして DBRC に登録されます。物理コピーの完了に必要な時間はデータ・セットのサイズによって変わります。

物理コピーはデータベース・リカバリー・ユーティリティ (DFSURDB0) が入力として使用します。物理コピー時間は DBRC がリカバリー用に JCL を生成するのに使用します。イメージ・コピーがクリーン・イメージ・コピー (SMSNOCIC) の場合、DBRC は必要な変更累積データ・セットを含み、イメージ・コピーの停止時



間後にデータベースに対する変更を適用するためにログに記録します。イメージ・コピーが並行イメージ・コピー (SMSCIC) の場合、DBRC は必要な変更累積データ・セットを含み、イメージ・コピーの開始時間後にデータベースに対する変更を適用するためにログに記録します。

DFSMS 並行コピー機能を使用することによって、データベース・イメージ・コピー 2 ユーティリティはデータベースの可用性を高めています。停止状態またはアクティブなデータベースをコピーできます。データベースが停止状態であれば、論理コピーが完了した後で再始動して、データベースの更新を継続できます。IMS では、物理コピーの完了を待たずにデータベースを使用できます。リカバリー不能データベースは、ユーティリティを実行する前に停止する必要がありますので注意してください。

また、更新のためにデータベースを解放する前に、物理コピーが完了するまで待つこともできます。これは、イメージ・コピーが特定の目的 (例えば、月末処理) に必要な場合に役立ちます。このような場合には、物理コピーを待ってからデータベースを再始動したほうが安全です。

並行コピー DBDS 選択ステートメントを指定することによって、データベース・イメージ・コピー 2 ユーティリティの並行コピー・オプションを選択できます。

## 高速複製機能

高速複製機能は、z/OS DFSMSdss コマンドの COPY FASTREP(REQ)、および FlashCopy 機能を持つエンタープライズ・ストレージ・サーバー (ESS) またはスナップショット機能を持つ RAMAC 仮想アレイ (RVA) によって提供される高速複製コピー技術を使用します。

高速複製機能は、イメージ・コピーの作成とリカバリー時のデータベース・データ・セット (DBDS) 復元のために高速複製技術を使用します。高速複製イメージ・コピーは、並行コピー・オプションのダンプ定様式イメージ・コピーとは対照的に、DBDS の正確なコピーです。

並行コピー・プロセスとは異なり、高速複製コピー・プロセスは論理コピー・フェーズと物理コピー・フィズに分かれていません。クリーン・イメージ・コピーを作成する場合、コピー・プロセスの間はデータベースを使用できません。ただし、高速複製プロセスは速度の点で、並行コピー・プロセスの論理コピー・フェーズに匹敵します。

イメージ・コピーが完了すると、データベース・イメージ・コピー 2 ユーティリティは、SMSOFFLC または SMSONLC イメージ・コピーとして DBRC に登録します。クリーン・イメージ・コピーを作成する場合、データベース・イメージ・コピー 2 はシステム・コンソールにイメージ・コピーの完了を通知します。ファジー・イメージ・コピーを作成する場合、通知は送信されません。

高速複製 DBDS 選択ステートメントを指定することによって、データベース・イメージ・コピー 2 ユーティリティの高速複製オプションを選択できます。

## 複数データベース・データ・セット入力

データベース・イメージ・コピー 2 ユーティリティでは、ユーティリティの 1 回の実行で複数の DBDS をコピーできます。コピーする DBDS ごとに、1 回の実行で複数の制御ステートメントを指定できます。

データベース・イメージ・コピー 2 ユーティリティは、1 回の呼び出しで、制御ステートメントに指定されたすべての DBDS を DFSMSdss コマンドに引き渡します。並行コピー・オプションを使用する場合、DFSMSdss は複数のダンプ・プロセスを並行に開始し、すべての DBDS について論理コピー・フェーズがきわめて短時間で完了します。複数の DBDS をコピーするのに高速複製オプションを使用する場合、DFSMSdss は複数のコピー・プロセスを並行に開始し、イメージ・コピー・プロセス全体が、複数のコピー要求を一度に処理するときよりも、素早く完了します。

**制約事項:** 同じテープ・ボリューム上にある出力データ・セットに複数の DBDS を並行してコピーすることはサポートされていません。この制限は、仮想テープ・ボリュームをインプリメントする IBM 仮想テープ・サーバーを使用することによって回避できます。または、並行コピー機能を使用する場合は、複数の入力データ・セットを 1 つの出力データ・セットに連結する同一データ・セット・オプションを使用できます。高速複製機能は、同一データ・セット・オプションをサポートしていません。

並行して処理できる DFSMSdss ダンプ・タスクまたはコピー・タスクの数は、使用できるリソースの量によって異なります。1 回の実行で複数のデータ・セットをコピーする場合、1 つの DBDS をコピーする場合に比べ、特定リソース (仮想ストレージ、テープ装置) がより多く必要となります。必要なだけのリソースが得られないと、他の DFSMSdss タスクが終了して必要量のリソースが使用できるようになるまで、いくつかの DFSMSdss タスクが後回しにされることがあります。

グループ名を指定して、グループとしてコピーされる DBDS のセットを識別することができます。グループ名を指定しない場合、同じユーティリティ実行でコピーされる DBDS ごとに別々の処理オプションを指定できます。これにより、ユーティリティの単一実行でコピーされる一部の DBDS イメージ・コピーがクリーン・イメージ・コピー (すなわち整合したイメージ・コピー) になり、同じ実行内でコピーされる他の DBDS のイメージ・コピーがファジー・イメージ・コピーになります。

並行コピー・オプションを使用して、同じユーティリティ実行内で KSDS のファジー・コピーを作成し、他の DBDS もコピーする場合は、KSDS だけをコピーするときに比べて、イメージ・コピー処理が失敗して DFS3145A メッセージが出力される可能性が高くなります。KSDS だけをコピーする場合、DFS3145A エラーによる失敗になるまでに、ユーティリティは DFSMSdss DUMP コマンドを何回か再試行します。しかし、同じ実行内で他の DBDS もコピーしているときは、再試行がありません。

高速複製オプションを使用して、他の DBDS も問題なく作成される同じユーティリティ実行で KSDS のファジー・イメージ・コピーを作成することができます。高速複製オプションを使用する場合、データ・セット・タイプは重要ではありません。

## グループ名の指定

グループ名ユーティリティー制御ステートメントを指定することによってデータベース・イメージ・コピー 2 ユーティリティーの 1 回の実行でコピーする DBDS の集合を示すために、グループ名を指定することができます。

グループ名機能では、グループ単位でクリーン・イメージ・コピー処理をモニターできます。グループ名を指定すると、状況が高速複製機能のイメージ・コピー・プロセス全体の完了と並行コピー機能の論理および物理フェーズの完了メッセージで報告されます。

グループ名は、DBRC グループ (DB グループ、DBDS グループ、または CA グループ) の名前にすることができます。ユーティリティーは、その名前を持つグループが RECON 内に存在するかを検査しませんし、存在する場合にそのグループのメンバーがユーティリティーによってコピーされるように指定されているかも検査しません。ただし、ユーティリティーのグループ名ステートメントで DB グループ名または DBDS グループ名を使用すると、操作が簡単になります。

グループ名ステートメントの後ろには、そのグループのメンバーである (または、メンバーとして扱うべき) 各 DBDS を指定する制御ステートメントが続きます。1 つの DB グループまたは DBDS グループに含まれるすべてのデータ・セットを 1 つの名前付きグループに含めておけば、その全部を 1 回のデータベース・イメージ・コピー 2 実行でコピーできます。クリーン・イメージ・コピーがとられるとき、いずれかの DB グループまたは DBDS グループでデータベース/区域を停止するには、/DBR DATAGROUP コマンドを使用できます。そのグループの適切な完了通知が提供された場合、/START DATAGROUP コマンドによってデータベースまたは区画を再開できます。

グループ・ステートメントを指定する場合、グループ・メンバーの処理オプションをグループ・レベルで指定します。グループに対する指定のオプションまたはデフォルトのオプションは、そのグループ中の DBDS ステートメントで個々に指定されたオプションに優先します。

## 複数のイメージ・コピーの単一出力データ・セット

DFSMSdss には、複数の入力データ・セットを 1 つの出力データ・セットに入力する機能があります。データベース・イメージ・コピー 2 ユーティリティーの並行コピー機能は、同一データ・セット・オプションを提供することによってこの機能を活用します。

同一データ・セット・オプションは、DFSMSdss ダンプ・ヘッダー・レコードが先頭に指定された複数のダンプを次々と単一出力データ・セットに書き込むことによって、1 回のユーティリティーの実行で、最大 255 の DBDS のイメージ・コピーをその出力データ・セットに連結します。RECON では、コピーされた各 DBDS のイメージ・コピー・レコードにこの出力データ・セットが記録されます。指定された DBDS が 255 個を超えていると、ユーティリティーの実行は中止されます。

最大 255 の DBDS インスタンスのイメージ・コピー出力を含む単一出力データ・セットは、同じテープ・ボリューム上の別々のデータ・セット内に複数のイメージ・コピー・インターフェースを生成する、JCL 指定で実現されるスタッキングと対比することができます。

同一データ・セット・オプションを使用したイメージ・コピーの連結により、テープ・メディアの使用効率が高まり、イメージ・コピーをとるために割り振られるテープ・ボリュームの数が減少します。

ただし、データベース・リカバリー・ユーティリティーを使ってリカバリーを行うと、スタックされたイメージ・コピーから各 DBDS を復元するために、テープ・ボリューム内での個別の読み取り移動が必要になるという欠点があります。また、連結によって 1 つのデータ・セットにする場合、物理的コピー処理が逐次化され、物理的完了時に通知をしてクリーン・コピーを作成する間の使用不可時間が長くなる可能性があります。

制約事項: 同一データ・セット・オプションは以下の項目によってサポートされません。

- データベース・イメージ・コピー 2 ユーティリティーの高速複製機能
- REUSE 属性を持つ RECON データ・セットに登録される DBDS

同一データ・セット・オプションをインプリメントするには、以下のステップに従います。

1. 指定する最初の DBDS 選択ステートメントでは、通常使用する出力データ・セットを定義します。
2. 後続の DBDS 選択ステートメントでは、
  - 列 1 に S を指定します。
  - 出力データ・セットは定義してはなりません。

## 高速複製イメージ・コピーの動的データ・セット名

高速複製機能を使用する場合、データベース・イメージ・コピー 2 ユーティリティーで、出力データ・セットのデータ・セット名と DD 名を動的に生成できます。高速複製 DBDS 選択ステートメントでの適切な指定のみ行い、高位修飾子を定義する HLQ 仕様ステートメントを含める必要があります。HLQ 仕様ステートメントを使用して、データベース・イメージ・コピー 2 ユーティリティーで、動的に生成した名前に低位修飾子としてタイム・スタンプを付加することができます。

SMS 管理 DASD を使用する場合、DFSMSdss では、正しい割り振り属性を持つデータ・セットを作成できます。DD ステートメントを含めたり、DBDS 制御ステートメントに OUTPUT DD 名を指定したりする必要はありません。

非 SMS 管理 DASD を使用する場合、ターゲット・ボリュームを指定するために DD ステートメントを指定する必要があります。DBDS 制御ステートメントに OUTPUT DD 名を指定する必要があります。DFSMSdss は、ソース・データベース・データ・セットから割り振り属性の残りを決めることができます。

データ・セット名と DD 名を手動で定義するように選択した場合、出力データ・セット名を JCL に指定し、適切な OUTPUT DD 名を DBDS 制御ステートメントに指定する必要があります。

## イメージ・コピー完了通知

クリーン・イメージ・コピーをとる場合、データベース・イメージ・コピー 2 ユーティリティーは、並行コピー機能の論理または物理フェーズがいつ終了するのか

と、高速複製機能のイメージ・コピー・プロセス全体がいつ終了するのかをシステム・コンソールに通知します。この通知は、データベースまたはグループが始動可能であることを示します。

データベース・イメージ・コピー 2 ユーティリティは、ファジー・イメージ・コピーをとる場合は完了通知を行いません。ファジー・イメージ・コピーが完了しても、データベースを可用性状況に戻すための処置は不要です。

#### 並行コピー機能の論理的完了通知

データベース・イメージ・コピー 2 ユーティリティは、並行コピー機能を使用して作成したイメージ・コピーが論理的に完了すると DFS3121I メッセージを発行します。DFSMSdss が並行コピーの初期設定フェーズを完了すると、そのイメージ・コピーは論理的に完了します。クリーン・イメージ・コピーがとられ、イメージ・コピーが物理的に完了しないうちにデータベースへの更新を再開する場合、このメッセージはデータベースがオンライン・システム上で開始できる状態になったことを示します。ユーザーが、データベースの開始を自動化できるように、メッセージはシステム・コンソールに送られます。

グループ・ステートメントを使用すると、そのグループ名に対して完了通知が出されます。グループ・ステートメントでは、どのような名前でも使用できますが、事前に定義されているグループ (CA グループ、DB グループ、DBDS グループなど) の名前と一致する名前を選択したほうが有用です。

- ユーティリティに対してグループ名を指定しておくと、そのグループに対して一度だけ DFS3121A メッセージが出されます。
- グループ名を指定しないときは、データベース、HALDB 区画、または区域ごとに一度このメッセージが出されます。

DFS3121A メッセージが論理的完了時に出されるのは、クリーン・イメージ・コピーがとられており、コピーが論理的に完了したら更新が許可されることになっている (ユーティリティ制御ステートメントで XL が指定された) 場合のみです。このメッセージに続いていくつかの DFS3121I メッセージが出され、コピーするグループまたはデータベースについて DBDS を示します (1 つの区域に 1 つの DFS3121I メッセージが出され、コピーするエリア・データ・セットを示します)。DFS3121A メッセージはシステム・コンソールと SYSPRINT データ・セットに送られます。DFS3121I メッセージは SYSPRINT データ・セットだけに送られます。

この例では、グループ・ステートメントを指定したときに SYSPRINT に出される論理的完了のメッセージを示します。

```
DFS3121A LOGICAL COPY COMPLETE FOR GROUP groupname; 0 OF 5 DATA SETS FAILED
DFS3121I COPIED DB/AREA dbnameA DDN dnameA1 DSN dsnameA1
DFS3121I COPIED DB/AREA dbnameA DDN dnameA2 DSN dsnameA2
DFS3121I COPIED DB/AREA dbnameA DDN dnameA3 DSN dsnameA3
DFS3121I COPIED DB/AREA dbnameB DDN dnameB1 DSN dsnameB1
DFS3121I COPIED DB/AREA dbnameC DDN dnameC1 DSN dsnameC1
```

この例では、グループ・ステートメントを指定しないときに出される論理的完了のメッセージを示します。

```
DFS3121A LOGICAL COPY COMPLETE FOR DB/AREA dbnameA; 0 OF 3 DATA SETS FAILED
DFS3121I COPIED DB/AREA dbnameA DDN ddnameA1 DSN dsnameA1
DFS3121I COPIED DB/AREA dbnameA DDN ddnameA2 DSN dsnameA2
DFS3121I COPIED DB/AREA dbnameA DDN ddnameA3 DSN dsnameA3
DFS3121A LOGICAL COPY COMPLETE FOR DB/AREA dbnameB; 0 OF 1 DATA SETS FAILED
DFS3121I COPIED DB/AREA dbnameB DDN ddnameB1 DSN dsnameB1
DFS3121A LOGICAL COPY COMPLETE FOR DB/AREA dbnameC; 0 OF 1 DATA SETS FAILED
DFS3121I COPIED DB/AREA dbnameC DDN ddnameC1 DSN dsnameC1
```

グループ、データベース、HALDB 区画、または区域について選択されているすべての DBDS オカレンスで論理コピーが完了したときに、データベース許可がデータベース・イメージ・コピー 2 によってリリースされます。これにより、アプリケーション処理が更新アクティビティを再開することができます。

論理コピー・プロセスがいずれかの DBDS に対して失敗すると、DFS3121A メッセージは失敗した DBDS を示します。DBDS のイメージ・コピーが失敗したことが DFS3121A によって示される場合は、DBDS のコピーを再度試行してからグループまたはデータベースを開始することを検討できます。論理コピー・フェーズで失敗した DBDS ごとに、メッセージ DFS3122A も発行されます。

#### 並行コピー機能の物理的完了通知

クリーン・イメージ・コピーを作成するために並行コピー機能を使用し、物理コピー・フェーズが完了するまでデータベースの更新を許可しない場合、イメージ・コピーが物理的に完了すると、データベース・イメージ・コピー 2 ユーティリティはシステム・コンソールにメッセージ DFS3141A を発行します。このメッセージは、イメージ・コピーが RECON に記録されて、データベースまたは区域が開始できる状態になったことを示します。

DFS3141A は、グループ名が指定されていると、DFS3121A と同様にそのグループ名で報告を行います。それ以外の場合は、処理されたデータベース、HALDB 区画、および区域ごとに DFS3141A メッセージが出されます。DFS3141 メッセージを見れば、どのデータ・セットが正常に処理されたかが分かります。

SYSPRINT 出力の例を次に示します。

```
DFS3141A PHYSICAL COPY COMPLETE FOR GROUP groupname; 0 OF 4 DATA SETS FAILED
DFS3141I COPIED DB/AREA dbname1 DDN ddname1 DSN dsname1
DFS3141I COPIED DB/AREA dbname1 DDN ddname2 DSN dsname2
DFS3141I COPIED DB/AREA dbname3 DDN ddname3 DSN dsname3
DFS3141I COPIED DB/AREA dbname4 DDN ddname4 DSN dsname4
```

DFS3141A メッセージは、システム・コンソールと SYSPRINT に送られます。DFS3141I メッセージは、SYSPRINT だけに送られます。グループ、データベース、HALDB 区画、または区域について選択しているすべての DBDS オカレンスで物理コピーが完了すると、許可がリリースされます。

物理コピー・プロセスがいずれかの DBDS に対して失敗すると、DFS3141A メッセージは失敗した DBDS を示します。DBDS のイメージ・コピーが失敗したことが DFS3141A によって示される場合は、DBDS のコピーを再度試行してからグループまたはデータベースを開始することを検討できます。物理コピー・フェーズで失敗した DBDS ごとに、メッセージ DFS3144A も発行されます。

#### 高速複製機能のイメージ・コピー完了通知

データベース・イメージ・コピー 2 ユーティリティは、高速複製機能を使用して作成したクリーン・イメージ・コピーが完了すると DFS3141A メッセージを発行します。このメッセージは、イメージ・コピーが RECON に記録されて、データベースまたは区域が開始できる状態になったことを示します。DFS3141A は、グループ名が指定されているとそのグループ名で報告を行います。それ以外の場合、データベース・イメージ・コピー 2 ユーティリティは、処理されたデータベース、HALDB 区画、および区域ごとに DFS3141A メッセージを発行します。DFS3141A メッセージは、正常に処理された個々のデータ・セットを示します。

SYSPRINT 出力の例を次に示します。

```
DFS3141A PHYSICAL COPY COMPLETE FOR GROUP groupname; 0 OF 4 DATA SETS FAILED
DFS3141I COPIED DB/AREA dbname1 DDN ddname1 DSN dsname1
DFS3141I COPIED DB/AREA dbname1 DDN ddname2 DSN dsname2
DFS3141I COPIED DB/AREA dbname3 DDN ddname3 DSN dsname3
DFS3141I COPIED DB/AREA dbname4 DDN ddname4 DSN dsname4
```

DFS3141A メッセージは、システム・コンソールと SYSPRINT に送られます。DFS3141I メッセージは、SYSPRINT だけに送られます。グループ、データベース、HALDB 区画、または区域について選択されているすべての DBDS オカレンスで高速複製イメージ・コピーが完了すると、許可がリリースされます。

高速複製コピー・プロセスがいずれかの DBDS に対して失敗すると、DFS3141A メッセージは失敗した DBDS を示します。DBDS のイメージ・コピーが失敗したことが DFS3141A によって示される場合は、DBDS のコピーを再度試行してからグループまたはデータベースを開始することを検討できます。物理コピー・フェーズで失敗した DBDS ごとに、メッセージ DFS3144A も発行されます。

## DFSMSdss SET PATCH コマンドの指定

このトピックにはプロダクト・センシティブ・プログラミング・インターフェース情報が含まれています。

DFSMSdss は、特定のオフセットでパッチ・バイトの値を設定することによって、DFSMSdss 処理をカスタマイズできるようにするパッチ・エリアを提供します。データベース・イメージ・コピー 2 ユーティリティは、DFSMSdss パッチ・バイトを一時的に設定するための 2 つの方法を提供します。

### SET PATCH 制御ステートメント

SET PATCH 制御ステートメントを使用することによって、データベース・イメージ・コピー 2 ユーティリティの現在の実行でコピーする DFSMSdss 処理オプションを指定できます。制御ステートメントでは、DFSMSdss SET PATCH コマンドのオフセットおよび値をデータベース・イメージ・コピー 2 ユーティリティに指定することができます。

各 SET PATCH 制御ステートメントには、最大 10 の SET PATCH 指定を含めることができます。複数の SET PATCH 制御ステートメントを含めることができます。

モジュール DFSUDMT2 内の SET PATCH コマンド

データベース・イメージ・コピー 2 ユーティリティーは、ザップできる総称 SET PATCH コマンド (SET PATCH 00=00) をモジュール DFSUDMT2 内に定義し、DFSMSDss commands DUMP または COPY とともに DFSMSDss に受け渡すことができます。その場合、パッチは、イメージ・コピー 2 のジョブ・ステップについてのみ有効で、他の DFSMSDss 処理には無効です。

ユーティリティーが SET PATCH コマンドを DFSMSDss に渡せるように、DFSUDMT2 内で ZAP を適用して SET PATCH 00=00 文字ストリングの 1 つでパッチ・バイトとパッチの値を必要な値 (長さ 2 の文字ストリング)、例えば SET PATCH 44=FF に置き換えることができます。モジュール内の 1 つ以上の SET PATCH コマンドの変更ができます。ユーティリティーは DFSMSDss に、パッチ・バイト文字ストリングが 00 でない SET PATCH コマンドを渡します。ユーティリティーは、コマンドを DFSMSDss に渡す前に、コマンドの構文を検査したり、パッチ・バイトまたはパッチ値の妥当性を検査したりしないことに注意してください。

インストールによって使用が制限されている場合は、PATCH キーワード付きで DFSMSDss SET コマンドを発行する許可が与えられているかを確認する必要があります。

関連資料:

 z/OS: DFSMSDss パッチ・エリア



---

## 第 14 章 オンライン・データベース・イメージ・コピー・ユーティリティー (DFSUICP0)

オンライン・データベース・イメージ・コピー・ユーティリティー (DFSUICP0) は、オンライン・システムがデータベースの更新を行っているときに、データベースの現状のままのイメージ・コピーを作成するために使用します。イメージ・コピーは、リカバリーの目的で使用されます。

DFSUICP0 ユーティリティーは、その始動時および完了時に、システム・ログのタイム・スタンプを印刷します。

出力イメージ・コピー上の最初のレコードは、ダンプ・ヘッダー・レコードです。これには、データ ID、作成日、作成時刻といった情報が含まれます。作成日と作成時刻は、後で入力を検証するためにデータベース・リカバリー・ユーティリティー (DFSURDB0) で必要になるものです。

DFSUICP0 ユーティリティーは、出力イメージ・コピーの 1 つまたは 2 つのコピーを作成できます。コピーを 2 つ指定すると、コピーの実行中に入出力エラーが起こっても、ユーティリティーは他方のコピーで最後まで続行するという利点が得られます。パフォーマンスがいくぶん低下しますが、実行全体をやり直す必要はありません。

DFSUICP0 ユーティリティーは、システムまたはユーティリティーで障害が発生した後に、最後のユーティリティー・チェックポイントからイメージ・コピー・ジョブを再始動するための特別なユーティリティー・チェックポイントをユーティリティーの実行時にとることができます。ユーティリティー・チェックポイント機能は、DFSUCKPT および DFSURSRT の DD ステートメントを含めて、DISP=KEEP または CATLG と、固有でない通し番号を使用して出力イメージ・コピー・データ・セットを定義する DD ステートメントをコーディングすることによって有効になります。

このユーティリティーは、バッチ・メッセージ処理プログラム (BMP) として実行されます。

サブセクション:

- 206 ページの『制約事項』
- 206 ページの『前提条件』
- 206 ページの『要件』
- 207 ページの『推奨事項』
- 207 ページの『入出力』
- 208 ページの『戻りコード』
- 208 ページの『JCL 指定』

## 制約事項

DFSUICP0 ユーティリティを実行するときには、次のような制約があります。

- HSAM と GSAM データベースはコピーできません。
- MSDB と DEDB はコピーできません。
- VSAM KSDS の索引部分は、KSDS 全体をコピーしないとコピーできません。
- データ共用環境では、複数の IMS サブシステムが更新アクセスのためにデータベースをオープンしている場合に、DFSUICP0 ユーティリティを使用することができません。別の方法として、データベース・イメージ・コピー・ユーティリティ (DFSUDMP0) の並行コピー・オプションを DFSUICP0 ユーティリティの代わりに使用できます。
- ユーティリティ・チェックポイント再始動機能は、電源障害または DASD のスペース不足が原因でジョブが失敗すると、QSAM レコードが QSAM バッファから書き込まれないため、イメージ・コピー・ジョブを再始動できません。そのようなことが起こったときは、DBRC コマンドの CHANGE.SUBSYS と DELETE.SUBSYS を使用して、オンライン・イメージ・コピー SSID を RECON データ・セットから削除しなければなりません。古い SSID を RECON から削除した後で、別のオンライン・イメージ・コピー・ジョブの実行依頼ができます。
- イメージ・コピー・ユーティリティを HALDB オンライン再編成と並行して実行することはできません。
- オンライン・イメージ・コピー・ユーティリティは、電源障害または DASD のスペース不足が原因で停止したときは、QSAM レコードが QSAM バッファから書き込まれていないため、再始動させることはできません。そのようなことが起こったときは、DBRC コマンドの CHANGE.SUBSYS と DELETE.SUBSYS を使用して、オンライン・イメージ・コピー SSID を RECON データ・セットから削除しなければなりません。古い SSID を RECON から削除した後で、別のオンライン・イメージ・コピー・ジョブの実行依頼ができます。
- データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

## 前提条件

リカバリー不能データベースのイメージ・コピーをとるときは、DFSUICP0 ユーティリティを実行する前に、データベース・アクセスを READ または READ ONLY 状況に設定する必要があります。これを行うために、マスター端末オペレーター (MTO) は /DBD コマンドを出すことができます。このユーティリティは、データベースへの並行更新が可能になっている場合は、失敗します。

## 要件

オンライン・データベース・イメージ・コピー・ユーティリティを実行するときには、次のような要件があります。

- このユーティリティには、コピーするデータベースを指定し、OLIC=YES オペランドを含んでいる PSB が必要です。この PSB には、1 つ以上の PCB を含

めることができますが、オンライン・システム定義の中で APPLCTN マクロで定義されていないならばなりません。The PSBGEN LANG= keyword must not specify PL/I.

コピーするデータ・セットが 2 次データ・セット・グループの OSAM データ・セットであって、ダンプ処理を高速化するために順次バッファリングが使用されているときは、PSB には、その 2 次データ・セット・グループのセグメントに関する SENSEG が少なくとも 1 つ含まれていなければなりません。

- デフォルトのブロック・サイズを JCL に指定しない場合は実行時に計算されるので、オンライン・データベース・イメージ・コピー・ユーティリティが作成するすべての出力コピーに標準ラベルを使用する必要があります。
- アプリケーション・プログラムがデータベースを更新しているときに DFSUICP0 ユーティリティを使用してそのデータベースのイメージ・コピーを作成する場合、そのプログラムによる変更をデータベースのリカバリー時に適用しなければなりません。変更をログから適用するには、データベース・リカバリー・ユーティリティ (DFSURDB0) を使用します。リカバリーのために必要な最初のログ・データ・セットのタイム・スタンプは、SYSOUT に印刷されます。

## 推奨事項

DFSUICP0 ユーティリティを使用するときは、次のような推奨事項があります。

- ログしなくてデータベースを更新するバッチ・ジョブを実行した直後にイメージ・コピーをとってください。イメージ・コピーでは、リカバリーが必要な場合にデータベースの保全性が維持されます。ログ・テープを使用すると、バッチ・ジョブの開始までリカバリーできるため、そこでバッチ・ジョブを再処理することができます。ただし、その結果のデータベースは、前のバッチ実行後のデータベースと論理的には同一であっても、同一でない場合があります。バッチ・ジョブは反復可能でない場合があるため、そのことを前提としてください。データベースが同一でないと、前のバッチ・ジョブの実行後に作られたログ・テープは、再理後は有効でなくなります。したがって、リカバリー処理は、イメージ・コピーで開始し、ログ・テープを適用し、ログがとられなかったバッチ実行を再処理し、その後で別のログ・テープを適用するという順序で行うことはできません。

## 入出力

DFSUICP0 ユーティリティへの 1 次入力、コピーするデータベース・データ・セットです。DFSUICP0 ユーティリティの 1 次出力は、データベース・リカバリー・ユーティリティ (DFSURDB0) への入力として使用できるデータベース・データ・セットのイメージ・コピーです。

DFSUICP0 ユーティリティの入出力は、以下の表に示すとおりです。

表 8. DFSUICP0 ユーティリティの入出力

入力	出力
RECON	SYSPRINT メッセージと統計
DBD ライブラリー	イメージ・コピー・データ・セット

コピーする 1 つ以上のデータベース・データ・セット

表 8. DFSUICP0 ユーティリティの入出力 (続き)

入力	出力
ユーティリティ制御ステートメント	

## 戻りコード

DFSUICP0 ユーティリティは以下の戻りコードを発行します。

コード 意味

- 0 すべての操作が正常に完了した
- 4 警告メッセージが出された
- 8 1 つ以上の操作が正常に完了しなかった
- 12 再始動時にエラーが起こった
- 16 すべての操作を完了しないうちに、ジョブを終了させる重大エラーが起こった

これらの戻りコードは、後続のジョブ・ステップの EXEC ステートメントで COND= パラメーターを指定すると、テストすることができます。

ユーティリティの実行中にチェックポイント・データ・セットでエラーを受け取ると、メッセージが印刷されますが、ユーティリティは続行します。それ以降のユーティリティ・チェックポイントはとられません。

再始動時に再始動データ・セットでエラーを受け取ると、メッセージが印刷され、ユーティリティは異常終了します。

## JCL 指定

DFSUICP0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。DFSUICP0 ユーティリティは、標準の z/OS ジョブとして実行されます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含める必要があります。

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

コピーするデータ・セットの DD ステートメントは、制御領域のジョブ制御言語データ・セットにあり、コピー・ジョブ自体にはありません。

### EXEC ステートメント

このステートメントは必要なステートメントが入っているカタログ式プロシージャを呼び出すことができますが、次の形式にすることもできます。

```
▶▶EXEC—PGM=DFSRRCO0,PARM=—'BMP,DFSUICP0,psbname,,destname'—▶▶
└─,IMSPLEX=plexname┘└─,DBRCGRP=xxx┘▶▶
```

## **BMP および DFSUICP0**

ユーティリティ領域を定義します。

### **psbname**

オンライン・システム定義の中で APPLCTN マクロによって記述され、コピーするデータベースを指定しており、OLIC=YES パラメーターを含んでいる PSB の名前です。

### **destname**

クリティカル・エラー・メッセージの出力先です (デフォルト宛先は z/OS コンソールです)。

### **IMSPLEX**

DBRC が結合する IMSplex を指定します。

### **DBRCGRP**

3 文字の DBRC グループ ID を左寄せで指定し、必要に応じて空白で埋めます。DBRC グループ ID は、DBRC SCI 登録出口ルーチン DSPSCIX0 に渡されます (DSPSCIX0 が存在する場合)。IMS に付属の DSPSCIX0 のサンプル版は、DBRC グループ ID としてユーザーが DBRC に与えた値を戻します。このパラメーターはオプションであり、DBRCGRP 実行パラメーターによってオーバーライドされることがあります。

推奨事項: DBRC グループ ID を判別するには、DBRCGRP パラメーターの代わりに DBRC SCI 登録出口 DSPSCIX0 を使用します。

通常の IMS 定位置パラメーターを続けて置くことができます。

### **DD ステートメント**

DFSUICP0 ユーティリティは、必須またはオプションの DD ステートメントをいくつか使用します。

#### **STEPLIB DD**

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

#### **DFSRESLB DD**

IMS SVC モジュールが入っている許可ライブラリーを指します。

#### **IMS DD**

ダンプするデータベースを記述する DBD が入っているライブラリーを定義します。これは DSNNAME=IMS.DBDLIB であるのが普通です。このデータ・セットは、直接アクセス・ボリュームに置いておかなければなりません。

#### **SYSPRINT DD**

出力メッセージ・データ・セットを定義します。このデータ・セットは、テープ、直接アクセス・ボリューム、またはプリンターに置いておくことも、出力ストリーム (SYSOUT) に入れて送ることもできます。

#### **SYSIN DD**

入力制御ステートメント・データ・セットを定義します。このデータ・セット

は、テープまたは直接アクセス・ボリュームに置いておくことも、入力ストリーム (DD \* または DD DATA) に入れて送ることもできます。

#### **dataout1 DD**

ダンプされた出力データ・セットの最初のコピーを定義します。DD ステートメントは、ダンプするデータ・セットごとに 1 つ必要です。DD 名には、1 文字から 8 文字までの任意の文字ストリングを使えますが、その DD 名は関連のユーティリティー制御ステートメントにも指定されている必要があります。出力装置は直接アクセス装置またはテープのいずれかでなければなりません。標準ラベルを使用しなければなりません。

ユーザー・ブロック・サイズは、DCB オペランドの BLKSIZE サブパラメーターで指定できます。ユーティリティーは、端数切り捨てによってブロック・サイズをデータベース・データ・セットの論理レコード・サイズの偶数倍数に丸めたものに 8 を加え、この値を最も近いダブルワードに丸めます。指定するブロック・サイズは論理レコード長より大きくし、装置の最大値に等しいかそれより小さくしなければなりません。ブロック・サイズの指定がないと、装置の最大ブロック・サイズを端数切り捨てによって LRCL の偶数倍数に丸めた値が使用されます。

例外: 装置が 3380 である場合、論理レコード長が 23 KB 以下であれば、ブロック・サイズとして 23 KB が使用されます。論理レコード長が 23 KB より大きい場合、ブロック・サイズは、32 KB (端数切り捨てによって論理レコード長の偶数倍数に丸めた値) となります。

制約事項: 論理レコード長は DCB オペランドで指定できません。

再始動機能が使用されるときは、イメージ・データ・セットの後処理が KEEP または CATLG になっていることを確かめてください。

#### **dataout2 DD**

これが必要になるのは、関連のユーティリティー制御ステートメントがダンプを 2 部コピーするよう要求している場合だけです。dataout2 DD に要求される条件は、dataout1 と同じです。dataout2 に指定するブロック・サイズは、dataout1 に指定したものと異なっていて構いません。

#### **DFSUCKPT DD**

ユーティリティーの実行に固有のユーティリティー・チェックポイント情報をこのユーティリティーが書き込む、オプションのチェックポイント・データ・セットを定義します。直接アクセス装置上の単一トラックが必要です。データ・セットの特性は、このユーティリティーが指定します。

#### **DFSURSRT DD**

ジョブの再始動のときに前のユーティリティー・チェックポイントを使用することを示している、オプションの再始動データ・セットを定義します。

#### **RECON1 DD**

最初の DBRC RECON データ・セットを定義します。この RECON1 データ・セットは、制御領域が使用しているものと同じ RECON1 データ・セットでなければなりません。

### RECON2 DD

2 番目の DBRC RECON データ・セットを定義します。この RECON2 データ・セットは、制御領域が使用しているものと同じ RECON2 データ・セットでなければなりません。

### RECON3 DD

3 番目の DBRC RECON データ・セットを定義します。この RECON3 データ・セットは、制御領域が使用しているものと同じ RECON3 データ・セットでなければなりません。


動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

### DFSCTL DD


順次バッファリング (SB) の活動化を要求する SBPARM 制御ステートメントが入っているデータ・セットを記述します。SB を条件付きで活動化させると、OSAM DB データ・セットのバッファリングのパフォーマンスが向上し、ジョブ時間が短縮されます。


DFSCTL ファイルは順次データ・セットにすることも、PDS のメンバーにすることもできます。レコード形式は F、FB、または FBS のいずれかで、レコード長は 80 でなければなりません。このデータ・セットは、直接アクセス装置またはテープに置いておくことも、入力ストリームに入れて送ることもできます。この DD ステートメントはオプションです。


関連概念:

 データベース・バックアップ・コピーの作成 (データベース管理)

関連資料:

 IMSBATCH プロシージャ (システム定義)

 順次バッファリング制御ステートメント (システム定義)

 プログラム仕様ブロック (PSB) 生成ユーティリティ (システム・ユーティリティ)

---

## DFSUICP0 ユーティリティの制御ステートメント

DFSUICP0 ユーティリティでは、ユーティリティ制御ステートメントを含める必要があります。

オンライン・データベース・イメージ・コピー・ユーティリティのユーティリティ制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

1 ステートメント ID

これは文字「D」でなければなりません。D は、このステートメントがオンライン・データベース・イメージ・コピー・ユーティリティの制御ステートメントであることを示します。

2 コピー数

これは、必要とするコピー数に応じて、1 か 2 にしなければなりません。

3 これは空白にしておきます。

**4-11 dbdname**

これは、ダンプするデータ・セットの名前が入っている物理 DBD の名前にしなければなりません。

**13-20 入力**の DD 名

これは、ダンプする入力データ・セットの DD 名でなければなりません。これは参照される DBD 内になければなりません。対応する DD ステートメントを DFSUICP0 に渡す必要はありません。

**22-29 出力**の DD 名

これは、1 次出力データ・セットの DD 名でなければなりません。対応する DD ステートメントが指定されていないなければなりません。

制約事項: 出力データ・セットは、コピーするデータベースの DD 名と同じ名前にすることができません。

**31-38 コピー**の DD 名

これは、ダンプされたデータ・セットの 2 番目のコピーの DD 名にしなければなりません。2 桁目が 1 のときは、このフィールドは空白のままにしておきます。空白にしない場合は、対応する DD ステートメントを指定する必要があります。

**40-43 チェックポイント**

これは、4 桁の数字で、ユーティリティー・チェックポイント間隔を指定します。このフィールドが検査されるのは、このユーティリティーの実行に使用される JCL に DFSUCKPT DD ステートメントが含まれている場合だけです。このフィールドを空白、非数字、またはゼロにしておく、デフォルトとして 5000 が使用されます (非数字かゼロのときは、フィールド形式が無効であるという警告メッセージが出されます)。

**44-80** コメントは、44 桁目から 80 桁目に入れることができます。

---

## DFSUICP0 ユーティリティーの例

これは、DFSUICP0 ユーティリティーの使用法の例です。

この節にある例では、桁合わせを容易にするために、SYSIN ステートメントの上に次のようなコメント行が示されています。

```
/** +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

このコメント行は単なる参考用です。

この節の例で、動的割り振りなしで DBRC を使用する場合は、以下の表に示す DD ステートメントをサンプル JCL に追加する必要があります。

図 56. 動的割り振りなしで DBRC を使用するための DD ステートメント



```
//RECON1 DD DSNAME=RECON1,DISP=SHR
//RECON2 DD DSNAME=RECON2,DISP=SHR
//RECON3 DD DSNAME=RECON3,DISP=SHR
```

この例は、HIDAM データベースに関連する 4 つの OSAM データ・セット・グループをユーティリティー・チェックポイントをとってコピーするために使用される JCL とユーティリティー制御ステートメントを示しています。

```
/*
//OLIC EXEC PGM=DFSRR00,PARM='BMP,DFSUICP0,HHTASK41'
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//SYSPRINT DD SYSOUT=A
//IMS DD DSNAME=IMS.DBDLIB,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//DFSUCKPT DD DSNAME=OLIC2.CKPT2,DISP=(NEW,KEEP),
// UNIT=SYSDA,VOL=SER=IMSQAW,SPACE=(TRK,(1,1))
//DMP1 DD DSNAME=OLIC2.IMAG1.OSAM,DISP=(NEW,KEEP),
// UNIT=TAPE,LABEL=(1,SL)
//DMP2 DD DSNAME=OLIC2.IMAG2.OSAM,DISP=(NEW,KEEP),
// UNIT=TAPE,LABEL=(1,SL)
//DMP3 DD DSNAME=OLIC2.IMAG3.OSAM,DISP=(NEW,KEEP),
// UNIT=TAPE,LABEL=(1,SL)
//DMP4 DD DSNAME=OLIC2.IMAG4.OSAM,DISP=(NEW,KEEP),
// UNIT=TAPE,LABEL=(1,SL)
/* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

//SYSIN	DD *			
D1	DH41SK01	DHSK0101	DMP1	1000
D1	DH41SK01	DHSK0102	DMP2	1000
D1	DH41SK01	DHSK0103	DMP3	2000
D1	DH41SK01	DHSK0104	DMP4	1000

```
/*
//DFSCCTL DD *
SBPARM ACTIV=COND
/*
```

## DFSUICP0 の障害後のイメージ・コピー・ジョブの再始動

DFSUICP0 の障害後のイメージ・コピー・ジョブの再始動について説明します。

2 つのオプション DD ステートメントを使用すると、システムまたはユーティリティーに障害が起こった後で、イメージ・コピー・ジョブを再始動することができます。これらのステートメントがオンライン・データベース・イメージ・コピー・ユーティリティーの JCL に含まれていないと、ユーティリティー・チェックポイント機能も再始動機能も使用できません。

DFSUCKPT DD ステートメントは、ユーティリティーの実行に固有のユーティリティー・チェックポイント情報をこのユーティリティーが書き込むデータ・セットを定義します。このユーティリティー・チェックポイント情報には、ボリューム通し番号や相対レコード番号があります。DFSUCKPT DD ステートメントが含まれていると、デフォルトによって、オンライン・データベース・イメージ・コピー・ユーティリティーは、5000 個のレコードがコピーされるたびにユーティリティー・チェックポイントを書き込みます。ユーティリティー・チェックポイント間隔は、制御ステートメントで別の間隔を指定することによって変更できます。

DFSURSRT DD ステートメントは、ジョブを再始動するとき使用するユーティリティー・チェックポイント・データ・セットを定義します。DFSUCKPT と DFSURSRT は、同じデータ・セットを定義することができます。

再始動機能を使用するには、イメージ・コピー・データ・セットを定義する DD ステートメントに、DISP=KEEP または CATLG と、非特定ボリューム通し番号が指定されている必要があります。後処理として KEEP または CATLG を指定しておく、再始動させる場合に、ボリュームが正しく巻き戻されます。ボリューム通し番号を特定しないで指定しておく、オペレーターは、再始動時に複数のボリュームを順序に関係なくマウントすることができます。

関連資料:

211 ページの『DFSUICP0 ユーティリティーの制御ステートメント』

---

## 第 3 部 リカバリー・ユーティリティー

リカバリー・ユーティリティーは、データベース、高速機能 DEDB 区域、および HALDB 区画のリカバリーを行うために使用します。

各トピックでは、1 つのユーティリティーを紹介し、その作動方法を説明し、その使用のための要件および制約事項を定義し、例を提供します。



---

## 第 15 章 バッチ・バックアウト・ユーティリティー (DFSBBO00)

バッチ・バックアウト・ユーティリティー (DFSBBO00) は、プログラムが開始される前の箇所まで、あるいはチェックポイントまたは同期点までデータベースをリカバリーするために使用します。

DFSBBO00 ユーティリティーは、バックアウトするエラーのあったプログラムの PSB と、そのプログラムが行った更新のログを使用して、通常の IMS バッチ・ジョブとして動作します。

DFSBBO00 ユーティリティーが役立つかどうかは、発生したエラーの種類、IMS システムの構成、およびリカバリーを行うときのステップが正しい順序であるかどうかによって左右されます。

DFSBBO00 ユーティリティーは、DBRC からの情報を使用して入力ログが有効かどうかを検査します。DFSBBO00 ユーティリティーは、既に完了しているバックアウトや進行中の再始動を使用して行われるバックアウトの実行を回避します。

DFSBBO00 ユーティリティーは、リモート・サイト・リカバリー (RSR) 構成のアクティブ・サイトのデータベースで実行できます。DFSBBO00 ユーティリティーは、トラッキング・サイトでは使用できません。

以下の図は、DFSBBO00 ユーティリティーのフロー・ダイアグラムです。

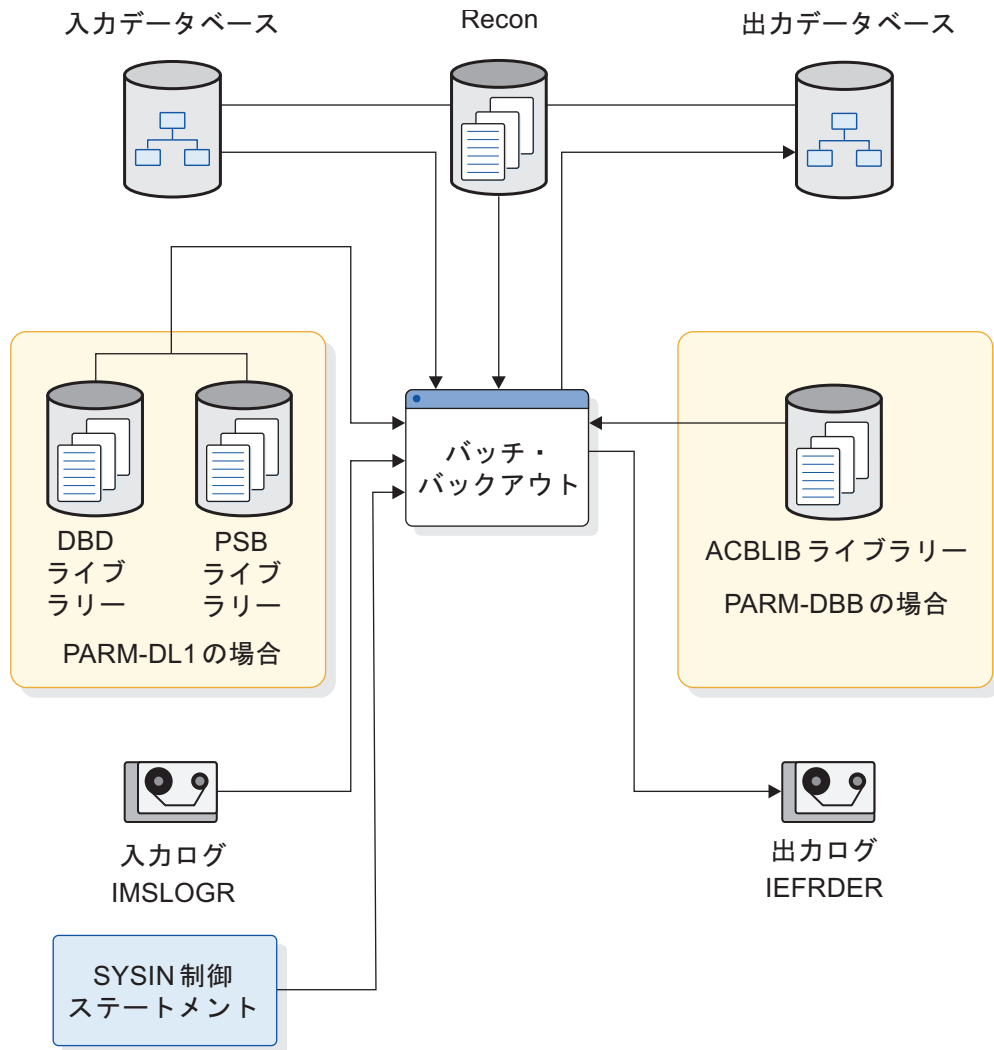


図 57. DFSBBO00 ユーティリティで必要なデータ・セット

サブセクション:

- 『制約事項』
- 219 ページの『前提条件』
- 219 ページの『要件』
- 220 ページの『推奨事項』
- 220 ページの『入出力』
- 222 ページの『戻りコード』
- 224 ページの『JCL 指定』

### 制約事項

このユーティリティを使用するときは、次のような制約があります。

- DFSBBO00 ユーティリティは、他のアプリケーションがデータベース更新にアクセスできるようになっていると、そのデータベース更新をバックアウトしません。オンライン・データベースの場合、更新を行った MPP、BMP、または IFP が同期点までくると、直ちに他のアプリケーションでその更新が使用できるよう

になります。IRLM を使用するバッチ・ジョブからチェックポイントが出された場合、そのバッチ・ジョブが行った更新はいずれも、他の IMS システムに置かれたアプリケーションで使用できるようになります。これらの場合では、最後の同期点以後に行われた更新のみをバックアウトできます。DFSBB000 ユーティリティーは、IRLM を使用しないバッチ・ジョブの場合には、指定されたチェックポイントまで更新をバックアウトします。このオプションの使用は、他のアプリケーションがデータベース更新にアクセスできないことを確かめてからにしてください。

- 正常終了したバッチ・ジョブが IRLM を使用していた場合、そのジョブはバックアウトできません。
- 使用するログは、再編成の前に作成されてはなりません。
- このユーティリティーは、入力ログが IMS バッチ・ジョブによって作成されたものでなければ、リカバリー不能なデータベースをバックアウトしません (そのログを保存した結果も同じです)。
- DFSBB000 ユーティリティーは、RSR トラッキング・サイトでは使用できません。
- DFSBB000 ユーティリティーは、全機能 (HALDB および非 HALDB) データベースのみで機能します。
- データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。

## 前提条件

DFSBB000 ユーティリティーを実行する前に、特定の前提条件が満たされていることを確認する必要があります。

このユーティリティーを使用するときは、次のような制約があります。

- DBRC に登録されていないデータベースで書き込みエラーが発生した場合、DFSBB000 ユーティリティーを実行する前に、そのデータベースをリカバリーする必要があります。前もってリカバリーをしないままバックアウトを試みた場合は、結果を予測できません。DBRC がなければ、リカバリーが正常に完了したことを検証できません。

データベースが DBRC に登録されている場合は、バックアウトより前にリカバリーを実行する必要があるのは、入出力エラーが原因でバックアウトが途中で失敗した場合だけです。バックアウトが正常に完了した場合は、リカバリーをもっと適切な時期まで延期できます。

## 要件

DFSBB000 ユーティリティーを実行するには、さまざまな要件を満たす必要があります。

このユーティリティーを使用するときは、次のような要件があります。

- DFSBB000 ユーティリティーのリリース・レベルは、入力ログ・データ・セットを作成した IMS バッチ領域コントローラー DFSRR000 のリリース・レベルと一致する必要があります。

- 入力として使用しているログを作成した IMS 領域が DBRC と IRLM を使用しているときは、DFSBB00 ユーティリティも DBRC と IRLM を使用しなければなりません。
- 高速機能データベースを参照する PSB で DFSBB00 ユーティリティを実行するときは、DBB 領域を指定しておき、IMS 管理の ACB を使用可能にしない場合はオンライン・アプリケーションで使用されたものと同じ ACBLIB を使用しなければなりません。
- このユーティリティは、入力ログが IMS バッチ・ジョブによって作成されたものでなければ、リカバリー不能なデータベースをバックアウトしません (そのログを保存した結果も同じです)。
- DBRC=C を使用するときには、自分がバックアウトしているジョブが完了した後で、そのジョブと同じデータベースを他のアプリケーションが変更していないことを確認する必要があります。また、SYSIN データ・セットに BYPASS LOGVER ユーティリティ制御ステートメントも指定する必要があります。DBRC=C の場合、ログ検査は行われません。
- オンライン再編成 PSB をバックアウトするために、DLI 領域タイプを使用して DFSBB00 ユーティリティを実行する必要があります。
- このユーティリティを RECON データ・セットに登録されていない IMS カタログ・データベースで使用するには、追加の EXEC パラメーター DFSDF を指定する必要があります。DFSDF= は、未登録の IMS カタログ・データベースの名前を含む IMS.PROCLIB データ・セットの DFSDFxxx メンバーの、3 文字からなる接尾部を指定します。それらの名前は、DATABASE ステートメントの UNREGCATLG パラメーターを用いて指定されます。例えば、次のようになります。

```
//STEP01 EXEC PGM=DFSRR00,
//          PARM=(DBB,DFSBB00,DFSCP01,,,,,,,,,N,N,IRLM,,,,,,,,
//          ,,'DFSDF=CAT')
```

3 文字からなる接尾部 CAT を、未登録の IMS カタログ・データベースの名前を指定する DFSDFxxx メンバーの接尾部で置き換えてください。

## 推奨事項

DFSBB00 ユーティリティには、使用と実行に関連した推奨事項があります。

DB/DC システムまたは DBCTL システムからのログを使用する際に、仮想ストレージ域の容量が限定されている場合、または使用可能な仮想専用ストレージが十分でないことを示すエラーを受け取った場合は、READBACK ステートメントを使用してください。READBACK ステートメントにより、ログの順方向読み取りの間にデータベース変更を仮想ストレージに保管するのではなく、ログを逆方向に読み取ってバックアウトを行うように、ユーティリティに指示します。

ただし、IMS バッチ入力ログには READBACK ステートメントは必要ありません。これらのジョブは自動的に READBACK を使用するからです。

## 入出力

DFSBB00 ユーティリティへの 1 次入力、プログラムの更新をバックアウトする必要があるデータベースです。DFSBB00 ユーティリティの 1 次出力は、



入力と同じデータベースですが、プログラムの更新が除去されています。  
DFSBB00 ユーティリティは、出力ログも生成する場合があります。

DFSBB00 ユーティリティの入出力は、以下の表に示すとおりです。

表 9. DFSBB00 ユーティリティで必要なデータ・セット

入力	出力
RECON	
バックアウトする必要がある更新を持つデータベース。	更新がバックアウトされたデータベース。
parm=DLI なら、DBD/PSB ライブラリー	IEFRDER (出力ログ)
parm=DBB なら、ACBLIB ライブラリー	
1 つ以上の入力ログ	
SYSIN 制御ステートメント	

DFSBB00 ユーティリティへの入力は、次に挙げるものから成り立ちます。

- バックアウトするデータベース更新が入っているログ・データ・セット (SLDS または OLDS、あるいはその両方、テープまたは DASD、あるいはその両方)。バックアウトされる更新がバッチ・ジョブからのものであるときは、そのジョブの 1 回の実行でとられた完全ログを指定する必要があります。バックアウトされるデータベース更新が IMS DB/DC または DBCTL サブシステムによって行われたものである場合は、すべての更新がその UOR で完了していることを DFSBB00 ユーティリティが認識できるだけのログ・データ・セットが含まれていなければなりません。バックアウトしているデータベースで起こった入出力エラーでリカバリーしていないものがない場合は、再始動に失敗した結果とられたログ・データ・セットは必要ありません。

動的バックアウト障害の場合は、DFSBB00 ユーティリティには、UOR の開始を示す X'5607' レコードから UOR の終了を示す X'07' レコードまでの間のすべてのレコードが必要です。場合によっては、バックアウトは、動的障害の後とられた IMS チェックポイントが入っているログを使用して行うことができます。動的バックアウト障害の原因となった問題がそのチェックポイントのデータ収集の妨げとなっているときは、そのチェックポイント・データはバックアウトに使用できません。

未完了 UOR および未確定 UOR の場合は、DFSBB00 ユーティリティには、UOR の開始を示す X'5607' レコードから次の再始動 (これは含まれない) までの間のすべてのレコードが必要です。

BYPASS LOGVER ユーティリティ制御ステートメントが使用される場合は、UOR の開始時にアクティブであったデータ・セットから、その UOR によって影響を受けたすべてのデータベースが停止されたときにアクティブであったデータ・セットまでの間のログ・データ・セットをすべて入れてください。影響を受けたデータベースが停止された後で、そのいずれかで /START コマンドが出されたときは、その箇所までのすべてのデータ・セットを入れてください。

ACTIVE または COLDSTART ユーティリティ制御ステートメントのどちらかが使用されているときは、最後の同期点またはアプリケーション・チェックポイ

ントから再始動までのすべてのログ・データ・セットを入れておかなければなりません。再始動の前に COLDSTART または ACTIVE 制御ステートメントを使用して DFSBBO00 ユーティリティを初めて実行するときは、入力ログに、IMS チェックポイントと、終了時にアクティブであったすべての非 BMP アプリケーションの先頭を入れておく必要があります。

- バックアウトする必要がある更新を持つデータベース。更新のために使用された PCB に入っているデータベースに関係するすべてのデータ・セットを与える必要があります。
- 何をバックアウトすべきかを決定するオプションの制御ステートメント

DFSBBO00 ユーティリティからの出力には、次のものがあります。

- 最後の同期点までバックアウトされた (IRLM を使用しないバッチ・ジョブで CHKPT ステートメントが使用されていたときは、特定のチェックポイントまでバックアウトされた) 未完了のトランザクションまたはジョブによって行われた変更を持つ入力データベース
- バックアウトされたデータベースのどちらかで順方向リカバリーの必要が起こった場合に備えて、データベース・リカバリー・ユーティリティへの入力として保管しておくべき出力ログ

DFSBBO00 ユーティリティからの出力ログは、変更累積ユーティリティの入力として必要になる場合もあります (DFSBBO00 ユーティリティ稼働時間が変更累積の全期間内にある場合)。

このログは、後続の DFSBBO00 ユーティリティ実行の入力に使用しないでください。

## 戻りコード

DFSBBO00 ユーティリティは、多数の戻りコードのいずれかを出すことができます。戻りコードが返されるたびに、メッセージが印刷されます。

DFSBBO00 ユーティリティの戻りコードとそれに関連付けられたメッセージは以下のとおりです。

コード 意味

- 0 バックアウトが正常に完了した (DFS395I)
- 4 PSB が正しくない (DFS396I)
- 8 データベースがオープンできない (DFS397I)
- 12 データベース入出力エラー (DFS398I)
- 16 バッファ・プールが小さすぎる (DFS399I)
- 20 入力ログがオープンできない (DFS400I)
- 24 DBRC への呼び出しが失敗した (DFS401I)
- 28 データベース・レコードがログにない (DFS888I)

入力ログがオンライン IMS サブシステムからのものであれば、このコードは、バックアウト基準に合致する UOR がログになかったことを意味します。正しいログと正しい PSB 名を指定したかを確認してください。PSB 名

とログが正しかった場合は、レコードが正常にコミットされていて、バッチ・バックアウトが不要であることを検証するために、ログを検査することができます。

- 32 IMSLOGR DD ステートメントにブロック・サイズがない (DFS890I)
- 36 入力ログのレコードが無効 (DFS894I)
- 40 予想外のレコードが見付かった (DFS896A)
- 48 指定された CHKPT が見付からない (DFS958I)
- 52 指定された CHKPT が最後のスケジュール内にはない (DFS959I)
- 60 入力ログが DD DUMMY と指定されたか、あるいは複数のログ DD ステートメントが指定されたが、正しい DD 名または順序が指定されなかった (DFS2296A)
- 64 PSB でのバックアウト処理が未完了 (DFS2298A)
- 68 ログ・シーケンス・エラーが入力ログに見付かった (DFS3278A)
- 72 無効なオプション・ステートメントが SYSIN に見付かった (DFS898A)
- 80 入力ログのタイプと矛盾する制御ステートメントが SYSIN データ・セットに見付かった
- 88 psbname で指定した PSB のデータベースでバックアウトが未完了 (DFS3283A)
- 96 2 種類の制御ステートメントが SYSIN データ・セットに見付かった。これらは一緒には使用できない
- 108 ログ・データ・セットが JCL の中で正しい順序になっていないか、メッセージに示されたログ・データ・セットに入っているデータが正しくない
- 112 入力制御ステートメントと EXEC PARM で暗黙に指定された基準に合致する保留中のバックアウトが RECON バックアウト・レコードに示されていない (DFS3290I)
- 120 進行中の再始動が行うものと同じバックアウトを行わせることを暗黙に指定した制御ステートメントが DFSBBO00 ユーティリティーに渡された (DFS3292I)
- 124 DFSBBO00 ユーティリティーはバックアウトを行ったが、そのバックアウトが必要であると RECON バックアウト・レコードに示されていない (DFS3293W)
- 128 入力ログがバックアウトに対して無効である (DFS3294A)
- 132 READBACK 制御ステートメントが使用された。システム・チェックポイントはバックアウトが必要であることを示しているが、入力として与えられたログ・データ・セットに元のログ・データが入っていない (DFS3295A)
- 140 SYSIN データ・セットに BYPASS LOGVER ユーティリティー制御ステートメントを入れずに DBRC=C を指定した (DFS3296A)

これらの戻りコードは、後続のジョブ・ステップの EXEC ステートメントで COND= パラメーターを指定すると、テストすることができます。

以下の戻りコードが出されると、ABENDU007I が出されます。

コード 意味

- 20 IMSLOGR DD がオープンできないか (DFS400I)、永続入出力エラーが起こった (DFS319I)
- 56 EXEC パラメーターで正しくない PSB が指定された (DFS428I)
- 60 IMSLOGR DD DUMMY または装置タイプが認識されなかった (DFS2296A)
- 68 このバックアウト実行には DBRC が必要である (DFS044I)
- 72 このバックアウト実行には IRLM が必要である (DFS045I)
- 76 X'42' ログ・レコードが見付からない (DFS091I)
- 80 IRLM がアクティブであった場合、正常に完了したジョブがバックアウトできない (DFS173I)

## JCL 指定

DFSBB000 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。DFSBB000 ユーティリティは、標準の z/OS ジョブとして実行されます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含めることができます。

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

DFSBB000 ユーティリティの EXEC ステートメントは、必要なステートメントが入っているカタログ式プロシーチャーを呼び出すことができますが、特定の形式を使用して JCL に指定することもできます。

このステートメントは次の形式になっていなければなりません。

```
//stepname EXEC PGM=DFSRR00,  
// PARM='DBB,DFSBB000,psbname,nnn'
```

または

```
//stepname EXEC PGM=DFSRR00,  
// PARM='DLI,DFSBB000,psbname,nnn'
```

### DBB

DFSBB000 ユーティリティが DBB タイプ領域で実行されることを指定します。事前に作られたブロックが使用されます。IMSACB DD ステートメントは必須です。

### DLI

DFSBB000 ユーティリティが DLI タイプ領域で実行されることを指定します。事前に作られたブロックは使用されません。IMS DD ステートメントが必要です。

## psbname

バックアウトするためにプログラムによって使用される PSB の名前

HALDB オンライン再編成機能が行ったデータベース再編成をバックアウトする場合、HALDB オンライン再編成機能を使用した 8 文字の PSB 名を指定する必要があります。HALDB オンライン再編成機能用の PSB 名は、ゼロで始まり、その後 7 文字の区画名が続きます。例えば、0POHIDKA のようになります。

## nnn

データベース・バッファ・プール用に必要な 1 KB ブロックの数

正常終了したバッチ・ジョブが EXEC ステートメントの DBRC を使用していたが IRLM を使用していなかった場合に、そのバッチ・ジョブをバックアウトするときは、DBRC パラメーターに値「C」を指定しなければなりません。

DBRC パラメーターが使用されているバッチ・ジョブの DBRC サブシステム・レコードに異常終了のマークが付けられていると、DFSBB000 ユーティリティーは、DBRC パラメーターに「Y」が指定されたのと同じ働きをします。

## DD ステートメント

DFSBB000 ユーティリティーは、必須の DD ステートメントおよびオプションの DD ステートメントを数多く使用します。

DD ステートメントで定義する DD 名は小文字で入力しますが、有効なものであれば、どの DD 名でも構いません。

### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

### IMS DD

IMS.PSBLIB データ・セットおよび IMS.DBDLIB データ・セット (PARM='DLI,...') を記述します。これらのデータ・セットは、直接アクセス・ボリュームに置いておかなければなりません。

### IMSACB DD

IMS.ACBLIB データ・セット (PARM='DBB,...') を記述します。このデータ・セットは直接アクセス・ボリュームに置いておかなければなりません。IMS DD ステートメントは、IMS.PSBLIB ライブラリーと IMS.DBDLIB ライブラリーを連結します。

重要: 汎用順次アクセス方式 (GSAM) データベースはサポートされません。バッチ・バックアウト・ユーティリティー (DFSBB000) では、IMS バッチ・ジョブと同じ規則に従う必要があります。PSB に GSAM データベースがあるために、IMS バッチ・ジョブで IMS DD ステートメントが必要な場合、バッチ・バックアウト・ユーティリティー (DFSBB000) でも IMS DD ステートメントが必要です (バッチ・バックアウト・ユーティリティー (DFSBB000) が GSAM データベースにアクセスしない場合でも)。

注: ただし、GSAM DB はバックアウトされませんが、XRST 呼び出しを使用して BMP の再始動プロセス中に再配置されます。XRST 呼び出しは、データ・セット・ポインターを、呼び出しで指定されているチェックポイント ID に位置変更します。アプリケーションが始動する時、そのポイントから始まり、前に進みます。XRST 呼び出しで指定されたチェックポイント ID は、非 GSAM の DB が (動的バックアウトまたはバッチ・バックアウトのいずれかを使用して) バックアウトされた時と同じ ID でなければなりません。

#### **IMSLOGR DD**

入力ログ・ファイルを記述します。これは、テープまたは DASD に置いておくことができます。

**重要:** モデル DSCB 名は、IMSLOGR ステートメントまたは IMSLOGxx DD ステートメントで参照しないでください。DCB に DSCB 名を指定すると、DFSBB000 ユーティリティは、マルチボリューム・ログ・データ・セットの最初のボリュームだけを処理します。バックアウトは正常に完了しますが、データベースが損傷する可能性があります。

FREE=CLOSE は使用してはなりません。入力ログ・ファイルをオープンした後、それをクローズして、再度オープンすることができます。FREE=CLOSE を使用すると、再オープンが失敗します。

#### **IMSLOGxx**

追加の入力ログを記述します。このデータ・セットは、OLDS または SLDS、あるいはその両方でも構いません。接尾部の xx には、任意の英数字を使用できます。

IMSLOGxx DD ステートメントは、入力ログ・データ・セットを指定します。入力ログ・データ・セットが 1 つしかない場合は、IMSLOGR DD ステートメントで指定します。入力ログ・データ・セットが複数ある場合は、これらのデータ・セットは、IMSLOGR を最も古いログの DD 名として、作成順にリストされていないなければなりません。

#### **IEFRDER DD**

バックアウト時に作成されるシステム・ログを記述します。このデータ・セットは、テープまたは DASD に置かれています。

#### **IEFRDER2 DD**

バックアウト時に作成される 2 次システム・ログを記述します。このデータ・セットは、テープまたは DASD に置かれています。このステートメントは、デュアル・ログ出力が必要な場合にのみ使用します。

#### **database DD**

バックアウトを必要とする PCB のデータベース・データ・セットのための DD ステートメントを記述します。このデータ・セットは、直接アクセス・ボリュームに置いておく必要がありますが、動的に割り振ることができます。EXEC ステートメントで参照された PSB が必要とするデータベースで動的割り振りを使用している場合は、データベース DD ステートメントは必要ありません。

#### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。この DD ステートメントは必須です。

このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリーム (DD \* または DD DATA) に入れて送ることもできます。

また、このデータ・セットは、シスプレックス環境内のカップリング・ファシリティー (CF) への接続に使用されるパラメーターも記述します。CFNAMES ステートメントは、バックアウトされる IMS の CFNAMES ステートメント全体と一致している必要があります。障害を起こした IMS で記述されている CFIRLM、CFVSAM、および CFOSAM を指定してください。

#### RECON1 DD

最初の DBRC RECON データ・セットを定義します。

#### RECON2 DD

2 番目の DBRC RECON データ・セットを定義します。

#### RECON3 DD

RECON1 または RECON2 にエラーが見付かったとき使用される、オプションの DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

#### SYSIN DD




ユーティリティ制御ステートメントが入っているデータ・セットを定義します。ユーティリティ制御ステートメントが必要でない場合、このステートメントは省くことができます。このデータ・セットは、テープまたは直接アクセス・ボリュームに置いておくことも、入力ストリーム (DD \* または DD DATA) に入れて送ることもできます。このデータ・セットは、物理順次データ・セットでなければなりません。レコード形式は F で、レコード長は 80 でなければなりません。

#### DFSCTL DD



SB 制御ステートメントが入っているステートメント・イメージ・ファイルを指すオプション・ステートメントです。これは、順次データ・セットにすることも、PDS のメンバーにすることもできます。レコード形式は F、FB、または FBS のいずれかで、レコード長は 80 でなければなりません。

SBIC 制御ステートメントは、このユーティリティへの入力として必要な SB イメージ・キャプチャー・ログ・レコードを作成するために、実行されるプログラムの /DFSCTL 入力データ・セットに入れておかなければなりません。

関連概念:

-  IMS バッファ・プール (システム定義)
-  データベースのバックアウト (データベース管理)
-  HALDB オンライン再編成での IMS ユーティリティの使用 (データベース管理)

関連資料:

-  順次バッファリング制御ステートメント (システム定義)
-  DBBBATCH プロシージャ (システム定義)

## DFSBB00 ユーティリティーのユーティリティー制御ステートメント

制御ステートメントを使用して、DFSBB00 ユーティリティーの処理オプションを定義します。

サブセクション

- 229 ページの『ABEND ステートメント』
- 229 ページの『ABENDMSG ステートメント』
- 229 ページの『ACTIVE ステートメント』
- 230 ページの『BYPASS LOGVER ステートメント』
- 230 ページの『BYPASS SEQVER ステートメント』
- 231 ページの『CHKPT ステートメント』
- 231 ページの『COLDSTART ステートメント』
- 232 ページの『NOREADBACK ステートメント』
- 232 ページの『READBACK ステートメント』

制御ステートメントは、SYSIN DD ステートメントが指す SYSIN データ・セットに入れることにより、ユーティリティーに渡すことができます。

IMS バッチ・ジョブのバックアウトについて、どのユーティリティー制御ステートメントも使用されない場合は、入力ログで最後のチェックポイントの後に現れた EXEC ステートメントに指定されている PSB のすべてのデータベース更新がバックアウトされます。IMS バッチ・ジョブからどのチェックポイントも出されていないと、「IMS 開始」ログ・レコード以後のすべてのデータベース更新がバックアウトされます。入力ログに「IMS 開始」レコードも、チェックポイント・レコードも入っていない場合、バックアウト要求は拒否されます。

入力ログが IMS DB/DC または DBCTL サブシステムからのもので、どの制御ステートメントも使用されていないと、DFSBB00 ユーティリティーは、RECON バックアウト・レコードに示されているバックアウトだけを行います。

COLDSTART または ACTIVE 制御ステートメントを使用した前の DFSBB00 ユーティリティーの実行の結果として、RECON バックアウト・レコードに未完了 UOR と未確定 UOR が示されていると、DFSBB00 ユーティリティーは、該当する制御ステートメントが指定されている場合にのみ、再始動が完了する前にその UOR をバックアウトします。ACTIVE および COLDSTART 制御ステートメントの説明の項を参照してください。

DBRC を使用しないで DFSBB00 ユーティリティーを実行する場合、入力ログが IMS DB/DC または DBCTL サブシステムからのもので、どの制御ステートメントも使用されていないときは、DFSBB00 ユーティリティーは動的バックアウト障害だけをバックアウトします。



## ABEND ステートメント

ABEND ステートメントは、SYSUDUMP または SYSABEND DD ステートメントが指定されているときに U507 異常終了ダンプをとるために使用します。この U507 異常終了は無条件であり、DFSBBO00 ユーティリティの実行が終了すると起こります。

このユーティリティ制御ステートメントは固定形式です。その桁位置は、次に説明するとおりです。

位置 説明

1-5 ステートメント ID

1 桁目から 5 桁目までには ABEND という文字を入れます。これらの文字は、制御ステートメントであることを意味します。

6 これはブランクにしておきます。

7-80 7 桁目から 80 桁目までには、コメントを入れることができます。

## ABENDMSG ステートメント

ABENDMSG ステートメントは、SYSUDUMP または SYSABEND DD ステートメントが指定されているときに U507 異常終了ダンプをとるために使用します。この U507 異常終了は、特定の障害が発生したときに出されます。この異常終了は、その前に障害を示すメッセージが必ず出されます。

このユーティリティ制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

1-8 ステートメント ID

1 桁目から 8 桁目までには ABENDMSG という文字を入れます。これらの文字は、制御ステートメントであることを意味します。

9 これはブランクにしておきます。

10-80 10 桁目から 80 桁目までには、コメントを入れることができます。

## ACTIVE ステートメント

ACTIVE ステートメントは、未完了 UOR だけをバックアウトするよう、ユーティリティに指示するために使用します。このステートメントが指定されていると、ユーティリティは、ERE NOBMP が行われる前に、未完了バッチ・メッセージ処理プログラム (BMP) からコミットされていない更新をバックアウトします。ERE NOBMP が完了しており、しかも DBRC がアクティブである場合は、ユーティリティは、ACTIVE ステートメントを使用せずにバックアウトを行います。

ACTIVE 制御ステートメントが使用できるのは、IMS DB/DC と DBCTL 環境からの入力ログの場合だけです。このステートメントが使用されていると、DFSBBO00 ユーティリティは、すべての未完了 UOR と未確定 UOR のリストを、入力ログに入っている情報からコンパイルし、DBRC にその情報がまだなければ、そのリストを DBRC に渡します。

このユーティリティー制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

**1-6** ステートメント ID

1 桁目から 6 桁目までには ACTIVE という文字を入れます。これらの文字は、制御ステートメントであることを意味します。

**7-80** 7 桁目から 80 桁目までには、コメントを入れることができます。

## **BYPASS LOGVER ステートメント**

BYPASS LOGVER は、入力ログが無効またはバックアウトが不要であることを RECON 情報が示しているときに、バックアウトを行うために使用します。DBRC=C を指定するときは、BYPASS LOGVER ステートメントも使用しなければなりません。ACTIVE ステートメントと COLDSTART ステートメントの項で説明してあるように、DFSBB00 ユーティリティーが通常、他の PSB に関する未完了 UOR と未確定 UOR のリストを DBRC に渡すことになっている場合、BYPASS LOGVER ステートメントを使用すると DFSBB00 ユーティリティーはそのリストをコンパイルしません。DFSBB00 ユーティリティーは、バックアウトされる PSB に関する活動を DBRC に知らせます。BYPASS LOGVER ステートメントは、IMS バッチ・ジョブからの入力ログまたは IMS DB/DC および DBCTL 環境からの入力ログの場合に使用できます。

このユーティリティー制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

**1-13** ステートメント ID

1 桁目から 13 桁目までには BYPASS LOGVER という文字を入れます。これらの文字は、制御ステートメントであることを意味します。

**14-80** 14 桁目から 80 桁目までには、コメントを入れることができます。

## **BYPASS SEQVER ステートメント**

BYPASS SEQVER は、ログ・レコードのシーケンス検査を禁止するために使用します。BYPASS SEQVER ステートメントは、IMS バッチ・ジョブからの入力ログまたは IMS DB/DC および DBCTL 環境からの入力ログの場合に使用できます。

このユーティリティー制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

**1-13** ステートメント ID

1 桁目から 13 桁目までには BYPASS SEQVER という文字を入れます。これらの文字は、制御ステートメントであることを意味します。

**14-80** 14 桁目から 80 桁目までには、コメントを入れることができます。

## CHKPT ステートメント

オプションの CHKPT ステートメントは、バックアウト先となる、以前のチェックポイントを識別するために使用します。この制御ステートメントは、IRLM を使用していなかった IMS バッチ・ジョブからの入力ログの場合にのみ使用できます。

CHKPT ステートメントは、BMP をバックアウトするときは使用しないでください。DFSBB000 ユーティリティーは、BMP をバックアウトする場合、ログの終わりから逆方向へ読み取っていき、最初に現れたチェックポイントを必ず使用します。

このユーティリティー制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

### 1-5 ステートメント ID

1 桁目から 5 桁目までには CHKPT という文字を入れます。これらの文字は、制御ステートメントであることを意味します。

6 この桁はブランクにしておきます。

### 7-14 チェックポイント ID

これは、CHKPT 呼び出しで IMS へ渡される 8 文字のチェックポイント ID です。この ID は、CHKPT 呼び出しが行われたときに、メッセージ DFS681I の一部として表示されます。

15 この桁はブランクにしておきます。

16-80 16 桁目から 80 桁目までには、コメントを入れることができます。

## COLDSTART ステートメント

COLDSTART ステートメントは、EXEC ステートメントに指定された PSB を使用して、未完了 UOR を持つすべての全機能データベースをバックアウトするために使用します。COLDSTART が指定されていると、据え置きバックアウト、未完了 UOR、および未確定 UOR が、指定された PSB についてバックアウトされます。

COLDSTART 制御ステートメントが使用できるのは、IMS DB/DC と DBCTL 環境からの入力ログの場合だけです。COLDSTART ステートメントが使用されていると、DFSBB000 ユーティリティーは、すべての未完了 UOR と未確定 UOR のリストを、入力ログに入っている情報からコンパイルし、DBRC にその情報がまだなければ、そのリストを DBRC に渡します。

COLDSTART ステートメントは、COLDBASE または COLDSYS の再始動前に、未完了 UOR と未確定 UOR をバックアウトするために使用されます。

COLDBASE の再始動を待ってからバックアウトを行う場合、次の条件の 1 つが存在するときは COLDSTART ステートメントを使用してください。

- DFSBB000 ユーティリティーが DBRC なしで実行される。
- BYPASS LOGVER ステートメントが DFSBB000 ユーティリティーの実行のために使用されている。

COLDSYS の再始動を待ってから未完了 UOR または未確定 UOR をバックアウトする場合、次の条件の 1 つが存在しているときは、DFSBB00 ユーティリティーに対して COLDSTART ステートメントを使用してください。

- DFSBB00 ユーティリティーが DBRC なしで実行される。
- BYPASS LOGVER ステートメントが DFSBB00 ユーティリティーの実行のために使用されている。
- DFSBB00 ユーティリティーの実行で ACTIVE または COLDSTART ステートメントが指定されなかったため、未完了 UOR と未確定 UOR のリストが DBRC へ渡されるようになっている。

このユーティリティー制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

**1-9** ステートメント ID

1 桁目から 9 桁目までには COLDSTART という文字を入れます。これらの文字は、制御ステートメントであることを意味します。

**10** この桁はブランクにしておきます。

**11-80** 11 桁目から 80 桁目までには、コメントを入れることができます。

### **NOREADBACK** ステートメント

バッチ入力ログで NOREADBACK ステートメントを使用すると、DB 更新ログ・レコードを、それがどこから得られたものかに関係なく、初期順方向読み取りの間にバッファに入れて、入力の逆方向読み取りの際には入れないようにします。NOREADBACK を使用すると、DFSBB00 ユーティリティーは、このバッファリングでデータ・スペース・ストレージとローカル・ストレージの両者を使用できません。

NOREADBACK ステートメントは、オンライン入力ログには使用しません。

位置 説明

**1-10** ステートメント ID

1 桁目から 10 桁目までには NOREADBACK という文字を入れます。これらの文字は、制御ステートメントであることを意味します。

**11** この桁はブランクにしておきます。

**11-80** 11 桁目から 80 桁目までには、コメントを入れることができます。

### **READBACK** ステートメント

READBACK ステートメントは、ログを順方向に読み取っているときデータベース変更を仮想ストレージに保管するのではなく、ログを逆方向に読み取ってバックアウトを行うようにユーティリティーに指示するために使用します。

DASD 装置上にある IMS バッチ入力ログでは、ログを逆方向に読み取る必要がある場合は、READBACK ステートメントが必要です。ログがテープ上にある場合は、READBACK ステートメントは不要です。それらのジョブは逆方向読み取りの

みを使用しているからです。磁気テープ装置からログを読み取るときに READBACK を使用すると、入出力パフォーマンスが低下することがあります。

**重要:** 非常に大規模なログ・データ・セットを読み取るときにデフォルト・アクションを使用すると、仮想ストレージが不足して異常終了となることがあります。

位置 説明

#### 1-8 ステートメント ID

1 桁目から 8 桁目までには READBACK という文字を入れます。これらの文字は、制御ステートメントであることを意味します。

9 この桁はブランクにしておきます。

10-80 10 桁目から 80 桁目までには、コメントを入れることができます。

---

## DFSBBO00 ユーティリティーの例

次の例は、PSB PLVAPZ12 を使用するプログラムによって行われたデータベース変更をバックアウトするための JCL を示しています。

```
//*
//EXAMPLE EXEC PGM=DFSRR00,
//          PARM='DBB,DFSBBO00,PLVAPZ12,008,1,,,,,,IMSS,,Y,Y,IRLM'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSABEND DD SYSOUT=A
//IMSACB DD DSN=IMS.ACBLIB,DISP=SHR
//IEFRDER DD DSN=LGBKOUTF,DISP=(NEW,KEEP),
//          UNIT=SYSDA,VOL=SER=000000,
//          DCB=(RECFM=VB,BLKSIZE=8192,LRECL=8188,BUFNO=12),
//          SPACE=(CYL,(1,1))
//IMSLOGR DD DSN=DSHR.OLDSP0,DISP=OLD,
//          UNIT=SYSDA,VOL=SER=IMSQAD
//IMSLOG00 DD DSN=DSHR.OLDSP1,DISP=OLD,
//          UNIT=SYSDA,VOL=SER=IMSQAD
//IMSLOG01 DD DSN=DSHR.OLDSP2,DISP=OLD,
//          UNIT=SYSDA,VOL=SER=IMSQAD
//DFSVSAMP DD *
VSRBF=512,50
VSRBF=1024,50
VSRBF=2048,50
//DBHVSAM1 DD DSN=DSHR.FJVHSG1K,DISP=SHR
//DBHVSAM2 DD DSN=DSHR.FJVHSG1E,DISP=SHR
//HIDAM DD DSN=DSHR.FKVHIG1E,DISP=SHR
//HIDAM2 DD DSN=DSHR.FKVHIG2E,DISP=SHR
//XDLBT04I DD DSN=DSHR.FKVHIIXK,DISP=SHR
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
```

---

## RSR 環境における DFSBBO00 ユーティリティーの実行

DFSBBO00 ユーティリティーは、リモート・サイト・リカバリー (RSR) 構成のアクティブ・サイトのデータベースで実行できます。DFSBBO00 ユーティリティーは、トラッキング・サイトでは使用できません。

DFSBBO00 ユーティリティーは、他の IMS バッチ・ジョブと同様に、トランスポート・マネージャー・サブシステム (TMS) を使用して、アクティブ・サイトのログ・データをトラッキング・サイトに送信します。DFSBBO00 ユーティリティーは、完了したバックアウトについて DBRC に通知するときに、トラッキング・サイ

トに送られるログ・レコードに同じ情報を記録するため、トラッキング・サイト RECON データ・セットが更新されます。

リモート・テークオーバーが行われた後は、このユーティリティーを新しいアクティブ・サイトで使用して、古いアクティブ・サイトで実行されたアプリケーションのコミットされていない変更があればそのバックアウトを実行することができます。通常、新しいアクティブ・サイトでは、リモート・テークオーバーが行われなかった場合に古いアクティブ・サイトで使用するときと同じようにこのユーティリティーを使用します。

ただし、新しいアクティブ・サブシステムの最初の再始動が緊急時コールド・スタート (/ERE COLDSYS) であった場合は、(緊急時再始動の前に) まず COLDSTART ステートメントを使って DFSBBO00 ユーティリティーを実行する必要があります。古いアクティブ・サイトで既にこれを実行したかどうかに関係なく、行わなければなりません。未完了の作業単位と関連付けられているデータベースはいずれも、バックアウトされるまでは許可の対象にはできません。

リモート・テークオーバーが行われた後は、このユーティリティーを実行して、古いアクティブ・サイトで完了されなかったバッチ・ジョブすべてについてコミットされていない変更をバックアウトする必要があります。これは、トラッキング・システムのシャットダウン直後に行うこともできるし、もっと適切な時期まで延期しても構いません。

以前のバックアウトが古いアクティブ・サイトでコールド・スタートの後まで延期されている場合は、DFSBBO00 ユーティリティーを使用して、このバックアウトに対処しなければなりません。これは、トラッキング・システムのシャットダウン直後に行うこともできるし、もっと適切な時期まで延期しても構いません。

---

## 第 16 章 データベース変更累積ユーティリティ (DFSUCUM0)

データベース変更累積ユーティリティは、データベース・リカバリー・ユーティリティに渡すリカバリー情報を簡素化するために使用します。

DFSUCUM0 ユーティリティは、ログ・データ・セットに入っている情報を使用します。出力は順次データ・セットの形式になっています。ユーティリティによる処理には次の事柄が含まれます。

1. データベース変更レコード以外のすべてのレコードを取り除きます。
2. 除去の日付 (複数の場合もある) を指定し、その日付以前のすべてのデータベース・レコードを取り除きます。
3. 受け入れ可能なデータベース変更レコードをソートします。
4. 同じデータベース物理レコードを更新するすべてのデータベース変更レコードを結合します。

その結果のレコードはデータベース内のデータ・セット順に並べられます。また、このユーティリティは、マイナー・フィールド RBA またはキーとしてソートします。

DFSUCUM0 ユーティリティをある期間に何回か実行すると、追加のデータベース変更を取り入れ、有効でなくなった変更を削除することができます。

DFSUCUM0 ユーティリティは、IMS から独立して実行させることができます。DFSUCUM0 ユーティリティが z/OS と一緒に稼働するときは、出力が圧縮されます。この圧縮を行うには、z/OS のサービスが使用されます。

データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

以下の図に、データベース変更累積ユーティリティへの入力源、およびこのユーティリティが作成する出力についてのフロー・ダイアグラムを示します。

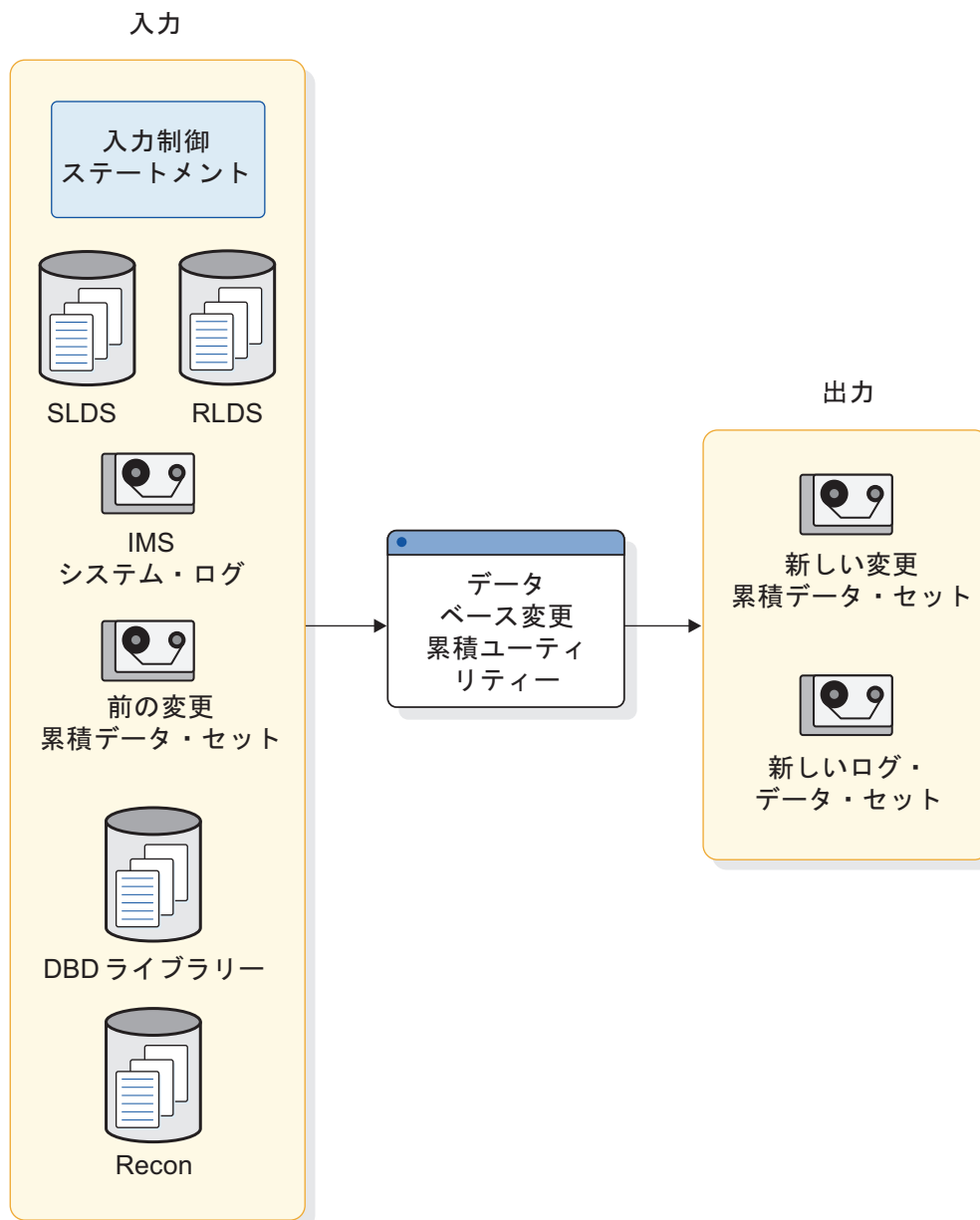


図 58. データベース変更累積ユーティリティ

サブセクション:

- 237 ページの『制約事項』
- 237 ページの『前提条件』
- 237 ページの『要件』
- 237 ページの『推奨事項』
- 237 ページの『入出力』
- 238 ページの『JCL 指定』
- 242 ページの『戻りコード』



## 制約事項

DFSUCUM0 ユーティリティーの制約事項には以下のものがあります。

- DBRC (データベース・リカバリー管理) を使用しているときは、このユーティリティーを使用してオプションのログ出力データ・セットが作成されたことは、RECON データ・セットには記録されません。その他の出力データ・セットが作成されたことは、RECON データ・セットに記録されます。
- データベース変更累積ユーティリティーは再始動できません。

## 前提条件

DFSUCUM0 ユーティリティーを実行する前に、リカバリーに必要なデータベース変更レコードが入っているすべてのオンライン・ログ・データ・セット (OLDS) を保存する必要があります。ログ保存ユーティリティー (DFSUARC0) 用の JCL を生成するために DBRC コマンド GENJCL.ARCHIVE を出して、その後 DFSUARC0 ユーティリティーを実行すると、OLDS を保存できます。

DFSUCUM0 ユーティリティーは z/OS ソート/マージ・プログラムを開始しますが、それが実行の前提条件となります。

## 要件

DFSUCUM0 ユーティリティーへのログ入力は、作成された順序になっていなければなりません。

## 推奨事項

変更累積ユーティリティーのための JCL を生成するには、DBRC コマンド GENJCL.CA を使用します。GENJCL.CA コマンドを使用すると、正しい UTC 除去時刻が適用され、すべての変更累積処理に対してユーティリティー制御ステートメントが正しくコーディングされるようになります。

## 入出力

DFSUCUM0 ユーティリティーへの 1 次入力は、ログ・データ・セットです。

DFSUCUM0 ユーティリティーの 1 次出力は、リカバリー用に最適化された累積データベース変更レコードが入っているデータ・セットです。DFSUCUM0 ユーティリティーは、追加出力も作成します。

データベース変更累積ユーティリティーへの入力は、次に挙げるものから成り立ちます。

- イメージ・コピー・ユーティリティーの最後の実行またはこのユーティリティーの最後の実行以降に作成されたすべての SLDS または RLDS。この入力には、このユーティリティーを前に実行したときの結果として得られた新しいログ出力データ・セットを含めることができます。新しいログ出力データ・セットについては、出力のセクションで説明しています。
- 前のデータベース変更累積ユーティリティー・データ・セット。このデータ・セットは、このユーティリティーの最後の実行からの出力です。
- 通常は IMS.DBDLIB と呼ばれる DBD ライブラリー。

- 除去の日付およびデータベース・ログ・レコードの処理方法を指定する制御ステートメント (ID、DB0 および DB1)。

データベース変更累積ユーティリティからの出力は、次に挙げるものから成り立ちます。

- 新しい変更累積ユーティリティ順次データ・セット。このデータ・セットには、DB0 制御ステートメントに指定されたデータベース/データ・セットの結合データベース・レコードが入ります。IMS では、このデータ・セットは圧縮されています。
- \*OTHER を指定する DB1 制御ステートメントのデータベースまたはデータ・セットで指定された再フォーマット設定済みのデータベース変更ログ・レコードが入っている、新しいログ出力データ・セット。新しいログ・データ・セットは、後続の CA 実行で使用されます。新しいログ・データ・セットは RECON には記録されません。新しいログ・データ・セットを使用して作成される後続の変更累積データ・セットは、DBRC NOTIFY.CA コマンドによって RECON データ・セットに記録する必要があります。新しいログ・データ・セットを使用することはお勧めしませんが、これらは DBRC サポートなしで一般的にシステムで作成され使用されています。
- VSAM データ・セットのリカバリーを行っている場合は、VSAM データ・セットの IMS データベース・リカバリーの正常なメッセージである、メッセージ IEC161I が出力されます。

データベース変更累積ユーティリティの入力源、およびこのユーティリティから得られる出力は、以下の表に示すとおりです。

表 10. データベース変更累積ユーティリティの入出力

入力	出力
RECON	新しいログ・データ・セット
入力制御ステートメント	新しい変更累積データ・セット
IMS システム・ログの SLDS と RLDS	
前の変更累積データ・セット	
DBD ライブラリー	

## JCL 指定

DFSUCUM0 ユーティリティは、標準の z/OS ジョブとして実行されます。DFSUCUM0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティ制御ステートメントが必要です。

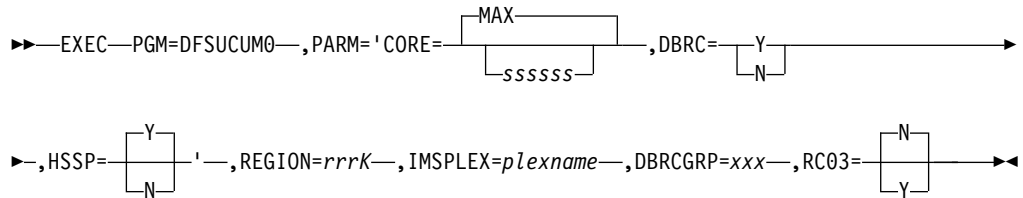
次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

**EXEC** ステートメント

DFSUCUM0 ユーティリティーの EXEC ステートメントは、必要なステートメントが入っているカタログ式プロシーチャーを開始することができますが、特定の形式で JCL に指定することもできます。

EXEC ステートメントを JCL に指定する場合、次のいずれかの形式になっていなければなりません。



### CORE=MAX|ssssss

z/OS SORT/MERGE プログラムがこのアプリケーションのために使用できるストレージの容量を、バイト数で表します。値の指定がないと、プログラムは CORE=MAX が指定されているものと見なします。CORE=MAX を指定すると、DFSUCUM0 はソート・ユーティリティーが使用できるストレージの容量に制限を設けず、ソート作業にはインストール・デフォルト値を使用します。

推奨事項: デフォルト指定の CORE=MAX を使用しないときに、何を指定するかが分からなければ、データ機能分類プログラム (DFSORT) で必要とされる最小限の量を使用してください。16 MB 境界より下に装置制御ブロック (UCB) を割り振る OEM ソート製品を使用する場合は、CORE=MAX を使用してください。

### DBRC

インストール・デフォルト・モジュール DFSIDEF0 の DBRC= の指定を変更するかどうかを、Y か N で指定します。JCL EXEC ステートメントで DBRC=Y と指定すると、このユーティリティーの実行時に DBRC が使用されます。JCL EXEC ステートメントで DBRC=N と指定すると、このユーティリティーの実行時に DBRC は使用されません。

インストール・デフォルト・モジュール DFSIDEF0 に DBRC=FORCE を指定した場合は、DBRC はこのユーティリティーの実行時に常に使用されます。DBRC=FORCE の指定を変更しようとする、メッセージ DFS044I が出され、ゼロ以外の戻りコードが出されます。

### HSSP

Y または N を指定することができます。HSSP=N を使用するの、高速機能ログ・レコードがなく、変更累積レコードとログ・レコードが共に 1 つのテープ・ドライブから入力される時だけです。デフォルトは Y です。

### REGION=rrrK

領域サイズです。領域サイズには ssssss の値に 100 KB を加えた大きさを指定し、ssssss 値の指定がない場合は 304 KB を指定してください。必要となる領域サイズを決めるには、多数の変数が関係します。例えば、入力および出力バッファ要件、制御ステートメントで指定されたデータベース/データ・セットの数、DBD のサイズなどです。

**IMSPLEX=plexname**

DBRC が結合する IMSplex を指定します。

**DBRCGRP=xxx**

このユーティリティーで使用する必要のある DBRC グループを指定します。3桁の英数字の値である変数 xxx は、RECON データ・セットで指定された DBRC グループ ID と一致していなければなりません。

**RC03=**

Y または N を指定できます。DBRC で許容される数を超えるログが DFSULOG DD ステートメントに含まれているときに、ユーティリティーが警告メッセージ DFS3523W と戻りコード 3 で終了するよう指定するには、Y を使用します。この状態は、GENJCL.CA コマンドが実行されてからユーティリティーが実行されるまでの間に ALLOC レコードが RECON データ・セットに書き込まれたときに発生します。一般に、この状態は次の GENJCL.CA コマンドが実行された後、ユーティリティーが再び実行されると解消します。デフォルトは N です。

**DD ステートメント**

DFSUCUM0 ユーティリティーは、数多くの必須およびオプションの DD ステートメントを使用します。

このユーティリティーで使用するデータ・セットを定義する DD ステートメントは、次のとおりです。

**STEPLIB DD**

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。

**SYSPRINT DD**

出力メッセージ・データ・セットを定義します。このデータ・セットは、テープ、直接アクセス・ボリューム、またはプリンターに置いておくことも、出力ストリーム (SYSOUT) に入れて送ることもできます。このデータ・セットに関して指定されている DCB パラメーターは、RECFM=FBA と LRECL=121 です。BLKSIZE を SYSPRINT DD ステートメントに指定する場合は、121 の倍数になっていなければなりません。

**IMS DD**

累積するデータベースをすべて記述する DBD が入っているライブラリーを定義します。これは DSNAME=IMS.DBDLIB であるのが普通です。このデータ・セットは、直接アクセス・ボリュームに置いておかなければなりません。

**SYSOUT DD**

z/OS ソート/マージ・プログラムのための出力メッセージ・データ・セットを定義します。このデータ・セットは、テープ、直接アクセス・ボリューム、またはプリンターに置いておくことも、出力ストリーム (SYSOUT) に入れて送ることもできます。z/OS ソート/マージ・プログラムは AP (すべてのメッセージをプリンターに送る) を指定します。

**SORTLIB DD**

z/OS ソート/マージプログラムの実行のためのロード・モジュールが入ってい

るデータ・セットを定義します。これは `DSNAME=SYS1.SORTLIB` であるのが普通です。このデータ・セットは、直接アクセス・ボリュームに置いておかなければなりません。

#### **SORTWKnn DD**

z/OS のソート/マージ・プログラムのための中間記憶データ・セットを定義します。このデータ・セットは、通常、直接アクセス・ボリュームに置いておきます。ただし、テープを使用することも可能です。

#### **DFSUCUMN DD**

新しい累積変更出力データ・セットを定義します。このデータ・セットはテープまたは直接アクセス・ボリュームに置いておくことができます。指定するブロック・サイズは、64 以上で、装置の最大値 (32760) 以下でなければなりません。ブロック・サイズの指定がない場合、デフォルトのブロック・サイズは装置の最大値です。装置が 3380 のときは、ブロック・サイズは 23476 バイトが使用されます。論理レコード長を `DCB` オペランドで変更することはできません。

#### **DFSUCUMO DD**

新しい累積変更データ・セットを作成するために、ログ入力データとマージする古い累積変更入力データ・セットを定義します。マージする古い累積変更がない場合は、次の `DD` ステートメントが必要です。

```
//DFSUCUMO DD DUMMY,DCB=BLKSIZE=100
```

このデータ・セットはテープまたは直接アクセス・ボリュームに置いておくことができます。

#### **DFSUDD1 DD**

新しいログ出力データ・セットを定義します。出力データ・セットは、テープまたは直接アクセス・ボリュームに置いておくことができます。指定するブロック・サイズは、64 以上で、装置の最大値 (32760) 以下でなければなりません。ブロック・サイズの指定がない場合、デフォルトのブロック・サイズは装置の最大値です。装置が 3380 のときは、ブロック・サイズは 23476 バイトが使用されます。論理レコード長および `RECFM=VBS` を `DCB` オペランドで変更することはできません。

このプログラムの中で指定されている `DCB` パラメーターは `RECFM=VBS` と `LRECL=32760` です。これらは変更できません。出力は、装置の最大容量までブロック化されます。標準ラベルを使用しなければなりません。

#### **DFSUDD2 DD**

再フォーマット設定された新しいソート出力ログ・データ・セットを定義します。これは診断の目的だけに使用され、`SO` 制御ステートメントが指定されているときに必要になるものです。出力は、テープまたは直接アクセス・ボリュームに置いておくことができ、装置の最大容量または 32760 のどちらか小さい方の値までブロック化されます。標準ラベルを使用しなければなりません。

#### **DFSULOG DD**

累積する変更レコードが入っているログ入力データ・セットを定義します。このデータ・セットはテープまたは直接アクセス・ボリュームに置いておくことができます。

データ・セットを連結すると、複数のログ・データ・セットを入力として使用できます。ログ入力は、作成された順序になっていなければなりません。DBRCは、ログ・データ・セットがその START TIME に従って発生順になっているか調べます。

#### **SYSIN DD**

制御ステートメント・データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス・ボリュームに置いておくことも、入力ストリーム (DD \* または DD DATA) に入れて送ることもできます。

#### **RECON1 DD**

最初の DBRC RECON データ・セットを定義します。制御領域について定義した RECON データ・セットと同じものでなければなりません。

動的割り振りを使用しているときは、RECON データ・セットの DD ステートメントを含めないようにしてください。

#### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。制御領域で 2 番目として定義した RECON データ・セットと同じものでなければなりません。

動的割り振りを使用しているときは、RECON データ・セットの DD ステートメントを含めないようにしてください。

#### **RECON3 DD**

3 番目の DBRC RECON データ・セットを定義します。制御領域で 3 番目として定義した RECON データ・セットと同じものでなければなりません。

動的割り振りを使用しているときは、RECON データ・セットの DD ステートメントを含めないようにしてください。

### 戻りコード

データベース変更累積ユーティリティの戻りコードは、次のとおりです。

コード 意味

- 0 すべての操作が正常に完了した
- 3 DFSULOG DD ステートメントに、DBRC で許容される数を超えるログ・レコードが含まれている
- 4 警告メッセージが出された
- 8 1 つ以上の操作が正常に完了しなかった
- 16 z/OS ソート/マージ・プログラムは正常終了しませんでした。


z/OS ソート/マージ・プログラムのインストールの際に ERRET=ABEND が指定されていた場合、その他の戻りコードは z/OS ソート/マージ・プログラムによって返される場合があります。

これらの戻りコードは、後続のジョブ・ステップの EXEC ステートメントで COND= パラメーターを指定すると、テストすることができます。

関連概念:

 RECON データ・セットの初期化および保守 (システム管理)

関連タスク:

 リカバリーのためのデータベース変更累積入力の使用 (データベース管理)

---

## DFSUCUM0 ユーティリティーの制御ステートメント

データベース変更累積ユーティリティーで使用する 4 つのユーティリティー制御ステートメントは、ID、DB0、DB1、および SO です。

特定のパラメーターが指定されている場合を除いて、これらのステートメントをすべて使用することも、あるいは任意に組み合わせて使用することもできます。

DBRC コマンドの GENJCL.CA を使用してデータベース変更累積ユーティリティー用の JCL を作成すると、これらの制御ステートメントをユーザーがコーディングする必要がなく、除去の日付と時刻を含めた JCL 指定は確実に正しくなります。ユーザーが自分で制御ステートメントをコーディングする場合は、ユーティリティーの EXEC ステートメントで必ず DBRC=Y を指定してください。これは、制御ステートメントが正しいことを DBRC が検証するようにするためです。

データベース変更累積ユーティリティーは、ユーティリティー制御ステートメント情報がない場合には、次のような制約を受けます。

- ユーティリティー制御ステートメントがない場合には、すべてのデータベース・ログ・レコードをソートし、結合して、新しい変更累積データ・セットを作成する必要があります。除去の日付がなく、基本キーの最大長は 10 バイトであると見なされます。
- DB0 または DB1 制御ステートメントがない場合には、すべてのデータベース・ログ・レコードをソートし、結合して、新しい変更累積テープを作成する必要があります。DBRC サポートを実行中の場合、DFSUCUM0 ユーティリティーは DBRC が提供する除去の日付を適用しますが、それ以外の場合は、ログ・レコードに対して除去の日付は適用されません。
- 変更累積ユーティリティー用の JCL に DB0 ステートメントを手動でコーディングし、除去時刻にオフセットを指定しない場合、ユーティリティーは、入力時刻がローカルであるものと見なして、UTC (Universal Coordinated Time) からのオフセットのローカル z/OS 設定を使用して、時刻を UTC に変換します。ゼロのオフセットを明示的にコーディングした場合、変更累積ユーティリティーは、入力タイム・スタンプをオフセットなしの UTC として解釈します。

制約事項: 複数の制御ステートメントを含める場合、あるステートメントに \*ALL または \*OTHER を指定すると、他の制御ステートメントで指定できる内容に影響することがあります。以下の規則があります。

- DB0 ステートメントを含めると、いずれの DB1 ステートメントにも \*ALL を指定できなくなります。
- いずれかの DB0 ステートメントに \*ALL を指定すると、DB1 ステートメントを含めることができなくなります。
- DB1 ステートメントに \*ALL を指定すると、他のすべての DB1 ステートメントに \*OTHER を指定できなくなります。

推奨事項: 変更累積ユーティリティーのための JCL を生成するには、DBRC コマンド GENJCL.CA を使用します。GENJCL.CA コマンドを使用すると、正しい UTC

除去時刻が適用され、すべての変更累積処理に対してユーティリティー制御ステートメントが正しくコーディングされるようになります。ただし、変更累積ユーティリティー用の JCL を手動でコーディングする場合は、正しい除去の日付と時刻を指定しなければなりません。

## 除去の日付と時刻の使用

変更累積レコード (または入力ログ) がイメージ・コピー時刻または再編成時刻の前後にわたる場合、入力ログには、イメージ・コピーまたは再編成時刻の前と後に作成された変更レコードが含まれます。イメージ・コピー時刻または再編成時刻に対応する除去の日付と時刻を指定する必要があります。こうすると、リカバリーに使用されないデータベース変更レコードが除外されます。

処理されるすべてのデータベース・ログ・レコードに除去の日付と時刻を 1 つ指定できます。除去の日付と時刻は、特定のデータベースを更新するすべてのデータベース・ログ・レコードに指定することもできるし、あるデータベース内のデータ・セットを更新するすべてのデータベース・ログ・レコードに指定することもできます。複数の除去の日付と時刻を指定することも可能であり、この場合は、データベース内のデータ・セットごとに異なる除去の日付と時刻を指定できます。

除去の日付を指定し、その時刻を指定しない場合は、時刻はゼロが指定されたものと見なされます。

入力ログ・テープの場合、変更累積ユーティリティーは、指定された除去の日付を各データベース更新レコードに入っている日付および時刻と比較します。除去の日付が更新レコードに入っている日付および時刻よりも後であれば、入力ログ・レコードが除去されます。

以前の変更累積テープの場合、このユーティリティーは、DBRC またはユーティリティー制御ステートメントで指定された除去の日付を、以前の変更累積テープの日付および時刻と比較します。除去の日付が以前の変更累積テープの日付および時刻よりも後であれば、入力ログ・レコードが除去されます。

これらのレコードに入っている日付と時刻は、データベース・ブロック内での最新の更新を表します。このような理由から、除去の日付よりも前の更新がそのレコードに入っている場合、その更新が除去されないことが起こりえますが、これは、その同じレコードに入っている最新の更新が指定された除去の日付よりも後であるためです。このことは、変更累積が再編成の前後に行われる場合に特に重要です。再編成の後にデータベースが初めて累積される際には、除去の日付を必ず指定してください。そうすることで、古いレコードが正しく廃棄されます。

データベース 1 をロードする

処理 - ログ 1

処理 - ログ 2

(ログ 1、ログ 2) を累積して累積ログ A に入れる

処理 - ログ 3

処理 - ログ 4

データベース 1 を再編成する

処理 - ログ 5



処理 - ログ 6

(ログ 5、ログ 6) を累積して累積ログ B に入れる

- 累積の実行では、累積ログ A が入力されなければ、除去の日付と時刻は必要ありません。
- 累積ログ B には、データベース 1 の再編成以後の変更ログ・レコードだけが入っています。

データベース 2 をロードする

データベース 1 とデータベース 2 の処理 - ログ 7

データベース 1 とデータベース 2 の処理 - ログ 8

データベース 2 を再編成する

(ログ 7、ログ 8) を累積して累積ログ C に入れる

- ここで除去の日付と時刻が必要になるのは、データベース 2 が再編成されたので、データベース 2 の前のログ・レコードを除去するためです。除去の日付と時刻がないと、再編成の前に存在していたデータベース・レコードが累積されるため、その後で同じブロックが更新されると、除去できない場合があります。再編成前に存在していた更新レコードが入っている変更累積データ・セットは、リカバリーの入力として使用されると、データベースを壊してしまいます。

除去の日付と時刻は、どの DB0 または DB1 ユーティリティー制御ステートメントでも指定できます。ログ・レコードは、制御ステートメントで指定されたレコードのうち、除去の日付以前に作成されたものが除去されます。古いデータベース変更累積データ・セットをこのユーティリティーの実行により更新した場合には、DB0 ID に一致しており、しかも除去の日付以前の日付を持つ古い累積変更レコードが除去され、新しい累積変更テープには書き込まれません。

## ID ステートメント

オプションの ID ステートメントを使用すると、この変更累積を実行するとき必要になるテーブル要件とソート要件の両方を記述することができます。このステートメントを使用するときは、最初に与えるステートメントにしなければなりません。このステートメントを使用しない場合は、デフォルト値が割り当てられます。このユーティリティー制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

### 1-2 ステートメント ID

1 桁目と 2 桁目には、ID という文字を入れておかなければなりません。これにより、このステートメントがデータベース変更累積ユーティリティー制御ステートメントであることが示されます。

### 3-30 3 桁目から 30 桁目までは空白にしておきます。

### 31-33 シーケンス ID の最大長 (オプション)

31 桁目から 33 桁目までには、DB0 制御ステートメントの結果として処理されるログ・レコードで参照されるすべてのデータベース・レコードに関して、最大レコード ID フィールドの長さを入力できます。OSAM デ

ータ・セットまたは ESDS に入っているデータベース・レコードの場合は、ID フィールドは RBN です。KSDS の場合は、レコード ID はルート・セグメントのキーです。

この最大長の値は、ソートの目的で、最大長より短いレコード ID に埋め込む 2 進ゼロの長さを計算するために使用されます。処理する VSAM KSDS がない場合は、この値を 4 (相対ブロック番号フィールドの長さ) と指定するか、ブランク (デフォルト値の 4 が使用される) にしてください。

処理する VSAM KSDS オカレンスがある場合は、この値を、1 から 256 までの範囲で指定して、左寄せするか、先行ゼロを入れます。あるいは、すべて省略することもできます。この値が省略されていて、1 つ以上の KSDS オカレンスが処理される場合、ユーティリティーは、リストされたすべての KSDS オカレンスに対する最大キー長を判別します。

このステートメントで値が指定されていたのに、それよりも大きな値が KSDS に関して検出された場合は、大きい値が指定値に優先します。

#### 41-45 最大レコード長 (max lrecl)

- 41 桁目から 45 桁目までには、データベース変更タイプのログ・レコードの最大長に 31 桁目から 33 桁目までに指定した最大シーケンス長を加えたものを入れます。デフォルト値は 4351 です。IMS (高速機能を除く) により作成されるデータベース変更タイプ・ログ・レコードの最大長は 4096 バイトであり、最大シーケンス長は 255 です。このパラメーターを指定するのは、高速機能のデータベース変更タイプ・レコードがログに書き込まれることが予想されるときです。高速機能では、このパラメーターは、どの区域の場合も、その最大制御インターバル・サイズに 255 を加え、それに MAX SEQ の長さを加え、それから 4 を引いたもの (4 より大きい MAX SEQ の長さが指定されている場合) にしなければなりません。値は、左寄せにするか、先行ゼロを入れてください。

MAX LRECL サイズは、IMS のリリースによって異なる場合があります。

46-80 コメントは、46 桁目から 80 桁目までに入れることができます。

## DB0 ステートメント

オプションの DB0 ステートメントは、新しい変更累積データ・セットへの出力として、どのレコードを累積するのかを記述するために使用します。このステートメントは、1 つ以上使用できます。DB0 ステートメントは、一般的には、何を入力ログ・データ・セットから累積して出力累積データ・セットに入れるかを指示するために使用されます。DB0 ステートメントで指定されたどのデータベースやデータ・セットの場合も、古い累積データ・セットからの入力、除去の日付が DB0 ステートメントで指定されているために除去される場合を除き、出力累積データ・セットに入れられます。それ以外のデータベースやデータ・セットの場合は、その古い累積データ・セットからの他の入力は、DBRC がアクティブでなければ、すべてが繰り越されて出力累積データ・セットに入れられます。DBRC がアクティブな場合は、これらのレコードは除去されます。

データベース名と DD 名のそれぞれの組み合わせは、1 つの制御ステートメントだけで指定できます。

DB0 ユーティリティー制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

**1-3** ステートメント ID

1 桁目から 3 桁目には、DB0 という文字を入れておかなければなりません。これにより、このステートメントがデータベース変更累積ユーティリティー制御ステートメントであることが示されます。

**4-11** dbname

- 4 桁目から 11 桁目までには、データベース名または「\*ALL」を入れることができます。\*ALL の場合は、4 桁目を空白にし、5 桁目から 8 桁目に文字 \*ALL を入れます。\*ALL を指定すると、すべてのレコードが累積され、12 桁目から 20 桁目までの除去の日付と時刻がすべてのレコードに適用され、DB1 ステートメントが指定できなくなります。DBRC が使用されているときは、DB0 制御ステートメントの \*ALL オプションを指定しても無効です。データベース名と DD 名が指定されたときは、除去の日付は、データベース名/DD 名の組み合わせが一致したレコードだけに適用されます。DD 名の指定がないと、除去の日付は、データベース名が一致したレコードすべてに適用されます。

HALDB データベースの場合は、必ず、DD 名を指定してください。変更累積 JCL を生成するためには、手動でコーディングするのではなく、DBRC GENJCL コマンドを使用してください。

**12-42** 除去の日付と時刻。除去の日付と時刻が指定されると、データベース名/DD 名の記述に一致しており、しかも除去の日付より前の日付を持つレコードがすべて除去されます。古い変更累積入力が指定されているときは、データベース名/DD 名の記述に一致しており、しかも除去の日付より前の日付であるすべてのレコードは新しい変更累積データ・セットにマージされません。このフィールドを空白にすると、除去の日付は使用されません。オフセットを指定しない場合、デフォルトでローカル・オフセットが設定されます。

**43-68** DD 名

43 桁目から 50 桁目まで、52 桁目から 59 桁目まで、および 61 桁目から 68 桁目までに、DD 名を最高 3 つ入れることができます。この DD 名は、指定されたデータベース名と結合されて、データベース・データ・セット ID となります。

DD 名は、左の桁から右の桁へ向かって入れる必要があります。この ID と一致するレコードはすべてソートされて累積され、除去の日付と時刻が適用されます。データベース名、除去の日付、および DD 名を指定した組み合わせは、追加の DB0 制御ステートメントをサブミットすることによって、必要な数だけ指定することができます。DD 名の指定がないと、除去の日付と時刻は、データベース名と一致したレコードすべてに適用されます。

DD 名はすべて左寄せしなければなりません。使用しない桁は、空白にしておきます。DD 名の指定はないが、4 桁目から 11 桁目に DB 名の指定があるときは、データベースに行ったすべての変更が累積変更デー

タ・セットに書き込まれます。指定する名前は、該当する DBDGEN のデータ・セットまたは区域ステートメントの DD1 キーワードで使用されたものと同じ名前にしなければなりません。高速機能への入力の場合は、これは、DD 名と区域名のうち、DBDGEN 時に使用されたものにすることができます。どちらの名前が使用されたか判別するには、DBDGEN を参照してください。HALDB OLR 対応のデータベースの場合は、必ず、DD 名を指定してください。A から J の DD 名と M から V の DD 名はどちらも、入力として使用できます。

推奨事項: 変更累積 JCL を生成するためには、DBRC GENJCL コマンドを使用してください。

#### 69-80 充てん文字/無視される

### DB1 ステートメント

このステートメントは、どのレコードを新しいログ出力データ・セットに書き込むかを記述するために使用してください。これらのレコードはソートされず、読み取られたときと同じ順序で書き込まれます。データベース変更レコードでないログ・レコードは、いずれも出力データ・セットには書き込まれません。データベース名および DD 名の組み合わせを記述する DB1 ステートメントは、いくつでも使用できます。

データベース名と DD 名のそれぞれの組み合わせは、1 つの制御ステートメントだけで指定できます。

DB1 ユーティリティー制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置 説明

#### 1-3 ステートメント ID

1 桁目から 3 桁目には、DB1 という文字を入れておかなければなりません。これにより、このステートメントがデータベース変更累積ユーティリティー制御ステートメントであることが示されます。

#### 4-11 dbname

4 桁目から 11 桁目までには、データベース名、\*ALL、または \*OTHER を入れることができます。\*ALL の指定があると、すべてのレコードが新しいログ・データ・セットに書き込まれるので、DB0 制御ステートメントが使用できなくなります。\*OTHER の指定があると、DB0 制御ステートメントで記述されていないレコードすべてが新しいログ・データ・セットに書き込まれます。除去の日付と時刻が 12 桁目に指定されていると、この制御ステートメントで指定され、除去の日付より前の日付を持つレコードはすべて新しいログ・データ・セットには書き込まれません。DB0 ステートメントに \*ALL を指定すると、DB1 ステートメントは指定できません。\*OTHER を指定する DB1 ステートメントは、1 つだけ与えることができます。

12-42 除去の日付と時刻。レコード ID の組み合わせに一致し、除去の日付より前の日付を持つレコードすべてが除去されます。

#### 43-68 DD 名

43 桁目から 50 桁目まで、52 桁目から 59 桁目まで、および 61 桁目から 68 桁目までに、DD 名を最高 3 つ入れることができます。この DD 名は、指定されたデータベース名と結合されて、データベース・データ・セット ID となります。

この ID と一致し、しかも除去の日付より後の日付のレコードがすべて新しいログ・データ・セットに書き込まれます。指定する DD 名はすべて左寄せしなければなりません。使用しない桁は空白にしておきます。DD 名の指定はないが、4 桁目から 11 桁目に DB 名の指定があるときは、データベースに行ったすべての変更が新しいデータベース・ログ・データ・セットに書き込まれます。指定する名前は、該当する DBDGEN のデータ・セットまたは区域ステートメントの DD1 キーワードで使用されたものと同じ名前にしなければなりません。高速機能への入力の場合は、これは、DD 名と区域名のうち、DBDGEN 時に使用されたものに行うことができます。どちらの名前が使用されたか判別するには、DBDGEN を参照してください。

#### 69-80 充てん文字/無視

### SO ステートメント

SO ステートメントでは、ソート・プログラムから得た再フォーマット設定出力ログを、ソート出力データ・セットに書き込むよう要求します。使用できる SO ステートメントは 1 つだけです。このステートメントは、問題を判別するときの診断ツールとなるものです。

指定できるオプションは 4 つあります。すなわち、どのオプションも指定しないか、DBNAME、DSID、または RBA/RBN を指定します。

DBNAME、DSID、RBA/RBN オプションは任意に組み合わせて指定できます。オプションを複数選択したときは、そのすべてが満足されていないとレコードは書き込まれません。どのオプションも選択されていないと、ソートされ、再フォーマット設定されたレコードすべてが書き込まれます。

SO ユーティリティ制御ステートメントは固定形式であり、その桁位置は次に説明するとおりです。

位置	説明
----	----

#### 1-2 ステートメント ID

1 桁目と 2 桁目には、SO という文字が入ります。これにより、このステートメントがデータベース変更累積ユーティリティ制御ステートメントであることが示されます。他のオプションの指定がないと、ソートされ、再フォーマット設定されたレコードすべてが書き込まれます。

3 3 桁目は空白にしなければなりません。

#### 4-11 dbname または空白

4 桁目から 11 桁目までには、データベース名を入れることもできますが、空白のままにしておいても構いません。dbname に名前の指定があると、その dbname を持つレコードのみが書き込まれます。

#### 12-14 dsid または空白


12 桁目から 14 桁目までには、データベース・データ・セット ID を入れることもできますが、ブランクのままにしておいても構いません。ブランクでない桁が 1 桁でもあるときは、必要なだけの先行ゼロを使って、すべての桁を数字で埋めなければなりません。データベース・データ・セット ID の指定があると、そのデータ・セット ID を持つレコードのみが書き込まれます。

15 15 桁目はブランクにしなければなりません。

#### 16-25 RBA または RBN

16 桁目から 25 桁目までには、1 つのレコードまたは多数のレコードの RBA (相対バイト・アドレス) または RBN (相対ブロック番号) を入れることができます。レコードに指定の RBA または RBN が含まれていると、その RBA または RBN を持つレコードのみが書き込まれます。RBA または RBN は、16 進数で指定してください。

関連資料:

 DBRC コマンド構文 (コマンド)

---

## DFSUCUM0 ユーティリティーの例

### DFSUCUM0 ユーティリティーの使い方の例

この節にある例では、桁合わせを容易にするために、SYSIN ステートメントの上に次のようなコメント行が示されています。

```
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

このコメント行は単なる参考用です。

この節の例で、使用する DBRC で動的割り振りを行わない場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

図 59. 動的割り振りなしで DBRC を使用するための DD ステートメント

```
//RECON1 DD DSN=RECON1,DISP=SHR  
//RECON2 DD DSN=RECON2,DISP=SHR  
//RECON3 DD DSN=RECON3,DISP=SHR
```

サブセクション:

- 『データベース・ログを累積する』
- 251 ページの『すべてのデータベース変更レコードを累積する』
- 252 ページの『古いデータ・セットを更新せずにデータベース・ログを累積する』

### データベース・ログを累積する

以下のコード・サンプルでは、2 つのデータベース・ログが累積されます。更新する古い累積変更データ・セットはありません。最初のデータベース (DI32DB01) はその全体が累積され、2007 年の 175 日目以前で、12 時以前のレコードがすべて除去されます。2 番目のデータベース (DI32DB02) は選択的に累積されます。変更累積データ・セットは、RECON に記録されます。

DD 名が DDI3IA のデータベース (DI32DB02) データ・セットは累積され、2007 年の 173 日目以前で、15 時以前のすべてのレコードが除去されます。DD 名が DDI30A のデータベース (DI32DB02) データ・セットは、そのレコードが新しいログ・データ・セットに書き込まれ、2007 年の 173 日目以前で、15 時以前のすべてのレコードが除去されます。他のレコードはすべて新しいログ・データ・セットに書き込まれます。

```
//STEP1 EXEC PGM=DFSUCUM0,PARM='CORE=512000,DBRC=Y'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSOUT DD SYSOUT=A
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK05 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK06 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//DFSUCUM0 DD DUMMY,DCB=BLKSIZE=100
//DFSUCUMN DD DSN=IMS.CUM1,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=CUMTAP
//DFSUDD1 DD DSN=IMS.NEWLOG,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=LOGTAP
//DFSULOG DD DSN=IMS.LOG1,DISP=OLD,
// UNIT=TAPE,VOL=SER=LTAPE1
// DD DSN=IMS.LOG2,DISP=OLD,
// UNIT=TAPE,VOL=SER=LTAPE2
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7---

//SYSIN DD *
DB0DI32DB01071751200000
DB0DI32DB02071731500000 DDI3IA
DB1DI32DB02071731500000 DDI30A
DB1 *OTHER
/*
```

## すべてのデータベース変更レコードを累積する

この例では、すべてのデータベース変更レコードが累積されます。ルート・セグメントの最大シーケンス・フィールド長が 4 バイトと指定されているのは、すべてのログ・レコードが HD タイプの編成を反映しているためです。VSAM KSDS タイプの変更レコードはありません。DBO 制御ステートメントは、すべてのレコードを累積し、2007 年の 200 日目以前のレコードをすべて除去することを指定します。古い変更累積データ・セットは新しい変更累積データ・セットとマージされます。除去の日付は古い累積データ・セットに適用されます。DFSUDD1 (新しいログ出力データ・セット) が DUMMY と定義されているのは、DB1 制御ステートメントが指定されていないためです。

```
//STEP1 EXEC PGM=DFSUCUM0,PARM='CORE=512000'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSOUT DD SYSOUT=A
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK05 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK06 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//DFSUCUM0 DD DSN=IMS.CUM1,DISP=OLD,
```

```

//          UNIT=TAPE,VOL=SER=CUMTAP
//DFSUCUMN DD DSN=IMS.CUM2,DISP=(NEW,KEEP),
//          UNIT=TAPE,VOL=SER=CUMTP2
//DFSUDD1  DD DUMMY
//DFSULOG  DD DSN=IMS.LOG1,DISP=OLD.
//          UNIT=TAPE,VOL=SER=LTAPE3
//          DD DSN=IMS.LOG2,DISP=OLD,
//          UNIT=TAPE,VOL=SER=LTAPE4
// * +----1----+----2----+----3----+----4----+----5----+----6----+----7---

//SYSIN     DD *
ID                               004
DB0 *ALL    072000000000
/*

```

## 古いデータ・セットを更新せずにデータベース・ログを累積する

この例 (例 2 と類似) では、2 つのデータベース・ログが累積されます。更新する古い累積変更データ・セットはありません。最初のデータベース (DI32DB01) はその全体が累積され、2007 年の 175 日目以前で、12 時以前のレコードがすべて除去されます。2 番目のデータベース (DI32DB02) は選択的に累積されます。

DD 名が DDI3IA のデータベース (DI32DB02) データ・セットは累積され、2007 年の 173 日目以前で、15 時以前のすべてのレコードが除去されます。

DD 名が DDI3OA のデータベース (DI32DB02) データ・セットは、そのレコードが新しいログ・データ・セットに書き込まれ、2007 年の 173 日目以前で、15 時以前のすべてのレコードが除去されます。

データベース (DI32DB01) データ・セット 001 は、ソートされ再フォーマット設定されたすべてのレコード (RBN が 100 となっている) がソート出力データ・セットに書き込まれます。

データベース名が DI32DB01、データ・セット ID が 001、および RBA が 100 の SORT ルーチンに渡されたすべてのログ・レコードが IMS.NEWLOG2 に書き込まれます。

他のレコードはすべて新しいログ・データ・セット IMS.NEWLOG1 に書き込まれ、このデータ・セットは診断の目的だけに使用されます。

```

//STEP1 EXEC PGM=DFSUCUM0,PARM='CORE=512000'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS     DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSOUT  DD SYSOUT=A
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK05 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//SORTWK06 DD UNIT=SYSDA,SPACE=(CYL,(2),,CONTIG)
//DFSUCUM0 DD DUMMY,DCB=BLKSIZE=100
//DFSUCUMN DD DSN=IMS.CUM1,DISP=(NEW,KEEP),
//          UNIT=TAPE,VOL=SER=CUMTAP
//DFSUDD1  DD DSN=IMS.NEWLOG1,DISP=(NEW,KEEP),
//          UNIT=TAPE,VOL=SER=LOGTAP
//DFSUDD2  DD DSN=IMS.NEWLOG2,DISP=(NEW,KEEP),
//          UNIT=TAPE,VOL=SER=LOGTAP
//DFSULOG  DD DSN=IMS.LOG1,DISP=OLD,

```



```
//          UNIT=TAPE,VOL=SER=LTAPE1
//          DD DSN=IMS.LOG2,DISP=OLD,
//          UNIT=TAPE,VOL=SER=LTAPE2
//* +---1---+---2---+---3---+---4---+---5---+---6---+---7---

//SYSIN      DD *
DB0DI32DB01071751200
DB0DI32DB02071731500000          DDI3IA
DB1DI32DB022007:173 15:15:30.2#8  DDI30A
DB1 *OTHER
SO DI32DB01001 0000000100
/*
```



---

## 第 17 章 データベース・リカバリー・ユーティリティー (DFSURDB0)

データベース・リカバリー・ユーティリティーは、IMS データベースの中の物理的に損傷を受けたデータ・セットをリカバリーするために使用します。データ・セットは実行ごとに 1 つだけがリカバリーされます。

このユーティリティーでは、データはアプリケーション論理エラーからリカバリーされません。データベース内のデータの保全性を保つのはユーザーの役割です。

このユーティリティーによるリカバリーでは、データベースのコピーの作成後にログに記録された変更を反映して、そのコピーが更新されます。

データベース・リカバリー・ユーティリティーは、特殊な IMS バッチ領域で実行されます。そのため、データベース・リカバリーは IMS システムから独立して行うことができます。バッチ領域は、ユーティリティー EXEC ステートメントの PARM キーワードに UDR として定義されます。

データベース・リカバリー・ユーティリティーは、ほとんどの動作環境でのリカバリーをサポートします。以下に例を示します。

- リモート・サイト・リカバリー (RSR) 構成
- 統合 HALDB オンライン再編成機能を使用する IMS システム

RSR トラッキング・サイトで、データベース・リカバリー・ユーティリティーは、アクティブ・サイトのデータベースがリカバリーされるのと同じリカバリー・ポイントまでデータベースをリカバリーします。

RSR データベース・レベル・トラッキングを使用する場合、データベース・リカバリー・ユーティリティーが稼働を停止した後、RSR 機能オンライン・フォワード・リカバリーがリカバリー・プロセスを完了します。

統合 HALDB オンライン再編成機能を使用するシステムでは、データベース・リカバリー・ユーティリティーを使用して、入力および出力データ・セットをリカバリーすることができます。

以下の図は、データベース・リカバリー・ユーティリティーのフロー・ダイアグラムです。

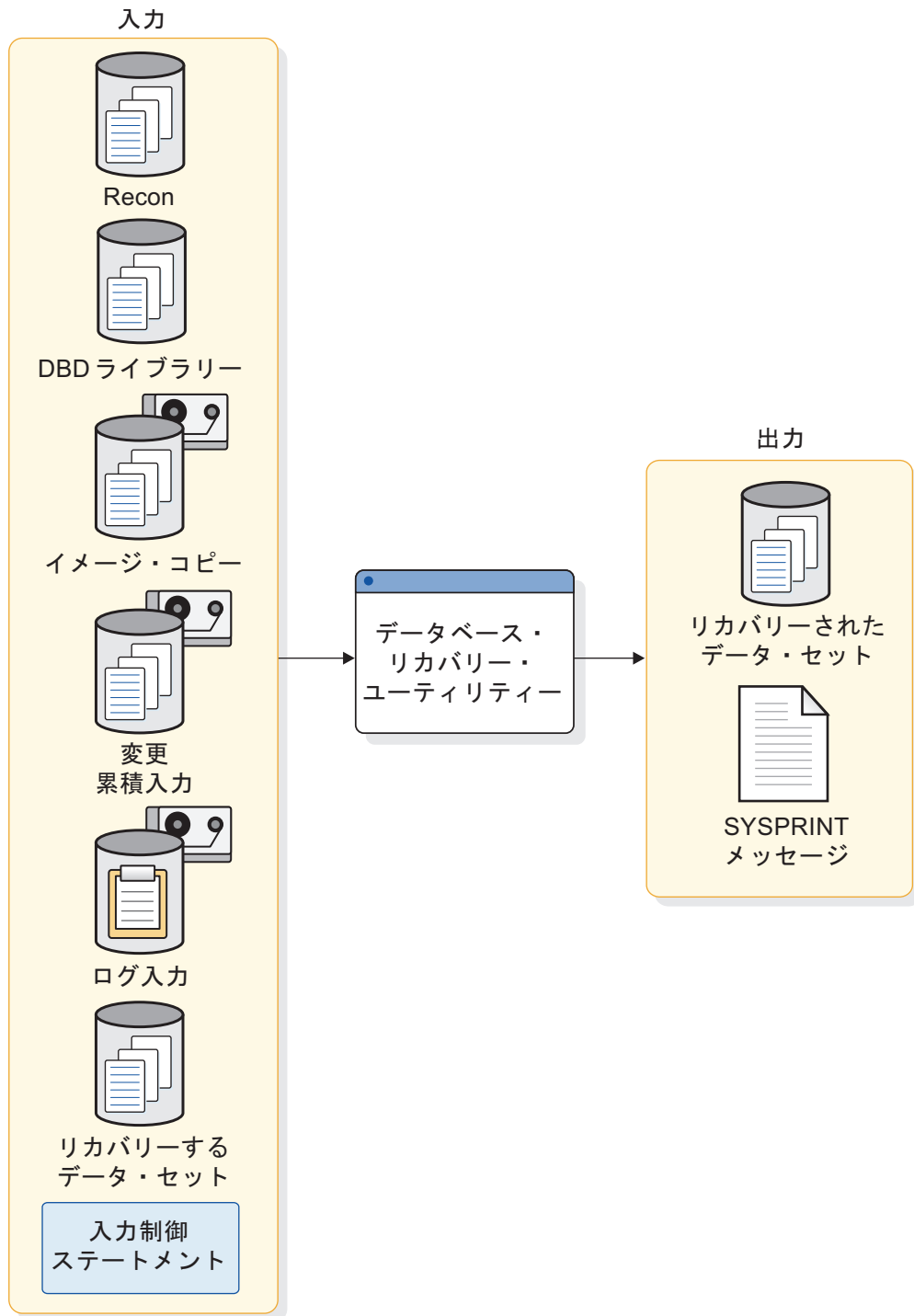


図 60. 데이터베이스·리카바리·유티리티어

サブセクション:

- 257 ページの『制約事項』
- 257 ページの『前提条件』
- 257 ページの『要件』
- 260 ページの『推奨事項』
- 260 ページの『入出力』

- 266 ページの『戻りコード』
- 263 ページの『JCL 指定』

## 制約事項

データベース・リカバリー・ユーティリティーを実行するときは、次のような制約があります。

- データベース・リカバリー・ユーティリティーは、HSAM または GSAM データベースでは使用できません。
- フォワード・リカバリーを行う場合に HD 再編成再ロード・ユーティリティーによってイメージ・コピーの代用として再ロードされたデータ・セットは使用できません。セグメントの物理位置は再ロード時に変わる可能性があります。その場合、データベース・リカバリー・ユーティリティーがリカバリーで使用するログに記録された位置とセグメントの位置は一致しません。
- ログ・マージ・ユーティリティーからの出力をリカバリーへの入力に使用しないでください。
- 非標準イメージ・コピー・データ・セットは、データベース・リカバリー・ユーティリティーへの入力として使用できません。非標準イメージ・コピーを使用してデータベースをリカバリーする手順は、IMS システムがアクティブなサブシステムであるか、トラッキング・サブシステムであるかによって多少異なります。
- データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。

## 前提条件

データベース・リカバリー・ユーティリティーへの入力として変更累積データ・セットを使用する場合は、データベース・リカバリー・ユーティリティーを実行する前に変更累積ユーティリティー (DFSUCUM0) を実行します。

イメージ・コピーをデータベース・リカバリー・ユーティリティーに含めない場合、データベース・リカバリー・ユーティリティーを実行する前にデータベースのバックアップ・コピーを復元する必要があります。

DEDB 多重エリア・データ・セット (MADS) をリカバリーする場合、MADS を使用不可にして、さらに「リカバリーが必要」に設定する必要があります。DBRC コマンド CHANGE.ADS を使用することによって、MADS を使用不可にできます。以下に例を示します。CHANGE.ADS ADDN(ADDN004) AREA(AREA002) DBD(DBD001) UNAVAIL

## 要件

データベース・リカバリー・ユーティリティーを実行するときは、次のような要件があります。

### 一般要件

- 並行イメージ・コピー (CIC) が使用される場合、CIC の最小 DBRC 共用レベルは 1 です。CIC を実行するときは、共用レベルを指定しなければなりません。

- イメージ・コピー 2 ユーティリティで作成したイメージ・コピーを入力として使用するとき、DBRC をアクティブにしておいてデータベース・リカバリー・ユーティリティを実行しなければなりません。
- 共用副次索引をリカバリーする場合は、共用副次索引の DBD ステートメントの DBDNAME パラメーターにリストされた最初の索引名のみを指定します。データベース・リカバリー・ユーティリティは、共用副次索引のすべての索引のデータ・セットをリカバリーします。
- HISAM アンロード・データ・セットをデータベース・リカバリー・ユーティリティへの DFSUDUMP データ・セット入力として使用するには、データベースをアンロードの直後に再ロードしなければなりません。これは、データベース・リカバリー・ユーティリティが、HISAM アンロード操作とリカバリー操作の間に作成されるいずれのデータベース更新ログ・レコードからも影響を受けなくするための手順です。この手順が必要な理由は、HISAM アンロード・データ・セットに、再編成された形式のデータベースが含まれるためです。HISAM アンロード操作と再ロード操作の間に作成されるログ・レコードは、再編成前のデータベースの物理的な状態を示しています。

重要: 以前の状態のデータベースを使用して作成されたログ・レコードを HISAM アンロード・データ・セットと組み合わせて使用すると、アンロード・データ・セットの方には新しい状態が反映されているため、データの整合性が損なわれる可能性があります。アンロードしたデータ・セットを再ロードすると、そのデータ・セットはアンロード・データ・セットにある新しく再編成されたデータベースの状態にマップされます。HISAM 再ロード操作後に作成されるすべてのデータベース・ログ・レコードは、新しく編成されたデータベースの状態を反映するため、データベース・リカバリー・ユーティリティが安全に使用できます。変更累積プロセスを使用する場合は、再ロード後に新しいジョブを開始してください。これにより、新しいジョブを開始すると、最新の再ロード前に生成されたすべての変更累積レコードは不整合を起こす危険がなくなります。

- このユーティリティを RECON データ・セットに登録されていない IMS カタログ・データベースで使用するには、追加の EXEC パラメーター DFSDF を指定する必要があります。DFSDF= は、未登録の IMS カタログ・データベースの名前を含む IMS.PROCLIB データ・セットの DFSDFxxx メンバーの、3 文字からなる接尾部を指定します。それらの名前は、DATABASE ステートメントの UNREGCATLG パラメーターを用いて指定されます。例えば、次のようにします。

```
//RCVA EXEC PGM=DFSRR00,
// PARM=(UDR,DFSURDB0,DFSCD000,,,,,,,,,N,N,,,,,,,,,,,,,,,,,,,,,
// 'DFSDF=CAT')
```

3 文字からなる接尾部 CAT を、未登録の IMS カタログ・データベースの名前を指定する DFSDFxxx メンバーの接尾部で置き換えてください。

#### 高速機能 DEDB 要件

- 高速機能 DEDB エリア・データ・セット (ADS) を DBRC を使用して RECON データ・セットに登録する場合、以下の要件が適用されます。

- データベース・リカバリー・ユーティリティーによってリカバリーされる ADS を示す dataset1 DD ステートメントでは、ADS の RECON データ・セットに登録される DD 名とデータ・セット名を指定する必要があります。
- ターゲット ADS は、RECON に「リカバリーが必要」とマークされていなければなりません。
- RECON データ・セットに登録されていない高速機能 DEDB ADS をリカバリーする場合、RECON データ・セットに NOFORCE 属性が指定されていないときは、dataset1 DD ステートメントに指定された DD 名がターゲット区域名に一致しなければなりません。
- RECON データ・セットに登録されていない高速機能 DEDB ADS をリカバリーする場合、RECON データ・セットに FORCER 属性が指定されているときは、データベース・リカバリー・ユーティリティーはエラー・メッセージを出力して終了します。
- DEDB 多重エリア・データ・セット (MADS) をリカバリーする場合、DEDB エリア・データ・セット作成ユーティリティーを使用して、リカバリーされた MADS の追加のコピーを作成する必要があります。

#### 必要なデータ・セット

- OSAM データ・セットをリカバリーするためには、リカバリーの実行前に、OSAM データ・セットを削除した後に再割り振りを行う必要があります。

以下の例では、IDCAMS を使用して OSAM データ・セットの削除と再割り振りを行う方法を示しています。

```

IF MAXCC = 0 -
THEN DO
  DELETE (IMSVS.IMSIVP.DFSIVD1) NONVSAM
  IF LASTCC NE 0 -
  THEN SET MAXCC = 0
  ELSE
  ALLOCATE -
    DSNAME('IMSVS.IMSIVP.DFSIVD1') -
    FILE(DFSIVD1) -
    RECFM(F,B,S) -
    LRECL(2048) -
    BLKSIZE(2048) -
    DSORG(PS) -
    NEW CATALOG -
    SPACE(15645 15645) -
    MAXVOL(28) -
    UNIT(SYSDA)
  END
ELSE

```

**推奨事項:** リカバリー・ジョブ・ステップの前のジョブ・ステップで、z/OS IDCAMS コマンドを使用して、マルチボリューム OSAM データ・セットの削除と割り振りを行います。削除と割り振りが行われない場合、これらのデータ・セットは、リカバリー・ジョブ・ステップで DISP=NEW を指定して割り振られます。データが入っている既存の OSAM データ・セットを使用して全体のリカバリーを実行しようとすると、予測不能な結果を引き起こすことになります。

- 全体のリカバリーは、有効なイメージ・コピー、変更累積、およびログ・レコード・データを使用するリカバリーです。このステートメントは、イメージ・コピーなしの USEDDBDS または順方向リカバリーには関係ありません。
- イメージ・コピー 2 で作成された高速複製コピーからの復元を意図した VSAM データ・セットのリカバリーでは、データベース・リカバリー・ユーティリティを実行する前にその VSAM データ・セットを削除し、再定義する必要があります。VSAM データ・セットをリカバリーするときには、VSAM 通知メッセージ IEC161I 072-053 (DATA SET WAS EMPTY) が返されるのが普通です。

**重要:** イメージ・コピーがユーティリティへの入力として含まれない場合のフォワード・リカバリーを除いて、データベース・リカバリー・ユーティリティは入力データ・セットが空であると予想します。データ・セットが空でない場合は、リカバリーが失敗し、IMS はメッセージ DFS0792A を発行します。

## 推奨事項

データベース・リカバリー・ユーティリティを実行する場合は DBRC を使用し、DBRC GENJCL コマンドによってリカバリー JCL を生成します。DBRC は、データベース・リカバリー・ユーティリティで使用する場合、以下の操作を行います。

- GENJCL コマンドを使用する場合、ユーティリティに対して正しい JCL を生成します。
- リカバリーで使用する正しい協定世界時 (UTC) オフセットを計算します。DBRC を使用しない場合、z/OS UTC オフセットが使用されます。
- リカバリーの入力として使用する正しいイメージ・コピーまたはデータ・セットを選択します。
- 入力として使用する正しい変更累積データ・セットを選択します。
- リカバリーの入力として使用する正しいログを選択します。
- リカバリーで使用する正しい除去時刻を計算します。
- DBRC が生成した JCL の実行時に上記のすべての入力の正確さを検証します。

データベースまたは区域が静止している場合は、QUIESCE フラグが設定された RECON データ・セットの、そのデータベースまたは区域の割り振りレコードに DEALLOC タイム・スタンプが記録され、そのタイム・スタンプは LIST.DB コマンド、LIST.DBDS コマンド、または QUERY API 要求を使用して取得できます。このユーティリティは、このタイム・スタンプを使用してデータベースのリカバリーを行うことができます。

## 入出力

データベース・リカバリー・ユーティリティ (DFSURDB0) の入力には、イメージ・コピー、変更累積データ・セット、およびログを含めることができます。ユーティリティの EXEC ステートメントで DBRC=Y を指定すると、ユーティリティの入力に対する JCL の指定が正しいかどうかを、DBRC が検査します。DBRC が指定が不正確であると判断すると、データベース・リカバリー・ユーティリティは終了します。



JCL の指定がすべて正しいことを確認するには、DBRC コマンド GENJCL.RECOV を使用して、データベース・リカバリ・ユーティリティ用の JCL を生成してください。

イメージ・コピーを以前のジョブでデータベースに適用した場合など、データベース・リカバリ・ユーティリティの入力としてイメージ・コピーを含めない場合、データベース・リカバリ・ユーティリティはリカバリ対象のデータ・セットを入力として使用します。

現在の実行時または前の実行時に障害が起こったため、ターゲット・データ・セットがオープンしないときは、そのデータ・セットを消去し、再割り振りしてからこのユーティリティを再実行してください。

要確認: VSAM データ・セットをリカバリするとき、VSAM 通知メッセージ IEC161I 072-053 が出されるのが普通です。

データベース・リカバリ・ユーティリティの入出力は、以下の表に示すとおりです。

表 11. データベース・リカバリ・ユーティリティの入出力

入力	出力
RECON データ・セット	SYSPRINT メッセージ
DBD ライブラリー	リカバリされたデータ・セット
イメージ・コピー	
変更累積入力	
ログ	
リカバリされるデータ・セット	
入力制御ステートメント	

## ログ

複数のログは、ログを日付と時刻順に連結して提供することができます。

以下の変更記録のみがデータベース・リカバリに対する有効な入力です。

- 正常実行時に IMS によって作成された SLDS または RLDS
- データベース変更累積ユーティリティ (DFSUCUM0) によって作成されたログ上の変更の累積
- SLDS とデータベース変更累積ユーティリティによって作成されたログの両方の結合 (SLDS に入っている変更は、変更累積ログに入っている変更よりも新しくなければなりません)

## イメージ・コピー

データベース・リカバリ・ユーティリティに与えられるデータベースのコピーには、次のものがあります。

- データベース・イメージ・コピー・ユーティリティ (DFSUDMP0) またはオンライン・データベース・イメージ・コピー・ユーティリティ (DFSUICP0) により作成されたイメージ・コピー。

- データベース・イメージ・コピー 2 ユーティリティ (DFSUDMT0) の並行コピー機能または高速複製機能によって作成されたイメージ・コピー。DFSUDMT0によって作成されたイメージ・コピーを使用する場合、データベース・データ・セットは DBRC を用いて登録しておきます。
- HISAM 再編成アンロード・ユーティリティ (DFSURUL0) によって作成された HISAM アンロード・データ・セット。

デュアル・イメージ・コピー・データ・セットが RECON データ・セットにリストされる場合は、データベース・リカバリー・ユーティリティは、INVALID とマークされている場合を除き、最初のイメージ・コピー・データ・セットを使用します。

z/OS コピー・ユーティリティで作成するイメージ・コピーなど、データベースの非標準イメージ・コピーをリカバリーで使用する場合、その非標準イメージ・コピーでデータベースを復元し、DBRC に非標準イメージ・コピーを通知してから、データベース・リカバリー・ユーティリティを実行します。非標準イメージ・コピーからリカバリーする手順は、クリーン非標準イメージ・コピーを使用しているか、ファジー非標準イメージ・コピーを使用しているかによって異なります。

クリーン非標準イメージ・コピーを使用している場合、以下の処理をリカバリーで実行します。

1. クリーン非標準イメージ・コピーからデータベースを復元します。
2. DBRC コマンドの NOTIFY.RECOV を発行して、リカバリーについての情報を RECON データ・セットに追加します。
3. ユーティリティ制御ステートメントをコーディングする場合、または USEDBDS パラメーター付きで DBRC コマンドの GENJCL.RECOV ステートメントを発行する場合、DFSUDUMP DD ステートメントに DUMMY または NULLFILE を指定します。
4. データベース・リカバリー・ユーティリティを実行することによって、ログからデータベースの変更、変更累積データ・セット、またはその両方を復元済みの非標準イメージ・コピーに適用します。

並行 (またはファジー) 非標準イメージ・コピーを使用している場合、リカバリーで以下の処理を実行します。

- RECOV パラメーター付きで CHANGE.DBDS コマンドを実行し、リカバリーが進行中であることを DBRC に通知します。
- ファジー非標準イメージ・コピーからデータベースを復元します。
- データベース・リカバリー・ユーティリティ制御ステートメントを手動でコーディングする場合、以下を指定します。
  - DFSUDUMP DD ステートメントに DUMMY または NULLFILE を指定します。
  - 並行ユーザー・イメージ・コピーがリカバリーで使用されたことを示すために 64 桁目に M を指定します。USER CIC を実行してからリカバリーを行う場合、M は 64 桁目ではなく 63 桁目に指定します。
  - リカバリーで特定のランタイムを指定する場合、ランタイム指定制御ステートメントを指定します。

- DBRC コマンドの GENJCL.RECOV を使用してデータベース・リカバリー・ユーティリティを生成する場合、以下を指定します。
  - DFSUDUMP DD ステートメントに DUMMY または NULLFILE を指定します。
  - 特定のランタイムをリカバリーに指定するには、USERIC パラメーターを指定します。
  - 最新の並行ユーザー・イメージ・コピーを使用するには、LASTUIC パラメーターを指定します。
- データベース・リカバリー・ユーティリティを実行することによって、ログからデータベースの変更、変更累積データ・セット、またはその両方を復元済みの非標準イメージ・コピーに適用します。

データベース・データ・セットのバックアップをとるのにオンライン・データベース・イメージ・コピー・ユーティリティを使用する場合は、変更を、イメージ・コピーと並行して行うだけでなく、イメージ・コピーの後でも行う必要があります。イメージ・コピーには、バッチ・データベース・イメージ・コピー・ユーティリティまたはオンライン・データベース・イメージ・コピー・ユーティリティのいずれかで作成されたデータ・セットが使用できます。HISAM の場合、イメージ・コピーには、HISAM 再編成アンロード・ユーティリティによって作成された再編成データ・セットが使用できます。オンライン・データベース・イメージ・コピー・ユーティリティは、適用可能な変更が入っている可能性のある最初のログ・テープのボリューム通し番号を SYSOUT に出力します。

ロギングしないでデータベースを更新するバッチ・ジョブを実行した直後にイメージ・コピーをとってください。こうすると、リカバリーが必要な場合に、データベースの保全性が維持されます。ログ・テープを使用すると、バッチ・ジョブの開始までリカバリーできるため、そこでバッチ・ジョブを再処理することができます。ただし、その結果のデータベースは、前のバッチ実行後のデータベースと論理的には同一であっても、ビット単位では同一でない場合があります。バッチ・ジョブは反復可能でない場合があるため、そのことを前提としてください。データベースがビット単位で同一でないと、前のバッチ・ジョブの実行後に作られたログ・テープは、再処理後は有効でなくなります。したがって、リカバリー処理は、イメージ・コピーで開始し、ログ・テープを適用し、ログがとられなかったバッチ実行を再処理し、その後で別のログ・テープを適用するという順序で行うことはできません。

## JCL 指定

データベース・リカバリー・ユーティリティは、標準の z/OS ジョブとして実行されます。次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

このステートメントは、必要なステートメントが入っているカタログ式プロシージャを呼び出すことができますが、次の形式にすることもできます。

```
//STEP EXEC PGM=DFSRR00,PARM='UDR,DFSURDB0,dbdname'
```

## UDR

リカバリー領域を指定します。

## dbdname

リカバリーするデータ・セットを含む (DBD ステートメントで定義された) データベースの名前。HALDB データベースでは、これはマスター・データベースの名前です。

BUF、SPIE といった通常の IMS 定位置パラメーターは、*dbdname* の後に置くことができます。

## DD ステートメント

### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

### IMS DD

リカバリーするデータベース・データ・セットを記述する DBD が入っているライブラリーを定義します。これは DSN=IMS.DBDLIB であるのが普通です。このデータ・セットは直接アクセス・ボリュームに置いておかなければなりません。

### SYSPRINT DD

出力メッセージ・データ・セットを定義します。このデータ・セットは、テープ、直接アクセス・ボリューム、またはプリンターに置いておくことも、出力ストリーム (SYSOUT) に入れて送ることもできます。SYSPRINT はブロック化できますが、121 の倍数にしなければなりません。

### SYSIN DD

入力制御データ・セットを定義します。これは、テープまたは直接アクセス・ボリュームに置いておくことも、入力ストリーム (DD \* または DD DATA) に入れて送ることもできます。

### DFSUDUMP DD

リカバリーに使用するイメージ・コピー入力データ・セットがある場合は、これを定義します。これには、バッチ・データベース・イメージ・コピー・ユーティリティー、オンライン・データベース・イメージ・コピー・ユーティリティー、データベース・イメージ・コピー 2 ユーティリティー、または HISAM 再編成アンロード・ユーティリティーのいずれかで作成されたデータ・セットが使用できます。このデータ・セットはテープまたは直接アクセス・ボリュームに置いておくことができます。

このステートメントの DD 名は、DFSUDUMP とは別のものにすることができます。DD 名が DFSUDUMP でないときは、ユーティリティー制御ステートメントの 22 桁目にもこの DD 名を入れておかなければなりません。

以下の条件のいずれかが満たされる場合、DD ステートメントは DUMMY になります。

- イメージ・コピーまたは HISAM アンロード・コピー入力提供されない場合
- HALDB オンライン再編成をリカバリーの開始点として使用する場合
- USEDBDS または USEAREA キーワードを GENJCL.RECOV コマンドに指定する場合

#### **DFSUCUM DD**

累積変更入力データ・セットを定義します。定義する累積変更入力がない場合は、このステートメントを DD DUMMY とコーディングする必要があります。このデータ・セットはテープまたは直接アクセス・ボリュームに置いておくことができます。

#### **DFSULOG DD**

ログ変更入力を定義します。適用するログ変更がない場合は、このステートメントを DD DUMMY とコーディングする必要があります。このデータ・セットはテープまたは直接アクセス・ボリュームに置いておくことができます。

データ・セットを連結すると、複数のログを入力として使用できます。そのためには、DD ステートメントを日付と時刻順にする必要があります。

#### **DFSUSNAP DD**

SNAP 出力データ・セットを定義します。これは、該当する場合に、診断の目的で特定のストレージ域を書き込むために使用されます。このデータ・セットは SYSOUT=\* と定義することも、属性 DCB=(RECFM=VBA,LRECL=125,BLKSIZE=1632) を使って定義することもできます。

#### **dataset1 DD**

リカバリーするデータ・セットを定義します。DD 名は、このデータ・セットを記述する DBD 内の名前と同じでなければなりません。dataset1 DD ステートメントが含まれる場合、dataset1 で表される値は DFSURDB0 ユーティリティ制御ステートメントの 13 から 20 桁目に指定される DD 名と一致しなければなりません。

DEDB のときは、この DD ステートメントは、リカバリーする区域のエリア・データ・セットを定義し、DD 名は、その区域を記述している DBD にあるものと同じでなければなりません。HALDB のときは、この DD ステートメントは、リカバリーする区画の区画データ・セットを定義します。DD 名は、区画定義ユーティリティで定義された区画名でなければなりません。

区域が DBRC RECON データ・セットに登録されているときは、DD 名と DS 名は、ターゲット区域の ADS リストに登録されている名前と一致していなければなりません。DBRC RECON データ・セットに区域が登録されていないで、DBRC RECON データ・セットが NOFORCER 属性を持っている場合、DD 名は、その区域名と同じものでなければならず、ユーティリティ制御ステートメントに指定されていなければなりません。区域に複数のエリア・データ・セット (MADS) があり、データベース・コピー 2 ユーティリティによって作成された SMSNOCIC、SMSCIC、SMSONLC、または SMSOFFLC イメージ・コピーがリカバリーへの入力として使用されている場合、すべてのエリア・データ・セットの DD ステートメントを含める必要があります。リカバリーされるエリア・データ・セットは、イメージ・コピー・データ・セットの内容から判別されます。

### DFSVSAMP DD

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。この DD ステートメントは次の場合に必要です。

- 変更累積入力だけが使用される場合
- ログ入力を使用される場合
- HISAM アンロードからのデータを入力として VSAM ESDS をリカバリーする場合
- ヌル・イメージ・コピー・データ・セットを入力としてリカバリーを行う場合

このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリーム (DD \* または DD DATA) に入れて送ることもできます。

### SYSABEND DD または SYSUDUMP DD

ダンプ・データ・セットを定義します。これらの DD ステートメントはオプションです。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

### RECON1 DD

最初の DBRC RECON データ・セットを定義します。この RECON1 データ・セットは、制御領域が使用しているものと同じ RECON1 データ・セットでなければなりません。

### RECON2 DD

2 番目の DBRC RECON データ・セットを定義します。この RECON2 データ・セットは、制御領域が使用しているものと同じ RECON2 データ・セットでなければなりません。

### RECON3 DD

3 番目の DBRC RECON データ・セットを定義します。この RECON3 データ・セットは、制御領域が使用しているものと同じ RECON3 データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

### DFSVDUMP DD

DFSVDUMP DD ステートメントは、DUMMY として生成されます。

## 戻りコード

データベース・リカバリー・ユーティリティからは、次のような戻りコードが返されます。

コード 意味

- 0 すべての操作が正常に完了した
- 4 警告メッセージが出された
- 8 複数の操作が正常に完了しなかった
- 16 すべての操作を完了する前にジョブを終了させる重大エラーが起こった

これらの戻りコードは、後続のジョブ・ステップの EXEC ステートメントで COND= パラメーターを指定すると、テストすることができます。

関連概念:

- ➡ IMS バッファ・プール (システム定義)
- ➡ 操作タスクの理解 (システム管理)
- ➡ DBRC による共用レベルの割り当て (システム管理)

関連タスク:

- ➡ データベース・リカバリー戦略のプラン (データベース管理)

---

## DFSURDB0 ユーティリティーの制御ステートメント

データベース・リカバリー・ユーティリティーでは、ABEND と NOSEQCK の 2 つのユーティリティー制御ステートメントを使用します。これらのステートメントは別々に使用することもできるし、一緒に使用することもできます。

サブセクション:

- 『ABEND ステートメント』
- 『NOSEQCK ステートメント』
- 268 ページの『データベース・リカバリー・ステートメント』
- 270 ページの『非標準並行イメージ・コピーのランタイム指定ステートメント』

### ABEND ステートメント

ABEND ユーティリティー制御ステートメントは、異常条件が現れたときに、ユーザー ABEND 302 を出してユーティリティーを終了させることを要求するために使用します。SYSUDUMP DD ステートメントが指定されていると、ストレージ・ダンプがとられます。このステートメントの指定がない場合、データベース・リカバリー・ユーティリティーは、なんらかの異常条件が現れたときにエラー・メッセージを出しますが、処理を続けます。

この制御ステートメントは、データベース・リカバリー・ユーティリティー制御ステートメントの前に置いておかなければなりません。

この制御ステートメントの形式では、ABEND を 1 桁目からコーディングします。

### NOSEQCK ステートメント

NOSEQCK ユーティリティー制御ステートメントは、ユーティリティーが入力ログのシーケンス検査を行わないよう要求するために使用します。

**重要:** このステートメントを使用するときは、ログが正しい順序になっていないとリカバリーできない場合があるので、注意が必要です。

この制御ステートメントは、データベース・リカバリー・ユーティリティー制御ステートメントの前に置いておかなければなりません。

この制御ステートメントの形式では、NOSEQCK を 1 桁目からコーディングします。

## データベース・リカバリー・ステートメント

データベース・リカバリー・ユーティリティ制御ステートメントは以下の桁位置を使用する固定形式です。

位置 説明

1 ステートメント ID

データベース・リカバリー・ユーティリティ制御ステートメントの ID。これは、文字「S」でなければなりません。

2 この桁はブランクにしておきます。

3 この桁はブランクにしておきます。

4-11 dbdname

リカバリーするデータ・セットが入っているデータベースを記述する DBD の名前。HALDB データベースでは、DD 名はリカバリー対象のデータ・セットが入っている区画の名前にします。EXEC ステートメントの PARM フィールドに DBD 名を入力する場合、その DBD 名はデータベース・リカバリー・ユーティリティ制御ステートメント内の DBD 名に一致する必要があります。

12 この桁はブランクにしておきます。

13-20 データ・セットまたは区域の DD 名。

- リカバリーするデータ・セットまたは区域名の DD 名。これは、DBD および *dataset1* DD ステートメントの DD 名と同じものでなければなりません。

リカバリーへの入力としてデータ・セット名を指定するために、DBRC はアクティブでなければなりません。

高速複製イメージ・コピー以外のイメージ・コピーからのリカバリーでは、JCL ストリームは、リカバリーする DBDS のデータ・セット名をコーディングする必要がある *dataset1* DD ステートメント (データ・セットの DD 名に対応するステートメント) を含んでいなければなりません。

高速複製イメージ・コピーからのリカバリーでは、対応する *dataset 1* DD ステートメントはオプションです。この *dataset1* DD ステートメントが含まれている場合、ターゲット・データ・セットまたは区域名は、この DD ステートメントによって与えられます。これが含まれていない場合、ターゲット・データ・セットまたは区域名は DBRC によって与えられます。

21 ブランク

22-29 入力の DD 名

イメージ・コピー入力として使用されるデータ・セットの DD 名。このフィールドをブランクにすると、デフォルトの DD 名「DFSUDUMP」が指定されたものと見なされます。

30 ブランク



**31-61** RCVTIME パラメーターが GENJCL.RECOV コマンドの入力である場合に指定されるタイム・スタンプ。それ以外の場合、これらの桁位置は空白になります。制御ステートメントのこれらの桁はコメントとして処理され、DBRC をオフにしてユーティリティを実行すると無視されます。

位置 説明

**31-47** この桁は、日付と時刻を yydddhhmmssthmiju の形式で指定します。ここで、yy= 年、ddd= 日付、hh= 時、mm= 分、ss= 秒、thmiju= 100 万分の 1 秒です。

**48** この桁位置は、オフセットの符号 + または - を指定します。

**49-52** この桁位置は、UTC からのオフセットを HHMM の形式で指定します。

**53-61** ブランク

または、その代わりに:

位置 説明

**31-52** この桁は、句読点で区切られたタイム・スタンプを yy.ddd hh:mm:ss.thmiju の形式で指定します。ここで、yy= 年、ddd= 年間通算日、hh= 時、mm= 分、ss= 秒、thmiju= 100 万分の 1 秒です。

**53** ブランク

**54** この桁位置は、オフセットの符号 + または - を指定します。

**55-59** この桁位置は、UTC からのオフセットを HH:MM の形式で指定します。

**60-61** ブランク

または、その代わりに:

位置 説明

**31-54** この桁は、句読点で区切られたタイム・スタンプを (年を 4 桁で表示) を yyyy.ddd hh:mm:ss.thmiju の形式で指定します。ここで、yyyy= 年、ddd= 年間通算日、hh= 時、mm= 分、ss= 秒、thmiju= 100 万分の 1 秒です。

**55** ブランク

**56** この桁位置は、オフセットの符号 + または - を指定します。

**57-61** このオプションの桁位置は、UTC からのオフセットを HH:MM の形式で指定します。この桁位置を省略すると、デフォルトは現在の z/OS オフセット値 (CVTLDTO) から導き出されます。

**62** ブランク

**63** リカバリー・タイプ。有効な値は以下のとおりです。

**C** イメージ・コピー・データ・セットがこのリカバリーに使用されていないことを示します。

- V RECEIVE タイプ・リカバリーが RSR トラッキング・サイトで実行中であることを示します。
- M ユーザー並行イメージ・コピーがリカバリーに使用されていることを示します。

64 データ・セット情報の提供方法を示します。有効な値は次のとおりです。

- D 完全修飾データ・セット名が DBRC によって提供されており、DD ステートメントがジョブ・ストリームで不必要であることを示します。D 仕様は、GENJCL.RECOV コマンドの NOBDSDD パラメーターを指定した場合と同じです。

#### ブランク

DBDS の DD ステートメントが生成された JCL に含まれていることを示します。これはデフォルトのパラメーターです。64 桁目をブランクのままにすると、GENJCL.RECOV コマンドの DBDSDD パラメーターを指定するのと同じです。

65-80 ユーザーのコメントを入れることができます。

### 非標準並行イメージ・コピーのランタイム指定ステートメント

ユーザーの並行イメージ・コピーからのリカバリーで特定のランタイムを指定するには、ランタイム指定制御ステートメントを使用します。

ランタイム指定制御ステートメントは、1 桁目に M を入力し、2 から 32 桁目にランタイムを入力することによって示します。

## DFSURDB0 ユーティリティーの例

DFSURDB0 ユーティリティーのサンプル JCL について以下の例に示します。

この節にある例では、桁合わせを容易にするために、SYSIN ステートメントの上に次のようなコメント行が示されています。

```
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

このコメント行は単なる参考用です。

この節の例で、使用する DBRC で動的割り振りを行わない場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

図 61. 動的割り振りなしで DBRC を使用するための DD ステートメント

```
//RECON1 DD DSN=RECON1,DISP=SHR
//RECON2 DD DSN=RECON2,DISP=SHR
//RECON3 DD DSN=RECON3,DISP=SHR
```

サブセクション:

- 271 ページの『HDAM OSAM データ・セット JCL のリカバリー』
- 271 ページの『DEDB のリカバリー』
- 272 ページの『複数の区画を持つデータベースの区画データ・セットのリカバリー』

- 272 ページの『高速複製リカバリーによる完全修飾データ・セット名のリカバリー』
- 273 ページの『並行ユーザー・イメージ・コピーを使用したリカバリー』
- 273 ページの『タイム・スタンプによるリカバリー』

## HDAM OSAM データ・セット JCL のリカバリー

この例は、DD32DB01 という名前のデータベースに置かれている DBHD3B という DD 名を持つ HDAM OSAM データ・セットをリカバリーするための JCL を示します。入力は、イメージ・コピー・データ・セットと複数のシステム・ログ・データ・セットから得られます。イメージ・データ・セットの DD 名は、制御ステータメントの DFSUDUMP にデフォルト指定されません。

ログ入力は連結できますが、日付と時刻順になっていなければなりません。

```
//STEP1 EXEC PGM=DFSRR00,PARM='UDR,DFSURDB0,DD32DB01'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//DUMPDS DD DSN=IMS.DBBOUT1,DISP=(OLD,KEEP),
// UNIT=TAPE,VOL=SER=BDMP3,LABEL=(,SL)
//DFSUCUM DD DUMMY
//DFSULOG DD DSN=IMSLOG.MONDAY,DISP=(OLD,KEEP),
// UNIT=TAPE,VOL=SER=LOG1,LABEL=(,SL)
// DD DSN=IMSLOG.TUESDAY,DISP=(OLD,KEEP),
// UNIT=TAPE,VOL=SER=LOG2,LABEL=(,SL)
//DBHD3B DD DSN=IMS.DBHD3B,DISP=(NEW,KEEP),
// UNIT=SYSDA,VOL=SER=DBASE2,
// SPACE=(CYL,(20,10))
//DFSVSAMP DD input for VSAM and OSAM buffers and options
/* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

```
//SYSIN DD *
S DD32DB01 DBHD3B DUMPDS
/*
```

## DEDB のリカバリー

この例は、DI32DB01 という名前のデータベースに置かれている AREANAM1 という区域名を持つ DEDB エリアをリカバリーするための JCL を示します。入力、イメージ・コピー・データ・セットと変更累積データ・セットから得られます。DDNAME1 と DSNAME1 は、リカバリーする区域の ADS リストにある DD 名と DS 名です。追加パラメーターは DBRC に YES (Y) を指定し、GSGNAME が与えられます。

```
///STEP1 EXEC PGM=DFSRR00,
// PARM='UDR,DFSURDB0,DI32DB01,,,,,,,,,Y,,,,,,,,,GSGNAME1'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1210
//DFSUDUMP DD DSN=IMS.DBAOUT1,DISP=(OLD,KEEP),
// UNIT=TAPE,VOL=SER=BDMP1,LABEL=(,SL)
//DFSUCUM DD DSN=IMS.CUM1,DISP=(OLD,KEEP),
// UNIT=TAPE,VOL=SER=DBCUM1,LABEL=(,SL)
//DFSULOG DD DUMMY
//DDNAME1 DD DSN=DSNAME1,DISP=OLD
//DFSVSAMP DD DSN=IMS.VSAM.PARM(OPTIONS),DISP=SHR
//RECON1 DD DSN=RECON1,DISP=SHR
```

```
//RECON2 DD DSN=RECON2,DISP=SHR
//RECON3 DD DSN=RECON3,DISP=SHR
/* +-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----

//SYSIN DD *
S DI32DB01 AREANAM1
/*
```

## 複数の区画を持つデータベースの区画データ・セットのリカバリー

この例は、複数の区画を持つデータベース内の最初の区画データ・セットをリカバリーするための JCL を示します。MASTERDB は、RECON に記録され、DBDLIB に入っているマスター・データベース名です。PART001A は、同様に RECON に記録されている、区画 PART001 の最初の部分についての区画 DD 名です。

```
//RECOVDB EXEC PGM=DFSRR00,
// PARM='UDR,DFSURDB0,MASTERDB,,,,,,,,,Y,N'
//IMS DD DSN=IMSTESTG.IMS910.DBDLIB,DISP=SHR
// DD DSN=IMSTESTG.I11X.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DFSUDUMP DD DSN=IMSVS.PART001.IC.ICDSN1,DISP=OLD,
// UNIT=SYSDA,
// VOL=SER=222222
//DFSULOG DD DSN=DBRC.RLDS000,DISP=SHR,
// VOL=SER=000000,UNIT=SYSDA
// DD DSN=DBRC.RLDS001,DISP=SHR,
// VOL=SER=000000,UNIT=SYSDA
//DFSUCUM DD DUMMY
//PART001A DD DSN=MASTERDB.PART.DATASET.A00001,DISP=SHR
//RECON1 DD DSN=IMSTESTL.IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMSTESTL.IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMSTESTL.IMS.RECON3,DISP=SHR
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//SYSIN DD *
S MASTERDB PART001A
/*
```

## 高速複製リカバリーによる完全修飾データ・セット名のリカバリー

この例では、制御ステートメントの 64 桁目の D によって示されるように、高速複製を使用して、リカバリー・ユーティリティー完全修飾データ・セット名が DBRC によって提供される場合のリカバリー JCL を示します。DD ステートメントはジョブ・ストリームに必要ありません。D 仕様は、GENJCL.RECOV コマンドの NOBDSDD パラメーターを指定した場合と同じです。

```
//RECOVERY JOB
/*
//STEP1 EXEC PGM=DFSRR00,PARM='UDR,DFSURDB0,DXVNTZ02'
/*
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//DFSUDUMP DD DSN=IMS.DXVNTZ02.XDLBT04I.IC,
// DISP=(OLD,KEEP),DCB=BUFNO=10
//DFSVDUMP DD DUMMY
//DFSUCUM DD DUMMY
//DFSULOG DD DSN=IMSL0G.MONDAY,DISP=(OLD,KEEP),
```

```

//          UNIT=TAPE,VOL=SER=LOG1,LABEL=(1,SL),
//          DCB=RECFM=VB,
//          DISP=OLD
//DFSVSAMP DD *
1024,4
4096,4
8192,4
//*-----1-----2-----3-----4-----5-----6-----7--

//SYSIN    DD *
S  DXVNTZ02 XDLBT04I          D
/*

```

## 並行ユーザー・イメージ・コピーを使用したリカバリー

この例は、データベース・リカバリー・ステートメントの 63 桁目の M で示されるように、並行ユーザー・イメージ・コピー (CIC) をリカバリーで使用する場合はリカバリー JCL を示します。データベース・リカバリー・ステートメントに続いて、並行ユーザー・イメージ・コピーのランタイムを指定するランタイム・ステートメントがあります。

```

//RECOVER  JOB
//*
//STEP1 EXEC PGM=DFSRR00,PARM='UDR,DFSURDB0,DHVNTZ02'
//STEPLIB  DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS      DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//RECON1   DD DSN=IMS.RECON1,DISP=SHR
//RECON2   DD DSN=IMS.RECON2,DISP=SHR
//RECON3   DD DSN=IMS.RECON3,DISP=SHR
//HIDAM    DD DSN=IMSTESTL.DHVNTZ02.FKXXI01E,
//          DISP=OLD
//DFSUDUMP DD DUMMY
//DFSVDUMP DD DUMMY
//DFSUCUM  DD DUMMY
//DFSULOG  DD DSN=IMSLOG.MONDAY,DISP=(OLD,KEEP),
//          UNIT=TAPE,VOL=SER=LOG1,LABEL=(1,SL),
//          DCB=RECFM=VB,
//          DISP=OLD
//DFSVSAMP DD *
1024,4
4096,4
8192,4
//SYSABEND DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//SYSIN    DD *
//*-----1-----2-----3-----4-----5-----6-----7--

S  DHVNTZ02 HIDAM          M
M062501219001-0700
/*

```

## タイム・スタンプによるリカバリー

この例は、タイム・スタンプ・リカバリーのリカバリー JCL を示します。31 桁目に示される時間は、DBRC コマンドの GENJCL.RECOV の RCVTIME パラメーターで入力された時間に対応します。

**要確認:** HALDB 区画のタイム・スタンプは、HALDB データベースのすべての区画のリカバリーと、それに引き続いて、関連する 1 次索引間接リスト・データ・セット (ILDS) の再作成が必要です。

```

//RECOVERY JOB
//*
//RCV1 EXEC PGM=DFSRR00,REGION=0M,PARM='UDR,DFSURDB0,DIVNTZ02'
//*
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//DBHVSAM1 DD DSN=IMSTESTL.DIVNTZ02.FJXXS01K,
// DISP=OLD
//DFSUDUMP DD DSN=IMS.DIVNTZ02.DBHVSAM1.IC,
// UNIT=SYSDA,
// VOL=(PRIVATE,,,,SER=(222222)),
// LABEL=(1,SL),
// DISP=(OLD,KEEP),DCB=BUFNO=10
//DFSVDUMP DD DUMMY
//DFSUCUM DD DUMMY
//DFSULOG DD DUMMY
//DFSVSAMP DD *
1024,4
4096,4
8192,4
//SYSABEND DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//SYSIN DD *
/*+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7--
S DIVNTZ02 DBHVSAM1 2006.250 12:12:49.123456 -07:00
/*

```

関連資料:

291 ページの『第 20 章 HALDB 索引/ILDS 再作成ユーティリティー (DFSPREC0)』

---

## 第 18 章 DEDB エリア・データ・セット比較ユーティリティ (DBFUMMH0)

DEDB エリア・データ・セット比較ユーティリティ (DBFUMMH0) は、複数エリア・データ・セット (MADS) のすべてのエリア・データ・セットのコピーを比較することにより、その MADS の物理レコードにエラーがないかどうかを調べるために使用します。

DBFUMMH0 ユーティリティは、次に挙げるものの制御インターバル (CI) を比較します。

- ルート・アドレス可能部分と独立オーバーフロー部分
- 再編成した作業単位 (UOW)
- 順次部分 (順次従属セグメントが DBDGEN 時に AREA ステートメントで定義されている場合)

比較の結果が等しくない場合には、最高 10 個までの一致しないレコードの完全ダンプが SYSPRINT DD ステートメントで指定されたメディア上に印刷されます。次に、比較処理は、エリア・データ・セットの終わりまで続行します。

比較処理が終わると、一致しなかった CI の数と一致した CI の数を示すメッセージが SYSPRINT データ・セットに印刷されます。このユーティリティは、指定された各エリア・データ・セットに入っているエラー・キュー・エレメント (EQE) も検査し、区域の EQE 状況を SYSPRINT データ・セットに印刷します。

各 CI の比較は、比較の対象となる CI のカウントが 2 に等しいかそれより大きいときだけ行われます。比較の対象となる CI のカウントは、入出力エラーまたは EQE が検出されるたびに、1 ずつ減っていきます。

このユーティリティは、/STOP REGION コマンドで即時に停止させることができます。

サブセクション:

- 『制約事項』
- 276 ページの『前提条件』
- 276 ページの『要件』
- 276 ページの『推奨事項』
- 276 ページの『入出力』
- 277 ページの『JCL 指定』
- 278 ページの『戻りコード』

### 制約事項

プログラムは同じジョブ・ステップ内でユーティリティまたはデータベースを切り替えることができないため、DBFUMMH0 ユーティリティを実行する場合は、別のジョブ・ステップで実行しなければなりません。

DBFUMMH0 は次のものは比較しません。

- 制御 CI (区域の 1 番目と 2 番目の CI)
- 残余部分に置かれているすべての CI (DBDGEN 時に定義されたスペースが VSAM 定義で定義されたものより小さい場合)

DBFUMMH0 ユーティリティーは、次の場合に終了します。

- 比較の対象となるエリア・データ・セットまたは区域が /STOP AREA または /STOP ADS コマンドで停止されたとき
- 比較の対象となるエリア・データ・セットまたは区域が内部停止コマンドで停止されたとき
- 比較の対象となるエリア・データ・セットのカウン트가減少して 1 になったとき

DBFUMMH0 ユーティリティーは再始動できません。再始動 (REST) パラメーターを使用しても、無視されます。

DBFUMMH0 ユーティリティーは、高速機能データベース (DEDB) のみで機能します。

データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。

## 前提条件

現在、DBFUMMH0 ユーティリティーの使用および実行に対して文書化された前提条件はありません。

## 要件

現在、DBFUMMH0 ユーティリティーに対して文書化された要件はありません。

## 推奨事項

現在、DBFUMMH0 ユーティリティーの使用および実行に対して文書化された推奨事項はありません。

## 入出力

DBFUMMH0 ユーティリティーは、DEDB オンライン・ユーティリティー・コマンドで指定された入力パラメーターが入っているデータ・セットを入力として受け入れます。

DBFUMMH0 ユーティリティーは次の出力を作成します。

- 最高 10 個までの不一致レコードの印刷ダンプ。SYSPRINT データ・セットに印刷されます。
- エラー・キュー・エレメント (EQE) 状況報告書。SYSPRINT データ・セットに印刷されます。
- 不一致 CI の数と一致 CI の数を示したメッセージ。これも SYSPRINT データ・セットに印刷出力されます。



## エラー・キュー・エレメント (EQE) 状況報告書の例

以下の例は、エラー・キュー・エレメント (EQE) 状況報告書を示しています。

```
AREA EQE STATUS
      RBA   ADS01   ADS02   ADS03   ADS04
1. 00000800      *
2. 00001400      *           *
3. 00001800           *
4. 00001C00      *           *
5. 00002000      *
6. 00002400           *
7. 00002800      *   *   *   *
8. 00002C00           *
9. 00003000           *
10. 00003400      *
EQE COUNT      5       2       2       7
NO DATA AVAILABLE FOR CI STARTING AT 00000800
```

## 不一致レコードの印刷ダンプの例

以下の例は、不一致レコードの印刷ダンプを示しています。

```
UNMATCHED CI AT      AREA=AREA01      RBA=00000800
ADS02  0000  F0F1F2F3.....C1C2C3C4  *0123.....ABCD*
ADS03  0000  F0F1F2F3.....A1C2C3C4  *0123.....BCD*
ADS04  0000  F0F1F2F3.....C1C2C3C4  *0123.....ABCD*
ADS02  0020  F4F5F6F7.....C5C6C7C8  *4567.....EFGH*
ADS03  0020  00000000.....C5000000  *.....E...*
ADS04  0020  F4F5F6F7.....C5C6C7C8  *4567.....EFGH*
ADS02  03C0  F4F5F6F7.....C5C6C7C8  *4567.....EFGH*
ADS03  03C0  00000000.....C5000000  *.....E...*
ADS04  03C0  F4F5F6F7.....C5C6C7C8  *4567.....EFGH*
ALL    03E0  D1D2D3D4.....E2E3E4E5  *JKLM.....STUV*
DFS3753I  COMPARISON NOT PERFORMED FOR ADS=ADS01
          REASON: NO DATA AVAILABLE DUE TO EQE
```

## JCL 指定

DBFUMMH0 ユーティリティーは、標準の z/OS ジョブとして実行されます。DBFUMMH0 ユーティリティーの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。DBFUMMH0 ユーティリティーの処理オプションを制御するには、SYSIN DD ステートメントに DEDB オンライン・ユーティリティー・コマンドをコーディングする必要があります。

DEDB エリア・データ・セット比較ユーティリティーを実行するには、次のステートメントが必要です。

- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

このステートメントは、必要な JCL が入っている FPUTIL プロシーチャーを指定することができますが、PGM=DFSRRC00 の形式にすることもできます。

### DD ステートメント

#### STEPLIB DD

DEDB エリア・データ・セット比較ユーティリティーが入っているライブラリーを記述します。

#### STEPDAT DD

最初に検索される専用 VSAM ユーザー・カタログを記述します。定義済み領域がユーザー・カタログにカタログされる場合に、必要です。

#### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

#### SYSIN DD

ユーティリティー制御ステートメントが入っている入力制御データ・セットを記述します。


#### SYSPRINT DD

出力メッセージと統計が入っている出力データ・セットを記述します。

### 戻りコード

現在、DBFUMMH0 ユーティリティーに対して文書化された戻りコードはありません。

関連資料:

 FPUTIL プロシージャ (システム定義)

---

## DBFUMMH0 ユーティリティーの制御ステートメント

DBFUMMH0 ユーティリティー制御ステートメントは、DEDB オンライン・ユーティリティー・コマンドの形式で実行依頼されます。制御ステートメントは、SYSIN DD ステートメントにコーディングされます。制御ステートメントは、比較する入力エリア・データ・セットを記述するためにも使用されます。

### コマンドのフォーマット

パラメーターをフリー・フォームで指定します。

例外: フィールドは、最大文字数が 120 文字であることを除いて、何らかの特定の列に限定されません。

各ステートメントは新しい行で始める必要があります。コマンド名は、最初を非空白文字で開始し、1 つの空白または等号で終了します。コメントを示すときは、先頭文字にアスタリスクを使用します。

コマンドでオペランドが必要な場合は、次の文字を非空白か、等号以外かでオペランド欄を開始します。コマンドで複数のオペランドを使用する場合は、オペランドをコンマで分離します。オペランド欄は、1 つの空白、または行の最後で終了します。ただし、EBCDIC 値の一部である文字 (引用符で囲まれた文字ストリングとして指定) は、終了コンマまたは終了空白としてカウントしません。

行のオペランド欄に続く文字は、コメントとして処理されます。

スキャン・ユーティリティおよび削除ユーティリティのオペランドは、領域に固有です。複数の領域が各実行で処理される場合、QUITCI、V5COMP、および EXCLUDE などのキーワードを領域ごとに指定する必要があります。これらの各オペランドに適用可能な値は、各 AREA コマンドに対する各 GO コマンドの前にコーディングする必要があります。例えば、ユーティリティの実行ですべての領域に対して V5COMP 関数が必要な場合、各 AREA コマンドに対する各 GO コマンドの前に V5COMP をコーディングする必要があります。

## コマンドの継続

以下のように懸垂コンマを使用することで、オペランドの間でオペランド欄を継続することができます。

```
STARTSEQ OP='FIELD=FLD1',      (最初の行)
          VALUE=X'C4C5C2'      (継続行)
```

終了の引用符、コンマ、および再オープンの引用符を以下のように使用して、引用符付きストリングを継続します。

```
STARTROOT X'C1C2C3C4C',      (最初の行)
           '5C6C7',          (2 番目の行)
           'C8C9'            (最後の行)
```

懸垂コンマとコメントの間に 1 つのブランクを使用して、各行にコメントを含めることができます。

### AREA (必須)

どの区域を処理するのかを、制御領域 DD ステートメント上の名前指定します。このコマンドは、各 GO コマンドまたは各 RUN コマンドの後で繰り返さなければなりません。名前は 1 文字から 8 文字までにしなければなりません。最初の文字は英字にし、残りの文字は英数字にします。英字には、@、#、および \$ が含まれます。

### BUFNO (オプション)

DEDB を読み書きするために使用するバッファの数を指定します。

再編成ユーティリティ以外のユーティリティの場合は、少なくとも 7 つのバッファが必要です。BUFNO の指定がない場合は、デフォルトが使用されます。デフォルトのバッファ数は、作業単位 (UOW) 当たりの制御インターバル (CI) 数に 7 を加えたものです。

### DDNAME (オプション)

比較するエリア・データ・セットを指定します。比較ユーティリティの場合、DDNAME ステートメントで指定するエリア・データ・セットは、可用性状況になっていなければなりません。DDNAME ステートメントは、最大 7 個まで指定できます。

### TYPE (必須)

実行のタイプとして、比較、または作成のいずれかのユーティリティを指定します。

**重要:** このコマンドは必須であり、最初の GO コマンドの前か、あるいは SYSIN ファイルの終わりの前に入れておかなければなりません。

プログラムは同じジョブ・ステップ内でユーティリティーを切り替えることができないため、ジョブ・ステップ内に指定できる TYPE コマンドは 1 つだけです。

関連資料:

376 ページの『DBFUHDR0 ユーティリティーの実行』

---

## DBFUMMH0 ユーティリティーの例

これらの例は、2 つ、7 つ、または使用可能なすべてのエリア・データ・セットを比較するためのサンプル JCL を示しています。

サブセクション:

- 『2 つのエリア・データ・セットの比較』
- 『7 つのエリア・データ・セットの比較』
- 281 ページの『使用可能なすべてのエリア・データ・セットの比較』

### 2 つのエリア・データ・セットの比較

以下の例は、2 つのエリア・データ・セットを比較するためのサンプル JCL とユーティリティー制御ステートメントを示しています。

```
//UTL101 EXEC FPUTIL,DBD=DEDBJN23
//*
/*          DEDBJN23 IS THE TARGET DATABASE
/*
//SYSIN    DD *
TYPE COMPARE
AREA DB23AR2
DDNAME DB23AR21
DDNAME DB23AR22
GO
/*
```

### 7 つのエリア・データ・セットの比較

以下の例は、7 つのエリア・データ・セットを比較するためのサンプル JCL とユーティリティー制御ステートメントを示しています。

```
//UTL102 EXEC FPUTIL,DBD=DEDBJN23
//*
/*          DEDBJN23 IS THE TARGET DATABASE
/*
//SYSIN    DD *
TYPE COMPARE
AREA DB23AR2
DDNAME DB23AR21
DDNAME DB23AR22
DDNAME DB23AR23
DDNAME DB23AR24
DDNAME DB23AR25
DDNAME DB23AR26
DDNAME DB23AR27
GO
/*
```

## 使用可能なすべてのエリア・データ・セットの比較

以下の例は、ある区域の使用可能なエリア・データ・セットをすべて比較するためのサンプル JCL とユーティリティー制御ステートメントを示しています。

```
//UTL103 EXEC FPUTIL,DBD=DEDBJN23
//*
//*      DEDBJN23 IS THE TARGET DATABASE
//*
//SYSIN   DD *
TYPE COMPARE
AREA DB23AR2
GO
/*
```



---

## 第 19 章 DEDB エリア・データ・セット作成ユーティリティー (DBFUMRI0)

DEDB エリア・データ・セット作成ユーティリティーは、オンライン・トランザクション処理時に複数 DEDB エリア・データ・セットから 1 つ以上のコピーを作成するために使用します。

アプリケーション・プログラムは、このコピー作成処理の間、続行することができます。DEDB 初期設定ユーティリティーは必要ありません。このユーティリティーは、欠陥制御インターバル (CI) があるデータ・セットを使用して、欠陥 CI のないコピーを作成することができます。アプリケーション・プログラムから要求された書き込みは、使用可能なデータ・セットとこの新しいデータ・セットの両方に対して実行されます。

DBFUMRI0 ユーティリティーを使用して、仮想記憶オプション (VSO) 区域の追加のコピーを作成することもできます。ただし、VSO 区域の複数のコピーの必要性は、非 VSO 区域の場合よりも小さくなります。

VSO 区域からの CI が z/OS データ・スペースに入れられると、IMS がアクティブである限りその CI がアプリケーションで使用できるため、VSO 区域の複数エリア・データ・セットの必要性が軽減されます。あらゆる読み取り要求の場合にこの CI がデータ・スペースから取り出され、読み取りエラーを排除します。この CI に対する更新は、データ・スペースと DASD の両方に書き込まれます。DASD 書き込み中に書き込みエラーが発生しても、CI は、そのデータ・スペースからの読み取りと、そのデータ・スペースに対する更新を行う際に使用可能です。

VSO 区域のコピーを作成するときの DBFUMRI0 ユーティリティーのパフォーマンスは、ユーティリティーの実行時にデータ・スペースに入っている CI の数に左右されます。この区域がプリロードされているか、あるいは頻繁にアクセスされる場合、すべての CI または多くの CI を (DASD ではなく) データ・スペースから取り出して、新しいエリア・データ・セットにコピーできるため、読み取り時間が短縮されます。

このユーティリティーの始動時にデータ・スペースに入っていない VSO 区域の CI は、DASD から読み取られる際にデータ・スペースにコピーされます。ユーティリティーが終了するときには、この区域の CI はすべてデータ・スペースに入っているため、その後は DASD から読み取る必要がなくなります。

VSO 区域の順次従属 (SDEP) 部分は z/OS データ・スペースに保持されないため、DBFUMRI0 ユーティリティーは、SDEP を新しいエリア・データ・セットにコピーするためには、今までどおり DASD からその SDEP を読み取らなければなりません。

VSO 区域は、DBFUMRI0 ユーティリティーの書き込み機能に影響しません。

DBFUMRIO ユーティリティは、使用可能なデータ・セットの CI で読み取りエラーを検出すると、同じ区域の別のデータ・セットを使用します。使用可能なすべてのデータ・セットで同じ CI で読み取りエラーが検出されると、DBFUMRIO ユーティリティは終了します。

DBFUMRIO ユーティリティは、新しいデータ・セットで書き込みエラーを検出した場合も終了します。

/STOP REGION コマンドを発行することにより、DBFUMRIO ユーティリティが完了する前に、このユーティリティを停止することができます。

ヒント: DBFUMRIO ユーティリティのコピー処理の区域フォーマット設定フェーズにより、XRF テークオーバーおよび /STOP REGION コマンドの実行に遅延が生じる場合があります。IMS は、区域フォーマット設定段階が完了するまで、XRF テークオーバーおよび /STOP REGION コマンドの両方を中断します。エリア・データ・セットを事前にフォーマット設定することにより、この遅延を回避することができます。

サブセクション:

- 『制約事項』
- 『前提条件』
- 『要件』
- 285 ページの『推奨事項』
- 285 ページの『入出力』
- 285 ページの『戻りコード』
- 286 ページの『JCL 指定』

## 制約事項

DEDB エリア・データ・セット作成ユーティリティを使用するときは、次のような制約があります。

- DBFUMRIO ユーティリティは再始動できません。必ず最初から実行し直してください。
- データ共用環境の XRF アクティブ IMS システムで DBFUMRIO ユーティリティが実行される場合、テークオーバー処理は、フォーマット設定フェーズが終わるまで中断されます。
- DBFUMRIO ユーティリティは、高速機能データベース (DEDB) のみで機能します。
- データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

## 前提条件

現在、DBFUMRIO ユーティリティに対して文書化された前提作業はありません。

## 要件

DBFUMRIO ユーティリティを実行するには、以下の操作上の要件を満たす必要があります。



- DBFUMRI0 ユーティリティは、別のジョブ・ステップで実行しなければなりません。プログラムは、同じジョブ・ステップ内でユーティリティまたはデータベースを切り替えることはできません。
- CI のサイズは、同じ区域の他のデータ・セットについて定義されているものと同じでなければなりません。

データ・セットは、VSAM 定義で指定されます。データ・セットに割り振られたスペースが、同じ区域の他のデータ・セットに定義されたスペース未満の場合、そのデータ・セットは、十分なスペースが割り振られるまで 1 次割り振りによって拡張されます。高速機能データ・セットの場合、2 次割り振りは無視されます。データ・セットが複数のボリュームを使用するよう定義されている場合、前のボリューム上のスペースがすべて使用されない限り、次のボリューム上にスペースは割り振られません。

## 推奨事項

DBFUMRI0 ユーティリティを XRF 環境で実行する場合、XRF テークオーバーの際の遅延を防止するには、DBRC=N を指定した DEDB 初期設定ユーティリティ (DBFUMIN0) を使用して新しい ADS を事前にフォーマット設定します。IMS は、DBFUMRI0 ユーティリティのフォーマット設定フェーズが完了するまで、/STOP REGION コマンドの実行だけでなく、XRF テークオーバーも中断します。

エリア・データ・セットが事前にフォーマット設定されている場合、DBFUMRI0 ユーティリティは、フォーマット設定フェーズをスキップします。

DBFUMIN0 ユーティリティを実行して、DBFUMRI0 ユーティリティ用にエリア・データ・セットを事前にフォーマット設定する場合、DEDB エリア・データ・セット作成ユーティリティのコピー・フェーズが完了しないうちに新しい ADS が DBRC で使用可能にならないようにするには、DBRC=N が必要です。

## 入出力

DEDB エリア・データ・セット作成ユーティリティは次の入力を使用します。

- コピーするエリア・データ・セット
- DBRC RECON データ・セット
- コマンドで与えた入力パラメーターが入っているデータ・セット

DEDB エリア・データ・セット作成ユーティリティからは、次のような出力が得られます。

- AVAILABLE データ・セットの 1 つ以上のコピー
- 出力メッセージと統計が入っているデータ・セット

## 戻りコード

現在、DBFUMRI0 ユーティリティに対して文書化された戻りコードはありません。

## JCL 指定

DBFUMRIO ユーティリティは、標準の z/OS ジョブとして実行されます。DBFUMRIO ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。DBFUMRIO ユーティリティの処理オプションを制御するには、SYSIN DD ステートメントに DEDB オンライン・ユーティリティ・コマンドをコーディングする必要があります。

### EXEC ステートメント

このステートメントは、必要な JCL が入っている FPUTIL プロシーチャーを指定することができますが、次の形式にすることもできます。

```
PGM=DFSRR00
```

### DD ステートメント

#### STEPLIB DD

DEDB データ・セット作成ユーティリティが入っているライブラリーを記述します。

#### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

#### SYSIN DD

ユーティリティ制御ステートメントが入っている入力制御データ・セットを記述します。

#### SYSPRINT DD

出力メッセージと統計が入っている出力データ・セットを記述します。

---

## DBFUMRIO ユーティリティの制御ステートメント

DBFUMRIO ユーティリティ制御ステートメントは、DEDB オンライン・ユーティリティ・コマンドの形式で実行依頼されます。コマンドは、SYSIN DD ステートメントにコーディングされます。制御コマンドは、作成される入力エリア・データ・セットの記述も行います。

### コマンドのフォーマット

パラメーターをフリー・フォームで指定します。

例外: フィールドは、最大文字数が 120 文字であることを除いて、何らかの特定の列に限定されません。

各ステートメントは新しい行で始める必要があります。コマンド名は、最初を非ブランク文字で開始し、1 つのブランクまたは等号で終了します。コメントを示すときは、先頭文字にアスタリスクを使用します。

コマンドでオペランドが必要な場合は、次の文字を非ブランクか、等号以外かでオペランド欄を開始します。コマンドで複数のオペランドを使用する場合は、オペランドをコンマで分離します。オペランド欄は、1 つのブランク、または行の最後で終了します。ただし、EBCDIC 値の一部である文字 (引用符で囲まれた文字ストリングとして指定) は、終了コンマまたは終了ブランクとしてカウントしません。

行のオペラント欄に続く文字は、コメントとして処理されます。

スキャン・ユーティリティーおよび削除ユーティリティーのオペラントは、領域に固有です。複数の領域が各実行で処理される場合、QUITCI、V5COMP、および EXCLUDE などのキーワードを領域ごとに指定する必要があります。これらの各オペラントに適用可能な値は、各 AREA コマンドに対する各 GO コマンドの前にコーディングする必要があります。例えば、ユーティリティーの実行ですべての領域に対して V5COMP 関数が必要な場合、各 AREA コマンドに対する各 GO コマンドの前に V5COMP をコーディングする必要があります。

## コマンドの継続

以下のように懸垂コンマを使用することで、オペラントの間でオペラント欄を継続することができます。

```
STARTSEQ OP='FIELD=FLD1',      (最初の行)
          VALUE=X'C4C5C2'      (継続行)
```

終了の引用符、コンマ、および再オープンの引用符を以下のように使用して、引用符付きストリングを継続します。

```
STARTROOT X'C1C2C3C4C',      (最初の行)
          '5C6C7',           (2 番目の行)
          'C8C9'             (最後の行)
```

懸垂コンマとコメントの間に 1 つのブランクを使用して、各行にコメントを含めることができます。

### AREA (必須)

どの区域を処理するのかを、制御領域 DD ステートメント上の名前指定します。このコマンドは、各 GO コマンドまたは各 RUN コマンドの後で繰り返さなければなりません。

名前は 1 文字から 8 文字までにしなければなりません。最初の文字は英字にし、残りの文字は英数字にします。英字には、@、#、および \$ が含まれます。

### BUFNO (オプション)

DEDB を読み書きするために使用するバッファの数を指定します。

再編成ユーティリティー以外のユーティリティーの場合は、少なくとも 7 つのバッファが必要です。BUFNO の指定がない場合は、デフォルトが使用されます。デフォルトのバッファ数は、作業単位 (UOW) 当たりの制御インターバル (CI) 数に 7 を加えたものです。

BUFNO の指定がない場合は、デフォルトが使用されます。デフォルトのバッファ数は、作業単位 (UOW) 当たりの制御インターバル (CI) 数に DEDB の階層レベル数 + 12 を加えたものです。

高速 DEDB 直接再編成ユーティリティーでの BUFNO の使用法の詳しい説明については、376 ページの『バッファおよび BUFNO コマンド』を参照してください。

### DDNAME (必須)

作成または比較するエリア・データ・セットを指定します。作成ユーティリティーの場合、DDNAME ステートメントで指定するエリア・データ・セットは、DBRC RECON データ・セットで使用不可状況になっていなければなりません。DDNAME ステートメントは、最大 6 個まで指定できます。

## GO (オプション)

一連の要求を区切るために使用します。2 つ以上の区域の処理、2 つ以上の出口ルーチンの使用、あるいは 2 つ以上のデータ範囲の処理を 1 つのジョブ・ステップで行う場合は、必ず指定する必要があります。

## TYPE (必須)

CREATE ユーティリティーには CREATE 演算子が必要です。

**重要:** このコマンドは必須であり、最初の GO コマンドの前か、あるいは SYSIN ファイルの終わりの前に入れておかなければなりません。

プログラムは同じジョブ・ステップ内でユーティリティーを切り替えることができないため、ジョブ・ステップ内に指定できる TYPE コマンドは 1 つだけです。

---

## DBFUMRIO ユーティリティーの例

これらの例は、1 つ、2 つ、または 5 つの新しいエリア・データ・セットを作成するためのサンプル JCL を示しています。

サブセクション:

- 『1 つの新しいエリア・データ・セットを作成するための JCL』
- 『2 つの新しいエリア・データ・セットを作成するための JCL』
- 289 ページの『5 つの新しいエリア・データ・セットを作成するための JCL』

### 1 つの新しいエリア・データ・セットを作成するための JCL

以下の例は、1 つの新しいエリア・データ・セットを作成するためのサンプル JCL とユーティリティー制御ステートメントを示しています。

```
//UTL1 EXEC FPUTIL,DBD=DEDBJN23
//*
//*      DEDBJN23 IS THE TARGET DATABASE
//*
//SYSIN DD *
TYPE CREATE
AREA DB23AR3
DDNAME DB23AR33
GO
/*
```

### 2 つの新しいエリア・データ・セットを作成するための JCL

以下の例は、2 つの新しいエリア・データ・セットを作成するためのサンプル JCL とユーティリティー制御ステートメントを示しています。

```
//UTL2 EXEC FPUTIL,DBD=DEDBJN23
//*
//*      DEDBJN23 IS THE TARGET DATABASE
//*
//SYSIN DD *
TYPE CREATE
AREA DB23AR3
DDNAME DB23AR33
DDNAME DB23AR34
GO
/*
```

## 5 つの新しいエリア・データ・セットを作成するための JCL

以下の例は、5 つの新しいエリア・データ・セットを作成するためのサンプル JCL とユーティリティー制御ステートメントを示しています。

```
//UTL4    EXEC FPUTIL,DBD=DEDBJN23
//*
//*      DEDBJN23 IS THE TARGET DATABASE
//*
//SYSIN   DD *
TYPE CREATE
AREA DB23AR3
DDNAME DB23AR33
DDNAME DB23AR34
DDNAME DB23AR35
DDNAME DB23AR36
DDNAME DB23AR37
GO
/*
```



---

## 第 20 章 HALDB 索引/ILDS 再作成ユーティリティ (DFSPREC0)

HALDB 索引/ILDS 再作成ユーティリティ (DFSPREC0) は、PHIDAM データベースの 1 次索引データ・セットを再作成するためと、間接リスト・データ・セット (ILDS) を更新しなかった区画リカバリーまたはオフライン区画再編成後に PHIDAM または PHIDAM データベースのいずれかの ILDS を再作成するために使用します。

DFSPREC0 ユーティリティは、HALDB 区画データベース・データ・セット (DBDS) をスキャンし、HALDB 区画データから ILDS と PHIDAM の 1 次索引を再作成します。DFSPREC0 ユーティリティの 1 回の実行で、対象の区画の 1 次索引または ILDS のいずれかのみ、またはその両方を再作成できます。

HALDB 索引/ILDS 再作成ユーティリティは、Type=ULU 領域でバッチ・ユーティリティとして実行されます。

DFSPREC0 ユーティリティが処理するのは一度に 1 つの区画のみです。そのため、同時に複数の区画に対して DFSPREC0 ユーティリティを実行できるため、HALDB データベースにある複数の区画の ILDS と 1 次索引の再作成にかかる時間を大幅に短縮することができます。DFSPREC0 ユーティリティは、1 回の実行で、1 つの HALDB 区画の 1 次索引と ILDS の両方を再作成できます。

ユーティリティ制御ステートメントで指定するリカバリーのタイプに応じて、DFSPREC0 ユーティリティは VSAM 更新モードまたは VSAM ロード・モードを使用して ILDS を再作成することができます。VSAM ロード・モードを使用する場合、フリー・スペースに関して VSAM DEFINE ステートメントで指定した内容が ILDS に適用され、CI および CA の分割が両方とも削減されてパフォーマンスが向上します。

データベース内に存在しなくなったセグメントの ILE が大量に ILDS に含まれている場合、ILDS を再作成すると索引項目の数が大幅に変わることがあります。未使用の ILE は、再作成された ILDS から除去されます。

DFSPREC0 ユーティリティは統合 HALDB オンライン再編成機能をサポートします。再編成機能が停止すると DFSPREC0 ユーティリティは 1 次索引と ILDS を再作成できますが、カーソルはアクティブ状態のままです。

サブセクション:

- 292 ページの『制約事項』
- 292 ページの『前提条件』
- 292 ページの『要件』
- 293 ページの『入出力』
- 293 ページの『JCL 指定』
- 294 ページの『戻りコード』

## 制約事項

DFSPREC0 ユーティリティは、全機能 HALDB データベースのみで機能します。

データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

PHIDAM 索引付きデータ・セットまたは ILDS を削除し、再作成する前に再定義する必要があります。

## 前提条件

1 次索引と ILDS の再作成に加えて区画のデータベース・データ・セットのリカバリーも行う必要がある場合は、データベース・データ・セットのリカバリーを行ってから、索引/ILDS 再作成ユーティリティを実行してください。索引/ILDS 再作成ユーティリティは、1 次索引と ILDS の再作成を行いながら区画データ・セットを読み取ります。そうでない場合、入力した区画を索引/ILDS 再作成ユーティリティの実行前にリカバリーする必要はありません。

PHIDAM 区画の ILDS と 1 次索引のと ILDS の両方を再作成する必要がある場合は、まず 1 次索引の再作成を行ってから ILDS を再作成してください。PHIDAM 区画の場合、索引/ILDS 再作成ユーティリティは 1 次索引を使用して ILDS の再作成を行います。ユーティリティ制御ステートメントに BOTH または BOTHF を指定して、索引/ILDS 再作成ユーティリティの 1 回の実行で 1 次索引と ILDS の両方を再作成する場合、索引/ILDS 再作成ユーティリティは自動的に、まず 1 次索引の再作成を行います。

## 要件

- ユーティリティ制御ステートメントで ILEF パラメーターまたは BOTHF パラメーターを指定して、索引/ILDS 再作成ユーティリティの VSAM ロード・モード・オプションを使用する場合、処理を行うにはユーティリティに 5 つの使用可能な z/OS データ・スペースが必要です。
- このユーティリティを RECON データ・セットに登録されていない IMS カタログ・データベースで使用するには、追加の EXEC パラメーター DFSDF を指定する必要があります。DFSDF= は、未登録の IMS カタログ・データベースの名前を含む IMS.PROCLIB データ・セットの DFSDFxxx メンバーの、3 文字からなる接尾部を指定します。それらの名前は、DATABASE ステートメントの UNREGCATLG パラメーターを用いて指定されます。例えば、次のようになります。

```
//PREC1 EXEC PGM=DFSRR00,REGION=0M,  
// PARM=(ULU,DFSPREC0,DFSCD000,,,,,,,,,,,,,N,N,,,,,,,,,,,,,,,,,,,,,  
// 'DFSDF=CAT')
```

3 文字からなる接尾部 CAT を、未登録の IMS カタログ・データベースの名前を指定する DFSDFxxx メンバーの接尾部で置き換えてください。



## 入出力

HALDB 索引/ILDS 再作成ユーティリティは、制御ステートメントを使用します。制御ステートメントでは、DFSPPREC0 ユーティリティで実行する区画名およびリカバリーのタイプを指定できます。

HALDB 索引/ILDS 再作成ユーティリティからの出力には、再作成された索引、再作成された ILDS、および PHIDAM 1 次索引と ILDS に挿入された項目数の報告書が含まれます。報告書は SYSPRINT データ・セットに書き込まれ、その中には HALDB 索引/ILDS 再作成ユーティリティから返されたエラー・メッセージも含まれます。

HALDB 索引/ILDS 再作成ユーティリティは、再作成された PHIDAM 索引、再作成された ILDS、またはこの両方を作成します。DFSPPREC0 ユーティリティは、SYSPRINT データ・セットに送られた行の総数についても報告します。この行の総数は、作成された索引数および ILE 項目数を示します。

## JCL 指定

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を指定する DD ステートメント

ユーティリティ制御ステートメントも含める必要があります。

### EXEC ステートメント

EXEC ステートメントは次の形式になっていなければなりません。

```
PGM=DFSRR00,PARM='ULU,DFSPPREC0,dbname,,,,,,,,,Y,N' Y=DBRC REQUIRED
```

EXEC カードの dbname は、HALDB マスター・データベースの名前です。この HALDB を構成する区画の 1 つの ILDS および 1 次索引が、DFSPPREC0 ユーティリティの実行で再作成されます。再作成する区画は、SYSIN DD で入力されるユーティリティ制御ステートメントで選択します。

このデータベースは動的に割り当てられます。

### DD ステートメント

#### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

#### DFSRESLIB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

#### IMS DD

分析するデータベースを記述する DBD と PSB が入っているライブラリーを定

義します。これらのデータ・セットは、直接アクセス装置に置いておかなければなりません。このステートメントは必須であり、必ず DBD ライブラリーを定義しなければなりません。

#### **RECON1 DD**

この DD ステートメントはオプションですが、最初の DBRC RECON データ・セットを定義します。これを指定しない場合は、RECON1 データ・セットを識別する IMS.SDFSRESL データ・セットに RECON1 メンバーがなければなりません。

#### **RECON2 DD**

この DD ステートメントはオプションですが、2 番目の DBRC RECON データ・セットを定義します。これを指定しない場合は、RECON2 データ・セットを識別する IMS.SDFSRESL データ・セットに RECON2 メンバーがなければなりません。

#### **RECON3 DD**

この DD ステートメントはオプションですが、3 番目の DBRC RECON データ・セットを定義します。これを指定しない場合は、RECON3 データ・セットを識別する IMS.SDFSRESL データ・セットに RECON3 メンバーがなければなりません。

#### **SYSIN DD**

このプログラムのための入力制御データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。LRECL と BLKSIZE は共に 80 でなければなりません。

#### **SYSPRINT DD**

メッセージ出力データ・セットを定義します。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=120 です。BLKSIZE を指定するときは、120 の倍数になっていなければなりません。

この DD ステートメントは必須です。

SYSPRINT データ・セットは、ユーザーにエラー・メッセージを戻すためにも使用されます。

#### **SYSUDUMP DD**

ダンプ・データ・セットを定義します。

#### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。

### **戻りコード**

索引/ILDS 再作成ユーティリティーが生成する戻りコードは、メッセージ DFS1982I で戻されます。

関連概念:

➡ HALDB オンライン再編成での IMS ユーティリティーの使用 (データベース管理)

関連資料:

➡ DBBBATCH プロシージャ (システム定義)

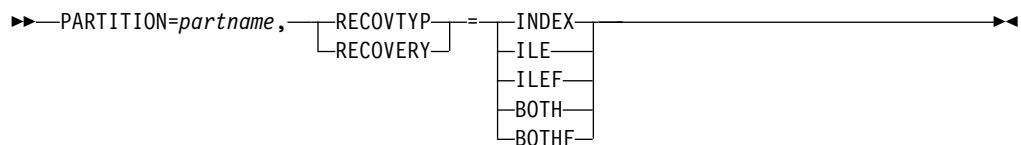
➡ DLIBATCH プロシージャ (システム定義)

---

## DFSPREC0 ユーティリティーの制御ステートメント

ユーティリティー制御ステートメントを使用して、HALDB 索引/ILDS 再作成ユーティリティーの処理オプションを記述できます。

HALDB 索引/ILDS 再作成ユーティリティー制御ステートメントは定位置形式であり、以下の構文図に示すパラメーターを含めることができます。パラメーターの桁位置は、構文図のすぐ後で説明しています。



DFSPREC0 ユーティリティー制御ステートメントのキーワードおよびパラメーターは、以下の定位置に入力する必要があります。

位置 説明

**1-10** 固定文字キーワード。これは文字PARTITION=でなければならない

**11-17** 区画名。再作成する 1 次索引データ・セットまたは ILDS、あるいはその両方を含む HALDB 区画の名前。

**18** コンマ

**19-27** 固定文字キーワード。これは RECOVTYP または RECOVERY のいずれかでなければなりません。

**28-32** リカバリー・タイプ。以下の文字値のいずれかを指定します。

### **BOTH**

指定した区画の 1 次索引と ILDS の両方を再作成します。

### **BOTHF**

指定した区画の 1 次索引と ILDS の両方を再作成します。フリー・スペースのオプションを使用して ILDS を再作成します。フリー・スペースのオプションを選択する場合、ユーティリティーには処理するために使用可能な 5 つの z/OS データ・スペースが必要です。

### **ILE**

指定した区画の ILDS のみを再作成します。

### **ILEF**

フリー・スペースのオプションを使用して、指定した区画の ILDS のみ

を再作成します。フリー・スペースのオプションを選択する場合、DFSPREC0 には処理するために使用可能な 5 つの z/OS データ・スペースが必要です。

## INDEX

指定した区画の 1 次索引のみを再作成します。

---

## DFSPREC0 ユーティリティーの例

以下のサブトピックに示す DFSPREC0 のサンプル JCL は、区画 PARTNAM の ILDS と 1 次索引の両方を再作成します。

最初のサンプルは、VSAM 更新モードを使用して ILDS を再作成します。2 つ目のサンプルは、フリー・スペースのオプションを使用して ILDS を再作成します。

サブセクション:

- 『VSAM 更新モードの JCL の例』
- 『VSAM ロード・モードの JCL の例』

### VSAM 更新モードの JCL の例

以下のサンプル JCL は、ILDS と 1 次索引の両方を再作成します。この例では、PARTNAM は、EXEC ステートメントの定位置パラメーター MSTRDBNM で指定された HALDB マスター・データベースの区画を示します。

```
//STPPD EXEC PGM=DFSRR00,
//      PARM='ULU,DFSPREC0,MSTRDBNM,,,,,,,,,Y,N',
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//      DD DSN=IMS.PSBLIB,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1200
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//SYSIN DD *
PARTITION=PARTNAM,RECOVTYP=BOTH
/*
```

### VSAM ロード・モードの JCL の例

以下のサンプル JCL は、フリー・スペースのオプションを使用して ILDS と 1 次索引の両方を再作成します。この例では、PARTNAM は、EXEC ステートメントの定位置パラメーター MSTRDBNM で指定された HALDB マスター・データベースの区画を示します。

```
//STPPD EXEC PGM=DFSRR00,
//      PARM='ULU,DFSPREC0,MSTRDBNM,,,,,,,,,Y,N',
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//      DD DSN=IMS.PSBLIB,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1200
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//SYSIN DD *
PARTITION=PARTNAM,RECOVTYP=BOTH
/*
```

## DFSPREC0 ユーティリティーの実行

DFSPREC0 ユーティリティーは、VSAM 更新モードを使用する更新オプション、または VSAM ロード・モードを使用するフリー・スペースのオプションのいずれかを使用して実行できます。

ILDS の再作成において、VSAM 更新モードと VSAM ロード・モードの違いは以下のとおりです。

### VSAM 更新モード

ユーティリティー制御ステートメントで ILE または BOTH を指定すると、DFSPREC0 ユーティリティーは順番に区画データ・セットを読み取って、ILE のターゲット・セグメントを読み取るとすぐに ILDS 内の各間接リスト項目 (ILE) を更新します。DFSPREC0 ユーティリティーは、VSAM 更新モードを使用して ILDS を更新します。DFSPREC0 ユーティリティーが区画データ・セットの読み取りを終了すると、ILDS の再作成も終了しています。VSAM 更新モードでは、ILDS 内のフリー・スペースの追加や保守は行われません。

以下の場合に VSAM 更新モードを使用します。

- リカバリー目的で、削除されたセグメント用の ILE を保持する場合
- 最小限の更新のみを必要とする ILDS を更新する場合

### VSAM ロード・モード

ユーティリティー制御ステートメントで ILEF または BOTHF を指定すると、DFSPREC0 ユーティリティーは順番に区画データ・セットを読み取りますが、すぐに ILE を更新する代わりに、DFSPREC0 ユーティリティーは 5 つのデータ・スペースのいずれかに ILE を格納します。DFSPREC0 ユーティリティーは、区画データ・セットの読み取りを終了すると、ILE を間接リスト・キー (ILK) の順にソートしてから、VSAM ロード・モードを使用して 1 回の受け渡しですべての ILE を ILDS にロードします。各 ILE は、ILDS を定義する VSAM KSDS DD ステートメントに指定されたフリー・スペースの量と共にロードされます。この方法をフリー・スペースのオプションと呼ぶのはそのためです。

フリー・スペースのオプションでは、処理を行うには DFSPREC0 ユーティリティーに 5 つの使用可能な z/OS データ・スペースが必要です。

また、フリー・スペースのオプションでは DFSPREC0 ユーティリティーを実行する際に、空の VSAM ILDS データ・セットも 1 つ必要です。ILDS データ・セットの削除および再定義のジョブは、DFSPREC0 ユーティリティーを実行する前に行っておく必要があります。

以下の場合に VSAM ロード・モードを使用します。

- パフォーマンス向上のため、ILDS にフリー・スペースを追加する場合
- 相当な量の更新を必要とする ILDS を再作成する場合
- ILDS 内の CI と CA の分割を除去する場合
- アクティブ区画内に存在しなくなったセグメントの ILE の ILDS を除去する場合

頻繁に更新が必要な ILDS の場合は、VSAM 更新モードよりフリー・スペースのオプションの方が処理が速く行えます。また、フリー・スペースを追

加して将来の分割を防ぐ以外に、CI と CA の分割を除去することで、以後の HALDB 区画の再編成およびリカバリーのパフォーマンスを向上することができます。

---

## 第 21 章 MSDB ダンプ・リカバリー・ユーティリティー (DBFDBDR0)

MSDB ダンプ・リカバリー・ユーティリティー (DBFDBDR0) は、MSDBINIT データ・セットの新しいコピーの作成や、選択された 1 つ以上の MSDB が入っている新しい MSDBINIT データ・セットの作成を行うために使用します。また、IMS システムが異常終了し、緊急時再始動で MSDB をリカバリーできないときに、MSDB を再構成するために使用します。

新しい MSDBINIT データ・セットを作成する場合、DBFDBDR0 ユーティリティーは、ダンプ・データ・セット (MSDBDUMP) またはチェックポイント・データ・セット (MSDBCP1 および MSDBCP2) を使用します。XRF 環境では、チェックポイント・データ・セットとして、MSDBCP1 と MSDBCP2 または MSDBCP3 と MSDBCP4 のどちらでも使用できます。

IMS システムが異常終了し、緊急時再始動で MSDB をリカバリーできないときに、MSDB を再構成する場合、RECOVERY モードで DBFDBDR0 ユーティリティーを実行します。DBFDBDR0 ユーティリティーは、チェックポイント・データ・セットおよび関連のシステム・ログを使用して、MSDB の再構成と、MSDBINIT データ・セットの新しいコピーの作成を行います。

MSDB は、ログに記録されたすべての変更を MSDBDUMP データ・セットに適用することによって MSDBINIT データ・セット上に再構成されます。

両方のチェックポイント・データ・セットの DD ステートメントが同じデータ・セットを指定していると、ユーティリティーは、そのデータ・セットに入っているチェックポイントからリカバリーを行います。

MSDBDUMP データ・セットの形式はチェックポイント・データ・セットの形式と同じなので、MSDBDUMP データ・セットを両方のチェックポイント DD ステートメントに指定しておくこと、そのデータ・セットからリカバリーを行うことができます。

選択された 1 つ以上の MSDB が入っている新しい MSDBINIT を作成する場合、MSDBDUMP データ・セットを入力として UNLOAD モードで DBFDBDR0 ユーティリティーを実行します。MSDB は、ダンプされたときと同じ変更レベルで MSDBINIT データ・セットにアンロードされます。

DBFDBDR0 ユーティリティーは、z/OS バッチ領域で実行されるオフライン・ユーティリティーです。

以下の図に示すのは、MSDB ダンプ・リカバリー・ユーティリティーが MSDB のアンロードと再構成に使用するデータ・セットのフロー・ダイアグラムです。

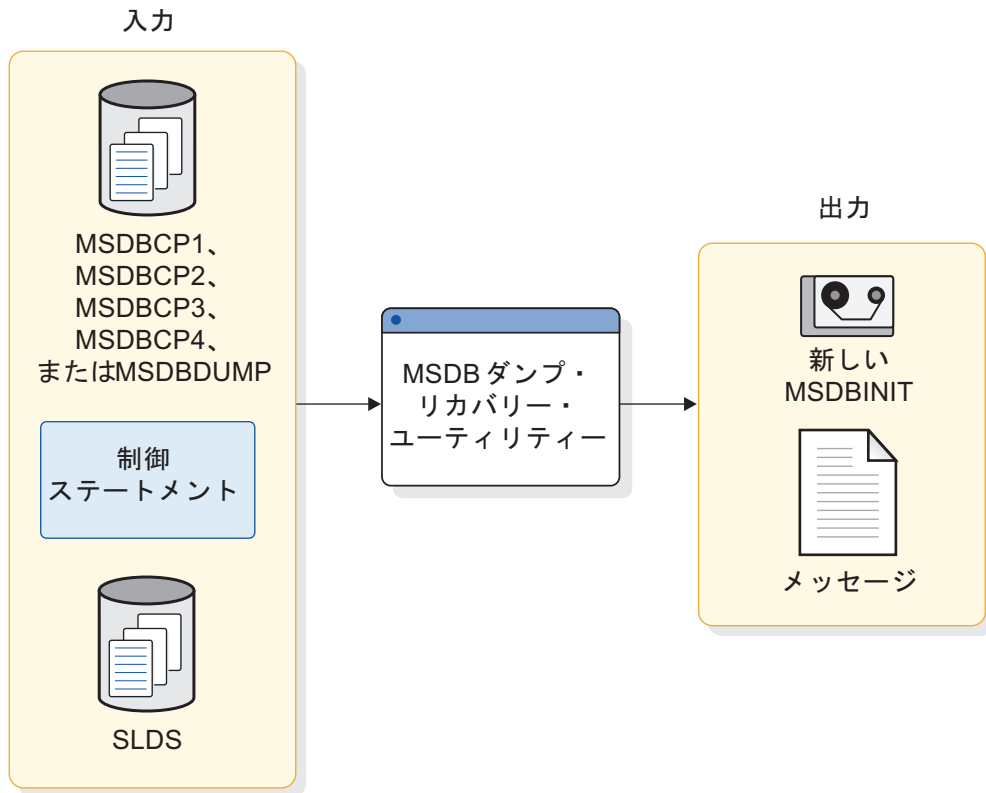


図 62. 入力データ・セットと出力データ・セットを指定する MSDB ダンプ・リカバリー・ユーティリティ

サブセクション:

- 『制約事項』
- 『前提条件』
- 301 ページの『要件』
- 301 ページの『推奨事項』
- 301 ページの『入出力』
- 302 ページの『JCL 指定』
- 303 ページの『戻りコード』

### 制約事項

DBFDBDR0 ユーティリティは、主記憶データベース (MSDB) のみで機能します。

データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

### 前提条件

現在、DBFDBDR0 ユーティリティに対して文書化された前提条件はありません。



## 要件

DBFDBDR0 ユーティリティは少なくとも 512 KB の仮想ストレージを必要とします。

MSDB ダンプ・リカバリー・ユーティリティは、MSDB ダンプ・データ・セットまたはチェックポイント・データ・セットを作成するとき使用したのと同じリリース・レベルのものを実行しなければなりません。

MSDB をアンロードする場合は、以下の点に注意してください。

- MSDBDUMP データ・セットを指定する必要があります。
- 制御データ・セットは UNLOAD 制御ステートメントと指定していなければなりません。

MSDB を再構成する場合は、以下の点に注意してください。

- MSDB チェックポイント・データ・セット (MSDBCP1 と MSDBCP2) を指定する必要があります。ユーティリティは、有効なイメージ・コピーのうちどちらが古い方かを判断し、その箇所からリカバリーを行います。

XRF 環境では、MSDB チェックポイント・データ・セット (MSDBCP1、MSDBCP2、MSDBCP3、および MSDBCP4) を指定する必要があります。ユーティリティは、新しい方の対の MSDB チェックポイント・データ・セットを選択します。その対の中で、ユーティリティは、有効なイメージ・コピーのうちどちらが古い方かを判断し、その箇所からリカバリーを行います。

- 古い方のチェックポイント・データ・セットの作成以後にログに記録された MSDB 変更が入っている IMS ログ・データ・セットを指定することができません。
- 制御データ・セットは RECOVERY 制御ステートメントと指定していなければなりません。

## 推奨事項

現在、DBFDBDR0 ユーティリティに対して文書化された推奨事項はありません。

## 入出力

DBFDBDR0 ユーティリティへの 1 次入力、MSDB ダンプ・データ・セットまたは MSDB チェックポイント・データ・セットです。DBFDBDR0 ユーティリティの 1 次出力は、新しい MSDBINIT です。出力には、メッセージ・データ・セットも含まれます。

MSDB ダンプ・リカバリー・ユーティリティの入出力は、以下の表に示すとおりです。

表 12. MSDB ダンプ・リカバリー・ユーティリティに必要なデータ・セット

入力	出力
MSDBCP1、MSDBCP2、MSDBCP3、MSDBCP4、または MSDBDUMP	新しい MSDBINIT
制御ステートメント	メッセージ

表 12. MSDB ダンプ・リカバリー・ユーティリティーに必要なデータ・セット (続き)

入力	出力
SLDS	

入力データ・セットには、次のものがあります。

- MSDB ダンプ・データ・セット (MSDBDUMP) または MSDB チェックポイント・データ・セット (MSDBCP1 および MSDBCP2)。ダンプ・データ・セットは、/DBDUMP コマンドで作成されます。チェックポイント・データ・セットは、システム・チェックポイントごとに自動的に作成されます。出力データ・セットは、チェックポイントが現れるたびに、MSDBCP1 と MSDBCP2 との間で交互に切り替わります。

XRF 環境では、MSDB チェックポイント・データ・セットは MSDBCP1、MSDBCP2、MSDBCP3、および MSDBCP4 です。出力データ・セットは、チェックポイントが現れるたびに、MSDBCP1 と MSDBCP2 との間で、あるいは MSDBCP3 と MSDBCP4 との間で交互に切り替わります。

- IMS システム・ログ・データ・セット (オプション)。
- 制御データ・セット (オプション)。

1 次出力は、新しい MSDBINIT データ・セットです。メッセージ・データ・セットも生成されます。

## JCL 指定

DBFDBDR0 ユーティリティーは、標準の z/OS ジョブとして実行されます。DBFDBDR0 ユーティリティーの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティー制御ステートメントを JCL ステートメントに含めることが必須であり可能です。

### EXEC ステートメント

DBFDBDR0 ユーティリティーの EXEC ステートメントは、必要なステートメントが入っているカタログ式プロシージャーを呼び出すことができますが、次の形式を使用して JCL に指定することもできます。

```
PGM=DBFDBDR0
```

### DD ステートメント

DBFDBDR0 ユーティリティーは、必須の DD ステートメントおよびオプションの DD ステートメントを数多く使用します。

### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。

### MSDBDUMP DD

すべての MSDB または選択した MSDB のダンプが入っているデータ・セットを記述します。

**MSDBCP1 DD**

すべての MSDB のイメージ・コピーが入っているチェックポイント・データ・セットを記述します。

**MSDBCP2 DD**

すべての MSDB のイメージ・コピーが入っているチェックポイント・データ・セットを記述します。

**MSDBCP3 DD**

XRF 環境において、すべての MSDB のイメージ・コピーが入っているチェックポイント・データ・セットを記述します。

**MSDBCP4 DD**

XRF 環境において、すべての MSDB のイメージ・コピーが入っているチェックポイント・データ・セットを記述します。

**IEFRDER DD**

古い IMS システム・ログが入っているデータ・セットを記述します。これはリカバリー操作のときに使用されるものです。

このデータ・セットは、複数のボリュームに置いておくことができます。2つのイメージ・コピーのうち古い方のチェックポイントが入っているボリュームおよびそれ以後のすべてのボリュームはマウントされていて、DD ステートメントに指定されていなければなりません。

**MSDBCTL DD**

80 バイト・レコードに制御ステートメントが入っているデータ・セットを記述します。この DD ステートメントはオプションです。

**MSDBPRT DD**

メッセージ・データ・セットを記述します。これは、テープ、直接アクセス装置、またはプリンターに置いておくことも、出力ストリームに入れて送ることもできます。

**MSDBINIT DD**

ユーティリティの実行後に、アンロードまたは再構成された MSDB が入っているデータ・セットを記述します。

**戻りコード**

DBFDBDR0 ユーティリティからは、次のような戻りコードが返されます。

コード 意味

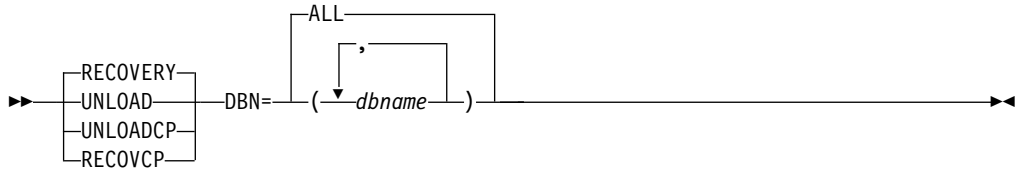
- 0 ユーティリティが正常に実行された
- 4 エラー・メッセージが印刷された
- 8 印刷データ・セットがオープンできない
- 12 印刷ルーチンの入出力エラー

---

**DBFDBDR0 ユーティリティの制御ステートメント**

DBFDBDR0 ユーティリティの制御ステートメントは入力を指定し、ユーティリティが MSDB をアンロードするか、あるいは再構成するかを指定します。これらの制御ステートメントは、MSDBCTL データ・セットから読み取られます。

制御ステートメントはフリー・フォームです。制御情報は、1 桁目から 71 桁目までに入れます。MSDB 名のリストを継続するには、ステートメントの最後の名前の上にコンマを挿入し、72 桁目に空白以外の文字を入れ、次のステートメントの 16 桁目からリストを継続します (1 桁目から 15 桁目までは空白にしておきます)。コメントは、この節の後の例に示されているように入れることができます。



### RECOVERY

MSDB を古い方の MSDB チェックポイント・データ・セット (MSDBCP1 または MSDBCP2) および関連の IMS システム・ログから再構成することを指定します。

特定のチェックポイント・データ・セットからリカバリーを行う場合は、MSDBCP1 と MSDBCP2 DD ステートメントは共に同じデータ・セットを指定していなければなりません。

### UNLOAD

MSDBDUMP DD ステートメントで定義した MSDB ダンプ・データ・セットを z/OS 順次データ・セット (MSDBINIT) にアンロードすることを指定します。ログを使つての更新は行われません。

### UNLOADCP

MSDBCPX DD ステートメントで定義した MSDB チェックポイント・データ・セットを z/OS 順次データ・セット (MSDBINIT) にアンロードすることを指定します。ログを使つての更新は行われません。

### RECOVCP

失敗した MSDBCPx データ・セットを、エラーがない残りの MSDBCPx データ・セットから (または、XRF 生成の IMS システムの場合は、最新のチェックポイントをとったデータが入っている MSDBCPx データ・セットから) 再作成することを指定します。

### DBN=ALL

データ・セットに入っているすべての MSDB を、指定された操作に従ってアンロードまたは再構成することを指定します。

### DBN=(dbname,...)

データ・セットに入っている MSDB の中で、指定された操作に従ってどれをアンロードまたは再構成するかを指定します。

DBN= の指定がないと、DBN=ALL と見なされます。

どの制御ステートメントも与えられていないと、RECOVERY と DBN=ALL が指定されたものと見なされます。

## DBFDBDR0 ユーティリティーの例

これらの例は、DBFDBDR0 ユーティリティーを実行するために必要な JCL とユーティリティー制御ステートメントを示しています。

サブセクション:

- 『MSDB のアンロード』
- 『古い方のチェックポイント・データ・セットからの MSDB の再構成』
- 306 ページの『MSDBDUMP データ・セットからの特定の MSDB のアンロード』
- 306 ページの『チェックポイント・データ・セットおよびシステム・ログによる 1 つの MSDB の再構成』
- 306 ページの『新しい方の対のチェックポイント・データ・セットおよびシステム・ログからの MSDB の再構成』
- 307 ページの『RECOVCP ユーティリティー制御ステートメントによる新しい MSDBCP1 データ・セットの再作成』

### MSDB のアンロード

この例では、すべての MSDB を MSDBDUMP データ・セットから MSDBINIT データ・セットにアンロードします。

```
//UN101 EXEC PGM=DBFDBDR0
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//MSDBPRT DD SYSOUT=A
//*
//MSDBINIT DD DSN=IMS.MSDBLM02,DISP=(NEW,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=IMSDCL,
//          DCB=(BLKSIZE=13030,RECFM=VBT,LRECL=13026),
//          SPACE=(CYL,1)
//MSDBDUMP DD DSN=IMS.LMDMP,DISP=SHR MSDB DUMP
//*
//MSDBCTL DD *
//          UNLOAD DBN=ALL
/*
```

### 古い方のチェックポイント・データ・セットからの MSDB の再構成

この例では、MSDB を古い方のチェックポイント・データ・セットと IMS システム・ログから再構成します。

```
//RC101 EXEC PGM=DBFDBDR0
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//MSDBPRT DD SYSOUT=A
//*
//MSDBINIT DD DSN=IMS.MSDBLM03,DISP=(NEW,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=IMSDCL,
//          DCB=(BLKSIZE=13030,RECFM=VBT,LRECL=13026),
//          SPACE=(CYL,1)
//MSDBCP1 DD DSN=IMS.LMCP1,DISP=SHR CHECKPOINT # 1
//MSDBCP2 DD DSN=IMS.LMCP2,DISP=SHR CHECKPOINT # 2
//IEFRDER DD DSN=IMS.LMLOG,DISP=SHR IMS OLDS/SLDS
//*
//MSDBCTL DD *
//          RECOVER ALL MSDB'S
//          RECOVERY DBN=ALL
/*
```

## MSDBDUMP データ・セットからの特定の MSDB のアンロード

この例では、1 つの MSDB を MSDBDUMP データ・セットから MSDBINIT データ・セットにアンロードします。

```
//UN201 EXEC PGM=DBFDBDR0
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//MSDBPRT DD SYSOUT=A
//* MESSAGE PRINT FILE
//MSDBINIT DD DSN=IMS.MSDBLM02,DISP=(NEW,CATLG,DELETE),
// UNIT=SYSDA,VOL=SER=IMSDCL,
// DCB=(BLKSIZE=13030,RECFM=VBT,LRECL=13026),
// SPACE=(CYL,1)
//MSDBDUMP DD DSN=IMS.LMDMP,DISP=SHR MSDB DUMP
//* UNLOAD ONLY MSDB 05
//MSDBCTL DD *
UNLOAD DBN=(MSDBLM05)
/*
```

## チェックポイント・データ・セットおよびシステム・ログによる 1 つの MSDB の再構成

以下の例では、1 つの MSDB を古い方のチェックポイント・データ・セットと IMS システム・ログから再構成します。

```
//RC202 EXEC PGM=DBFDBDR0
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//MSDBPRT DD SYSOUT=A
//* MESSAGE PRINT FILE
//MSDBINIT DD DSN=IMS.MSDBLM03,DISP=(NEW,CATLG,DELETE),
// UNIT=SYSDA,VOL=SER=IMSDCL,
// DCB=(BLKSIZE=13030,RECFM=VBT,LRECL=13026),
// SPACE=(CYL,1)
//MSDBCP1 DD DSN=IMS.LMCP1,DISP=SHR CHECKPOINT # 1
//MSDBCP2 DD DSN=IMS.LMCP2,DISP=SHR CHECKPOINT # 2
//IEFRDRE DD DSN=IMS.LMLOG,DISP=SHR IMS OLDS
//* RECOVER ONLY MSDB 05
//MSDBCTL DD *
RECOVERY DBN=(MSDBLM05)
/*
```

## 新しい方の対のチェックポイント・データ・セットおよびシステム・ログからの MSDB の再構成

以下の例では、特定の 1 つの MSDB を、新しい方の対のうちの古い方のチェックポイント・データ・セットと IMS システム・ログから再構成します。

```
//RC202 EXEC PGM=DBFDBDR0
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//MSDBPRT DD SYSOUT=A
//* MESSAGE PRINT FILE
//MSDBINIT DD DSN=IMS.MSDBLM03,DISP=(NEW,CATLG,DELETE),
// UNIT=SYSDA,VOL=SER=IMSDCL,
// DCB=(BLKSIZE=13030,RECFM=VBT,LRECL=13026),
// SPACE=(CYL,1)
//MSDBCP1 DD DSN=IMS.LMCP1,DISP=SHR CHECKPOINT # 1
//MSDBCP2 DD DSN=IMS.LMCP2,DISP=SHR CHECKPOINT # 2
//MSDBCP3 DD DSN=IMS.LMCP3,DISP=SHR CHECKPOINT # 3
//MSDBCP4 DD DSN=IMS.LMCP4,DISP=SHR CHECKPOINT # 4
//IEFRDRE DD DSN=IMS.LMLOG,DISP=SHR IMS OLDS
```

```

/*                                RECOVER ONLY MSDB 06
//MSDBCTL DD *
    RECOVERY DBN=(MSDBLM06)
/*

```

## RECOVCP ユーティリティー制御ステートメントによる新しい MSDBCP1 データ・セットの再作成

以下の例は、RECOVCP ユーティリティー制御ステートメントを使用して新しい MSDBCP1 データ・セットを再作成するための JCL を示しています。

```

//RECOVCP EXEC PGM=DBFDBDR0
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//MSDBPRT DD SYSOUT=A
/*                                MESSAGE PRINT FILE
//NEWCP DD DSN=IMS.LMCP1, new CHECKPOINT DS #1
//      DISP=(NEW,CATLG,DELETE),
//      UNIT=SYSDA,VOL=SER=IMSDCL,
//      DCB=(BLKSIZE=2048,RECFM=F,LRECL=2048),
//      SPACE=(CYL,1)
//MSDBCP1 DD DSN=IMS.LMCP2,DISP=SHR CHECKPOINT DS #2
//MSDBCP2 DD DSN=IMS.LMCP2,DISP=SHR CHECKPOINT DS #2
//@      IF XRF ENVIRONMENT, UNCOMMENT NEXT TWO LINES
//@MSDBCP3 DD DSN=IMS.LMCP3,DISP=SHR CHECKPOINT DS #3
//@MSDBCP4 DD DSN=IMS.LMCP4,DISP=SHR CHECKPOINT DS #4
//IEFRDER DD DSN=IMS.LMLOG,DISP=SHR IMS OLDS/SLDS
/*                                RECOVER ALL MSDB'S
//MSDBCTL DD *
    RECOVCP
/*

```





---

## 第 4 部 再編成ユーティリティおよび変換ユーティリティ

再編成ユーティリティおよび変換ユーティリティは、データベース、高速機能 DEDB 区域、HALDB 区画の再編成、および、データベースのタイプの変換を行うために使用します。

各トピックでは、1 つのユーティリティを紹介し、その作動方法を説明し、その使用のための要件および制約事項を定義し、例を提供します。



---

## 第 22 章 データベース事前再編成ユーティリティ (DFSURPR0)

データベース事前再編成ユーティリティは、他の論理関係解決ユーティリティによって使用される制御データ・セットを作成するために使用します。このユーティリティは、データベース・スキャン・ユーティリティにスキャンさせる必要のあるデータベースおよびセグメントがある場合、それらも示します。

非 HALDB の全機能データベースを HALDB に変換する際に、データベース事前再編成ユーティリティを使用して、1 つ以上の HALDB 区画を初期設定することができます。既存の HALDB データベースの新規の区画または変更された区画を初期設定するには、HALDB 区画データ・セット初期設定ユーティリティ (DFSUPNT0) をお勧めします。ただし、これらの目的のために DFSURPR0 を使用することもできます。

HALDB 全体、またはある範囲の区画を再編成するときは、データの入っていない残りの区画を初期設定するための最終ステップとして DFSUPNT0 を実行することもできます。データを受け取る区画は、再ロードまたは初期ロードの間に動的に初期設定されます。

区画の初期設定の際に、データベース事前再編成ユーティリティは、接頭部解決と更新が不要であることを示す制御データ・セット (CDS) を作成し、初期設定が必要であるとして RECON データ・セットでフラグの立っているすべてのデータ・セットを初期設定します。

全機能データベースを HALDB に変換するときに DFSURPR0 ユーティリティを使用して HALDB 区画を初期設定すると、必要な JCL の変更量が最小になります。これは、データベースの再編成に使用した同じ JCL を、データベースを HALDB に変換する際に使用できるためです。

全機能データベースを HALDB に変換するときに DFSURPR0 ユーティリティを使用して HALDB 区画を初期設定することによって、初期設定されていなかった区画を更新しようとするために起こる IMS 異常終了を避けることもできます。新規 HALDB データベースが RECON データ・セットで最初に定義されるとき、そのすべての区画が RECON データ・セット内で「区画の初期設定が必要」(PINIT) として記録されます。事前再編成の後に、新規の区画はすべて RECON データ・セット内で「初期設定済み」として記録されます。

以下のフロー・ダイアグラムは、データベース事前再編成ユーティリティを示します。

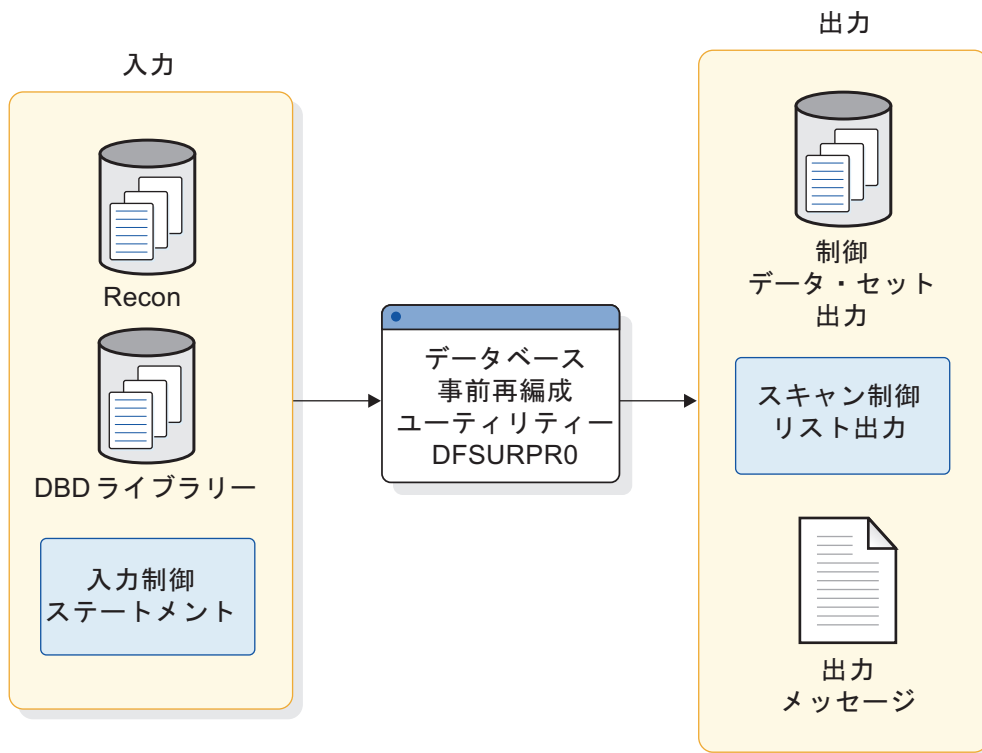


図 63. データベース事前再編成ユーティリティ

サブセクション:

- 『制約事項』
- 『前提条件』
- 313 ページの『要件』
- 313 ページの『推奨事項』
- 313 ページの『入出力』
- 314 ページの『JCL 指定』
- 316 ページの『戻りコード』

### 制約事項

DFSURPRO ユーティリティは、全機能 (HALDB および非 HALDB) データベースのみで機能します。

データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

### 前提条件

HALDB 区画を初期設定する場合、DFSURPRO ユーティリティでは、それらの区画が RECON データ・セット内で「区画の初期設定が必要」としてフラグが立てられている必要があります。RECON データ・セット内の区画定義の作成または変更を行う場合、「区画の初期設定が必要」のフラグは IMS によって自動的に立てられます。ただし、DASD 上で VSAM 区画データ・セットを削除および再定義する

場合は、ユーザーが自分で RECON データ・セットに区画初期設定フラグを設定してから DFSURPR0 ユーティリティを実行する必要があります。

以下のいずれかの方法により、VSAM HALDB 区画データ・セットについて、「区画の初期設定が必要」のフラグを RECON データ・セットに設定できます。

- オンライン区画定義ユーティリティ (%DFSHALDB) または DBRC のオンライン・コマンドかバッチ・コマンド (例えば CHANGE.DB DBD(*partition\_name*) PINIT など) を使用してから、区画を再ロードします。
- 区画をアンロードし、VSAM がそれを再定義した後、その区画を再ロードします。この場合、ユーザーは区画初期設定フラグを設定する必要がありません。最初の更新が実行されると PROCOPT L (ロード) が区画を内部で初期設定するためです。
- 影響を受ける区画を無条件に初期設定するには、バッチ区画初期設定ユーティリティ (DFSUPNT0) を使用します。指定された区画は、RECON データ・セットでの初期設定の状態にかかわらず初期設定されます。

注: VSAM データ・セットを HALDB 用に再定義する場合、VSAM DEFINE ステートメント内に REUSE パラメーターを指定する必要があります。

## 要件

索引付き非 HALDB データベースを初期ロードまたは再編成するときに副次索引が存在する場合は、副次索引を作成するとき使用される制御データ・セットを作成するために、索引付きデータベースに対してデータベース事前再編成ユーティリティを実行してください。また、非 HALDB データベースをロードまたは再編成する場合に、両方のデータベースが論理関係を含んでいるときも、このユーティリティを実行する必要があります。

HALDB 区画データ・セットを初期設定する場合は、RECON データ・セットを使用可能にしておく必要があります。

新規作成された HALDB 区画または変更された HALDB 区画に対して VSAM データ・セットを初期設定する場合、以下のことを行ってください。

1. DFSVSAMP DD カードのバッファ・プール定義で、DL/I がすべての HALDB 区画データ・セットを HALDB データベース用にオープンするのに十分な大きさが指定されていることを確認してください。
2. VSAM DEFINE CLUSTER ステートメントで REUSE オプションを使用して、すべての HALDB 区画 VSAM データ・セットを定義する。
3. 初期ロード、または再ロード用の再編成のために、データベース事前再編成ユーティリティを実行する。

## 推奨事項

現在、DFSURPR0 ユーティリティに対して文書化された推奨事項はありません。

## 入出力

データベース事前再編成ユーティリティの入出力は、以下の表に示すとおりです。

表 13. データベース事前再編成ユーティリティの入出力

入力	出力
RECON	出力メッセージ
DBD ライブラリー	スキャン制御リスト
入力制御ステートメント	制御データ・セット

このユーティリティへの入力は、ロード、再編成、またはその両方が行われるデータベースを指定するユーティリティ制御ステートメントから成るデータ・セットです。HALDB の場合は、このユーティリティの入力に、初期設定されていない 1 つ以上の区画も必要です。制御ステートメントで指定されたデータベースのために使用される DBD は、各データベースを、論理関係が解決された後に存在するように定義していなければなりません。接頭部更新ユーティリティが正常に実行されるまで、DBD を変更しないでください。

出力には、データベース・スキャン・ユーティリティ、データベース接頭部解決ユーティリティ、およびデータベース HD 再編成再ロード・ユーティリティによって使用される制御データ・セットが含まれます。さらに、出力は 1 つ以上の初期化された HALDB 区画の場合もあります。

このユーティリティから出される出力メッセージは、接頭部解決ユーティリティおよび接頭部更新ユーティリティの実行に先立って行っておく必要があるデータベース操作を示しています。以下に例を示します。

- メッセージ DFS861I の中で DBIL= の後にリストされたデータベースは、初期ロードしなければなりません。
- メッセージ DFS861I の中で DBR= の後にリストされたデータベースは、HD 再編成アンロード/再ロード・ユーティリティを使用して再編成しなければなりません。メッセージ DFS862I の中で DBS= の後にリストされたデータベースは、データベース・スキャン・ユーティリティを使用してスキャンしなければなりません。

このユーティリティによって作成されるその他の出力として、次のものがあります。

- 入力として与えられた制御ステートメントのリスト
- データベース・スキャン・ユーティリティが使用するためのスキャン制御ステートメントのオプション・デッキ
- エラー・メッセージ
- 終了メッセージ

データベース事前再編成ユーティリティの機能 ID は PO です。

## JCL 指定

データベース事前再編成ユーティリティは、標準の z/OS ジョブとして実行されます。次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント

- 入力および出力を定義する DD ステートメント

JCL 入力には、ユーティリティー制御ステートメントを含めることもできます。

#### EXEC ステートメント

EXEC ステートメントは次の形式になっていなければなりません。

```
PGM=DFSRR00,PARM='ULU,DFSURPR0'
```

SPIE、BUF といった通常の IMS 定位置パラメーターは、PARM フィールドの中でプログラム名の後に置くことができます。

#### DD ステートメント

##### DFSVSAMP DD

DL/I バッファラー・ハンドラーが必要とするバッファラー・プール情報が入っているデータ・セットを記述します。この DD ステートメントは、HALDB 区画に必須です。

##### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

##### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

##### IMS DD

入力制御ステートメントで指定されたデータベースを記述する DBD が入っているライブラリーを定義します。このデータ・セットは、直接アクセス装置に置いておかなければなりません。

この DD ステートメントは必須です。

##### SYSIN DD

入力制御ステートメントが入っています。このデータ・セットはテープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。

この DD ステートメントは必須です。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=80 です。BLKSIZE は、この DD ステートメントで指定しなければなりません。BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

##### SYSPRINT DD

メッセージ出力データ・セットを定義します。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=120 です。BLKSIZE は、この DD ステートメントで指定しなければなりません。BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

この DD ステートメントは必須です。

#### **SYSPUNCH DD**

パンチ・タイプ出力データ・セットを定義します。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。「OPTIONS=(PUNCH)」制御ステートメントがあるときは、この DD ステートメントが必要です。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=80 です。BLKSIZE は、この DD ステートメントで指定しなければなりません。BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないため、結果は予測不能です。

#### **DFSURCDS DD**

このプログラムのための出力データ・セットを定義します。このデータ・セットは、データベースのロード時に、接頭部解決ユーティリティーとデータベース・スキャン・ユーティリティーによって使用される制御データ・セットです。

HALDB の場合は、これらのユーティリティーは CDS では使用不可です。

このプログラムの中で指定されている DCB パラメーターは、RECFM=FB および LRECL=1600 です。BLKSIZE は、この DD ステートメントで指定しなければなりません。

この DD ステートメントは必須です。

#### **SYSABEND DD または SYSUDUMP DD**

ダンプ・データ・セットを定義します。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

#### **RECON1 DD**

最初の DBRC RECON データ・セットを定義します。

#### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。

#### **RECON3 DD**

RECON1 または RECON2 にエラーが見つかったとき使用される、オプションの DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。


### **戻りコード**

プログラム終了時に次の戻りコードが返されます。

コード 意味

- 0 エラーは検出されなかった
- 8 1 つ以上のエラー・メッセージが出された。

関連資料:

 DBBBATCH プロシージャ (システム定義)



## DFSURPRO ユーティリティーの制御ステートメント

DFSURPRO ユーティリティーでは、DBIL、DBR、DBM、および OPTIONS の 4 つのユーティリティー制御ステートメントが使用できます。

### DBIL ステートメント

DBIL ユーティリティー制御ステートメントは、初期ロードまたは再編成するデータベースを指定するために使用します。このステートメントは、論理ポインターまたはカウンターに影響するような変更が DBD に行われている場合の再編成に必要です。

```
▶▶ DBIL=--database name comments ▶▶
```

このステートメントは、1 つ以上指定することができます。DBIL ステートメントは、以下に準拠している必要があります。

- 各 DBD 名は左寄せし、長さは 8 文字までです。
- DBD 名が 8 文字未満のときは、8 文字になるよう、末尾ブランク文字を入れなければなりません。
- 各ステートメントの最後の項目の後に、ブランクを入れなければなりません。

HIDAM または PHIDAM データベースを初期ロードするときは、その DBD 名だけを DBIL 制御ステートメントに指定してください。HIDAM または PHIDAM 1 次索引の DBD 名も、副次索引の DBD 名も指定しないでください。

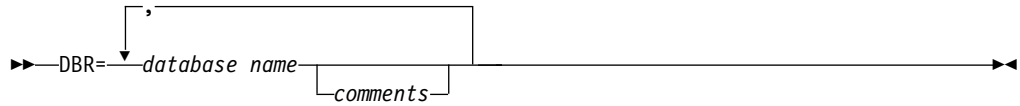
論理関係に影響するような構造上の変更が、データベース再編成処理でデータベースに行われたときは、HD 再ロード・ステップの前に次のことを行っておかなければなりません。

- 新しい DBD をアSEMBルし、バインドしてIMS DBD ライブラリーに入れておく。
- DBIL= ステートメントでデータベース名を指定して、新しい DBD に対してデータベース事前再編成ユーティリティーを再実行する。論理的に関係を持つ他のデータベースについても、DBIL の指定が必要になることがあります。

推奨事項: 論理ポインターまたはカウンターに影響するような変更が DBD に行われているときは、DBIL ステートメントを使用してください。このオプションは、ポインター・エラーを訂正するためにも使用しなければなりません。

### DBR ステートメント

DBR ユーティリティー制御ステートメントは、DOS/VSE DL/I から変換する、あるいは再編成するデータベースを指定するために使用します。



このステートメントは、1 つ以上指定することができます。各 DBD 名は左寄せし、長さは 8 文字までです。DBD 名が 8 文字未満のときは、8 文字になるよう、末尾空白文字を入れなければなりません。各ステートメントの最後の項目の後に、空白を入れなければなりません。

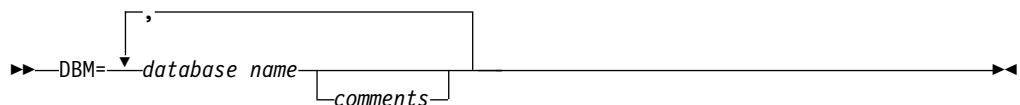
HISAM 再編成アンロード/再ロード・ユーティリティを使用して HISAM データベースを再編成する場合は、HISAM DBD 名を DBR 制御ステートメントに指定しないでください。ただし、HD 再編成アンロード/再ロード・ユーティリティを使用して HISAM データベースを再編成する場合は、HISAM DBD 名を DBR 制御ステートメントに指定しなければなりません。

HIDAM または PHIDAM データベースを再編成する場合は、その DBD 名だけを DBR 制御ステートメントに指定してください。HIDAM 1 次索引の DBD 名も、副次索引の DBD 名も指定しないでください。HALDB では、区画の初期設定で要求されない限り、事前再編成の必要はありません。

DBR は古い RBA 値を使用して、データベースの論理関係を解決します。

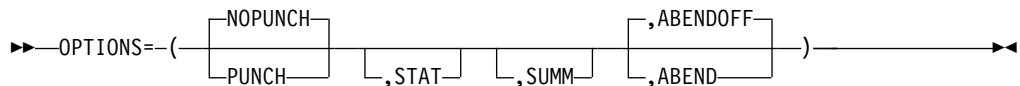
## DBM ステートメント

DBR ユーティリティ制御ステートメントは、制御データ・セット (CDS) 内の DB スキャン・ビットを非 HALDB データベースから HALDB データベースへのマイグレーションに対して使用不可にします。



## OPTIONS ステートメント

OPTIONS ユーティリティ制御ステートメントは、データベースの再編成時または初期ロード時に、オプションの情報を与えるかどうかを示します。



これらのパラメーターは定位置パラメーターではないので、どの順序でも指定できます。パラメーターを複数個指定するときは、パラメーターとパラメーターの間にコンマを入れてください。このステートメントで指定した情報は、データベース事前再編成ユーティリティ (DFSURPR0) およびデータベース接頭部解決ユーティリティ (DFSURG10) の実行時に得られる出力に影響します。

### PUNCH

これを指定すると、データベース・スキャン・リストが SYSPUNCH データ・

セットと SYSPRINT データ・セットの両方に書き込まれます。この出力は、SYSIN データ・セットを使用した、データベース・スキャン・ユーティリティーへの入力として使用されます。

#### **NOPUNCH**

これを指定すると、スキャン・リストは、SYSPUNCH DD ステートメントで定義されたデータ・セットに書き込まれません。NOPUNCH はデフォルトです。

#### **STAT**

これを指定すると、データベース接頭部解決ユーティリティー (DFSURG10) は更新されたセグメントに関する統計を累積します。

#### **SUMM**

これを指定すると、データベース接頭部解決ユーティリティー (DFSURG10) は、メッセージ DFS878I が出された回数を累積し、それを印刷します。


#### **ABENDOFF**

ABEND 機能をオフにします。これはデフォルトです。ABEND を指定すると、これは、ABENDOFF が指定されるまでジョブ・ステップ内で有効です。

#### **ABEND**

実行を異常終了させるような条件が起こると、ユーザー ABEND 955 を出して終了させます。SYSABEND または SYSUDUMP DD ステートメントがあると、ダンプが印刷されます。

関連概念:

 論理関係を追加する場合のユーティリティーの使用の要約 (データベース管理)

---

## **DFSURPRO ユーティリティーの例**

ここでの例は、DFSURPRO ユーティリティーの使用法を示しています。

DBM ステートメントは、非 HALDB のマイグレーション・アンロードに使用され、DBIL ステートメントは、%DFSHALDB または DBRC コマンド CHANGE.DB によって、区画の初期設定が必要であるとして記録されていた HALDB 区画を初期設定するために使用されます。

この例は、初期ロードする 2 つのデータベースで DFSURPRO を実行するために必要なジョブ制御ステートメントとユーティリティー制御ステートメントを示しています。この 2 つのデータベースの DBD 名は、PAYR と SKILLINV です。

```
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSURPRO'  
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR  
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR  
//IMS DD DSN=IMS.DBDLIB,DISP=SHR  
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1200  
//DFSURCDS DD DSN=IMS.RLCDS,DISP=(NEW,KEEP),  
// UNIT=SYSDA,VOL=SER=IMSMSC,DCB=(BLKSIZE=1600),  
// SPACE=(CYL,1)  
//SYSIN DD *,DCB=BLKSIZE=80  
DBIL=PAYRbbbb,SKILLINV  
/*
```

以下の図は、ユーザー・データが入った VSAM データ・セットに対して指定された REUSE を示しています。

```
//DBVDJ05 EXEC PGM=IDCAMS
//SYSIN DD *
DEFINE CLUSTER (NAME (IMSTESTS.DBVHDJ05.A00001) -
              TRK(3,1) RECSZ (1017.1017) -
              VOL (DSHR00) SHAREOPTIONS (3,3) -
              CISZ (1024) NIXD) REUSE
/*
```


以下の図に示す例では、ある HALDB 区画に初期設定が必要であることを DBRC に通知しています。データベース事前再編成ユーティリティーを実行する前に、この例のコマンドを使用してください。

```
//UPDRECNI EXEC PGM=DSPURX00
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSIN DD*
CHANGE.DB DBD(DBVHDJ05) PINIT
```

以下の図に示す例は、VSAM HALDB 区画データ・セットの定義に必要なジョブ制御ステートメントです。VSAM CLUSTER 定義に REUSE オプションを使用しています。

```
//PRERSTEP EXEC PGM=DFSRR00,
//          PARM=(ULU,DFSURPRO,,,,,,,,,,,,,Y,N,,N)
//*          PREREORGANIZATION-UTILITY
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSVSAMP DD DSN=IMS.DFSVSAMP(VSM885FP),DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//SYSUDUMP DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
//PRINTDD DD SYSOUT=A
//DFSURCDS DD DSN=PINIT.CDS,UNIT=SYSDA,DISP=(,PASS),
//          SPACE=(TRK,1),DCB=BLKSIZE=1600
//SYSIN DD *
DBR=DBVHDJ05
/*
```

関連概念:

 HALDB 区画の初期設定 (データベース管理)

---

## 第 23 章 HALDB マイグレーション・エイド・ユーティリティー (DFSMAID0)

HALDB マイグレーション・エイド・ユーティリティー (DFSMAID0) は、既存の全機能データベースをスキャンして、区画数、区画サイズ、および区画のキー範囲指定の推定値を返します。これらの値は、スキャンしたデータベースを HALDB に変換する際に使用できます。

DFSMAID0 ユーティリティーが返す情報は、ユーティリティーに対して実行依頼する基準に応じて異なります。

- 計画した HALDB データベースの各区画に必要なキー範囲が分かる場合は、ユーティリティー制御ステートメントに KR=キーワードを使用して、各区画のハイ・キーをユーティリティーに対して実行依頼できます。DFSMAID0 ユーティリティーは、各区画に必要なスペースを返します。
- 各区画に必要なセグメント・データの最大バイト数が分かる場合は、ユーティリティー制御ステートメントに MAX キーワードを使用して、バイト数を実行依頼できます。DFSMAID0 ユーティリティーは、計画された HALDB データベースに必要な区画数、および各区画に必要なハイ・キーを返します。
- 計画した HALDB データベースで、同じサイズの区画がいくつ必要なのか分かる場合は、ユーティリティー制御ステートメントに NBR= キーワードを使用して区画数を実行依頼できます。DFSMAID0 ユーティリティーは、各区画に必要なスペースおよび各区画に必要なハイ・キーを返します。

DFSMAID0 ユーティリティーが実行する各種のデータベース分析 (KR、MAX、または NBR) について、このユーティリティーが IMS データベースの全探索を行うのは 1 回だけです。

DFSMAID0 ユーティリティーは、統計の格納とソートにそれぞれ 2 GB の z/OS データ・スペースを 4 つまで使用します。DFSMAID0 ユーティリティーに必要な正確なデータ・スペースの数は、ユーティリティーが実行する分析の種類によって異なります。

ターゲット・データベースを処理するのに DFSMAID0 ユーティリティーが要する時間とストレージ量を減らすために、DFSMAID0 ユーティリティーはサンプリング・オプションを提供します。SAMPLE キーワードを指定すると、DFSMAID0 ユーティリティーは、ターゲット・データベースに含まれるレコードのサブセット 1 つのみを分析します。使用する z/OS データ・スペースは 3 つだけです。KR= キーワードを指定した場合、サンプリング・オプションはサポートされません。

サブセクション:

- 322 ページの『制約事項』
- 322 ページの『前提条件』
- 322 ページの『要件』
- 322 ページの『推奨事項』

- 『入出力』
- 323 ページの『JCL 指定』
- 324 ページの『戻りコード』

## 制約事項

DFSMAID0 ユーティリティーは、全機能データベースのみで機能します。

データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。

## 前提条件

現在、DFSMAID0 ユーティリティーに対して文書化された前提条件はありません。

## 要件

DFSMAID0 ユーティリティーでは、ターゲット・データベースの分析によって生成された統計は、データ・スペースを使用して格納およびソートする必要があります。

## 推奨事項

大規模なデータベースには、サンプリングを使用してください。サンプリングは、SAMPLE キーワードでサンプル・サイズを指定すると使用可能になります。SAMPLE キーワードは、NBR または MAX の EXEC ステートメント変数で使用できます。KR EXEC ステートメント変数には使用できません。

DFSMAID0 ユーティリティーに適したデータ・スペースが定義されていることを確認してください。通常、データ・スペースはインストール時に定義されます。z/OS User Step Initiation インストール・システム出口 (IEFUSI) によって、ユーザー作業に割り振られるデータ・スペースの数およびストレージ・サイズがさらに制限される場合もあります。

## 入出力

DFSMAID0 ユーティリティーは次の入力を使用します。

- HALDB 区画ごとのキー範囲
- HALDB 区画ごとの最大バイト数 (拡張のためのスペースを残す)
- 必要な HALDB 区画数
- ランダム・サンプルによる推定が必要な場合のサンプル・サイズ

DFSMAID0 ユーティリティーは、以下の出力を作成して、生成した報告書に表示します。

- ユーティリティーに対して与えられた入力制御ステートメント
- 総バイト数 — 接頭部 + 現行のデータベース用に存在するデータ長 (KB)
- データベース・レコード数
- タイプ別のセグメント数
- 新しい HALDB で作成された接頭部のサイズの増加分 (バイト)

- 新しい HALDB で作成された物理対による増加分 (バイト)

データベース全体の接頭部サイズの合計が増加するのは、論理親を指すために EPS (拡張ポインター・セット) が使用されるからであり、また、仮想論理子を置き換えるために物理論理子が追加されるからです。

以下の図は、DFSMAID0 ユーティリティで生成された統計報告書の例です。キーの長さについて、区画キーがダンプ形式で作成されます。

```
partition 1:
minimum key=
+0000 d2c1c1f1f1          |KAA11      |
maximum key=
+0000 d2f2f3f9 f9        |K2399      |
segment name  segments  bytes      pref-incr  pair-inc
1. 'K1      '      263      14728      2104      0
2. 'K2      '      37       1036      296       0
3. 'K3      '      68       3808      2176      0
4. 'K4      '      35        560       420       0
5. 'K5      '      46       1656      368       0
6. 'K6      '      40        640       480       0
```

上記は、各 HALDB 区画と、その後に総合計が続いた報告書を示しています。データベース全体の接頭部サイズの総増加分が示されます。この総増加分は、次の理由によるものです。

- 論理親を指すための EPS の使用
- 仮想論理子を置き換えるための物理論理子の追加による、合計バイト数の増加

区画が空の場合、最小キーは 0xFFFF..FF に設定され、最大キーは 0x0000..00 に設定されます。

## JCL 指定

DFSMAID0 ユーティリティは、標準の z/OS ジョブとして実行されます。DFSMAID0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含めることが必須であり可能です。

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を指定する DD ステートメント

### EXEC ステートメント

EXEC ステートメントは次の形式にできます。

```
PGM=DFSRRCO0,PARM='ULU,DFSMAID0,dbname,,0,,,,,,,,,Y,N'
```

EXEC ステートメントに含まれる変数は次のとおりです。

#### dbname

分析すべきターゲット・データベースの名前

- 0 ULU 再始動ではありません
- Y DBRC を活動化します
- N IRLM を活動化しません

**DD** ステートメント

**STEPLIB DD**

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

**DFSRESLB DD**

IMS SVC モジュールが入っている許可ライブラリーを指します。

**IMS DD**

分析するデータベースを記述する DBD と PSB が入っているライブラリーを定義します。このデータ・セットは、直接アクセス装置に置いておかなければなりません。このステートメントは必須であり、必ず DBD ライブラリーを定義しなければなりません。PSB ライブラリーが必要になるのは、PARM=DLI と指定されたときだけです。

**SYSIN DD**

このプログラムのための入力制御データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。システムの C プログラミングでは、SYSIN が DFSMAID0 への C 標準入力 (stdin) になります。LRECL の最大値は 600 バイトです。

**SYSPRINT DD**

メッセージ出力データ・セットを定義します。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

**SYSUDUMP DD**

オプションのダンプ・データ・セットを定義します。

**database DD**

データベース事前再編成ユーティリティの指示に従ってスキャンすべきデータベースを指定します。ユーザーが DFSMDA メンバーを用意していれば、データベースが動的に割り振られるため、この DD ステートメントは不要です。DD 名は、DBD に指定されている DD 名と一致する必要があります。このデータ・セットは、直接アクセス装置に置いておかなければなりません。

**DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。

**戻りコード**

プログラム終了時に次の戻りコードが返されます。これらの戻りコードは、オフセット X'50' のジョブ・ステップ・ドキュメンテーションにあります。

コード 意味

- 0 正常終了。



- 12 ユーティリティーは異常終了した
- 0100 キーに事前割り振りされたアクセス・リスト項目テーブル (ALET) スロットが少なすぎた
- 0101 recStats に事前割り振りされた ALET スロットが少なすぎた
- 0102 索引に事前割り振りされた ALET スロットが少なすぎた
- 0103 16 MB より大きいデータベース・レコードが検出された

メッセージ DFS2397E によって生成されるダンプでは、DSPSERV (データ・スペースの作成) マクロ戻りコードは 4 バイトではオフセット X'00' にあります。メッセージの説明は、「z/OS MVS™ Programming: Authorized Assembler Services Reference」に記載されています。理由コード xxNNNNxx 上のマスクの一部に NNNN を使用して、エラーの原因を判別します。

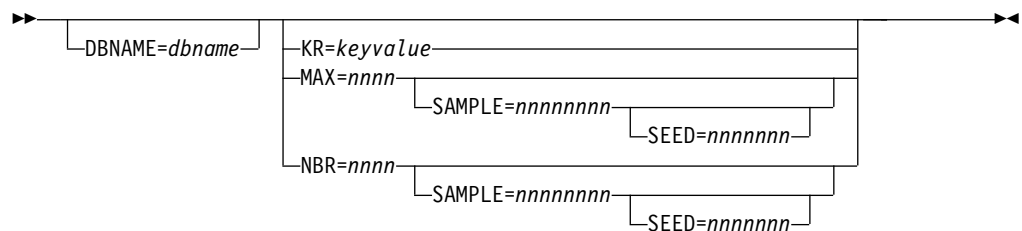
## DFSMAID0 ユーティリティーの制御ステートメント

HALDB マイグレーション・エイド・ユーティリティーの処理オプションを記述するために使用できる入力ステートメントは、6 つあります。

SYSIN 入力ファイルには、次の構文規則と制約があります。

- コメントの先頭として 1 桁目に「\*」を使用します。
- 大/小文字混合は構いません。
- 1 つのステートメントに複数の値が含まれるとき、その値を 2 つの行に分けてはなりません。しかし、複数のステートメントを使用することは可能です。
- 桁に関する制約はありません。
- 1 回の実行で行える分析は、KR、NBR、または MAX のいずれか 1 つです。

以下の構文図は、HALDB マイグレーション・エイド・ユーティリティー制御ステートメントの形式を示します。



### DBNAME

スキャンすべきデータベースを指定します。指定はオプションです。

**KR** KR は各 HALDB 区画に必要な最大キー値を指定します。各区画に対して 1 つの KR パラメーターが必要です。ただし、ハイ・キーの最大値または大きい値の区画は除きます。分析には必ず大きい値の区画が含まれます。このパラメーターは、MAX や NBR と同時に指定することはできません。

KR は SAMPLE キーワードをサポートしていません。両方を指定すると、予測不能な結果を引き起こすことになります。

キー値を指定する場合は、値が 16 進値または文字値のいずれであるかを指定し、キー値を単一引用符で囲む必要があります。例えば、文字キーの場合は `KR=C'key_value'` とコーディングし、16 進キーの場合は `KR=X'key_value'` とコーディングします。

キーを、DBD で定義した長さと同じにする必要はありません。キー・サイズいっぱいになるまで、`X'FF'` などの大きい値が埋め込まれます。

#### MAX

新しいデータベースの各 HALDB 区画に必要な最大バイト数を指定します。NBR や KR と同時に指定することはできません。

#### SAMPLE

SAMPLE は区分するランダム・サンプルのサイズを指定します。サンプル・サイズを指定するときは、同じステートメント内で後ろにシード・キーワードを置くことができます。サンプル・サイズ指定によるサンプリングの選択は、最初の制御ステートメントで行ってください。大規模なデータベースには、サンプリングをお勧めします。SAMPLE は、NBR または MAX とは併用できますが、KR とは併用できません。SAMPLE 指定は、他のどの指定より前に置いてください。

#### SEED

SEED は出発点となるランダム・シード値を指定します。SAMPLE を指定し、SEED を指定しないと、SEED はデフォルトで 1 となります。SAMPLE と SEED は、同じ制御ステートメントにコーディングしてください。

#### NBR

新しいデータベースに必要な HALDB 区画数を指定します。1 から 1001 を指定できます。MAX や KR と同時に指定することはできません。

#### 注:

1. SAMPLE、MAX、NBR のいずれかを含むステートメントは、1 つの入力セットに 1 つしか使用できません。
2. 1 つの入力セットでは、MAX、NBR、KR のいずれか 1 つしか使用できません。
3. KR ステートメントは、HALDB 区画の最大数と同じ数だけ使用できます。

---

## DFSMAIDO ユーティリティーの例

区画を制御するための入力パラメーターの使用法について、以下の例を示します。

サブセクション:

- 『3 つの区画定義』
- 327 ページの『MAXX キーワードの使用法』
- 327 ページの『区画分析中のサンプル』

### 3 つの区画定義

以下の例に示す入力では、3 つの区画が定義されています。最初の 2 つの区画は、例に示す値で制限されます。3 番目の区画は、キーがとりうる最大の値で終わります。

す。

図 64. 区画の定義

```
//SYSIN DD *  
KR=C'1050'  
KR=X'F4F5F6F7'
```

## MAXX キーワードの使用法

以下に示す例には、MAX キーワードが使われていて、各区画に最大 750,000,000 バイトを割り振るよう定めています。

図 65. MAX キーワードの使用

```
//SYSIN DD *  
MAX=750000000
```

## 区画分析中のサンプル

以下に示す例では、区画分析の際にサンプリングが実行されます。サンプル・サイズは 10000 です。SEED はデフォルトで 0 になります。区画数は 100 です。

図 66. 区画分析

```
//SYSIN DD *  
SAMPLE=10000  
NBR=100
```

---

## DFSMAID0 ユーティリティーのデータ・スペースおよびサンプリング

大規模な IMS データベースで DFSMAID0 ユーティリティーを効率的に使用するには、さまざまな分析を指定したときに DFSMAID0 ユーティリティーがどのように動作するかを理解する必要があります。

DFSMAID0 ユーティリティーは、データベース分析によって生成される統計の格納および処理のために、最大 4 つのデータ・スペース (各 2 GB) を割り振るよう要求することができます。DFSMAID0 が使用するデータ・スペース・ストレージ量を計算したり、さらに可能であれば、分析の土台としてターゲット・データベース内のレコードのサンプルを使用することによって、DFSMAID0 ユーティリティーがより効率的にデータ・スペースを使用できるようになります。

サブセクション:

- 『データ・スペースおよび分析の種類』
- 328 ページの『データ・スペースの補助ストレージ要件の計算』
- 328 ページの『サンプリングの概要と技法』

## データ・スペースおよび分析の種類

実行する分析の種類によっては、DFSMAID0 は、IMS データベース内の各レコードの圧縮統計を格納し、その後仮想メモリー内でこれらの統計をキーでソートする

場合もあります。IMS データベースには 10 億近くものレコードが格納されることもあるため、圧縮統計であっても、統計の格納とソートには膨大な数値計算が必要になり、かなりの量の CPU リソースを使用する可能性があります。

DFSMAID0 ユーティリティによるデータ・スペースの使用方法は、実行する分析の種類に応じて異なります。

- 入力制御ステートメントで KR キーワードを指定した場合、DFSMAID0 ユーティリティは統計の格納およびソートを行いません。DFSMAID0 ユーティリティの実行に要する時間は、そのほとんどが `get-next` 呼び出しによるデータベースの全探索に費やされます。サンプリングは適用されません。
- MAX キーワードを指定した場合、DFSMAID0 ユーティリティは、索引付きの IMS データベースのときを除いて、レコード統計を格納およびソートする必要があります。

レコード統計がキー・シーケンスになると、最後の区画を除いて、DFSMAID0 ユーティリティは指定した最大バイト数になるまで、繰り返し連続した区画を埋めていきます。

DFSMAID0 ユーティリティで IMS データベース全体をランダムにサンプリングし、補間された区画分析結果を生成することによって、格納とソートに要する時間とストレージを最小限に抑えることができます。

- NBR キーワードを指定した場合、DFSMAID0 ユーティリティは IMS データベースを全探索し、圧縮レコード統計をデータ・スペースに格納し、索引付きデータベースのときを除いて、レコード統計をキーでソートします。

レコード数が分かり、レコードがキー順でアクセス可能になると、DFSMAID0 は最大区画サイズを計算し、連続した区画を計算された最大値まで埋めていきます。

NBR を指定すると、サンプリングにより、格納するデータ量を大幅に減らすことができ、またソートのパフォーマンスも向上します。

## データ・スペースの補助ストレージ要件の計算

全データベース分析を実行する場合は、分析の実行に必要なストレージ量を次の計算で決定してください。

データ・スペースが 3 つ以上で、レコードが  $n$  個の場合:

$$n \times (4 + \text{ルート・キーの長さ} + 12 \times \text{セグメント・タイプの数}) + 12288$$

サンプリング分析を実行する場合は、分析の実行に必要なストレージ量を次の計算で決定してください。

データ・スペースがちょうど 3 つで、サンプル・サイズが  $n$  の場合:

$$n \times (8 + \text{ルート・キーの長さ} + 12 \times \text{セグメント・タイプの数}) + 12288$$

## サンプリングの概要と技法

DFSMAID0 ユーティリティは、複数データ・スペースのヒープ・ソート操作を使用して、DFSMAID0 ユーティリティが割り振ったレコード索引をキーの昇順に再配置します。ソートの計算量は  $n \cdot \log_2(N)$  に比例して増大します。ここで、 $n$  はサ

ンプル・サイズと全レコード数のうちで小さいほうです。次の理由から、サンプリングが従来より実用的であるため、ヒープ・ソート操作は、キャッシュ、テーブル索引バッファ (TLB)、その他同様の操作の妨げにはなりません。

使用しているデータベースの完全な分析を実行できないことがあるため、ここに示す例はサンプリング結果を確認するための方法を示しています。ただし、同じ結果になるとは限りません。

- オプションの制御カード `SAMPLE=n` を指定すると、`DFSMAID0` ユーティリティー区画はサイズが `n` レコードのランダム・サンプルを区分し、データベース内のレコード数とサンプル内のレコード数の比で、表形式の数値をスケールリングします。レコードの構造がほぼ似ている均質なデータベースでは、サンプリングが非常に正確に行われます。

例えば、1 から 2,147,483,646 の整数から 20,000 個の整数をランダムに選択し、サンプルをソートし、中央の 2 個 `sample[10000]` と `sample[10001]` の平均をとります。この試みを 100 回繰り返した結果、指定値のすべてが、実際の中央値である 1,073,741,823.5 の 1.78% 以内に入りました。

IMS データベースの二等分も、キーでソートし、中央値で二等分するという、同様な問題です。サイズ 20000 のサンプルでは、正確な結果の 1% から 2% 以内の結果が得られるはずですが、特にデータベースが静的でない場合に、2% 未満の精度向上のために、桁数が何桁も多くなるほどのデータ量を格納してソートする必要はありません。データベースの規模が大きくなるほど、同程度の精度を得るのに必要なサンプルの規模が大きくなるわけではありません。

- サンプルをデータベース全体のサイズまで拡大するための倍率は、以下のとおりです。

(データベース  
内のレコード数) ÷ (サンプル内のレコード数)

そのため定義上は、推定区画内のルート・セグメント数は、データベース内のルート・セグメントの総数と完全に一致しますが、その他の推定値は、データベースについて対応する総数の概算値です。

- サンプリングの安定度は、さまざまなランダム・シードを試すことで確認できます。シードは、ランダムマイザーの基本となるランダム値として使用されます。例えば「`sample=10000, seed=7`」と指定します。区分化は、レコードの格納もソートも行わないキー範囲 (KR) 分析を使用することで確認および改良できます。この場合も、サンプリングはオプションですが、大規模なデータベースにはお勧めします。

ランダムマイザーの出力は、キー生成アルゴリズムの入力パラメーターとして使用されます。このアルゴリズムは、データベースにランダムにアクセスするために使用されます。最適な結果を得るには、大きな奇数を指定してください。シードには素数が適しています。



---

## 第 24 章 HD 再編成再ロード・ユーティリティ (DFSURGL0)

HD 再編成再ロード・ユーティリティ (DFSURGL0) は、HD 再編成アンロード・ユーティリティ (DFSURGU0) によって作成された出力データ・セットを使用して、データベースおよび HALDB 区画を再ロードします。

DFSURGL0 ユーティリティは、以下のタイプのデータベースをサポートします。

- HDAM
- PHDAM
- HIDAM
- PHIDAM
- HISAM
- PSINDEX

DFSURGL0 ユーティリティは、データベース・レコードの該当するセグメントのすべてについてシーケンス検査を行います。

非パーティション・データベースに論理関係または 2 次索引が含まれる場合、DFSURGL0 ユーティリティは作業データ・セットを作成します。この作業データ・セットは、論理関係または 2 次索引関係を解決するために、データベース接頭部解決ユーティリティによって使用されます。HALDB パーティション・データベースを再編成するときに、ユーザーが論理関係または索引関係を解決する必要はありません。

PHDAM および PHIDAM データベースの場合、DFSURGL0 ユーティリティは、それぞれの区画の間接リスト・データ・セット (ILDS) を更新するためのいくつかのオプションを提供します。DFSURGL0 ユーティリティ制御ステートメントの指定または省略により、ILDS 更新オプションを選択することができます。

ILDS 更新オプションには以下の機能があります。

- 区画データ・セットの順次ロード中に、論理関係または 2 次索引のターゲット・セグメントが DFSURGL0 ユーティリティによって読み取られるたびに、ILDS を更新します。DFSURGL0 ユーティリティは、VSAM 更新モードを使用して ILDS を更新します。これは、ユーティリティ制御ステートメントが指定されない場合の DFSURGL0 ユーティリティのデフォルトの方式です。
- 区画データ・セットの順次ロード後の ILDS の更新は、VSAM ロード・モードおよび単一タスクの使用により完了します。VSAM ロード・モードを使用して、DFSURGL0 ユーティリティは VSAM DEFINE ステートメントで指定されたフリー・スペースを ILDS に入れます。このオプションは、ILDSINGLE ユーティリティ制御ステートメントを指定することにより選択できます。DFSURGL0 ユーティリティには、処理のために使用可能な z/OS データ・スペースが 5 つなければなりません。
- 非パーティション・データベースを HALDB に変換する場合、区画データ・セットの順次ロード後の ILDS の更新は、VSAM ロード・モードおよび複数のタスクの使用により完了します。VSAM ロード・モードを使用して、DFSURGL0

ユーティリティーは VSAM DEFINE ステートメントで指定されたフリー・スペースを ILDS に入れます。このオプションは、ILDSMULTI ユーティリティー制御ステートメントを指定することにより選択できます。DFSURGL0 ユーティリティーには、処理のために使用可能な z/OS データ・スペースが 5 つなければなりません。

- DFSURGL0 ユーティリティーが ILDS を更新しないようにします。このオプションを選択すると、HALDB 索引/ILDS 再作成ユーティリティー (DFSPREC0) などの他の方法で ILDS を更新する必要があります。このオプションは、NOILDS ユーティリティー制御ステートメントを指定することにより選択できます。

HDAM および HIDAM データベースの場合、このユーティリティーの機能は、ユーティリティー制御機能 (UCF) によって実行できます。UCF は HALDB データベースには使用できません。

HD 再編成再ロード・ユーティリティーのフロー・ダイアグラムを、以下の図に示します。

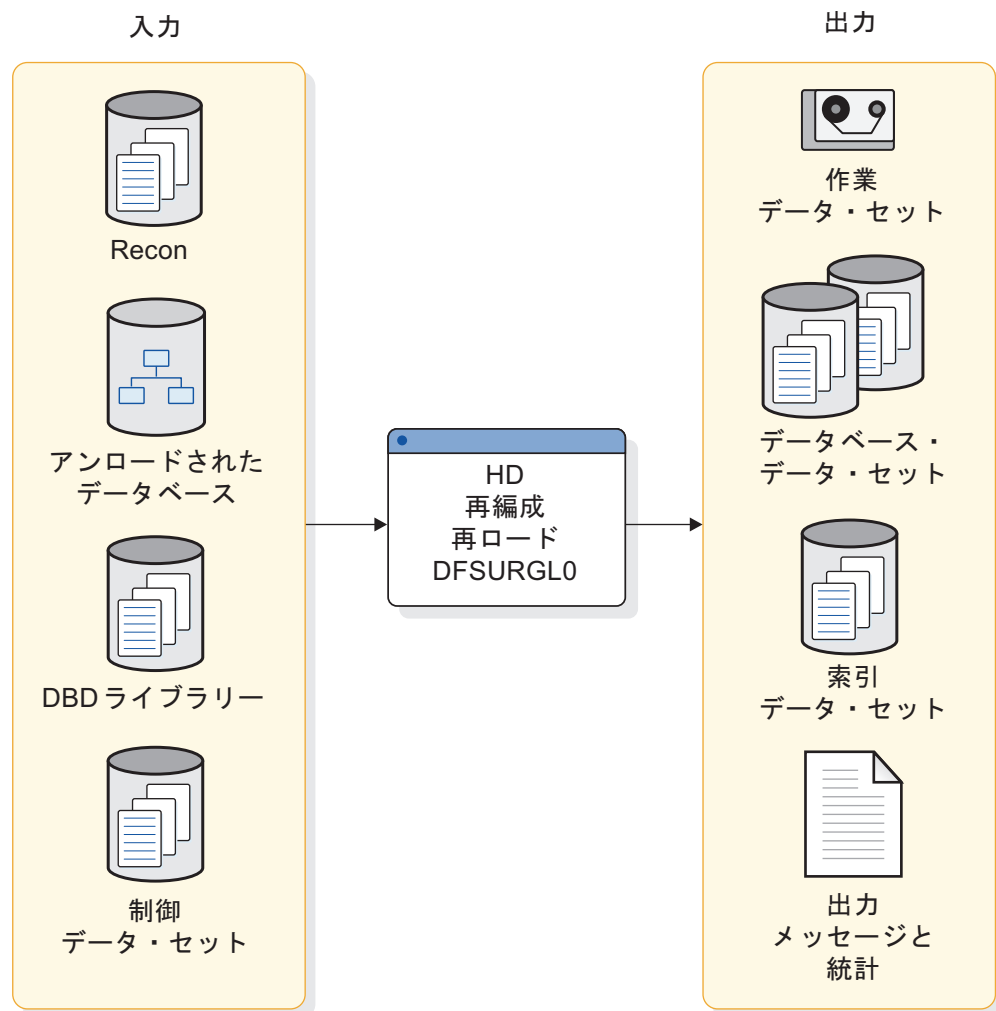


図 67. HD 再編成再ロード・ユーティリティー

サブセクション:



- 『制約事項』
- 334 ページの『前提条件』
- 334 ページの『要件』
- 334 ページの『推奨事項』
- 335 ページの『入出力』
- 335 ページの『JCL 指定』
- 339 ページの『戻りコード』

## 制約事項

DFSURGL0 ユーティリティを使用するときは、次のような制約があります。

- 次の場合は、DFSURGL0 ユーティリティを使用しないでください。
  - いずれかの論理子セグメントが物理パスまたは論理パスのいずれか (両方のパスからではなく) から削除されていたとき、両方向仮想対から両方向物理対に変更する場合。
  - 実論理子を、論理的に関係を持つ一方のデータベースから別のデータベースへ変更する場合。
  - HIDAM データベースへの索引となる HISAM データベースを再編成する場合。その場合には、HISAM 再編成再ロード・ユーティリティ (DFSURRL0) を使用してください。
- HALDB データベースを再ロードする場合は、DFSURGL0 ユーティリティでユーティリティ制御機能を使用することはできません。
- DFSURGL0 ユーティリティへの入力の際に、複数の HALDB アンロード・ファイルを連結できます。ただし、それらのアンロード・ファイルが HD 再編成アンロード・ユーティリティ (DFSURGU0) によって作成されるときに以下のいずれかの制御ステートメントが指定されていた場合を除きます。
  - MIGRATE
  - MIGRATX
  - FALLBACK
- ILDSINGLE SYSIN ステートメントは、既存の HALDB データベース区画を再編成するためにのみ使用できます。データベースを HALDB に変換する際には使用できません。
- ILDSMULTI SYSIN ステートメントは、データベースを HALDB に変換するためにのみ使用できます。既存の HALDB データベース区画の再編成には ILDSMULTI SYSIN ステートメントを使用しないでください。

DFSURGL0 ユーティリティは、全機能 (HALDB および非 HALDB) データベースのみで機能します。

データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

## 前提条件

DFSURGL0 ユーティリティを実行する前に、HD 再編成アンロード・ユーティリティ (DFSURGU0) を実行して、DFSURGL0 ユーティリティ用の入力ファイルを作成しなければなりません。

DBD を変更する場合は、HD 再編成再ロード・ユーティリティを実行する前に、以下の作業を完了する必要があります。

1. 新しい DBD をアSEMBルし、バインドしてIMS DBD ライブラリーに入れておく。
2. 新しいセグメントを追加するか、あるいはセグメントを削除する場合に、データベースが論理関係を含んでいるときは、新しい DBD に対してデータベース事前再編成ユーティリティを再実行する。
3. DBD 名が変更されて、その DBD が論理関係を含んでいるときは、新しい DBD に対してデータベース事前再編成ユーティリティを再実行する。

## 要件

DFSURGL0 には以下の要件があります。

- ILDSINGLE SYSIN ステートメントで HALDB 区画を再ロードする場合は、ILDS VSAM クラスターを削除して再定義する必要があります。
- PHIDAM データベースの場合、DFSUINPT DD ステートメントを HD 再ロードに実行依頼する前に、すべてのレコードがキー順になっていることを確認する必要があります。HD 再ロードは、入力 DD ステートメントのレコードの順序を確認しません。さらに、次の点に注意してください。
  - DFSUINPT DD ステートメントを連結する場合は、すべてのレコードが各区画内でキー順になっている必要があります。
  - 区画選択出口ルーチンに変更を加える場合、または区画選択出口ルーチンの使用後にキー範囲区分化を変更する場合は、DFSUINPT DD ステートメントを HD 再ロードに実行依頼する前に、すべてのレコードの再ソートが必要になることがあります。
- このユーティリティを RECON データ・セットに登録されていない IMS カタログ・データベースで使用するには、追加の EXEC パラメーター DFSDF を指定する必要があります。DFSDF= は、未登録のカタログ HALDB の名前またはメンバー名を指定します。例えば、次のようにします。

```
//HDRELOAD EXEC PGM=DFSRR00,  
// PARM=(ULU,DFSURGL0,DFSCD000,,,,,,,,,N,N,,,,,,,,,,,,,,,,,,,,,  
// 'DFSDF=CAT')
```

3 文字からなる接尾部 CAT を、未登録の IMS カタログ・データベースの名前を指定する DFSDFxxx メンバーの接尾部で置き換えてください。

## 推奨事項

DFSURGL0 ユーティリティの使用および実行については、以下の推奨事項があります。

- バッファ・プールのサイズは、データベースの再ロード時のパフォーマンスに影響します。最良の結果を得るために、データベース・レコード (ルート・セグメントとそのすべての従属セグメント) が収まるだけのブロック数がバッファ

ー・プールにあることを確かめてください。既に作成されているブロックは、セグメントが挿入されるときに参照されます。すなわち、前のブロックに入っている親または兄弟として指されます。これらの 2 セグメントの間に多数のセグメントが挿入されていると、親または兄弟が置かれているバッファはデータ・セットに書き込まれている可能性があるため、もう一度読み込む必要があります。

- 接頭部解決ユーティリティー (DFSURGL0) の作業データ・セットの DD ステートメントに DCB パラメーターを定義する場合、LRECL=900 が望ましい値ですが、2 次索引が存在しなければ、もっと小さな値 (300 程度の値) が使用できます。
- 論理関係または副次索引を持つ PHDAM 区画または PHIDAM 区画をロードするときは、DFSURGL0 ユーティリティーによる間接リスト・データ・セット (ILDS) の更新方法について、いくつかのオプションがあります。DFSURGL0 ユーティリティー制御ステートメントの指定または省略により、オプションを選択します。ユーティリティー制御ステートメントの指定を、その使用が推奨される環境とともに以下にリストします。

## 入出力

DFSURGL0 ユーティリティーへの 1 次入力、HD 再編成アンロード・ユーティリティーで作成されたアンロード・データ・セットです (ただし、ほかの入力データも必要です)。DFSURGL0 ユーティリティーの 1 次出力はロードされたデータベース・データ・セットであり、索引データ・セットと作業データ・セットを含むことがあります。

HD 再編成再ロード・ユーティリティーは、出力メッセージと統計を生成します。このユーティリティーは、要約報告書 (SYSIN データ・セットで STAT=DET を指定している場合は詳細報告書) を生成します。

HD 再編成再ロード・ユーティリティーの入出力は、以下の表に示すとおりです。

表 14. HD 再編成再ロード・ユーティリティーが使用するデータ・セット

入力	出力
RECON	出力メッセージと統計
アンロードされたデータベース	索引データ・セット
DBD ライブラリー	データベース・データ・セット
制御データ・セット	作業データ・セット

## JCL 指定

HD 再編成再ロード・ユーティリティーは、標準の z/OS ジョブとして実行されます。次の JCL ステートメントが必要です。

- ユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

EXEC ステートメントは次の形式になっていなければなりません。

PGM=DFSRR00,PARM='ULU,DFSURGL0,dbdname'

パラメーター ULU と DFSURGL0 は、ユーティリティー領域を記述します。変数 *dbdname* は、再ロードするデータベースを定義する DBD の名前です。SPIE、BUF、DBRC といった通常の IMS 定位置パラメーターは、*dbdname* の後に置くことができます。

関連資料: バッチ処理領域の実行について詳しくは、以下を参照してください。

- DBBBATCH プロシージャ (システム定義)
- DLIBATCH プロシージャ (システム定義)

#### DD ステートメント

##### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

##### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

##### IMS DD

EXEC ステートメントの PARM フィールドで参照される DBD が入っているライブラリーを記述します。(通常、これは IMS.DBDLIB です。) このデータ・セットは、直接アクセス装置に置いておかなければなりません。

注: HALDB 変更操作が開始されたが、オンライン変更機能が完了していない HALDB データベースを再ロードする場合、変更された DBD を含む DBD ライブラリーではなく、元の DBD を含む DBD ライブラリーを指定する必要があります。データベース内の一部またはすべてのセグメントが、変更された DBD が定義するフォーマットに従っているかどうかに関わらず、元の DBD を指定してください。IMS はステージング ACB ライブラリーから変更された DBD の情報を自動的に取得し、元の DBD との比較を要求します。

##### SYSPRINT DD

メッセージ出力データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

##### SYSIN DD

HALDB マイグレーション再ロード・ステートメントの NOILDS、ILDSMULTI、および ILDSINGLE について、入力制御ステートメント・データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。ユーティリティーへの入力になるユーティリティー制御ステートメントがない場合、この DD ステートメントは必要ではありません。

##### DFSUINPT DD

再ロードするデータが入っている入力データ・セットを記述します。これは、HD 再編成アンロード・ユーティリティーによって作成されたデータ・セットです。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておかなければなりません。HALDB データベースの場合、HD 再編成再ロード・ユーティリティーによって再ロードされるように複数のデータ・セットを連結で

きます。ただし、必要なデータ・セットをすべて入れ、重複するデータ・セットを入れていないことを必ず確認してください。HD 再ロードでは、データ・セットの確認が行われません。

要件: PHIDAM データベースの場合、DFSUINPT DD ステートメントを HD 再編成再ロード・ユーティリティに実行依頼する前に、すべてのレコードがキー順になっていることを確認する必要があります。HD 再編成再ロード・ユーティリティは、入力 DD ステートメントのレコードの順序を確認しません。さらに、次の点に注意してください。

- DFSUINPT DD ステートメントを連結する場合は、すべてのレコードが各区画内でキー順になっている必要があります。
- 区画選択出口ルーチンに変更を加える場合、または区画選択出口ルーチンの使用後にキー範囲区分化を変更する場合は、DFSUINPT DD ステートメントを HD 再編成再ロード・ユーティリティに実行依頼する前に、すべてのレコードの再ソートが必要になることがあります。

#### **DFSURWF1 DD**

再ロード時に作成する作業データ・セットを記述します。これは、論理関係または副次索引関係を解決するために、接頭部解決ユーティリティ (DFSURG10) によって入力として使用されるものです。再ロードするデータベースが、論理関係にも副次索引にも関連しないか、あるいはデータベースが PHIDAM または PHDAM の場合は、データ・セットの DD 名を DUMMY と指定することができます。PHIDAM または PHDAM データベースには接頭部解決は必要ありません。

DD ステートメントの DCB パラメーターとして、RECFM=VB と BLKSIZE が必要です。BLKSIZE は、ユーザーの初期ロード・プログラムの作業データ・セットまたはデータベース・スキャン・ユーティリティ (DFSURGS0) に関して指定されたものと同じになるように指定する必要があります。

推奨事項: LRECL=900 が望ましい値ですが、副次索引が存在しなければ、もっと小さな値 (300 程度の値) が使用できます。

このデータ・セットは、テープか直接アクセス装置のどちらかに置いておかなければなりません。

#### **database DD**

再編成するデータベース・データ・セットを定義します。このタイプのステートメントは、このデータベースを記述する DBD に入っているデータ・セットごとに 1 つ必要です。DD 名は、DBD にある DD 名と一致している必要があります。

これが HIDAM データベースのときは、索引を表しているデータ・セットにも、DD ステートメントが必要です。索引と関係がある DD ステートメントには、索引データベースの DBD に指定されている DD 名が含まれていなければなりません。このデータベースに関連する副次索引の場合も、DD ステートメントは必要ありません。

このデータ・セットは、直接アクセス装置に置いておかなければなりません。

#### **DFSURCDS DD**

このプログラムのための制御データ・セットを定義します。このデータ・セット

は、データベース事前再編成ユーティリティ (DFSURPRO) によって生成された出力でなければなりません。論理関係が存在するときは、この DD ステートメントが必要です。

このデータ・セットは、テープ装置か直接アクセス装置のどちらかに置いておかなければなりません。

#### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ・プール情報が入っているデータ・セットを記述します。このステートメントは必須です。

**推奨事項:** バッファ・プールのサイズは、データベースの再ロード時のパフォーマンスに影響します。最良の結果を得るために、データベース・レコード (ルート・セグメントとそのすべての従属セグメント) が収まるだけのブロック数がバッファ・プールにあることを確かめてください。既に作成されているブロックは、セグメントが挿入されるときに参照されます。すなわち、前のブロックに入っている親または兄弟として指されます。これらの 2 セグメントの間に多数のセグメントが挿入されていると、親または兄弟が置かれているバッファはデータ・セットに書き込まれている可能性があるため、もう一度読み込む必要があります。

**関連資料:** 制御ステートメント・フォーマットおよびバッファ・プール構造について詳しくは、『IMS バッファ・プール (システム定義)』を参照してください。

#### **DFSCTL DD**

順次バッファリングの活動化を要求する SBPARM 制御ステートメントが入っているデータ・セットを記述します。順次バッファリングを条件付きで活動化させると、OSAM DB データ・セットのバッファリングのパフォーマンスが向上し、ジョブ経過時間が短縮されます。

DFSCTL ファイルは順次データ・セットにすることも、PDS のメンバーにすることもできます。レコード形式は F、FB、または FBS のいずれかで、レコード長は 80 でなければなりません。このデータ・セットは、直接アクセス装置またはテープに置いておくことも、入力ストリームに入れて送ることもできます。

**関連資料:** SBPARM 制御ステートメントについての説明は、『順次バッファリング制御ステートメントの指定 (システム定義)』を参照してください。

#### **SYSABEND DD または SYSUDUMP DD**

ダンプ・データ・セットを定義します。どちらかのステートメントが指定されている場合、4 より大きい戻りコードが返されると、ABEND U0355 が起こります。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

#### **RECON1 DD**

最初の DBRC RECON データ・セットを定義します。

#### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。

#### **RECON3 DD**

RECON1 または RECON2 にエラーが見付かったとき使用される、オプション

の DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

## 戻りコード

プログラム終了時に次の戻りコードが返されます。

- 0 データベースの再ロードが正常に完了した
- 4 再ロードすべきセグメントがなかった
- 8 再ロード・カウントがアンロード・カウントと異なっている、または ILDS 処理でエラーが発生した
- 16 データベースの再ロードが正常に完了しなかった

関連資料:

437 ページの『第 31 章 ユーティリティー制御機能 (DFSUCF00)』

---

## DFSURGL0 ユーティリティーのユーティリティー制御ステートメント

DFSURGL0 ユーティリティーは、ユーザーが処理オプションを指定するために使用できる制御ステートメント NOILDS、ILDSMULTI、および ILDSINGLE を受け付けます。

制御ステートメントを指定しない場合、DFSURGL0 は区画データ・セットを順次ロードし、間接リスト項目 (ILE) のターゲット・セグメントがロードされるとすぐに、ILDS 内の各 ILE を更新します。DFSURGL0 は、「VSAM 更新モード」を使用して ILDS を更新します。DFSURGL0 が区画データ・セットのロードを終了すると、ILDS の再作成も終了します。VSAM 更新モードでは、ILDS 内のフリー・スペースの追加や保守は行われません。

次の場合は、ユーティリティー制御ステートメントを指定しないでください。

- リカバリー目的で、削除されたセグメント用の ILE を保持する場合
- 最小限の更新のみを必要とする ILDS を更新する場合

DFSURGL0 ユーティリティー制御ステートメントには、以下のものが含まれます。

### NOILDS

既存の HALDB データベースまたは新しく変換された HALDB データベースの再ロードのときに ILDS 処理を使用不可にします。

NOILDS を使用すると、新しい ILE が必要になるたびに ILDS データ・セットへの入出力が発生しなくなります。このオプションを指定すると、再ロードされる各区画に関連付けられた ILDS が、リカバリーが必要であるとして RECON データ・セットに記録されます。ILDS 再作成ユーティリティー (DFSREC0) を実行すると、別のジョブで ILDS が再作成されます。これにより、各 ILDS を作成するために HALDB 区画全体が読み取られるため、全体の実行時間が長くなります。このステートメントを大規模なアンロード・ファイ

ルに使用し、区画ごとに DFSURGL0 を別のジョブで実行すると、マルチタスキングのメリットが得られます。マイグレーションのアンロード・ファイルに対して、NOILDS の手法は役立ちます。

推奨事項: NOILDS 制御ステートメントが指定されると、DFSURGL0 ユーティリティーは ILDS を更新せずに区画をロードします。NOILDS オプションを指定すると、HALDB データベースを最も高速に再ロードできます。ただし、ご使用の HALDB データベースが論理関係または副次索引を使用する場合、索引/ILDS 再作成ユーティリティー (DFSURGL0) などの他の方法を使用して ILDS を更新する必要があります。NOILDS 制御ステートメントは、再編成再ロードおよびマイグレーション再ロードのどちらにも使用できます。

### ILDSMULTI

全機能 DL/I データベースを HALDB データベースにマイグレーションするためのマルチタスク ILDS 処理を使用可能にします。

ILDSMULTI を使用すると、ILK 順で ILE レコードをソートするために、複数の TCB が接続されます。DFSURGL0 ユーティリティーには、ロードされる区画ごとに合計 5 つのデータ・スペースが必要です。データ・スペースは、必要に応じて拡張されます。このオプションを使用する場合は、データ・スペースのシステム調整が必要になることがあります。ILDSMULTI はマイグレーション再ロードのみに使用されます。

推奨事項: 非パーティション・データベースを HALDB に変換する場合は ILDSMULTI 制御ステートメントを使用します。ILDSMULTI 制御ステートメントを指定すると、ILDSINGLE 制御ステートメントの場合と同様に、DFSURGL0 は区画データ・セットを順次ロードし、区画ごとにユーティリティーが必要とする 5 つのデータ・スペースのうちの 1 つに ILE を保管します。DFSURGL0 は区画データ・セットのロードを終了すると、ILE をその間接リスト・キー (ILK) によってソートした後、VSAM ロード・モードを使用して、1 度の受け渡しですべての ILE を ILDS にロードします。各 ILE は、ILDS を定義する VSAM KSDS DD ステートメントで指定された量のフリー・スペースでロードされます。

制約事項: ILDSMULTI 制御ステートメントを使用するには、DFSURGL0 ユーティリティーに、区画ごとにその処理をするために z/OS データ・スペースが 5 つ使用可能である必要があります。

### ILDSINGLE

単一のタスクにより、特定の区画について ILK (間接リスト・キー) 順で ILE (間接リスト・エントリ) をソートできるようにします。このオプションを使用すると、ILDS 内にフリー・スペースが組み込まれ、再ロードの実行時間が短縮されます。

ILDSINGLE を使用すると、ソートのために、ILE が単一の TCB に渡されます。ソートを行う TCB には、区画ごとに合計 5 つのデータ・スペースが必要です。データ・スペースは、必要に応じて拡張されます。このオプションを使用する場合は、データ・スペースのシステム調整が必要になることがあります。

ILDSINGLE を実行するには、実行前に VSAM クラスターを削除および再定義する必要があります。KSDS のフリー・スペースが有効になり、再ロード中の ILDS の CA または CI の分割が回避されます。ILE は ILK 順でソートされ、関連する区画の再ロード完了後に ILDS に挿入されます。処理中にエラー



が発生すると、識別メッセージが印刷され、リカバリーが必要であるとして ILDS が RECON データ・セットに記録されます。エラー発生後の ILDS の再作成には、ILDS 再作成ユーティリティを使用します。

ILDSINGLE=partname を指定したときに、ILDS にデータがある場合、ターゲット区画の再ロードが正常に完了した後でも、最初の複写キーが入力されるとエラー・メッセージ DFS871E が発行されます。

次に例を示します。

```
DFS871E  VSAM ERROR ENCOUNTERED LOADING ILDS WITH  
DDNAME=partname RC= 00000008 REASON= 00000008
```

次に、リカバリーが必要であるとして、ILDS が RECON データ・セットに記録されます。その後で実行を試みるたびに、ILDS に関連付けられた HALDB 区画へのアクセス前に、ABEND U3303 が発生します。メッセージ DFS047A、DFS3303I、および DFS0832I はエラーに関する有用な情報を提供します。区画、および関連する ILDS を再ロードするには、まず DBRC コマンド CHANGE.DBDS DBD(partname) DDN(ILDSddn) NORECOV を使用してから、ILDS VSAM クラスターを削除および再定義する必要があります。

ILDSINGLE は、マイグレーション・アンロード・ファイルに対して使用してはなりません。その場合には、ILDSMULTI を使用してください。

1 つのファイルの複数の HALDB 区画に対するアンロード・データは、入力として使用できます。DASD にアンロード・データ・セットが存在する場合は、複数の ILDSINGLE ジョブを並行して実行できます。処理は、ILDSINGLE=partname で指定されたターゲット区画に限定され、処理中に他の区画は無視されます。アンロード報告書によって、アンロードされたデータと再ロードされたデータが調整されます。ターゲット区画の FM 状況コードが戻された場合を除いて、メッセージもキーも印刷されません。

推奨事項:

ILDSINGLE 制御ステートメントを指定すると、DFSURGL0 ユーティリティは区画データ・セットを順次ロードし、区画ごとにユーティリティが必要とする 5 つのデータ・スペースのうち 1 つに ILE を保管します。DFSURGL0 ユーティリティは区画データ・セットのロードを終了すると、ILE をその間接リスト・キー (ILK) によってソートした後、VSAM ロード・モードを使用して、1 度の受け渡しですべての ILE を ILDS にロードします。各 ILE は、ILDS を定義する VSAM KSDS DD ステートメントで指定された量のフリー・スペースでロードされます。

制約事項: ILDSINGLE 制御ステートメントを使用するには、DFSURGL0 ユーティリティに、区画ごとにその処理をするために z/OS データ・スペースが 5 つ使用可能である必要があります。

ILDSINGLE 制御ステートメントは、次の場合に使用します。

- 相当な量の更新を必要とする ILDS を再作成する場合
- ILDS 内の CI と CA の分割を除去する場合
- ILDS にフリー・スペースを追加する場合
- 削除されたセグメントを参照する未使用 ILE を除去する場合
- VSAM 更新モードで実行する後続の再ロードの全体的な実行時間を短縮する場合

- DFSURGL0 ユーティリティの実行時間を改善する場合
- データベースを HALDB に変換する場合 (ILDSMULTI オプション指定時のみ)

ILDS が大量の更新を必要とする場合、VSAM 更新モードよりも VSAM ロード・モードのほうが高速です。また、フリー・スペースを追加して将来の分割を防ぐ以外に、CI と CA の分割を除去することで、以後の HALDB 区画の再編成およびリカバリーのパフォーマンスを向上することができます。

データベースを HALDB への変換の一部として再ロードする場合、ILDSINGLE 制御ステートメントはサポートされません。

ILDSINGLE オプションを使用して生成される ILDS は、フリー・スペースの追加のため、HD 再編成ユーティリティの他のオプションを使用して生成される ILDS よりも大きくなるのが普通です。ILDSINGLE により生成される ILDS は、ILEF または BOTHF のいずれかのオプションが指定されたときに、索引/ILDS 再作成ユーティリティ (DFSPREC0) によって再作成される ILDS と同等です。

---

## DFSURGL0 の出力メッセージと統計

HD 再編成再ロード・ユーティリティは、出力メッセージと統計を出力します。

統計は、メッセージの後の出力内に表示され、見出しの下に SEGMENT LEVEL STATISTICS が表示されます。

統計報告書のフィールドには以下のものがあります。

### SEGMENT NAME

この統計行が適用されるセグメントの名前

### SEGMENT LEVEL

このセグメントがデータベースに置かれている階層レベル。セグメントは、DBD に記述されているのと同じ順序で、また HD 再編成アンロード統計に現れるのと同じ順序で、上から下に向かってマップされます。

「TOTAL SEGMENTS BY SEGMENT TYPE」という見出しの下の 2 つのフィールドは次のとおりです。

### RELOADED

このセグメント・タイプがデータベース全体の再ロード時に出現した回数。ユーザー出口によって挿入されたセグメントは、この合計に含まれません。

### DIFFERENCE

このセグメントで再ロード・カウントがアンロード・カウントと等しければ、このフィールドはブランクです。ブランクでないときは、アンロードよりも再ロードのカウント数が大きければ、「+」が右側に付けられます。再ロードよりもアンロードのカウント数が大きければ、「-」が付けられます。

個々のセグメント・タイプの統計に続いて、「TOTAL SEGMENTS IN DATABASE」という見出しが示されます。フィールドは次のとおりです。

### UNLOADED

データベースにあって、HD 再編成アンロード・ユーティリティーによってカウントされたすべてのセグメントの総数。

### RELOADED

データベースにあって、HD 再編成再ロード・ユーティリティーによってカウントされたすべてのセグメントの総数。ユーザー出口によって挿入されたセグメントは、この合計に含まれません。

### DIFFERENCE

再ロードによるカウントとアンロードによるカウントが等しければ、フィールドはブランクです。等しくなければ、その差が印刷されます。

以下の図は、このユーティリティーから得られるメッセージと統計の例です。

```

H I E R A R C H I C A L   D I R E C T   D B   R E O R G   R E L O A D
      S E G M E N T   L E V E L   S T A T I S T I C S
      T O T A L   S E G M E N T S   B Y   S E G M E N T   T Y P E
SEGMENT      SEGMENT
NAME         LEVEL
K1           1           RELOADED           DIFFERENCE
K2           2           3
K3           3           2
K4           4           3
K5           2           2
K6           3           5
K8           3           3
K9           2           6
K10          2           0
K11          3           0
K12          3           0
K13          4           0
K14          4           0
K15          3           0
K15          3           0
      T O T A L   S E G M E N T S   I N   D A T A B A S E
      U N L O A D E D           R E L O A D E D           D I F F E R E N C E
      24                       24

```

SYSIN 制御ステートメントで STAT=DET を指定して HALDB データベースをロードして、詳細報告書を要求すると、その報告書には、区画ごとにロードされたセグメントのリストと、HALDB データベース全体のロード済みの全セグメントのリストが含まれます。

HD 再編成再ロード・ユーティリティーに対する連結入力データ・セットの場合、各入力ファイルからの出力統計が示されます。HD 再編成再ロード・ユーティリティーからの詳細報告書には、再ロードされたデータではなく、アンロードされたデータが反映されます。HD 再編成再ロード・ユーティリティーを実行した結果、キー範囲の変更、統合、または拡張が考えられるため、それらの報告書では区画名が明示されません。

以下の図は、他の区画のアンロード・データが入っている入力アンロード・ファイルに対して調整された、単一区画の統計の例です。

ILDSINGLE=PDHDOJA

```

      S E G M E N T   L E V E L   S T A T I S T I C S
      T O T A L   S E G M E N T S   B Y   S E G M E N T   T Y P E
SEGMENT      SEGMENT
NAME         LEVEL
      U N L O A D E D           R E L O A D E D           D I F F E R E N C E

```

J1	1	72	32 -
J2	2	9	64 -
J3	3	3	32 -
J4	4	3	32 -
J5	2	16	32 -
J6	3	12	32 -
J7	4	6	32 -
J8	5	3	32 -
J9	2	16	64 -
J10	3	9	
J11	4	0	
J7P	3	6	32 -
J12	2	79	64 -
J13	3	66	2 -
J14	4	40	
J13X	3	13	32 -
J15	3	41	

TOTAL UNLOADED	SEGMENTS IN DATA RELOADED	BASE DIFFERENCE
876	394	482

## DFSURGL0 の例

ここでの例は、DFSURGL0 ユーティリティのサンプル JCL を示しています。

この節の例で、使用する DBRC で動的割り振りを行わない場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

```
//RECON1 DD DSNAME=RECON1,DISP=SHR
//RECON2 DD DSNAME=RECON2,DISP=SHR
//RECON3 DD DSNAME=RECON3,DISP=SHR
```

サブセクション:

- 345 ページの『HDAM 再編成再ロードのための JCL』
- 345 ページの『HIDAM 再編成再ロード JCL』
- 345 ページの『HIDAM VSAM データベースのアンロードおよび再ロード』
- 346 ページの『NOILDS を使用した HIDAM HALDB マイグレーション再ロード』
- 347 ページの『ILDSMULTI を使用した HIDAM HALDB マイグレーション再ロード』

## HDAM 再編成再ロードのための JCL

この例は、HDAM 再編成再ロードのための JCL を示しています。

```
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSURGL0,DH32DB01',
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=OLD
//SYSPRINT DD SYSOUT=A
//DFSUINPT DD DSN=IMS.UNLOAD1,DISP=OLD,
// UNIT=TAPE,VOL=SER=TAPE11,LABEL=(,SL)
//DFSURWF1 DD DSN=IMS.WRKTAPE1,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=WKTAPE,LABEL=(,SL),
// DCB=(BLKSIZE=1008,LRECL=900,RECFM=VB)
//HDSKILLS DD DSN=DATABASE.SKILLS,DISP=(NEW,KEEP),
// UNIT=SYSDA,VOL=SER=DB0002,SPACE=(CYL,(10,10))
//DFSURCDS DD DSN=IMS.RLCDS,DISP=(OLD,KEEP),
// UNIT=SYSDA,VOL=SER=IMSMSC
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//DFSCCTL DD *
SBPARM ACTIV=COND
```

## HIDAM 再編成再ロード JCL

この例は、HIDAM 再編成再ロードのための JCL を示しています。1 次索引データベース・データ・セットも記述されています。RECON データ・セットは使用されません。

```
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSURGL0,HD32DB02',
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=OLD
//SYSPRINT DD SYSOUT=A
//DFSUINPT DD DSN=IMS.UNLOAD1,DISP=OLD,
// UNIT=TAPE,VOL=SER=TAPE11,LABEL=(,SL)
//DFSURWF1 DD DSN=IMS.WRKTAPE1,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=WKTAPE,LABEL=(,SL)
// DCB=(BLKSIZE=1008,LRECL=900,RECFM=VB)
//HDPAYROL DD DSN=DATABASE.PAYROLL,DISP=OLD,
// UNIT=SYSDA,VOL=SER=DB0001
//HDINDEXO DD DSN=DATABASE.INDEXO,DISP=(NEW,KEEP),
// UNIT=SYSDA,VOL=SER=DB0003,SPACE=(CYL,(10,10))
//DFSURCDS DD DSN=IMS.RLCDS,DISP=(OLD,KEEP),
// UNIT=SYSDA,VOL=SER=IMSMSC
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//DFSCCTL DD *
SBPARM ACTIV=COND
```

## HIDAM VSAM データベースのアンロードおよび再ロード

この例は、HIDAM VSAM データベースのアンロードおよび再ロードのための JCL を示しています。

```
//UNLOAD EXEC PGM=DFSRR00,PARM='ULU,DFSURGU0,DHVBTZ01'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
// DD DSN=IMS.PSBLIB,DISP=SHR
//DFSURGU1 DD DSN=UNLOAD,DISP=(NEW,PASS),
// UNIT=SYSDA,SPACE=(TRK,(10,5))
//PRINTDD DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DXSK0302 DD DSN=VVDX0302,DISP=OLD
//DXSK0301 DD DSN=VVDX0301,DISP=OLD
```

```

//DHSK0301 DD DSN=VVDH0301,DISP=OLD
//DFSVSAMP DD *
2048,10
//DFSCCTL DD *
SBPARM ACTIV=COND
/*
//STP98 EXEC PGM=IDCAMS
//SYSPPRINT DD SYSOUT=A
//SYSABEND DD SYSOUT=A
//VSA DD UNIT=SYSDA,DISP=OLD,VOL=SER=VSIMS
//SYSIN DD *
DELETE VVDH0301
DELETE VVDX0301
DELETE VVDX0302
DEF CL (NAME(VVDX0301) CYL(1 1) RECSZ(16 16) VOL(VSIMS) IXD-
CISZ(2048) FSPC(25) KEYS(10 5))
DEF CL (NAME(VVDH0301) TRK(10 5) RECSZ(2041 2041) VOL(VSIMS) NIXD-
CISZ(2048) )
DEF CL (NAME(VVDX0302) TRK(10 5) RECSZ(2041 2041) VOL(VSIMS) NIXD-
CISZ(2048) )
/*
//RELOAD EXEC PGM=DFSRR00,PARM='ULU,DFSURGL0,DHVBTZ01'
//STEP1 DD DSN=IMSCAT,DISP=SHR
//STEP2 DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
// DD DSN=IMS.PSBLIB,DISP=SHR
//DFSUINPT DD DSN=UNLOAD,DISP=(OLD,PASS),UNIT=SYSDA
//PRINT DD SYSOUT=A
//SYSPPRINT DD SYSOUT=A
//SYSUDUMP DD SYSOUT=A
//DHSK0301 DD DSN=VVDH0301,DISP=OLD
//DXSK0302 DD DSN=VVDX0302,DISP=OLD
//DXSK0301 DD DSN=VVDX0301,DISP=OLD
//DFSVSAMP DD *
2048,10
//DFSCCTL DD *
SBPARM ACTIV=COND
/*

```

## NOILDS を使用した HIDAM HALDB マイグレーション再ロード

この例は、HIDAM HALDB マイグレーション再ロードに NOILDS オプションを使用した JCL を示しています。新しい HALDB には 2 つの区画があります。

```

//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSURGL0,PHIDMSTR',
//STEP2 DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=OLD
//SYSPPRINT DD SYSOUT=A
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//DFSUINPT DD DSN=IMS.UNLOAD1,DISP=OLD,
// UNIT=TAPE,VOL=SER=TAPE11,LABEL=(,SL)
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//DFSCCTL DD *
SBPARM ACTIV=COND
//SYSIN DD *
NOILDS
//STEP02 EXEC PGM=DFSRR00,REGION=1300K,
// PARM='ULU,DFSPPREC0,PHIDMSTR,,,,,,,,,Y,N'
//STEP3 DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=OLD
//SYSPPRINT DD SYSOUT=A

```

```

//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//SYSIN DD *
PARTITION=PDHIDMA,RECOVTYP=ILE
//STEP03 EXEC PGM=DFSRR00,REGION=1300K,
// PARM='ULU,DFSREC0,PHIDMSTR,,,,,,,,,Y,N'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=OLD
//SYSPRINT DD SYSOUT=A
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//SYSIN DD *
PARTITION=PDHIDMB,RECOVTYP=ILE

```

## ILDSMULTI を使用した HIDAM HALDB マイグレーション再ロード

この例は、HIDAM HALDB マイグレーション再ロードに ILDSMULTI オプションを使用した JCL を示しています。

```

//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSURGL0,PHIDMSTR',
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=OLD
//SYSPRINT DD SYSOUT=A
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
//DFSUINPT DD DSN=IMS.UNLOAD1,DISP=OLD,
// UNIT=TAPE,VOL=SER=TAPE11,LABEL=(,SL)
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//DFSCTL DD *
SBPARM ACTIV=COND
//SYSIN DD *
ILDSMULTI

```





---

## 第 25 章 HD 再編成アンロード・ユーティリティ (DFSURGU0)

HD 再編成アンロード・ユーティリティ (DFSURGU0) は、IMS 全機能データベースの再編成、IMS 全機能データベースの構造の変更、および、IMS 全機能データベースにより使用される論理関係に関する情報の記録を行うために使用します。

DFSURGU0 ユーティリティは通常、HD 再編成再ロード・ユーティリティ (DFSURGL0) とペアで使用されます。HD 再編成再ロード・ユーティリティは、DFSURGU0 ユーティリティによって作成されたアンロード・データ・セットを読み取ることで IMS 全機能データベースを再ロードします。

DFSURGU0 ユーティリティは、次の目的で使用できます。

- HDAM、PHDAM、HIDAM、PHIDAM、PSINDEX、または HISAM データベースを順次データ・セットにアンロードします。
- 適切なユーティリティ制御ステートメントを指定することにより、PHDAM、PHIDAM、または PSINDEX の区画のサブセットをアンロードします。このユーティリティを HALDB データベースに対して実行するときに、ユーティリティ制御ステートメントを使用しなかった場合、DFSURGU0 ユーティリティはデータベース全体をアンロードします。
- 接頭部情報が入ったデータ・セットを生成します (論理関係が存在する場合)。
- HDAM、PHDAM、HIDAM、PHIDAM、または HISAM データベースに構造上の変更を行います。
- 論理親、論理兄弟、および論理子ポインター・フィールドと、データベースに置かれた論理親に関連するカウンター・フィールドを記録します。
- データベースを HALDB パーティション・データベースに変換し、HALDB からフォールバックします。

再編成されるデータベースの元の DBD を、構造上の変更を反映する新しい DBD で置き換える場合、DFSURGU0 ユーティリティは、再編成の過程でデータベースに対し一定の構造上の変更を加えることもできます。

DFSURGU0 ユーティリティで行うことのできる構造上の変更は、以下のとおりです。

- 同じタイプのすべてのセグメントが HD 再編成アンロード・ユーティリティの実行前にデータベースから削除されていれば、その既存のセグメント・タイプを DBD から削除します。
- 既存のセグメント・タイプ間の階層関係、または論理的に関係を持つセグメントの連結キーを変えない場合、新しいセグメント・タイプを新しい DBD に追加します。
- セグメントのシーケンス・フィールドのステートメントを除き、任意のフィールド・ステートメントを変更、追加、または削除します。ただし、IMS はセグメントのデータ内容を変更することはありません。

- 固定長セグメントの既存のセグメント長を変更します。IMS はデータ内容を変更しません。ただし、そのセグメントが小さくなったときは、データを切り捨てるために変更を行います。

セグメント長を大きくすると、そのセグメントの増加部分を充てんする文字として 2 進ゼロが使用されます。セグメントの拡張部分はユーザーの責任で置き換えてください。置き換えるには、IMS の下でアプリケーション・プログラムを更新モードで実行します。

- DL/I アクセス方式を変更します。OSAM 形式を VSAM 形式に変更することも、VSAM 形式を OSAM 形式に変更することもできます。どの DL/I アクセス方式も変更することができますが、HDAM または PHDAM は例外で、これらはどの索引方式にも変更できません。HISAM、HIDAM、および PHIDAM は HDAM または PHDAM に変更することができます。
- HDAM、PHDAM、HIDAM、および PHIDAM データベースに対するセグメント・ポインター・オプションを変更します。ただし、データベースに論理関係が入っている場合やカウンター、LT、または LP ポインターが変更された場合は、データベース事前再編成ユーティリティを再実行する必要があります。物理対から仮想対に変更する場合は、仮想となるセグメントのすべてのオカレンスを削除しなければなりません。仮想対は HALDB に対してサポートされません。

注: この制約事項は、HALDB データベースには適用されません。

アンロード操作を行う場合、このユーティリティはアプリケーション・プログラムとして機能し、一連の非修飾 GN 呼び出しを DL/I に出します。アンロード・ユーティリティを実行しても、それに伴って、完全なポインター健全性検査が行われることはありません。

データベースをアンロードするとき、HD 再編成アンロード・ユーティリティは、再編成をサポートするために、アンロードされた各セグメントに対して接頭部を追加します。非 HALDB データベースからアンロードされたセグメントに対して作成された接頭部は、HALDB データベースからアンロードされたセグメントに対して作成された接頭部とは異なります。

非 HALDB データベースからアンロードされたセグメントに対して作成された接頭部は、マクロ DFSURGUF によってマップされます。

HALDB データベースからアンロードされたセグメントに対して作成された接頭部は、マクロ DFSURGUP によってマップされます。アンロードされたセグメントが論理子である場合、DFSURGUP は間接リスト・キー (ILK) と拡張ポインター・セット (EPS) を含めます。

ILK には通常、以下の図に示すように、セグメントが最初に作成された際の相対バイト・アドレス (RBA)、区画 ID、および区画再編成番号が含まれています。ただし、HD データベースを HALDB にマイグレーションする場合、DFSURGU0 ユーティリティが作成するマイグレーション ILK には、区画 ID としてゼロが含まれ、通常は再編成番号を含むフィールドには、ターゲット・セグメントの読み取り元となる DASD のデータ制御ブロック (DCB) の数が含まれます。

ILK 接頭部			
	初期 RBA	区画 ID	再編成番号
バイト	4	2	2

図 68. ILK のフォーマット

このユーティリティの機能は、ユーティリティ制御機能に実行させることができます。

以下の図は、HD 再編成アンロード・ユーティリティのフロー・ダイアグラムです。

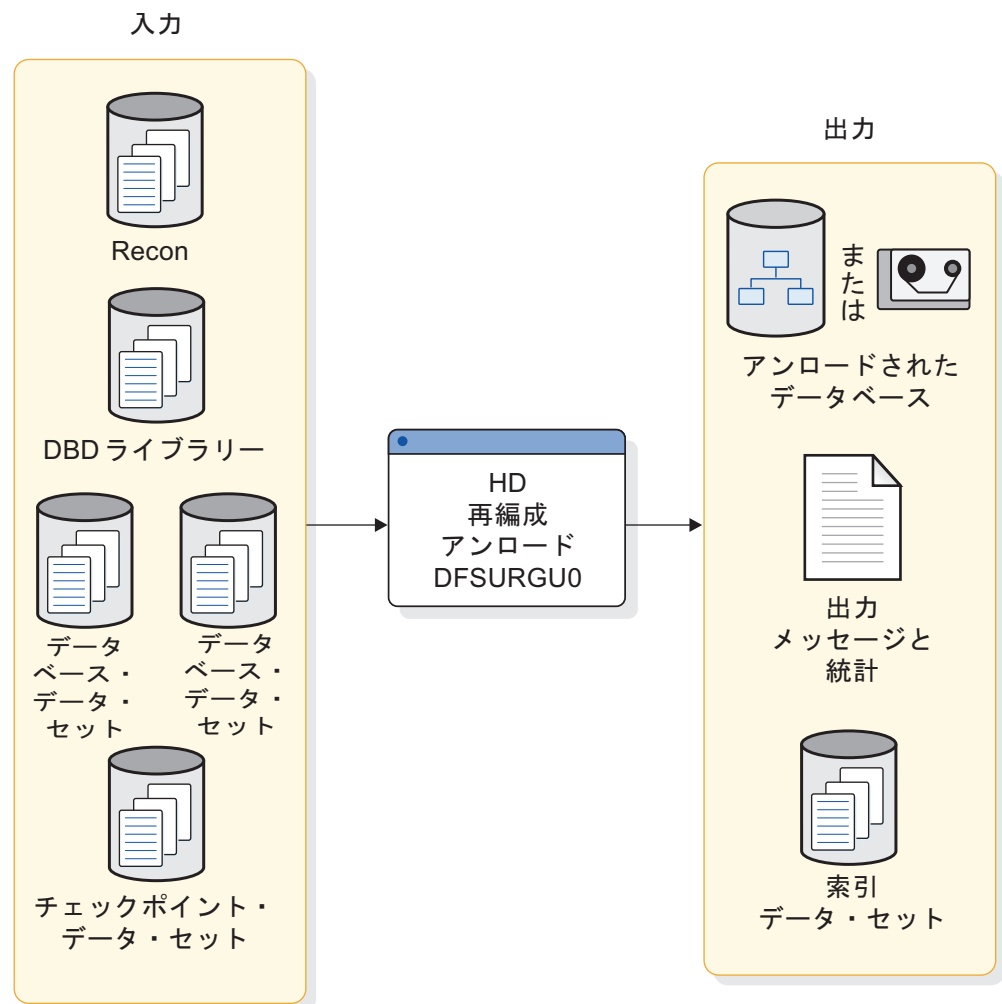


図 69. HD 再編成アンロード・ユーティリティ

サブセクション:

- 352 ページの『制約事項』
- 353 ページの『前提条件』

- 353 ページの『要件』
- 355 ページの『推奨事項』
- 355 ページの『入出力』
- 357 ページの『JCL 指定』
- 360 ページの『戻りコード』

## 制約事項

DFSURGU0 ユーティリティーには、次のような制約があります。

- HALDB に対してはユーティリティー制御機能がサポートされません。
- DFSURGU0 ユーティリティーの実行後は、データベースが再ロードされるまでデータベースの更新を行わないでください。アンロード・ファイルがデータベースの再ロードに使用される場合、アンロード・ファイルの作成後に行われた更新は失われます。
- 既存のセグメント・タイプの名前は変更しないでください。
- MIGRATE 制御ステートメントを使用するアンロードは、非 HALDB でのみ実行することができます。
- 索引アクセス方式を使用するよう HDAM データベースまたは PHDAM データベースの DL/I アクセス方式を変更することはできません。
- 次の場合は、HD 再編成アンロード・ユーティリティーを使用しないでください。
  - いずれかの論理子セグメントが物理パスまたは論理パスのいずれか (両方のパスからではなく) から削除されていたとき、両方向仮想対から両方向物理対に変更する場合。
  - 実論理子を、論理的に関係を持つ一方のデータベースから別のデータベースへ変更する場合。
  - 基本または副次索引を再編成するときは、索引データベースには HISAM 再編成ユーティリティーを使用してください (ただし、PSINDEX の場合は例外です)。
  - データベースで書き込みエラーが起こり、そのデータベースがリカバリーされていない場合。リカバリーを行ってからでなければ、このユーティリティーを実行できません。データベースが DBRC に登録されている場合に、このユーティリティーが DBRC を使用すると、DBRC は、書き込みエラーが起こってリカバリーがまだ行われていないことを認識し、このユーティリティーの許可要求を拒否します。
- DFSURGU0 ユーティリティーは、全機能 (HALDB および非 HALDB) データベースのみで機能します。
- データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。

KEYRANGE オプションを使用する場合、DFSURGL0 ユーティリティーには、次のような制約があります。

- VSAM データ・セット共用オプションでは、複数のリーダーを許可する必要があります。

- 予測不能な結果を回避するには、キー範囲は区画の境界上で開始および終了する必要があります。1 つ以上の区画に相当するデータをアンロードできますが、1 つのデータ・セットにアンロードする必要があります。これは、HD 再ロードでは入力データとして単一のデータ・セットのみがサポートされるためです。
- KEYRANGE オプションは MIGRATE= ステートメントに続いた場合のみ有効です。MIGRATX= ステートメントに続けると無効になります。
- KEYRANGE オプションには、データベース・データ・セットに対して IRLM=N、DBRC=N、および DISP=SHR を用いた DLI データベースのアンロードを行うことが必要です。
- マイグレーションの再ロードでは DBRC=Y を使用する必要があります、新しい HALDB 区画定義を持つ RECON データ・セットの割り振りが必要となります。その割り振りは、JCL を用いて、または DBRC RECON MDA メンバーを動的に使用して行われます。

## 前提条件

DFSURGU0 ユーティリティーの前提条件は次のとおりです。

- DBD から既存のセグメント・タイプを削除する場合は、DFSURGU0 ユーティリティーの実行前に、削除するタイプのセグメントすべてをデータベースから削除しておかなければなりません。

## 要件

DFSURGU0 の要件は次のとおりです。

- DFSURGU0 ユーティリティーは、データベースの現在の構造を記述している DBD に対して実行する必要があります。
- HALDB データベースのデータ・セット・フォーマットを OSAM から VSAM へ、または VSAM から OSAM へ変更するには、HALDB 定義をいったん削除し、RECON データ・セットで再定義しなければなりません。
- MIGRATX HD をアンロードするために HDAM または PHDAM データベースをアンロードするときは、ランダム化モジュールとすべての IMS ユーザー出口ルーチン (圧縮、疎索引など) を JOBLIB に含めなければなりません。通常の HD アンロードでは、疎索引は不要です。
- 固定長のセグメント長を増加させる場合、IMS のもとで更新モードで稼働するアプリケーション・プログラムを使用して、セグメントの拡張部分で充てん文字として使用される 2 進ゼロを置き換える必要があります。
- HDAM、PHDAM、HIDAM、および PHIDAM データベースに対するセグメント・ポインター・オプションを変更することができます。ただし、データベースに論理関係が入っている場合やカウンター、LT、または LP ポインターが変更された場合は、データベース事前再編成ユーティリティーを再実行する必要があります。物理対から仮想対に変更する場合は、仮想となるセグメントのすべてのオカレンスを削除しなければなりません。仮想対は HALDB に対してサポートされません。

注: 仮想対は HALDB ではサポートされないため、仮想となるすべてのセグメントの削除要件は HALDB には当てはまりません。

- 非固有キーを持つ副次索引データベースを HALDB に変換する場合は、HD 再編成再ロード・ユーティリティに入力する前に、別々の JCL ステップでアンロード・レコードのソートとマージを行って新規の /SX 値を作成する必要があります。非固有キーに関して、/SX 値がアンロード・レコードに生成されたときは、再ロード・ステップを実行する前に、/SX を入れることができるように HALDB DBD を変更しなければなりません。
- シンボリック・ポインティングを使用する副次索引データベースを HALDB に変換する場合は、別々の JCL ステップでアンロード・レコードのソートとマージを行う必要があります。ソートおよびマージ用の JCL の例は、366 ページの『シンボリック・ポインティングを使用する副次索引用のアンロード・レコードのソートおよびマージ』を参照してください。
- 副次索引を持つデータベースを HALDB パーティション・データベースに変換する場合は、DFSURGU0 ユーティリティを実行してそれぞれの副次索引データベースをアンロードする必要があります。新しい PSINDEX 向けの変換アンロード・レコードについて、間接リスト項目キー (ILK) および拡張ポインター・セット (EPS) の情報を構成するのに必要なソース・セグメントとターゲット・セグメントを解決するために、DFSURGU0 ユーティリティは 1 次データベースを読み取ります。副次索引セグメントで順次処理を行うと、1 次データベースで多数のランダム読み取りが行われることがあります。

複数の 2 次索引アンロードについて、1 次データベースでこのようなランダム・アクセスが行われると、DL/I 2 次索引データベースから HALDB PSINDEX へのマイグレーション・アンロードの総合パフォーマンスが低下します。

例えば、1 つの 1 次データベースに 7 つの 2 次索引データベースがあり、マイグレーション・アンロード・ジョブごとに 13 時間がかかるとすると、2 次索引データベースをすべてアンロードするのに要する合計時間は、7 x 13 で 91 時間です。この所要時間の多くは、基本データベースの読み取りを繰り返すことに費やされます。

- HD アンロード・ユーティリティを使用して非 HALDB から HALDB へのマイグレーションをする際に、KEYRANGE オプションを使用して複数のアンロード・ジョブを並列に実行することで、MIGRATE= のパフォーマンスを向上させることができます。これは、超大規模であるか、論理的に関連するセグメントを大量に含む非 HALDB をマイグレーションする場合に非常に役立ちます。
- KEYRANGE オプションの場合、
  - fromkeyval および tokeyval の両方を DBRC LIST.DB バッチ・コマンドの出力からコピーし、HDUNLOAD SYSIN DD ステートメントに貼り付けます。
  - (3,3) の VSAM データ・セット共用オプションが推奨されます。
  - パフォーマンスを最大化するには、1 つのステップで単一のパーティションのキー範囲をアンロードした後、対応する HALDB パーティションのマイグレーション再ロードを行います。
- このユーティリティを RECON データ・セットに登録されていない IMS カタログ・データベースで使用するには、追加の EXEC パラメーター DFSDF を指定する必要があります。DFSDF= は、未登録の IMS カタログ・データベースの名前を含む IMS.PROCLIB データ・セットの DFSDFxxx メンバーの、3 文字

からなる接尾部を指定します。それらの名前は、DATABASE ステートメントの UNREGCATLG パラメーターを用いて指定されます。例えば、次のようにします。

```
//HDUNLOAD EXEC PGM=DFSRR00,
// PARM=(ULU,DFSURGU0,DFSCD000,,,,,,,,,,,,,N,N,,,,,,,,,,,,,
// 'DFSDF=CAT')
```

3 文字からなる接尾部 CAT を、未登録の IMS カタログ・データベースの名前を指定する DFSDFxxx メンバーの接尾部で置き換えてください。

### 推奨事項

現在、DFSURGU0 ユーティリティに対して文書化された推奨事項はありません。

### 入出力

DFSURGU0 ユーティリティへの 1 次入力データベース・データ・セットです (ただし、ほかの入力データも必要です)。DFSURGU0 ユーティリティの 1 次出力はアンロード・データ・セットです。他の出力にはチェックポイント・データ・セットが含まれる場合があります、メッセージおよびアンロード統計を出力します。

HD 再編成アンロード・ユーティリティの入出力は、以下の表に示すとおりです。

表 15. HD 再編成アンロード・ユーティリティが使用するデータ・セット

入力	出力
RECON	チェックポイント・データ・セット
DBD ライブラリー	出力メッセージと統計
データベース・データ・セット	アンロードされたデータ・セット
チェックポイント・データ・セット	

HD 再編成アンロード・ユーティリティは、出力メッセージと統計を出力します。このユーティリティから得られるメッセージと統計の例を、以下の図に示します。

図 70. 出力メッセージと統計の例 — HD 再編成アンロード・ユーティリティ

```
H I E R A R C H I C A L D I R E C T D A T A B A S E R E O R G A N I Z A T I O N U N L O A D P A G E 01
DFS343W DDNAME DFSUCKPT WAS SPECIFIED AS DD DUMMY OR WAS OMITTED FOR FUNCTION DU
DFS342I RESTART NOT REQUESTED, NORMAL PROCESSING BEGINS
DFS344W DDNAME FOR SECOND COPY WAS NOT SUPPLIED, 1 COPY REQUESTED FOR FUNCTION DU
COPY 1 ON VOLUME(S) - USER02
DFS340I DATABASE DHONTZ04 HAS BEEN SUCCESSFULLY UNLOADED BY FUNCTION DU
          D A T A B A S E S T A T I S T I C S
SEGMENT LEVEL STATISTICS
MAXIMUM AVG MAXIMUM AVG SEGMENT SEGMENT RECORD LEVEL STATISTICS
TWINS TWINS CHILDREN CHILDREN NAME LEVEL BY SEG TYPE DB RECORD
1 1.00 8 7.00 K1 1 3 1.00
1 0.66 3 2.50 K2 2 2 0.66
2 1.50 1 0.66 K3 3 3 1.00
1 0.66 0 0.00 K4 4 2 0.66
2 1.66 1 0.60 K5 2 5 1.66
1 0.60 0 0.00 K6 3 3 1.00
```

4	2.00	0	0.00	K8	2	6	2.00
0	0.00	0	0.00	K9	2	0	0.00
0	0.00	0	0.00	K10	3	0	0.00
0	0.00	0	0.00	K11	3	0	0.00
0	0.00	0	0.00	K12	4	0	0.00
0	0.00	0	0.00	K13	4	0	0.00
0	0.00	0	0.00	K14	3	0	0.00
0	0.00	0	0.00	K15	3	0	0.00

H I E R A R C H I C A L D I R E C T D B R E O R G U N L O A D PAGE 02  
 TOTAL SEGMENTS IN DATABASE=24 AVERAGE DATABASE RECORD LENGTH=300 BYTES  
 DFS339I FUNCTION DU HAS COMPLETED NORMALLY RC=0

注: STAT=DET の場合、上記のデータベース統計のほかに、各 HALDB 区画の出力統計報告書があります。各区画統計報告書には、区画名が示され、データベース統計と同じフィールドがリストされます。

ページ見出しに続いて、この実行で選択されたオプションの結果として出された各種のメッセージが示されます。

1 次出力データ・セットが正常に完了すると、「COPY 1 ON VOLUME(S) - *volser1*」というメッセージが示されます。ボリューム通し番号のリストは、使用されたボリュームとその使用順序を示しています。2 番目のコピーが要求され、正常に完了していると、「COPY 2 ON VOLUME(S) - *volser2*」というメッセージが示されます。「DATABASE STATISTICS」という見出しがメッセージの次に示されます。

「SEGMENT LEVEL STATISTICS」という見出しの下のフィールドは、次のとおりです。

#### MAXIMUM TWINS

直接の親セグメントのもとに見付かった、このタイプのセグメントの最大数。ルート・レベルでは、この値は常に 1 です。

#### AVERAGE TWINS

直接の親セグメントのもとに見付かった、このタイプのセグメントの平均数。この値は、小数点以下 2 桁まで示されます。

#### MAXIMUM CHILDREN

指定の親のもとにある子セグメント (すべての従属レベルにあるもの) の最大数

#### AVERAGE CHILDREN

指定の親のもとにある子セグメント (すべての従属レベルにあるもの) の平均数。この値は、小数点以下 2 桁まで示されます。階層パスの最下位にあるセグメントの場合は、このフィールド・タイプは値がゼロになっています。

#### SEGMENT NAME

この統計行が適用されるセグメントの名前

#### SEGMENT LEVEL

このセグメントがデータベースに置かれている階層レベル。セグメントの記述は、DBD に記述されているのと同じ順序で、上から下に向かってマップされます。

「RECORD LEVEL STATISTICS」という見出しの下のフィールドは、次のとおりです。



## TOTAL SEGMENTS BY SEGMENT TYPE

このセグメント・タイプがデータベース全体に出現した回数。レベル 1 のセグメント・タイプのカウント・フィールドは、データベースに入っているデータベース・レコード (ルート・セグメント) の総数を示しています。

## AVERAGE COUNT PER DATABASE RECORD

このセグメント・タイプが、指定のデータベース・レコード内に出現した回数の平均数。この値は、小数点以下 2 桁まで示されます。

個々のセグメント・タイプの統計に続いて、データベースに置かれているすべてのセグメントの総数と平均データベース・レコード長 (バイト数) が示されます。平均データベース・レコード長には、データ長と接頭部サイズの両方が含まれます。レベル 1 のセグメント数に平均データベース・レコード長をかけて求めた積から、データベース内の総バイト数が分かります。すべての物理的に保管されるレコードは、使用可能なデータ位置をすべて使用するとは限らないため、この数字は、必要とするデータ・セット・スペースの近似値としてのみ使用することができます。

## JCL 指定

HD 再編成アンロード・ユーティリティーは、標準の z/OS ジョブとして実行されます。次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

HD 再編成アンロード・ユーティリティーからの出力は、オペレーティング・システムの可変ブロック化順次データ・セットです。出力は、出力装置が処理できる最大サイズにブロック化されるので、出力ボリュームには標準ラベルを使用しなければなりません。

### EXEC ステートメント

EXEC ステートメントは次の形式になっていなければなりません。

```
PGM=DFSRR00,PARM='ULU,DFSURGU0,dbname'
```

このステートメントは、HALDB および非 HALDB の両方のデータベースに使用されます。

パラメーター ULU と DFSURGU0 は、ユーティリティー領域を記述します。

*dbname* は、再編成するデータベースを記述する DBD の名前です。

SPIE、BUF、DBRC といった通常の IMS 定位置パラメーターは、*dbname* の後に置くことができます。

### DD ステートメント

DFSURGU0 ユーティリティーで使用される入力データ・セット、出力データ・セット、およびリソース・データ・セットは、DD ステートメントで識別されます。

DFSURGU0 ユーティリティーは、以下の必須 DD ステートメントとオプションの DD ステートメントを使用します。

### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

### IMS DD

再編成するデータベースを記述する DBD が入っているライブラリーを定義します (つまり、DSN=IMS.DBDLIB, DISP=SHR)。このデータ・セットは、直接アクセス装置に置いておかなければなりません。

注: HALDB 変更操作が開始されたが、オンライン変更機能が完了していない HALDB データベースをアンロードする場合でも、元の変更されていない DBD を含む DBD ライブラリーを指定する必要があります。データベース内の一部またはすべてのセグメントが、変更された DBD が定義するフォーマットに従っているかどうかは考慮しません。IMS はステージング ACB ライブラリーから変更された DBD の情報を自動的に取得し、元の変更されていない DBD との比較を要求します。

### SYSPRINT DD

メッセージおよび統計出力データ・セットを定義します。このデータ・セットは、テープ、直接アクセス装置、またはプリンターに置いておくことも、出力ストリームに入れて送ることもできます。このデータ・セットに関して指定されている DCB パラメーターは、RECFM=FBA と LRECL=121 です。BLKSIZE は SYSPRINT DD ステートメントで指定し、121 の倍数になっていなければなりません。

### SYSIN DD

入力制御ステートメント・データ・セットを、HALDB およびフォールバックに対してのみ定義する、オプション・データ・セットです。SYSIN データ・セットを省略した場合は、HALDB 全体がアンロードされます。SYSIN データ・セットを指定しても、内容が空の場合は、HALDB 全体がアンロードされます。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。

### DFSUCKPT DD

チェックポイントをとるために使用するデータ・セットを定義します。チェックポイントが必要でなければ、このステートメントは使用しないでください。このデータ・セットは直接アクセス装置に置いておくのが普通です。ただし、テープ・ボリュームを使用することも可能です。

### DFSURSRT DD

再始動を試みる場合にチェックポイント・データ・セットを定義します。再始動を試みない場合、このステートメントは省略してください。再始動を試みる場合は、最後のチェックポイントがとられたとき DFSUCKPT DD ステートメントが参照したのと同じデータ・セットを、このステートメントが参照していることを確かめてください。このデータ・セットは直接アクセス装置に置いておくのが普通です。ただし、テープ・ボリュームを使用することも可能です。

DFSURSRT DD ステートメントは、特殊なチェックポイント・レコードをチェックポイント・データ・セットと出力データ・セットに書き込みます。再始動が

必要であれば、チェックポイント・レコードがチェックポイント・データ・セットから取り出され、出力ボリュームが位置付けられ、データベース内に正しい位置が設定されます。統計テーブル・レコードが読み取られ、主記憶テーブルが正しく初期設定されます。その後、プログラムは通常の処理を続けます。

#### **DFSURGU1 DD**

1 次出力データ・セットを定義します。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。この DD ステートメントは必須です。

#### **DFSURGU2 DD**

2 次出力データ・セットを定義します。この DD ステートメントを使用するのは、出力のコピーを 2 つ要求するときです。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。

データベースのコピーは複数とることができます。コピーを 2 つ指定すると、実行中に入出力エラーが起こっても、ユーティリティは他方のコピーを使用して最後まで実行するという利点が得られます。この場合、パフォーマンスがいくぶん低下しますが、全体を再実行する必要はありません。

#### **database DD**

再編成するデータベース・データ・セットを定義します。このステートメントは、再編成するデータベースを記述する DBD に指定されているデータ・セットごとに 1 つ必要です。DD 名は、DBD にある DD 名と一致している必要があります。

HALDB データベースをアンロードする場合は、database DD ステートメントを指定しないでください。HALDB は動的割り振りを使用します。統合 HALDB オンライン再編成機能を使用する場合、動的割り振りはデータ・セット名に対して A から J または M から V の適切な接頭部を自動的に選択します。

これが HIDAM データベースのときは、索引を表しているデータ・セットにも、DD ステートメントが必要です。索引と関係付けるために使用される DD ステートメントは、索引データベースの DBD に指定されている DD 名を含んでいなければなりません。このデータベースに関連する 2 次索引の場合も、DD ステートメントは必要ありません。

このデータ・セットは、直接アクセス装置に置いておかなければなりません。

#### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ・プール情報が入っているデータ・セットを記述します。

このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。このステートメントは必須です。

#### **DFSCTL DD**

順次バッファリングの活動化を要求する SBPARM 制御ステートメントが入っているデータ・セットを記述します。順次バッファリングを条件付きで活動化させると、OSAM DB データ・セットのバッファリングのパフォーマンスが向上し、ジョブ経過時間が短縮されます。

DFSCTL ファイルは順次データ・セットにすることも、PDS のメンバーにすることもできます。レコード形式は F、FB、または FBS のいずれかで、レコー

ド長は 80 でなければなりません。このデータ・セットは、直接アクセス装置またはテープに置いておくことも、入力ストリームに入れて送ることもできます。

#### **DFSWRKnn**

ここで、nn は副次索引の数 (01 以上) です。DFSWRKnn データ・セットには、副次索引マイグレーション・アンロード・レコードが書き込まれます。

#### **DFSSRTnn**

ここで、nn は副次索引の数 (01 以上) です。DFSSRTnn データ・セットは、対応する DFSWRKnn 用に使用されるソート制御ステートメントです。

注: DFSSRTnn および DFSWRKnn データ・セットは、DBD 内での 2 次索引定義の順に割り振られます。このユーティリティーは、2 次索引名を DFSSRTnn または DFSWRKnn DD ステートメントまたはデータ・セットの接尾部に関係付けようとしません。2 次索引は、次の DFSSRTnn および DFSWRKnn データ・セットの対を昇順に使用します。

#### **SYSABEND DD または SYSUDUMP DD**

ダンプ・データ・セットを定義します。どちらかのステートメントが指定されているとき、4 より大きい戻りコードが返されると、ABEND U0347 が起こります。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

#### **RECON1 DD**

最初の DBRC RECON データ・セットを定義します。

#### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。

#### **RECON3 DD**

RECON1 または RECON2 にエラーが見つかったとき使用される、オプションの DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

### **戻りコード**

DFSURGU0 ユーティリティーは、完了時に以下の戻りコードを 1 つ以上返します。

コード 意味

- 0 データベースのアンロードが正常に完了した
- 4 1 つ以上の警告メッセージが出された
- 8 重大エラーが起こったか、コピー 1 で入出力エラーが起こった
- 12 戻りコード 4 と 8 の組み合わせが起こった
- 16 データベースのアンロードが正常に完了しなかった

関連資料:

437 ページの『第 31 章 ユーティリティー制御機能 (DFSUCF00)』

391 ページの『第 28 章 HISAM 再編成アンロード・ユーティリティ (DFSURUL0)』

📄 DBBATCH プロシージャ (システム定義)

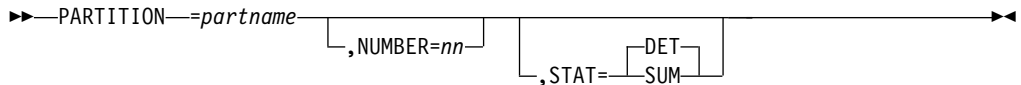
📄 DLIBATCH プロシージャ (システム定義)

## DFSURGU0 ユーティリティの制御ステートメント

DFSURGU0 ユーティリティでは、PARTITION、MIGRATE、KEYRANGE、MIGRATX、および FALLBACK のユーティリティ制御ステートメントが使用できます。

HD 再編成アンロード・ユーティリティ (DFSURGU0) で使用されるすべてのユーティリティ制御ステートメントは、HALDB データベースを処理するためのオプションまたはデータベースを HALDB に変換するためのオプションを指定します。

PARTITION ユーティリティ制御ステートメントは、以下に準拠している必要があります。



### PARTITION=

アンロードする HALDB 区画 (または複数の順次 HALDB 区画の最初の区画) の名前を指定します。

### NUMBER=

アンロードする順次 HALDB 区画の数を指定します。デフォルト設定は 1 です。

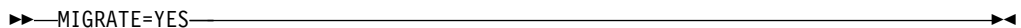
### STAT=

区画統計を生成するかどうかを指定します。区画統計は、STAT=DET の場合に生成されます。STAT=SUM の場合、区画統計は生成されません。デフォルトの設定は STAT=DET です。

SUM は、処理されたすべての区画についての要約報告書を作成します。このオプションは詳細報告書を抑止します。

DET は、詳細報告書を作成します。詳細報告書には、要約報告書に加えて、区画ごとの統計も含まれます。これはデフォルトです。

このユーティリティ制御ステートメントは、SYSIN ストリームに入れられます。



### MIGRATE=

非 HALDB データベースを、マイグレーションのためにアンロードするかどうかを指定します。DD カードは、関連するすべての副次索引データベースについて必要です。

▶▶—KEYRANGEFROMKEYfromkeyvalTOKEYtokeyval—KEYLENkeylenval—▶▶

#### KEYRANGE

KEYRANGE は、非 HALDB から HALDB へのマイグレーションにおいて、複数のアンロード・ジョブを並列に実行することでアンロードの効率を向上させるように設計されたものです。この機能呼び出すには、MIGRATE= ステートメントの直後に、列 1 に KEYRANGE ステートメントを追加してください。KEYRANGE は、MIGRATX= ステートメントに続けると無効になります。

FROMKEY、TOKEY、および KEYLEN パラメーターを使用してキー範囲を指定してください。

#### fromkeyval

ゼロ、あるいは 16 進数で表された前回の区画定義でのハイ・キーを使用してください。

#### tokeyval

16 進数で表された、宛先区画定義でのハイ・キー。

#### keylenval

英数字値で指定される、キーの長さ。

▶▶—MIGRATX=YES—▶▶

#### MIGRATX=

MIGRATX は、1 次データ・セットに加えて、副次索引データ・セットに対してアンロード・ファイルを作成します。それぞれの索引付きソース・セグメントに、データベースの 1 回の受け渡しで、複数の作業ファイルを作成します。これにより、PSINDEX へのマイグレーションのために HD 再編成アンロード・ユーティリティーで複数のマイグレーション・ジョブを実行しなくて済むようになります。作業ファイルには、それぞれ対応するソート制御ステートメントが生成されます。ソートされたファイルは、PSINDEX をロードするときに HD 再編成再ロード・ユーティリティーへの入力として使用できます。アンロードする 1 次データベースを参照するそれぞれの副次索引データベースについて、DFSWRKn および DFSSRTnn DD のカードが必要になります。

▶▶—FALLBACK=YES—▶▶

#### FALLBACK=

フォールバックのための HALDB データベースのアンロード (フォールバック・レコードを作成する) を指定します。

---

## DFSURGU0 ユーティリティーの例

ここでの例は、DFSURGU0 ユーティリティーの使用法を示しています。

この節にあるすべての例では、DBRC と動的割り振りを使用しています。

サブセクション:

- 363 ページの『2 つの HALDB 区画のアンロード』

- ・ 『チェックポイント機能と 2 つの出力コピーを使用した HIDAM データベースの再編成』
- ・ 364 ページの『チェックポイント・データ・セットを使用した異常終了後の再始動』
- ・ 365 ページの『データベースおよび 2 つの副次索引のアンロード』
- ・ 366 ページの『シンボリック・ポインティングを使用する副次索引用のアンロード・レコードのソートおよびマージ』
- ・ 367 ページの『短いキーの指定』
- ・ 367 ページの『新しい区画定義を使用してデータの範囲をマイグレーションする』
- ・ 368 ページの『長いキー (256 バイトまで) に対応するマイグレーション』

## 2 つの HALDB 区画のアンロード

この例では、区画 PDHDOJB がアンロードされた後、区画選択順序で次の区画もアンロードされます。STEPLIB の MDA メンバーを使用して、RECON が動的に割り振られます。

```
//HDUNLDCS JOB 'UNLOAD',MSGCLASS=A,MSGLEVEL=(1,1),CLASS=K,
//          REGION=2M,TIME=1440
//JOB CAT  DD DSN=VCATSHR,DISP=SHR
//HDUNLOAD EXEC PGM=DFSRR00,REGION=2048K,
//          PARM=(ULU,DFSURGU0,DBHDOJ01,,,,,,,,,,,,)
//STEPLIB DD DSN=IMSTESTL.TNUC0,DISP=SHR
//          DD DSN=IMSBLD.RESLIB,DISP=SHR
//SYS PRINT DD SYSOUT=A
//SYS DUMP DD SYSOUT=A
//DFSRESLB DD DSN=IMSBLD.CRESLIB,DISP=SHR
//DFSURGU1 DD DSN=HALDB2-3.UNLOAD,DISP=(,CATLG),
//          UNIT=SYSDA,SPACE=(CYL,(5,3)),DCB=BUFNO=5,VOL=SER=000000
//IMS      DD DSN=IMSTESTS.DBDLIB,DISP=SHR
//          DD DSN=IMSTESTG.DBDLIB,DISP=SHR
//SYSIN    DD *
PARTITION=PDHDOJB,NUMBER=2
/*
//DFSVSAMP DD *
VSRBF=1024,40
VSRBF=4096,40
VSRBF=8192,80
/*
```

推奨事項: HALDB 区画の DD 名は動的に割り振られるため、IBM では、この DD 名をコーディングしないことをお勧めします。

## チェックポイント機能と 2 つの出力コピーを使用した HIDAM データベースの再編成

この例では、チェックポイント機能と 2 つの出力コピーを使用して、HIDAM データベースが再編成されます。再始動は要求されていません。データベース DD ステートメントは 2 つあります。1 つは HIDAM OSAM データ・セットに対するもので、もう 1 つは HIDAM で使用される索引データベース (VSAM) に対するものです。この例が、単一データ・セット・グループ HDAM または PHDAM データベースのアンロードを示している場合は、DD ステートメントはデータベースにアクセスするのに必要ではありません。

```
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSURGU0,DI32DB02'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1210
//DFSUCKPT DD DSN=IMS.CHKPT,DISP=(NEW,KEEP),
// UNIT=SYSDA,VOL=SER=222222,SPACE=(TRK,(50))
//DFSURGU1 DD DSN=IMS.UNLOAD1,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=TAPE11,LABEL=(,SL)
//DFSURGU2 DD DSN=IMS.UNLOAD2,DISP=(NEW,KEEP),
// UNIT=TAPE,VOL=SER=TAPE21,LABEL=(,SL)
//HDPAYROL DD DSN=DATABASE.PAYROLL,DISP=OLD,
// UNIT=SYSDA,VOL=SER=DB0001
//HDINDEXO DD DSN=DATABASE.INDEXO,DISP=OLD,
// UNIT=SYSDA,VOLUME=SER=DB0003
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//DFSCCTL DD *
SBPARM ACTIV=COND
```

HDPAYROL DD ステートメントは、HIDAM データベースの OSAM データ・セットに対するものです。HDINDEXO DD ステートメントは、索引データベースに対するものです。

### チェックポイント・データ・セットを使用した異常終了後の再始動

この例では、異常終了が起こった後、例 1 の内容の実行が再始動されます。チェックポイント・データ・セットが再始動時に使用されます。

```
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSURGU0,DI32DB01',
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A DCB=BLKSIZE=1210
//DFSUCKPT DD DSN=IMS.CHKPT,DISP=(OLD,KEEP),
// UNIT=SYSDA,VOL=SER=222222
//DFSURSRT DD DSN=IMS.CHKPT,DISP=(OLD,KEEP),
// UNIT=SYSDA,VOL=SER=222222
//DFSURGU1 DD DSN=IMS.UNLOAD1,DISP=(MOD,KEEP),
// UNIT=TAPE,VOL=(,,2,SER=(TAPE11,TAPE12)),
// LABEL=(,SL)
//DFSURGU2 DD DSN=IMS.UNLOAD2,DISP=(MOD,KEEP),
// UNIT=TAPE,VOL=(,,2,SER=(TAPE21,TAPE22)),
// LABEL=(,SL)
//HDPAYROL DD DSN=DATABASE.PAYROLL,DISP=OLD,
// UNIT=SYSDA,VOL=SER=DB0001
//HDINDEXO DD DSN=DATABASE.INDEXO,DISP=OLD,
// UNIT=SYSDA,VOLUME=SER=DB0003
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//DFSCCTL DD *
SBPARM ACTIV=COND
```

DFSUCKPT DD ステートメントと DFSURSRT DD ステートメントは、同じデータ・セットを参照することができます。再始動が成功すると、古いチェックポイント・レコードは、次にとられたチェックポイントによって上書きされます。

1 次および 2 次出力 DD ステートメントは、2 つのボリュームを指定するよう変更されています。ユーティリティの前の実行では、ボリュームは 1 つだけが指定されていました。この例では、例えば、使用可能な追加ボリュームがないために、ボリュームの終わりで起こった入出力エラーの出力が原因で、前回の異常終了が起こったことが想定されています。



プログラムは、さまざまな終了原因を区別しないため、チェックポイントがとられた時点で使用中であったボリュームを、適用可能なチェックポイント・レコードに配置します。次に、ボリュームの強制終了 (FEOV) を出してボリューム切り替えを行い、新しい出力ボリュームで実行を続けます。

マルチボリューム出力実行の再始動時に大量のテープ処理が行われるのを避けるためには、正常に完了したボリュームを DD ステートメントから除いてからジョブを実行依頼するようにしてください。プログラムは出力をオープンし、現在マウントされているボリュームを調べて、それがチェックポイントがとられたときマウントされていたボリュームであったかどうかを確認めます。そのボリュームでなければ、プログラムは FEOV を出して、次のボリュームをマウントさせます。この結果、大量のテープ処理が行われることとなります。

## データベースおよび 2 つの副次索引のアンロード

HD 再編成アンロード・ユーティリティーは、作業ファイルの識別とソートのために、以下の報告書を作成します。MIGRATX を使用したときに、指定の副次索引のアンロードにどの DFSWRKnn が対応しているかを確認するためにこの報告書を参照します。

```
Work File Statistics
SINAME  WFNAME  SFNAME  RCDTOTAL  OFFSET  LENGTH
INDEX001 DFSWRK01 DFSSRT01 00000075 0069    0018
INDEX002 DFSWRK02 DFSSRT02 00000150 0069    0018
INDEX003 DFSWRK03 DFSSRT03 00000300 0069    0018
INDEX004 DFSWRK04 DFSSRT04 00000075 0069    0018
```

1 次データベースを 1 つと、その副次索引ベースを 2 つアンロードするには、以下の図の JCL を使用します。

```
//HDUNLOAD EXEC PGM=DFSRR00,
//          PARM=(ULU,DFSURGU0,DDPRIM01,9,0000,,0,,N,0,,,N,N,,N)
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS     DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSURGU1 DD DSN=HDAM2IX.UNLOAD,DISP=(SHR),
//          UNIT=SYSDA,SPACE=(CYL,(5,3)),DCB=BUFNO=5
//DDPRIM01 DD DSN=DDPRIM01,DISP=SHR
//DDINDEX1 DD DSN=INDEX001,DISP=SHR
//INDOVF01 DD DSN=INDOVF01,DISP=SHR
//DDINDEX2 DD DSN=INDEX002,DISP=SHR
//INDOVF02 DD DSN=INDOVF02,DISP=SHR
//DFSWRK01 DD DSN=DFSWRK01,DISP=(NEW,KEEP),VOL=SER=000000,
//          UNIT=SYSDA,SPACE=(CYL,(500,3)),DCB=BUFNO=5
//DFSSRT01 DD DSN=DFSSRT01,DISP=(NEW,KEEP),VOL=SER=000000,
//          UNIT=SYSDA,SPACE=(TRK,(1,0)),DCB=BUFNO=5
//DFSWRK02 DD DSN=DFSWRK02,DISP=(NEW,KEEP),VOL=SER=000000,
//          UNIT=SYSDA,SPACE=(CYL,(500,3)),DCB=BUFNO=5
//DFSSRT02 DD DSN=DFSSRT02,DISP=(NEW,KEEP),VOL=SER=000000,
//          UNIT=SYSDA,SPACE=(TRK,(1,0)),DCB=BUFNO=5
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//SYSIN DD *
MIGRATX=YES
```

以下の図の JCL は、副次索引データベース INDEX01 の作業ファイル DFSWRK01 をソートします。このソートでは、HD アンロードに MIGRATX=YES オプションを使用して、DFSSRT01 にソート制御ステートメントを使用します。ソートされた

出力内容は、HD 再ロード・ジョブのステップに渡されて、HALDB PSINDEX データベースの PSNDX001 をロードします。

```
//SORT01 EXEC PGM=SORT,PARM='CORE=MAX'
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
//SORTIN DD DSN=DFSWRK01,DISP(OLD,PASS),VOL=SER=000000
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(100,5),,CONTIG)
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(100,5),,CONTIG)
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(100,5),,CONTIG)
//SYSIN DD DSN=DFSSRT01,DISP=OLD,VOL=SER=000000
//SORTOUT DD DSN=INDEX001.SORTED.UNLOAD,DISP=(,PASS),
// UNIT=SYSDA,VOL=SER=000000,SPACE=(CYL,(500,3))
//HDRELOAD EXEC PGM=DFSRR00,
// PARM=(ULU,DFSURGL0,PSNDX001,9,0000,,0,,N,0,,,N,N,,N)
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSUNIPT DD DSN=INDEX001.SORTED.UNLOAD,DISP=OLD
//DFSURWF1 DD DUMMY
//SYSPRINT DD SYSOUT=*
//DFSVSAMP DD input for VSAM and OSAM buffers and options
```

## シンボリック・ポインティングを使用する副次索引用のアンロード・レコードのソートおよびマージ

シンボリック・ポインティングを使用する副次索引データベースを HALDB に変換する際、HD 再編成アンロード・ユーティリティーのレコードをソートするためのコーディングは、以下のステップに従ってください。

1. SORT FIELDS 制御ステートメントで使用するために、連結キーへのオフセットとキーの長さを DBD から入手する。
  - a. 副次索引連結キーの開始点へのオフセットが、アンロード・レコードの RDW から常に 63 バイトであるかを判別する。そのバイト数を、DFSORT に対する SORT FIELDS パラメーターのオフセット値として使用する。この値は、HALDB アンロード・レコードをマップするために使用する DFSURGUP マクロで定義されます。
  - b. ソースの長さ、ターゲットの長さ、および /SX 値の長さ (8) を足して、ソート・キーの長さを計算する。例えば、ソースの長さが 10 バイトで、ターゲットの長さが 10 バイトの場合は、ソート・キーの長さが次のようになります。

$$\text{source}(10) + \text{target}(10) + /\text{SX}(8) = 28$$

この値を、DFSORT に対する SORT FIELDS パラメーターの長さの値として使用する。

2. ファイルを分離するために、ヘッダー、トレーラー、およびアンロード・レコードを書き込む JCL ステップを、次の例のようにコーディングする。

```
//SORTIN DD DSN=UNLOAD.OUTPUT,DISP=SHR
//HEADER DD DSN=HEADER.FILE,DISP=(NEW,PASS)
//TRAILER DD DSN=TRAILER.FILE,DISP=(NEW,PASS)
//ULCOPY DD DSN=UNLOAD.COPY,DISP=(NEW,PASS)
//SYSIN DD *
OPTION COPY
OUTFIL INCLUDE=(5,2,CH,EQ,X'0080'),FNAMES=HEADER
OUTFIL INCLUDE=(5,2,CH,EQ,X'0290'),FNAMES=TRAILER
OUTFIL SAVE,FNAMES=ULCOPY
RECORD TYPE=V
END
```

3. アンロード・ファイルだけをソートする JCL ステップを、次の例のようにコーディングする。

```
//SORTIN DD DSN=IMSTESTS.HOSIX.UNLOAD.COPY,DISP=SHR
//SORTOUT DD DSN=IMSTESTS.HOSIX.UNLOAD.SORTED1,DIS=(,CATLG),
//          UNIT=SYSDA,VOL=SER=000000,SPACE=(CYL,(1,5))
//SYSIN DD *
SORT FIELDS=(63,28,CH,A)FILSZ=E1000
RECORD TYPE=V
END
```

4. ヘッダー、ソート済みアンロード、およびトレーラー・ファイルをマージする JCL ステップを、次の例のようにコーディングする。

```
//SORTIN DD DSN=IMSTESTS.HOSIX.UNLOAD.HEADER,DISP=(OLD,DELETE),
//          UNIT=SYSDA,VOL=SER=000000,SPACE=(CYL,(1,5))
//          DD DSN=IMSTESTS.HOSIX.UNLOAD.SORTED1,DISP=(OLD,DELETE),
//          UNIT=SYSDA,VOL=SER=000000,SPACE=(CYL,(1,5))
//          DD DSN=IMSTESTS.HOSIX.UNLOAD.TRAILER,DISP=(OLD,DELETE),
//SORTOUT DD DSN=IMSTESTS.HOSIX.UNLOAD.SORTED2,DSP=(NEW,KEEP),
//          UNIT=SYSDA,VOL=SER=000000,SPACE=(CYL,(1,5))
//SYSOUT DD SYSOUT=A
//SYSIN DD *
OPTION COPY
END
```

## 短いキーの指定

この例では、00000 より大きく K2000 以下のセグメント (key > lowkey & key <= highkey) を取得することにより、DLI データがキーの順序に従ってアンロードされ、RECON リストを作成した RECON 定義を使用して順に挿入されます。このような例は、ユーザー区画選択出口のない HALDB に対応します。

1. HALDB 区画定義の LIST.DB

2. 1 番目の区画についての RECON リスト

```
PARTITION HIGH KEY/STRING (CHAR): (LENGTH=5 ) K2000
PARTITION HIGH KEY/STRING (HEX): D2F2F0F0F0
```

3. RECON リストから得たキー値の 16 進表記を使用して、HDUNLOAD の SYSIN ステートメントを指定してください。KEYRANGE ステートメントは 1 桁目で始まり、キー・ワードのため (括弧内にあるキー値のためではない) 継続文字として「-」を使用して継続することもできますが、71 桁目を超える桁には表示できません。

```
MIGRATE=YES
KEYRANGE FROMKEY(F0F0F0F0F0) TOKEY(D2F2F0F0F0) KEYLEN(5)
```

または

```
KEYRANGE FROMKEY -
(F0F0F0F0F0) -
TOKEY -
(D2F2F0F0F0) -
KEYLEN(5)
```

4. アンロード・ファイルを HDRELOAD に入力する。

## 新しい区画定義を使用してデータの範囲をマイグレーションする

この例では、K2000 より大きく K6000 以下のセグメント (key > lowkey & key <= highkey) を取得することにより、DLI データがキーの順序に従ってアンロードされ、RECON リストを作成した RECON 定義を使用して順に挿入されます。

1. HALDB 区画定義の範囲の LIST.DB (2 から 6)
2. 1 番目から 4 番目の区画についての RECON リスト
 

```

PARTITION HIGH KEY/STRING (CHAR):          (LENGTH=5 )
      K2000
PARTITION HIGH KEY/STRING (HEX):
      D2F2F0F0F0

PARTITION HIGH KEY/STRING (CHAR):          (LENGTH=5 )
      K6000
PARTITION HIGH KEY/STRING (HEX):
      D2F6F0F0F0

```
3. RECON において定義されたキー値の 16 進表記を使用してアンロードするための SYSIN ステートメントの指定。
 

```

|...+....1....+....2....+....3....+....4....+....5....+....6..
MIGRATE=YES
KEYRANGE FROMKEY(D2F2F0F0F0) TOKEY(D2F6F0F0F0) KEYLEN(5)

```
4. アンロード・ファイルを HDRELOAD に入力する。

### 長いキー (256 バイトまで) に対応するマイグレーション

長いキー (256 バイトまで) に対応するマイグレーションのコーディングには、以下を含めてください。

1. RECON リスト。

```

PARTITION HIGH KEY/STRING (CHAR):          (LENGTH=129)
      J2000.....2.....3.....4.....
      ....5.....6.....7.....8.....
      .....9

PARTITION HIGH KEY/STRING (HEX):
      D1F2F0F0F0000000000000000000000000F2000000000000000000000000000000000
      0000F30000000000000000000000000000000000000000000000000000000000000000
      00000000F5000000000000000000000000000000000000000000000000000000000000
      000000000000F700000000000000000000000000000000000000000000000000000000
      0000000000000000F9

```

2. 長いキーに対応するマイグレーション・アンロード SYSIN のキー値は、1 桁目の KEYRANGE キーワード値に続く行で始まり、65 桁目を超えると折り返す。

```

|...+....1....+....2....+....3....+....4....+....5....+....6..
MIGRATE=YES
KEYRANGE FROMKEY (-
F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0
F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0
F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0
F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0
F0F0F0F0F0) -
TOKEY (-
D1F2F0F0F0000000000000000000000000000000000000000000000000000000000000
00F3000000000000000000000000000000000000000000000000000000000000000000
0000F50000000000000000000000000000000000000000000000000000000000000000
000000F700000000000000000000000000000000000000000000000000000000000000
00000000F9) -
KEYLEN(129)

```

---

## 第 26 章 高速 DEDB 直接再編成ユーティリティ (DBFUHDR0)

高速 DEDB 直接再編成ユーティリティ (DBFUHDR0) は、外部ストレージのフラグメント化の除去および制御インターバル (CI) 内でルート・セグメントと直接従属セグメントの順序付けを行うために使用します。

このユーティリティは、DB ユーティリティ従属タイプ領域で実行されます。

各区域のルート・アドレス可能部分内の CI は、作業単位 (UOW) にグループ化されます。DBFUHDR0 ユーティリティは、UOW を一度に 1 つずつ再編成します。各 UOW は、実ストレージ内で再編成され、単一のリカバリー単位として元の UOW にもう一度書き込まれます。実ストレージをこのように使用すると、パフォーマンスが向上し、ロギングが削減されます。

再編成される各 UOW は、再編成されて元の永続位置に書き込まれるまで、DBFUHDR0 ユーティリティにより排他的に保持されます。この期間、他のプログラムは UOW にアクセスできません。標準 IMS 高速機能のコミット処理は、変更する CI だけを書き込むのに使用されます。ストレージをこのように使用すると、パフォーマンスが向上し、ロギングが削減されます。

UOW 全体は、独立オーバーフロー (IOVF) 制御およびデータ CI を含め、1 つの単位としてコミットされるか、あるいは打ち切られます。UOW は、必要であれば、他のあらゆるオンライン・トランザクションと同じ方法でリカバリーできます。

DBFUHDR0 ユーティリティ制御ステートメントは、DEDB オンライン・ユーティリティ・コマンドの形式で実行依頼されます。

サブセクション:

- 『制約事項』
- 370 ページの『前提条件』
- 370 ページの『要件』
- 370 ページの『推奨事項』
- 370 ページの『入出力』
- 371 ページの『JCL 指定』
- 373 ページの『戻りコード』

### 制約事項

高速 DEDB 直接再編成ユーティリティは、別のジョブ・ステップで実行しなければなりません。プログラムは、同じジョブ・ステップ内でユーティリティまたはデータベースを切り替えることはできません。

DBFUHDR0 ユーティリティは、高速機能データベース (DEDB) のみで機能します。

データベースを変更するユーティリティーは、データベースが静止している間に行うことはできません。

## 前提条件

現在、DBFUHDR0 ユーティリティーに対して文書化された前提条件はありません。

## 要件

セグメント・シャントに使用されるオプションのデータ・セットは、LRECL=BLOCKSIZE=80 の固定ブロックでなければなりません。

## 推奨事項

現在、DBFUHDR0 ユーティリティーに対して文書化された推奨事項はありません。

## 入出力

DBFUHDR0 ユーティリティーへの 1 次入力、再編成する必要があるエリア・データ・セットです。DBFUHDR0 ユーティリティーの 1 次出力は、再編成されたエリア・データ・セットです。

以下の表は、DBFUHDR0 ユーティリティーへの入力およびこのユーティリティーからの出力を示したものです。

表 16. DBFUHDR0 ユーティリティーの入出力

入力	出力
再編成する必要があるエリア・データ・セット	ストレージがフラグメント化されずに論理的に順序付けられた UOW が入っている区域
DEDB ユーティリティー・コマンド・データ・セット	出力メッセージと統計

高速 DEDB 直接再編成ユーティリティーでは、入力として DEDB コマンドにより与えられた入力パラメーターが入っているデータ・セットを使用します。

高速 DEDB 直接再編成ユーティリティーにより、次のような出力が得られます。

- ストレージがフラグメント化されずに論理的に順序付けられた UOW が入っている区域
- 出力メッセージと統計が入っているデータ・セット

統計は、ユーティリティーの実行中に収集され、SYSPRINT データ・セットでユーザーに戻されます。統計には、次のものが含まれます。

- UOW 再編成アクティビティー
  - 再編成するよう要求された UOW の数
  - 実際に再編成された UOW の数
  - アンカー・ポイントが空であったためにスキップされた UOW の数

- 再編成に失敗した UOW の数。最初の 5 件の失敗について、再編成できなかった理由が示されます。
- 専用バッファ・セットの使用状況
  - 割り振られたバッファ・セットの総数
  - 専用バッファ・プールが拡張された回数
  - ルート・アドレス可能部分入力に使用されたバッファ・セットの数
  - 出力に使用されたバッファ・セットの数 (再編成されたルート・アドレス可能部分と IOVF データ CI の使用分を含む)
- スペース再利用アクティビティ
  - 解放された IOVF データ CI の数と、再使用された数
  - 新たに割り振られた IOVF データ CI の数
  - 区域内の空き IOVF データ CI の総数、および CI の総数に対する割合としての空き CI の数

#### 報告書の出力例

以下の例は、DBFUHDR0 ユーティリティーにより出力された報告書を示したものです。

```

IMS/FP DEDB UTILITY                                PAGE: 1
*****
*      ONLINE DEDB REORG  UTILITY.                  *
*      REORG AREA DB21AR1                            *
*****
TYPE HSR                                           00055600
*      THE TARGET DATA BASE IS                     00055700
*DBDNAME DEDBJN21                                  00055800
*      THE TARGET DATA BASE AREA IS                00055900
AREA DB21AR1                                       00056200
BUFNO=4
*ERROR TEST
GO                                                  00056300
AREA DB21AR1 HAS      15 UOW'S.
EACH UOW HAS 10 CI'S; 9 AP CI'S, 1 DOVF CI'S.
EACH CI IS 1K, ONE BUFFER SET REQUIRES 10K OF STORAGE
INDEPENDENT OVERFLOW HAS      3 OVERFLOW UNITS, A TOTAL OF 347 DATA CI'S
STATS FOR REORG OF AREA DB21AR1 :
# OF UOWS REQUESTED TO REORG=      15; LOW UOW      0, HIGH UOW      14
# OF UOWS ACTUALLY REORG'D=      6; LOW UOW      0, HIGH UOW      14
# OF UOWS SKIPPED=      9, ALL ANCHOR POINT CI'S WERE EMPTY
# OF IOVF CI'S FREED BY REORG: 141 - 135 (REUSED) =      6
# OF IOVF CI'S ALLOCATED:      6 (NEW) + 135 (REUSED) = 141
# OF FREE IOVF CI'S AFTER REORG= 206, PERCENT FREE= 59
PRIVATE BUFFER SET INITIAL ALLOCATION= 4, EXTENSION COUNT= 4
RAP INPUT BUFFER SETS= 2, OUTPUT/IOVF BUFFER SETS= 6
UOW      6 USED 28 BUFFERS FOR IOVF I/O, THE HIGHEST USAGE
PERFORMANCE STATS: # OF ASYNCHRONOUS READ AHEAD I/O'S= 7,
# OF WAITS FOR UOW LOCKS=      0,
# OF WAITS FOR PRIVATE BUFFERS=      3

IMS/FP DEDB UTILITY                                PAGE: 2
DFS2657I UTILITY EXECUTED AS REQUESTED

```

#### JCL 指定

DBFUHDR0 ユーティリティーは、標準の z/OS ジョブとして実行されます。  
 DBFUHDR0 ユーティリティーの JCL 指定には、JOB ステートメント、EXEC ス

テートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティー制御ステートメントを JCL ステートメントに含めることが必須であり可能です。

#### EXEC ステートメント

DBFUHDR0 ユーティリティーの EXEC ステートメントは、必要な JCL が入っている FPUTIL プロシージャーを呼び出すことができますが、次の形式で指定することもできます。

```
PGM=DFSRR00
```

#### DD ステートメント

##### STEPLIB DD

再編成ユーティリティーが入っているライブラリーを記述します。

##### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

##### SYSIN DD

ユーティリティー制御ステートメントが入っている入力制御データ・セットを記述します。

DEDB オンライン・ユーティリティーでは、QSAM を使用して、SYSIN データ・セットを読み取ります。入力は、ブロック化または非ブロック化、固定長または可変長のいずれでも構いません。レコードは数行の入力ステートメントまたはコマンドとして解釈され、文字は EBCDIC です。レコードの長さは、80 文字から 120 文字までです。必要なデータが入力行に収まる場合は、レコードは 80 文字より短くて構いません。

##### SYSPRINT DD

メッセージと統計が入っている出力データ・セットを記述します。

##### INDD DD

セグメント・シャントに使用されるオプションのデータ・セットを指示します。このデータ・セットは、固定ブロック LRECL=BLOCKSIZE=80 でなければなりません。

#### JCL の例

次の例は、再編成ユーティリティーを実行するための JCL とユーティリティー制御ステートメントを示しています。

```
//ORGDB01 EXEC FPUTIL,
//          DBD=DEDBJN01,REST=00
//*
//*          DBD=DBDNAME AS TARGET DATABASE FOR THIS UTILITY RUN
//*          REST=RESTART NUMBER FOR THIS RUN
//* //SYSIN   DD *
*****
*          HIGH SPEED DEDB ROOT REORGANIZATION UTILITY
*****
*
* COMMANDS and OPERATORS      COMMENTS
* TYPE HSREORG
*                               SET ERROR OPTION
ERROR HALT
*                               THE TARGET DATABASE IS
```



```

*                               DBDNAME DEDBJN01
*                               THE TARGET AREA IS
AREA DB1AREA1
GO
*                               THE TARGET DATABASE IS
*                               DBDNAME DEDBJN01
*                               THE TARGET AREA FOLLOWS
AREA DB1AREA2
GO

```

## 戻りコード

DBFUHDR0 ユーティリティは、以下の戻りコードの 1 つ以上を生成します。

コード 意味

- 0 ユーティリティは要求どおりに実行された
- 2 INDD 内のエラーをセグメント・シャント中に検出。ユーティリティは正常に実行されている。ユーティリティ出力を参照。
- 4 SYSPRINT エラーか、あるいは要求された数の UOW の再編成の失敗
- 8 パラメーター分析中のエラー、または複数のエラーが発生した。ユーティリティ出力を参照。

ゼロ以外のすべての戻りコードには、エラー・メッセージが一緒に出されます。

---

## DBFUHDR0 ユーティリティの制御ステートメント

DBFUHDR0 ユーティリティ制御ステートメントは、DEDB オンライン・ユーティリティ・コマンドの形式で実行依頼されます。コマンドは、SYSIN DD ステートメントにコーディングされます。制御コマンドは、再編成する入力エリア・データ・セットを記述するためにも使用されます。

### コマンドのフォーマット

パラメーターをフリー・フォームで指定します。

例外: フィールドは、最大文字数が 120 文字であることを除いて、何らかの特定の列に限定されません。

各ステートメントは新しい行で始める必要があります。コマンド名は、最初を非空白文字で開始し、1 つの空白または等号で終了します。コメントを示すときは、先頭文字にアスタリスクを使用します。

コマンドでオペランドが必要な場合は、次の文字を非空白か、等号以外かでオペランド欄を開始します。コマンドで複数のオペランドを使用する場合は、オペランドをコンマで分離します。オペランド欄は、1 つの空白、または行の最後で終了します。ただし、EBCDIC 値の一部である文字 (引用符で囲まれた文字ストリングとして指定) は、終了コンマまたは終了空白としてカウントしません。

行のオペランド欄に続く文字は、コメントとして処理されます。

スキャン・ユーティリティおよび削除ユーティリティのオペランドは、領域に固有です。複数の領域が各実行で処理される場合、QUITCI、V5COMP、および EXCLUDE などのキーワードを領域ごとに指定する必要があります。これらの各オ

ペランドに適用可能な値は、各 AREA コマンドに対する各 GO コマンドの前にコーディングする必要があります。例えば、ユーティリティの実行ですべての領域に対して V5COMP 関数が必要な場合、各 AREA コマンドに対する各 GO コマンドの前に V5COMP をコーディングする必要があります。

## コマンドの継続

以下のように懸垂コンマを使用することで、オペラントの間でオペラント欄を継続することができます。

```
STARTSEQ OP='FIELD=FLD1',      (最初の行)
          VALUE=X'C4C5C2'      (継続行)
```

終了の引用符、コンマ、および再オープン引用符を以下のように使用して、引用符付きストリングを継続します。

```
STARTROOT X'C1C2C3C4C',      (最初の行)
          '5C6C7',            (2 番目の行)
          'C8C9'              (最後の行)
```

懸垂コンマとコメントの間に 1 つのブランクを使用して、各行にコメントを含めることができます。

以下のコマンドは、DBFUHDR0 ユーティリティに適用できます。

### AREA (必須)

どの区域を処理するのかを、制御領域 DD ステートメント上の名前指定します。このコマンドは、各 GO コマンドまたは各 RUN コマンドの後で繰り返さなければなりません。名前は 1 文字から 8 文字までにしなければなりません。最初の文字は英字にし、残りの文字は英数字にします。英字には、@、#、および \$ が含まれます。

### BUFNO (オプション)

DBFUHDR0 ユーティリティの場合、DEDB ユーティリティ・コマンド BUFNO は、バッファ数ではなく、バッファ・セット数を指定します。バッファ・セットとは、完全な UOW を保持できる大きさのバッファのセットです。例えば、16 個の CI から成る UOW には、各バッファが 1 個の CI を保持できる大きさの、16 個のバッファから成るバッファ・セットが必要です。

高速 DEDB 直接再編成ユーティリティの場合、バッファの最低数は、次式で計算されます。

$$\text{Minimum number of buffers} = \text{number of hierarchic levels of DEDB} + 12$$

BUFNO の指定がない場合は、デフォルトが使用されます。デフォルトのバッファ数は、作業単位 (UOW) 当たりの制御インターバル (CI) 数に DEDB の階層レベル数 + 12 を加えたものです。

高速 DEDB 直接再編成ユーティリティでの BUFNO の使用法の詳しい説明については、376 ページの『バッファおよび BUFNO コマンド』を参照してください。

### ERRORACTION (オプション)

エラーが検出され、エラー・メッセージが印刷されたときに、指定のユーティリティにとらせるアクションを指定します。このコマンドはオプションですが、

必要な回数だけ指定することができます。ERRORACTION が指定されていないと、デフォルトとして STOP オペランドが使用されます。

#### **STOP**

ユーティリティを即時に停止させます。

#### **ABEND**

ユーティリティが U1039 異常終了ダンプを作成することを指定します。

#### **SCAN**

ユーティリティがエラーの有無を調べるために入力のスキャンを続けることを指定します。

#### **SCANRUN**

ユーティリティが、次の GO コマンドが現れた後で検出されたエラーを無視することを除けば、SCAN オペランドと同じ処理を指定します。

#### **GO (オプション)**

一連の要求を区切ります。2 つ以上の区域の処理、2 つ以上の出口ルーチンの使用、あるいは 2 つ以上のデータ範囲の処理を 1 つのジョブ・ステップで行う場合は、必ず指定する必要があります。

#### **STARTUOW (オプション)**

再編成する最初の作業単位 (UOW) を指定します。このコマンドはオプションですが、再編成ユーティリティで使用します。

#### **STOPAFTERFAIL# (オプション)**

使用可能なバッファがなかったために再編成に失敗した UOW の数を指定します。この失敗が起きると、ユーティリティは異常終了します。最大値は 999 です。STOPAFTERFAIL# の指定がない場合は、デフォルトが使用されます。デフォルト値は 5 です。STOPAFTERFAIL# が 0 と指定された場合、強制される制限はなく、ユーティリティは、DEDB 領域の終わりに達するまで、継続して、次の UOW の再編成を試みます。失敗した UOW が最大 999 までリストされます。

#### **STOPUOW**

再編成する最後の作業単位 (UOW) を指定します。このコマンドはオプションですが、再編成ユーティリティで使用します。

#### **TYPE (必須)**

実行のタイプとして、スキャン、削除、または再編成のいずれかのユーティリティを指定します。

**重要:** このコマンドは必須であり、最初の GO コマンドの前か、あるいは SYSIN ファイルの終わりの前に入れておかなければなりません。

プログラムは同じジョブ・ステップ内でユーティリティを切り替えることができないため、ジョブ・ステップ内に指定できる TYPE コマンドは 1 つだけです。DBFUHDR0 ユーティリティを呼び出すのに使用される TYPE コマンドのキーワードは REORG です。同義語は HSR、HSREORG、DR および R です。

STARTUOW コマンドを使用すると、再編成を開始する UOW を指定することによって、区域のルート・アドレス可能部分の一部を再編成することができます。STOPUOW コマンドは、再編成する最後の UOW を指定します。区域内の最初の

UOW は番号 0、2 番目は番号 1、3 番目は番号 2 であり、以下同様です。STARTUOW コマンドの指定がないと、ユーティリティーは、区域内の最初の UOW から再編成を開始し、STOPUOW コマンドに指定された UOW まで続行します。STOPUOW コマンドの指定がないと、ユーティリティーは、指定された開始 UOW から開始し、その区域の終わりまで処理します。開始 UOW と停止 UOW の両方を指定する場合、開始 UOW の値は、停止 UOW 以下でなければなりません。再編成される UOW が 1 つだけである場合、開始 UOW と停止 UOW は同じ値でなければなりません。どちらかの UOW に無効値が指定されると、エラー・メッセージが印刷され、データは再編成されません。UOW 番号は、10 進数形式でも、16 進数形式でも入力できます。

---

## DBFUHDR0 ユーティリティーの実行

DBFUHDR0 ユーティリティーの操作上の考慮事項として、BUFNO コマンド、セグメント・シャント、リカバリーと再始動、およびエラー処理があります。

サブセクション:

- 『バッファーおよび BUFNO コマンド』
- 377 ページの『セグメント・シャント』
- 379 ページの『リカバリーおよび再始動』
- 379 ページの『エラー処理』

### バッファーおよび BUFNO コマンド

DBFUHDR0 ユーティリティーの場合、BUFNO は、バッファー数ではなく、バッファー・セット数を指定します。BUFNO が指定されない場合、デフォルトは 3 つのバッファー・セットです。

BUFNO コマンドにバッファーを 4 つ以上指定すると、2 つのバッファー・セットはルート・アドレス可能部分入力に使用されるため、ユーティリティーは非同期先行読み取りを使用できるようになります。

非同期先行読み取りとは、ユーティリティーが、現行の UOW を再編成する間に次の UOW を読み取ることができることを意味しています。この結果、区域の独立オーバーフロー部分がルート・アドレス可能部分とは別の物理装置上にある場合は、パフォーマンスが著しく向上する可能性があります。

IOVF 部分とルート・アドレス可能部分が同じ物理装置上にある場合は、パフォーマンスはあまり向上しません。

DBFUHDR0 ユーティリティーは、バッファーの待機があった場合、専用バッファー・プールを動的に拡張して、さらに多くのバッファー・セットを得ます。許可される拡張の数は、元の BUFNO 指定と同じです。バッファー・プールをこのように動的に拡張することにより、ユーティリティーは、最小限のバッファー・セットで最大限のパフォーマンスを実現します。

UOW の再編成に最低限必要なバッファー・セットの数は、アクセスしなければならない独立オーバーフロー CI の数によって決まります。

- ルート・アドレス可能部分入力 UOW

- 出力のための、再編成されたルート・アドレス可能部分 UOW
- コミット・ポイントに達し、UOW が完全に再編成されるまで解放されるか、あるいは割り振られる、独立オーバーフロー CI

独立オーバーフロー CI に必要なバッファの数は予測できません。十分な数のバッファが使用できない場合、その UOW の再編成はできません。したがって、ユーティリティーは動的プール拡張を使用して、必要なだけバッファを入手します。

各区域に最適な BUFNO 指定は、経験で分かるようになります。指定拡張が必要でなく、過剰なバッファ・セットが指定されていない場合に、最も効率的な指定となります。

バッファは、当該ユーティリティーが処理を始めた時点で作成される専用バッファ・プールから動的に得られます。このバッファ・プールは、BUFNO コマンドに指定された限度まで、必要に応じて拡張することができます。バッファ・プール拡張は、1 バッファ・セット単位で行います。

バッファ・プールは、パフォーマンス上の理由から、永続的にページ固定です。バッファ・セットは永続的にページ固定であるため、実ストレージが必要です。ご使用の実ストレージが限られている場合は、BUFNO の指定が重要になります。DBFUHDR0 ユーティリティーを同時にいくつ実行するかについては、ページ固定を考慮して決めます。このユーティリティーをいくつ実行できるかについて、ストレージが影響を及ぼすことはありませんが、実行する数が多すぎると、システム・パフォーマンス全体が低下する可能性があります。

再編成される各区域ごとに、専用バッファ・プールが作成され、ページ固定されます。使用される実ストレージの容量が重大になる場合があります。例えば、デフォルトの BUFNO 指定を使用し、UOW がそれぞれ 16 KB というサイズの CI を 45 個更新する区域の場合、バッファ・プール拡張を計算に入れなくても、2160 KB の実ストレージが必要になります。特別な場合を除き、3 というデフォルトで十分です。ただし、最大のパフォーマンスを得るには、環境に合わせて調整しなければなりません。

## セグメント・シャント

セグメント・シャント は、指定されたセグメントを従属オーバーフロー (DOVF) または独立オーバーフロー (IOVF) に直接再編成する機能で、ルート・アンカー・ポイント制御インターバル (RAP CI) に現在スペースが存在する場合でも、RAP CI へのこれらのセグメントのコピーをバイパスします。セグメント・シャントにより、場合によっては、DEDB UOW を再編成し、新しい挿入に備えて RAP CI 内にスペースを確保できます。

入力データ・セット INDD を使用して、シャントするセグメントを指定できます。このデータ・セットは、80 の論理レコード長 (LRECL)、80 のブロック長、さらにレコード当たり 1 つの入力カードを持つ固定ブロックでなければなりません。入力カードは左寄せする必要があります。そうでないと、カードは無視されます。このデータ・セットは、セグメント・シャント機能を使用する場合にのみ必要です。

シャントできるセグメントは、直接従属セグメントだけです。ルート・セグメントおよび SDEP はシャントできません。

INDD データ・セットは、次の 3 つの入力カード・タイプで構成されます。

#### **ERRACTN=aaaa**

オプションの入力カード・タイプです。これを含めるときは、INDD データ・セットの最初のカードにしなければなりません。このカードの有効なオプションは、ERRACTN=CONT、ERRACTN=EXIT、および ERRACTN=TERM です。

ERRACTN=CONT がデフォルト・オプションです。INDD データ・セットの残りの部分でエラーが見付かった場合、無効なカードは無視され、入力カードの残りの部分は受け入れられます。

ERRACTN=EXIT が指定された場合、INDD データ・セットのカードに無効なデータが含まれていると、データ・セット全体が無視され、再編成ジョブは、INDD データ・セットが指定されていない場合と同様に進行します。

ERRACTN=TERM が指定された場合、INDD データ・セットのカードに無効なデータが含まれていると、データ・セット全体が無視されます。再編成ジョブは実行されず、DL/I 状況として Z0 が返されます。

#### **AREA=areaname**

単一区域の入力ストリームを開始します。この区域の入力ストリームは、次の AREA= カードか、またはファイルの終わりで終了します。入力ストリーム内の後続のカードはすべて、関連した AREA= カードについてシャントするセグメントと見なされます。

#### **segmentn**

前の AREA= カードと関連したセグメント名が含まれます。

以下の例は、12 の区域、1 つのルート・セグメント ROO1、直接従属 DD1、DD2、DD3、DD4、および DD5 を含む DEDB を示しています。

```
ERRACTN=CONT
AREA=AREA_1
DD1
DD2
DD3
AREA=AREA_10
DD4
DD2012
AREA=AREA_11
DD1
DD4
```

この例では、次のようになります。

- ERRACTN=CONT が指定されているため、無効なカードはすべて無視されます。
- AREA\_1 では、セグメント DD1、DD2、および DD3 は RAP CI に再編成されませんが、スペースがあれば、ROOT、DD4、および DD5 は RAP に挿入できます。
- AREA\_2 から AREA\_9 と AREA\_12 では、区域は、セグメント・シャントなしに、通常のものとして再編成されます。
- AREA\_10 では、セグメント DD4 はシャントされ、カード DD2012 は無視されます。

- AREA\_11 では、セグメント DD1 および DD4 は両方ともシャントされます。

## リカバリーおよび再始動

DBFUHDR0 ユーティリティが異常終了した場合、データベースまたは区域には、データ保全性を維持するための終結処置は必要ありません。失敗したときに処理中だった UOW は記憶され、その間にその区域に対して別のユーティリティが実行されない限り、このユーティリティは、その区域に対して次に呼び出しが行われたときに、失敗した地点から再開します。ただし、このユーティリティの実行中に IMS または z/OS が異常終了した場合は、失敗した地点が記憶されません。

DBFUHDR0 ユーティリティの実行中に IMS またはシステムが失敗した後でも、終結処置は不要です。必要なのは、再編成され、コミットされた最後の UOW が元どおり、完全に DASD に書き込まれたことを確認することだけです。これは通常の高速機能 REDO 処理であり、IMS をコールド・スタートしているときに緊急時再始動または順方向リカバリー・ユーティリティにより行われます。

DBFUHDR0 ユーティリティでは、REST=00 の場合と REST=01 の場合とで処理に差異はありません。


## エラー処理

以下の表は、DBFUHDR0 ユーティリティによる入出力エラー処理と、入出力エラー後の区域の状況を要約したものです。

表 17. DBFUHDR0 ユーティリティによる入出力エラー処理の要約

エラー・タイプ	処置	区域の状況
読み取りエラー (ルート・アドレス可能部分 UOW または独立オーバーフロー制御インターバルのいずれかにおいて)	UOW は失敗し、統計が収集され、ユーティリティ処理は続行する (重大なエラーでない場合)	使用可能
書き込みエラー	処理は続行する	使用可能

関連資料:

 FPUTIL プロシージャ (システム定義)





## 第 27 章 HISAM 再編成再ロード・ユーティリティ (DFSURRL0)

HISAM 再編成再ロード・ユーティリティは、HISAM 再編成アンロード・ユーティリティ (DFSURUL0) の出力データ・セットを使用して HISAM データベースを再ロードするために使用します。

DFSURRL0 ユーティリティは、さらに次のような目的でも使用できます。

- HISAM 再編成アンロード・ユーティリティから得た再編成済み出力データ・セットから、副次索引を作成するかマージします。
- HISAM 再編成アンロード・ユーティリティによってアンロードされた HISAM データベースの 1 次索引を再ロードします。

HISAM 再編成再ロード・ユーティリティは、実行されるたびに統計報告書を作成します。OPTIONS ユーティリティ制御ステートメントで NSTAT パラメータを使用して、統計報告書を抑止することができます。

以下の図は、HISAM 再編成再ロード・ユーティリティのフロー・ダイアグラムです。

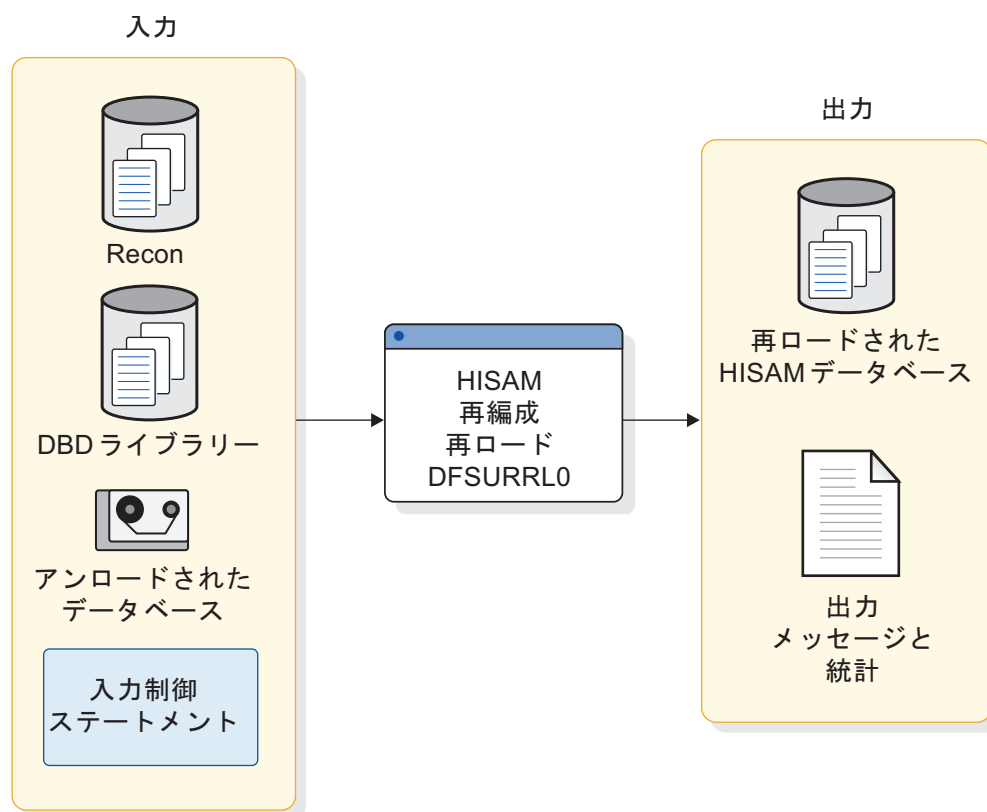


図 71. HISAM 再編成再ロード・ユーティリティ (DFSURRL0)

サブセクション:

- 『制約事項』
- 『前提条件』
- 『要件』
- 『推奨事項』
- 383 ページの『入出力』
- 385 ページの『JCL 指定』
- 387 ページの『戻りコード』

## 制約事項

HISAM 再編成再ロード・ユーティリティを使用するときは、次のような制約があります。

- このユーティリティは、PSINDEX データベースを再ロードするためには使用できません。
- このユーティリティが使用できるのは、論理レコード長とブロック・サイズの変更を行うときだけです。データベースの構造上の変更を行うときは、HD 再編成再ロード・ユーティリティ (DFSURGL0) を使用します。
- このユーティリティは、副次索引が付いている HISAM データベース、または論理関係の中で使用されている直接アドレス・ポインタを持つセグメントが置かれている HISAM データベースを再編成するためには使用できません。この場合には、HD 再編成アンロード・ユーティリティ (DFSURGU0) と HD 再編成再ロード・ユーティリティ (DFSURGL0) を使用してください。
- DFSURRL0 ユーティリティは、全機能非 HALDB データベースのみで機能します。
- データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。
- アンロードされた高速機能の副次索引データベースを DFSURRL0 に入力することはできません。高速機能の副次索引データベースを再編成するには、IDCAMS (データ操作ユーティリティ) の REPRO コマンドを発行してください。高速機能の副次索引データベースのリカバリーには、IMS ベンダー製品に含まれるデータベース・リカバリー・ユーティリティ (DFSURDB0) または索引ビルダー・ツールのいずれかを使用してください。

## 前提条件

現在、DFSURRL0 ユーティリティに対して文書化された前提条件はありません。

## 要件

データベースを再ロードするための IMS リリースは、そのデータベースをアンロードするために使用したものと同一リリースを使用しなければなりません。

## 推奨事項

現在、DFSURRL0 ユーティリティに対して文書化された推奨事項はありません。

## 入出力

DFSURRL0 ユーティリティーへの 1 次入力、HISAM 再編成アンロード・ユーティリティー (DFSURUL0) で作成されたアンロード・データ・セットです。

DFSURRL0 ユーティリティーの 1 次出力は、再ロード・データ・セットです。

HISAM 再編成再ロード・ユーティリティーの入出力は、以下の表に示すとおりです。

表 18. DFSURRL0 ユーティリティーが使用するデータ・セット

入力	出力
RECON	再ロードされた HISAM データベース用の VSAM KSDS データ・セット
アンロードされたデータベース	場合により、オーバーフロー OSAM データ・セット
DBD ライブラリー	出力メッセージと統計
入力制御ステートメント	

### 報告書の出力例

HISAM 再編成再ロード・ユーティリティーは、再ロードされたデータ・セット・グループ別にメッセージ、統計、および監査証跡を提供します。このユーティリティーから得られるメッセージと統計の例を、説明情報と一緒に以下の図に示しています。

```

      H I E R A R C H I C A L   I N D E X E D   S E Q U E N T I A L
      D A T A B A S E   R E O R G A N I Z A T I O N   R E L O A D
      D A T A   S E T   G R O U P   S T A T I S T I C S
DATABASE - DIVNTZ04
PRIMARY DD - DBHVSAM1
OVERFLOW DD - DBHVSAM2

DEPENDENT OVERFLOW CHAINS (#)
PRIMARY ROOTS OVERFLOW DEPENDENTS NO. LONGEST SHORTEST AVG
3             2             2             1 1             1.00 1

ROOT WITHOUT OVERFLOW CHAINS (BYTES)
NO. LONGEST SHORTEST AVERAGE
1  0             0             0.00

      S E G M E N T   L E V E L   S T A T I S T I C S
SEGMENT SEGMENT TOTAL SEGMENTS BY SEGMENT TYPE
NAME    LEVEL    RELOADED    DIFFERENCE
J1      1          3
J2      2          3
J3      3          1
J4      4          1
J5      2          4
J6      3          4
J7      4          2
J8      5          1
J9      2          3
J10     3          3
J11     4          0
J7P     3          2
J12     2          0
J13     3          0
J14     4          0
J13X    3          0

```

```

TOTAL SEGMENTS IN DATA SET GROUP
UNLOADED          RELOADED          DIFFERENCE
27                27
DFS340I  DATABASE DIVNTZ04 HAS BEEN SUCCESSFULLY RELOADED BY FUNCTION SR
DFS339I  FUNCTION SR HAS COMPLETED NORMALLY RC=00

```

この実行でいずれかのオプションが選択されていると、さまざまなメッセージが生成され、ページ見出しの直後に示されます。

通常、統計は、HISAM 再編成再ロード・ユーティリティーが実行されるたびにとられます。そのために再ロード時間がいくぶん増加するので、プロセッサ時間で料金を請求されるインストール・システムでは、統計の記録を抑止することもできます。これを行うには、OPTIONS ユーティリティー制御ステートメントで NSTAT パラメーターを使用します。

データ・セット・グループ別のメッセージに続いて、「DATA SET GROUP STATISTICS」という見出しが示されます。

#### **DATABASE**

データベース名

#### **PRIMARY DD**

データ・セット・グループの VSAM KSDS DD 名

#### **OVERFLOW DD**

データ・セット・グループの VSAM ESDS DD 名

#### **PRIMARY ROOTS**

ロードされたルートの数

#### **OVERFLOW DEPENDENTS**

ロードされた従属レコードの数

#### **DEPENDENT OVERFLOW CHAINS (#)**

VSAM ESDS にチェーニングされた従属レコードを持つルートに関する統計:

**NO** VSAM ESDS 従属レコードを持つ VSAM KSDS の数

#### **LONGEST**

1 つの VSAM KSDS からチェーンが切り離された VSAM ESDS 従属レコードの最大数

#### **SHORTEST**

1 つの VSAM KSDS からチェーンが切り離された VSAM ESDS 従属レコードの最小数 (非ゼロ)

#### **AVERAGE**

VSAM KSDS (チェーンを持つものの 1 つ) からチェーンが切り離された VSAM ESDS 従属レコードの平均数

#### **ROOTS WITHOUT OVERFLOW CHAINS (BYTES)**

VSAM ESDS 従属レコードを持っていない VSAM KSDS に関する統計:

**NO.** VSAM ESDS 従属レコードを持っていないルートの数

#### **LONGEST**

VSAM ESDS 従属レコードを持っていない最大ルート・レコード (バイト数)

### SHORTEST

VSAM ESDS 従属レコードを持っていない最小ルート・レコード (バイト数)

### AVERAGE

VSAM ESDS 従属レコードを持っていないルート・レコードの平均長 (バイト数)

「SEGMENT LEVEL STATISTICS」の節には、以下のフィールドがあります。

#### SEGMENT NAME

この統計行が適用されるセグメントの名前

#### SEGMENT LEVEL

このセグメントがデータベースに置かれている階層レベル

#### TOTAL SEGMENTS BY SEGMENT TYPE

再ロードされたセグメントの総数および再ロードされた数とアンロードされた数の差

#### RELOADED

アンロードされたこのタイプのセグメントの総数

#### DIFFERENCE

再ロードとアンロードの数が等しければ、このフィールドは空白になっています。等しくなければ、その差が印刷されます。

データベースが共用副次索引ベースの一部であるときは、セグメントの出現回数は、常に最初のセグメント名の下に示されます。他のセグメント名の出現回数はすべてゼロになっています。IMS は、どのセグメント名が統計と一致しているかを区別できませんが、出現回数は正確です。

「TOTAL SEGMENTS IN DATA SET GROUP」の節には、以下のフィールドがあります。

#### UNLOADED

HISAM 再編成アンロード・ユーティリティ (DFSURUL0) によってアンロードされたセグメントの総数

#### RELOADED

HISAM 再編成再ロード・ユーティリティ (DFSURRL0) によって再ロードされたセグメントの総数

#### DIFFERENCE

上記 2 つの総計に差があれば、その差

### JCL 指定

DFSURRL0 ユーティリティは、標準の z/OS ジョブとして実行されます。DFSURRL0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。オプションのユーティリティ制御ステートメントを 1 つ JCL ステートメントに含めることができます。

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

#### EXEC ステートメント

EXEC ステートメントは次の形式になっていなければなりません。

```
PGM=DFSRR00,PARM='ULU,DFSURRL0'
```

SPIE、BUF、DBRC といった通常の IMS 定位置パラメーターは、 PARM フィールドの中でプログラム名の後に置くことができます。

#### DD ステートメント

DFSURRL0 ユーティリティは、必須 DD ステートメントとオプションの DD ステートメントを多数使用します。

DFSURRL0 ユーティリティの DD ステートメントには、以下のものが含まれます。

#### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。APF 無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

#### DFSRESLB DD

IMS SVC モジュールが入っている APF 許可ライブラリーを指します。

#### IMS DD

再編成するデータベースを記述する DBD が入っているライブラリーを定義します。このデータ・セットは、直接アクセス装置に置いておかなければなりません。

#### SYSPRINT DD

出力メッセージおよび統計データ・セットを定義します。このデータ・セットは、テープ、直接アクセス装置、またはプリンターに置いておくことも、出力ストリームに送ることもできます。

#### DFSUINxx DD

アンロードされた入力データ・セットを定義します。最初に再ロードする入力データ・セット・グループは DD 名 DFSUIN01 で定義され、後続の入力データ・セットごとに数値が 1 ずつ増えます。

#### vsamout1 DD

再ロードする VSAM KSDS を定義します。DD 名は、このデータ・セットがアンロードされるときに参照された DBD に入っている名前と同じものでなければなりません。この DD ステートメントでさらに大きなスペース・パラメーターを指定すると、データベース・スペース割り振りのサイズを増やすことができます。

#### vsamout2 DD

再ロードする VSAM ESDS 出力データ・セットを定義します。名前は、このデ

ータ・セットがアンロードされる時に参照された DBD に入っている DD 名と同じものでなければなりません。この DD ステートメントでさらに大きなスペース・パラメーターを指定すると、データベース・スペース割り振りのサイズを増やすことができます。

要件: VSAM データベースの場合は、DEFINE 制御ステートメントでスペースを変更する必要があります。

#### **SYSIN DD**

入力制御情報データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることができます。ユーティリティーへの入力としてユーティリティー制御ステートメントが指定されていない場合、この DD ステートメントは必要ありません。

#### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。この DD ステートメントは必須です。

このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。

#### **SYSABEND DD または SYSDUMP DD**

ダンプ・データ・セットを定義します。これらの DD ステートメントはオプションです。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

#### **RECON1 DD**

最初の DBRC RECON データ・セットを定義します。

#### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。

#### **RECON3 DD**

RECON1 または RECON2 にエラーが見付かったとき DBRC が使用する、オプションのデータ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

### **戻りコード**

DFSURRL0 ユーティリティーは、次の戻りコードを 1 つ以上生成します。

プログラムが終了すると、次の戻りコードの 1 つが返されます。

コード 意味

- 0 すべての操作が正常に完了した
- 4 1 つ以上の警告メッセージが出された
- 8 1 つ以上の操作が正常に完了しなかった
- 16 重大エラーが起こったためプログラムが終了した

関連概念:

➡ IMS バッファ・プール (システム定義)

関連資料:

437 ページの『第 31 章 ユーティリティー制御機能 (DFSUCF00)』

➡ DBBBATCH プロシージャ (システム定義)

➡ DLIBATCH プロシージャ (システム定義)

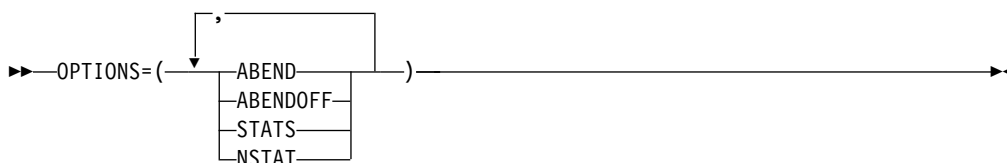
---

## DFSURRL0 ユーティリティーの制御ステートメント

HISAM 再編成再ロード・ユーティリティーには、OPTIONS というオプションの制御ステートメントが 1 つあります。

OPTIONS 制御ステートメントを使用して、以下について DFSURRL0 ユーティリティーの状態を制御します。

- 異常終了時にダンプを作成するかどうか
- 実行のたびにロード統計を生成するかどうか



### OPTIONS=

OPTIONS 制御ステートメントであることを示します。

### ABEND

実行を終了させるような条件が起これると、ABEND U0359 を出して終了させます。SYSABEND または SYSUDUMP DD ステートメントが指定されていると、ダンプが印刷されます。

### ABENDOFF

ABEND 機能をオフにします。異常条件が起これるとプログラムは終了しますが、ABEND コードは出されず、また、ダンプもとられません。

例外: ABENDOFF は初期デフォルトです。ただし、その前に ABEND が指定されていると、ABEND は ABENDOFF が JCL コードの中で現れるまで有効です。

### STATS

HISAM 再編成アンロード・ユーティリティーによって統計がとられていれば、ロードされた数を、HISAM 再編成アンロード・ユーティリティーから得た数と突き合わせて比較します。

OPTIONS=STATS と指定しておくか、あるいは OPTIONS ステートメントを使用していないと、再ロードごとに統計がとられます。

### NSTAT

HISAM 再編成アンロード・ユーティリティーがとった統計を無視させます。



OPTIONS=NSTAT と指定すると、このジョブ・ステップでは統計はとられません。

---

## DFSURRL0 ユーティリティーの例

以下の図は、DFSURRL0 ユーティリティーの JCL の例を示しています。

図 72. DFSURRL0 ユーティリティーの JCL の例

```
//HSRELOAD EXEC PGM=DFSRR00,PARM='ULU,DFSURRL0,,,01,,,,,,Y,N'  
//*****  
//* EXECUTE DATA BASE REORG RELOAD FOR DIVNTZ04 */  
//*****  
//IMS DD DSN=IMSVS.DBDLIB,DISP=SHR  
// DD DSN=I11X.DBDLIB,DISP=SHR  
//RECON1 DD DSN=IMSVS.RECON1,DISP=SHR  
//RECON2 DD DSN=IMSVS.RECON2,DISP=SHR  
//RECON3 DD DSN=IMSVS.RECON3,DISP=SHR  
//SYSPRINT DD SYSOUT=A  
//DFSUIN01 DD DSN=DATAOUT1,DISP=(OLD,KEEP),UNIT=SYSDA,VOL=SER=USER02  
//DBHVSAM1 DD DSN=DIVNTZ04.JJXS01K,DISP=SHR  
//DBHVSAM2 DD DSN=DIVNTZ04.JJXS01E,DISP=SHR  
//DFSVSAMP DD *  
2048,4  
IOBF=(8192,4)  
/*
```



---

## 第 28 章 HISAM 再編成アンロード・ユーティリティー (DFSURUL0)

HISAM 再編成アンロード・ユーティリティー (DFSURUL0) は、HISAM データベースをアンロードして再編成し、HISAM データベース内に 2 次索引を作成するために使用します。

HISAM 再編成アンロード・ユーティリティーは、次に挙げる用途のために使用します。

- HISAM データベースをアンロードします。
- 再編成された出力を作成します。これは、データベース・リカバリー・ユーティリティーまたは HISAM 再編成再ロード・ユーティリティー (DFSURRL0) への入力として使用できます。
- 接頭部解決ユーティリティー (DFSURG10) が作成する索引作業データ・セットを、HISAM 再編成再ロード・ユーティリティー (DFSURRL0) が使用できるようにフォーマット設定します。HISAM 再編成再ロード・ユーティリティーは、このデータ・セットを使用して、副次索引を作成したり、共用副次索引があれば、これと索引作業データ・セットのレコードをマージしたりします。

3380 以外の装置では、DFSURUL0 は出力を出力装置のブロック・サイズにブロック化します。装置が 3380 である場合、論理レコード長が 23 KB 以下であれば、ブロック・サイズとして 23 KB が使用されます。論理レコード長が 23 KB より大きい場合、ブロック・サイズは、32 KB (端数切り捨てによって論理レコード長の偶数倍数に丸めた値) となります。ブロック化因数は実行時に判断されるので、すべての出力ボリュームに IBM 標準ラベルを使用する必要があります。

新しい DBD が新しいサイズで生成されている場合は、まず新しい DBD を使用してデータベースをアンロードします。次に、DFSMS アクセス方式サービス・プログラムを実行して、旧データ・セットを削除し、新しいブロック・サイズと論理レコード長 (厳密に言えば、VSAM では論理的に同等のもの、すなわち、制御インターバル・サイズと LRECL) を含む新しいデータ・セットを定義してください。その後、HISAM 再編成再ロード・ユーティリティー (DFSURRL0) を使用して、データベースを再ロードできます。DBD とは異なる制御インターバルまたはレコード・サイズを指定する場合は、ユーティリティー制御ステートメントの 50 から 80 桁目に CHNG=CARD をコーディングし、CHANGE 制御ステートメントを使用して新しいサイズを指定します。その他の場合は、このフィールドに CHNG=DBD を使用します。

ユーティリティー制御機能で、このユーティリティーの機能を実行することもできます。

以下の図は、このユーティリティーのフロー・ダイアグラムです。

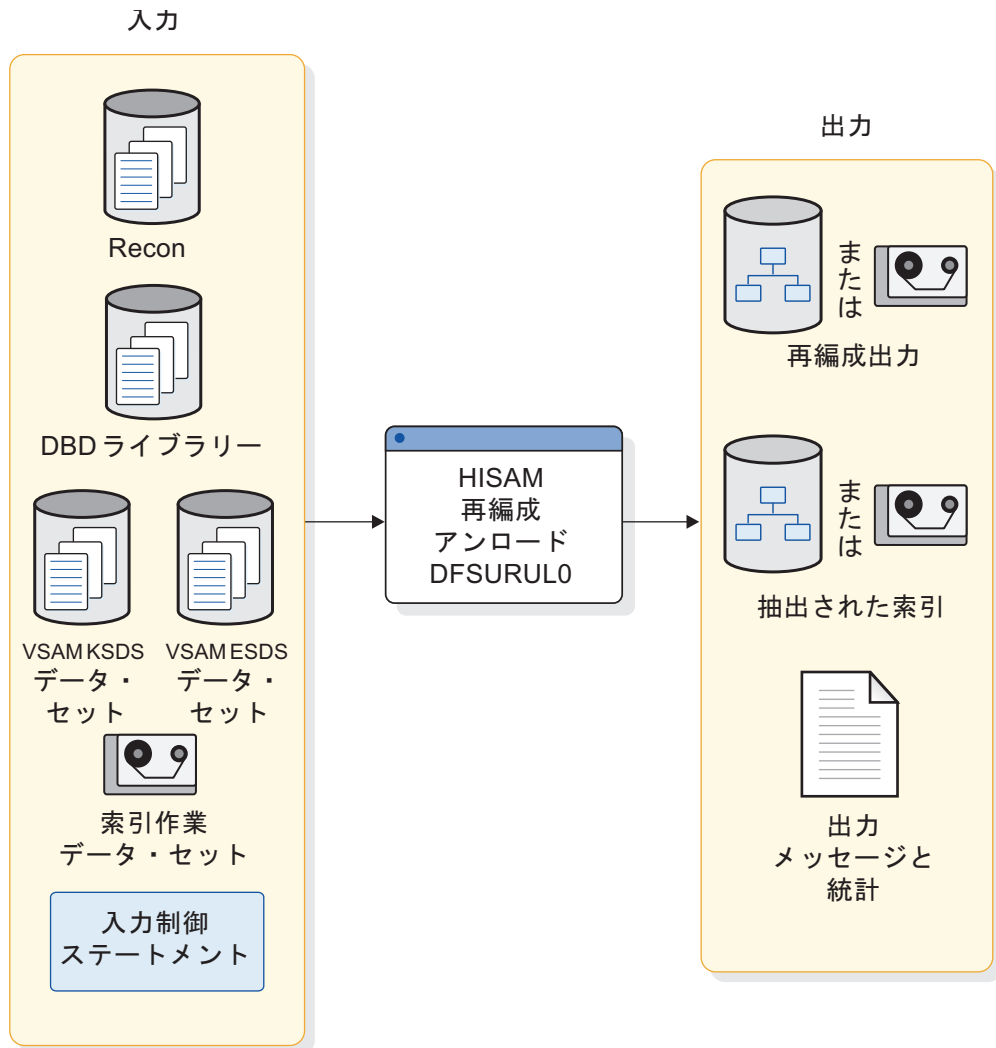


図 73. HISAM 再編成アンロード・ユーティリティ (DFSURRL0)

サブセクション:

- 『制約事項』
- 393 ページの『前提条件』
- 393 ページの『要件』
- 394 ページの『推奨事項』
- 394 ページの『入出力』
- 398 ページの『JCL 指定』
- 400 ページの『戻りコード』

### 制約事項

HISAM 再編成アンロード・ユーティリティには、次のような使用上の制約があります。

- このユーティリティは、PSINDEX データベースをアンロードできません。PSINDEX データベースをアンロードするには、HD 再編成アンロード・ユーティリティ (DFSURGU0) を使用します。

- HISAM 再編成アンロード・ユーティリティは、データベース編成に対して構造上あるいはその他の変更を行うことはできません。これらの目的のためには、HD 再編成アンロード・ユーティリティ (DFSURGU0) と HD 再編成再ロード・ユーティリティ (DFSURGL0) を使用してください。
- HISAM データベースに直接アドレス論理親ポインターを持つ論理子セグメントが含まれているときは、このユーティリティはそのデータベースをアンロードできません。
- このユーティリティは、SHISAM データベースをアンロードできません。
- このユーティリティは、DBRC に登録されていない別名を持つ共用副次索引のアンロードまたは再ロードには使用しないでください。 使用した場合、IMS は、PSB で参照されたデータベースが DBRC に登録されていないことを示すメッセージ DFS194W を出します。その後 IMS は、ユーザーがサブシステムに対して排他使用を指定したときと同じように該当データベースを扱います。
- 順次バッファリングはこのユーティリティをサポートしません。
- DFSURUL0 ユーティリティは、全機能非 HALDB データベースのみで機能します。
- データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。
- 高速機能の副次索引データベースを DFSURUL0 に入力することはできません。高速機能の副次索引データベースを再編成するには、IDCAMS (データ操作ユーティリティ) の REPRO コマンドを発行してください。高速機能の副次索引データベースのリカバリーには、IMS ベンダー製品に含まれる、データベース・リカバリー・ユーティリティ (DFSURDB0) または索引ビルダー・ツールのいずれかを使用してください。

## 前提条件

現在、DFSURUL0 ユーティリティに対して文書化された前提条件はありません。

## 要件

DFSURUL0 ユーティリティの要件は次のとおりです。

- このユーティリティをリカバリーの目的に使用する場合は、HISAM 再編成再ロード・ユーティリティ (DFSURRL0) を使用して、データベースに変更を適用する前にデータベースを再ロードする必要があります。変更を適用する前に、データベースを即時に再ロードしないと、セグメントは別の位置に再ロードされてしまいます。アンロードから再ロードまでの間に作成されたログは元の位置を参照しているため、リカバリーは不可能となります。
- データベースで書き込みエラーが起こり、そのデータベースがリカバリーされていない場合、このユーティリティを実行する前にリカバリーを行う必要があります。データベースが DBRC に登録されている場合に、このユーティリティが DBRC を使用すると、DBRC は、書き込みエラーが起こってリカバリーがまだ行われていないことを認識し、このユーティリティの許可要求を拒否します。

- 新しい DBD は、旧 DBD と同じ名前であればなりません。旧 DBD と新 DBD は、別々の 2 つのライブラリーが使用され、アンロード時と再ロード時に該当するほうのライブラリーが参照されるのであれば、システムに同時に置いておくことができます。

## 推奨事項

現在、DFSURULO ユーティリティーに対して文書化された推奨事項はありません。

## 入出力

DFSURULO ユーティリティーへの 1 次入力、再編成される VSAM KSDS データ・セットまたは ESDS データ・セットです。DFSURULO ユーティリティーの 1 次出力は、再編成された VSAM KSDS データ・セットまたは ESDS アンロード・データ・セットです。

DFSURULO ユーティリティーの入出力は、以下の表に示すとおりです。

表 19. DFSURULO ユーティリティーが使用するデータ・セット

入力	出力
RECON	出力メッセージと統計
DBD ライブラリー	抽出された索引
VSAM KSDS データ・セット	再編成出力
VSAM ESDS データ・セット	
索引作業データ・セット	
入力制御ステートメント	

## 報告書の出力例

HISAM 再編成アンロード・ユーティリティーは、データベースの内容に関するメッセージと統計をデータ・セット・グループ別に出力します。さらに、このユーティリティーは監査証跡も出力します。以下の図は、このユーティリティーが作成した出力メッセージと統計の例です。

### H I E R A R C H I C A L I N D E X E D S E Q U E N T I A L D A T A B A S E R E O R G A N I Z A T I O N U N L O A D D A T A S E T G R O U P S T A T I S T I C S

```

DATA BASE - DIVNTZ04
  PRIMARY DD- DBHVSAM1
  OVERFLOW DD- DBHVSAM2
PRIMARY ROOTS          OVERFLOW ROOTS          OVERFLOW DEPENDENTS
IN      OUT  DELETED    IN      DELETED    IN      OUT
3        3    0          0        0          2        2
TOTAL NUMBER OF RECORDS OUT =10
COPY 1 ON VOLUME(S)- USER02
ROOT OVERFLOW CHAINS (#)  DEPENDENT OVERFLOW CHAINS (#)  ROOTS W/OUT OVERFLOW CHAINS (BYTES)
NO.  LONGEST  SHORTEST  AVG  NO.  LONGEST  SHORTEST  AVG  NO.  LONGEST  SHORTEST  AVG
0    0         0         0.00  2    1         1         1.00  0    0         0         0.00
          S E G M E N T   L E V E L   S T A T I S T I C S
MAXIMUM  AVERAGE  MAXIMUM  AVERAGE  SEGMENT  SEGMENT  TOTAL SEGMENTS  AVERAGE COUNT PER
TWINS    TWINS    CHILDREN CHILDREN  NAME     LEVEL   BY SEGMENT TYPE  DATA BASE RECORD
1         1.00    12       8.00    J1       1       3                 1.00
2         1.00    2        0.66    J2       2       3                 1.00
1         0.33    1        1.00    J3       3       1                 0.33
1         1.00    0        0.00    J4       4       1                 0.33
2         1.33    3        1.75    J5       2       4                 1.33

```

1	1.00	2	0.75	J6	3	4	1.33
1	0.50	1	0.50	J7	4	2	0.66
1	0.50	0	0.00	J8	5	1	0.33
2	1.00	3	1.66	J9	2	3	1.00
1	1.00	0	0.00	J10	3	3	1.00
0	0.00	0	0.00	J11	4	0	0.00
2	0.66	0	0.00	J7P	3	2	0.66
0	0.00	0	0.00	J12	2	0	0.00
0	0.00	0	0.00	J13	3	0	0.00
0	0.00	0	0.00	J14	4	0	0.00
0	0.00	0	0.00	J13X	3	0	0.00

TOTAL SEGMENTS IN DATA SET GROUP=27    AVG DATA SET GROUP RECORD LENGTH=203 BYTES

この実行でいずれかのオプションが選択されていると、さまざまなメッセージが生成され、ページ見出しの直後に示されます。

データ・セット・グループ別のメッセージに続いて、「DATA SET GROUP STATISTICS」という見出しが示されます。

#### DATA BASE

データベース名

#### PRIMARY DD

データ・セット・グループの VSAM KSDS DD 名

#### OVERFLOW DD

データ・セット・グループの VSAM ESDS DD 名

#### PRIMARY ROOTS

VSAM KSDS 中のルート・レコードに関する統計:

**IN**    読み取られた旧 VSAM KSDS ルートの数

**OUT**   書き込まれた新 VSAM KSDS ルートの数

#### DELETED

削除された旧 VSAM KSDS ルートの数

#### OVERFLOW ROOTS

VSAM ESDS 中の旧ルートに関する統計:

**IN**    読み取られた旧 VSAM ESDS ルートの数

#### DELETED

削除された旧 VSAM ESDS ルートの数

#### OVERFLOW DEPENDENTS

VSAM ESDS 中の従属レコードに関する統計:

**IN**    読み取られた VSAM ESDS 中の旧従属レコードの数

**OUT**   書き込まれた VSAM ESDS 中の新従属レコードの数

#### TOTAL NUMBER OF RECORDS OUT

書き込まれた VSAM KSDS ルートと VSAM ESDS 従属の両方のレコードの数。この総計には、少なくとも 1 つのヘッダー・レコードが含まれます。また、複数の統計レコードが含まれる場合もあります。すなわち、データ・セットの先頭にあつて、再ロード・プログラムのテーブル初期設定レコードとして使用された以上のレコードと、データ・セットの最後にあつて、HISAM 再ロー

ド・ユーティリティが、再ロードされた数とアンロードされた数を比較できるようにするために、セグメント・タイプ別にアンロードされた総計が取められている 1 つ以上のレコードです。

統計テーブル・レコードは、DBD 中の各セグメント・タイプ用の 20 バイトと、統計レコード全体を取めておくために必要な各 LRECL 用の 28 バイト・ヘッダーから成り立っています。統計レコードを取めておくために必要な LRECL の概数は、次の式を用いると求めることができます。(セグメント・タイプの数 × 20)/(LRECL)

求めた結果に 2 をかけると、書き込まれたレコードの総数に加えらる LRECL の概数が得られます。

#### **COPY 1 ON VOLUME(S) - volser1**

1 次出力データ・セットは正常に完了しています。ボリューム通し番号のリストは、使用されたボリュームとその使用順序を示しています。2 番目のコピーが要求され、その 2 番目のコピーが正常に完了していると、さらに「COPY 2 ON VOLUME(S) - volser2」というメッセージが示されます。

#### **ROOT OVERFLOW CHAINS (#)**

VSAM KSDS にあって、VSAM ESDS にあるルート・レコードにつながるオーバーフロー・チェーンを持つルート・レコードに関する統計:

**NO.** オーバーフロー・チェーンを持つルートの数

##### **LONGEST**

1 つの VSAM KSDS ルートからチェーンが切り離された VSAM ESDS ルートの最大数

##### **SHORTEST**

1 つの VSAM KSDS ルートからチェーンが切り離された VSAM ESDS の最小数 (非ゼロ)

##### **AVERAGE**

1 つの VSAM KSDS ルート (チェーンを持つルートの 1 つ) からチェーンが切り離された VSAM ESDS ルートの平均数

#### **DEPENDENT OVERFLOW CHAINS (#)**

VSAM ESDS に従属がある VSAM KSDS ルートに関する統計:

**NO.** VSAM ESDS 従属レコードを持つルートの数

##### **LONGEST**

1 つのルートからチェーンが切り離された VSAM ESDS 従属レコードの最大数

##### **SHORTEST**

1 つのルートからチェーンが切り離された VSAM ESDS 従属レコードの最小数 (非ゼロ)

##### **AVERAGE**

1 つのルート (従属を持つルートの 1 つ) からチェーンが切り離された VSAM ESDS 従属レコードの平均数

#### **ROOTS WITHOUT OVERFLOW CHAINS (BYTES)**

従属を持っていない VSAM KSDS ルートに関する統計:

**NO.** 従属チェーンを持っていないルートの数



### **LONGEST**

VSAM ESDS 従属レコードを持っていない最大データベース・レコード (バイト数)

### **SHORTEST**

VSAM ESDS 従属レコードを持っていない最小データベース・レコード (バイト数)

### **AVERAGE**

VSAM ESDS 従属レコードを持っていないルートの平均データベース・レコード長 (バイト数)

「SEGMENT LEVEL STATISTICS」の節には、以下のフィールドがあります。

### **MAXIMUM TWINS**

直接の親セグメントのもとに見付かった、このタイプのセグメントの最大数。ルート・レベルでは、この値は常に 1 です。

### **AVERAGE TWINS**

直接の親セグメントのもとに見付かった、このタイプのセグメントの平均数。この値は、小数点以下 2 桁まで示されます。

### **MAXIMUM CHILDREN**

指定の親のもとにある子セグメント (すべての従属レベルにあるもの) の最大数

### **AVERAGE CHILDREN**

指定の親のもとにある子セグメント (すべての従属レベルにあるもの) の平均数。この値は、小数点以下 2 桁まで示されます。階層パスの最下位にあるセグメントの場合は、このフィールド・タイプは値がゼロになっています。

### **SEGMENT NAME**

この統計行が適用されるセグメントの名前

### **SEGMENT LEVEL**

このセグメントがデータベースに置かれている階層レベル

### **TOTAL SEGMENTS BY SEGMENT TYPE**

このセグメント・タイプがデータベース全体に出現した回数。レベル 1 のセグメント・タイプのカウント・フィールドは、データベースに入っているデータベース・レコード (ルート・セグメント) の総数を示しています。

データベースが共用副次索引データベースの一部であるときは、セグメントの出現回数は、常に最初のセグメント名の下に示されます。他のセグメント名の出現回数はすべてゼロになっています。

### **AVERAGE COUNT PER DATABASE RECORD**

このセグメント・タイプが、指定のデータベース・レコード内に出現した回数の平均数。この値は、小数点以下 2 桁まで示されます。

このデータ・セット・グループの個々のセグメント・タイプ統計に続いて、次の 2 つのフィールドがあります。

### **TOTAL SEGMENTS IN DATA SET GROUP**

データ・セット・グループに含まれるセグメントの総数

## AVERAGE DATA SET GROUP RECORD LENGTH

データベース・レコードのうち、このデータ・セット・グループに保管されている部分の平均長 (バイト数)

## JCL 指定

DFSURUL0 ユーティリティは、標準の z/OS ジョブとして実行されます。DFSURUL0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含める必要があります。

次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

DFSURUL0 ユーティリティの EXEC ステートメントは、固有の形式を使用します。

EXEC ステートメントは次の形式になっていなければなりません。

```
PGM=DFSRR00,PARM='ULU,DFSURUL0'
```

SPIE、BUF、DBRC といった通常の IMS 定位置パラメーターは、PARM フィールドの中のプログラム名の後に置くことができます。

### DD ステートメント

DFSURUL0 ユーティリティは、必須 DD ステートメントとオプションの DD ステートメントを多数使用します。

DFSURUL0 ユーティリティは、次の DD ステートメントを使用します。

#### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。APF 無許可ライブラリーを IMS.SDFSRESL に連結させるため、STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

#### DFSRESLB DD

IMS SVC モジュールが入っている APF 許可ライブラリーを指します。

#### IMS DD

再編成するデータベースを記述する DBD が入っているライブラリーを定義します (DSN=IMS.DBDLIB,DISP=SHR)。このデータ・セットは、直接アクセス装置に置いておかなければなりません。

#### SYSPRINT DD

メッセージおよび統計出力データ・セットを定義します。このデータ・セットは、テープ、直接アクセス装置、またはプリンターに置いておくことも、出力ストリームに送ることもできます。このデータ・セットに関して指定されている

DCB パラメーターは、RECFM=FBA と LRECL=133 です。BLKSIZE は SYSPRINT DD ステートメントで指定し、133 の倍数になっていなければなりません。

#### **SYSIN DD**

入力制御ステートメント・データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。

#### **vsamksds DD**

再編成する VSAM KSDS を定義します。DD 名は、このデータ・セットを記述する DBD に入っている名前と同じものでなければなりません。また、この名前は、このジョブ・ステップの SYSIN データ・セットに入っているユーティリティー制御ステートメントにも指定されている必要があります。このタイプの DD ステートメントは、再編成する VSAM KSDS ごとに 1 つ存在していなければなりません。

この DD ステートメントは、HISAM データベースの 1 次データ・セットを表します。副次索引を作成する場合は、1 つ以上の DD ステートメントを指定する必要があります。その場合、各 DD ステートメントは、再編成する副次索引データ・セットを表します。

#### **vsamesds DD**

再編成する VSAM ESDS を定義します。DD 名は、このデータ・セットを記述する DBD に入っている名前と同じものでなければなりません。このタイプの DD ステートメントは、再編成する VSAM ESDS ごとに 1 つ存在していなければなりません。

#### **dataout1 DD**

再編成された出力データ・セットの最初のコピーを定義します。このタイプの DD ステートメントは、再編成する VSAM データ・セット・グループごとに 1 つ必要です。名前はどの名前でも構いませんが、関連付けられたユーティリティー制御ステートメントに指定されている必要があります。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておかなければなりません。

この出力データ・セットはイメージ・コピー・データ・セットとして使用できます。アンロード・データ・セットをイメージ・コピーとして DBRC で登録するには、HISAM 再ロード・ユーティリティーを使用してデータベースを再ロードしなければなりません。

#### **dataout2 DD**

再編成された出力データ・セットの 2 番目のコピーを定義します。このステートメントはオプションですが、出力のコピーが 2 つ要求された場合にのみ必要です。複数の出力コピーを指定すると、次のような利点があります。一方のコピーで永続エラーが発生しても、もう一方のボリュームでジョブが正常に終了するまで続けられるため、すべてを再実行する必要はありません。

dataout1 DD ステートメントの項で説明した要件は、dataout2 にも適用されません。

#### **indexwrkds DD**

接頭部解決プログラムから得られ、副次索引情報が入っている出力データ・セット DD 名 (DFSURIDX) を記述します。ユーティリティー制御ステートメント

がタイプ「X」のときは、このステートメントは必要です。その他の場合は、省略できます。DD 名は、制御ステートメントの 40 桁目から始まる名前と同じものでなければなりません。

#### **DFSEXTDS DD**

制御ステートメントで指定されたとおりに共用副次索引から分割され、アンロードされたレコードを書き込みます。DFSEXTDS DD はオプションですが、ユーティリティー制御ステートメントの 3 桁目に「E」の指定がある場合にのみ必要です。DCB 属性は、出力装置タイプと使用される VSAM LRECL に基づいて、動的に判断されます。標準ラベルを使用する必要があります。

#### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。この DD ステートメントは必須です。

このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、入力ストリームに入れて送ることもできます。

#### **SYSUDUMP または SYSABEND DD**

ダンプ・データ・セットを定義します。これらの DD ステートメントはオプションです。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

#### **RECON1 DD**

最初のデータベース・リカバリー管理 (DBRC) RECON データ・セットを定義します。この RECON1 データ・セットは、制御領域が使用しているものと同じ RECON1 データ・セットでなければなりません。

#### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。この RECON2 データ・セットは、制御領域が使用しているものと同じ RECON2 データ・セットでなければなりません。

#### **RECON3 DD**

RECON1 または RECON2 にエラーが見つかったとき DBRC が使用する、オプションの RECON データ・セットを定義します。この RECON3 データ・セットは、制御領域が使用しているものと同じ RECON3 データ・セットでなければなりません。

制約事項: 動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

### **戻りコード**

HISAM 再編成アンロード・ユーティリティーは戻りコードを返しますが、エラーが起こった場合は、戻りコードより前にプログラム実行の結果を詳しく説明した番号付きメッセージを SYSPRINT データ・セットへ送ります。


戻りコードの意味は次のとおりです。

コード 意味

- 0 要求されたすべての操作が正常に完了した
- 4 1 つ以上の操作が正常に完了しなかった

- 8 重大エラーが起こったためジョブが終了した
- 12 戻りコード 4 と 8 の組み合わせが起こった
- 16 DD 名 SYSIN をオープンできなかった


関連概念:


 IMS バッファ・プール (システム定義)

関連資料:

437 ページの『第 31 章 ユーティリティー制御機能 (DFSUCF00)』

349 ページの『第 25 章 HD 再編成アンロード・ユーティリティー (DFSURGU0)』

 DBBBATCH プロシージャ (システム定義)

 DLIBATCH プロシージャ (システム定義)

## DFSURUL0 ユーティリティーの制御ステートメント

DFSURUL0 ユーティリティーには、その処理オプションを定義するために、CHANGE および OPTIONS の 2 つの制御ステートメントがあります。

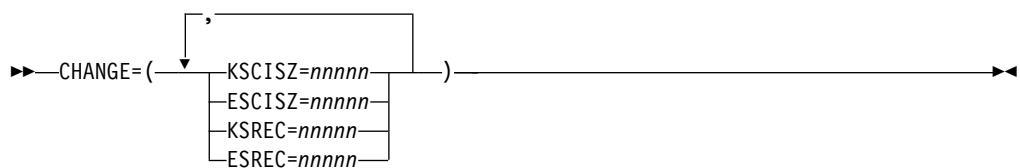
位置 説明

- 1 「R」か「X」のいずれかを入れなければなりません。
  - R** HISAM 再編成アンロード・ユーティリティー制御ステートメントであることを示します。
  - X** 副次索引再編成制御ステートメントであることを示します。

デフォルトはありません。この桁位置がブランクのままになっていると、エラー・メッセージが出されます。
- 2 必要とする出力コピー数に応じて、1 または 2 を入れなければなりません。デフォルトはありません。この桁位置がブランクのままになっていると、エラー・メッセージが出されます。
- 3 「M」、「E」、「R」のいずれかを入れるか、副次索引再編成 (1 桁目が「X」) 制御ステートメントのときはブランクにしなければなりません。この桁位置の値に何も指定しない場合、デフォルトは M です。
  - M** データベースの初期ロード時または再編成時に (接頭部解決ユーティリティーを使用して) 作成された索引作業データ・セットを、既存の副次索引にマージすることを示し、データ・セットがなければ、副次索引を作成するために使用されます。
  - E** (このステートメントの 49 桁目の定数で定義された)副次索引を、共用索引データベースから、あるいは索引作業データ・セット (接頭部解決ユーティリティーから得たもの) から抽出することを示します。
  - R** 副次索引の中の、このステートメントの 49 桁目の定数と一致するセグメントを、作業データ・セットで見付かった一致するセグメントで置き換えることを示します。置き換えられる副次索引セグメントを保管しておく必要があるときは、置き換えを行う前に抽出機能を実行してください。

- 4-12 再編成する VSAM データ・セット・グループの DD 名を含んでいる DBD の名前を入れなければなりません。
- 13-21 再編成する VSAM データ・セットの KSDS DD 名を入れなければなりません。DD 名とは、HISAM データベースの 1 次データ・セット名と副次索引データベースの副次索引 DD 名です。この DD 名を、参照される DBD ステートメントの中に指定する必要があり、対応する DD ステートメントを指定する必要があります。
- 22-30 1 次出力データ・セットの DD 名を入れなければなりません。対応する DD ステートメントを指定する必要があります。
- 31-39 再編成された出力データ・セットの 2 番目のコピーの DD 名を入れなければなりません。このフィールドに DD 名が入っているときは、対応する DD ステートメントを指定する必要があります。2 桁目が 1 のときは、このフィールドはブランクのままにしておきます。
- 40-48 この制御ステートメントがタイプ「X」のときは、副次索引作業データ・セットの DD 名を入れなければなりません。
- 49 このステートメントの 3 桁目に「E」または「R」が指定されたときは、共用副次索引の DBD 生成の中で指定されている 1 バイトの定数を入れなければなりません。
- 50-80 コメントを入れることも、制御インターバル (CI) またはレコード・サイズの変更を指定することもできます。CI またはレコード・サイズの変更にどの値を使用するかを指定するには、CHNG=DBD または CHNG=CARD とコーディングします。この場合、DBD を指定すると、アンロード・ユーティリティーは KSDS と ESDS CI およびレコード・サイズに DBD 値を使用し、CARD を指定すると、アンロード・ユーティリティーは 1 桁目から CHANGE で始まる制御ステートメントで指定された値を使用します。

## CHANGE ステートメント



これは、オプションの制御ステートメントです。このステートメントを使用する場合は、キーワードを少なくとも 1 つ指定しなければなりません。

### CHANGE=

CHANGE 制御ステートメントであることを示します。

#### KSCISZ

新しい KSDS CI サイズをバイト数で指定します。サイズは 512 の倍数で指定してください。

#### ESCISZ

新しい ESDS CI サイズをバイト数で指定します。サイズは 512 の倍数で指定してください。

### KSREC

新しい KSDS レコード・サイズを指定します。

### ESREC

新しい ESDS レコード・サイズを指定します。

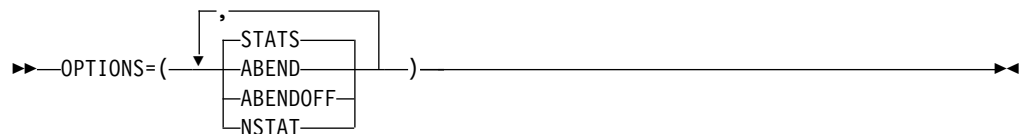
*nnnnn*

必要であれば、先行ゼロを含めて 5 桁の 10 進数で指定します。最大値は 32767 です。

例: この例では、CHANGE ステートメントは、新しい KSDS CI サイズと新しい ESDS CI サイズを指定するために使用されています。

CHANGE=(KSCISZ=01024,ESCISZ=02048)

## OPTIONS ステートメント



これは、オプションの制御ステートメントです。

### OPTIONS=

OPTIONS 制御ステートメントであることを示します。

### STATS

統計を SYSPRINT データ・セットへ出力するとともに、HISAM 再編成再ロード・ユーティリティへ送ります。STATS はデフォルトです。

### ABEND

ABEND 機能をオンにします。実行を終了させるような条件が起こると、ABEND U0359 を出して実行を終了させます。SYSABEND または SYSUDUMP DD ステートメントが指定されていると、ダンプが印刷されません。

### ABENDOFF

ABEND 機能をオフにします。

例外: ABENDOFF は初期デフォルトです。ただし、その前に ABEND が指定されていると、ABEND は ABENDOFF が指定されるまで有効です。

### NSTAT

統計出力が生成されないことを指定します。このパラメーターを使用すると、HISAM 再編成再ロード・ユーティリティも統計出力を無視します。

---

## DFSURUL0 ユーティリティの例

ここでの例は、DFSURUL0 ユーティリティの使用法を示しています。

この節にある例では、桁合わせを容易にするために、SYSIN ステートメントの上に次のようなコメント行が示されています。

```
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7
```

このコメント行は単なる参考用であり、したがって必須ではありません。

この節の例で、使用する DBRC で動的割り振りを行わない場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

図 74. 動的割り振りなしで DBRC を使用するための DD ステートメント

```
//RECON1 DD DSNAME=RECON1,DISP=SHR
//RECON2 DD DSNAME=RECON2,DISP=SHR
//RECON3 DD DSNAME=RECON3,DISP=SHR
```

サブセクション:

- 『索引作業データ・セット』
- 405 ページの『索引レコードの抽出、置き換え、およびマージ』
- 405 ページの『副次索引の作成とマージ、定数の置き換え、および HISAM 再編成再ロード・ユーティリティーで使用するための準備』

## 索引作業データ・セット

この例では、接頭部解決ユーティリティーから渡された索引作業データ・セットは、次のものを作成するために使用されます。

- 共用副次索引データベースに置かれている 2 つの副次索引をアンロードしたものの
- 3 番目の副次索引をアンロードしたもの

```
//INDREOR JOB 1,1,MSGLEVEL=1
//*
//STEP1 EXEC PGM=DFSRRCO0,
//          PARM='ULU,DFSURUL0,,,1,,,,,,,,,N,N'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1330
//DBIDX1 DD DSN=IMS.INDX1,DISP=SHR
//DXOUT1 DD DSN=IMS.DBXOUT1,DISP=(MOD,KEEP),
//          UNIT=TAPE,VOL=SER=DXOUT1,LABEL=(,SL)
//DBIDX2 DD DSN=IMS.INDX2,DISP=SHR
//DXOUT2 DD DSN=IMS.DBXOUT2,DISP=(,KEEP),
//          UNIT=TAPE,VOL=SER=DXOUT2,LABEL=(,SL)
//NDXDS DD DSN=IMS.NDXWDS,DISP=(OLD,DELETE)
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7
//SYSIN DD *
OPTIONS=(NSTAT,ABEND)
X1MDIX1DB01 DBIDX1 DXOUT1 NDXDS CREATE NEW SECONDARY
INDEX
X1MDIX1DB02 DBIDX1 DXOUT1 NDXDS ADD SECONDARY INDEX RECS
TO EXISTING ONE ABOVE
X1MDIX2DB01 DBIDX2 DXOUT2 NDXDS RECORDS FROM SAME WORK
DS PUT INTO DIFFERENT
データベース
/*
```



この例の出力は HISAM 再編成再ロード・ユーティリティーへの入力として使用され、副次索引が作成されます。副次索引データベースを作成するには、アクセス方式サービス・プログラム・ユーティリティーが実行されている必要があります。

DBRC および IRLM は、EXEC PARM ステートメントでオフになっています。OPTIONS ステートメントは、統計が必要でないこと、重大メッセージが出されたら ABEND U0359 を出すことを指定しています。

### 索引レコードの抽出、置き換え、およびマージ

この例では、共用索引の DBDGEN 中に定数「B」が定義されている索引レコード群が抽出され、置き換えられ、マージされます。オプションは操作と操作の間で変更され、最初の操作では統計をとり、2 番目の操作では統計をとらず、3 番目の操作では再度統計をとるようにしています。ABEND オプションは、2 番目の OPTIONS ステートメントで変更されています。

```
//IDEREORG JOB 1,1,MSGLEVEL=1
/*
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSURUL0'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1330
//DBIDX2 DD DSN=IMS.INDX2,DISP=SHR
//DBIDX1 DD DSN=IMS.INDX1,DISP=SHR
//DBXOUT1 DD DSN=IMS.DBXOUT1,DISP=(MOD,KEEP),
// UNIT=TAPE,VOL=SER=DXOUT1,LABEL(,SL)
//DFSEXTDS DD DSN=IMS.EXTDS1,DISP=(MOD,KEEP),
// UNIT=TAPE,VOL=SER=DEXTDS,LABEL(,SL)
//NDXWDS1 DD DSN=IMS.XWDS1,DISP=(OLD,PASS)
//NDXWDS2 DD DSN=IMS.XWDS2,DISP=(OLD,PASS)
//NDXWDS3 DD DSN=IMS.XWDS3,DISP=(OLD,PASS)
/* +-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7
//SYSIN DD *
OPTIONS=(STATS,ABEND)
X1EDIX3DB01 DBIDX1 DBXOUT1 NDXWDS1 BUNLOAD INDEX EXTRACT
THOSE MARKED WITH
CONSTANT B INCLUDING
THOSE ON WORK DATA SET

OPTIONS=(ABENDOFF,NSTAT)
X1RDIX4DB01 DBIDX2 DBXOUT1 NDXWDS2 BUNLOAD INDEX REPLACING
THOSE HAVING CONSTANT B
WITH THOSE FROM INDEX
WORK DATA SET

OPTIONS=(STATS,ABEND)
X1MDIX4DB02 DBIDX2 DBXOUT1 NDXWDS3 MERGE ALL RECS OF WORK
DATA SET AND CREATE A
SHARED INDEX

/*
//DFSVSAMP DD *
1024,10
/*
```

### 副次索引の作成とマージ、定数の置き換え、および HISAM 再編成再ロード・ユーティリティーで使用するための準備

この例では、JCL は次のことを行うためのものです。

- 2 つの別々の副次索引の作成に使用される 2 つの出力データ・セットを作成する。

- 接頭部解決ユーティリティにより作成された索引作業データ・セットを既存の副次索引 (定数を 1 つだけ持つ共用副次索引として以前に作成されていたもの) とマージすることにより、2 つの副次索引をマージする。
- 共用副次索引に既に入っている 1 つの定数を、索引作業データ・セットに入っている定数で置き換える。
- 共用索引から定数を抜き出して、それを、もう 1 つの別の副次索引を作成するために HISAM 再編成再ロード・ユーティリティが使用できる形式にする。

各操作は、OPTIONS ステートメントで区切られています。ここには JCL は一度しか示されていませんが、各操作は別々の実行として扱われます。すべての操作を 1 回の実行で行うには、DBOUT1 と DBOUT2 に DISP=MOD と指定しておく必要があります。

```
//INDXREOR JOB 1,1,MSGLEVEL=1
//*
//STEP1 EXEC PGM=DFSRRCO0,PARM='ULU,DFSURULO'
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//IMS DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A,DCB=BLKSIZE=1330
//DBIX1A DD DSN=IMS.DBIX1A,DISP=OLD
//DBIX2B DD DSN=IMS.DBIX2B,DISP=OLD
//DBOUT1 DD DSN=IMS.DBOUT1A,DISP=(,KEEP),
// UNIT=SYSDA,SPACE=(CYL,(2,1))
//DBOUT2 DD DSN=IMS.DBOUT2A,DISP=(,KEEP),
// VOL=SER=DBOUT2,LABEL=(,SL)
//INDXWDS1 DD DSN=IMS.INDXWDS1,DISP=(OLD,DELETE)
//INDXWDS2 DD DSN=IMS.INDXWDS2,DISP=(OLD,DELETE)
//DFSEXTDS DD DSN=IMS.EXTSD1,DISP=(,KEEP),
// VOL=SER=EXTSD1,LABEL=(,SL)
/* +----1-----+----2-----+----3-----+----4-----+----5-----+----6-----+----7
//SYSIN DD *
X1MDI32XDB1 DBIX1A DBOUT1 INDXWDS1 DBIX1A TO BE CREATED
X1MDI33XDB2 DBIX2B DBOUT2 INDXWDS2 DBIX2B TO BE CREATED
OPTIONS=(STATS,ABEND)
X1MDI32XDB1 DBIX1A DBOUT1 INDXWDS2 MERGE ONE EXISTING
WITH NEW OPTIONS=(NSTAT,ABENDOFF)
X2RDI32XDB1 DBIX1A DBOUT1 DBOUT2 INDXWDS1 AREPLACE ANY EXISTING
A'S WITH ONES FROM
WORK DATA SET OPTIONS=STATS
X1EDI32XDB1 DBIX1A DBOUT1 INDXWSD2 BTAKE B CONSTANTS FROM
SHARED INDX AND/OR
WORK DATA SET AND PUT
OUT TO DFSEXTDS DD
STATEMENT

/*
//DFSVSAMP DD *
1024,10
/*
```

**重要:** 「X」または「R」制御ステートメントを、共用索引 DBD に入っているすべての別名に対して指定する必要があります。いずれかの別名で「X」または「R」ステートメントが省略されている場合、その別名が表す実際のデータは、共用副次索引データベースが削除または再定義されたときに削除される可能性があります。

---

## 第 29 章 MSDB - DEDB 間変換ユーティリティ (DBFUCDB0)

MSDB - DEDB 間変換ユーティリティ (DBFUCDB0) は、既存の主記憶データベース (MSDB) を高速処理データベース (DEDB) に変換するために使用します。このユーティリティを使用すると、DEDB から MSDB へフォールバックすることもできます。このユーティリティは、z/OS バッチ領域で実行されるオフライン・ユーティリティです。

DBFUCDB0 ユーティリティは、IBM IMS Fast Path Basic Tools に付属の DEDB アンロード/再ロード・ユーティリティと連携します。ただし、同等の代替製品を使用することができます。別の製品を使用する場合、DBFUCDB0 ユーティリティとともに提供される変換出口ルーチンおよびフォールバック出口ルーチンを変更する必要があります。

サブセクション:

- 『制約事項』
- 『前提条件』
- 408 ページの『要件』
- 408 ページの『推奨事項』
- 408 ページの『入出力』
- 408 ページの『JCL 指定』
- 410 ページの『戻りコード』

### 制約事項

DBFUCDB0 ユーティリティには、以下のような使用上の制約があります。

- DEDB に変換できるのは、非端末関連 MSDB だけです。
- 変換できる MSDB は、DBFUCDB0 ユーティリティの実行 1 回につき 1 つだけです。
- 各 MSDB は、単一の DEDB 区域に変換されます。
- 各 MSDB セグメントは、レベル「01」のセグメントに変換されます。
- MSDB と DEDB DBD との間でフィールドの整合性検査は行われません。
- アンロードされた形式で DBFUCDB0 ユーティリティによって出力された RAP RBA 値について、適切な昇順になっているかどうかの検査は行われません。
- DBFUCDB0 ユーティリティは、高速機能データベース (DEDB) のみで機能します。
- データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

### 前提条件

現在、DBFUCDB0 ユーティリティに対して文書化された前提条件はありません。

## 要件

IMS Fast Path Basic Tools とともに提供される DEDB アンロード/再ロード・ユーティリティ以外のアンロードおよび再ロードの製品を使用する場合、DBFUCDB0 ユーティリティの変換出口ルーチン (DBFUCDX0) およびフォールバック出口ルーチン (DBFUCDX1) がそのアンロードおよび再ロードの製品と連携するように、これらの出口ルーチンを変更する必要があります。

## 推奨事項

現在、DBFUCDB0 ユーティリティに対して文書化された推奨事項はありません。

## 入出力

DBFUCDB0 ユーティリティへの 1 次入力、非端末関連 MSDB です。  
DBFUCDB0 ユーティリティの 1 次出力は、1 つの区域が入っている DEDB データベースです。DBFUCDB0 ユーティリティは、要約報告書も生成します。

以下の表は、DBFUCDB0 ユーティリティへの入力およびこのユーティリティからの出力を示したものです。

表 20. DBFUCDB0 ユーティリティの入出力

入力	出力
非端末関連 MSDB	1 つの区域が入っている DEDB
制御ステートメント	要約報告書

## 報告書の出力例

このユーティリティは、実行を完了すると、要約報告書を生成します。報告書は SYSPRINT データ・セットに送られます。以下の図は、報告書の例を示しています。

```
CONVERSION UTILITY SUMMARY REPORT
TYPE OF CONVERSION = CONVERT/FALLBACK
TOTAL NUMBER OF RECORDS PROCESSED FOR AREA XXXXXXXX IS NNNN
XXXXXXXX = the name of the area that was processed
NNNN     = the total number of segments processed for the area
```

## JCL 指定

DBFUCDB0 ユーティリティは、標準の z/OS ジョブとして実行されます。  
DBFUCDB0 ユーティリティの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。1 つ以上のユーティリティ制御ステートメントを JCL ステートメントに含めることが必須であり可能です。

### EXEC ステートメント

DBFUCDB0 ユーティリティの EXEC ステートメントは、特定の形式を使用して JCL に指定します。

以下の形式を使用します。

//STEPNAME EXEC PGM=DBFUCDB0

#### **DD** ステートメント

DBFUCDB0 ユーティリティは、必須の DD ステートメントおよびオプションの DD ステートメントを数多く使用します。

DBFUCDB0 ユーティリティの DD ステートメントには、以下のものがあります。

#### **STEPLIB DD**

IMS.SDFSRESL を定義します。ここには、IMS 中核と必要なアクション・モジュールが入っています。

#### **MACBLIB DD**

**MSDB** IMS ACB ライブラリーを定義します。このライブラリーでは、変換するデータベースの DMB により、このデータベースは MSDB と指定されます。

#### **DACBLIB DD**

**DEDB** IMS ACB ライブラリーを定義します。このライブラリーでは、変換するデータベースの DMB により、このデータベースは DEDB と指定されます。

#### **MSDBINIT DD**

新しい DEDB 区域にロードされる MSDB セグメントが入っている MSDBINIT データ・セットを定義します。このデータ・セットは、CONVERT オプションが指定されている場合に入力として使用されます。この DD ステートメントが必要なのは、CONVERT オプションの場合だけであり、FALLBACK オプションの場合は不要です。

#### **DURDBDFN DD**

再ロード・プロセッサ (IMS Fast Path Basic Tools の一部) によって使用される DMB のフォーマット済みコピーが入っているデータ・セットを定義します。

#### **MSDBLCHG DD**

MSDBINIT レコード形式で変更レコードが格納されるデータ・セットを定義します。このデータ・セットは、MSDBINIT データ・セットを作成するために、MSDB 保守ユーティリティへの入力として使用されます (MSDBINIT データ・セットは、フォールバックが実行されているときに MSDB をストレージに再ロードするのに使用されます)。MSDBLCHG は、FALLBACK オプションが指定されている場合の出力データ・セットです。この DD ステートメントが必要なのは、FALLBACK オプションの場合だけであり、CONVERT オプションの場合は不要です。

#### **areaname DD**

再ロード・ユーティリティが新しい DEDB 区域をロードするために使用する MSDB 変換データが入っているデータ・セットを定義します。DD ステートメントは、DBDGEN 入力制御ステートメント内の AREA 名ごとに 1 つ必要です。JCL ステートメント上の DD 名は、入力制御ステートメント内の AREA 名と同じものでなければなりません。

### **SYSIN DD**

入力制御ステートメント・データ・セットを定義します。

### **SYSOUT DD**

DFSORT プログラムからのメッセージに使用される出力メッセージ・データ・セットを定義します。

### **SORTWKnn DD**

DFSORT プログラムが使用する中間記憶データ・セットを定義します。

### **SYSPRINT DD**

このユーティリティーのエラー・メッセージ、戻りコード、および要約報告書を入れる出力データ・セットを定義します。

## 戻りコード

終了時に DBFUCDB0 ユーティリティーによって、以下の戻りコードの 1 つ以上が生成されます。

コード 意味

- 1 TYPE ステートメントが脱落しているか、あるいは無効である
- 2 データベース・ステートメントが無効である
- 3 MSDB ACB ライブラリー (MACBLIB) は、入力制御ステートメントに指定されたデータベースが MSDB でないことを示している
- 4 入力制御ステートメントに指定された MSDB は、非端末関連 MSDB ではない
- 5 DEDB ACB ライブラリー (DACBLIB) は、入力制御ステートメントに指定されたデータベースが DEDB でないことを示している
- 6 指定された MSDB= メンバーが、MSDB ACB ライブラリー (MACBLIB) で見付からなかった
- 7 指定された DEDB メンバーが、新しい DEDB ライブラリー (DACBLIB) で見付からなかった
- 8 ランダマイザー・モジュール xxxxxxxx のロード時にエラーが検出された
- 9 SYSIN DD ステートメントが脱落している
- 10 SYSPRINT DD ステートメントが脱落している
- 11 MSDBINIT DD ステートメントが脱落している
- 12 AREADCB DD ステートメントが脱落している
- 13 MACBLIB DD ステートメントが脱落している
- 14 DACBLIB DD ステートメントが脱落している
- 15 MSDBLCHG DD ステートメントが脱落している
- 16 AREAIN DD ステートメントが脱落している

関連資料:

 z/OS: SORTWKdd DD ステートメント

---

## DBFUCDB0 ユーティリティーの制御ステートメント

DBFUCDB0 ユーティリティーの制御ステートメントは、**CONVERT** および **FALLBACK** です。

ユーティリティー制御ステートメントについて以下に説明します。

### **TYPE=CONVERT|FALLBACK**

ユーティリティーの実行モードとして変換またはフォールバックを指示します。**TYPE** は、1 桁目から始めます。

#### **CONVERT**

ユーティリティーを変換モードにすることを指示します。MSDB は DEDB に変換されます。

#### **FALLBACK**

ユーティリティーをフォールバック・モードにすることを指示します。DEDB は MSDB に変換されます。

### **MSDB=xxxxxxx,DEDB=yyyyyyyy**

どのデータベースを処理するのかを指示します。変換ユーティリティーは、データベースを一度に 1 つ変換します。この制御ステートメントは、1 桁目から始めます。

このステートメントの形式は次のとおりです。

**MSDB=xxxxxxx,DEDB=yyyyyyyy**

**CONVERT** オプションの場合、MSDB パラメーターは、変換元 MSDB の名前です。DEDB パラメーターは、変換先 DEDB の名前です。DEDB 名と MSDB 名が同じである必要はありません。

**FALLBACK** オプションの場合、MSDB パラメーターは、フォールバック・プロシージャの変換先 MSDB の名前です。DEDB パラメーターは、フォールバックの変換元 DEDB です。DEDB 名と MSDB 名が同じである必要はありません。DBD 内の AREA パラメーターには、アンロードされる DEDB 内の区域の名前が入ります。

---

## DBFUCDB0 ユーティリティーの例

これらの例は、変換とフォールバックの両方でこのユーティリティーを実行するために必要な JCL を示しています。

サブセクション:

- 『MSDB の DEDB への変換』
- 413 ページの『DEDB を MSDB に戻す方法』

### **MSDB の DEDB への変換**

変換処理の最初のステップでは、MSDBDUMP または MSDBCPx (チェックポイント) データ・セットのいずれか一方から MSDBINIT データ・セットを作成します。これは、MSDB ダンプ・リカバリー・ユーティリティーを使用して行われます。この例では、SAVINGS という名前の MSDB が入っている新しい MSDBINIT

データ・セットが作成されます。MSDB ダンプ・リカバリー・ユーティリティが完了すれば、MSDB - DEDB 間変換ユーティリティを実行することができます。

MSDB - DEDB 間変換ユーティリティは、MSDB セグメントのための入力ソースとして MSDB ダンプ・リカバリー・ユーティリティにより作成される MSDBINIT データ・セットを使用します。

SYSIN 制御ステートメントは、このユーティリティの実行モードが CONVERT であることを示します。データベース制御ステートメントは、SYSIN ストリームにも含まれています。MSDB SAVINGS が SAVINGS という名前の DEDB に変換されています。DEDB SAVINGS には SAVE1 という名前の区域があり、これには、MSDBINIT からのセグメントがロードされます。

MSDB.ACBLIB データ・セットには、データベース SAVINGS が MSDB であることを指定する DMB が入っています。DEDB.ACBLIB データ・セットには、データベース SAVINGS が DEDB であることを指定する DMB が入っています。

DBT.FORMAT データ・セットには、変換により生じた出力が入ります。この出力には、MSDB SAVINGS のセグメントごとに DBT アンロード形式のレコードが入ります。

次のステップでは、DFSORT プログラムまたは同等のプログラムを実行します。SYSIN ストリーム内の入力制御ステートメントは、DBT.FORMAT データ・セットのレコードを RAP RBA 値の昇順によりソートします。このステップが完了すると、DBT.FORMAT データ・セットを削除できます。

DBT.FORMAT.SAVE1 データ・セットには、ソートにより生じた出力が入ります。このデータ・セットは、DEDB アンロード/再ロード・ユーティリティ (IBM IMS Fast Path Basic Tools for z/OS の一部) によって使用できるようになります。

出力データ・セット DBT.FORMAT.SAVE1 は、新しい DEDB 区域をロードするために DEDB アンロード/再ロード・ユーティリティ (IMS Fast Path Basic Tools の一部) への入力として使用されます。この新しい DEDB 区域が SAVE1 です。

以下の図は、MSDB を DEDB に変換する例を示しています。

```
//RECOVERY EXEC PGM=DBFDBDR0
//* EXECUTE THE DUMP RECOVERY UTILITY USING MSDBDUMP AS INPUT
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//MSDBINIT DD DSN=MSDBINIT,DISP=(,KEEP,DELETE),
//          UNIT=SYSDA,VOL=SER=IMS333,
//          DCB=(BLKSIZE=13030,RECFM=VBT,LRECL=13026)
//MSDBDUMP DD DSN=IMS.MSDBDUMP,DISP=OLD
//MSDBCTL DD *
//          UNLOAD DBN=(SAVINGS)
//
//CONVERT EXEC PGM=DBFUCDB0
//* EXECUTE MSDB-TO-DEDB CONVERSION UTILITY WITH THE CONVERT OPTION
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//MACBLIB DD DSN=MSDB.ACBLIB,DISP=SHR
//DACBLIB DD DSN=DEDB.ACBLIB,DISP=SHR
//MSDBINIT DD DSN=IMS.MSDBINIT,DISP=SHR
//DURDBDFN DD DSN=DURDBDFN,DISP=(,CATLG,DELETE),UNIT=SYSDA,
//          VOL=SER=333333,SPACE=(TRK,(1,1))
```



```

//SAVE1 DD DSN=DBT.FORMAT,DISP=(,CATLG,DELETE),UNIT=SYSDA,
// VOL=SER=IMS333,SPACE=(CYL,(1,1)),
// DCB=(BLKSIZE=13030,RECFM=VB)
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
TYPE=CONVERT
MSDB=SAVINGS,DEDB=SAVINGS
/*
//SORT EXEC PGM=SORT
/* EXECUTE SORT UTILITY TO ORDER RBAS IN DBT.FORMAT DATA SET
//STEPLIB DD DSN=program.lib,DISP=SHR
//SYSOUT DD SYSOUT=A
//SORTIN DD DSN=DBT.FORMAT,DISP=SHR
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTOUT DD DSN=DBT.FORMAT.SAVE1,DISP=(,CATLG,DELETE),
// UNIT=SYSDA,VOL=SER=IMS333,SPACE=(CYL,(1,1)),
// DCB=(BLKSIZE=13030,RECFM=VB)
//SYSIN DD *
SORT FIELDS=(5,77,CH,A) /* Control statements to order RAP RBA */
/*
//DELETE EXEC PGM=IEBGENER
/* EXECUTE IEBGENER TO DELETE DATA SET DBT.FORMAT
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DISP=(OLD,DELETE,KEEP),
// DSN=DBT.FORMAT
//SYSUT2 DD DUMMY
/*
//RELOAD EXEC PGM=FABCUR3
/* EXECUTE THE DBT RELOAD UTILITY TO LOAD AREA SAVE1
//SAVE1 DD DSN=SAVE1,DISP=OLD
//DURDATA DD DSN=DBT.FORMAT.SAVE1,DISP=OLD
//DURBDFN DD DSN=DURBDFN,DISP=OLD
//DURIWRK DD
//DURAUDIT DD
//SYSPRINT DD
//SYSIN DD

```

## DEDB を MSDB に戻す方法

フォールバック手順の最初のステップでは、DEDB アンロード/再ロード・ユーティリティ (IMS Fast Path Basic Tools の一部) を使用して DEDB をアンロードします。

フォールバック手順の DEDB は SAVINGS という名前が付けられています。これが、MSDB に変換されます。区域名は SAVE1 です。

次のステップでは、MSDB - DEDB 間変換ユーティリティを実行します。SYSIN ストリーム内の入力制御ステートメントは、この実行が MSDB から DEDB へのフォールバックであることを示します。

MSDB.ACBLIB データ・セットには、SAVINGS の DMB が MSDB であることを指定する DBD が入っています。DEDB.ACBLIB データ・セットには、SAVINGS の DMB が DEDB であることを指定する DBD が入っています。

MSDBLCHG は、変換によって生じた出力データ・セットです。これには、SAVINGS からのセグメントがすべて入っています。形式は、MSDBINIT データ・セットの形式と同じです。MSDBLCHG は、MSDB 保守ユーティリティーへの入力として使用されます。

次のステップでは、IMS.MSDBLCHG データ・セットを入力として使用して MSDB 保守ユーティリティーを実行します。MSDB 保守ユーティリティーは、MSDBLCHG データ・セットからのレコードを、MSDBINIT データ・セットへ追加します。

MSDBINIT データ・セットを使用して、その新しい MSDB セグメントをオンラインにすることができます。

以下の図は、DEDB を変換して MSDB に戻す例を示しています。

```
//UNLOAD EXEC PGM=FABCUR1
//* EXECUTE THE DBT UNLOAD UTILITY TO UNLOAD AREA SAVE1.
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//OLDACB DD DSN=MSDB.ACBLIB,DISP=SHR
//NEWACB DD DSN=DEDB.ACBLIB,DISP=SHR
//SAVE1 DD DSN=SAVE1,DISP=OLD
//DURD0010 DD DSN=DBT.SEGMENTS.SAVE1,DISP=(,CATLG,DELETE)
//DURDBDFN DD
//DURS0010 DD
//DURAUDIT DD
//SYSPRINT DD
//SYSIN DD
//
//
//SORT EXEC PGM=SORT
//* SORT THE OUTPUT OF THE UNLOAD
//SORTIN DD DSN=DBT.SEGMENTS.SAVE1,DISP=OLD
//SORTOUT DD DSN=DBT.FORMAT.SAVE1,DISP=(,CATLG,DELETE)
//SYSIN
//SORTWKnn
//
//
//FALLBACK EXEC PGM=DBFUCDB0
//* EXECUTE THE CONVERSION UTILITY WITH THE FALLBACK OPTION
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//MACBLIB DD DSN=MSDB.ACBLIB,DISP=SHR
//DACBLIB DD DSN=DEDB.ACBLIB,DISP=SHR
//MSDBLCHG DD DSN=IMS.MSDBLCHG,DISP=(,CATLG,DELETE),
// VOL=SER=IMS333,SPACE=(CYL,(1,1)),
// DCB=(LRECL=13026,BLKSIZE=13030,RECFM=VBT)
//SAVE1 DD DSN=DBT.FORMAT.SAVE1,DISP=OLD
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
TYPE=FALLBACK
MSDB=SAVINGS,DEDB=SAVINGS
/*
//MAINT EXEC PGM=DBFDBMA0
//* EXECUTE THE MSDB MAINTENANCE UTILITY
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//MSDBPRT DD SYSOUT=A
//MSDBOLD DD DSN=OLD.MSDBINIT,DISP=OLD
//MSDBLCHG DD DSN=IMS.MSDBLCHG,DISP=OLD
```

```
//MSDBACB DD DSN=MSDB.ACBLIB,DISP=OLD
//MSDBNEW DD DSN=NEW.MSDBINIT,DISP=(,CATLG,DELETE)
//MSDBCTL DD *
          DBN=SAVINGS MODE=INSERT
```

関連資料:

299 ページの『第 21 章 MSDB ダンプ・リカバリー・ユーティリティ (DBFDBDR0)』

---

## MSDB の DEDB への変換

変換には、MSDB ダンプ・リカバリー・ユーティリティ (DBFDBDR0) と、IBM IMS Fast Path Basic Tools for z/OS の DEDB アンロード/再ロード・ユーティリティまたは同等のユーティリティを使用する必要があります。

ヒント: MSDB に既に指定されている単一の値は固定長セグメントを暗黙指定するものであるため、BYTES パラメーターを変更する必要はありません。

MSDB を DEDB に変換するには、以下のようにします。

1. MSDB DBD に、以下の DBD の変更を行います。
  - a. MSDB DBD 内の ACCESS パラメーターの、MSDB から DEDB への変更
  - b. ランダマイザー名についての RMNAME パラメーターの指定
  - c. DEDB 区域についての AREA ステートメントの指定
2. 変換する MSDB のための DEDB DBD を作成します。
3. データベース記述生成ユーティリティおよびアプリケーション制御ブロック生成ユーティリティを実行して、新しいアプリケーション制御ブロック・ライブラリーを作成します。
4. MSDB ダンプ・リカバリー・ユーティリティを実行して、MSDBINIT データ・セットを作成します。MSDBINIT データ・セットには、新しい DEDB 区域にロードされる MSDB セグメントを入れます。
5. CONVERT オプションを指定して、MSDB - DEDB 間変換ユーティリティを実行します。このユーティリティにより、変換される MSDB ごとにアンロード形式でデータ・セットが作成されます。
6. MSDB ごとにアンロード形式で RAP RBA 値の昇順によりデータ・セットをソートします。
7. IMS 高速機能 DEDB 再ロード・ユーティリティを実行して、これらのデータ・セットを DEDB にロードします。

関連タスク:

『DEDB から MSDB へのフォールバックの実行』

---

## DEDB から MSDB へのフォールバックの実行

フォールバックには、MSDB 保守ユーティリティ (DBFDBMA0) および DEDB アンロード/再ロード・ユーティリティ (IBM IMS Fast Path Basic Tools for z/OS の DEDB アンロード/再ロード・ユーティリティなど) を使用する必要があります。

フォールバックを実行するためには、次のことを行います。

1. アンロード・ユーティリティを実行して、DEDB をアンロードします。このユーティリティにより、MSDB - DEDB 間変換ユーティリティの入力が作成されます。
2. FALLBACK オプションを指定して、MSDB - DEDB 間変換ユーティリティを実行します。FALLBACK オプションにより、MSDBLCHG データ・セットが作成されます。このデータ・セットには、MSDB 保守ユーティリティへの入力となる変更レコードが入っています。
3. MSDBLCHG データ・セットをルート・キーによってソートします。MSDBLCHG データ・セット・レコードの形式は、MSDBINIT データ・セット・レコードの形式と同じです。必要な SORT 制御ステートメントは、ルート・キーの長さおよびデータ・タイプによって異なります。データ・セットを昇順にソートします。
4. MSDB 保守ユーティリティを実行して、新しい MSDBINIT データ・セットを作成します。作成された MSDBINIT を使用して、MSDB をストレージに再ロードすることができます。

関連タスク:

415 ページの『MSDB の DEDB への変換』

---

## DBFUCDB0 ユーティリティの変換およびフォールバックの出口ルーチン

MSDB - DEDB 間変換ユーティリティ (DBFUCDB0) は、出口ルーチンを使用することにより、MSDB レコードを DEDB アンロード形式に変換したり、フォールバックの際に DEDB アンロード形式から MSDBINIT 形式に戻す変換を行ったりします。

DEDB アンロード形式は、DEDB データベースをアンロードおよび再ロードするために使用するユーティリティまたはツールに固有です。IMS とともに出荷される DBFUCDX0 出口ルーチンおよび DBFUCDX1 出口ルーチンは、別売品である IBM IMS Fast Path Basic Tools for z/OS に付属の DEDB アンロード/再ロード・ユーティリティをサポートするようにコーディングされています。別の製品を使用して DEDB データベースをアンロードおよび再ロードする場合、その製品がサポートする形式に MSDB レコードをフォーマット設定するように、DBFUCDX0 出口ルーチンおよび DBFUCDX1 出口ルーチンの両方を変更する必要があります。

### 変換出口ルーチン (DBFUCDX0)

アンロード形式に変換するための出口ルーチン (DBFUCDX0) は、パラメーター・リストを使って呼び出します。リストの最初のワードは機能コードです。機能コードは次のとおりです。

- 0 INIT 機能。この機能は、変換処理の始めで一度呼び出されます。この機能を使用すれば、必要とする特別なセットアップ処理をすべて行うことができます。
- 4 CONVERT 機能。この機能は、変換する入力 MSDB レコードのアドレスと、変換されたレコードの出力位置を示します。この機能コードには、それ以外の情報も含まれます。以下のリストに、機能コードの後のワードにあるアドレスを示します。

ワード 内容

- 2 変換する MSDBINIT レコードのアドレス
  - 3 変換されたレコードの出力域のアドレス
  - 4 DBFDMCB によってマップされた DMCB のアドレス
  - 5 DBFDMAC によってマップされた DMAC のアドレス
  - 6 DBFBMSDB によってマップされた BHDR のアドレス
- 8 TERM 機能。この機能は、変換処理の終わりで一度呼び出されます。この機能を使用すれば、必要とする特別な終結処理をすべて行うことができます。

### フォールバック出口ルーチン (DBFUCDX1)

フォールバック手順を実行するための出口ルーチン (DBFUCDX1) は、アンロードされた形式のレコードを MSDBINIT 形式に変換します。このルーチンは、パラメーター・リストを使って呼び出します。リストの最初のワードは機能コードです。機能コードは次のとおりです。

- 0 INIT 機能。この機能は、変換処理の始めで一度呼び出されます。この機能を使用すれば、必要とする特別なセットアップ処理をすべて行うことができます。
  - 4 FALLBACK 機能。この機能は、変換する入力 MSDB レコードのアドレスと、変換されたレコードの出力位置を示します。この機能コードには、それ以外の情報も含まれます。以下のリストに、機能コードの後のワードにあるアドレスを示します。
- ワード 内容
- 2 変換する入力レコードのアドレス
  - 3 変換されたレコードの出力域のアドレス
- 8 TERM 機能。この機能は、変換処理の終わりで一度呼び出されます。この機能を使用すれば、必要とする特別な終結処理をすべて行うことができます。

### 新しい出口ルーチンを使用する手順

これらの変換には、ユーザー固有の出口ルーチンを使用することができます。処理は次のとおりです。

1. 新しい出口ルーチンを作成して、MSDBINIT レコードを新しいアンロード形式に変換します。このルーチンは、指定されたルーチン (DBFUCDX0) と同じ名前を持っていなければなりません。
2. 新しい DBFUCDX0 を変換ユーティリティー・ロード・モジュールにバインドします。
3. JCL でアンロード/再ロード・ユーティリティーを実行する準備をします。

FALLBACK オプションに同じ手順を使用します。新しい出口ルーチンは、DBFUCDX1 という名前にしなければなりません。



---

## 第 30 章 データベース部分再編成ユーティリティ (DFSPRCT1 および DFSPRCT2)

データベース部分再編成ユーティリティは、HDAM または HIDAM データベースの中でユーザーが指定する範囲を再編成するために使用します。

範囲とは、連続したキー値を持つ HIDAM レコードのグループ、または連続した相対ブロック番号を持つ HDAM レコードのグループのどちらかです。範囲は、キーまたはブロック番号の最小値と最大値の対を用いて指定します。

関連データベースは、49 個まで処理でき、セグメント・タイプは、500 個まで再編成に関与できます。スキャンおよび再ロード・アクションは、ポインター解決で 500 個まで指定できます。KEYRANGE または FROMAREA は、10 個まで指定でき、TOAREA は、各 FROMAREA または KEYRANGE の後に 10 個まで指定できます。

データベース調査ユーティリティ (DFSPRSUR) は、データベースの物理編成を記述する報告書を作成するために使用できます。この情報は、再編成の範囲を決定する上で役立ちます。

データベース部分再編成ユーティリティは、他のさまざまな再編成ユーティリティと類似した次のような操作を 1 回の実行で実行します。

- 特定の範囲内のすべてのルート・セグメントとそのルート・セグメントに從属するすべてのセグメントを、階層順序でアンロードする。
- これらのルート・セグメントと從属セグメントを、階層順序で特定の連続フリー・スペースに再ロードする。
- 再ロードされたセグメントに関係する、次のようなポインターをすべて解決する。
  - 同じデータベース内で論理的に関係を持つセグメント
  - 他のデータベース内で論理的に関係を持つセグメント
  - 選択した範囲の境界にあるルート・セグメントの物理兄弟ポインター

再編成を行うときは、次の 2 ステップで行います。

1. DFSPRCT1 ユーティリティでは、データベースの使用を必要としないで、ユーザー・エラーの有無を調べるために事前再編成機能を実行します。
2. DFSPRCT2 ユーティリティでは、データベースをオフラインにして、アンロード/再ロード/ポインター解決機能を実行します。

データベース部分再編成ユーティリティは再始動機能を備えています。アンロード/再ロード・フェーズの前と各ソート・フェーズの終了時と接頭部更新フェーズの開始時に、ユーティリティ・チェックポイントがとられます。再始動はどのチェックポイントからでも行うことができます。再始動を行う前に、バッチ・バックアウト・ユーティリティ (DFSBO00) を使用して、チェックポイントがとられた後に行われた変更を元に戻しておかなければなりません。

サブセクション:

- 『制約事項』
- 『前提条件』
- 『要件』
- 『推奨事項』
- 『入出力』
- 422 ページの『JCL 指定』
- 426 ページの『戻りコード』

## 制約事項

データベース部分再編成ユーティリティを使用するときは、次のような制約があります。

- データベースに対する構造上の変更はできません。
- 論理的に関係を持つ HISAM データベースでは、論理子または論理親セグメントの中に直接ポインターを持つことはできません。
- DFSPRCT1 ユーティリティと DFSPRCT2 ユーティリティは、全機能非 HALDB データベースのみで機能します。
- データベースを変更するユーティリティは、データベースが静止している間に実行することはできません。

## 前提条件

現在、データベース部分再編成ユーティリティに対して文書化された前提条件はありません。

## 要件

現在、データベース部分再編成ユーティリティに対して文書化された要件はありません。

## 推奨事項

現在、データベース部分再編成ユーティリティに対して文書化された推奨事項はありません。

## 入出力

データベース部分再編成ユーティリティは、2 ステップで実行されます。各ステップでの入力を、次の節で説明します。

このユーティリティによっていくつかの報告書が作成されます。

### ステップ 1: 事前再編成

データベース部分再編成の第 1 ステップでは、次のものから成る初期設定機能が実行されます。

- レコードの範囲を指定している制御ステートメントの読み取り
- ステップ 2 で使用するための制御テーブルの作成



- ポインター解決を必要とする論理的に関係を持つデータベースの判別
- 報告書の作成

ステップ 1 では、再編成するデータベースの DBD と、レコード範囲とソート・オプションを定義しているユーティリティー制御ステートメントが入力として必要です。

1 次出力は、部分再編成を行うためにステップ 2 で使用される制御テーブルのセットです。ステップ 2 で使用する PSB を作成するために、PSB ソース・ステートメントを作成することができますが、これはオプションです。(この PSB は、必要となる 4 つの GSAM 作業データ・セットのための PCB を含んでいることが必要であり、ステップ 2 の前にアSEMBルして結合しておかなければなりません。再編成するデータベースのために PSB 生成を行っておけば、以後に行う同じデータベースの再編成では、同じ処理を繰り返す必要はありません。ステップ 1 では、2 つのスキャン報告書も作成されます。「REQUIRED SEGMENT SCAN」報告書には、ステップ 2 の処理で自動的にスキャンされる論理的に関係のあるデータベースの、論理的に関係のあるセグメントがすべてリストされます。「OPTIONAL SEGMENT SCAN」報告書には、スキャンをオプションとして行う論理的に関係のあるデータベースの、論理的に関係のあるセグメントがすべてリストされます。(スキャンを行うオプション・セグメントを選択するには、SCANSEG 制御ステートメントをステップ 2 の入力として用意してください。)

スキャンでは、スキャンされるすべてのセグメント・タイプが現れるたびに検査され、そのオカレンスごとに作業レコードが作成されます。ステップ 2 では、ポインター更新を必要とするセグメント・オカレンスの位置の検出を迅速化するために、これらの作業レコードが使用されます。オプション・スキャン・セグメントでスキャンが行われないときは、ステップ 2 の接頭部更新フェーズはポインター・チェーンに従って、更新すべきセグメントの位置を突き止めるようにしなければなりません。

オプション・セグメントのスキャンを行うと、次のようなときに利点があります。

- 同じデータベース内の他のセグメントがスキャンを必要とするとき。
- オプション・スキャン・セグメントの大部分が、論理的に関係を持つデータベースに移動される必須スキャン・セグメントを指すポインターを持っているとき。

その他のステップ 1 の出力としては、報告書および戻りコードがあり、場合によってはエラー・メッセージもあります。

## ステップ 2: アンロード、再ロード、ポインター解決

第 2 ステップでは、ステップ 1 で作成された制御テーブルおよびユーザーが指定した制御ステートメントが読み取られます。指定されたレコード範囲に従って、すべてのセグメント (ルートとその従属) は、階層順に中間データ・セットにアンロードされます。これらのレコードが占有していたデータベース内のスペースは解放されます。同様に、ユーザーの指定に従って、レコードはデータベース内のフリー・スペースの範囲に再ロードされ、新しいレコード位置が作業レコード内に保管されます。その後、論理的に関係を持つすべてのデータベースがスキャンされ、移動されたレコードを指すポインターを見付けます。ポインター解決が必要な箇所を示した作業レコードが作成されます。次に、すべての作業レコードは、データベース名

とセグメント名に従って、(z/OS SORT によって) ソートされます。最後に、新しい位置を持つレコードを指すすべてのポインターが、新しい位置を指すように変更されます。

ステップ 2 の出力には、部分的に再編成されたデータベースと、何が行われたかを示す報告書と戻りコードがあります。実行が失敗した場合には、エラー・メッセージが出されます。

## JCL 指定

データベース部分再編成ユーティリティーは、2 つの標準の z/OS ジョブとして IMS バッチ処理領域で実行されます。次の JCL ステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

ステップ 1 の JCL ステートメント: *DFSPRCT1*

この節では、ステップ 1 の EXEC ステートメントと DD ステートメントについて説明します。

ステップ 1 の EXEC ステートメント

EXEC ステートメントは、実行するプログラムを記述します。ステートメントの形式は次のとおりです。

```
PGM=DFSPRCT1,PARM=()
```

ステップ 1 の DD ステートメント

次の DD ステートメントは、ステップ 1 での必須およびオプションの入力および出力データ・セットを定義します。

### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。

### IMS DD

再編成するデータベースおよび論理的に関係を持つすべてのデータベースを記述する DBD が入っているライブラリーを定義します。

### SYSPRINT DD

メッセージおよび報告書出力データ・セットを定義します。このデータ・セットは、テープ、直接アクセス装置、またはプリンターに置いておくことも、出力ストリームに入れて送ることもできます。

このデータ・セットの DCB パラメーターは、RECFM=FBA と LRECL=121 です。BLKSIZE は DD ステートメントで指定し、121 の倍数になっていなければなりません。

この DD ステートメントは必須です。

### SYSIN DD

このジョブのための入力制御データ・セットを定義します。このデータ・セット

は、テープまたは直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。LRECL は 80 と指定しておかなければなりません。

#### **SYSABEND DD または SYSUDUMP DD**

ダンプ・データ・セットを定義します。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

#### **SYSPUNCH DD**

ユーザーが PSB ソース・ステートメントを作成することを選択した場合、それらを入れるデータ・セットを定義します。

このデータ・セットの DCB パラメーターは、RECFM=FBS と LRECL=80 です。BLKSIZE はこの DD ステートメントで指定し、80 の倍数になっていなければなりません。

#### **DFSPRCOM DD**

ステップ 2 で使用するための制御テーブルを入れるデータ・セットを定義します。

#### ステップ 2 の JCL ステートメント: **DFSPRCT2**

この節では、ステップ 2 の EXEC ステートメントと DD ステートメントについて説明します。

#### ステップ 2 の **EXEC** ステートメント

EXEC ステートメントは、実行するプログラムを記述します。形式は次のとおりです。

```
PGM=DFSRR00,PARM=(DLI,DFSPRCT2,psbname)
```

DLI は領域を記述し、DFSPRCT2 はステップ 2 のプログラムを指定し、*psbname* は、再編成するデータベースが入っている PSB の名前です。通常の IMS 定位置パラメーターは、PARM フィールドの中で *psbname* の後に置くことができます。

#### ステップ 2 の **DD** ステートメント

次の DD ステートメントは、ステップ 2 での必須およびオプションの入力および出力データ・セットを定義します。

#### **STEPLIB DD**

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

#### **DFSRESLB DD**

IMS SVC モジュールが入っている許可ライブラリーを指します。

#### **IMS DD**

再編成するデータベースおよび論理的に関係を持つすべてのデータベースを記述する DBD が入っているライブラリーを定義します。このライブラリーには、EXEC ステートメントで指定された PSB と、必要な GSAM DBD も入っている必要があります。

**DFSPRF1 DD**

中間作業データ・セットを定義します。このデータ・セットには、DISP=NEW と指定しておきます。

**DFSPRF2 DD**

中間作業データ・セットを定義します。これは GSAM データ・セットです。DCB パラメーターとして、LRECL=18 と RECFM=FB が必要です。このデータ・セットには、DISP=NEW と指定しておきます。ただし、DFSPR001 より大きい番号を持つチェックポイントから再始動を実行する場合は除きます。この場合、DISP=OLD と指定しておきます。このデータ・セットには、GSAM DBD の生成が必要です。

**DFSPRF3 DD**

中間作業データ・セットを定義します。これは GSAM データ・セットです。DCB パラメーターとして、LRECL=18 と RECFM=FB が必要です。このデータ・セットには、DISP=NEW と指定しておきます。ただし、DFSPR001 より大きい番号を持つチェックポイントから再始動を実行する場合は除きます。この場合、DISP=OLD と指定しておきます。このデータ・セットには、GSAM DBD の生成が必要です。

**DFSPRF4 DD**

中間作業データ・セットを定義します。これは GSAM データ・セットです。DCB パラメーターとして、LRECL=1000 と RECFM=VB が必要です。このデータ・セットには、DISP=NEW と指定しておきます。ただし、DFSPR001 より大きい番号を持つチェックポイントから再始動を実行する場合は除きます。この場合、DISP=OLD と指定しておきます。このデータ・セットには、GSAM DBD の生成が必要です。

**DFSPRF5 DD**

中間作業データ・セットを定義します。これは GSAM データ・セットです。DCB パラメーターとして、LRECL=1000 と RECFM=VB が必要です。このデータ・セットには、DISP=NEW と指定しておきます。ただし、DFSPR001 より大きい番号を持つチェックポイントから再始動を実行する場合は除きます。この場合、DISP=OLD と指定しておきます。このデータ・セットには、GSAM DBD の生成が必要です。

**DFSPRF6 DD**

中間作業データ・セットを定義します。このデータ・セットには、DISP=NEW と指定しておきます。ただし、DFSPR001 より大きい番号を持つチェックポイントから再始動を実行する場合は除きます。この場合、DISP=OLD と指定しておきます。

**DFSPRF7 DD**

中間作業データ・セットを定義します。このデータ・セットには、DISP=NEW と指定しておきます。ただし、DFSPR002 より大きい番号を持つチェックポイントから再始動を実行する場合は除きます。この場合、DISP=OLD と指定しておきます。

**DFSPRF8 DD**

中間作業データ・セットを定義します。このデータ・セットには、DISP=NEW と指定しておきます。ただし、DFSPR003 より大きい番号を持つチェックポイントから再始動を実行する場合は除きます。この場合、DISP=OLD と指定しておきます。

**DFSPRF9 DD**

中間作業データ・セットを定義します。このデータ・セットには、DISP=NEW と指定しておきます。ただし、DFSPR004 より大きい番号を持つチェックポイントから再始動を実行する場合を除きます。この場合、DISP=OLD と指定しておきます。

**DFSPRFWA DD**

中間作業データ・セットを定義します。このデータ・セットには、DISP=NEW と指定しておきます。

**SORTLIB DD**

z/OS のソート/マージ・プログラムのためのロード・モジュールが入っているデータ・セットを定義します。この DD ステートメントは必須です。

**SORTWKnn DD**

z/OS のソート/マージ・プログラムのための中間記憶データ・セットを定義します。

**IEFRDER DD**

IMS ログ・データ・セットを定義します。このデータ・セットは直接アクセス装置に置いておかなければなりません。このステートメントは必須です。

このデータ・セットが DD DUMMY と指定されているときは、チェックポイント/再始動機能は使用できません。

**SYSPRINT DD**

メッセージおよび報告書出力データ・セットを定義します。このデータ・セットは、テープ、直接アクセス装置、またはプリンターに置いておくことも、出力ストリームに入れて送ることもできます。このステートメントは必須です。

このデータ・セットの DCB パラメーターは、RECFM=FBA と LRECL=121 です。BLKSIZE はこの DD ステートメントで指定し、121 の倍数になっていなければなりません。

**DFSPRCOM DD**

ステップ 1 で作成する、制御テーブルを入れるデータ・セットを定義します。

**SYSIN DD**

このジョブのための入力制御データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。

**SYSOUT DD**

z/OS ソート/マージ・プログラムのためのメッセージ出力データ・セットを定義します。DD 名は、呼び出される SORT の生成時に指定したものです (DD 名は、通常、SYSOUT です)。このデータ・セットは、プリンター、テープ、または直接アクセス装置に置いておくことも、出力ストリームに入れて送ることもできます。この DD ステートメントは必須です。

**database DD**

再編成するデータベース・データ・セット、論理的に関係を持つすべてのデータベース・データ・セット、およびすべての副次索引データ・セットを定義します。この DD 名は、DBD に指定されている DD 名と一致している必要があります。

これが HIDAM データベースのときは、索引データ・セットの DD ステートメントも必要です。この DD 名は、DBD の索引データ・セットに指定されている DD 名と一致している必要があります。

これらのデータ・セットは、直接アクセス装置に置いておかなければなりません。DISP=OLD が望ましい値です。

#### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ・プール情報が入っているデータ・セットを記述します。この DD ステートメントは必須です。

#### **DFSCTL DD**

順次バッファリング (SB) の活動化を要求する SBPARM 制御ステートメントが入っているデータ・セットを記述します。

DFSCTL ファイルは順次データ・セットにすることも、PDS のメンバーにすることもできます。レコード形式は F、FB、または FBS のいずれかで、レコード長は 80 でなければなりません。このデータ・セットは、直接アクセス装置またはテープに置いておくことも、入力ストリームに入れて送ることもできます。

#### **SYSABEND DD または SYSUDUMP DD**

ダンプ・データ・セットを定義します。これらの DD ステートメントはオプションです。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

#### **RECON1 DD**

最初の DBRC RECON データ・セットを定義します。

#### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。

#### **RECON3 DD**

RECON1 または RECON2 にエラーが見付かったとき使用される、オプションの DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

### **戻りコード**

DFSPRCT1 モジュールおよび DFSPRTC2 モジュールは以下の戻りコードを返しますが、エラーが起こった場合は、戻りコードより前に、プログラム実行の結果を詳しく説明した番号付きメッセージを SYSPRINT データ・セットへ送ります。

コード 意味

- 0 要求されたすべての操作が正常に完了した
- 4 警告メッセージが出された
- 8 ジョブを終了させる重大エラーが起こった

関連資料:

39 ページの『第 4 章 データベース調査ユーティリティ (DFSPRSUR)』

217 ページの『第 15 章 バッチ・バックアウト・ユーティリティー  
(DFSBB00)』

➞ DBBATCH プロシージャ (システム定義)

➞ DLIBATCH プロシージャ (システム定義)

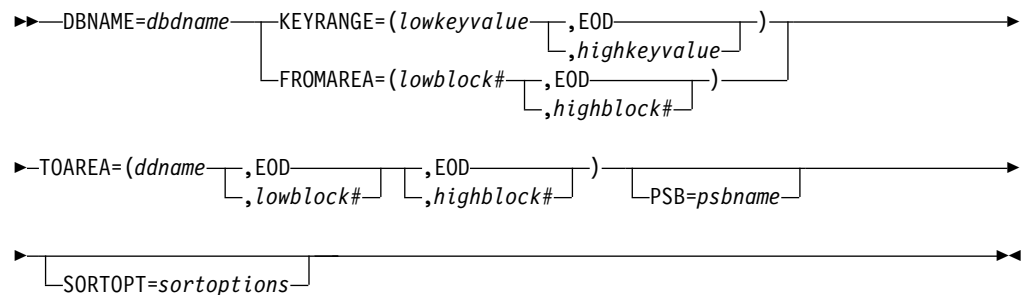
## データベース部分再編成ユーティリティーの制御ステートメント

入力ステートメントは、データベース部分再編成ユーティリティーの処理オプションを記述するために使用します。

入力ステートメントは、以下のガイドラインに準拠している必要があります。

- ステートメントの先頭の文字は、 17 桁目よりも前から始まっていなければなりません。
- 1 つのステートメントで複数のキーワードを指定することはできません。
- 入力ステートメントに空白があると (オペランドの途中の空白を除き)、空白の後に続く文字はコメントと見なされます。
- キーワードのオペランドが 1 つの入力ステートメントに入りきらないときは、継続ステートメントを使うことができます。
- 継続が許されるのは、キーワードのオペランドとオペランドの間だけであり、オペランドの途中では許されません。ただし、KEYRANGE キー・オペランドは例外です。
- ステートメントを継続するには、72 桁目に空白以外の文字を入れなければなりません。
- 継続ステートメントの先頭文字は、 16 桁目から始めなければなりません。
- 1 桁目にアスタリスクを入れると、コメントだけのステートメントが指定できます。

### ステップ 1 の DFSPRCT1 ユーティリティー制御ステートメント



#### DBNAME

部分的に再編成するデータベースを指定します。このオペランドは、HD 編成の DBD の名前にしなければなりません。

要件: DBNAME は最初のキーワードとして必要であり、一度しか使えません。

#### KEYRANGE

再編成するキーの範囲を 1 つ指定します。データベースが HIDAM のときは、KEYRANGE は少なくとも 1 つ指定しなければなりません。KEYRANGE

は 10 個まで指定できます。オペランドはルート・セグメント・キーまたは総称キーであり、どちらも最大 255 バイトです。キーは、文字値か 16 進値のどちらかで指定します。文字キーがデフォルトです。値の先頭に X を付け、値を単一引用符で囲むと、キーを 16 進数で表すことができます。例えば、KEYRANGE=(X'C8C5E7', X'D2C5E8') のようにします。上限がデータベースの終わりであれば、最大キー値を EOD として指定できます。

データベースが HDAM のときは、KEYRANGE は無効です。

#### FROMAREA

再編成するブロックの範囲を 1 つ指定します。データベースが HDAM のときは、FROMAREA は少なくとも 1 つ指定しなければなりません。これらのキーワードは 10 個まで指定することができます。オペランドは、ルート・アドレス可能域内のブロック番号です。上限がルート・アドレス可能域内の最後のブロックであれば、最大ブロック番号を EOD として指定できます。

データベースが HIDAM のときは、FROMAREA は無効です。

#### TOAREA

再編成したセグメントを入れるデータベースの区域を 1 つ指定します。TOAREA キーワードは、再編成するデータベース内のデータ・セット・グループごとに 1 つ指定し、それらをまとめて 1 つのセットにします。指定された KEYRANGE または FROMAREA ごとに、1 セットずつ必要です。ddname は、編成するデータベース内のデータ・セットを定義している DD の名前にしなければなりません。

EOD は、最小ブロック番号と最大ブロック番号のどちらの値として指定しても構いません。EOD を最小ブロック番号として指定したときは、セグメントは、オーバーフロー域の終わりから置かれていきます。この場合、最大ブロック番号は指定しません。EOD を最大ブロック番号として指定したときは、TOAREA は、最小ブロック番号からデータ・セット・グループの終わりまでになります。

#### PSB

ステップ 1 で生成し、ステップ 2 で使用する PSB の名前を指定します。前のデータベース部分再編成の実行で PSB が既に生成されていて、新しい PSB を必要としないときは、このキーワードを省くことができます。

#### SORTOPT

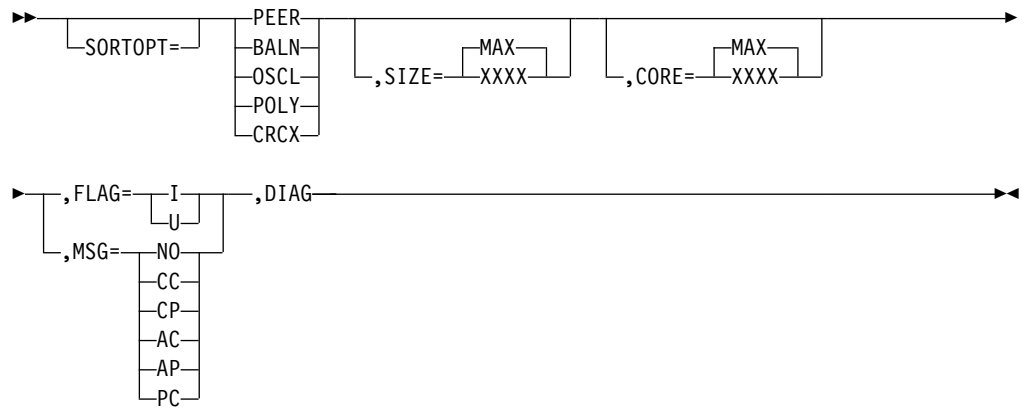
すべてのソートに関するオプションを指定します。SORTOPT はオプションであり、一度だけ指定できます。

制約事項: 引用符で囲まれた文字ストリングに入れられる文字数は、69 文字までです。

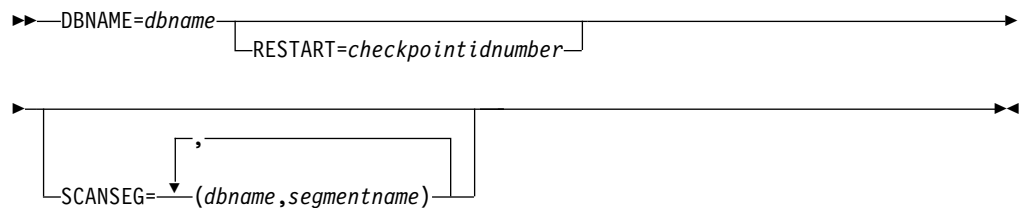
使用するソート・オプションだけをオペランド・ストリングに入れます。省いた SORTOPT オペランドを示すために別のコンマを使用する必要はありません。データベース部分再編成でソートを行うときのデフォルト・オプションは、通常の z/OS ソート・オプションのデフォルトです。

データベース部分再編成ユーティリティで使用できるソート・オプションは、次のとおりです。





## ステップ 2 の DFSPRCT2 ユーティリティー制御ステートメント



### DBNAME

部分的に再編成するデータベースを指定します。 *dbname* は、ステップ 1 で指定したものと同一 DB 名でなければなりません。

### RESTART

指定したチェックポイントから部分再編成を再始動させることを指定します。RESTART を指定するときは、一度しか使用できません。他のキーワードは、DBNAME を除き、すべて無視されます。

### SCANSEG

スキャン処理の対象とするセグメント・タイプを指定します。SCANSEG ステートメントは、ステップ 2 の 1 回の実行に 1 つ以上使用できます。ステップ 1 のオプション報告書のオプション・スキャンの節にリストされている DBD 名およびセグメント名のみがオペランドとして使用できます。(スキャンを必要とするものとしてリストされているセグメントは、このステートメントで指定する必要はありません。これらは自動的にスキャンされます。)

ヒント: 同じデータベース内の他のセグメントのスキャンが必要な場合は、オプション・セグメントをスキャンの対象として含めておくと、パフォーマンスが向上します。

## データベース部分再編成ユーティリティーの例

DFSPRCT1 モジュールおよび DFSPRCT2 モジュールのサンプル JCL を以下の例に示します。

この節にある例では、桁合わせを容易にするために、SYSIN ステートメントの上に次のようなコメント行が示されています。

```
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

このコメント行は単なる参考用です。

この節の例で、使用する DBRC で動的割り振りを行わない場合は、以下の図に示す DD ステートメントをサンプル JCL に追加する必要があります。

```
//RECON1 DD DSN=IMS.RECON1,DISP=SHR
//RECON2 DD DSN=IMS.RECON2,DISP=SHR
//RECON3 DD DSN=IMS.RECON3,DISP=SHR
```

サブセクション:

- 『DFSPRCT1 の実行』
- 432 ページの 『DFSPRCT2 の実行』

## DFSPRCT1 の実行

次の例は、データベース部分再編成ユーティリティのステップ 1 を実行するための JCL およびユーティリティ制御ステートメントを示しています。

```
//LOAD1 EXEC PGM=DFSRR00,
// PARM='DLI,DFSDDL0,PRPSB01L,,,,,,,,,N,N'
//PRINTDD DD SYSOUT=A
//PR01DD DD DSN=PR01RW00,DISP=SHR
//PR01IDD DD DSN=PR01I,DISP=SHR
//IMS DD DSN=IMS.PSBLIB,DISP=SHR
// DD DSN=IMS.DBDLIB,DISP=SHR
//IEFRDER DD DUMMY
//DFSVSAMP DD input for VSAM and OSAM buffers and options
//SYSIN DD DSN=IMS.DBTDATA(PR01DT),DISP=SHR
//PTSTE17 EXEC PGM=DFSPRCT1,COND=EVEN
//IMS DD DSN=IMS.PSBLIB,DISP=SHR
// DD DSN=IMS.DBDLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A,DCB=(BLKSIZE=1210)
//SYSUDUMP DD SYSOUT=A
//SYSPUNCH DD DSN=&&DPSB10,SPACE=(TRK,(2,1)),UNIT=SYSDA,
// DISP=(,PASS)
//DFSPRCOM DD DSN=&&DPR10,SPACE=(TRK,(2,1)),UNIT=SYSDA,
// DISP=(,PASS)
//* +----1----+----2----+----3----+----4----+----5----+----6----+----7---
```

```
//SYSIN DD *
KEYRANGE=(000010,000020)
TOAREA=(PR01DD,1,EOD)
KEYRANGE=(000030,000040)
TOAREA=(PR01DD,1,EOD)
KEYRANGE=(000050,000060)
TOAREA=(PR01DD,1,EOD)
KEYRANGE=(000070,000080)
TOAREA=(PR01DD,1,EOD)
KEYRANGE=(000090,000100)
TOAREA=(PR01DD,1,EOD)
KEYRANGE=(000110,000120)
TOAREA=(PR01DD,1,EOD)
KEYRANGE=(000130,000140)
TOAREA=(PR01DD,1,EOD)
PSB=PTSTE17
```

次に、ステップ 1 からの出力報告書のサンプルを示します。以下の図は入力ステートメントを示しています。

```

IMS PARTIAL REORGANIZATION - STEP 1 INPUT STATEMENTS          137/89    PAGE    1
0.....1.....2.....3.....4.....5.....6.....7.....8
123456789012345678901234567890123456789012345678901234567890
  DBNAME=PR07RW12
  KEYRANGE=(000100,000200)
  TOAREA=(PR07DD,005,007)
  PSB=PTSTN07

```

以下の図は範囲の値を示しています。

```

IMS PARTIAL REORG - RANGE VALUES FOR DBD - PR01RW00 137/89 PAGE    2
  KEYRANGE = '000100'
            TO '000200'
  TOAREA   = 00000005 TO 00000007          DDNAME = PR07DD

```

以下の図は、必須スキヤンの対象となるセグメントの名前を示しています。

```

IMS PARTIAL REORG - REQUIRED SEGMENT SCAN FOR DBD - PR01RW00 137/89 PAGE 3
                      NO SEGMENT FOUND FOR REQUIRED SCAN

```

以下の図は、オプション・スキヤンの対象となるセグメントの名前を示しています。

```

IMS PARTIAL REORG - OPTIONAL SEGMENT SCAN FOR DBD - PR01RW00 137/89 PAGE 4
                      NO SEGMENT FOUND FOR OPTIONAL SCAN

```

### PSB の例

以下の図は、PSB キーワードがステップ 1 の入力制御ステートメントに含まれている場合に、ステップ 1 で生成される PSB ソース・ステートメントを示しています。

```

PCB   TYPE=DB,NAME=PR07RW12,KEYLEN=12,PROCOPT=GIR
      SENSEG  NAME=D,PARENT=0
      SENSEG  NAME=F,PARENT=D
PCB   TYPE=DB,NAME=PR071,KEYLEN=6,PROCOPT=G
      SENSEG  NAME=INDEX,PARENT=0
PCB   TYPE=DB,NAME=PR07R,KEYLEN=12,PROCOPT=GIR
      SENSEG  NAME=A,PARENT=0
      SENSEG  NAME=C,PARENT=A
PCB   TYPE=GSAM,DBDNAME=DFSPRF2,PROCOPT=L
PCB   TYPE=GSAM,DBDNAME=DFSPRF3,PROCOPT=L
PCB   TYPE=GSAM,DBDNAME=DFSPRF4,PROCOPT=L
PCB   TYPE=GSAM,DBDNAME=DFSPRF5,PROCOPT=L
PSBGEN LANG=COBOL,CMPAT=YES,PSBNAME=PTSTN07

```

### DBD の例

以下の図は、ステップ 2 の実行前に生成しておかなければならない DBD を示しています。

```

DBD  NAME=DFSPRF2,ACCESS=(GSAM,BSAM)          *
DATASET DD1=DFSPRF2,RECFM=FB,RECORD=18,BLOCK=10
DBDGEN
FINISH
END
DBD  NAME=DFSPRF3,ACCESS=(GSAM,BSAM)          *
DATASET DD1=DFSPRF3,RECFM=FB,RECORD=18,BLOCK=10
DBDGEN
FINISH
END
DBD  NAME=DFSPRF4,ACCESS=(GSMA,BSAM)
DATASET DD1=DFSPRF4,RECFM=VB,RECORD=1000,BLOCK=1
DBDGEN
FINISH

```

```

END
DBD  NAME=DFSPRWF5,ACCESS=(GSAM,BSAM)
DATASET DD1=DFSPRWF5,RECFM=VB,RECORD=1000,BLOCK=1
DBDGEN
FINISH
END

```

## DFSPRCT2 の実行

以下の図は、ステップ 2 を実行するために必要な JCL およびユーティリティー制御ステートメントを示しています。

```

//STEP2 EXEC PGM=DFSRR00
//          PARM=(DLI,DFSPRCT2,PTSTN07)
//STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSN=IMS.SDFSRESL,DISP=SHR
//DFSPRWF1 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(10,5))
//DFSPRWF2 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(01,1)),
//          DCB=(RECFM=FB,LRECL=18,BLKSIZE=180)
//DFSPRWF3 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(01,1))
//          DCB=(RECFM=FB,LRECL=18,BLKSIZE=180)
//DFSPRWF4 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(01,1))
//          DCB=(RECFM=VB,LRECL=1000,BLKSIZE=1004)
//DFSPRWF5 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(01,1))
//          DCB=(RECFM=VB,LRECL=1000,BLKSIZE=1004)
//DFSPRWF6 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(01,1))
//DFSPRWF7 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(01,1))
//DFSPRWF8 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(01,1))
//DFSPRWF9 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(01,1))
//DFSPRWF0 DD DISP=(,PASS),UNIT=SYSDA,SPACE=(CYL,(01,1))
//SORTLIB DD DSN=SYS1.SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,1)
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,1)
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,1)
//IMS DD DSN=IMS.PSBLIB,DISP=SHR
//      DD DSN=IMS.DBDLIB,DISP=SHR
//IEFRDER DD DSN=IMS.LOG1,DISP=(NEW,KEEP),
//          UNIT=TAPE,VOL=SER=222222
//SYSPRINT DD SYSOUT=A,DCB=(BLKSIZE=1210)
//DFSRCOM DD DSN=*.PTSTN07.DFSRCOM,DISP=(OLD,KEEP)
//SYSUDUMP DD SYSOUT=A
//SYSOUT DD SYSOUT=A
//PR07DD DD DSN=PR07RW12,DISP=OLD
//PR07RDD DD DSN=PR07R,DISP=OLD
//PR07RIDD DD DSN=PR07RI,DISP=OLD
//PR07IDDD DD DSN=PR07I,DISP=OLD
//* +----1---+----2---+----3---+----4---+----5---+----6---+----7---

//SYSIN DD *
          DBNAME=PR07RW12
//DFSVSAMP DD *
          1024,4
          4096,8
//DFSCCTL DD *
          SBPARAM ACTIV=COND
/*

```

次に、ステップ 2 からの出力報告書のサンプルを示します。以下の図は入力ステートメントを示しています。

```

IMS PARTIAL REORGANIZATION - STEP 2 INPUT STATEMENTS 137/89 PAGE 1
0.....1.....2.....3.....4.....5.....6.....7.....8
1234567890123456789012345678901234567890123456789012345678901234567890
          DBNAME=PR07RW12

```

以下の図はアンロード統計を示しています。

```

IMS PARTIAL REORG - UNLOAD STATS FOR RANGE 1 FOR DBD - PR07RW12 137/89 PG 2
*
SEGMENT STATISTICS *
RECORD STATISTICS *
SEGMENT  SEG  DSG  BLOCK  % OF SEG IN  SAME BLK AS:  AVEGRAGE  AVERAGE  AVERAGE  TOTAL  AVG SEG PER
NAME      LVL  NUM  SIZE  PHY-TWIN     PHY-PAR     TWINS     CHILDREN  LENGTH  SEGMENTS  DB RECORD
-----  ---  ---  ---  -----  -----  -----  -----  -----  -----  -----
D         1    1   4096   79.3        N/A         N/A        2.2      44.0     29        1.0
F         2    1   4096   90.6        92.2        2.2       0.0      36.0     64        2.2
TOTAL SEGMENTS UNLOADED = 93
AVERAGE DATABASE RECORD LENGTH = 123.2
NUMBER OF ROOT ANCHOR POINTS PER BLOCK = 1
KEYRANGE = '000100'
          TO '000200'

```

「PARTIAL REORGANIZATION-UNLOAD STATISTICS」報告書のフィールドは次のとおりです。

**SEGMENT NAME**

アンロードまたは再ロードされるセグメント・タイプの名前

**SEG LVL**

アンロードまたは再ロードされるセグメント・タイプの階層レベル番号

**DSG NUM**

アンロードまたは再ロードされるセグメント・タイプのデータ・セット・グループ番号

**BLOCK SIZE**

データ・セット・グループのブロック・サイズ

**% OF SEG IN SAME BLK AS PHY-TWIN**

以前にアンロードまたは再ロードされた物理兄弟セグメントと同じデータベース・ブロックを、このタイプのセグメントが占めている割合

**% OF SEG IN SAME BLK AS PHY-PAR**

以前にアンロードまたは再ロードされた物理親セグメントと同じデータベース・ブロックを、このタイプのセグメントが占めている割合

**AVERAGE TWINS**

兄弟ファミリー内の物理兄弟の平均数

**AVERAGE CHILDREN**

このセグメント・タイプの子の平均数

**AVERAGE LENGTH**

このタイプのセグメントの平均長

**TOTAL SEGMENTS**

アンロードまたは再ロードされるセグメントの総数

**AVE SEG PER DB RECORD**

データベース・レコード当たりのセグメントの平均数

```

IMS PARTIAL REORGANIZATION - UNLOAD STATISTICS FOR RANGE 1
FOR DBD - PR07RW12 137/89 PAGE 3
RANGE OF UNLOADED SEGMENTS
DATA SET      LOW BLOCK  HIGH BLOCK
GROUP NUMBER  NUMBER    NUMBER
2             2         5

```

「RANGE OF UNLOADED SEGMENTS」報告書のフィールドは次のとおりです。

**RANGE OF UNLOADED SEGMENTS**

アンロードされるセグメントの実際の範囲。(HDAM データベースでは、この番号は、指定された範囲と異なることがあります。)

**DATA SET GROUP NUMBER**

アンロードまたは再ロードされるセグメント・タイプのデータ・セット・グループ番号

**LOW BLOCK NUMBER**

データ・セット・グループ内でアンロードされたセグメントの最小ブロック番号

**HIGH BLOCK NUMBER**

データ・セット・グループ内でアンロードされたセグメントの最大ブロック番号

```

IMS PARTIAL REORGANIZATION - UNLOAD STATISTICS FOR RANGE 1
FOR DBD - PR07RW12      137/89      PAGE  4
      DISTRIBUTION OF DATABASE RECORDS
NUMBER OF   OBSERVED   PERCENT   CUMULATIVE   CUMULATIVE
BLOCKS     FREQUENCY  TOTAL    PERCENT     REMAINDER
1           27         93.10   93.10       6.90
2           2          6.90   100.00      0.00
MAXIMUM BLOCKS FOR A DATABASE RECORD =  2
MEAN OBSERVED FREQUENCY                =  1.07
  
```

「DISTRIBUTION OF DATABASE RECORDS」報告書のフィールドは次のとおりです。

**DISTRIBUTION OF DATABASE RECORDS**

示されている物理ブロックの数にまたがってアンロードされたデータベース・レコードの分布。この報告書は、1 ブロックから 40 ブロックのみまでを作表し、40 ブロックを超える分布は 1 つの項目に累積されます。

**NUMBER OF BLOCKS**

1 つのデータベース・レコードが占めている物理ブロックの数

**OBSERVED FREQUENCY**

この分布に関して観察されたデータベース・レコードの数

**PERCENT TOTAL**

アンロードされた総データベース・レコード数に対し、観察されたデータベース・レコードが占める割合

**CUMULATIVE PERCENT**

この時点までの割合の合計

**CUMULATIVE REMAINDER**

この時点までに残っている割合の合計

**MAXIMUM BLOCKS FOR A DATABASE RECORD**

1 つのデータベース・レコードが占めているブロックの最大数

**MEAN OBSERVED FREQUENCY**

アンロードされたデータベース・レコードが占めているブロックの平均数

以下の図は、再ロード統計を示しています。

```

IMS PARTIAL REORGANIZATION - UNLOAD STATISTICS FOR RANGE 1
FOR DBD - PR07RW12      137/89      PAGE  5
SEGMENT  SEG  DSG  BLOCK  % OF SEG IN  SAME BLK AS:  RELOAD  DIFFERENCE
NAME     LVL  NUM  SIZE   PHY-TWIN    PHY-PAR      COUNT   RELOAD-UNLOAD
-----  ---  ---  -----  -----
  
```

D	1	1	4096	96.6	N/A	29	0
F	2	1	4096	98.4	98.4	64	0
TOTAL SEGMENTS RELOADED =			93				

**SEGMENT NAME**

アンロードまたは再ロードされるセグメント・タイプの名前

**SEG LVL**

アンロードまたは再ロードされるセグメント・タイプの階層レベル番号

**DSG NUM**

アンロードまたは再ロードされるセグメント・タイプのデータ・セット・グループ番号

**BLOCK SIZE**

データ・セット・グループのブロック・サイズ

**% OF SEG IN SAME BLK AS PHY-TWIN**

以前にアンロードまたは再ロードされた物理兄弟セグメントと同じデータベース・ブロックを、このタイプのセグメントが占めている割合

**% OF SEG IN SAME BLK AS PHY-PAR**

以前にアンロードまたは再ロードされた物理親セグメントと同じデータベース・ブロックを、このタイプのセグメントが占めている割合

**RELOAD COUNT**

再ロードされたセグメントの数

**DIFFERENCE RELOAD-UNLOAD**

アンロードされたセグメント数と再ロードされたセグメント数との差

**TOTAL SEGMENTS RELOADED**

特定の範囲について再ロードされたセグメントの総数

以下の図は、再ロードされる一定範囲のセグメントについての報告書サンプルです。

```

IMS PARTIAL REORGANIZATION - UNLOAD STATISTICS FOR RANGE 1
  FOR DBD - PR07RW12      137/89      PAGE 6
                RANGE OF RELOADED SEGMENTS
DATA SET      LOW BLOCK  HIGH BLOCK  BYTE COUNT INSERTED
GROUP NUMBER  NUMBER     NUMBER     TO OVERFLOW
1             5         6         N/A
DFS3000I SUCCESSFUL COMPLETION OF PARTIAL REORGANIZATION

```

「RANGE OF RELOADED SEGMENTS」報告書のフィールドは次のとおりです。

**DATA SET GROUP NUMBER**

再ロードされるセグメント・タイプのデータ・セット・グループ番号

**LOW BLOCK NUMBER**

データ・セット・グループ内で再ロードされたセグメントの最小ブロック番号

**HIGH BLOCK NUMBER**

データ・セット・グループ内で再ロードされたセグメントの最大ブロック番号

**BYTE COUNT INSERTED TO OVERFLOW**

オーバーフロー域に挿入されたバイト数 (HDAM のみ)

以下の図はスキャン統計を示しています。

```
IMS PARTIAL REORGANIZATION - UNLOAD STATISTICS FOR RANGE 1
      FOR DBD - PR07RW12      137/89      PAGE   7
DATABASE NAME  SEGMENT NAME  SCAN COUNT
PR05R         D              100
TOTAL SEGMENTS SCANNED =      100
DFS3000I SUCCESSFUL COMPLETION OF PARTIAL DATABASE REORGANIZATION
```

「PARTIAL REORGANIZATION SCAN STATISTICS」報告書のフィールドは次のとおりです。

**DATABASE NAME**

スキャンされたデータベースの名前

**SEGMENT NAME**

スキャンされたセグメント・タイプの名前

**SCAN COUNT**

このセグメント・タイプについてスキャンされたセグメントの数



## 第 31 章 ユーティリティー制御機能 (DFSUCF00)

ユーティリティー制御機能 (UCF) は、IMS データベース再編成ユーティリティーの実行とオプションを制御するために使用します。

UCF は、他のユーティリティーの実行を制御するユーティリティーです。UCF は、フリー・フォームでコーディングされた制御ステートメントにより駆動されます。複数の制御ステートメントを指定して、1 つのジョブ・ステップ内の 1 つ以上のデータベースに対して複数のユーティリティーを実行すること (例えば、さまざまな再編成ユーティリティーやバッチ・イメージ・コピー・ユーティリティーの実行) ができます。

UCF には以下の利点があります。

- 再始動処理が用意されており、これは単一の EXEC パラメーターまたは制御ステートメントで開始させることができます。
- 制御ストリーム内の大部分の操作は、都合のよいときに停止させ、再始動させることができます。
- 未処理の要応答オペレーター宛メッセージ (WTOR) 機能を使用すると、ジョブを停止させることも、特定のオプションを入力することもできます。
- 処理中のデータ・レコードにアクセスするためのユーザー出口が用意されています。
- UCF は、制御ステートメントに指定された値に基づいて制御データ・セットを作成します。この制御データ・セットは、特定の操作上の問題からユーザーを保護するようにすべての機能を編成して実行します。

サブセクション:

- 『制約事項』
- 438 ページの『前提条件』
- 438 ページの『要件』
- 441 ページの『推奨事項』
- 441 ページの『JCL 指定』
- 447 ページの『戻りコード』

### 制約事項

ユーティリティー制御機能には、次のような制約があります。

- UCF は、以下のユーティリティーをサポートしません。
  - 変更累積ユーティリティー
  - データベース・バックアウト・ユーティリティー
  - データベース・イメージ・コピー 2 ユーティリティー
  - データベース・リカバリー・ユーティリティー
  - データベース調査ユーティリティー
  - オンライン・データベース・イメージ・コピー・ユーティリティー

- データベース部分再編成ユーティリティー
- UCF は、HALDB データベースをサポートしません。
- UCF は、GSAM データベースを初期ロードするためには使用しないでください。
- FUNCTION=ZB は、DBRC がアクティブなときは RECON を更新しません。
- DFSUCF00 ユーティリティーは、全機能非 HALDB データベースのみで機能します。
- データベースを変更するユーティリティーは、データベースが静止している間に実行することはできません。
- UCF は、高速機能の副次索引データベースをサポートしません。

## 前提条件

現在、DFSUCF00 ユーティリティーに対して文書化された前提条件はありません。

## 要件

ユーティリティー制御機能には以下の要件があります。

### 一般要件

- ユーザー作成ルーチンでオープンしたデータ・セットは、明示的にクローズしなければなりません。そうしないと、UCF は異常終了します。
- UCF を 1 回実行して VSAM データベースを再編成するときは、アンロード機能を要求するユーティリティー制御ステートメントで EXEC=STOP を指定しておきます。アンロード操作が完了すると、UCF は実行を停止します。ここでアクセス方式サービス・ユーティリティーを使用すると、データベースを削除して再割り振りができるので、その後で UCF を再始動させることができます。

複数のデータベースをアンロードする場合、「STOP」を各アンロード機能に指定することも、最後のアンロード機能だけに指定して、すべてのデータベースがアンロードされた後で UCF を停止させることもできます。UCF 機能は、データベース名の照合順序 (例えば、DBX が DBY より先) で行われます。「STOP」は、最後のデータベース名を参照するステートメントに入れ直してください。

- データベースの DBD が変更されていて、論理関係を含んでいる場合にそのデータベースを再編成するときは、次の条件の 1 つに該当する場合は、アンロード機能の後 UCF の実行を停止させなければなりません。
  - カウンター、LT、または LP ポインターのいずれかが変更された。
  - 論理関係にかかわるセグメントの追加または削除による変更があった。
  - DBD 名が変更されたが、その DBD は論理関係を含んでいる。

UCF の実行を停止するには、最後のアンロード機能で EXEC=STOP パラメーターを使用してください。新しい DBD が生成された後で、UCF を (再始動ではなく) 新たに実行して、再ロードを行う必要があります。

- FUNCTION=OP カードは、チェックポイント値と要求パラメーターがエラーを起こした UCF ステップの FUNCTION=OP カードと同じになっていなければなりません。
- 停電などでシステムが突然停止した場合は、再始動を始める前に、クローズしていない出力データ・セットを終了させなければなりません。
- データベースが論理関係を持っている場合は、UCF の制御下で完全にユーティリティを実行するか、あるいは UCF を使用せずにユーティリティを実行しなければなりません。UCF を使用するユーティリティと、UCF を使用しないユーティリティが混在していると、予測できない結果が起こる可能性があります。

#### ロードおよび再ロードの要件

- ユーザーの初期ロード・プログラムを再始動する場合や、HD 再編成再ロード・ユーティリティが再ロード中であった HDAM データベースで再始動を行う場合は、次の注意事項に従ってください。
  - この再始動のためのチェックポイント値は、エラーを起こした UCF 機能に指定されているものと同じ値でなければなりません。
  - ルート・シーケンス・フィールドが非固有の場合、再始動が行われたチェックポイントの後に挿入されたルート・セグメントがあれば、それはデータベースに残っています。再始動に有効な条件は制御された終了 (アプリケーション・プログラムが戻るときに、ゼロ以外の戻りコードを返すだけです)。
  - ルート・シーケンス・フィールドが固有でなく、チェックポイントの後に挿入されたセグメントがあるためにデータベース内に存在するセグメントについて、ルート・セグメントの挿入が行われた場合は、データが比較されます。セグメント・データが同じであれば、古いセグメントはその置換セグメントでオーバーレイされ、従属セグメントは、次に続くユーザー/再ロードによる挿入により再挿入されるため、除去されます。これは、固有のルート・セグメントが見付かるまでの間だけ行われます。新しいキーを持つセグメントまたは異なるデータを持つセグメントが現れた後で、重複するものが見付かった場合は、LB 状況コードが返されます。

再始動時に、ルートが既にデータベースに存在しており、パス CALL を使用してルートとその従属セグメントを挿入する場合に、パス CALL のそのルートに従属セグメントを挿入しようとする、失敗します。

- ユーザー作成ルーチンでオープンしたデータ・セットは、明示的にクローズしなければなりません。そうしないと、UCF は異常終了します。
- UCF を 1 回実行して VSAM データベースを再編成するときは、アンロード機能を要求するユーティリティ制御ステートメントで EXEC=STOP を指定しておきます。アンロード操作が完了すると、UCF は実行を停止します。ここでアクセス方式サービス・ユーティリティを使用すると、データベースを削除して再割り振りができるので、その後で UCF を再始動させることができます。

複数のデータベースをアンロードする場合、「STOP」を各アンロード機能に指定することも、最後のアンロード機能だけに指定して、すべての

データベースがアンロードされた後で UCF を停止させることもできます。UCF 機能は、データベース名の照合順序 (例えば、DBX が DBY より先) で行われます。「STOP」は、最後のデータベース名を参照するステートメントに入れ直してください。

- データベースの DBD が変更されていて、論理関係を含んでいる場合にそのデータベースを再編成するときは、次の条件の 1 つに該当する場合は、アンロード機能の後 UCF の実行を停止させなければなりません。
  - カウンター、LT、または LP ポインターのいずれかが変更された。
  - 論理関係にかかわるセグメントの追加または削除による変更があった。
  - DBD 名が変更されたが、その DBD は論理関係を含んでいる。

UCF の実行を停止するには、最後のアンロード機能で EXEC=STOP パラメーターを使用してください。新しい DBD が生成された後で、UCF を (再始動ではなく) 新たに実行して、再ロードを行う必要があります。

- FUNCTION=OP カードは、チェックポイント値と要求パラメーターがエラーを起こした UCF ステップの FUNCTION=OP カードと同じになっていなければなりません。
- 停電などでシステムが突然停止した場合は、再始動を始める前に、クローズしていない出力データ・セットを終了させなければなりません。
- データベースが論理関係を持っている場合は、UCF の制御下で完全にユーティリティーを実行するか、あるいは UCF を使用せずにユーティリティーを実行しなければなりません。UCF を使用するユーティリティーと、UCF を使用しないユーティリティーが混在していると、予測できない結果が起こる可能性があります。
- ユーザーの初期ロード・プログラム、HD 再編成再ロード・ユーティリティーまたはその両方を UCF のもとで再始動する場合、次のような制約があります。
  - 再始動は、VSAM データ・セットにだけ適用されます。
  - 複数のロード PCB を 1 つの PSB に入れることができます。ただし、初期ロード時、または初期ロードの再始動時に、初期ロード・プログラムが使用できるロード PCB は、1 回の FUNCTION=IL ステートメントの実行につき 1 つだけです。
  - ルート・セグメントには、ユーザーの HDAM ランダム化モジュールがそのルート・セグメントを突き止められるように、キーが付いていなければなりません。
  - 再始動は、キー・フィールドバック域と等しいキーを持つルートの後に置かれた次のルート・セグメントの ISRT から始めなければなりません。
  - 入力ファイルの位置変更は、ユーザー作成初期ロード・プログラムで行わなければなりません。
- 初期ロードまたは再ロード・プログラムの再始動時に、再始動の起点となる有効なチェックポイントがないと UCF が判断すると、UCF はロードまたは再ロードの始めからやり直します。既にデータベースに書き込まれているセグメントがあると、メッセージ DFS730I や、理由コード

I,30 が出されます。そのようなときは、再始動を再び試みる前に、データベース・データ・セットを消去して、再割り振りしておく必要があります。

領域サイズは、600 KB が実行のために最低限必要です。

再始動処理の場合は、PARM='ULU,DFSUCF00,,,0001' と指定しておきます。

## 推奨事項

現在、DFSUCF00 ユーティリティーに対して文書化された推奨事項はありません。

## 入出力

DFSUCF00 は、xxSYSIN DD ステートメントを読み取ることによって他のユーティリティー・プログラムを実行します。出力はそれらのユーティリティー自体の実行結果です。

## JCL 指定

ユーティリティー制御機能は、標準の z/OS ジョブとして実行されます。次の JCL ステートメントが必要です。

- ユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

EXEC ステートメントは次の形式にできます。

```
PGM=DFSRR00,PARM='ULU,DFSUCF00'
```

必要なジョブ制御ステートメントとユーティリティー制御ステートメントが入っているプロシージャーの形式にすることもできます。

### DD ステートメント

#### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

#### DFSRESLB DD

IMS SVC モジュールが入っている許可ライブラリーを指します。

#### IMS DD

データベースとその処理ブロックを記述する DBD および PSB が入っているライブラリー (つまり、DSN=IMS.DBDLIB と IMS.PSBLIB) を定義します。

#### DFSPPRINT DD

出力メッセージ・データ・セットを定義します。このデータ・セットは、テープ、直接アクセス装置、またはプリンターに置いておくことも、出力ストリームに入れて送ることもできます。このファイルは、DCB オペランド LRECL=121 および RECFM=FBA でプログラムに指定されています。BLKSIZE は、この

DD ステートメントで指定しなければなりません。BLKSIZE を指定しない場合、BLKSIZE にはデフォルトがないので結果を予測できません。

#### **DFSYSIN DD**

入力制御ステートメント・データ・セットを定義します。このデータ・セットはテープ、直接アクセス装置、またはシステム読取装置に置いておくことができます。このファイルは、DCB オペランド LRECL=80 と RECFM=FB でプログラムに指定されています。BLKSIZE は、この DD ステートメントで指定することができます。

#### **DFSNJRNL DD**

新しい UCF ジャーナル・データ・セットを定義します。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。ここには、再始動処理時に使用される制御レコードとチェックポイント・レコードが入ります。このデータ・セットの指定は必須です。これは、DCB パラメーター RECFM=VB、BLKSIZE=4008、および LRECL=4000 でプログラムに指定されています。DISP=(,KEEP) は、ユーザーが指定しなければなりません。

ボリューム・シーケンス番号を、VOLUME パラメーターに 1 と指定しておくことも必要です。

#### **DFSQJRN DD**

再始動処理を必要とする前回の実行で作成された、古い UCF ジャーナル・データ・セットを定義します。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。この指定は必須です。再始動するときは、これを、次のような DCB 値を使用してプログラムで定義します。この値とは、すなわち、RECFM=VB、LRECL=4000、および BLKSIZE=4008 です。再始動しないときは、これを DD DUMMY と指定しなければなりません。

#### **DFSNCDS DD**

このプログラムのための新しい制御データ・セットを定義します。このデータ・セットは、テープまたは直接アクセス装置に置いておくことができ、常に DISP=(,KEEP) として定義されます。このデータ・セットの指定は必須であり、DCB パラメーター LRECL=1600 および RECFM=FB でプログラムに指定されています。BLKSIZE の指定は必須です。BLKSIZE を指定しない場合、デフォルトがないので結果を予測できません。再始動処理情報については、DFSOCDS DD を参照してください。

#### **DFSOCDS DD**

再始動処理を必要とする前回の実行で作成された、古い制御データ・セットを定義します。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。再始動するときは、この指定は必須です。

LRECL=1600 および RECFM=FB で DCB をプログラムに定義します。再始動しないときは、このデータ・セットを DD DUMMY と指定できます。この DD ステートメントで BLKSIZE の指定が必須であるのは、このデータ・セットがラベルなしテープに置かれている場合だけです。

#### **DFSRDER DD**

省略可能ですが、DD DUMMY と指定することもできます。IMS システム・ログを定義します。システム・ログは、初期ロードまたは再ロード・ユーティリティでは現在使用されていません。

### **DFSCNTRL DD**

機能ユーティリティーを接続 (アタッチ) する前に作成される制御データ・セットを定義します。このデータ・セットの指定は必須です。これは、テープまたは直接アクセス装置に置いておくことができます。プログラムは、LRECL=80、RECFM=FB、および BLKSIZE=80 で DCB を定義します。

制約事項: DD DUMMY を使ってこのデータ・セットを指定することはできません。

### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。この DD ステートメントが必要とされるのは、使用されたデータベースに VSAM データ・セットまたは OSAM データ・セットがある場合です。

次に示した DD ステートメントは、ジョブ実行が接頭部解決実行とかかわりがある場合に必要です。

### **SYSOUT DD**

z/OS ソート/マージ・プログラムが使用する出力データ・セットを定義します。このデータ・セットは、テープ、直接アクセス装置、またはプリンターに置いておくことも、SYSOUT ストリームに入れて送ることもできます。DCB パラメーターは、ソート・プログラムにより設定されます。

### **SORTWKnn DD**

z/OS のソート/マージ・プログラムのための中間記憶データ・セットを定義します。nn の指定は、2 桁の数字で行います。

次に示した DD ステートメントは、ジョブ実行が論理関係を持つデータベースの再編成、初期ロード、接頭部解決ユーティリティー、または接頭部更新ユーティリティーの実行に関連がある場合に必要です。

### **DFSURWF1 DD**

論理関係または副次索引関係を解決するために使用されるデータ・セットを定義します。このデータ・セットは、接頭部解決ユーティリティーへの入力として使用されます。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。このデータ・セットは再始動処理で必要になる場合もあるため、DISP=KEEP と指定しておきます。RECFM=VB、LRECL=900、および BLKSIZE は、この DD ステートメントに指定しなければなりません。特定の DFSURWF1 データ・セットへの参照はすべて、UCF の 1 回の実行中に含まれている必要があります。(割り込みと再始動は、1 回の実行の一部と見なされます。)

### **DFSURWF2 DD**

中間ソート作業データ・セットを定義します。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。そのサイズは、DFSURWF1 DD ステートメントで定義されたデータ・セットのサイズとほぼ同じです。このプログラムの中で指定されている DCB パラメーターは RECFM=VB と LRECL=900 です。BLKSIZE は、この DD ステートメントで指定しなければなりません。

### **DFSURWF3 DD**

この特定の実行で論理関係にかかわりを持つセグメントの、すべての更新データ

が入っている出力作業データ・セットを定義します。このデータ・セットは、接頭部解決ユーティリティによって作成され、接頭部更新ユーティリティによって使用されます。これは、テープまたは直接アクセス装置に置いておくことができます。プログラムに定義されている DCB パラメーターは、RECFM=VB と LRECL=900 です。BLKSIZE は、この DD ステートメントで指定しなければなりません。接頭部更新ユーティリティの実行中に UCF が終了した場合に再始動できるように DISP=KEEP を指定しておきます。

#### DFSURIDX DD

処理される DBD に副次索引が存在するとき使用される出力作業データ・セットを定義します。このデータ・セットについての要件は、DFSURWF3 について説明したものと同じです。この DD ステートメントが必要になるのは、副次索引が存在するときだけです。このプログラムの中で指定されている DCB パラメーターは、RECFM=VB および LRECL=900 です。BLKSIZE は DFSURIDX DD ステートメントで指定しなければなりません。

次の DD ステートメントは、副次索引再編成アンロードのとき REQUEST=EXTRACT が指定されているとき必要です。

#### DFSEXTDS DD

制御ステートメントで指定されたとおりに共用副次索引から抽出されたレコードのアンロード版を作るときに使用します。この DD ステートメントはオプションですが、FUNCTION=RU 制御ステートメントで E (REQUEST=EXTRACT) が指定されているときだけは必要です。DCB 属性は、使用される出力装置タイプと VSAM LRECL に基づいて、動的に決定されます。標準ラベルを使用しなければなりません。

この実行時に機能ユーティリティによって参照されるすべてのデータベースと、この実行時に作成されるすべての出力データ・ファイルを定義するための DD ステートメントも必要です (例えば、再編成アンロードやバッチ・イメージ・コピー)。再編成再ロード・ユーティリティによって使用される入力ファイルも定義する必要があります。以下の表は、さまざまな DD ステートメントでの FUNCTION キーワードの使い方を示したサマリー表です。

表 21. JCL DD ステートメントのサマリー表

DD ステートメント	ユーティリティ制御ステートメントで 사용되는 FUNCTION= キーワード																
	UCF*	OP	DR	DU	DX	IL	IM	PR	PU	RR	RU	SN	SR	SU	SX	ZB	ZM
IMS	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
DFSPPRINT	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
DFSYSIN	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
DFSNJRNL	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
DFSJRNL	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
DFSNDCS	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
DFSOCDS	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
DFSRDER	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
DFSCNTRL	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
DFSVSAMP			V	V	V	V	V		V	V	V	V	V	V	V	V	
SYSOUT						R		R									
SORTLIB						R		R									
SORTWKnn						R		R									
DFSURWF1			R		R	R		O									
DFSURWF2					R			R									



表 21. JCL DD ステートメントのサマリー表 (続き)

DD ステートメント	ユーティリティ制御ステートメントで使用される FUNCTION= キーワード																
	UCF*	OP	DR	DU	DX	IL	IM	PR	PU	RR	RU	SN	SR	SU	SX	ZB	ZM
DFSURWF3					R			R	R								
DFSURIDX											R						
DFSEXTDS											E						
データ・セット DD			R	R	R	R	R		R	R	R	R	R	R	R	R	R

キー:

R 必須

D 制約を受けていないとき DD DUMMY と指定

V VSAM 編成ファイルのときは必須

E REQUEST=EXTRACT オプションの指定があるときは必須

O 必須だが DD 名の変更が可能

\* この DD ステートメントは UCF の実行に関係するものです。

キーワードのサマリー表

以下の表は、キーワードの使い方を機能ユーティリティ別に要約した表です。「使用回数」の欄は、UCF の各実行で各キーワードが現れる回数を示しています。UCF は、各機能ステートメントについて作業域を使用して、キーワードの値を調べます。この固定長作業域がいっぱいになると、UCF は DFS385A メッセージを出します。

表 22. UCF FUNCTION サマリー表

キーワード	使用回数	機能																
		OP	DR	DU	DX	IL	IM	PR	PU	RR	RU	SN	SR	SU	SX	ZB	ZM	
FUNCTION	N	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
SEQ	N		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
COND	1	1																
REQUEST	N	N	N	N	N		N	N	N	N	N	N	N	N	N			
EXEC	N		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
CKPNT	N	1	1	1	1	1	1		1	1	1	1	1	1	1			
IDLEN	1																	
RECID	N																	
DBNAME	N		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
RELATE	N															1	1	
PURGDT	N																	
KDSDD	N									1	1		1	1	1			
CUMOUT	1																	
CUMIN	N																	
ILPGM	N					1												
DSDDNAM	N					D												
OUTDDS	N			2	2	1	2	D			2	D		2	2			
INDDS	N		1		1			1	1	1			D		D			
LOGIN	N																	
LOGOUT	1																	

表 22. UCF FUNCTION サマリー表 (続き)

キーワード	使用回数	機能															
		OP	DR	DU	DX	IL	IM	PR	PU	RR	RU	SN	SR	SU	SX	ZB	ZM
EXITRTN	N		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
SEGNAME	N											1					
SCANTYP	N											1					
DBDDDS	N		D	D	D	D	1		D	1	1	D	1	1	1	1	
COPY	N										1			1	1		
EXTRACT	N										1						
IDXIN	N										1						
ILPSBNAM	N					1											
MODULE	N																1
CSECT	N																1
VERIFY	N															E	E
REP	N															E	E
VALUE	N															1	1
MSGNUM	N	N															
SICON	N										1						
PURGTM	N																
RBNID	N																1
EXITRLD	N					1								1			
WF1DDDS	N		1		1												

キー:

ブランク

ブランク・スペースは、このキーワードがこの機能に使用できないことを意味します。

**N** このキーワードは何回でも使用できます。

**D** このキーワードは 20 回までを限度として何回でも使用できます。この 20 の限度は、機能要求ごとにすべての「D」の DD 名を加算することで判断されます。

**E** このキーワードは、1 回だけ使用できます。この表に E と示されたキーワードが同じシステム制御ステートメントでその他に指定されていない場合だけ使用できます。(VERIFY キーワードと REP キーワードを同じ制御ステートメントで指定することはできません。)

**1, 2** このキーワードは、この回数を上限として使用できます。

以下の表は、各種の制御ステートメントを作るとき、最低限指定しなければならないキーワードを示しています。

表 23. 機能別の UCF 必須キーワード

キーワード	機能															
	OP	DR	DU	DX	IL	IM	PR	PU	RR	RU	SN	SR	SU	SX	ZB	ZM
FUNCTION	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
DBNAME		X	X	X	X	X			X	X	X	X	X	X	X	
KDSDD										X			X	X		
ILPGM					X									X		
OUTDDS						X				X			X			
INDDS														X		
LOGIN																
SEGNAME											X					

表 23. 機能別の UCF 必須キーワード (続き)

キーワード	機能															
	OP	DR	DU	DX	IL	IM	PR	PU	RR	RU	SN	SR	SU	SX	ZB	ZM
DBDDDS						X									X	
ILPSENAM					X											
MODULE																X
VERIFY															X	X
REP															X	X
VALUE															X	X
RBNID															X	

## 戻りコード

UCF が終了すると、以下の表の戻りコードがレジスター 15 に渡されます。

表 24. UCF 戻りコード

コード	意味	処置
0	処理が正常に終了した。	DFSNJRNL データ・セットおよび DFSNCDS データ・セットを消去できます。
4	警告メッセージが出された。あるいは、EXEC=STOP が指定されていたため終了した。	警告がエラーでないことを確認してください。エラーがない場合は、戻りコード 0 と同じ処置をとってください。エラーまたは EXEC=STOP のために再始動したい場合は、DFSNJRNL と DFSNCDS が保管されていなければなりません。
8	重大エラーが発生した。UCF は終了した。	エラーを訂正して、再始動処理を開始してください。DFSNJRNL データ・セットおよび DFSNCDS データ・セットは、再始動時に DFSOJRNL および DFSOCDS として使用可能になっていなければなりません。
12	UCF に障害が起こり、リカバリ不能エラーで終了した。	エラーを訂正して、UCF を始めからやり直してください。再始動処理を試みてもなりません。再始動はできないので、DFSNJRNL データ・セットおよび DFSNCDS データ・セットを消去できます。

UCF が終了し、ゼロ以外の戻りコードが返された場合は、終了時に使用されていたすべてのデータ・セットが再始動時に必要になることがあります。以下のデータ・セットは、特定の条件が起こったときに、保管しておくことが必要です。

### DFSURWF1



初期ロード時、HD 再編成再ロード時、およびデータベース・スキャン時に作成されたものです。これらのデータ・セットが必要になるのは、これらの機能の 1 つが実行しているときに UCF が終了した場合です。

### DFSURWF3

接頭部解決ユーティリティーにより接頭部更新ユーティリティーのために作

成されたものです。接頭部解決ユーティリティーの実行中に UCF が終了したときは、このデータ・セットを保管しておく必要はありません。これは、接頭部解決が再始動されると、再び作成されるためです。ただし、接頭部更新ユーティリティーの実行中に UCF が終了したときは、WF3 データ・セットは再始動のために保管しておく必要があります。

関連概念:

-  環境に合わせた IMS システムの調整 (システム定義)
-  IMS システム定義プロセスの概要 (システム定義)

関連資料:

321 ページの『第 23 章 HALDB マイグレーション・エイド・ユーティリティー (DFSMAID0)』

---

## DFSUCF00 ユーティリティーの制御ステートメント

DFSUCF00 ユーティリティーの制御ステートメントはフリー・フォームでコーディングできますが、各ステートメントには有効なキーワードを少なくとも 1 つ指定しなければなりません。

キーワード間は、コンマで区切らなければなりません。ステートメントは、72 桁目にブランク以外の文字を入れ、後続のステートメントを 72 桁目よりも前の任意の桁から入力していくことにより継続させることができます。1 つの完結した制御ステートメントは、最初のステートメントと、継続ステートメント (存在する場合) から構成されます。以下に例を示します。

```
FUNCTION=OP  
FUNCTION=DX,DBNAME=dbname,OUTDDS=ddname,          x  
      INDDS=ddname
```

この例で、FUNCTION=OP は 1 行に収められた 1 つの完成したステートメントであり、FUNCTION=DX... は 2 行に収められた 1 つの完成したステートメントです。

あるパラメーターがいくつかの値を括弧で囲んで持つことができる場合、それらの値に定位置はないので、任意の順序で指定することができます。例えば、次のようになります。

```
      ,REQUEST=(SUMM,MSGALL)  
      ,REQUEST=(MSGALL,SUMM,STATS)
```

各ユーティリティー制御ステートメントは、FUNCTION キーワードを含んでいなければなりません。このキーワードは次のように定義します。

- FUNCTION=xx、ただし、xx は次のとおりです。
  - OP** オプション・ステートメント
  - DR** HD 再編成再ロード・ユーティリティー
  - DU** HD 再編成アンロード・ユーティリティー
  - DX** HD 再編成アンロード・ユーティリティーと再ロード・ユーティリティーの組み合わせ
  - IL** ユーザーの初期ロード・プログラム

<b>IM</b>	バッチ・データベース・イメージ・コピー・ユーティリティ
<b>PR</b>	接頭部解決ユーティリティ
<b>PU</b>	接頭部更新ユーティリティ
<b>RR</b>	副次索引再ロード
<b>RU</b>	副次索引アンロード
<b>SN</b>	データベース・スキャン・ユーティリティ
<b>SR</b>	HISAM 再編成再ロード・ユーティリティ
<b>SU</b>	HISAM 再編成アンロード・ユーティリティ
<b>SX</b>	HISAM 再編成アンロードと再ロード・ユーティリティの組み合わせ
<b>ZB</b>	データベース ZAP
<b>ZM</b>	モジュール ZAP

- これらの機能は、機能ユーティリティやデータベース初期ロードあるいはその両方を呼び出すための要求です。

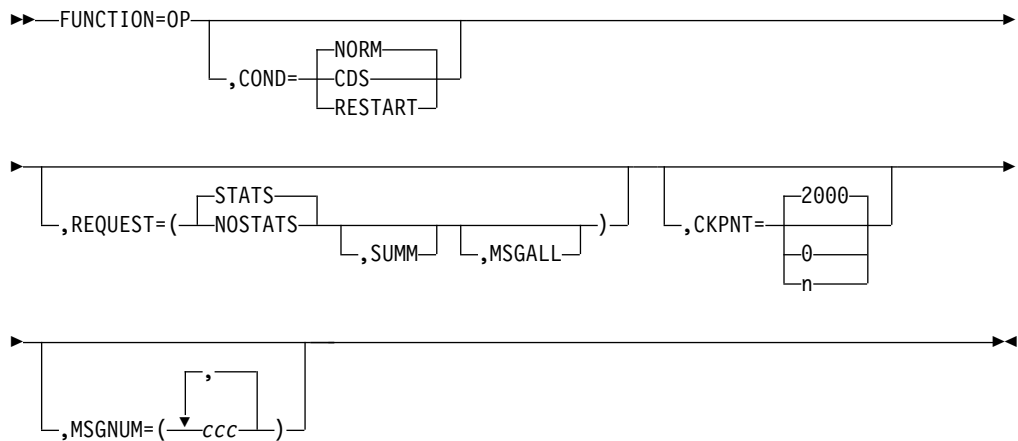
サブセクション:

- 『FUNCTION=OP ステートメント』
- 451 ページの 『FUNCTION=DR ステートメント』
- 452 ページの 『FUNCTION=DU ステートメント』
- 454 ページの 『FUNCTION=DX ステートメント』
- 457 ページの 『FUNCTION=IL ステートメント』
- 459 ページの 『FUNCTION=IM ステートメント』
- 460 ページの 『FUNCTION=PR ステートメント』
- 462 ページの 『FUNCTION=PU ステートメント』
- 464 ページの 『FUNCTION=RR ステートメント』
- 465 ページの 『FUNCTION=RU ステートメント』
- 468 ページの 『FUNCTION=SN ステートメント』
- 470 ページの 『FUNCTION=SR ステートメント』
- 473 ページの 『FUNCTION=SU ステートメント』
- 475 ページの 『FUNCTION=SX ステートメント』
- 478 ページの 『FUNCTION=ZB ステートメント』
- 480 ページの 『FUNCTION=ZM ステートメント』

## FUNCTION=OP ステートメント

UCF はデータベース事前再編成ユーティリティを代行するので、このステートメントにある REQUEST= オプションは、報告書作成を制御するためにデータベース接頭部解決によって使用されます。そのために、REQUEST キーワードは主に事前再編成/接頭部解決のために、また二次的に UCF デフォルトのためにコーディングします。

ステートメントの形式は次のとおりです。



この制御ステートメントは、UCF 実行に関する、いくつかの指定を行います。

#### COND=

UCF のこの実行のときの処理のタイプを指定します。COND= は、UCF の実行ごとに 1 回だけ使用できます。COND=RESTART は、再始動処理が必要であることを意味します。COND=CDS は、データ要求の有効性検査を目的とした制御データ・セットを、入力制御ステートメントから作成します。COND=NORM はデフォルトであり、通常の処理を意味します。

#### REQUEST=STATS|NOSTATS

STATS は、各機能ユーティリティーが処理を完了したときに統計を印刷することを指定します。STATS はデフォルトです。NOSTATS は、統計を印刷しないことを指定します。UCF の初回の実行で REQUEST=NOSTATS が指定された場合、後続の再始動ステップでもすべて、REQUEST=NOSTATS が指定されていなければなりません。このパラメーターは、接頭部解決によっても、その出力報告書を制御するために使用されます。

#### REQUEST=SUMM

機能ユーティリティーが処理を完了したときに統計の要約を印刷することを指定します。このパラメーターは、接頭部解決によっても、その出力報告書を制御するために使用されます。

#### REQUEST=MSCALL

すべての A および W タイプのメッセージを異常終了要求として設定することを指定します。このパラメーターは、診断エイドとしてだけ使用されます。

#### CKPNT=

UCF の実行時にデフォルト指定として使用するチェックポイント間隔を指定します。指定する値は、0 から 999999 までの範囲のものでなければなりません。どの機能制御ステートメントの場合も、CKPNT キーワードの指定がないと、この値がデフォルトとして使用されます。FUNCTION=OP に CKPNT の指定がないと、どの UCF 制御ステートメントでも、2000 がデフォルトの CKPNT 値として使用されます。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。デフォルト値については、それぞれ該当する制御ステートメントを参照してください。

#### MSGNUM=

これは、エラー・ポイント異常終了フラグを設定するために使用されます。値

ccc は、メッセージ ID DFScccl から抜き出されたメッセージ番号です。この番号は、後続の処理でこのメッセージが出された場合に異常終了を引き起こす条件を設定するのに使用されます。

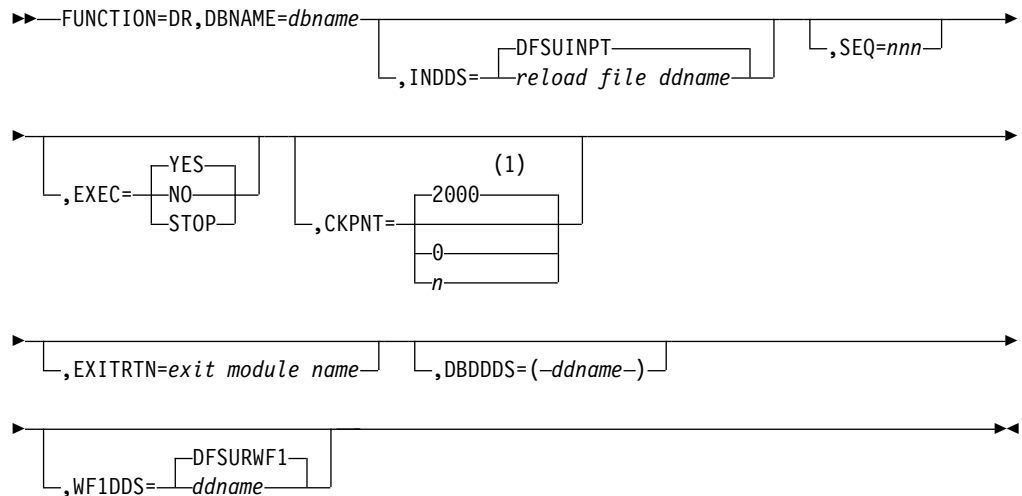
## FUNCTION=DR ステートメント

この制御ステートメントは、UCF に HD 再編成再ロード・ユーティリティ (DFSURGL0) を実行させて、HD 再編成アンロード・ユーティリティ (DFSURGU0) によって作成されたデータ・セットから HDAM または HIDAM データベースを再ロードするためのものです。

HD 再編成再ロード・ユーティリティ・ステップは、正常終了した時点で、内部的にバッチ・データベース・イメージ・コピー・ユーティリティ要求を生成します。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=DR 制御ステートメントの形式は、次のとおりです。



注:

- 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

### DBNAME=

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

### INDDS=

再ロードするデータが入っている入力データ・セットの DD 名を指定します。これは、FUNCTION=DU 制御ステートメント (HD 再編成アンロード・ユーティリティ) の OUTDDS キーワードで指定されたデータ・セットです。INDDS キーワードはデータベース名を指定してはならず、この制御ステートメントに指定できるのは 1 回だけです。DD 名の指定がない場合、デフォルトとして DFSUINPT が使用されます。

**SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

**EXEC=**

この制御ステートメントを実行するかどうかを指定します。EXEC=STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。

EXEC=YES と指定した場合、この機能は処理されます。EXEC=NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

**CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、ルート・セグメントの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。CKPNT の指定がない場合、デフォルトは、FUNCTION=OP 制御ステートメントの CKPNT に指定された値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

**EXITRTN=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていないなければなりません。

**DBDDDS=**

すべてのデータベース・データ・セットがこの機能の実行前に定義されているかを調べることができるようにします。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。指定された DD 名のいずれにも DD ステートメントがない場合は、メッセージが出され、UCF を終了させるために再始動準備が開始されます。

**WF1DDS=**

接頭部解決ユーティリティーへの入力として渡される出力作業データ・セットを定義します。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。デフォルト (DFSURWF1) 以外の DD 名を使用する場合、その DD 名は、UCF の実行の中でのこのデータ・セットへのすべての参照について、矛盾があってはなりません。

**FUNCTION=DU ステートメント**

この制御ステートメントは、UCF に HD 再編成アンロード・ユーティリティーを実行させて、HDAM または HIDAM データベースを QSAM 形式のデータ・セットへアンロードするためのものです。DBD が変更されており、論理関係を含んでいる場合、次のいずれかの条件に該当するときは、アンロード機能の後で UCF の実行を停止させなければなりません。

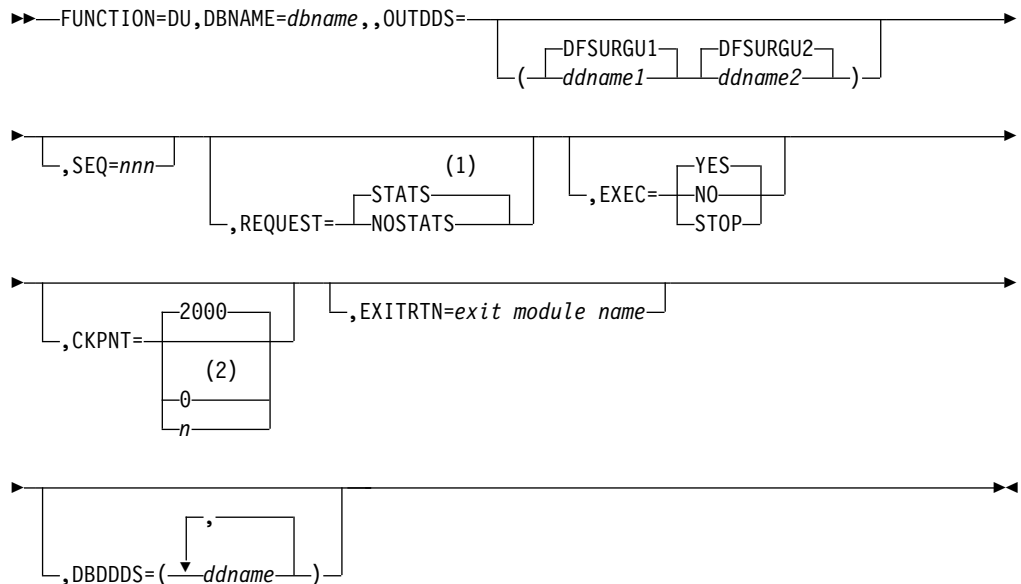
- カウンター、LT、または LP ポインターのいずれかが変更された。
- 論理関係にかかわりを持つセグメントのレベルが、セグメントの追加または削除により変更された。
- DBD 名が変更されたが、その DBD は論理関係を含んでいる。



UCF の実行を停止するには、最後のアンロード機能で EXEC=STOP パラメーターを使用してください。新しい DBD が生成された後で、UCF を (再始動ではなく) 新たに実行して、再ロードを行う必要があります。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=DU 制御ステートメントの形式は、次のとおりです。



注:

- 1 STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。
- 2 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

#### DBNAME=

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

#### OUTDDS= (DFSURGU1 ddname-1) | (DFSURGU2 ddname-2)

HD 再編成アンロード・ユーティリティの 1 次または 2 次データ・セットを定義する DD 名を 2 つまで指定します。これらのデータ・セットは、テープまたは直接アクセス装置に置いておくことができます。(複数の出力コピーを指定すると、一方のコピーで永続エラーが発生しても、もう一方のボリュームでジョブが正常に終了するまで続けられるという利点があります。) UCF の 1 回の実行に複数の DU 機能が含まれている場合、各機能が固有の出力データ・セットを指定していなければなりません。ddname-1 の指定がない場合、デフォルトの名前である DFSURGU1 が指定されているものと見なされます。

ddname-2 の指定がない場合は、デフォルトの名前である DFSURGU2 が使用されているものと見なされます。一時データ・セットは、イメージ・コピー出力データ・セットには使用できません。

#### **SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

#### **REQUEST=**

STATS は、この機能ユーティリティーが処理を完了した後で統計を印刷することを指定します。NOSTATS は、統計を印刷しないことを指定します。STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。

#### **EXEC=**

この制御ステートメントを実行するかどうかを指定します。EXEC=STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。EXEC=YES と指定した場合、この機能は処理されます。EXEC=NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

VSAM データベースを再編成するときは、EXEC=STOP を指定しておかなければなりません。これについては、438 ページの『要件』の項で詳しく説明してあります。

#### **CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、ルート・セグメントの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

#### **EXITRTN=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていないければなりません。

#### **DBDDDS=**

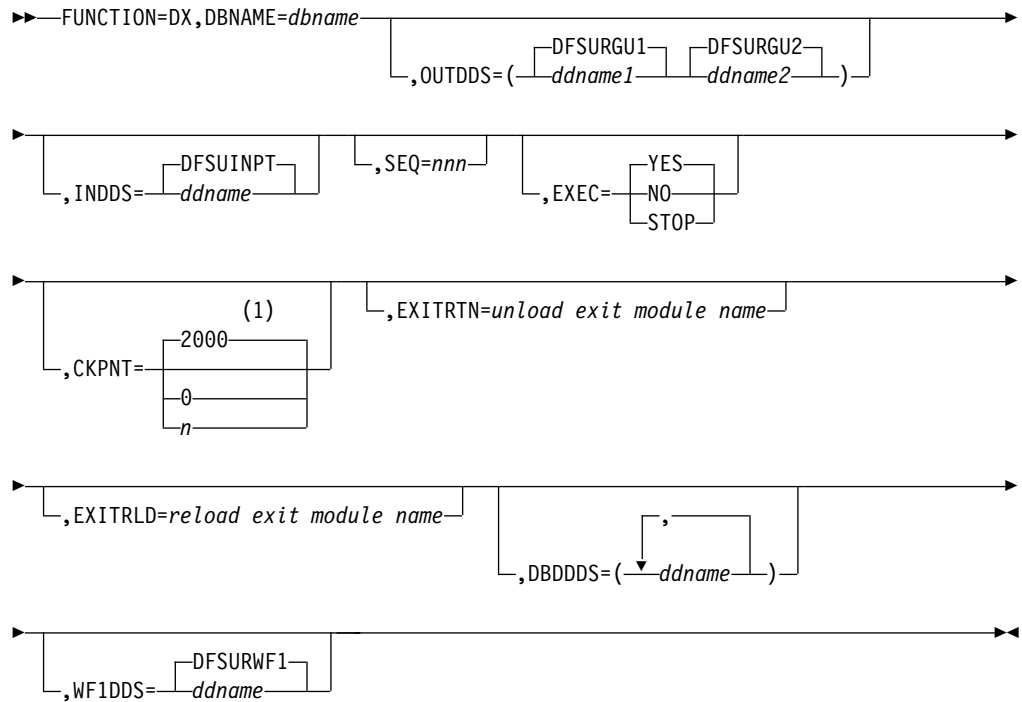
すべてのデータベース・データ・セットがこの機能の実行前に定義されているかを調べることができるようにします。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。指定された DD 名のいずれにも DD ステートメントがない場合は、メッセージが出され、UCF を終了させるために再始動準備が開始されます。

### **FUNCTION=DX ステートメント**

この制御ステートメントは、FUNCTION=DR と FUNCTION=DU の箇所の説明されているように、UCF に HD 再編成アンロード・ユーティリティーと再ロード・ユーティリティーを実行させて、データベースの再編成を行うためのものです。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=DX 制御ステートメントの形式は、次のとおりです。



注:

- 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

#### DBNAME=

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

#### OUTDDS=

HD 再編成アンロード・ユーティリティーの 1 次または 2 次データ・セットを定義する DD 名を 2 つまで指定します。これらのデータ・セットは、テープまたは直接アクセス装置に置いておくことができます。(複数の出力コピーを指定すると、一方のコピーで永続エラーが発生しても、もう一方のボリュームでジョブが正常に終了するまで続けられるという利点があります。) UCF の 1 回の実行に複数の DX 機能が含まれている場合は、各機能が固有の出力データ・セットを指定していなければなりません。ddname-1 の指定がない場合、デフォルトの名前である DFSURGU1 が指定されているものと見なされます。ddname-2 の指定がない場合は、デフォルトの名前である DFSURGU2 が使用されているものと見なされます。

#### INDDS=

再ロードするデータが入っている入力データ・セットを指定します。この DD 名は、データベース再編成のアンロード中は、OUTDDS キーワードに指定した DD 名と一致していなければなりません。データ・セットは、テープか直接ア

アクセス装置のどちらかに置いておかなければなりません。DD 名の指定がない場合、デフォルトの名前である DFSUINPT が指定されているものと見なされます。

#### **SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

#### **EXEC=**

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

EXEC=STOP パラメーターは、制御ステートメント全体に対して有効です。STOP はアンロードと再ロードの両機能で実行されるため、後続の機能の実行を続けるには再始動が必要になります。

VSAM データベースを再編成するときは、EXEC=STOP を指定しておかなければなりません。これについては、438 ページの『要件』の項で詳しく説明してあります。

#### **CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、ルート・セグメントの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

#### **EXITRTN=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、この再編成のアンロード・フェーズ時に制御を渡す出口ルーチンを指定します。アンロード出口モジュール名は、この機能が認識しているライブラリーに入っていないければなりません。

#### **EXITRLD=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、この再編成の再ロード・フェーズ時に制御を渡す出口ルーチンを指定します。再ロード出口モジュール名は、この機能が認識しているライブラリーに入っていないければなりません。

#### **DBDDDS=**

すべてのデータベース・データ・セットがこの機能の実行前に定義されているかを調べることができるようにします。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。指定された DD 名のいずれにも DD ステートメントがない場合は、メッセージが出され、UCF を終了させるために再始動準備が開始されます。

#### **WF1DDS=**

接頭部解決ユーティリティーへの入力として渡される出力作業データ・セットを定義します。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。デフォルト (DFSURWF1) 以外の DD 名を使用する

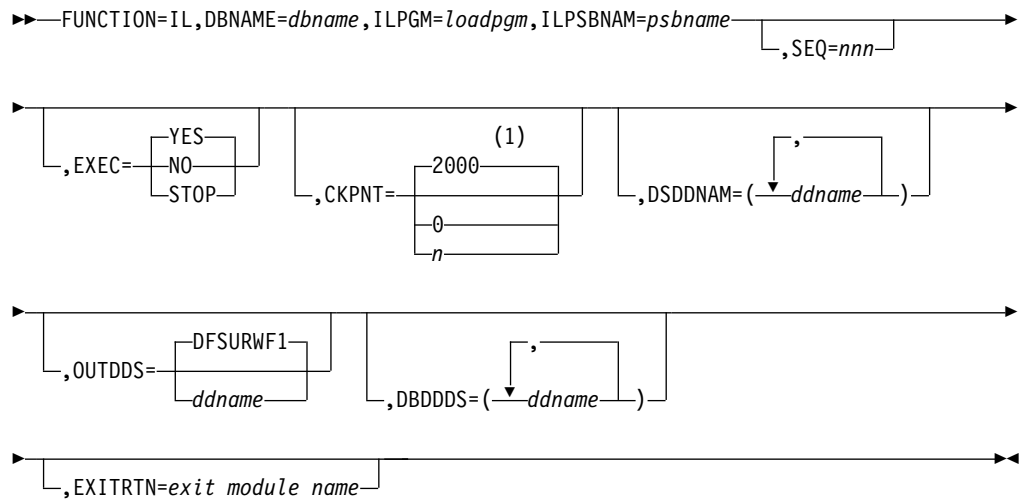
場合、その DD 名は、UCF の実行の中でのこのデータ・セットへのすべての参照について、矛盾があってはなりません。

## FUNCTION=IL ステートメント

この制御ステートメントは、ユーザーが指定した初期ロード・プログラムを接続 (アタッチ) します。これらのプログラムは、UCF のもとで再始動機能を使用することができます。ほとんどの場合、これらのプログラムに少しの変更を加えるだけで、UCF との正しいインターフェースが得られます。

この機能では、DD 名の指定は 20 個までにしてください。

初期ロード・プログラムのための FUNCTION=IL 制御ステートメントの形式は、次のとおりです。



注:

- 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

### DBNAME=

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

### ILPGM=

接続 (アタッチ) する初期ロード・プログラムの名前を定義します。このモジュールは、LINKLIB に置いておくか、JOBLIB または STEPLIB データ・セットに置いておかなければなりません。あるいは、BLDL コマンドおよび ATTACH コマンドにより探索されるため、LINKPACK 区域に置いておかなければなりません。

### ILPSBNAM=

ユーザー初期ロード・プログラムが使用する PSB の名前を定義します。この PSB は、IMS DD ステートメントによって定義されたデータ・セットに置かな

ければなりません。このキーワードは、完結した IL 制御ステートメントごとに 1 回だけ指定できます。どの制御ステートメントの場合も、2 回以上は指定できません。

#### **SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

#### **EXEC=**

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

#### **CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、データベースにロードされたルートの数に等しく、1 から 999999 までの範囲になければなりません。

機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントで CKPNT が指定されていない場合、この機能のデフォルトは 2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

#### **DSDDNAM=**

すべてのデータ・セットがこのプログラムの実行前に定義されているかを調べるように初期ロード機能に指示します。UCF は、この DD 名に対して DEVTYPE マクロを実行します。このデータ・セットの DD ステートメントがない場合、メッセージが出され、UCF を終了させるために再始動準備が開始されます。

#### **OUTDDS=**

接頭部解決ユーティリティへの入力として渡される入力作業データ・セットを定義します。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。デフォルト (DFSURWF1) 以外の DD 名を使用する場合、その DD 名は、UCF の実行の中でのこのデータ・セットへのすべての参照について、矛盾があってはなりません。

#### **DBDDDS=**

すべてのデータベース・データ・セットがこの機能の実行前に定義されているかを調べます。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。指定された DD 名のいずれにも DD ステートメントがない場合は、メッセージが出され、UCF を終了させるために再始動準備が開始されます。

#### **EXITRTN=**

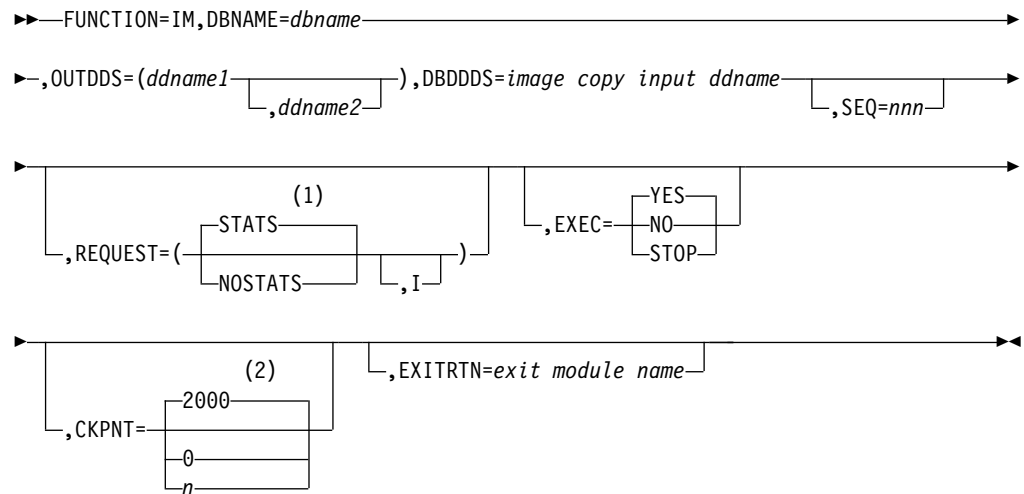
ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていなければなりません。

## FUNCTION=IM ステートメント

この制御ステートメントは、UCF にバッチ・データベース・イメージ・コピー・ユーティリティー (DFSUDMP0) を実行させて、データベースにデータ・セットの出力コピーを作成するためのものです。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=IM 制御ステートメントの形式は、次のとおりです。



注:

- 1 STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。
- 2 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

### DBNAME=

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

### OUTDDS=

イメージ・コピー出力データ・セットにコピーを最大 2 つ指定します。

*ddname-1* は 1 次出力データ・セットを定義し、*ddname-2* は 2 次データ・セットを定義します。それらのデータ・セットは、テープまたは直接アクセス装置に置いておくことができます。(複数の出力コピーを指定すると、一方のコピーで永続エラーが発生しても、もう一方のボリュームでジョブが正常に終了するまで続けられるという利点があります。) 複数の FUNCTION=IM ステートメントが指定されている場合、イメージ・コピーは、OUTDDS DD 名の照合順序に基づく順で行われます。OUTDDS データ・セットが割り振られるか、あるいは新しい世代別データ・セット・グループ (GDG) が作成される場合は、これ

らのデータ・セットは、オープン・エラーを防止するために、DD 名の照合順序で JCL に入れておかなければなりません。

#### **DBDDDS=**

イメージ・コピー処理への入力となるデータベース・データ・セットの DD 名を指定します。このデータ・セットは直接アクセス装置に置いておかなければなりません。また、制御ステートメントごとに 1 つ指定する必要があります。REQUEST=I が指定されている場合、データ・セットは KSDS でなければなりません。

#### **SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの機能制御ステートメントとリンクさせます。

#### **REQUEST=**

STATS は、この機能ユーティリティーが処理を完了した後で統計を印刷することを指定します。NOSTATS は、統計を印刷しないことを指定します。STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。

REQUEST=I は、KSDS の索引のイメージ・コピーを指定します。これが指定される場合、DBDDDS キーワード値は KSDS を参照するものでなければなりません。このイメージ・コピーで使用できるリカバリーは、トラック・リカバリーだけです。

#### **EXEC=**

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

#### **CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、論理レコードの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

#### **EXITRTN=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていなければなりません。

### **FUNCTION=PR ステートメント**

この制御ステートメントは、UCF にデータベース接頭部解決ユーティリティー (DFSURG10) を実行させるためのものです。このユーティリティーは、1 つ以上のデータベースのロード時、再編成時、またはその両方を行っているとき、生成された情報を作業データ・セットに累積します。このユーティリティーから得られる出



力データ・セット DFSURWF3 には、論理関係情報が置かれている各セグメントに適用される 1 つ以上の更新されたレコードが入っています。

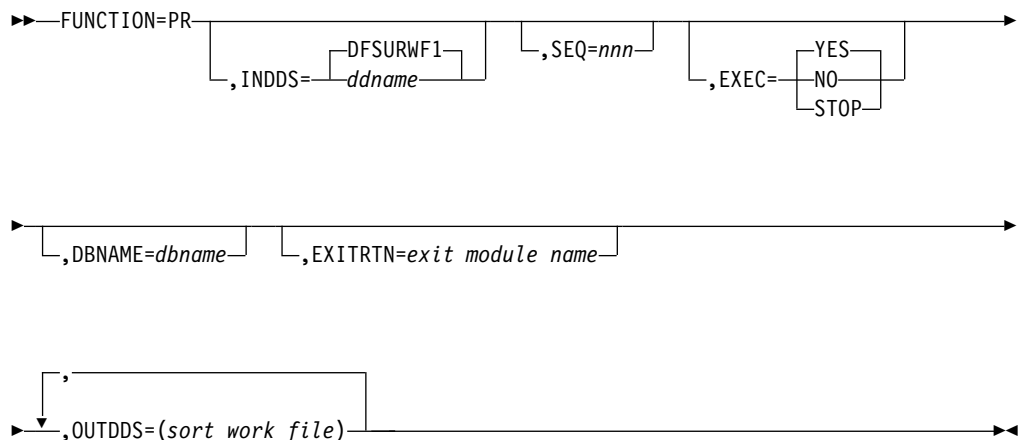
このデータ・セットの内容は、UCF を使用せずに接頭部解決ユーティリティーを実行して生成された同じデータ・セットと比較すると、異なっている場合があります。これは、データベースを、論理子を持っていない論理親となっているデータベース・レコードに論理的に関連付けたときに発生する可能性があります。この場合、一部のポインターは解決を必要とせず、それらのレコードは廃棄されます。非 UCF 制御のユーティリティーを実行するには、事前再編成ユーティリティー (DFSURPR0) 用の DBIL 制御ステートメントをコーディングする必要があるため、2 回のユーティリティー実行の間で廃棄されたレコードの数とタイプに違いが生じる場合があります。ユーティリティーの実行においては、最終的なデータベースの内容はこの違いによっては影響されません。

更新されたレコードは、データベースとセグメント別に物理位置順にソートされています。更新される接頭部フィールドとしては、論理親、論理兄弟、および論理子ポインター・フィールドと論理親に関連するカウンター・フィールドがあります。このプログラムは、指定があれば、副次索引データベースを作成または更新するのに必要な情報が入っている出力データ・セットを作成します。

STATS 報告書と SUMM 報告書の制御は、FUNCTION=OP ステートメントの REQUEST= キーワードで行います。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=PR 制御ステートメントの形式は、次のとおりです。



#### INDDS=

データベース・スキャン・ユーティリティーによって生成された DFSURWF1 またはユーザー定義のデータ・セットを、データベース接頭部解決ユーティリティーへの入力の 1 つとして使用することを指定します。

#### SEQ=

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

#### EXEC

この制御ステートメントを実行するかどうかを指定します。STOP と指定した

場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

**DBNAME=**

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。この機能は、DBNAME パラメーターの指定の有無に関係なく、作業ファイル上のすべてのデータベースに対して実行されます。

**EXITRTN=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていないければなりません。

**OUTDDS=**

すべてのソート用作業ファイル・データ・セットがこの機能の実行前に定義されているかを調べることができるようにします。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。ソート用作業ファイルのいずれにも DD ステートメントが指定されていない場合は、メッセージが出され、再始動準備によって UCF が終了します。

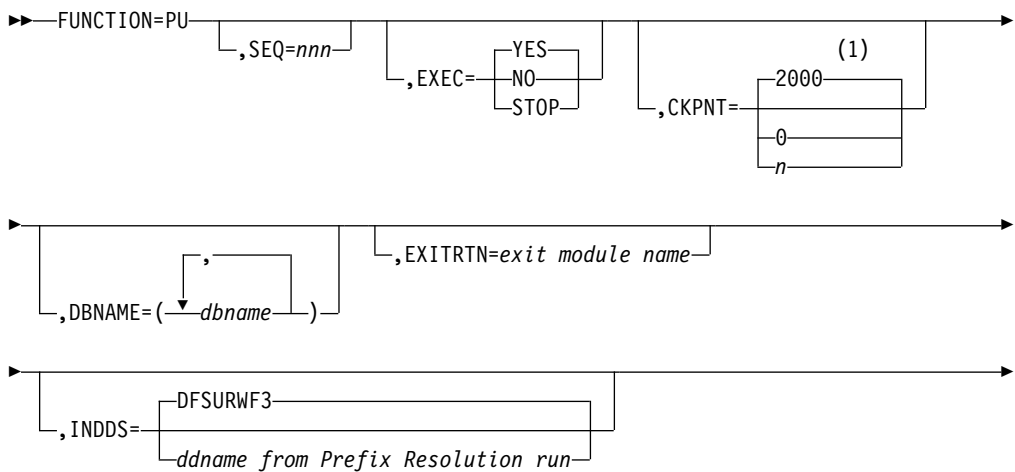
**FUNCTION=PU ステートメント**

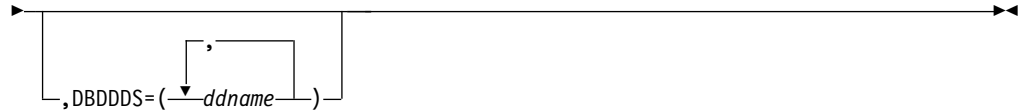
この制御ステートメントは、UCF にデータベース接頭部更新ユーティリティ (DFSURGP0) を実行させるためのものです。このユーティリティは、接頭部解決ユーティリティによって生成された出力を使用して、データベースのロードまたは再編成により接頭部情報が影響を受けた各セグメントの接頭部を更新します。

このステートメントは、通常の処理のとき UCF によって自動的に生成されるので、その使用はオプションです。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=PU 制御ステートメントの形式は、次のとおりです。





注:

- 1 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

#### SEQ=

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

#### EXEC=

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

#### CKPNT=

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、入力作業レコードの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

#### DBNAME=

実行の目的で、また、WTOR のパラメーターを設定するために UCF が必要とするデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。この機能は、DBNAME パラメーターの指定の有無に関係なく、作業ファイル上のすべてのデータベースに対して実行されます。

#### EXITRTN=

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていないなければなりません。

#### INDDS=

入力データ・セットの DD 名を指定します。この特定の UCF 実行にはデフォルト (WF3 データ・セット) が適用されます。接頭部解決の実行がこの UCF の実行の外で行われた場合、データ・セットは、他の DD 名 (ユーザーが指定したもの) を持つことができます。

#### DBDDDS=

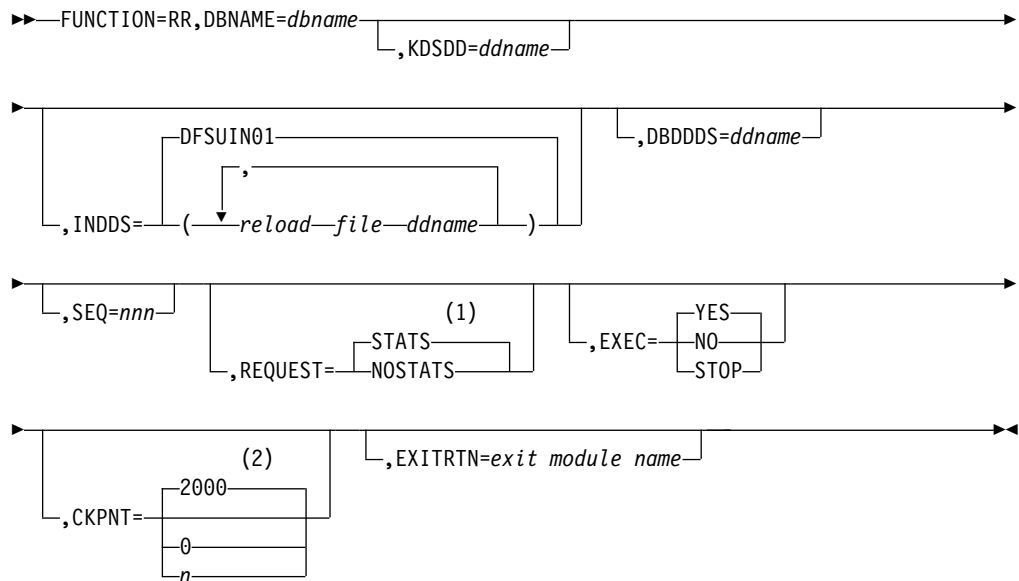
すべてのデータベース・データ・セットがこの機能の実行前に定義されているかを調べることができるようにします。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。指定された DD 名のいずれにも DD ステートメントがない場合は、メッセージが出され、UCF を終了させるために再始動処理が開始されます。

## FUNCTION=RR ステートメント

この制御ステートメントは、UCF に HISAM 再編成再ロード・ユーティリティー (DFSURRL0) を実行させて、副次索引内でメンバーを作成、マージ、あるいは置き換えるためのものです。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=RR 制御ステートメントの形式は、次のとおりです。



注:

- 1 STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。
- 2 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

### DBNAME=

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

### KDSDD=

操作を行うキー順データ・セットの DD 名を指定します。このキーワードは、キー順データ・セットに DD ステートメントがあるかどうかを調べるために使用します。

### INDDS=

再ロードするデータが入っている入力データ・セットを指定します。この DD 名は、データベース再編成のアンロード中は、OUTDDS キーワードに指定した DD 名と一致していなければなりません。このキーワードを指定しない場合、

デフォルトは DFSUIN01 です。このデータ・セットは、テープか直接アクセス装置のどちらかに置いておくことができます。

#### **IBDDS=**

この機能で必要とするオーバーフロー (ESDS) データベース・データ・セットの DD 名を指定します。各制御ステートメントに指定できる DBDDDS は 1 つだけです。

#### **SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの機能制御ステートメントとリンクさせます。

#### **REQUEST=**

STATS は、この機能ユーティリティーが処理を完了した後で統計を印刷することを指定します。NOSTATS は、統計を印刷しないことを指定します。STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。

#### **EXEC=**

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

#### **CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、論理レコードの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CHKPNT=0 の場合、この機能については、チェックポイントはとられません。

#### **EXITRTN=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていなければなりません。

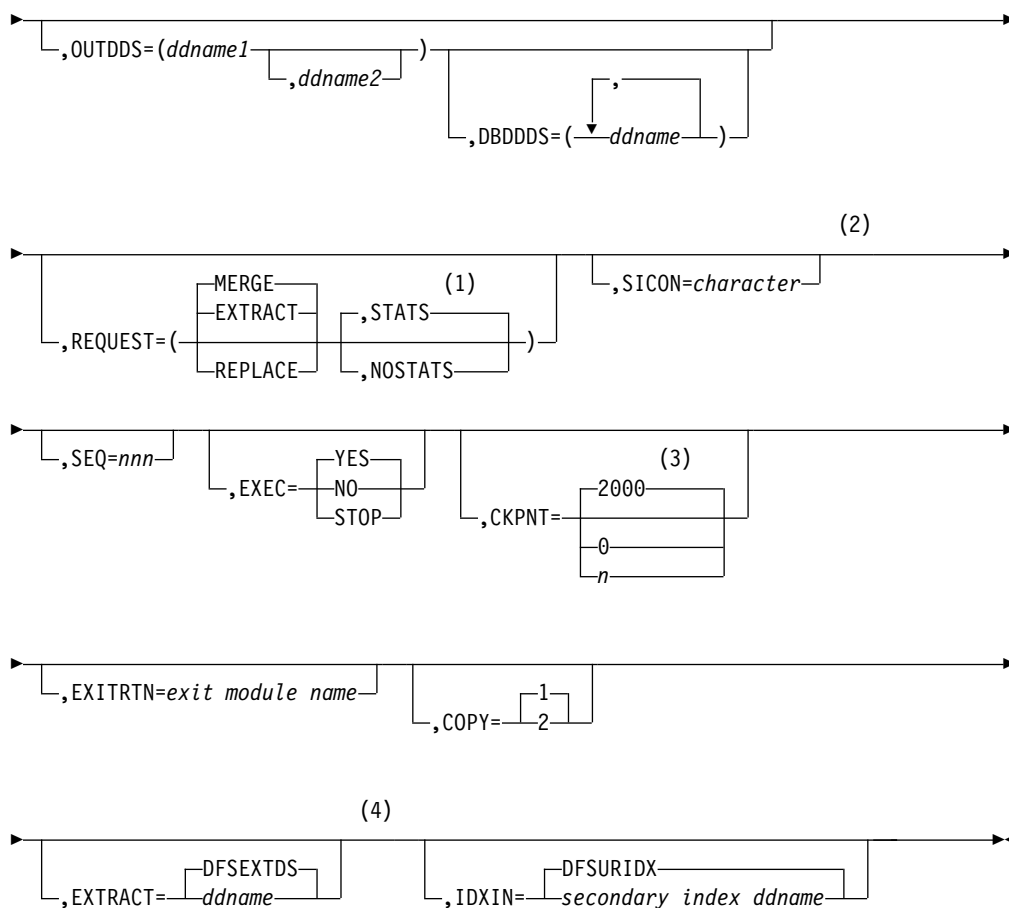
### **FUNCTION=RU ステートメント**

この制御ステートメントは、UCF に HISAM 再編成アンロード・ユーティリティー (DFSURUL0) を実行させて、HISAM 再編成再ロード・ユーティリティーへの入力作業データ・セットを作成するものです。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=RU 制御ステートメントの形式は、次のとおりです。

▶▶—FUNCTION=RU,DBNAME=*dbname*,KDSDD=*ddname*————▶▶



注:

- 1 STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。
- 2 REQUEST=EXTRACT または REQUEST=REPLACE のどちらかが指定された場合に必要です。
- 3 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。
- 4 REQUEST=EXTRACT のときだけ指定できます。

#### DBNAME=

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

#### KDSDD=

再編成するデータベースの VSAM KSDS を定義します。DD 名は、このデータ・セットを記述する DBD に入っている名前と同じものでなければなりません。このキーワードは、HISAM データベースの再編成のたびに DBDDDS キーワードと一緒に使用します。

**OUTDDS=**

副次索引出力データ・セットのコピーを 2 つまで指定します。ddname-1 は 1 次出力データ・セットを定義します。ddname-2 を指定する場合は、COPY キーワードの値が 2 でなければなりません。これらのコピー・データ・セットは、テープまたは直接アクセス装置のどちらかにおいておくことができます。

**DBDDDS=**

すべてのデータベース・データ・セットがこの機能の実行前に定義されているかを調べることができるようにします。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。指定された DD 名のいずれにも DD ステートメントがない場合は、メッセージが出され、UCF を終了させるために再始動準備が開始されます。

**REQUEST=EXTRACT または MERGE または REPLACE**

EXTRACT を指定すると、副次索引 (SICON キーワードにより定義されたもの) は、共用索引データベースからか、あるいは接頭部解決で得られた索引作業データ・セットから抽出されます。REQUEST=EXTRACT を指定する場合は、SICON キーワードと EXTRACT キーワードが必要です。

MERGE を指定すると、データベース初期ロード時または再編成時に (接頭部解決ユーティリティを使用して) 作成された索引作業データ・セットが既存の副次索引にマージされ、そのデータ・セットがなければ副次索引が作成されます。MERGE はデフォルトです。

REPLACE を指定すると、副次索引にあって、SICON キーワードで指定された文字定数に一致するセグメントが、作業データ・セットに見付かった一致するセグメントと置き換えられます。置き換えられる副次索引セグメントを保管しておく必要があるときは、置き換えを行う前に抽出操作を実行してください。置き換え操作には、SICON キーワードが必要です。

**REQUEST=STATS または NOSTATS またはオプション・ステートメントでの指定**

STATS は、この機能ユーティリティが処理を完了した後で統計を印刷することを指定します。NOSTATS は、統計を印刷しないことを指定します。STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。

**SICON=**

共用副次索引の DBD 生成時に定義された 1 バイト定数を指定します。このキーワードは、この制御ステートメントで REQUEST=EXTRACT または REQUEST=REPLACE が定義されている場合に必要です。

**SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

**EXEC=**

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

#### **CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、ルート・セグメントの数と等しくなっています。機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

#### **EXITRTN=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていないければなりません。

#### **COPY=**

出力データ・セットのコピーを複数指定できるようにします。各制御ステートメントに指定できる COPY は 1 つだけです。COPY=1 と指定すると、コピーが 1 つだけ得られます。これはデフォルトです。COPY=2 と指定すると、追加の出力コピーが得られます。COPY=2 は、OUTDDS キーワードに ddname-2 が指定されている場合に指定します。(複数の出力コピーを指定すると、一方のコピーで永続エラーが発生しても、もう一方のボリュームでジョブが正常に終了するまで続けられるという利点があります。)

#### **EXTRACT=**

EXTRACT データベース・データ・セットの DD 名を定義します。REQUEST=EXTRACT が使用された場合は、このステートメントを指定しなければなりません。DD 名の指定がない場合、DFSEXTDS がデフォルトです。

#### **IDXIN=**

接頭部解決ユーティリティーから得られ、副次索引情報が含まれている出力データ・セット (DFSURIDX) を記述します。このキーワードの指定は必須であり、各制御ステートメントに 1 つだけ指定できます。このキーワードは、HISAM 再編成アンロード・ユーティリティーの索引 DD ステートメントに対応しています。DD 名の指定がない場合、DFSURIDX がデフォルトです。

### **FUNCTION=SN ステートメント**

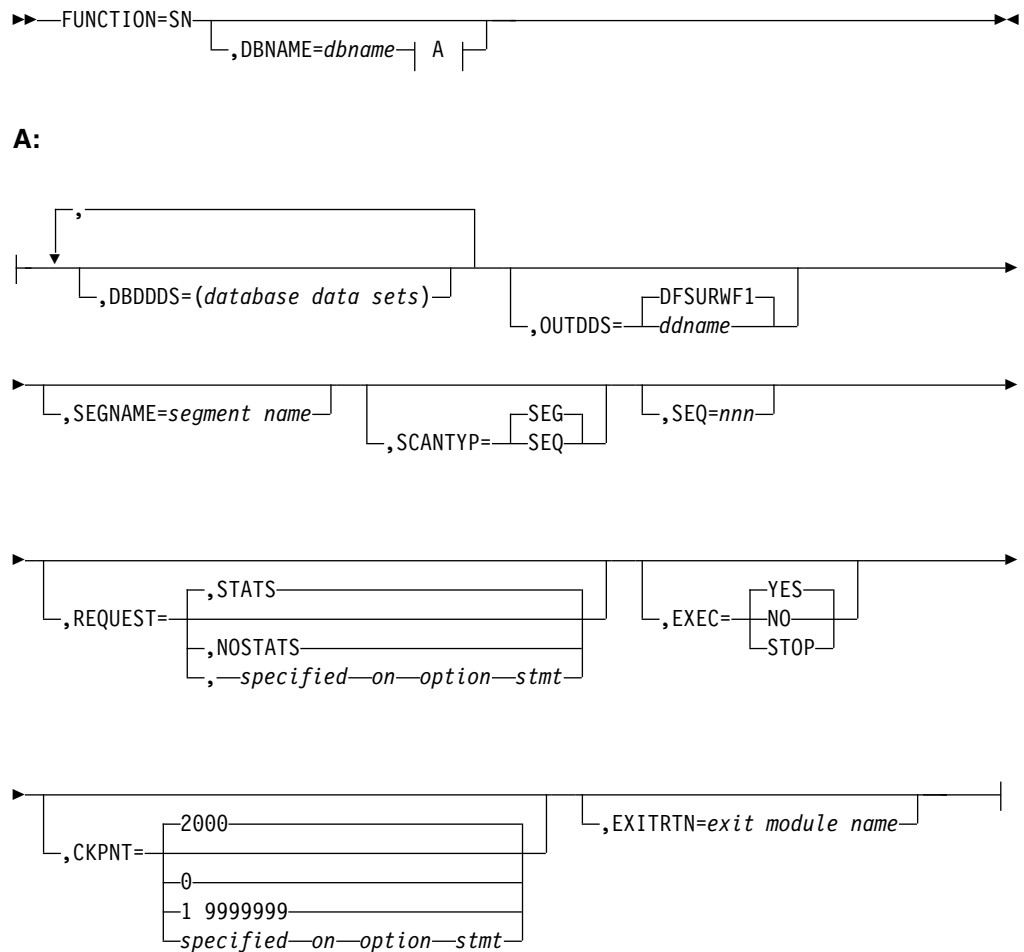
この制御ステートメントは、UCF にデータベース・スキャン・ユーティリティー (DFSURGS0) を実行させるためのものです。このユーティリティーは、他のデータベースのロードまたは再編成により影響を受けた論理関係を含んでいる、再編成されていないデータベースをスキャンするためのものです。また、データベース接頭部解決ユーティリティーによって使用される出力作業データ・セットも生成します。

SCAN 機能は、初期ロードと HD 再ロードのときに実行されます。SCAN が必要でないときは、FUNCTION=SN ステートメントで EXEC=NO が指定されていない限り、DFSURWF1 DD ステートメントがなければなりません。

この機能では、DD 名の指定は 20 個までにしてください。

このステートメントは、通常の処理のとき UCF によって自動的に生成されるので、その使用はオプションです。FUNCTION=SN 制御ステートメントの形式は、次のとおりです。





**DBNAME=**

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

**DBDDDS=**

すべてのデータ・セットがこの機能の実行前に定義されているかどうかを調べます。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。指定された DD 名のいずれにも DD ステートメントがない場合は、メッセージが出され、UCF を終了させるために再始動準備が始まります。

**OUTDDS=**

論理関係を解決するために使用される出力データ・セットを定義します。このデータ・セットは、接頭部解決ユーティリティーへの入力として使用されます。デフォルト (DFSURWF1) 以外の DD 名を使用する場合、その DD 名は、UCF の実行の中でのこのデータ・セットへのすべての参照について、矛盾があつてはなりません。

**SEGNAME=**

スキャンして探すセグメント名を定義します。このキーワードは、この同じ制御ステートメントの中で DBNAME= キーワードと一緒に使用されます。これは、完結した制御ステートメントごとに 1 つだけ指定できます。

**SCANTYPE=**

この同じ制御ステートメントと関連付けられた DBNAME でスキャンを行う順序を定義します。このキーワードは、完結した制御ステートメントごとに 1 つだけ指定できます。値 SEQ は、検索が非修飾 GN 呼び出しを使用してデータベースの先頭から行われていくことを意味します。値 SEG は、検索がセグメント名で修飾された GN 呼び出しを使用してデータベースの先頭から行われていくことを意味します。

**SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

**REQUEST=**

STATS は、この機能ユーティリティーが処理を完了した後で統計を印刷することを指定します。NOSTATS は、統計を印刷しないことを指定します。

STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。STATS はデフォルトです。

**EXEC=**

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

**CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、読み取られたルート・セグメントの数に等しい呼び出しの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

**EXITRTN=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていなければなりません。

**FUNCTION=SR ステートメント**

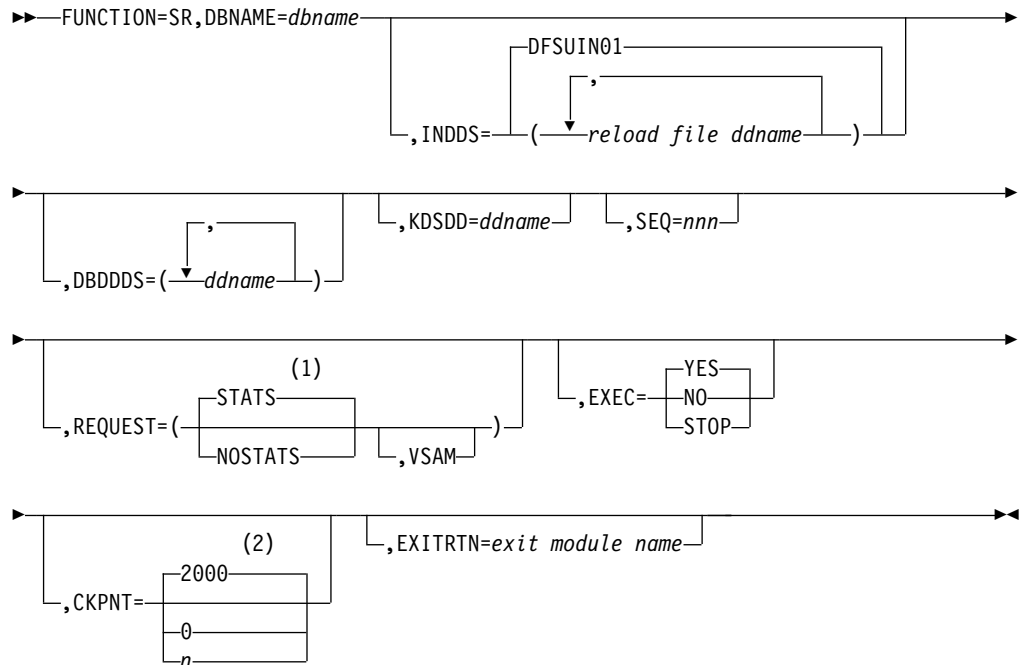
この制御ステートメントは、UCF に HISAM 再編成再ロード・ユーティリティー (DFSURRL0) を実行させるためのものです。このユーティリティーを使用すると、HISAM 再編成アンロード・ユーティリティーによって作成された QSAM 形式のデータ・セットから、HISAM または HIDAM 1 次索引データベースを再ロードすることができます。

HISAM データベースの再編成は、HD アンロード/再ロード・ユーティリティーではなく、HISAM アンロード/再ロード・ユーティリティーを使用すると、大幅に高速化します。

HISAM アンロード/再ロード・ユーティリティーは、論理レコード長およびブロック・サイズの変更以外の構造上の変更を行うために使用することはできません。ただし、HD アンロード/再ロード・ユーティリティーを使用すると、データベースに対して構造上の変更を行うことができます。HISAM 再編成アンロード/再ロード・ユーティリティーを使用して、副次索引の索引が付けられていたり、あるいは論理関係で使用される直接アドレス・ポインターを持つセグメントが入っている HISAM データベースを再編成することはできません。その場合には、HD 再編成アンロード/再ロード・ユーティリティーを使用しなければなりません。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=SR 制御ステートメントの形式は、次のとおりです。



注:

- 1 STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。
- 2 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

**DBNAME=**

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

**INDDS=**

再ロードするデータが入っている入力データ・セットを指定します。これは、

FUNCTION=SU 制御ステートメント (HISAM 再編成アンロード・ユーティリティ) から作成されたデータ・セットです。このキーワードを指定しない場合、デフォルトは DFSUIN01 です。

**DBDDDS=**

再ロードするデータベース・データ・セットがあるかどうかを検査するために使用します。

**KDSDD=**

再ロードする VSAM KSDS 出力データ・セットを定義します。この DD 名は、HISAM 再編成アンロード (FUNCTION=SU) 時に KSDSD キーワードに使用された DD 名と同じものでなければなりません。

**SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

**REQUEST=**

STATS は、この機能ユーティリティが処理を完了した後で統計を印刷することを指定します。NOSTATS は、統計を印刷しないことを指定します。STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。

**VSAM**

OSAM 入力再ロード・コピーを、VSAM データ・セットに再ロードすることを指定します。

FUNCTION=SR 制御ステートメントで REQUEST=VSAM キーワードを使用するときは、次のような制約があります。

- OSAM データベースがアンロードされ、しかも REQUEST=VSAM キーワードが指定された場合、出力は VSAM 形式になります。アクセス方式サービス・ユーティリティを実行して必要なデータ・セットを作成し、必要な DBD 変更をすべて行ってからでないと、HISAM 再ロード・ユーティリティは実行できません。
- REQUEST=VSAM は、OSAM から VSAM へ変換を行うときだけ使用できます。
- VSAM データベースがアンロードされるときは、再ロードされるデータベースは、REQUEST= キーワード・ステートメントで何を指定したかに関係なく、VSAM です。アクセス方式サービス・ユーティリティで元のデータ・セットを消去して再割り振りするか、あるいはアクセス方式サービス・ユーティリティにより新しいデータ・セット名を作成してからでないと、HISAM 再ロード・ユーティリティは実行できません。

**EXEC=**

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

**CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定し

ます。この値は、論理レコードの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。CKPNT の指定がない場合、デフォルトは、FUNCTION=OP 制御ステートメントの CKPNT に指定された値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

**EXITRTN=**

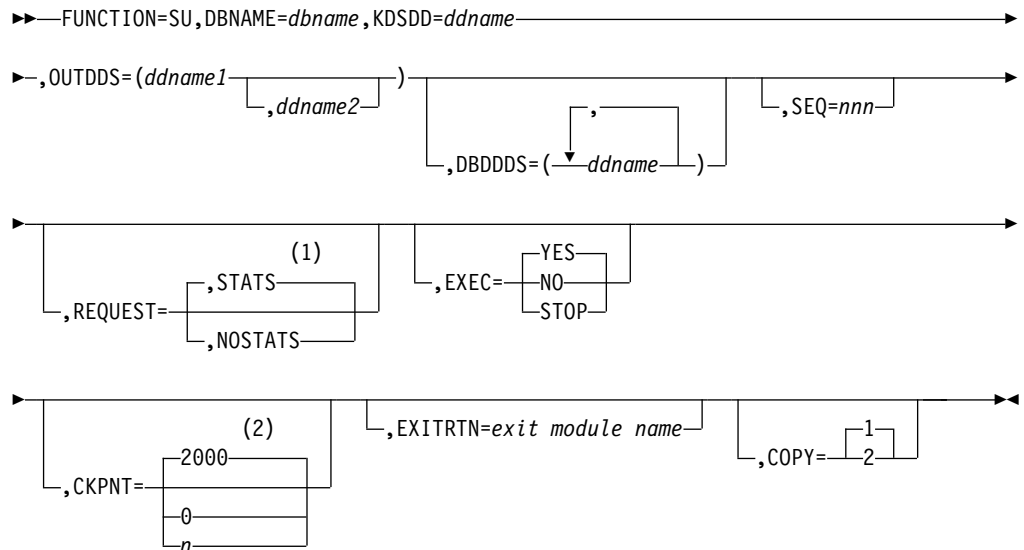
ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていないとできません。

**FUNCTION=SU ステートメント**

この制御ステートメントは、UCF に HISAM 再編成アンロード・ユーティリティ (DFSURUL0) を実行させるためのものです。このユーティリティは、HISAM データベースをアンロードし、再編成された出力を作成します。この出力は、データベース・リカバリー・ユーティリティまたは HISAM 再編成再ロード・ユーティリティのいずれか一方への入力として使用することができます。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=SU 制御ステートメントの形式は、次のとおりです。



注:

- 1 STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。
- 2 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

**DBNAME=**

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

**KDSDD=**

再編成するデータベースの VSAM KSDS を定義します。DD 名は、このデータ・セットを記述する DBD に入っている名前と同じものでなければなりません。このキーワードは、HISAM データベースの再編成のたびに DBDDDS キーワードと一緒に使用します。

**OUTDDS=**

アンロード・データ・セットのコピーを 2 つまで指定します。ddname-1 は 1 次出力データ・セットを定義し、ddname-2 は 2 次データ・セットを定義します。ddname-2 を指定する場合は、COPY キーワードの値が 2 でなければなりません。これらのコピー・データ・セットは、テープまたは直接アクセス装置のどちらかにおいておくことができます。

**DBDDDS=**

すべてのデータベース・データ・セットがこの機能の実行前に定義されているかを調べるのに使用します。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。指定された DD 名のいずれにも DD ステートメントがない場合は、メッセージが出され、UCF を終了させるために再始動準備が始まります。

**SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

**REQUEST=**

STATS は、この機能ユーティリティーが処理を完了した後で統計を印刷することを指定します。NOSTATS は、統計を印刷しないことを指定します。

STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。

**EXEC=**

この制御ステートメントを実行するかどうかを指定します。EXEC=STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。EXEC=YES と指定した場合、この機能は処理されます。EXEC=NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

VSAM データベースを再編成するときは、EXEC=STOP を指定しておかなければなりません。

**CKPNT=**

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、ルート・セグメントの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。

FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

**EXITRTN=**

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていないければなりません。

**COPY=**

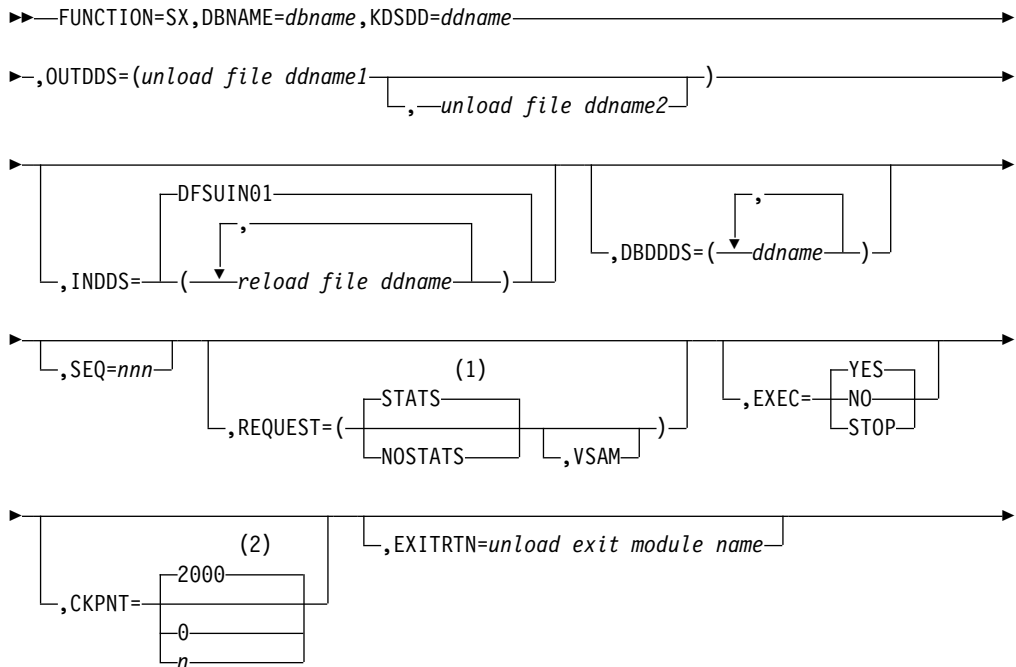
出力データ・セットのコピーを複数指定できるようにします。各制御ステートメントに指定できる COPY は 1 つだけです。COPY=1 と指定すると、コピーが 1 つだけ得られます。これはデフォルトです。COPY=2 と指定すると、追加の出力コピーが得られます。COPY=2 は、OUTDDS キーワードに ddname-2 が指定されている場合に指定します。(複数の出力コピーを指定すると、一方のコピーで永続エラーが発生しても、もう一方のボリュームでジョブが正常に終了するまで続けられるという利点があります。)

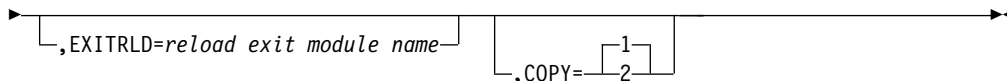
**FUNCTION=SX ステートメント**

この制御ステートメントは、UCF に、HISAM データベースを再編成するために HISAM 再編成アンロードおよび再ロード・ユーティリティーを実行させるためのものです。FUNCTION=SX は、FUNCTION=SU と FUNCTION=SR の指定を結合したものです。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=SX 制御ステートメントの形式は、次のとおりです。





注:

- 1 STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。
- 2 機能で CKPNT の指定がない場合、デフォルトは FUNCTION=OP 制御ステートメントに指定された CKPNT 値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは 2000 です。

**DBNAME=**

この機能を適用するデータベースを指定します。値 *dbname* は DBD ライブラリーのメンバーとして存在する DBD 名であり、この名前を持つ実際のデータベースでもあります。

**KDSDD=**

再編成するデータベースの VSAM KSDS を定義します。DD 名は、このデータ・セットを記述する DBD に入っている名前と同じものでなければなりません。このキーワードは、HISAM データベースの再編成のたびに DBDDDS キーワードと一緒に使用します。

**OUTDDS=**

アンロード・データ・セットのコピーを 2 つまで指定します。ddname-1 は 1 次出力データ・セットを定義し、ddname-2 は 2 次データ・セットを定義します。ddname-2 を指定する場合は、COPY キーワードの値が 2 でなければなりません。これらのコピー・データ・セットは、テープまたは直接アクセス装置のどちらかにおいておくことができます。

**INDDS=**

再ロードするデータが入っている入力データ・セットを指定します。これは、FUNCTION=SU 制御ステートメント (HISAM 再編成アンロード・ユーティリティ) から作成されたデータ・セットです。このキーワードを指定しない場合、デフォルトは DFSUIN01 です。

**DBDDDS=**

すべてのデータベース・データ・セットがこの機能の実行前に定義されているかを調べるのに使用します。UCF は、どの DD 名に対しても DEVTYPE マクロを実行します。指定された DD 名のいずれにも DD ステートメントがない場合は、メッセージが出され、UCF を終了させるために再始動準備が開始されます。

**SEQ=**

突き合わせ順序 ID を用いて、ZAP 機能 (ZB と ZM) をこの制御ステートメントとリンクさせます。

**REQUEST=**

STATS は、この機能ユーティリティが処理を完了した後で統計を印刷することを指定します。NOSTATS は、統計を印刷しないことを指定します。STATS と NOSTATS のどちらも指定されていない場合、デフォルトは、



FUNCTION=OP 制御ステートメントで指定された値です。FUNCTION=OP 制御ステートメントに STATS も NOSTATS も指定されていない場合、デフォルトは STATS です。

#### VSAM

OSAM 入力再ロード・コピーを、VSAM データ・セットに再ロードすることを指定します。

FUNCTION=SX 制御ステートメントで REQUEST=VSAM キーワードを使用するときは、次のような制約があります。

- OSAM データベースがアンロードされ、しかも REQUEST=VSAM キーワードが指定された場合、出力は VSAM 形式になります。HISAM 再ロード・ユーティリティーは、アクセス方式サービス・ユーティリティーを実行して必要なデータ・セットを作成し、必要な DBD 変更をすべて行ってから、実行してください。
- REQUEST=VSAM は、OSAM から VSAM へ変換を行うときだけ使用できます。
- VSAM データベースがアンロードされる場合は、再ロードされるデータベースは、REQUEST= キーワード・ステートメントで何を指定したかに関係なく、VSAM です。アクセス方式サービス・ユーティリティーで元のデータ・セットを消去して再割り振りするか、あるいはアクセス方式サービス・ユーティリティーにより新しい DSNNAME を作成してからでないと、HISAM 再ロード・ユーティリティーは実行できません。

#### EXEC=

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

EXEC=STOP パラメーターは、制御ステートメント全体に対して有効です。STOP はアンロードと再ロードの両機能で実行されるため、後続の機能の実行を続けるには再始動が必要になります。

VSAM データベースを再編成するときは、EXEC=STOP を指定しておかなければなりません。

#### CKPNT=

この機能の実行中に使用される、ユーザー指定のチェックポイント間隔を指定します。この値は、ルート・セグメントと論理レコードの数に等しく、1 から 999999 までの範囲にあるものでなければなりません。CKPNT の指定がない場合、デフォルトは、FUNCTION=OP 制御ステートメントの CKPNT に指定された値です。FUNCTION=OP 制御ステートメントに CKPNT が指定されていない場合のデフォルトは、2000 です。CKPNT=0 は、この機能については、チェックポイントをとらないことを意味します。

#### EXITRTN=

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、この再編成のアンロード・フェーズ時に制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていなければなりません。

### EXITRLD=

ユーザーがレコードを調べたり、統計をコンパイルしたりできるようにするために、この再編成の再ロード・フェーズ時に制御を渡す出口ルーチンを指定します。出口モジュール名は、この機能が認識しているライブラリーに入っていないとあればなりません。

### COPY=

出力データ・セットのコピーを複数指定できるようにします。各制御ステートメントに指定できる COPY は 1 つだけです。COPY=1 と指定すると、コピーが 1 つだけ得られます。これはデフォルトです。COPY=2 と指定すると、追加の出力コピーが得られます。COPY=2 は、OUTDDS キーワードに ddname-2 が指定されている場合に指定します。(複数の出力コピーを指定すると、一方のコピーで永続エラーが発生しても、もう一方のボリュームでジョブが正常に終了するまで続けられるという利点があります。)

### FUNCTION=ZB ステートメント

この制御ステートメントは、データベース・ブロックに対して ZAP を行い、エラーを訂正させたり、異常終了を強制的に起こすためのものです。サポートされるのは、保守援助機能 SPZAP プログラムの VERIFY 制御ステートメントと REP 制御ステートメントだけです。

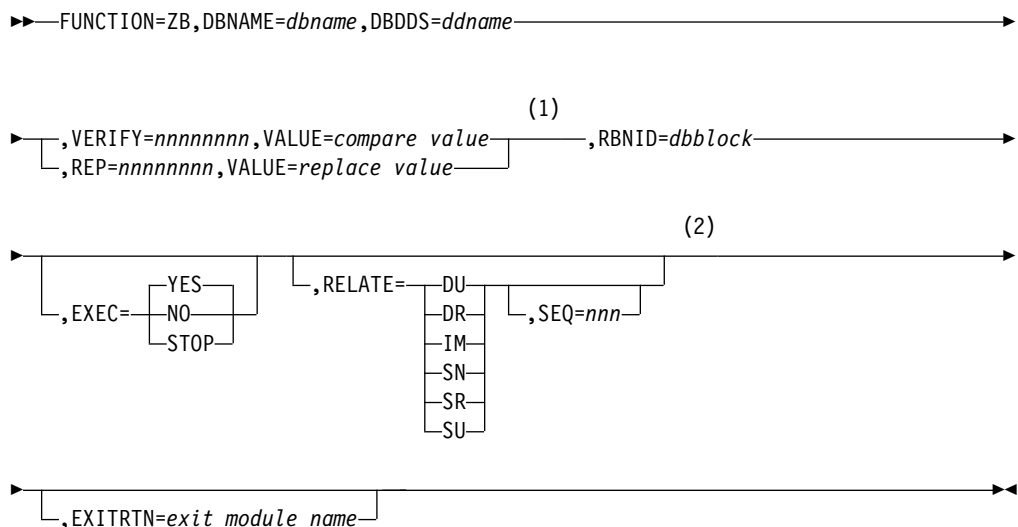
制約事項: このユーティリティー機能は、IBM の技術プログラム・サポート担当者だけが使用するものです。他の人が使用するときは、IBM 担当者の指示を受けて行ってください。

FUNCTION=ZB は、DBRC がアクティブであるときは RECON を更新しません。

ヒント: この機能の使用後に、イメージ・コピーをとります。

この機能では、DD 名の指定は 20 個までにしてください。

FUNCTION=ZB 制御ステートメントの形式は、次のとおりです。



注:

- 1 完結した制御ステートメントごとに、**VERIFY** キーワードまたは **REP** キーワードのどちらか一方を指定しておくことができます。これらのキーワードは同じステートメントに指定することはできません。ある完結したステートメントで **VERIFY** キーワードを指定し、別の完結したステートメントで **REP** キーワードを指定するときは、その実行順序が非常に重要です。ステートメントは **UCF** によって自動的にソートされないため、必ず **VERIFY** を **REP** の前に置くようにしなければなりません。**VERIFY** および **REP** 制御ステートメントは、同じジョブ・ステップに入れておかなければなりません。**REP** ステートメントが **VERIFY** ステートメントよりも後のステップに現れる場合、**REP** ステートメントは **VERIFY** ステートメントとの関連がなくなります。
- 2 **SEQ** キーワードを付けずに **RELATE** キーワードを使用すると、**UCF** は、**RELATE** キーワードで指定された関係機能と関連するデータベースのすべてで **ZAP** を実行します。**SEQ** キーワードと **RELATE** キーワードを同じ完結した制御ステートメントで使用すると、**ZAP** を特定の機能に限定することができます。

**DBNAME=**

この機能を適用するデータベースを指定します。値 **dbname** は **DBD** ライブラリーのメンバーとして存在する **DBD** 名であり、この名前を持つ実際のデータベースでもあります。

**DBDDDS=**

この機能で必要とするデータベース・データ・セットの任意の **DD** 名を指定します。各制御ステートメントに指定できる **DBDDDS** は 1 つだけです。

**VERIFY=**

関連の **VALUE=** フィールドの比較の対象となるオフセット (**RBNID** 内の **RBA** のアドレスからみたもの) を 16 進数で定義します。**VERIFY** が失敗すると、**ZAP** 全体が終了します。**VERIFY** は、8 桁の 16 進数で指定する必要があります。

**REP=**

関連の **VALUE=** データで置き換えられるオフセット (ゼロからみたもの) を 16 進数で定義します。**REP** は、8 桁の 16 進数で指定する必要があります。

**VALUE=**

データを定義します (16 進数表記を使用し、例えば、**C "A"** はステートメントでは **C1** となります)。このデータは、**VERIFY** キーワードと関連する場合は比較フィールドとして、また、**REP** キーワードと関連する場合は置き換えデータとして操作の対象となるものです。正確に 8 桁の 16 進文字で指定しなければなりません。これは、4 バイトのデータを、制御ステートメントごとに変更または比較する必要があることを意味しています。このキーワードは、制御ステートメントごとに 1 つ指定しなければなりません。

**RBNID=**

関連の **VERIFY=** または **REP=** データにより操作の対象となるデータベース・ブロックを 16 進表記で定義します (例えば、**RBNID= 00001000**)。OSAM-HISAM データ・セットの場合、これは、相対レコード番号 (**RRN**) です。OSAM 非 HISAM データ・セットの場合、相対ブロック番号 (**RBN**) です。VSAM データ・セットの場合、これは、ブロック内の最初のバイトの相

対バイト・アドレス (RBA) です。このキーワードを指定する場合は、正確に 8 桁の 16 進文字で指定しなければなりません。この値はパックされて、4 バイト・フィールドに入れられます。

#### **EXEC=**

この制御ステートメントを実行するかどうかを指定します。STOP と指定した場合、UCF は、この機能が完了した時点で処理を終了します。YES と指定した場合、この機能は処理されます。NO と指定した場合、この制御ステートメントは実行されません。EXEC=YES がデフォルトです。

#### **RELATE=**

ある機能制御ステートメントを別のものに関連付けます。値 xx は、関連付けられている機能制御ステートメントで定義されるもので、これについては、FUNCTION キーワードの項で説明してあります。関連付けられる制御ステートメントは、SEQ キーワードを使用して定義します。これが RELATE キーワードと同じ制御ステートメントに指定されていないと、1 つの完結した制御ステートメントに含まれる RELATE 指定すべてに ZAP が適用されます。RELATE キーワードを使用すると、FUNCTION=ZB 制御ステートメントを次の機能と関連付けることができます。

**DU** (DFSURGU0 を表す)

**SN** (DFSURGS0 を表す)

**DR** (DFSURGL0 を表す)

**IM** (DFSUDMP0 を表す)

**SU** (DFSURUL0 を表す)

**SR** (DFSURRL0 を表す)

Sequence=nnn は、この ZAP 機能を、同じシーケンス番号を持つ別の UCF 機能制御ステートメントとリンクさせます。RELATE キーワードが指定されていない場合は、このキーワードは無効です。

#### **EXITRTN=**

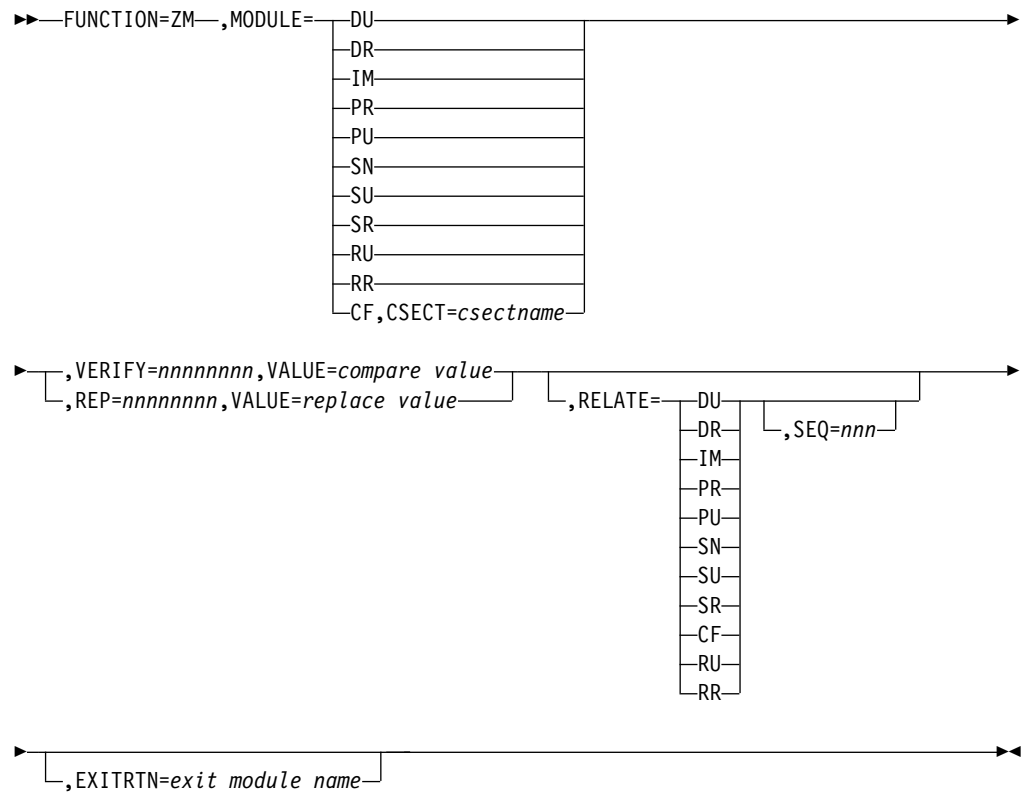
ブロックがストレージにあるときに制御を受け取って、ZAP が行われたかどうかを確かめ、必要な場合はそのブロックの統計をコンパイルするために使用します。

### **FUNCTION=ZM ステートメント**

この制御ステートメントは、ZAP を特定の UCF モジュールに適用させるためのものです。ZAP はストレージで適用され、論理エラーを訂正し、また、異常終了を強制的に引き起こします。サポートされるのは、保守援助機能 SPZAP プログラムの VERIFY 制御ステートメントと REP 制御ステートメントだけです。

**制約事項:** このユーティリティー機能は、IBM の技術プログラム・サポート担当者だけが使用するものです。他の人が使用するときは、経験のある人の指示を受けて行ってください。

FUNCTION=ZM 制御ステートメントの形式は、次のとおりです。



#### MODULE=

ZAP の対象となるロード・モジュールを定義します。このキーワードは、制御ステートメントごとに 1 つだけ指定できます。以下のモジュールの 1 つを指定しなければなりません。

- DU (DFSURGU0 を表す)
- SN (DFSURGS0 を表す)
- DR (DFSURGL0 を表す)
- SU (DFSURUL0 を表す)
- IM (DFSUDMP0 を表す)
- SR (DFSURRL0 を表す)
- PR (DFSURG10 を表す)
- CF (DFSUCF00 を表す)
- PU (DFSURGP0 を表す)
- RU (DFSURUL0 を表す)
- RR (DFSURRL0 を表す)

#### CSECT=

ZAP の対象となるロード・モジュール内の CSECT を定義します。このキーワードを省略した場合、ロード・モジュールは、エントリー・ポイントが置かれている CSECT で変更され、アドレス・ゼロからみたエントリー・ポイントのオフセットとなります。(特定のロード・モジュールの有効な CSECT 名について

は、IMS システム定義一覧表を参照してください。)このキーワードは、MODULE=CF が指定された場合にのみ有効であり、それ以外の場合は無視されます。

#### **VERIFY=**

関連の VALUE= フィールドの比較の対象となるオフセット (RBNID 内の RBA のアドレスからみたもの) を 16 進数で定義します。VERIFY が失敗すると、ZAP 全体が終了します。VERIFY は、8 桁の 16 進数で指定する必要があります。

#### **REP=**

関連の VALUE= データで置き換えられるオフセット (ゼロからみたもの) を 16 進数で定義します。REP は、8 桁の 16 進数で指定する必要があります。

#### **VALUE=**

データを定義します (16 進数表記を使用し、例えば、C "A" はステートメントでは C1 となります)。このデータは、VERIFY キーワードと関連する場合は比較フィールドとして、また、REP キーワードと関連する場合は置き換えデータとして操作の対象となるものです。正確に 8 桁の 16 進数で指定しなければなりません。これは、4 バイトのデータを、制御ステートメントごとに変更または比較する必要があることを意味しています。VALUE キーワードは、制御ステートメントごとに 1 つ指定しなければなりません。

#### **RELATE=**

この ZAP ステートメントを別の UCF 制御ステートメントと関連付けます。このキーワードは、ZAP ステートメントに 1 つだけ指定できます。

値 *xx* は、関連付けられている制御ステートメントで定義されるもので、これについては、FUNCTION キーワードの説明の項で説明してあります。関連付けられる機能制御ステートメントは、SEQ キーワードを使用して定義します。これが RELATE キーワードと同じ制御ステートメントに指定されていないと、1 つの完結した制御ステートメントに含まれる RELATE 指定すべてに ZAP が適用されます。RELATE キーワードを使用すると、FUNCTION=ZM 制御ステートメントを次の機能と関係付けることができます。

**DU** (DFSURGU0 を表す)

**SN** (DFSURGS0 を表す)

**DR** (DFSURGL0 を表す)

**SU** (DFSURUL0 を表す)

**IM** (DFSUDMP0 を表す)

**SR** (DFSURRL0 を表す)

**PR** (DFSURG10 を表す)

**CF** (DFSUCF00 を表す)

**PU** (DFSURGP0 を表す)

**RU** (DFSURUL0 を表す)

**RR** (DFSURRL0 を表す)

### SEQ=

この ZAP 機能を、同じシーケンス番号を持つ別の UCF 機能制御ステートメントにリンクさせます。RELATE キーワードが指定されていない場合は、このキーワードは無効です。

### EXITRTN=

ZAP が実行されたとき制御を受け取るために使用します。

---

## DFSUCF00 ユーティリティーの例

これらの例は、サンプル JCL を使用したユーティリティー制御機能の使用法を示しています。

サブセクション:

- 『UCF の実行に必要な最小限の JCL』
- 『再始動時に UCF を実行するための JCL』
- 484 ページの『再始動時に UCF を実行するための JCL および制御ステートメント』

### UCF の実行に必要な最小限の JCL

以下の図は、制御ステートメント入力データ・セットと UCF の実行に最低限必要な JCL を示しています。

```
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSUCF00'  
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR  
//DFSRESLB DD DSNAME=IMS.SDFSRESL,DISP=SHR  
//DFSPPRINT DD SYSOUT=A,DCB=(BLKSIZE=605,LRECL=121,  
// RECFM=FBA)  
//IMS DD DSNAME=IMS.PSBLIB,DISP=SHR  
// DD DSNAME=IMS.DBDLIB,DISP=SHR  
//DFSJRNLL DD DSNAME=NJRNLL,DISP=(,KEEP),  
// UNIT=SYSDA,SPACE=(CYL,(2,2)),DCB=BLKSIZE=1600  
// DCB=(RECFM=VB,BLKSIZE=4008,LRECL=4000)  
//DFSJRNLL DD DUMMY  
//DFSNCDS DD DSNAME=NCDS,DISP=(,KEEP),  
// UNIT=SYSDA,SPACE=(CYL,(2,2)),DCB=BLKSIZE=1600  
//DFSOCDS DD DUMMY  
//DFSORDER DD DUMMY  
//DFSYSIN DD *  
UCF CONTROL STATEMENT INPUT  
/*  
//DFSCNTRL DD DSNAME=CNTRL,UNIT=SYSDA,SPACE=(CYL,(2,2)),  
// DCB=BLKSIZE=80  
//*OTHER JCL AS REQUIRED BY  
//*UCF CONTROL STATEMENTS  
//*SPECIFIED BY THE USER.
```

### 再始動時に UCF を実行するための JCL

以下の図は、再始動が必要になったときに UCF を実行するのに使用される JCL を示したものです。STEP1 は、EXEC ステートメントのパラメーター・フィールドを使った再始動を示しています。DFSYSIN DD ステートメントは、この実行では DD DUMMY と指定されています。

```
//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSUCF00,,0001'  
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR  
//DFSRESLB DD DSNAME=IMS.SDFSRESL,DISP=SHR
```

```

//DFSPPRINT DD SYSOUT=A
//IMS      DD DSNAME=IMS.PSBLIB,DISP=SHR
//        DD DSNAME=IMS.DBDLIB,DISP=SHR
//DFSJRNRL DD DSNAME=NJRNL2,DISP=(,KEEP),
//        UNIT=SYSDA,SPACE=(CYL,(2,2)),DCB=(RECFM=B,BLKSIZE=4008,LRECL=4000)
//DFSJRNRL DD DSNAME=NJRNL,DISP=(OLD,KEEP)
//DFSNCDS  DD DSNAME=NCDS2,DISP=(,KEEP),
//        UNIT=SYSDA,SPACE=(CYL,(2,2)),DCB=BLKSIZE=1600
//DFSOCDS  DD DSNAME=NCDS,DISP=(OLD,KEEP)
//DFSORDER DD DUMMY
//DFSYSIN  DD DUMMY,DCB=BLKSIZE=80
//DFSCNTRL DD DSNAME=CNTRL,UNIT=SYSDA,SPACE=(CYL,(2,2)),
//        DCB=BLKSIZE=80
//*
//*OTHER JCL AS REQUIRED TO
//*IDENTIFY DATA SETS REQUIRED
//*FOR THE ACCESS METHOD BEING RESTARTED.

```

## 再始動時に UCF を実行するための JCL および制御ステートメント

以下の図は、再始動が必要になったとき、制御ステートメントを使用して UCF を実行するために使用される JCL を示しています。

```

//STEP1 EXEC PGM=DFSRR00,PARM='ULU,DFSUCF00'
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//DFSRESLB DD DSNAME=IMS.SDFSRESL,DISP=SHR
//DFSPPRINT DD SYSOUT=A
//IMS      DD DSNAME=IMS.PSBLIB,DISP=SHR
//        DD DSNAME=IMS.DBDLIB,DISP=SHR
//DFSJRNRL DD DSNAME=NJRNL1,DISP=(,KEEP),
//        UNIT=SYSDA,SPACE=(CYL,(2,2)),DCB=(RECFM=B,BLKSIZE=4008,LRECL=4000)
//DFSJRNRL DD DSNAME=NJRNL,DISP=(OLD,KEEP)
//DFSNCDS  DD DSNAME=NCDS1,DISP=(,KEEP),
//        UNIT=SYSDA,SPACE=(CYL,(2,2)),DCB=BLKSIZE=1600
//DFSOCDS  DD DSNAME=NCDS,DISP=(OLD,KEEP)
//DFSORDER DD DUMMY
//DFSYSIN  DD *
//        FUNCTION=OP,COND=RESTART
//*
//DFSCNTRL DD DSNAME=CNTRL,UNIT=SYSDA,SPACE=(CYL,(2,2)),
//        DCB=BLKSIZE=80
//*
//*OTHER JCL AS REQUIRED TO
//*IDENTIFY DATA SETS REQUIRED
//*FOR THE ACCESS METHOD BEING RESTARTED.

```

---

## DFSUCF00 ユーティリティーの実行

DFSUCF00 ユーティリティーの実行には、いくつか操作上の考慮事項があります。通常の処理、終了エラー処理、再始動処理、ユーザー指定出口ルーチンの処理などについてです。

サブセクション:

- 485 ページの『通常の処理』
- 486 ページの『初期ロード・アプリケーション・プログラムに関する考慮事項』
- 488 ページの『終了/エラー処理』
- 489 ページの『再始動処理』
- 490 ページの『ユーザー指定出口ルーチンの処理』



## 通常の処理

通常の処理を行うには、データベース・スキャン (DFSURGS0)、接頭部解決 (DFSURG10)、および接頭部更新 (DFSURGP0) 以外のすべてのユーティリティー機能を管理するための制御ステートメントが必要です。UCF は、次の場合を除き、これら 3 つの機能については、必要とするすべてのステートメントを自動的に生成します。

- アンロードまたは再ロードだけが要求された場合は、データベース・スキャン、接頭部解決、および接頭部更新処理は自動的に行われます。ただし、これらのユーティリティーの実行を求める制御ステートメントはユーザーが指定しなければなりません。
- 初期ロードだけが要求された場合は、これらのユーティリティーの処理は自動的に行われます。ただし、それらの制御ステートメントのキーワードが、処理を行わないように指示している場合を除きます。データベース・スキャンの実行を要求するには、制御ステートメントが 1 つ必要です。その後に出されるスキャン要求は無視されます。

制御ステートメントは一度にまとめて読み取られ、制御データ・セットが作成されます。制御データ・セットに入っている項目がすべて相互参照され、矛盾する要求が行われていないか、また、すべての論理関係機能がユーザーから要求されているのか、あるいは UCF により生成されたのかが調べられます。

UCF は、制御ステートメントを実行依頼したときの順序に関係なく、ユーティリティーを特定の順序で実行します。実行の順序は次のとおりです。

1. HISAM 再編成アンロード
2. HISAM 再編成再ロード
3. データベース・スキャン
4. HD 再編成アンロード
5. HD 再編成再ロード
6. 初期ロード
7. 接頭部解決
8. 接頭部更新
9. 副次索引の再編成アンロード
10. 副次索引の再編成再ロード
11. イメージ・コピー (バッチ)
12. データベース ZAP
13. モジュール ZAP

同じ機能がいくつかの制御ステートメントで要求されたときは、UCF は機能をデータベース名の昇順で実行します。同じ機能を指定した複数の制御ステートメントが同じデータベース名も指定している場合は、これらのステートメントは UCF によって読み取られた順序で実行されます。

制御ステートメントは、1 つの機能内でデータベース名の順に実行されます。同一の機能およびデータベース名内で、それらは実行依頼された順に実行されます。

処理は、次に開始する機能を判断し、イベントをジャーナル・データ・セットに記録し、該当のユーティリティーを接続 (アタッチ) することにより、進められます。

すべての機能の編成と実行は、特定の操作上の問題が起こる可能性のある場合に、それを防ぐような形で行われます。例えば、論理子が論理親を指す直接ポインターを持っているとき、論理親が置かれているデータベースを再編成する場合を考えてみます。データベース・スキャンを実行する前に、論理親データベースがアンロードされて再ロードされると、論理関係は壊れてしまいます。しかし、UCF は、論理子データベースを先にスキャンしてから、論理親データベースをアンロードして再ロードします。

独立型ユーティリティーの統計報告書と要約報告書は、UCF 出力の一部です。統計を抑止するオプションがこれらのユーティリティー内で指定されている場合、統計を印刷するかどうかは、REQUEST キーワード値 STATS および NOSTATS で決定されます。

HD 再編成再ロード・ユーティリティー・ステップは、正常終了した時点で、内部的にバッチ・データベース・イメージ・コピー・ユーティリティー要求を生成します。このステップは、まだ処理すべき優先順位の高い他のステップがある場合は、即時には行われません。副次索引再ロード・ステップは、バッチ・データベース・イメージ・コピー・ユーティリティー・ステップを開始させません。

## 初期ロード・アプリケーション・プログラムに関する考慮事項

このトピックには汎用プログラミング・インターフェース情報が含まれています。

初期ロード・プログラムは UCF の制御下で実行できるので、UCF 再始動機能を利用できます。ほとんどの場合、これらのプログラムを少し変更するだけで、正しいインターフェースが得られます。プログラムは、再始動処理が行われるとき、WTOR 処理中止要求が入力されたとき、あるいはチェックポイントがとられたときを認識できるように変更しておく必要があります。インターフェースは、次のようになっています。

Register 0 contains a 4-word parameter list as follows:

- 1st word = DBPCB list
- 2nd word = DFSPRINT data set
- 3rd word = PST
- 4th word = During restart of the user's  
initial load program, the address  
of area containing the last segment  
loaded prior to checkpoint.

Register 1 contains the DBPCB list

ユーザーの初期ロード・プログラムの再始動処理が進行中の場合は、プログラムに入ったときに、状況コードとして UR (再始動を意味する) が返され、ロード PCB に入ります。別のパラメーターとして、PCB キー・フィードバック域に入っている完全連結キーも渡されます。このキー・フィードバック域に入っている情報によって、再始動のタイプが判断され、次の 2 つの条件が適用されます。

- その情報がゼロでなければ、再始動処理は終了箇所から始まります。
- その情報がゼロであれば、再始動処理はプログラムの先頭から始まります。

ユーザーのプログラムは、次にロードするルート・セグメントまたは従属セグメントで応答する必要があるため、ユーザーの入出力ファイルはアプリケーション・プログラムで調整する必要があることがあります。

入力ファイル上の再始動箇所を判断するためにさらに検査が必要になるので、実際のセグメント・データも指定されます (これは、例えば、非固有フィールドや順序付けされていないフィールドがある場合に重要になります。)

上記のほか、ユーザーのプログラムは、ユーザーの入出力ファイルでチェックポイントをとる必要があること、オペレーターの応答の結果、処理を終了させる必要が出てきたことを DL/I 状況コードが示しているときを認識できるように変更しておく必要があります。この状況コードは、呼び出しを行うと状況がブランクになるようなときだけ返されます。

DL/I 状況コードとその意味は次のとおりです。

- UC** チェックポイントがとられました。データベースは、ロードされたばかりのルートの前に置かれた論理レコードの箇所でチェックポイントがとられます。
- UR** これは再始動です。
- US** オペレーターは、初期ロード・プログラムがその処理を中止するように要求しました。
- UX** UC と US の組み合わせ。

US と UX が同時に起こったときは、次のチェックポイントの箇所で処理を中止できます。

初期ロード・アプリケーション・プログラムに適用される制約があることに注意してください。

#### 初期ロード出口ルーチン

このトピックには汎用プログラミング・インターフェース情報が含まれています。

UCF チェックポイント・モジュール (DFSUCP90) は、ロード挿入モジュール (DFSDDLE0) から直接制御を受け取るため、UCF にユーザーの初期ロード・プログラムのチェックポイントをとらせ、未処理の WTOR に対する応答があれば、それを処理させることができます。他方、DFSUCP90 モジュールは、ユーザーの出口ルーチンとのインターフェースとなるため、ユーザーはチェックポイント間隔を変更するか、作業データ・セットのチェックポイントをとるか、あるいはその両方を行うことができます。1 つの完全なデータベース・レコードを書き込まないと、DFSUCP90 は、ユーザー出口ルーチンによるチェックポイント要求を受け入れません。2 番目のデータベース・レコードが存在する場合は、最初のレコードが完了し、DFSUCP90 を正常に実行できることを示します。

初期ロード出口ルーチンは、データベース・レコード内の最後のセグメントが挿入された後、レコードごとに 1 回のみ呼び出されます。チェックポイント・ロジック以外では、コードはデータベース・レコード境界上であって、次のレコードを挿入できるようになっていなければなりません。再始動が発生する場合、これはレコード境界上に書き込まれたデータからである必要があります。

出口ルーチンとのインターフェースは次のようになっています。

Register 1 contains a 3-word parameter list as follows:  
1st word = common area defined by DSECT UCFCMVEC  
2nd word = DFSPRINT data set  
3rd word = PST

共通域にはフィールド U7CURCKP および U7CKPTNK があります。  
U7CURCKP フィールドには、現在有効なチェックポイント間隔が入っています。  
U7CKPTNK フィールドは、4 バイト・カウンターとして働きます。ここには、チェックポイントをとる前に処理しておくべきレコードの数が入っています。

次の方法のいずれかにより、個々のデータ・セットのチェックポイントと、初期ロード・プログラムで現在有効なチェックポイントとを同期させることができます(その方法は任意です)。

- カウントをモニターし、カウンター (U7CKPTNK) がゼロになったときデータ・セットのチェックポイントをとる。
- チェックポイントを強制的にとるためにカウンターを 16 進数のゼロにクリアしてから、データ・セットのチェックポイントをとる。

DFSUCP90 に戻るとき、U7CKPTNK フィールドがゼロであれば、チェックポイント・レコードはジャーナル・データ・セットに書き込まれ、このフィールドは U7CURCKP フィールドに入っている値に再初期設定されます。

初期ロード出口ルーチンは、VSAM データ・セットについてのみ有効です。ユーザー出口が OSAM について指定される場合、出口ルーチンは無視されます。

## 終了/エラー処理

エラー検査は、ユーティリティーの実行中だけでなく、完了した後も行われます。エラーがない場合、イベントの完了が記録され、処理停止要求があるかどうかを検査されます。処理停止要求が出されている場合は、再始動メッセージが生成され、ジョブは終了し、戻りコードとして 4 が返されます。処理停止要求が出されていない場合は、次の機能が判断され、通常の処理の項で説明したように処理が続行されます。

実行のエラー検査フェーズでエラーが検出されると、イベントの完了はエラーとして記録され、処理が終了して、機能が再始動できるようにします。ジョブ全体が完了したときは、戻りコードが渡されます。正常に完了したときは、再始動処理は必要でないため、ジャーナル・データ・セットに書き込まれた特殊なレコードにより止められます。正常以外の終了は再始動できます。統計は、各機能が終了したときと、ジョブの終了時に印刷されます。

ゼロの戻りコードがレジスター 15 に返された場合は、ユーザーの初期ロード・プログラムが正常に完了し、後で再始動を行う必要がないことを意味します。しかし、後で再始動が必要である場合は、レジスター 15 に 4 より大きい戻りコードが返されます。この場合には、チェックポイントがとられ、次のルート・セグメントから再始動が行われます。

チェックポイント/再始動

UCF には、異常終了したときとユーザーから終了要求があったときのために、内部チェックポイント/再始動機能が備わっています。チェックポイント機能はジャーナル・データ・セットを必要とするので、ユーザーは UCF のためにその割り振りを行っておく必要があります。通常、バッチ処理に使用される IMS システム・ログは、この場合には使用されません。チェックポイントは、機能ユーティリティーが始動または終了されたとき、および制御機能が始動または終了されたときにとられます。チェックポイントは、ユーザー指定のレコード・カウントおよびデフォルトのレコード・カウントの結果として、機能ユーティリティーの処理内の特定の箇所でもとられます。これらのチェックポイントごとに、レコードがジャーナル・データ・セットに書き込まれ、チェックポイントのタイプとそのタイプに見合う再始動制御情報が定義されます。

これらのチェックポイントは、再始動プロセッサで使用されるので、z/OS チェックポイントではなく内部チェックポイントです。チェックポイントの頻度は、ユーザー・オプションとして指定することができますが、2000 個のデータベース関連レコードごとにチェックポイントをとるというデフォルトの設定を適用することもできます。

## 再始動処理

再始動処理が必要なときは、プログラムの最後の終了時に使用されていた制御データ・セットが読み取られ、最後に使用されたジャーナル・ログも読み取られます。最後に開始された機能または完了した機能 (あるいはその両方) は、ジャーナル・レコードを制御データ・セットに入っている機能と突き合わせることによって判断され、処理は通常の処理の場合と同様に続行されます。

最後にどのタイプのチェックポイント・レコードがジャーナル・データ・セットに書き込まれたかに応じて、再始動機能は次のいずれかの方法で行われます。

- レコードが機能ユーティリティー (つまり、データベース・ユーティリティーの 1 つまたはユーザーの初期ロード・プログラム) の開始を示していれば、再始動処理は、開始された機能から始まります。レコードが機能ユーティリティーの終了を示していれば、再始動処理は、次に実行される機能から始まります。
- レコードが制御機能 (制御データ・セットの作成など) の開始または終了を示していれば、再始動処理は、開始された制御機能または次に実行される機能のどちらから始まります。
- レコードがチェックポイントを示していれば、再始動処理はその箇所から始まり、IMS データ・セットは必要に応じて位置付けられます。

再始動処理を行うためには、チェックポイントがとられた時点で使用中であったすべてのデータ・セットが使用可能になっていなければなりません。

再始動時に、マルチボリューム出力データ・セットの位置を変更する場合は、元の実行で使用されなかったボリューム通し番号を除去するよう、DD ステートメントを変更する必要があります。これを行っておかないと、出力データ・セットの位置が正しく指定されません。

接頭部解決ユーティリティーの再始動は、実行の最初から行わなければなりません。このユーティリティーでは、終了時点までに作成されたデータ・セットすべて

を消去し、初めて始動するときと同様にユーティリティーを再始動させなければなりません。

## ユーザー指定出口ルーチンの処理

このトピックには汎用プログラミング・インターフェース情報が含まれています。

UCF を使用すると、どのユーティリティーにおいても、ユーザー指定出口ルーチンがレコードを調べ、統計をコンパイルすることができます。ユーザー出口ルーチンの処理中にはどの IMS 制御データも変更できませんが、セグメント・レコードに要求される長さが守られる限り、ユーザー・データは変更できます。ユーザー出口ルーチンは、ルーチンの処理中に、すべての IMS データを未変更のままにしておかなければなりません。HD 再編成アンロード・ユーティリティー (DFSURGU0) および HD 再編成再ロード・ユーティリティー (DFSURGL0) のユーザー出口ルーチンは、セグメントを削除または挿入し、ロードされた後のブロックを調べることができます。

ユーザー出口ルーチンを指定するには、特定の機能に関連するユーティリティー制御ステートメントで EXITRTN キーワードをコーディングし、出口ルーチンの名前を指定します。ユーザー出口ルーチンは、LINKLIB に置いておくか、STEPLIB データ・セットまたは JOBLIB データ・セットに定義しておかなければなりません。

ユーザー出口ルーチンに入ると、レジスター 1 は、次の 3 つの項目から成るパラメーター・リストを指します。

- データのアドレス
- DFSPRINT DCB のアドレス
- 正常に完了 (非異常終了) したときの区画指定テーブル (PST) のアドレス

最初のパラメーターによってアドレス指定されたデータは、イメージ・コピー・レコードです。最初の呼び出しでは、出口は IC ヘッダー・レコードで呼び出されません。

ユーザー出口ルーチンをコーディングするための考慮事項:

- データをアドレス指定するには、RDW はデータの一部でないののでそれについて調整します。
- ファイル・レコードの長さを調整するには、RDW を変更します。
- ユーザー出口ルーチンは、24 ビットのアドレッシング・モードで呼び出され、その常駐モードは 16 MB 境界より下でなければなりません。

ユーザー出口ルーチンに入ったのが最終回であると、データのアドレスが 0 になります。

HD 再編成再ロード・ユーティリティーのユーザー出口に入る入り口点には、次の 2 つがあります。

- 再ロード・レコードがアンロード・テープから読み取られたときは、オフセット 0
- レコードがデータベースにロードされたときは、オフセット 4

ユーザー出口ルーチンは、MOVE MODE 操作のとき PUT マクロ・ステートメントを出すと、DFSPRINT データ・セットに書き込むことができます。DFSPRINT

データ・セットは、出口ルーチンに入る前にオープンされます。出口ルーチンが使用する非ユーティリティー・データ・セットは、そのルーチンがオープンし、クローズしなければなりません。

HD 再編成ユーティリティーで使用されるユーザー出口ルーチンは、戻りコードをレジスター 15 に返すことによって、セグメントの後処理をユーティリティーに知らせることができます。HD 再編成ユーティリティーに認識される戻りコードには、次のものがあります。

コード 意味

- 0 正常に処理する
- 4 出口ルーチンに渡されたセグメントを削除する
- 8 レジスター 1 が指しているセグメントを現行セグメントの前に挿入する。このセグメントは、再ロード・ユーティリティーの統計に含まれません。最初に渡されたものと同じセグメントと一緒に出口ルーチンに戻ります。

戻りコード 8 の考慮事項:

- HD アンロードでは、戻りコード 8 は意味を持たず、無視されます。
- オフセット 0 で入力された HD 再ロードに戻るには、現行レコードの前にレジスター 1 が指しているアンロード・レコードを処理します。この場合、レジスター 1 は、レコードの最初のデータ・バイト (RDW の後の最初のバイト) を指すはずでず。

推奨事項: ユーザー出口に渡されたレコードを変更しようとししないでください。これにより、ユーザー ABEND 0805 が発生します。次の呼び出しで、HD 再ロードは、直前の呼び出しで戻されたのと同じレコードと共に同じオフセットで出口ルーチンに入ります。

- オフセット 4 で入力された HD 再ロードに戻るには、通常どおりに処理します。

セグメント名とセグメント・レベルは、PST が指している PCB にあります。挿入するセグメントの論理子は、それぞれのデータベースに別々に挿入しなければなりません。

保守援助機能

UCF には、2 種類の保守援助機能が用意されています。エラー・ポイント異常終了と、データベース ZAP およびモジュール ZAP です。

制約事項: ZAP 援助機能は、IBM 技術プログラム・サポート担当者だけが、あるいは権限を持つ人が正しい指示を受けて使用するものです。

エラー・ポイント異常終了

UCF では、選択的異常終了要求が可能です。処理の途中で診断メッセージが出されたときは、プログラムに異常終了を引き起こすべき箇所を、制御ステートメントで選択することができます。REQUEST=MSGALL を指定するということは、A または W タイプの診断メッセージはすべて、異常終了要求として設定されることを指示しています。MSGNUM キーワードに正確にメッセージを指定しておけば、処理の途中でそのメッセージが出されたとき、そこで異常終了させることができます。

## データベース ZAP またはモジュール ZAP

UCF を使用すると、データベース・ブロック、UCF モジュール、または UCF ユーティリティを ZAP して、異常終了条件を強制的に引き起こしたり、論理エラーを訂正したりできます。この ZAP 機能のための制御ステートメント規則は、SPZAP プログラムの制御ステートメント規則に従います。しかし、サポートされるのは、VERIFY および REP 制御ステートメントだけであり、その使用にはいくつかの制約があります。データは正確に 8 バイトにして、2 バイトの 16 進数形式で指定しなければなりません。データを、コンマやスペースで区切らないでください。

モジュールの ZAP がストレージで行われるのに対し、データベース ZAP はディスク上で行われます。ZAP ステートメントに RELATE キーワードが指定されていると、モジュール ZAP は、関連のモジュールの実行前に行われます。データベース ZAP は、アンロード・ユーティリティ機能の前とロード・ユーティリティ機能の後で行われます。

モジュール ZAP は、RELATE キーワードと SEQ キーワードで制限していない限り、指定されたモジュール名の個々の実行の前に行われます。あるモジュール ZAP で RELATE が指定されている場合、一致するモジュール名を持つ機能制御ステートメントだけが ZAP されます。SEQ が指定されている場合は、ZAP はさらに制限され、一致するシーケンス番号を持つモジュールだけに適用されます。データベース ZAP は、アンロード・ユーティリティ機能の前、ロード・ユーティリティ機能の後、および要求されたすべての機能ユーティリティが実行された後で行われます。

表 25. データベース ZAP

アンロード・ユーティリティ機能の前	ロード・ユーティリティ機能の後
IM	SR
SU	DR
RU	RR
DU	RV

上記以外のデータベース ZAP は、いずれも、最後に要求された機能ユーティリティが実行された後で行われます。

### 要応答オペレーター宛メッセージ機能

要応答オペレーター宛メッセージ (WTOR) は、UCF ユーザーにメッセージを出し、それに対するユーザーからの応答を処理します。UCF により与えられる未処理 WTOR を使用すると、その実行の状況を判断するために UCF に照会し、チェックポイント値を変更し、UCF を停止させ、しばらくしてから再始動させることができます。

制約事項: 再始動は、WTOR では開始できません。

WTOR は、ユーティリティの実行と実行との間、およびチェックポイント時に処理されます。



以下の図は、メッセージ DFS367I がコンソールに送られてきたので、UCF の実行状況を要求した結果、UCF がデータベース「DBNAME」で HD 再編成再ロード・ユーティリティーを実行中であることが分かった場合を示しています。(機能が PR の場合、DBNAME はメッセージには含まれません。)

```
@09 *DFS367I UTILITY CONTROL FACILITY RUNNING,
      ENTER REQUESTS AS NEEDED
r 9, status
IEE600I REPLY TO 09 IS 'STATUS'
DFS369I FUNCTION IS DR FOR DATABASE DBDNAME
@10 *DFS367I UTILITY CONTROL FACILITY RUNNING,
      ENTER REQUESTS AS NEEDED
```

最後の DFS367I メッセージに対する応答として、例えば、チェックポイント値を次のように即時に変更することを要求できます。

```
r 10,ckpnt=100
```

あるいは、例えば、次のように UCF に処理を中止することを要求することもできます。

```
r 10,END
```

UCF は、実行中のユーティリティーによる END 要求を認識すると、処理を中止します。

さらに、メッセージ DFS367I に対する応答として、特定の制御応答とオプション応答を入れることができます。入力できる情報を以下の表にまとめておきます。

表 26. メッセージ DFS367I への制御応答とオプション応答

制御応答	オプション応答
END	ID
	FUNCTION=
FUNCTION xx END	SEQ=
	EXEC=
STATUS	値
	REQUEST=
	CKPNT=

1 つの応答で入力できる制御応答は 1 つに限られます。

#### END

現在の機能における、またはその後の次のチェックポイント (先に現れた方のチェックポイント) で処理を中止します。

#### FUNCTION xx END

指定された機能の最初の実行の後で処理を中止します。FUNCTION=OP、COND=RESTART と指定した制御ステートメントを DFSYSIN データ・セットに追加してジョブを再実行依頼すると、FUNCTION xx END の後で UCF を再始動できます。

#### STATUS

これを指定すると、処理中の機能と、(可能な場合には) 処理中のデータベースまたはデータ・セットを説明した WTO メッセージが出されます。状況要求

は、ユーティリティーの実行と実行の間、またはチェックポイント時のうち、どちらかが最初に起こった箇所で処理されます。

FUNCTION= ID および SEQ= ID はどちらも、単独で入れることができますが、EXEC= ID と一緒に入れることもできます。ただし、EXEC= ID は、FUNCTION= ID または SEQ= ID のどちらかが入力された場合にのみ入力できます。

FUNCTION= と SEQ= の両方を同じ応答に入力する場合は、同じシーケンス番号の制御データ・セット項目と一致していなければなりません。例えば、ユーザーの入力が FUNCTION=PR,SEQ=004 である場合、SEQ=004 を持つ制御データ・セットを FUNCTION=PR と関連付けておく必要があります。

FUNCTION= ID がオプション応答で入力されない場合、FUNCTION=OP が指定されたものと見なされます。

制約事項: FUNCTION=DX と FUNCTION=SX は、WTOR 応答では入力できません。

REQUEST= または CKPNT= の値は単独でも、ID と一緒でも入力できます。

応答の中で 1 つの ID が重複して使用されていると、その ID で最後に入力された値だけが受け付けられます。

関連資料:

 z/OS: SPZAP 制御ステートメント

---

## 第 5 部 報告書ユーティリティーおよびテスト・ユーティリティー

報告書ユーティリティーおよびテスト・ユーティリティーは、データベースについて、および、IMS 内でのそのデータベースの使用について報告とテストを行うために使用されます。

各トピックでは、1 つのユーティリティーを紹介し、その作動方法を説明し、その使用のための要件および制約事項を定義し、例を提供します。



---

## 第 32 章 データベース・モニター報告書印刷ユーティリティー (DFSUTR30)

データベース・モニター報告書印刷ユーティリティー (DFSUTR30) はオフライン・プログラムであり、IMS バッチ・システムの実行中に DB モニター (DFSMNTB0) によって収集された情報を要約した報告書を作成します。

このユーティリティーが作成する報告書には、次のものがあります。

- VSAM バッファークラッシュ報告書
- VSAM 統計報告書
- データベース・バッファークラッシュ報告書
- プログラム入出力報告書
- DL/I 呼び出し要約報告書
- 分布付録報告書
- モニター・オーバーヘッド報告書

ここでは、報告書の各フィールドについて説明し、その後その報告書の使用方法の要約を掲載しています。

サブセクション:

- 『制約事項』
- 『前提条件』
- 498 ページの『要件』
- 498 ページの『推奨事項』
- 498 ページの『入出力』
- 498 ページの『JCL 指定』
- 499 ページの『戻りコード』

### 制約事項

DFSUTR30 ユーティリティーは、DB モニター (DFSMNTB0) によって作成されるデータ・セット上のデータ・レコードに左右されます。報告される時刻および報告書に反映される統計の正確さは、データ・セットに入っているレコードによって異なります。さまざまなイベントのレコードは、開始イベント・レコードと終了イベント・レコードといった、対になっているものと考えられます。イベントは、その両方が受け取られない限り、カウントも報告もされません。

### 前提条件

DFSUTR30 ユーティリティーを実行する前に、DFSUTR30 ユーティリティーで使われる入力データ・セットを作成するため DB モニターを使用可能にする必要があります。

## 要件

現在、DFSUTR30 ユーティリティーに対して文書化された要件はありません。

## 推奨事項

現在、DFSUTR30 ユーティリティーに対して文書化された推奨事項はありません。

## 入出力

DFSUTR30 ユーティリティーへの 1 次入力、DB モニターの出力データ・セットです。DFSUTR30 ユーティリティーの 1 次出力は、バッチ・システムでのデータベース・アクティビティーを要約するさまざまな報告書のいずれかです。

表 27. DFSUTR30 ユーティリティーの入出力

入力	出力
IMSMON または IMSLOG データ・セット	報告書
分析制御データ・セット	

## JCL 指定

DFSUTR30 ユーティリティーはバッチ・モードで稼働し、モニター・トレース期間ごとに 1 つのジョブ・ステップが実行されます。

次のものがが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

EXEC ステートメントはプログラム名を指定します。このステートメントの形式は次のとおりです。

```
PGM=DFSUTR30
```

### DD ステートメント

#### STEPLIB DD

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。

#### SYSPRINT DD

出力データ・セットを指定します。SYSOUT=A とするのが普通です。

#### SYSUT1 DD

入力データ・セットを指定します。これは、DB モニター・モジュール DFSMNTB0 によって書き込まれたラベル付きデータ・セットです。別のデータ・セット (ddname= および dsname=IMSMON) またはシステム・ログ (dsname=IMSLOG) も使用できます。

## ANALYSIS DD

分析制御データ・セットを指定します。このファイルはカード・イメージ形式になっていなければなりません。デフォルト・パラメーター (分布報告書は作成されない) を使用するときは、DD DUMMY と指定しておけば、入力はテープ上の最初のトレース間隔となります。

### 分析制御データ・セット

分析制御データ・セットには、3つのレコード・タイプがあります。それは、分布付録報告書を要求するもの、分布間隔を再定義するもの、どのトレース間隔を処理するかを指定するものです。


- 分布付録報告書を作成する場合は、1桁目から始めて DISTRIBUTION または DIS と指定します。
- デフォルトの分布間隔を変更する場合は、制御ステートメントを Dn n1,n2,.. の形式で指定します。
- 入力データ・セット上のどのトレース間隔 (デフォルトである最初のもの以外) を処理するかを指示するには、1桁目から始めて FILE=nn または FILE=n と指定します。


FILE= の指定は OS ファイルを参照しません。これは、OS ファイル内に記録されたトレース間隔を参照します。

### 戻りコード

現在、DFSUTR30 ユーティリティに対して文書化された戻りコードはありません。

関連概念:

 IMS のモニター (オペレーションおよびオートメーション)

 IMS モニター報告書 (システム管理)

---

## DFSUTR30 ユーティリティの例

ここでの例は、DFSUTR30 ユーティリティのサンプル JCL を示しています。

以下の図は、通し番号が IMSDA1 のテープから最初のトレース間隔に関する 1 組の報告書を得るための JCL を示しています。

```
//*  
//STEP1 EXEC PGM=DFSUTR30  
//STEPLIB DD DSNAME=IMS.SDFSRESL,DISP=SHR  
//SYSPRINT DD SYSOUT=A  
//SYSUDUMP DD SYSOUT=A  
//SYSUT1 DD DSNAME=IMSMON,DISP=(OLD,KEEP),  
// UNIT=TAPE,VOL=SER=IMSDA1  
//ANALYSIS DD *  
DISTRIBUTION  
/*
```

D13 の分布を変更し、2 番目のトレース間隔を指定するには、分析制御データ・セットを以下の図に示されているように変更する必要があります。

```
//ANALYSIS DD *  
DISTRIBUTION  
FILE=2  
D13 ,,2,4,6,8,10,12
```

この例では、最初の 2 つの間隔は変更されず、それぞれ 0 と 1 のままになっています。



---

## 第 33 章 プログラム分離トレース報告書ユーティリティ (DFSPIRPO) (Program-Isolation-Trace Report utility (DFSPIRPO))

プログラム分離トレース報告ユーティリティ (DFSPIRPO) は、プログラム分離 (PI) トレースによって作成された X'67FA' ログ・レコードから報告書を印刷するために使用します。

プログラム分離 (PI) トレースは、すべてのプログラム分離ロック・マネージャーの ENQ 要求と DEQ 要求をトレース・テーブルに書き込みます。DFSPIRPO ユーティリティが作成する報告書は、待機が必要であった ENQ (リソースが即時使用可能にならなかったもの) だけを示します。

PRINT 制御ステートメントを使用して、DFSPIRPO ユーティリティが作成する報告書を、ある期間に限定することができます。

以下のトピックに追加情報があります。

- 『制約事項』
- 『前提条件』
- 502 ページの『要件』
- 502 ページの『推奨事項』
- 502 ページの『入出力』
- 503 ページの『JCL 指定』
- 505 ページの『戻りコード』

### 制約事項

DFSPIRPO ユーティリティ制御ステートメントのキーワードは、繰り返して指定することはできません。

DFSPRIRPO ユーティリティは、全機能 (HALDB および非 HALDB) データベースのみで機能します。

### 前提条件

DFSPIRPO ユーティリティを使用する前に、必ず一定の前提条件が満たされているようにする必要があります。

DFSPIRPO ユーティリティを実行する前に、プログラム分離 (PI) トレースを有効にして X'67FA' ログ・レコードを作成する必要があります。DFSPIRPO ユーティリティは、このログ・レコードからプログラム分離トレース報告書を生成します。プログラム分離トレースは、コマンド /TRACE SET ON PI OPTION ALL を出すと有効にできます。

## 要件

DFSPIRPO ユーティリティを実行するには、操作上の必要条件を 1 つ以上満たしていなければなりません。

ログ・レコードをソートするために、DFSPIRPO ユーティリティに対して 3 つ以上のソート作業データ・セットを定義する必要があります。

## 推奨事項

現在、DFSPIRPO ユーティリティに対して文書化された推奨事項はありません。

## 入出力

DFSPIRPO ユーティリティへの 1 次入力、プログラム分離トレースによって作成された X'67FA' ログ・レコードを含む、ログ入力データ・セットです。

DFSPIRPO ユーティリティの 1 次出力は、必要なリソースを待機しなかったプログラム分離 ENQ の報告書です。

以下の表は、DFSPIRPO ユーティリティへの入力およびこのユーティリティからの出力を示したものです。

表 28. DFSPIRPO ユーティリティの入出力

入力	出力
X'67FA' ログ・レコードを含むログ入力データ・セット	待機する必要があったプログラム分離 ENQ の報告書
制御ステートメント	メッセージ

プログラム分離トレース報告ユーティリティによって作成される報告書の内容は、次のとおりです。

- 要求されたリソース (DMB 名、DCB 番号、および 4 バイトの 16 進 ID (RBA))
- ENQ 要求の時刻 (呼び出しの時刻)
- 待機経過時間 (要求側タスクがリソースが使用可能になるまで待たされた時間)。PI トレース・タイミングが働いているときは、 /TRACE ALL コマンドを使用してください。

例外: 高速機能の場合の経過待機時間は記録されません。

- 要求側 PSB と保持側 PSB の名前
- ID、DCB、および DMB 別の待機の総数

高速機能環境では、要求されたリソースは、4 バイトの 16 進 ID (RBA) の先頭バイトに応じて変化します。ID の先頭バイトは、ロック・サブ ID である EPSTLKID と同じ内容を表します。以下のリストは、EPSTLKID の 16 進 ID および要求されたリソースの DMB 名内の名前を示しています。

### EPSTLKID ID

名前

(EPSTCILK)X'00'  
DEDB エリア名

X'F0' NOTAVAIL

(EPSTMDLK)X'F1'  
MSDB シンボル名

X'F2' NOTAVAIL

X'F3' NOTAVAIL

(EPSTARLK)X'F8'  
DEDB エリア名

X'FF' NOTAVAIL

ID の残りのバイトには、EPSTLKID の内容に関係なく、CI の上位 3 バイトが入ります。

PI トレース・タイミングが働いているときは、ENQ 要求の PI トレース X'67FA' ログ・レコードは、どの ID の場合も、その ID が獲得されたときと同じ順にログ・データ・セットに入れられます。したがって、ENQ 要求の要求側トランザクションは、同じ ID に対する次の ENQ 要求で待機が必要とされれば、その ENQ 要求の保持側トランザクションと見なされます。PI トレースの期間、タイミングが働いていなければ、可能性は低いものの、PI トレース・ログ・レコードが ID の獲得時と同じ順序にならない場合も考えられます。すなわち、ENQ 要求は ID を獲得したが、その ID が PI トレース・ロガー・バッファに追加される直前に、優先度の高いタスクが同じ ID に対して ENQ 要求を出して割り込みをかけると、このようなことが起こる場合があります。この 2 番目のタスクは待機させる必要がありますが、その PI トレース・ログ・レコードは、保持側タスクに対応するレコードより前に置かれます。

制約事項: タイミングが働いていれば、トレース報告ユーティリティーによって使用されるログ・レコードは、リソースが獲得された後にだけログ・バッファに追加されるので、このようなことは起こりません。

## JCL 指定

DFSPIRPO ユーティリティーは、標準の z/OS ジョブとして実行されます。DFSPIRPO ユーティリティーの JCL 指定には、JOB ステートメント、EXEC ステートメント、および DD ステートメントが含まれます。オプションのユーティリティー制御ステートメントを 1 つ JCL ステートメントに含めることができます。

次のものがが必要です。

- JOBLIB DD ステートメントまたは STEPLIB DD ステートメント
- EXEC ステートメント
- 入力データ・セットと出力データ・セットを指定する DD ステートメント

**JOBLIB DD** ステートメントまたは **STEPLIB DD** ステートメント

JOBLIB DD ステートメントまたは STEPLIB DD ステートメントは、このユーティリティー・プログラムが入っているプログラム・ライブラリーを記述します。このステートメントの形式は次のとおりです。

```
//JOB LIB DD DSNAME=IMS.SDFSRESL,DISP=SHR
```

### **EXEC** ステートメント

EXEC ステートメントはこのユーティリティー・プログラムを呼び出します。このステートメントの形式は次のとおりです。

```
//PITRACE EXEC PGM=DFSPIRPO
```

### **DD** ステートメント

#### **LOGTAPE DD**

ログ入力データ・セットを記述します。ログが複数のデータ・セットにまたがる場合は、それらのデータ・セットは連結されます。形式は次のとおりです。

```
//LOGTAPE DD DSNAME=nnnn,DISP=OLD,VOL=SER=xxxxxx,  
// UNIT=YYYY
```

ただし、nnnn はデータ・セット名、xxxxxx はボリューム通し番号、YYYY はログ・データ・セットの入力タイプです。

#### **PRINT DD**

報告書データ・セットを記述します。形式は次のとおりです。

```
//PRINT DD SYSOUT=A
```

#### **SORTLIB DD**

ソート・プログラム・ライブラリーを記述します。形式は次のとおりです。

```
//SORTLIB DD DSNAME=SYS1.SORTLIB,DISP=SHR
```

#### **SYSOUT DD**

ソートのためのメッセージ出力データ・セットを記述します。フォーマットは次のとおりです。

```
//SYSOUT DD SYSOUT=A
```

#### **SORTWK01-32 DD**

ソート・プログラムの作業データ・セットを記述します。定義するスペースは、同じでなくても構いません。データ・セットは、少なくとも 3 つ必要です。これらのデータ・セットは直接アクセス装置に置いておくのが普通ですが、テープ・ボリュームを使用することも可能です。ディスク・ソートの場合、形式は次のとおりです。

```
//SORTWKnn DD UNIT=SYSDA,SPACE=(CYL,(5),,CONTIG)
```

ソート・プログラムの作業域はディスク上の別々の装置タイプにあってはいけないため、SYSDA は単一の装置タイプを参照しなければなりません。

#### **SYSPRINT DD**

システム・メッセージ・データ・セットを記述します。形式は次のとおりです。

```
//SYSPRINT DD SYSOUT=A
```


#### **SYSIN DD**

オプションのユーティリティー制御ステートメントが入っているデータ・セットを記述します。入力ストリームに入れておくときは、この DD ステートメントは DD \* にしておくのが普通です。オプションのユーティリティー制御ステートメントが省かれた場合は、このステートメントも省くことができます。

## 戻りコード

DFSPIRP0 ユーティリティーでは戻りコードはありません。

関連概念:

 プログラム分離およびロック・トレースの入手 (システム管理)

---

## DFSPIRP0 ユーティリティーの制御ステートメント

オプションの DFSPIRP0 ユーティリティー制御ステートメントは、報告書で必要とする開始時刻と停止時刻を指定するために使用します。この制御ステートメントは、プログラム出力に印刷されます。

DFSPIRP0 ユーティリティー制御ステートメントの形式は、次のとおりです。

```
▶▶ ┌──PRINT──┐ ┌──START=HHMM,──┐ ┌──STOP=HHMM,──┐ ┌──DATE=MM/DD──┐ ▶▶
```

START と STOP には、HH/MM で時 (00 から 99 まで) と分 (00 から 59 まで) を指定します。

DATE には、MM/DD で月 (01 から 12 まで) と日 (01 から指定された月の最大日数まで) を指定します。

PRINT は、命令フィールドに指定しなければなりません。キーワードは、コマで区切り、任意の順序で入れることができます。

制約事項: キーワードは反復できません。

重要: この時刻はすべて、トレース期間の開始を基準にしているので、このユーティリティーの制御ステートメントは変更されません。ただし、このユーティリティーを使用して、現地時間に変更を加えたトレース期間に対する報告書を作成すると問題が起こります。

データは、PRINT の前後に 1 つ以上のスペースを入れて、1 桁目から 71 桁目までに入れることができます。パラメーターの次にスペースを入れると、データの終わりを表します。このスペースの後に入るものは、コメントと見なされます。

PRINT とだけ指定された場合、制御ステートメントが省かれた (SYSIN データ・セットが指定されない) 場合、あるいはブランクの制御ステートメントが与えられている場合は、ログ・データ・セット全体が検索されます。PRINT と START だけが指定された場合、ログ・データ・セットは、開始時刻から最後まで検索されます。PRINT と STOP だけが指定された場合、ログ・データ・セットは、最初から停止時刻まで検索されます。

START 時刻と STOP 時刻は、指定された日付を基準にしており、DATE が指定されない場合は、PI トレースが開始された日付、すなわち PI トレース・ログ・レコードに記録されている日付を基準にしています。PRINT と DATE だけが指定された場合、ログ・データ・セットは、その日付の 00:00:00 から検索され、レコード

の有無が調べられます。START が指定されずに DATE と STOP だけが指定された場合、ログ・データ・セットは、その日付の 00:00:00 から、その日付を基準にした停止時刻になるまで検索されます。

指定する DATE は、PI トレースが開始された日付から 12 日以内の日付でなければなりません。

---

## DFSPIRPO ユーティリティーの例

以下の例は、DFSPIRPO ユーティリティーを使用するためのサンプル JCL を示します。

次の例では、待機を必要とし、PI トレースが開始された日付の午前 9 時と午前 9 時 30 分の間に要求されたすべての ENQ の報告書を要求しています。

```
//JOB LIB DD DSNAME=IMS.SDFSRESL,DISP=SHR
/*
// EXEC PGM=DFSPIRPO
//LOGTAPE DD DSNAME=IMSLLOG,DISP=OLD,
// UNIT=TAPE,VOL=SER=XXXXXX
//PRINT DD SYSOUT=A
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(5),,CONTIG)
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(5),,CONTIG)
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(5),,CONTIG)
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(5),,CONTIG)
//SORTWK05 DD UNIT=SYSDA,SPACE=(CYL,(5),,CONTIG)
//SORTWK06 DD UNIT=SYSDA,SPACE=(CYL,(5),,CONTIG)
//SYSOUT DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
//SORTLIB DD DSNAME=SYS1.SORTLIB,DISP=SHR
//SYSIN DD *
PRINT START=0900,STOP=0930
/*
```

午後 11 時 30 分から午前 1 時 10 分までの報告書が必要ならば、制御ステートメントは次のようになります。

```
PRINT START=2330,STOP=2510
```

以下の図は、PI トレースが 11 月 25 日に初めて開始された場合に内容がすべて同じになる例を示しています。

```
PRINT START=1000,STOP=1530,DATE=11/27
PRINT START=3400,STOP=3930,DATE=11/26
PRINT START=5800,STOP=6330
```

報告書見出しの日付は、PI トレースが初めて開始されたときのログからとられる日付です。報告書に記載される呼び出しの時刻および開始/停止時刻は、この日付を基準にした時刻です。

---

## 第 34 章 SB テスト・ユーティリティ (DFSSBHD0)

SB テスト・ユーティリティは、順次バッファリング (SB) をチューニングし、その問題判別を行うために使用します。このユーティリティを使用すると、アプリケーション・プログラムまたは別のユーティリティを前に実行したときに出された SB バッファ・ハンドラー呼び出しシーケンスを再度処理することができます。

SB テスト・ユーティリティは、入力として SB イメージ・キャプチャー・ログ・レコードを受け付けることができます。SB イメージ・キャプチャー・ログ・レコードは、IMS プログラムの JCL の //DFSCCTL ファイルに SBIC 制御ステートメントを含めることによって生成できます。そのプログラムの実行中に、SB バッファ・ハンドラーに対するすべての内部 IMS 呼び出しがキャプチャーされ、IMS ログに記録されます。

SB テスト・ユーティリティが処理できるのは、このユーティリティと同じ PSB を使用する 1 回のプログラム実行の SB イメージ・キャプチャー・ログ・レコードだけです。このユーティリティの入力データ・セットに同じ PSB を使用した複数回のプログラム実行からの SB イメージ・キャプチャー・ログ・レコードが入っている場合は、このユーティリティでプログラムのどの実行を行うかを指定する SELECT ユーティリティ制御ステートメントを含める必要があります。SELECT ステートメントを含めない場合、SB テスト・ユーティリティは、プログラムの最初の実行の SB イメージ・キャプチャー・ログ・レコードのみを処理します。

SB テスト・ユーティリティでは、SB バッファ・ハンドラー呼び出しは、通常のプログラムの実行時に出されたものと同じように処理されます。ただし、SB バッファ・ハンドラーは DB 読み取り入出力を出すよう指定する DBIO ユーティリティ制御ステートメントが含まれない限り、DB 読み取り入出力をシミュレートするだけです。

SB テスト・ユーティリティでは、アプリケーションによって生成された SB バッファ・ハンドラー呼び出しシーケンスを再作成し、再実行することができます。SB テスト・ユーティリティは、入出力参照パターンを分析する SB バッファ・ハンドラーのアルゴリズムに関する問題、さらには、他の SB バッファ・ハンドラー関連の問題を文書化し、修正テストを行うときに役立ちます。

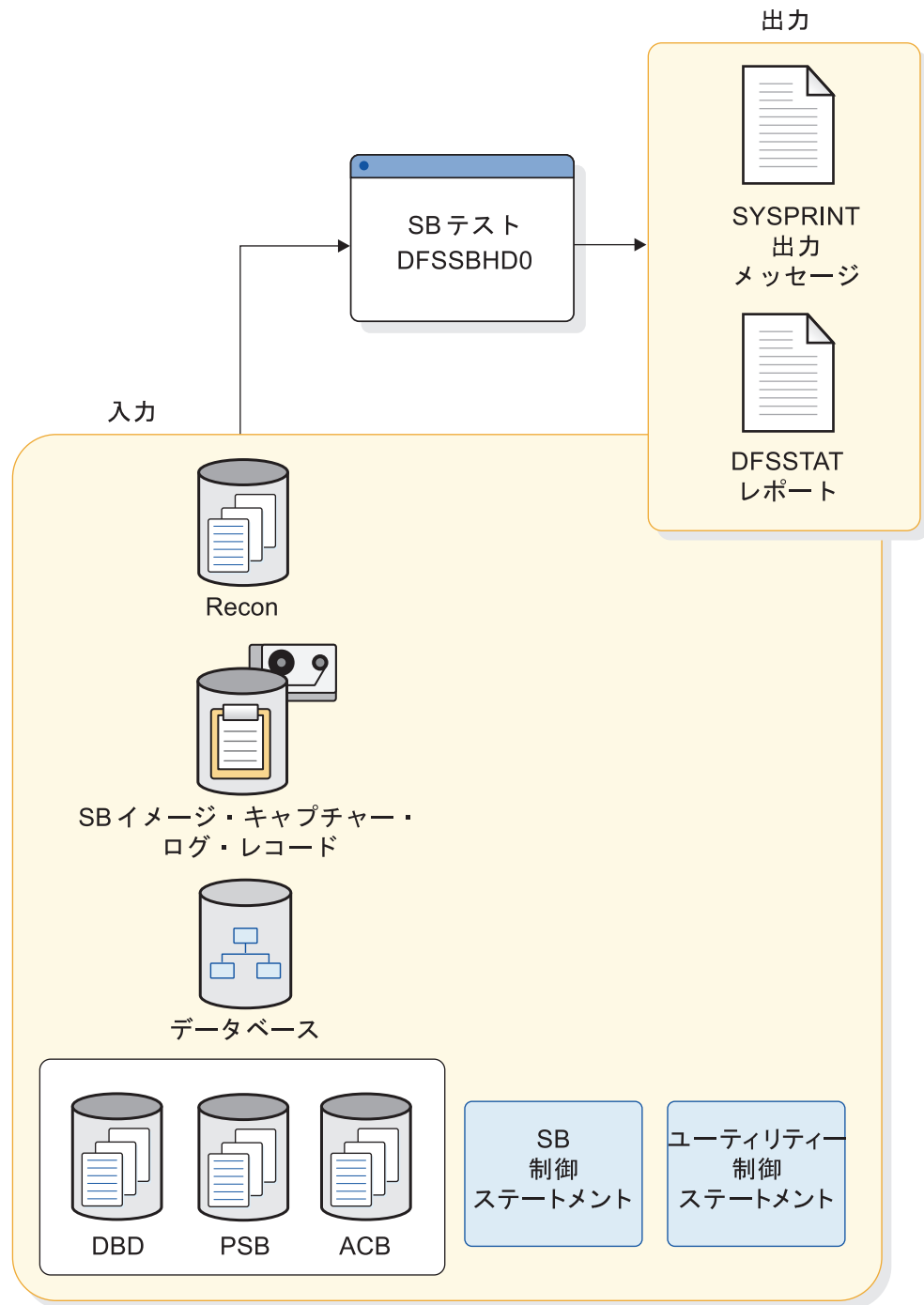
SB テスト・ユーティリティをチューニング時に使用すると、SB アルゴリズムの SB パラメーター値 (バッファ・セットの数) を変えてテストし、その変更による影響を //DFSSTAT 報告書で検討することができます。

**重要:** SB バッファ・ハンドラー呼び出しシーケンスを SB テスト・ユーティリティが再実行した場合、同じ SB バッファ・ハンドラー・パラメーター (BUFSETS 値) が使用されたとしても、元の実行と同数の順次読み取りとランダム読み取りが行われるとは限りません。これは、双方の実行時に無効とされた SB バッファが常に同じであるとは限らないからです。

SB テスト・ユーティリティーを実行しても、データベース・データ・セットは更新されません。このユーティリティーは、元の SB イメージ・キャプチャーからこのユーティリティーの実行までの間に行われたデータベース変更 (ISRT、REPL、DLET など) には影響されません。

以下の図は、SB テスト・ユーティリティーで必要になるデータ・セットを示しています。





領域タイプによる

図 75. SB テスト・ユーティリティーに必要なデータ・セット

以下のトピックに追加情報があります。

- 510 ページの『制約事項』
- 510 ページの『前提条件』
- 510 ページの『要件』
- 510 ページの『推奨事項』
- 510 ページの『入出力』

- 511 ページの『JCL 指定』
- 514 ページの『戻りコード』

## 制約事項

SB テスト・ユーティリティーには、次のような制約があります。

- SB テスト・ユーティリティーは、その実行ごとに、1 回のプログラム実行の SB イメージ・キャプチャー・ログ・レコードだけを処理します。
- このユーティリティーはバッチ領域のみで実行できます。
- このユーティリティーは、ロード処理オプションを指定する DB PCB については SB イメージ・キャプチャー・ログ・レコードを処理しません。
- このユーティリティーは、システム全体の動作ではなく、個々のプログラムの順次バッファリングの動作のみをシミュレートできます。グローバルなシステム最適化と、1 つのプログラムのローカルな SB 最適化とを混同しないでください。
- DFSSBHD0 ユーティリティーは、全機能 (HALDB および非 HALDB) データベースのみで機能します。

## 前提条件

SB イメージ・キャプチャー・ログ・レコードを SB テスト・ユーティリティーへの入力として使用できるようにするには、その前に、//DFSCTL ファイルに SBIC 制御ステートメントを含む JCL を使用してアプリケーション・プログラムまたはユーティリティーを実行する必要があります。

## 要件

アプリケーションまたはユーティリティーの SB バッファ・ハンドラー呼び出しをシミュレートするには、SB テスト・ユーティリティーが以下にアクセスできなければなりません。

- アプリケーションで使用する PSB
- アプリケーションまたはユーティリティーが参照するすべての DBD
- アプリケーションまたはユーティリティーがアクセスするデータベース・データ・セット

## 推奨事項

IMS モニターは SB テスト・ユーティリティーにとって重要な報告書を提供しないため、SB テスト・ユーティリティーの実行中は、IMS モニターを使用しないでください。

## 入出力

以下の表に SB テスト・ユーティリティーの入出力を示します。

表 29. SB テスト・ユーティリティーの入出力

入力	出力
RECON	DFSSTAT レポート
SB イメージ・キャプチャー・ログ・レコード データベース	SYSPRINT 出力メッセージ

表 29. SB テスト・ユーティリティの入出力 (続き)

入力	出力
DBDLIB、PSBLIB、ACBLIB (領域タイプによりいずれか)	
SB 制御ステートメント	
ユーティリティ制御ステートメント	

SB テスト・ユーティリティは、次の入力を使用します。

- SB イメージ・キャプチャー・ログ・レコードが入っている SYSUT1 データ・セット
- DBIO ステートメントと SELECT ユーティリティ制御ステートメントが入っているオプションの SYSIN ファイル。
- SB イメージ・キャプチャー・ログ・レコードを生成したアプリケーションによって使用される PSB、DBD およびデータベース

SB テスト・ユーティリティを実行すると、次のような出力が得られます。

- SB 要約報告書と SB 詳細報告書が入っている DFSSTAT ファイル
- SYSIN から読み取られたユーティリティ制御ステートメントを、このユーティリティによって書き込まれたメッセージと一緒にリストしている SYSPRINT ファイル

## JCL 指定

SB テスト・ユーティリティは、標準の z/OS ジョブとして実行されます。次のステートメントが必要です。

- ご使用のシステムの指定と一致するようにユーザーが定義する JOB ステートメント
- EXEC ステートメント
- 入力および出力を定義する DD ステートメント

### EXEC ステートメント

EXEC ステートメントは、次のいずれかの形式になっていなければなりません。

- SB イメージ・キャプチャー・ログ・レコードを生成したプログラムが PSBLIB からの PSB を使用して稼働していた場合  
`PGM=DFSRR00,PARM='DLI,DFSSBHD0,psbname'`
- SB イメージ・キャプチャー・ログ・レコードを生成したプログラムが ACBLIB からの PSB を使用して稼働していた場合  
`PGM=DFSRR00,PARM='DBB,DFSSBHD0,psbname'`
- SB イメージ・キャプチャー・ログ・レコードを生成したプログラムが PSB を使用しないで稼働しているユーティリティ・プログラムであった場合  
`PGM=DFSRR00,PARM='ULU,DFSSBHD0,dbdname'`

*psbname* パラメーターには、SB イメージ・キャプチャー・ログ・レコードを生成したアプリケーションによって使用された PSB の名前を指定します。*dbdname* パラ

メーターには、SB イメージ・キャプチャー・ログ・レコードを生成したユーティリティーによって使用された DBD の名前を指定します。

SPIE、BUF、DBRC といった通常の IMS 定位置パラメーターは、*psbname* または *dbdname* の後に置くことができます。

#### **DD** ステートメント

##### **STEPLIB DD**

IMS.SDFSRESL を指します。ここには、IMS 中核と必要なアクション・モジュールが入っています。無許可ライブラリーを IMS.SDFSRESL と連結させて STEPLIB を無許可にする場合は、DFSRESLB DD ステートメントが必要です。

##### **DFSRESLB DD**

IMS SVC モジュールが入っている許可ライブラリーを指します。

##### **IMS DD**

このユーティリティーを DLI または ULU 領域で実行させるとき使用される PSB または DBD が入っているライブラリーを定義します。

##### **IMSACB DD**

このユーティリティーを DBB 領域で実行させるとき使用される PSB と DBD の ACB が入っているライブラリーを定義します。

##### **DFSVSAMP DD**

DL/I バッファ・ハンドラーが必要とするバッファ情報が入っているデータ・セットを記述します。この DD ステートメントは必須です。

##### **DFSCCTL DD**

SB 制御ステートメントが入っているカード・イメージ・ファイルを指します。これは、順次データ・セットにすることも、PDS のメンバーにすることもできます。レコード形式は F、FB、または FBS のいずれかで、レコード長は 80 でなければなりません。

SBIC 制御ステートメントは、このユーティリティーへの入力として必要な SB イメージ・キャプチャー・ログ・レコードを作成するために、実行されるプログラムの //DFSCCTL 入力データ・セットに入れておかなければなりません。

SB テストが問題判別のために使用されているときは、SBSNAP、SBESNAP および SNAPDEST 制御ステートメントは、必要に応じて、//DFSCCTL カード・イメージ入力データ・セットに入れておくことができます。

##### **SYSABEND DD または SYSUDUMP DD**

ダンプ・データ・セットを定義します。これらの DD ステートメントはオプションです。両方のステートメントが存在するときは、最後に出現するステートメントがダンプで使用されます。

##### **IEFRDR DD**

順次バッファのテスト時に作成されるシステム・ログを記述します。このデータ・セットは、テープまたは DASD に置かれています。

##### **IEFRDR2 DD**

順次バッファのテスト時に作成される 2 次システム・ログを記述します。このデータ・セットは、テープまたは DASD に置かれています。このステートメントは、デュアル・ログ出力が必要な場合にのみ使用します。

### **RECON1 DD**

最初の DBRC (データベース・リカバリー管理) RECON データ・セットを定義します。

動的割り振りを使用しているときは、これらの RECON データ・セット DD 名を使用しないでください。

### **RECON2 DD**

2 番目の DBRC RECON データ・セットを定義します。

### **RECON3 DD**

RECON1 または RECON2 にエラーが見付かったとき使用される、オプションの DBRC RECON データ・セットを定義します。この RECON データ・セットは、制御領域が使用しているものと同じ RECON データ・セットでなければなりません。

### **database DD**

SB イメージ・キャプチャー・ログ・レコードを生成したアプリケーションまたはユーティリティーによって使用されるデータベース・データ・セットを定義します。

### **SYSUT1 DD**

SB イメージ・キャプチャー・ログ・レコードが入っているデータ・セットを定義します。この DD ステートメントの指定は必須です。このデータ・セットの代表例としては、IMS ログ・データ・セットまたは IMS ログ・データ・セットから抽出したもののどちらかです。このデータ・セットは、テープに置くことも、DASD に置くこともできます。

### **SYSIN DD**

ユーティリティー制御ステートメントが入っているカード・イメージ・データ・セットを指示します。このデータ・セットは、テープまたは DASD に置いておくことも、入力ストリームに入れて送ることもできます。データ・セットのレコード形式は F、FB、または FBS のいずれかで、レコード・サイズは 80 でなければなりません。この DD ステートメントはオプションです。

### **SYSPRINT DD**

メッセージ出力データ・セットを定義します。このデータ・セットは、テープ、DASD、またはプリンターに置いておくことも、出力ストリームに入れて送ることもできます。以下の DCB パラメーターは、DFSSBHD0 から指定されるものであり、この DD ステートメントに指定しないでください。

- RECFM=FBA
- LRECL=121
- BLKSIZE=605

### **DFSSTAT DD**

//DFSSTAT ファイルを指します。このファイルは、順次バッファリング統計を書き込むのに使用されるものです。このステートメントの DCB 属性は、IMS モジュールによって次のように設定されます。

- RECFM=FBA
- LRECL=133
- BLKSIZE=1330

## 戻りコード

現在、DFSSBHD0 ユーティリティに対して文書化された戻りコードはありません。

関連概念:

➡ IMS バッファ・プール (システム定義)

関連資料:

➡ 順次バッファリング制御ステートメント (システム定義)

➡ DBBBATCH プロシージャ (システム定義)

➡ DLIBATCH プロシージャ (システム定義)

---

## DFSSBHD0 ユーティリティの制御ステートメント

DFSSBHD0 ユーティリティは、次の 2 つの制御ステートメントを使用します:  
DBIO および SELECT。

DBIO ステートメントは、SB テスト・ユーティリティが実際の DB 読み取り入出力呼び出しを使用するのか、シミュレートした読み取り入出力呼び出しを使用するのかを指定するために使用します。SELECT ステートメントは、SB イメージ・キャプチャー・ログに複数のプログラム実行が含まれている場合に特定のプログラム実行を選択するために使用します。

### DBIO ステートメント

DBIO ステートメントは、SB バッファ・ハンドラーが実際の DB 読み取り入出力を出すのか、あるいは DB 読み取り入出力をシミュレートするだけなのかを指定するのに使用します。実際の DB 読み取り入出力を出すと、ジョブ・ステップの経過時間が大幅に増加します。ただし、チューニング時にバッファ・セットの数を覚えてユーティリティを複数回実行してみることによって、DB 読み取り入出力の時間を測定し、比較することができます。DBIO ステートメントの形式は次のとおりです。



### YES

SB バッファ・ハンドラーが DB 読み取り入出力を出すことを指定します。

YES を指定すると、OSAM バッファ・ハンドラーから出された OSAM 読み取り入出力操作だけが出力されます。VSAM データベース・データ・セットに対する入出力操作と出力の DB 入出力操作は、このユーティリティの実行中には出されません。

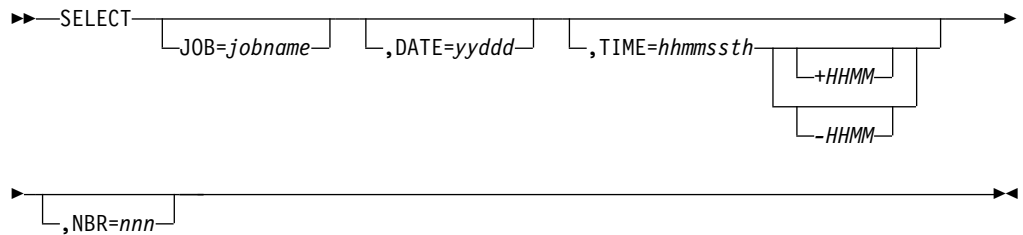
**重要:** SB イメージ・キャプチャーが行われてからこのユーティリティが実行されるまでに DB が再編成されるか、再ロードされており、その新しい DB バージョンの高位使用相対バイト・アドレス (RBA) が元の DB バージョンのそれよりも小さくなっている場合は、DBIO YES は使用しないでください。その

理由は、実際の高位使用 RBA よりも高い RBA の場合、SB バッファ・ハンドラーの呼び出しを記述している SB イメージ・キャプチャー・ログ・レコードは、DBIO YES が指定されると、処理されないからです。そのため、SB バッファ・ハンドラーの呼び出しシーケンスは、このユーティリティーの実行時と元のプログラムの実行時とは同じでないため、比較できません。

**NO** SB バッファ・ハンドラーが DB 読み取り入出力をシミュレートするだけであることを指定します。NO がデフォルトです。DBIO NO が指定されていても、このユーティリティーは、DB データ・セットにアクセスしなければなりません。

## SELECT ステートメント

SELECT ステートメントは、同じ PSB を使用してプログラムを複数回実行することによって生成された SB イメージ・キャプチャー・ログ・レコードが入力データ・セットに入っているときに使用します。このステートメントを使用して、特定のアプリケーション実行の SB イメージ・キャプチャー・ログ・レコードだけを選択してください。SELECT ステートメントは 1 つしか指定できません。SELECT ステートメントが指定されないと、最初のプログラム実行が選択されます。SELECT ステートメントの形式は次のとおりです。



SELECT というワードは 1 桁目から始め、その後に空白を 1 つだけ置き、その後 1 つ以上のキーワード・パラメータを続けなければなりません。複数のキーワード・パラメータは、間を 1 個のコンマ (,) で区切り、最後のキーワード・パラメータの後に少なくとも 1 個の空白を置かなければなりません。キーワード・パラメータはどのような順序でも構いません。

制約事項: SELECT ステートメントは次の行に継続できません。

### JOB=

どのアプリケーションの SB イメージ・キャプチャー・ログ・レコードを選択するのかを、その jobname で指定します。

### DATE=

そのアプリケーションが始動されたときの日付を指定します。

### TIME=

そのアプリケーションが始動されたときの大体の時刻を指定します。この値は、すべての先行ゼロと後続ゼロを含め、8 桁の数字で指定しなければなりません。つまり、hh (時)、mm (分)、ss (秒)、t (10 分の 1 秒)、h (100 分の 1 秒) として指定します。このキーワードは、DATE= キーワードを付けても指定できますが、付けなくても構いません。

TIME= キーワードが指定されていると、このユーティリティーは、指定された時刻以前に開始されたプログラム実行のどのプログラム開始レコードも無視します。次の UTC からのオフセットはオプションです。現在の UTC オフセットが、夏時間の開始または終了の調整によって、イメージ・キャプチャー・ログ・レコードが作成されたときに有効であった値と異なる場合にのみ、必要です。

+|-

UTC への時間帯のオフセットの符号を指定します。

HH UTC へのオフセットの総時間数を指定します。

MM オフセットの分単位の時間を指定します。これは、00、15、30、45 にできます。

NBR=

特定のプログラムが領域内で、しかも 100 分の 1 秒以内に、 $n^{\text{th}}$  番目に行われたときの SB イメージ・キャプチャー・ログ・レコードを選択することを指定します。このキーワードは、他のキーワードでは、選択するアプリケーション開始レコードを十分に固有に識別できない場合に使用すると便利です。

---

## DFSSBHD0 ユーティリティーの例

これらの例は、DFSSBHD0 ユーティリティーのためのサンプル JCL を示しています。

以下の図は、SB テスト・ユーティリティーを実行するための JCL を示しています。

図 76. SB テスト・ユーティリティーを実行するための JCL

```
// EXEC DLIBATCH,MBR=DFSSBHD0,PSB=xxxxxxx
//SKILLDB DD DSN=user.db.data.dsname,DISP=SHR
//SKILLIX DD DSN=user.db.indx.dsname,DISP=SHR
//SYSUT1 DD DSN=imslog,DISP=SHR
//DFSVSAMP DD *
4096,6
//DFSCCTL DD *
SBPARAM ACTIV=COND,BUFSETS=10
//SYSIN DD *
DBIO YES
//DFSSTAT DD SYSOUT=A
//SYSPRINT DD SYSOUT=A
/*
```



---

## 第 6 部 付録



---

## 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す。本書の他言語版を IBM から入手できる場合があります。ただし、ご利用にはその言語版の製品もしくは製品のコピーを所有していることが必要な場合があります。

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。

〒103-8510

東京都中央区日本橋箱崎町19番21号

日本アイ・ビー・エム株式会社

法務・知的財産

知的財産権ライセンス渉外

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。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

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 Director of Licensing*  
*IBM Corporation*  
*North Castle Drive, MD-NC119*  
*Armonk, NY 10504-1785*  
*US*

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せんが、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。

記載されている性能データとお客様事例は、例として示す目的でのみ提供されています。実際の結果は特定の構成や稼働条件によって異なります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名前はすべて架空のものであり、類似する個人や企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。これらのサンプル・プログラムは特定物として現存するままの状態を提供されるものであり、いかなる保証も提供されません。IBM は、お客様の当該サンプル・プログラムの使用から生ずるいかなる損害に対しても一切の責任を負いません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (年).

このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。

© Copyright IBM Corp. \_年を入れる\_.

---

## プログラミング・インターフェース情報

本書では、IMS によって提供されるプロダクト・センシティブ・プログラミング・インターフェースとそれに関連する情報を記述しています。

プロダクト・センシティブ・プログラミング・インターフェースにより、お客様のインストール済み環境で、このソフトウェア製品の診断、修正、モニター、修復、調整、またはチューニングなどの作業を実行することができます。これらのインターフェースを使用すると、IBM のソフトウェア製品の詳細設計や実装に対する依存関係が生じます。このためプロダクト・センシティブ・プログラミング・インターフェースは上記の特別な目的にだけ使用してください。詳細設計やその実現方法に依存しているので、このようなインターフェースに合わせて作成したプログラムは、新しい製品のリリース、バージョンで実行するとき、または保守サービスの結果として、変更が必要になることがあります。プロダクト・センシティブ・プログラミング・インターフェースとそれに関連する情報は、セクションやトピックの単位の場合はその冒頭で識別され、それ以外の場合は「プロダクト・センシティブ・プログラミング・インターフェース」というマーケティングで識別されます。IBM では、上記の冒頭部での識別の記述、およびその記述を参照する本書内のすべての記述を、そのような記述によって示される全体コピーまたは部分コピーに含めるよう求めています。

---

## 商標

IBM、IBM ロゴおよび [ibm.com](http://www.ibm.com)<sup>®</sup> は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。

Adobe、Adobe ロゴ、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

Java<sup>™</sup> およびすべての Java 関連の商標およびロゴは Oracle やその関連会社の米国およびその他の国における商標または登録商標です。

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。

---

## 製品資料に関するご使用条件

これらの資料は、以下のご使用条件に同意していただける場合に限りご使用いただけます。

### 適用される条件

このご使用条件は、IBM Web サイトのすべてのご利用条件に追加して適用されません。

### 個人使用

これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布（頒布、送信を含む）または表示（上映を含む）することはできません。

### 商業的使用

これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

### 権利

ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。

お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入 関連法規を含む、すべての関連法規を遵守するものとします。

IBM は、これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態を提供され、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なしで提供されます。

---

## IBM オンライン・プライバシー・ステートメント

サービス・ソリューションとしてのソフトウェアも含めた IBM ソフトウェア製品（「ソフトウェア・オファリング」）では、製品の使用に関する情報の収集、エンド・ユーザーの使用感の向上、エンド・ユーザーとの対話またはその他の目的のために、Cookie はじめさまざまなテクノロジーを使用することがあります。多くの場合、ソフトウェア・オファリングにより個人情報が収集されることはありません。

IBM の「ソフトウェア・オファリング」の一部には、個人情報を収集できる機能を持つものがあります。ご使用の「ソフトウェア・オファリング」が、これらの Cookie およびそれに類するテクノロジーを通じてお客様による個人情報の収集を可能にする場合、以下の具体的事項をご確認ください。

この「ソフトウェア・オファリング」は、Cookie もしくはその他のテクノロジーを使用して個人情報を収集することはありません。

この「ソフトウェア・オファリング」が Cookie およびさまざまなテクノロジーを使用してエンド・ユーザーから個人を特定できる情報を収集する機能を提供する場合、お客様は、このような情報を収集するにあたって適用される法律、ガイドライン等を遵守する必要があります。これには、エンドユーザーへの通知や同意の要求も含まれますがそれらには限られません。

このような目的での Cookie を含む様々なテクノロジーの使用の詳細については、IBM の『IBM オンラインでのプライバシー・ステートメント』(<http://www.ibm.com/privacy/details/jp/ja/>) の『クッキー、ウェブ・ビーコン、その他のテクノロジー』および『IBM Software Products and Software-as-a-Service Privacy Statement』(<http://www.ibm.com/privacy/details>) を参照してください。





## 参考文献

この参考文献のリストには、IMS バージョン 13 ライブラリーのすべての資料、補足資料、資料コレクション、および IMS バージョン 13 ライブラリーで引用されているアクセシビリティ関連の資料が記載されています。

表題	頭字語	資料番号
IMS V13 アプリケーション・プログラミング	APG	SA88-5465
IMS V13 アプリケーション・プログラミング API	APR	SA88-5463
IMS V13 コマンド 第 1 巻: IMS コマンド A-M	CR1	SA88-5466
IMS V13 コマンド 第 2 巻: IMS コマンド N-V	CR2	SA88-5467
IMS V13 コマンド第 3 巻: IMS コンポーネント および z/OS コマンド	CR3	SA88-5472
IMS V13 コミュニケーションおよびコネクション	CCG	SA88-5482
IMS V13 データベース管理	DAG	SA88-5485
IMS V13 データベース・ユーティリティー	DUR	SA88-5492
IMS Version 13 Diagnosis	DGR	GC19-3654
IMS V13 出口ルーチン	ERR	SA88-5494
IMS V13 インストール	INS	GA88-5697
IMS Version 13 Licensed Program Specifications	LPS	GC19-3663
IMS V13 メッセージおよびコード 第 1 巻: DFS メッセージ	MC1	GC43-1535
IMS V13 メッセージおよびコード 第 2 巻: DFS 以外メッセージ	MC2	GC43-1536
IMS V13 メッセージおよびコード 第 3 巻: IMS 異常終了コード	MC3	GC43-1537
IMS V13 メッセージおよびコード 第 4 巻: IMS コンポーネント・コード	MC4	GC43-1538
IMS V13 オペレーションおよびオートメーション	OAG	SA88-7059
IMS V13 リリース計画	RPG	GA88-7071
IMS V13 システム管理	SAG	SA88-7072
IMS V13 システム定義	SDG	GA88-7073
IMS V13 システム・プログラミング API	SPR	SA88-7077
IMS V13 システム・ユーティリティー	SUR	SA88-7100

## 補足資料

表題	資料番号
<i>Program Directory for Information Management System Transaction and Database Servers V13.0</i>	GI10-8914
<i>Program Directory for Information Management System Transaction and Database Servers V13.0 Database Value Unit Edition V13R1</i>	GI10-8966
<i>Program Directory for Information Management System Transaction and Database Servers V13.0 Transaction Manager Value Unit Edition V13R1</i>	GI10-9001
IRLM メッセージおよびコード	GC19-2666

## 資料コレクション

表題	フォーマット	資料番号
IMS バージョン 13 製品キット	CD	SK5T-8864

## IMS バージョン 13 ライブラリーで引用されているアクセシビリティ ー関連の資料

表題	資料番号
z/OS TSO/E 入門	SA88-8632
z/OS TSO/E ユーザーズ・ガイド	SA88-8638
z/OS 対話式システム生産性向上機能 (ISPF) ユーザーズ・ガイド 第 1 巻	SC88-8965

## 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  
キーボード・ショートカット x  
機能 x  
イメージ・コピー  
オンライン・データベース・イメージ・コピー・ユーティリティ  
(DFSUICP0) 205  
データベース・イメージ・コピー 2 ユーティリティ (DFSUDMT0) 177  
コピー完了通知 200  
データベース・イメージ・コピー・ユーティリティ (DFSUDMP0) 163  
オンライン再編成 (OLR)  
HALDB (高可用性ラージ・データベース)  
PDU を使用した DBRC での OLR 機能の登録 121  
オンライン・データベース・イメージ・コピー・ユーティリティ (DFSUICP0)  
出力 207  
推奨事項 207  
制約事項 206  
説明 205  
前提条件 206  
戻りコード 208  
ユーティリティ制御機能の使用が制限された 437  
ユーティリティ制御ステートメント 211  
要件 206  
リカバリーおよび再始動 213  
例 212  
JCL の要件  
DD ステートメント 209  
EXEC ステートメント 208  
QUIESCE データベース状況 206

### [カ行]

間接リスト・データ・セット (ILDS)  
再作成  
フリー・スペースのオプション  
297, 341

間接リスト・データ・セット (ILDS) (続き)  
再作成 (続き)  
VSAM ロード・モード 297, 341  
パフォーマンス  
フリー・スペース, 再作成 297, 342  
フリー・スペース  
再作成中にフリー・スペースを含める 297, 341  
キーボード・ショートカット x  
区画定義ユーティリティ  
HALDB (高可用性ラージ・データベース)  
DBRC での OLR 機能の登録 121  
区画デフォルト情報画面  
イメージ・コピー JCL 124  
受け取り JCL 124  
オンライン・イメージ・コピー JCL 124  
再使用可能? 124  
自動定義 125  
データベース名 122  
Anchor 122  
automatic definition 122  
block size 123  
Bytes 123  
data set name prefix 122  
default JCL 124  
free block freq. factor 123  
Free space percentage 123  
high block number 123  
input data set 122  
max. image copies 123  
module name 122  
partition ID 122  
recovery period 124  
recovery utility JCL 124  
use defaults for DS groups 122  
グループ名ステートメント  
ユーティリティ制御ステートメント, DFSUDMT0 191  
高可用性ラージ・データベース (HALDB)  
パフォーマンス  
ILDS 297  
DFSMAID0 (HALDB マイグレーション・エイド・ユーティリティ) 321  
HALDB 索引/ILDS 再作成ユーティリティ 291  
HALDB マイグレーション・エイド・ユーティリティ (DFSMAID0) 321

高可用性ラージ・データベース (HALDB) (続き)  
ILDS  
再作成, DFSPREC0 291  
構成  
HALDB (高可用性ラージ・データベース) 143  
高速 DEDB 直接再編成ユーティリティ (DBFUHDR0)  
エラー処理 379  
概要 369  
収集された統計 370  
推奨事項 370  
制御ステートメント 373  
制約事項 369  
セグメント・シャント 377  
前提条件 370  
入出力 370  
バッファの使用法 376  
戻りコード 373  
ユーティリティの実行 376  
要件 370  
リカバリーおよび再始動 379  
BUFNO コマンド 376  
DD ステートメント 372  
EXEC ステートメント 372  
JCL 指定 371  
JCL の例 372  
QUIESCE データベース状況 369  
高速機能  
DEDB ユーティリティ  
コマンドの継続 69, 70, 77, 89, 279, 287, 374  
MSDB ユーティリティ  
MSDBINIT DATASET 154  
高速機能 (Fast Path)  
DEDB 順次従属削除ユーティリティ (DBFUMDL0) 75  
DEDB ユーティリティ  
高速直接再編成ユーティリティ 369  
順次従属スキャン・ユーティリティ (DBFUMSC0) 85  
初期設定ユーティリティ (DBFUMIN0) 65  
DEDB エリア・データ・セット作成ユーティリティ (DBFUMRI0) 283  
DEDB エリア・データ・セット比較ユーティリティ (DBFUMMH0) 275

高速機能 (Fast Path) (続き)

DEDB、再編成

高速直接再編成ユーティリティ  
369

MSDB ユーティリティ

MSDB ダンプ・リカバリー・ユー  
ティリティ (DBFDBDR0) 299

MSDB 保守ユーティリティ  
(DBFDBMA0) 147

MSDB - DEDB 間変換ユーティリ  
ティ (DBFUCDB0) 407

高速機能データベース 3

高速複製

データベース・イメージ・コピー 2 ユ  
ーティリティ (DFSUDMT0)

動的データ・セット名 200

高速複製機能

DBDS 選択ステートメント 189

構文図

読み方 viii

## [サ行]

再編成

DEDB

セグメント・シャント 377

再ロード・ユーティリティ

(DFSURGL0)

統計 342

報告書 342

サンプリング

HALDB マイグレーション・エイド・  
ユーティリティ (DFSMAID0) 328

順次バッファリング

チューニング

SB テスト・ユーティリティ  
(DFSSBHD0) 507

テスト

SB テスト・ユーティリティ  
(DFSSBHD0) 507

問題判別

SB テスト・ユーティリティ  
(DFSSBHD0) 507

SB テスト・ユーティリティ

(DFSSBHD0) 507

商標 519, 521

制御ステートメント

オンライン・データベース・イメ  
ージ・コピー・ユーティリティ  
(DFSUICP0) 211

データベース事前再編成ユーティリ  
ティ (DFSURPR0)

概要 317

データベース接頭部更新ユーティリ  
ティ (DFSURGP0)

概要 21

制御ステートメント (続き)

データベース調査ユーティリティ  
(DFSPRSUR) 43

データベース部分再編成ユーティリ  
ティ (DFSPRCT1 および  
DFSPRCT2) 427

データベース変更累積ユーティリ  
ティ (DFSUCUM0)

概要 243

データベース・イメージ・コピー 2 ユ  
ーティリティ (DFSUDMT0)

概要 186

データベース・イメージ・コピー・ユ  
ーティリティ (DFSUDMP0) 174

データベース・スキャン・ユーティ  
リティ (DFSURGS0)

概要 35

データベース・リカバリー・ユーテ  
ィリティ (DFSURDB0)

概要 267

バッチ・バックアウト・ユーティリ  
ティ (DFSBB000)

概要 228

プログラム分離トレース報告ユーテ  
ィリティ (DFSPIRP0) 505

ユーティリティ制御機能

(DFSUCF00)

概要 448

DBFDBDR0 (MSDB ダンプ・リカバ  
リー・ユーティリティ) 304

DBFDBMA0 (MSDB 保守ユーティ  
リティ) 152

DBFUCDB0 (MSDB - DEDB 間変換  
ユーティリティ) 411

DBFUMIN0 (DEDB 初期設定ユーテ  
ィリティ) 70

DEDB 初期設定ユーティリティ  
(DBFUMIN0) 70

DFSBB000 (バッチ・バックアウト・  
ユーティリティ)

概要 228

DFSMAID0 (HALDB マイグレーシ  
ョン・エイド・ユーティリティ) 325

DFSPIRP0 (プログラム分離トレース報  
告ユーティリティ) 505

DFSPRCT1 および DFSPRCT2 (デー  
タベース部分再編成ユーティリ  
ティ) 427

DFSPREC0 (HALDB 索引/ILDS 再作  
成ユーティリティ) 295

DFSPRSUR (データベース調査ユーテ  
ィリティ) 43

DFSSBHD0 (SB テスト・ユーティ  
リティ)

概要 514

制御ステートメント (続き)

DFSUCF00 (ユーティリティ制御機  
能)

概要 448

DFSUDMP0 (データベース・イメ  
ージ・コピー・ユーティリティ) 174

DFSUDMT0 (データベース・イメ  
ージ・コピー 2 ユーティリティ)

概要 186

DFSUICP0 (オンライン・データベー  
ス・イメージ・コピー・ユーティ  
リティ) 211

DFSUPNT0 (HALDB 区画データ・セ  
ット初期設定ユーティリティ) 104

DFSURGL0 (HD 再編成再ロード・ユ  
ーティリティ) 339

DFSURGP0 (データベース接頭部更新  
ユーティリティ)

概要 21

DFSURGS0 (データベース・スキャ  
ン・ユーティリティ)

概要 35

DFSURGU0 (HD 再編成アンロード・  
ユーティリティ) 361

DFSURPR0 (データベース事前再編成  
ユーティリティ)

概要 317

DFSURRL0 (HISAM 再編成再ロ  
ード・ユーティリティ) 388

DFSURUL0 (HISAM 再編成アンロ  
ード・ユーティリティ)

概要 401

HALDB 区画データ・セット初期設定  
ユーティリティ (DFSUPNT0) 104

HALDB 区画定義ユーティリティ  
(%DFSHALDB) 113

HALDB 索引/ILDS 再作成ユーティ  
リティ (DFSPREC0) 295

HALDB マイグレーション・エイド・  
ユーティリティ (DFSMAID0) 325

HD 再編成アンロード・ユーティリ  
ティ (DFSURGU0) 361

HD 再編成再ロード・ユーティリ  
ティ (DFSURGL0) 339

HISAM 再編成アンロード・ユーテ  
ィリティ (DFSURUL0)

概要 401

HISAM 再編成再ロード・ユーティ  
リティ (DFSURRL0) 388

MSDB ダンプ・リカバリー・ユーテ  
ィリティ (DBFDBDR0) 304

MSDB 保守ユーティリティ  
(DBFDBMA0) 152

MSDB - DEDB 間変換ユーティリ  
ティ (DBFUCDB0) 411

制御ステートメント (続き)  
SB テスト・ユーティリティ  
(DFSSBHD0)  
概要 514  
%DFSHALDB (HALDB 区画定義ユー  
ティリティ) 113  
セグメント  
セグメント・シャント、DEDB 377  
セグメント・シャント 377

## [夕行]

データベース  
再編成  
DEDB 369  
バックアップ  
オンライン・データベース・イメー  
ジ・コピー・ユーティリティ  
(DFSUICP0) 205  
データベース・イメージ・コピー 2  
ユーティリティ  
(DFSUDMT0) 177  
データベース・イメージ・コピー・  
ユーティリティ  
(DFSUDMP0) 163  
変換  
DEDB を MSDB に 415  
MSDB を DEDB に 415  
DEDB  
高速直接再編成ユーティリティ  
369  
データベース ZAP 機能  
ユーティリティ制御機能  
(DFSUCF00) 492  
データベース事前再編成ユーティリティ  
(DFSURPR0)  
出力メッセージ 313  
制約事項 312  
説明 311  
戻りコード 316  
ユーティリティ制御ステートメント  
概要 317  
DBIL= 317  
DBR= 317  
OPTIONS= 318  
例 319  
JCL の要件  
DD ステートメント 315  
EXEC ステートメント 315  
QUIESCE データベース状況 312  
データベース接頭部解決ユーティリティ  
(DFSURG10)  
概要 3  
推奨事項 6  
制約事項 5  
前提条件 6

データベース接頭部解決ユーティリティ  
(DFSURG10) (続き)  
入出力 6  
戻りコード 7  
要件 6  
EXEC ステートメントの PARM フィ  
ールド・オプション 8  
EXEC ステートメントの例 8  
JCL の要件  
DD ステートメント 10  
EXEC ステートメント 8  
JCL の例 13  
QUIESCE データベース状況 5  
z/OS ソート/マージ・プログラム 8  
データベース接頭部更新ユーティリティ  
(DFSURGP0)  
概要 15  
再始動 27  
推奨事項 17  
制約事項 16  
前提条件 17  
入出力 17  
戻りコード 18  
ユーティリティ制御ステートメント  
概要 21  
ABEND 22  
CHKPT 21  
RSTRT 21  
SNAP 21  
要件 17  
例 22  
JCL の要件  
DD ステートメント 18  
EXEC ステートメント 18  
QUIESCE データベース状況 16  
データベース調査ユーティリティ  
(DFSRSUR)  
出力 40  
推奨事項 39  
制約事項 39  
説明 39  
前提条件 39  
入力 40  
戻りコード 40  
ユーティリティ制御機能の使用が制  
限された 437  
ユーティリティ制御ステートメント  
43  
要件 39  
例 45  
JCL の要件  
DD ステートメント 41  
EXEC ステートメント 41  
QUIESCE データベース状況 39  
データベースの再編成 309  
データベースのバックアップ 161

データベースの報告書作成とテスト 495  
データベースのリカバリー 215  
データベース部分再編成ユーティリティ  
(DFSPRCT1 および DFSPRCT2)  
推奨事項 420  
ステップ 1 事前再編成 420  
ステップ 2 アンロード/再ロード/ポ  
インター解決 421  
制約事項 420  
説明 419  
前提条件 420  
チェックポイント/再始動 419  
戻りコード 426  
ユーティリティ制御機能の使用が制  
限された 437  
ユーティリティ制御ステートメント  
427  
要件 420  
例 429  
DD ステートメント (ステップ 2) 423  
JCL の要件  
DD ステートメント (ステップ  
1) 422  
EXEC ステートメント (ステップ  
1) 422  
EXEC ステートメント (ステップ  
2) 423  
QUIESCE データベース状況 420  
データベース部分再編成ユーティリティ  
(DFSPRCT1)  
ステップ 1 のユーティリティ制御ス  
テートメント 427  
データベース部分再編成ユーティリティ  
(DFSPRCT2)  
ステップ 2 のユーティリティ制御ス  
テートメント 427  
データベース変更累積ユーティリティ  
(DFSUCUM0)  
除去の日付と時刻 244  
推奨事項 237  
制約事項 237  
説明 235  
前提条件 237  
入出力 237  
戻りコード 242  
ユーティリティ制御機能の使用が制  
限された 437  
ユーティリティ制御ステートメント  
概要 243  
DB0 ステートメント 246, 247  
DB1 ステートメント 248  
ID ステートメント 245  
SO ステートメント 249  
要件 237  
例 250

データベース変更累積ユーティリティ  
(DFSUCUM0) (続き)  
JCL の要件  
DD ステートメント 240  
EXEC ステートメント 238  
QUIESCE データベース状況 235  
データベース・イメージ・コピー 2 ユー  
ティリティ (DFSUDMT0)  
グループ名 199  
高速複製機能 197  
コピー完了通知 200  
推奨事項 179  
制約事項 177  
説明 177  
前提条件 178  
データ・セット、複数入力 198  
同一データ・セット・オプション 199  
動的データ・セット名 200  
入出力 180  
並行コピー機能 196  
ユーティリティ制御ステートメント  
概要 186  
グループ名ステートメント 191  
高速複製 DBDS 選択ステートメン  
ト 189  
並行コピー DBDS 選択ステートメン  
ト 186  
HLQ 仕様ステートメント 193  
SET PATCH 制御ステートメント  
195  
要件 178  
例 185  
DFSMSdss SET PATCH コマンド  
203  
JCL の要件  
DD ステートメント 181  
EXEC ステートメント 181  
QUIESCE データベース状況 177  
データベース・イメージ・コピー・ユーテ  
ィリティ (DFSUDMP0)  
制約事項 164  
説明 163  
前提条件 165  
入出力 167  
戻りコード 168, 180  
ユーティリティ制御ステートメント  
174  
要件 166  
リモート・サイト・リカバリー 163  
例 175  
JCL 指定 168  
JCL の要件  
DD ステートメント 171  
EXEC ステートメント 168  
QUIESCE データベース状況 164

データベース・スキャン・ユーティリティ  
(DFSURGS0)  
再始動 38  
出力メッセージ 31  
推奨事項 31  
スキャン・オプション  
SEG 35  
SEQ 35  
制約事項 31  
説明 29  
前提条件 31  
入出力 31  
戻りコード 31  
ユーティリティ制御ステートメント  
概要 35  
ABEND 37  
CHKPT 36  
DBS 35  
RSTRT 37  
要件 31  
JCL の指定および例 32  
JCL の要件  
DD ステートメント 32  
EXEC ステートメント 32  
JCL の例 38  
QUIESCE データベース状況 31  
データベース・モニター報告書印刷ユーテ  
ィリティ (DFSUTR30)  
推奨事項 498  
制約事項 497  
説明 497  
前提条件 497  
入出力 498  
分析制御データ・セット 499  
戻りコード 499  
要件 498  
例 499  
JCL 指定 498  
JCL の要件  
DD ステートメント 498  
EXEC ステートメント 498  
データベース・リカバリー・ユーティリテ  
ィ (DFSURDB0)  
推奨事項 260  
制約事項 257  
説明 255  
前提条件 257  
入出力 260  
戻りコード 266  
ユーティリティ制御ステートメント  
概要 267  
ABEND 267  
NOSEQCK 267  
S (データベース・リカバリー) 268  
要件 257  
リモート・サイト・リカバリー 255

データベース・リカバリー・ユーティリテ  
ィ (DFSURDB0) (続き)  
例 270  
DFSUSNAP DD ステートメント 265  
JCL の要件  
DD ステートメント 264  
EXEC ステートメント 263  
QUIESCE データベース状況 257  
RSR 環境 255  
SNAP 出力データ・セット属性 265  
データ・セット  
データベース・イメージ・コピー 2 ユ  
ーティリティ (DFSUDMT0)  
動的データ・セット名 200  
連結データ・セット  
連結 HALDB データ・セットの表  
示 143  
DFSURDB0 SNAP 出力データ・セッ  
ト 265  
HALDB  
連結データ・セットの表示 143  
特記事項  
商標 519, 521  
特記事項 519

## [ハ行]

パーティション・データベース  
情報画面  
区画選択ルーチン 120  
データベース名 120  
database organization 121  
number of data set groups 121  
recoverable? 121  
RSR global service group 120  
RSR トラッキング・タイプ 120  
share level 120  
バックアップ  
オンライン・データベース・イメー  
ジ・コピー・ユーティリティ  
(DFSUICP0) 205  
データベース・イメージ・コピー 2 ユ  
ーティリティ (DFSUDMT0) 177  
データベース・イメージ・コピー・ユ  
ーティリティ (DFSUDMP0) 163  
バッチ・バックアウト・ユーティリティ  
(DFSBB000)  
推奨事項 220  
制約事項 218  
説明 217  
前提条件 219  
入出力 220  
戻りコード 222  
ユーティリティ制御ステートメント  
概要 228  
ABEND ステートメント 229

バッチ・バックアウト・ユーティリティ  
(DFSBB00) (続き)  
ユーティリティ制御ステートメント  
(続き)  
ABENDMSG ステートメント 229  
ACTIVE ステートメント 229  
BYPASS LOGVER ステートメント  
230  
BYPASS SEQVER ステートメント  
230  
CHKPT ステートメント 231  
COLDSTART ステートメント 231  
NOREADBACK ステートメント  
232  
READBACK ステートメント 232  
要件 219  
リモート・サイト・リカバリー 233  
例 233  
JCL の要件  
DD ステートメント 225  
EXEC ステートメント 224  
QUIESCE データベース状況 218  
RSR 環境における 233  
バッファ  
高速 DEDB 直接再編成ユーティリ  
ティー (DBFUHDR0) 376  
BUFNO コマンド 376  
DBFUHDR0 (高速 DEDB 直接再編成  
ユーティリティ) 376  
バッファ・セット 376  
非同期先行読み取り 376  
フリー・スペース  
ILDS  
再作成中にフリー・スペースを含め  
る 297, 341  
プログラム分離トレース報告書ユーティ  
リティ (DFSPIR0)  
(Program-Isolation-Trace Report utility  
(DFSPIR0))  
ユーティリティ制御ステートメント  
505  
例 506  
プログラム分離トレース報告ユーティリ  
ティ (DFSPIR0)  
推奨事項 502  
制約事項 501  
説明 501  
前提条件 501  
入出力 502  
戻りコード 505  
要件 502  
JCL の指定および例 503  
JCL の要件  
DD ステートメント 504  
EXEC ステートメント 504  
JOB LIB DD ステートメント 503

並行コピー機能  
DBDS 選択ステートメント 186

## 【マ行】

モジュール ZAP 機能  
ユーティリティ制御機能  
(DFSUCF00) 492

## 【ヤ行】

ユーザー出口ルーチンの処理 490  
ユーティリティ  
オンライン・データベース・イメ  
ジ・コピー・ユーティリティ  
(DFSUICP0) 205  
高速 DEDB 直接再編成ユーティリ  
ティ (DBFUHDR0) 369  
データベース事前再編成ユーティリ  
ティ (DFSURPR0) 311  
データベース接頭部解決ユーティリ  
ティ (DFSURG10) 3  
データベース接頭部更新ユーティリ  
ティ (DFSURGP0) 15  
データベース調査ユーティリティ  
(DFSURPR0) 39  
データベース部分再編成ユーティリ  
ティ (DFSURPR0) および  
DFSURPR2) 419  
データベース変更累積ユーティリ  
ティ (DFSUCUM0) 235  
データベース・イメージ・コピー 2 ユ  
ーティリティ (DFSUDMT0) 177  
データベース・イメージ・コピー・ユ  
ーティリティ (DFSUDMP0) 163  
データベース・スキャン・ユーティ  
リティ (DFSURGS0) 29  
データベース・モニター報告書印刷ユ  
ーティリティ (DFSUTR30) 497  
データベース・リカバリー・ユーテ  
ィリティ (DFSURDB0) 255  
バッチ・バックアウト・ユーティリ  
ティ (DFSBB00) 217  
プログラム分離トレース報告ユーテ  
ィリティ (DFSPIR0) 501  
ユーティリティ制御機能  
(DFSUCF00) 437  
DBFDBMA0 147  
DBFUDA00 (DEDB 変更ユーティリ  
ティ) 55  
DEDB エリア・データ・セット作成ユ  
ーティリティ (DBFUMRI0) 283  
DEDB エリア・データ・セット比較ユ  
ーティリティ (DBFUMMH0) 275

ユーティリティ (続き)  
DEDB 順次従属削除ユーティリティ  
(DBFUMDL0) 75  
DEDB 順次従属スキャン・ユーティ  
リティ (DBFUMSC0) 85  
DEDB 初期設定ユーティリティ  
(DBFUMIN0) 65  
DEDB 変更ユーティリティ  
(DBFUDA00) 55  
DFSURPR1 419  
DFSURPR2 419  
DFSURPR3 39  
DFSUCF00 437  
DFSURG10 3  
DFSURGL0 331  
DFSURGP0 15  
DFSURGS0 29  
DFSURGU0 349  
DFSURPR0 311  
DFSURRL0 381  
DFSURUL0 391  
HD 再編成アンロード・ユーティリ  
ティ (DFSURGU0) 349  
HD 再編成再ロード・ユーティリ  
ティ (DFSURGL0) 331  
HISAM 再編成アンロード・ユーテ  
ィリティ (DFSURUL0) 391  
HISAM 再編成再ロード・ユーテ  
ィリティ (DFSURRL0) 381  
MSDB ダンプ・リカバリー・ユーテ  
ィリティ (DBFDBDR0) 299  
MSDB 保守ユーティリティ  
(DBFDBMA0) 147  
SB テスト・ユーティリティ  
(DFSDBHD0) 507  
ユーティリティ制御機能 (DFSUCF00)  
エラー・ポイント異常終了 491  
実行するデータベース ZAP 476  
終了/エラー処理 488  
初期ロード出口ルーチン 487  
初期ロード・アプリケーション・プロ  
グラムに関する考慮事項  
関連する DL/I 状況コード 487  
説明 486  
チェックポイント・モジュール  
(DFSUCP90), UCF の 488  
処理のタイプ  
再始動 489  
終了/エラー 488  
正常 485  
ユーザー出口ルーチンの処理 490  
推奨事項 441  
制約事項 437, 438  
説明 437  
前提条件 438  
チェックポイント/再始動機能 488

ユーティリティ制御機能 (DFSUCF00) (続き)  
データベース ZAP 機能 492  
データベース・ユーティリティの実行 485  
入出力 441  
保守援助機能 491  
モジュール ZAP 機能 492  
戻りコード 447  
ユーティリティ制御ステートメント  
イメージ・コピー (IM) 459  
概要 448  
初期ロード (IL) 457  
接頭部解決 (PR) 461  
接頭部更新 (PU) 462  
データベース ZAP (ZB) 478  
データベース・スキャン (SN) 468  
副次索引アンロード (RU) 465  
副次索引再ロード (RR) 464  
モジュール ZAP (ZM) 480  
ユーザー初期ロード (IL) 457  
FUNCTION=OP 449  
HD 再編成アンロード (DU) 453  
HD 再編成アンロードと再ロードの組み合わせ (DX) 455  
HD 再編成再ロード (DR) 451  
HISAM 再編成アンロード (SU) 473  
HISAM 再編成アンロードと再ロード組み合わせ (SX) 475  
HISAM 再編成再ロード (SR) 471  
ユーティリティ制御ステートメントで指定するキーワード  
最低限の条件 446  
例、JCL 483  
FUNCTION= キーワード  
制御ステートメントの要件 448  
DR (HD 再編成再ロード・ユーティリティを表す) 451  
DU (HD 再編成アンロード・ユーティリティを表す) 453  
IL (初期ロード・プログラムを表す) 457  
IM (イメージ・コピー・ユーティリティを表す) 459  
PR (接頭部解決ユーティリティを表す) 461  
PU (接頭部更新ユーティリティを表す) 462  
RR (副次索引再ロードを表す) 464  
RU (副次索引アンロードを表す) 465  
SN (データベース・スキャン・ユーティリティを表す) 468  
SR (HISAM 再編成再ロード・ユーティリティを表す) 470

ユーティリティ制御機能 (DFSUCF00) (続き)  
FUNCTION= キーワード (続き)  
SU (HISAM 再編成アンロード・ユーティリティを表す) 473  
SX (HISAM 再編成アンロードと再ロードを表す) 475  
ZB (データベース ZAP を表す) 478  
ZM (モジュール ZAP を表す) 480  
JCL の要件  
DD ステートメント 441  
EXEC ステートメント 441  
QUIESCE データベース状況 437  
WTOR (要応答オペレーター宛メッセージ) 機能 492  
ユーティリティ制御ステートメント  
オンライン・データベース・イメージ・コピー・ユーティリティ (DFSUICP0) 211  
データベース事前再編成ユーティリティ (DFSURPR0)  
概要 317  
データベース接頭部更新ユーティリティ (DFSURGP0)  
概要 21  
データベース調査ユーティリティ (DFSPRSUR) 43  
データベース部分再編成ユーティリティ (DFSPRCT1 および DFSPRCT2) 427  
データベース変更累積ユーティリティ (DFSUCUM0)  
概要 243  
データベース・イメージ・コピー 2 ユーティリティ (DFSUDMT0)  
概要 186  
データベース・イメージ・コピー・ユーティリティ (DFSUDMP0) 174  
データベース・スキャン・ユーティリティ (DFSURGS0)  
概要 35  
データベース・リカバリ・ユーティリティ (DFSURDB0)  
概要 267  
バッチ・バックアウト・ユーティリティ (DFSBO00)  
概要 228  
プログラム分離トレース報告ユーティリティ (DFSPIRP0) 505  
ユーティリティ制御機能 (DFSUCF00)  
概要 448  
DBFDBDR0 (MSDB ダンプ・リカバリ・ユーティリティ) 304

ユーティリティ制御ステートメント (続き)  
DBFDBMA0 (MSDB 保守ユーティリティ) 152  
DBFUCDB0 (MSDB - DEDB 間変換ユーティリティ) 411  
DBFUMIN0 (DEDB 初期設定ユーティリティ) 70  
DEDB 初期設定ユーティリティ (DBFUMIN0) 70  
DFSBO00 (バッチ・バックアウト・ユーティリティ)  
概要 228  
DFSMAID0 (HALDB マイグレーション・エイド・ユーティリティ) 325  
DFSPIRP0 (プログラム分離トレース報告ユーティリティ) 505  
DFSPRCT1 および DFSPRCT2 (データベース部分再編成ユーティリティ) 427  
DFSPREC0 (HALDB 索引/ILDS 再作成ユーティリティ) 295  
DFSPRSUR (データベース調査ユーティリティ) 43  
DFSBBHD0 (SB テスト・ユーティリティ)  
概要 514  
DFSUCF00 (ユーティリティ制御機能)  
概要 448  
DFSUDMP0 (データベース・イメージ・コピー・ユーティリティ) 174  
DFSUDMT0 (データベース・イメージ・コピー 2 ユーティリティ)  
概要 186  
DFSUICP0 (オンライン・データベース・イメージ・コピー・ユーティリティ) 211  
DFSUPNT0 (HALDB 区画データ・セット初期設定ユーティリティ) 104  
DFSURGL0 (HD 再編成再ロード・ユーティリティ) 339  
DFSURGP0 (データベース接頭部更新ユーティリティ)  
概要 21  
DFSURGS0 (データベース・スキャン・ユーティリティ)  
概要 35  
DFSURGU0 (HD 再編成アンロード・ユーティリティ) 361  
DFSURPR0 (データベース事前再編成ユーティリティ)  
概要 317  
DFSURRL0 (HISAM 再編成再ロード・ユーティリティ) 388



ユーティリティ制御ステートメント (続き)

- DFSURUL0 (HISAM 再編成アンロード・ユーティリティ)  
概要 401
  - HALDB 区画データ・セット初期設定  
ユーティリティ (DFSUPNT0) 104
  - HALDB 区画定義ユーティリティ  
(%DFSHALDB) 113
  - HALDB 索引/ILDS 再作成ユーティリティ (DFSPREC0) 295
  - HALDB マイグレーション・エイド・ユーティリティ (DFSMAID0) 325
  - HD 再編成アンロード・ユーティリティ (DFSURGU0) 361
  - HD 再編成再ロード・ユーティリティ (DFSURGL0) 339
  - HISAM 再編成アンロード・ユーティリティ (DFSURUL0)  
概要 401
  - HISAM 再編成再ロード・ユーティリティ (DFSURRL0) 388
  - MSDB ダンプ・リカバリー・ユーティリティ (DBFDBDR0) 304
  - MSDB 保守ユーティリティ (DBFDBMA0) 152
  - MSDB - DEDB 間変換ユーティリティ (DBFUCDB0) 411
  - SB テスト・ユーティリティ (DFSSBHD0)  
概要 514
  - %DFSHALDB (HALDB 区画定義ユーティリティ) 113
- 要応答オペレーター宛メッセージ (WTOR) 機能  
ユーティリティ制御機能 (DFSUCF00) 492

## [ラ行]

- リカバリー
  - MSDB ダンプ・リカバリー 304
- リモート・サイト・リカバリー (RSR)
  - グローバル・サービス・グループ
    - HALDB (High Availability Large Database) 120
  - シャドウイング
    - HALDB (High Availability Large Database) 120
- DBTRACK 120
- RCVTRACK 120

## A

- ABEND ステートメント
  - データベース・スキャン・ユーティリティ (DFSURGS0) 37
  - データベース・リカバリー・ユーティリティ (DFSURDB0) 267
  - バッチ・バックアウト・ユーティリティ (DFSBB000) 229
- ABEND 制御ステートメント
  - データベース接頭部更新ユーティリティ (DFSURGP0) 22
- ABEND パラメーター
  - データベース事前再編成ユーティリティ (DFSURPR0) 319
- HISAM 再編成アンロード・ユーティリティ (DFSURUL0) 403
- HISAM 再編成再ロード・ユーティリティ (DFSURRL0) 388
- ABENDMSG ステートメント
  - バッチ・バックアウト・ユーティリティ (DFSBB000) 229
- ABENDOFF パラメーター
  - データベース事前再編成ユーティリティ (DFSURPR0) 319
- HISAM 再編成アンロード・ユーティリティ (DFSURUL0) 403
- HISAM 再編成再ロード・ユーティリティ (DFSURRL0) 388
- ACTION ステートメント
  - MSDB 保守ユーティリティ (DBFDBMA0) 152
- ACTIVE ステートメント
  - バッチ・バックアウト・ユーティリティ (DFSBB000) 229

## B

- BYPASS LOGVER ステートメント
  - バッチ・バックアウト・ユーティリティ (DFSBB000) 230
- BYPASS SEQVER ステートメント
  - バッチ・バックアウト・ユーティリティ (DFSBB000) 230

## C

- CHANGE=ステートメント
  - HISAM 再編成アンロード・ユーティリティ (DFSURUL0) 402
- CHKPT ステートメント
  - データベース接頭部更新ユーティリティ (DFSURGP0) 21
  - データベース・スキャン・ユーティリティ (DFSURGS0) 36

CHKPT ステートメント (続き)

- バッチ・バックアウト・ユーティリティ (DFSBB000) 231
- CKPNT= キーワード  
制御ステートメント
  - UCF FUNCTION=DR 452
  - UCF FUNCTION=DU 454
  - UCF FUNCTION=DX 456
  - UCF FUNCTION=IL 458
  - UCF FUNCTION=IM 460
  - UCF FUNCTION=OP 450
  - UCF FUNCTION=PU 463
  - UCF FUNCTION=RR 465
  - UCF FUNCTION=RU 468
  - UCF FUNCTION=SN 470
  - UCF FUNCTION=SR 472
  - UCF FUNCTION=SU 474
  - UCF FUNCTION= SX 477
- COLDSTART ステートメント
  - バッチ・バックアウト・ユーティリティ (DFSBB000) 231
- COPY= キーワード  
制御ステートメント
  - UCF FUNCTION=RU 468
  - UCF FUNCTION=SU 475
  - UCF FUNCTION= SX 478
- CSECT= キーワード
  - UCF FUNCTION=ZM 制御ステートメント 481

## D

- DATA= キーワード
  - MSDB 保守ユーティリティの MSDBINIT ステートメント 155
- DATE= キーワード
  - SB テスト・ユーティリティの SELECT ステートメント 515
- DB0 ステートメント
  - データベース変更累積ユーティリティ (DFSUCUM0) 246, 247
- DB1 ステートメント
  - データベース変更累積ユーティリティ (DFSUCUM0) 248
- DBDDDS= キーワード  
制御ステートメント
  - UCF FUNCTION=DR 452
  - UCF FUNCTION=DU 454
  - UCF FUNCTION=DX 456
  - UCF FUNCTION=IL 458
  - UCF FUNCTION=IM 460
  - UCF FUNCTION=PU 463
  - UCF FUNCTION=RU 467
  - UCF FUNCTION=SN 469
  - UCF FUNCTION=SR 472
  - UCF FUNCTION=SU 474

- DBDDDS= キーワード (続き)  
制御ステートメント (続き)  
UCF FUNCTION= SX 476  
UCF FUNCTION= ZB 479
- DBDLIB  
メンバー 119  
HALDB (高可用性ラージ・データベース)  
143
- DBDS 選択ステートメント  
高速複製機能 189  
並行コピー機能 186
- DBFDBDR0 (MSDB ダンプ・リカバリー・ユーティリティ)  
推奨事項 301  
制約事項 300  
説明 299  
前提条件 300  
入出力 301  
戻りコード 303  
ユーティリティ制御ステートメント  
304  
要件 301  
例 305  
JCL 指定 302  
JCL の要件  
DD ステートメント 302  
EXEC ステートメント 302  
RECON 状況 300
- DBFDBMA0 (MSDB 保守ユーティリティ)  
機能  
置き換え 158  
削除 158  
挿入 158  
変更 158  
推奨事項 149  
制約事項 148  
説明 147  
前提条件 149  
入出力 149  
戻りコード 151  
ユーティリティ制御ステートメント  
152  
ACTION ステートメント 152  
RUN ステートメント 152  
要件 149  
例 156  
JCL の要件  
DD ステートメント 151  
EXEC ステートメント 151  
QUIESCE データベース状況 148
- DBFUCDB0 (MSDB - DEDB 間変換ユーティリティ)  
推奨事項 408  
制約事項 407  
説明 407
- DBFUCDB0 (MSDB - DEDB 間変換ユーティリティ) (続き)  
前提条件 407  
入出力 408  
フォールバック 415  
ユーティリティ制御ステートメント  
411  
ユーティリティの実行 415  
要件 408  
要約報告書 408  
例 411  
DD ステートメント 409  
DEDB の MSDB への変換 415  
EXEC ステートメント 408  
JCL 指定 408  
QUIESCE データベース状況 407
- DBFUDA00 (DEDB 変更ユーティリティ)  
推奨事項 56  
制約事項 55  
説明 55  
前提条件 56  
入出力 56  
戻りコード 57  
ユーティリティ制御ステートメント  
58  
要件 56  
DD ステートメント 57  
EXEC ステートメント 57  
JCL 指定 57  
JCL の例 62
- DBFUHDR0 (高速 DEDB 直接再編成ユーティリティ)  
エラー処理 379  
概要 369  
収集された統計 370  
推奨事項 370  
制御ステートメント 373  
制約事項 369  
セグメント・シャント 377  
前提条件 370  
入出力 370  
バッファの使用法 376  
戻りコード 373  
ユーティリティの実行 376  
要件 370  
リカバリーおよび再始動 379  
BUFNO コマンド 376  
DD ステートメント 372  
EXEC ステートメント 372  
JCL 指定 371  
JCL の例 372  
QUIESCE データベース状況 369
- DBFUMDL0 (DEDB 順次従属削除ユーティリティ)  
推奨事項 76  
制御ステートメント 77  
制約事項 75  
説明 75  
前提条件 75  
入出力 76  
戻りコード 76  
要件 75  
JCL の指定および例 76  
JCL の要件  
DD ステートメント 76  
EXEC ステートメント 76  
JCL の例 81  
QUIESCE データベース状況 75
- DBFUMIN0 (DEDB 初期設定ユーティリティ)  
推奨事項 67  
制約事項 65  
説明 65  
前提条件 66  
入出力 67  
戻りコード 67  
ユーティリティ制御ステートメント  
70  
要件 66  
例 72  
JCL の要件  
DD ステートメント 68  
EXEC ステートメント 68  
QUIESCE データベース状況 65
- DBFUMMH0 (DEDB エリア・データ・セット比較ユーティリティ)  
推奨事項 276  
制御ステートメント 278  
制約事項 275  
説明 275  
前提条件 276  
入出力 276  
戻りコード 278  
要件 276  
例 280  
JCL 指定 277  
JCL の要件  
DD ステートメント 277  
EXEC ステートメント 277  
QUIESCE データベース状況 275
- DBFUMRI0 (DEDB エリア・データ・セット作成ユーティリティ)  
推奨事項 285  
制御ステートメント 286  
制約事項 284  
説明 283  
前提条件 284  
入出力 285

- DBFUMRI0 (DEDB エリア・データ・セット作成ユーティリティ) (続き)  
 戻りコード 285  
 要件 284  
 例 288  
 JCL 指定 286  
 JCL の要件  
 DD ステートメント 286  
 EXEC ステートメント 286  
 QUIESCE データベース状況 284
- DBFUMSC0 (DEDB 順次従属スキャン・ユーティリティ)  
 推奨事項 86  
 スキャンの範囲 95  
 制御ステートメント 88  
 制約事項 86  
 説明 85  
 前提条件 86  
 入出力 86  
 戻りコード 87  
 要件 86  
 JCL の指定および例 87  
 JCL の要件  
 DD ステートメント 87  
 EXEC ステートメント 87  
 JCL の例 94  
 QUIESCE データベース状況 86
- DBIL= ステートメント  
 データベース事前再編成ユーティリティ (DFSURPR0) 317
- DBIO ステートメント  
 SB テスト・ユーティリティ (DFSSBHD0) 514
- DBNAME= オペランド  
 制御ステートメント  
 UCF FUNCTION=DR 451  
 UCF FUNCTION=DU 453  
 UCF FUNCTION=DX 455  
 UCF FUNCTION=IL 457  
 UCF FUNCTION=IM 459  
 UCF FUNCTION=PR 462  
 UCF FUNCTION=PU 463  
 UCF FUNCTION=RR 464  
 UCF FUNCTION=RU 466  
 UCF FUNCTION=SN 469  
 UCF FUNCTION=SR 471  
 UCF FUNCTION=SU 474  
 UCF FUNCTION=SX 476  
 UCF FUNCTION=ZB 479  
 データベース調査ユーティリティ (DFSPRSUR) 44  
 部分的 DB 再編成  
 ステップ 1 (DFSURPR0) 427  
 ステップ 2 (DFSURPR0) 429
- DBN= ステートメント  
 MSDB ダンプ・リカバリー・ユーティリティ 304  
 MSDB 保守ユーティリティ  
 ACTION ステートメント 153  
 MSDB 保守ユーティリティの変更ステートメント 155
- DBR=ステートメント  
 データベース事前再編成ユーティリティ (DFSURPR0) 317
- DBS= ステートメント  
 データベース・スキャン・ユーティリティ (DFSURGS0) 35
- DD ステートメント  
 オンライン・データベース・イメージ・コピー・ユーティリティ (DFSUICP0) 209  
 データベース事前再編成ユーティリティ (DFSURPR0) 315  
 データベース接頭部解決ユーティリティ (DFSURG10) 10  
 データベース接頭部更新ユーティリティ (DFSURGP0) 18  
 データベース調査ユーティリティ (DFSPRSUR) 41  
 データベース部分再編成ユーティリティ (DFSURPR0) 422  
 データベース部分再編成ユーティリティ (DFSURPR0) 423  
 データベース変更累積ユーティリティ (DFSUCUM0) 240  
 データベース・イメージ・コピー 2 ユーティリティ (DFSUDMT0) 181  
 データベース・イメージ・コピー・ユーティリティ (DFSUDMP0) 171  
 データベース・スキャン・ユーティリティ (DFSURGS0) 32  
 データベース・モニター報告書印刷ユーティリティ (DFSUTR30) 498  
 データベース・リカバリー・ユーティリティ (DFSURDB0) 264  
 バッチ・バックアウト・ユーティリティ (DFSBB000) 225  
 プログラム分離トレース報告ユーティリティ (DESPIRP0) 504  
 ユーティリティ制御機能 (DFSUCF00) 441
- DEDB エリア・データ・セット作成ユーティリティ (DBFUMRI0) 286  
 DEDB エリア・データ・セット比較ユーティリティ (DBFUMMH0) 277  
 DEDB 順次従属削除ユーティリティ (DBFUMDL0) 76  
 DEDB 順次従属スキャン・ユーティリティ (DBFUMSC0) 87
- DD ステートメント (続き)  
 DEDB 初期設定ユーティリティ (DBFUMIN0) 68  
 EXEC ステートメント 87  
 HD 再編成アンロード・ユーティリティ (DFSURGU0) 357  
 HISAM 再編成アンロード・ユーティリティ (DFSURUL0) 398  
 HISAM 再編成再ロード・ユーティリティ (DFSURRL0) 386  
 MSDB ダンプ・リカバリー・ユーティリティ (DBFDBDR0) 302  
 MSDB 保守ユーティリティ (DBFDBMA0) 151  
 SB テスト・ユーティリティ (DFSSBHD0) 512
- DDNAME  
 HALDB (High Availability Large Database) 118
- DEDB  
 DEDB の MSDB への変換 415  
 MSDB の DEDB への変換 415
- DEDB エリア・データ・セット作成ユーティリティ (DBFUMRI0)  
 推奨事項 285  
 制御ステートメント 286  
 制約事項 284  
 説明 283  
 前提条件 284  
 入出力 285  
 戻りコード 285  
 要件 284  
 例 288  
 JCL 指定 286  
 JCL の要件  
 DD ステートメント 286  
 EXEC ステートメント 286  
 QUIESCE データベース状況 284
- DEDB エリア・データ・セット比較ユーティリティ (DBFUMMH0)  
 推奨事項 276  
 制御ステートメント 278  
 制約事項 275  
 説明 275  
 前提条件 276  
 入出力 276  
 戻りコード 278  
 要件 276  
 例 280  
 JCL 指定 277  
 JCL の要件  
 DD ステートメント 277  
 EXEC ステートメント 277  
 QUIESCE データベース状況 275

DEDB 順次従属削除ユーティリティ  
(DBFUMDL0)

処理の範囲 82  
推奨事項 76  
制御ステートメント 77  
制約事項 75  
説明 75  
前提条件 75  
入出力 76  
戻りコード 76  
要件 75  
JCL の指定および例 76  
JCL の要件

DD ステートメント 76  
EXEC ステートメント 76  
JCL の例 81  
QUIESCE データベース状況 75

DEDB 順次従属スキャン・ユーティリ  
ティー (DBFUMSC0)

推奨事項 86  
スキャンの範囲 95  
制御ステートメント 88  
制約事項 86  
説明 85  
前提条件 86  
入出力 86  
戻りコード 87  
要件 86  
EXEC ステートメント 87  
JCL の指定および例 87  
JCL の要件

DD ステートメント 87  
EXEC ステートメント 87  
JCL の例 94  
QUIESCE データベース状況 86

DEDB 初期設定ユーティリティ  
(DBFUMIN0)

推奨事項 67  
制約事項 65  
説明 65  
前提条件 66  
入出力 67  
戻りコード 67  
ユーティリティ制御ステートメント  
70  
要件 66  
例 72  
JCL の要件

DD ステートメント 68  
EXEC ステートメント 68  
QUIESCE データベース状況 65

DEDB 変更ユーティリティ  
(DBFUDA00)

推奨事項 56  
制約事項 55  
説明 55

DEDB 変更ユーティリティ  
(DBFUDA00) (続き)

前提条件 56  
入出力 56  
戻りコード 57  
ユーティリティ制御ステートメント  
58  
要件 56  
DD ステートメント 57  
EXEC ステートメント 57  
JCL 指定 57  
JCL の例 62

DFPXPMB 143, 144

DFSBB00 (バッチ・バックアウト・ユ  
ーティリティ)

推奨事項 220  
制約事項 218  
説明 217  
前提条件 219  
入出力 220  
戻りコード 222  
ユーティリティ制御ステートメント  
概要 228  
ABEND ステートメント 229  
ABENDMSG ステートメント 229  
ACTIVE ステートメント 229  
BYPASS LOGVER ステートメント  
230  
BYPASS SEQVER ステートメント  
230  
CHKPT ステートメント 231  
COLDSTART ステートメント 231  
NOREADBACK ステートメント  
232  
READBACK ステートメント 232  
要件 219  
リモート・サイト・リカバリー 233

JCL の要件

DD ステートメント 225  
EXEC ステートメント 224  
QUIESCE データベース状況 218  
RSR 環境における 233

DFSMAID0 (HALDB マイグレーショ  
ン・エイド・ユーティリティ)

概要 321  
サンプリング 327, 328  
出力メッセージと統計 321  
推奨事項 322  
制約事項 322  
前提条件 322  
データ・スペース 327  
データ・スペース、サイズの計算 328  
データ・スペースおよび分析の種類  
327  
入出力 322  
戻りコード 324

DFSMAID0 (HALDB マイグレーショ  
ン・エイド・ユーティリティ) (続き)  
ユーティリティ制御ステートメント  
325

要件 322  
DD ステートメント 324  
EXEC ステートメント 323  
JCL 指定 323  
JCL の例 326  
QUIESCE データベース状況 322

DFSMSdss

SET PATCH コマンド 203

DFSPIR0 (プログラム分離トレース報告  
ユーティリティ)

推奨事項 502  
制約事項 501  
説明 501  
前提条件 501  
入出力 502  
戻りコード 505  
ユーティリティ制御ステートメント  
505  
要件 502  
例 506  
JCL の指定および例 503  
JCL の要件

DD ステートメント 504  
EXEC ステートメント 504  
JOBLIB DD ステートメント 503

DFSPRCT1

データベース部分再編成ユーティリ  
ティー 420

DFSPRCT2 (データベース部分再編成  
ユーティリティ)

ユーティリティ制御機能の使用が  
制限された 437

DFSPRCT1 および DFSPRCT2 (データ  
ベース部分再編成ユーティリティ)

ユーティリティ制御ステートメント  
427

DFSPRCT1 (データベース部分再編成ユ  
ーティリティ)

推奨事項 420  
制約事項 420  
説明 419  
前提条件 420  
要件 420  
例 429

QUIESCE データベース状況 420

DFSPRCT2 (アンロード/再ロード/ポ  
インター解決ステップ 2)

データベース部分再編成ユーティリ  
ティー 421

DFSPRCT2 (データベース部分再編成ユ  
ーティリティ)

推奨事項 420

- DFSPRCT2 (データベース部分再編成ユーティリティ) (続き)  
 制約事項 420  
 説明 419  
 前提条件 420  
 要件 420  
 例 429  
 QUIESCE データベース状況 420
- DFSPREC0 (HALDB 索引/ILDS 再作成ユーティリティ)  
 概要 291  
 制約事項 292  
 入出力 293  
 フリー・スペースのオプション 297  
 戻りコード 294  
 ユーティリティ制御ステートメント 295  
 例 296  
 JCL の要件  
 DD ステートメント 293  
 EXEC ステートメント 293  
 QUIESCE データベース状況 292  
 VSAM ロード・モード 297
- DFSPRSUR (データベース調査ユーティリティ)  
 出力 40  
 推奨事項 39  
 制約事項 39  
 説明 39  
 前提条件 39  
 入力 40  
 戻りコード 40  
 ユーティリティ制御機能の使用が制限された 437  
 ユーティリティ制御ステートメント 43  
 要件 39  
 例 45  
 JCL の要件  
 DD ステートメント 41  
 EXEC ステートメント 41  
 QUIESCE データベース状況 39
- DFSSBHD0 (SB テスト・ユーティリティ)  
 イメージ・キャプチャー・ログ・ロード 507  
 出力 511  
 推奨事項 510  
 制約事項 510  
 説明 507  
 前提条件 510  
 入力 511  
 戻りコード 514  
 ユーティリティ制御ステートメント  
 概要 514  
 DBIO 514
- DFSSBHD0 (SB テスト・ユーティリティ) (続き)  
 ユーティリティ制御ステートメント (続き)  
 SELECT 515  
 要件 510  
 JCL の要件  
 DD ステートメント 512  
 EXEC ステートメント 511  
 JCL の例 516
- DFSUCF00 (ユーティリティ制御機能)  
 エラー・ポイント異常終了 491  
 終了/エラー処理 488  
 初期ロード出口ルーチン 487  
 初期ロード・アプリケーション・プログラムに関する考慮事項  
 説明 486  
 処理のタイプ  
 再始動 489  
 正常 485  
 ユーザー出口ルーチンの処理 490  
 推奨事項 441  
 制約事項 437, 438  
 説明 437  
 前提条件 438  
 チェックポイント/再始動機能 488  
 データベース ZAP 機能 492  
 データベース・ユーティリティの実行 485  
 入出力 441  
 保守援助機能 491  
 モジュール ZAP 機能 492  
 ユーティリティ制御ステートメント  
 概要 448  
 FUNCTION=OP 449  
 FUNCTION= キーワード  
 制御ステートメントの要件 448  
 JCL の要件  
 DD ステートメント 441  
 EXEC ステートメント 441  
 QUIESCE データベース状況 437  
 WTOR (要応答オペレーター宛メッセージ) 機能 492
- DFSUDMP0 (データベース・イメージ・コピー・ユーティリティ)  
 制約事項 164  
 説明 163  
 前提条件 165  
 入出力 167  
 戻りコード 168, 180  
 ユーティリティ制御ステートメント 174  
 要件 166  
 リモート・サイト・リカバリー 163  
 例 175  
 JCL 指定 168
- DFSUDMP0 (データベース・イメージ・コピー・ユーティリティ) (続き)  
 JCL の要件  
 DD ステートメント 171  
 EXEC ステートメント 168  
 QUIESCE データベース状況 164
- DFSUDMT0 (データベース・イメージ・コピー 2 ユーティリティ)  
 グループ名 199  
 高速複製機能 197  
 コピー完了通知 200  
 推奨事項 179  
 制約事項 177  
 説明 177  
 前提条件 178  
 データ・セット、複数入力 198  
 同一データ・セット・オプション 199  
 動的データ・セット名 200  
 入出力 180  
 並行コピー機能 196  
 ユーティリティ制御ステートメント  
 概要 186  
 グループ名ステートメント 191  
 高速複製 DBDS 選択ステートメント 189  
 並行コピー DBDS 選択ステートメント 186  
 HLQ 仕様ステートメント 193  
 SET PATCH 制御ステートメント 195  
 要件 178  
 例 185  
 DFSMSdss SET PATCH コマンド 203  
 JCL の要件  
 DD ステートメント 181  
 EXEC ステートメント 181  
 QUIESCE データベース状況 177
- DFSUICP0 (オンライン・データベース・イメージ・コピー・ユーティリティ)  
 出力 207  
 推奨事項 207  
 制約事項 206  
 説明 205  
 前提条件 206  
 戻りコード 208  
 ユーティリティ制御機能の使用が制限された 437  
 ユーティリティ制御ステートメント 211  
 要件 206  
 リカバリーおよび再始動 213  
 例 212  
 JCL の要件  
 DD ステートメント 209  
 EXEC ステートメント 208

DFSUICP0 (オンライン・データベース・イメージ・コピー・ユーティリティ) (続き)	DFSURGP0 (データベース接頭部更新ユーティリティ)	DFSURGU0 (HD 再編成アンロード・ユーティリティ) (続き)
QUIESCE データベース状況 206	概要 15	DD ステートメント 357
DFSUPNT0 (HALDB 区画データ・セット初期設定ユーティリティ)	再始動 27	EXEC ステートメント 357
概要 99	推奨事項 17	JCL 指定 357
推奨事項 100	制約事項 16	JCL の要件
制約事項 100	前提条件 17	DD ステートメント 357
前提条件 100	入出力 17	JCL の例 362
入出力 101	ユーティリティ制御ステートメント	QUIESCE データベース状況 352
戻りコード 101	概要 21	RECON 状況 352
ユーティリティ制御ステートメント 104	ABEND 22	DFSURPR0 (データベース事前再編成ユーティリティ)
要件 100	CHKPT 21	出力メッセージ 313
例 104	RSTRT 21	制約事項 312
DD ステートメント 102	SNAP 21	説明 311
EXEC ステートメント 101	要件 17	戻りコード 316
JCL の指定および例 101	JCL の要件	ユーティリティ制御ステートメント
QUIESCE データベース状況 100	DD ステートメント 18	概要 317
DFSURG10 (データベース接頭部解決ユーティリティ)	EXEC ステートメント 18	DBIL= 317
概要 3	QUIESCE データベース状況 16	DBR= 317
推奨事項 6	DFSURGS0 (データベース・スキャン・ユーティリティ)	OPTIONS= 318
制約事項 5	再始動 38	例 319
前提条件 6	出力メッセージ 31	JCL の要件
入出力 6	推奨事項 31	DD ステートメント 315
戻りコード 7	スキャン・オプション	EXEC ステートメント 315
要件 6	SEG 35	QUIESCE データベース状況 312
EXEC ステートメントの PARM フィールド・オプション 8	SEQ 35	DFSURRL0 (HISAM 再編成再ロード・ユーティリティ)
EXEC ステートメントの例 8	制約事項 31	出力メッセージと統計 383
JCL の要件	説明 29	推奨事項 382
DD ステートメント 10	前提条件 31	制約事項 382
EXEC ステートメント 8	入出力 31	説明 381
JCL の例 13	戻りコード 31	前提条件 382
QUIESCE データベース状況 5	ユーティリティ制御ステートメント	入出力 383
z/OS ソート/マージ・プログラム 8	概要 35	戻りコード 387
DFSURGL0 (HD 再編成再ロード・ユーティリティ)	ABEND 37	ユーティリティ制御ステートメント 388
出力メッセージと統計 335	CHKPT 36	要件 382
推奨事項 334	DBS 35	JCL 指定 385
制約事項 333	RSTRT 37	JCL の要件
説明 331	要件 31	DD ステートメント 386
前提条件 334	JCL の指定および例 32	EXEC ステートメント 386
統計 342	JCL の要件	JCL の例 389
フリー・スペースのオプション 341	DD ステートメント 32	QUIESCE データベース状況 382
報告書 342	EXEC ステートメント 32	DFSURUL0 (HISAM 再編成アンロード・ユーティリティ)
戻りコード 339	JCL の例 38	出力メッセージと統計 394
ユーティリティ制御ステートメント 339	QUIESCE データベース状況 31	推奨事項 394
要件 334	DFSURGU0 (HD 再編成アンロード・ユーティリティ)	制約事項 392
例 344	推奨事項 355	説明 391
QUIESCE データベース状況 333	制約事項 352	前提条件 393
VSAM ロード・モード 341	説明 349	入出力 394
	前提条件 353	戻りコード 400
	入出力 355	ユーティリティ制御ステートメント
	戻りコード 360	概要 401
	ユーティリティ制御ステートメント 361	要件 393
	要件 353	

DFSURUL0 (HISAM 再編成アンロード・ユーティリティ) (続き)  
例 404  
CHANGE=ステートメント 402  
JCL 指定 398  
JCL の要件  
DD ステートメント 398  
EXEC ステートメント 398  
OPTIONS=ステートメント 403  
QUIESCE データベース状況 392  
DFSUSNAP DD ステートメント  
属性 265  
DFSUTR30 (データベース・モニター報告書印刷ユーティリティ)  
推奨事項 498  
制約事項 497  
説明 497  
前提条件 497  
入出力 498  
分析制御データ・セット 499  
戻りコード 499  
要件 498  
例 499  
JCL 指定 498  
JCL の要件  
DD ステートメント 498  
EXEC ステートメント 498  
DSDDNAM= データ・セット  
UCF FUNCTION=IL 制御ステートメント 458  
DSPXPDA 140  
DSPXPEA 141  
出力データ・セット名 141  
データベース名 141  
DSPXPPIA 142  
データベース名 142  
Input data set name 142  
Input member name 142  
Processing option 142  
DSPXPKE パネル 116  
DSPXPLA 132  
アクション 133  
データ・セット名接頭部 133  
ID 133  
name 133  
DSPXPLB 134  
DSPXPOA 140  
DSPXRUN コマンド 113

## E

ESCISZ= キーワード  
HISAM 再編成アンロード・ユーティリティ (DFSURUL0) 402

ESREC= キーワード  
HISAM 再編成アンロード・ユーティリティ (DFSURUL0) 403  
EXEC ステートメント 87  
オンライン・データベース・イメージ・コピー・ユーティリティ (DFSUICP0) 208  
高速 DEDB 直接再編成ユーティリティ (DBFUHDR0) 372  
データベース事前再編成ユーティリティ (DFSURPR0) 314  
データベース接頭部解決ユーティリティ (DFSURG10) 8  
データベース接頭部更新ユーティリティ (DFSURGP0) 18  
データベース調査ユーティリティ (DFSPRSUR) 41  
データベース部分再編成ユーティリティ (DFSPRCT1) 422  
データベース部分再編成ユーティリティ (DFSPRCT2) 423  
データベース変更累積ユーティリティ (DFSUCUM0) 238  
データベース・イメージ・コピー 2 ユーティリティ (DFSUDMP0) 181  
データベース・イメージ・コピー・ユーティリティ (DFSUDMP0) 168  
データベース・スキャン・ユーティリティ (DFSURGS0) 32  
データベース・モニター報告書印刷ユーティリティ (DFSUTR30) 498  
データベース・リカバリー・ユーティリティ (DFSURDB0) 263  
バッチ・バックアウト・ユーティリティ (DFSBO00) 224  
プログラム分離トレース報告ユーティリティ (DFSPIRP0) 504  
ユーティリティ制御機能 (DFSUCF00) 441  
DBFUCDB0 (MSDB - DEDB 間変換ユーティリティ) 408  
DBFUHDR0 (高速 DEDB 直接再編成ユーティリティ) 372  
DEDB エリア・データ・セット作成ユーティリティ (DBFUMRI0) 286  
DEDB エリア・データ・セット比較ユーティリティ (DBFUMMH0) 277  
DEDB 順次従属削除ユーティリティ (DBFUMDL0) 76  
DEDB 初期設定ユーティリティ (DBFUMIN0) 68  
HD 再編成アンロード・ユーティリティ (DFSURGU0) 357  
HISAM 再編成アンロード・ユーティリティ (DFSURUL0) 398

EXEC ステートメント (続き)  
HISAM 再編成再ロード・ユーティリティ (DFSURRL0) 386  
MSDB ダンプ・リカバリー・ユーティリティ (DBFDBDR0) 302  
MSDB 保守ユーティリティ (DBFDBMA0) 151  
MSDB - DEDB 間変換ユーティリティ (DBFUCDB0) 408  
SB テスト・ユーティリティ (DFSSBHD0) 511  
EXEC= ステートメント  
制御ステートメント  
UCF FUNCTION=DR 452  
UCF FUNCTION=DU 454  
UCF FUNCTION=DX 456  
UCF FUNCTION=IL 458  
UCF FUNCTION=IM 460  
UCF FUNCTION=PR 461  
UCF FUNCTION=PU 463  
UCF FUNCTION=RR 465  
UCF FUNCTION=RU 467  
UCF FUNCTION=SN 470  
UCF FUNCTION=SR 472  
UCF FUNCTION=SU 474  
UCF FUNCTION= SX 477  
UCF FUNCTION=ZB 480  
EXITRLD= キーワード  
制御ステートメント  
UCF FUNCTION=DX 456  
UCF FUNCTION= SX 478  
EXITRTN= キーワード  
制御ステートメント  
UCF FUNCTION=DR 452  
UCF FUNCTION=DU 454  
UCF FUNCTION=IL 458  
UCF FUNCTION=IM 460  
UCF FUNCTION=PR 462  
UCF FUNCTION=PU 463  
UCF FUNCTION=RR 465  
UCF FUNCTION=RU 468  
UCF FUNCTION=SN 470  
UCF FUNCTION=SR 473  
UCF FUNCTION=SU 475  
UCF FUNCTION= SX 477  
UCF FUNCTION=ZB 480  
UCF FUNCTION=ZM 483  
EXTRACT= キーワード  
UCF FUNCTION=RU 制御ステートメント 468  
**F**  
FIELD= キーワード  
MSDB 保守ユーティリティの  
MSDBINIT ステートメント 155

「File」アクション・バー 118  
 FIXED= キーワード  
 MSDB 保守ユーティリティの  
 ACTION ステートメント 154  
 FROMAREA= オペランド  
 部分的 DB 再編成  
 ステップ 1 (DFSPRCT1) 428  
 DB データベース調査ユーティリ  
 ティー 44  
 FUNCTION= キーワード  
 ユーティリティ制御機能 (UCF)  
 DR (HD RR 再ロード・ユーティ  
 リティをを表す) 451  
 DU (HD 再編成アンロード・ユー  
 ティリティをを表す) 452  
 DX (HD RR アンロードと再ロー  
 ド・ユーティリティ (組み合わ  
 せ) を表す) 454  
 IL (初期ロード・プログラムを表  
 す) 457  
 IM (イメージ・コピー・ユーティ  
 リティをを表す) 459  
 PR (接頭部解決ユーティリティ  
 を表す) 460  
 PU (接頭部更新ユーティリティ  
 を表す) 462  
 RR (副次索引再ロードを表す) 464  
 RU (副次索引アンロードを表す)  
 465  
 SN (データベース・スキャン・ユ  
 ーティリティをを表す) 468  
 SR (HISAM 再編成再ロード・ユー  
 ティリティをを表す) 470  
 SU (HISAM 再編成アンロード・  
 ユーティリティをを表す) 473  
 SX (HISAM 再編成アンロードと再  
 ロード・ユーティリティ (組み  
 合わせ) を表す) 475  
 ZB (データベース ZAP を表す)  
 478  
 ZM (モジュール ZAP を表す) 480  
 FUNCTION=OP ステートメント  
 ユーティリティ制御機能  
 (DFSUCF00) 449

## H

HALDB (High Availability Large  
 Database)  
 行コマンド 136  
 区画  
 最大数 119  
 情報  
 区画定義ユーティリティを使用し  
 たエクスポート 141

HALDB (High Availability Large  
 Database) (続き)  
 パフォーマンス  
 ILDS 342  
 DFSMAID0 (HALDB マイグレーショ  
 ン・エイド・ユーティリティ) 321  
 DSPXPEA  
 DSPXPEA を参照 141  
 「File」アクション・バー 119  
 HALDB マイグレーション・エイド・  
 ユーティリティ (DFSMAID0) 321  
 HALDB 区画データ・セット初期設定ユ  
 ーティリティ (DFSUPNT0)  
 概要 99  
 推奨事項 100  
 制約事項 100  
 前提条件 100  
 入出力 101  
 戻りコード 101  
 ユーティリティ制御ステートメント  
 104  
 要件 100  
 例 104  
 DD ステートメント 102  
 EXEC ステートメント 101  
 JCL の指定および例 101  
 QUIESCE データベース状況 100  
 HALDB 区画定義ユーティリティ  
 (%DFSHALDB)  
 インポート 116  
 エクスポート 116  
 概要 107  
 使用 116  
 推奨事項 108  
 制約事項 108  
 前提条件 108  
 対話式アプリケーション 116  
 入出力 108  
 バッチ JCL の要件  
 DD ステートメント 112  
 フォアグラウンド JCL の要件  
 DD ステートメント 110  
 メインスクリーン 117  
 メインパネル 117  
 戻りコード 113  
 ユーティリティ制御ステートメント  
 113  
 要件 108  
 例 113, 115  
 ISPF パネル  
 HALDB 区画定義ユーティリ  
 ティ (%DFSHALDB) 117  
 Partitioned Databases (パーティショ  
 ン・データベース) パネル 117

HALDB (高可用性ラージ・データベース)  
 インターフェース  
 HALDB 区画定義ユーティリ  
 ティ (%DFSHALDB) 116  
 区画  
 区画定義ユーティリティを使用し  
 たオープン 119, 136  
 区画定義ユーティリティを使用し  
 たコピー 136  
 区画定義ユーティリティを使用し  
 た削除 136  
 区画定義ユーティリティを使用し  
 た作成 119  
 区画定義ユーティリティを使用し  
 た手動定義 126  
 区画定義ユーティリティを使用し  
 た情報の印刷 136  
 区画定義ユーティリティを使用し  
 た定義 119  
 区画定義ユーティリティを使用し  
 た変更 119  
 区画定義ユーティリティ 126  
 アクセス 116  
 ハイ・キー値 126  
 パネル 116  
 フィールドの変更 126  
 HALDB データ・セット連結の表  
 示 143  
 RECON に対する影響 107  
 区画定義ユーティリティを使用した  
 自動区画定義 125  
 区画定義ユーティリティを使用した  
 手動区画定義 126  
 区画定義ユーティリティを使用した  
 データベース情報のインポート 142  
 区画定義ユーティリティを使用した  
 データベースの削除パネル 140  
 区画定義ユーティリティを使用した  
 データ・セット・グループの定義  
 130  
 区画変更画面 128  
 区画 ID フィールド 128  
 区画選択ストリング 129  
 F11 130  
 F5 129  
 F6 130  
 Partition high key 128  
 構成  
 リスト 144  
 情報  
 区画定義ユーティリティを使用し  
 たインポート 119, 142  
 区画定義ユーティリティを使用し  
 たエクスポート 119  
 区画定義ユーティリティを使用し  
 たオープン 140



- HALDB (高可用性ラージ・データベース) (続き)
  - 情報 (続き)
    - 区画定義ユーティリティを使用し  
た削除 119, 140
    - 区画定義ユーティリティを使用し  
た表示 119
    - 区画定義ユーティリティを使用し  
た変更 119
- データベース区画のリスト 132
  - アクション 135
  - 最大ブロック 135
  - 表示 132
  - モジュール 135
  - Anchor 135
  - Bytes 135
  - FBFF 135
  - 「File」アクション・パー選択 133
  - FSFF 135
  - name 135
- バッチ・インポート 112
- パフォーマンス
  - ILDS 297
- マスター
  - 値 119
- DBDLIB 143
- DSPXPDA 140
- DSPXPGA 130
- DSPXPGB 131
- DSPXPGD 131
- DSPXPJA
  - DSPXPJA を参照 142
- DSPXPKE パネル 116
- DSPXPLA
  - DSPXPLA を参照 132
- DSPXPLB 134
- DSPXPOA 140
- DSPXRUN コマンド 113
  - 「File」アクション・パー  
アクション 119
- HALDB 区画定義ユーティリティ  
(%DFSHALDB)
  - 使用 116
  - メインスクリーン 117
  - メインパネル 117
  - Partitioned Databases (パーティシ  
ョン・データベース) パネル 117
- HALDB 索引/ILDS 再作成ユーティリ  
ティ 291
- ILDS
  - 再作成、DFSPREC0 291
- IMS 構成 143
- ISRDDN 143
- RECON データ・セット 143
- HALDB (高可用性ラージ・データベース)
  - 区画定義ユーティリティ
    - DBRC での OLR 機能の登録 121
- HALDB 索引/ILDS 再作成ユーティリ  
ティ (DFSPREC0)
  - 概要 291
  - 制約事項 292
  - 入出力 293
  - フリー・スペースのオプション 297
  - 戻りコード 294
  - ユーティリティ制御ステートメント  
295
  - 例 296
  - JCL の要件
    - DD ステートメント 293
    - EXEC ステートメント 293
  - QUIESCE データベース状況 292
  - VSAM ロード・モード 297
- HALDB マイグレーション・エイド・ユ  
ーティリティ (DFSMAID0)
  - 概要 321
  - サンプリング 327, 328
  - 出力メッセージと統計 321
  - 推奨事項 322
  - 制約事項 322
  - 前提条件 322
  - データ・スペース 327
  - データ・スペース、サイズの計算 328
  - データ・スペースおよび分析の種類  
327
  - 入出力 322
  - 戻りコード 324
  - ユーティリティ制御ステートメント  
325
  - 要件 322
  - DD ステートメント 324
  - EXEC ステートメント 323
  - JCL 指定 323
  - JCL の例 326
  - QUIESCE データベース状況 322
- HD 再編成アンロード・ユーティリテ  
ィ (DFSURGU0)
  - 推奨事項 355
  - 制約事項 352
  - 説明 349
  - 前提条件 353
  - 入出力 355
  - 戻りコード 360
  - ユーティリティ制御ステートメント  
361
  - 要件 353
  - DD ステートメント 357
  - EXEC ステートメント 357
  - JCL 指定 357
  - JCL の要件
    - DD ステートメント 357
- HD 再編成アンロード・ユーティリテ  
ィ (DFSURGU0) (続き)
  - JCL の例 362
  - QUIESCE データベース状況 352
  - RECON 状況 352
- HD 再編成再ロード・ユーティリテ  
ィ (DFSURGL0)
  - 出力メッセージと統計 335
  - 推奨事項 334
  - 制約事項 333
  - 説明 331
  - 前提条件 334
  - 統計 342
  - フリー・スペースのオプション 341
  - 報告書 342
  - 戻りコード 339
  - ユーティリティ制御ステートメント  
339
  - 要件 334
  - 例 344
  - DD ステートメント 335
  - EXEC ステートメント 335
  - JCL 335
  - QUIESCE データベース状況 333
  - VSAM ロード・モード 341
- High Availability Large Database  
(HALDB)
  - パフォーマンス
    - ILDS 342
- HISAM 再編成アンロード・ユーティリテ  
ィ (DFSURUL0)
  - 出力メッセージと統計 394
  - 推奨事項 394
  - 制約事項 392
  - 説明 391
  - 前提条件 393
  - 入出力 394
  - 戻りコード 400
  - ユーティリティ制御ステートメント  
概要 401
  - 要件 393
  - 例 404
  - CHANGE=ステートメント 402
  - JCL 指定 398
  - JCL の要件
    - DD ステートメント 398
    - EXEC ステートメント 398
    - OPTIONS=ステートメント 403
  - QUIESCE データベース状況 392
- HISAM 再編成再ロード・ユーティリテ  
ィ (DFSURRL0)
  - 出力メッセージと統計 383
  - 推奨事項 382
  - 制約事項 382
  - 説明 381
  - 前提条件 382

HISAM 再編成再ロード・ユーティリティ  
ー (DFSURRL0) (続き)  
入出力 383  
戻りコード 387  
ユーティリティ制御ステートメント  
388  
要件 382  
JCL 指定 385  
JCL の要件  
DD ステートメント 386  
EXEC ステートメント 386  
JCL の例 389  
QUIESCE データベース状況 382  
HLQ 仕様ステートメント  
ユーティリティ制御ステートメン  
ト、DFSUDMT0 193

## I

ID ステートメント  
データベース変更累積ユーティリティ  
ー (DFSUCUM0) 245  
IDXIN= キーワード  
UCF FUNCTION=RU 制御ステート  
メント 468  
ILDS (間接リスト・データ・セット)  
再作成  
フリー・スペースのオプション  
297, 341  
VSAM ロード・モード 297, 341  
パフォーマンス  
フリー・スペース、再作成 297,  
342  
フリー・スペース  
再作成中にフリー・スペースを含め  
る 297, 341  
ILPGM= パラメーター  
UCF FUNCTION=IL 制御ステートメ  
ント 457  
ILPSBNAM= キーワード  
UCF FUNCTION=IL 制御ステートメ  
ント 457  
INCR= キーワード  
MSDB 保守ユーティリティの  
ACTION ステートメント 154  
INDDS= キーワード  
制御ステートメント  
UCF FUNCTION=DR 451  
UCF FUNCTION=DX 455  
UCF FUNCTION=PR 461  
UCF FUNCTION=PU 463  
UCF FUNCTION=RR 464  
UCF FUNCTION=SR 471  
UCF FUNCTION=SX 476  
ISPF  
バッチ・ジョブ 111

ISPF メンバー・リスト 118  
表示 118  
ISPSTART 113  
ISPTLIB 141

## J

JOBLIB DD ステートメント  
プログラム分離トレース報告書ユーテ  
ィリティ (DFSPIRP0)  
(Program-Isolation-Trace Report  
utility (DFSPIRP0)) 503  
JOB= キーワード  
SB テスト・ユーティリティの  
SELECT ステートメント 515

## K

KDSDD= キーワード  
制御ステートメント  
UCF FUNCTION=RR 464  
UCF FUNCTION=RU 466  
UCF FUNCTION=SR 472  
UCF FUNCTION=SU 472  
UCF FUNCTION=SX 476  
KEYRANGE= オペランド  
部分的 DB 再編成  
ステップ 1 (DFSPRCT1) 427  
DB データベース調査ユーティリ  
ー 44  
KEY= オペランド  
MSDB 保守ユーティリティの  
MSDBINIT ステートメント 155  
KSCISZ= キーワード  
HISAM 再編成アンロード・ユーテ  
ィリティ (DFSURUL0) 402  
KSREC= キーワード  
HISAM 再編成アンロード・ユーテ  
ィリティ (DFSURUL0) 403

## M

MODE= キーワード  
データベース調査ユーティリティ  
(DFSPRSUR) 45  
MSDB 保守ユーティリティ  
ACTION ステートメント 153  
MSDB  
DEDB の MSDB への変換 415  
MSDB の DEDB への変換 415  
MSDB ダンプ・リカバリー・ユーティリ  
ティ (DBFDBDR0)  
推奨事項 301  
制約事項 300  
説明 299

MSDB ダンプ・リカバリー・ユーティリ  
ティ (DBFDBDR0) (続き)  
前提条件 300  
入出力 301  
戻りコード 303  
ユーティリティ制御ステートメント  
304  
要件 301  
例 305  
JCL 指定 302  
JCL の要件  
DD ステートメント 302  
EXEC ステートメント 302  
RECON 状況 300  
MSDB データベース  
置き換え 158  
削除 158  
挿入 158  
変更 158  
MSDB 保守ユーティリティ  
(DBFDBMA0)  
機能  
置き換え 158  
削除 158  
挿入 158  
変更 158  
推奨事項 149  
制約事項 148  
説明 147  
前提条件 149  
入出力 149  
戻りコード 151  
ユーティリティ制御ステートメント  
152  
ACTION ステートメント 152  
RUN ステートメント 152  
要件 149  
例 156  
JCL の要件  
DD ステートメント 151  
EXEC ステートメント 151  
QUIESCE データベース状況 148  
MSDB - DEDB 間変換ユーティリティ  
フォールバック 415  
DBT アンロード/再ロード以外のツ  
ールの使用 416  
MSDB - DEDB 間変換ユーティリティ  
(DBFUCDB0)  
推奨事項 408  
制約事項 407  
説明 407  
前提条件 407  
入出力 408  
ユーティリティ制御ステートメント  
411  
ユーティリティの実行 415

MSDB - DEDB 間変換ユーティリティ  
(DBFUCDB0) (続き)  
要件 408  
要約報告書 408  
例 411  
DD ステートメント 409  
DEDB の MSDB への変換 415  
EXEC ステートメント 408  
JCL 指定 408  
QUIESCE データベース状況 407  
MSDBDUMP データ・セット  
MSDB ダンプ・リカバリー・ユーティ  
リティ 299  
/DBDUMP コマンド 302  
MSGNUM= キーワード  
UCF FUNCTION=OP 制御ステートメ  
ント 450

## N

NBR= キーワード  
SB テスト・ユーティリティの  
SELECT ステートメント 516  
NOPUNCH パラメーター  
データベース事前再編成ユーティリ  
ティ (DFSURPR0) 319  
NOREADBACK ステートメント  
バッチ・バックアウト・ユーティリ  
ティ (DFSBB000) 232  
NOSEQCK ステートメント  
データベース・リカバリー・ユーティ  
リティ (DFSURDB0) 267  
NSTAT キーワード  
HISAM 再編成アンロード・ユーティ  
リティ (DFSURUL0) 403  
HISAM 再編成再ロード・ユーティリ  
ティ (DFSURRL0) 388

## O

OLR (online reorganization)  
HALDB (高可用性ラージ・データベー  
ス)  
PDU を使用した DBRC での OLR  
機能の登録 121  
OPTIONS=ステートメント  
データベース事前再編成ユーティリ  
ティ (DFSURPR0) 318  
HISAM 再編成アンロード・ユーティ  
リティ (DFSURUL0) 403  
HISAM 再編成再ロード・ユーティリ  
ティ (DFSURRL0) 388  
OUTDDS= パラメーター  
制御ステートメント  
UCF FUNCTION=DU 453

OUTDDS= パラメーター (続き)  
制御ステートメント (続き)  
UCF FUNCTION=DX 455  
UCF FUNCTION=IL 458  
UCF FUNCTION=IM 459  
UCF FUNCTION=PR 462  
UCF FUNCTION=RU 467  
UCF FUNCTION=SN 469  
UCF FUNCTION=SU 474  
UCF FUNCTION=SU 474  
UCF FUNCTION=SU 474  
UCF FUNCTION=SU 474

## P

PDS ディレクトリ 118  
PROC= キーワード  
MSDB 保守ユーティリティ  
(DBFDBMA0) 152  
PSB= オペランド  
部分的 DB 再編成  
ステップ 1 (DFSPRCT1) 428  
PUNCH パラメーター  
データベース事前再編成ユーティリ  
ティ (DFSURPR0) 318

## R

RBNID= キーワード  
UCF FUNCTION=ZB 制御ステートメ  
ント 479  
READBACK ステートメント  
バッチ・バックアウト・ユーティリ  
ティ (DFSBB000) 232  
RECON データ・セット  
HALDB (高可用性ラージ・データベー  
ス) 143  
RECOVCP パラメーター  
MSDB ダンプ・リカバリー 304  
RELATE= キーワード  
UCF FUNCTION=ZB 制御ステートメ  
ント 480  
UCF FUNCTION=ZM 制御ステート  
メント 482  
Remote Site Recovery (RSR)  
トラッキング・タイプ  
HALDB (High Availability Large  
Database) 120  
REP= キーワード  
UCF FUNCTION=ZB 制御ステートメ  
ント 479  
UCF FUNCTION=ZM 制御ステート  
メント 482  
REQUEST= キーワード  
制御ステートメント  
UCF FUNCTION=DU 454  
UCF FUNCTION=IM 460

REQUEST= キーワード (続き)  
制御ステートメント (続き)  
UCF FUNCTION=OP 450  
UCF FUNCTION=RR 465  
UCF FUNCTION=RU 467  
UCF FUNCTION=SN 470  
UCF FUNCTION=SR 472  
UCF FUNCTION=SU 474  
UCF FUNCTION=SU 474  
RESTART= キーワード  
データベース部分再編成  
ステップ 2 (DFSPRCT2) 429  
RSTRT 制御ステートメント  
データベース接頭部更新ユーティリ  
ティ (DFSURGP0) 21  
RSTRT=ステートメント  
データベース・スキャン・ユーティリ  
ティ (DFSURGS0) 37  
RUN ステートメント  
MSDB 保守ユーティリティ  
(DBFDBMA0) 152

## S

S (データベース・リカバリー) ステート  
メント  
データベース・リカバリー・ユーティ  
リティ (DFSURDB0) 268  
SB テスト・ユーティリティ  
(DFSBBHD0)  
イメージ・キャプチャー・ログ・レコ  
ード 507  
出力 511  
推奨事項 510  
制約事項 510  
説明 507  
前提条件 510  
入力 511  
戻りコード 514  
ユーティリティ制御ステートメント  
概要 514  
DBIO 514  
SELECT 515  
要件 510  
JCL の要件  
DD ステートメント 512  
EXEC ステートメント 511  
JCL の例 516  
SCANSEG= キーワード  
データベース部分再編成 ステップ 2  
(DFSPRCT2) 429  
SEG オプション  
データベース・スキャン・ユーティリ  
ティ (DFSURGS0) 35

SEGNAME= キーワード  
UCF FUNCTION=SN 制御ステートメント 469  
SELECT ステートメント  
SB テスト・ユーティリティ  
(DFSSBHD0) 515  
SEQ オプション  
データベース・スキャン・ユーティ  
リティー (DFSURGS0) 35  
SEQ= キーワード  
制御ステートメント  
UCF FUNCTION=DR 452  
UCF FUNCTION=DX 456  
UCF FUNCTION=IL 458  
UCF FUNCTION=IM 460  
UCF FUNCTION=PR 461  
UCF FUNCTION=PU 463  
UCF FUNCTION=RR 465  
UCF FUNCTION=RU 467  
UCF FUNCTION=SN 470  
UCF FUNCTION=SR 472  
UCF FUNCTION=SU 474  
UCF FUNCTION=SX 476  
UCF FUNCTION=ZM 483  
SET PATCH  
DFSMSdss コマンド 203  
SET PATCH 制御ステートメント  
ユーティリティー制御ステートメン  
ト、DFSUDMT0 195  
SICON= キーワード  
UCF FUNCTION=RU 制御ステート  
メント 467  
SNAP 出力データ・セット DCB 属性  
265  
SNAP 制御ステートメント  
データベース接頭部更新ユーティリ  
ティー (DFSURGP0) 21  
SO ステートメント  
データベース変更累積ユーティリ  
ティー (DFSUCUM0) 249  
SORTOPT= キーワード  
部分的 DB 再編成  
ステップ 1 (DFSURPCT1) 428  
STARTHEXT コマンド  
DEDB オンライン・ユーティリティー  
91  
STAT パラメーター  
データベース事前再編成ユーティリ  
ティー (DFSURPR0) 319  
STATS キーワード  
HISAM 再編成再ロード・ユーティリ  
ティー (DFSURRL0) 388  
STATS パラメーター  
HISAM 再編成アンロード・ユーティ  
リティー (DFSURUL0) 403

STOPHEXT コマンド  
DEDB オンライン・ユーティリティー  
79, 92  
SUMM パラメーター  
データベース事前再編成ユーティリ  
ティー (DFSURPR0) 319

## T

TIME= キーワード  
SB テスト・ユーティリティーの  
SELECT ステートメント 515  
TOAREA= キーワード  
データベース調査ユーティリティー  
(DFSRSUR) 45  
部分的 DB 再編成  
ステップ 1 (DFSURPCT1) 428  
TO= キーワード  
MSDB 保守ユーティリティーの  
MSDBINIT ステートメント 155  
TRACE= キーワード  
MSDB 保守ユーティリティーの  
ACTION ステートメント 154

## U

UNLOAD パラメーター  
MSDB ダンプ・リカバリー・ユーテ  
リティー 304  
UNLOADCP パラメーター  
MSDB ダンプ・リカバリー・ユーテ  
リティー 304

## V

VALUE= キーワード  
制御ステートメント  
UCF FUNCTION=ZB 479  
UCF FUNCTION=ZM 482  
VERIFY= キーワード  
制御ステートメント  
UCF FUNCTION=ZB 479  
UCF FUNCTION=ZM 482  
VSAM  
更新モード  
HALDB 索引/ILDS 再作成ユーテ  
リティー 297  
HD 再編成再ロード・ユーティリ  
ティー (DFSURGL0) 339  
ロード・モード  
HALDB 索引/ILDS 再作成ユーテ  
リティー 297  
HD 再編成再ロード・ユーティリ  
ティー (DFSURGL0) 341

VSAM (仮想記憶アクセス方式)  
UCF FUNCTION=SR 制御ステートメ  
ント 472

## W

WF1DDS= キーワード  
UCF FUNCTION=DR 制御ステート  
メント 452

## [特殊文字]

%DFSHALDB (HALDB 区画定義ユーテ  
リティー)  
インポート 116  
エクスポート 116  
概要 107  
推奨事項 108  
制約事項 108  
前提条件 108  
対話式アプリケーション 116  
入出力 108  
バッチ JCL の要件  
DD ステートメント 112  
フォアグラウンド JCL の要件  
DD ステートメント 110  
戻りコード 113  
ユーティリティー制御ステートメント  
113  
要件 108  
例 113, 115





プログラム番号: 5635-A04  
5655-DSM  
5655-TM2

Printed in Japan

SA88-5492-04



日本アイ・ビー・エム株式会社  
〒103-8510 東京都中央区日本橋箱崎町19-21

Spine information:

IMS バージョン 13

データベース・ユーティリティー

