

IMS
バージョン 15.1.0

システム管理

IBM

IMS

バージョン 15.1.0

システム管理

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、 979 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 15 (プログラム番号 5635-A06)、IMS Database Value Unit Edition V15.01.00 (プログラム番号 5655-DS5)、IMS Transaction Manager Value Unit Edition V15.01.00 (プログラム番号 5655-TM4)、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： SC27-6795-00

IMS

Version 15.1.0

System Administration

(November 13, 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	xiii
前提知識	xiii
新規および変更された情報の識別方法	xiii
IMS 15 のアクセシビリティ機能	xiv
ご意見の送付方法	xv

第 1 部 IMS システム管理の概要 1

第 1 章 IMS システムの概要	3
管理アクティビティの概要	3
IMS の環境および構成	5
DB/DC 環境	7
DBCTL 環境 (DBCTL environment)	11
DCCTL 環境	14
DB バッチ環境	19
TM バッチ環境	19
IMSplex の概要	20

第 2 章 システム管理者のための概念 35

アプリケーション・プログラムのシステム・サポート	35
IMS での動的割り振り	36
Base Primitive Environment (BPE) の概要	37
BPE コンポーネントのトレース	39
Common Service Layer の概要	43
IMSplex 内の CSL	44
CSL マネージャー	47
CSL での Single Point of Control (SPOC) プログラム	52
REXX SPOC API	54
グローバル・オンライン変更	55
ACB ライブラリー・メンバー・オンライン変更	56
グローバル TM リソース管理	58
自動 RECON 損失通知の概要	58
CSL を備えた IMSplex の構成	59
CSL 構成例	61
Common Queue Server の概要	71
キュー構造	74
リソース構造	75
CQS 構造機能	76
CQS リカバリー機能	77
タイプ 2 コマンド環境	78
拡張端末オプション	79
APPC	79
従属領域処理のセキュリティ	80
DB/DC 環境および DCCTL 環境での MPP スケジューリング	81
使用不能なデータに対するアプリケーション・プログラムのスケジューリング	82
高速機能 (Fast Path)	83

DBCTL 環境での高速機能	85
自動化操作プログラムのアプリケーション・プログラム	85
システム・ロギングとプロセス継続性	86
チェックポイント操作	87
DB/DC および DCCTL でのロッキング・メカニズムとデータベース健全性	88
出口点および出口ルーチン	89
データ・キャプチャー出口ルーチン	90
HALDB 区画選択出口ルーチン	91
z/OS 自動リスタート・マネージャー (ARM)	91

第 3 章 IMS システムの文書化 93

IMS システムの要件の抽出	93
設計の検討への参加	94
命名規則の設定	95
データ・ディクショナリーの使用	97
システムの特性の文書化	97

第 4 章 IMS のオペレーションおよびリカバリーの概要 101

操作タスクの理解	101
自動化オペレーション	102
リカバリー作業の理解	103
例: リカバリーが行われないシステム	104
リカバリー可能なシステムの要件	105
リカバリー処理のステップ	105
IMS におけるリカバリーのメカニズム	106
逆方向リカバリー	136
順方向リカバリー	137
IMS システムのシャットダウン	137
再始動時のリカバリー	138
リカバリーの複雑性	139
IMS リカバリーが不可能な場合	139
リカバリーのオーバーヘッド	140
IMS Application Menu	141

第 2 部 IMS システム管理の考慮事項とタスク 145

第 5 章 z/OS インターフェースの考慮事項 147

IMS のための z/OS インターフェースの考慮事項	147
インストールの問題の予防	147
JCL のセットアップ	148
従属領域の JES の構成	148
一部の必須の非標準 z/OS マクロをそれらのオリジナル・ライブラリーで保持	149
IBM 提供の z/OS プログラム特性テーブルの更新	149

必要な IMS リンクの z/OS へのインストール	152
タイプ 2 SVC モジュールのインストール	155
メモリー・ベースのデータ・セット ENQ 管理の使用可能化	156
IMS 異常終了フォーマット設定モジュール	156
印刷ダンプ出口制御表へのオフライン・ダンプ・フォーマット設定ルーチンの追加	158
DBRC タイプ 4 SVC のバインド	159
許可プログラム機能での IMS システム・データ・セットの許可	159
APPC/MVS 管理ダイアログの更新	160
RRS アーカイブ・ログ・ストリームの考慮事項	160
システム管理機能 DDCONS パラメーターの考慮事項	160
IRLM のための z/OS インターフェースに関する考慮事項	161
第 6 章 VTAM インターフェースの考慮事項	163
ネットワーク制御プログラム (NCP) の遅延設定	164
第 7 章 CSL の管理	165
CSL を使用した z/OS 自動リスタート・マネージャの使用	165
グローバル・オンライン変更の管理	166
グローバル・オンライン変更の使用可能化	166
グローバル・オンライン変更の使用不可能化	167
混合オンライン変更有効範囲を持つ IMSplex の保守	167
自動 RECON 損失通知の管理	168
IMSplex のモニター	169
IMSplex 問題の診断	169
第 8 章 CSL ODBM の管理	171
CSL 内での ODBM クライアントの登録	172
ODBM および RRS	172
ODBM と ODBA	172
ODBM を使用する ODBA アプリケーション・サーバーの構成	173
ODBM メッセージ経路指定	174
ODBM とセキュリティ	174
ODBM アカウンティング	175
CSL ODBM アカウンティング用のログ・ストリームの構成	176
ODBM アカウンティング用の SMF ログ・レコード	176
第 9 章 CSL OM の管理	181
CSL OM コマンドのルーティング	181
OM API を使用したコマンドの発行	182
CSL OM 監査証跡	183
CSL OM 監査証跡のログ・ストリームの構成	184
OM 監査ログ・レコード・フォーマット	185
OM ログ・レコードの印刷	186
CSL OM コマンド・セキュリティ	188

OM を介して入力されるコマンドに対する RACF 許可	189
IMS コマンド、RACF アクセス権限およびリソース名の表	190
CSL OM セキュリティー・ユーザー・ルーチン	199
第 10 章 CSL RM の管理	201
CSL RM により管理される情報	201
データベース、DEDB エリア、およびトランザクションに対するグローバル情報の保守	202
メッセージ宛先リソース情報の保守	205
OLCSTAT データ・セットに対するグローバル情報の保守	205
IMSplex 全体パラメーターの保守	206
IMS による、シスプレックス直列化プログラム管理のためのグローバル情報の維持方法	206
CSL RM 用のリソース構造の二重化要件	207
CSL RM によるリソース構造の再移植方法	207
z/OS によるリソース構造再作成方法	207
第 11 章 CSL SCI の管理	209
CSL SCI のセキュリティ	209
第 12 章 CQS の管理	211
CQS の始動に必要な情報の記録	211
障害クライアント・テークオーバー時の CQS へのクライアント接続の確立	211
CQS へのアクセスの許可	212
CQS 登録の許可	212
CQS 構造への接続の許可	213
CQS における構造変更の使用	213
CQS システム・チェックポイントの使用	214
CQS 構造チェックポイントの使用	216
CQS 構造フルの防止	217
CQS 構造オーバーフロー機能	218
キュー・スペース通知出口ルーチン (DFSQSSP0) を使用した共用メッセージ・キュー使用量のモニター	219
CQS 構造のフル・モニター	220
CQS 構造オーバーフローを伴う構造フル・モニターの使用	221
CQS における構造再作成	221
z/OS システム管理の再作成および CQS	222
CQS 管理の再作成	222
z/OS および CQS による構造再作成の開始	223
CQS 構造の再移植	223
CQS 構造リカバリー	224
CQS 構造のコピー	225
CQS における z/OS 構造の二重化	225
第 13 章 IMSRSC リポジトリの管理	229
RS カタログ・リポジトリ内の IMSRSC リポジトリの指定の更新	231
RS カタログ・リポジトリからの IMSRSC リポジトリの除去	232

IMSRSC リポジトリの定義と状況の表示	233
IMSRSC リポジトリを含む IMS の初期化	233
IMSRSC リポジトリの CSL RM 管理	235
IMSRSC リポジトリを含む CSL RM の初期化	235
CSL RM、IMS、および Repository Server の終了	238

第 14 章 IMS サービスの考慮事項 239

サービス SMP/E SYSMOD のタイプ	239
サービス SYSMOD のパッケージ化	240
IMS サービスの入手	241
保守に関する推奨	242
保守のインストールの準備状況の査定	242
新規または移行済みの IMS システムの保守サービス・レベルの更新	242
既存の IMS 実動システムのサービス・レベルの保守	243
保守の取得とインストール	244
単一システム上の IMS サービスのインストール	245
RECEIVE/APPLY/ACCEPT (標準的なシーケンス)	246
APPLY なしの ACCEPT (事前生成モード)	247
APPLY の前に ACCEPT (システム定義の必要なサービス)	251
IMSplex への IMS サービスのインストール	252
インストールおよび保守に関する一般的な問題	252
IMS システム定義が原因で APPLY のみの状態の SYSMOD に回帰が起こるのを防ぐ	253
非システム定義ターゲット・ライブラリーをビルドするための JCL の生成	254
IVP ダイアログの場合の保守の適用	254
z/OS のアップグレード	254
適切な SYSLIB 連結の確認	255
バインダー戻りコードの適切な解釈	256
SMP/E を使用したサンプル出口ルーチンのインストールとバインド	256
IMS の新しいバージョンへのマイグレーション	256

第 15 章 共用キューの計画 259

共用キュー環境での CQS の役割	259
共用キュー環境での IMS の役割	259
IMS がキューに関するインタレストを登録する方法	260
共用キュー環境でのメッセージ・フロー	262
作業単位 (UOW) トラッキング	263
端末自動ログオン	263
メッセージ通信およびプログラム間通信	263
動的制御ブロック	264
IMS キュー・マネージャー	264
MTO メッセージ	265
逐次アプリケーション処理	265
逐次トランザクション処理	265
共用キュー環境での会話型トランザクション	266
共用キュー環境の構成	266
共用キューの使用可能化	269

CFRM ポリシーの定義	270
z/OS ログ・ストリームの定義	272
CQS パラメーターの定義	273
共用キューの IMS パラメーター	273
Common Queue Server の使用	275
キュー制御機能を使用した構造のモニターおよび再キューイング	278
IMS コマンドによる CQS へのアクセス	279
リスト構造	280
関連するプリンターの使用	282
IMS フロントエンド切り替えの計画	283
シスプレックス全体にわたる処理に適しているかどうかの判別	283
シスプレックス環境での APPC および OTMA メッセージの管理	284
IMS 送達不能キューの管理	285
急送メッセージ・ハンドラーの使用	286
延期トランザクションの再キューイング	287
AOI トランザクションのスケジューリング	287
共用キュー環境での MSC の計画	287
共用キュー環境でのパフォーマンスのチューニング	288
共用キュー環境での高速機能トランザクションに関する推奨	289

第 16 章 IMS 環境でのデータ共用 291

データ共用の概要	291
DBRC およびデータ共用サポート	293
データ共用サポートがない場合のデータベース保全性	295
アプリケーションのデータ・アクセス方法	295
IMS システムのデータベースの共用方法	296
データベース・レベル共用における IRLM の使用	297
ブロック・レベル共用	298
シスプレックス・データ共用内の複数の IRLM	298
IRLM プロシージャラーの設定	299
IRLM のためのシステム初期設定	301
シスプレックス・データ共用の IRLM の定義	302
更新アクティビティーを行うデータベース・レベルでのデータ共用	303
複数の読み取りアクセスが行われるデータベース・レベルでのデータ共用	304
ブロック・レベルでのデータ共用	305
データを共用する IMS システムの調整	306
インストール・タスク	306
データ共用環境におけるバッチ・システムのための DBRC および IRLM サポート	307
データ共用からのデータベースの除外	308
実行 JCL の調整	308
z/OS オペレーティング・システムの調整	309
共用オプションによる VSAM データ・セット定義の調整	309
データ共用の開始および停止	310
IRLM の開始	310
IRLM の停止	311
データベース処理の開始および停止	311
データ共用システムのモニター	313

IRLM アクティビティーの状況の入手	313
コンポーネントおよびリソースの表示	313
RECON データ・セット内の情報のモニター カップリング・ファシリティー上の構造のモニター	315
DBRC を使用したデータベース割り振りおよびアクセスの制御	319
オンライン DBRC コマンド	320
データベース可用性の改善	321
オンライン変更の制御	321
データベースの再編成	321
オンライン・データベース・イメージ・コピーの作成	322
通常オペレーションの維持	322
バッチ更新プログラムのスケジューリング	322
オンライン更新プログラムの復元	323
更新機能の転送	324
データベースの再編成	325
イメージ・コピーの排他制御での作成	325
共用データベース・イメージ・コピーの作成	326
データベースの共用レベルを下げる	327
障害からのリカバリー	328
システム・ログの管理	328
リカバリー手順の計画	329
リカバリー機能	330
DBRC なしのリカバリー	332
IMS の障害後の再始動	332
DBRC の障害後の再始動	333
IRLM を伴うリカバリー	333
シスプレックス環境でのデータ共用	335
シスプレックス・データ共用の概念と用語	335
XRF およびシスプレックス・データ共用	339
シスプレックス・データ共用構成の例	339
シスプレックス・データ共用をどのような場合に使用するか	344
シスプレックス・データ共用のためのバッチ・ジョブ変更	344
カップリング・ファシリティー構造のサイズの計算	345
OSAM および VSAM 構造のサイズ変更	348
シスプレックス・データ共用の障害からのリカバリー	349
第 17 章 VTAM 汎用リソース・グループの計画	355
VTAM 汎用リソース・グループを使用するための要件	355
VTAM 汎用リソース・グループの制約事項	355
VTAM 汎用リソース類似性	356
VTAM 汎用リソース・グループの作成	357
APPC 汎用リソース名の指定	357
VTAM 汎用リソース・グループ内の IMS とのセッションの開始	357
XRF と VTAM 汎用リソース	358
ログオン・プロシージャの変更	359
APPLID フィールドのオーバーライド	359

類似性を終了する時期の判別	359
持続している類似性の終了	360
汎用リソース・グループからの IMS システムの除去	361
端末状況のリセット	361
類似性が持続している IMS 障害後のログオン	362
VTAM 汎用リソース・メンバー選択の制御	362
IMSplex 間の整合性の保証	363
IMS ESTAE プロセスにおける類似性管理のバイパス	364

第 18 章 IMSplex 内のトランザクション・マネージャーのリソースの計画 367

リソース名の一意性	367
リソース・タイプ整合性	368
グローバル呼び出し可能サービス	368
トランザクション・マネージャーのリソース	369
TM リソース: APPC 記述子	369
TM リソース: VTAM LTERM	369
TM リソース: MSNAME	370
TM リソース: VTAM 端末ノード	370
TM リソース: トランザクション	371
TM リソース: ユーザー名	372
TM リソース: ユーザー ID	372
RM およびリソース構造の IMS アクティビティーへの影響	373

第 19 章 IMS のセキュリティー 375

IMS でのデータ・セット暗号化のサポート	375
バッチ・ログ・データ・セットの暗号化	377
BPE トレース・データ・セットの暗号化	377
変更累積データ・セットの暗号化	378
IMS Connect レコーダー・データ・セット (非 BPE トレース) の暗号化	378
CQS SRDS データ・セットの暗号化	379
IMS 外部トレース・データ・セットの暗号化	379
GSAM データベース・データ・セットの暗号化	380
イメージ・コピー・データ・セットの暗号化	380
オンライン・ログ・データ・セット (OLDS) の暗号化	380
SLDS/RLDS (IMS アーカイブ・ログ・データ・セット) の暗号化	383
全機能 VSAM データベース・データ・セット (非 HALDB または非 OLR 対応) の暗号化	384
全機能 VSAM データベース・データ・セット (HALDB、OLR 対応) の暗号化	384
z/OS ログ・ストリームのオフロード・データ・セットおよびステージング・データ・セットの暗号化	384
先行書き込みデータ・セット (WADS) の暗号化	385
DB/DC および DCCTL セキュリティーの概要	386
保護できる DB/DC および DCCTL リソース	386
DB/DC および DCCTL システム定義時のセキュリティーの定義	387
DB/DC および DCCTL リソース用のセキュリティー機能	388

IMS DB/DC および DCCTL 用のセキュリティー の設計	389
端末からのアクセスの制限	390
トランザクションとコマンドの許可	392
RACF を使用した物理端末の保護	394
ネットワーク・セキュリティー資格情報の伝搬	395
マスター端末のセキュリティーに関する考慮事項	401
AO アプリケーション・プログラムのセキュリ ティー	401
時間制御操作のセキュリティー	407
高速機能アプリケーション・プログラムのセキュ リティー	408
JMP アプリケーション・プログラムのセキュ リティー	409
セキュリティーと CPI-C ドリブン・アプリケー ション・プログラム	409
ODBA アプリケーション・プログラムのセキュ リティー	409
RACF データ・スペースの使用	410
MSC および共用キュー環境でのセキュリティー	410
APPC/IMS セキュリティー	415
ETO 端末のセキュリティー	415
DB/DC および DCCTL 従属領域とそれぞれの リソースの保護	416
OTMA のセキュリティー	419
IMS Connect のセキュリティー	420
DB/DC 環境および DCCTL 環境向けの IMS セキ ユリティーの活動化	421
セキュリティー出口ルーチンの準備	421
RACF セキュリティー計画の準備	422
APSB SAF セキュリティーの使用可能化および 使用不可化	426
RS カタログ・リポジトリおよび IMSRSC リ ポジトリへのアクセスの制限	426
DB/DC および DCCTL 用のシステム始動時のセ キュリティーの制御	430
DB/DC 環境および DCCTL 環境におけるオンラ インでのセキュリティー変更の実装	433
DB/DC 環境および DCCTL 環境でのセキュリテ ィー違反の制御	433
その他のアクセス管理方式に関する考慮事項	434
物理的セキュリティー	434
DB/DC および DCCTL における表示バイパス およびパスワード・マスキングの使用	434
リソースの保護	435
暗号化: アクセス制御に代わるもの	437
DBCTL 環境のセキュリティーに関する考慮事項	438
保護できる DBCTL リソース	438
システム定義時に行われる DBCTL セキュリテ ィーの選択	439
DBCTL リソース用のセキュリティー機能および セキュリティー・タイプ	439
DBCTL セキュリティーの設計に関する考慮事項	440
IMS DBCTL セキュリティーの活動化	442
DBCTL セキュリティー用のシステム始動の制御	442

オンラインでの DBCTL セキュリティー変更の 実装	443
DBCTL セキュリティー違反の制御	444
第 20 章 IMS の制御	445
システムのモニター	445
分析のための IMS システム・ログ情報の処理	445
IMS システム・ログ・ユーティリティーの使用	445
IMS データベース生産性向上ツール	446
システムのリカバリー	446
システム・リソースの変更および制御	447
データ共用の制御	448
DBRC を使用したデータ共用の制御	448
データ共用システムのモニター	448
ログ・データ・セットの特性の制御	449
サブシステム接続および切断	449
第 21 章 IMS の開始または再始動	451
第 22 章 IMS のシャットダウン	453
第 23 章 システムのテスト	455
テスト・システムの必要性	456
テスト・システムの設定	456
テスト・データベースの設定	457
操作手順のテスト	457
IMS テスト環境でのモニター	458
DB/DC 環境でのモニター	459
ネットワークの作動可能性の確認	459
DB/DC 環境および DCCTL 環境でのネットワ ーク・テスト	460
DBCTL 環境でのテスト	460
DCCTL 環境でのテスト	460
IMS テスト・エイド	460
DB/DC 環境および DCCTL 環境におけるバッ チ端末シミュレーターによるオンライン実行のシ ミュレーション	461
DB/DC 環境および DCCTL 環境での MFS 形 式のオンライン・テスト	461
テストのための動的リソース定義とオンライン変 更の使用	462
DB/DC 環境および DCCTL 環境での SYSIN/SYSOUT を使用したプログラム・テスト	463
DB/DC 環境および DCCTL 環境でのテレプロ センシング・ネットワーク・シミュレーターを使用 したネットワーク・テスト	464
キュー制御機能によるパフォーマンスとストレス のテスト	464
第 24 章 IMS モニター・データの収集 および解釈	465
モニター手順の確立	465
パフォーマンス目標の確立	466
ワークロード管理の計画	468
モニター・アクティビティーおよび技法の決定	476
詳細モニター用ツール	480

IMS モニター	480
z/OS 汎用トレース機能 (GTF)	482
z/OS コンポーネント・トレース (CTRACE) サービス	483
プログラム分離およびロック・トレースの入手	483
IMS トレース機能	485
OTMA メッセージ・ルーティングのパフォーマンスの分析	486
MSC ネットワークでのパフォーマンス情報の調整	486
DB/DC 環境および DCCTL 環境での高速機能システムのモニター	487
DB/DC 環境および DCCTL 環境でのトランザクション・フロー	487
DB/DC 環境および DCCTL 環境での IMS モニター	492
トランザクション・レベル統計のモニター	492
DBCTL 環境でのモニター手順	493
DBCTL 環境でのパフォーマンス目標の確立	495
DBCTL 環境でのモニター・アクティビティおよび技法の決定	497

第 25 章 システム設計の変更 501

アプリケーションの変更の評価	501
アクティブ IMS システムへの変更されたアプリケーションの紹介	504
非アクティブ ACB ライブラリーのオンラインでのサイズ変更	505
ACB ライブラリーの静的割り振りから動的割り振りへのマイグレーション	505
システム定義の変更の計画の立案	506
システム定義プロセスの管理	506
必要なシステム定義タイプの判別	507
DB/DC 環境および DCCTL 環境でのネットワーク定義の変更	507
システム・チューニングの変更	507
リソース使用状況の変更	508
アプリケーションおよびデータベース設計の変更	508
DB/DC 環境および DCCTL 環境での通信設計の変更	509
オンライン・システム定義の変更の管理	509
システム修正変更がオンライン変更を使用できるかどうかの判断	509
オンライン変更を計画する際の考慮事項	510
容量計画の実行	512

第 26 章 システムのチューニング 515

パフォーマンスの管理	516
変更管理機能	517
設計変数	518
予測のタイプ	518
チューニングのための z/OS および IMS パラメータの初期設定	518
z/OS ディスパッチング優先順位の割り当て	519
パフォーマンスに対する IMS オプションの選択	520
IMS リソース (バッファ・プールを除く) に対する競合の回避	523

IMS バッファ・プールのチューニング	526
IMS が制御する入出力とページングとのトレードオフ	531
パス長さの最小化	534
DB/DC 環境および DCCTL 環境での通信ネットワークに関する考慮事項	535
IMS システム・データ・セットの配置に関するガイドライン	536
入出力サブシステムの構成	537
DB/DC 環境および DCCTL 環境でのアプリケーション最適化	537
DL/I に関する考慮事項	538
共用キュー環境でのパフォーマンスの計画	538
パフォーマンスの問題の識別と訂正	539
ページング率の検討	540
プロセッサ・リソースに関する問題の検出	540
入出力リソース競合を除去するためのチューニング	541
DB/DC 環境および DCCTL 環境での通信サブシステムの競合	541
DB/DC 環境および DCCTL 環境での IMS メッセージ処理	543
DB/DC 環境および DCCTL 環境での入力キューイングとスケジューリング/終了	545
プログラムのロードと初期設定	547
プログラム実行時間	548

第 27 章 オンライン変更の実行 551

管理対象 ACB 環境でのオンライン変更	553
IMS が ACB を管理する場合のリソースの活動化の概要	554
IMS が ACB を管理する場合の変更されたリソースの活動化	555
IMS が ACB を管理する場合の新規リソースの活動化	557
IMS カタログが共用されている場合の活動化	559
IMSplex 内のシステムのサブセットでの PSB の活動化	561
オンライン変更機能	569
ローカル・オンライン変更機能の概要	570
グローバル・オンライン変更機能の概要	577
コールド・スタートの実行	601

第 28 章 IMS スプール API による出力の印刷 605

設計および操作上の考慮事項	605
データ・マネージャーとしての IMS スプール API	608
印刷データ・セットの特性	608
変更 (CHNG) 呼び出し	609
オプションの設定 (SETO) 呼び出し	610
出力 DD ステートメント	610
IMS スプール API へのデータの書き込み	610
挿入呼び出し	611
PURG 呼び出し	611
ROLL および ROLB 呼び出し	612
SETS、SETU、および ROLS 呼び出し	612

特殊な考慮事項 — 許可される記述子	612	データ・セット命名規則	655
印刷データ・セットの制御	612	イメージ・コピー・データ・セットの命名規則	656
高速 代替 PCB	613	重複イメージ・コピー・データ・セットの命名規	657
XRF 環境	613	則	657
割り振りエラーについて	614	変更累積データ・セットの命名規則	657
第 29 章 IMS 内から外部システムへの		データベースのバックアップ・コピー	658
アクセス	615	イメージ・コピー・ユーティリティー	
第 3 部 DBRC 管理	617	(DFSUDMP0、DFSUDMT0、DFSUICP0)	658
第 30 章 DBRC の概要	619	並行イメージ・コピー	663
DBRC コンポーネント	620	将来の利用と再利用のためのイメージ・コピー・	
DBRC タスク	621	データ・セットの作成	664
DBRC コマンドおよび API 要求	622	管理されるイメージ・コピー数の制御	665
第 31 章 ログ情報の記録および制御	625	イメージ・コピー・データ・セットおよび	
RECON データ・セットのログ・レコードの変更	627	GENMAX のリカバリー期間	665
OLDS のアーカイブ	627	変更累積データ・セットのリカバリー期間と	
DBRC ログ関連のコマンド	628	GRPMAX	666
第 32 章 データベースのリカバリー	631	イメージ・コピー・データ・セットの再使用	668
第 33 章 DBRC はリカバリーにどのよ		HSSP イメージ・コピー	669
うに役立つか	635	HISAM コピー (DFSURUL0 および	
リカバリー JCL の生成	635	DFSURRL0)	669
ユーティリティー JCL の妥当性検査	636	標準外イメージ・コピー・データ・セット	670
第 34 章 データベースのオープンと更		バックアップ・コピーの頻度および保存	671
新に関する情報の記録	639	ログ・レコードの変更累積	671
第 35 章 データ共用のサポート	641	累積した SLDS または RLDS (変更累積) の圧	
データ共用のレベル	641	縮	672
RECON データ・セット内のデータ共用情報の記録	641	どのようなときに変更累積が必要か	673
DBRC による共用レベルの割り当て	642	DBDS グループの考慮事項	676
第 36 章 IMSplex のサポート	643	DB グループ	677
第 37 章 DBRC の IMS に対しての定		DBRC グループ	677
義	645	HALDB オンライン再編成に関する DBRC の考慮	
第 38 章 データベースとデータベ		事項	678
ース・データ・セットの登録	647	オンライン再編成に影響されるリカバリー関連レ	
第 39 章 リカバリーの計画	649	コードの保守	680
リカバリー・メカニズムの設定	649	オンライン再編成処理	681
リカバリー機能	649	第 41 章 DBRC セキュリティー	683
DBRC なしのリカバリー	652	DBRC コマンドおよび API 要求のセキュリティー	683
IMS の障害後の再始動	652	RACF による DBRC コマンドと API 要求の許	
DBRC の障害後の再始動	653	可	684
IRLM 構成を伴うリカバリー	653	DSPDCAX0 出口ルーチンによる DBRC コマン	
バッチ・バックアウト	653	ドおよび API 要求の許可	685
第 40 章 DBRC システムの考慮事項	655	DSPDCAX0 出口ルーチンと RACF の両方を使	
DBRC 使用上の一般的な考慮事項	655	用した DBRC コマンドおよび API 要求の許可	685
		コマンド許可のためのリソース名	686
		RECON データ・セットの DBRC セキュリティー	
		のオーバーライド	690
		第 42 章 RECON データ・セットの初	
		期化および保守	691
		RECON データ・セットの計画の考慮事項	692
		RECON データ・セットの最初の初期化	692
		RECON データ・セットの競合問題の回避	692
		RECON データ・セットへの初期アクセス	702
		RECON データ・セット内のレコード	704
		RECON ヘッダー・レコード	704
		ログ・データ・セット・レコード	705

データベース・リカバリー・レコード	706
RECON データ・セット内のレコードのマッピング	715
RECON データ・セットの保守	716
RECON データ・セットのバックアップ	716
不要な RECON レコードの削除	716
RECON データ・セットの再編成	719
損傷した RECON データ・セットの置き換え	721
RECON データ・セットのリカバリー	721
廃棄された RECON データ・セットの置き換え	723
RECON データ・セットの修復	723
RECON 損失の通知	725
RECON データ・セットに行われた変更のトラッキング	726

第 43 章 DBRC のヒント 727

タイム・スタンプを元の現地時間で出力するための RECON データ・セットの変更	727
RECON データ・セット内の最後の SLDS 停止時刻の検出	727
GENMAX の調整 (その値に達したか値が大きすぎる場合)	729
GRPMAX の調整 (その値に達したか値が大きすぎる場合)	730
オープン・オンライン PRILOG レコードのクローズ	731
サブシステム・レコード (SSYS) の処理	733
DB/AREA レコードから SSYS 間の許可の不整合を除去する方法	734
ログの変更累積プロセスの再始動	734
処理対象がないとされた場合の変更累積作業の再始動	735
ログ・データ・セットを移動する	735
データ・セットをカタログする	736
テスト環境で複数コールド・スタートを行なう	737
潜在的な RECON データ・セット・エンキュー問題の削減	738

第 4 部 IMS システム・リカバリー 741

第 44 章 拡張回復機能の概要 743

XRF の概要	743
IMSplex 内の XRF	744
XRF の利点を生かせるインストール・システムのタイプ	744
XRF 概念と用語	745
XRF 複合システム	747
DBCTL 機能	750
XRF テークオーバー (XRF takeover)	750
XRF 要件	756
XRF プロセスでのコンポーネントの役割	758
XRF プロセスに与える IMS の利点	758
z/OS および z/OS エレメントの利点	761
XRF の監視の確立	768
XRF プロセスのフェーズ	771
XRF プロセスの初期設定フェーズ	771
XRF プロセスの同期フェーズ	773

XRF プロセスのトラッキング・フェーズ	774
XRF プロセスのテークオーバー・フェーズ	776
XRF プロセスのテークオーバーの後処理フェーズ	786
XRF プロセスの終了フェーズ	788
XRF フェーズのサイクル	789
XRF 複合システムの編成	789
1 つの CPC を持つ 1 つの XRF 複合システム	789
2 つの CPC を持つ 1 つの XRF 複合システム	791
2 つの CPC を持つ 1 つの XRF 複合システムと XRF のない IMS	792
3 つの CPC を持つ 2 つの XRF 複合システム	793
4 つの CPC を持つ 2 つの XRF 複合システム	796
XRF 複合システムの計画	798
XRF の制限事項	798
代替 IMS での非 XRF 作業負荷	798
システム間連絡リンクの使用	799
XRF 複合システムの端末	800
テークオーバーが端末ユーザーに与える影響	805
端末切り替えに VTAM 所有権が与える影響	810
クラス 1 端末の VTAM 所有権についての計画	810
USERVAR を使用する XRF についての NCP 計画に関する考慮事項	811
パフォーマンスの考慮事項	812
z/OS 計画に関する考慮事項	812
XRF のためのシステムの準備	814
IMS 実行 JCL の調整	814
XRF に対する IMS システム定義マクロ・ステートメントのコーディング	815
XRF のための IMS.PROCLIB メンバーの定義	818
VTAM USERVAR テーブルの定義	827
XRF 構成における IMS データ・セットの配置	828

第 45 章 IMSplex でのリカバリー 835

第 5 部 IMS 報告書の使用 837

第 46 章 DB モニター報告書 839

VSAM バッファー・プール報告書	839
VSAM バッファー・プール報告書の使用方法	839
VSAM バッファー・プール報告書のフィールド	840
VSAM 統計報告書	846
VSAM 統計報告書の使用方法	846
VSAM 統計報告書のフィールド	847
データベース・バッファー・プール報告書	852
データベース・バッファー・プール報告書の使用方法	852
データベース・バッファー・プール報告書のフィールド	852
プログラム入出力報告書	854
プログラム入出力報告書の使用方法	855
プログラム入出力報告書のフィールド	855
DL/I 呼び出し要約報告書	857
DL/I 呼び出し要約報告書の使用方法	857
DL/I 呼び出し要約報告書のフィールド	857
分布付録報告書	860

分布付録報告書の作成方法	862
分布できるイベントとそのデフォルトの範囲	862
モニター・オーバーヘッド報告書	864
モニター・オーバーヘッド報告書のフィールド	864
第 47 章 IMS モニター報告書	865
トランザクションのフローと IMS モニターのイベント	865
IMS モニター・トレース・イベントの間隔	868
IMS モニター報告書の概要	869
モニター実行の文書化	870
IMS モニター報告書オカレンスの検査	872
従属領域内のアクティビティのモニター	873
データベース処理意図の競合の検出	876
チェックポイントの影響の検査	876
領域占有の測定	877
アプリケーション・プログラムの経過時間のモニター	877
アプリケーション・プログラム DL/I 呼び出しのための入出力のモニター	880
MFS アクティビティのモニター	884
メッセージ・キュー処理のモニター	885
チェックポイントの影響の検出	885
トランザクション・キューイング報告書	886
データベース・バッファのモニター	887
回線アクティビティのモニター	888
メッセージ処理効率のモニター	889
IMS 内部リソースの使用	890
IMS モニター出力の度数分布の使用法	893
度数分布出力の入手方法	895
度数分布範囲の定義方法	896
分布付録の解釈	897
MSC の IMS モニター報告書	899
MSC トラフィック報告書	899
MSC 要約報告書	901
MSC キューイング要約報告書	902
第 48 章 DBCTL の IMS モニター報告書	903
IMS モニター・トレース・イベントの間隔	904
IMS モニター報告書の概要	905
モニター実行の文書化	906
従属領域内のアクティビティのモニター	909
アプリケーション・プログラムの経過時間のモニター	913
アプリケーション・プログラム DL/I 呼び出しのための入出力のモニター	915
トランザクション・キューイング報告書	916
データベース・バッファのモニター	917
IMS 内部リソースの使用	918
IMS モニター出力の度数分布の使用法	920
分布付録出力の解釈	923
第 49 章 DCCTL の IMS モニター報告書	925
IMS モニター・トレース・イベントの間隔	925

IMS モニター報告書の概要	926
モニター実行の文書化	927
従属領域内のアクティビティのモニター	930
チェックポイントの影響の検査	933
領域占有の測定	933
アプリケーション・プログラムの経過時間のモニター	934
アプリケーション・プログラム DL/I 呼び出しのための入出力のモニター	936
MFS アクティビティのモニター	939
メッセージ・キュー処理のモニター	940
チェックポイントの影響の検出	940
トランザクション・キューイング報告書	941
回線アクティビティのモニター	942
メッセージ処理効率のモニター	943
IMS 内部リソースの使用	944
IMS モニター出力の度数分布の使用法	945
分布付録出力の解釈	949
第 50 章 //DFSSTAT 報告書	951
PST 会計報告書	951
VSAM バッファ・プール報告書 (バッチ領域の場合のみ)	952
OSAM バッファ・プール報告書 (バッチ領域の場合のみ)	953
順次バッファリング要約報告書	954
順次バッファリング明細報告書	957
第 51 章 統計分析報告書、ログ・トランザクション報告書、およびログ・レコード分析	967
統計分析ユーティリティの報告書	967
トランザクション負荷の計算	968
プログラム間トラフィックの評価	969
未送信メッセージ数の入手	970
重要なトランザクションの監査	970
ログ・トランザクション分析ユーティリティの報告書	970
スケジューリング・アクティビティの検査	971
IMS 会計情報	974

第 6 部 付録	977
特記事項	979
プログラミング・インターフェース情報	981
商標	981
製品資料に関するご使用条件	982
IBM オンライン・プライバシー・ステートメント	983
参考文献	985
索引	X-1

本書について

これらのトピックでは、単一の IMS[™]、または 1 つの単位として動作する 1 つ以上の IMS システム (IMSplex) の管理と操作に関するガイダンス情報を提供します。これらのトピックでは、IMS システムの設計、文書化、操作、保守、およびリカバリーについて説明するとともに、データベース・リカバリー管理 (DBRC) 機能、拡張リカバリー機能 (XRF)、IMSRSC リポジトリ、およびリポジトリ・サーバーについても説明します。これらのトピックには、IMSplex に含めることができる IMS Base Primitive Environment (BPE)、IMS Common Queue Server (CQS)、および IMS Common Service Layer (CSL) に関する情報のほか、IMSplex 内でデータとメッセージ・キューを共用するための情報も含まれます。

この情報は、IBM[®] Knowledge Center で参照できます。

前提知識

本書を使用する際には、z/OS[®] および IMS の基本概念、インストールされている IMS システムを理解しておいてください。IMS を実行できる環境には、DB バッチ、DCCTL、TM バッチ、DB/DC、DBCTL があります。インストールするシステムに適用される環境について理解しておく必要があります。また、データベース処理と DL/I で使用するアクセス方式についての基本知識が必要です。さらに、さまざまなタイプの DL/I 呼び出しの目的、IMS アプリケーション・プログラムの構造、アプリケーション・プログラム設計に関連したタスクを知っておくと役立ちます。

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。

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 (改訂マーカー) 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、改訂マーカーは含まれていません。

改訂マーカーは、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。

IMS 15 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 15 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 15 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 15 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 15 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) を参照してください。

ご意見の送付方法

お客様のご意見を送り返していただくことは、弊社が正確な情報を提供し、品質の高い情報を提供するうえで重要なことです。本書またはその他の IMS 関連資料についてコメントのある場合、次のいずれかの方法でお送りください。

- IBM Knowledge Center のトピックの下部にある「**Contact Us**」タブをクリックします。
- imspubs@us.ibm.com に E メールを送信します。必ず、資料タイトルと資料番号を記載してください。

弊社が迅速かつ正確に対応するために、ご意見をお送りいただく資料の内容、その掲載箇所、改善のためのご提案について可能な限り多くの情報を記載してください。

第 1 部 IMS システム管理の概要

このトピックでは、IMS システムを紹介します。また、IMS システムを管理する中心的概念についても紹介します。

第 1 章 IMS システムの概要

このトピックでは、IMS システムの管理アクティビティーの概要を説明します。また、IMS システムを管理する中心的な概念についても紹介します。

管理アクティビティーの概要

IMS オンライン・システムを管理するには、IMS オンライン・システムを設計し、アプリケーションの要件を満たす操作手順を確立し、エンド・ユーザーに対応する実動システムを保守して、新しいアプリケーションまたは主要な設計変更を現行システムに組み込む必要があります。

これらの責任を果たすためには、アプリケーション開発サイクルで発生する多くのアクティビティーを調整しなければなりません。それらのアクティビティーの実行の結果として、次のものが得られます。

- IMS ネットワークの文書化
- IMS のシステム定義パラメーターと実行制御パラメーターの指定
- 操作のプロシーチャー制御
- モニターおよび監査証跡の方法

実動モードに入った後、管理者はアクティビティーを継続しながら次のものをサポートします。

- 操作とエンド・ユーザー・サービスの監査
- 実動統計のモニターと収集
- オンライン・システム設計の変更を制御するプロシーチャーの確立
- アプリケーションおよび IMS を変更した後のオンライン・システムのテスト

以下の図は、管理アクティビティーの概要を示しています。この図の最初の列に示したアクティビティーは設計段階で発生し、2 列目のアクティビティーはアプリケーション・コードの開発時に発生し、3 列目のアクティビティーはテスト時に発生します。中央の縦の線は、実動モードへのマイグレーションを示すマークです。

実動モードを開始する前の管理アクティビティーとして、次の 2 つの主要なアクティビティーがあります。

- システムの生成および JCL (ジョブ制御言語) の作成
- 操作プロシーチャーの開発

この図で中央線の右側にある項目は、継続中のシステム管理アクティビティーを反映しています。「PRODUCTION」というヘッダーが先頭にある列では、システムの日常の操作とパフォーマンスの把握に重点が置かれています。その次の列のアクティビティーは、維持管理に関するものです。アプリケーションの小規模な設計変更、問題解決、IMS のすべての維持管理がこのカテゴリーに含まれます。一番右側の列は、追加アプリケーションの組み込みやアプリケーション・パッケージの実装に必要なアクティビティーを示しています。大規模な追加の場合は、図の左側に

示したのと同様なアクティビティーになります。端末やネットワークの修正を必要としないその他の設計変更は、IMS をシャットダウンしなくても処理できます。このような単純な設計変更の場合、関連のアクティビティーが保守モードでの操作手順改訂の原因となります。

始動後は、IMS のシステム定義と操作手順の改訂が重要なアクティビティーになります。その状況では、オンライン操作の間のアプリケーションの変更を決定する場合もあります。その場合、モニターとパフォーマンス分析に基づいたシステム・パフォーマンスの解釈が重要なアクティビティーとなります。

図では、1 つの行は 1 つのアクティビティー・セットを表しており、各アクティビティー・セットには次のような独自の特性があります。

- ユーザー要求の分析には、必要な IMS 機能と予期される作業負荷についてのアプリケーション文書を調べることが含まれます。
- オンライン要件の収集は、IMS システム定義、システム・データ・セットの割り振り、初期 JCL についての指定に関係します。
- IMS ネットワークの準備には、システムとの対話とネットワーク構築アクティビティーが含まれます。
- セキュリティー・プロシージャの確立には、セキュリティ方法の設計と実施が含まれます。
- 操作計画の開発では、操作制御のための文書を作成し、実動サイクルの制御の監査に備えます。
- モニター戦略の開発は、システムのモニターとパフォーマンス・データの収集に帰結します。
- パフォーマンスの基準の確立は、パフォーマンス分析とチューニング・アクティビティーに帰結します。

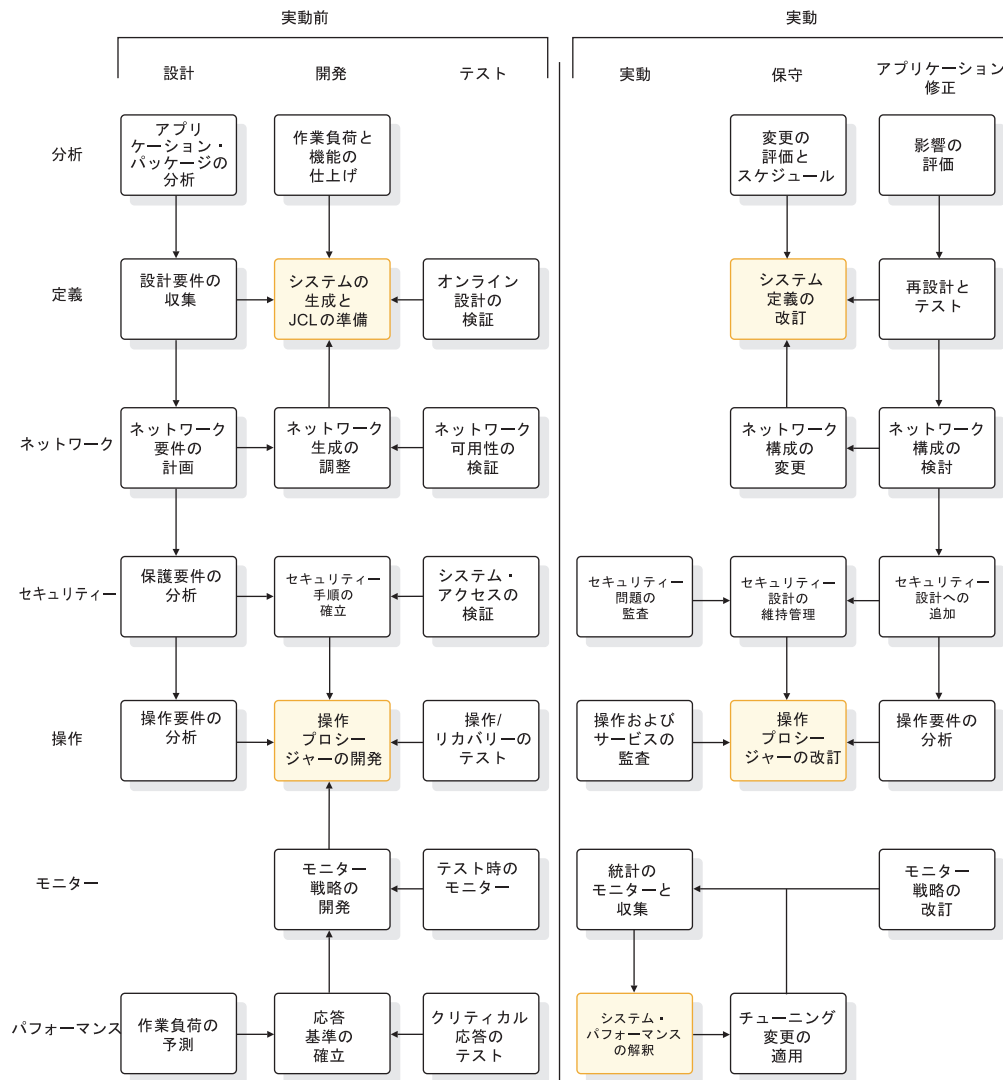


図 1. IMS システム管理アクティビティーの図

関連資料: 本書では、操作手順の確立に必要な詳細な計画立案作業については説明していません。操作手順の確立については、「IMS V15 オペレーションおよびオートメーション」を参照してください。

IMS の環境および構成

IMS は、IMS Database Manager (DB) と IMS Transaction Manager (TM) の 2 つの主要コンポーネントで構成されています。これらは、一緒に使用することもできますし、独立して使用することもできます。一緒に使用すると、DB/DC 環境が構成されます。

Database Manager を単独で使用すると、バッチ環境とデータベース制御 (DBCTL) 環境を生成できます。Transaction Manager を単独で使用した場合は、データ通信制御 (DCCTL) 環境を生成できます。データ共用および拡張回復機能 (XRF) は、環境と見なされることが少なくありませんが、ここに挙げた 3 つの環境の特殊な事例です。

3つの環境 (DBCTL、DB/DC、および DCCTL) のほかに、IMS を IMSplex として構成できます。

IMS の主要なコンポーネントを使用して、業務上の要件に基づいたシステム定義プロセスの一部としてユーザーの環境を構成します。

DB/DC、DCCTL、および DBCTL の各環境はすべて、オンライン IMS システムと見なされます。

各 IMS 環境は、ハードウェアとプログラムのさまざまな組み合わせからなっており、それぞれ異なる処理目標をサポートします。オンライン環境とそれがサポートする目標を、以下の表に示します。

表 1. IMS オンライン環境

環境	データ・プロセスの目標
DBCTL (11 ページの『DBCTL 環境 (DBCTL environment)』を参照)	<ul style="list-style-type: none"> • Transaction Manager を使用せずにネットワーク・トランザクションを処理する。つまり、Database Manager をトランザクション管理サブシステム (例えば CICS®) と一緒に使用する。 • ある一定のインターバルで DB バッチを使用してバッチ・アプリケーション・プログラムを実行する (例えば、給与計算を処理したり在庫報告書を作成する)。 • DB バッチを使用してデータベース・ユーティリティーを実行する。
DB/DC (7 ページの『DB/DC 環境』を参照)	<ul style="list-style-type: none"> • 満足できるリアルタイム・パフォーマンスで端末ユーザーがデータを検索し、データベースを修正できるようにする。(代表的なアプリケーションは、金融機関業務処理、航空座席予約、販売受注などである。) • 検索したデータが最新のものであることを保証する。 • 通信ネットワーク内の複数の CPU 間でトランザクション処理を分散する。 • ある一定のインターバルで DB バッチを使用してバッチ・アプリケーション・プログラムを実行する (例えば、給与計算を処理したり在庫報告書を作成する)。 • DB バッチを使用してデータベース・ユーティリティーを実行する。 • 外部サブシステムにアクセスするアプリケーション・プログラム、または Db2® for z/OS のデータなどの、外部サブシステム内のデータにアクセスするアプリケーション・プログラムを実行する。
DCCTL (14 ページの『DCCTL 環境』を参照)	<ul style="list-style-type: none"> • Database Manager を使用せず、Transaction Manager を外部のデータベース管理サブシステムと一緒に使用してネットワーク・トランザクションを処理する。 • DBRC を使用して、再始動用のシステム・ログ情報を保守する。 • 外部サブシステムにアクセスするアプリケーション・プログラム、または Db2 for z/OS のデータなどの、外部サブシステム内のデータにアクセスするアプリケーション・プログラムを実行する。

関連概念:

20 ページの『IMSplex の概要』

DB/DC 環境

DB/DC 環境では、データは並行して実行されているいくつかのアプリケーション用に中央で管理され、端末ユーザーが使用できるようになります。データベース・リカバリー管理 (DBRC) 機能は、データベースの可用性、データ共用、システム・ロギングの管理に役立ちます。

以下の図は、DB/DC 環境の例を示しています。

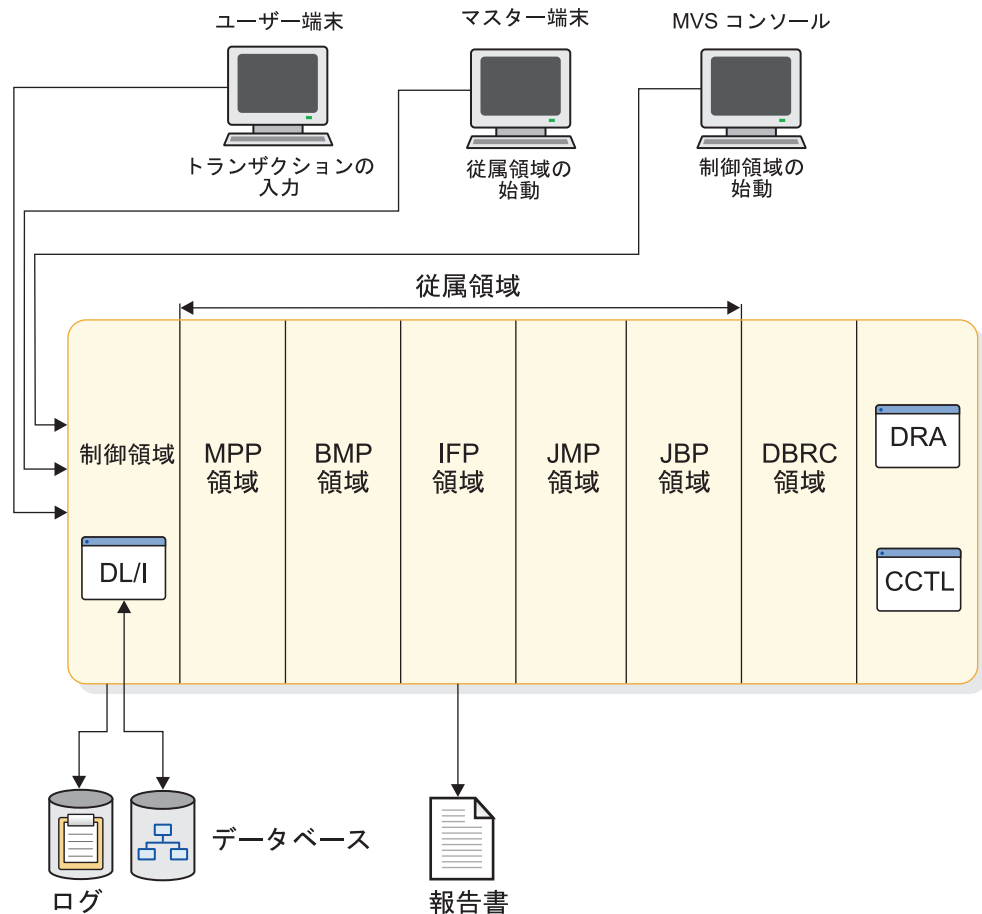


図 2. DB/DC 環境の例

図では、DL/I が制御領域の一部として示されていますが、DL/I をここで実行する必要はありません。DL/I は独自のアドレス・スペースで実行できます。

DB/DC 環境およびその機能を理解するには、トランザクションを理解することが重要です。DB/DC 環境での基本の作業単位はトランザクションです。トランザクション処理には次のものが含まれます。

- 端末で入力された作業要求の受信。要求はトランザクション・コードの形態をとり、実行する作業の種類とそのために必要なデータを識別します。
- 作業を実行するプログラムの呼び出しと、端末オペレーターへの応答 (例えば、実行される作業の肯定応答や照会への応答) の準備。
- 作業を要求した端末への応答の送信。

最も単純な種類のトランザクションには 2 つのメッセージが関与します。そのメッセージとは、端末ユーザーからの入力メッセージと、それに対する出力メッセージです。アプリケーション・プログラムは入力元以外の端末へメッセージを送信することもでき、トランザクションを生成できます。

関連概念:

487 ページの『DB/DC 環境および DCCTL 環境でのトランザクション・フロー』

制御領域

IMS 制御領域は制御プログラムを保持します。制御プログラムは制御領域アドレス・スペース内で常時実行されており、他の領域内での処理を制御します。

IMS 制御領域は、直接に、または DL/I 分離アドレス・スペース (DLISAS) を通じて、すべての DL/I 呼び出しにサービスします。IMS 制御領域はオンライン・アプリケーション・プログラムからアクセスできるすべてのデータベースを所有し、それらのデータベースへのすべての物理入出力を行います。IMS 制御領域はメッセージの処理を監視し、接続された端末のすべての通信トラフィックを監視します。IMS 制御領域はまた、再始動とリカバリーの目的で情報を管理し、IMS システム・ログを操作します。

IMS 制御領域は、通常、z/OS の START コマンドを使用して始動されます。IMS 制御領域は、その後、自動的に DBRC アドレス・スペースを開始します。

データベース・リカバリー管理機能

データベース・リカバリー管理機能 (DBRC) は、ログおよびデータベースのリカバリーを管理する上で役立ちます。この機能は、データベースを共用する IMS システムに対して、共用データベースへのアクセスを許可したり、阻止することによってデータ共用環境も管理します。

DBRC は固有のアドレス・スペースで実行されますが、IMS 制御領域に従属しています。DBRC は、すべてのオンライン IMS システムと、データ共用を使用するすべてのシステムで必要です。

関連概念:

619 ページの『第 30 章 DBRC の概要』

従属領域

従属領域は、制御領域とは異なるアドレス・スペースですが、IMS に依存しており、IMS は、この領域で、トランザクションを処理するアプリケーションをスケジュールします。従属領域は、z/OS の START コマンドまたは、IMS マスター端末からの /START REGION コマンドによって開始されます。

以下に、各種タイプの従属領域を示します。

- メッセージ処理プログラム (MPP) 領域。

MPP 領域はマスター端末オペレーターによって始動されるか、制御プログラムが実行されている場合は JCL によって始動されます。制御プログラムは MPP 領域内のアプリケーション・プログラムをスケジュールします。その後、アプリケーション・プログラムが稼働し、オンライン・データベースへアクセスし、メ

ッセージ・キューからトランザクション入力を入力します。アプリケーション・プログラムは、z/OS ファイルへアクセスしたり、z/OS チェックポイントを発行したりすることはできません。アプリケーション・プログラムの出力メッセージは、LTERM かその他のアプリケーション・プログラムへ宛先指定することができます。アプリケーション・プログラムは、MPP 領域が処理する作業がない場合でも、MPP 領域内にスケジュールした状態で残することができます。MPP 領域は、その領域が処理する作業がさらに発生するまで、待ち状態 (入力待ち状態) のままです。

- バッチ・メッセージ処理 (BMP) 領域。

z/OS は BMP 領域をスケジュールします。BMP 領域内のアプリケーション・プログラムは、制御領域によってではなく、各領域を始動するために使用される JCL によって判別されます。それらのアプリケーション・プログラムは制御領域が所有するデータベースと、BMP 領域が所有する z/OS データ・セットにアクセスできます。z/OS データ・セットには、高速処理データベース (DEDB) と主記憶データベース (MSDB) が含まれます。

BMP 領域内のアプリケーション・プログラムは入力および出力のメッセージ・キューにアクセスでき、入力待ちモードで実行することもできます。入力メッセージ・キューにアクセスするには、BMP 領域用の JCL でアクセスしたいトランザクション・コードを指定します。このトランザクション・コードを指定すると、アプリケーション・プログラムの仕様ブロック (PSB) 内にある端末プログラム連絡ブロック (PCB) を使用して出力メッセージ・キューにもアクセスできるようになります。入力メッセージ・キューへのアクセスがない場合でも、出力 LTERM を指定するか領域用の JCL でトランザクション・コードを指定すると、アプリケーション・プログラムから出力メッセージを発行できます。

- IMS 高速機能 (IFP) 領域。

IFP 領域では、次の 2 つのタイプのプログラムが実行されます。

- 高速機能のメッセージを処理するアプリケーション・プログラム。これらはメッセージ・ドリブン・プログラムと呼ばれます。
- DEDB を処理するユーティリティ。これらは BMP です。

- Java™ メッセージ処理 (JMP) 領域。

JMP 領域は、Java で書かれたアプリケーションか、または Java と OO COBOL の両方で書かれたアプリケーションでメッセージを処理します。JMP 領域では、31 ビットまたは 64 ビットの Java 仮想マシン (JVM) をロードすることができます。

- Java バッチ処理 (JBP) 領域。

JBP 領域は、Java で書かれたアプリケーションか、または Java と OO COBOL の両方で書かれたアプリケーションでバッチ操作を処理します。JBP 領域では、31 ビットまたは 64 ビットの Java 仮想マシン (JVM) をロードすることができます。

関連資料:

- 高速機能処理のタイプおよび高速機能が組み込まれた DB/DC 環境の管理の詳細については、「IMS V15 データベース管理」を参照してください。

- JMP および JBP 領域で実行できるアプリケーションの作成の詳細については、「IMS V15 アプリケーション・プログラミング」を参照してください。

マスター端末

マスター端末は、DB/DC 環境のコントロール・センターとなります。マスター端末オペレーター (MTO) はシステムのすべての操作局面を知っている必要があり、入力されるすべての IMS コマンドの目的と処置に精通している必要があります。

マスター端末の特性は次のとおりです。

- マスター端末は、システムの始動、停止、再始動を行うコマンドを入力するために使用されます。
- マスター端末は、1 つの論理端末としてシステム・メッセージを受信します。
- ネットワークおよび接続端末の基本制御は、マスター端末を通して行われます。マスター端末は通信回線の始動と停止を行うことができ、論理端末を物理端末の宛先へ割り当てることができます。
- システムの状況は、マスター端末から表示できます。例えば、処理するトランザクションの数、アクティブなプログラムとデータベースの数、通信回線の状況などの項目を要求できます。
- プログラムまたはデータベースのエラーが発生した場合、マスター端末からコマンドを入力することにより、影響を受けたりソースに対するそれ以上のプロセスを防止したり端末への入出力アクティビティを防止したりできます。これらのリカバリー処置は、異常終了が発生した後のシステム全体のリカバリーに適用することもできます。
- 例外条件が発生した場合は、マスター端末からオンライン・ログ・データ・セット (OLDS) の状況を表示でき、OLDS の機能を制御できます。

マスター端末が操作不能になった場合は、オペレーティング・システムのコンソールをバックアップとして使用できます。MTO は IMS をシステム・コンソールから操作することも、あるいはマスター端末 LTERM (論理端末) を代替の端末に割り当てることもできます。メッセージは、LTERM がシステム・コンソールへ割り当てられるまで、旧マスター端末 LTERM へ引き続き送られます。システム・コンソールのアドレスは LINE 1 PTERM 1 です。システム・コンソールを使用して操作を続行した場合、端末のセキュリティはマスター端末のセキュリティと同じです。

DB/DC 環境でサポートされるデータベース

DB/DC 環境では、すべての全機能データベース (HSAM、SHSAM、HISAM、SHISAM、HDAM、PHDAM、HIDAM、PHIDAM、および PSINDEX) をサポートします。

DB/DC 環境内の BMP 領域は、GSAM データベースにアクセスできます。BMP 領域は、DB/DC が外部サブシステム・インターフェースをサポートしているため、外部サブシステム (例えば Db2 for z/OS) にもアクセスできます。

高速機能の高速処理データベース (DEDB) および主記憶データベース (MSDB) もサポートされます。

注: PHDAM、PHIDAM、および PSINDEX は High Availability Large Database (HALDB) 用に追加されたデータベース・タイプです。これらのデータベース・タイプは、それぞれ HDAM、HIDAM、副次索引と同様に区画分割されています。

DB/DC での高速 DB リカバリー領域

高速 DB リカバリー領域は、別個の IMS 制御領域であり、IMS をモニターし、障害を検出し、障害の生じた IMS によりロックされた IMS 所有データベース・リソースをすべてリカバリーして、次の完全再始動を待たずにそれらを他の IMS システムで使用できるようにします。

Db2 for z/OS データベースなど、IMS 所有でないデータベース・リソースの場合、高速 DB リカバリー領域は、この目的のためにオプションとして ESAF 未確定通知出口ルーチン (DFSFDN0) を提供します。

高速 DB リカバリー領域は、IMS システム定義により提供されるカタログ式プロシージャによって実行されます。高速 DB リカバリー領域は、トラッキングする IMS を開始した後で開始する必要があります。

高速 DB リカバリーに対して DB/DC IMS を使用可能にするには、IMS プロシージャに FDRMBR パラメーターを指定します。FDRMBR パラメーターは、DB/DC システムを高速 DB リカバリー可能として定義します。

DB/DC 環境でのデータ共有

データは、従属領域やその他の IMS システムとの間で共有できます。その他のシステムとは、DB/DC または DBCTL です。

データをブロック・レベルで共有したい場合は、共有に関与するすべての環境に内部リソース・ロック・マネージャー (IRLM) が存在していなければなりません。IRLM は独自のアドレス・スペース内で実行されます。

拡張回復機能の実行

拡張回復機能 (XRF) はいくつかのプログラムを組み合わせたもので、これにはエンド・ユーザーに高水準の IMS 可用性を提供する 2 つの DB/DC 環境が含まれます。

一方の環境はアクティブであり、アクティブ・システムと呼ばれます。もう一方の環境は最初の環境のプロセスを常時追跡しており、代替システムと呼ばれます。代替システムは、アクティブ・システムに障害が起きるか、計画的なテークオーバーが (例えば保守を行うために) 開始された場合、いつでも処理をテークオーバーすることができます。

関連概念:

743 ページの『第 44 章 拡張回復機能の概要』

DBCTL 環境 (DBCTL environment)

DBCTL 環境は DB/DC 環境に似ており、DL/I 領域が処理対象のデータベースを所有します。DL/I は DBCTL 環境にも存在しますが、DL/I は独自のアドレス・スペースの中で実行する必要があります。DBCTL に必要なデータベース・リカバリー管理 (DBRC) 機能は、データベースの可用性、データ共有、システム・ロギング、およびデータベースのリカバリーを管理するのに役立ちます。

DBCTL と DB/DC の最大の相違点は、DBCTL がユーザー端末、マスター端末、およびメッセージ処理をサポートしないことです。したがって、MPP 領域は存在しません。BMP 領域は、バッチ・アプリケーションとユーティリティーだけが使用します。ただし、外部プログラム・サブシステムはメッセージを処理するインターフェース、つまりコーディネーター・コントローラ (CCTL) を使用できます。これと同じインターフェースが DB/DC 制御領域にも存在するため、DB/DC 環境で CCTL を使用することも可能です。CCTL と制御領域との間のインターフェースは、データベース・リソース・アダプター (DRA) です。DRA は CCTL と同じアドレス・スペースに常駐します。

CCTL は、メッセージ・トラフィックの処理と、アプリケーション・プログラムのスケジューリングをすべて DBCTL 環境の外部で行います。CCTL はインターフェースを介してデータベース呼び出しを制御領域に渡し、制御領域はそれらの呼び出しを DL/I に送り、インターフェースを介して結果を CCTL に返します。

本書で、IMS オンライン・システムについて述べているトピックは、DB/DC および DBCTL の両方に適用されます。例外については、DBCTL には適用されない旨が表記されています。

以下の図は、DBCTL 環境の例を示しています。

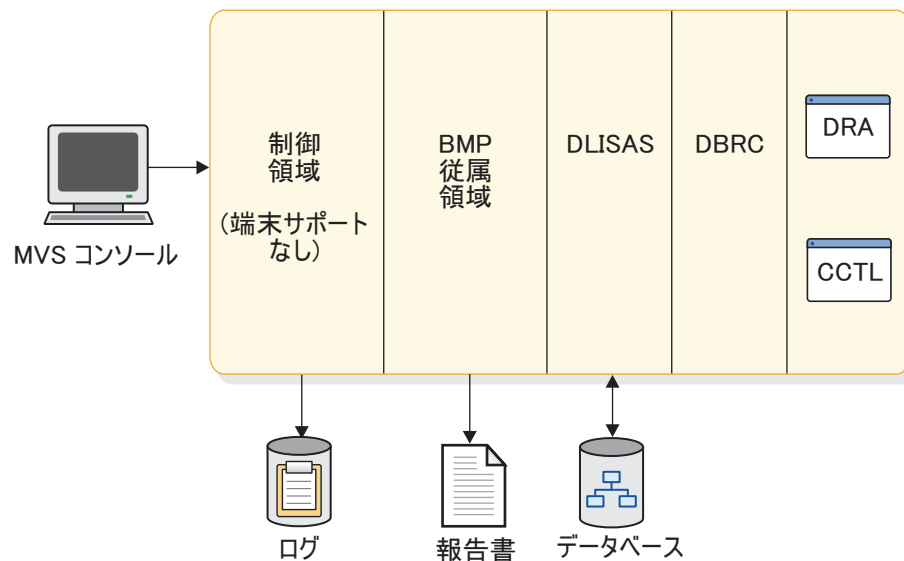


図 3. DBCTL 環境の例

DBCTL 環境にマスター端末が含まれていない場合でも、IMS コマンドを使用して環境を制御できます。その場合、メッセージ処理を制御するコマンドおよびコマンド機能は操作できませんが、それ以外は操作できます。それらのコマンドおよびコマンド機能は、MVS™ コンソールか 2 次コンソールを通して入力できます。制御領域は、コマンドを先頭文字のスラッシュ (/) によって認識します。システム定義の過程で、または実行パラメーターとして、異なる文字を最初の文字として選択できます。

注: 本書に記載されているマスター端末オペレーター (MTO) とは、DB/DC の MTO または DBCTL オペレーターを示します。

コマンドからの出力メッセージは、コマンドを入力したコンソールへ送信されます。非送信請求出力を受信するために他のコンソールを指定することもできます。それらのコンソールは、IMS 生成マクロ IMSCTRL を使用して定義したカテゴリーに属するコンソールです。

DBCTL によってサポートされるデータベース

DBCTL 環境では、すべての全機能データベース (HSAM、SHSAM、HISAM、SHISAM、HDAM、PHDAM、HIDAM、PHIDAM、および PSINDEX) をサポートします。

DBCTL 環境内の BMP 領域は、GSAM データベースにアクセスできます。BMP 領域は、DBCTL が外部サブシステム・インターフェースをサポートしているため、外部サブシステム (例えば Db2 for z/OS) にもアクセスできます。

DBCTL 環境は、高速機能の高速処理データベース (DEDB) をサポートします。

注: PHDAM、PHIDAM、および PSINDEX は High Availability Large Database (HALDB) 用に追加されたデータベース・タイプです。これらのデータベース・タイプは、それぞれ HDAM、HIDAM、副次索引と同様に区画分割されています。

DBCTL での高速 DB リカバリー領域

高速 DB リカバリー領域は、別個の IMS 制御領域であり、IMS をモニターし、障害を検出し、障害の生じた IMS によりロックされた IMS 所有データベース・リソースをすべてリカバリーして、次の完全再始動を待たずにそれらを他の IMS システムで使用できるようにします。

Db2 for z/OS など、IMS 所有でないデータベース・リソースの場合、高速 DB リカバリー領域は、オプションの ESAF Indoubt Notification (未確定通知) 出口ルーチン (DFSFDN0) をこの目的で提供します。

高速 DB リカバリー領域は、IMS システム定義により提供されるカタログ式プロシージャによって実行されます。高速 DB リカバリー領域は、トラッキングする IMS を開始した後で開始する必要があります。

DBCTL サブシステムで高速 DB リカバリーを使用可能にするには、DBC プロシージャに FDRMBR パラメーターを指定します。この FDRMBR パラメーターは、DBCTL システムを高速 DB リカバリー可能として定義します。

データ共有

DB/DC 環境と同様に、従属領域間およびその他の IMS システムとの間でデータを共有できます。その他のシステムとは、DB/DC 環境または DBCTL 環境です。

データをブロック・レベルで共有したい場合は、共有に関与するすべての環境に IRLM が存在しなければなりません。

代替 DBCTL 環境

DBCTL 環境で XRF を持つことはできませんが、2 つの DBCTL 環境 (アクティブと代替) を実行でき、それによってシステム使用可能性を高めることができます。

ただし、代替 DBCTL 環境はアクティブ環境のプロセスを追跡しません。コンソール・オペレーターは代替システムをアクティブ環境にするため、代替システムに対して緊急時再始動コマンド (/ERESTART) を使用しなければなりません。

DCCTL 環境

DCCTL は IMS Transaction Manager のサブシステムですが、データベース・コンポーネントは持っていません。DCCTL 環境は DB/DC 環境によく似ています。主な相違点は、DCCTL 制御領域がデータベースを所有せず、DL/I データベース呼び出しをサービスしないことです。

DCCTL サブシステムでは IMS カタログをサポートしません。

DCCTL は、IMS 外部サブシステム接続機能 (ESAF) または Db2 for z/OS リカバリー可能リソース・マネージャー・サービス接続機能 (RRMS) と結合して、外部サブシステム (例えば Db2 for z/OS) に対して Transaction Manager 機能を提供します。DCCTL 環境では、トランザクション処理と端末管理は DB/DC 環境でのトランザクション処理と端末管理と同じです。DCCTL には次のものに必要なプログラミング・サポートが含まれています。

- マスター端末サポート
- 端末ネットワーク・サポート
- データ通信
- メッセージ処理
- トランザクション処理
- アプリケーション・プログラムの実行
- IMS コマンドの実行

DCCTL はオンライン変更、メッセージ形式サービス (MFS)、複数システム結合機能 (MSC)、データベース・リカバリー管理 (DBRC) もサポートします。

DCCTL 環境では、データベース・コマンドとデータベース関連キーワードを除くすべての IMS コマンドがサポートされています。

DBRC は必須であり、再始動用のシステム・ログ情報を保持するために使用されます。DCCTL 環境での DBRC は、ログをトランザクション用にだけ保持します。外部データベース・サブシステムは、独自にデータベース・ログを保持しなければなりません。

DCCTL は次の 3 種類のアドレス・スペースからなっています。

- 制御領域
- DBRC
- 従属領域 (最大 999 個)

従属領域と DBRC は制御領域に従属します。

DCCTL 制御領域には、次の 3 つのコンポーネントが含まれています。

- データ通信マネージャー。これは端末の状態と入出力メッセージのトラフィックを制御します。また、このプログラムには DC リソースへの無許可アクセスを防止するセキュリティー管理機能も含まれます。

- メッセージ・マネージャー。これは、データ通信マネージャーからの端末入力と、Transaction Manager のスケジューリング・サービスとの間で読み書きと入出力のインターフェースとなります。
- Transaction Manager。これは MPP、BMP、および IFP を管理し、それらの従属領域に入っているアプリケーション・プログラムをスケジューリングし、アプリケーション・プログラミング・インターフェース (API) を所有し、API へ応答します。

それぞれのマネージャー (データ通信マネージャー、Transaction Manager、メッセージ・マネージャー) は各自のリソースの使用と、システムに障害が起きたときのリソースのリカバリーの可能性を制御します。DB/DC 従属領域と同様、MPP、BMP、IFP、JMP、および JBP 従属領域は、アプリケーション・プログラムをスケジューリングするために Transaction Manager によって使用されます。

以下の図は、外部サブシステムへ付加された DCCTL 環境を表しています。

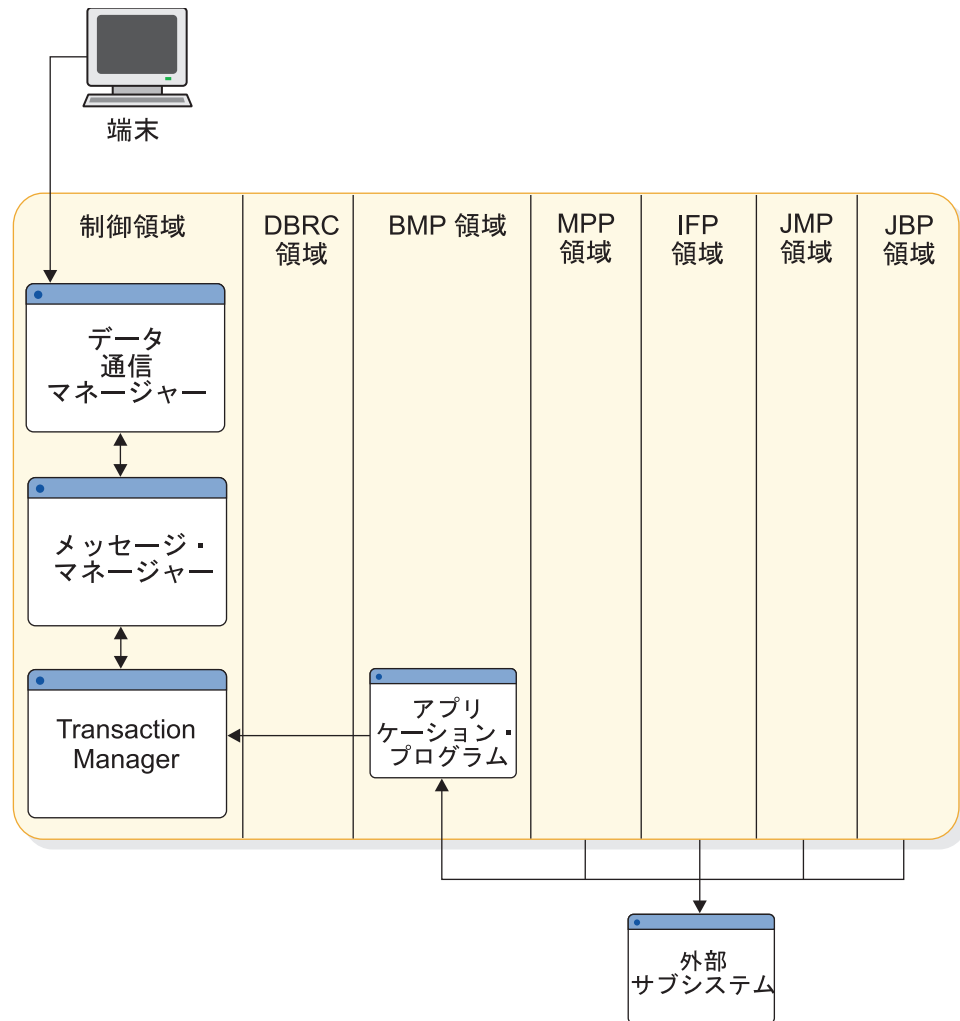


図 4. DCCTL 環境および付加されたサブシステムの例

DCCTL は同期点リカバリー・プロセスと、接続された外部サブシステムを調整します。DCCTL は、アプリケーションが同期点に到達した時点でデータベースの更新と端末メッセージがコミットされるようにします。

DCCTL 制御領域を別のサブシステムへ接続するには、2 つの方法があります。まず、DCCTL に制御領域 EXEC パラメーター SSM を使用させ、PROCLIB メンバーを選択できます。あるいは、/START SUBSYSTEM SSM コマンドを使用できます。このコマンドにより、DCCTL は、IMS 始動時にこのオプションが要求されなかった場合でも、別のサブシステムに接続することが可能になります。

従属領域 EXEC パラメーター SSM を従属領域用に指定することもできます。制御領域 SSM メンバー定義は、従属領域が 1 つ以上の外部サブシステム接続を選択できるようにします。SSM メンバーは、外部サブシステムへの接続を防止するために定義を含んでいなくても (ヌル・メンバーでも) かまいません。

/START SUBSYSTEM SSM コマンドを使用した後、従属領域が外部サブシステム接続を必要とする場合は、アクティブの従属領域をいったん停止してから再始動しなければなりません。

IMS が従属領域と外部サブシステムとの間に接続を確立した後、接続された領域間にスレッドが作成されます。このスレッドは、それ以後のアプリケーション・プログラム呼び出しやデータのコミットに使用され、障害が起きた状況ではデータのバックアウトに使用されます。

DCCTL によって管理されるアプリケーション・プログラムは、DB/DC 環境で DC マネージャーと TM マネージャーによって管理されるアプリケーションと同じです。

関連資料:

- ESAF および RRSAF インターフェースについて詳しくは、615 ページの『第 29 章 IMS 内から外部システムへのアクセス』を参照してください。
- /START SUBSYSTEM SSM コマンドの使用法について詳しくは、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

DCCTL によってサポートされるデータベース

IMS の DCCTL 構成は、データベースと従属領域の組み合わせをサポートし、これらの組み合わせでの互換通信のフロントエンドとなります。

以下のデータベースおよび従属領域の組み合わせは、IMS の DCCTL 構成用の互換通信のフロントエンドになります。

- BMP 領域用の GSAM データベース
- 以下の Db2 for z/OS データベース:
 - 外部サブシステム接続機能 (ESAF) を介した JMP、JBP、BMP、MPP、および IFP の各領域用
 - DB2[®] リカバリー可能リソース・サービス接続機能 (RRSAF) を介した JMP および JBP の各領域用

制約事項: DCCTL では、以下のデータベース・タイプをサポートしません。

- 高速機能データベース

- 全機能データベース
 - IMS カタログ、HALDB 全機能システム・データベース
- 全機能データベースまたは高速機能データベース。

GSAM データベースでは、DCCTL は順次の非 IMS データ・セットを BMP で使用します。アプリケーション・プログラムは、入出力 PCB を使用して、GSAM データ・セットに対し、シンボリック・チェックポイント (CHKP) 呼び出しと拡張再始動 (XRST) 呼び出しを発行することもできます。CHKP と XRST 呼び出しを発行する機能により、データ・セットの位置変更ができます。

DCCTL が ESAF または RRSF インターフェースを介して Db2 for z/OS データベースにアクセスしたとき、制御領域は、その他のサブシステムとの連絡を開始します。DCCTL がアクセスできるその他のサブシステムは、ユーザーが提供する IMS.PROCLIB データ・セットのサブシステム・メンバー (SSM) の中で定義されます。このサブシステム定義には、DCCTL が他のサブシステムと通信するために使用する情報が入っています。

関連資料:

- IMS から外部サブシステムにアクセスする方法については、615 ページの『第 29 章 IMS 内から外部システムへのアクセス』を参照してください。
- RRSF については、「DB2 for z/OS アプリケーション・プログラミングおよび SQL 解説書」を参照してください。

DCCTL は、外部サブシステムにアクセスする必要があるオンライン IMS アプリケーション用にトランザクション管理を実施します。

DCCTL がサポートするアプリケーション呼び出し

DCCTL へ渡されたアプリケーション・プログラム呼び出しは、その呼び出し機能がサポートされていないか、データベース PCB が呼び出しリストの一部として渡された場合、AD 状況コードを受け取ります。

DCCTL 環境では、アプリケーション・プログラムに次のデータ通信呼び出しが使用できます。

- AUTH
- CHNG
- CMD
- GCMD
- GN
- GU
- ICAL
- ISRT
- PURG
- SETO

DCCTL 環境では、アプリケーション・プログラムから次のシステム・サービス呼び出しが使用できます。

- APSB
- CHKP
- DPSB
- INIT
- INQY
- LOG
- ROLB
- ROLL
- ROLS
- SETS
- SETU
- SYNC
- XRST

DCCTL と DB/DC との比較

DCCTL と DB/DC 環境の TM 部分は非常によく似ています。

次のような類似点があります。

- 制御領域と従属領域の初期設定と終了
- システム定義とシステム生成
- 再始動
- データベース定義を除去しないステージ 1 入力。DCCTL システムを定義するにはいくつかの変更を行う必要があります。システム定義用のマクロ分析の詳細については、「IMS V15 システム定義」を参照してください。
- 診断

出口ルーチン

DCCTL 環境で他のサブシステム・リソースにアクセスする IMS 出口ルーチンまたは既存の IMS アプリケーション・プログラムに変更を加える必要はありません。しかし、他のサブシステムと IMS データベースにアクセスする呼び出しが混在しているアプリケーション・プログラムには、変更を加える必要があります。データベース PCB を使用するすべての DL/I 呼び出しでは、状況コード AD を受け取ります。

自動化操作プログラム・トランザクション

自動化操作プログラム・トランザクションは、IMS トランザクションの始動と同じ方法で始動されます。自動化操作プログラム・トランザクションは、DL/I 呼び出しを使用して DCCTL コマンドのサブセットを発行する権限を持つ IMS アプリケーション・プログラムとして実行されます。

DB バッチ環境

バッチ環境は、アプリケーション・プログラムと DL/I ルーチンが常駐するバッチ領域 (1 つのアドレス・スペース) から構成されます。ここで実行されるバッチ・ジョブは、オペレーティング・システム・ジョブと同様、JCL によって開始されます。

以下の図は、1 つのバッチ環境を表しています。この環境では、TSO 端末を介してバッチ・ジョブがサブミットされ、更新ファイルからの読み取り、データベースへの書き込み、および報告書の作成を行うためにアプリケーション・プログラムが実行されます。例えば、販売レコード (在庫減) と供給レコード (在庫増) を読み取り、それらに応じてデータベースを更新し、在庫販売報告書を印刷するような在庫管理アプリケーションが実行される場合があります。

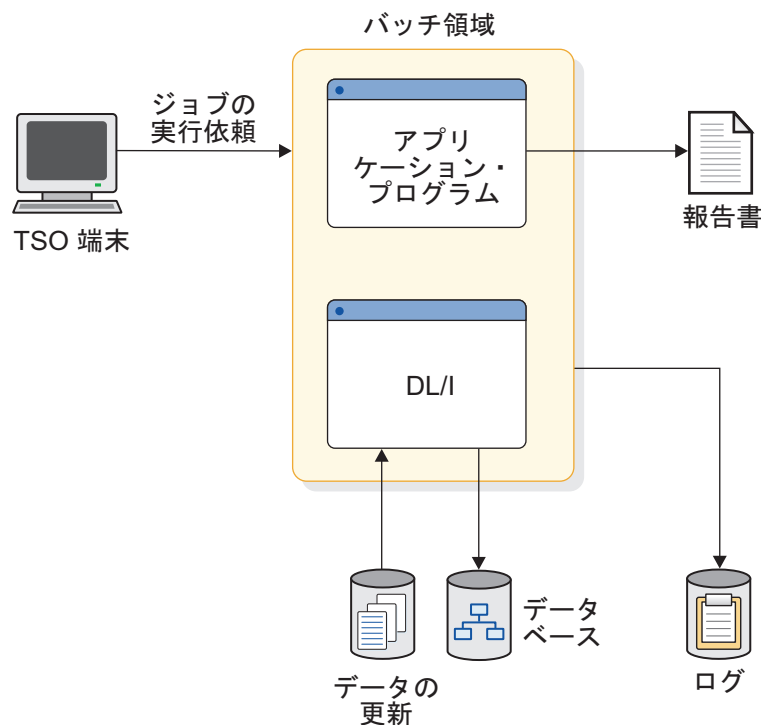


図 5. DB バッチ環境の例

TM バッチ環境

IMS TM は、アプリケーション・プログラムの実行用のバッチ領域をサポートします。IMS アプリケーションは、バッチでは SQL 呼び出しを発行するのに ESAF を使用することはできません。このサポートは外部サブシステムによって提供されます。

IMS TM バッチ環境では、Db2 for z/OS を 2 つの方法のどちらかで接続できます。まず、TM バッチ領域実行 JCL で SSM パラメーターを使用し、MBR パラメーターでバッチ・プログラムの実際の名前を指定できます。あるいは、バッチ領域実行 JCL に DDITV02 DD ステートメントをコーディングし、MBR パラメーターで Db2 for z/OS モジュール名、DSNMTV01 を指定できます。

有効な TM バッチ領域タイプは、DBB、DLI、または UPB のいずれかです。その他のすべての領域タイプは、TM バッチ環境には適用できません。

TM バッチ環境用の生成プログラム仕様ブロック (GPSB) は、DBBBATCH プロシージャおよび DLIBATCH プロシージャの PSB パラメーターを使用して指定します。

Db2 for z/OS などの他の外部サブシステムとのログ同期を有効にするには、IEFRDER ログ DD ステートメントが必要です。

関連資料:

- 外部サブシステムが Db2 for z/OS である場合、バッチ・プログラムから SQL 呼び出しを発行できるようにするために必要なステップについては、「DB2 for z/OS アプリケーション・プログラミングおよび SQL 解説書」を参照してください。
- 追加オプションまたは追加要件については、「IMS V15 システム定義」を参照してください。

IMSplex の概要

IMSplex は、相互に連携する IMS および z/OS コンポーネントで構成されます。

IMSplex では、複数システム結合機能などの接続されたシステムおよび XRF などの複製されたシステムが使用されます。

IMSplex は、以下のように定義できます。

- リソースまたはメッセージ・キュー (あるいはその両方) および作業負荷を共有するために連携する IMS システムの 1 つの集合。
- 共通サービス層 (CSL) を使用して Resource Manager (RM) なしに Single Point of Control を持つ単一の IMS システム。この構成では、IMS タイプ 2 コマンドを使用できます。これらのコマンドは、自動化操作プログラム (AOP) により、Operations Manager (OM) API を介してのみ発行できます。オプションで、このタイプの IMSplex では、CSL Open Database Manager (ODBM) を使用することもできます。
- RM を含む CSL を使用する、単一の IMS システム。

IMSplex の説明と、Parallel Sysplex® (以降、シスプレックスと呼びます) の説明を比較してください。シスプレックスは、相互に連携する複数の z/OS イメージで、結合機能によって接続されます。1 つ以上の IMSplex システムを 1 つ以上の z/OS イメージ上に定義できます。ただし、シスプレックス内のすべての z/OS イメージ上に IMS インスタンスを定義する必要はありません。

データ共有の概念は、IMSplex、特にリソースおよび作業負荷を共有する IMSplex にとって重要です。

これにより、IMS バッチ・ジョブまたはオンライン・アプリケーション、またはその両方を IMSplex 内のどこでも実行することができます。また、このようなバッチ・ジョブやアプリケーションは、共有 IMS データベース内のすべてのデータにアクセスできます。業務上の要件に応じて、IMS の作業負荷を分散できます。データ共有を有効にした後、バッチおよびトランザクションの作業負荷を IMSplex 全体

に分散させる計画を作成し、その計画を実行する必要があります。その手法の 1 つに、共用キュー (SQ) を使用する方法があります。SQ を使用すると、IMSplex 内のあらゆる IMS システムに対して、ネットワークからトランザクションを入力でき、共用キュー・グループ内のあらゆる IMS システムからトランザクションを実行できます。ある IMS システムが使用不可でも、別のシステムがその作業を処理できます。

複数の IMS システムがリソースまたはメッセージ・キューを共用する場合、IMS 共通サービス層 (CSL) を使用することにより管理および操作を簡易化できます。CSL により、単一イメージとしての観点が提供されて、複雑さが軽減されます。CSL を使用すると、複数の IMS システムを単一システムのように管理できます。例えば、タイプ 1 コマンドを複数の MTO または Single Point of Control (SPOC) から入力してローカル・オンライン変更を行う代わりに、1 つの SPOC からタイプ 2 コマンドを入力してグローバル・オンライン変更を実行できます。以下の図では、CSL を使用する場合と使用しない場合の IMSplex 管理の相対的な複雑さを比較しています。

CSL を使用した IMSplex 管理には、以下の利点があります。

- 改善されたシステム管理
- 単一システム・イメージ
- 単一制御点から行われる一層容易な操作
- IMS システム全体にわたって行われるリソースの共用

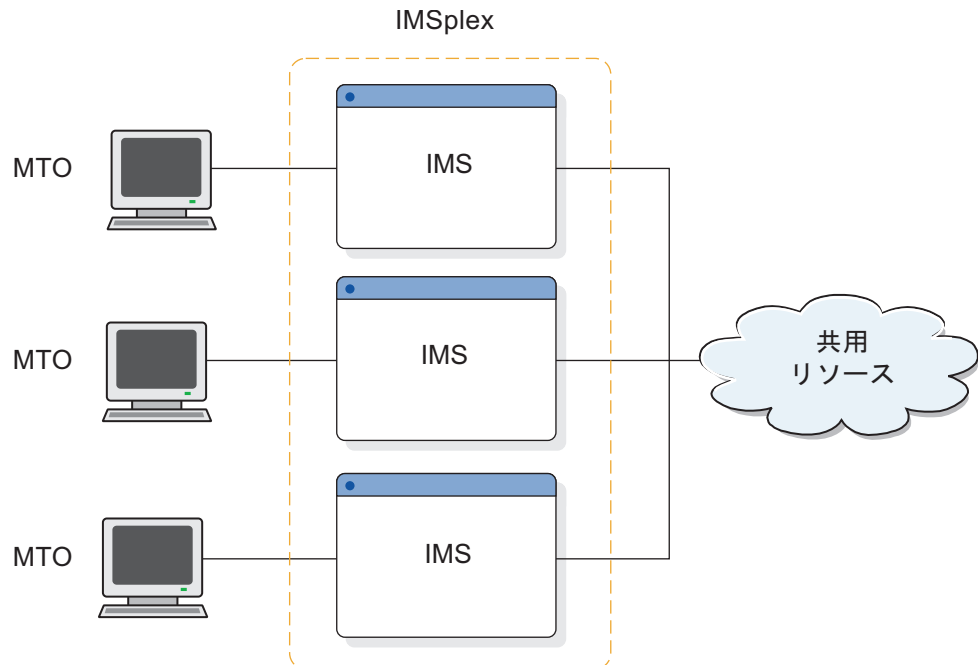


図 6. CSL を持たない IMSplex での複雑な管理

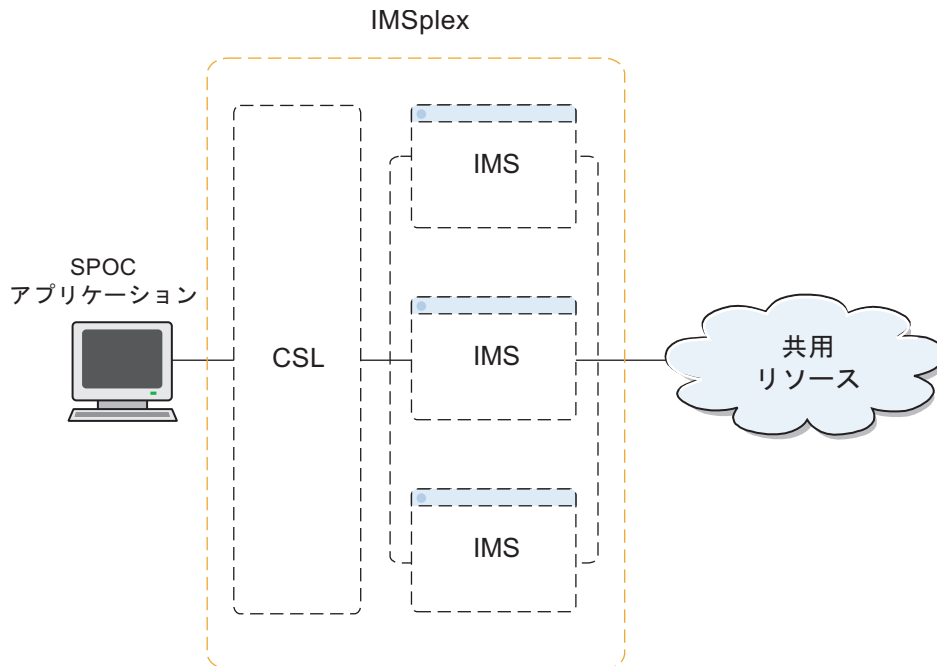


図 7. CSL を備えた *IMSplex* での単純化された管理

CSL により *IMSplex* を管理すると、結果として、環境は複数の *IMSplex* コンポーネント により構成されます。*IMSplex* コンポーネントは、*IMSplex* コンポーネント間でのリソースの管理、操作の管理、または通信の円滑化のいずれかを行います。*IMSplex* コンポーネントを以下に示します。

- Operations Manager (OM)
- Resource Manager (RM)
- Open Database Manager (ODBM)
- 構造化呼び出しインターフェース (SCI) (Structured Call Interface (SCI))
- IMS Connect
- データベース・リカバリー管理 (DBRC)
- Common Queue Server (CQS)
- リポジトリ・サーバー (RS) (Repository Server (RS))
- TSO 単一制御点 (SPOC)
- 自動プログラム用の REXX SPOC API
- オンライン制御領域

SCI に登録されたプログラムは、すべて *IMSplex* コンポーネントと見なされます。ユーザーやベンダーも、SCI に登録されるプログラムを作成することができます。それらのプログラムも *IMSplex* コンポーネントと見なされます。バッチ領域や従属領域 (MPR、BMP、および IFP) は *IMSplex* コンポーネントではありません。バッチ領域内で実行される DBRC コードは SCI に登録されますが、バッチ領域自体はコンポーネントではありません。*IMSplex* コンポーネントは、初期設定されて *IMSplex* に結合されると、*IMSplex* メンバー になります。

関連概念:

5 ページの『IMS の環境および構成』

IMSplex 内で使用される z/OS コンポーネントおよびシステム・サービス

IMSplex 内で使用される z/OS コンポーネントには、カップリング・ファシリティーとそのキャッシュ、リスト、およびロック構造が含まれます。

IMSplex は、以下のような z/OS システム・サービスを利用します。

z/OS システム間カップリング・ファシリティー

IMSplex は、通信サービスに XCF を使用します。XCF サービスは、あるシステム上の権限のあるプログラムが、同じシステム上の、または他のシステム上のプログラムと通信することを可能にします。

自動再始動管理 (ARM)

IMSplex は、リカバリー・サービスに ARM を使用します。ARM によって再始動されるのは、サブシステム、z/OS 障害後の別の z/OS イメージ上の作業負荷のすべての要素、および障害が起きた z/OS イメージです。

システム間拡張サービス (XES)

IMSplex は、データ共有サービスおよびシステム管理の再ビルド・サポートに XES を使用します。

IMSplex の一部である IMS コンポーネントおよびシステム・サービス

IMSplex には、いくつかの異なる IMS コンポーネント (IMS サブシステム、DBRC および RECON データ・セット、IMS Connect、および TSO SPOC) を含めることができます。IMS は、ロック・サービスに IRLM を使用します。

さらに、多くの IMS アドレス・スペースにより、IMSplex 内でシステム・サービスが提供されます。これらのアドレス・スペースは、すべて Base Primitive Environment (BPE) を使って作成されており、その BPE はトレース、メッセージ・フォーマット設定、構文解析、ストレージ管理、サブディスパッチング、およびシリアルライゼーションなどのシステム・サービスを提供します。

BPE システム・サービスは、共通キュー・サーバー (CQS)、ならびに CSL の Open Database Manager (ODBM)、Operations Manager (OM)、Resource Manager (RM)、Structured Call Interface (SCI)、Repository Server (RS)、およびオプションとして、データベース・リカバリー管理機能 (DBRC) により使用されますが、これらはそれぞれ他のシステム・サービスの役に立っています。

Common Queue Server

共用キューからのデータ・オブジェクトを受信、保守、および配布します。

共通サービス層 (CSL)

以下のコンポーネントを含みます。

Open Database Manager

IMSplex 内の DBCTL 環境および DB/TM 環境の IMS DB システムにより管理される IMS データベースにアクセスできるようにします。ODBM は、IMS Connect を介して TCP/IP クライアントをサポートし、さらに、Db2 for z/OS や WebSphere® Application Server for z/OS などの、IMS ODBA インターフェー

スを使用するアプリケーション・プログラムを実行するアプリケーション・サーバーをサポートします。

Operations Manager

コマンドの送付、コマンド応答の収集、コマンド自動化のための API の提供、およびカスタマイズとセキュリティーのためのユーザー出口の提供を行います。

Resource Manager

グローバル・リソース情報の保守、リソース整合性の確保、グローバル・オンライン変更などの IMSplex 全体のプロセスの調整を行います。

さらに、RM は、IMSRSC リポジトリのインターフェースおよびアクセスも提供します。

構造化呼び出しインターフェース


メッセージの送付、IMSplex のメンバーの登録および登録解除、IMSplex メンバーのセキュリティー認証などを行います。

Repository Server

リソースおよび記述子の定義を保管するために使用できるデータ・ストレージ機構である、IMSRSC リポジトリを管理します。

IMS も、ロック・サービスに IRLM を使用します。

関連概念:

 [IMSRSC リポジトリの概要 \(システム定義\)](#)

IMSplex 内の複数システム結合機能

複数システム結合機能 (MSC) を使用すると、複数システムに作業負荷を分散してバランスを取ることができます。

MSC は、IMS サブシステム間のトランザクション・トラフィックおよびメッセージ・トラフィックを分散します。MSC では IMS との通信機能およびスケジューリング機能が拡張されており、トランザクションが実際には複数の IMS サブシステム間で経路指定されていても、1 つの仮想 IMS システムとして認識できます。

MSC ネットワークは、共用キューを使用する IMSplex と共存可能です。MSC と IMSplex の一時的な共存は、MSC ネットワークから IMSplex 構成へマイグレーションする場合に役立ちます。また、MSC リンクが IMSplex を IMSplex 外の IMS システムに接続した時などの永続的な共存もあります。

どちらの共存例においても、MSC 環境と IMSplex 環境の両方にわたるトランザクション・メッセージの適正な経路指定と処理を考慮する必要があります。これは、それぞれの環境で異なる経路指定方式が使用されるからです。通常、MSC ネットワークは SYSID を使用して、トランザクションを特定の IMS システムに経路指定します。共用キューを使用する IMSplex システムは、トランザクションを共用キュー上でトランザクションに関するインタレストを登録した任意の IMS システムに対して使用可能にすることにより、トランザクションを経路指定します。MSC ネットワークと共用キューを使用する IMSplex が共存する場合、これらの経路指定方式は両方ともトランザクションに適用されます。

MSC システムと IMSplex システムの共存の詳細については、「IMS V15 コミュニケーションおよびコネクション」を参照してください。

IMSplex でのシステムのリカバリー

IMS システムのリカバリーは、拡張回復機能 (XRF) により管理できます。さらに、IMS 高速データベース・リカバリー (FDBR) 領域もリカバリー用に使用できます。

XRF システムごとにテークオーバーを行います。XRF は、アクティブ IMS システムと同じ物理データベースおよび多数の同じ物理リソースに依存します。

XRF は、ローカル・システム上で使用されるときと同じ方法で、IMSplex 内で使用することが可能です。XRF 代替システムは、IMSplex 内で固有の IMSID を持っており、IMSplex に定義されており、しかも CSL へのアクセスを持っている必要があります。SCI は、XRF 代替が常駐する z/OS イメージ上に常駐する必要があります。XRF を使用して全 IMSplex をリカバリーすることはできませんが、IMSplex 内の個別 IMS をリカバリーすることは可能です。

IMS Database Control (DBCTL) ウォーム・スタンバイ領域および IMS 高速データベース・リカバリー (FDBR) 領域もリカバリー用に使用できます。

FDBR FDBR 領域は、IMS サブシステムをモニターして、その IMS サブシステムが失敗した場合、自動的にデータベース・リソース (共用データベースおよびエリア) をリカバリーすることができます。

DBCTL スタンバイ

1 つの DBCTL サブシステムがアクティブの間、待機代替サブシステムとして別の DBCTL サブシステムを開始できます。この代替サブシステムは、アクティブ・サブシステムを (XRF 複合システムの中の代替サブシステム同様に) トラッキングしませんが、再始動コマンドを待つ完全に初期設定済みの IMS DBCTL です。

IMSplex には、XRF、FDBR、および DBCTL スタンバイ領域を含めることができます。

関連資料:

- XRF の詳細については、741 ページの『第 4 部 IMS システム・リカバリー』を参照してください。
- FDBR の詳細については、「IMS V15 オペレーションおよびオートメーション」を参照してください。

IMSplex の定義および調整

IMSplex の始動に必要な手順を確立し、IMS PROCLIB データ・セットの DFSCGxxx メンバーまたは DFSDFxxx メンバーの CSL セクションに適切なパラメーターを指定する必要があります。

以下の各トピックでは、CSL を備えた IMSplex に必要なそれぞれの IMSplex コンポーネントの初期設定方法について説明します。

これらのトピックでは、IMSplex 内に定義する必要があるコンポーネントについて簡単に説明します。

関連資料:

- IMSplex システム定義について詳しくは、「IMS V15 システム定義」を参照してください。
- IMS 制御領域の定義、システム定義マクロ、および PROCLIB メンバー・データ・セットについて詳しくは、「IMS V15 システム定義」を参照してください。

関連概念:

58 ページの『グローバル TM リソース管理』

関連タスク:

28 ページの『タイプ 2 コマンド環境のための簡易 IMSplex の定義』

CQS の定義:

CQS を定義するには、CQS および BPE PROCLIB メンバー・データ・セットを作成し、実行データ・セットを定義します。

以下のステップは、CQS の定義を要約したものです。

1. MSGQ、EMHQ、および RSRC 構造に対し、カップリング・ファシリティー・リソース管理 (CFRM) ポリシーを作成する。
2. 適用可能な z/OS ポリシーを定義する。
3. CFRM ポリシーを活動化する。
4. CQS および BPE PROCLIB メンバー・データ・セットを作成する。
5. すべての CQS 実行データ・セットを定義する。
6. CQS 環境をカスタマイズする。
7. CQS 構造への接続を許可する。
8. z/OS プログラム特性テーブルを更新する。

IMS 制御領域の定義:

IMSplex 内で機能するように IMS を初期設定するには、コマンド処理のために OM と、またリソース妥当性検査、状況更新、およびグローバル・オンライン変更のために RM と通信するように IMS 制御領域を定義する必要があります。

定義処理のステップを以下に示します。

- システム定義マクロで IMSID= キーワードを使用して、IMSplex 内の各 IMS システムを一意的に識別する。
- 各 IMS システムの DFSCGxxx PROCLIB メンバー・データ・セットで、または DFSDfxxx PROCLIB メンバー・データ・セットの CSL セクションで IMSPLEX= キーワードを使用して、各 IMS システムに対して IMSplex を識別する。

IMS システムと IMSplex を正しく定義するよう変更する必要がある PROCLIB メンバー・データ・セットには、次のものがあります。

- DFSCGxxx
- DFSDfxxx
- DFSPBxxx

- DFSSQxxx
- DFSVSMxx

ODBM の定義:

IMS.PROCLIB データ・セットの CSLDIxxx メンバーを使用して、ODBM を初期化するためのパラメーターを指定します。CSLDIxxx の一部のパラメーターは、ODBM 実行パラメーターでオーバーライドすることができます。

IMSPLEX(NAME=*plxnm*) パラメーターを使用して CSLDIxxx に IMSplex 名を指定する場合、PROCLIB データ・セットのメンバー CSLOIxxx、CSLRIxxx、CSLSIxxx、DFSDFxxx、および DFSCGxxx に指定されたものと同じ IMSplex 名を指定する必要があります。

CSLDCxxx PROCLIB メンバーを使用して、IMS データ・ストアへの ODBM 接続を構成するパラメーターを指定します。

OM の定義:

IMS.PROCLIB データ・セットの CSLOIxxx メンバーを使用して、OM を定義するためのパラメーターを指定します。CSLOIxxx の一部のパラメーターは、OM 実行パラメーターでオーバーライドすることができます。

IMSPLEX(NAME=*plxnm*) パラメーターを使用して CSLOIxxx に IMSplex 名を指定する場合、CSLDIxxx、CSLRIxxx、CSLSIxxx、DFSDFxxx、および DFSCGxxx PROCLIB メンバーに指定されたものと同じ IMSplex 名を指定する必要があります。

RM の定義:

IMS.PROCLIB データ・セットの CSLRIxxx メンバーを使用して、RM を定義するためのパラメーターを指定します。CSLRIxxx の一部のパラメーターは、RM 実行パラメーターでオーバーライドすることができます。

IMSPLEX(NAME=*plxnm*) パラメーターを使用して CSLRIxxx に IMSplex 名を指定する場合、CSLDIxxx、CSLOIxxx、CSLSIxxx、DFSDFxxx、および DFSCGxxx PROCLIB メンバーに指定されたものと同じ IMSplex 名を指定する必要があります。

SCI の定義:

IMS.PROCLIB データ・セットの CSLSIxxx メンバーを使用して、SCI アドレス・スペースを定義するためのパラメーターを指定します。CSLSIxxx の一部のパラメーターは、SCI 実行パラメーターでオーバーライドすることができます。

IMSPLEX(NAME=*plxnm*) パラメーターを使用して CSLSIxxx に IMSplex 名を指定する場合、CSLDIxxx、CSLOIxxx、CSLRIxxx、DFSDFxxx、および DFSCGxxx PROCLIB メンバーに指定されたものと同じ IMSplex 名を指定する必要があります。

タイプ 2 コマンド環境のための簡易 IMSplex の定義:

RM を使用せずに簡易 IMSplex を定義するには、IMSplex の定義時に、IMS PROCLIB データ・セットの DFSCGxxx メンバーまたは DFSDFxxx PROCLIB メンバーの CSL セクションに、RMENV=N を指定します。

RM なしで CSL を起動するには、IMS PROCLIB データ・セットの DFSCGxxx メンバーまたは DFSDFxxx メンバーの SCIPROC パラメーターに RMENV=N を指定する必要があります。

関連概念:

55 ページの『グローバル・オンライン変更』

47 ページの『簡易 CSL 構成』

関連タスク:

25 ページの『IMSplex の定義および調整』

➡ CSL OM の開始 (オペレーションおよびオートメーション)

➡ CSL SCI の開始 (オペレーションおよびオートメーション)

関連資料:

61 ページの『CSL 構成例』

➡ IMS PROCLIB データ・セットの DFSCGxxx メンバー (システム定義)

➡ IMS PROCLIB データ・セットの DFSDFxxx メンバー (システム定義)

RECON 損失通知用または **DBRC** 並列 **RECON** アクセス用の **IMSplex** 名の定義:

自動 RECON 損失通知または並列 RECON アクセスを使用可能にするには、CHANGE.RECON IMSPLEX コマンドを使って、RECON データ・セットの IMSplex 名を指定する必要があります。

自動 RECON 損失通知の使用が、IMSplex での CSL の唯一の使用となる場合、必要な CSL アドレス・スペースは SCI のみです。

この RECON にアクセスするすべての DBRC インスタンスは、DBRC SCI 登録出口ルーチン (DSPSCIX0) または EXEC ステートメントのキーワード・パラメーター IMSPLEX= を使って、RECON データ・セットの IMSplex 名を指定する必要があります。SCI を使用するには、DBRC では、DBRC インスタンスが実行される IMSplex の名前を与える必要があります。

DBRC SCI 登録出口ルーチン (DSPSCIX0) は、SCI への登録に先だって DBRC が呼び出す出口ルーチンです。並列 RECON アクセスを使用可能にしている場合、この出口ルーチンには SCI 登録に必要な IMSplex 名を提供する必要があります。この出口ルーチンを使用しなければ、DBRC は、サンプル版の出口ルーチンが使用された場合のように継続されます。つまり、DBRC が SCI に登録を行わないこととなります。DSPSCIX0 は、許可ライブラリーまたは LINKLST に入れておく必要があります。TSO 環境では、DSPSCIX0 を格納するライブラリーは TSO TASKLIB ライブラリーであることが必要です。

IMSplex に複数の RECON セットが定義されている場合、*group_id* も DSPSCIX0 出口ルーチンまたは RECON データ・セットに定義する必要があります。*group_id* の EXEC ステートメントのキーワード・パラメーターは、DBRCGRP=xxxx です。RECON データ・セットに *group_id* を定義するには、CHANGE.RECON IMSPLEX(*imsplex_name*,*group_id*) コマンドを使用します。

グローバル・オンライン変更の定義:

グローバル・オンライン変更を CSL で定義されている IMSplex 内で使用可能にするには、IMS PROCLIB データ・セットの DFSCGxxx メンバーまたは DFSDFxxx メンバーの CSL セクションを指定する必要があります。

DFSCGxxx PROCLIB メンバーおよび DFSDFxxx PROCLIB メンバーの CSL セクションには、IMSplex 内のすべての IMS システム (RM なしで定義されたシステムを除く) に共通の、SCI、OM、および RM に関連したパラメーターが含まれています。OLC= パラメーターでは、グローバル・オンライン変更を表す GLOBAL か、またはローカル・オンライン変更を表す LOCAL を指定する必要があります。

OLC=GLOBAL は、オンライン変更が IMSplex 全体にわたって調整されることを意味します。グローバル・オンライン変更が、INITIATE OLC コマンドにより準備およびコミットされます。グローバル・オンライン変更が使用可能にされており、リソース構造が定義されている場合には、MODBLKS、FORMAT、および ACBLIB の各データ・セットは IMSplex 全体で整合するはずですが、ただし、リソース整合性検査が NORSCCC キーワードによって省略されている場合は別です。

OLC=LOCAL は、オンライン変更がそれぞれの IMS にローカルで適用されることを意味します。ローカル・オンライン変更は、それぞれのローカル IMS で、/MODIFY PREPARE コマンドおよび /MODIFY COMMIT コマンドを使用して、準備されてコミットされます。ローカル・オンライン変更は、IMSplex 全体では手動で調整する必要があります。

CSL を備えた IMSplex に対しグローバル・オンライン変更が使用可能とされている場合、OLCSTAT データ・セットを DFSCGxxx PROCLIB メンバー・データ・セットの OLCSTAT= パラメーターまたは DFSDFxxx PROCLIB メンバー・データ・セットの CSL セクションを使用して指定します。次のことを行うには、グローバル・オンライン変更ユーティリティ (DFSUOLC0) を実行します。

- IMS が初めてコールド・スタートする前に、OLCSTAT データ・セットを初期設定する。
- OLCSTAT データ・セットがエラーにより使用できなくなった後に、OLCSTAT データ・セットを再作成する。
- OLCSTAT データ・セットをアンロックする。

OLCSTAT データ・セット:

z/OS が IMS に割り当てる、グローバル OLCSTAT データ・セットは、可変サイズのレコードをサポートする基本順次アクセス方式 (BSAM) データ・セットです。OLCSTAT データ・セットは、動的な割り振りおよび割り振り解除が可能です。MODSTAT データ・セットとは異なり、OLCSTAT データ・セットは、IMS システムの稼働中に常に使用可能にする必要はありません。

OLCSTAT データ・セットの属性

OLCSTAT の割り振りの前に、データ・セット属性を定義しておく必要があります。以下の表は、OLCSTAT データ・セットの推奨属性を示します。最大 65 個の IMS レコードを使用できます。

表 2. OLCSTAT データ・セットの推奨属性

データ・セット属性	値
DSORG	Sequential
RECFM	V
LRECL	5204
BLKSIZE	5208

これらの属性を定義しないと、IMS が OLCSTAT データ・セットの割り振りを試みたとき、その割り振りは失敗します。

これらの属性の定義後、DFSCGxxx PROCLIB データ・セット・メンバー内の OLCSTAT パラメーターを使用して、データ・セットに名前を付ける必要があります。

OLCSTAT データ・セット形式

OLCSTAT データ・セットは、ヘッダーと以下の形式を持つ任意の数の IMS レコードから構成されます。

- VERS=1 の OLCSTAT 形式

```
hlenhverivermstr OLCINP modifyid MODBLKSz x IMSACBb x FORMATc x
ilenrveriverims  prepare_timestamp commit_timestamp
```

- VERS=2 の OLCSTAT 形式

```
hlenhverivermstr OLCINP modifyid MODBLKSz x IMSACBb x FORMATc x
MOLCstate MOLCtoken MOLCid
ilenrveriverims  prepare_timestamp commit_timestamp
```

ヘッダーには、最後のオンライン変更フェーズのマスター、進行ロックでのオンライン変更、変更 ID、アクティブ MODBLKS DD 名、アクティブ IMSACB DD 名、アクティブ FORMAT DD 名、および最後のオンライン変更で変更されたライブラリーなどの情報が含まれます。ヘッダーは、グローバル・オンライン変更ユーティリティ (DFSUOLC0) によって初期設定されます。

hlen OLCSTAT データ・セットのヘッダー・レコード長。ヘッダー・レコードは最初の行です。

hver OLCSTAT データ・セットのヘッダー・バージョン。

iver IMS バージョン。

mstr 最後のオンライン変更フェーズ (準備、コミット、またはアボート) のマスターであった IMS システムの IMS ID。

ilen OLCSTAT データ・セットの IMS レコード長。OLCSTAT データ・セットには、ゼロまたは 1 つ以上の OLCSTAT データ・セットの IMS レコードが含まれます。

rver OLCSTAT データ・セットの IMS レコード・バージョン。

iver IMS バージョン。

ims IMS ID。

OLCSTAT データ・セット・ヘッダー

以下の例に、バージョン 2 用のサンプル OLCSTAT データ・セット・ヘッダーを示します。

```
RECORD SEQUENCE NUMBER - 1
000000 00000080 00000002 F1F0F1F0 40404040 40404040 40404040 40404040
000020 F0F0F0F1 40D4D6C4 C2D3D2E2 C140D540 C9D4E2C1 C3C2C140 D540C6D6
000040 C140D540 D4D6D3C3 C3D4D7E3 40BF88D7 940BE91E 82402006 283F2321
000060 028D4040 40404040 40404040 40404040 40404040 40404040 40404E40
000080 00000050 00000001 F1F0F1F0 40404040 C9D4E2F1 40F2F0F0 F6F2F8F3
0000A0 F0F5F2F7 4060F0F7 7AF0F040 F2F0F0F6 F2F8F340 F2F3F2F1 F0F3F440
0000C0 F0F04040 40404040 40404040 4040404E
```

```
F0F0F0F0 *.....1010 0000*
D9D4C1E3 *0001 MODBLKSA N IMSACBA N FORMAT*
03465440 *A N MOLCCMPT ..P..Z.. ..... *
40404040 *.. + *
40F2F3F2 *...&....1010 IMS1 2006283 232*
60F0F77A *0527 -07:00 2006283 2321034 -07:*
          *00 + *
```

```
IDC0005I NUMBER OF RECORDS PROCESSED WAS 1
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION CODE WAS 0
IDCAMS SYSTEM SERVICES
```

TIME: 1

それぞれの IMS レコードは、オンライン変更ライブラリーに関する最新の IMS 情報を含んでいます。その IMS は、最後のオンライン変更に関与したか、または最後のオンライン変更以後にコールド・スタートされたものです。コールド・スタートされた IMS がまだ存在しない場合、OLCSTAT データ・セットには IMS レコードは含まれません。

IMS または IMSplex のオンライン変更の準備処理の 1 ステップとして、オンライン変更コピー・ユーティリティ (DFSUOCU0) を実行する必要があります。DFSUOCU0 ユーティリティは、新規の定義を含むソース・ライブラリーをターゲット・ライブラリーにコピーします。IMS システムのクローンが作られておらず、ライブラリーが共用されていないような IMSplex では、DFSUOCU0 ユーティリティを、IMSplex 内のすべての IMS システムで実行しなければならない場合があります。IMS システムのクローンが作られており、ライブラリーが共用されている IMSplex では、DFSUOCU0 ユーティリティを、最上位の IMS レベルにある 1 つの IMS.SDFSRESL データ・セットに対して一度だけ実行しなければならない場合があります。

DFSUOCU0 ユーティリティは、ステージング・ライブラリーの内容を、パラメーターで指定されたターゲット・ライブラリーにコピーすることができます。このユーティリティは、OLCSTAT データ・セットを使用して、ターゲット・ライブラリー値 G (ライブラリーのターゲットが、このユーティリティで決定されたアクティブ・ライブラリーであることを意味する) をサポートします。ターゲットは、IMSplex 内の IMS システムが現在使用していないライブラリーです。新規ターゲット・ライブラリー G を指定すると、DFSUOCU0 ユーティリティは OLCSTAT データ・セットを読み取ってターゲット・ライブラリーを決定します。

グローバル・オンライン変更コマンド INITIATE OLC PHASE(PREPARE) および INITIATE OLC PHASE(COMMIT) をこの順序で発行すると、非アクティブ・ライブラリーがアクティブ・ライブラリーに変更されます。

関連資料:

➡ グローバル・オンライン変更ユーティリティー (DFSUOLC0) (システム・ユーティリティー)

➡ オンライン変更コピー・ユーティリティー (DFSUOCU0) (システム・ユーティリティー)

IMSplex での動的リソース定義

IMSplex には、DRD を使用可能にしている IMS システムと DRD を使用可能にしていないその他のシステムを含めることができます。このような種類の IMSplex 環境を DRD 混在 IMSplex と呼びます。

DRD 混在 IMSplex で MODBLKS パラメーターを含むコマンドが発行されると、MODBLKS パラメーターは無視されます。

DRD 混在 IMSplex でのアプリケーション・プログラム追加方法を示すため、2 つのシナリオが提供されています。

シナリオ 1: DRD 混在 IMSplex にアプリケーション・プログラムを追加

この IMSplex では、IMS1 が DRD を使用可能にしており、IMS2 と IMS3 が DRD を使用可能にしていません。

1. ステージング ACBLIB に入れる PSB に対し ACBGEN を実行します。
2. MODBLKS 生成を実行して、IMS2 および IMS3 システムに対しステージング MODBLKS データ・セット内で APPLCTN プログラムを定義します。
3. IMS1 で CREATE PGM コマンドを発行するか、または (プログラムが BMP 従属領域か JBP 従属領域でスケジュールされるようにする場合は) プログラム作成ユーザー出口 (PGMCREAT) を使用して、アプリケーション・プログラムを動的に作成します。
4. ステージング MODBLKS データ・セットと ACBLIB を非アクティブな MODBLKS データ・セットと ACBLIB にコピーします。
5. 以下のコマンドを発行して、MODBLKS データ・セットおよび ACBLIB に対しグローバル・オンライン変更を行います。

```
INITIATE OLC PHASE(PREPARE) TYPE(ACBLIB,MODBLKS)
INITIATE OLC PHASE(COMMIT)
```

MODBLKS キーワードは IMS1 システムにより無視されます。ただし、IMSplex は、1 つのオンライン変更インスタンスによりプログラムを同期し、追加します。MODBLKS キーワードを含むコマンドはリジェクトされません。この場合、IMS1 の ACBLIB に対し別のオンライン変更が必要となります。

シナリオ 2: DBCTL IMS システムを含む DRD 混在 IMSplex 内にアプリケーション・プログラムとその MFS フォーマットを追加

IMS1 と DBCTL1 は DRD を使用可能にしています。IMS2 と DBCTL2 は DRD を使用可能にしていません。


1. ステージング ACBLIB に入れる PSB に対し ACBGEN を実行します。
2. MODBLKS 生成を実行して、IMS2 および DBCTL2 システムに対しステージング MODBLKS データ・セット内で APPLCTN プログラムを定義します。
3. MFS ユーティリティーを実行して、IMS1 および IMS2 システムに対するステージング FMTLIB に MFS フォーマットを作成します。
4. IMS1 システムおよび DBCTL1 システムで CREATE PGM コマンドを発行するか、または (プログラムが BMP 従属領域か JBP 従属領域でスケジューラされるようにする場合は) プログラム作成ユーザー出口 (PGMCREAT) を使用して、アプリケーション・プログラムを動的に作成します。
5. ステージング MODBLKS データ・セットと ACBLIB を非アクティブな MODBLKS データ・セットと ACBLIB にコピーします。
6. 以下のコマンドを発行して、MODBLKS、FMTLIB、および ACBLIB に対しグローバル・オンライン変更を行います。


```
INITIATE OLC PHASE(PREPARE) TYPE(ACBLIB,FMTLIB,MODBLKS)
INITIATE OLC PHASE(COMMIT)
```

MODBLKS キーワードは、IMS1 システムおよび DBCTL1 システムにより無視されます。FMTLIB キーワードは、DBCTL1 システムおよび DBCTL2 システムにより無視されます。ACBLIB キーワードは、4 つのシステムすべてで処理されます。これにより、1 つのオンライン変更インスタンスで IMSplex が同期され、アプリケーション・プログラムとその MFS フォーマットが追加されます。MODBLKS キーワードを含むコマンドはリジェクトされません。この場合、IMS2、DBCTL1、および DBCTL2 に対し追加のオンライン変更が必要となります。

関連資料: DRD で何ができるかについての説明は、「IMS V15 システム定義」を参照してください。

関連資料:

 CREATE PGM コマンド (コマンド)

 PGMCREAT ユーザー出口ルーチン・タイプ (出口ルーチン)

IMSplex セキュリティーの確立

IMS オンライン・システムを保護する方法は、IMSplex 内の IMS システムに対しても違いはありません。保護できるリソースやそれらのリソースを保護するために使用できる機能は似ています。スタンドアロン IMS システムで存在するタイプのセキュリティ問題は、IMSplex 内の IMS システムでも存在します。

IMS のセキュリティを実装する際には、次のような基本事項の決定が必要です。

- 保護が必要なリソースの決定。
- リソースへのアクセスを必要とするユーザーの決定。
- ユーザーが必要とするリソースへのアクセス・レベルの決定。

スタンドアロン IMS システムで行われるセキュリティ検査に加えて、CSL を備えた IMSplex は、追加のセキュリティ検査を実行します。

第 2 章 システム管理者のための概念

このトピックでは、IMS のリソース制御の中心となる概念について説明します。この情報の以降のトピックは、これらの概念を完全に理解していることを前提としています。

アプリケーション・プログラムのシステム・サポート

IMS の下でアプリケーション・プログラムを実行するには、制御ブロックを定義および生成して、IMS.DBDLIB、IMS.PSBLIB、および IMS.ACBLIB ライブラリーに配置する必要があります。

IMS の下でアプリケーション・プログラムを実行するには、以下の操作を行う必要があります。

- プログラム仕様ブロック (PSB) 生成ユーティリティーを使用して、PSB 生成によりプログラムとその論理端末および論理データ構造の使用を記述します。この PSB は、PSB ライブラリーという 1 つ以上の IMS システム・ライブラリーに維持されます。
- データベース記述子ブロック (DBD) を作成して、IMS データベースへのアクセス権限を取得します。DBD は、DBD ライブラリーというシステム・ライブラリーにアセンブルされます。
- PSB および DBD の制御ブロックを結合して、アプリケーション制御ブロック (ACB) という内部フォーマットに拡張します。アプリケーション制御ブロック保守ユーティリティーが、ACB を作成するために使用され、ACB は ACB ライブラリーに配置されます。

アプリケーション・プログラマーおよびデータベース管理者は、PSB、DBD、および ACB ライブラリー・データ・セットの命名規則を把握する必要があります。

システム管理者は、以下を決定する必要があります。

- ACB ライブラリーを JCL によって割り振るか、または動的に割り振るか (DFSMDA マクロを使用)

ACB ライブラリー・データ・セットの動的割り振りのために DFSMDA マクロの使用を開始するには、以下のステップを実行します。

1. ACBLIBA および ACBLIBB データ・セット用の DFSMDA メンバーを作成します。DFSMDA メンバーは、IMS STEPLIB 連結または IMSDALIB DD ステートメントで指定されたデータ・セットに配置できます。
 2. IMS および DL/I JCL プロシージャから、IMSACBA および IMSACBB DD ステートメントを除去します。
 3. IMS を停止してから、DFSMDA メンバーを使用して再始動します。
- ACB を 64 ビット・ストレージにロードするかどうか

実行時に、アプリケーション・プログラムで必要な ACB は、31 ビットの非常駐 ACB ストレージ・プールに存在する必要があります。スケジューリング時

に、IMS では、まず非常駐ストレージ・プールが検索されて、ACB の有無が確認されます。ACB がストレージ・プールに存在しない場合、IMS では、ACB ライブラリーまたは 64 ビットのストレージ・プールから 31 ビットの非常駐 ACB ストレージ・プールに ACB をロードできます。

ACB 用の 64 ビットのストレージ・プールを使用可能にするには、以下のステップを実行します。

1. ACB ライブラリーの非常駐 ACB メンバーの総数にこれらのメンバーのサイズを乗算して、64 ビットのストレージ・プールに割り振るストレージの容量を計算します。
2. DFSDFxxx PROCLIB メンバーの DATABASE セクションで、ACBIN64 パラメーターに 64 ビット ACB ストレージ・プールのサイズ (ギガバイト単位) を指定します。ほぼすべてのインストール済み環境では、64 ビット ACB ストレージ・プールに 1 または 2 ギガバイトを指定すれば十分です。
3. IMS を停止してから、ACBIN64 パラメーターを使用して再始動します。

64 ビット ACB ストレージ・プールを使用可能にした後、IMS では、ACB を必要とするアプリケーション・プログラムが初めてスケジュールに入れられたときに、このプールにデータが追加されます。それ以降は、この ACB を必要とするアプリケーション・プログラムがスケジュールに入れられたときに、IMS では、ACB ライブラリーの代わりに、この 64 ビットのストレージ・プールから ACB を読み取ります。

31 ビットの非常駐ストレージ・プールおよび 64 ビットのストレージ・プールの両方から ACB を除去する必要がある場合は、DELETE DB および DELETE PGM コマンドを使用します。その他のコマンドを使用して、これらのプールから ACB を除去することはできません。

関連資料: PSB、DBD、および ACB の生成のさらに詳細な概要については、「IMS V15 データベース管理」を参照してください。

関連資料:

756 ページの『XRF 要件』

IMS での動的割り振り

データ・セットが制御領域プロシージャの JCL を使用して指定された場合、それらのデータ・セットは制御領域の始動時に初期割り振りされます。一部のデータ・セットは、動的に割り振ることができます。

以下のデータ・セットを必要なときに動的に割り振り、使用しなくなったときには割り振り解除するよう設定することができます。

動的な割り振りと割り振り解除の対象となるデータ・セットを宣言するには、IMS マクロ DFSMDA を使用します。

- データベース・データ・セットの動的割り振りは、/START コマンドを使用して明示的に、またはアプリケーション・プログラムのスケジュール時に暗黙に行うことができます。データベース・データ・セットは、/DBRECOVERY コマンドを使用して割り振りを解除できます。

- DEDB エリア・データ・セットでは、暗黙の割り振りはアプリケーション・プログラムによる最初のアクセスで発生します。/STOP コマンドはそのデータ・セットの割り振り解除も行います。
- IMS モニター・データ・セットは、IMS モニターを /TRACE SET ON コマンドで始動したときに動的に割り振ることができ、/TRACE SET OFF コマンドで割り振りを解除できます。
- 再始動のための入力として必要な RECON データ・セット、オンライン・ログ・データ・セット (OLDS)、先行書き込みデータ・セット (WADS)、システム・ログ・データ・セット (SLDS) は、動的に割り振ることができます。
- ACB ライブラリーは、動的に割り振ることができます。
- 高可用性ラージ・データベース (HALDB) では、動的な割り振りの対象となるのは DBRC に登録されたデータ・セットだけです。割り振りでは、DFSMDA のメンバーを検索または処理しません。HALDB に割り振られた DD 名には、A から J、X、または L の文字が含まれ、その後に接尾部として 7 バイトの HALDB 区画名が付きます。

生成された HALDB 区画の DD 名と同じ名前を使用する割り振りがすでに存在する場合、そのデータ・セット名は IMS オンライン環境で DBRC に登録されたデータ・セット名と照合されます。データ・セット名が一致しなければ、割り振りは失敗します。

動的割り振りを使用するすべてのデータ・セットはカタログしなければなりません。ただし、IMS モニター・データ・セットだけは例外で、これはカタログしてはなりません。JCL で初期に割り振られたデータ・セットは、制御領域の実行中に動的に割り振り解除して再割り振りできます。

関連資料: IMS マクロ DFSMDA について詳しくは、「IMS V15 システム定義」を参照してください。

Base Primitive Environment (BPE) の概要

IMS Base Primitive Environment (BPE) は、他の多くの IMS コンポーネントが構築された、共通システム・サービス・ベースです。BPE は、トレース、メッセージ・フォーマット設定、構文解析、ストレージ管理、サブディスパッチング、およびシリアルライゼーションなどのサービスを提供します。

以下のIMS コンポーネントが BPE を使用しています。

- Common Queue Server (CQS)
- IMS Connect
- Open Database Manager (ODBM)
- Operations Manager (OM)
- Resource Manager (RM)
- リポジトリ・サーバー (RS) (Repository Server (RS))
- 構造化呼び出しインターフェース (SCI) (Structured Call Interface (SCI))

オプションとして、データベース・リカバリー管理機能 (DBRC) で BPE を使用できます。

BPE を使用する IMS コンポーネントが開始されたとき、コンポーネントは BPE サービス・モジュールのコピーを IMS プログラム・ライブラリーからアドレス・スペースにロードします。IMS コンポーネントのモジュールは、そのコンポーネントに固有です。ただし、BPE サービス・モジュールはさまざまなアドレス・スペースに対して共通です。したがって、基本システム・サービス機能は、BPE を使用するアドレス・スペースのそれぞれに対して同一です。次の図は、BPE、IMS コンポーネント、および IMS プログラム・ライブラリーの関係を示しています。

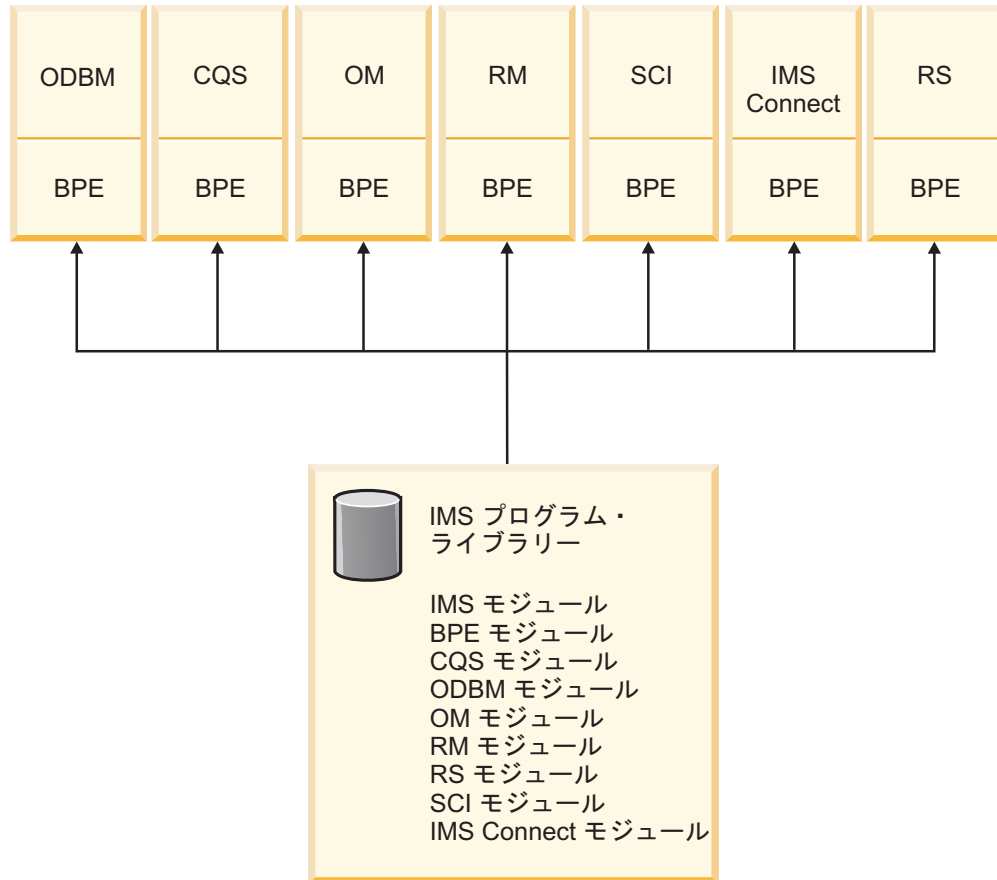


図 8. BPE および IMS コンポーネント

多くの場合、BPE は、IMS コンポーネントのアドレス・スペース内で隠れた層になっています。しかし、BPE に対して、以下の外部インターフェースを使用することができます。

構成 BPE PROCLIB メンバー・データ・セット内のステートメントを使用して、BPE を使用するアドレス・スペースに関する特定の属性を、起動時に構成することができます。例えば、BPE 管理トレース・テーブルに対してデフォルトのレベルおよびサイズを設定することができます。PROCLIB データ・セットのメンバー BPE を使用した BPE 定義については、「IMS V15 システム定義」で説明しています。

コマンド

BPE が提供する、BPE 管理リソースを操作するためのコマンドの小さな集合を使用することが可能です。例えば、BPE 管理ユーザー出口ルーチンおよびトレース・テーブルの属性を表示および変更することが可能です。BPE

コマンドについては、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」に説明があります。

ユーザー出口ルーチン

ユーザー出口ルーチンを作成して、IMS コンポーネント・アドレス・スペースの操作をカスタマイズすることが可能です。BPE と共に実行するコンポーネントは、BPE ユーザー提供の出口ルーチン・サービスを使用して、コンポーネントに固有のユーザー提供出口ルーチン呼び出すことができます。BPE も独自のユーザー出口ルーチンを持っています。BPE を介して呼び出されるユーザー出口ルーチンは、標準 BPE ユーザー出口ルーチン・インターフェースによって制御を受け取り、BPE ユーザー出口ルーチン呼び出し可能サービスの使用を許されます。BPE ユーザー出口については、「IMS V15 出口ルーチン」に説明があります。

メッセージおよび異常終了

BPE には、固有のメッセージおよび異常終了コードがあります。「IMS V15 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」に、BPE の異常終了コードとサービス戻りコード、および BPE メッセージの説明があります。

BPE コンポーネントのトレース

BPE コンポーネント (BPE、CQS、DBRC、IMS Connect、ODBM、OM、RM、または SCI) に対してトレースを使用可能にして、トレース情報をメモリーに書き込むか、またはメモリーと外部トレース・データ・セットの両方に書き込むかを制御できます。

このトピックでは、BPE コンポーネントに対するトレースの使用可能化に関連するタスクを説明します。

BPE トレースの使用可能化

BPE トレースを使用可能にして、IMS アドレス・スペースに対して常にオンにするには、IMS.PROCLIB データ・セットの BPE 構成パラメーター・メンバー (BPECFG= で指定) 内で TRCLEV ステートメントを定義または変更する必要があります。また、UPDATE TRACETABLE コマンドを使用して、BPE トレースを動的に開始することもできます。

BPE 構成パラメーター・メンバーは、開始されるアドレス・スペースに対する BPE 実行環境の設定を定義します。BPE トレース・レコードは、内部 (メモリー専用) トレース・テーブルおよび外部データ・セットに書き込むことができます。

デフォルトでは、内部トレース・テーブルに書き込みます (EXTERNAL=NO)。外部データ・セットに書き込むには、TRCLEV ステートメント上で EXTERNAL=YES と設定し、BPE 構成パラメーター・メンバーに EXTTRACE パラメーターを指定する必要があります。BPE がトレースする世代別データ・セット・グループ (GDG) も定義する必要があります。EXTERNAL=YES と指定すると、トレース・データは外部データ・セットおよび内部トレース・テーブル両方に書き込まれます。

アドレス・スペースの実行中に、BPE 構成の PROCLIB メンバー内の外部トレース・データ・セット指定を動的に変更し、メンバーをリフレッシュすることができます。例えば、外部トレース・データ・セットなしで実行中の場合に、BPE

PROCLIB メンバーを編集し、外部トレース・データ・セット指定を追加して、アドレス・スペースを再開することなく外部トレースの使用を開始できます。

関連資料:


 BPE UPDATE TRACETABLE コマンド (コマンド)

内部トレース・テーブルへの書き込み:

アドレス・スペースに対するデフォルトの BPE 実行環境の設定では、内部トレース・テーブルに書き込みます (EXTERNAL=NO)。

BPE トレース・レコードを内部トレース・テーブルに書き込むには、IMS.PROCLIB データ・セットの BPE 構成パラメーター・メンバー (BPECFG=) の TRCLEV パラメーターに、タイプ、レベル、および、オプションとして、トレース・テーブルに割り振られるストレージ・ページ数を指定します。

関連資料:

 IMS PROCLIB データ・セットの BPE 構成パラメーター・メンバー (システム定義)

外部データ・セットへの書き込み:

外部データ・セットに書き込むには、TRCLEV ステートメント上で EXTERNAL=YES と設定し、BPE 構成パラメーター・メンバーに EXTTRACE パラメーターを指定する必要があります。

BPE トレース・レコードを外部データ・セットに書き込むには、以下の手順を実行します。

1. 外部トレース・データ・セットの世代別データ・グループ (GDG) を定義します。
2. GDG モデルを定義します。
3. IMS.PROCLIB データ・セットの BPE 構成パラメーター・メンバー (BPECFG=) に EXTTRACE パラメーターを指定します。EXTTRACE ステートメントには、ステップ 1 で定義した GDG の名前を指定する必要があります。
4. BPE 外部トレース・データ・セットに書き込むトレース・テーブルを指定します。これは、以下の方法のうち的一方、または両方を使って行えます。
 - 外部化するトレース・テーブルに対する TRCLEV ステートメント上の他の TRCLEV オプションに EXTERNAL=YES パラメーターを含めます。TRCLEV ステートメントは、IMS PROCLIB データ・セットの BPE 構成パラメーター・メンバー (BPECFG=) 内にコーディングされます。

ヒント: TRCLEV ステートメントへの変更が処理されるのは、アドレス・スペースが開始されたときのみです。変更を行ったときにアドレス・スペースが実行中の場合は、アドレス・スペースが再開されるまで変更は効力を持ちません。

- NAME パラメーターに外部化するトレース・テーブルを指定し、かつ EXTERNAL=YES を指定した UPDATE TRACETABLE コマンドを発行します。このコマンドは、TRCLEV パラメーター指定に関係なく、アドレス・スペースの実行中に外部トレースを動的にオンにするために使用されます。

5. オプション: 外部データ・セットへのトレースを開始する際、アドレス・スペース開始時に EXTTRACE パラメーターを指定しなかった場合には、以下のようになります。
 - a. アドレス・スペースの実行中に、EXTTRACE データ・セットを IMS.PROCLIB データ・セットの BPE 構成メンバーに追加します。
 - b. OPTION(REREAD) オプションを指定した UPDATE TRACETABLE コマンドを発行して、BPE が新たに更新された EXTTRACE パラメーターを処理するようにします。

EXTTRACE と TRCLEV ステートメント、BPECFG=、および IMS.PROCLIB データ・セットについては、「IMS V15 システム定義」で説明されています。

外部トレース・データ・セットの定義:

GDG を定義するには、z/OS DEFINE GENERATIONDATAGROUP コマンドを使用します。このコマンド内の NAME キーワードで指定した値は、IMS PROCLIB データ・セットの BPE 構成パラメーター・メンバー (BPECFG=) の EXTTRACE ステートメントの GDGDEF パラメーター内のデータ・セット名と一致している必要があります。

IDCAMS ジョブからの **GDG** の定義

z/OS DEFINE GENERATIONDATAGROUP コマンドは、以下の例に示すように、IDCAMS ジョブから発行できます。

```
//DEFGDG JOB ...
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
DEFINE GENERATIONDATAGROUP -
(NAME(BPEEXTRC.GDG01) -
NOEMPTY -
SCRATCH -
LIMIT(255))
```

TSO セッションからの **GDG** の定義

z/OS DEFINE GENERATIONDATAGROUP コマンドは、以下の例に示すように、TSO セッションから発行できます。

```
DEFINE GENERATIONDATAGROUP (NAME(BPEEXTRC.GDG01) NOEMPTY SCRATCH LIMIT(255))
```

TSO バッチ・ジョブからの **GDG** の定義

z/OS DEFINE GENERATIONDATAGROUP コマンドは、以下の例に示すように、TSO コマンド・バッチ・ジョブから発行できます。

```
//DEFGDG JOB ...
//STEP1 EXEC PGM=IKJEFT01
//SYSSTPRINT DD SYSOUT=A
//SYSTSIN DD *
DEFINE GENERATIONDATAGROUP (NAME(BPEEXTRC.GDG01) NOEMPTY SCRATCH LIMIT(255))
```

DEFINE GENERATIONDATAGROUP コマンドについて詳しくは、「z/OS DFSMS カタログのためのアクセス方式サービス・プログラム」を参照してください。

トレース・データの追加情報については、「IMS Version 15 Diagnosis」を参照してください。

GDG モデル・データ・セットの定義:

GDG エントリーを作成する際に使用されるデータ制御ブロック (DCB) を提供する必要があります。これは、世代別データ・グループ (GDG) モデル・データ・セットの定義時に提供することができます。GDG モデル・データ・セットを定義するには、z/OS プログラム IEFBR14 を使用します。

IEFBR14 を使用した **GDG** モデル・データ・セットを定義するためのサンプル **JCL**

以下の例では、DCB 情報を含む、GDG モデル・データ・セットを定義するためのサンプル JCL を示します。

```
/* STEP1 - Define the GDG Base
//STEP2 EXEC PGM=IEFBR14
//BLDDSCB DD DSN=BPEEXTRC.GDG01,
//          DISP=(NEW,KEEP),
//          UNIT=SYSDA,
//          VOL=SER=PAGE01,
//          SPACE=(TRK,(0)),
//          DCB=(DSORG=PS,LRECL=32756,RECFM=VB,BLKSIZE=32760)
```

EXTTRACE ステートメント上の DSN パラメーターを使用して DCB 情報を受け渡すこともできます。ただし、データ・セット属性をオーバーライドすることはできません。

データ・セットを定義する際には、RECFM=VB とし、LRECL は BLKSIZE より 4 バイト小さくしなければなりません。サポートされる最小の BLKSIZE は 8340 です。

推奨事項: 大きなブロック・サイズ (最大値の 32760 など) を使用してください。BLKSIZE を大きくすることにより、1 つのブロックに複数のレコードが書き込まれるようになります。さらに、サポートしている最小ブロック・サイズが大きくなっても、BPE 外部トレース定義を変更する必要はありません。

IEFBR14 の詳細については、「z/OS MVS JCL 解説書」を参照してください。

BPE 外部トレースの開始および停止

BPE 外部トレースを開始するには、IMS.PROCLIB データ・セットの BPE 構成パラメーター・メンバー (BPECFG=) の TRCLEV= パラメーターで EXTERNAL=YES キーワードを指定するか、EXTERNAL(YES) パラメーターを指定した UPDATE TRACETABLE コマンドを発行する必要があります。

どのトレース・テーブルが外部トレースをオンにしたかを調べるには、BPE DISPLAY TRACETABLE コマンドを使用します。

外部トレース・データ・セットの 1 つに書き込み中に IMS で入出力エラーが起きた場合、IMS は現行データ・セットをクローズして割り振り解除を行い、新規データ・セットをオープンして割り振ります。IMS が新規データ・セットを割り振れない、またはオープンできない場合、IMS は GDG データ・セットを使用するすべて

のコンポーネントに対し BPE トレースを使用不可にします。IMSにデータ・セットの割り振りおよびオープン処理を再試行させるには、BPE UPDATE TRACETABLE コマンドを発行します。

GDG データ・セットの入出力容量では作成されているトレース・レコードの量に対応できない場合は、IMS は、待機して他の IMS 作業を遅らせるよりも、入出力システムが追いつくまでトレース・レコードの書き込みをスキップします。

特定のトレース・テーブルのタイプに対して BPE 外部トレースを停止するには、次のようにします。

1. UPD TRTAB EXTERNAL(NO) コマンドを発行します。
2. NAME パラメーターにテーブル・タイプを指定します。

全テーブルに対して BPE 外部トレースを停止するには、UPD TRTAB NAME(*) EXTERNAL(NO) を発行します。外部トレースされているテーブル・タイプがなくなると、BPE は最後のトレース・データ・セットをクローズし、割り振り解除します。

BPE コマンドについては、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」に説明があります。

BPE 外部トレース・データ・セットのフォーマット設定および表示

BPE トレース・エントリーのフォーマット設定および表示を行うには、対話式問題管理システム (IPCS) を使用します。IPCS を使用した BPE トレース・エントリーの表示については、「IMS Version 15 Diagnosis」を参照してください。

Common Service Layer の概要

IMS 共通サービス層 (CSL) は、IMSplex におけるシステム管理作業に必要なインフラストラクチャーを提供する IMS マネージャー・アドレス・スペースの集合です。CSL アドレス・スペースとしては、Open Database Manager (ODBM)、Operations Manager (OM)、Resource Manager (RM)、Structured Call Interface (SCI) があります。

CSL は、IMS 基本プリミティブ環境 (BPE) 層に構築されます。そのため、すべての BPE 外部要素 (コマンド、メッセージ、異常終了、構成、およびユーザー出口インターフェースなど) は、すべての CSL マネージャー・アドレス・スペースに適用されます。

IMS CSL は、以下のものを提供します。

- 改善されたシステム管理
- 単一システム・イメージ
- 単一制御点による使用の容易性
- 全 IMS システムにわたるリソースの共有

単一 CSL は、同一オペレーティング・システム上、または複数オペレーティング・システム上の IMSplex 内の複数の IMS システムをサービスすることができます。

推奨事項: それぞれのタイプの複数の CSL マネージャーを IMSplex 内で活動化してください。CSL で定義された IMSplex 内の IMS 制御領域は、IMSplex において少なくとも 1 つの OM がアクティブで、SCI が IMSplex のすべてのオペレーティング・システムに存在する場合を除き、始動できません。RM は IMS 制御領域の始動には必要ありませんが、この推奨は RM にもあてはまります。

それぞれのオペレーティング・システムに CSL マネージャーすべてを配置することには、利点もあれば欠点もあります。利点としては、システム・パフォーマンスがよくなるという点があります。IMSplex メンバーが同一オペレーティング・システム上の CSL マネージャーと通信する速度は、IMSplex メンバーが別のオペレーティング・システム上の CSL マネージャーと通信する場合より速くなります。欠点としては、それぞれのオペレーティング・システムでアドレス・スペースの数が増加するという点があります。しかし、IMSplex 内に定義されている CSL マネージャーのインスタンスが 1 つだけの場合は、それが ODBM、OM、RM のいずれであるかに関係なく、障害が起きた場合に、その CSL マネージャーのバックアップは存在しません。

共用キューまたはシスプレックス・テクノロジーを使用しない場合は、単純化された CSL 構成の利点を活用し、CSL OM を介してタイプ-2 コマンド を発行することができます。

関連概念:

47 ページの『簡易 CSL 構成』

IMSplex 内の CSL

IMS CSL は、IMSplex において、複数の IMS システムを、単一のもののように見なすことができるので、それらの管理の複雑性が軽減されます。IMSplex は、1 つのユニットとして作動できる、1 つ以上の IMS サブシステム (制御、マネージャー、またはサーバー) です。

このようなサブシステムは、一般的に (常時とは限らない)、以下のことを行います。

- データベース、リソース、またはメッセージ・キュー (あるいはいずれかの組み合わせ) の共用
- z/OS シスプレックス環境で稼働する。
- IMS CSL を組み込む。

IMSplex 内において、複数の IMS サブシステムを、あたかも 1 つのシステムであるかのように、管理することができます。例えば、ローカルでのオンライン変更時に IMS システム 1 つ 1 つにコマンドを入力する代わりに、1 つの単一制御点からコマンドを入力することで、そのコマンドが IMSplex 内の各 IMS システム上で実行されます。IMSplex は、ローカルまたは非シスプレックス環境でも存在可能です。CSL の使用はオプションです。

IMSplex のコンポーネントは、一般的にそれ自身のアドレス・スペースで実行される IMS 定義のエンティティであって、リソースを管理するか、操作を管理するか、または他の IMS 定義のエンティティ間の通信を支援します。これらのコン

ポーネントは、それに必要な始動プロシージャーが初期化されると、*IMSplex* メンバー になります。IMSplex コンポーネントの例としては、次のようなものがあります。

- IMS サブシステム (DB/DC、DBCTL、DCCTL)
- Resource Manager
- Operations Manager
- Open Database Manager
- 構造化呼び出しインターフェース
- IMS Connect
- DLIBATCH または DBBBATCH 領域
- リポジトリ・サーバー (RS) (Repository Server (RS))

DLIBATCH および DBBBATCH 領域は、RM および OM と対話しないという点で IMSplex コンポーネントの特殊なタイプと見なされます。

CSL が組み込まれた標準的な IMSplex 環境を以下の図に示します。異なる 3 つのオペレーティング・システム (OS) イメージがあります。それぞれの OS には、IMS 制御領域と SCI があります。さらに、OS 1 には OM と RM、OS 2 には OM と ODBM、および OS 3 には ODBM があります。この 3 つの OS イメージは、データベース共用構造、メッセージ・キュー構造、およびリソース構造を持つカップリング・ファシリティーを共用します。

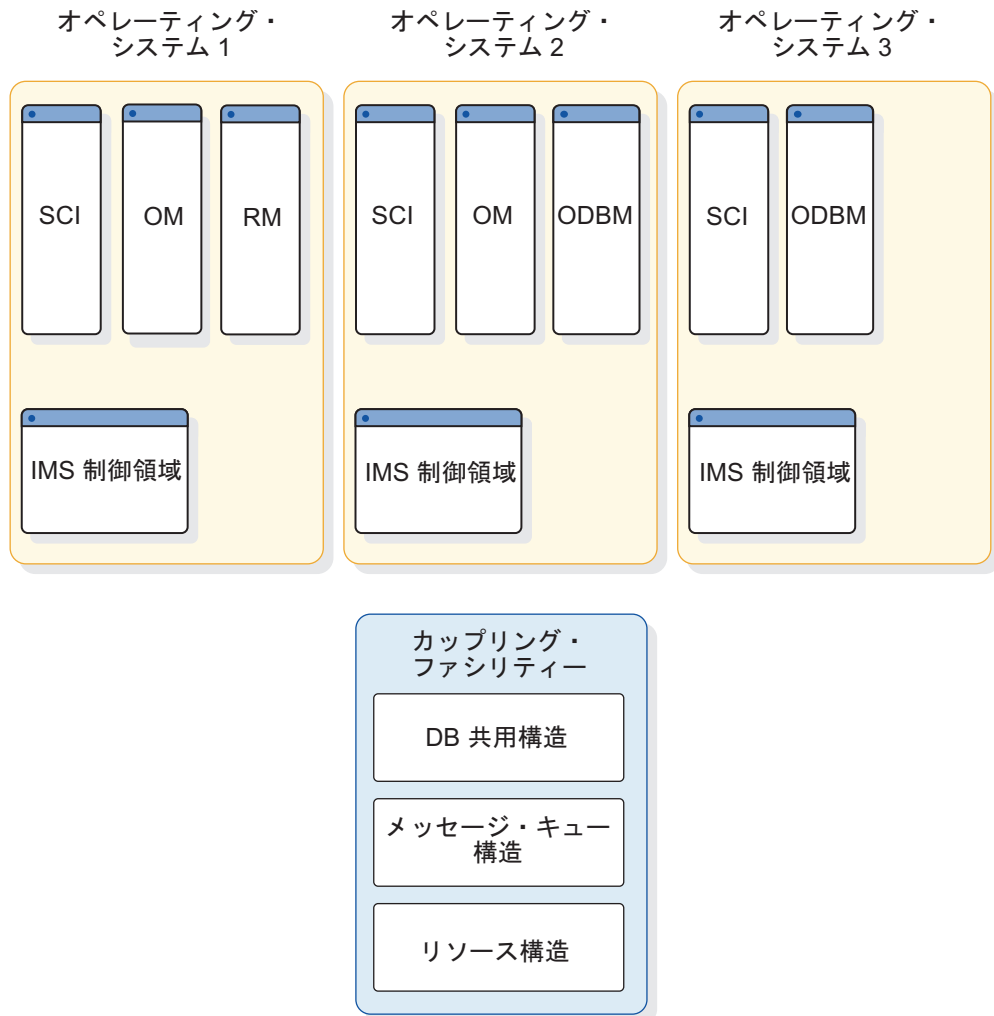


図 9. CSL が組み込まれた IMSplex 環境

CSL には以下の IMS アドレス・スペースが組み込まれており、この一部はオプションです。

- ODBM
- OM
- RM
- SCI

IMSplex に関与するアドレス・スペースは次のとおりです。

- IMS 制御領域アドレス・スペース
- IMS CSL マネージャー・アドレス・スペース (ODBM、OM、RM、SCI)
- IMS サーバー・アドレス・スペース (共通キュー・サーバー)
- IMS 以外のアドレス・スペース
- RS のアドレス・スペース

IMSplex 内の IMS システムの IMSID は、固有であることが必要です。

簡易 CSL 構成

ご使用の IMS 構成が RM サービスを必要としない場合は、OM および SCI を引き続き使用して、タイプ-2 IMS コマンドを活用することができます。

コマンドを発行するには、TSO SPOC、あるいは OM API を用いるユーザー作成またはベンダー作成の自動化プログラムを使用します。

IMS で IMSRSC リポジトリを使用可能にするには、RM を含む CSL が必要です。

IMS Application Menu は、TSO SPOC、Syntax Checker、IVP などのようなアプリケーションを開始するための、共通インターフェースを提供します。「IMS Application Menu」の詳細については、「IMS V15 インストール」を参照してください。

関連概念:

43 ページの『Common Service Layer の概要』

関連タスク:

28 ページの『タイプ 2 コマンド環境のための簡易 IMSplex の定義』

CSL マネージャー

CSL は、発展を続ける IMS アーキテクチャーのキー・コンポーネントです。IMSplex 内で CSL を使用することで、システムを管理を改善するための基盤が提供されます。

CSL アドレス・スペースは、CSL マネージャーとも呼ばれています。

- Open Database Manager
- Operations Manager
- Resource Manager
- 構造化呼び出しインターフェース

CSL Open Database Manager の概要

Open Database Manager (ODBM) は、IMSplex 内の DBCTL 環境または DB/TM 環境用に構成された IMS DB システムが管理する IMS データベースへの分散ローカル・アクセスを実現します。

単独で使用する場合でも IMS Connect と共に使用する場合でも、ODBM は、さまざまな分散ローカル環境から IMS データベースにアクセスするアプリケーション・プログラムの開発を容易にするために、各種インターフェースをサポートします。ODBM は、以下のインターフェースをサポートします。

- IMS Universal Database リソース・アダプター
- IMS Universal JDBC ドライバー
- IMS Universal DL/I ドライバー
- ODBA インターフェース
- ユーザー作成 ODBM クライアント・アプリケーション・プログラム用の ODBM CSLDMI インターフェース

上記のインターフェースのサポートに加え、ODBM は以下の機能を備えます。

- 同期点コーディネーターとして z/OS リソース・リカバリー・サービスを必要とするかしないかに関係なく、1 フェーズ・コミットをサポートします。ODBM パラメーター RRS=N を指定すると、IMS との通信にデータベース・リソース・アダプター (DRA) が使用されます。
- 別名に基づいて IMS システムに着信データベース要求を経路指定します。IMS システムの別名は、CSLDCxxx PROCLIB メンバーで ODBM に対して定義され、クライアント・アプリケーション・プログラムによって着信データベース要求で指定されます。

クライアント・アプリケーション・プログラムがデータベース・アクセス要求で別名を指定しない場合、ODBM はラウンドロビン方式の分散を使用して、複数の IMS システム間にその要求を経路指定します。この分散処理により、ODBM はそれぞれの着信要求を、ODBM に対して定義されているアクティブな IMS システムに経路指定します。

- ODBM クライアント・アプリケーション・プログラムで IMSplex 内の他の論理区画からデータベースにアクセスできるようにします。
- ODBM を介して ODBA インターフェースを使用する z/OS アプリケーション・プログラムの DL/I 処理時に、IMS 制御領域を予期しない終了から保護します。
- ODBM を使用するように構成されている ODBA アプリケーションでは、RRS 制御同期点の処理に参加することが可能な RRMS リソース・マネージャーとして機能します。
- リソース・マネージャーとして機能し、ローカル RRS トランザクションの単一フェーズ・コミット処理に必要な呼び出しを RRS に発行します。
- IMS Connect との併用により、DRDA 仕様を使用するクライアント・アプリケーション・プログラムに対して、完全な Distributed Relational Database Architecture™ (DRDA) のターゲット・サーバーとして機能します。

ODBM および IMS Connect はいずれも DRDA をサポートするため、DRDA 仕様の一部である分散データ管理 (DDM) 体系のコマンドを使用することで、IMS と通信する独自の DRDA ソース・アプリケーション・サーバーを開発できます。

ODBM は、z/OS システム管理機能 (SMF) を利用して、CPU 使用量などの ODBM アカウンティング情報のロギングと取り出しを行います。ODBM アドレス・スペースのロギングは、オプション・パラメーター LOGOPT=ACCOUNTING が ODBM 初期設定メンバー CSLDIxxx で指定された場合に活動化されます。

ODBM は、共通サービス層 (CSL) のアドレス・スペースで実行される、オプションの IMSplex コンポーネントです。ODBM は、通信には CSL の Structured Call Interface (SCI) サービスを使用し、コマンド処理には CSL の Operations Manager (OM) サービスを使用します。

ODBM 機能を使用するには、IMSplex 内に 1 つの ODBM インスタンスを定義する必要があります。各 z/OS イメージは複数の ODBM を持つことができます。IMSplex 内で ODBM の複数インスタンスが定義されている場合、そのすべての ODBM インスタンスが IMSplex 内のすべての z/OS イメージからの作業を実行できます。

以下の図は、ODBM を使用する IMS 構成の概要を示します。

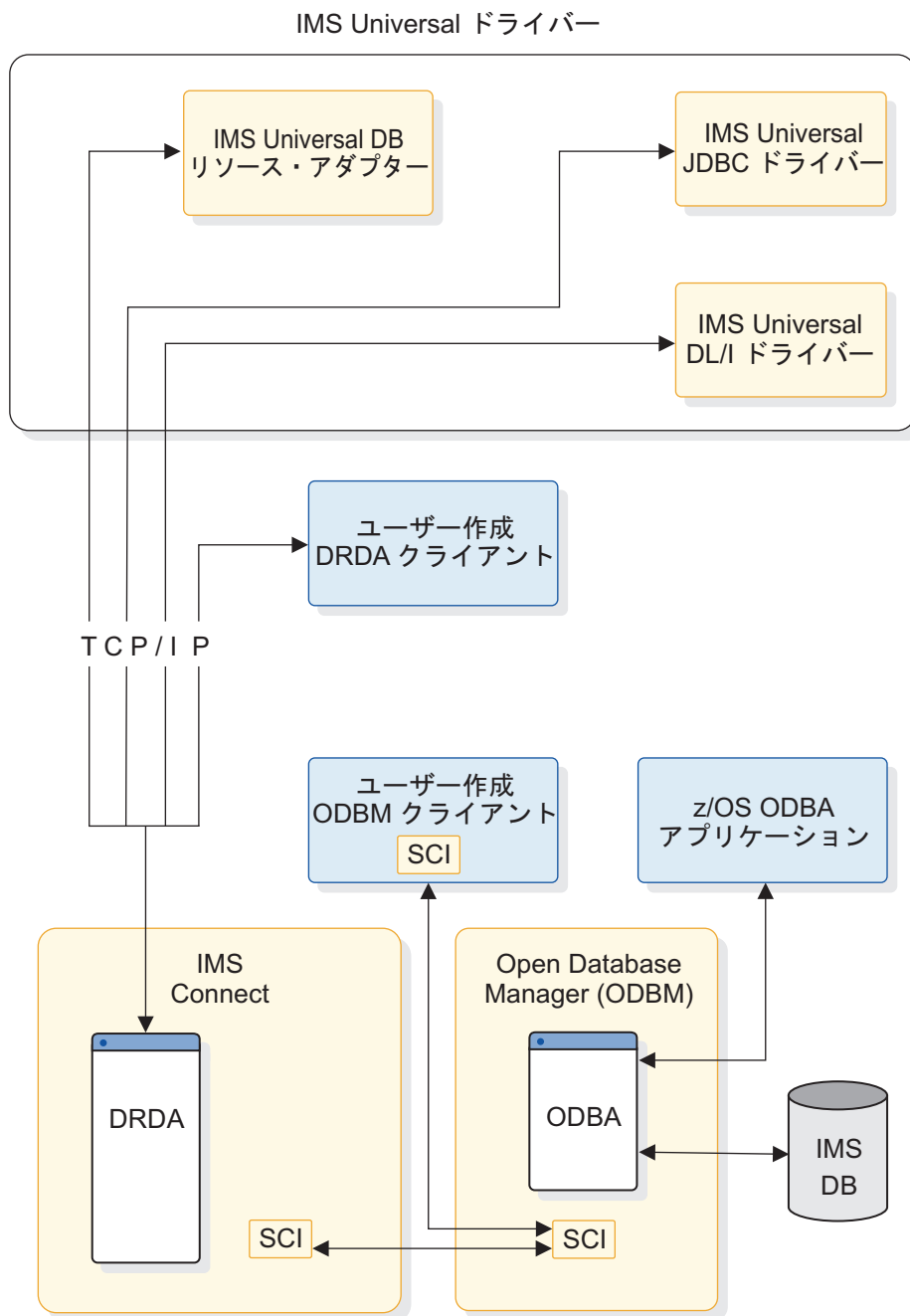


図 10. ODBM を使用する IMS 構成の概要

関連概念:

171 ページの『第 8 章 CSL ODBM の管理』

🔗 DRDA のための IMS サポート を使用したプログラミング (アプリケーション・プログラミング)

Operations Manager の概要

OM は、IMSpIex の操作を制御します。OM は、コマンドを発行し、応答を受け取るためのアプリケーション・プログラミング・インターフェース (OM API) を提供しています。単一制御点 (SPOC) インターフェースから、コマンドを OM にサブミットすることができます。SPOC インターフェースとしては、TSO SPOC および REXX SPOC API があります。コマンドをサブミットするためのユーザー固有のアプリケーションを作成することもできます。

OM は、IMSpIex に以下の機能を提供します。

- IMS コマンドを、そのコマンド用に登録した IMSpIex メンバーに経路指定する。
- 個々の IMSpIex メンバーからのコマンド応答を 1 つの応答に統合して、統合された応答をコマンドの発信元に戻す。
- 自動化操作プログラム・コマンド用の API を提供する。
- コマンド処理クライアントをサポートするための、コマンド登録用汎用インターフェースを提供する。
- コマンドおよび応答編集用のユーザー出口およびコマンド・セキュリティ用のユーザー出口を提供する。

CSLOIxxx PROCLIB メンバー上の IMSPLEX() キーワードの AUDITLOG= パラメーターを指定することにより、OM ログ・レコードを z/OS システム・ロガーのログ・ストリームに書き込めます。ログ・レコードは、コマンド入力、関連コマンド出力、および非送信請求メッセージ出力の監査証跡となります。

OM 機能を使用するためには、IMSpIex 内に OM が 1 つ定義されている必要があります。個々の z/OS イメージは複数の OM を持つことができます。IMSpIex 内で複数の OM が定義されている場合、定義されている OM のどれもが、IMSpIex 内のどの z/OS イメージからの作業でも実行することができます。

関連資料:

61 ページの『CSL 構成例』

Resource Manager の概要

RM は、IMSpIex 内の複数の IMS システムで共用されるリソースを管理するのを支援します。RM は、グローバル・リソースを管理し、グローバルオンライン変更など、IMSpIex 全体のプロセスを調整する基盤を提供します。

RM は、IMSpIex に以下の機能を提供します。

- IMSpIex 内の全 RM がアクセスできるカップリング・ファシリティ・リスト構造であるリソース構造内のグローバル・リソース情報を保守する。
- トランザクション、Iterm、または msname として定義されたリソースが IMSpIex 内の全 IMS システムで同じリソース・タイプとして定義されるように、リソースの整合性を確保する。
- リソース・サービスをサポートする。
- クライアント・サービスをサポートする。
- 共通キュー・サーバー (CQS) を使ってグローバル・リソース情報を保守する。
- IMSpIex 全体のプロセス (グローバル・オンライン変更など) を調整する。

RM により、システム管理者は、IMSplex 内で複数の IMS システムで共用されるリソースを管理することができます。RM は、グローバル・リソースを管理し、IMSplex 全体に渡るプロセスを調整するための基盤を提供します。

推奨事項: IMSplex でリソース構造はオプションですが、リカバリー機能を向上させるためにリソース構造を定義してください。共用キューを使用しており、かつ IMSplex 内の逐次アプリケーションを管理するには、少なくとも 1 つの RM が、定義されたリソース構造によりアクティブとなっている必要があります。

RM 機能を使用するためには、IMSplex 内で少なくとも 1 つの RM が定義されていなければなりません。リソース構造が定義されていれば、各 z/OS イメージ上で 1 つ以上の RM を持つことができます。リソース構造が定義されていない場合、1 つの RM しか持てません。RM 機能が不要の場合は、DFSCGxxx PROCLIB メンバー・データ・セット上で RMENV=N を指定して、RM なしの IMS システムを構成することができます。どの RM も IMSplex 内のどの z/OS イメージからの作業も処理することができます。

関連資料:

61 ページの『CSL 構成例』

CSL Structured Call Interface の概要

SCI は、IMSplex のメンバーが相互に通信し合えるようにします。IMSplex メンバー間の通信は、単一の z/OS イメージ内または複数の z/OS イメージ間で行なうことができます。個々の IMSplex メンバーは、他のメンバーの所在や使用する通信インターフェースについて知っている必要はありません。

SCI は、IMSplex に以下の機能を提供します。

- IMSplex 内でメッセージと要求を経路指定する。
- IMSplex メンバーを登録および登録解除する。
- メンバーが IMSplex に加わったとき、または抜けたとき、IMSplex メンバーに知らせる。
- メンバーが IMSplex に加わったとき、メンバーのセキュリティー認証を準備する。
- クライアントとサーバーを基礎の通信テクノロジーから切り離すための単一呼び出しインターフェースを備えている。

SCI サービスを必要とする IMSplex メンバーは、その z/OS イメージ上に SCI がなければなりません。各 z/OS イメージ上において、それぞれの IMSplex に置ける SCI アドレス・スペースは多くても 1 つです。

注: DBRC 自動 RECON 損失通知と並列 RECON アクセスは、SCI を使用しますが、RM または OM は使用しません。SCI アドレス・スペースは、自動 RECON 損失通知のための他の CSL マネージャーのアドレス・スペースを必要とせずに、起動することができます。

推奨事項: IMSplex 内の CSL マネージャーの各アドレス・スペースを初期化してください。

関連資料:

61 ページの『CSL 構成例』

CSL での Single Point of Control (SPOC) プログラム

単一制御点 (SPOC) とは、マスター端末を使用する代わりに、IMSplex 内のすべての IMS システムの操作を管理できるようにするプログラムのことです。SPOC を使うと、IMSplex 内の全メンバーに同時にコマンドを出すことができます。

SPOC は単一の OM と通信します。その OM は、SCI を介して、IMSplex 内の他のすべての IMS 制御領域と通信します。

SPOC は、IMSplex ではオプションです。WTOR、MTO、および E-MCS コンソール用の既存のコマンド・インターフェースは、タイプ 1 コマンドについてのみサポートされています。タイプ 2 コマンドを使用する場合には、SPOC を使用する必要があります。ほとんどのコマンド処理では、z/OS マスター・コンソール、IMS マスター端末、またはユーザー端末の代わりに SPOC を使用できます。

SPOC のタイプ:

- 3270 型 TSO 端末では、TSO SPOC は、ISPF パネル・インターフェースを持つシステム管理アプリケーションです。IMS Application Menu を使用して、TSO SPOC を開始することができます。TSO SPOC はまた OM 監査証跡も表示しますが、これには IMS が出したメッセージおよびオペレーターが発行したコマンドのログが記録されています。
- バッチ SPOC ユーティリティは、OM サービスを使って IMS オペレーター・コマンドを IMSplex に送信します。このユーティリティは、OM API でサポートされているすべてのコマンドを受け入れます。バッチ SPOC ユーティリティの起動は、標準 JCL ステートメントを使って行います。
- REXX SPOC API を使用すれば、自動化プログラムは SPOC 機能を使用できます。REXX SPOC API を使用してコマンドを自動化することもできます。
- ベンダー作成またはユーザー作成の SPOC: OM API を使用して、または OM API にアクセスして SPOC 機能を実行するプログラムです。

IMSplex 内に複数のタイプの SPOC を置くことができ、同時にいくつかの SPOC でもアクティブにできます。

SPOC は、以下の機能を IMSplex に提供します。

- 単一コンソールから IMSplex 内のすべての IMS システムを操作できるようにすることによる IMSplex の単一システム・ビュー
- 複数の IMS システムからのコマンド応答の統合表示
- IMSplex 内の任意の IMS 制御領域に接続されている IMS 端末へのメッセージの送信

以下の図は、IMSplex 内の SPOC アプリケーションを示します。この例における IMSplex には、3 つの同一の OS イメージがあります。それぞれのイメージに IMS 制御領域、IMS CQS アドレス・スペース、SCI、OM、および RM があります。この 3 つの OS イメージは、データベース共用構造、メッセージ・キュー構造、およびリソース構造が組み込まれたカップリング・ファシリティーを共有します。この IMSplex 構成には、共用データベースおよび RECON データ・セットも組み込まれています。

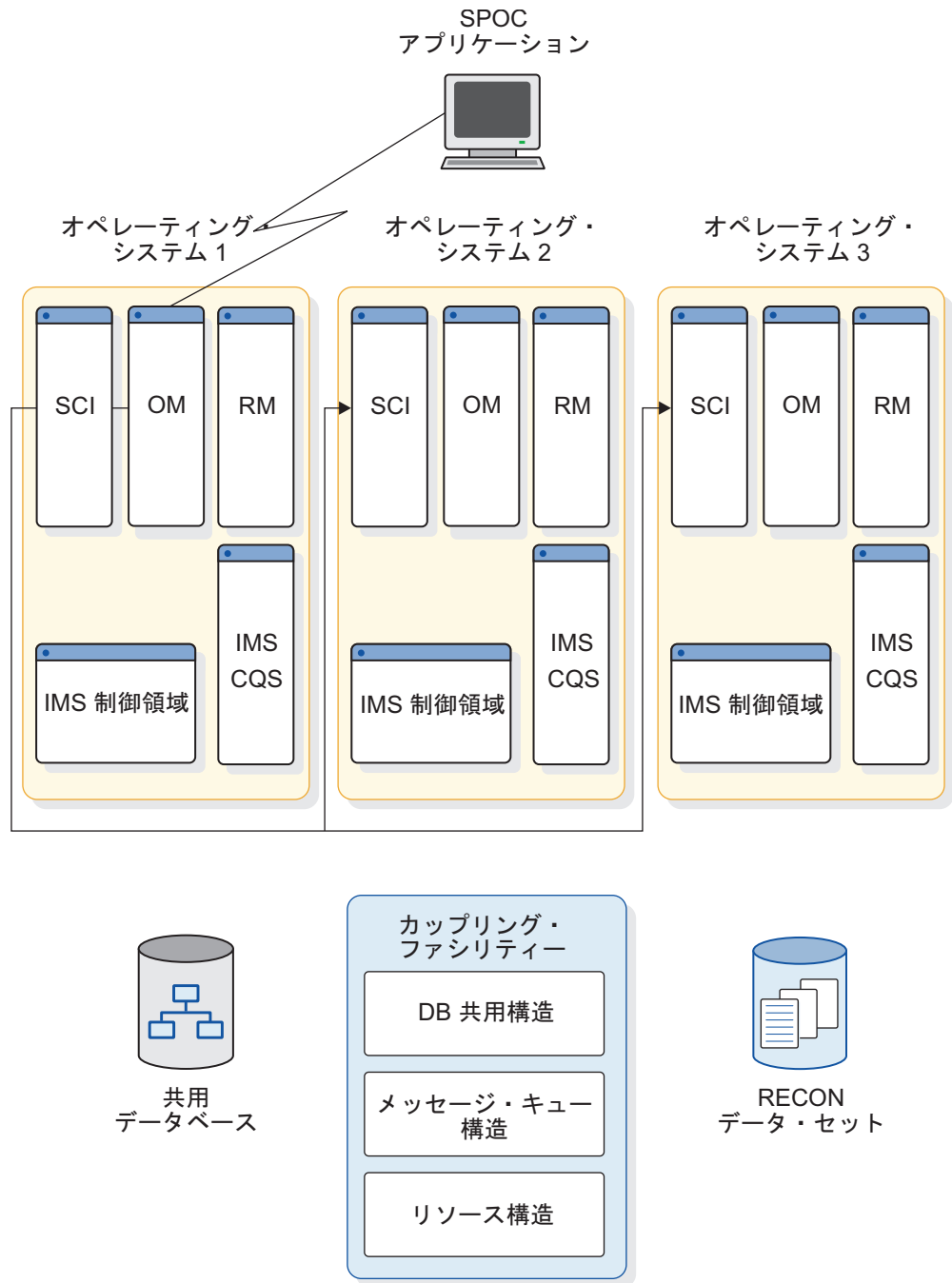


図 11. IMSplex 内の SPOC アプリケーション

以下の図に、複数の SPOC ユーザーが含まれる IMSplex 環境を示します。この IMSplex 環境には 4 つの IMS 制御領域 (それぞれに自身の SCI) と 1 つの OM があります。複数の SPOC ユーザーとして、2 つの SPOC TSO/ISPF アプリケーション、REXX SPOC API、およびベンダー作成の SPOC プログラムがあります。各 SPOC が IMSplex 環境にアクセスすることができます。

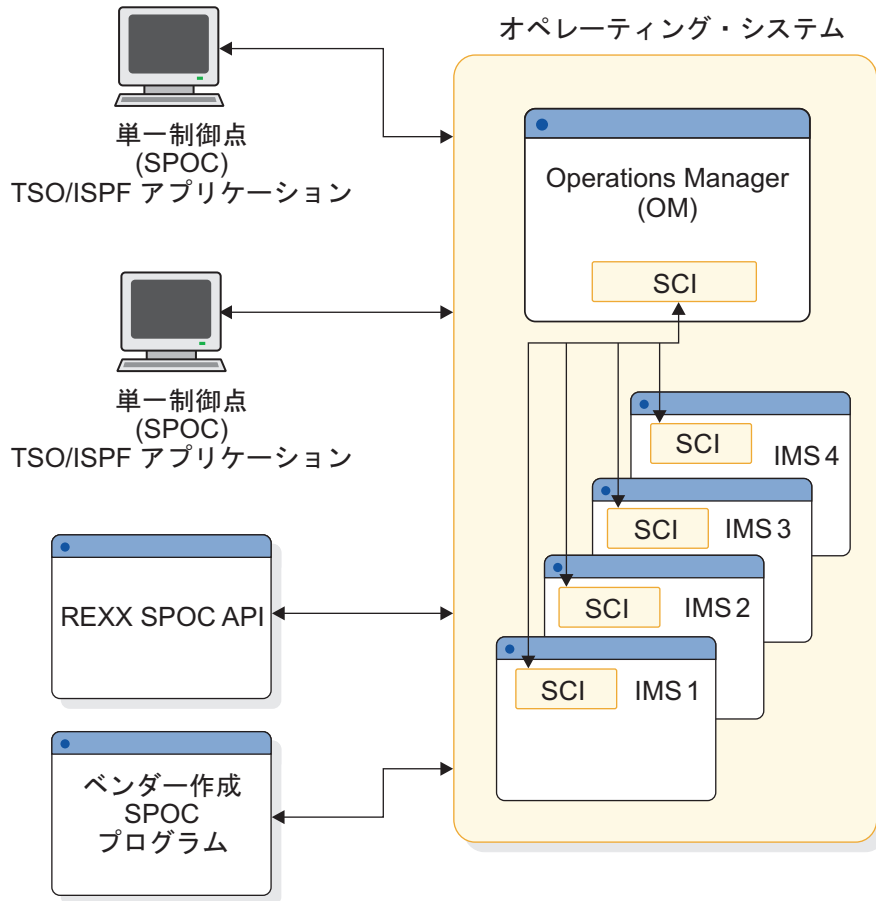


図 12. IMSplex 内の複数の SPOC ユーザー

関連概念:

☞ TSO SPOC アプリケーションによる IMS の制御 (オペレーションおよびオートメーション)

『REXX SPOC API』

関連タスク:

☞ 「IMS Application Menu」からの IVP の開始 (インストール)

関連資料:

☞ IMS タイプ 2 コマンド・フォーマット (コマンド)

REXX SPOC API

REXX SPOC API では、REXX プログラミング言語で作成された自動化操作プログラムのアプリケーション・プログラムは、IMSplex に対して IMS オペレーター・コマンドを発行したり、コマンド応答を取り出すことができます。

コマンド応答は、XML ステートメントの形式になっています。REXX アプリケーション・プログラムは、コマンド応答を調べて、必要な場合は、さらに処理を実行できます。

例えば、REXX アプリケーション・プログラムは、以下のコマンドを発行して、トランザクションのキュー・カウントを検査できます。

```
QRY TRAN NAME(prod001) SHOW(QCNT)
```

応答により高いキュー・カウントが示された場合、REXX アプリケーション・プログラムは、他のコマンドを発行して、追加の領域を始動したり、トランザクションのクラスを変更したりできます。

REXX アプリケーション・プログラムを作成し、それを TSO、バッチ、または Tivoli® NetView® for z/OS 環境で実行できます。REXX アプリケーション・プログラムが IMS オペレーター・コマンドを発行する前に、SCI、OM、および IMS 制御領域アドレス・スペースを開始する必要があります。

グローバル・オンライン変更

グローバル・オンライン変更機能は、IMSplex 内のすべての IMS システムのリソースをオンラインで変更します。

グローバル・オンライン変更の利点として次のものがあります。

- IMSplex 内のすべての IMS システムに関して、オンライン変更のコマンドを 1 回発行するだけでリソースをオンラインで変更する単純化されたプロセス
- それぞれの IMS 停止せず、しかも IMS 手動でオンライン変更と調整しないような継続処理
- IMSplex 内の一部の IMS システムでオンライン変更がコミットされ、他の IMS システムではコミットされないという状態が生じない

グローバル・オンライン変更を使用して、マスター IMS 制御領域は、IMSplex 内のすべての IMS システムにある、以下のリソースのオンライン変更を調整します。

- データベース (ACBLIB 内の DMB)
- データベース・ディレクトリー (MODBLKS 内の DDIR)
- MFS フォーマット (FMTLIB)
- プログラム (ACBLIB 内の PSB)
- プログラム・ディレクトリー (MODBLKS 内の PDIR)
- トランザクション (MODBLKS 内の SMB)
- 宛先コード (MODBLKS 内の RCTE)

制約事項: リソース定義の動的定義が使用可能になっている場合、オンライン変更機能は以下のタイプのリソースに対してサポートされません。

- ACB の IMS 管理が使用可能である場合のデータベース (DBD) およびプログラム・ビュー (PSB) リソース。
- 動的リソース定義が使用可能である場合の MODBLKS リソース。

グローバル・オンライン変更は、ユーザーまたは OM によって指定されたとおりに、マスター IMS 制御領域を介して作動します。OM は、RM がアクティブであれば、RM を使用して、IMSplex 内のその他の IMS システムとのオンライン変更のフェーズを調整します。オンライン変更の開始は、OM API を介してコマンド INITIATE OLC を発行することによって行います。その時点で、マスター IMS 制

御領域を指定できます。ユーザーがマスター IMS 制御領域を選択しなければ、OM が IMS システムのいずれかをマスター IMS 制御領域として選択します。

グローバル・オンライン変更を IMSplex で指定する場合、以下のコンポーネントが必要です。

- CSL と、少なくとも 1 つの OM で定義された IMSplex
- CQS (IMSplex 内にリソース構造がある場合)

リソース構造は、IMSplex 全体にわたるプロセス状況を保管するのに使用されます。また、これにより、オンライン変更時のリカバリー機能 (障害が発生した場合) が向上します。

- OLCSTAT データ・セット。これは、最初の IMS コールド・スタートを行う前に、グローバル・オンライン変更ユーティリティ (DFSUOLC0) で初期設定しておく必要があります。

RMENV=N を指定することによって IMSplex から RM を除外する場合、各 IMS システムは、固有の OLCSTAT データ・セットを持っている必要があり、その OLCSTAT データ・セットには、それを所有する IMS システムの IMSID が含まれており、他の IMSID が含まれてはなりません。

- DFSCGxxx PROCLIB メンバー・データ・セット内の OLC=GLOBAL パラメーターと OLCSTAT= パラメーター
- IMS EXEC ステートメントの CSLG= パラメーター

グローバル・オンライン変更には、少なくとも 1 つの RM とリソース構造をお勧めしますが、それらは必須ではありません。RM またはリソース構造を使用しないことにした場合は、以下を考慮してください。IMSplex で RM を使用しない場合、OLCSTAT データ・セットに入れられるのは、その OLCSTAT データ・セットを所有する IMS システムの IMSID だけです。別の IMSID または複数の IMSID を持つ OLCSTAT データ・セットが含まれている IMS システムを再始動しようとすると、異常終了になります。IMS は、OLCSTAT データ・セットを所有する IMS システム以外の IMS システムによって発行された INITIATE OLC コマンドおよび TERMINATE OLC コマンドをリジェクトします。

グローバル・オンライン変更を使用可能にしないことにした場合、IMS プロシージャ内の MODSTAT データ・セットはオプションです。IMS 制御領域のサンプル JCL には、MODSTAT データ・セットおよび MODSTAT2 データ・セット (XRF の場合) の DD ステートメントが含まれます。グローバル・オンライン変更を使用可能にする IMSplex に対して MODSTAT DD ステートメントを指定した場合、MODSTAT および MODSTAT2 (XRF の場合) データ・セットも存在していなければなりません。

関連タスク:

28 ページの『タイプ 2 コマンド環境のための簡易 IMSplex の定義』

ACB ライブラリー・メンバー・オンライン変更

IMS システムが IMSplex 環境で ACB ライブラリーを使用する場合は、ACB メンバー・オンライン変更 (OLC) 機能を使用して、ACB ライブラリーの個々のメンバーの追加または変更、あるいは ACB ライブラリー全体の変更を行い、さらに、

IMSplex を静止させたり、アクティブな ACB ライブラリーのリフレッシュを行ったりせずに、新規メンバーまたは変更したメンバーをオンラインにすることができます。

ACB メンバー・オンライン変更機能は、ACB の IMS 管理が使用可能な場合は適用されません。この場合は、DDL またはいずれかの IMS カタログ・データ取り込みユーティリティーを使用して IMS に変更を実行依頼することで、個々の ACB メンバーをオンラインで変更することができます。変更がオンライン・システムで自動的にアクティブにならない場合は、IMPORT DEFN SOURCE(CATALOG) コマンドを発行してください。

メンバー OLC の影響を受けないメンバーは静止されません。

ACB ライブラリーにオンライン変更を加えられるのは、以下の IMSplex 環境です。

- 単一の IMS システムで構成される IMSplex。この構成では、IMS は、OM と SCI からなる CSL を持つ OLCSTAT データ・セットを使用する必要があります。RM は必要ではありません。
- 複数の IMS システムで構成される IMSplex。この構成では、各 IMS システムが、同じ共用 OLCSTAT データ・セット、または各自のローカル OLCSTAT データ・セットを使用する必要があります。CSL は OM、SCI、および RM で構成されていなければなりません。リソース構造をお勧めしますが、必須ではありません。

ACBLIB メンバーに対しグローバル・オンライン変更を行えるのは、IMSplex 中のみです。

IMSplex 内で ACBLIB メンバーを共用にするのか、専用にするのかを指定できます。以下に、ACBLIB メンバー・オンライン変更プロセスを OLCSTAT データ・セットを共用しているすべての IMS システム間で調整する例を中心に説明します。

- OM が、コマンド・マスター IMS となる IMS システムを選択します。
- コマンド・マスター IMS が、RM を使って OLCSTAT データ・セットを共用している他の IMS システムとメンバー OLC 処理の調整を行います。
- DFSCGxxx または DFSDFxxx PROCLIB メンバーにおいて、ACBSHR= を Y または N に指定する必要があります。
 - Y は、OLCSTAT 内のすべての IMS システムが同じアクティブおよび非アクティブな ACBLIB を使用していることを示します。
 - N は、OLCSTAT 内の各 IMS が各自専用のアクティブおよび非アクティブな ACBLIB を使用していることを示します。
- OLCSTAT データ・セットを共用している IMSplex 内のすべての IMS システムは、ACBSHR= の値を指定する必要があります。
- メンバー OLC では、共用されているかどうかにかかわらず、すべてのアクティブな ACBLIB が更新されます。

ACB メンバー・オンライン変更では、IMS タイプ 2 コマンドのみ使用されます。ACB ライブラリー・メンバー・オンライン変更の実行に関わるコマンドは、次のとおりです。

- INITITATE OLC PHASE(PREPARE)

- INITIATE OLC PHASE(COMMIT)
- TERMINATE OLC
- QUERY MEMBER TYPE(IMS)
- QUERY OLC SHOW(RSCLIST)

ACB メンバー・オンライン変更の実行に必要なステップについては、591 ページの『オンラインでの IMS.ACBLIB メンバーの変更または追加』を参照してください。変更されたリソースの以前のバージョンにフォールバックするには、完全なライブラリー切り替えとともに、完全なオンライン変更プロセスを行ってください。

グローバル TM リソース管理

RM とリソース構造を使用すると、IMSplex 内の Transaction Manager のリソースを一層効率よく管理できるようになります。IMS は、LTERM (VTAM[®])、ノード (VTAM 単一セッション)、ユーザー ID (インストール・システムがシングル・サインオン制約を要求する場合)、およびユーザーに関して、名前の一意性を強制します。

また IMS は、メッセージ宛先ついてリソース・タイプ整合性を保ち、グローバル呼び出し可能サービスをサポートします。

関連タスク:

25 ページの『IMSplex の定義および調整』

自動 RECON 損失通知の概要

SCI は、自動 RECON 損失通知で必要とされる唯一の CSL マネージャーです。CSL を自動 RECON 損失通知専用で使用している場合、OM および RM は必要ありません。ただし、SCI は、それぞれの z/OS イメージで使用可能でなければなりません。DFSCGxxx または DFSDFxxx メンバーの CSL セクションを使用して IMS 制御領域を開始する場合は、OM が (そしておそらく RM も) 必要となります。

ある特定の RECON を使用する DBRC インスタンスはすべて、同一の IMSplex に結合する必要があります。複数の IMS 共用グループが単一の RECON を使用するように IMS システムを構成することができます。この構成は、共用グループが同一 IMSplex の一部となっている場合にサポートされます。ただし、それぞれの共用グループが固有の IMSplex として識別される場合には、RECON を複数の RECON に分割して、IMSplex 当たり 1 つとなるようにする必要があります。また、制御領域と DBRC アドレス・スペースの両方が SCI を使用する場合、それらを同一の IMSplex に結合する必要もあります。複数の IMS 共用グループが同じ IMSplex に属しながらも、固有の RECON を使用するように IMS システムを構成することができます。

推奨事項: 同じ IMSplex 内の各 RECON は、固有の DBRC 共用グループ ID を持っている必要があります。これは、IMSplex 内の複数の RECON に対し並列 RECON アクセスが使用可能とされた場合に必要となります。

関連資料: 自動 RECON 損失通知の構成についての詳細は、「IMS V15 データベース管理」を参照してください。

CSL を備えた IMSplex の構成

CSL なしで実行されている IMS 制御領域は、いったん停止してから、再始動して CSL に接続する必要があります。CSL が IMSplex に導入されている場合、IMS 制御領域を一度に 1 つずつ停止して再始動することができます。CSL を導入するために IMSplex 全体をシャットダウンする必要はありません。

CSL を備えた IMSplex を構成するには、以下の IMSplex コンポーネントのそれぞれについて少なくとも 1 つを初期設定する必要があります。

- DB/DC、DCCTL、または DBCTL IMS 制御領域
- それぞれの IMS 制御領域用の DBRC
- Operations Manager (OM)
- IMSplex メンバーが稼働するそれぞれのオペレーティング・システム上の Structured Call Interface (SCI)
- CQS (メッセージ・キューが共用される場合、または RM をリソース構造と一緒に使用してグローバル・リソースを管理する場合)

Resource Manager (RM) とリソース構造も初期設定することを強くお勧めします。RM とリソース構造は、必須ではありませんが、リソースをグローバルに共用または管理することはできません。

オプションとして、以下のものを IMSplex に含めることができます。

- CSL Open Database Manager。これは、IMSplex 内の DBCTL 環境または DB/TM 環境用に構成された IMS DB システムが管理する IMS データベースへのローカル分散アクセスを提供します。
- DB/DC または DBCTL 環境での 1 つ以上のデータベースのデータ共用
- IMS 制御領域ごとに 1 つの DL/I アドレス・スペースと 1 から 999 個の従属領域
- 共用キュー。

重要: すべての IMS 制御領域がメッセージ・キューを共用する必要があります。共用しない場合には、共用キューを使用しないでください。共用されるキューとそうでないキューがある混合環境で実行すると、OM API を介して入力されるタイプ 2 コマンドの出力が予測不能であったり、誤解を招くようなものになる場合があります。

- 任意の数のバッチ (DLIBATCH または DBBBATCH) 領域。OM と RM はバッチ領域とは相互作用しません。ですから、それらはバッチ領域の管理には影響しません。
- 共用リソース定義用の IMSRSC リポジトリ、および Repository Server (RS) アドレス・スペース

CSL を組み込むには、CSLG= パラメーターを指定して (または、IMS PROCLIB データ・セットの DFSDFxxx メンバーに COMMON_SERVICE_LAYER セクションが組み込まれている場合は、DFSDF= パラメーターを指定して)、IMS 制御領域を開始する必要があります。CSL の最小構成は、IMSplex コンポーネント (TSO SPOC および CQS のような IMSplex コンポーネントを含む) が置かれているそれ

それぞれのオペレーティング・システムに SCI が 1 つのみ、IMSplex 内の任意の場所に 1 つ以上の OM、および IMSplex 内の任意の場所に 1 つ以上の RM を持つような構成です。

RM および OM の数は、ユーザーの要件によって異なります。CSL の構成規則の概要を以下に示します。

- 所定の IMSplex について、オペレーティング・システムで実行できる SCI は 1 つだけです。同じオペレーティング・システム上の複数の IMS システムは、同じ SCI のインスタンスを使用して同じ IMSplex 内で通信します。同じオペレーティング・システム上の 1 つ以上の IMS システムが複数の IMSplex と関連付けられている場合は、各 IMSplex について別の SCI のインスタンスを実行する必要があります。
- 少なくとも 1 つの OM が IMSplex 内でアクティブでなければなりません。1 つ以上のオペレーティング・システム上で 1 つ以上の OM をアクティブにすることができます。
- RM は必須ではありませんが、IMSplex で少なくとも 1 つの RM をアクティブにすることをお勧めします。任意の数のオペレーティング・システム上で 1 つの RM をアクティブにすることができます。複数の RM がある場合には、リソース構造も持っている必要があります。
- リポジトリを使用するように IMS が構成されている場合は、RM を持つ CSL 構成が必要です。

IMSplex のためのリソース構造の構成

リソース構造は、共通キュー・サーバー (CQS) によってサポートされるもので、RM が管理するリソースのリポジトリです。CQS は、単一 IMSplex 内で共有される固有名付きクライアント・リソースを、カップリング・ファシリティー・リスト構造を使用して管理します。

リソース構造はオプションです。ただし、リソース構造がないと、一部の IMSplex 機能を使用できません。

クライアントは、リソース構造を使用して、リソース情報や制御ブロック情報などを共用できます。リソース名は構造内で固有です。リソースは、更新したり、削除したり、照会したりすることができます。リソースを入れるために、基本カップリング・ファシリティー・リスト構造が使用されます。リソース構造は、オーバーフロー構造をサポートしません。

CQS は、リソース構造に関して、構造リカバリー、構造チェックポイント、およびオーバーフロー処理をサポートしません。また、パフォーマンスを向上させる目的で、リソース構造の変更はログに記録されません。ただし、リソースのリカバリーについては、リソース構造を自動的に二重化することができ、それは CFRM ポリシーに指定することにより行えます。

CQS は、リソース構造に最初に接続するときに、リソース構造の属性を定義します。CQS は、リソース構造を永続 ENTRYKEY リスト構造として定義します。リソース構造上のリスト項目は、付加属性エリアとデータ入力項目を含むように定義されます。リソース構造は、SVC ダンプ、構造変更、および自動再作成が行えるように定義されます。

推奨事項: IMSplex 内では複数の RM と 1 つのリソース構造を使用してください。リソース構造がない場合、RM のグローバル・リソース共有機能を使用することはできません。また、リソース構造がない場合、1 つの RM しか始動できません。その 1 つの RM がダウンした場合、バックアップ RM はないこととなります。しかしながら、DBCTL 環境において、リソース構造がない場合、1 つの RM でグローバル・オンライン変更を実行することはできます。

タイプ 2 コマンド環境のための簡易 IMSplex の構成

タイプ 2 コマンドを使用する予定で、スタンドアロン IMS システムまたはデータベースを共用するのみの複数の IMS システムがある場合、RM を使用しない IMSplex を定義できます。これにより、CSL を使用して簡易 IMSplex が作成されます。

関連資料:

- RM が使用されない場合のオンライン変更の制約事項については、55 ページの『グローバル・オンライン変更』を参照してください。
- RM を使用しない IMSplex の定義については、以下を参照してください。
 - 28 ページの『タイプ 2 コマンド環境のための簡易 IMSplex の定義』
 - IMS V15 システム定義
- CSL 構成の詳細については、『CSL 構成例』を参照してください。

CSL 構成例

一般に、IMS 制御領域 (DL/I、DBRC、従属領域)が CSLの使用を必要とする場合、SCI、OM、および RM がすべて必要です。ただし、1 つの IMSplex 内でさまざまな CSL の構成が可能です。例えば、CSL 構成に ODBM を含めたり、RM を省略したりしてもかまいません。

基本的な CSL コンポーネントは、以下のような CSL マネージャーです。

Open Database Manager

Open Database Manager (ODBM) は、IMSplex 内ではオプションです。パフォーマンスと可用性を向上させるために、IMSplex 内に追加 ODBM を定義することができます。

Operations Manager

IMSplex 内には、少なくとも 1 つの OM がなければなりません。パフォーマンスと可用性を向上させるために、IMSplex 内に追加 OM を定義することができます。

Resource Manager

IMSplex 内の IMS が RM サービスを必要とする場合、IMSplex が初期設定されるときに少なくとも 1 つの RM が使用可能でなければなりません。リソース構造が使用されていれば、パフォーマンスと可用性を向上させるために IMSplex 内に追加 RM を定義することができます。しかし、リソース構造が使用されていない場合、IMSplex 内で 1 つの RM しか始動できません。RM なしで IMSplex を構成し、タイプ 2 コマンドを使用できます。

IMSplex で IMSRSC リポジトリが使用可能になっている場合は、RM を含む CSL の構成が必要です。

構造化呼び出しインターフェース

IMSplex メンバーが実行されている z/OS イメージは、それぞれが 1 つの SCI が必要です。特定の IMSplex の各オペレーティング・システム・イメージで使用できる SCI は 1 つのみです。

CSL が組み込まれた IMSplex 構成例

以下の図に、CSL、SPOC、および自動手順を組み込んだ IMSplex の構成例を示します。

- OS イメージには、ODBM、OM、SCI、RM、IMS 制御領域、IMS Connect、および IMS CQS 用のアドレス・スペースがあります。
- OS イメージは、カップリング・ファシリティとデータベースを共有します。
- SPOC アプリケーション、自動化アプリケーション、マスター端末、およびエンド・ユーザー端末すべてが OS イメージにアクセスします。
- IMS 用の TCP/IP ゲートウェイである IMS Connect を介して OS イメージにアクセスする、オープン・データベース・リソース・アダプターおよび API も示してあります。

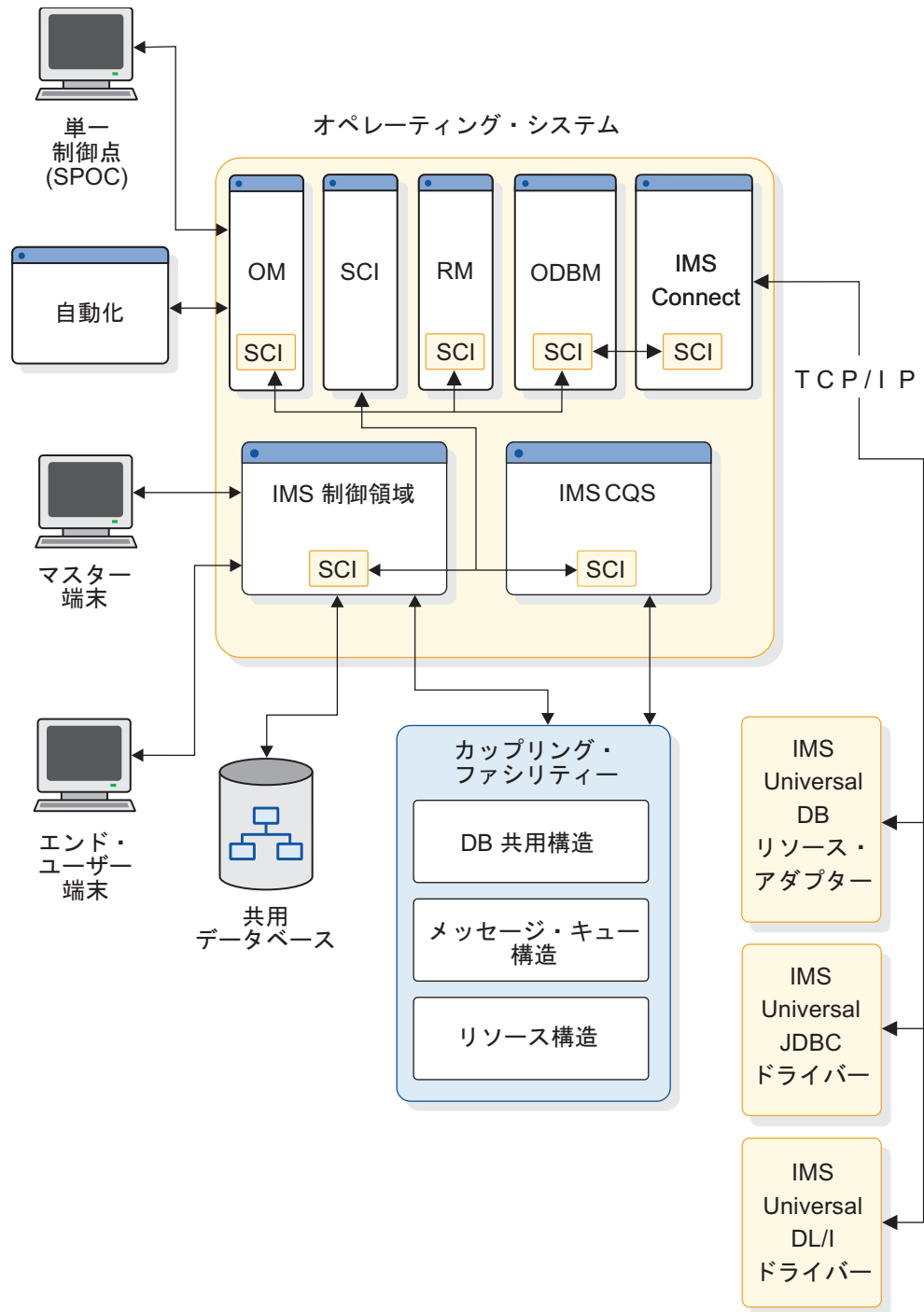


図 13. CSL が組み込まれた IMSplex 構成例

最速の通信を行うためには、各 z/OS イメージに OM と RM を定義してください。IMSplex コンポーネントは、そのコンポーネントと同じ z/OS イメージにある OM または RM と通信するほうが、別の z/OS イメージにある OM または RM と通信するよりも迅速になります。しかし、この構成は、一方で、各 z/OS イメージ上のアドレス・スペースの数を増やし、その結果、稼働環境がより複雑になります。

注: IMSplex 全体で RM と OM が、それぞれ、1 つずつしか定義されていない場合、障害の際にそのマネージャーの作業を行うバックアップがありません。

推奨事項: IMSplex 全体で複数の RM、OM、および SCI を定義してください。

IMSplex 最小 CSL 構成

以下の図は、IMSplex 内の CSL に可能な、最小の構成を示します。各 OS イメージには、IMS 制御領域と SCI があります。さらに、最初の OS イメージには OM があります。始動パラメーターに RMENV=NO が指定されました。

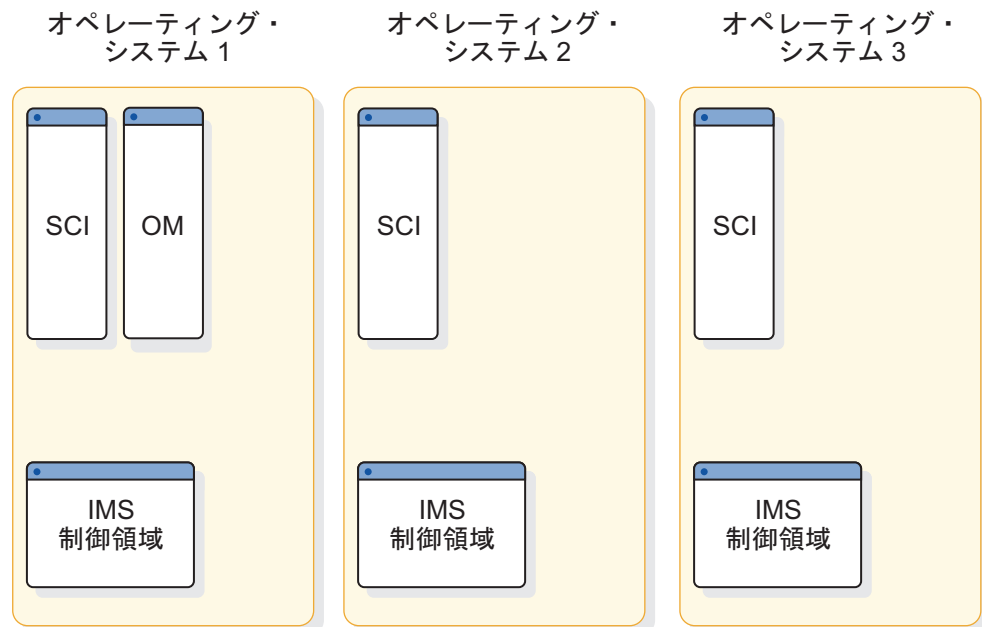


図 14. IMSplex 最小 CSL 構成

この図では、各 z/OS イメージには個別の IMS 制御領域があるため、個別の SCI があります。OM を 1 つの z/OS イメージ上に置き、それを IMSplex 内の他のイメージで使用することができます。

IMSplex 混在バージョン CSL 構成

以下の図は、IMSplex 内に IMS の複数のバージョンがある、さらに複雑な構成を示します。

- オペレーティング・システム 1 には、IMS バージョン 12 制御領域があり、OM および RM を実行しています。
- オペレーティング・システム 2 には、IMS バージョン 13 制御領域、ODBM、および OM があります。
- オペレーティング・システム 3 には、IMS バージョン 11 制御領域があります。

3 つすべての OS イメージがカップリング・ファシリティーを共有します。カップリング・ファシリティー内にリソース構造があります。したがって、別の OS イメ

ージに追加 RM を定義することができます。

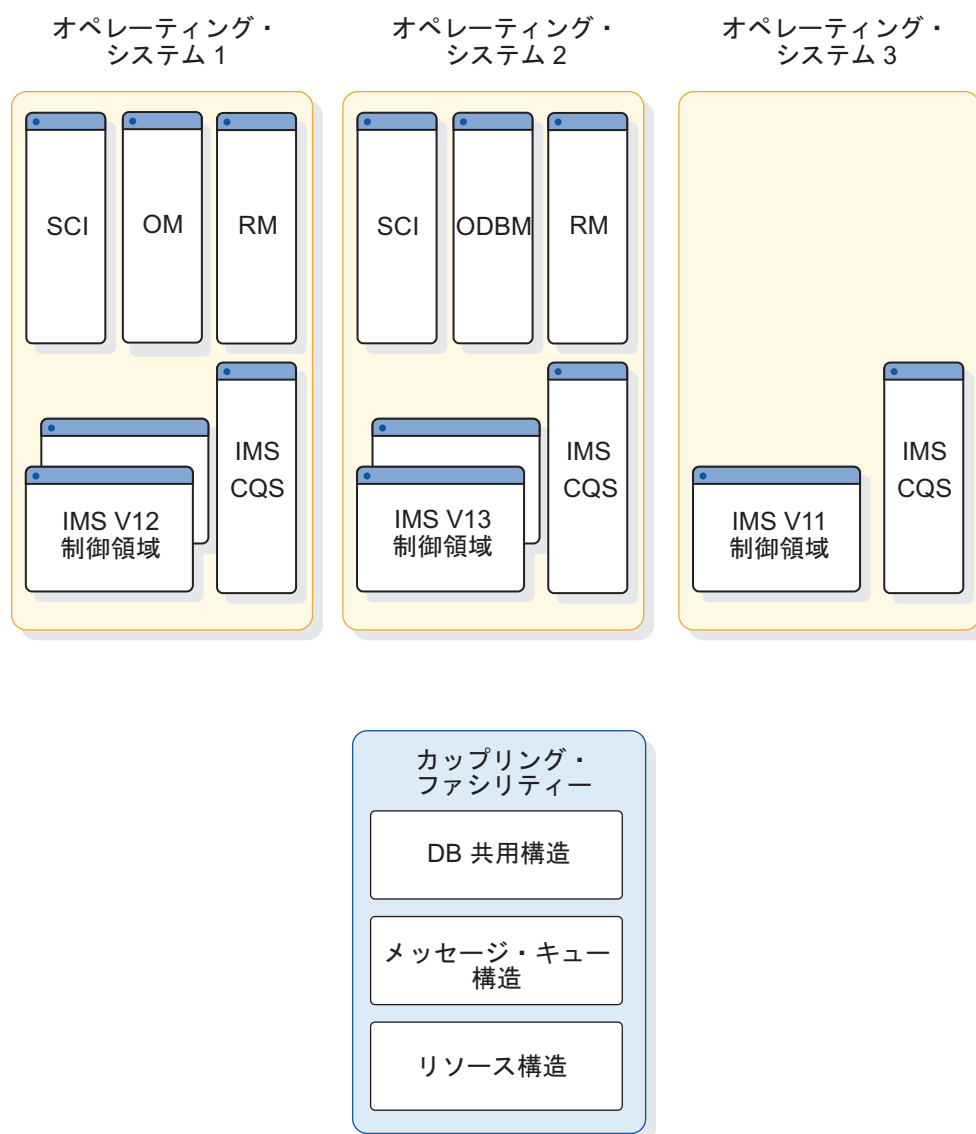


図 15. IMSplex 混在バージョン CSL 構成

IMSplex DBCTL CSL 構成

以下の図は、DBCTL 環境における共通サービス層の構成を示します。ここでは、3つの OS イメージそれぞれに IMS DBCTL 制御領域があります。すべてに SCI と OM があります。2 つには ODBM があります。リソース構造が定義されていないため、ここ OS1 では、IMSplex で使用できる RM は 1 つのみです。

推奨事項: DBCTL 用のリソース構造を定義してください。

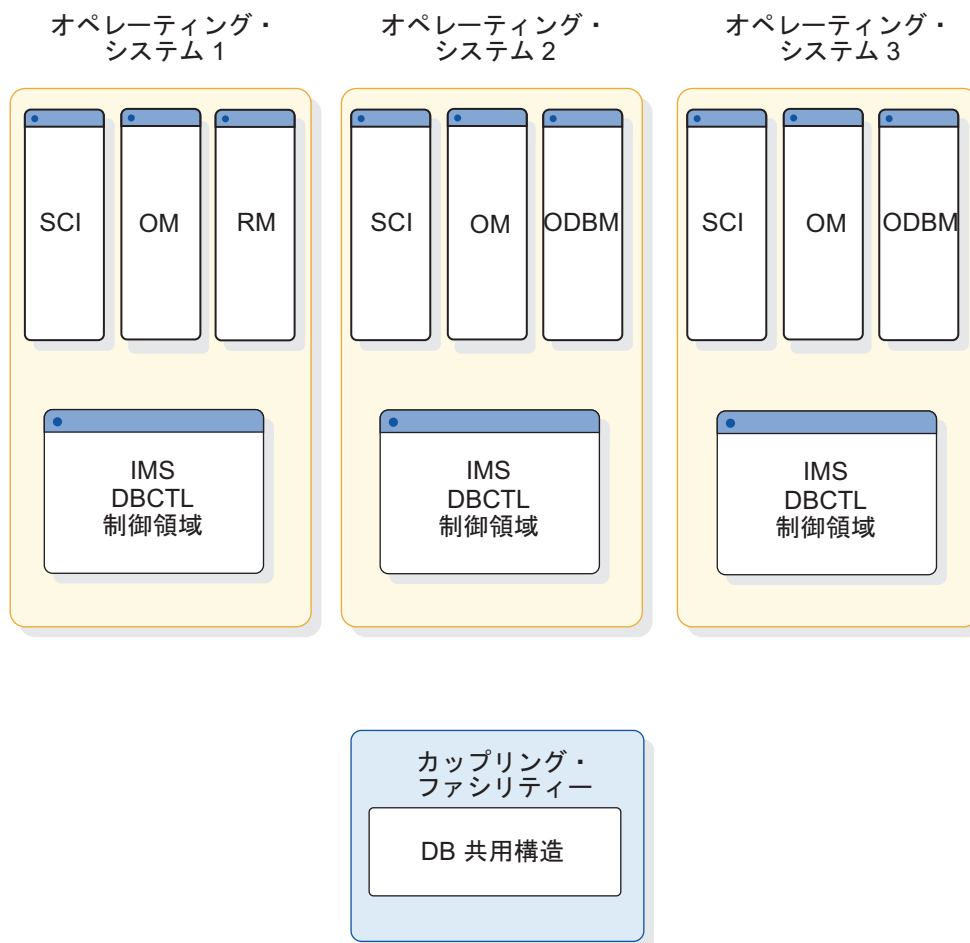


図 16. IMSplex DBCTL CSL 構成

この図では、システム全体のパフォーマンスを向上させるために、各 OS イメージで OM が定義されています。グローバル・オンライン変更を使用しない場合は、RM サービスも RM アドレス・スペースも必要ありません。そのような構成においては、CSL に SCI と OM が組み込まれます。

IMSplex 最小 ODBM 構成

次の図は、3 つの IMS システムを含んでいる IMSplex 内の ODBM の最小構成例を示しています。各 OS イメージには、IMS 制御領域、SCI、および ODBM があります。さらに、最初の OS イメージには OM (これは ODBM に必要です) がありますが、RM がありません。始動パラメーターに RMENV=NO が指定されました。

示されている ODBM 構成は、データ共用環境ではありません。各 IMS システムは、独自のデータベースを別々に管理します。

また、オペレーティング・システム 3 上のデータベースには、IMS Universal ドライバーなどの ODBM クライアントはアクセスできません。オペレーティング・システム 3 では ODBM のインスタンスが実行されていないからです。

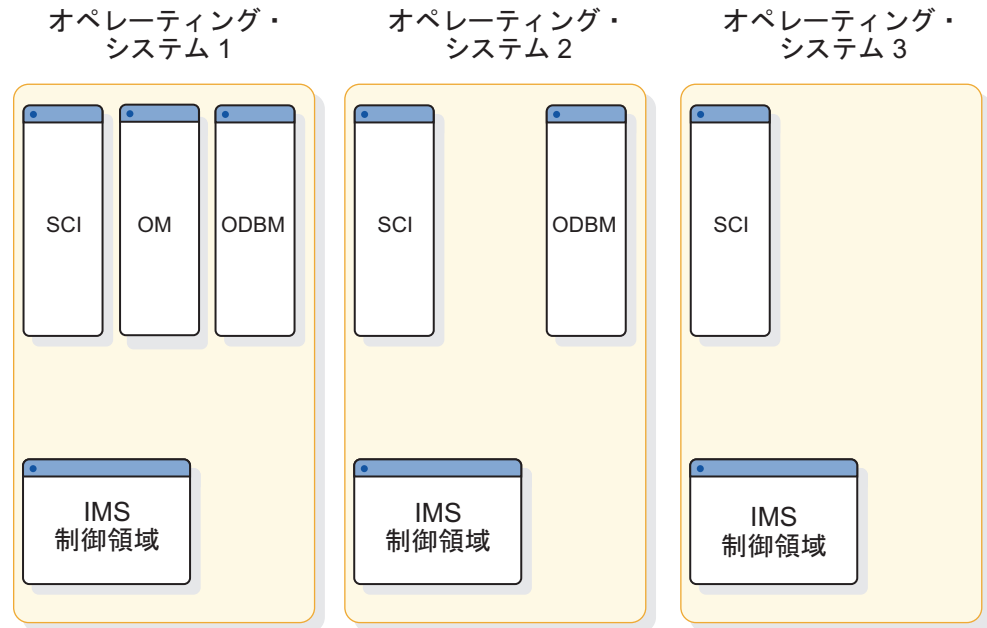


図 17. IMSplex 最小 ODBM 構成

この図では、各 z/OS イメージには個別の IMS 制御領域があるため、個別の SCI があります。OM を 1 つの z/OS イメージ上に置き、それを IMSplex 内の他のイメージで使用することができます。ODBM クライアントがアクセスする必要があるデータベースを含んでいる各 z/OS イメージ上では、ODBM のインスタンスがアクティブでなければなりません。

CSL がない IMSplex 内の共用キュー

以下の図では、3 つの OS イメージそれぞれに IMS 制御領域と IMS CQS が定義されています。それぞれの OS イメージがマスター端末オペレーター (MTO) コンソールに関連付けられています。3 つの OS イメージすべてが、データベース共用構造およびメッセージ・キュー構造が組み込まれたカップリング・ファシリティーを共用します。CSL マネージャー・アドレス・スペースは定義されていません。

この図は、CSL がない共用キュー IMSplex 環境の例を示しています。

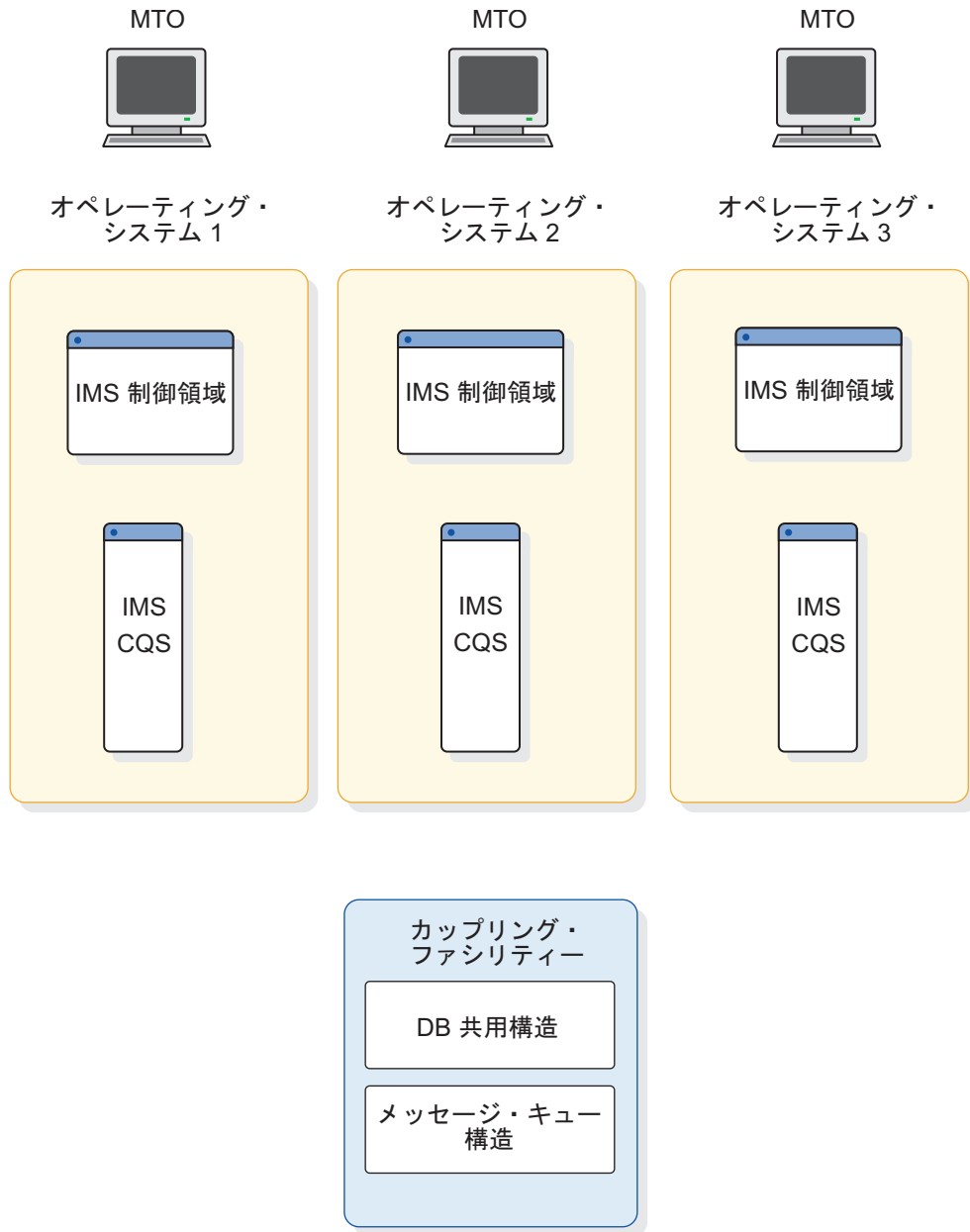


図 18. CSL がない IMSplex 内の共用キュー

CSL がある IMSplex 環境での共用キュー

以下の図では、3 つの OS イメージが定義されています。それぞれ同様に、IMS 制御領域、IMS CQS、およびすべての CSL マネージャー (SCI、OM、および RM) が組み込まれています。この 3 つの OS イメージは、データベース共用構造、メッセージ・キュー構造、およびリソース構造が組み込まれたカップリング・ファシリティを共有します。SPOC アプリケーションは、この OS イメージのどれにでもアクセスできます。

この図は、CSL がある共用キュー IMSplex 環境の例を示しています。

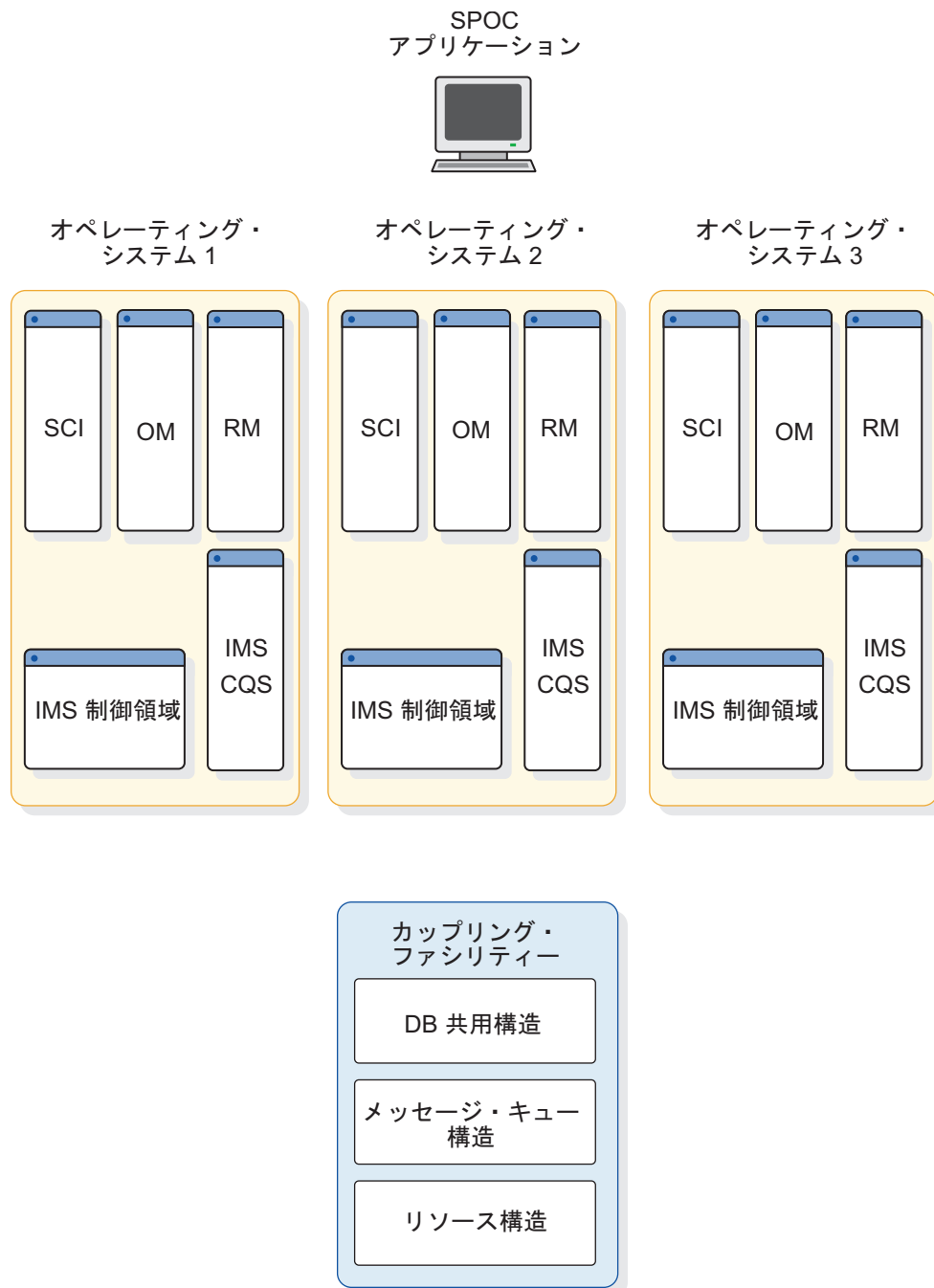


図 19. CSL がある IMSplex 環境での共用キュー

IMSRSC リポジトリを含む IMSplex 構成

以下の図は、IMSRSC リポジトリを含むサンプル IMSplex 構成を示しています。

このリポジトリを構成するコンポーネントは、CSL、OM、RM、SCI、TSO SPOC または自動化アプリケーション・プログラム、Repository Server (RS)、RS カタログ・リポジトリ・データ・セット、および IMSRSC リポジトリ・データ・セット です。

オプションで、この構成には、CQS アドレス・スペース、およびリソース構造を持つ z/OS システム間カップリング・ファシリティを含めることができます。RM がリソース構造を使用している場合、CQS は必須です。

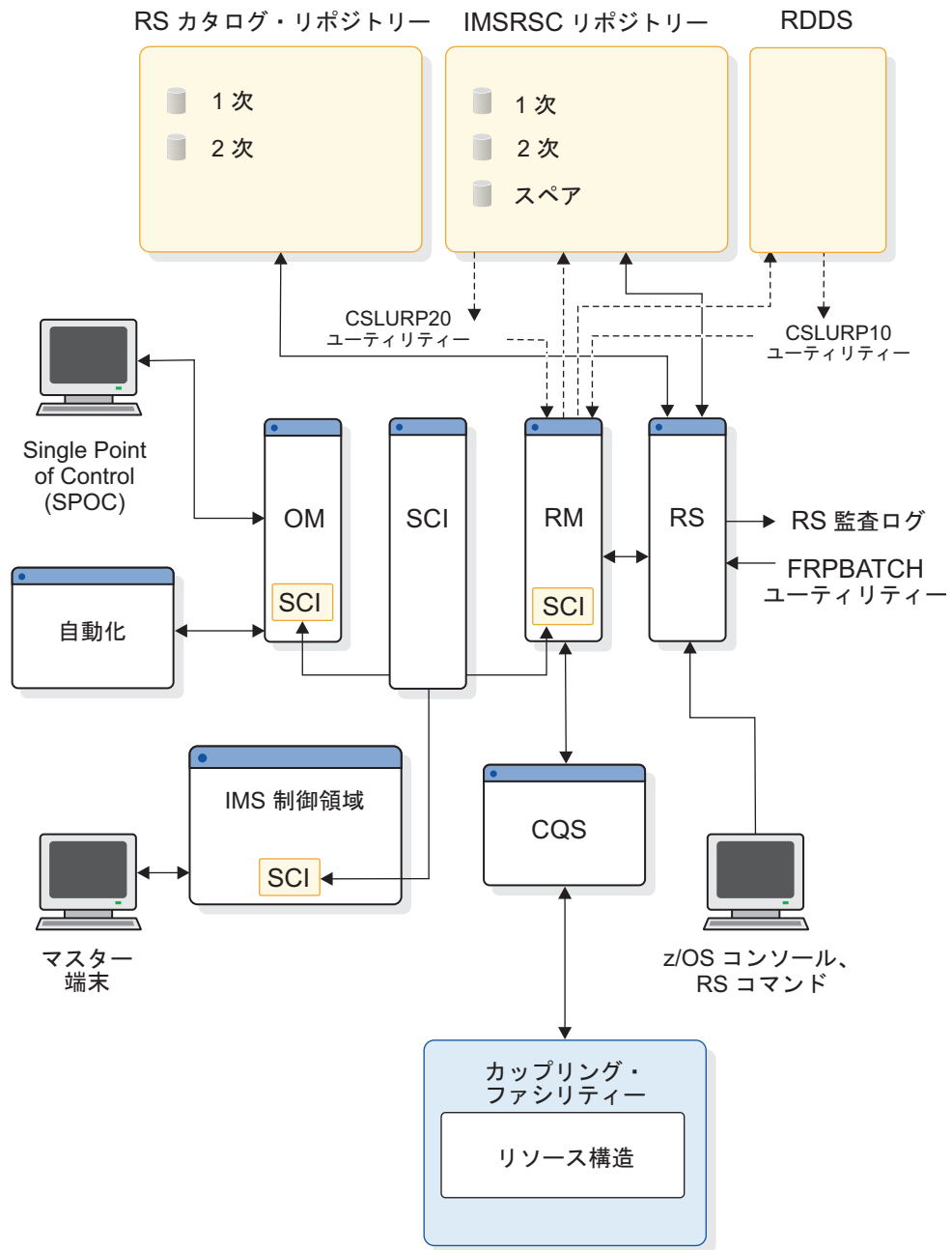


図 20. IMSRSC リポジトリを含む IMSplex 構成

IMSplex 単一システム CSL 構成

単一システム IMSplex 環境では、共通サービス層の各アドレス・スペースは単一 OS イメージで定義されます。この単純な構成を以下の図に示します。OS イメージは、複数の IMS 制御領域と、SCI、RM、OM、および ODBM を 1 つずつ用いて定義されています。

オペレーティング・システム

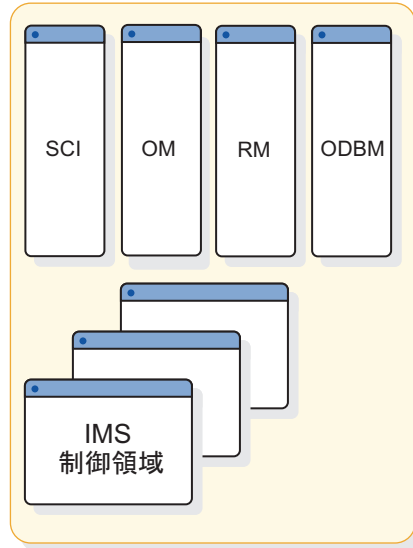


図 21. IMSplex 単一システム CSL 構成

関連概念:

- 50 ページの『Operations Manager の概要』
- 50 ページの『Resource Manager の概要』
- 51 ページの『CSL Structured Call Interface の概要』

➡ IMSRSC リポジトリーの概要 (システム定義)

関連タスク:

- 28 ページの『タイプ 2 コマンド環境のための簡易 IMSplex の定義』

Common Queue Server の概要

共通キュー・サーバー (CQS) は、カップリング・ファシリティー・リスト構造 上のデータ・オブジェクトを管理する汎用サーバーです。CQS は、キュー構造 およびリソース構造 という 2 タイプの構造を管理します。CQS は、複数クライアントのために、こうした構造からのデータ・オブジェクトを受信、保守、および配布します。IMS は、その共用キューおよび共用リソースの両方の管理に CQS を使用する CQS クライアントの一例です。

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。

CQS は、データ・オブジェクトのリポジトリーとして z/OS カップリング・ファシリティーを使用します。カップリング・ファシリティー内のストレージは、構造と呼ばれる別のオブジェクトに分割されます。許可プログラムは構造を使用し、データ共用および高速シリアライゼーションを実装します。カップリング・ファシリティーは、リスト構造に従ってデータを保管し、配置します。キュー構造には、キューとして認識されている、同じ名前を共用するデータ・オブジェクトの集合が含まれています。リソース構造には、一意の名前を持つリソースとして編成されているデータ・オブジェクトが含まれます。

CQS は、z/OS 上で稼働します。CQS クライアントも z/OS 上で稼働する必要があります。CQS は、クライアントによって開始できる、別個になったアドレス・スペース内で実行されます。

クライアントは、CQS マクロ・ステートメントがサポートする CQS 要求を使用して、CQS と通信します。これらのマクロを使用することによって、CQS クライアントは CQS と通信して、共用カップリング・ファシリティ構造上のクライアント・データを操作することができます。以下の図には、クライアント、CQS、およびカップリング・ファシリティ間の通信および関係が示されています。

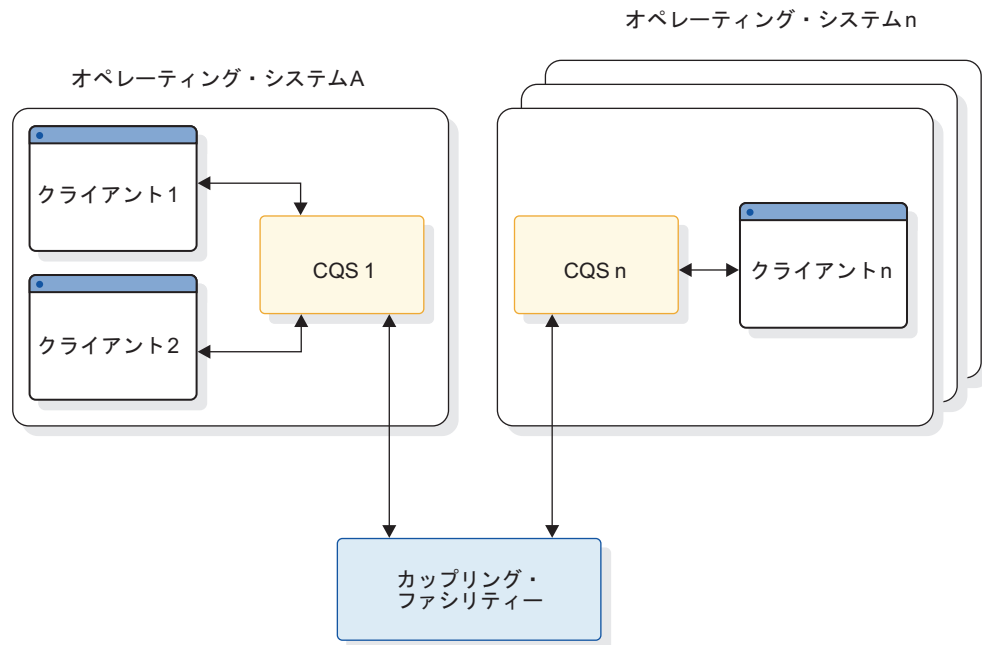


図 22. クライアント・システム、CQS、およびカップリング・ファシリティ

関連資料:

- CQS 要求については、「IMS V15 システム・プログラミング API」に説明があります。
- シスプレックスのセットアップに関する完全な詳細については、「z/OS MVS シスプレックスのセットアップ」を参照してください。

CQS の利点

CQS によって、ユーザーは並列シスプレックス (シスプレックス) 環境の利点を享受することができます。次のような利点があります。

自動的な作業負荷平準化

CQS は、任意の参加クライアント・システムが処理できるように、データ・オブジェクトを共用キューに置きます。

どの参加クライアント・システムでも、CQS を使用して共用キューからデータ・オブジェクトを検索することができます。

増加への対応

カスタマーは、作業負荷の増加に合わせて新しいシステムを追加することができます。

信頼性

共用キューおよびリソースの両方に関しては、1 つのクライアント・システムに障害が発生した場合、残りのクライアント・システムがその作業を処理します。

CQS コンポーネント

CQS は、以下のコンポーネントを使用します。

基本構造

共用キューが入る z/OS カップリング・ファシリティのリスト構造の 1 つ。

リソース構造

一意的に名前の付いているリソースが含まれている、z/OS カップリング・ファシリティ・リスト構造。

オーバーフロー構造

基本構造が、インストール・システムで指定されたオーバーフローしきい値に達したときに共用キューが入る、z/OS カップリング・ファシリティのリスト構造の 1 つ。オーバーフロー構造はオプションです。

z/OS ログ・ストリーム

構造ペアに接続されているすべての CQS からのすべての CQS ログ・レコードを含んでいる、共用 z/OS ログ・ストリーム。このログ・ストリームは、共用キューのリカバリーが必要な場合、重要です。各構造ペアは、関連するログ・ストリームを持っています。

チェックポイント・データ・セット

CQS システム・チェックポイント情報が含まれているローカル・データ・セット。

構造リカバリー・データ・セット (SRDS)

構造ペアの共用キューに対する構造チェックポイント情報が入っている共用データ・セット。各構造ペアは、関連する SRDS を 2 つ持っています。

CQS 機能

CQS は、以下の機能を実行します。

CQS 要求の処理

キュー構造またはリソース構造上のデータ・オブジェクト、または CQS にアクセスするためにクライアントが使用する、設計済みインターフェース。

キュー上の作業の通知

クライアントはインタレストを共用キューに登録します。空のキューが空ではなくなった場合、CQS は登録されたクライアントに通知します。

レコード再始動およびリカバリー情報

CQS は、z/OS システム・ロガー内の再始動およびリカバリーに必要なすべての情報を記録します。

CQS システム・チェックポイント

CQS システム・チェックポイントは、CQS ログに対する特定の CQS に関連するログ・レコードを書き込みます。ログ・レコードには、CQS の再始動およびリカバリー作業に必要な情報が含まれています。

構造チェックポイント

構造チェックポイントでは、CQS はリカバリー目的のために、キューを構造ペアから SRDS へコピーします。構造チェックポイントの開始方法については、276 ページの『構造チェックポイントの開始』を参照してください。

構造再作成

構造再作成は、構造の別のインスタンスに初期構造インスタンスから再構成された同じ名前とデータを割り振れるようにする z/OS プロセスです。

オーバーフロー処理

CQS は、キュー・フル状態を防ぐのに役立つオーバーフロー・オプションを提供します。基本リスト構造がオーバーフローしきい値に到達した場合、CQS は基本構造のサイズを動的に増やそうとするか、選択されたキューの負荷をオーバーフロー構造にオフロードするか、選択されたキューについての要求をリジェクトします。

CQS 要件

CQS には、カップリング・ファシリティーの要件があり、それらは「IMS V15 リリース計画」で説明されています。

関連資料: 下記の事項について詳しくは、「z/OS MVS シスプレックスのセットアップ」を参照してください。

- カップリング・ファシリティーを使用するシスプレックスへのマイグレーションに必要な計画
- シスプレックスのハードウェア構成
- カップリング・ファシリティーを使用できるソフトウェア・プロダクト
- 結合環境にマイグレーションするための作業
- シスプレックスのハードウェアおよびソフトウェアのインストールのためのチェックリスト

キュー構造

キュー構造は、一部のオブジェクトが同じ名前を持つデータ・オブジェクトの集合が含まれるカップリング・ファシリティー・リスト構造です。同じ名前を持つデータ・オブジェクトは、同じキュー上にあるものと見なされます。

キュー構造は、キュー構造がフルになるのを防ぐために関連するオーバーフロー構造を割り振ることができる、構造オーバーフローをサポートしています。基本キュー構造およびその関連するオーバーフロー構造は、構造ペアとして認識されます。

CQS は、キュー構造リスト・ヘッダーを、CQS が使用するための専用キュー・タイプ 11 個、クライアントが使用するためのクライアント・キュー・タイプ 11 個に物理的に分割します。クライアント・キュー・タイプは、クライアントによって定義されます。クライアントは、トランザクションなどの 1 つの作業タイプに関連

するキューをグループ化することができます。キュー・タイプは、1 から 255 までの値を持つことができます。11 を超えるキュー・タイプはすべて、物理的なキュー・タイプの 1 つにマップされます。

CQS は、キュー構造上の専用キューおよびクライアント・キューを管理します。CQS は、専用キューを使用して、CQS 要求用のクライアント・データ・オブジェクトを操作します。各クライアント・キュー・タイプを、1 つのクライアントでそれぞれ異なるタイプの作業に使用することができます。クライアントは、ユーザーがそれに関して定義した作業のタイプに基づいて、処理可能なキュー・タイプについてのみ、インタレストを登録します。

専用キュー・タイプ、CQS によって管理される

以下の表は、5 つの専用キュー・タイプとそれらに関してクライアントが処理する作業を示します。

キュー・タイプ	説明
コールド・キュー	コールド・スタートされたクライアントまたは CQS に関する、疑わしいデータ・オブジェクトが入っている
制御キュー	CQS がリスト構造と制御プロセス (構造チェックポイントや構造リカバリーなど) の管理に使用する制御リスト項目が入っている
削除キュー	CQSDEL 要求の処理に使用される中間キュー
ロック・キュー	CQSREAD 要求によってロックされたデータ・オブジェクトが入っている
移動キュー	CQSMOVE 要求の処理に使用される中間キュー

リソース構造

リソース構造は、共通サービス層の Resource Manager によって使用、CQS によって管理されている、固有名が付けられたリソースを含む、カップリング・ファシリティー・リスト構造です。

この構造は、複数の Resource Manager が IMSplex 内に存在するときにグローバル・リソース情報を保守するために、代表的に使用されます。リソース構造によって、CQS が IMSplex 内でリソース管理を実行することが可能になります。

CQS は、リソース構造リスト・ヘッダーを、CQS が使用するための専用リソース・タイプ 11 個、クライアントが使用するためのクライアント・リソース・タイプ 11 個に物理的に分割します。クライアント・リソース・タイプは、クライアントによって定義されます。リソース・タイプは、1 から 255 までの値を持つことができます。11 を超えるリソース・タイプはすべて、物理的なリソース・タイプの 1 つにマップされます。

クライアントは、リソース構造を使用して、リソース情報の共用、ブロック情報およびその他の情報の制御を行うことができます。リソース名は構造内で固有です。リソースは、更新、照会、または削除することが可能です。リソースを入れるために、基本カップリング・ファシリティー・リスト構造が使用されます。

CQS 構造機能

CQS は、構造状況および容量のモニター、および構造リカバリーを可能にする機能を提供します。これらの関数の一部は組み込み関数であり、調整は必要ありません。その他の機能はオプションで、ご使用のシステムがそれらの機能を必要としたときにセットアップまたは開始することが可能です。

構造オーバーフロー

CQS は、キュー構造がフルに近づいていて、フル構造を防ぐための処置をとるときに、自動的に警告を出す構造オーバーフロー機能を提供します。構造の使用量がオーバーフローしきい値に達した場合、CQS は構造変更を開始して、構造をより大きくしようとします。この変更が失敗した場合、CQS は、オーバーフロー構造を割り振り、選択されたキューをオーバーフロー構造に移動させるか (オーバーフロー構造を定義する場合)、新規のデータ・オブジェクトが選択されたキューに置かれることのないようにするかのいずれかを行います。

重要: オーバーフロー処理は、リソース構造に対してサポートされていません。

CQS クライアント要求 CQSPUT および CQSDEL は、パラメーター FEEDBACK= および FEEDBACKLEN= を使用することにより、現在の構造使用状況に関する情報を提供できます。IMS は、この機能を使用して、共用キューに対するキュー・スペース通知出口ルーチン (DFSQSSP0) に、IMS 共用メッセージ・キュー構造に関する構造使用状況情報を提供します。構造使用状況情報には、基本構造およびオーバーフロー構造内の割り振り済みおよび使用中の項目およびエレメントの総数が含まれています。この情報は、メッセージ・キュー構造が満杯にならないようにするのに役立ちます。DFSQSSP0 が提供する情報は、IMS 共用 EMH 構造に関するものではなく、IMS 共用メッセージ・キュー構造に関するものだけです。

構造再作成

構造再作成は、構造の別のインスタンスに同じ名前を割り振り、初期構造インスタンスから再構成されたデータが含まれるようにする z/OS プロセスです。z/OS は、z/OS が構造を再作成するシステム管理再作成をサポートしています。z/OS はまた、ユーザー管理再作成をサポートしています。ユーザーが構造を再作成します。構造再作成は、オペレーター・コマンドを使用して手動で開始することもできますし、CQS または z/OS によって自動的に開始することもできます。

CQS によって、キュー構造およびリソース構造に対するシステム管理再作成が可能になります。CQS は、構造コピー機能および構造リカバリー機能をサポートするユーザー管理再作成を提供します。

構造コピーは、計画された再構成のための別の構造に、構造の内容をコピーします。構造コピーは、リソース構造およびキュー構造に対してサポートされています。

構造リカバリーは、構造障害の後、構造チェックポイント・データ・セットおよび CQS ログから構造をリカバリーします。構造リカバリーは、キュー構造に対してサポートされています。

構造の二重化

CQS は、z/OS の二重化 機能を使用できます。二重化は、オペレーティング・システムが構造の二重化 (バックアップ) コピーを作成し、通常メインライン・オペレーションが行われている間 2 つの構造を保守しているときに起こります。構造に障害が起こった場合、または構造への接続が逸失した場合は、オペレーティング・システムは、影響を受けていない構造インスタンスへ切り替えます。

関連概念:

217 ページの『CQS 構造フルの防止』

221 ページの『CQS における構造再作成』

CQS リカバリー機能

CQS は、システム・シャットダウンまたは障害が起こったときの、処理中作業、キュー、およびリソースをリカバリーする機能を提供します。これらのリカバリー機能には標準装備のものあり、それらについては調整は必要ではありません。

その他の機能はオプションで、それらの機能が必要になったときにセットアップまたは開始することが可能です。

システム・チェックポイント

障害が起こったときに CQS 再始動を可能にするために、CQS はすべての制御ブロックおよびテーブルの「スナップショット」を定期的に作成し、その情報を z/OS ログに書き込みます。このプロセスはシステム・チェックポイント と呼ばれます。システム・チェックポイントは、CQS またはクライアントによって開始することが可能で、IMS コマンドを使用して手動で開始することも可能です。

CQS ロギングおよび z/OS システム・ロガー

CQS は z/OS システム・ロガーを使用して、CQS がキュー構造をリカバリーして再始動するために必要な情報を記録します。CQS は、使用するリカバリー可能なカップリング・ファシリティ・リスト構造ペアに関するログ・レコードを、それぞれ別々のログ・ストリームに書き込みます。ログ・ストリームは、その構造を共用するすべての CQS アドレス・スペースで共用します。システム・ロガーは、カップリング・ファシリティ・リスト構造上のキューを共用するすべての CQS アドレス・スペースに関するマージされたログを提供します。

重要: リソース構造およびリカバリー不能 (RECOVERABLE=NO) キュー構造への変更はログに記録されません。

構造チェックポイント

障害が発生したときのキュー構造リカバリーを可能にするために、CQS は、すべてのキュー構造上のキューの「スナップショット」を定期的に作成します。このプロセスは構造チェックポイント と呼ばれます。構造チェックポイントは、CQS またはクライアントによって開始することが可能で、IMS コマンドを使用して手動で開始することも可能です。

CQS 構造チェックポイントを自動的に取るよう CQS を構成するには、2 つの構造チェックポイント間のインターバルで個々の CQS アドレス・スペースに書き込み

可能な最大ログ・レコード数を指定する必要があります。この数値に到達した場合、CQS は自動的に構造チェックポイントを実行します。

重要: 構造チェックポイントは、リソース構造に対してサポートされていません。

CQS クライアント要求

CQS クライアント・システムは、CQS 要求で構成される汎用インターフェースを使用して、CQS と通信します。

関連概念:

211 ページの『CQS の始動に必要な情報の記録』

216 ページの『CQS 構造チェックポイントの使用』

 [CQS クライアント要求 \(システム・プログラミング API\)](#)

タイプ 2 コマンド環境

タイプ 2 コマンド環境では、Operations Manager (OM) および構造化呼び出しインターフェース (SCI) を備えた共通サービス層 (CSL) が必要です。Resource Manager (RM) は必要ありません。CSL を持つ IMSplex (単一 IMS の IMSplex を含みます) は、いずれもタイプ 2 コマンド環境を構成します。

タイプ 2 コマンド環境には、単一のスタンドアロン IMS システム、リソースを共用しない複数の IMS システム、または RM が組み込まれているフル機能 IMSplex を含めることができます。タイプ 2 コマンド環境は、DB/DC、DCCTL、および DBCTL 環境と共にも使用できます。


一部のタイプ 2 コマンド (オープン・データベース機能に関連したタイプ 2 コマンドなど) には、追加の要件があります。

タイプ 2 コマンド環境では、OM コマンド・セキュリティーも使用できます。タイプ 2 コマンド環境では、SCI セキュリティーも考慮する必要があります。

タイプ 2 コマンドは、IMS に付属の自動化操作プログラムのアプリケーション・プログラム (TSO 単一制御点 (SPOC) アプリケーションなど) から発行する必要があります。IMS には、そのようなアプリケーション・プログラムがIMS との通信に使用できるインターフェースを 2 つ提供します。Operations Manager (OM) アプリケーション・プログラミング・インターフェース (API) と REXX SPOC API です。

関連概念:

20 ページの『IMSplex の概要』

 [TSO SPOC アプリケーションによる IMS の制御 \(オペレーションおよびオートメーション\)](#)

 [CSL OM での REXX SPOC API 環境 \(システム・プログラミング API\)](#)

関連タスク:

28 ページの『タイプ 2 コマンド環境のための簡易 IMSplex の定義』

関連資料:

拡張端末オプション

拡張端末オプション (ETO) は IMS TM の機能の 1 つであり、システム定義に組み込むことができます。

ETO では、以下のことができます。

- システムを再定義せずに IMS に VTAM 端末を追加できます。
- ユーザー LTERM とリモート LTERM (MSC リンク用) を IMS へ動的に追加できます。
- ETO を使用してマスター端末を定義することはできません。
- ETO と一緒に XRF 監視リンクを使用することはできません。

関連資料:

- システム定義での ETO を指定する方法については、IMS V15 システム定義を参照してください。
- ETO システム・セキュリティー機能については、375 ページの『第 19 章 IMS のセキュリティー』を参照してください。

APPC

IMS は、2 つのシナリオで拡張プログラム間通信機能 (APPC) 会話をサポートします。APPC/IMS と明示的な CPI-C ドリブン・インターフェースです。

この 2 つの方法は、アプリケーション・プログラムがアクセスする保護リソースの更新と同期を管理するサブシステムが異なります。

APPC/IMS の方法では、SYNCLVL = NONE または SYNCLVL = CONFIRM であれば、IMS が同期点マネージャーになります。SYNCLVL = SYNCPT の場合、z/OS リソース・リカバリー・サービス (RRS) が同期点マネージャーになります。

CPI-C ドリブンの方法では、次のようになります。

- 同期点マネージャーは、z/OS の RRS 機能です。
- Resource Manager は IMS
- 保護リソースのアクセスと更新を行うプログラムは、APPC/MVS アプリケーション・プログラム

APPC/IMS はプログラム間通信用の形式とプロトコルを定義します。APPC/IMS を使用すると、アプリケーションをネットワーク全体に分散し、基礎となるハードウェア・アーキテクチャーとソフトウェア環境に関係なく互いに通信させることができます。APPC/IMS は論理装置タイプ 6.2 (LU 6.2) サポートを実施するための機能を提供します。

リモート LU 6.2 装置またはサブシステムとのすべての対話には、APPC/MVS が使用されます。IMS は APPC/MVS サービスを通してセッションにアクセスします。IMS と LU 6.2 装置は APPC/IMS を使用して互いにアクセスし、既存アプリケーション・プログラムのコーディング変更を必要としません。IMS アプリケーシ

ョン・プログラムにわずかな修正を加えることによって、共通プログラミング・インターフェース (CPI) 通信ドリブン・アプリケーション・プログラムは IMS アプリケーション・プログラムと通信できます (また、IMS アプリケーション・プログラムとして実行できます)。


ただし、暗黙トランザクションとの LU 6.2 同期会話には制限があります。トランザクションが複数のドーター・トランザクションを生成し、次にそれらが他のトランザクションを生成し、ドーター・トランザクションの 1 つが応答を提供する可能性があるような場合、その結果は予測不能となります。場合によっては、トランザクションの実行シーケンスによって、LU は DFS2082 メッセージを受け取り、その応答はデフォルトの TP 名 DFSASYNC に送られます。その他の場合は、LU が応答を受け取り、DFS2082 メッセージは出されません。

APPC/IMS フラッディング制御

APPC/IMS には、オプションのフラッディング制御機能が組み込まれています。この機能は、アクティブな APPC 会話の数が DFSDCxxx PROCLIB メンバーの APPCMAXC= パラメーターに定義されているフラッディングしきい値を超えた場合に、64 ビット・ストレージで APPC 要求をキューに入れます。デフォルトのしきい値は 5000 アクティブ APPC 会話です。

IMS は、64 ビット・ストレージ内でキューに入れられた APPC 要求の数が APPCMAXC= パラメーターの 2 番目の値位置で定義されている最大数に達した場合、あるいは 64 ビット・ストレージ内でのキューイングが使用不可で、アクティブ APPC 会話の最大数に達した場合に、APPC/IMS と CPI-C の両方についての入力を含め、z/OS からのすべての APPC 入力を停止します。

関連情報:

 3303 (メッセージおよびコード)

従属領域処理のセキュリティー

端末、トランザクション、コマンド、およびその他のタイプの許可別にセキュリティー検査を実行できますが、従属領域内でスケジュールされたアプリケーション・プログラムがリソース・アクセス・セキュリティー (RAS) を使用してアクセスできるリソースを制限することによっても、セキュリティーを実装できます。

RAS は、RACF[®]、RACF セキュリティー・クラス、およびユーザー ID を使用して、リソースとそのリソースを使用できる従属領域を定義します。RAS セキュリティーを実装するには、RACF セキュリティー・クラスで、保護したいトランザクション、PSB、および LTERM のためのリソース・プロファイルを定義する必要があります。リソース・プロファイルで、各リソースの使用を許可したい各従属領域のユーザー ID を指定することも必要です。

RACF は、IMS に対する外部セキュリティー製品であり、セキュリティー・アクセス機能 (SAF) を使用して IMS からアクセスされます。RACF は、IBM z/OS Security Server と一緒にライセンスが交付されます。本書で RACF の使用を指示している場合、他の同等のセキュリティー製品を選択して使用することもできます。

従属領域で実行するアプリケーション・プログラムがリソースにアクセスしようとすると、RACF は、リソースのセキュリティー・クラス・プロファイルを検査して、アプリケーションが常駐している従属領域のユーザー ID がそのリソースについて許可されているかどうかを確認します。リソース・プロファイルにそのユーザー ID がリストされている場合は、RACF はアクセスを許可しますが、そうでない場合は、RACF はアクセスを拒否します。

RAS セキュリティー付きのリソース・アクセス・セキュリティー・ユーザー出口 (RASE) などの出口ルーチンも使用できます。これらの出口ルーチンでは、従属領域処理用のセキュリティー検査のカスタマイズが可能です。

DB/DC 環境および DCCTL 環境での MPP スケジューリング

MPP 領域の初期化が完了すると、その仮想ストレージ内にあるアプリケーション・プログラムを実行できます。スケジューリング・アルゴリズムを使用することにより、制御プログラムはプロセス用のメッセージを選択します。トランザクション・コードを使用して、適切なアプリケーション・プログラムが IMS.PGMLIB データ・セットから従属領域ストレージへロードされます。

アプリケーション・プログラムは、システム定義内でそのトランザクション・コードに関連付けるよう宣言された PSB 名によって識別されます。MPP 用に IMS TM が使用する規則は、アプリケーション・プログラム名が PSB 名と同じものであるということです。その後、最初のメッセージ・セグメントがメッセージ・キューから使用可能にされ、アプリケーション・プログラムへ制御が渡されます。

スケジューリング・アルゴリズムは、アプリケーション・プログラムが実行するプロセスの量も制御します。1 つのプログラムのスケジューリングの中で処理されるメッセージの数の限界を指定することができます。この数値に到達した場合、IMS TM は次のことを実行します。

- 優先順位が同じかそれ以上のトランザクションがキューに入っている場合、IMS TM はアプリケーション・プログラムを終了します。領域は、別のプログラムをそのストレージへスケジュールできるようになります。IMS はスケジューリング・アルゴリズムを使用して、スケジュールするプログラムを選択します。
- 優先順位が同じかそれ以上のトランザクションがキューに入っておらず、メッセージがまだ現行アプリケーション・プログラム用にキューの中に入っている場合、領域では高速スケジュール変更が行われ、続くメッセージをアプリケーション・プログラムへ戻します。
- 等しい優先順位のトランザクションがキューに入っている場合、PROCLIM 値に達していなければ、IMS はトランザクションの高速スケジュール変更を許可します。PROCLIM 値に達している場合は、IMS は高速スケジュール変更を許可しないで、その他のトランザクションを処理します。

高速スケジュール変更は、IMS のスケジューリング・アルゴリズムの影響も受けれます。このアルゴリズムでは、以下の要因 (およびその他) が考慮されます。

- MAXRGN 値
- PARLIM 値
- トランザクションがスケジュールされている領域の現行数

- スケジュールされたトランザクション用にそれ以上メッセージが存在しない場合、IMS TM はその領域用にプロセス可能な別の作業がないか判別します。
- プロセス可能な状態の別の作業がない場合、IMS TM はその領域を疑似入力待ち (疑似 WFI) 状態にできるかどうかを判別します。この判別によって、次のいずれかの処置が実行されます。
 - 領域を疑似 WFI 状態にできる場合、その領域はトランザクション用にスケジュールされた状態のまま残り、その領域用に別のメッセージが入力されるまで待機します。次のメッセージが、スケジュールされたトランザクション用のものである場合、そのメッセージはアプリケーション・プログラムへ渡されます。次のメッセージが別のトランザクション用のものである場合、IMS TM はアプリケーション・プログラムを終了し、その新しいメッセージを処理する新しいアプリケーション・プログラムをスケジュールします。
 - 領域を疑似 WFI 状態にできない場合、IMS TM はアプリケーション・プログラムにそれ以上メッセージが存在しないことを知らせ、アプリケーション・プログラムは終了します。

マスター端末オペレーターはバッチ・メッセージ領域を始動する JCL の項目によって、バッチ・メッセージ・プログラムを直接スケジュールします。使用するプログラムと PSB は、EXEC ステートメントの中で明示的に指定します。

使用不能なデータに対するアプリケーション・プログラムのスケジュールリング

IMS Transaction Manager は、あるアプリケーション・プログラムがアクセスできる全機能データベースの一部が使用不能な場合でも、そのアプリケーション・プログラムをスケジュールします。アプリケーション・プログラムは、使用不能なデータを処理するときに、データが使用不能であることにセンシティブである場合もセンシティブでない場合もあります。

使用不能なデータにセンシティブであるアプリケーション・プログラムは、IMS がアプリケーションをスケジュールするときに INIT 呼び出しを出します。INIT 呼び出しは、アプリケーション・プログラムに必要なデータが使用不能である場合には状況コードを PCB に入れて戻すように IMS に指示します。これにより、プログラムは適切な処置を実行できます。

使用不能なデータにセンシティブでないアプリケーション・プログラムは、INIT 呼び出しを出しません。使用不能なデータにセンシティブでないアプリケーション・プログラムが使用可能でないデータにアクセスしようとする、IMS はユーザー異常終了コード 3303 でアプリケーション・プログラムを終了し、そのアプリケーション・プログラムが行った更新をバックアウトします。アプリケーション・プログラムが 3303 異常終了を 10 回生成すると、IMS は、その PSB を停止することによって、アプリケーション・プログラムをそれ以上再スケジュールしないようにします。

データが使用不能であるためにアプリケーション・プログラムが終了した場合、IMS は、そのアプリケーション・プログラムが処理していた入力メッセージを延期キューに入れます。延期キューは、トランザクションのタイプごとに別個のものが

存在します。順次トランザクションは、延期キューには入れられず、次に処理されるメッセージとして通常のキューに戻され、USTOPPED 状態になります。

IMS は、ほとんどのメッセージが失敗して延期キューに置かれていると判別した場合、そのトランザクション・タイプのプロセスを停止します。そのトランザクションが開始されたとき、またはそのトランザクションのプロセスの中で使用されるデータベースが始動されたときに、該当の延期キューに入っているメッセージが通常のキューへ転送され、あらためてそのメッセージのプロセスが試みられます。

関連資料:

- 使用不能なデータに対するアプリケーション・プログラムのスケジューリングの追加情報については、「IMS V15 アプリケーション・プログラミング」を参照してください。
- 延期キューについて詳しくは、「IMS V15 システム定義」の『延期キューを使用したトランザクションのスケジューリング』を参照してください。

高速機能 (Fast Path)

高速機能は、単純なトランザクションのパフォーマンスを改善するために使用します。データ通信要件が、データベースの高速な更新と照会を伴った大量のトランザクション用のものである場合、高速機能は全機能 DL/I プロセスにいくつかの利点を提供します。

これらの要件を持つアプリケーション・プログラムの例としては、金融機関業務処理でのテラー・トランザクションや小売業での POS トランザクション (在庫更新) があります。高速機能の入出力メッセージは、急送メッセージ・ハンドラー (EMH)、バイパス・メッセージ・キューイング、および優先順位スケジューリングを使用します。ほとんどの端末は高速機能を実行する能力を備えています。しかし、応答モードで実行できない端末は、高速機能を実行する能力がありません。

DB/DC 環境では、高速機能は Database Manager と Transaction Manager を必要とし、IMS オンライン・システムの一部を構成します。制御プログラムは、高速機能と DL/I プログラムの並行処理を管理します。

DCCTL 環境では、高速機能処理およびトランザクションをサポートしますが、高速機能データベースはサポートしません。

関連資料:

- 高速機能で使用するデータベースの設計、定義、初期設定、モニター、チューニングについて詳しくは、「IMS V15 データベース管理」を参照してください。
- 高速機能アプリケーション・プログラミングについては、「IMS V15 アプリケーション・プログラミング」を参照してください。

高速機能データベース

DL/I データベースに加えて、高速機能ではその他の 2 つのデータベース・タイプ、つまり主記憶データベース (MSDB) と高速処理データベース (DEDB) を使用できます。これらのデータベースは、高い可用性を必要とするアプリケーション・

プログラム用に設計されています。これら 2 つのタイプによって、高いアクティビティ内での高速応答と、容量の大きいデータ内での区分アクセスとを選択できます。

関連資料: これらの高速機能データベースの設計の利点と実装については、「IMS V15 データベース管理」を参照してください。

高速機能のための従属領域使用

高速機能処理プログラムの大多数は、メッセージ処理プログラム (MPP) によく似た機能を備えています。メッセージ・ドリブン・プログラムは MPP に対応し、高速機能従属領域 (IFP 領域) 内で実行されます。これらのプログラムは入力待ちモードで実行されるため、プログラムの実行は従属領域操作と等価です。並列スケジューリングがサポートされているため、別の従属領域でプログラムの別のコピーを実行することもできます。

再編成機能やリカバリー関連機能など、高速処理データベースの保守の大部分をオンラインで実行できるため、IMS オンライン・システムは高速機能ユーティリティ領域のスケジューリングを考慮したものでなければなりません。

高速機能のアプリケーション・プログラムとユーティリティは、メッセージ処理プログラムや BMP と並行してアクティブにできます。複数システム結合機能 (MSC) を使用する IMS オンライン・システムでも高速機能トランザクションを処理することができます。ただし、MSC リンクを通して受け取ったメッセージ入力を高速機能アプリケーション・プログラムへ宛先指定したり、高速機能入力出口ルーチンへ渡したりすることはできません。この制約事項は、システム間連絡 (ISC) 接続を使用して受け取ったメッセージ入力には適用されません。

高速機能トランザクション

高速機能アプリケーション・プログラムは、IMS 入力メッセージ・キュー処理をバイパスするトランザクションによって駆動されます。トランザクションは、高速機能専用トランザクションとして宣言できます。初期編集の後、入力メッセージは出口ルーチンへ渡されます。このルーチンは、トランザクションが実行される従属領域を判別するのに役立ちます。メッセージはプログラムのストレージ内にある高速機能メッセージ処理領域へ追加され、その後、メッセージ・キューへの入出力なしにメッセージ・ドリブン・プログラムから使用可能になります。

トランザクションは、高速機能利用可能として宣言することもできます。入力後、トランザクションはユーザー出口ルーチンへも渡され、ユーザー出口ルーチンはトランザクションを制御プログラム・ストレージのメッセージ保持領域へ直接渡すべきか、それとも通常のメッセージ・キュー処理として IMS へ送るべきかを決定します。キューのバイパスの結果、再びトランザクションがメッセージ・ドリブン・プログラムへ提示されます。

制御プログラムのストレージ内での高速機能メッセージの制御は、急送メッセージ処理と呼ばれます。いずれかの検査により、メッセージが、単一セグメント入出力メッセージを使用する要件を満たしていることが保証されます。高速機能入力編集/経路指定出口ルーチンは DBFHAGU0 です。IMS は、高速機能と全機能の両方のトランザクションの完全な入力編集のために EMH バッファを使用できます。

DBCTL 環境での高速機能

DBCTL 環境では、高速機能は高速処理データベース (DEDB) を使用できるプログラムのパフォーマンスとデータ可用性を改善します。DBCTL 環境は、DEDB に関連した高速機能の諸機能だけをサポートします。主記憶データベース (MSDB) 機能を実行することはできません。

BMP または CCTL スレッドは、DEDB へアクセスするために PSB をスケジューリングできます。並列スケジューリングがサポートされているため、別の BMP または CCTL スレッドで PSB の別のコピーを実行することもできます。高速機能のアプリケーション・プログラムとユーティリティーは、BMP と並行してアクティブにできます。

DEDB の保守の大部分 (再編成やリカバリー関連機能など) はオンラインで実行できるため、IMS DBCTL 環境は高速機能ユーティリティー領域のスケジューリングを考慮したものでなければなりません。

自動化操作プログラムのアプリケーション・プログラム

DL/I の CMD 呼び出しか ICMD 呼び出しを使用して IMS オペレーター・コマンドのサブセットを発行できるアプリケーション・プログラムは、自動化操作 (AO) を行うアプリケーション・プログラムと呼ばれます。

AO アプリケーション・プログラムは、DB/DC 環境および DCCTL 環境でのみ CMD 呼び出しを使用できます。

AO アプリケーション・プログラムは、すべての環境で ICMD 呼び出しを使用できます。

CMD 呼び出しまたは ICMD 呼び出しを発行した場合、そのオペレーター・コマンドが実行され、コマンド応答の最初のセグメントが AO アプリケーション・プログラムの入出力域に置かれます。それに続く応答の部分は、GCMD 呼び出し (CMD を発行した場合)、または RCMD 呼び出し (ICMD を発行した場合) によって入手されます。

セキュリティを維持するために、どの AO アプリケーション・プログラムがオペレーター・コマンドを実行でき、どのコマンドを実行できるかを定める必要があります。AO アプリケーション・プログラムは、単一のコマンドか一連のコマンドを実行できます。

関連資料:

- CMD 呼び出しおよび ICMD 呼び出しの保護については、375 ページの『第 19 章 IMS のセキュリティ』を参照してください。
- 自動化操作 (AO) を行うアプリケーション・プログラムについて詳しくは「IMS V15 オペレーションおよびオートメーション」を参照してください。

AO アプリケーション・プログラムを使用する場合は、IMS AO 機能をデータベース・リカバリー・サービスと併用して、データベースおよびエリアのリカバリーを自動化することも可能です。リカバリーを制御するには、タイプ 2 の AO アプリケーション・プログラムで ICMD 呼び出しを使用して /RECOVER コマンドを実行する方法もあります。

関連資料: /RECOVER コマンドの詳細については、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

システム・ロギングとプロセス継続性

データの保全性を保護するため、オンライン IMS は外部のセキュリティー検査と各種の内部的な技法の両方を使用して、システムへ入力されたトランザクションとデータベース更新アクティビティーを記録します。

オンライン・システム・アクティビティーを記録する基本ツールは、IMS システム・ロギングです。各種のロギング・データ・セットに格納されたデータには、再始動、リカバリー、統計、監査の各目的で使用される情報が入っています。

IMS ログ・データは、次の 4 種類のデータ・セットに記録されます。

- オンライン・ログ・データ・セット (OLDS)
- 先行書き込みデータ・セット (WADS)
- 再始動データ・セット (RDS)
- システム・ログ・データ・セット (SLDS)

オンライン・システムは最低でも 3 つの OLDS と 1 つの WADS、単一の RDS を使用し、これらはすべて DASD 上にだけ常駐します。1 つ以上のオンライン・ログ・データ・セットがいっぱいになった場合は、IMS ログ保存ユーティリティーを使用してそれらのデータ・セットをシステム・ログ・データ・セットへ保存できます。SLDS には DASD またはテープ・メディアを使用できます。バッチ・システムはシステム・ログ・データ・セットを使用し、テープと DASD のどちらにでもログを記録できます。

オンライン・システムは OLDS を循環方式で使用します。OLDS の重複ロギングがインストール要件である場合は、1 対のデータ・セット (1 次と 2 次) を割り当てなければなりません。OLDS の DD 名は、1 次データ・セットの場合には文字ストリング DFSOLP、2 次データ・セットの場合は DFSOLS でそれぞれ始まりま
す。8 文字からなる DD 名は、「OLDS ID」と呼ばれる固有の接尾部 (00 から 99) で終わります。システムの初期設定時に DD ステートメントによって決定されたか、DFSDFxxx PROCLIB メンバーの LOGGER セクションに含まれる命令によって決定されたとおりに、単一ロギングまたは重複ロギングのいずれかが実行されます。

WADS は、OLDS バッファーに入っていて OLDS へまだ書き込まれていないログ・レコードのコピーが入っている小さなデータ・セットです。DASD へログを記録する場合 (オンライン処理に必要)、固定長ブロックならば直接検索が容易になります。WADS では (さまざまなブロック形式の) 大きな固定長ブロックを OLDS へ書き込むことができ、ブロックを再書き込みする必要がありません。ログ・データがすでに OLDS へ書き込まれている場合は、WADS が再利用されます。

システム障害が発生した場合、WADS 内のログ・データを使用して OLDS がクローズされます。このクローズ・プロセスは緊急時再始動の一部として発生するか、ログ・リカバリー・ユーティリティーのオプションの 1 つとして発生します。

IMS では、最大 10 個の WADS が許容され、そのうちの 1 つまたは 2 つのみがアクティブで、残りは予備です。書き込みエラーが検出された場合、エラーを検出した WADS が予備の WADS に置き換えられます。WADS から OLDS がクローズされるときに読み取りエラーが起きた場合に備えてバックアップを作成しておく必要がある場合は、重複 WADS ロギングもサポートされます。

オンライン IMS システムは、始動に使用されるログ・データ・セットを制御します。オンライン IMS システムは、再始動データ・セットへ書き込まれたチェックポイント識別テーブル内の項目と、DBRC RECON データ・セット内に記録されたログ・データ・セット情報を利用します。自動再始動を使用している場合は、システム・コンソールから発行された /START IMS コマンドにより適切な種類の再始動が実行されます。通常、この再始動では結果的に OLDS レコードが使用され、以前のシステム・ログを使用せずに正常再始動が完了します。再始動処理が初期チェックポイントの前で異常終了した場合、自動再始動に適した再始動は、異常終了した再始動と同じタイプ (/NRESTART または /ERESTART) です。

チェックポイント操作

チェックポイント操作は、中断された操作の再始動に使用できる情報を記録するために IMS が使用する基本技法です。チェックポイントのときに記録された状況情報を使用して、IMS はメッセージ・キューとデータベース変更の内容を復元します。

チェックポイントは、システムのシャットダウンと始動に不可欠の部分です。また、チェックポイント操作を適切な間隔で行っている場合、システム障害の前のチェックポイントと継続点より後のチェックポイントの間で行われる、再処理の量が削減されます。一部のプロセス・オーバーヘッドはチェックポイント情報に関連したのですが、これはシステムの効率的な再始動のために許容できるトレードオフです。

XRF 複合システムの中で、アクティブ IMS システム上で取得された SNAPQ チェックポイント・レコードは、同期フェーズ中に代替 IMS システム上に制御ブロックを構築するために使用されます。

IMS 内部チェックポイントは、前もって決められたインターバルで自動的に発生するようにスケジュールされます。このインターバルは、作成されるシステム・ログ・レコードの数の増分によって指定されます。オンライン IMS イベントが個々のログ・レコード・タイプとともにログに記録されるときに、カウントが保持されます。増分が指定された値を超えた時点で、チェックポイント処理が呼び出されます。IMS チェックポイントは、/CHECKPOINT コマンドの発行を許可されたアプリケーション・プログラムとマスター端末オペレーターから明示的に呼び出すこともできます。

高速 DB リカバリー領域は、IMS システムのチェックポイント・レコードをモニターし、データベース・リソース・リカバリーでそれらを使用します。

DB/DC および DCCTL でのロッキング・メカニズムとデータベース保全性

IMS ではロックを選択することができ、プログラム分離 (PI) ロッキングか、内部リソース・ロック・マネージャー (IRLM) のサービスを使用することができます。IRLM コンポーネントは、データ共有の不可欠の部分として使用されます。

プログラム分離により、DB/DC 環境でアクティブ状態のアプリケーション・プログラムのすべてのアクティビティ (データベースの変更とメッセージの作成) は、システムでアクティブ状態のその他のどのアプリケーション・プログラムからも分離されます。この分離は、前者のアプリケーション・プログラムが同期点に到達することによって自らが修正または作成したデータが有効であることを確認するまで持続します。

注: PI ロック・マネージャーは、IMS システムで最大 63 の待機アプリケーション・プログラムをサポートします。ロックを待機する必要があるアプリケーション・プログラムが PI ロック・マネージャーのサポートする最大を超えた場合、アプリケーション・プログラムは異常終了 2478 を出して終了し、IMS によってバックアウトされ、ロックは解放されます。リクエストがメッセージ・ドリブン・プログラム (MPP、JMP、IFP、または BMP) である場合、メッセージはキューに返され、再処理されます。

ロッキング・メカニズムは、次のことにも使用されます。

- 異常終了したアプリケーション・プログラムの影響を除去する
- ROLL、ROLB、ROLS のいずれかの呼び出しに必要となるプロセスを行う
- デッドロック状態を解決する

上記のすべてのプロセスについて、データベース更新と保留出力メッセージの除去が前の同期点から現在の状況まで実行されます。同期点 は、アプリケーション・プログラムを再始動できる時点として定義されます。アプリケーション・プログラム用のそのような最初の点は、初期スケジューリングです。最も一般的な同期点は、メッセージ・キューへの GU が発生したときです。単一メッセージ・モードのプログラムは、次のメッセージを求める要求を発行することにより、プロセス・サイクルの開始と前の作業の完了を示します。このとき、一時的な宛先へのキューに入れられた出力メッセージが最終的な宛先へ送られ、データベース更新がコミットされます。

アプリケーション・プログラムは、同期点を強制する CHKP 呼び出しを発行することもできます。複数メッセージ・モードで実行されているアプリケーション・プログラムか、トランザクション駆動でない BMP の場合、同期点は初期スケジューリングか直前の CHKP 呼び出しの時点です。

プログラム分離のもう 1 つの局面は、セグメント発生レベルでのデータベース更新の制御です。スケジューリング・プロセス中に、IMS はアプリケーション・プログラムの使用するデータベースに対する意図を分析します。アプリケーション・プログラムがデータベースの排他使用を必要とすることが原因で、現在スケジュールされているトランザクションと、これからスケジュールしようとするトランザクションのデータベース使用状況に競合が存在する場合、スケジューリング・プロセスは別のトランザクション・コードを選択して再試行しなければなりません。排他意図が要件でない場合 (通常はそうです)、アプリケーション・プログラムは並行してス

ケジュールされます。IMS はロッキング・メカニズムを使用して、データベース・セグメントのインターリーブされた所有権を制御します。アプリケーション・プログラムは実行の際、データベース・レコード上でエンキューし、更新後またはアプリケーション・プログラムが同期点に到達したときに、それらのリソースを解放します。

発生のある可能性があるデッドロック状態は、アプリケーション・プログラムと端末オペレーターには認識されない方法で解決されます。IMS がデッドロック状態を検出すると、そのデッドロックに関与しているアプリケーション・プログラムの 1 つが特殊な異常終了コードで異常終了します。異常終了の結果、終了したプログラムのアクティビティは前の同期点まで動的にバックアウトされます。そのプログラムの保留リソースは解放されます。これにより、他のアプリケーション・プログラムは処理を完了することができます。特殊コードによって、処理中だったトランザクションが保管されます。そのアプリケーション・プログラムはスケジュール変更されます。

DBCTL では、デッドロック状態によって強制的に CCTL スレッドが異常終了させられた場合、そのスレッドは保管されず、DBCTL によって再試行されません。CCTL は、ある特定のデッドロック終了コードを受け取ると、そのトランザクションを再試行します。

動的にバックアウトするために BMP が選択された場合、その BMP はスケジュール変更できず、最新の同期点で終了します。BMP が入力を求めてメッセージ・キューへアクセスしなかった場合、または CHKP 呼び出しを発行しなかった場合、BMP はスケジュールリングのときに終了し、すべての BMP データベース更新アクティビティは無効にされます。

IMS DB のシャットダウン時または異常終了時に ODBA スレッドがアクティブであると、ODBA アプリケーションのスレッドは終了します。ODBA アプリケーション・プログラムは終了しませんが、これ以上スレッド上で呼び出しを行うことはできません。

出口点および出口ルーチン

特別な処理を実行するサブルーチン (出口ルーチン と呼びます) を IMS に提供することにより、IMS で作業単位をどのように処理するかを変更することができます。IMS は、ロジック・フローのさまざまな点 (出口点 と呼びます) で出口ルーチン呼び出しを行います。これにより、IMS でその作業をどのように実行するかを制御することができます。

IMS で作業単位をどのように処理するかを変更する例として、(MSC 環境で) 未完了メッセージを問い合わせ、その発信元がテスト・システムであるか実動システムであるかを判別する場合などがあります。テスト・システムから発信された場合、テスト・システムに関するデータを収集するアプリケーションに、そのメッセージを経路指定する必要があります。メッセージが実動システムから発信された場合は、おそらくその宛先を変更する必要はありません。

メッセージの問い合わせを行うには、TM および MSC メッセージ経路指定および制御ユーザー出口ルーチンをコーディングします。このルーチンに DFSMSCEO という名前を付けて、これを IMS.SDFSRESL ライブラリーに配置して、そのライブ

ラリ (または連結ライブラリ) にバインドすると、IMS でのメッセージの受信時に、IMS はその出口ルーチン呼び出します。出口ルーチンは、その処理を完了すると制御を IMS に返し、次に IMS は、元の宛先またはテスト・アプリケーションへのメッセージの処理を再開します。

出口ルーチンには、必須のものとおプションのものがあります。IBM 提供の出口ルーチンには、そのまま使用できるものと使用前に変更が必要なものがあります。

関連資料: 出口点および出口ルーチンの詳細については、「IMS V15 出口ルーチン」を参照してください。

データ・キャプチャー出口ルーチン

ご使用のシステムで IMS DB と Db2 for z/OS の両方のデータベースを使用する場合、IMS DL/I と Db2 for z/OS のリレーショナル・データベースのデータを重複させることが必要な場合があります。

例えば、構造化照会言語 (SQL) で書かれた Db2 for z/OS アプリケーション・プログラムが、IMS DB データベースからのデータを必要とする場合があります。また、徐々にサイトを Db2 for z/OS に変換している場合や、Db2 for z/OS のリレーショナル・テクノロジーを何らかの IMS データに利用したい場合などが考えられます。

2 つのタイプのデータベース間でデータを重複させるには、データ・セグメントへの更新が、確実に両方のデータベースに対して適切な時点で行われるようにしなければなりません。IMS DB データベースから Db2 for z/OS データベースへ更新を複製するプロセスは、データ伝搬と呼ばれます。

IMS DB から Db2 for z/OS へデータを伝搬させる方法には、次の 2 つがあります。

- IMS DataPropagator は、データ伝搬と出口ルーチンのサポートを提供する IBM ライセンス・プログラムです。
- データ・キャプチャー出口ルーチン。これは、データ伝搬用のルーチンを確立するためにユーザーが作成する出口ルーチンです。これはアセンブラ言語、C 言語、COBOL、または PL/I のいずれかで作成でき、データ伝搬を必要とするアプリケーション・プログラムから呼び出されます。

注: データ・キャプチャー出口ルーチンは、CICS からは使用できません。DBCTL はこの出口ルーチンを使用できますが、BMP 用にだけ使用できます。

関連資料:

- データ伝搬のシステム要件については、「IMS V15 システム定義」を参照してください。
- データ・キャプチャー出口ルーチンに関連したデータベースの考慮事項については、「IMS V15 データベース管理」を参照してください。
- データ・キャプチャー出口ルーチンの作成方法については、「IMS V15 出口ルーチン」を参照してください。
- For more information about IMS DataPropagator について詳しくは、以下を参照してください。

- *IMS DataPropagator for z/OS: An Introduction*
- *IMS DataPropagator for z/OS Concepts*

HALDB 区画選択出口ルーチン


インストール・システム定義の High Availability Large Database (HALDB) 区画選択出口ルーチンはすべて、アセンブルおよびバインドを行って、ロード・ライブラリーに入れる必要があります。

選択するライブラリーは、IMS SDFSRESL 連結の一部である必要があります。また、すべての HALDB 区画選択出口ルーチンは、再入可能としてリンクされる必要があります。

HALDB 区画選択出口ルーチンの名前は、DBD 生成時に HALDB 区画定義ユーティリティまたは DBRC INIT.DB コマンドで指定できます。

HALDB 区画選択出口ルーチンを定義する場合、HALDB 区画定義ユーティリティで定義される区画ストリング値パラメーターまたは DBRC INIT.PART コマンドの KEYSTRNG パラメーターについても検討する必要があります。このパラメーターは、ユーザー出口ルーチンに HALDB 区画選択に使用できるオプションのデータを提供します。このパラメーターは必須ではなく、省略することができます。

関連資料:

 HALDB 区画選択出口ルーチン (DFSPSE00) (出口ルーチン)

z/OS 自動リスタート・マネージャー (ARM)

z/OS 自動リスタート・マネージャー (ARM) を使用すると、z/OS ハードウェアまたはソフトウェアの障害の発生後に、サブシステム (またはジョブ) を再始動できます。

さらに、サブシステムをある z/OS システムから他のシステムに移動する必要がある z/OS ハードウェアまたはソフトウェアの障害の場合、ARM は同じ再始動グループに定義されているすべてのサブシステムを 1 つのグループとして、残っている z/OS システムに移します。

IMS は、以下の環境で ARM をサポートします。

- TM-DB
- DCCTL
- DBCTL
- XRF
- FDBR

DL/I、DBB および IMS ユーティリティはサポートされません。IMS 制御領域が ARM によって再始動される唯一の領域です。

重要: DL/I SAS および DBRC 領域は IMS 制御領域により内部的に始動されます。IMS 従属領域は、自動的には再始動されません。これらの領域は通常は、IMS 制御領域が再始動した後に再始動されるためです。

IMS が ARM への登録呼び出しで使用するエレメント名は IMSID です。エレメント・タイプは SYSIMS です。ARM では重複したエレメント名は使用できません。ARM を使用する場合、オンライン・システムおよび FDBR システムの IMSID は固有でなければなりません。

ARM は、SYSIMS に対してデフォルト ARM レベルとして 1 を与えます。

IMSID は、シスプレックス内で重複してはなりません。IMS を実行している z/OS または CPC で障害が発生すると、ARM はその IMS を、存続している z/OS に移そうとします。IMSID が固有でない場合、ARM は、その IMS を、障害の発生した CPC から、同じ IMSID の IMS がすでに入っているところに移します。

IMS が z/OS によって取り消された場合、CANCEL または FORCE コマンドに ARMRESTART オプションが指定されていれば、IMS は ARM によって自動的に単に再始動されます。

IMS は、以下のユーザー異常終了テーブルを保持していて、以下の異常終了のいずれかが発生すると、ARM から登録解除されます。

1. U0020: USER 20 - MODIFY
2. U0028: USER 28 - /CHE ABDUMP
3. U0604: USER 604 - /SWITCH
4. U0758: USER 758 - QUEUES FULL
5. U0759: USER 759 - QUEUE I/O ERROR
6. U2476: USER 2476 - CICS TAKEOVER

これらの異常終了の最初の 3 つは、オペレーター介入の結果です。また、最後の 3 つの異常終了については、何らかの外部変更を行ってからでないと、IMS を再始動することはできません。

第 3 章 IMS システムの文書化

オンライン IMS システムの管理をサポートする計画を立てる場合、文書化に伴ういくつかの責務について考慮しなければなりません。

このアプリケーション要件の文書を検討することは、IMS オンライン・システムを設計したり、その設計に必要な変更に対応したりする場合に必要な作業です。

IMS システムの要件の抽出

オンライン環境でのアプリケーションの有効範囲と影響についての分析は、設計段階と、アプリケーション変更が提案されたときに行われます。

この際に、次のソースを検討することにより、アプリケーション要件の詳細を評価しなければなりません。

- プログラムの仕様と論理
- 実施計画
- ビジネス要件の要約
- 設計変更要求

アプリケーション文書を検査する場合は、次のような種類の情報を抽出しなければなりません。

- IMS 機能に対する要件
- データベース要件
- アプリケーション作業負荷の予測
- ネットワーク定義要件
- セキュリティーに関する考慮事項
- 操作要件
- 監査およびヒストリーに関する推奨
- パフォーマンス要因
- 端末要件
- FPBUF 要件

システムでインテリジェント・リモート・ステーションを使用しなければならない場合は、次のことを行わなければなりません。

- 分散アプリケーション処理をサポートするために使用できる IMS の機能 (例えば、ISC、MSC、LU 6.2 など) を選択します。
- IMS が提供する端末サポートを識別します (ETO を含む)。
- 使用するシステムの高速機能要件を識別します。
- アプリケーション要件のインテリジェント・リモート・ステーションへのオフロードと、特殊なコンポーネントまたは画面フォーマット設定の使用を評価します。

- ・ インテリジェント・リモート・ステーションに常駐して IMS と通信するプログラムの設計を支援し、IMS とインテリジェント・リモート・ステーションが使用するシステム・ネットワーク体系 (SNA) プロトコルを識別します。

設計の検討への参加

オンライン IMS システムの管理者は、その仕様の適切な範囲の詳細を扱います。管理者はオンライン・システムについて計画し、また続いてその仕様を決定する必要があります。アプリケーション・パッケージの開発が進むにつれ、設計を検討することが必要になります。

以下の表は、収集する必要のある情報およびその情報がどのようにシステム管理タスクに関連するかを要約したものです。

アプリケーション設計仕様に加えて、アプリケーション開発チームがスケジュールと責任についての管理文書を保守する場合があります。管理者は自分自身の要件と工程表 (例えば、操作手順についての完了日、テスト実施日など) を使用して、その計画に貢献しなければなりません。

表 3. 管理上の設計検討の使用

設計段階	必要な情報	管理用タスク
設計の検討 1	<ul style="list-style-type: none"> ・ プロジェクトの有効範囲 ・ ハードウェアおよびソフトウェア要件 ・ エンド・ユーザーおよび開発の接点 	<ul style="list-style-type: none"> ・ IMS 機能の要件を分析する ・ 文書化計画に寄与する ・ 標準への準拠を検査する ・ ネットワークの影響を評価する ・ DP 操作の影響を評価する ・ 作業負荷を予測する
設計の検討 2	<ul style="list-style-type: none"> ・ MFS および画面使用特性の使用 ・ エンド・ユーザー制御のエレメント ・ ネットワーク計画 ・ オペレーター制御用のポインター ・ トランザクション負荷 	<ul style="list-style-type: none"> ・ MFS ライブラリー管理を確立し、形式名を識別する ・ ディクショナリーの使用を調整する ・ 命名の基準を検査する ・ ネットワーク要件を追跡する ・ セキュリティ戦略を計画する ・ RTO と MTO のプロシーチャーを開始する ・ 処理の作業負荷を予測する
設計の検討 3	<ul style="list-style-type: none"> ・ メッセージ定義 ・ 会話型の属性 ・ データベースおよびプログラム ・ システム・リソース要件 ・ メッセージ編集の必要性 ・ リカバリーに関する考慮事項 ・ セキュリティと監査 	<ul style="list-style-type: none"> ・ メッセージ・キューを計算する ・ SPA データを計算する ・ システムおよび JCL を指定する ・ システム要件およびハードウェア計画を仕上げる ・ メッセージ編集コーディングを指定する ・ リカバリー・プロシーチャーを開始する ・ セキュリティ設計を開発する

表 3. 管理上の設計検討の使用 (続き)

設計段階	必要な情報	管理用タスク
データベース・アプリケーション設計の検討	<ul style="list-style-type: none"> データベース保守 データベース共用 妥当性検査および受け入れの計画 パフォーマンス予測 モニター計画 	<ul style="list-style-type: none"> オンライン・データベースおよびイメージ・コピー要件を文書化する システム保全性オプションを選択する システム可用性を確立する パフォーマンス基準を確立し、モニター計画を立てる
論理の検討	<ul style="list-style-type: none"> モニター・ポインター プログラムのプリロード 仮想記憶の必要性 データベースの処理意図の競合 	<ul style="list-style-type: none"> モニター戦略を開発する 従属領域の計画を立てる スケジューリング・アルゴリズムを開発する バッファ・プールおよびシステム・データ・セット・リソースを見積もる

関連資料: 設計の検討の目的と有効範囲について詳しくは、「IMS V15 データベース管理」を参照してください。

命名規則の設定

アプリケーション仕様と IMS オンライン・システム設計の管理における重要な部分は、リソースの命名規則を維持することです。

多数のリソースを持つ大きなシステムを定義する場合、リソースの特性をリソース名から認識できれば、次のように多くの利点があります。

- システム定義入力を簡単に検査でき、変更の識別も容易になります。
- MTO 制御がより効果的で効率的になり、エラーを少なくできます。
- アプリケーション設計を修正する場合、すでに定義済みのリソースを容易に認識でき、不明確な追加リソースや不要な追加リソースを作成せずに済みます。

データベース管理者と協力し、最低でも次のリソースについて命名規則を確立してください。

- データベース、およびその DD 名とデータ・セット名
- イメージ・コピーおよび変更累積データ・セット名
- セグメント名およびフィールド名
- PSB およびプログラム名
- トランザクション・コード
- MFS 形式名
- LTERM 名およびノード名
- ETO 端末名およびユーザー名
- LU 6.2 記述子名
- オンライン・ログ・データ・セット名
- システム・ログ・データ・セット名
- IMS モニター出力データ・セット名
- リンク名および IMS システム ID (複数システム結合の場合)

- 高速 DB リカバリー領域名
- High Availability Large Database (HALDB)、および HALDB 区画、DD 名、およびデータ・セット名

以下の表に、IMS がオンライン・アプリケーション用に制御するリソースに適用できる命名規則の例を示します。

表 4. 命名規則の例

リソース	命名規則	説明	
トランザクション	Taaatsss	T	トランザクション
		aaa	アプリケーション ID
		t	更新トランザクションの場合 U、または照会トランザクションの場合 R
		sss	トランザクション・シーケンス
LTERM 名	cnxiiii	c	ローカルの場合 L、交換の場合 S、非交換の場合 N
		nn	端末タイプを表す 2 文字のコード
		x	画面サイズ、プリンター、コンポーネントのいずれかを示す 1 文字の属性
		iiii	4 文字の ID
MFS (MSG 名) (MID および MOD)	aaaiiii	aaa	アプリケーション ID
		iiii	4 文字の ID
MFS (FMT 名) (DIF および DOF)	aaaiii	aaa	アプリケーション ID
		iii	3 文字の形式 ID
モジュール名	Maaaiiii	M	モジュール名
		aaa	アプリケーション ID
		iiii	4 文字の ID
ジョブ名	Jaaannnn	J	ジョブ名
		aaa	アプリケーション ID
		nnnn	4 文字のジョブ ID

命名規則の詳細については、以下を参照してください。

- IMS V15 データベース管理 (データベース、PSB、プログラム、および HALDB の命名規則に関する推奨について)
- IMS V15 システム定義 (制限されている名前のリストについて)
- IMS V15 コミュニケーションおよびコネクション (ETO 用の特定の命名規則について)

データ・ディクショナリーの使用

IBM DB/DC データ・ディクショナリー・ライセンス・プログラム (プログラム番号 5740-XXF) を使用すると、そのいくつかの機能はシステムとネットワークの文書化に役立ちます。この製品は対話式プロセスをサポートしており、都合の良いときに端末を使用して文書を作成できます。

この製品は、データベース、セグメント、フィールド、PSB、PCB、トランザクション、プログラム、システムの標準カテゴリを提供します。これらのカテゴリを使用して、オンライン IMS システム・リソースの詳細な説明を構築できます。プロシージャと実行の JCL を (「ユーザー・データ」として) 記録することもできます。ステージ 1 マクロのうち、APPLCTN、DATABASE、および TRANSACT の 3 つは、カード出力として生成できます。

拡張性機能を使用して、各端末について端末の件名を定義し、記録する属性の選択を指定できます。「説明」と「ユーザー・データ」のセグメントを使用して、その他のフリー・フォームのテキストを端末の件名に関連付けることができます。

IBM DB/DC データ・ディクショナリーの詳細については、「*IBM DB/DC Data Dictionary Administration and Customization Guide*」(SH20-9174) を参照してください。

システムの特性の文書化

管理者は、実動システムの設計と操作を文書化し、独立した情報本体として作成しなければなりません。この資料はアプリケーション要件から得られ、IMS 機能の使用目的も含みます。

以下は、主要な文書化アクティビティーについて説明したものです。

- IMS システム定義の文書化

IMS オンライン・システムの設計の詳細は、システム定義マクロの仕様に反映されます。システム定義プロセスの最初のステージへの入力、主要な文書化ツールとして使用できます。パラメーターの選択についてのコメントを組み込んでおくと、定義変更を制御するのに役立ちます。

IMSCTRL マクロ・ステートメントは、ネットワーク内の現行 ETO 記述子の報告書を作成する ETO オプションを備えています。この ETO 記述子報告書を使用して、現行ネットワーク定義をモニターできます。

関連資料: パラメーターに関するコメントを付ける方法の詳細および IMSCTRL マクロについては、「*IMS V15 システム定義*」を参照してください。

- DB/DC 環境および DCCTL 環境での IMS ネットワークの文書化

システムの文書化をより広い視野で捕らえると、IMS ネットワークの詳細を文書化する必要があります。この文書化は、アプリケーション・システムの初期計画が使用可能になると同時に開始してください。早期に開始する利点は、物理ネットワークについてと、それを IMS 接続によって論理ネットワークとして使用する方法について詳しく知ることができる点です。オンライン IMS システムが、

VTAM に対して定義されたネットワークの一部である端末を使用する場合は、システム・プログラミング・スタッフによって開発された文書の一部を使用できることがあります。

詳細な文書化によって、次のことが可能になります。

- 端末装置の機能と操作を詳しく知る
 - アプリケーションが端末を通常と異なる方法で使用しようとしているかどうかを知り、もしそうなら潜在的な問題を調べる
 - IMS システム定義ステージ 1 の入力準備をする
 - ETO 記述子を作成する
 - LU 6.2 記述子を作成する
 - インストールとネットワーク構築の計画を立てる
 - 全体構成のサブセットの各操作局面を理解する
- DB/DC 環境および DCCTL 環境での端末プロファイルの文書化

主要な設計が確定した後には必須ネットワークを文書化する方法の 1 つは、個々の端末の使用目的と特性を記録することです。次の事項が入っているプロファイルを構成してください。

- 端末タイプ、その必須機能、それが使用する接続のタイプ
- 装置について選択されたオプションとその選択理由
- アプリケーションによる端末の使用方法の特性
- 端末またはユーザーへ関連付けることができる、提案された LTERM 名
- VTAM サポート装置の場合は、ノード名と伝送特性
- 提案された使用法の範囲と、その装置を非 IMS ユーザーと共用するかどうか
- その装置に適切な診断プロシージャ
- ユーザー・プロファイルとユーザー・プロファイル・セキュリティー

動的端末では、将来の変更を容易にするため、動的端末の特性とユーザーの記録を保守する必要があります。

- APPC/IMS のトランザクション・プロファイル名の文書化

トランザクション・プログラム名 (TPN) の定義は、APPC/MVS リソースの TP_Profile の中に入っています。TP_Profile の中で、TP プロファイル・データ・セットが個々の TPN の属性情報を提供します。さまざまな特性を持つ TPN を、それらの TPN に関連付けられている個々の LU 名に対して定義できます。

関連資料: TP 名について詳しくは、「IMS V15 コミュニケーションおよび接続」を参照してください。

- 実動システムの構成の文書化

個々の端末と端末のクラスターをネットワーク内で接続する方法を示した構成マップを作成する必要があります。それによって、MTO に対してネットワーク制御を指定する方法について理解を深めることができます。また、構成マップがあると、さまざまな技術専門家と対話する場合にも役立ちます。このマップは次のことを示していなければなりません。

- プロセッサまたはホスト・コンピューター
- チャンネルと回線 (回線タイプも含む)
- 通信コントローラー
- 制御装置および端末接続機構
- VTAM ノード名
- XRF システムの場合は、USERVAR 名または MNPS ACB 名
- 代替の接続または構成

起こり得る問題を解決するには、問題がある端末または制御装置を識別できなければなりません。装置に識別ラベルを付けることにより、エンド・ユーザーとサービス技術員の両方を支援できます。その端末に適切な IMS アドレスまたはノード名と LTERM に加えて、ハードウェア・アドレス、回線 ID、必要な場合はその他のデータも含めてください。

第 4 章 IMS のオペレーションおよびリカバリーの概要

どのラージ・システムの場合でも、オペレーションとリカバリーは同時進行するものです。なぜなら、オペレーションとはシステムが円滑に稼働し続けるようにする日常のアクティビティーのことを指し、リカバリーとは障害が起こったシステムをオンラインに戻すのに必要なアクティビティーのことを指すからです。

オペレーションについての情報において、リカバリーの際に行う必要があることを説明していなければ、システム (またはシステムの一部) に問題が起こった場合にこの情報では不十分です。

オペレーションに関連したもう 1 つの重要な作業は、日常のオペレーションとリカバリーの両方についてのプランを作成することです。このプラン作成作業の一環として、IMS のオペレーターとシステム・プログラマーのそれぞれの役割を指定する手順を作成します。この手順により、これらの各役割の責任が明確になります。

このトピックでは、IMS のオペレーションとリカバリーの両方の概要を示し、この両方の作業のために IMS および z/OS が提供するツールのいくつかについて説明します。

操作タスクの理解

操作用プランの作成作業には、機能とツールの選択および操作手順とエンド・ユーザー手順の作成という 2 つの主要部分があります。

最初に、システム操作用に IMS が提供するツールをどのように使用するかを決める必要があります。これには、例えば、ログのセットアップ時に重複ロギングを使用するかどうかの選択、データベースのバックアップ・コピーを作成する頻度の決定、およびデータベースのリカバリーの制御に DBRC を使用するかどうかについての選択などが含まれます。

次に、IMS の操作手順と使用手順を作成する必要があります。操作手順では、以下の事柄の実行方法をオペレーターに指示することが必要です。

- IMS の開始および再始動
- IMS のコントロール
- IMS を修正するためのオンライン変更
- IMS のシャットダウン
- 各種の IMS ユーティリティーの実行
- IMS 障害およびその他の障害からのリカバリー

これらの手順は、第一には、IMS をマスター端末から操作するマスター端末オペレーター (MTO) 用のものです。第二には、これらは IMS の円滑な稼働状態を維持するのを援助する人々、例えば、より複雑な IMS リカバリー問題に対処するリカバリーの専門家が行う手順です。

エンド・ユーザー手順では、エンド・ユーザーに以下の事柄の実行方法を指示します。

- 端末の操作
- IMS への接続の確立
- IMS との通信 (特定のアプリケーションを使用)
- IMS への接続の終了
- 各種のエラー条件が検出された場合の応答

本書は、操作に関するすべての事柄を取り扱ったものではありません。システム操作のさまざまな局面の詳細については、他の IMS トピックに記載されています。

関連資料:

- 「IMS V15 オペレーションおよびオートメーション」は、IMS の操作に使用するコマンドに関するガイダンス・レベルの情報と詳細情報を提供します。
- 「IMS V15 データベース・ユーティリティー」および「IMS V15 システム・ユーティリティー」は、IMS で使用可能な各種のユーティリティーに関するガイダンス・レベルの情報と詳細情報を提供します。

自動化オペレーション

自動化オペレーションは、インストールの生産性を向上させるのに役立つツールおよび技法です。システムはますます複雑化し、メッセージ・トラフィックが増え続けていることから、特定の作業の自動化によってシステムの効率化が図れます。

自動化オペレーションを使用すると、次のことが可能です。

- エラーを最小限に抑える
- 可用性を向上させる
- 問題の診断と防止を促進させる

IMS には、ユーザーがオペレーションを自動化するのに役立つ、以下のツールが用意されています。

- 時間制御操作 (TCO)
- 自動化操作プログラム・インターフェース (AOI) (Automated Operator Interface (AOI))
- REXX SPOC API

IMS オペレーションは Tivoli NetView for z/OS を使用して自動化することもできます。Tivoli NetView for z/OS 機能は、IMS から独立しているため、情報を収集して、IMS で使用可能でないコマンドを出すことができます。

タスクを自動化することには、多数の利点があります。オペレーターが行うジョブの多くは、システムのモニターやリカバリー・コマンドの送出的ように、単純で繰り返しの多い作業です。多くの場合、これらのジョブを自動化して、操作手順の実装などといったより複雑なアクティビティーのためにオペレーターを解放することが可能です。

IMS 環境では、TCO および AOI を使用して以下のことを行うと、オペレーターの生産性を向上できます。

- オペレーターの生産性と正確度の向上。オペレーターの作業を自動化すれば、手順が単純化され、オペレーター入力が削減されて、オペレーターのエラーが最小限に抑えられます。例えば TCO は、システム状況のモニター、メッセージ領域と通信回線の開始、およびユーザーへのシステム状況の通知を自動的に実行するため、オペレーター入力を削減できます。
- 問題判別の促進。オペレーターの基本的なジョブは、問題を迅速に迂回または修正することです。問題に関する十分な情報が入手できないと、オペレーターが問題を迅速に修正できない可能性があります。自動化オペレーションは特に問題判別に適しています。例えば TCO は、問題を迅速に識別して訂正できるように、必要な情報の収集と診断用の分析を自動的に実行できます。

自動化できるオペレーションを識別してそれらを実装するためには、インストール・システムでのオペレーションについて熟知している必要があります。さまざまなリソースを収集して分析し、どの作業が自動化の候補として適しているのかを識別する必要があります。分析すべきリソースとしては、次のものが挙げられます。

- システム・ログ
- 問題管理報告書
- ヘルプ・デスクへの電話の記録
- オペレーターのメモ

繰り返しの多い作業や予測可能な作業は自動化の候補として適しています。

オペレーションの自動化の処理は、反復して行う必要のある処理です。自動手順を作成して使用してみた後には、それらの評価を行う必要があります。さらに、この処理においては、他にもあらたに自動化できる作業がないかどうかを考える必要があります。

関連資料:

- IMS で提供される自動化ツールについて詳しくは、「IMS V15 オペレーションおよびオートメーション」を参照してください。
- Tivoli NetView for z/OS について詳しくは、「Tivoli NetView for z/OS インストール: 入門」または「Tivoli NetView for z/OS ユーザーズ・ガイド」を参照してください。

リカバリー作業の理解

リカバリーとは、障害が発生したシステム、アプリケーション・プログラム、またはデータベースを復元して、通常オペレーションに戻す作業です。

リカバリーは、オペレーションの中で最も大きく最も複雑な部分を占めます。そのため、あらゆるオペレーション説明は主としてリカバリーに関する説明となります。

以下に、システムのリカバリーに含まれるものを示します。

- データベース
- ユーザーのデータ処理要求
- その処理を行うプログラム (アプリケーション・プログラム)

- ユーザーに送信された出力

リカバリー可能なシステムとは、次のことが保証されるシステムです。

- データが破損しない。
- データベースに対する未完了の変更は保管されない。

データの破損には 2 つの場合があります。1 つは、データが物理的に破損する場合、つまり、データが常駐しているディスクまたはテープが損傷を受けるか、または間違った場所に置かれる場合です。もう 1 つは、データが「論理的に」破損する場合、つまり、データが不正になるか、または他のデータとの関係を失う場合です。データ保全性を保証するシステムでは、データが破損したり、まだ処理が完了していないデータが保管されたりすることが決してないようになっています。

例: リカバリーが行われないシステム

どのようにして不具合が起こり得るのかを示すために、リカバリー・メカニズムを備えていない仮定のシステムを考えてみましょう。カスタマーの注文を処理するトランザクションをユーザーが要求するとします。

アプリケーション・プログラムは、作業の一環として次のことを行います。

- データベース内の在庫数を減算する。
- 費用をカスタマーの未払い請求書に追加する。
- その金額を毎日の総売上用アキュムレーターに追加する。

以下のリストは、さまざまな障害のシナリオを示します。

- システム障害の影響

在庫数が減算された時点から課金が割り当てられるまでの間にシステムに障害が起こった場合はどうなるでしょうか。ユーザーがこのトランザクションは完了したと考えた場合は、データベース内のカスタマーの請求書レコードと総売上レコードは誤ったものとなってしまいます。また、ユーザーがこのトランザクションは失敗したと考えてそれを再実行した場合は、在庫数がもう一度減算されて、やはりそのレコードは誤ったものとなります。どちらのケースでも、データ保全性は損なわれています。

- プログラム異常終了の影響

プログラムがデータベース内のレコードの変更を開始してから異常終了したらどうなるでしょうか。結果は、システム障害の影響の場合と同じになります。つまり、データベースへの変更が途中までしか行われていない状態になります。途中でしか変更されていないデータベースでは、その保全性は損なわれています。

- 入出力エラーの影響

プログラムが在庫レコードの読み取りを試行して、装置エラーを検出したとします。なんらかの入出力の問題が原因で、そのレコードは獲得不能になっています。したがって、そのプログラムは実行できず、カスタマーの注文レコードが埋められません。それだけでなく、このレコードに依存している他の作業もすべて未完了となります。

同様に、プログラムがこのレコードの読み取りと更新を実行できることを予期して別のいくつかのレコードをすでに変更していた場合はどうなるでしょうか。この場合も、データの全体としての健全性は損なわれています。なぜなら、そのデータは部分的にしか更新されていないからです。

- キューの損失の影響

ラージ・システムでは、作業要求はしばしば保管されて (キューに入れられて)、後に作業負荷が許容範囲のときに、処理されます。また、プログラムが要求元 (またはその他の場所) に返送する出力も、しばしばキューに入れられて、後に作業負荷が許容範囲のときに、送信されます。

要求が入力された時点からその要求がキューから取り出されて実行されるまでの間にシステムに障害が起こった場合は、その要求は失われる可能性があります。キューに入れられた出力についても、これと同じことが言えます。アプリケーション・プログラムが出力を提供した時点から、その出力をキューから取り出して実際に送信できるようになるまでの間にシステムに障害が起こった場合は、それらの処理結果が失われる可能性があります。

リカバリー可能なシステムの要件

システムがリカバリー可能であるためには、システムはデータベースに対して途中で完了していない (したがって誤りである) 変更が行われないようにデータベースを保護する必要があります。システムは、データベースの物理的な破損 (データベースが部分的または全体的に使用不可になるか、消失すること) に対処可能であることが必要です。また、入出力キューを保護してそれらが破損することがないようにする必要があります。

リカバリー処理のステップ

リカバリー処理には 2 つの主要な作業があります。それは、問題が発生する前にリカバリーの際に何を行うかを計画することと、問題が発生した後にその問題に対処してそれを修正することです。

IMS リカバリー用のプランを作成する作業は、次のように細分できます。

- ロギングのセットアップ
- チェックポイントの設定
- バックアップ・コピーの作成
- プロシージャの作成と責任の割り当て

また、対応に関する作業も次のように細分できます。

- エラーの通知
- IMS のシャットダウン (必要な場合のみ)。完全にシャットダウンする場合と部分的にシャットダウンする場合があります。
- エラーの診断と修正 (迂回する場合もある)
- リカバリー用のユーティリティとサービスの使用 (必要な場合)。これらはオンラインでもオフラインでも実行できます。
- IMS (または IMS の修正したばかりの部分) の再始動

当然のことながら、実際のリカバリーが上記の 2 つのリストに示されているように簡単に済むことはめったにありません。

IMS におけるリカバリーのメカニズム

IMS は、リカバリー可能であるように設計されており、このようなエリアで役立つメカニズムをいくつか備えています。これらのメカニズムの中には、自動的に機能するものとユーザーの関与を必要とするものがあります。

リカバリーが正常に行われるかどうかは以下の 2 点にかかっています。

- 戻るべきセーフ・ポイント (保全性が維持されていることが分かっているポイント) をもっている。これは、そこではデータが正しいことが分かっている、そこでレコードを作成するポイントです。
- このセーフ・ポイント以降に行われた処理をトラッキングしてある。

この 2 つのタイプの情報が収集されれば、ほとんど常にリカバリーが可能です。

同期点

厳密には、障害発生後にシステムを再始動できるようにするために必要なのはログだけです。システムを正しく開始させてあり、それ以降に行ったことをすべてログに記録してあるのであれば、常にリカバリーが可能です。このケースの再始動では、システム用に書き込まれたあらゆるログ・レコードをシステムが読み取る必要があります。

最初にシステムを開始してから経過した時間が長ければ長いほど、書き込まれたログ・レコードの数が多く、その分再始動にはより多くの時間がかかります。

定義: 同期点 は、データ保全性またはシステム保全性が確保された既知のポイントです。これは、データベースに対するすべての変更が完了した時点のことです。障害によって再始動が必要となった場合は、最新の同期点から処理を再開できます。

同期点をもつことの価値は、IMSの再始動時に、IMS がその同期点の設定以前に作成されたすべてのログを無視できることです。同期点を設定する頻度が低いと、再始動時に IMS が読み取らなければならないログ・データの量が増えるため、同期点の価値がそれだけ下がります。したがって、同期点を定期的に設定する必要があります。そのようにすれば、問題が起こっても、最後の同期点を比較的最近に設定してあるため、あるかぎりのログを調べる必要がありません。

IMS は、2 つのタイプの同期点を使用します。すなわち、IMS 自体が取る同期点 (システム・チェックポイント と呼びます) および IMS の下で実行される個々のアプリケーション・プログラムが取る同期点 (アプリケーション・プログラム同期点 またはアプリケーション・プログラム・コミット・ポイント と呼びます) です。

システム・チェックポイント:

IMS は、自動的に定期的なシステム・チェックポイントを取ります。したがって、IMS の再始動が必要な場合は、最後のチェックポイント (あるいは、当該の作業が終了したときに最後のチェックポイントがまだ完了していなかった場合は、それ以前のチェックポイント) から IMS を開始できます。

システム・チェックポイント間のインターバルは IMS が書き込んだログ・レコードの数によって測定されるため、チェックポイントは経過時間ではなくアクティビティを測定するものさしとなります。システム・チェックポイント間のインターバルは IMS のインストール時に選択します。

リカバリー用の準備に関するほとんどの選択の場合と同様に、チェックポイントを頻繁に取ることによる影響と、リカバリーをより高速に行えることの利点とをはかりに掛ける必要があります。IMS はチェックポイントを取る作業を延期させないので、トランザクションの速度が低下しますが、これはそれほど大きな影響の要因とはなりません。

推奨事項: システム・パフォーマンスの向上のためにシステム・チェックポイントを抑制しないでください。これは、緊急時再始動には常に少なくとも 2 つ前のシステム・チェックポイントまで (あるいは以前に再始動した時点まで) 戻る必要があるためです。

IMS MTO も、/CHECKPOINT コマンドを出せば、いつでも追加のシステム・チェックポイントを要求できます。

以下に、IMS が必ずシステム・チェックポイントを取る場合を示します。

- IMS の始動時
- 正規のインターバルが経過したとき
- MTO がコマンド (/DBRECOVERY、UPDATE DB STOP(ACCESS) OPTION(FEOV)、/CHECKPOINT、または /SWITCH OLDS CHECKPOINT) を入力したとき
- IMS のシャットダウン時

IMS がリソース定義データ・セットまたは IMSRSC リポジトリで使用可能であり、AUTOEXPORT=AUTO、RDDDS、または REPO が DFSDFxxx PROCLIB メンバーに指定されている場合、IMS は、チェックポイント処理が正常に実行されて終了するたびに、最後の自動エクスポートまたは EXPORT DEFN コマンド以降に行われた MODBLKS リソースのリソース定義でのすべての変更を RDDDS またはリポジトリに自動的にエクスポートします。

アプリケーション同期点:

IMS システム・チェックポイントは IMS サブシステムのリカバリーには役立ちませんが、アプリケーション・プログラムに障害が起こった場合にそれを再始動することもできなければなりません。IMS では、アプリケーション・プログラム (バッチとオンラインの両方) による同期点の設定が可能です。

アプリケーション・プログラム同期点の目的は次の 2 つです。

- 中間完了ポイント (そこまでに終了した作業は正しいものと判断される場所) のマーク付け。これ以降のリカバリーは、すべてこのポイントから行えます。
- プログラムが更新したデータベース・レコード上で保持されているロックの解除、およびそのプログラムが作成したすべての出力メッセージのエンキュー。エンキューされたメッセージは、IMS によってそれぞれの宛先に送信可能です。

アプリケーション・プログラム同期点はコミット・ポイント と呼ばれることがあります。同期点の設定により、プログラムは自分が行った処理が正確で完了していることを明示し、他のアプリケーションが使用できるようにデータをリリースします。

ロギング

ログは、アクティビティーのリスト、実行済み作業のリスト、およびすでに行われた変更のリストです。何も問題がなかったときのシステムの状態、およびそれ以降にシステムで何が行われたかを把握することで、障害発生時にシステムをリカバリーできます。

ログは、リカバリーおよび再始動を可能にします。IMS は、作動中には常にイベント情報をログ・レコードに書き込みます。このログ・レコードには、以下の情報が記録されます。

- IMS がいつ始動していつシャットダウンされたか
- プログラムがいつ開始されていつ終了したか
- データベースに加えられた変更
- 受信されたトランザクション要求と送信された応答
- アプリケーション・プログラムのチェックポイント
- システム・チェックポイント

以下の図は、IMS オンライン環境でのロギング・プロセスの仕組みを示します。バッチ環境でのロギングはこれよりも単純です。

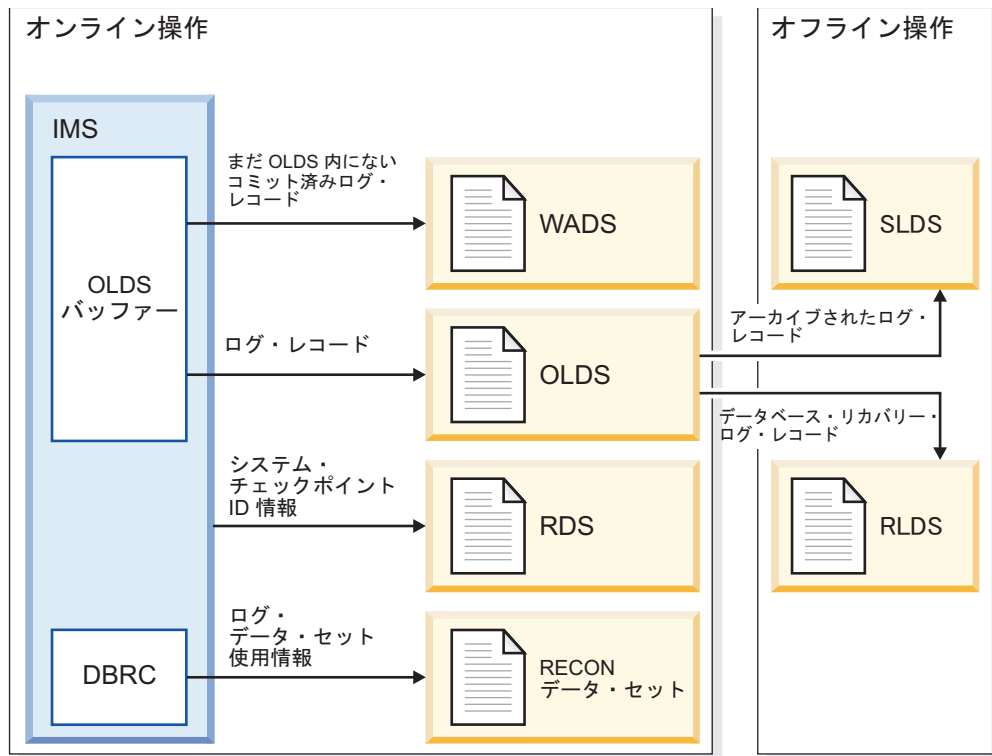


図 23. ロギング・プロセスの概要

IMS は、ログ・レコードをオンライン・ログ・データ・セット (OLDS) に書き込むことにより、ログ・レコードを外部化します。パフォーマンスを向上させ、OLDS 上のスペースを最適化するために、不完全なバッファまたは部分的に埋められたバッファは、リカバリーを可能にするために必要な場合には先行書き込みデータ・セット (WADS) に書き込まれます。WADS は、高い書き込み率を持つ高速 DASD データ・セットです。完全なログ・バッファのみが、OLDS に書き込まれます。ログ・データが OLDS に存在すれば、これに相当する WADS レコードは無視されます。

IMS は一連の OLDS を循環的に使用するため、そのうちの 1 つの OLDS がいっぱいになっても IMS はロギングを継続できます。同様に、ある OLDS への書き込み中に入出力エラーが起こった場合でも、IMS はその障害が発生した OLDS を分離して別の OLDS に切り替えられるので、ロギングを継続できます。一度 OLDS が使用されると、IMS ログ保存ユーティリティにより、DASD またはテープ上のシステム・ログ・データ・セット (SLDS) にアーカイブするためにその OLDS を使用することができます。このユーティリティは、IMS 始動パラメーター (ARC=) を使用して自動的に実行できます。

最後の使用可能 OLDS が満杯になりそうになると、IMS はユーザーに警告を出すので、ユーザーは使用済みの OLDS のアーカイブが完了したことを確認することができます。あるいはシステムに新しい OLDS を追加することができます。IMS ログ保存ユーティリティを使用してユーザーが手動で OLDS を SLDS にアーカイブすることもできます。SLDS は、DASD またはテープ上に置くことができます。アーカイブされた OLDS は、新しいログ・データのために再使用できるようになります。SLDS は、データベース・リカバリー・プロセスへの入力として使用します。

OLDS をアーカイブする場合は、IMS が OLDS のログ・レコードのサブセットをリカバリー・ログ・データ・セット (RLDS) と呼ばれる別のログ・データ・セットに書き込むよう要求できます。RLDS に含まれているのは、データベース・リカバリーに必要なログ・レコードだけです。

IMS は、稼働中に自分のアクティビティをロギングする一方で、定期的にシステム・チェックポイントを取り、そのチェックポイント通知を再始動データ・セット (RDS) と呼ばれる別のデータ・セットに書き込みます。IMS は、再始動時にこの RDS を使用して、再始動の起点とすべき正しいチェックポイントを判別します。

バッチ環境では、IMS はログ・レコードを直接 SLDS に書き込むため、OLDS も WADS も使用しません。オンライン環境の場合とまったく同じように、バッチ SLDS は DASD、テープ、または大容量記憶装置に置くことができます。SLDS が DASD にある場合は、IMS ユーティリティを使用してログ・レコードを DASD からテープまたは他の DASD にコピーできます。

IMS ログ・データ・セット:

これらのトピックでは、OLDS、WADS、SLDS、RLDS、再始動データ・セット (RDS)、RECON データ・セット、CQS のために使用される z/OS ログ・データ・セット、CQS システム・チェックポイント・データ・セット、および CQS 構造チェックポイント・データ・セットについて説明します。

オンライン・ログ・データ・セット:

IMS は、OLDS をオンライン環境でのみ使用します。OLDS には、再始動、リカバリー、およびバッチと動的バックアウトの両方に必要なすべてのログ・レコードが含まれています。OLDS は、これらのログ・レコードを IMS が SLDS にアーカイブするまで保持します。

OLDSDEF ステートメントを使用して、IMS プロシージャ・ライブラリー (IMS.PROCLIB) 内のすべての OLDS を定義してください。OLDS は、直接アクセス装置上に事前に割り当てられている必要があります。また、IMS の実行中に /START OLDS コマンドを使用して追加の OLDS を動的に割り当てることもできます。

IMS は、基本順次アクセス方式 (BSAM) を使用してログ・レコードを OLDS に書き込み、動的バックアウトを実行する際に OLDS を読み取ります。データ・セット (単数) と呼ばれてはいますが、OLDS は実際には複数データ・セットから構成されていて、それらが 1 つ 1 つ順番に循環方式で使用されます。OLDS には少なくとも 3 つ以上、ただし 100 個を超えないデータ・セットを割り当てる必要があります。

関連資料: OLDS のデータ・セットの割り振りについては、「IMS V15 システム定義」の『データ・セットの割り振り』を参照してください。

OLDS が重複ロギングを使用するように指定できます。重複ロギングとは、情報を 2 つのログに重複して記録することです。重複ロギングを使用する場合は、1 次または 2 次データ・セットのいずれかに入出力エラーが発生すると、IMS はエラーがない方の OLDS をクローズし、さらにリカバリー管理 (RECON) データ・セット内でエラーのある方の OLDS に入出力エラーとクローズ・エラーがあることを意味するマークを付けます。その後、IMS は次の使用可能なデータ・セットの対を処理対象としてロギングを継続します。重複ロギングの場合は、データ・セットの最小数は 3 対、最大数は 100 対です。

IMS は、ユーザーが割り当てただけの数の OLDS を使用します。また IMS は、現行 OLDS を変更するたびにメッセージを出します。このメッセージにより、クローズされた OLDS と次に使用される OLDS が識別されます。

以下のイベントのいずれかが起こった場合

- IMS が OLDS をいっぱいにした
- 入出力エラーが発生した
- 以下のコマンドのいずれかを発行した
 - /SWITCH OLDS
 - /DBDUMP DB
 - /DBRECOVERY DB
 - UPDATE DB STOP(UPDATES) OPTION(FEOV)
 - UPDATE DB STOP(ACCESS) OPTION(FEOV)

IMS は次のことを行います。

- 次の OLDS を開く。

- IMS が新しい OLDS を使用していることを DBRC と MTO に通知する。
- 現行 OLDS (重複ロギングを使用している場合は 1 次と 2 次の両方) をクローズする。

IMS が最後の使用可能 OLDS を使用しているときには、IMS は MTO に対して使用可能な OLDS スペースがそれ以上ないことを知らせるアラートを送ります。すべての OLDS が満杯になった時点でアーカイブが完了しなかった場合は、IMS は OLDS スペースが使用可能になるまで待ちます。IMS は、まだアーカイブされていないアクティブ・データを含んでいる OLDS に対してはログを書き込みません。そのため、ログ保存ユーティリティを使用して OLDS スペースを解放する必要があります。最後の割り当て済み OLDS の使用後は、IMS は最初の OLDS を再使用します (それがアーカイブ済みである場合)。

/STOP コマンドを使用すると、OLDS を停止して、動的に割り振りを解除できます。停止させられた OLDS は、以後は循環プロセスに組み入れられなくなります。

推奨事項: OLDS のリカバリーを必要とするエラーが起こった場合は、当該の OLDS を停止してください。

制約事項: 現行 OLDS は停止できません。また、現在使用可能な OLDS が 2 つ以下である場合はどの OLDS も停止できません。

同様に、/START コマンドを使用すると、OLDS を開始してそれを動的に割り当てられます。IMS は、ある再始動からその次の再始動までの間の OLDS の状況 (使用中、停止済みなど) を保存します。

OLDS をアーカイブする各ジョブが、それらの OLDS が作成された順序どおりに完了しない場合があります。例えば、ある OLDS がまだアーカイブされていないうちに後続の OLDS がアーカイブ済みとなってしまう場合などです。このようなことが起こったときは、IMS はメッセージ DFS3259I を出して、次の使用可能 OLDS を使用します。

DBRC RECON データ・セットには、それぞれの IMS サブシステムごとの OLDS に関する情報が含まれています。RECON データ・セット内の情報は、ある OLDS が使用可能であるのか、それともアーカイブする必要があるアクティブ・ログ・データを含んでいるのかを指示します。

先行書き込みデータ・セット:

IMS は、オンライン環境でのみ先行書き込みデータ・セット (WADS) を使用します。WADS には、OLDS バッファに入っているが、まだ OLDS には書き込まれていないコミット済みログ・レコードのコピーが含まれています。

ログ効率を最大限に高めるために、IMS はログの先行書き込み機能を使用して、部分的に満たされたブロックを (OLDS ではなく) WADS に書き込みます。IMS は、適切なログ・データを OLDS に書き込んだ後、継続的に WADS スペースを再使用します。

このログ先行書き込み機能は、IMS がデータベースに変更を書き込む前にすべてのログ・レコードがログに確実に揃っているようにします。IMS は、次の場合にはデータベースを更新します。

- IMS がデータベース・バッファを再使用する必要があるとき (これがコミット前である場合)
- コミット中
- VSAM がバックグラウンドで書き込み中であるとき

IMS に障害が起こった場合は、WADS 内のログ・データを使用して OLDS の内容を完全なものにしてから、IMS 緊急時再始動の一環として、またはログ・リカバリー・ユーティリティのオプションとして、その OLDS をクローズしてください。OLDS を緊急時再始動時にクローズする場合は、障害発生時に使用中であった WADS を含める必要があります。

WADS は、拡張カウント・キー・データ (ECKD™) アーキテクチャーをサポートする DASD 装置上に事前割り振りし、フォーマット設定しておく必要があります。FORMAT WADS | ALL キーワードを /NRESTART または /ERESTART コマンドのいずれかで使用して、WADS をフォーマットしてください。WADS はすべて、同じタイプの装置上に置かれている必要があり、さらに同じスペース割り当てがなされていなければなりません。/START WADS コマンドを使用して追加の WADS を動的に割り当てることもできます。

WADS は、制御インターバル (CI) のサイズが 4 KB (4096 バイト)、2 次スペース割り振りが 0 の線形 VSAM データ・セットとして、SHAREOPTIONS(3 3) パラメーターを使用して定義する必要があります。

WADS への書き込みには、High Performance FICON® for z Systems™ (zHPF) および zHyperWrite のハードウェア機能を使用できます。zHPF は、LPAR 上で使用可能になっている場合、スループットを向上させます。zHyperWrite は、複製の待ち時間を短縮するためのオプションとして使用されます。DFSDFxxx PROCLIB メンバーの LOGGER セクション内で ZHYPERWRITE= パラメーターを使用して、zHyperWrite を使用可能または使用不可にすることができます。

IMS の再始動時には、WADS に対する以下の指定をどれでも変更できます。

- WADS の数
- WADS のシーケンス
- WADS の名前
- 単一 WADS または重複 WADS の使用

推奨事項: 潜在的なリソースの競合を除去するために、WADS は OLDS 用に使用する装置とは別の使用頻度の低い装置に配置してください。

WADS を OLDS のいずれかと同じ装置上に置き、さらにその OLDS に対してフルトラック・ブロッキング (ブロックの大きさがトラック全体の大きさに等しい) を使用する場合は、その装置が低頻度 OLDS シークに対処できなければなりません。それでも競合が発生する可能性は残ります。

WADS と OLDS が同じ装置上にある場合は、ログ保存ユーティリティ (DFSUARC0) または動的バックアウトが原因で、アーカイブされている OLDS とアクティブ WADS との間で重大な競合が起こる可能性があります。

関連資料:

DFSDFxxx メンバーの LOGGER セクション (システム定義)

システム・ログ・データ・セット:

IMS は、SLDS をオンライン環境とバッチ環境の両方で使用します。オンライン環境では、SLDS にはアーカイブされた OLDS データが含まれています。バッチ環境では、SLDS には現行ログ・データが含まれています。

ログ保存ユーティリティを実行するたびに SLDS が 1 つ作成されます。1 つの SLDS には 1 つまたは複数の OLDS からのデータを入れられます。SLDS をデータベース・リカバリー・ユーティリティ (データベース・リカバリー、データベース変更累積、およびバッチ・バックアウト) への入力として使用してください。SLDS を IMS の緊急時再始動時に使用することもできます。SLDS は、DASD、テープ、またはその他の大容量記憶装置に保管できます。

DBRC は、RECON データ・セット内の SLDS に関する情報を維持します。

- バッチ・サブシステムの場合は、DBRC は PRILOG および SECLOG レコードの SLDS 情報を維持する。
- オンライン・サブシステムの場合は、DBRC は、ログ保存ユーティリティの実行時にユーザーが RLDS を指定しなかった場合にのみ、PRILOG および SECLOG レコードの SLDS 情報を維持する。そうでない場合は、DBRC は PRISLD および SECSLD レコードの SLDS 情報を維持します。

ログ保存ユーティリティは、どの OLDS をアーカイブ中で、どの SLDS を作成中かを DBRC に指示します。IMS オンライン・システムはアーカイブ済みの OLDS を再使用できます。

すべてのログ・レコードを OLDS から SLDS にコピーしたい場合が一般的ですが、特定のレコードを指定することも可能です。スペースの節約のために一部のログ・レコード・タイプを SLDS から取り除く場合は、ログ保存ユーティリティの実行時に NOLOG キーワードを指定してください。SLDS が必ず含んでいなければならないレコードは、データベース・リカバリー、バッチ・バックアウト、または IMS 再始動の際に必要な可能性のあるレコードです。除くことができるレコードは次のとおりです。

X'10' セキュリティ違反レコード

X'45' チェックポイント時に書き込まれた統計レコード

X'5F' 呼び出しトレース・レコード

X'67' 通信 (SNAP) トレース・レコード

X'69' 無許可 ID レコード (3275 ディスプレイ端末の場合)

再始動読み取り処理に必要なログ・データを OLDS から入手できない場合は常に、IMS は、IMS 再始動時に SLDS を動的に割り振ります。OLDS が使用不能となる場合としては、その OLDS がアーカイブ済みであるとき、または次のことが当てはまる場合が考えられます。

- その OLDS がすでに再使用されている。
- PRIOLDS および SECOLDS レコードが RECON データ・セットからすでに削除されている。

IMS が SLDS を動的に割り振られるようにするためには、SLDS 装置タイプの動的割り振りマクロ (DFSMDA) を指定する必要があります。動的割り振りに必要なデータ・セット名とボリューム情報は DBRC が提供します。

リカバリー・ログ・データ・セット:

ログ保存ユーティリティーを実行して SLDS を作成する場合、リカバリー・ログ・データ・セット (RLDS) の作成も要求できます。RLDS は、DASD、テープ、またはその他の大容量ストレージ・デバイスに保管できます。

RLDS に含まれるのは、データベース・リカバリーに必要な以下のログ・レコードだけです。

X'24' データベース・エラー・レコード

X'3730'

同期点レコード

X'4001'

チェックポイント・レコード

X'4084'

高速機能 DMCB/DMAC 制御ブロックが入るチェックポイント・レコード

X'4098'

高速機能チェックポイントの終了を表すチェックポイント・レコード

X'42' チェックポイント ID レコード

X'5612'

フェーズ 2 終了コミット・レコード

X'5701'

データベース開始更新レコード

X'59' 高速機能データベース変更レコード

X'505x'

データベース変更レコード

IMS は、RECON データ・セット内の PRILOG および SECLOG レコードの RLDS 情報を維持します。DBRC は、データベース・リカバリー・ユーティリティー用およびデータベース変更累積ユーティリティー用の JCL の作成時に、可能であれば常に SLDS の代わりに RLDS を使用します。RLDS に含まれている情報は SLDS の情報より少ないため、SLDS でなく RLDS を使用する方が効率的です。

再始動データ・セット:

IMS は、再始動データ・セット (RDS) にシステム・チェックポイント情報を書き込みます。それぞれのチェックポイントでは、IMS はチェックポイント ID テーブルを作成または更新します。IMS は IMS 再始動時にこのテーブルを使用して、システムをどのチェックポイントから再始動するのかを判別します。

再始動時になんらかの理由で RDS が使用不能な場合は、IMS は必要なチェックポイント情報をログから入手できます。ただし、ログのみを使用する場合は再始動処理がかなり長引く可能性があります。

一般的には、ユーザーが RDS の内容を知っている必要はありません。IMS は、必要とする情報を RDS 内で検索し、その情報を再始動時に自動的に使用します。

RECON データ・セット (RECON data set):

DBRC は、情報を自動的に RECON データ・セットに記録します。RECON データ・セットは両方とも同一の情報を含んでいるため、本書ではこれらを参照する場合は単一のデータ・セットとして扱います。

以下に示すように、IMS は RECON データ・セットを多くの状況で使用します。

- ウォーム・スタート時、通常再始動時、および緊急時再始動時。RECON データ・セットは、DBRC に登録されている DBDS ごとに、最新のログ・データが OLDS または SLDS のいずれのデータ・セットに含まれているのかを示します。
- ロギング時。RECON は当該の OLDS に対する RECON の最新の状況およびその OLDS がアーカイブ済みかどうかを示す。
- リカバリー・ユーティリティーの場合。DBRC が正しいデータ・セットを選択する。

z/OS ログ・データ・セット:

IMS 共通キュー・サーバー (CQS) は、IMS 共用キュー内のデータに関する情報を z/OS ログ・データ・セットに記録します。z/OS システム・ロガーは、OLDS が IMS のために果たす役割と同じ役割を CQS に対して果たします。このロガーは、CQS がカップリング・ファシリティ内の構造をリカバリーして障害後に再始動するのに必要なすべての情報を記録します。

CQS は、カップリング・ファシリティ・リスト構造のそれぞれの対ごとのログ・レコードを別々のログ・ストリームに書き込みます。このログ・ストリームは、この構造の対を共用するすべての CQS サブシステム間で共用されます。z/OS はこれらのログ・ストリームをマージして、リカバリーを可能にします。

関連資料: z/OS システム・ロガーの詳細については、「z/OS MVS プログラミング: アセンブラー・サービス ガイド」の『システム・ロガー・サービスの使用』を参照してください。

CQS システム・チェックポイント・データ・セット:

各 CQS サブシステムは、カップリング・ファシリティ内の構造対ごとのシステム・チェックポイント・データ・セットを維持します。

CQS サブシステムは、システム・チェックポイントを取るたびに、制御情報をこのデータ・セットに書き込みます。システム・ロガー・ログ・ストリームにログ・レコードも書き込みます。

このシステム・チェックポイント・データ・セットは CQS サブシステム間で共用されません。

関連概念:

214 ページの『CQS システム・チェックポイントの使用』

CQS 構造リカバリー・データ・セット:

CQS サブシステムは、構造チェックポイントを取るたびに、メッセージ・キューのスナップショットを構造リカバリー・データ・セットに書き込みます。システム・ロガー・ログ・ストリームにログ・レコードも書き込みます。構造リカバリー・データ・セットは、メッセージ・キューのリカバリーに使用されます。

シスプレックス内の CQS サブシステムは、この構造リカバリー・データ・セットを共用します。それぞれの構造ごとに 1 対 (2 つ) のデータ・セットが存在します。CQS は、チェックポイントのたびにこの 2 つのデータ・セットを交互に使用します。

関連概念:

224 ページの『CQS 構造リカバリー』

ログ・レコードのアーカイブ:

オンライン・システムの場合は、OLDS から SLDS へのログ・レコードのアーカイブを自動的にでも手動でも開始できます。

自動アーカイブ:

ご使用のシステムがアクティビティーの多いラージ・システムである場合は、自動アーカイブを使用すればアーカイブ処理への介入を最小限にできます。自動アーカイブにより、絶えずロギングをモニターしてアーカイブすべきときを判別する必要がなくなります。

デフォルトでは、IMS は各 OLDS をそれが満杯になったときにアーカイブします。しかし、IMS が OLDS をアーカイブする前に満杯になっていなければならない OLDS の個数を指定して、アーカイブが行われる頻度を制御できます。自動アーカイブを制御するには、ARC= 実行パラメーターを使用するか、/START コマンドの AUTOARCH キーワードを使用します。

推奨事項: IMS で OLDS を再利用できるようにするには、前もって OLDS をアーカイブしておく必要があります。OLDS スペースが使い尽くされてしまわないように、必ずそれを回避できるだけの頻度でアーカイブを行うようにしてください。OLDS スペースが使い尽くされてしまうと、IMS は OLDS スペースが使用可能になるまで待つこととなります。

関連資料: ARC= 実行パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。/START コマンドの AUTOARCH キーワードについて詳しくは、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

手動アーカイブ:

まれにまたは不定期にしかアーカイブを行わない場合は、ユーザーが自らアーカイブを開始できます。手動でログをアーカイブするには、ログ保存ユーティリティー (DFSUARC0) を使用してください。

DBRC GENJCL.ARCHIVE コマンドを使用すると、ログ保存ユーティリティー用の JCL を生成できます。このコマンドは、DBRC コマンド要求、データベース・リカ

バリー管理ユーティリティー (DSPURX00) または IMS オンライン・コマンド (/RMGENJCL) を使用して出すことができます。

SLDS は、DASD またはテープに置くことができます。バッチ・システムの場合も、SLDS をディスク上に置いているのであれば、アーカイブはディスク・スペースを解放するのに役立ちます。ログ保存ユーティリティーを使用して、SLDS を DASD からテープにコピーしてください。通常 DASD とテープではそれぞれブロック・サイズが異なるため、このユーティリティーはログ・レコードをコピーするときにそれらをブロック化し直します。

IMS は、OLDS がいっぱいになるか、またはクローズされるか、あるいはその両方が起こると、必ず DBRC に通知します。DBRC は RECON データ・セットを更新し、OLDS が保存可能になったことを示します。

ログ保存ユーティリティーを使用すると、保存される OLDS が IMS によって連続して作成されたものである限り、複数の OLDS を単一の SLDS に保存することができます。ユーティリティーに与えられる JCL は、保存される OLDS とどれだけの OLDS が保存されるかを定義します。DBRC の GENJCL 機能により、次の指定を行うことができます。

- 作成する JCL に含める必要のある OLDS
- まだ保存されていないすべての OLDS を含めること

指定したすべての OLDS が正常に保存されると、DBRC は RECON データ・セットを更新し、それらの OLDS がオンライン・システムによって再利用可能になったことを示します。ログ保存ユーティリティー・ジョブが失敗した場合は、ジョブを再実行してください。

自動保存を指定していない場合は、このユーティリティーを実行する JCL を作成する必要があります。自動保存を指定すると、IMS は、指定の数の OLDS がいっぱいになるかクローズされたときに、DBRC GENJCL 機能を呼び出してユーティリティー用の JCL を生成します。

GENJCL 出力用の DBRC JCLOUT DD ステートメントが内部読み取りプログラムに向けられると、保存ジョブが自動的に開始されます。以下の図は、ログ保存ユーティリティーの概要を示します。

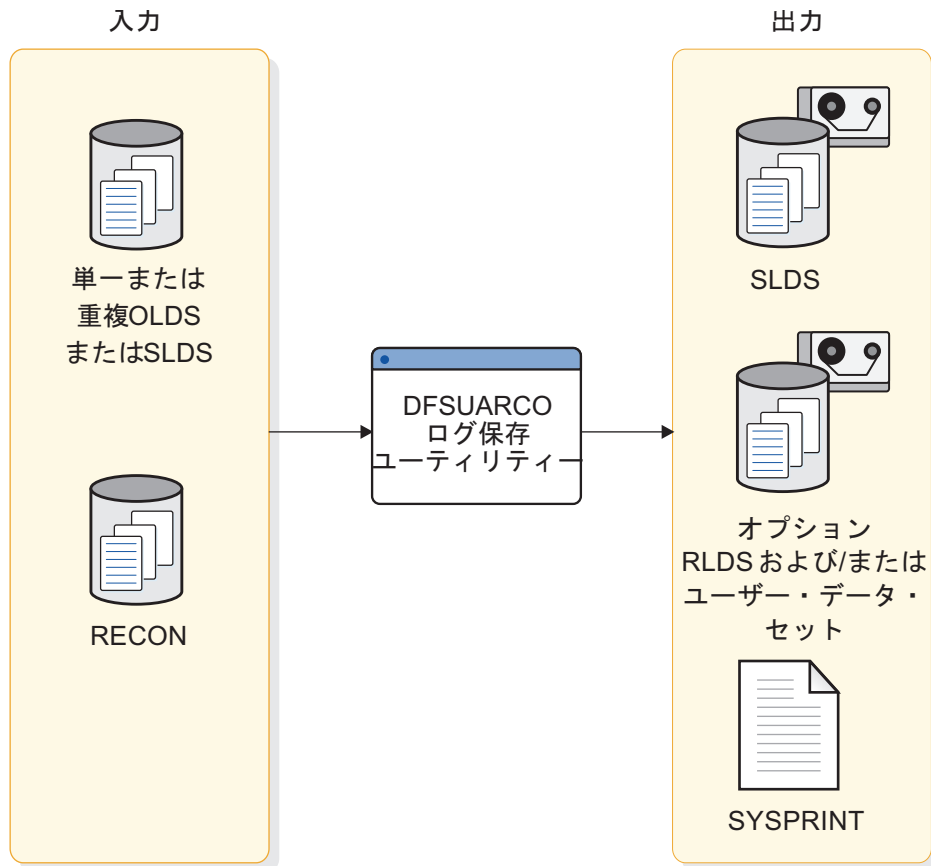


図 24. ログ保存ユーティリティの概要

SLDS または RLDS のコピー:

ログ保存ユーティリティを使用して SLDS または RLDS を新規データ・セットにコピーできます。ただし、GENJCLARCHIVE コマンドを使用してこれらのデータ・セットをコピーするための JCL を生成することはできません。

また、ログ保存ユーティリティを使用して RLDS またはユーザー・データ・セットを作成することもできます。以下のいずれかを行う場合は、ログ保存 COPY 制御ステートメントを使用してください。

- ログ・レコードのコピー先としたいユーザー・データ・セットを指定する。
- ユーザー・データ・セットにコピーする必要があるログ・レコードを判別する。
- データベース・リカバリーに必要なログ・レコードをすべて RLDS にコピーすることを指定する。


アーカイブのカスタマイズ:

ログ・レコードを処理して特定のログ・レコードをユーザー・データ・セットにコピーするためのユーザー出口ルーチンを作成できます。例えば、バッチ・メッセージ処理プログラム (BMP) を再始動するのに必要なすべてのレコードをユーザー・データ・セットにコピーできます。

アーカイブをカスタマイズするには、ログ保存ユーティリティの制御ステートメントを使用して、出口ルーチン用のエントリー・ポイントを指定してください。以下に、IMS が出口ルーチンに制御を与える時点を示します。

- ログ保存ユーティリティの初期設定時
- IMS が OLDS を読み取るとき
- ログ保存ユーティリティの終了時

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:

 ログ保存ユーティリティ (DFSUARC0) (システム・ユーティリティ)

ログのトレース:

ロギング・アクティビティをトレースしてパフォーマンス上の問題または IMS の問題を診断できます。あらゆるトレースの場合と同様に、ログ・トレースを使用する場合にトレードオフの関係にあるのは、診断能力の向上とトレースの実行によるオーバーヘッドです。

ログのトレースによるオーバーヘッドは、他のタイプのトレースを実行する場合よりも大きくなる可能性があります。特に、外部トレースを要求する場合、つまり、トレース自体がログに記録される場合はそうです。しかし、外部トレースを外部トレース・データ・セットに書き込むようにすれば、OLDS に余分な負担をかけずに済みます。

次のいずれか 1 つの方法でログ・トレースを指定してください。

- IMS の初期化時に、OPTIONS 制御ステートメントで DLOG パラメーターを使用します。
- /TRACE コマンドを使用して、DLOG キーワードを省略します。ログ・トレースのオン/オフと、ログ・トレースを OLDS と外部トレース・データ・セットのどちらのログに記録するかについての制御を行えます。

高速機能ロギングの削減:

IMS が VSAM 制御インターバル (CI) 全体を書き込む前に DEDB に対する更新をストレージ内に保持するため、ユーザーは、高速機能データに対するロギングを削減できます。ログのボリュームは、置換 (REPL) 呼び出しの際に各ログ・レコードの変更されたデータのみをロギングするようにすれば削減されます。ただし、ロギングを削減できるのは、セグメントの長さが未変更のままである場合のみです。

IMS プロシージャーまたは DBC プロシージャーの LGNR パラメーターを使用して、IMS が VSAM 制御インターバル (CI) 全体をログに記録する前に保持されるべき高速機能 DEDB バッファーの変更の最大数を決定してください。LGNR パラメーターに使用する値を評価するには、高速機能ログ分析ユーティリティ (DBFULTA0) を使用してください。

関連概念:

 DBCTL での高速機能 EXEC パラメーター (システム定義)

☞ DCCTL または DB/DC での高速機能 EXEC パラメーター (システム定義)

関連資料:

☞ 高速機能ログ分析ユーティリティー (DBFULTA0) (システム・ユーティリティー)

バッチ・ジョブ・ログのトラッキングのための **DBRC** の使用:

IMS オンライン・サブシステムはログのトラッキングに常に **DBRC** を使用しますが、バッチ・ジョブは **DBRC** を使用する必要がありません。バッチ・ジョブの場合に **DBRC** を使用すると、**DBRC** はどのバッチ・ジョブがどの **SLDS** を作成したかについてのみのトラッキングを行います。

推奨事項: **DBRC** をバッチ・ジョブに使用して、バッチ **SLDS** のトラッキングを手動で行わなくても済むようにしてください。

読み取り専用 (PROCOPT=G) バッチ・ジョブについてはログを作成する必要はありませんが、更新ジョブの場合はログを作成しなければなりません。**DBRC** を使用する更新ジョブの場合は、ログ・データ・セット用の **JCL** の中で **DD NULLFILE** または **DD DUMMY** を使用することはできません。

IMS のシステム定義時に **IMSCTRL** マクロで **DBRC** キーワードを使用して、**DBRC** の使用を指定してください。IMS の実行中にプロシージャ **DBBBATCH** および **DLIBATCH** の中で **DBRC=** 実行パラメーターを使用すれば、システム定義時に指定した値を変更できます。システム定義時に **FORCE** キーワードを指定した場合は **DBRC** を使用する必要があります。ただし、ログ保存ユーティリティー (バッチのみ)、ログ・リカバリー・ユーティリティー、またはバッチ・バックアウト・ユーティリティーを実行する場合は除きます。

ログに記録されたデータベース変更のリカバリーのための統合:

データベース変更累積ユーティリティー (**DFSUCUM0**) を使用して、**SLDS** または **RLDS** にあるデータベース変更レコードを抽出し、統合できます。データベース変更累積ユーティリティーは、データベース変更レコードをリカバリー用に最適化し、変更累積データ・セットに保管します。変更累積データ・セットは、データベース・リカバリー・ユーティリティー (**DFSURDB0**) への入力として使用できます。

IMS を実行しているうちに、**SLDS** または **RLDS** の数が増えていきます。これらのデータ・セットを使用して損失または損傷したデータベースをリカバリーできますが、これらを変更せずに使用するのには次の理由により非効率的です。

- 各 **SLDS** または **RLDS** には、IMS サブシステム全体のアクティビティーとすべてのデータベース用のすべてのデータ・セットのレコードが含まれている。しかし、データベースをリカバリーする際にリカバリーするのは通常は単一のデータ・セットだけです。したがって、**SLDS** および **RLDS** の内容の多くは適用されません。
- **SLDS** および **RLDS** には、1 つ 1 つのデータベース・レコードに加えられたそれぞれの変更が古い順にすべて保管されている。あるレコードがデータ・セットが最後にバックアップされてから現在までに 100 回変更されたとすると、**SLDS** または **RLDS** にはその 100 回分のすべての変更が含まれていることになりま

す。しかし、リカバリーの際には、関係があるのは当該のデータ・セットが逸失した時点においてそのデータがもっていた値だけであり、それ以外の 99 の変更は無関係です。


データベース変更累積ユーティリティを使用して、累積されている SLDS および RLDS 全体を前もってソートし、さらに、それらを圧縮して合理化できます。このユーティリティは、以下のことを行います。

- データベースのリカバリーに関連したログ・レコードだけを取り出す
- それらのレコードをデータベース内のデータ・セット別にソートする
- 個々のレコード内の各部分での最新の変更をすべて検索する。

このユーティリティが変更累積データ・セットを作成するにつれて、IMS は連続して出現する単一文字 (ブランクやゼロなど) を圧縮します。IMS は、リカバリーの際にこのデータを再び展開します。

データベース変更累積ユーティリティの実行は必須ではありませんが、これを定期的を使用すると、データベース・リカバリーが高速化されます。その代わりに、データベース変更累積ユーティリティをリカバリーが必要になったときだけ (データベース・リカバリー・ユーティリティを実行する直前に) 実行しても構いません。まだ累積されていないログ情報の量によっては、データベース・リカバリー・ユーティリティだけを実行するよりも、この 2 つのユーティリティを実行した方が、リカバリーに要する合計時間が削減される可能性があります。

関連資料:

 データベース変更累積ユーティリティ (DFSUCUM0) (データベース・ユーティリティ)

データベース変更累積ユーティリティへの入力:

アーカイブ済みログ・データ (SLDS および RLDS) をデータベース変更累積ユーティリティへの入力として使用するのに加え、IMS ログのサブセットまたは以前の変更累積データ・セットを使用することもできます。このユーティリティは、累積された変更を新しい変更累積データ・セットに書き込みます。

ログ・データがテープ上にある場合は、データベース変更累積ユーティリティへの入力としてすべてのログ・ボリュームとログ・ボリュームのサブセットのどちらかを指定しても構いません。ログ・ボリュームのサブセットを指定した場合は、DBRC はそのサブセットが完全なものであるかどうかをそれぞれの DBDS ごとに検査します。ある DBDS についてログ・ボリュームのサブセットが完全なものであるとは、以下の条件がすべて真である場合を言います。

- そのサブセット内の最初のボリュームが、以下のいずれかのイベントの発生以降に当該 DBDS に対して行われた最初の変更を含んでいるボリュームである。
 - 最後の変更累積
 - 最後のイメージ・コピー
 - DBRC によるこのエリア用 ALLOC レコードの作成 (イメージ・コピーが同時イメージ・コピーであった場合)

- 更新が実行されている最中にイメージ・コピーが取られた場合に、すべての更新が確実に考慮されるようにするために、イメージ・コピーの実行時より前のログで検出されたチェックポイント ID が含まれている。
- 残りのボリュームは順になっている。
- データ共用環境では、DBDS への変更を含んでいるすべてのログが含まれている。

ログ・ボリュームのサブセットを指定するには、DBRC GENJCL.CA コマンドを使用してください。特定の数のログ・ボリュームをボリュームによって (VOLNUM キーワードを使用) またはタイム・スタンプによって (CATIME キーワードを使用) 要求できます。

使用するログ・ボリュームのサブセットが完全であっても不完全であっても、後でデータベース変更累積ユーティリティを実行する際に、変更累積データ・セットを入力として使用できます。しかし、変更累積データ・セットをデータベース・リカバリー・ユーティリティへの入力として使用できるのは、そのデータ・セットが完全なログ・サブセットに相当するものである場合だけです。

関連資料:

GENJCL.CA コマンド (コマンド)

データベース・リカバリーおよびデータベース変更累積ユーティリティの除去時刻:

除去時刻は、DBRC がリカバリーの開始点としてイメージ・コピーまたは HALDB オンライン再編成を使用して、リカバリーする必要がある可能性があるデータベースまたはエリアの変更の収集を開始するために使用するタイム・スタンプです。このタイム・スタンプより前の変更は、変更累積データ・セットから除去され、データベースのリカバリーに考慮されません。

イメージ・コピーの除去時刻

更新が行われていない (データベースの RECON 内にアクティブの割り振りレコードがない) 場合は、イメージ・コピーの実行時間が除去時刻として使用されます。イメージ・コピーの実行時に更新が行われている場合、DBRC は、イメージ・コピーの実行時より前のログ上のチェックポイント・タイム・スタンプを検査します。全機能データベース (HALDB を含む) の場合、前のログ上の 1 つのチェックポイントのみが考慮されます。高速機能データベースの場合、前のログ上の 2 つのチェックポイントが考慮されます。除去時刻が適時に前進し続けるようにするために、IMS によるデータベースの更新のたびに、各ログ上にチェックポイントが存在することを確認してください。

全機能データベースと高速機能データベースの両方とも、以下のタイプのイメージ・コピーの場合は、イメージ・コピーの実行時間が除去時刻として選択されます。

- データベースが更新に利用不可の間に、データベース・イメージ・コピー (DFSUDMP0) ユティリティを使用して、イメージ・コピーを作成した (BATCH)。

- データベースが更新処理に利用不可の間に、データベース・イメージ・コピー 2 ユーティリティーが DFSMS 高速レプリケーションを呼び出して、イメージ・コピーを作成した (SMSOFFLC)。
- データベースが更新処理に利用不可の間に、データベース・イメージ・コピー 2 を使用して、イメージ・コピーを作成した (SMSNOCIC)。
- オンライン・データベース・イメージ・コピー・ユーティリティーを使用して、イメージ・コピーを作成した (ONLINE)。

その他のタイプのファジー・イメージ・コピーの場合、DBRC は、ファジー・イメージ・コピーの実行時にデータベースが割り振られている限り、それより前のチェックポイント ID を使用して、すべての変更がデータベース・リカバリーまたはデータベース変更累積ユーティリティーに確実に含まれるようにします。ファジー・イメージ・コピーの実行時にデータベースが IMS システムによって割り振られていない場合は、イメージ・コピーの実行時間が、そのデータベースの除去時刻として選択されます。

ファジー・イメージ・コピーが取られた時点のすべての IMS システム上のアクティブ割り振りが考慮され、IMS システムでアクティブ割り振りが検出された場合、以下のように、全体で最も古い除去時刻が選択されます。

- データベースが更新処理に使用可能の間に、データベース・イメージ・コピー・ユーティリティーを使用して、イメージ・コピーを作成した (CONCUR)。並行イメージ・コピーは、更新も実行されているときに行われる「ファジー」コピーです。データ・セットは、イメージを完成するためにログを使用します。
- データベース・イメージ・コピー 2 ユーティリティーが DFSMS 高速レプリケーションを呼び出して、更新処理と並行してイメージ・コピーを取った (SMSONLC)。このイメージ・コピーは「ファジー」コピーであるため、データ・セットを使用可能な状態にリカバリーするには、ログを適用する必要があります。
- データベースが更新処理に使用可能な間に、データベース・イメージ・コピー 2 を使用して、イメージ・コピーを作成した (SMSCIC)。このイメージ・コピーは「ファジー」コピーであるため、データ・セットを使用可能な状態にリカバリーするには、ログを適用する必要があります。

DBRC は、以下のように、ログ上のチェックポイント ID を使用して、ファジー・イメージ・コピーでの除去時刻を判別し、すべての更新が確実に含まれるようにします。

- 全機能データベース (HALDB) の場合: イメージ・コピーの開始時刻より前のログ・ボリュームで検出されたチェックポイント ID について、そのチェックポイント ID が割り振りより前である場合は、それを選択します。そうでない場合は、割り振り時刻が、除去時刻として選択されます。
- 高速機能データベースの場合: イメージ・コピーの開始時刻より前のログ・ボリュームで、少なくとも 2 つチェックポイントが検出された場合、そのログ・ボリュームの開始時刻が割り振りより前であれば、そのログ・ボリュームの開始時刻が使用されます。そうでない場合は、割り振り時刻が、除去時刻として選択されます。

HALDB のリカバリーのための入力として使用できる、選択されたイメージ・コピーの除去時刻より後の、HALDB オンライン再編成 (OLR) が検出された場合、そ

の OLR の実行時間が除去時刻として使用されます。

変更累積グループ:

DBRC を使用して、データベース変更累積ユーティリティによる変更累積の対象となる DBDS をグループに分類できます。これらの DBDS のグループは、変更累積グループと呼ばれます。

変更累積グループは、DBRC INIT.CAGRP コマンドを使用して CAGRP レコードを RECON データ・セットに書き込めば定義できます。このレコード内で、DBRC は、データベース名とデータ・セット DD 名によって識別される変更累積グループを構成する DBDS と、RECOVPD キーワードによって識別されるリカバリー期間をリストします。変更累積グループは 1 グループ当たり最大 32,767 のメンバーを持つことができます。

INIT.CAGRP コマンドを使用して変更累積グループを定義するためには、その前に INIT.DBDS コマンドを使用して DBRC に各メンバーを識別させる必要があります。各 DBDS は 1 つの変更累積グループにしか所属できません。

CHANGE.CAGRP コマンドを使用すれば、変更累積グループのメンバーを追加または削除できます。

制約事項: データベース安全性が損なわれる可能性があるため、データベース変更累積ユーティリティの実行中は CHANGE.CAGRP コマンドを出さないでください。

将来の利用のための変更累積データ・セットの定義:

特定の変更累積グループ用の変更累積データ・セットを将来の使用のために定義しておけます。DBRC INIT.CA コマンドを使用して、これらのデータ・セットが存在することを DBRC に通知し、さらにグループの定義時に INIT.CAGRP コマンドに REUSE キーワードを指定してください。

INIT.CAGRP コマンドの GRPMAX キーワードで、定義できる変更累積データ・セットの数を指定します。

変更累積データ・セットの再使用:

DBRC が古い変更累積データ・セットを再使用できるようにするには、REUSE キーワードを使用して変更累積グループを定義し、さらに GENJCL.CA コマンドを使用して、データベース変更累積ユーティリティ・ジョブのための JCL を生成してください。

データベース変更累積ユーティリティは、特定の変更累積グループ用の使用可能変更累積データ・セットがすべて使用済みで、変更累積データ・セットの数がすでに最大数に達している場合は、最も古い変更累積データ・セットを再使用します。変更累積データ・セットの再使用とは、DBRC が当該の変更累積データ・セットのデータ・セット名、ボリューム、物理スペース、および RECON データ・セット内のレコードを、あたかもそれらが空の変更累積データ・セットのものであるかのごとくに使用することを意味します。

データ・セットを再使用するのではなく、NOREUSE キーワードを使用してグループを定義した場合は、DBRC は最も古い変更累積データ・セット用の RECON レコードを削除します。このケースでは、DBRC はそのデータ・セットをスクラッチしません。しかし、DBRC は今後はそのデータ・セットを認識しなくなるため、ユーザーがそのデータ・セットのスクラッチまたはトラッキングを行う必要があります。

ログ・データ・セットについての選択の指定:

このトピックでは、ロギングに關与する以下の各種のデータ・セットを定義する際に行う必要のある選択について説明します。

関連資料: CQS データ・セットの定義方法は、「IMS V15 システム定義」を参照してください。

オンライン・ログ・データ・セットの定義:

これらのトピックでは、OLDS の定義に關連した作業について説明します。

OLDS ロギングについての単一か重複かの選択:

IMS では、ログ情報を単一のデータ・セットまたは 2 つの同一のデータ・セットに記録できます。

定義: 単一ロギング は単一のデータ・セットを使用し、重複ロギング は 2 つのデータ・セット (両方のデータ・セットは同一) を使用します。110 ページの『オンライン・ログ・データ・セット』で説明しているように、使用されるのが単一ロギングであっても重複ロギングであっても、IMS はデータ・セットのセットに情報を書き込みます。

推奨事項: OLDS はシステム保全性にとって最も重要であるため、可能な限り重複ロギングを使用してください。OLDSDEF ステートメントを使用して、IMS.PROCLIB 内で重複ロギングを指定してください。

重複ロギングについては、IMS には (1 対の) OLDS のどちらか一方に入出力エラーが起こったときのためのオプションが 2 つ用意されています。最初のオプションは、その対を廃棄して問題のない対に切り替えるためのものです。この振る舞いは、まさしく単一ロギング・モードの場合と同様です。

2 番目のオプションは、対になっている OLDS の両方に障害が起こらない限り、IMS がその OLDS の対を廃棄することがないようにするためのものです。そのようにしておくこと、問題のなかった OLDS 対のすべてに障害が起こった場合に、IMS は単一ロギング・モードに機能を低下させて、それぞれの対の問題のない方の OLDS だけを使用することが可能となります。この時点からは、IMS は当初から単一ロギング・モードであったかのように振る舞います。また、このオプションは操作手順の単純化も実現します。例えば、(単一または重複) ロギングを使用していて書き込みエラーが発生した場合は、ログ・リカバリー・ユーティリティのみを使用するだけで OLDS を終結処理できます。

本質的に、IMS がその中で稼働するロギング状態としては、次の 3 つがあります。

- 単一ロギング。この場合は、冗長性が一切なく、エラー発生時には最高の維持が行われます。問題のない OLDS があと 2 つだけになると、IMS は終了します。
- DEGRADE=NO の指定付きの重複ロギング。この場合の振る舞いは、単一ロギングの場合と同様です。ただし、データが 1 つのログにではなく 2 つのログに書き込まれることだけが異なります。さらに、単一ロギング・モードのように、問題のない OLDS の対があと 2 つだけになると IMS は終了します。
- DEGRADE=YES の指定付き重複ロギング。このケースでは、OLDS の各対に少なくとも 1 つの書き込みエラーがある状態になると、IMS は単一ロギング・モードに切り替えて、各対の問題のない方のすべての OLDS を対象としてロギングを開始します。

3 つのケースのすべてにおいて、IMS は、書き込み可能なデータ・セットがあと 2 つだけになった時点で終了します。なんらかの理由で IMS が書き込める問題のないデータ・セットが 1 つもなくなった場合は、IMS は ABENDU0616 を出して終了します。ログ・エラーについての詳細は IMS V15 オペレーションおよびオートメーションを参照してください。

OLDS の数の定義:

IMS を始動するためには、少なくとも 3 つの OLDS (または OLDS の対) を定義する必要があります。ただし、追加の OLDS (最大 100 まで) を定義できます。

使用する OLDS は、初期設定時に OLDSDEF ステートメントを使用して IMS.PROCLIB データ・セット内で定義する必要があります。その後は追加の OLDS を動的に割り振ることができます。DFSMDA マクロを使用して、動的に割り振りたい OLDS を指定してください。後で /START OLDS コマンドを使用すれば、OLDS を追加できます。

関連資料: DFSMDA マクロの詳細については、「IMS V15 システム定義」を参照してください。

定義する OLDS の数を決める際には、アーカイブの頻度とオンラインにしておきたいデータの量を考慮してください。IMS は、OLDS をアーカイブした後でのみ再使用します。定義する OLDS の数とアーカイブの頻度との間には一貫性がなければなりません。例えば、アーカイブを頻繁に行うのであれば、通常は定義する OLDS の数を少なくするように計画します。

推奨事項: システム障害を回避するために、重複ロギングを使用する場合でも、最小数より多い OLDS を定義することをお勧めします。

各 OLDS のスペースの定義:

各 OLDS のサイズを定義するときには、SLDS のサイズと場所 (DASD またはテープ) も考慮する必要があります。SLDS がテープ上にある場合は、SLDS ポリユームのサイズと、OLDS をアーカイブする予定頻度を考慮に入れてください。

SLDS が DASD 上にある場合は、アーカイブされる OLDS をすべて入れられるだけの十分なスペースを割り振る必要があります。OLDS のアーカイブ時に各 OLDS の SLDS ポリユームが満杯になっているようにするため、各 OLDS に対して十分なスペースを割り振ってください。あるいは、2 つの OLDS が満杯になった

ときにアーカイブを開始すればよいように、それぞれの OLDS のサイズを SLDS ボリュームの半分にすることも推奨されています。

また、各 OLDS のサイズを定義するときには、緊急時再始動や BMP の再始動などのようなタスクを行うためにオンラインにしておきたいデータの量も考慮してください。例えば、ご使用のシステムのアクティビティーが非常に多く、大量のトランザクションを処理する場合は、より大きな OLDS を定義することを検討してください。大きな OLDS を定義しておくこと、緊急時再始動に必要なレコードがオンラインになる可能性が高まります。

OLDS のブロック・サイズの定義:

OLDS のブロック・サイズを決定する主な要因は OLDS 装置のトラック・サイズです。OLDS のブロック・サイズは装置のトラック・サイズを超えることはできません。トラック当たりのログ・データ量を最大にするブロック・サイズ (例えば、3380 または 3390 DASD の場合ハーフトラック) を定義します。IMS は満杯になった OLDS バッファのみを OLDS に書き込むため、大きい OLDS ブロック・サイズを設定すると、DASD スペースをより効率的に使用することができます。

OLDS のブロック・サイズの選択は注意深く行う必要があります。なぜなら、OLDS が確立した後に OLDS のサイズを変更したい場合は、オンライン作業を停止し、すべての OLDS をアーカイブし、さらにそれらのスクラッチと再割り振りを行って、それらのブロック・サイズが同一のものであることを確認しなければならないからです。スクラッチと再割り振りを行う際には、OLDS エントリーを DBRC RECON データ・セットから削除する必要があります。OLDS のブロック・サイズを、2048 の倍数 (z/Architecture[®] モードで実行していない) から 4096 の倍数 (z/Architecture モードで実行している) に変更することにより、WADS が OLDS ブロック・サイズの変更に従って再フォーマットされることを確実にするには、/NRE FORMAT WA コマンドまたは /NRE FORMAT ALL コマンドを実行する必要があります。OLDS のブロック・サイズの変更後は、SLDS からだけしかオンライン作業を再始動できません。

推奨事項: 再始動後にはただちにチェックポイントを取ってください。これ以降の再始動で、SLDS ではなく、新しい OLDS を使用できるようにするためです。

各 OLDS のブロック・サイズは同じでなければなりません。OLDS のブロック・サイズは、以下の要件をすべて満たしている必要があります。

- IMS が z/Architecture モードで実行していない場合、ブロック・サイズは 2048 バイト (2 KB) の倍数でなければならない。IMS が z/Architecture モードで実行している場合、ブロック・サイズは 4096 (4 KB) の倍数でなければならない。
- ブロック・サイズは最小でも 6144 でなければならない。

推奨事項: 8 KB を最小限とするか、あるいは最大のメッセージ・セグメントの長さを使用してください。

- ブロック・サイズは最大でも 30,720 バイトを超えてはならない。これは、BSAM がサポートする 2048 の倍数の中で最大のものです。

- 拡張フォーマット OLDS がサポートされている場合は、何個のブロックが 1 つのトラックに収まるかについての計算をするときに、その計算を行う前に、この他に 32 バイトの SAM オーバーヘッドを IMS ブロック・サイズに追加する必要があります。

大規模順次データ・セット・サポートを使用している場合は、OLDS が 65,535 トラックを超えてもかまいません。このサポートを利用するには、65,535 を超えるトラックを持つハードウェアを使用する必要があります。

初期設定中に、IMS は OLDS に指定されたブロック・サイズが最大ログ・レコードを確実に処理できるだけの十分な大きさであるようにします。指定されたブロック・サイズが小さすぎる場合は、IMS はその OLDS データ・セットを廃棄して、次の OLDS を調べます。初期設定の終了時になっても使用可能な OLDS の対が最小限の 3 つより少ない場合は、IMS は異常終了 0073 で終了します。

OLDS 用の装置の定義:

OLDS をアクセス不能な状態にするシステム障害が起こってもシステム全体が停止しないように OLDS を構成してください。OLDS 重複ロギングを定義する場合は、1 つの OLDS 対のそれぞれのデータ・セットを別々の装置に定義し、さらに可能であれば、別々の制御装置およびチャンネルに定義してください。

以下の図は、複数の装置からなる OLDS 構成を示します。この構成では、1 つの装置が故障しても、まだ他の 2 つの装置上の OLDS にアクセスできます。例えば、1 次 OLDS 1 および 2 次 OLDS 1 へのロギング中に装置 B に障害が起きた場合でも、ログ保存ユーティリティーへの入力として 1 次 OLDS 1 を使用できます。そして、IMS は、次に使用可能な OLDS の対 (1 次 OLDS 3 および 2 次 OLDS 3) を使用してロギングを継続します。1 次 OLDS 3 および 2 次 OLDS 3 のどちらもアーカイブされていないとき、IMS は、次に使用可能な対 (1 次 OLDS 6 および 2 次 OLDS 6) を使用してロギングを継続します。1 次 OLDS のステップは次の装置へと移り、シーケンスは最初に戻ります。

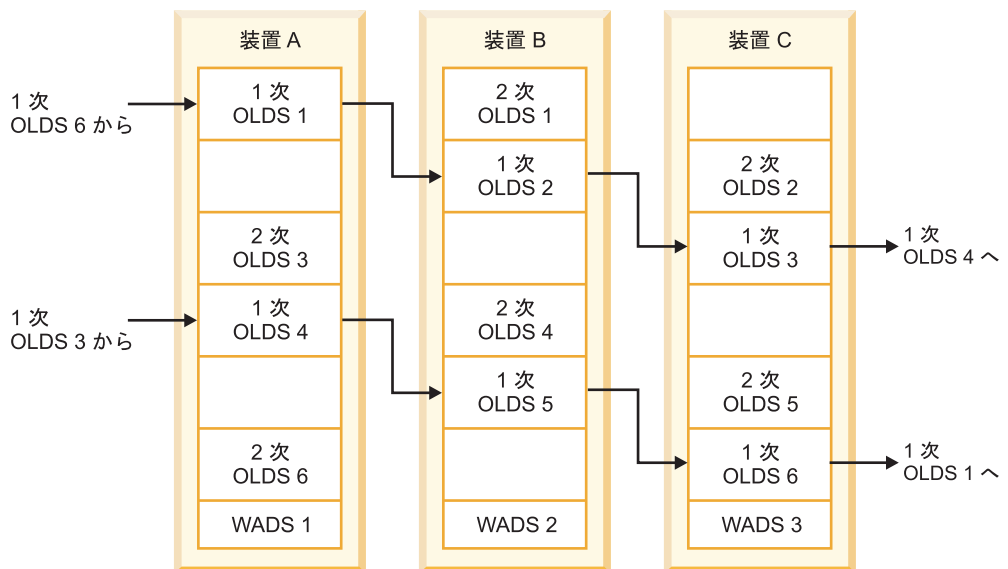


図 25. OLDS 構成の例

OLDS の特性の変更:

アーカイブ済みの OLDS を消去して再割り振りする前に、DBRC DELETE.LOG
OLDS(ddname) SSID (name) コマンドを使用して、DBRC RECON データ・セット
内のこの OLDS 用のログ制御レコードを削除する必要があります。

スクラッチされて再割り振りされた OLDS 用のログ制御レコードを削除しなければ
ならないのは、そのログ制御レコードはその OLDS がアーカイブ済みであるという
ことだけしか指示していないからです。この OLDS が IMS 再始動またはバッチ・
バックアウトのために必要な場合は、DBRC はアーカイブ・ジョブによって作成さ
れた SLDS の代わりにこの OLDS を使用するよう IMS に指示します。

関連概念:

619 ページの『第 30 章 DBRC の概要』

OLDS バッファの定義:

2 から 9999 個 (デフォルトは 5) の OLDS バッファを定義できます。バッファ
ーの数は IMS の再始動時に変更可能です。

以下の状況では、OLDS バッファの数を増加することをお勧めします。

- IMS が OLDS バッファのために頻繁に待ち状態になる。
- 動的バックアウトを頻繁に行う。

DFSDFxxx PROCLIB メンバーの LOGGER セクションにある OLDSDEF ステ
ートメントの BUFNO パラメーターに、OLDS バッファの数を指定します。

IMS は、OLDS ログ・バッファが DFSMS 拡張フォーマットのデータ・セット
として割り振られており、BUFSTOR=64 が DFSDFxxx IMS.PROCLIB メンバーの
LOGGER セクションにある OLDSDEF ステートメントで指定されている場合に
のみ、OLDS ログ・バッファを 2 GB の境界より上に配置します。

システム・ログ・データ・セットの定義:

システム・ログ・データ・セット (SLDS) を定義するには、単一または重複のい
ずれかの SLDS ロギングを選択し、SLDS のブロック・サイズを定義することを検討
してください。

- SLDS ロギングについての単一か重複かの選択

OLDS および WADS の場合と同様に、SLDS に対して単一ロギングと重複ロギ
ングのどちらを行うかを選択できます。テープにアーカイブする場合は、SLDS
制御ステートメントのボリューム強制終了 (FEOV) キーワードを使用して、ボリ
ューム当たりのログ・ブロック数を指定すると、1 次ボリュームと 2 次ボリ
ュームが同じデータを含むように強制できます。IMS は、指定された数のブロック
を書き込むと、1 次データ・セットおよび 2 次データ・セットの両方で、強
制的にボリュームの終わりにします。

重複ロギングを使用するためには、それぞれの SLDS ごとに 1 次と 2 次の両
方の DD ステートメントを提供してください。

- SLDS のブロック・サイズの定義

SLDS のブロック・サイズは、アーカイブされる OLDS のブロック・サイズとは異なっていても構いません。しかし、FEOV キーワードを使用している場合は、1 次 SLDS と 2 次 SLDS のブロック・サイズは同じでなければなりません。

リカバリー・ログ・データ・セットの指定:

RLDS に対して重複ロギングを使用するには、ログ保存ユーティリティー用の JCL 内で各 RLDS ごとに 1 次と 2 次の両方の DD ステートメントを提供します。

再始動データ・セットの指定:

再始動データ・セット (RDS) に 1 シリンダー分のスペースを割り振ります。チェックポイント ID テーブルが使用するのは 1トラックですが、IMS はその他のリカバリー情報も RDS に書き込みます。

ログの内容:

IMS は、OLDS にアクティビティーを記録します。CQS は、z/OS ログ・ストリームにアクティビティーを記録します。それぞれ異なるアクティビティーは、おの別個のログ・レコードとして記録されます。

一般的には、ユーザーがログ・レコードの内容を知っている必要はありません。IMS および CQS は、リカバリーを実行する際に正しいレコードを識別してそれらを自動的に使用します。

ただし、ログに関して、以下の事項に留意してください。

- ログの縮小

ログの縮小の一環として、IMS はログ・データを圧縮します。IMS は、ログ・レコードのセグメント・データ部分に連続して出現する単一文字 (ブランクやゼロなど) を圧縮します。この圧縮は、DL/I の挿入 (ISRT)、削除 (DLET)、および置換 (REPL) 呼び出しの結果として行われる更新に適用されます。

圧縮と対になっているのが展開です。バックアウトおよびデータベース・リカバリーの実行時に、IMS はデータベース・バッファ内のデータを展開します。変更累積データ・セットの場合は、IMS は変更累積レコード内のデータを展開します。

- ロギングおよびデータ・キャプチャー出口ルーチン

IMS は、データ・キャプチャー出口ルーチン用のログ・レコードを作成してこのルーチンが呼び出されたことを示すというはしません。また IMS は、どのログ・レコードにもこの出口ルーチンの名前を書き込みません。IMS は、アプリケーション・プログラムとこの出口ルーチンを区別しません。このデータ・キャプチャー出口ルーチンを広範囲にわたって使用すると、IMS システム・アカウント情報とパフォーマンス・モニター情報が、アプリケーション・プログラムのシステム使用について、それらが実際に使用している量よりも多い使用量を示す可能性があります。

関連資料: 複雑なりカバリー問題を解決するためにログ・レコードを調べる必要がある場合は、「IMS Version 15 Diagnosis」を参照してください。

バックアップ

データ・セットのバックアップ・コピーは、リカバリー可能システムでチェックポイントが果たす目的と同じ目的を果たせます。つまりこれは、そこから処理を再始動できる場所を定義するという事です。

IMSには、次のようないくつかのタイプのバックアップ・コピーを作成できるようにするためのユーティリティーが用意されています。

データベースのバックアップ・コピー:

IMSが通常システム・チェックポイントを取るときには、IMSはDL/I用および高速機能用の内部制御情報を記録しますが、データベースの内容は一切記録しません。データベースが破損した場合には、最後のシステム・チェックポイントを調べてもそのデータベースをリカバリーすることはできません。システム・ログを調べればどのような変更が行われたのかが分かりますが、オリジナルのデータベースそのものがなければ、リカバリーは不可能です。

推奨事項: 最初にデータベースをロードした後、それらすべてのバックアップ・コピーを作成します。また、新しいバックアップ・コピーを定期的を作成することも必要です。バックアップ・コピーが新しいものであればあるほど、リカバリー時に処理しなければならないログ変更レコードの数が少なくて済むため、リカバリーに要する時間が削減されます。

IMSには、データベースのバックアップ・コピーを作成するためのユーティリティーが数種類用意されています。

- イメージ・コピー・ユーティリティー

IMSデータベース・イメージ・コピー・ユーティリティーを使用すると、データベースに対する変更の前と後にそのデータベースの『スナップショット』を取ることができます。これらのスナップショットはイメージ・コピーと呼ばれます。

定義: イメージ・コピーは、データベースの現状のままのイメージです。イメージ・コピー・ユーティリティーは、コピーの際にデータベースの物理的なフォーマットは変更しません。イメージ・コピーはデータのバックアップ・コピーで、データベースのリカバリー処理とバックアウト処理を高速化するのに役立ちます。

イメージ・コピー・ユーティリティーには、データベース・イメージ・コピー・ユーティリティー (DFSUDMP0)、データベース・イメージ・コピー 2 ユーティリティー (DFSUDMT0)、およびオンライン・データベース・イメージ・コピー・ユーティリティー (DFSUICP0) があります。

- HISAM 再編成アンロード・ユーティリティー (DFSURUL0)

このユーティリティーを使用すると、あるHISAMデータベースがアンロードされて再編成されるときに、そのデータベース全体を1回のパスで処理できます(イメージ・コピー・ユーティリティーは各データベース・データ・セットをそれぞれ個別に処理します)。このユーティリティーは、データベースがオフラインであるときに稼働します。

推奨事項: データベースをアンロードする前に、データベースのイメージ・コピーを作成します。イメージ・コピーは、アンロード処理中に障害が発生した場合に、リカバリー目的で使用できます。 HISAM 再編成再ロード・ユーティリティーを使用せずに、イメージ・コピー・データ・セットと同等のものを作成することもできます。


データベースをアンロードした後は、そのデータベースをオンラインに戻す前に HISAM 再編成再ロード・ユーティリティー (DFSURRL0) を使用します。データベースを再ロードしないと、アプリケーション・プログラムはそのデータベースを使用することはできませんが、古い編成のままのデータ・セットを使用することになります。すると、バックアップ・コピーとそのデータベース用に作成されたログ・レコードとが一致しなくなるため、そのバックアップ・コピーを使用してデータベースをリカバリーした場合はデータ安全性が損なわれてしまいます。

これらのユーティリティーは、DBRC と一緒にでも別個にでも実行できます。また、各種の z/OS ユーティリティーを使用してバックアップ・コピーを作成することもできますが、これらのユーティリティーは DBRC とは相互作用しないため、使用する IMS システムの定義のされ方によっては安全性に関する問題が起こる可能性があります。

関連概念:

663 ページの『並行イメージ・コピー』

関連資料:

 HISAM 再編成再ロード・ユーティリティー (DFSURRL0) (データベース・ユーティリティー)

メッセージ・キューのバックアップ・コピー:

メッセージ (エンド・ユーザーによって入力されたトランザクション要求と、それらのユーザーに返信される応答) は、処理される前にキューに保管されます。IMS メッセージ・キューを共用していない場合は、メッセージはメッセージ・キュー・データ・セットに保管されます。このデータ・セットはディスクと仮想記憶域に分かれて常駐します。

共用キュー環境では、IMS メッセージはカップリング・ファシリティ上で保持されます。IMS は、通常システム・チェックポイントを取るときに、データベースの内容を記録しないのとまったく同じように、メッセージ・キューの内容を記録しません。

IMS を (異常終了ではなく) 正常にシャットダウンした場合は、(カップリング・ファシリティ上ではなく) 仮想記憶域内の変更済みメッセージはすべてメッセージ・キュー・データ・セットのディスクの部分に自動的に書き込まれます。したがって、定期的に IMS をシャットダウンしている場合は、メッセージ・キューを手動でバックアップする必要はほとんどまたはまったくありません。

IMS をシャットダウンせずに長時間稼働している場合は、メッセージ・キューの定期的なバックアップが必要となる可能性があります。非共用キュー環境では、/CHECKPOINT SNAPQ コマンドを使用してメッセージ・キューをバックアップしてください。このコマンドは IMS をシャットダウンしません。

共用キュー環境では、共用キューを定期的にバックアップする必要があります。
/CQCHKPT コマンドを使用して、IMS メッセージを共通キュー・サーバーの構造
リカバリー・データ・セットにコピーしてください。

メッセージ・キューのバックアップを取っていると、メッセージ・キュー・デー
タ・セットに問題が起こったときにリカバリーに要する時間が削減されます。

システム・データ・セットのバックアップ・コピー:

バックアップ・コピーを作成する必要があるデータ・セットは、データベース・デ
ータ・セットおよびメッセージ・キュー・データ・セットではありません。
IMS システム・データ・セットのバックアップ・コピーも作成する必要がありま
す。これらには、ACB ライブラリー (IMS.ACBLIBx) および MFS ライブラリー
(IMS.FORMATx) が含まれます。

IMS には、システム・データ・セットのバックアップ・コピーを作成するための特
別なユーティリティやコマンドは用意されていません。しかし、IEBCOPY などの
z/OS ユーティリティを使用すれば、定期的にバックアップ・コピーを作成でき
ます。これらのコピーは、z/OS システム・ライブラリーのバックアップ・コピ
ーを定期的に作成する際に同時に作成できます。システム・データ・セットとデー
タベースを同時にバックアップすることもできるので便利です。

オンライン変更を行った場合は、非アクティブ・データ・セットとアクティブ・デ
ータ・セットの切り替え後にアクティブ・データ・セットのバックアップ・コピ
ーを作成する必要があります。

また、BACKUP.RECON コマンドを使用して RECON データ・セットを定期的に
バックアップすることも必要です。

リカバリー用のユーティリティおよびサービス

IMS には、リカバリー可能システムのリカバリーおよび維持に役立つユーティリ
ティとサービスが多数用意されています。

このようなユーティリティおよびサービスを以下に示します。

- データベース関連のユーティリティ。これらのユーティリティは、データベ
ースのイメージ・コピーの作成、データベース変更レコードの累積およびソ
ート、リカバリーの高速化、およびデータベース・リカバリーに使用します。
- システム関連のユーティリティ。これらのユーティリティは、ログ・デー
タ・セットの管理およびリカバリー、または分析報告書の生成によってシステム
を管理する作業の一助として使用します。
- トランザクション・マネージャー関連のユーティリティ。これらのユーティ
リティは、トランザクションの管理と制御の一助として使用します。

また、IBM は、IMS 生産性向上ツールも多数提供しており、これにより IMS 環境
を強化できます。これらのツールは、ご使用の IMS ユーティリティのオペレ
ーションの自動化と高速化に役立ちます。これらはまた、ご使用の IMS データベ
ースの分析、管理、リカバリー、および修理を支援します。これらのツールについ
ては、Web ページ [http://www.ibm.com/software/data/db2imstools/products/
ims-tools.html](http://www.ibm.com/software/data/db2imstools/products/ims-tools.html) で詳しく学習できます。

データベース・リカバリー管理機能およびリカバリー

IMS データベース・リカバリー管理 (DBRC) 機能は、データベース・リカバリー用の IMS ユーティリティーの機能を拡張して、IMS データベースをより容易にリカバリーできるようにしています。DBRC は、DL/I データベースのリカバリーと高速機能の高速処理データベース (DEDB) のリカバリーの両方を手助けできます。

DBRC は 2 つのレベルの制御 (ログ制御と共用制御) を提供します。

ログ制御のみを使用する場合は、DBRC は IMS 用の OLDS の使用と再使用を制御します。

共用制御を使用する場合は、DBRC は次のことを行います。

- IMS 用 OLDS の使用と再使用の制御
- リカバリー管理 (RECON) データ・セット内のリカバリー関連情報の記録
- データベース・リカバリーの援助
- リカバリー関連ユーティリティー用のジョブ制御言語 (JCL) の生成
- データベースのスケジューリングの登録と制御

関連概念:

619 ページの『第 30 章 DBRC の概要』

自動化操作プログラム・インターフェース

IMS 自動化操作プログラム・インターフェース (AOI) を使用すると、アプリケーション・プログラムから IMS コマンドを発行できるようになり、MTO に経路指定された IMS メッセージを代行受信できます。

AOI を使用すれば、インストール・システムに合うように調整したプロシージャーを作成して、リカバリー処理の一部を自動化できます。この機能については、「IMS V15 オペレーションおよびオートメーション」の『自動化オペレーション用のツール』で説明されています。

拡張回復機能 (XRF) を使用する IMS 複合システムでは、AOI はアクティブになっている IMS サブシステム上で実行され、XRF テークオーバーが行われると、それ以降 AOI はその新しいアクティブ・サブシステム上で実行されるようになります。

IMS DBCTL 環境におけるリカバリー

IMS DBCTL 環境は、トランザクション・マネージャーではなく、IMS データベース・マネージャーで構成され、この環境では、1 つ以上のトランザクション管理サブシステム (CICS など) から DL/I データベースや DEDB にアクセスできます。

これらのトランザクション管理サブシステムは、コーディネーター・コントローラー (CCTL) として知られています。CCTL は、データベース・リソース・アダプター (DRA) と呼ばれるインターフェースを使用して DBCTL サブシステムと通信します。DBCTL 制御領域と CCTL との間の接続またはパスは、CCTL の初期設定の一環として作成されます。これらの接続はスレッドと呼ばれます。

CCTL は別個のサブシステムであるため、IMS DBCTL サブシステムと CCTL との間では互いに障害が分離されます。つまり、一般的には、DBCTL サブシステム

の障害が原因で CCTL が終了するようなことはないということです。同様に、CCTL の障害が原因で DBCTL サブシステムに障害が起こることは通常ありません。一般的には、CCTL 障害が DBCTL サブシステムに影響を及ぼす可能性があるのは、その障害が発生した時点において CCTL が DL/I で実行中のスレッドを 1 つまたは複数もっていたときだけです。

IMS DBCTL サブシステムは、DBCTL サブシステムによって作成されたログ・データ以外のログ・データからは再始動できません。同様に、DBCTL IMS サブシステム以外のサブシステムは、DBCTL サブシステムによって作成されたログ・データを使用して再始動することはできません。換言すると、IMS DBCTL サブシステムの再始動はそのシステムによって作成されたログ・データを使用して行う必要があるということです。ただし、他のサブシステム (IMS DB/DC、バッチ、または DBCTL) によって作成されたログ・データ・セットを組み合わせ使用してデータベース・リカバリーを行うことはできます。

DBCTL は DBRC ログ制御機能を必要とするため、DBRC と DBCTL を一緒に使用する必要があります。IMS DBCTL システムによって作成されたログ・レコードは、IMS DB/DC システムとの互換性があります (ただし、サブシステムの再始動の場合は除きます)。

XRF 複合システムでのリカバリー

拡張回復機能 (XRF) を代替 IMS サブシステムとして使用します。このサブシステムは、アクティブな IMS サブシステムのログ・データをモニターし、アクティブ・サブシステムに障害が発生した場合にはそのサブシステムの処理負荷をテークオーバーします。

XRF 以外のリカバリー機構について言えることはすべて XRF 複合システムにも当てはまります。しかし XRF を使用する場合は、IMS は、異常イベントの発生後にユーザーがデータを使用できない状態になる時間を削減するためにより多くのことを行えます。

自動テークオーバーのトリガーとなる条件として、以下のものを指定できます。

- IMS 制御領域の障害
- z/OS の全体的な障害
- 単一の中央演算処理装置複合システム (CPC) の障害
- IMS STATUS 出口ルーチンが必要となる内部リソース・ロック・マネージャー (IRLM) 障害
- IMS TPEND 出口ルーチンが必要となる仮想記憶通信アクセス方式 (VTAM) 障害

また、障害が起こっていない状態で XRF テークオーバーを手動で開始して、ユーザーに与える混乱を最小限に抑えながらシステムに対してプランされた変更を加えることもできます。

関連概念:

743 ページの『第 44 章 拡張回復機能の概要』

逆方向リカバリー

IMS またはアプリケーション・プログラムに障害が起こった場合は、誤った変更または加えたくない変更をデータベースから除去する必要があります。逆方向リカバリーすなわちバックアウト を実行すれば、これらの誤った更新を除去できます。

バックアウトには次の 3 つのタイプがあります。

- 動的バックアウト
- 緊急時再始動時のバックアウト
- バッチ・バックアウト

最初の 2 つのタイプのバックアウトは IMS が自動的に実行します。最後の 1 つはユーザーが手動で開始してください。

以下のいずれかが起こった場合は、常に IMS はオンライン環境でのデータベース変更を自動的に (動的に) バックアウトします。

- アプリケーション・プログラムが異常終了した。
- アプリケーション・プログラムがロールバック呼び出しを出した。
- アプリケーション・プログラムが使用不能なデータベースへのアクセスを試行した。
- デッドロックが発生した。

バッチ環境では、バッチ・ジョブが異常終了したかまたはロールバック呼び出しを出した場合は IMS がデータベース変更を動的にバックアウトするように指定できます。

システム障害後の再始動処理の間に、IMS は障害発生時に実行中のアプリケーション・プログラムがあったかどうかと、それらがバックアウトが必要な変更を行ったかどうかを判別します。再始動する前に、IMS はそのような変更に関するログ・レコードをスキャンし、次に影響を受けたデータベースから当該の変更をバックアウトします。ログのスキャン中に IMS がメモリーを使い尽くしてしまった場合は、変更のバックアウトを手動で行うよう指示するメッセージを受け取ることがあります。

IMS バッチ・バックアウト・ユーティリティー (DFSBB00) を使用すると、DL/I 領域、DBB 領域、またはオンライン・プログラムの実行中に行われたデータベース変更を除去できます。このユーティリティーを使用すると、最後のチェックポイント以降の変更をバックアウトできます。当該のバッチ領域でデータ共用が使用されておらず、その領域が BMP でない場合は、特定のチェックポイントを選択できます。BMP の場合はチェックポイントを指定しないでください。なぜなら、このユーティリティーはログ上の最後のチェックポイントまで必ず BMP をバックアウトするからです。

動的バックアウトまたは緊急時再始動時のバックアウトが失敗した場合は、IMS はバックアウトが完了していないデータベースを停止し、ユーザーがそれらのデータベースを再始動するときにバックアウトを再実行します。

順方向リカバリー

順方向リカバリーでは、IMS ログから情報を再構成し、その情報をデータベースに再適用します。

順方向リカバリーでは、破損した情報または作業を再構成し、それをデータベースに追加します。バックアウトでは、誤った変更または行いたくない変更を情報から除去します。IMS サブシステムの場合は、順方向リカバリーの作業はユーザーが行います (IMS はそれを手助けするユーティリティを提供します)。トランザクションのバックアウト (およびそれらのトランザクションに関連するデータベース更新) は IMS によって自動的に処理されます。バックアウトは通常はユーザーの関与を必要としませんが、アプリケーション・プログラムにそのアプリケーションが担当したトランザクションのバックアウト処理を制御させるようにしても構いません。バッチ・プログラムのバックアウトは、自動的に実行させることも IMS ユーティリティを使用して行うこともできます。

IMS システムのシャットダウン

問題の中には IMS サブシステムの障害の原因となるほど重大なものがあります。このようなケースでは、IMS は自動的にシャットダウンします。しかし、その他のケースでは IMS は稼働し続けるので、手動で IMS をシャットダウンしてからでなければリカバリーを行えません。その際には、IMS サブシステムの一部だけのシャットダウン (部分的シャットダウン) と全体のシャットダウン (完全シャットダウン) のどちらも実行可能です。

推奨事項: 基本的な目的は、ユーザーが IMS 機能を使用できる状態を保持することであるため、部分的シャットダウンではシステムをリカバリーできない場合にのみ、完全シャットダウンを使用することをお勧めします。

IMS の一部だけが誤動作している場合は、その部分だけをシャットダウンし、IMS の残りの部分は機能するままにして、生産性を落とさずに済む可能性があります。例えば、IMS と他のデータベースには処理を継続させておいて、障害のあるデータベースをオフラインにする (アプリケーション・プログラムから使用不能な状態にする) ことができます。また、誤動作している端末を切り離すことが可能で、その一方で他のすべての機能は影響を受けずに作動し続けます。

IMS の一部だけをシャットダウンするには、誤動作中のコンポーネントを停止させるコマンドを使用してください。

IMS を制御付きでシャットダウンすることもできます。この制御付きシャットダウンが推奨されます。なぜなら、このシャットダウンではシステム内の現行情報が保管されるため、後でシステムを簡単かつ正確に再始動できるからです。

DRD が使用可能で、自動エクスポートが使用不可の場合、動的に定義されたリソースはすべて、IMS をシャットダウンする前に手動でエクスポートする必要があります。

IMS をシャットダウンするには、キーワード FREEZE、DUMPQ、または PURGE のいずれかを指定して /CHECKPOINT コマンドを使用してください。

関連資料:

 /CHECKPOINT コマンド (コマンド)

再始動時のリカバリー

IMS は、再始動時に多くのタイプの障害からリカバリーできるようになっています。いくつかのケースでは、リカバリーは自動的に行われます。例えば、トランザクションとそれらに関連したデータベース更新のバックアウトなどがこれに当たります。その他のケースでは、ユーザーが再始動コマンドでリカバリーのタイプを要求します。例えば、メッセージ・キューを直前のバックアップ・コピーから再作成するよう IMS に指示できます。

一般的に、IMS を再始動する方法は、IMS をシャットダウンした方法、または IMS にどのような障害が起こったかによって決まります。

- IMS の一部だけをシャットダウンした (例：あるデータベースをオフラインにしたか、あるいはある回線を切り離れた) 場合は、その部分だけを再始動する必要があります。
- IMS 全体をシャットダウンしたか、IMS 自体に障害が起こった場合は、その IMS サブシステム全体を再始動する必要があります。

障害が起こったコンポーネントをリカバリーしたら、それを再始動しても構いません。例えば、入出力エラーのためにデータベースをオフラインにし、その後それをリカバリーしたら、そのデータベースを再びアプリケーションが使用できるようになります。

IMS の一部だけを再始動するには、停止させたコンポーネントを開始するコマンドを使用してください。

IMS サブシステム全体を再始動するためには、その前にまず必要なリカバリーをすべて実行する必要があります。その後、以下の 3 つの方法のいずれかで IMS を再始動できます。

- 通常再始動。障害が起こっていない IMS サブシステムを初期設定します。通常再始動では、コールド・スタート (IMS の以前の実行を一切参照せずに IMS を再始動する場合) とウォーム・スタート (/CHECKPOINT コマンドを使用して終了された IMS サブシステムを再始動する場合) のどちらも可能です。通常再始動のコマンドは /NRESTART です。
- 緊急時再始動。障害が起こった IMS サブシステムを初期設定します。緊急時再始動時には、IMS はトランザクションとアクティブ領域をリセットし、データベースとメッセージ・キューを最新の同期点での状態に復元します。バッチおよびバッチ・メッセージ処理 (BMP) 領域は手動で再始動する必要があります。緊急時再始動コマンドは /ERESTART です。
- 自動再始動。この場合は、適切な再始動コマンドを IMS が自動的に選択するため、MTO の介入が少なく済み、再始動の高速化が可能です。オペレーターは再始動コマンドを入力しません。その代わりに IMS 制御領域用の JCL に AUTO=Y をコーディングして、自動再始動を指定します。

リカバリーの複雑性

開始場所を突き止めて、それ以降に行ったことをトラッキングしただけで常にシステムをリカバリーできるのであれば、話は簡単です。残念ながら、多くの場合、データやアプリケーションにだけでなく、リカバリー・メカニズムそのものにも複雑な事態が持ち上がります。

IMS ログ (DASD 上またはテープ上にある) は、他のあらゆるデータ・セットとまったく同じように、入出力問題の被害を受けます。スペースが使い尽くされまったり、ドライブに物理的な問題があることが発覚したりすることがあります。リカバリーに関与する内部 IMS モジュールは、オペレーティング・システムとハードウェアに全面的に依存しています。そのため、なんらかの理由 (例：電源障害、異常終了、ハードウェアの誤動作など) によってシステムが破損した場合は、これらのモジュールによって実行されるリカバリー処理が不完全となることがあります。

したがって、問題からのリカバリーは 2 つの部分からなる処理となります。最初に IMS リカバリー・ツール自体をリカバリーして、次にデータとアプリケーションをリカバリーすることができます。

IMS リカバリーが不可能な場合

IMS は、数種類のエラーからは自動的にリカバリーできます。さらに IMS には、他の多くのエラーからもリカバリーできるようにするためのメカニズムが用意されています。しかし、IMS にはリカバリーできないエラーがいくつかあります。

一般的に、これらのリカバリー不能エラーは次の 2 つのタイプに分かれます。

- アプリケーション論理エラーまたは入力エラー

実行中のアプリケーション・プログラムによって引き起こされたエラーを検出した場合は、誤ったデータのバックアウトはきわめて簡単です。しかし、そのプログラムへの入力が誤っていたり、論理エラーがある場合でも、プログラムは完了するまで実行されてしまうため、このような問題を検出できない可能性があります。そのアプリケーション・プログラムの実行が完了した (あるいはそのプログラムがデータをコミットした) 後に、その誤ったデータが他のプログラムによって使用される場合があり、そのためにこの問題がシステム全体に波及してしまう可能性があります。

IMS では、コミット済みのデータを自動的にまたは結果を保証するバックアウトの方法がありません。例えば影響を受けたデータベース内の個々のレコードを編集するなどしてユーザーがデータを手動でバックアウトすることは可能ですが、このような処理は往々にして困難です。さらに、オリジナル・エラーが引き起こされた時点から時間がたてばたつほど、誤ったデータをすでに使用してしまったプログラムの数が増え、成功する可能性はますます少なくなります。そのうえ、この誤ったデータの結果として生成された出力または実行されたアクションは、すでにデータベースの境界の外側に出て行ってしまっています。

推奨事項: アプリケーション・ロジックまたは入力のエラーを最小限にするには、新しいアプリケーションをオンラインにする前に、それらに対して広範なテスト (個別テストと統合システム・テストの両方) を行う必要があります。ま

た、アプリケーション・プログラムにユーザー入力に対する広範な妥当性検査を行わせるようにすると、やはり入力エラーを最小限にするのに役立ちます。

- IMS リカバリー機能の誤用 (偶発的または故意) を含む操作エラー

リカバリーが正常に行われるようにするためには、日常の IMS オペレーションを適切に行うことと、リカバリー機能の適正使用の両方が必要です。この適正使用には、次のようなことが含まれます。

- 正しいデータ・セットの使用
- 通常オペレーション時におけるジョブの不適切な取り消しの回避
- ログの定期的な保守
- 正しいログ・ボリュームの正しいシーケンスでの使用
- 適切なユーティリティーの適宜使用

IMS を適切に操作するのに困難があってはならないので、本書の他のトピックでは必要な手順をセットアップする方法およびガイドラインについて説明します。ただし、IMS または IMS のリカバリー機能 (あるいはその両方) の操作が不適切であった場合は、IMS はそのような不適切な使用によって引き起こされた問題を訂正できないことがあります。

リカバリーのオーバーヘッド

リカバリーの価値は明白ですが、コストなしではできません。例えば、ロギングは時間とスペースを要します。ログ・レコードを作成する際にはいくらかの処理時間を必要としますし、各ログ・レコードはデータ・セット上でそれぞれスペースを使用します。また、チェックポイントが書き込まれるたびに、バックアップ・コピーの作成の場合と同様に、やはり時間とスペースが消費されます。

実際問題として、基本的なリカバリー体系を備えることは必須であり、これについては選択の余地はありません。ただし、どの程度入念なリカバリー体系にするのかは選択できます。

一般的に、日常のオペレーションに組み込まれているリカバリーのための準備が広範であればあるほど、エラーの発生後にシステムをより速やかに実動状態に戻せるようになります。トレードオフの関係にあるのは、リカバリーの速さと、リカバリー用の維持のための日常のオーバーヘッドです。また、オンライン・システムをダウンさせた場合のコストについても考慮する必要があります。

このトレードオフについての判断を下す際には、リカバリーを必要とするエラーの発生頻度を考慮に入れてください。エラーは確実に起こるものですが、そのためのリカバリーの準備をするのに値するのは、まれにしか起こらないエラーよりも、むしろ頻繁に発生するエラーの方です。リカバリー・メカニズムのための準備をどの程度までするのかについての決定は、インストールにおけるニーズと優先順位をしっかりと把握し、この両者のそれぞれの重要性をはかりに掛けたうえで行う必要があります。

IMS Application Menu

「IMS Application Menu」は、ISPF を使用して TSO 上で稼動する IBM 提供の IMS アプリケーションへの共通インターフェースを提供します。

以下のアプリケーションは、ISPF を使用して TSO 上で稼働します。

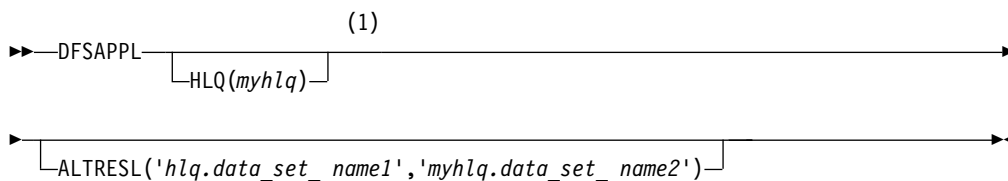
- Single Point of Control (SPOC)
- Manage Resources
- HALDB 区画定義ユーティリティー
- IMS 構文チェッカー
- インストール検査プログラム (IVP)
- IVP Export utility
- IPCS with IMS Dump Formatter (IPCS)
- Abend Search and Notification (ASN)

ヒント: 「IMS Application Menu」にアクセスできるようにするには、IMS.SDFSEXEC データ・セットを SYSPROC DD 連結に組み込みます。

重要: IPCS が必ず「IMS Application Menu」の開始より前に開始されるようにしてください。そうしないと、メッセージ DFSIX103 が表示されます。

DFSAPPL コマンドを使用して「IMS Application Menu」を開始します。次のように、TSO コマンドと EXEC コマンドがあり、どちらでも使用できます。

- TSO %DFSAPPL HLQ(*myhlq*)
- EXEC 'IMS.SDFSEXEC(DFSAPPL)' 'HLQ(*myhlq*)'



注:

- 1 このコマンドを最初に使用するときは、**HLQ** パラメーターが必要です。その後は、**HLQ** はオプション・パラメーターとなります。

ただし、次のとおりです。

DFSAPPL

「IMS Application Menu」を開始するためのコマンド

HLQ IMS 配布データ・セットの高位修飾子を指定するためのキーワード

このコマンドを最初に使用するときは、**HLQ** パラメーターが必要です。このパラメーターを指定しなかった場合、コマンドは最新に指定された高位修飾子を使用します。このパラメーターはオプションです。

myhlq

IMS 配布データ・セットの高位修飾子

ALTRESL

ロード・モジュールを含んでいるデータ・セット名のリストを指定するためのキーワード

ALTRESL パラメーターを指定する場合は、データ・セット名のリストに SDFSRESL を含める必要があります。ALTRESL パラメーターを指定しなかった場合は、*myhlq.SDFSRESL* が ISPLLIB データ・セットとして使用されます。

myhlq.data_set_name1

ロード・モジュールを含んでいるデータ・セットの完全修飾名

注: アプリケーションの中には、ISPTABL データ・セットを必要とするものがあります。ISPTABL データ・セットが割り振り済みであると、そのデータ・セットが引き続き使用されます。ISPTABL データ・セットが使用されていない場合は、TSO 接頭部またはユーザー ID を高位修飾子として使用して、新しいデータ・セットが割り振られます。

以下の図に、「IMS Application Menu」を示します。

```
Help
-----
IMS Application Menu
Command ==> 2_____

Select an application and press Enter.

      1 Single Point of Control (SPOC)
      2 Manage resources
      3 Reserved for future use
      4 HALDB Partition Definition Utility (PDU)
      5 Syntax checker for IMS Parameters (SC)
      6 Installation Verification Program (IVP)
      7 IVP Export Utility (IVPEX)
      8 IPCS with IMS Dump Formatter (IPCS)
      9 Abend Search and Notification (ASN)
```

図 26. IMS Application Menu

「IMS Application Menu」を使用する場合は、アプリケーションを選択して Enter (実行) キーを押せば、任意の TSO または ISPF アプリケーションを開始できます。





また、ローカル ISPF オプション・メニューから「IMS Application Menu」にリンクすることもできます。以下のパネルはその一例です。

```
)BODY
      Local Option Menu
Option ==>_ZCMD


      I   IMS Application Menu
      .
      .
)PROC
&ZSEL = TRANS(TRUNC(&ZCMD, '.'))
      I, 'CMD(%DFSAPPL HLQ(myhlq)) NEWAPPL(DFS) NOCHECK'
      .
      .
)END
```


関連資料: IMS アプリケーション・メニュー内のいずれかの ISPF パネルについて詳しくは、アプリケーションに付属する ISPF オンライン・ヘルプを参照してください。


関連概念:

-  TSO SPOC アプリケーションによる IMS の制御 (オペレーションおよびオートメーション)
-  IMS 構文チェッカー (システム定義)
-  IMS インストール検査プログラム (IVP) の概要 (インストール)
-  IPCS および IMS オフライン・ダンプ・フォーマッターの使用 (診断)

関連資料:

-  HALDB 区画定義ユーティリティ (%DFSHALDB) (データベース・ユーティリティ)

関連情報:

-  IMS 異常終了の検索および通知 (メッセージおよびコード)

第 2 部 IMS システム管理の考慮事項とタスク

以下のトピックでは、IMS システムの日常のオペレーションのための情報と必要な手順について説明します。

第 5 章 z/OS インターフェースの考慮事項

以下のトピックでは、IMS および IRLM を z/OS にインストールする際に考慮しなければならない情報および必要なステップについて説明します。

重要: z/OS および VTAM インターフェースのステップが完了した後、ユーザーは z/OS を IPL し、CLPA または MLPA=xx のいずれか一方 (あるいはその両方) を指定する必要があります。また、IMS は 31 ビット処理モードでも 64 ビット処理モードでも実行できることに注意してください。

IMS のための z/OS インターフェースの考慮事項

z/OS 上に IMS を確実に正しく完全にインストールするためには、考慮すべき多数の要件と必要なステップがあります。以下のトピックで、これらの要件と必要なアクションについて説明します。

インストールの問題の予防

以下の処置を確実に実行して、z/OS への IMS のインストール中に問題が起こらないようにしてください。

- IMS ステージ 2 定義では、z/OS マクロ・ライブラリーを使用してください。IMS は、z/OS の下でのみ稼動します。
- IMS がロードされ実行される元となるライブラリーを、適切な許可表に組み込み、制御領域が APF 許可されたプログラムとして実行されるようにします。z/OS では、IMS は許可プログラムとして実行されます。

関連資料: APF 許可については、「z/OS MVS 初期設定およびチューニング 解説書」の IEAAPFxx (許可プログラム機能リスト) を参照してください。

- IMS.SDFSRESL は LNKLSTxx (SYS1.LINKLIB に連結されたデータ・セット) に置かず、JOBLIB または STEPLIB DD ステートメントを使用してください。もし IMS.SDFSRESL が LNKLSTxx にあると、異なる IMS リリース・レベル (そのリリース自身の IMS.SDFSRESL が正しく APF 許可されていない) はモジュールを LNKLSTxx からロードすることができます。互換性のないリリース・レベルのモジュールは、予測できない結果を引き起こす原因となる可能性があります。
- プログラム特性テーブルを更新します。IMS 制御領域は、ジョブ・ステップ・タスクまたはシステム・タスクとして作動します。制御領域の実行はすべて、監視プログラム状態です。詳しくは、149 ページの『IBM 提供の z/OS プログラム特性テーブルの更新』を参照してください。

関連資料: z/OS 上で実行するときのシステム保全性の維持に関する追加情報については、「z/OS MVS Programming: Authorized Assembler Services Reference」を参照してください。

JCL のセットアップ

以下のリストには、z/OS JCL をセットアップするための要件が含まれています。

- JOB ライブラリーまたは STEP ライブラリーは、制御領域に対して、APF 許可されている必要があります。従属領域に対して、PGMLIB は許可されている必要はなく、STEPLIB として SDFSRESL に連結することができます。
- EXEC ステートメントは、制御領域に対して PGM=DFSVMVRC0 を指定する必要があります。
- 以下のライブラリーは APF 許可されている必要があります。
 - IMS.SDFSRESL
 - IMS.MODBLKSA および IMS.MODBLKSB
 - IMS.SDXRRESL
 - IMS.SDFSJLIB
 - ユーザーの DB2 モジュールのロード先となるライブラリー (DFSESL DD または JOBLIB または STEPLIB)

関連資料: z/OS JCL についての詳細は、「IMS V15 システム定義」に記載されているシステム定義プロセスに関する情報を参照してください。

従属領域の JES の構成

開始タスクとして実行されている IMS 従属領域が終了したときは、EOM ブロードキャストが常に行われますが、バッチ・ジョブが終了したときは、EOM ブロードキャストが行われない場合があります、そのために処理されない場合があります。

IMS 従属領域をバッチ・ジョブとして実行する場合、ジョブとそのジョブで使用されたイニシエーターの両方が終了するまで、EOM ブロードキャストは行われません。したがって、従属領域がバッチ・ジョブとして実行されており、説明されたようなストレージ問題が発生した場合、EOM クリーンアップ・サービスは、イニシエーターが終了するまで呼び出されません。

JES2 環境では、IBM 定義の出口を実装することにより、従属領域で使用されるイニシエーターをジョブの終了時に自動的に停止するように設定できます。

サブシステム・インターフェース (SSI) ジョブ選択用の JES2 出口 32 を使用して、ジョブの終了時にイニシエーターを終了 (排出) するか、終了して再起動するように指定します。

1. ジョブの終了時にイニシエーターを終了して再起動するには、応答バイトのビット 4 を設定します。
2. ジョブの終了時にイニシエーターを終了するには、ビット 5 を設定します (詳しくは、「z/OS V1R12.0 JES2 Installation Exits」 (SA22-7534-12) を参照してください)。

JES3 環境では、イニシエーターの GROUP 定義で EXRESC パラメーターの UNALOPT オプションに DYNAMIC または DEMAND を指定することにより、従属領域で使用されているイニシエーターをジョブの終了時に自動的に停止するように設定できます (詳しくは、「z/OS JES3 Initialization and Tuning Guide」 (SA22-7549-11) を参照してください)。

一部の必須の非標準 z/OS マクロをそれらのオリジナル・ライブラリーで保持

特定の IMS モジュールのアセンブリーでは、標準の z/OS システム・マクロ・ライブラリーには含まれていない z/OS マクロが必要とされます。これらの要件は IMS および z/OS の保守により変わることがあるため、これらのマクロは本来のライブラリーに保持しておき、SYS1.MODGEN (または SYS1.AMODGEN) に対して IMS が生成した JCL を使用します。


IBM 提供の z/OS プログラム特性テーブルの更新

以下のモジュールは、z/OS: BPEINI00、CQSINIT0、DFSMVRC0、および DXRRLM00 と一緒に出荷されるデフォルトのプログラム特性テーブル (PPT) に事前定義されています。

デフォルトの z/OS PPT を変更しなかった場合、これらの IMS モジュールはすでに PPT に組み込まれているので、何もアクションを実行する必要はありません。これらのモジュールのデフォルト項目を削除した場合は、デフォルト項目を復元する必要があります。これについては、後続のセクションで説明します。

関連資料: PPT の更新については、「z/OS MVS 初期設定およびチューニング 解説書」を参照してください。

関連タスク:

 z/OS プログラム特性テーブルへの項目の追加 (システム定義)

z/OS プログラム特性テーブルへの IMS 項目の追加

IMS オンライン環境 (DB/DC、DBCTL、DCCTL) では、この PPT 項目が必要です。IMS BATCH のみを使用する場合は、この項目は必要ありません。

必要な入力の例を以下に示します。この例は IMS.INSTALIB データ・セットに含まれています。「SCHEDxx の更新 - PPT 項目」という表題の正しい項目については、「IMS V15 インストール」の『IVP ジョブおよびタスク』を参照してください。

この項目を作るには、SYS1.PARMLIB データ・セットの SCHEDxx メンバーを編集します。以下の項目を SCHEDxx メンバーに追加します。

	/* IMS ONLINE CONTROL REGION	*/
PPT PGMNAME(DFSMVRC0)	/* PROGRAM NAME = DFSMVRC0	*/
CANCEL	/* PROGRAM CAN BE CANCELLED	*/
KEY(7)	/* PROTECT KEY ASSIGNED IS 7	*/
NOSWAP	/* PROGRAM IS NOT-SWAPPABLE	*/
NOPRIV	/* PROGRAM IS NOT PRIVILEGED	*/
SYST	/* PROGRAM IS A SYSTEM TASK	*/
DSI	/* DOES REQUIRE DATA SET INTEGRITY	*/
PASS	/* PASSWORD PROTECTION ACTIVE	*/
AFF(NONE)	/* NO CPU AFFINITY	*/
NOPREF	/* NO PREFERRED STORAGE FRAMES	*/

DFSMVRC0 プログラムの PPT 項目には、上記のように NOSWAP の指定が必要です。

z/OS プログラム特性テーブルへの CQS 項目の追加

CQS を使用する場合は、CQSINIT0 または BPEINI00 のいずれかの z/OS PPT 項目が必要です。

CQSINIT0 項目のサンプルを以下に示します。このサンプルは IMS.INSTALIB データ・セットに含まれています。BPEINI00 項目のサンプルは、『z/OS プログラム特性テーブルへの BPE 項目の追加』という見出しの下にあります。「SCHEDxx の更新 - PPT 項目」という表題の正しい項目については、「IMS V15 インストール」の『IVP ジョブおよびタスク』を参照してください。

この項目を作るには、SYS1.PARMLIB データ・セットの SCHEDxx メンバーを編集します。以下の項目を SCHEDxx メンバーに追加します。

```
PPT PGMNAME(CQSINIT0)      /* CQS - COMMON QUEUE SERVER      */
                                /* PROGRAM NAME = CQSINIT0        */
                                /* PROGRAM CAN BE CANCELLED      */
                                /* PROTECT KEY ASSIGNED IS 7     */
                                /* PROGRAM IS NOT-SWAPPABLE      */
                                /* PROGRAM IS NOT PRIVILEGED     */
                                /* PROGRAM IS A SYSTEM TASK      */
                                /* DOES REQUIRE DATA SET INTEGRITY */
                                /* PASSWORD PROTECTION ACTIVE    */
                                /* NO CPU AFFINITY               */
                                /* NO PREFERRED STORAGE FRAMES  */
```

CQSINIT0 プログラムの PPT 項目には、上記のように NOSWAP の指定が必要です。

z/OS プログラム特性テーブルへの BPE 項目の追加

BPE プログラム特性テーブルの項目 BPEINI00 は、以下のようなさまざまな IMS アドレス・スペースで使用されます。

次のようなアドレス・スペースがあります。

- Common Queue Server (CQS)
- データベース・リカバリー管理 (DBRC) - オプション
- IMS Connect
- Open Database Manager (ODBM)
- Operations Manager (OM)
- Resource Manager (RM)
- 構造化呼び出しインターフェース (SCI) (Structured Call Interface (SCI))

これらのアドレス・スペースのうちのいずれかを使用する場合、PPT に BPEINI00 項目を入れておかなければなりません。

この項目を作るには、SYS1.PARMLIB データ・セットの SCHEDxx メンバーを編集します。以下の項目を SCHEDxx メンバーに追加します。

```
PPT PGMNAME(BPEINI00)      /* BPE - BASE PRIMITIVE ENVIRONMENT */
                                /* PROGRAM NAME = BPEINI00        */
                                /* PROGRAM CAN BE CANCELLED      */
                                /* PROTECT KEY ASSIGNED IS 7     */
                                /* PROGRAM IS NOT-SWAPPABLE      */
                                /* PROGRAM IS NOT PRIVILEGED     */
                                /* REQUIRES DATA SET INTEGRITY  */
                                /* CANNOT BYPASS PASSWORD PROTECTION */
```



```

SYST          /* PROGRAM IS A SYSTEM TASK          */
AFF(NONE)     /* NO CPU AFFINITY                          */
NOPREF        /* NO PREFERRED STORAGE FRAMES             */

```

SCHEDxx の変更を有効にするために、以下の処置のいずれかを行います。

- z/OS システムを再始動する。
- z/OS SET SCH= コマンドを発行する。

z/OS プログラム特性テーブルの IMS Connect 項目の追加

IMS Connect は、事前定義されている z/OS プログラム特性テーブル (PPT) 項目 BPEINI00 を使用しますが、HWSHWS00 もサポートされています。

以下に示す例では、BPEINI00 の代わりに HWSHWS00 を使用しています。PPT に両方の項目を含めることもできます。例えば、IMS Connect を含む混在バージョンの IMS 環境の場合などです。HWSHWS00 はデフォルトの PPT には組み込まれていないため、使用する場合は PPT に HWSHWS00 を追加する必要があります。

推奨事項: IMS Connect 始動 JCL で BPEINI00 を使用すれば、PPT に HWSHWS00 を追加する必要はありません。

例

TCP/IP 通信の場合のみ、z/OS PPT に以下の項目を追加してください。

```

PPT PGMNAME(HWSHWS00) /* PROGRAM NAME = HWSHWS00          */
      CANCEL          /* PROGRAM CAN BE CANCELED           */
      KEY(7)          /* PROTECT KEY ASSIGNED IS 7        */
      SWAP            /* PROGRAM IS SWAPPABLE              */
      NOPRIV          /* PROGRAM IS NOT PRIVILEGED         */
      DSI             /* REQUIRES DATA SET INTEGRITY     */
      PASS            /* CANNOT BYPASS PASSWORD PROTECTION*/
      SYST            /* PROGRAM IS A SYSTEM TASK          */
      AFF(NONE)       /* NO CPU AFFINITY                   */
      NOPREF          /* NO PREFERRED STORAGE FRAMES      */

```

クライアント通信のローカル・オプションを単独で使用する場合でも、TCP/IP 通信で使用する場合でも、z/OS PPT に以下の項目を追加してください。

```

PPT PGMNAME(HWSHWS00) /* PROGRAM NAME = HWSHWS00          */
      CANCEL          /* PROGRAM CAN BE CANCELED           */
      KEY(7)          /* PROTECT KEY ASSIGNED IS 7        */
      NOSWAP          /* PROGRAM IS NOT SWAPPABLE         */
      NOPRIV          /* PROGRAM IS NOT PRIVILEGED         */
      DSI             /* REQUIRES DATA SET INTEGRITY     */
      PASS            /* CANNOT BYPASS PASSWORD PROTECTION*/
      SYST            /* PROGRAM IS A SYSTEM TASK          */
      AFF(NONE)       /* NO CPU AFFINITY                   */
      NOPREF          /* NO PREFERRED STORAGE FRAMES      */

```

z/OS プログラム特性テーブルへの IRLM 項目の追加

IRLM を使用する場合は、z/OS PPT 項目が必要です。

必要な入力の例を以下に示します。この例は IMS.INSTALIB データ・セットに含まれています。「SCHEDxx の更新 - PPT 項目」という表題の正しい項目については、「IMS V15 インストール」の『IVP ジョブおよびタスク』を参照してください。

この項目を作るには、SYS1.PARMLIB データ・セットの SCHEDxx メンバーを編集します。以下の項目を SCHEDxx メンバーに追加します。

```

PPT PGMNAME(DXRRLM00) /* IRLM - RESOURCE LOCK MANAGER */
CANCEL /* PROGRAM NAME = DXRRLM00 */
KEY(7) /* PROGRAM CAN BE CANCELLED */
NOSWAP /* PROTECT KEY ASSIGNED IS 7 */
NOPRIV /* PROGRAM IS NOT-SWAPPABLE */
SYST /* PROGRAM IS NOT PRIVILEGED */
DSI /* PROGRAM IS A SYSTEM TASK */
PASS /* DOES REQUIRE DATA SET INTEGRITY */
AFF(NONE) /* PASSWORD PROTECTION ACTIVE */
NOPREF /* NO CPU AFFINITY */
NOPREF /* NO PREFERRED STORAGE FRAMES */

```

DXRRLM00 PPT 項目には、上記のように NOSWAP の指定が必要です。

必要な IMS リンクの z/OS へのインストール

これらのモジュールとプロシーチャーを、使用する z/OS システムにインストールする必要があります。

IMS システム定義のステージ 2 では、以下の変更を行う場合があります。

- IMS.SDFSRESL への CTC チャネル終了付加ルーチンのバインド (CTC オプション付きの MSC が定義されている場合)
- IMS.PROCLIB へのカタログ式プロシーチャーのコピー

さらに、JCL を実行して、以下のモジュールを IMS.SDFSRESL にバインドする必要があります。

- タイプ 2 SVC ルーチン。DFSIDEF マクロ内で TYPE=GEN,SVC2=xxx を指定すると (ここで、xxx はタイプ 2 SVC 番号)、モジュールをバインドするために使用できるサンプル JCL が生成されます。
- DBRC タイプ 4 SVC ルーチン。DFSIDEF マクロ内で TYPE=GEN,SVC4=yyy を指定すると (ここで、yyy はタイプ 4 SVC 番号)、モジュールをバインドするために使用できるサンプル JCL が生成されます。

以下の表に、IMS システムを z/OS の下で実行する場合に必要な処置の概要を示します。

表 5. z/OS 下での実行の場合に必要なステップ (IMS 環境別)

処置	DB チ・シス テム	DBCTL システム	DB/DC システム	DCCTL システム
タイプ 2 SVC を z/OS 中核にバインドする	あり	あり	あり	あり
以下の方式のいずれかを使用して、SYS1.NUCLEUS からタイプ 2 SVC をロードする	あり	あり	あり	あり
<ul style="list-style-type: none"> 中核モジュール・ローダー機能 SYS1.IPLPARM メンバー NUCLSTxx SYS1.PARMLIB メンバー NUCLSTxx 				

表 5. z/OS 下での実行の場合に必要なステップ (IMS 環境別) (続き)

処置	DB チ・シス テム	DBCTL システム	DB/DC システム	DCCTL システム
DBRC タイプ 4 SVC モジュールを LPALIB (または、オプションで MLPA ライブラリー) にバインドする	あり	あり	あり	あり

以下の表に、z/OS インターフェースに必要なモジュールを示します。この表は、モジュールがバインドされた後の、モジュール名を配布ライブラリー (IMS.ADFSLOAD) に、ロード・モジュール名をターゲット・ライブラリー (IMS.SDFSRESL) に示します。

表 6. z/OS インターフェース・モジュール:

IMS.ADFSLOAD	IMS.SDFSRESL	説明
DFSVC200 ²	IGCiii ²	タイプ 2 SVC ベクトル・ルーチン ³
DSP00MVS ¹	IGC00nnn ¹	DBRC タイプ 4 SVC ルーチン ³

注:

iii タイプ 2 SVC 番号を指定します。

nnn 符号付き 10 進数のタイプ 4 SVC 番号。例えば、SVC 255 は 25E です。

¹ これらのモジュールは、RENT および REFR 属性でバインドされる必要があります。

² このモジュールは、RENT、REFR、および SCTR バインダー・オプションでバインドされる必要があります。これらのモジュールは、SYS1.NUCLEUS に置かれます。

³ JCL を実行して、これらのモジュールをバインドする必要があります。

IMS SVC モジュール

IMS は、バッチ、ユーティリティー、DBCTL、DCCTL、および DB/DC IMS 制御プログラム機能の場合、200 ~ 255 の範囲のタイプ 2 監視プログラム呼び出し (SVC) を使用します。IMS は、DBRC 機能の場合、200 から 255 の範囲のタイプ 4 監視プログラム呼び出し (SVC) を使用します。

同一の z/OS システムに異なるリリース・レベルの IMS をインストールしようとしている場合は、タイプ 2 SVC および タイプ 4 SVC には下方への互換性があることに注意してください。SVC の IMS 15 レベルは、IMS バージョン 13 および IMS 14 で使用できます。しかし、IMS バージョン 13 レベルは、IMS 14 でも IMS 15 でも使用できず、IMS 14 レベルは IMS 15 では使用できません。

重要: DFSIDEF0 モジュールで **SVC2=** または **SVC4=** パラメーターを指定する場合は、DFSIDEF0 モジュールが、IMS 制御領域の JOBLIB 連結か STEPLIB 連結の中、または z/OS LINKLIST 連結の中に組み込まれた APF 許可ライブラリー内にあることを確認してください。

タイプ 2 SVC

タイプ 2 SVC を使用するには、以下のステップを実行します。

1. 以下のいずれかの場所に SVC2= キーワードを指定して、タイプ 2 SVC 番号を定義します。タイプ 2 SVC 番号を定義するために SVC2= キーワードを使用しなかった場合、IMS はデフォルト値の 254 を使用します。
 - IMS PROCLIB データ・セットの DFSPBxxx メンバー内。DFSPBxxx メンバーで指定した値は、DFSIDEF マクロで指定された値をオーバーライドします。
 - JCL 内。JCL 内で指定した値は、DFSPBxxx メンバー内で指定された SVC2= キーワード値および DFSIDEF マクロで指定された値をオーバーライドします。
 - 次のコードを指定して、DFSIDEF0 モジュールの DFSIDEF マクロで指定します。


```
DFSIDEF TYPE=PARM,SVC2=
```

DFSIDEF マクロで指定した値は、DFSPBxxx メンバー内または JCL 内で指定された値によってオーバーライドされます。
2. DFSIDEF マクロで SVC2= パラメーターを使用してタイプ 2 SVC 番号を定義した場合は、DFSIDEF0 モジュールをアセンブルして IMS.SDFSRESL にリンクします。
3. JCL を実行してタイプ 2 SVC ルーチン DFSVC200 を IGCxxx として再リンクします。ここで、xxx はタイプ 2 SVC 番号です。ルーチンを再リンクするためのサンプル JCL を生成するには、DFSIDEF マクロ内で TYPE=GEN,SVC2=xxx を指定します。
4. タイプ 2 SVC ルーチンを z/OS 中核にバインドします。

タイプ 4 SVC

タイプ 4 SVC を使用するには、以下のステップを実行します。

1. タイプ 4 SVC 番号を定義するために、DFSIDEF0 モジュールの DFSIDEF マクロで次のコードを指定します。SVC4= キーワードを使用してタイプ 4 SVC 番号を指定しなかった場合、IMS はデフォルト値の 255 を使用します。


```
DFSIDEF TYPE=PARM,SVC4=
```
2. DFSIDEF0 モジュールをアセンブルして、IMS.SDFSRESL にリンクします。
3. JCL を実行して、タイプ 4 SVC ルーチン DSP00MVS を IGC00yyy として再リンクします。ここで、yyy はゾーン 10 進数タイプ 4 の値の EBCDIC 表現です。DFSIDEF マクロで TYPE=GEN,SVC4=yyy を指定したときに生成されるサンプル JCL を使用して、ルーチンを再リンクすることができます。
4. タイプ 4 SVC ルーチンを LPA ライブラリーまたは MLPA ライブラリーにバインドします。

IMS SVC および DBRC SVC の z/OS への定義

IMS SVC および DBRC SVC を z/OS に定義する際には、以下のフォーマットに従います。

例:

```
SVC Parm 254,REPLACE,TYPE(2)
SVC Parm 255,REPLACE,TYPE(4)
```

関連資料: SVC の z/OS への定義の詳細については、z/OS MVS 初期設定およびチューニング 解説書を参照してください。

関連タスク:

『タイプ 2 SVC モジュールのインストール』

タイプ 2 SVC モジュールのインストール

IMS タイプ 2 SVC を z/OS 中核に組み込む必要があります。

重要: SYS1.NUCLEUS が 2 次エクステントを持つことはできません。z/OS は 2 次エクステントを認識できません。

IMS タイプ 2 SVC を z/OS 中核に取り込むには、以下のいずれかの作業を実行します。

- タイプ 2 SVC を z/OS 中核にバインドする。

以下の 2 つの方法のいずれかで、タイプ 2 SVC を z/OS 中核にバインドできます。

1. バッチ・ジョブでバインド・ユーティリティを呼び出す。
 2. SMP/E USERMOD 用の RECEIVE と APPLY を作成して実行する。
- 中核モジュール・ローダー機能を使用して、SYS1.NUCLEUS からタイプ 2 SVC をロードする。
 1. z/OS 中核にロードする必要のある IMS SVC のリストが含まれている中核モジュール・リスト (NML) を作成する。IMS は IEANS001 NML を使用します。
 2. タイプ 2 SVC をアSEMBルして SYS1.NUCLEUS にバインドする。

この方法は、例として IVP 材料に組み込まれています。

- SYS1.IPLPARM メンバー NUCLSTxx を使用して、SYS1.NUCLEUS からタイプ 2 SVC をロードする。
 1. IMS.SDFSRESL からの IMS SVC を SYS1.NUCLEUS にバインドする。

重要: z/OS システム・プログラマーに問い合わせ、使用すべき適切な NUCLSTxx メンバーを決定します。LOADxx メンバーとそれに関連した NUCLSTxx メンバーの両方が SYS1.IPLPARM に入っていないと、IMS は割り込み禁止待ち状態に入り、IPL プロセスが停止します。
 2. IMS SVC 用の INCLUDE ステートメントを、SYS1.IPLPARM の NUCLSTxx メンバーに定義する。
- SYS1.PARMLIB メンバー NUCLSTxx を使用して、SYS1.NUCLEUS からタイプ 2 SVC をロードする。
 1. IMS.SDFSRESL からの IMS SVC を SYS1.NUCLEUS にバインドする。

重要: z/OS システム・プログラマーに問い合わせて、使用すべき適切な NUCLSTxx メンバーを決定します。LOADxx メンバーとそれに関連した NUCLSTxx メンバーの両方が SYS1.PARMLIB に入っていないと、IMS は割り込み禁止待ち状態に入り、IPL プロセスが停止します。

2. IMS SVC 用の INCLUDE ステートメントを、SYS1.PARMLIB の NUCLSTxx メンバーに定義する。

メモリー・ベースのデータ・セット ENQ 管理の使用可能化

z/OS の MEMDSENQMGMT 機能により、ジョブおよびサブシステムでは、動的に割り振られるデータ・セットに対してメモリー・ベースのデータ・セット ENQ 管理を使用できるようになります。これは、スケジューラー作業領域ベース (SWA ベース) のデータ・セット ENQ 管理よりも高速です。z/OS システムでこの機能を使用可能にすると、動的に割り振られるデータ・セットに対するデータ・セット ENQ は、メモリー内で管理されます。

MEMDSENQMGMT 機能を使用可能にするには、以下の方法のいずれかを使用します。

- ALLOCxx PARMLIB メンバーを更新して、SYSTEM MEMDSENQMGMT 値を ENABLE に設定する。

```
SYSTEM MEMDSENQMGMT(ENABLE)
```

この設定は IPL を越えても有効のままであるため、この方法が推奨されます。

- システム・コマンド SETALLOC SYSTEM,MEMDSENQMGMT=ENABLE を発行し、IMS を再始動して変更を有効にする。

IMS 異常終了フォーマット設定モジュール

IMS は、IMS 異常終了フォーマット設定モジュール (DFSAFMX0) をインストールします。ユーザーによるセットアップは不要であり、モジュール DFSAFMD0 をホスト z/OS システムにインストールする必要はありません。

モジュール DFSAFMD0 は、IMS V11 以降には必要ありませんが、IMS ライブラリー内の DFSAFMD0 を直接指すユーザーをサポートするために、このモジュールは引き続き添付されています。IMS V11 以降に含まれているモジュール DFSAFMD0 は、IMS バージョン 10 以前の異常終了フォーマット設定を提供するために使用できます。

IMS オンライン・ダンプ・フォーマット設定サポートを使用する場合は、IMS V10 をオンラインまたはバッチで実行するすべての z/OS 論理区画 (LPAR) に、モジュール DFSAFMD0 をインストールしておく必要があります。インストールするモジュール DFSAFMD0 のバージョンは、z/OS LPAR で実行する、バージョン 11 より前の IMS バージョンのうちで最上位のものと同一以上でなければなりません。

モジュール DFSAFMD0 の処理は、異常終了時に実行されるオンライン・ダンプ・フォーマット設定専用です。これは、異常終了時に書き込まれた SDUMP データ・セットまたは SYSDUMP データ・セットを使用して行われる、オフライン・ダンプ・フォーマット設定には効果がありません。

推奨事項: オンライン・ダンプ・フォーマット設定を使用する場合は、IMS V11 以降へのマイグレーションが完了し、IMS の以前のバージョンを実行する可能性がなくなるまで、z/OS システムからモジュール DFSAFMD0 をアンインストールしないでください。

モジュール DFSAFMD0 と、IMS V11 以降によってインストールされる DFSAFMX0 動的異常終了出口モジュールは、同じシステム上に共存できます。

関連資料: z/OS ダンプ・フォーマット設定出口のインストールについて詳しくは、「z/OS MVS インストール・システム出口」を参照してください。

異常終了フォーマット設定モジュールのバインド (DFSAFMD0)

同じ論理区画 (LPAR) 上にある IMS バージョン 13、IMS 14、および IMS 15 のインスタンスなど、複数のバージョンの IMS を同時に実行する計画であり、IMS バージョン 13 および IMS 14 にオンライン・ダンプ・フォーマット設定を使用する場合は、IMS 異常終了フォーマット設定モジュール (DFSAFMD0) を、IEAVADFM ダンプ機能インストール・システム出口ルーチン名リストに含まれる z/OS 出口としてインストールする必要があります。

IMS コードを LPAR 上の IMS V11 以降からのみ実行する計画の場合は、モジュール DFSAFMD0 をインストールする必要はありません。

モジュール DFSAFMD0 をバインドするには、以下のようにします。

1. モジュール DFSAFMD0 を SYS1.LPALIB または MLPA ライブラリーにバインドします。
2. モジュール名 DFSAFMD0 を SYS1.LPALIB 内の z/OS ロード・モジュール IGC0805A の IEAVADFM CSECT に追加します。このプロセスにより、IMS モジュール DFSAFMD0 は z/OS SNAP、SYSABEND、および SYSUDUMP 処理のときに制御を受け取り、ダンプ内の IMS 制御ブロックをフォーマット設定することができます。

関連資料: IEAVADFM インストール・プロセスに関する追加情報については、「z/OS MVS インストール・システム出口」を参照してください。

異常終了フォーマット設定モジュールのアンインストール (DFSAFMD0)

IMS バージョン 11 以降にマイグレーションした後、ホストの z/OS システムからモジュール DFSAFMD0 を除去できます。

モジュール DFSAFMD0 をアンインストールするには、以下のようにします。

1. DFSAFMD0 という名前を、SYS1.LPALIB 内のモジュール IGC0805A の IEAVADFM CSECT から除去します。この名前を除去することで、オペレーティング・システムが次の IPL 時にモジュール DFSAFMD0 を異常終了フォーマット設定出口としてインストールしないようにできます。
2. DFSAFMD0 をバインドした SYS1.LPALIB または MLPA ライブラリーからモジュール DFSAFMD0 を除去する。
3. CLPA を使用して再始動し、上記の変更を使用可能にする。

以前に (IMS IVP で行われるように) AMASPZAP ユーティリティを使用して、モジュール DFSAFMD0 を IEAVADFM CSECT にザップしていた場合は、IEAVADFM CSECT から名前 DFSAFMD0 を除去するのに AMASPZAP ユーティリティを使用する必要があります。IEAVADFM CSECT は、8 バイトの項目の表の後に、出口名リストの終わりを示すゼロが入っている最後の 4 バイト項目を続けたものです。それぞれの 8 バイト項目には、ダンプ・フォーマット設定出口ルーチンの名前が入っています。DFSAFMD0 が表内の最後の項目でない場合は、DFSAFMD0 項目を除去するだけでなく、後続の項目をすべて移動させて、表の最後の前にはすべてゼロの項目が決して存在しないようにする必要があります。

以下の例で、モジュール DFSAFMD0 を IEAVADFM CSECT から除去する方法を示します。

1. AMASPZAP ユーティリティを使用して、IEAVADFM CSECT の現在の内容をダンプする。

```
//DMPVADFM JOB ...
//STEP001 EXEC PGM=AMASPZAP
//SYSLIB DD DSN=SYS1.LPALIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
DUMP IGC0805A IEAVADFM
/*
```

2. AMASPZAP ダンプ・ジョブ出力から、IEAVADFM CSECT の内容を調べる。モジュール DFSAFMD0 (16 進では X'C4C6E2C1C6D4C4F0') が入っている項目を見つけてください。

```
**CCHHR- 0022000421 RECORD LENGTH- 000BA0 MEMBER NAME IGC0805A CSECT NAME IEAVADFM
000000 C4C6E2C1 C6D4C4F0 D4E8C4D4 D7E7F0F0 00000000 00000000 00000000 00000000
000020 00000000 07FE0000 00000008 00000000
```

3. AMASPZAP ユーティリティを使用して、モジュール DFSAFMD0 を含んでいる項目をゼロで置き換える。上記の出力例では、モジュール DFSAFMD0 が IEAVADFM CSECT 内の最初の項目で、別の1つの項目がそれに続いています。モジュール DFSAFMD0 を除去するためには、以下に示すように、項目 2 を移動して項目 1 とし、項目 2 をすべてゼロになるようにザップする必要があります。

```
/ZAPVADFM JOB ...
/STEP001 EXEC PGM=AMASPZAP
/SYSLIB DD DSN=SYS1.LPALIB,DISP=SHR
/SYSPRINT DD SYSOUT=A
/SYSIN DD *
NAME IGC0805A IEAVADFM
VER 0000 C4C6E2C1C6D4C4F0
VER 0008 D4E8C4D4D7E7F0F0
REP 0000 D4E8C4D4D7E7F0F0
REP 0008 0000000000000000
```

印刷ダンプ出口制御表へのオフライン・ダンプ・フォーマット設定ルーチンの追加

オフライン・ダンプ・フォーマット設定モジュール名を、SYS1.PARMLIB メンバー BLSCECT にある印刷ダンプ出口制御表に追加します。

項目は以下を含んでいる必要があります。

モジュール名 DFSOFMD0

出口フラグ 0
ユーザー verb IMSDUMP

IPCS ダイアログ (IPCS ISPF 選択 2.6) のコンポーネント分析セクションから、IMS ダンプ・フォーマッターを入手することもできます。

SDFSRESL が LNKLSTxx がない場合、IPCS ユーザーが DFSOFMD0 をロードできるようにするには、JOBLIB または STEPLIB 連結に SDFSRESL がなければなりません。

関連資料:

- 出口制御表については、「z/OS MVS 初期設定およびチューニング 解説書」を参照してください。
- IMS ダンプ・オプションの制御およびオフライン・ダンプ・フォーマッターについての詳細は、「IMS Version 15 Diagnosis」を参照してください。

DBRC タイプ 4 SVC のバインド

DBRC タイプ 4 SVC を LPALIB または MLPA ライブラリーにバインドします。これは IGC00nnn という名前です。ここで nnn は符号付き 10 進数の SVC 番号です。

許可プログラム機能での IMS システム・データ・セットの許可

領域タイプ CTL (DB/DC 領域タイプ)、DBC (DBCTL 領域タイプ)、または DCC (DCCTL 領域タイプ) を持つ JOBLIB/STEPLIB を使用する場合は、すべての JOBLIB/STEPLIB の連結は APF 許可されている必要があります。

以下の IMS システム・データ・セットは APF 許可されている必要があります。

- IMS.SDXRRESL
- IMS.SDFSRESL
- IMS.SDFSJLIB
- IMS.MODBLKSA、IMS.MODBLKSB
- DB2 モジュールおよび表のロード先となる DFSESL DD、または JOBLIB または STEPLIB

これらのデータ・セットに加えて、DB/DC または DCCTL 環境では SYS1.CSSLIB が APF 許可されている必要があります。これは、ユーザーが APPC/z/OS を使用するかどうかとは無関係に適用されます。SYS1.CSSLIB が LNKLSTxx にあって LNKLSTxx が許可されていても、SYS1.CSSLIB が IEAAPFxx にもなければなりません。これは、IMS が、LNKLSTxx 連結を使用しないで SYS1.CSSLIB にアクセスするためです。SYS1.CSSLIB は明示的に APF 許可されている必要があります。

推奨事項: 複数レベルの IMS が稼働している場合、あるいは新たなバージョンまたはリリース・レベルへマイグレーションしようとしている場合は、IMS.SDFSRESL を LNKLSTxx に入れてはなりません。

IMS は、データ・セット許可に関する z/OS 規則に準拠しています。IMS ジョブ・ステップを許可する場合、そのジョブ・ステップで使用されるすべてのライブラリーも許可します。ある IMS バッチ領域を無許可として実行するためには、

IMS.SDFSRESL に無許可ライブラリーを連結してください。連結するには、バッチ・ジョブに、IMS.SDFSRESL を指す DFSRESLB DD ステートメントがある必要があります。

z/OS に対してライブラリーを許可するには、SYS1.PARMLIB の PROGxx メンバーを使用することもできます。

関連資料: 「z/OS MVS 初期設定およびチューニング 解説書」に記載されている IEAAPFxx に関する情報を参照してください。

APPC/MVS 管理ダイアログの更新

IMS TP プロファイルを用いて APPC/MVS 管理ダイアログ・ユーティリティーを使用する場合は、トランザクション・スケジューラーとして「IMS」を最初に追加しておく必要があります。これを行うには、変数 QASTSPE が定義されている非表示パネル ICQASE00 に 1 行を追加する必要があります。

追加する行の形式は次の通りです。

```
IMS,DFSTPPE0'
```

現在割り当てられている最終行の単一引用符 (') を、正符号 (+) に変えることも必要です。

さらに、IMS.SDFSEXEC を TSO SYSPROC 連結に追加し、IMS.SDFSPLIB を TSO ISPLIB 連結に追加しなければなりません。

このパネルの詳しい変更方法については、「z/OS MVS 計画: APPC/MVS 管理」の『ダイアログのカスタマイズ』を参照してください。

RRS アーカイブ・ログ・ストリームの考慮事項

z/OS リソース・リカバリー・サービス 機能を使用する場合は、オプションの RRS アーカイブ・ログ・ストリームが MVS ロガーにさらに大量のロギングを行なうことが分かっているので注意してください。

RRS アーカイブ・ログを使用する場合は、システム・ログ・ストリームのパフォーマンスへの影響を十分にモニターしている必要があります。IMS も RRS もいずれも RRS アーカイブ・ログ・ストリームのデータを診断目的には使用しません。したがって、この機能を使用する必要はありません。

システム管理機能 DDCONS パラメーターの考慮事項

起こりうる処理の遅延を回避するために、APAR OY31613 での記載に従って、システム管理機能 (SMF) DDCONS パラメーターの設定を検討する必要があります。

オプションは DDCONS です。取りうる 2 つのサブオプションは YES と NO です。

YES (デフォルト) 統合を行うよう要求します。

NO 統合を行わないよう要求します。

IRLM のための z/OS インターフェースに関する考慮事項

z/OS 上に IRLM を確実に正しく完全にインストールするためには、考慮すべき多数の要件と必要なステップがあります。

以下のステップで、これらの要件と必要なアクションについて説明します。

- IRLM CTRACE モジュールを z/OS リンク・リストに追加する IRLM CTRACE スタート・ストップ・ルーチン・ロード・モジュールである DXRRL183 は、z/OS リンク・リスト (LL) 上になければなりません。このモジュールは、IRLM 用の自動リスタート・マネージャー (ARM) サポートも含んでいます。
- IRLM を許可プログラム機能 (APF) で許可する IMS.SDXRRESL システム・データ・セットは、APF 許可されている必要があります。
- IRLM サブシステム名を作成する z/OS 事前調整により、すでに IRLM および JRLM がサブシステム名として定義されています (ユーザーによって削除されている場合を除きます)。ユーザーはこれらの名前を使用することも、独自の名前を定義することもできます。z/OS システムで実行するそれぞれの IRLM ごとに z/OS サブシステム名項目を作成します。2 つの IRLM が同一の z/OS システム上にある場合は、それぞれが固有の z/OS サブシステム名を持っていないければなりません。

IRLM が発行する各メッセージには、IRLM z/OS サブシステム名 (開始プロシージャの IRLMNM) と ID (開始プロシージャの IRLMID) を連結したものが含まれます。どの IRLM がその特定のメッセージを出したか容易に識別できるような命名規則をお勧めします。以下の IRLM コマンドは、この IRLM あるいは共用グループに関連する、すべての IRLM 名および ID を表示します。

```
F irImproc,STATUS,ALLI
```

- IRLM 項目を z/OS プログラム特性テーブル (PPT) に追加する z/OS 事前調整により、すでに PPT 項目が DXRRLM00 に対して定義されています (ユーザーによって削除されている場合を除きます)。
- 印刷ダンプ出口制御テーブルを更新する 項目は以下を含んでいる必要があります。

モジュール名 DXRRLM50

出口フラグ 0

ユーザー verb IRLM

下記のいずれかが正しいことを確認します。

- 印刷ダンプ・フォーマット設定モジュール DXRRLM50 が SYS1.LINKLIB にあること。
- ダンプを印刷するジョブに、モジュールが入っているライブラリーを指定する JOBLIB ステートメントまたは STEPLIB ステートメントが含まれていること。

関連資料:

- オフライン・ダンプ・フォーマッターに関する詳細については、「IMS Version 15 Diagnosis」を参照してください。

- PPT への IRLM 項目の追加について詳しくは、151 ページの『z/OS プログラム特性テーブルへの IRLM 項目の追加』を参照してください。
- 以下の事項については、z/OS MVS 初期設定およびチューニング 解説書を参照してください。
 - z/OS に対するサブシステムの定義。
 - IEAAPFxx。
 - メッセージへの応答と、トレース・オプションおよびパラメーターを含めるための PARMLIB メンバーのセットアップ。

第 6 章 VTAM インターフェースの考慮事項

使用する IMS システムが VTAM を必要とする場合は、VTAM モード・テーブルには IMS に対して定義されているすべての VTAM 端末用の項目が含まれていなければなりません。

ユーザーは、ログオン時にこのテーブルの項目名を、以下のいずれにも使用できます。

- VTAM VARY コマンドの LOGMODE パラメーター
- /OPNDST コマンドの MODE パラメーター
- 他の端末の INIT SELF コマンドのパラメーター
- TERMINAL マクロの MODETBL パラメーター

MODETBL パラメーターは、ACF/VTAM LOGON または SCIP 出口 CINIT で提供された、他の入力を変更します。 所定の端末によるすべての並列セッションの MODETBL 名は、同じです。クロスドメイン・リソースに対して MODETBL を指定してはなりません。

モード・テーブル項目はセッション・パラメーターを作成し、これにより、IMS と端末との間に確立されたセッションを制御します。

関連資料: VTAM 論理装置の BIND パラメーターのリストについては、「IMS V15 コミュニケーションおよびコネクション」を参照してください。

以下の端末をすべて定義してください。

- 3600、3614、および SLU P を LUTYPE=0 として
- SLU 1 を LUTYPE=1 として
- SLU 2 を LUTYPE=2 として
- LU 6 を LUTYPE=6 として

SLUTYPE1 として定義した 3770P または 3790 端末は、モード・テーブル項目で「不在」として定義する必要があります。 SLU 1 端末を 2 次端末に対して「例外」または「確定応答」として定義できます。 SLUTYPEP として定義された端末の場合、BIND コマンドの最初の 7 バイトにはオプションは許されません。

IMS 用の VTAM APPL マクロに PARSESS=NO を指定した場合、VTAM 並列セッション・サポートは組み込まれません。この場合、IMS は VTAM START パラメーターの MAXAPPL キーワードにおいて「1」としてカウントします。

IMS 用の VTAM APPL マクロに PARSESS=YES を指定した場合、VTAM 並列セッション・サポートがシステムに組み込まれます。IMS は VTAM START パラメーターの MAXAPPL キーワードにおいて「2」としてカウントします。

関連資料: 並列セッションに対する IMS サポートの詳細は、「IMS V15 コミュニケーションおよびコネクション」を参照してください。

重要: z/OS および VTAM インターフェースのステップの完了後には、z/OS を始動して、CLPA か MLPA=xx、またはその両方を指定する必要があります。

ネットワーク制御プログラム (NCP) の遅延設定

HOST マクロの DELAY パラメーターの値を 0、またはユーザーのシステムの他の作業を考慮したうえで、可能な限り小さな値に設定します。

第 7 章 CSL の管理

CSL の管理に関連した作業は、一般的には、システム管理者またはシステム・オペレーターが行います。

CSL を使用した z/OS 自動リスタート・マネージャーの使用

CSL アドレス・スペース (ODBM、OM、RM、SCI) は、要求された場合、z/OS 自動リスタート・マネージャー (ARM) に登録できます。ARM は、開始タスクの可用性を改良できる z/OS リカバリー機能です。タスクが失敗したか、またはタスクが実行されているシステムに障害が起きた場合、ARM は、オペレーターの介入なしにタスクを再始動することができます。

IBM は、自動再始動管理にポリシー・デフォルトを提供しています。ユーザーはこのデフォルトを使用することもできるし、CSL アドレス・スペースの再始動を指定する固有の ARM ポリシーを定義することもできます。ARM ポリシーは、システムに障害が起きたとき、または CSL アドレス・スペースに障害が起きたとき、何を行うかを指定します。

ARM を使用可能にするには、次の 2 つの方法のどちらかで ARMRST=Y を指定します。

- 以下の CSL アドレス・スペース初期設定 PROCLIB メンバー・データ・セットの中で
 - ODBM の場合、CSLDIxxx
 - OM の場合、CSLOIxxx
 - RM の場合、CSLRIxxx
 - SCI の場合、CSLSIxxx
- 実行パラメーターとして

ARM が使用可能になると、CSL アドレス・スペースは、ARM エlement名を使用して ARM に登録されます。次の表に、ARM エlement名の定義を示します。

表 7. ARM エlement名

CSL アドレス・スペース	ARM エlement名
ODBM	『CSL』 + odbmname + 『OD』
OM	『CSL』 + omname + 『OM』
RM	『CSL』 + rmname + 『RM』
SCI	『CSL』 + sciname + 『SCI』

注: CSL アドレス・スペースの名前は、実行パラメーターとして、またはその CSL アドレス・スペースの初期設定 PROCLIB メンバー・データ・セットの中で定義された名前です。例えば、PROCLIB メンバー・データ・セット CSLOIxxx の中で OMNAME=OM1A である場合、ARM エlement名は CSLOM1AOM です。

ARM ポリシーの中で、各 CSL アドレス・スペース用に適切な ARM エLEMENT 名を使用してください。詳しくは、「z/OS MVS シスプレックスのセットアップ」を参照してください。

モジュールの中に、各 CSL アドレス・スペース用の異常終了テーブルが存在します。

- ODBM の場合、CSLDARM0
- OM の場合、CSLOARM0
- RM の場合、CSLRARM0
- SCI の場合、CSLSARM0

このテーブルは、異常終了が起きた後、ARM が CSL アドレス・スペースを再始動しない異常終了をリストします。このテーブルは、変更することができます。

グローバル・オンライン変更の管理

このトピックでは、グローバル・オンライン変更を IMSplex 用に使用可能および使用不可にする方法について説明します。このトピックでは、混合オンライン変更の有効範囲を備えた IMSplex の保守についても説明します。

グローバル・オンライン変更の使用可能化

グローバル・オンライン変更の使用可能化には、IMSplex 内のすべての IMS システムを同時にシャットダウンする必要はありません。一度に 1 つの IMS のグローバル・オンライン変更を使用可能にすることができます。すべての IMS システムがグローバル・オンライン変更をサポートするよう切り替えた後は、IMSplex はグローバル・オンライン変更のみをサポートします。

グローバル・オンライン変更では CSL が必要とされます。以下のステップは、CSL 環境がすでに確立されていることを前提としています。グローバル・オンライン変更を使用可能にするには、以下のステップを一度実行する必要があります。

1. OLCSTAT データ・セット特性を定義する。
2. OLCSTAT を初期化する。

OLCSTAT データ・セットの定義と初期設定を行った後、一度に 1 つの IMS に対して以下のことを実行することができます。

1. MODSTAT DD ステートメントを IMS 制御領域の JCL から除去する。
2. MODSTAT2 DD ステートメントを IMS 制御領域の JCL から除去する (XRF の場合)。
3. グローバル・オンライン変更に関連した DFSCGxxx PROCLIB メンバー・データ・セットのパラメーターを定義する。
 - a. OLC=GLOBAL 指定し、グローバル・オンライン変更を使用可能にする。
 - b. オンライン変更データ・セットの名前を OLCSTAT= に指定する。
IMSplex 内のすべての IMS システムは、同じ OLCSTAT データ・セットを指定する必要があります。
4. IMS をシャットダウンする。
5. IMS をコールド・スタートします。

バッチ IMS システムのマイグレーションを行う場合、DBBATCH JCL を変更することにより、調整した (グローバル) オンライン変更を使用可能にできます。

- OLCSTAT DD ステートメントの追加
- MODSTAT DD ステートメントの削除

グローバル・オンライン変更の使用不可能化

グローバル・オンライン変更を使用不可にする際に、IMSplex 内のすべての IMS システムを同時にシャットダウンする必要はありません。また、グローバル・オンライン変更を使用不可にしたなら、IMSplex はローカル・オンライン変更のみをサポートします。

グローバル・オンライン変更を使用不可にする準備ができれば、一度に 1 つの IMS に対して以下のステップを実行します。

1. IMS をシャットダウンする。
2. OLC=GLOBAL を OLC=LOCAL に変更してローカル・オンライン変更を使用可能にするために、オンライン変更用の DFSCGxxx PROCLIB メンバー・データ・セットのパラメーターを変更する。
3. INITMOD ジョブを実行して、MODSTAT データ・セットを初期化する。
4. MODSTAT DD ステートメントを IMS 制御領域の JCL に組み込む。
5. MODSTAT2 DD ステートメントを IMS 制御領域の JCL に組み込む (XRF の場合)。
6. IMS をコールド・スタートする。

一般に、グローバル・オンライン変更時にダウンした IMS は、再び立ち上げる際にはコールド・スタートさせる必要があります。コールド・スタートが必要となるのは、一致しない現行オンライン変更ライブラリーに対してログ・レコードの処理を行って再始動されるのを防ぐためです。そのようにすると、再始動が失敗したり、あるいは後で重大エラーが生じたりする可能性があります。しかし、IMS がダウンしたのが、最後のグローバル・オンライン変更のときのみであり、再始動が最後のグローバル・オンライン変更と矛盾しない場合には、IMS をウォーム・スタートさせることができます。

IMS のダウンが最後のオンライン変更時のみに起きた場合でも、その再始動のタイプが最後のオンライン変更のタイプと矛盾する場合、あるいは IMS のダウンが複数のグローバル・オンライン変更で起きた場合には、IMS はコールド・スタートさせる必要があります。

混合オンライン変更有効範囲を持つ IMSplex の保守

IMSplex で、グローバル・オンライン変更をサポートする IMS システムとそれをサポートしない IMS システムの両方が混合している場合、ユーザーがオンライン変更をグローバル・オンライン変更と手動で調整する。

OLC=GLOBAL で定義される IMS システムは、グローバル・オンライン変更に参加できます。

例えば、IMSplex 全体にわたって一度に 1 つの IMS のグローバル・オンライン変更を使用可能にするときには、混合 IMSplex にすることができます。このマイグレ

ーションの間にオンライン変更を実行することにした場合、OLC=LOCAL を指定した IMS のオンライン変更を、手動でグローバル・オンライン変更と調整する必要があります。

関連資料: グローバル・オンライン変更プロセスについて詳しくは、577 ページの『グローバル・オンライン変更機能の概要』を参照してください。

自動 RECON 損失通知の管理

RECON に対する IMSplex 名が指定された後、RECON 損失通知は自動です。

自動 RECON 損失通知は、1 つの DBRC インスタンスが RECON エラーを処理したらすぐにアクティブにされます。そのエラーは、SCI を介して他の DBRC インスタンスへ即時に伝えられます。RECON エラーを通知された DBRC インスタンスは、通知を受けたことを示すメッセージ (DSP1141I) を発行します。

RECON のセットを共用する DBRC インスタンスはすべて、同じ IMSplex を使用して、すべての DBRC インスタンスへ RECON エラーを確実に通知することも必要です。ある特定の RECON のセットに初めて RECON 損失通知を使用するとき、IMSplex 名が RECON に設定されます。その後で別の IMSplex を使用してまたは IMSplex を使用しないで RECON のアクセスを試みても、それはリジェクトされ、メッセージ DSP1136A が発行されます。

ARLN がアクティブの場合、または並列 RECON アクセスが使用可能な場合で、RECON のセットに関連付けられた IMSplex を変更するときは、CHANGE.RECON コマンド上のパラメーター IMSPLEX() | NOPLEX を使用します。CHANGE.RECON IMSPLEX() | NOPLEX コマンドの実行時にいずれかの DBRC インスタンスがアクティブである場合、このコマンドはリジェクトされ、メッセージ DSP1137I が発行されます。以下のリストに、RECON のセットに関連付けられた IMSplex を変更するための手順を簡単に要約します。

1. 現行 IMSplex 上のすべての DBRC アクティビティーが終わるのを待つ。
2. IMSplex 名を変更するために DBRC ユーティリティー・ジョブを実行依頼する。
3. ユーザー出口ルーチン DSPSCIX0 で IMSplex 名を変更する。
4. 新規の IMSplex SCI が作動可能であることを確認する。
5. 新規 IMSplex 上で DBRC アクティビティーを再開する。

注: RECON 内の IMSplex 名を変更するのは、SCI 登録出口ルーチンが変更されて新規の IMSplex 名を戻すようになったときだけにしてください。

自動 RECON 損失通知を停止またはリセットするには、IMSplex 名をクリアする必要があります。以下に、RECON のセットに関連付けられた自動 RECON 損失通知を停止するための手順を簡単に要約します。

1. 現在の IMSplex 上のすべての DBRC アクティビティーが終了するまで待つ。
2. IMSplex 名を変更するために DBRC ユーティリティー・ジョブを実行依頼する。
 - a. コマンド CHANGE.RECON IMSPLEX() | NOPLEX を発行する。

要件: 並列 RECON アクセスが使用可能で、ARLN を使用不可にする場合、CHANGE.RECON NOPLEX コマンド発行前に CHANGE.RECON ACCESS(SERIAL) コマンドを発行して並列 RECON アクセスを使用不可にする必要があります。

関連資料:

- 自動 RECON 損失通知についての詳細は、725 ページの『RECON 損失の通知』を参照してください。
- 並列 RECON アクセスについての詳細は、692 ページの『並列モードでの RECON データ・セットへのアクセス』を参照してください。

IMSplex のモニター

CSL を備えた IMSplex 内の IMS システムのモニターは、他の環境のスタンドアロン IMS システムをモニターするのと違いはありません。しかし、全体としては、IMSplex のモニターは、個々の IMS システムのモニターとは異なります。

システム状況は QRY IMSPLEX SHOW(ALL) コマンド発行により確認できますが、このコマンドは IMSplex 内でどの IMS システムがアクティブであるか、またそれらシステムの状況を表示するものです。

IMSplex をある側面からモニターするのに便利な QUERY コマンドがこの他にいくつかあります。QUERY コマンドは、特定の IMS システムに経路指定されている場合を除き、IMSplex 全体で作動します。例えば、次のようになります。

- QRY TRAN QNT(GT,*nn*)。ここで、*nn* はユーザーが指定する数です。

応答では、キュー・カウントの高いトランザクションが表示されます。

- QRY DB STATUS(ALLOCF,STOSCHD)

応答では、割り振り失敗のある、または停止されたデータベースが表示されます。この他にもさまざまな状況が指定できます。

IMSplex 問題の診断

OM、RM、および SCI の各共通サービス層アドレス・スペースは、内部エラーに対して SDUMP を作成します。CSL ダンプは SYS1.DUMP データ・セット内にあります。

IMSplex 問題をうまく診断するには、以下の種類のシステム情報のうちの 1 つ以上を収集する必要がある場合があります。

- CSL メンバーがある各論理区画 (LPAR) からの SYSLOG
- IMSplex のメンバーおよびその状況を表示するタイプ 2 QUERY コマンドの出力
- CSL アドレス・スペースの z/OS SVC ダンプ

CSLOIxxx PROCLIB メンバー・データ・セット上の IMSPLEX キーワードの AUDITLOG= パラメーターを指定することにより、OM ログ・レコードを z/OS

ロガー・データ・ストリームに書き込むことができます。ログ・レコードは、コマンド入力、関連コマンド出力、および非送信請求メッセージ出力の監査証跡を提供します。

監査証跡中のメッセージは、TSO SPOC を使用して表示することができます。アクション・バーを使って、「SPOC」 > 「監査証跡」を選択し、該当するデータ・ストリーム名および時間を入力します。

IMSplex 問題の診断について詳しくは、「IMS Version 15 Diagnosis」を参照してください。

第 8 章 CSL ODBM の管理

CSL Open Database Manager (ODBM) は、IMSplex 内の DBCTL システムおよび DB/TM IMS システムで IMS DB によって管理されるデータベースへのアクセスをサポートします。

ODBM は、以下のリソース・アダプターと API を使用するアプリケーション・プログラムからのデータベース接続要求およびアクセス要求を管理します。

- IMS Universal Database リソース・アダプター
- IMS Universal JDBC ドライバー
- IMS Universal DL/I ドライバー
- Open Database Access (ODBA) インターフェース
- ODBM CSLDMI インターフェース

ODBA アプリケーション・プログラムと CSLDMI インターフェースに合わせて直接作成されたアプリケーション・プログラムの場合を除いて、すべてのデータベース接続要求およびアクセス要求は、IMS Connect によってリソース・アダプターおよび API から ODBM に送られます。ODBM は、IMS Connect から受け取ったデータベース接続要求およびアクセス要求を、要求されたデータベースを管理している IMS システムに送ります。

IMS Connect を通じて ODBM と通信する IMS Universal ドライバーは、低レベル通信プロトコルとしてオープン・スタンダードの分散リレーショナル・データベース体系 (DRDA) を使用します。ODBM は、DRDA の一部である分散データ管理 (DDM) 体系から着信するデータベース要求を、IMS が想定する DL/I 呼び出しに変換します。ODBM は、IMS 出力をクライアントに返すとき、応答を元の DDM プロトコルに変換します。

ODBA アプリケーション・プログラムと、ODBM CSLDMI インターフェースを使用するアプリケーション・プログラムは、DRDA を使用せず、代わりに ODBA インターフェースがサポートする DL/I 呼び出しを使用します。

既存の ODBA アプリケーションは、ODBM を使用することで、DL/I 処理中の ODBA アプリケーションの予期せぬ終了に起因する IMS の異常終了を防ぐこともできます。ODBM が ODBA アプリケーション・プログラムをサポートするには、アプリケーション・プログラム自体の変更は不要であり、IMS DB への接続を定義する DFSPRP マクロにわずかな変更を加えるだけで済みます。

ODBM から見ると、IMS Connect など、CSLDMI インターフェースを通じて ODBM と直接インターフェースするアプリケーション・プログラムは、ODBM クライアントです。ユーザーは、ODBM CSLDMI インターフェースを使用して独自の ODBM クライアントを開発できます。ODBM クライアント・アプリケーション・プログラムは、IMSplex 内のすべての LPAR で IMS DB によって管理されるデータベースにアクセスできます。

関連概念:

CSL 内での ODBM クライアントの登録

アプリケーション・プログラムが ODBM を介して IMS データベースにアクセスするには、その前にアプリケーション・プログラム、またはそれらのアプリケーション・サーバーを ODBM にクライアントとして登録する必要があります。ODBM CSLDMI インターフェースの CSLDMREG 要求を発行して、ODBM クライアントを ODBM に登録します。

ODBM クライアントを ODBM に登録した後、ODBM クライアント上で実行されるアプリケーション・プログラムは、ODBM を介して IMS への ODBA 呼び出しを発行することができます。

ODBM クライアントは、ODBM に登録する前にまず SCI に登録する必要があります。ODBM クライアントは、IMS への ODBA 呼び出しの発行を完了したら、CSLDMDRG 要求を発行して登録を解除できます。

CSLDMREG および CSLDMDRG 要求など、ODBM CSLDMI インターフェースの詳細については、「IMS V15 システム・プログラミング API」を参照してください。

ODBM および RRS

CSL Open Database Manager (ODBM) は、z/OS リソース・リカバリー・サービスを使用しても、使用しなくても実行することができます。デフォルトでは、ODBM は RRS と共に実行されます。

ODBM を RRS とともに実行するときは、RRS が同期点マネージャーになり、UOR に関連付けられている、IMS 以外のリソース・マネージャーを含め、すべてのリソース・マネージャーを調整します。RRS と共に実行するとき、ODBM は 2 フェーズの同期点プロトコルを使用します。

RRS なしで実行するとき、ODBM は、データベース・リソース・アダプター (DRA) インターフェースを介して IMS DB に接続する CCTL と同様に機能しますが、CICS CCTL とは異なり、ODBM は同期点マネージャーとしては機能しません。すべての同期点管理は、ODBM クライアントによって実行される必要があります。

DRA インターフェースの詳細については、「IMS V15 システム・プログラミング API」の『DRA (データベース・リソース・アダプター)』を参照してください。

ODBM に対する RRS のサポートは、ODBM 初期設定 PROCLIB メンバー CSLDIxxx 内の RRS キーワードによって制御されます。これについては、「IMS V15 システム定義」で説明されています。

ODBM と ODBA

Open Database Manager (ODBM) は、Open Database Access (ODBA) インターフェースをサポートおよび使用します。

ODBM は、ODBA インターフェースを使用して IMS と通信します。CSLDCxxx ODBM 構成 PROCLIB メンバー内のパラメーターの多くは、DFSPRP マクロ・パラメーターから作成される DFSxxxx0 始動テーブルで ODBA が使用するパラメーターと同じです。

ODBM は ODBA を使用するため、Db2 for z/OS や WebSphere Application Server for z/OS などの ODBA アプリケーション・サーバーは、ODBA ではなく ODBM を介して IMS に接続するように構成できます。ODBM を介して接続すると、DL/I 処理中に DB2 ストアード・プロシージャまたは WebSphere Application Server アプリケーション・プログラムが予期せずに終了した場合に、U0113 異常終了が発生するのを防ぐことができます。

ODBM を使用するように ODBA アプリケーション・サーバーを構成する場合、その ODBA アプリケーション・サーバーの下で実行される既存のアプリケーション・プログラムを変更する必要はありません。

ODBM を使用する ODBA アプリケーション・サーバーの構成

Db2 for z/OS や WebSphere Application Server for z/OS など、Open Database Access (ODBA) インターフェースを使用するアプリケーション・サーバーを、IMS 共通サービス層 (CSL) の Open Database Manager (ODBM) コンポーネントを介して IMS に接続するように構成することができます。

前提条件:

ODBA を使用するアプリケーション・サーバーを、ODBM を介して IMS に接続するように構成するには、ODBA インターフェースを介した IMS データベースへのアクセス (コミュニケーションおよびコネクション)の説明に従って ODBA 環境を構成する必要があります。

また、ODBM には CSL Structured Call Interface (SCI) と CSL Operations Manager (OM) も必要です。

制約事項: ODBA アプリケーション・サーバーは、ODBM および IMS と同じ z/OS LPAR 上にある必要があります。

ODBA アプリケーション・サーバーを ODBM を使用するように構成するには、次のようにします。

1. データベース・リソース・アダプター (DRA) の始動テーブル DFSxxxx0 (ここで xxxx は、APSB 呼び出しで指定される IMS システムの別名) の作成に使用される DFSPRP マクロで、ODBA 接続属性に加えて、IMSPLEX パラメーターを指定します。必要に応じて、ODBMNAME パラメーターも指定します。

ODBMNAME パラメーターが指定されていない場合、ODBA は、CSL Structured Call Interface (SCI) がデータベース接続要求を経路指定するために使用する ODBM インスタンスを選択します。

2. オプション: ODBA アプリケーション・サーバーを、ODBA 呼び出しが処理される専用コンテキストを取得および管理するように構成します。ODBA アプリケーション・サーバーが専用コンテキストを取得しない場合、ODBA が ODBA アプリケーションに代わって専用コンテキストを取得および管理します。

以下のサンプル JCL は、DFSSYS10 という名前の DRA 始動テーブルをセットアップします。ODBM の使用は、IMSPLEX=PLEX1 および ODBMNAME=ODBM1 の指定によって示されます。MINTHRD および MAXTHRD の値は、スレッドの除去と再作成の繰り返しを避けるために使用するものと同じ値です。

```
//DFSSYS10 JOB A,JIM,CLASS=Q,MSGLEVEL=(1,1),MSGCLASS=A,
// USER=username,PASSWORD=password,
// REGION=3072K
/*ROUTE PRINT THISCPU/CECTOOL
/*
/*-----
/* Build ODBA PRP member *
/*-----
/* SYS1, CRESLIB
/******
/* *****
/* ASSEMBLE AND LINK DFSSYS10 USING IMS MACROS
/* Test TIMER = 100 Error. We will see RC=08. *****
/******
//ASMLNK10 EXEC ASMLK11A,
// PARM.LINK='SIZE=(880K,64K),NCAL,LET,,XREF,LIST',
// LNKOUT='IMSTESTL.TNUC0'
//ASSEM.SYSIN DD * SUFFIX SYS1
DFSPPR DSECT=NO,FPBUF=10,FPBOF=5,CNBA=150,DBCTLID=IMS1,
MINTHRD=10,MAXTHRD=10,FUNCLV=2,IDRETRY=100,
IMSPLEX=PLEX1,ODBMNAME=ODBM1,SOD=A,TIMEOUT=2,
DSNAME=IMS.RESLIB,DDNAME=DFSIVD1
/*
//LINK.SYSIN DD *
NAME DFSSYS10(R)
/*
/*
```

関連概念:

 CSL の定義と調整 (システム定義)

関連資料:

 DRA 始動テーブル (システム・プログラミング API)

ODBM メッセージ経路指定

IMSplex 内の IMS システムを ODBM に定義するときは、各 IMS システムに 1 つ以上の別名も割り当てます。別名は、ODBM クライアントと ODBM がデータベース・アクセス要求を IMS システムに経路指定するために使用します。

IMSplex 内の複数の IMS システムで別名が共用される場合、ODBM はラウンドロビン・アルゴリズムを使用して、共用別名を指定する着信要求を共用 IMS システムに配布します。

同様に、着信要求が別名を空白のままにしてある場合も、ODBM はラウンドロビン・アルゴリズムを使用して、着信要求を IMSplex 内で ODBM が使用できるすべての IMS システムに配布します。

ODBM とセキュリティー

CSL Open Database Manager (ODBM) は、ユーザー認証自体もユーザー許可自体も行いません。

ODBM と共に IMS Connect を使用している場合は、要求メッセージが ODBM に到達する前に、IMS Connect のセキュリティーを使用してそれらの要求メッセージのユーザー ID を認証することができます。RACF を直接呼び出せることに加え、IMS Connect は、IMS DB との通信のセキュリティー検査のカスタマイズを容易にするための IMS DB セキュリティー・ユーザー出口ルーチン (HWSAUTH0) を提供します。

RACF を使用して IMS Connect でユーザー ID を認証する場合、IMS Connect が IMS DB への ODBM クライアント接続を認証するために RACF RACROUTE REQUEST=VERIFY 呼び出しを発行したときに、RACF 統計を収集できるようにすることもできます。

IMS でユーザーの権限を検査して、APSB およびリソース・アクセス管理 (RAS) のセキュリティーを使用することにより、PSB を割り振るか、IMS リソースにアクセスすることができます。

APSB セキュリティーは、ODBASE パラメーターを指定することで有効になります。RAS セキュリティーは、ISIS パラメーターで指定されます。ODBASE および ISIS パラメーターは、IMS または DBC の始動プロシージャー、または DFSPBxxx PROCLIB メンバーで指定できます。

ユーザー作成の ODBM クライアントが、RACF などのセキュリティー製品のセキュリティー・オブジェクトを渡す場合、ODBM は RACROUTE REQUEST=VERIFY を呼び出して、APSB スレッドのアクセス制御環境エレメント (ACEE) を作成します。その後、IMS は、PSB の割り振りまたは他のリソースへのアクセスのための APSB 許可または RAS 許可の際に、ACEE を使用できます。

関連概念:

375 ページの『第 19 章 IMS のセキュリティー』

🔗 セキュリティーの確立と定義 (コミュニケーションおよび接続)

関連タスク:

🔗 IMS Connect に対する RACF セキュリティー統計の使用可能化 (コミュニケーションおよび接続)

関連資料:

🔗 UPDATE IMSCON TYPE(CONFIG) コマンド (コマンド)

🔗 ODACCESS ステートメント (システム定義)

ODBM アカウンティング

ODBM アカウンティング機能は、ODBM アドレス・スペース内で処理されるトランザクションについて、CPU 使用量などのアカウンティング情報のロギングを実行します。この情報は、チャージバックに使用できます。

ODBM は、z/OS システム管理機能 (SMF) を利用して、ODBM アカウンティング情報のロギングと取り出しを行います。

CSL ODBM アカウンティング用のログ・ストリームの構成

ODBM アドレス・スペースのロギングは、オプション・パラメーター LOGOPT=ACCOUNTING が ODBM 始動メンバー CSLDIxxx で指定された場合に活動化されます。

1. IMS PROCLIB データ・セットの CSLDIxxx メンバー内で、LOGOPT=ACCOUNTING を指定します。
2. 変更した CSLDIxxx メンバーを使用して ODBM を開始します。
3. z/OS システム管理機能 (SMF) タイプ 29、サブタイプ 1 のログを処理して、ODBM のワークロードおよびチャージバック・データのアカウント処理を行います。SYS1.PARMLIB 内の SMF parmlib メンバー SMFPRMxx は、SMF ロギングの方式とターゲット・データ・セットを定義します。

ロギング方式には、ログ・ストリームとデータ・セットの 2 つがあります。一時点では、これらの方式のどちらか 1 つのみが使用されます。次に示すのは、ログ・ストリーム定義の例です。

```
DEFAULTLSNAME(IFASMF.ALLSYS.DEFAULT)
LSNAME(IFASMF.ALLSYS.DATA,TYPE(29))
```


次に示すのは、データ・セット定義の例です。

```
DSNAME(SYSA.MAN1,SYSA.MAN2,SYSA.MAN3)
```

SMF ユーティリティの IFASMF DL (ログ・ストリームの場合) または IFASMF DP (データ・セットの場合) を発行することで、SMF ログを取得できます。これらのユーティリティの出力は、どちらも SMF ログの未加工ダンプです。

ログに記録された情報には、スケジュールされた PSB の期間中の単一の作業単位 (UOW) からのすべてのアクティビティが含まれます。

関連情報:

 [SMF のセットアップと管理](#)

ODBM アカウンティング用の SMF ログ・レコード

ODBM アカウンティング用に予約されている SMF レコードは、タイプ 29 (x'1D')、サブタイプ 1 のものです。SMF ログ・レコードは、SMF ヘッダー、BPE ヘッダー、および ODBM アカウンティング情報の 3 つの主要なセクションで構成されます。これらはすべて、単一のレコード・インスタンスに含まれます。

BPESMF29 マクロには、SMF タイプ 29 用と BPE ヘッダー用の両方の DSECT が含まれています。SMF ヘッダー内のフィールド smf29bhs には、SMF 29 レコードの先頭から BPE ヘッダーまでのオフセットが含まれています。

CSLDSMF マクロには、ODBM アカウンティング情報 (レコード・サブタイプ 1) の DSECT が含まれています。SMF ヘッダー内のフィールド smf29sts には、SMF 29 レコードの先頭から ODBM アカウンティング情報セクションまでのオフセットが含まれています。

各セクションの詳細を以下の表に示します。

表 8. SMF ヘッダー (タイプ 29) - BPESMF29 DSECT

名前	長さ	タイプ	説明
smf29len	2	バイナリー	レコード長
smf29seg	2	バイナリー	セグメント記述子
smf29flg	1	バイナリー	システム・インディケータ・フラグ: <ul style="list-style-type: none"> • x'80' - 予約済み • x'40' - 使用されたサブタイプ • x'20' - 予約済み • x'10' - MVS/SP バージョン 4 以上 • x'08' - MVS/SP バージョン 3 • x'04' - MVS/SP バージョン 2 • x'02' - VS2 • x'01' - 予約済み
smf229rty	1	バイナリー	レコード・タイプ (29 進数または x'1D')
smf29tme	4	バイナリー	午前 0 時からの経過時間 (100 分の 1 秒単位)
smf29dte	4	パック	レコードが SMF バッファーに移動された日付 (パック 10 進数)
smf29sid	4	EBCDIC	システム識別
smf29ssi	4	EBCDIC	サブシステム ID
smf29sty	2	バイナリー	レコード・サブタイプ
smf29trn	2	バイナリー	このレコード内のトリプレットの数
*	2		予約済み
smf29bhs	4	バイナリー	レコードの先頭から BPE ヘッダーまでのオフセット
smf29bhl	2	バイナリー	BPE ヘッダーの長さ
smf29bhn	2	バイナリー	BPE ヘッダーの番号
smf29sts	4	バイナリー	レコードの先頭からサブタイプ固有のセクションまでのオフセット
smf29stl	2	バイナリー	サブタイプ固有のセクションの長さ
smf29stn	2	バイナリー	サブタイプ固有のセクションの番号

表 9. BPE ヘッダー - BPESMF29_BPEHDR DSECT (レコードの先頭からのオフセットが smf29bhs の位置にある)

名前	長さ	タイプ	説明
smf29bh_fieldFlags	4	バイナリー	フィールド・フラグ
smf29bh_asType	4	EBCDIC	このレコードが書き込まれるアドレス・スペース・タイプ
smf29bh_jobName	8	EBCDIC	ジョブまたは開始タスク名
smf29bh_asName	8	EBCDIC	1. バッチ/IMS 制御領域関連のアドレス・スペース (CTL、DLI、DBRC、DEP) 用の IMSID (または DBCTL RSENAME) 2. BPE 管理アドレス・スペース用の BPE アドレス・スペース「システム名」
smf29bh_crType	1	バイナリー	関連制御領域のタイプ: <ul style="list-style-type: none"> • x'00' - 制御領域アドレス・スペース以外 • x'01' - TM/DB IMS • x'02' - DBCTL IMS • x'03' - DCCTL IMS • x'04' - FDBR 領域
smf29bh_flag1	1	バイナリー	フラグ・バイト: <ul style="list-style-type: none"> • x'80' - VUE として登録される IMS または BPE アドレス・スペース • x'40' - IMS 従属領域または DBCTL/ODBA スレッドによって書き込まれるレコード • x'20' - IMS バッチ領域によって書き込まれるレコード • x'10' - IRLM 使用の IMS • x'08' - DBRC 使用の IMS • x'04' - 共用キュー使用の IMS • x'02' - CSL 使用の IMS • x'01' - 予約済み
*	2		予約済み
smf29bh_asVersion	3	EBCDIC	アドレス・スペースのバージョン番号 (バイナリー)
smf29bh_bpeVersion	3	EBCDIC	BPE バージョン番号または 0 (バイナリー)
smf29bh_asid	2	バイナリー	このレコードが書き込まれるアドレス・スペースの ASID
*	4		予約済み
smf29bh_startStck	8	EBCDIC	1. バッチおよび制御領域関連のアドレス・スペース (CTL、DLI、DBRC、DEP) の IMS バッチまたは制御領域の先頭の STCK 値 2. BPE ベースのアドレス・スペースのアドレス・スペース先頭の STCK 値
smf29bh_stck	8	EBCDIC	レコードが作成された時点の現行 STCK

表 10. ODBM アカウンティング情報サブタイプ 1 – CSLDSMF DSECT (レコードの先頭からのオフセットが smf29sts の位置にある)

名前	長さ	タイプ	説明
smf29sty1_pver	4	バイナリー	バージョン番号
smf29sty1_func	4	バイナリー	機能番号
smf29sty1_aliasName	4	EBCDIC	別名 (使用可能な場合)
smf29sty1_plexName	8	EBCDIC	IMSpIex 名
smf29sty1_psbName	8	EBCDIC	PSB 名 (使用可能な場合)
smf29sty1_apSBToken	16	EBCDIC	APSB トークン 注: このトークンは、x'08' レコード内の IMS ユーザー定義トークンと関連しています。
smf29sty1_apSBSTCK	8	EBCDIC	APSB STCK
smf29sty1_clientidlen	4	バイナリー	クライアント ID の長さ
smf29sty1_clientidoff	4	バイナリー	ODBM サブタイプ・レコード の先頭からクライアント ID 名までのオフセット (利用可能な場合)
smf29sty1_useridlen	4	バイナリー	ユーザー ID の長さ
smf29sty1_useridoff	4	バイナリー	ODBM サブタイプ・レコード の先頭からユーザー ID 名までのオフセット (利用可能な場合)
smf29sty1_racfIen	4	バイナリー	RACF ID の長さ
smf29sty1_racfIoff	4	バイナリー	ODBM サブタイプ・レコード の先頭から RACF ID 名までのオフセット (利用可能な場合)
smf29sty1_UOWtime	8	EBCDIC	合計 UOW 時間 (STCK 単位)
smf29sty1_zIIPtime	8	EBCDIC	合計 zIIP 時間 (STCK 単位)
smf29sty1_numDLI	4	バイナリー	DLI 呼び出しの総数
smf29sty1_numSQL	4	バイナリー	実行可能 SQL コマンドの総数

第 9 章 CSL OM の管理

このトピックでは、OM に関連する管理タスクについて説明しています。

CSL OM コマンドのルーティング

IMSplex 環境においては、OM に対して発行されたコマンドは、単一の IMS システムに対して発行されたコマンドと異なる動きをすることがあります。OM に対して発行されたコマンドは、デフォルト解釈では、IMSplex 内の、アクティブであって、そのコマンドを処理するように登録されている全 IMS システムに経路指定されます。

IMSplex 内の 1 つ以上の特定のコマンド処理クライアントにコマンドを経路指定したい場合、CSL OM 要求の ROUTE パラメーターを使用するか、CSL OM API を使用してください。OM コマンドに ROUTE パラメーターがある場合、IMS は経路リスト中の最上レベルの IMS をコマンド・マスターとして選択します。例えば、IMS 15 CQS を持つ IMS 15 システム、および IMS 14 SCI を持つ別の IMS 14 システムを含む IMSplex 構成において、INIT OLC コマンド (ROUTE=ANY コマンド) が発行されると、IMS 15 システムがコマンド・マスターとして選択されます。

TSO SPOC を使用しても経路指定情報を指定することができます。ROUTE パラメーターの指定方法については、「IMS V15 システム・プログラミング API」を参照してください。TSO SPOC で提供されるオンライン・ヘルプも参照してください。

IMSplex 内では、コマンドは SPOC または 自動化操作プログラム (AOP) から発行され、OM に送られます。次に、そのコマンドは IMS システムに経路指定されます。各 IMS システムはそのコマンド出力を戻します。次に、OM は、個々の応答を統合し、統合された出力を、要求を発信したクライアントに XML タグでカプセル化して送り返します。以下の図は、このルーティングを示しています。OS1 には、SCI、SPOC TSO/ISPF アプリケーション、および自動化プログラムがあります。OS2 には、SCI 付きの OM と、SCI 付きの IMS 制御領域があります。OS3 には、それぞれが SCI 付きの 2 つの IMS 制御領域があります。コマンドは、OS1 の SCI から OS2 の OM に経路指定されます。その OM に関連付けられている SCI は、そのコマンドを他の IMS システムに、他のシステムの SCI を介して、経路指定されます。

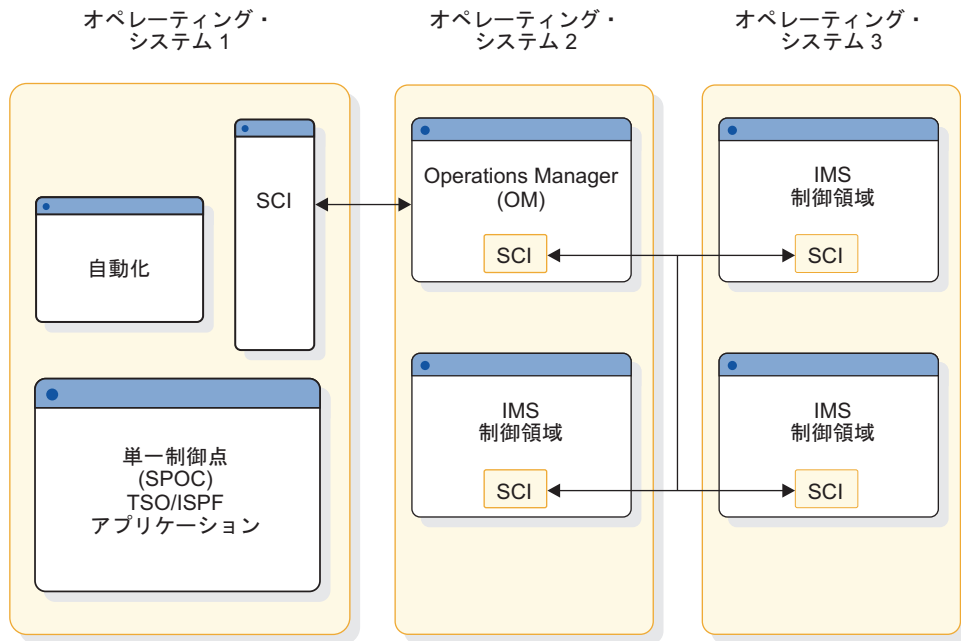


図 27. CSL が組み込まれた IMSplex 内のコマンド経路指定

コマンドが OM に発行されると、コマンドの応答は XML タグでカプセル化されます。

IMS コマンドならびにその応答については、「IMS V15 コマンド 第 1 巻: IMS コマンド A-M」および「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

関連概念:

- ➡ CSL OM XML 出力の解釈 (システム・プログラミング API)
- ➡ CSL OM 自動化操作プログラム・クライアント (システム・プログラミング API)

関連タスク:

- ➡ OM コマンド処理クライアントの登録 (システム・プログラミング API)

関連資料:

- ➡ CSL OM クライアントの作成 (システム・プログラミング API)

OM API を使用したコマンドの発行

MTO 端末やエンド・ユーザー端末などの IMS 端末から、タイプ 1 コマンドを入力できます。SPOC および OM API を通じて、多くのタイプ 1 コマンドとすべてのタイプ 2 コマンドを入力できます。OM API からコマンドを入力すると、OM は、そのコマンドを処理できるすべての IMS システムにコマンドを経路指定できます。OM は、IMS システムからのコマンド応答を、それらの受信時に整理して 1 つのバッファーに入れます。

コマンドの有効範囲がローカルのみ有的时候、コマンドが IMS マスター端末、IMS エンド・ユーザー端末、または IMS 自動化操作プログラム (AOP) から入力された

場合、そのコマンドは、コマンドが入力されたローカル IMS によってのみ処理されます。しかし、リソース構造が IMSplex の一部である場合、 /STOP USER や /STOP NODE などの一部のコマンドは、RM でグローバル状況になります。それらのコマンドを SPOC から入力すると、IMSplex 内のすべての IMS システムに経路指定され、グローバルな効果を持つ可能性があります。TSO SPOC によりコマンドを個々の IMS システムに経路指定できます。

CSL には、タイプ 2 コマンドの形式が用意されています。そのコマンド形式の利点を以下に示します。

- リソース名とコマンド・キーワードの矛盾の除去
- BPE パーサーを使用した優れたコマンド構文検査
- コマンド verb のセットの単純化
- 並列コマンド処理
- リソース名選択のためのフィルターおよびワイルドカード
- QUERY コマンド出力での選択出力フィールドのみのフィルター操作と表示
- グローバル情報へアクセスできるのは 1 つの IMS のみ

このコマンド形式を使用して入力するコマンドは、SPOC を介してのみ入力できます。ですから、このコマンド形式は、システム・コンソール、マスター端末、エンド・ユーザー端末、DL/I 呼び出し、APPC クライアント、あるいは OTMA インターフェースを使って入力しないでください。この形式は、単純化された BPE 構文解析規則に基づいています。これらのコマンドは、IMSplex が提供する、IMSplex 内のデータベースおよびキューを共用する IMS システムの単一システム・イメージと連動します。

タイプ 2 コマンドの形式は、次のとおりです。キーワードとパラメーターはオプションです。

```
ActionVerb ResourceType Keyword(parameter) Keyword(parameter) . . .
```

CSL OM 監査証跡

OM 監査証跡には、OM API を介して処理されたコマンドの入出力が反映されます。ログに記録できるのは、コマンドの起点、処理されたコマンド、およびその処理の結果です。

OM はログ・レコードを z/OS システム・ロガーのログ・ストリームに書き込み、コマンド入力、関連コマンド出力、および非送信請求メッセージ出力の監査証跡を作成できます。OM 監査証跡機能を使用する前に、ログ・ストリームを作成する必要があります。z/OS システム・ロガーおよびカップリング・ファシリティ構造の定義が必要です。OM ログ・レコードを z/OS ログ・ストリームに書き込むには、CSLOIxxx PROCLIB メンバー上の IMSPLEX() キーワードの AUDITLOG= パラメーターを指定します。

以下のシナリオでは、OM が作業をログに記録するため、ログ・ストリームが使用可能になったときに OM 監査証跡内のデータが失われることはありません。

- ログ・ストリームへの接続が開始時に失敗した場合でも、初期設定は続行されます。ログ・ストリームが使用可能になった後に、接続がセットアップされます。

- OM のログ・ストリームへの書き込みが失敗すると、OM はログ・ストリームが使用可能になるのを待ち、もう一度書き込みプロセスを実行します。

個々のコマンドまたはコマンド応答に関連制御データを加えた長さが 32,760 バイトより長い場合、複数のログ・レコードにまたがって分割されます。こうしたログ・レコードには、ログ・レコードを単一のコマンドまたはコマンド応答に再組み立てするためにログ・フォーマット設定ユーティリティーが使用する制御データが含まれています。

関連概念:

➡ IVP によって検査されるその他のサンプル・アプリケーション (インストール)

➡ OM 監査証跡 (オペレーションおよびオートメーション)

関連タスク:

185 ページの『OM 監査ログ・レコード・フォーマット』

CSL OM 監査証跡のログ・ストリームの構成

IMS IVP では、ほとんどのユーザーに広く適用するように設定されたパラメーターを使用して、ログ・ストリームを定義しています。OM 監査証跡のログ・ストリームを定義する際に、ご使用のシステムの構成および要件に応じて、これらのパラメーターを変更することを検討してください。

変更を検討する IVP 設定のパラメーターには、以下が含まれます。

LS_SIZE(200)

LS_SIZE は、オフロード・データ・セットのサイズを制御します。ご使用のシステムでは、データ・セットを大きくすると利点が得られる場合があります。

LOWOFFLOAD(20)

LOWOFFLOAD は、オフロードの頻度に影響を与えます。LOWOFFLOAD のパーセンテージを低くする (例えば、0) ことは、一般的にオフロードの発生頻度が少なくなることを意味し、ログ・ストリームをリアルタイムで表示するのではない限り、最も効率的であると考えられます。

HIGHOFFLOAD(50)

HIGHOFFLOAD は、オフロードの頻度に影響を与えます。

HIGHOFFLOAD 値を高くする (例えば、70) ことは、一般的にオフロードの発生頻度が少なくなることを意味し、効率が向上します。

また、IMS IVP では、システム・ロガーに対してステージング・データ・セットへの二重化も定義しません。ご使用のシステムが、MVS データ・スペースの二重化で提供されるよりも高水準のデータの冗長性を必要とする場合は、二重化が行われる条件を定義する DUPLEXMODE パラメーターとともに、下記の定義に STG_DUPLEX(YES) を追加して、外部二重化を使用可能にすることを検討してください。ただし、高水準のデータの冗長性は、二重化オプションのパフォーマンスへの影響と対比して検討する必要があります。

IMS IVP では、以下のパラメーターを使用して、ログ・ストリームを定義します。

```

DEFINE STRUCTURE NAME(IMSOM2Q01)
    LOGSNUM(1)
    AVGBUFSIZE(4000)
    MAXBUFSIZE(32760)

DEFINE LOGSTREAM NAME(SYSLOG.OM2Q01.LOG)
    STRUCTNAME(IMSOM2Q01)
    HLQ(IXGLOGR)
    LS_STORCLAS(LOGGER1)
    LS_DATACLAS(LOGGER1)
    LS_MGMTCLAS(LOGGER)
    LS_SIZE(200)
    LOWOFFLOAD(20)
    HIGHOFFLOAD(50)

```

推奨事項:

- ログ・ストリーム最大バッファ・サイズ (MAXBUFSIZE) を 32,760 バイトに設定します。この設定により、すべてのコマンド・データおよびコマンド応答データが確実に監査証跡に書き込まれるようになります。設定を小さくすると、いくつかの OM 監査証跡ログ・レコードが切り捨てられます。
- ログ・ストリームを OM 監査証跡専用にします。

関連概念:

- ➡ IBM Redbooks: システム・ロガー
- ➡ z/OS: システム・ロガー・サービスの使用

関連タスク:

- ➡ z/OS: システム・ロガー・アプリケーションの計画

関連資料:

- ➡ IMS PROCLIB データ・セットの CSLOIxxx メンバー (システム定義)

OM 監査ログ・レコード・フォーマット

各 Operations Manager (OM) 監査ログ・レコードには、ログ・レコード接頭部が含まれ、その後、そのレコード固有のデータが続きます。

ログ・レコード接頭部をマップするには、マクロ CSLZLGPf を使用します。OM 監査ログ・レコード・フォーマットを表示するには、マッピング・マクロ CSLOLGRC をアセンブルします。

個々のログ・レコードの DSECT は、希望するログ・レコードのフォーマット指定ステートメントを含めて ILOGREG マクロをアセンブルすることで取得できます。

以下の表は、OM 監査ログ・レコードを示しています。表には、OM 監査ログ・レコードごとに、以下の内容が示されています。

- ログ・レコードのタイプおよびサブタイプ
- レコードをマップするマクロ
- レコードが書き込まれる原因となるイベント

表 11. OM 監査ログ・レコード

タイプ	サブタイプ	マッピング・マクロ	ログ・レコードが書き込まれる原因となるイベント
X'06'	X'01'	CSLOLGCM	OM 入力ユーザー出口ルーチン呼び出す前に、コマンド入力ログに記録されました。
X'06'	X'02'	CSLOLGCM	OM 入力ユーザー出口ルーチン呼び出した後で、そのユーザー出口ルーチンがコマンド入力を変更した後に、コマンド入力ログに記録されました。
X'08'	X'01'	CSLOLGCR	OM 出力ユーザー出口ルーチン呼び出す前に、コマンド応答出力ログに記録されました。
X'08'	X'02'	CSLOLGCR	OM 出力ユーザー出口ルーチン呼び出した後で、そのユーザー出口ルーチンがコマンド応答出力を変更した後に、コマンド応答出力ログに記録されました。
X'09'	X'01'	CSLOLGOU	OM 出力ユーザー出口ルーチン呼び出す前に、非送信請求出力メッセージがログに記録されました。
X'09'	X'02'	CSLOLGOU	OM 出力ユーザー出口ルーチン呼び出した後で、そのユーザー出口ルーチンが非送信請求出力メッセージを変更した後に、非送信請求出力メッセージがログに記録されました。

OM ログ・レコードの印刷

z/OS システム・ログから OM ログ・レコードを印刷するには、IMS ファイル選択・フォーマット設定印刷ユーティリティー (DFSERA10) と出口ルーチン CSLOERA3 を使用します。

次の図の例は、z/OS システム・ログからログ・レコードを印刷するために必要な JCL を示しています。

z/OS システム・ログから OM ログ・レコードを印刷する JCL

```
//CSLERA1 JOB MSGLEVEL=1,MSGCLASS=A,CLASS=K
//STEP1 EXEC PGM=DFSERA10
//STEPLIB DD DISP=SHR,DSN=IMS.SDFSRESL
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SYSLOG.OM.AUDIT.TRAIL.LOG,
// SUBSYS=(LOGR,IXGSEXIT),
// DCB=(BLKSIZE=32760)
//SYSIN DD
CONTROL CNTL H=EOF
OPTION PRINT EXITR=CSLULALE
END
//
```

32 760 バイトより長いコマンド入力およびコマンド応答のログ・レコードは、複数のセグメントに分割されます。非送信請求出力は、最大長が 320 バイトです。各セグメントは、OM からの個別の IXGWRITE 呼び出しで、MVS ロガーに送信されます。そのため、長いログ・レコードは、ログ・ストリーム上に複数のレコードとして現れます。CSLOERA3 および CSLULALE フォーマット設定ユーティリティーが、複数のセグメントに分離されたログ・レコードを再アSEMBルできるようにするために、コマンド応答およびコマンド入力のログ・レコードに制御データが追加されます。コマンド入力またはコマンド応答以外のログ・レコードを含めて、単一セグメントのログ・レコードを示すために、ログ・レコード接頭部にビットが追加されます。ログ・レコードは、ログ・レコードに示されたタイム・スタンプに基

づき、ログ・ストリームに送信されたのと同じ順序で印刷されます。MAXBUFSIZE が 32 760 バイトより低い値に設定されている場合、ログ・レコードは複数のセグメントに分割されません。代わりに、MAXBUFSIZE の長さを超えるログ・レコードは切り捨てられます。

推奨事項: MAXBUFSIZE は 32760 バイトに設定してください。65,535 セグメントを超えるログ・レコードは引き続き切り捨てられます。現在のところ、この制限は、2GB より長いログ・レコードにのみ影響を与えます。

OM ログ・レコードを印刷するための DD ステートメントは、次のとおりです。

STEPLIB

DSN= は、IMS.SDFSRESL を指します。これには、IMS ファイル選択・フォーマット設定印刷ユーティリティ DFSERA10 が入っています。

SYSUT1

DSN= は、CSLOIxxx PROCLIB メンバーの AUDITLOG= パラメーターで指定された OM ログ・ストリーム名を指します。

OM ログ・レコードを印刷するための制御ステートメントは、次のとおりです。

H= 印刷するログ・レコードの数を指定します。すべてのログ・レコードを印刷するには、H=EOF を指定します。

EXITR=CSLULALE

各ログ・レコードをフォーマット設定するために呼び出される OM ログ・レコード出口ルーチンを識別します。出口ルーチン CSLULALE は、OM 監査証跡を syslog に似た形式でフォーマットします。出口ルーチン CSLOERA3 も使用できます。出口ルーチン CSLOERA3 は、レコード・タイプと各レコードのタイム・スタンプ情報を含めて、レコードをダンプ形式で印刷し、最大 32 760 バイトのレコードの内容を提供します。

ログ・データを特定の時刻範囲に制限するには、下の図に示すように、SUBSYS ステートメントで FROM パラメーターと TO パラメーターを使用します。この DD カードは、2007 年の 42 日目の 11:00 から 12:00 までのログ・レコードを印刷します。

ログ・データを特定の時刻範囲に制限する **JCL**


```
//SYSUT1 DD DSN=SYSLOG.OM.AUDIT.TRAIL.LOG,
//          SUBSYS=(LOGR,IXGSEXIT,
//          'FROM=(2007/042,11:00:00),TO=(2007/042,12:00:00)'),
//          DCB=(BLKSIZE=32760)
```

日時は、グリニッジ標準時 (GMT) で指定します。この時刻値の秒数のフィールドはオプションです。ローカルの日時を使用するには、次のテーブルに示すように、LOCAL キーワードを追加します。

ローカルの日時を使用して、ログ・データを特定の時刻範囲に制限する **JCL**

```
//SYSUT1 DD DSN=SYSLOG.OM.AUDIT.TRAIL.LOG,
//          SUBSYS=(LOGR,IXGSEXIT,
//          'FROM=(2007/042,11:00:00),TO=(2007/042,12:00:00),LOCAL'),
//          DCB=(BLKSIZE=32760)
```

関連資料:

 ファイル選択およびフォーマット設定印刷ユーティリティー (DFSERA10) (システム・ユーティリティー)

CSL OM コマンド・セキュリティ

OM コマンド・セキュリティはコマンド処理時にオプションで実行されます。

コマンド・セキュリティによって、以下が可能になります。

- どのユーザー ID が OM を介して IMS コマンドを入力できるかを制御するユーザー。
- ユーザー ID をアプリケーション・プログラム・アドレス・スペースに関連付ける。
- ユーザー ID を、TSO SPOC にログオンしたエンド・ユーザーにする。

CMDSEC= パラメーターは、OM 始動プロシージャ (CSLOM)、OM 初期設定 PROCLIB メンバー・データ・セット (CSLOIxxx)、および DFSCGxxx PROCLIB メンバー・データ・セットで使用可能です。OM 始動プロシージャの一部としてこのパラメーターが出されると、それはすべての IMS コマンド、タイプ-1 およびタイプ-2、に適用されます。これが DFSCGxxx PROCLIB メンバー・データ・セットを使って出された場合は、OM を介して入力されたタイプ-1 コマンドにのみ適用されます。OM と IMS のセキュリティの相違点を、以下の表に示します。

表 12. OM と IMS セキュリティの比較

セキュリティ方式	N	A	E	R
OM 実行パラメーター (CSLOM および CSLOIxxx)	許可検査は実行されません。これはデフォルトです。	コマンド許可検査のために RACF と CSL OM セキュリティ・ユーザー・ユーザー・出口ルーチンの両方を呼び出します。	コマンド許可検査のために CSL OM セキュリティ・ユーザー・ユーザー・出口ルーチンを呼び出します。	コマンド許可検査のために、RACF を呼び出します。コマンドは、OPERCMDS リソース・クラスの一部です。
DFSCGxxx PROCLIB メンバー・データ・セット	許可検査は実行されません。これはデフォルトです。OM がコマンド許可検査を行うこともできます。	コマンド許可検査のために、RACF および IMS コマンド許可出口ルーチン (DFSCCMD0) の双方を呼び出します。	コマンド許可検査のために、IMS コマンド許可出口ルーチン (DFSCCMD0) を呼び出します。	コマンド許可検査のために、RACF を呼び出します。コマンドは、CIMS リソース・クラスの一部です。

推奨事項: IMS コマンド・セキュリティではなく、OM コマンド・セキュリティを使用してください。

OM API でサポートされる全コマンドに関する RACF アクセス権限 (READ または UPDATE) およびリソース名については、190 ページの『IMS コマンド、RACF アクセス権限およびリソース名の表』に説明があります。RACF 権限は、そのコマンドが登録されたときに指定されたアクセス権限を示します。

コマンドは、CSLOMBLD 要求を用いて OM に登録されます。RACF PERMIT コマンドに対するアクセス権限は、コマンドが登録されたときに指定されたアクセス権限と一致しなければなりません。CSLOMBLD によるコマンドの登録について詳しくは、「IMS V15 システム・プログラミング API」を参照してください。

OM を介して入力されるコマンドに対する RACF 許可

すべての OM セキュリティ検査で、RACF OPERCMDS クラスを使用します。リソース名は、IMS.imsplexname.commandverb.commandkeyword の形式です。

IMS

OM を通じて入力される IMS コマンドを保護するすべての RACF リソースの高位修飾子。

imsplexname

コマンドの許可が与えられる IMSplex の名前。IMSplex 名は、先頭に、文字「CSL」が入っている必要があります。例えば、CSLplex1 です。

commandverb

コマンド verb の名前。

commandkeyword

基本コマンド・キーワードまたはそのコマンドのリソース・タイプ。

例: IMSplex でのコマンドに対するユーザーの許可

以下の例は、さまざまなユーザーを異なるコマンドに対して許可する RACF 定義の例を示しています。

```
/* allow UPD TRAN command on any IMSplex for user ID Jim */
RDEFINE OPERCMDS IMS.*.UPD.TRAN UACC(NONE)
PERMIT IMS.*.UPD.TRAN CLASS(OPERCMDS) ID(JIM) ACCESS(UPDATE)

/* allow INIT OLC command on IMSplex CSLPLEXA for user ID Sandy */
RDEFINE OPERCMDS IMS.CSLPLEXA.INIT.OLC UACC(NONE)
PERMIT IMS.CSLPLEXA.INIT.OLC CLASS(OPERCMDS) ID(SANDY) ACCESS(UPDATE)

/* allow any QRY command on any IMSplex for user ID Tom */
RDEFINE OPERCMDS IMS.*.QRY.* UACC(NONE)
PERMIT IMS.*.QRY.* CLASS(OPERCMDS) ID(TOM) ACCESS(READ)

/* allow any commands on any IMSplex for user ID Betty */
RDEFINE OPERCMDS IMS.* UACC(NONE)
PERMIT IMS.* CLASS(OPERCMDS) ID(BETTY) ACCESS(UPDATE)

SETROPTS CLASSACT(OPERCMDS)
SETROPTS RACLIST(OPERCMDS) REFRESH
```

例: RACF グループの使用と、タイプ 2 コマンド・セキュリティのための OPERCMDS クラス

以下の例は、RACF グループの一部であるすべてのユーザー ID があらゆる IMS オペレーター・コマンドを使用することを許可する RACF 定義を示しています。この例には、RACF 管理の 2 つの重要なエレメントが示されています。

- RACF グループ
- ワイルドカード文字

```

/* define IMS operator group */
ADDGROUP IMSOPER
CONNECT (PEDRO,ROSA,PETER,MATT) GROUP(IMSOPER)

/* allow any commands on any IMSplex for group IMSOPER */
RDEFINE OPERCMDS IMS.* UACC(NONE)
PERMIT IMS.* CLASS(OPERCMDS) ID(IMSOPER) ACCESS(UPDATE)

SETROPTS CLASSACT(OPERCMDS)
SETROPTS RACLIST(OPERCMDS) REFRESH

```

グループ定義を使用すると、RACF プロファイルの管理が単純化されます。そのグループを、データ・セットや OPERCMDS プロファイルなど、多数の RACF プロファイルに対して許可できます。ユーザーがジョブを変更した場合、そのユーザーの、各 RACF リソース・プロファイルに対するアクセスを削除する代わりに、そのユーザー ID をグループ定義から除去するだけで済みます。

ワイルドカード文字を RACF リソース名で使用する利点は、RACF リソース定義の数を最小に保持できるということです。また、OPERCMDS プロファイルの作成後に定義された一致オペレーター・コマンドは、指定されていないユーザーから引き続き保護されます。

例: RACF APPL クラスに関連付けられているユーザーの許可

以下の例は、RACF APPL クラスに関連付けられているユーザーを許可するプロファイルを追加するための RACF 定義を示しています。

```

/* add a profile */
RDEFINE APPL JOBNAME UACC(NONE)
PERMIT JOBNAME CLASS(APPL) ID(USERID OR GROUPNAME) ACCESS(UPDATE)

```

IMS コマンド、RACF アクセス権限およびリソース名の表

以下の表は、IMS タイプ 1 およびタイプ 2 コマンド・セキュリティの OM コマンド・セキュリティに適用されます。この表は、IMS コマンド verb 別およびキーワード別に、RACF セキュリティ検査に使用するリソース名と許可を示しています。

例えば、PLX01 という名前の IMSplex に対する ACTIVATE NODE コマンドを出したいとすれば、ユーザー ID は、OPERCMDS クラス内のプロファイル IMS.CSLPLX01.ACT.NODE に対する UPDATE 権限をもっていなければなりません。READ 権限しかもっていなかった場合には、ACTIVATE NODE コマンドは失敗し、「権限不足」を示すメッセージが出されます。

通常、ユーザーは、READ 権限をもつ表示または照会のコマンドを出すことができます。リソースの状態を変更するには、UPDATE 権限が必要です。

注: リソース名の中でワイルドカードを使用することにより、必要な RACF リソースの数を減らすことができます。例えば、すべての表示コマンドを使用する権限を付与するには、IMS.CSLPLX01.DIS.* を使用します。

表 13. IMS コマンドのリソース名および RACF 権限

コマンド verb	コマンド・キーワード	権限	リソース名
ACT	LINK	UPDATE	IMS.plxname.ACT.LINK

表 13. IMS コマンドのリソース名および RACF 権限 (続き)

コマンド verb	コマンド・キーワード	権限	リソース名
ACT	NODE	UPDATE	IMS.plxname.ACT.NODE
ALL	LU	UPDATE	IMS.plxname.ALL.LU
ASS	CLASS	UPDATE	IMS.plxname.ASS.CLASS
ASS	CPRI	UPDATE	IMS.plxname.ASS.CPRI
ASS	INPUT	UPDATE	IMS.plxname.ASS.INPUT
ASS	LCT	UPDATE	IMS.plxname.ASS.LCT
ASS	LPRI	UPDATE	IMS.plxname.ASS.LPRI
ASS	LTERM	UPDATE	IMS.plxname.ASS.LTERM
ASS	NPRI	UPDATE	IMS.plxname.ASS.NPRI
ASS	OUTPUT	UPDATE	IMS.plxname.ASS.OUTPUT
ASS	PARLIM	UPDATE	IMS.plxname.ASS.PARLIM
ASS	PLCT	UPDATE	IMS.plxname.ASS.PLCT
ASS	SEGNO	UPDATE	IMS.plxname.ASS.SEGNO
ASS	SEGSZ	UPDATE	IMS.plxname.ASS.SEGSZ
ASS	TRAN	UPDATE	IMS.plxname.ASS.TRAN
ASS	USER	UPDATE	IMS.plxname.ASS.USER
BRO	ACT	READ	IMS.plxname.BRO.ACT
BRO	LINE	READ	IMS.plxname.BRO.LINE
BRO	LTERM	READ	IMS.plxname.BRO.LTERM
BRO	MASTER	READ	IMS.plxname.BRO.MASTER
BRO	NODE	READ	IMS.plxname.BRO.NODE
BRO	PTERM	READ	IMS.plxname.BRO.PTERM
BRO	USER	READ	IMS.plxname.BRO.USER
CHA	APPC	UPDATE	IMS.plxname.CHA.APPC
CHA	CCTL	UPDATE	IMS.plxname.CHA.CCTL
CHA	CPLOG	UPDATE	IMS.plxname.CHA.CPLOG
CHA	DESC	UPDATE	IMS.plxname.CHA.DESC
CHA	DIR	UPDATE	IMS.plxname.CHA.DIR
CHA	FDR	UPDATE	IMS.plxname.CHA.FDR
CHA	LINK	UPDATE	IMS.plxname.CHA.LINK
CHA	NODE	UPDATE	IMS.plxname.CHA.NODE
CHA	PSWD	UPDATE	IMS.plxname.CHA.PSWD
CHA	SUBSYS	UPDATE	IMS.plxname.CHA.SUBSYS
CHA	SURV	UPDATE	IMS.plxname.CHA.SURV
CHA	TRAN	UPDATE	IMS.plxname.CHA.TRAN
CHA	UOR	UPDATE	IMS.plxname.CHA.UOR
CHA	USER	UPDATE	IMS.plxname.CHA.USER
CHE		UPDATE	IMS.plxname.CHE
CHE	DUMPQ	UPDATE	IMS.plxname.CHE.DUMPQ
CHE	FREEZE	UPDATE	IMS.plxname.CHE.FREEZE

表 13. IMS コマンドのリソース名および RACF 権限 (続き)

コマンド verb	コマンド・キーワード	権限	リソース名
CHE	PURGE	UPDATE	IMS.plxname.CHE.PURGE
CHE	STATISTICS	UPDATE	IMS.plxname.CHE.STATISTICS
CLS	NODE	UPDATE	IMS.plxname.CLS.NODE
CQC	SHRQ	UPDATE	IMS.plxname.CQC.SHRQ
CQC	SYSTEM	UPDATE	IMS.plxname.CQC.SYSTEM
CQQ	STATISTICS	READ	IMS.plxname.CQQ.STATISTICS
CQS	SHUTDOWN	UPDATE	IMS.plxname.CQS.SHUTDOWN
CRE	DB	UPDATE	IMS.plxname.CRE.DB
CRE	DBDESC	UPDATE	IMS.plxname.CRE.DBDESC
CRE	IMSCON	UPDATE	IMS.plxname.CRE.IMSCON
CRE	LTERM	UPDATE	IMS.plxname.CRE.LTERM
CRE	MSLINK	UPDATE	IMS.plxname.CRE.MSLINK
CRE	MSNAME	UPDATE	IMS.plxname.CRE.MSNAME
CRE	MSPLINK	UPDATE	IMS.plxname.CRE.MSPLINK
CRE	PGM	UPDATE	IMS.plxname.CRE.PGM
CRE	PGMDESC	UPDATE	IMS.plxname.CRE.PGMDESC
CRE	RTC	UPDATE	IMS.plxname.CRE.RTC
CRE	RTCDESC	UPDATE	IMS.plxname.CRE.RTCDESC
CRE	TRAN	UPDATE	IMS.plxname.CRE.TRAN
CRE	TRANDESC	UPDATE	IMS.plxname.CRE.TRANDESC
DBD	DB	UPDATE	IMS.plxname.DBD.DB
DBR	AREA	UPDATE	IMS.plxname.DBR.AREA
DBR	DB	UPDATE	IMS.plxname.DBR.DB
DBR	DATAGRP	UPDATE	IMS.plxname.DBR.DATAGRP
DEL	DB	UPDATE	IMS.plxname.DEL.DB
DEL	DBDESC	UPDATE	IMS.plxname.DEL.DBDESC
DEL	DEFN	UPDATE	IMS.plxname.DEL.DEFN
DEL	IMSCON	UPDATE	IMS.plxname.DEL.IMSCON
DEL	LE	UPDATE	IMS.plxname.DEL.LE
DEL	LTERM	UPDATE	IMS.plxname.DEL.LTERM
DEL	MSLINK	UPDATE	IMS.plxname.DEL.MSLINK
DEL	MSNAME	UPDATE	IMS.plxname.DEL.MSNAME
DEL	MSPLINK	UPDATE	IMS.plxname.DEL.MSPLINK
DEL	PGM	UPDATE	IMS.plxname.DEL.PGM
DEL	PGMDESC	UPDATE	IMS.plxname.DEL.PGMDESC
DEL	PSWD	UPDATE	IMS.plxname.DEL.PSWD
DEL	RTC	UPDATE	IMS.plxname.DEL.RTC
DEL	RTCDESC	UPDATE	IMS.plxname.DEL.RTCDESC
DEL	TERMINAL	UPDATE	IMS.plxname.DEL.TERMINAL
DEL	TRAN	UPDATE	IMS.plxname.DEL.TRAN

表 13. IMS コマンドのリソース名および RACF 権限 (続き)

コマンド verb	コマンド・キーワード	権限	リソース名
DEL	TRANDESC	UPDATE	IMS.plxname.DEL.TRANDESC
DEQ	AOITKN	UPDATE	IMS.plxname.DEQ.AOITKN
DEQ	LINE	UPDATE	IMS.plxname.DEQ.LINE
DEQ	LTERM	UPDATE	IMS.plxname.DEQ.LTERM
DEQ	LU	UPDATE	IMS.plxname.DEQ.LU
DEQ	MSNAME	UPDATE	IMS.plxname.DEQ.MSNAME
DEQ	NODE	UPDATE	IMS.plxname.DEQ.NODE
DEQ	SUSPEND	UPDATE	IMS.plxname.DEQ.SUSPEND
DEQ	TMEM	UPDATE	IMS.plxname.DEQ.TMEM
DEQ	TRAN	UPDATE	IMS.plxname.DEQ.TRAN
DEQ	USER	UPDATE	IMS.plxname.DEQ.USER
DIAG	ADDRESS	UPDATE	IMS.plxname.DIAG.ADDRESS
DIAG	BLOCK	UPDATE	IMS.plxname.DIAG.BLOCK
DIAG	LTERM	UPDATE	IMS.plxname.DIAG.LTERM
DIAG	NODE	UPDATE	IMS.plxname.DIAG.NODE
DIAG	SNAP	UPDATE	IMS.plxname.DIAG.SNAP
DIAG	TRAN	UPDATE	IMS.plxname.DIAG.TRAN
DIAG	USER	UPDATE	IMS.plxname.DIAG.USER
DIS	ACT	READ	IMS.plxname.DIS.ACT
DIS	AFFIN	READ	IMS.plxname.DIS.AFFIN
DIS	AOITKN	READ	IMS.plxname.DIS.AOITKN
DIS	APPC	READ	IMS.plxname.DIS.APPC
DIS	AREA	READ	IMS.plxname.DIS.AREA
DIS	ASMT	READ	IMS.plxname.DIS.ASMT
DIS	CCTL	READ	IMS.plxname.DIS.CCTL
DIS	CONV	READ	IMS.plxname.DIS.CONV
DIS	CPLOG	READ	IMS.plxname.DIS.CPLOG
DIS	CQS	READ	IMS.plxname.DIS.CQS
DIS	DB	READ	IMS.plxname.DIS.DB
DIS	DBD	READ	IMS.plxname.DIS.DBD
DIS	DESC	READ	IMS.plxname.DIS.DESC
DIS	FDR	READ	IMS.plxname.DIS.FDR
DIS	FPV	READ	IMS.plxname.DIS.FPV
DIS	HSB	READ	IMS.plxname.DIS.HSB
DIS	HSSP	READ	IMS.plxname.DIS.HSSP
DIS	LINE	READ	IMS.plxname.DIS.LINE
DIS	LINK	READ	IMS.plxname.DIS.LINK
DIS	LTERM	READ	IMS.plxname.DIS.LTERM
DIS	LU	READ	IMS.plxname.DIS.LU
DIS	MASTER	READ	IMS.plxname.DIS.MASTER

表 13. IMS コマンドのリソース名および RACF 権限 (続き)

コマンド verb	コマンド・キーワード	権限	リソース名
DIS	MODIFY	READ	IMS.plxname.DIS.MODIFY
DIS	MSNAME	READ	IMS.plxname.DIS.MSNAME
DIS	NODE	READ	IMS.plxname.DIS.NODE
DIS	OASN	READ	IMS.plxname.DIS.OASN
DIS	OLDS	READ	IMS.plxname.DIS.OLDS
DIS	OTMA	READ	IMS.plxname.DIS.OTMA
DIS	OVERFLOWQ	READ	IMS.plxname.DIS.OVERFLOWQ
DIS	POOL	READ	IMS.plxname.DIS.POOL
DIS	PGM	READ	IMS.plxname.DIS.PGM
DIS	PSB	READ	IMS.plxname.DIS.PSB
DIS	PTERM	READ	IMS.plxname.DIS.PTERM
DIS	Q	READ	IMS.plxname.DIS.Q
DIS	QCNT	READ	IMS.plxname.DIS.QCNT
DIS	RECOVERY	READ	IMS.plxname.DIS.RECOVERY
DIS	RTC	READ	IMS.plxname.DIS.RTC
DIS	SHUTDOWN	READ	IMS.plxname.DIS.SHUTDOWN
DIS	STATUS	READ	IMS.plxname.DIS.STATUS
DIS	STRUC	READ	IMS.plxname.DIS.STRUC
DIS	SUBSYS	READ	IMS.plxname.DIS.SUBSYS
DIS	SYSID	READ	IMS.plxname.DIS.SYSID
DIS	TIMEOVER	READ	IMS.plxname.DIS.TIMEOVER
DIS	TMEM	READ	IMS.plxname.DIS.TMEM
DIS	TRACE	READ	IMS.plxname.DIS.TRACE
DIS	TRACKING	READ	IMS.plxname.DIS.TRACKING
DIS	TRAN	READ	IMS.plxname.DIS.TRAN
DIS	UOR	READ	IMS.plxname.DIS.UOR
DIS	USER	READ	IMS.plxname.DIS.USER
END	LINE	UPDATE	IMS.plxname.END.LINE
END	NODE	UPDATE	IMS.plxname.END.NODE
END	USER	UPDATE	IMS.plxname.END.USER
ERE		UPDATE	IMS.plxname.ERE
ERE	BACKUP	UPDATE	IMS.plxname.ERE.BACKUP
ERE	COLDBASE	UPDATE	IMS.plxname.ERE.COLDBASE
ERE	COLDCOMM	UPDATE	IMS.plxname.ERE.COLDCOMM
ERE	COLDSYS	UPDATE	IMS.plxname.ERE.COLDSYS
EXC	LINE	UPDATE	IMS.plxname.EXC.LINE
EXC	NODE	UPDATE	IMS.plxname.EXC.NODE
EXC	USER	UPDATE	IMS.plxname.EXC.USER
EXI	CONV	UPDATE	IMS.plxname.EXI.CONV
EXP	DEFN	UPDATE	IMS.plxname.EXP.DEFN

表 13. IMS コマンドのリソース名および RACF 権限 (続き)

コマンド verb	コマンド・キーワード	権限	リソース名
IDL	LINE	UPDATE	IMS.plxname.IDL.LINE
IDL	LINK	UPDATE	IMS.plxname.IDL.LINK
IDL	NODE	UPDATE	IMS.plxname.IDL.NODE
IMP	DEFN	UPDATE	IMS.plxname.IMP.DEFN
INIT	OLC	UPDATE	IMS.plxname.INIT.OLC
INIT	OLREORG	UPDATE	IMS.plxname.INIT.OLREORG
LOC	DB	UPDATE	IMS.plxname.LOC.DB
LOC	PGM	UPDATE	IMS.plxname.LOC.PGM
LOC	TRAN	UPDATE	IMS.plxname.LOC.TRAN
LOG		UPDATE	IMS.plxname.LOG
MOD	ABORT	UPDATE	IMS.plxname.MOD.ABORT
MOD	COMMIT	UPDATE	IMS.plxname.MOD.COMMIT
MOD	PREPARE	UPDATE	IMS.plxname.MOD.PREPARE
MON	LINE	UPDATE	IMS.plxname.MON.LINE
MSA	LINK	UPDATE	IMS.plxname.MSA.LINK
MSA	MSNAME	UPDATE	IMS.plxname.MSA.MSNAME
MSA	SYSID	UPDATE	IMS.plxname.MSA.SYSID
MSA	TRAN	UPDATE	IMS.plxname.MSA.TRAN
MSV	MSNAME	UPDATE	IMS.plxname.MSV.MSNAME
MSV	SYSID	UPDATE	IMS.plxname.MSV.SYSID
NRE		UPDATE	IMS.plxname.NRE
NRE	CHKPT	UPDATE	IMS.plxname.NRE.CHKPT
OPN	NODE	UPDATE	IMS.plxname.OPN.NODE
PST	LINE	UPDATE	IMS.plxname.PST.LINE
PST	LINK	UPDATE	IMS.plxname.PST.LINK
PST	LTERM	UPDATE	IMS.plxname.PST.LTERM
PST	MSPLINK	UPDATE	IMS.plxname.PST.MSPLINK
PST	REGION	UPDATE	IMS.plxname.PST.REGION
PST	TRAN	UPDATE	IMS.plxname.PST.TRAN
PUR	APPC	UPDATE	IMS.plxname.PUR.APPC
PUR	FPPROG	UPDATE	IMS.plxname.PUR.FPPROG
PUR	FPRGN	UPDATE	IMS.plxname.PUR.FPRGN
PUR	LINE	UPDATE	IMS.plxname.PUR.LINE
PUR	LTERM	UPDATE	IMS.plxname.PUR.LTERM
PUR	MSNAME	UPDATE	IMS.plxname.PUR.MSNAME
PUR	TRAN	UPDATE	IMS.plxname.PUR.TRAN
QRY	AREA	READ	IMS.plxname.QRY.AREA
QRY	DB	READ	IMS.plxname.QRY.DB
QRY	DBDESC	READ	IMS.plxname.QRY.DBDESC
QRY	IMS	READ	IMS.plxname.QRY.IMS

表 13. IMS コマンドのリソース名および RACF 権限 (続き)

コマンド verb	コマンド・キーワード	権限	リソース名
QRY	IMSCON	READ	IMS.plxname.QRY.IMSCON
QRY	IMSPLEX	READ	IMS.plxname.QRY.IMSPLEX
QRY	LE	READ	IMS.plxname.QRY.LE
QRY	LTERM	READ	IMS.plxname.QRY.LTERM
QRY	MEMBER	READ	IMS.plxname.QRY.MEMBER
QRY	MSLINK	READ	IMS.plxname.QRY.MSLINK
QRY	MSNAME	READ	IMS.plxname.QRY.MSNAME
QRY	MSLINK	READ	IMS.plxname.QRY.MSPLINK
QRY	NODE	READ	IMS.plxname.QRY.NODE
QRY	OLC	READ	IMS.plxname.QRY.OLC
QRY	OLREORG	READ	IMS.plxname.QRY.OLREORG
QRY	PGM	READ	IMS.plxname.QRY.PGM
QRY	PGMDESC	READ	IMS.plxname.QRY.PGMDESC
QRY	POOL	READ	IMS.plxname.QRY.POOL
QRY	RM	READ	IMS.plxname.QRY.RM
QRY	RTC	READ	IMS.plxname.QRY.RTC
QRY	RTCDESC	READ	IMS.plxname.QRY.RTCDESC
QRY	STRUCTURE	READ	IMS.plxname.QRY.STRUCTURE
QRY	TRACE	READ	IMS.plxname.QRY.TRACE
QRY	TRAN	READ	IMS.plxname.QRY.TRAN
QRY	TRANDESC	READ	IMS.plxname.QRY.TRANDESC
QRY	USER	READ	IMS.plxname.QRY.USER
QRY	USEREXIT	READ	IMS.plxname.QRY.USEREXIT
QRY	USERID	READ	IMS.plxname.QRY.USERID
QUE	LTERM	UPDATE	IMS.plxname.QUE.LTERM
QUE	TRAN	UPDATE	IMS.plxname.QUE.TRAN
QUI	NODE	UPDATE	IMS.plxname.QUI.NODE
RDI	MASTER	READ	IMS.plxname.RDI.MASTER
REC	ADD	UPDATE	IMS.plxname.REC.ADD
REC	REMOVE	UPDATE	IMS.plxname.REC.REMOVE
REC	START	UPDATE	IMS.plxname.REC.START
REC	STOP	UPDATE	IMS.plxname.REC.STOP
REC	TERMINATE	UPDATE	IMS.plxname.REC.TERMINATE
REFRESH	USEREXIT	UPDATE	IMS.plxname.REFR.USEREXIT
RMC		UPDATE	IMS.plxname.RMC
RMD		UPDATE	IMS.plxname.RMD
RMG		UPDATE	IMS.plxname.RMG
RMI		UPDATE	IMS.plxname.RMI
RML		READ	IMS.plxname.RML
RMN		UPDATE	IMS.plxname.RMN

表 13. IMS コマンドのリソース名および RACF 権限 (続き)

コマンド verb	コマンド・キーワード	権限	リソース名
RST	LINE	UPDATE	IMS.plxname.RST.LINE
RST	LINK	UPDATE	IMS.plxname.RST.LINK
RST	MSPLINK	UPDATE	IMS.plxname.RST.MSPLINK
RST	NODE	UPDATE	IMS.plxname.RST.NODE
RST	USER	UPDATE	IMS.plxname.RST.USER
RTA	DUMPQ	UPDATE	IMS.plxname.RTA.DUMPQ
RTA	FREEZE	UPDATE	IMS.plxname.RTA.FREEZE
RTA	UNPLAN	UPDATE	IMS.plxname.RTA.UNPLAN
SEC	APPC	UPDATE	IMS.plxname.SEC.APPC
SEC	OTMA	UPDATE	IMS.plxname.SEC.OTMA
SMC	MASTER	UPDATE	IMS.plxname.SMC.MASTER
SMC	TERMINAL	UPDATE	IMS.plxname.SMC.TERMINAL
STA	APPC	UPDATE	IMS.plxname.STA.APPC
STA	AREA	UPDATE	IMS.plxname.STA.AREA
STA	AUTOARCH	UPDATE	IMS.plxname.STA.AUTOARCH
STA	CLASS	UPDATE	IMS.plxname.STA.CLASS
STA	DB	UPDATE	IMS.plxname.STA.DB
STA	DATAGRP	UPDATE	IMS.plxname.STA.DATAGRP
STA	DC	UPDATE	IMS.plxname.STA.DC
STA	ISOLOG	UPDATE	IMS.plxname.STA.ISOLOG
STA	LINE	UPDATE	IMS.plxname.STA.LINE
STA	LTERM	UPDATE	IMS.plxname.STA.LTERM
STA	LU	UPDATE	IMS.plxname.STA.LU
STA	MADSIOT	UPDATE	IMS.plxname.STA.MADSIOT
STA	MSNAME	UPDATE	IMS.plxname.STA.MSNAME
STA	NODE	UPDATE	IMS.plxname.STA.NODE
STA	OLDS	UPDATE	IMS.plxname.STA.OLDS
STA	OTMA	UPDATE	IMS.plxname.STA.OTMA
STA	PGM	UPDATE	IMS.plxname.STA.PGM
STA	REGION	UPDATE	IMS.plxname.STA.REGION
STA	RTC	UPDATE	IMS.plxname.STA.RTC
STA	SB	UPDATE	IMS.plxname.STA.SB
STA	SERVGRP	UPDATE	IMS.plxname.STA.SERVGRP
STA	SUBSYS	UPDATE	IMS.plxname.STA.SUBSYS
STA	SURV	UPDATE	IMS.plxname.STA.SURV
STA	THREAD	UPDATE	IMS.plxname.STA.THREAD
STA	TMEM	UPDATE	IMS.plxname.STA.TMEM
STA	TRAN	UPDATE	IMS.plxname.STA.TRAN
STA	TRKARCH	UPDATE	IMS.plxname.STA.TRKARCH
STA	USER	UPDATE	IMS.plxname.STA.USER


表 13. IMS コマンドのリソース名および RACF 権限 (続き)

コマンド verb	コマンド・キーワード	権限	リソース名
STA	VGR	UPDATE	IMS.plxname.STA.VGR
STA	WADS	UPDATE	IMS.plxname.STA.WADS
STA	XRCTRACK	UPDATE	IMS.plxname.STA.XRCTRACK
STO	ADS	UPDATE	IMS.plxname.STO.ADS
STO	APPC	UPDATE	IMS.plxname.STO.APPC
STO	AREA	UPDATE	IMS.plxname.STO.AREA
STO	AUTOARCH	UPDATE	IMS.plxname.STO.AUTOARCH
STO	BACKUP	UPDATE	IMS.plxname.STO.BACKUP
STO	CLASS	UPDATE	IMS.plxname.STO.CLASS
STO	DB	UPDATE	IMS.plxname.STO.DB
STO	DATAGRP	UPDATE	IMS.plxname.STO.DATAGRP
STO	DC	UPDATE	IMS.plxname.STO.DC
STO	LINE	UPDATE	IMS.plxname.STO.LINE
STO	LTERM	UPDATE	IMS.plxname.STO.LTERM
STO	LU	UPDATE	IMS.plxname.STO.LU
STO	MADSIOT	UPDATE	IMS.plxname.STO.MADSIOT
STO	MSNAME	UPDATE	IMS.plxname.STO.MSNAME
STO	NODE	UPDATE	IMS.plxname.STO.NODE
STO	OLDS	UPDATE	IMS.plxname.STO.OLDS
STO	OTMA	UPDATE	IMS.plxname.STO.OTMA
STO	PGM	UPDATE	IMS.plxname.STO.PGM
STO	REGION	UPDATE	IMS.plxname.STO.REGION
STO	RTC	UPDATE	IMS.plxname.STO.RTC
STO	SB	UPDATE	IMS.plxname.STO.SB
STO	SERVGRP	UPDATE	IMS.plxname.STO.SERVGRP
STO	SUBSYS	UPDATE	IMS.plxname.STO.SUBSYS
STO	SURV	UPDATE	IMS.plxname.STO.SURV
STO	THREAD	UPDATE	IMS.plxname.STO.THREAD
STO	TMEM	UPDATE	IMS.plxname.STO.TMEM
STO	TRAN	UPDATE	IMS.plxname.STO.TRAN
STO	USER	UPDATE	IMS.plxname.STO.USER
STO	VGR	UPDATE	IMS.plxname.STO.VGR
STO	WADS	UPDATE	IMS.plxname.STO.WADS
STO	XRCTRACK	UPDATE	IMS.plxname.STO.XRCTRACK
SWI	OLDS	UPDATE	IMS.plxname.SWI.OLDS
SWI	SYSTEM	UPDATE	IMS.plxname.SWI.SYSTEM
SWI	WADS	UPDATE	IMS.plxname.SWI.WADS
TERM	OLC	UPDATE	IMS.plxname.TERM.OLC
TERM	OLREORG	UPDATE	IMS.plxname.TERM.OLREORG
TES	MFS	UPDATE	IMS.plxname.TES.MFS


表 13. IMS コマンドのリソース名および RACF 権限 (続き)

コマンド verb	コマンド・キーワード	権限	リソース名
TRA	SET	UPDATE	IMS.plxname.TRA.SET
UNL	DB	UPDATE	IMS.plxname.UNL.DB
UNL	PGM	UPDATE	IMS.plxname.UNL.PGM
UNL	SYSTEM	UPDATE	IMS.plxname.UNL.SYSTEM
UNL	TRAN	UPDATE	IMS.plxname.UNL.TRAN
UPD	AREA	UPDATE	IMS.plxname.UPD.AREA
UPD	DATAGRP	UPDATE	IMS.plxname.UPD.DATAGRP
UPD	DB	UPDATE	IMS.plxname.UPD.DB
UPD	DBDESC	UPDATE	IMS.plxname.UPD.DBDESC
UPD	IMS	UPDATE	IMS.plxname.UPD.IMS
UPD	IMSCON	UPDATE	IMS.plxname.UPD.IMSCON
UPD	LE	UPDATE	IMS.plxname.UPD.LE
UPD	MSLINK	UPDATE	IMS.plxname.UPD.MSLINK
UPD	MSNAME	UPDATE	IMS.plxname.UPD.MSNAME
UPD	MSPLINK	UPDATE	IMS.plxname.UPD.MSPLINK
UPD	OLREORG	UPDATE	IMS.plxname.UPD.OLREORG
UPD	PGM	UPDATE	IMS.plxname.UPD.PGM
UPD	PGMDESC	UPDATE	IMS.plxname.UPD.PGMDESC
UPD	POOL	UPDATE	IMS.plxname.UPD.POOL
UPD	RM	UPDATE	IMS.plxname.UPD.RM
UPD	RTC	UPDATE	IMS.plxname.UPD.RTC
UPD	RTCDESC	UPDATE	IMS.plxname.UPD.RTCDESC
UPD	TRACE	UPDATE	IMS.plxname.UPD.TRACE
UPD	TRAN	UPDATE	IMS.plxname.UPD.TRAN
UPD	TRANDESC	UPDATE	IMS.plxname.UPD.TRANDESC
VUN	AREA	UPDATE	IMS.plxname.VUN.AREA

関連概念:

 IMS 操作タスク用のコマンド (オペレーションおよびオートメーション)

関連資料:

 同等の IMS タイプ 1 およびタイプ 2 コマンド (コマンド)


CSL OM セキュリティー・ユーザー出口ルーチン

OM は、コマンド処理時にオプションの OM セキュリティー・ユーザー定義出口ルーチンを呼び出します。OM プロシージャの CMDSEC パラメーターに A または E も同時に指定すると、その出口ルーチンはユーザー定義セキュリティ検査を実行します。OM 入力出口ルーチンの後に、OM セキュリティー・ユーザー定義出口ルーチンに制御が与えられます。


CSL OM セキュリティー出口ルーチンは、BPE 出口リスト PROCLIB メンバー・データ・セットの EXITDEF ステートメントに TYPE=SECURITY と定義されま

す。このタイプのユーザー定義出口ルーチンは 1 つ以上指定できます。この出口が呼び出されると、このタイプのすべてのユーザー定義出口ルーチンが、`EXITS=keyword` で指定された順序で駆動されます。

関連概念:

 [IMS PROCLIB データ・セット・メンバーを使用した Common Service Layer の定義 \(システム定義\)](#)

関連資料:

 [CSL OM セキュリティー・ユーザー出口 \(出口ルーチン\)](#)

第 10 章 CSL RM の管理

Resource Manager (RM) の管理に関連したタスクは、一般的には、システム管理者またはシステム・オペレーターが行います。

CSL RM により管理される情報

Resource Manager (RM) を作成して、リソース構造内のリソースを作成、更新、照会、または削除できます。

リソース構造に情報が保管されるリソースの例としては、次のものがあります。

- データベース、DEDB エリア、およびトランザクションに対するグローバル・リソース状況情報
- メッセージ宛先リソース情報。例: APPC 記述子、VTAM LTERM、MSNAME、VTAM 端末ノード、ユーザー、およびユーザー ID
- OLCSTAT データ・セットに対するグローバル情報。例: 現行オンライン変更ライブラリー
- IMSplex 全体パラメーター
- スケジュールされた逐次プログラム

RM は、リソース構造上のリソースに関する以下の情報を保守します。

リソース名

長さが 11 バイトのクライアント定義のリソース名。

名前タイプ

同じタイプ内のすべてのリソースは固有の名前を持つようにするリソース属性。

RM は、この規則を強制します。名前タイプ内のリソースは、異なるリソース・タイプを持つことができます。

リソース・タイプ

CQS がカップリング・ファシリティ・リスト構造上のリソースを物理的にグループ化できるようにする、クライアント定義のリソース属性。RM は、最大 255 個のリソース・タイプをサポートします。

RM は、グローバル情報を保管するには、リソース・タイプ 253 を使用します。リソース・タイプ 253 は、リソース・タイプ 11、22、33、44、...253 と同じ物理的リスト・ヘッダーにハッシュされます。クライアントは、同じ物理的リスト・ヘッダーにマップされるリソース・タイプを RM のグローバル情報として定義します。

リソース・バージョン

リソースが更新された回数。RM は、リソース構造上の更新の順番を識別できるようにバージョンを使います。

リソース所有者

リソースの所有者を意味するリソース属性。この属性はオプションです。

所有者が識別される場合、クライアントは、リソースの単一所有者を強制する責任があります。例えば、IMS は、リソース (ノード、LTERM、ユーザー、およびユーザー ID) の単一のアクティブ・インスタンスを強制するために、そのリソースに対して所有者を設定することができます。IMSplex 内の 1 つの IMS にサインオンしているユーザーまたはユーザー ID は、同時にその IMSplex 内の別の IMS にサインオンすることはできません。IMSplex 内の 1 つの IMS 上でアクティブな lterm は、同時にその IMSplex 内の別の IMS でアクティブになることはできません。

リソース・データ

そのリソースに関するすべての追加データが入っているリソース属性。

RM は、RM の初期化中または UPDATE RM コマンドの処理中に、Repository Server に登録し、IMSRSC リポジトリに接続します。RM がリポジトリ構造とともに定義されている場合、RM はそのリソース構造を使用して、リポジトリ名、リポジトリ・タイプ、監査アクセスの値、および RS z/OS システム間カップリング・ファシリティ (XCF) グループ情報などのリポジトリ情報を保管します。このリソース構造により、IMSplex 内のすべての RM が同じリポジトリを使用することが保証されます。

RM によって管理されるリソース構造に関する情報を表示するには、QUERY STRUCTURE コマンドを使用します。このコマンドについては、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」に詳細な情報があります。

関連概念:

235 ページの『IMSRSC リポジトリの CSL RM 管理』

データベース、DEDB エリア、およびトランザクションに対するグローバル情報の保守

IMS は、IMSplex 全体にわたってデータベース、DEDB エリア、およびトランザクションに対するコマンド状況を保守するのに RM リソース構造を使用します。このグローバル状況により、すべての IMS システムが IMSplex 内のデータベース、DEDB エリア、およびトランザクションを単一のデータベース、エリア、およびトランザクションとして表示できます。

IMS システムが再始動した場合、RM 内に設定されているグローバル・コマンド状況を使用し、適宜その状況をリソースに適用します。

グローバル・リソースの状況は、DBRC への登録の有無に関わりなくほとんどのデータベース、DEDB エリア、およびトランザクションに対し保守されます。共用副次索引データベースおよび MSDB に対してはデータベース状況は保守されません。

RM がデータベース状況または DEDB エリア状況を取得する時にエラーが発生すると、当該リソースに対してグローバル状況が保守されず、メッセージ DFS3308E が発行されます。DFS3308E には RM エラーの戻りコードおよび理由コードがリストされ、その後には、データベース・リソースまたは DEDB エリア・リソースに対するグローバル状況が使用不可であることを示すメッセージ DFS3500I が表示されます。これ以降の SCOPE(ALL) または GLOBAL を指定したすべての DB コマン

ドまたは AREA コマンドは RM 状況が更新されないようにします。エラー修正後、UPD IMS SET を発行して、グローバル状況を再び使用可能にすることができます。

オンライン変更がコミットされた後、追加されるすべてのデータベース、DEDB エリア、およびトランザクションに対してグローバル状況が取得されます。CREATE DB コマンドおよび CREATE TRAN コマンドに対するグローバル・コマンド状況も RM から取得されます。データベース、DEDB エリア、またはトランザクションは、追加リソースのグローバル状況が取得、処理されるまで、使用できません。追加リソースのグローバル状況を取得中または処理中にエラーが発生した場合、追加リソースにはエラーのマークが付きまます。

RM 内のデータベース、DEDB エリア、およびトランザクションに対してグローバル状況を保守するには、IMS 初期設定時に IMS PROCLIB PROCLIB のメンバー・データ・セット DFSDFxxx または DFSCGxxx 内に適切な PLEXPARM 値を指定します。

グローバル・リソースの状況は、IMSpIex 内で始動する最初の IMS システムにより設定され、その値はシステム上の PLEXPARM パラメーターから取得されます。RM を使ってこの情報をグローバルに保守することにより、例えば、あるデータベースをグローバルに停止でき、かつ IMSpIex に結合するいずれの IMS システムもこのデータベースが停止していることを認識します。同様に、オフライン・プロセスにより使用中のデータベースへのアクセスまたは更新を、IMSpIex に結合する IMS システムが行うことを防止できます。

データベース、DEDB エリア、およびトランザクション・リソースの最後の重大なグローバル状況のみが、RM 構造で保守されます。リソースに対して複数の更新コマンドが実行された場合、最新のコマンド状況のみが保守され、グローバル状況として照会可能になります。例えば、UPD TRAN NAME(TRANA) STOP(SCHD) コマンドが発行された場合、トランザクション TRANA に対するグローバル状況 STOSCHD が保守されます。その後、UPDATE TRAN NAME(TRANA) STOP(Q) コマンドが発行された場合は、トランザクション TRANA に対するグローバル状況 STOQ が保守されます。STOSCHD 状況と STOQ 状況の両方を保守するには、UPDATE TRAN NAME(TRANA) STOP(Q,SHCD) コマンドを発行する必要があります。

また、UPD IMS SET (PLEXPARM()) コマンドを使用して、PLEXPARM を動的に変更できます。UPD IMS コマンドに関する詳細については、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

RM は、CQS と協力して、クライアントに代わってリソース構造にアクセスします。RM はリソースを照会し、保守するための要求を発行し、クライアントに影響のあるリソース構造の変更があった場合、クライアントに知らせます。

IMSpIex 内の IMS システムは、引き続きリソース定義を保有します。RM は、IMSpIex 全体にわたるリソース定義の整合性を保証しません。IMS システム内のリソース定義を更新するには、グローバル・オンライン変更を使用できます。

以下の表は、IMS がどのように始動されるかに基づいて、リソースのグローバル状況が IMS によって、どのように取得および更新されるかを示します。

表 14. IMS によるリソースのグローバル状況の取得および更新方法

始動タイプ	状況の取得および更新方法
IMS 初期化	IMS は RM から PLEXPARM 値を読み取り、RM 値が優先するのか、または IMS 定義値を使用するのかを判断します。IMS は、その他の後続の IMS システムについては RM 内の PLEXPARM 値を更新します。RM のグローバル状況が保守されている場合、IMS は、RM のリソース構造からデータベース、DEDB エリア、およびトランザクションのグローバル状況を取得します。
コールド・スタート、または /ERE COLDBASE または /ERE COLDSYS コマンド	データベース、DEDB エリア、およびトランザクションのグローバル状況は、RM のリソース構造から取得されます。
ウォーム・スタートおよび緊急時再始動	ログ・レコードのローカル状況が最初に適用されます。ログ・レコード処理の完了後、RM から読み取ったグローバル状況が IMS システムのダウン中に更新されたと IMS が判断した場合、RM から読み取った RM のグローバル状況がリソースに適用されます。

データベース・リソース用に保管された情報

データベース用に RM が保守する情報は次のとおりです。

- データベース名
- データベース・タイプ
- データベース・アクセス・タイプ
- データベースが HALDB 区画の場合の HALDB マスター名
- データベースの状況

データベースに対しグローバル状況の更新を起こさせるコマンドおよびシステム処置には、次のものがあります。

- GLOBAL キーワード付きの /DBDUMP DB
- GLOBAL キーワード付きの /DBRECOVERY DB
- GLOBAL キーワード付きの /START DB
- GLOBAL キーワード付きの /STOP DB
- 以下のキーワードのいずれかを指定した UPDATE DB: START(Access) SET(ACCTYP) STOP(Access | SCHD | UPDATES) SET(LOCK(ON|OFF)) および SCOPE=ALL

グローバル・データベース情報は、UPD IMS SET(PLEXPARM(GSTSDB(N))) コマンドを使用して RM 構造から削除されます。

DEDB エリア・リソース用に保管された情報

DEDB エリア用に RM が保守する情報は次のとおりです。

- DEDB 名
- DEDB エリア・アクセス
- DEDB エリア名
- DEDB エリア状況

DEDB エリアに対しグローバル状況の更新を起こさせるコマンドおよびシステム処置には、次のものがあります。

- GLOBAL キーワード付きの /DBRECOVERY AREA
- GLOBAL キーワード付きの /START AREA
- GLOBAL キーワード付きの /STOP AREA
- 以下のキーワードのいずれかを指定した UPDATE AREA: START(ACCESS) SET(ACCTYPE) STOP(ACCESS |SCHD) および SCOPE=ALL

グローバル DEDB エリア情報は、UPD IMS SET(PLEXPARM(GSTSAREA(N))) コマンドを使用して RM 構造から削除します。

トランザクション・リソース用に保管された情報

RM がトランザクションについて保守する情報としては、トランザクション名およびトランザクション状況があります。

トランザクションに対しグローバル状況の更新を起こさせるコマンドおよびシステム処置には、次のものがあります。

- UPD TRAN STOP(Q) SCOPE=ALL
- UPD TRAN STOP(SCHD) SCOPE=ALL
- UPD TRAN START(Q) SCOPE=ALL
- UPD TRAN START(SCHD) SCOPE=ALL

グローバル状況を NULL にリセットするには、UPD IMS SET(PLEXPARM (GSTSTRAN=N)) コマンドを発行します。

メッセージ宛先リソース情報の保守

RM が IMSplex 内でリソース構造を使用する場合、IMS が IMS Transaction Manager のリソースを共有していることを示すことが可能です。このメッセージ宛先リソース情報には、端末、ユーザー、ユーザー ID、LTERM、MSNAME、および APPC 記述子が含まれます。

メッセージ宛先リソース情報を保守するには、IMS PROCLIB データ・セットの DFSDCxxx メンバー上で STM=YES と指定します。DFSDCxxx に関する詳細については「IMS V15 システム定義」を参照してください。

TM リソースの詳細については、367 ページの『第 18 章 IMSplex 内のトランザクション・マネージャーのリソースの計画』を参照してください。

OLCSTAT データ・セットに対するグローバル情報の保守

OLCSTAT データ・セット名は、RM リソース構造内で保守されます。これにより、IMSplex 内のグローバル・オンライン変更を行うことのできるすべての IMS システムが同じ OLCSTAT データ・セット名を使用するようになります。

IMSplex 内で最初に始動する IMS システムが、その OLCSTAT 名を OLCSTAT データ・セットに書き込みます。その後始動する IMS システムは、それぞれの OLCSTAT 名を RM 内の OLCSTAT 名と比較します。OLCSTAT 名が一致しない場合、IMS は 2800、コード 09 で異常終了します。

IMSplex 全体パラメーターの保守

PLEXPARM パラメーターである GSTSAREA=、GSTSDB=、および GSTSTRAN= は、グローバル状況を保守するかどうかを定義します。これらのパラメーターの値は、RM リソース構造内で保守されます。

これにより、IMSplex 内のすべての IMS システムがグローバルな PLEXPARM 値に対して同じ値を持つようになります。IMSplex 内で最初に始動する IMS システムが、PROCLIB 内の自身のメンバー・データ・セット DFSCGxxx または DFSDFxxx からグローバルな PLEXPARM 値を RM リソース構造に書き込みます。その後始動する IMS システムは、RM 内に定義された値を使用します。始動中の IMS システム内の PLEXPARM 値が RM 内の PLEXPARM 値と一致しない場合、RM 値により IMS 値がオーバーライドされることを示すメッセージ DFS3425I が発行されます。

IMS による、シスプレックス直列化プログラム管理のためのグローバル情報の維持方法

シスプレックス直列化プログラム管理を使用することにより、共用キュー環境のユーザーは、直列として定義されているアプリケーション・プログラムが、IMSplex 内の別の IMS システム内で並列にスケジューラれるのを防止することができます。

スケジュールに入れられている直列 PSB に関する情報は Resource Manager (RM) リソース構造で保守されており、それにより、その直列 PSB がいつでも IMSplex 全体で 1 つの IMS でのみスケジュールに入れられることが保証されます。

IMSplex 内の IMS が直列 PSB をスケジュールに入れるためには、その IMS は、スケジューリングを完了する前に、RM リソース構造にその PSB の固有インスタンスを正常に作成する必要があります。別の IMS が、その PSB インスタンスが固有でないことを検出すると、PSB のスケジューリング処理が打ち切られます。

IMS PROCLIB データ・セットの DFSCGxxx メンバー内の GBL_SERIAL_PGM パラメーターを使用して、シスプレックス逐次プログラム管理機能を活動化または非活動化することができます。デフォルトでは、シスプレックス直列化プログラム管理はアクティブです。

シスプレックス逐次プログラム管理には、以下のコンポーネントおよび構造が必要です。

- Common Service Layer (CSL) の Operations Manager (OM)、Structured Call Interface (SCI)、および RM のコンポーネント
- カップリング・ファシリティーで定義された RM リソース構造
- 共用メッセージ・キュー

アクティブになるように定義された共用キュー、RM、および RM 構造

CSL RM 用のリソース構造の二重化要件

CQS は、リソースの更新のログを記録しないし、リソース構造のチェックポイントもサポートしません。しかし、CQS は、構造の障害に備えて、バックアップとしてリソース構造の自動二重化をサポートします。

リソース構造をリカバリー可能にしたい場合、カップリング・ファシリティ・リソース管理 (CFRM) ポリシーに、自動的に二重化されるようにリソース構造を定義する必要があります。

CSL RM によるリソース構造の再移植方法

リソース構造とその複製 (複製があれば) に障害が発生した場合、CQS が新しい構造を割り振ることができれば、CQS は、RM に構造を再移植するように通知します。RM は、そのローカル制御ブロック内の情報から構造を再移植します。

RM は、次にそのクライアントに、構造を移植するディレクティブを発行します。リソース構造のみに存在していた RM または IMS リソースは失われます。新しいリソース構造が割り振られるとき、クライアントは、リソース構造の再移植を調整することを選択できます。

RM 再移植ディレクティブを受け取ったすべてのアクティブな IMS システムは、以下の情報を使って RM を更新します。

- 自身の PLEXPARM 値
- 自身のトランザクション
- グローバル状況を持つデータベース・リソース、DEDB エリア・リソース、およびトランザクション・リソースの自身のローカル・リスト (グローバル状況が保守されている場合)
- OLCSTAT 名
- VTAM リソースの場合のみ、端末、ユーザー (または ISC サブプール)、LTERM、ユーザー ID、および APPC 記述子名 (IMS 初期設定時に STM=YES が DFSDCxxx に指定されている場合)

IMSRSC リポジトリでリソース定義の保管が有効になっている場合、RM がリソース構造の再移植を促されると、各 RM はそのリポジトリ情報をリソース構造に書き込みます。

RM が再移植を要求したときに、すべての IMS システムがダウンしている場合、データベース、DEDB エリア、またはトランザクションのグローバル状況などの IMS 情報は、新しい RM 構造内では使用可能となりません。

z/OS によるリソース構造再作成方法

リソース構造は、システム管理される再作成を定義されているので、z/OS は、CQS が立ち上がっていなくて、構造を作成できない場合、自動的に構造を再作成します。構造に障害が発生した場合、または z/OS が構造との接続を失った場合、z/OS は構造を再作成することはできません。

CQS が立ち上がっていて、再作成が SETXCF START,REBUILD コマンドで開始されると、CQS は構造をコピーします。構造に障害が発生すると、リソース構造は構造チェックポイントをサポートしていないので、構造リカバリーは開始されません。

第 11 章 CSL SCI の管理

このトピックでは、SCI に関連する管理タスクについて説明しています。

CSL SCI のセキュリティー

IMSplex にアクセスするためには、クライアントは、最初に、SCI に登録する必要があります。クライアントは、クライアントが実行しているアドレス・スペースのユーザー ID が SCI に登録する権限を持っている場合にのみ、登録できます。IMSplex の RACF FACILITY クラス・プロファイルにクライアント・アドレス・スペースの権限を定義します。

RACF FACILITY クラスのプロファイル名は、`CSL.imsplex_name` の形式になっている必要があります。ここで、`imsplex_name` は、保護される IMSplex の名前です。`imsplex_name` は、文字「CSL」の後に PROCLIB データ・セットのメンバー CSLSIxxx 内の IMSPLEX パラメーターに定義された IMSplex 名が続いたものです。

クライアントが SCI に登録する CSLSCREG 要求を発行すると、SCI はセキュリティーを検査します。CSLSCREG 要求への対応として、SCI は、RACF に対して RACROUTE REQUEST=AUTH 呼び出しを発行して、クライアントが SCI に登録する権限を持っているかどうかを調べます。RACF は、CSLSCREG 要求を発行したアドレス・スペースのユーザー ID を検査します。このユーザー ID には、少なくとも SCI への登録のための UPDATE 権限がなければなりません。

UACC(NONE) プロファイルがなく、特定のユーザー ID に一致するプロファイルもない RACF FACILITY クラスを持つ IMSplex は、無保護です。SCI に登録する CSLSCREG 要求は、無保護 IMSplex に対して許可されます。

重要: SCI アドレス・スペースは、自ら登録を行いますが、セキュリティー許可は必要ありません。

例: IMSplex である PLEX1 は、CSLPLEX1 という IMSplex ID を持っています。IMSUSER1 と IMSUSER2 のユーザーのみが、PLEX1 内の SCI に登録できるプロファイルを定義するには、次の RACF コマンドを発行します。

```
RDEFINE FACILITY CSL.CSLPLEX1 UACC(NONE)
PERMIT CSL.CSLPLEX1 CLASS(FACILITY) ID(IMSUSER1) ACCESS(UPDATE)
PERMIT CSL.CSLPLEX1 CLASS(FACILITY) ID(IMSUSER2) ACCESS(UPDATE)
SETROPTS CLASSACT(FACILITY)
```

第 12 章 CQS の管理

このトピックでは、共通キュー・サーバーの使用に関連したシステム管理の作業について説明します。

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。

CQS の始動に必要な情報の記録

CQS は z/OS システム・ロガーを使用して、CQS がキュー構造をリカバリーして再始動するために必要なすべての情報を記録します。CQS は、使用する各カップリング・ファシリティ・リスト構造ペアに関するログ・レコードを、それぞれ別々のログ・ストリームに書き込みます。ログ・ストリームは、そのキュー構造を共用するすべての CQS アドレス・スペースで共用します。

z/OS システム・ロガーは、カップリング・ファシリティ・リスト構造上のキューを共用するすべての CQS アドレス・スペースに関するマージされたログを提供します。

重要: リソース構造への変更はログに記録されません。

CQS が z/OS システム・ログを使用するには、まず最初に、ログ・ストリームおよび関連するリソースを z/OS に定義する必要があります。

CQS には、ログ・レコードを印刷するためのファイル選択・フォーマット設定印刷ユーティリティも用意されています。

ログ・ストリームの定義について詳しくは、「z/OS MVS シスプレックスのセットアップ」を参照してください。

ファイル選択・フォーマット設定印刷ユーティリティの詳細については、「IMS V15 システム・ユーティリティ」を参照してください。

関連概念:

77 ページの『CQS リカバリー機能』

障害クライアント・テークオーバー時の CQS へのクライアント接続の確立

クライアント (例: IMS) が XRF テークオーバー機能をサポートしている場合、いずれか 1 つのクライアントが、障害が起きたクライアントの作業をテークオーバーする必要があります。障害が起きたクライアントに接続されている CQS は、アクティブではない場合があります。したがって、テークオーバー処理時は、障害が起きたクライアントからの作業をテークオーバーするクライアントは別の CQS に接続される必要があります、別のクライアントからの作業をテークオーバーしていることを示す必要もあります。

この時点で、CQS は、障害が起きたクライアントに接続されているログ・レコードを使用して、CQS の再始動と同じ処理を実行する必要があります。通常、CQS 障害クライアントの接続再始動 (ウォーム・スタート) は自動で、アクションを実行する必要はありません。

障害クライアントの接続の間に、CQS は、障害が起きたクライアントに接続されている CQS に対する構造からのログ・トークンを読み取ります。

- CQS がログ・トークンを検出した場合、CQS は、障害が起こったクライアントに対するウォーム・スタート処理を実行します。
- CQS がログ・トークンを検出しない場合、CQS は WTOR CQS0033A を発行します。この時点では、以下のいずれかを行うことができます。
 - クライアントの接続をコールド・スタートする
 - クライアントの接続要求をリジェクトする
 - 新しいログ・トークンを指定する

推奨事項: CQS が障害クライアントの接続再始動時にログ・トークンを受け入れない場合は、接続をコールド・スタートしてください。複数の CQS が実行されている場合、1 つの CQS 構造チェックポイントは、以前に障害が起きて再始動されなかった別の CQS に対するログ・レコードをパージする場合があります。

CQS へのアクセスの許可

インストール・システムに RACF または他のセキュリティー製品がインストールされている場合、セキュリティー管理者は、クライアントの CQS 構造に対する接続とアクセスの能力を制御するプロファイルを定義することができます。

CQS 登録の許可

クライアントが CQS に登録するための CQSREG 要求を出すと、CQS は、そのクライアントが CQS への登録の許可を受けているかどうかを判別するための RACROUTE REQUEST=AUTH 呼び出しを行います。RACF は、CQSREG 要求を出したクライアントのユーザー ID をチェックします。CQS に登録するためには、このユーザー ID に少なくとも UPDATE 権限が含まれている必要があります。

RACF セキュリティー管理者は、CQS への登録を制御するために、FACILITY クラスでプロファイルを定義することができます。プロファイル名は、CQS.cqs_id という形式でなければなりません。ここで、cqs_id は保護すべき CQS の ID です。cqs_id の値は、CQSIPxxx PROCLIB メンバー・データ・セットで定義されたサブシステム名 (SSN) の後に、文字 CQS を続けたものです。例えば、SSN が ABC であれば、cqs_id の値は ABCCQS になります。

CQS 登録を許可するための RACF コマンド

CQSUSER1 および CQSUSER2 以外のユーザーの登録を防止するための CQS 用プロファイルを定義するには、以下の例に示す RACF コマンドを実行します。

```
RDEFINE FACILITY CQS.ABCCQS UACC(NONE)
PERMIT CQS.ABCCQS CLASS(FACILITY) ID(CQSUSER1) ACCESS(UPDATE)
PERMIT CQS.ABCCQS CLASS(FACILITY) ID(CQSUSER2) ACCESS(UPDATE)
SETROPTS CLASSACT(FACILITY)
```

CQS 構造への接続の許可

クライアントが CQS 構造に接続するための CQSCONN 要求を出すと、CQS は、そのクライアントが構造へのアクセスの許可を受けているかどうかを判別するための RACROUTE REQUEST=AUTH 呼び出しを行います。RACF は、CQSCONN 要求を出したクライアントのユーザー ID をチェックします。このユーザー ID は、少なくとも CQS による構造への接続のための UPDATE 権限を持っている必要があります。

RACF セキュリティー管理者は、CQS 構造に対する接続を制御するために、FACILITY クラスでプロファイルを定義する必要があります。プロファイル名は、CQSSTR.*structure_name* という形式でなければなりません。ここで、*structure_name* は、保護すべき基本 CQS 構造の名前です。CQSSGxxx および CQSSLxxx の PROCLIB メンバー・データ・セットで定義するものと同じ構造名を使用してください。

CQSSTR.*structure_name* プロファイルは、CQS による指定された構造へのアクセスを制御するだけで、IXL マクロを使用する構造に対する直接アクセスは制御しません。IXLSTR.*structure_name* という形式の RACF プロファイルを定義することによって、構造の直接アクセスを制御することができます。このようなプロファイルを作成する場合は、構造への CQS のアクセスを実行するためのユーザー ID を指定する必要があります。

関連資料: カップリング・ファシリティ構造への直接アクセスの保護に関する詳細については「z/OS MVS Programming: Sysplex Services Guide」内の『カップリング・ファシリティ要求を許可する』を参照してください。

構造名の定義について詳しくは、「IMS V15 システム定義」を参照してください。基本構造とオーバーフロー構造は 1 つの単位と見なされているので、CQS はオーバーフロー構造名の独立したチェックは行いません。

例: IMSMSGQ01 という名前の CQS 基本構造のプロファイルを定義するため、およびユーザー CQSUSER だけがそれに接続できるようにするためには、以下の例に示された RACF コマンドを実行します。

```
RDEFINE FACILITY CQSSTR.IMSMGQ01 UACC(NONE)
PERMIT CQSSTR.IMSMGQ01 CLASS(FACILITY) ID(CQSUSER) ACCESS(UPDATE)
SETROPTS CLASSACT(FACILITY)
```

特定の CQS 構造に対するプロファイルを定義しない場合、その構造は保護されず、どのユーザー ID でも、CQSCONN 要求を出せばその構造にアクセスすることができます。

CQS における構造変更の使用

構造変更は、構造サイズを変更するために、または構造内のオブジェクトを再配布するために使用することができる、CQS によってサポートされた z/OS プロセスです。CQS は、基本キュー構造、オーバーフロー・キュー構造、およびリソース構造に対する構造変更をサポートしています。CQS では、以下のようにして、基本構造またはオーバーフロー構造のサイズを動的に変更することができます。

構造変更を可能にするためには、CFRM ポリシーをアクティブにし、このポリシー内の INITSIZE および SIZE パラメーターを定義してください。構造サイズに関する詳細については、「IMS V15 システム定義」を参照してください。

構造サイズを変更するには、以下の XES コマンドを入力してください。

```
SETXCF START,ALTER,STRNAME=strname,SIZE=size
```

size の値は、CFRM ポリシーの INITSIZE と SIZE の値の間の範囲内であることが必要です。

自動構造変更は、構造フルしきい値に到達したときに、構造サイズまたは比率を入力するエレメントを自動的に変更することができる z/OS 機能です。CQS は、キュー構造およびリソース構造に対する自動構造変更をサポートしています。自動構造変更を可能にするためには、INITSIZE、SIZE、および ALLOWAUTOALT(YES) によって定義された CFRM ポリシーをアクティブにしてください。

重要: カップリング・ファシリティ・ストレージが制限された場合、自動構造変更によって使用可能になった構造は、z/OS によってサイズが縮小される候補になります。キュー構造に対する自動構造変更を使用可能にするときは注意してください。z/OS がキュー構造サイズを縮小する場合、キュー構造が不必要にオーバーフローする原因となる場合があります。これが起こるのを防ぐには、MINSIZE (最小サイズ) を使用して、それ以下では z/OS が構造を縮小しない CFRM ポリシーを定義してください。

CFRM ポリシー構造定義内の FULLTHRESHOLD パラメーターは、構造に対するパーセント・フルしきい値を指定します。これは、自動変更処理発生前にエンタリーやエレメントなどカップリング・ファシリティのリソースが到達しなければならないパーセント・フルのしきい値です。FULLTHRESHOLD パラメーターのデフォルト値は、80 パーセントです。CQSSGxxx PROCLIB メンバーのパラメーター OVFLWMAX は、CQS オーバーフロー処理が発生するために使用中でなければならない構造内のエレメントのパーセントを指定します。OVFLWMAX のデフォルト値は、70 パーセントです。

推奨事項: ALLOWAUTOALT(YES) を使用する場合、OVFLWMAX 値が FULLTHRESHOLD 値より少なくとも 5 パーセント大きくなるよう、これらのデフォルト値のうち片方または両方を変更します。これは、エンタリー対エレメントの比率および構造のサイズを変更し、CQS オーバーフロー処理を回避する機会を z/OS に与えるためです。また、これらのパラメーターを異なる値に設定しておけば、自動変更処理および CQS オーバーフロー処理の両方が同時に発生して、余計なオーバーヘッドが生じる可能性は低くなります。

自動構造変更について詳しくは、「z/OS MVS シスプレックスのセットアップ」を参照してください。

CQS システム・チェックポイントの使用

CQS システム・チェックポイント (リカバリーに使用されるチェックポイント・データ・セット) は、少なくとも 1 つのキュー構造を管理している場合、CQS に適用されます。CQS がリソース構造のみを管理する場合は、システム・チェックポイントは適用されません。

CQS 情報のリカバリーのためのシステム・チェックポイントで、CQS は再始動およびリカバリーの情報が入ったログ・レコードを CQS ログに書き込みます。CQS は、チェックポイントが進行中の間はアクティビティーを停止することができません。

CQS は、次に示すそれぞれの状態において、システム・チェックポイントを実行します。

- クライアントが CQSCHKPT FUNC=CHKPTSYS 要求を出したとき
- CQS が書き込むログ・レコードの数が CQSSLxxx PROCLIB メンバー・データ・セットの SYSCHKPT= パラメーターで指定された値に達したとき
- クライアントが IMS であり、/CQCHKPT SYSTEM コマンドが実行されたとき
- クライアントの RESYNC が終了したとき
- 構造チェックポイントが正常に終了したとき
- 再始動の終了時

このほか、CQS はシャットダウンなどの重要なイベント中にシステム・チェックポイントを実行します。

CQS チェックポイント・データ・セット

CQS は、各構造ペアごとにチェックポイント・データ・セットを維持管理しています。CQS はそのチェックポイント・データ・セットを書き込んでおき、それを再始動時に使用します。

チェックポイント・データ・セットは、CQS 初期設定時に動的に割り当てられます。構造のチェックポイント・データ・セット DSNAME は、PROCLIB メンバー・データ・セットの CQSSLxxx の中の CHKPTDSN= パラメーターを使用して定義します。

システム・チェックポイント後の CQS の再始動方法

CQS の再始動時に、CQS は、最後のシステム・チェックポイントからログ・レコードを読み取って、コミット済みデータ・オブジェクトの環境を復元し、キュー構造上のコミットされていないデータ・オブジェクトをバックアウトします。システム・チェックポイントの頻度によって、この再始動が影響を受けます。CQS は、チェックポイントが頻繁に起きない場合は、チェックポイントがより頻繁に起きる場合より多くのログ・レコードを読み取る必要があります。CQS ログは複数の CQS によって共用されているので、CQS の再始動時間は、再始動されている CQS だけでなく、複数の CQS によって書き込まれたログ・レコードの数によって影響を受けます。

CQS は再始動の終了時に初期システム・チェックポイントをとります。

関連概念:

115 ページの『CQS システム・チェックポイント・データ・セット』

CQS 構造チェックポイントの使用

構造チェックポイントは、構造障害の後 CQS がキューをリカバリーできるように、キュー構造上の共用キューのスナップショットを実行し、そのデータを構造リカバリー・データ・セット (SRDS) に書き込みます。構造チェックポイント処理では、すべてのリカバリー可能データ・オブジェクトを構造ペアから SRDS へコピーします。

リカバリー不能データ・オブジェクトに関しては、キュー名および UOW はコピーされますが、実際のデータ・オブジェクトはコピーされません。クライアントは、データ・オブジェクトを共用キューへ挿入するための CQSPUT FUNC=PUT 要求が出されると、そのデータ・オブジェクトがリカバリー可能かどうかを指定します。例えば、IMS がクライアントの場合、高速機能入力メッセージを除き、すべてのデータ・オブジェクトがリカバリー可能とマークされます。

重要: 構造チェックポイントは、リソース構造に対してサポートされていません。キュー構造のみサポートします。

コピー操作を実行するとき、CQS は、チェックポイントをとっている間に構造が変化しないようにするために、構造に対するすべてのアクティビティを停止します。CQS が構造チェックポイントの進行中に作業を処理するための要求を受け取ると、その要求は構造チェックポイントの完了後まで保留にされます。

推奨事項: CQS がチェックポイントをとっている間は構造に関する他の作業は処理できないので、構造チェックポイントの処理は、ピーク時以外の時間に行うよう考慮してください。

すべての共用キューが SRDS にコピーされた後で、各 CQS は、再始動チェックポイントのタイム・スタンプが必ず現行の構造チェックポイントより新しくなるようにするために、システム・チェックポイントを実行します。次に、構造チェックポイント・プロセスは、構造リカバリーに不要なログ・レコードをすべて削除して、ロガーが CQS ログ内のスペースを再使用できるようにし、ログがいっぱいになるのを防ぎます。ログ・レコードを削除してしまうと、CQS はそのログ・レコードにアクセスできなくなり、その結果、それらのレコードを構造リカバリーや CQS 再始動に使用できなくなります。有効な構造チェックポイント・データが入っている SRDS が 1 つだけの場合、その構造チェックポイントより前に作成されたログ・レコードがすべて削除されます。両方の SRDS に有効な構造チェックポイント・データが含まれている場合、一番古い構造チェックポイントより前に書き込まれたログ・レコードがすべて削除されます。

構造チェックポイントの時に CQS がアクティブでなかった場合、CQS はシステム・チェックポイントを開始できません。これは、再始動チェックポイントが少なくとも 1 つの構造チェックポイント分だけ古いということを示します。両方の SRDS に有効な構造チェックポイント・データが含まれている場合は問題ありません (まだ CQS 再始動チェックポイントが最も古い構造チェックポイントより新しいため、その再始動ログ・レコードは削除されません)。ただし、これが最初または唯一の有効な構造チェックポイントの場合、または CQS が 2 つの構造チェックポイントにわたってダウンした場合、その CQS の再始動に必要なログ・レコードは削除されます。このような状況では、CQS を再始動するためにコールド・スタートが必要な場合があります。

推奨事項: 構造のコールド・スタートの後、または SRDS が削除および再定義された時は、構造チェックポイントを開始してください。構造チェックポイントは、その構造を共用するすべての CQS が開始された後に開始する必要があります。これにより、SRDS に初期構造チェックポイントが提供されます。また、共用キューのスナップショットを更新し、ログ・レコードを定期的に削除するために、構造チェックポイントを一定間隔で開始してください。

構造リカバリーが必要である場合、SRDS および CQS のログを使用して共用キューをリカバリーします。CQS は、最初に新しい構造を SRDS から再移植します。CQS はその後で構造チェックポイントの完了時以来のすべてのログ・レコードを読み取ります。ログ・レコード読み取りのための時間の長さは、ログ内にどれだけログ・レコードがあるかによって異なります。構造チェックポイントを頻繁に行うと、それだけ構造リカバリー時に読み取りが必要なログ・レコードの数が少なくなります。ログ・レコードの削除も、ログがいっぱいになるのを防止するために役立ちます。ログ・ストリームがいっぱいになると、CQS は最も古い構造チェックポイントまたは CQS システム・チェックポイントより古いすべてのログ・レコードを削除します。CQS は、その後で構造チェックポイントをとります。

CQS は、次に示すそれぞれの状態において、構造チェックポイントを実行します。

- z/OS ログがいっぱいになるかまたはいっぱいに近付いたとき
- 構造リカバリーが成功した後
- オーバーフローしきい値プロセスが成功した後
- クライアントが CQSCHKPT FUNC=CHKPTSTR 要求を出したとき
- クライアントが IMS であり、/CQCHKPT SHAREDQ コマンドが実行されたとき
- クライアントが CQSDISC 要求で CQS の正常終了を要求した場合の、CQS の正常終了時。クライアントが IMS の場合、/CQSET SHUTDOWN SHAREDQ コマンドを入力することにより、CQS 終了時に構造チェックポイントを要求することができます。IMS は次に、IMS の正常終了時に /CHECKPOINT FREEZE|DUMPQ|PURGE コマンドによってこの要求を CQS へ渡します。
- ユーザー指定のログ・レコード番号に達したとき

関連概念:

77 ページの『CQS リカバリー機能』

CQS 構造フルの防止

構造の使用量を管理して、構造フル状態を回避してください。リソース構造またはキュー構造がフルになった場合、CQS はメッセージ CQS0205E を発行します。

構造フルを防ぐ方法には、さまざまな方法があります。

- キュー構造に対する CQS 構造オーバーフロー機能
- キュー・スペース通知出口ルーチン (DFSQSSP0) を使用した共用メッセージ・キュー使用量のモニター
- キュー構造およびリソース構造に対する、CQS によって使用される、z/OS 構造フル・モニター機能

これらのメカニズムを使用して、構造フル状態が近づいてきたときに警告を発行し、フル構造を防ぐためのアクションを実行してください。

関連概念:

76 ページの『CQS 構造機能』

CQS 構造オーバーフロー機能

CQS は、キュー構造がフルに近づいていて、フル構造を防ぐための処置をとるときに、自動的に警告を出す構造オーバーフロー機能を提供します。構造の使用量がオーバーフローしきい値に達した場合、CQS は構造変更を開始して、構造をより大きくしようとします。この変更が失敗すると、CQS は、オーバーフロー構造を割り振り、選択されたキューに関連付けられたデータ・オブジェクトをオーバーフロー構造に移動するか (オーバーフロー構造を定義している場合)、または選択されたキューにデータ・オブジェクトが入れられることをリジェクトします。CQS は、キューを選択している間、構造に対するすべてのアクティビティを停止します。

定義: オーバーフローしきい値 は、CQS がオーバーフロー・モードに入る前に使用されている必要のある、基本構造のパーセンテージです。デフォルト・オーバーフローしきい値は 70% ですが、PROCLIB データ・セットの CQSSGxxx メンバー内の OVFLWMAX パラメーターを定義することで、このデフォルトを変更することができます。

重要: 構造オーバーフローは、リソース構造に対してサポートされていません。

CQS が構造のサイズの変更に失敗した場合、構造はオーバーフロー・モードに入ります。オーバーフロー・モードでは、CQS は、構造上の最大スペースをオーバーフロー処理に対する候補として使用しているキューを選択します。十分なキューが選択されたために基本構造の使用量がオーバーフローしきい値を 20% 下回った場合、CQS はキューを選択することを停止します。オーバーフローに対してキューが選択されている間は、構造に対するアクティビティは一時的に停止されます。CQS は、候補キュー名の付いたキュー・オーバーフロー・ユーザー提供の出口ルーチンを駆動します。その場合、この出口はオーバーフロー処理に対して承認またはリジェクトします。承認されたキューはオーバーフロー・モードに入ります。オーバーフロー構造が定義されている場合、CQS はオーバーフロー構造を割り振り、承認されたキューに関連付けられているデータ・オブジェクトをオーバーフロー構造に移動します。オーバーフロー構造が定義されていない場合、CQS は承認されたキューに対する CQSPUT 要求をリジェクトします。オーバーフロー構造は、OVFLWSTR パラメーターを使用して PROCLIB データ・セットの CQSSGxxx メンバー内で定義することが可能です。

オーバーフロー構造が割り振られる場合、CQS がキューにそれ以上項目がないことを検出するまで、キューはオーバーフロー・モードにとどまります。(キューは 15 分おきにスキャンされます。) 空のキューが検出された後、そのキューはオーバーフロー・モードから削除され、通常の処理に戻ります。スキャンは定期的に行われるため、一時的に空のキューがオーバーフロー・モードにとどまる場合があります。すべてのキューが削除されると、CQS はオーバーフロー・モードを終了します。オーバーフロー構造が割り振られていない場合は、1 次構造の使用量がオーバーフローしきい値を 20% 下回ったときに CQS はオーバーフロー・モードを終了します。

要確認: 大容量メッセージ (キュー・マネージャーの長メッセージ・レコードのサイズを超えるメッセージ) は、複数の部分に分割されて共用キュー構造に入れられます。メッセージの最初の部分は、トランザクション作動可能キューまたは LTERM 作動可能キューに入れられます。2 番目とそれ以降の部分は、ステージング・キューと呼ばれる 2 次キューに入れられます。トランザクション・キューまたは LTERM キューがオーバーフロー処理の対象として選択された場合、そのキューにある大容量メッセージの最初の部分のみがオーバーフロー構造に移動されます。ステージング・キューにあるメッセージの部分はすべて、基本構造に残されません。ステージング・キューは、オーバーフロー処理に適格ではありません。LGMSGSZ パラメーターを使用して、キュー・マネージャーの長メッセージ・レコードのサイズを変更できます。

キュー・スペース通知出口ルーチン (DFSQSSP0) を使用した共用メッセージ・キュー使用量のモニター

IMS は、カップリング・ファシリティ上での共用メッセージ・キュー構造の全体的な使用量に関するフィードバック機能を備えた、共用キュー用のキュー・スペース通知出口ルーチン (DFSQSSP0) を提供しています。これにより、キュー構造をモニターして満杯になることを防止するソリューションを出口に実装できます。

IMS は、共用キューを使用して実行されている場合、トランザクション・メッセージを IMS 共用メッセージ・キュー・カップリング・ファシリティ構造に格納します。構造の使用量は、システムがどの程度ビジーか、トランザクション処理速度がどれだけ高速かに応じて、ビジネス・オペレーション中に変化します。メッセージの到着頻度がメッセージの処理速度より高くなると、構造がオーバーフローする可能性があります。共用キューがいっぱいになると、システム可用性に影響が出ることがあります。

キュー・スペース通知出口ルーチンを介して、基本構造およびオーバーフロー構造の両方にどの程度のメッセージ・キュー構造が使用されているかについての情報を受け取ることができます。

- 合計および使用中の構造項目の数。
- 合計および使用中の構造エレメントの数。

IMS が共用キュー構造にメッセージを書き込んだり、共用キュー構造からメッセージを削除したりするたびに、この構造使用量情報が更新されます。DFSQSSP0 出口パラメーター・リスト DFSPARM は、IMS とユーザー出口 DFSQSSP0 との間の通信のための入力パラメーターと出力パラメーター提供します。




キュー・スペース通知出口パラメーター・リスト DFSPARM では、既存の PARM DSECT の下で以下のフィールドを使用して構造使用量情報を取得できます。

- QSPCFBKP - 構造のフィードバック域のアドレス
- QSPCFBKL - 構造のフィードバック域の長さ

フィードバック域の詳細については、IMS に付属の CQSSFBA マクロを参照してください。

注: 共有キューのキュー・スペース通知出口 (DFSQSSP0) は、IMS 共有メッセージ・キュー構造のみに関する構造使用状況情報を提供します。この出口は、IMS 共有 EMH 構造に関する情報は提供しません。

関連資料:

-  キュー・スペース通知出口ルーチン (DFSQSPC0/DFSQSSP0) (出口ルーチン)
-  CQSDEL 要求 (システム・プログラミング API)
-  CQSPUT 要求 (システム・プログラミング API)

CQS 構造のフル・モニター

構造がフルに近づいたときに警告を発行し、フル構造を防ぐために、z/OS 構造フル・モニター機能を、キュー構造およびリソース構造に対して使用することができます。

構造フル・モニターが使用可能なとき、z/OS は構造の使用量をモニターします。使用中のエントリーまたはエレメントの数が構造フルしきい値に到達した場合、z/OS はハイライトされた IXC585E メッセージを発行し、システム・プログラマーに対して構造フル状態が差し迫っていることを警告します。自動変更が使用可能になっている場合、z/OS は、自動的に構造変更を開始し、構造サイズを増加させたり、エントリーに対するエレメントの割合を変更したりします。

構造フル・モニターは、80% のデフォルトしきい値によって自動的に使用可能になります。CFRM ポリシー FULLTHRESHOLD パラメーターを使用して、別のしきい値を定義してください。FULLTHRESHOLD(0) を使用して CFRM ポリシーを定義し、構造フル・モニターを使用不可にしてください。構造の使用量がしきい値を下回った場合、z/OS は IXC586I メッセージを発行します。

以下のコマンドは、有効な構造フルしきい値を表示します。

```
D XCF,STRUCTURE,STRNAME=strname
```

以下の例では、コマンド D XCF,STRUCTURE,STRNAME=IMSRSRC01 が実行され、構造フルしきい値は 80% です。

```
STRNAME: IMSR SRC01
STATUS: NOT ALLOCATED
POLICY SIZE      : 4096 K
POLICY INITSIZE  : N/A
FULLTHRESHOLD   : 80
REBUILD PERCENT : N/A
DUPLEX          : DISABLED
PREFERENCE LIST : LF03
ENFORCEORDER    : NO
EXCLUSION LIST  IS EMPTY
```

以下の例は、すべてのエントリーが使用中のために構造がフルになっていることを表す IXC585E メッセージを示しています。

```
*IXC585E STRUCTURE IMSR SRC01 IN COUPLING FACILITY LF03, 725
PHYSICAL STRUCTURE VERSION B4704775 92D95302,
IS AT OR ABOVE STRUCTURE FULL MONITORING THRESHOLD OF 80%.
ENTRIES: IN USE:      4874 TOTAL:      4874, 100% FULL
ELEMENTS: IN USE:      19 TOTAL:      4872,  0% FULL
```

以下の例は、IXC586I メッセージを示します。

IXC586I STRUCTURE IMSRRC01 IN COUPLING FACILITY LF03, 772
PHYSICAL STRUCTURE VERSION B4704775 92D95302,
IS NOW BELOW STRUCTURE FULL MONITORING THRESHOLD.

関連資料: 構造フル・モニターおよび CFRM ポリシー内の FULLTHRESHOLD および ALLOWAUTOALT キーワードに関する詳細については「z/OS MVS システムのセットアップ」を参照してください。

CQS 構造オーバーフローを伴う構造フル・モニターの使用

キュー構造には、構造オーバーフロー機能 (CQS 機能) とともに構造フル・モニター機能 (z/OS 機能) を使用することができます。

オーバーフローしきい値は、CQS に対して定義された値です。構造フルしきい値は、z/OS に対して定義された値です。オーバーフローしきい値が構造フルしきい値に近く、自動変更が使用可能な場合、CQS および z/OS の両方が構造変更を同時に開始し、構造がフルになるのを防ごうとすることがあります。

構造フルしきい値に達したことを z/OS が検出したときに CQS によって開始された構造変更が進行中だった場合、z/OS は CQS によって開始された構造変更を停止し、それ自身の構造変更を開始します。CQS が、CQS 構造変更が失敗したことを検出した場合、たとえ z/OS によって開始された構造変更が構造の使用量をオーバーフローしきい値以下に減少させた場合でも、CQS はオーバーフロー・モードに入ります。

推奨事項: いつ構造がオーバーフロー・モードに入るかを制御するためにどのオーバーフローしきい値を定義するかを決定するときには、構造フルしきい値を考慮してください。構造フルしきい値を使用している場合、オーバーフロー・モードに不必要に入ることを避けるために、構造フルしきい値をオーバーフローしきい値より低く定義してください。構造フルしきい値がオーバーフローしきい値より低い場合、z/OS は、構造がオーバーフロー・モードに入る前に構造フルしきい値処理を行うことができます。

関連資料:

- PROCLIB データ・セット の CQSSGxxx メンバーに関する詳細については、「IMS V15 システム定義」を参照してください。
- キュー・オーバーフロー・ユーザー提供の出口ルーチンに関する詳細については、「IMS V15 出口ルーチン」を参照してください。
- CQSPUT 要求に関する詳細については、「IMS V15 システム・プログラミング API」を参照してください。

CQS における構造再作成

構造再作成は、構造の別のインスタンスに初期構造インスタンスから再構成された同じ名前とデータを割り振れるようにする z/OS プロセスです。z/OS は、システム管理の再作成、CQS 管理の再作成、および構造の二重化をサポートします。

CQS は、キュー構造およびリソース構造に対するシステム管理の再作成および CQS 管理の再作成をサポートします。CQS は、構造再作成時、構造に対するすべてのアクティビティを停止することに注意してください。

関連概念:

76 ページの『CQS 構造機能』

z/OS システム管理の再作成および CQS

システム管理再作成とは、z/OS が構造を再作成する z/OS プロセスです。z/OS は、構造内容を新規構造にコピーします。システム管理の再作成は、キュー構造およびリソース構造に対してサポートされています。活動している CQS がない場合、システム管理の再作成は実行されるだけです。CQS が活動している場合、CQS はユーザー管理の再作成を実行し、構造のコピーを実行します。

システム管理の再作成は、主に計画された再構成に対して使用してください。再作成が SETXCF START,REBUILD コマンドによって開始され、かつ CQS が使用可能ではないため構造のコピーが実行できない場合、z/OS は構造のコピーを実行します。

制約事項: システム管理の再作成は、カップリング・ファシリティ障害、構造障害、接続の逸失のいずれにも対処しません。これらの障害をハンドルするためには、CQS 管理の再作成が必要です。

システム管理の再作成に対する構造を使用可能にするためには、以下のパラメーターを CFRM 結合データ・セット・ユーティリティー・ジョブに追加してジョブ制御言語 (JCL) を実行し、システム管理の再作成機能を使用して CFRM 結合データ・セットを形式設定してください。

ITEM NAME(SMREBLD) NUMBER(1)

CQS 管理の再作成

CQS 管理の再作成は、CQS が構造再作成を管理するプロセスです。CQS は、CQS 管理の再作成の 2 つのバリエーション、構造のコピーおよび構造リカバリーをサポートします。

構造のコピーは、計画された再構成または接続逸失のために、構造の内容を別の構造にコピーします。また、構造のコピーを使用して、新規 CFRM ポリシー属性を活動化することができます。構造リカバリーは、構造障害の後、SRDS および z/OS ログから構造をリカバリーします。

1 つの CQS が構造への接続を逸失し、別の CQS がその構造への接続を引き続き持っている場合、CQS は構造再作成を管理し、構造のコピーを実行します。すべての CQS がリソース構造への接続を逸失した場合、構造リカバリーが試行されますが、構造リカバリーがリソース構造に対してまたはリカバリー不能なキュー構造に対してサポートされていないため (RECOVERABLE=NO) 失敗します。

カップリング・ファシリティまたはキュー構造に障害が起こった場合、CQS は構造リカバリーを実行します。

リソース構造に障害が起こった場合、リソース構造は逸失し、構造再作成は実行されません。リソース構造はチェックポイントおよびロギングをサポートしていないため、CQS は構造リカバリーを実行することができません。CQS クライアントは、障害の起きたリソース構造を再移植することが可能です。CQS は新規リソース構造の割り振りを試行します。

新規構造が正常に割り振られた場合、CQS は再移植構造イベントとともにクライアント構造イベント出口を駆動します。その場合、CQS クライアントは構造を再移植する必要があります。新規構造が正常に割り振られなかった場合、CQS は、構造障害イベントとともに構造出口イベントを駆動します。構造は、再移植に対してアクセス可能ではありません。構造の割り振りの障害の原因となった環境の問題を訂正し、構造が割り振りおよび再移植されるようにしてください。

z/OS および CQS による構造再作成の開始

構造再作成は、z/OS オペレーター、CQS、または z/OS によって開始することができます。

- z/OS オペレーターは、構造再作成を開始し、以下のコマンドを使用してキューをコピーまたはリカバリーすることができます。
`SETXCF START,REBUILD,STRNAME=strname,LOCATION=NORMAL/OTHER`
- CQS は、CQS 初期設定時に、空の構造と有効な SRDS (SRDS 内の有効な構造チェックポイントを示す) を検出した場合、構造再作成を開始します。CQS が空の構造および有効 SRDS を検出した場合、CQS は、イベント通知機能 (ENF) 35 イベント処理時に構造再作成を開始することができます。
- 接続の逸失に対する再作成しきい値に到達した場合、z/OS は構造再作成を開始します。接続の逸失に対する再作成しきい値は、CFRM ポリシー `REBUILDPERCENT` キーワードにより定義されます。`REBUILDPERCENT` デフォルトは 1 です。システム・プログラマーが `REBUILDPERCENT` を定義していないとき、構造への接続を逸失した CQS が存在する場合は、z/OS は再作成を開始します。
- 構造のコピーが CQS 障害のために打ち切られ、他の CQS は障害が起きた CQS がマスターであるかどうか判別できない場合、再作成は構造リカバリーとして開始されます。

CQS 構造の再移植

構造の再移植は、CQS クライアントが、障害の起きたリソース構造を再移植するプロセスです。CQS はリソース更新をログに記録しない、またはリソース更新のチェックポイントを指定しないので、CQS はリソース構造に対する構造リカバリーをサポートしません。

リソース構造およびその二重化に障害が起こった場合、CQS クライアントはそのリソース構造を再移植することができます。CQS は、新規構造の割り振りを試行します。この割り振りが正常に行われた場合、CQS はクライアントに対して、再移植するよう通知します。その場合、CQS クライアントは構造を再移植する必要があります。リソース構造上のみ保持されていたリソースはすべて逸失します。

CQS が新規構造の割り振りに失敗した場合、CQS はクライアントに対して、構造に障害が起きたことを通知します。シスプレックス環境がその後で変更され、その結果 CQS が新規リソース構造を割り振ることができるようになった場合、CQS はクライアントに対して、その時点で再移植するよう通知します。または、構造の割り振りの障害の原因となった環境の問題を訂正し、構造が割り振りおよび再移植されるようにしてください。

CQS は CQS クライアント間のリソース構造の再移植を調整しません。そのため、クライアントは、リソース構造の再移植を行う場合は、その同期化を行う必要があります。構造の再移植は、すべてのオブジェクトの復元を保証するわけではありません。オブジェクトの一部が逸失する場合があります。

CQS 構造リカバリー

構造リカバリー機能は、構造障害の後、SRDS および z/OS ログから、構造上にあるデータ・オブジェクトをリカバリーします。

重要: 構造リカバリーは、リソース構造に対してサポートされていません。

構造障害の後、構造が空の場合または構造が CQS 制御情報のみを含む場合、構造をリカバリーする必要がある場合があります。構造リカバリー時、CQS は構造を割り振り、(以前のチェックポイントからの有効クライアント・データを含んでいる) SRDS および CQS ログのいずれかから、または CQS ログそれ自身からの構造を再移植します。

CQS は構造チェックポイントから構造をリカバリーし、構造リカバリー・データ・セットからのデータ・オブジェクトとともにその構造を再移植します。CQS は、構造チェックポイントが起こるときに開始されるログを読み取り、構造チェックポイントの後に起こった変更とともに構造を更新します。

基本構造が空で、すべての SRDS が有効構造チェックポイント・データを含んでいない場合、CQS はリカバリーに対する CQS ログを使用することが可能かどうかを判断します。ログ・ストリーム内の最初のログ・レコードがログの先頭ログ・レコードである場合、そのログ・ストリームはリカバリーに必要なすべてのログ・レコードを含み、CQS はそれらのログ・レコードを使用して構造リカバリーを完了することができます。

CQS が、以前の構造再作成が正常にコンプリートしなかったことを検出した場合、CQS は別の再作成を開始します。

基本構造が CQS 制御情報のみを含み、構造を割り振った CQS が、再作成が必要かどうかを判断できない場合で、SRDS のいずれかが有効であるとき、またはすべてのログ・レコードが使用可能なとき、CQS は再作成を開始します。

いずれの SRDS も有効でなく、ログ・レコードが以前の構造チェックポイントによって削除されている場合、CQS はその構造を再作成することはできません。このケースの場合、CQS は WTOR CQS0034A を発行し、ユーザーにどうするかを尋ねます。構造をコールド・スタートするか、またはこの CQS を取り消すことができます。

構造再作成が開始されたときに構造にアクセスしている CQS がない場合、構造は SRDS および CQS ログからリカバリーされます。リカバリー不能データ・オブジェクト (IMS 高速機能入力メッセージなど) は失われます。データ・オブジェクトは SRDS から読み取られ、新しい構造にコピーされます。その後 CQS は、ログを読み取って、構造を現行ポイントに戻します。これまでに構造チェックポイントが開始されていなかった場合には、そのログには、構造リカバリーに必要なレコードがすべて含まれています。この場合は、構造は CQS ログのみからリカバリーされます。

クライアントは、CQSCONN 要求を使用して、構造再作成中に作業を行うことができるかどうかを指定することができます。構造リカバリーが進行中の間は、CQS は構造に対するすべてのアクティビティーを停止します。これは、CQS 要求が、構造リカバリーがコンプリートされるまで保留されていることを意味します。

CQSCONN 要求を使用して構造に接続するときに WAITRBLD=NO を指定することで、構造再作成の間も CQS 要求を継続させることが可能です。このケースの場合、構造リカバリーはしばらくの間構造アクティビティーを停止させますが、構造はもっと早く使用可能になります。

関連概念:

224 ページの『CQS 構造リカバリー』

CQS 構造のコピー

構造のコピー機能は、計画された再構成または接続逸失などの計画されていないアクティビティーに対して、データ・オブジェクトのすべて（リカバリー可能およびリカバリー不能の両方）を構造から新規構造にコピーします。

構造のコピーを使用して、構造のロケーション、または SIZE、INITSIZE、および PREFLIST などの CFRM ポリシー内で定義されたその他の属性を変更することができます。構造再作成が開始された場合、構造のコピーを実行するためには、その構造へのアクセスを持つ CQS が少なくとも 1 つ存在する必要があります。

CQS における z/OS 構造の二重化

構造の二重化は、キュー構造およびリソース構造の障害のリカバリーに対する、オプションの z/OS 管理プロセスです。このプロセスでは、z/OS は、障害が起こる前に構造の両面コピーを作成し、通常オペレーション時の二重化状態でのそれらの構造を保守します。

キュー構造に障害が起こり、かつ二重化が使用可能な場合、z/OS は、影響を受けていない構造インスタンスに切り替えます。キュー構造に障害が起こり、かつ二重化が使用可能ではない場合、CQS は、最新のチェックポイントおよび z/OS ログ・エントリからのデータを基にした構造を再作成します。障害発生時の二重化キュー構造の利点は、CQS 管理構造再作成のオーバーヘッドを回避できることです。

二重化が使用可能で、かつリソース構造に障害が起こった場合、z/OS は影響を受けていない構造インスタンスへ切り替えます。二重化が使用可能ではなく、かつリソース構造に障害が起こった場合、リソース構造はチェックポイントまたはロギングをサポートしないため、データ・オブジェクトは逸失します。CQS は、制御情報を使用してリソース構造を再移植します。CQS はクライアントに対して、構造を再移植するよう通知します。リソース構造の再移植が必要な場合は、クライアントがそれを行う必要があります。

推奨事項: リソース構造に対する構造の二重化を使用可能にしてください。

構造のインスタンスの両方に障害が同時に起こった場合、構造の二重化は機能せず、すべてのデータ・オブジェクトが逸失します。障害の起こった構造がリソース

構造の場合、CQS クライアントはその構造を再移植する必要があります。障害の起こった構造がキュー構造の場合、CQS は構造再作成を使用して構造をリカバリーします。

構造の二重化はオプションです。これを使用するには、z/OS 二重化機能を使用可能にする必要があります。以下のステップを実行し、この機能を使用可能にしてください。

1. シスプレックスを二重化可能として定義する
2. 以下のパラメーターを、使用している CFRM 結合データ・セット・フォーマット・ユーティリティに追加する ITEM NAME(SMDUPLEX) NUMBER(1)
3. システム管理二重化が CFRM 観点から使用可能になっている環境へマイグレーションする。混乱を引き起こさない CFRM 結合データ・セット・マイグレーションが必要です。システム管理二重化をサポートする、あるレベルでの z/OS システムだけが、二重化が可能なシステム管理 CFRM 結合データ・セットを使用することができます。そのため、以下のステップを実行してください。
 - a. CFRM を使用しているシスプレックス内のすべてのシステムを、システム管理二重化をサポートする z/OS レベルに、徐々にマイグレーションする。
 - b. システム管理二重化可能 CFRM 結合データ・セットをフォーマットし、構成に対する基本および代替 CFRM 結合データ・セットとしてそれらを使用状態にする。

重要: z/OS 二重化を使用可能にした後で、中断なしに下位レベルの CFRM 結合データ・セット (システム管理二重化可能でない結合データ・セット) に戻ることはできません。それを行うためには、システム管理二重化可能データ・セットを使用しているすべてのシステムの、シスプレックス全体にわたる IPL が必要になります。

いったん上位レベル CFRM 結合データ・セットがシスプレックス内で使用状態になると、混乱を起こさずにシステム管理二重化を開始および停止することが可能です。この機能のオン/オフを行うには、たとえば CFRM 結合データ・セットが使用状態の場合でも、CFRM ポリシー DUPLEX パラメーターを変更するか、SETXCF START/STOP,REBUILD,DUPLEX オペレーター・コマンドを使用してください。

特定の構造に対してシステム管理二重化を使用可能にするには、その構造を二重化可能として定義する必要があります。構造を二重化可能として定義すると、その構造は、システム管理の再作成可能としても定義されます。以下のパラメーターを、使用している CFRM アクティブ・ポリシーに追加してください。

DUPLEX (ENABLED)

または

DUPLEX(ALLOWED)

DUPLEX(ENABLED) が CFRM アクティブ・ポリシー内で定義されている場合、システム・プログラマーまたは z/OS は、二重化の再作成を内部で開始することができます。z/OS は、タイマー、または特定のイベント (接続、切断、およびポリシー変更など) の検出を基にした、二重化の再作成の開始をトリガーします。CQS が初期化され、DUPLEX(ENABLED) によって定義された構造に接続する場合、z/OS は二重化の再作成を開始します。

DUPLEX(ALLOWED) が CFRM アクティブ・ポリシー内で定義されている場合、システム・プログラマーは以下のコマンドを使用して、二重化の再作成を初期化する必要があります。

```
SETXCF START,REBUILD,DUPLEX,STRNAME=strname
```

重要: DUPLEX(ENABLED) を使用してオーバーフロー構造を定義する場合、IMS 初期設定はオーバーフロー構造を割り振り、二重化が開始されます。IMS 初期設定が、オーバーフロー構造は必要ではないと判断した場合、構造は削除され、二重化は終了します。この不必要なオーバーヘッドを回避するには、CQS 初期設定時に DUPLEX(ALLOWED) を使用してオーバーフロー構造を定義し、構造がオーバーフロー・モードに入るときに SETXCF コマンドを使用して二重化を開始してください。

二重化が確立されると、構造は無期限にその状態にとどまります。エラー (リンク障害、構造障害、および CFRM ポリシー変更) が発生した場合、二重化は、z/OS によって内部で停止させることが可能です。システム・プログラマーは、以下のコマンドを使用して、二重化を明示的に停止させることができます。ここで KEEP=OLD を指定すると古い構造が保持され、KEEP=NEW を指定すると新規構造が保持されます。

```
SETXCF STOP,REBUILD,DUPLEX,STRNAME=strname,KEEP=OLD/NEW
```

計画された再構成 (CFRM ポリシー変更または保守のためのカップリング・ファシリティー・オフラインの実行など) がサポートされます。構造再作成は、二重化を確立した構造に対しては許可されないため、その二重化を最初に停止させる必要があります。以下のステップを実行してください。

1. 二重化を停止する。

以下のコマンドを実行して、二重化を停止し、構造をシンプレックス・モードに切り替える。

```
SETXCF STOP,REBUILD,DUPLEX,STRNAME=strname,KEEP=OLD/NEW
```

2. 再構成する。

計画された再構成に必要な変更を作成する。

3. 二重化の再作成を開始する。

以下のコマンドを実行し、新規二重化の再作成を開始する。

```
SETXCF START,REBUILD,DUPLEX,STRNAME=strname
```

第 13 章 IMSRSC リポジトリの管理

IMSRSC リポジトリのシステム定義を実行した後は、それを使用して IMS データベース、トランザクション、プログラム、および宛先コードのリソース (および記述子) 定義を保管することができます。

リポジトリには、IMSpIex 内の最大 64 の IMS システムのリソースおよび記述子定義を維持することができます。

IMS は、Resource Manager (RM) のアドレス・スペースを介してリポジトリにアクセスし、保管されているリソース定義の読み取りと書き込みを行います。IMS は、RM リポジトリの読み取り要求を発行して、自動インポート、IMPORT コマンド、および QUERY コマンドの処理中に、保管されているリソース定義を読み取ります。IMS は、RM リポジトリの更新要求を発行して、EXPORT コマンドの処理中に、そのランタイム・リソース定義を、保管されているリソース定義としてリポジトリに書き込みます。

IMS は RM リポジトリの削除要求を発行して、DELETE DEFN コマンドの処理中に、保管されているリソース定義をリポジトリから削除します。

RDDS to Repository ユーティリティ (CSLURP10) に使用して、リソース定義データ・セット (RDDS) の内容を IMSRSC リポジトリにコピーできます。

Repository to RDDS ユーティリティ (CSLURP20) を使用して、IMSRSC リポジトリから RDDS を生成できます。

リポジトリを使用可能にして IMS を開始する前に、リポジトリのシステム定義を実行する必要があります。IMS システムがクローン化されていない場合は、DFSDFxxx メンバーの DYNAMIC_RESOURCES セクションで AUTOEXPORT=AUTO および AUTOEXP_IMSID=THIS_IMS を設定することを検討してください。クローン化されている場合は、AUTOEXPORT=AUTO および AUTOEXP_IMS=ALL_DEFINED を設定することを検討してください。

リポジトリを使用可能にして IMS を開始する前に、リポジトリのシステム定義を実行する必要があります。

Repository Server (RS) アドレス・スペースのバッチ・インターフェース (FRPBATCH) を使用して、リポジトリの管理タスクを実行することができます。また、RS アドレス・スペースに MODIFY (F) *reposervername* コマンドを使用して、いくつかのリポジトリ管理タスクを実行することができます。

以下は、FRPBATCH インターフェースを使用して実行できるいくつかのリポジトリ管理タスクの例です。

- リポジトリを RS カタログ・リポジトリ・データ・セット に追加する
- RS カタログ・リポジトリ・データ・セット からリポジトリを除去する
- リポジトリ・データ・セット・ペアの状況を変更する
- リポジトリを名前変更する

- リポジトリを開始する
- リポジトリを停止する
- リポジトリ・データ・セットを更新する

以下は、MODIFY (F) *reposervername* コマンドを使用して実行できるいくつかのリポジトリ管理タスクの例です。


- リポジトリ・データ・セット・ペアの状況を変更する
- リポジトリを開始する
- リポジトリを停止する
- RS の監査設定を変更する
- セキュリティー製品から変更を取得するために、ストレージ内にあるセキュリティー・テーブルを動的にリフレッシュする
- RS をシャットダウンする

いくつかの MODIFY (F) コマンドと FRPBATCH コマンドは同等です。


表 15. 同等の Modify (F) コマンドと FRPBATCH コマンド

MODIFY (F)	FRPBATCH	注記
--	ADD	
ADMIN DISPLAY	LIST	
ADMIN START	START	
ADMIN STOP	STOP	IMSRSC リポジトリを停止する
--	RENAME	
--	DELETE	
ADMIN DSCHANGE	DSCHANGE	
--	UPDATE	
AUDIT	--	監査レベルを変更する
SECURITY	--	ストレージ内にあるプロファイルをリフレッシュする
SHUTDOWN	--	RS を停止する。z/OS STOP (P) インターフェースを介した STOP コマンドに類似。

関連概念:


 [IMSRSC リポジトリの概要 \(システム定義\)](#)


関連タスク:

 [IMSRSC リポジトリの定義 \(システム定義\)](#)

関連資料:

 [IMPORT DEFN SOURCE\(REPO | RDDS\) コマンド \(コマンド\)](#)

 [EXPORT コマンド \(コマンド\)](#)

 [QUERY コマンド \(コマンド\)](#)

➡ MODIFY reposername コマンド (コマンド)

➡ Repository to RDDS ユーティリティ (CSLURP20) (システム・ユーティリティ)

➡ RDDS to Repository ユーティリティ (CSLURP10) (システム・ユーティリティ)

関連情報:

➡ FRPBATCH でのコマンド (システム・プログラミング API)

RS カタログ・リポジトリ内の IMSRSC リポジトリの指定の更新

IMSRSC リポジトリの名前、自動オープン・オプション、セキュリティー・クラス、およびデータ・セットを更新できます。

リポジトリの名前は、RENAME FRPBATCH コマンドを使用して変更できます。その他のすべてのリポジトリの詳細は、UPDATE FRPBATCH コマンドを使用して変更できます。

注: リポジトリを更新するには、その前に停止する必要があります。

以下に、さまざまな更新操作を実行した場合の影響を示します。

- リポジトリの名前変更:

リポジトリの名前を変更する場合は、新規名でリポジトリを参照するように Resource Manager (RM) を変更する必要があります。新規名でリポジトリを参照するように RM を変更するには、以下のようにします。

1. UPDATE RM TYPE(REPO) REPOTYPE(IMSRSC) SET(REPO(N)) コマンドを発行して、RM がリポジトリを使用しないようにします。
2. QUERY RM TYPE(REPO) SHOW(ALL) コマンドを発行して、すべての RM でリポジトリの使用が不可になっていることを確認します。
3. すべての RM で、新規リポジトリ名を持つように IMS PROCLIB データ・セットの CSLRIxxx メンバーを変更します。
4. UPDATE RM TYPE(REPO) REPO(TYPE(IMSRSC)) SET(REPO(Y)) コマンドを発行して、RM がリポジトリを使用できるようにします。

- UPDATE FRPBATCH コマンドの AUTOOPEN オプションの切り替え:

AUTOOPEN オプションの変更は、次回リポジトリを開始したときに有効になります。

- セキュリティー・クラスの変更:



UPDATE FRPBATCH コマンドに SECURITYCLASS(NULL) を指定すると、リポジトリ定義の既存の値がすべてリセットされます。SECURITY Repository Server (RS) コマンドを使用すると、ストレージ内にあるセキュリティー設定を即時にリフレッシュできます。このコマンドを使用しない場合、新しい値は、次回リポジトリを開始したときに有効になります。

- リポジトリが使用しているデータ・セットの変更:

UPDATE FRPBATCH コマンドの REPDS*n*RID パラメーターと REPDS*n*RMD パラメーターを使用して、IMSRSC リポジトリ・データ・セット の各ペアに対応するデータ・セットの名前を変更することができます。このアクションは、リポジトリ定義内のそのリポジトリ・データ・セットのペアの状況は変更しません。例えば、リポジトリ・データ・セット・ペア 3 の状況が COPY1 のときに、UPDATE FRPBATCH コマンドを使用してリポジトリ索引データ・セット(RID) とリポジトリ・メンバー・データ・セット (RMD) の名前を変更しても、次回リポジトリをオープンしたとき、新規データ・セットはやはり COPY1 として扱われます。したがって、リポジトリがオフラインのときに UPDATE FRPBATCH コマンドを使用してリポジトリ・データ・セットを操作すると決めた場合は、次回リポジトリを開始する前に、データ・セットの実際の状態が、RS カタログ・リポジトリに保管されている状態と互換性があることを確認する必要があります。

重要: リポジトリ・データ・セットを誤って更新すると、データが消失する可能性があります。データ・リカバリーが正常に実行されたことを確認するまでは、変更を予定しているデータ・セットのバックアップを保持しておいてください。







関連概念:

-  [IMSRSC リポジトリの概要 \(システム定義\)](#)
-  [IMSRSC リポジトリ・データ・セットの更新処理中のリカバリー \(オペレーションおよびオートメーション\)](#)

関連タスク:

『RS カタログ・リポジトリからの IMSRSC リポジトリの除去』

関連資料:

-  [QUERY RM コマンド \(コマンド\)](#)
-  [UPDATE RM コマンド \(コマンド\)](#)
-  [IMS PROCLIB データ・セットの CSLRIxxx メンバー \(システム定義\)](#)
-  [FRPBATCH での UPDATE コマンド \(システム・プログラミング API\)](#)
-  [FRPBATCH での RENAME コマンド \(システム・プログラミング API\)](#)
-  [F reposervername,SECURITY \(コマンド\)](#)

RS カタログ・リポジトリからの IMSRSC リポジトリの除去


IMSRSC リポジトリを Repository Server (RS) のカタログ・リポジトリから除去するには、リポジトリの使用を停止するための手順を実行し、DELETE FRPBATCH コマンドを実行します。


リポジトリが RS カタログ・リポジトリに追加された後に名前変更されているため、そのリポジトリを除去し、この IMS の IMS リポジトリ機能を使用し続けたい場合は、ステップ 1 を実行しないでください。

1. リポジトリの使用を停止するための手順を実行してください。


2. DELETE FRPBATCH コマンドを実行して、RS カタログ・リポジトリからリポジトリを除去します。

関連概念:


 [IMSRSC リポジトリの概要 \(システム定義\)](#)

 [IMS リポジトリの索引データ・セットとメンバー・データ・セット \(システム定義\)](#)

関連タスク:

 [IMSRSC リポジトリの使用からのフォールバック \(システム定義\)](#)

関連資料:

 [FRPBATCH での DELETE コマンド \(システム・プログラミング API\)](#)


IMSRSC リポジトリの定義と状況の表示

IMSRSC リポジトリに関する情報を表示するには、FRPBATCH LIST コマンドのいずれか、または `F reposervername,ADMIN` コマンドを使用します。

Repository Server (RS) カタログ・リポジトリにリストされているリポジトリについて、以下の情報を表示できます。

- リポジトリの名前
- リポジトリが開始されているか、停止されているか、またはクローズされているか
- リポジトリが最後に更新された時刻、および誰が更新したか
- RS によって管理されているリポジトリ名を表示するには、MODIFY コマンドの `F reposervername,ADMIN DISPLAY` を使用します。
- 特定のリポジトリの詳細を表示するには、FRPBATCH コマンドの LIST REPOSITORY、または MODIFY コマンドの `F reposervername,ADMIN,DISPLAY(repositoryname)` を使用します。
- RS カタログ・リポジトリ内のすべてのリポジトリの詳細を表示するには、FRPBATCH コマンドの LIST STATUS、または MODIFY コマンドの `F reposervername,ADMIN` を使用します。

関連概念:

 [IMSRSC リポジトリの概要 \(システム定義\)](#)

関連資料:

 [F reposervername,ADMIN \(コマンド\)](#)

 [FRPBATCH での LIST コマンド \(システム・プログラミング API\)](#)

IMSRSC リポジトリを含む IMS の初期化

IMS の初期化中、IMS は IMS PROCLIB データ・セットの DFSDFxxx メンバーの REPOSITORY セクションを読み取ります。

IMS が DRD で使用可能になっていない (MODBLKS=OLC が DFSDFxxx メンバーの Common Service Layer セクションまたは DFSCGxxx メンバーに指定されている) か、IMS が RMENV=N で定義されている (IMS は RM サービスを使用していない) 場合、REPOSITORY セクションは無視されます。DFS4402W メッセージが出され、REPOSITORY セクションが無視されることを示します。

TYPE=IMSRSC が CSLRIxxx メンバーに指定されている場合、IMS は IMSRSC リポジトリの Resource Manager (RM) への接続を試行します。RM が、IMS が要求するリポジトリ・タイプを管理するように定義されている場合は、RM のリポジトリ接続要求に対して正常な戻りコードが返されます。IMS がリポジトリの RM に正常に接続すると、DFS4404I メッセージが出されます。

RM が、IMS が要求しているリポジトリ・タイプに対して使用可能になっていない場合は、RM のリポジトリ接続要求に対してエラー戻りコードが返されます。IMS はメッセージ DFS4400E を出し、IMS の初期化は異常終了コード 0400 によって終了します。RM からのその他のエラー戻りコードおよび理由コードがある場合は、RM 戻りコードおよび理由コードが含まれた DFS4401E メッセージが出されます。DFS4401E エラー・メッセージの後、IMS の初期化は異常終了コード 0400 によって終了します。

AUTOIMPORT=AUTO が指定され、DFSDFxxx メンバーに REPOSITORY セクションが定義されている場合は、IMS の初期化中、DFS3499I ではなく、DFS3409I メッセージが出されます。これは、IMS がまだリポジトリに接続していないためです。DFS3499I メッセージはコールド・スタート中に出され、リソース定義のインポート元を示します。

IMS がリポジトリで使用可能になっており、AUTOIMPORT=AUTO が DFSDFxxx メンバーに指定され、保管されている IMS のリソース定義がリポジトリに含まれている場合は、自動インポートがリポジトリからのものであることを示す DFS3399I メッセージが出されます。


リポジトリに、IMS の保管されたリソース定義が含まれていない場合は、リソース定義データ・セット (RDDS) または MODBLKS データ・セットが使用されていることを示す DFS3399I メッセージが出されます。


リポジトリが空の場合は、リポジトリからリソースがインポートされていないことを示す DFS4405W メッセージが出されます。

AUTOIMPORT=REPO または RDDS あるいは MODBLKS が指定されている場合は、IMS の初期化中に DFS3499I メッセージが出されます。



コールド・スタートまたは IMPORT コマンドの処理中に、保管されているリソース定義がリポジトリからインポートされると、DEFNTYPE が IMPORT に設定されます。

関連概念:

 IMSRSC リポジトリの概要 (システム定義)

 DFS メッセージ (メッセージおよびコード)

関連資料:

-  IMS PROCLIB データ・セットの DFSDFxxx メンバー (システム定義)
-  IMS PROCLIB データ・セットの DFSCGxxx メンバー (システム定義)

IMSRSC リポジトリの CSL RM 管理

Common Service Layer (CSL) Resource Manager (RM) は、Repository Server (RS) のアドレス・スペースと対話して、IMSRSC リポジトリ内に保管されているリソース定義を管理します。


RM は、RM の初期化中または UPDATE RM コマンドの処理中に、Repository Server に登録し、IMSRSC リポジトリに接続します。RM がリポジトリ構造とともに定義されている場合、RM はそのリソース構造を使用して、リポジトリ名、リポジトリ・タイプ、監査アクセスの値、および RS z/OS システム間カップリング・ファシリティ (XCF) グループ情報などのリポジトリ情報を保管します。このリソース構造により、IMSplex 内のすべての RM が同じリポジトリを使用することが保証されます。

リポジトリを使用可能にして RM を開始する前に、リポジトリのシステム定義を実行する必要があります。特に、RM リポジトリ・パラメーターは、IMS PROCLIB データ・セットの CSLRIxxx メンバーに定義されます。

UPDATE RM コマンドは、RM によるリポジトリの使用を動的に使用可能にするために使用します。


QUERY RM コマンドは、RM アドレス・スペースによって管理されているリポジトリの状況に関する情報を取得するために使用します。

関連概念:

-  IMSRSC リポジトリの概要 (システム定義)


428 ページの『CSL RM アドレス・スペースでの IMSRSC リポジトリのセキュリティの設定に関する考慮事項』


関連タスク:

-  IMSRSC リポジトリの定義 (システム定義)

関連資料:

201 ページの『CSL RM により管理される情報』

-  QUERY RM コマンド (コマンド)

-  UPDATE RM コマンド (コマンド)

IMSRSC リポジトリを含む CSL RM の初期化

Resource Manager (RM) の初期化中、RM は IMS PROCLIB データ・セットの CSLRIxxx メンバーを読み取って、REPOSITORY セクションを処理します。

CSLRIxxx メンバーに REPOSITORY セクションを定義することにより、RM システムによって管理される IMSRSC リポジトリ名とリポジトリ・タイプを定義することができます。REPOSITORY セクションが定義されている場合、RM は、RM

の初期化中に自動的に Repository Server (RS) アドレス・スペースに登録します。RM は、CSLRIxxx メンバーに定義された名前でもリポジトリに接続します。

RM の初期化中に RS REGISTER または CONNECT 要求でエラーが発生すると、RS からの戻りコードまたは理由コードとともに CSL2510E または CSL2511E エラー・メッセージが出されます。さらに、CSL2502A または CSL2503A メッセージが出されます。RM は、初期化を完了せず、作業の受け入れ準備が整ったことを示す CSLSCRDY 要求を SCI に出しません。IMSpIex 内のすべての RM が RS またはリポジトリを待機している場合、初期化中で、RM サービスを使用している IMS 制御領域は、DFS3306A メッセージにより RM を待機します。アクティブな IMS は、RM サービスが使用不可なことを示す DFS3306A メッセージを出します。

RM 初期化中の CSL2510E メッセージの後には、RM が RS アドレス・スペースの開始を待機中であることを示す CSL2502A メッセージが出されます。RM は、RS への登録に成功するか終了されるまで、5 秒おきに RS への登録を試行します。RM は、CANCEL コマンドを使用して終了する必要があります。RM は、SCI に登録されていないので、SCI SHUTDOWN 要求を使用して終了することはできません。リポジトリの REGISTER および CONNECT は RM 初期設定の初期段階に含まれており、MVS STOP コマンドを処理するためのスレッドがまだセットアップされていないため、MVS STOP コマンドを使用して RM を終了することはできません。RM が RS への登録に成功すると、強調表示されている CSL2502A メッセージが削除されます。

RM 初期化中の CSL2511E メッセージの後には、RM が、CSL2503A メッセージに指定された名前とタイプを使用してリポジトリへの接続を待機していることを示す CSL2503A メッセージが表示されます。RM は、リポジトリへの接続に成功するか終了されるまで、5 秒おきにリポジトリへの接続を再試行します。RM は、CANCEL コマンドを使用して終了する必要があります。RM は、SCI に登録されていないので、SCI SHUTDOWN 要求を使用して終了することはできません。リポジトリの REGISTER および CONNECT は RM 初期設定の初期段階に含まれており、MVS STOP コマンドを処理するためのスレッドがまだセットアップされていないため、MVS STOP コマンドを使用して RM を終了することはできません。RM がリポジトリへの接続に成功すると、強調表示されている CSL2503A メッセージが削除されます。

CSLRIxxx メンバーのリポジトリ・セクションの構文解析エラーが発生すると、CSL25xx エラー・メッセージが出されます。これが、IMSpIex 内で開始されている最初の RM の場合 (RM にリソース構造があるかないかに関係なく)、RM は、リポジトリが使用可能になっていない状態で起動します。CSLRIxxx メンバーのエラーが修正されると、UPDATE RM コマンドを使用して、RM でリポジトリを使用可能にすることができます。

リソース構造ありで RM が開始されており、IMSpIex でリポジトリが使用可能になっている場合、リポジトリ名のエラーが発生すると CSL25xx エラー・メッセージが出され、その後、リソース構造から読み取られた名前を持つリポジトリに RM が接続されます。

RM の開始時に、CSLRIxxx メンバーとリソース構造で XCFGROUP 名の不一致があった場合、RM は異常終了し、異常終了コード 0010 とサブコード X'00000634' が出されます。

RM の開始時に、RM の初期化モジュールがストレージを取得できないと、RM は異常終了し、異常終了コード 0010 とサブコード X'00000635' または X'00000636' が出力されます。

CONNECT 要求によってリポジトリに接続する最初の RM が、リポジトリで使用される索引フィールドとキー・フィールドを定義します。後続の RM は、そのリポジトリの索引フィールドとキー・フィールドの情報を使用します。RM がリポジトリに正常に接続すると、CSL2500I メッセージが出力されます。さらに、RM がリポジトリのキー・フィールドと索引フィールドを初期化し、リポジトリが空の場合、CSL2501I メッセージが出力されます。

RM がリソース構造を使用している場合は、最初の RM が、CSLRIxxx メンバーからのリポジトリ名、リポジトリ・タイプ、および監査アクセスの情報をリソース構造に保管します。後続の RM は、リソース構造からリポジトリ名、リポジトリ・タイプ、および監査アクセスの情報を取得します。リソース構造内の情報と RM の CSLRIxxx メンバー内の値の間で不一致がある場合は、その不一致を示す CSL2512W メッセージが表示されます。RM は、リソース構造から取得したリポジトリ名とタイプに接続します。




RM がリソース構造に接続できない場合、CSL2515E メッセージがエラー戻りコードおよび理由コードとともに出力されます。RM は、リポジトリ・サービスなしで処理を続行します。

エラーが解決された後は、UPDATE RM コマンドを使用して動的にリポジトリを使用可能にできます。


RM がリソース構造を使用していない場合、RM は常に、CSLRIxxx メンバーに指定された名前のリポジトリに接続します。

RM がリソース構造を使用していない環境では、RM の再始動と再始動の間に、CSLRIxxx メンバー内のリポジトリ名を変更しないでください。





関連概念:

-  IMSRSC リポジトリの概要 (システム定義)
-  DFS メッセージ (メッセージおよびコード)
-  CSL SCI 要求 (システム・プログラミング API)

関連タスク:

-  IMS リソース・グループの動的定義の使用可能化 (システム定義)

関連資料:

-  UPDATE RM コマンド (コマンド)
-  IMS PROCLIB データ・セットの CSLRIxxx メンバー (システム定義)
-  CSL メッセージ (メッセージおよびコード)
-  RM 異常終了コード (メッセージおよびコード)

CSL RM、IMS、および Repository Server の終了

IMS または Repository Server (RS) が終了すると、Resource Manager (RM) は、コミットされていない処理中の作業単位をすべてキャンセルします。RM は終了すると、RM が管理しているすべてのリポジトリから切断し、RS アドレス・スペースからの登録を解除します。


RM は、Structured Call Interface (SCI) 通知出口を使用して、通常の IMS 終了および異常な IMS 終了をすべて listen します。処理中の作業は、未確定状態になる場合があります。RM は、終了する IMS に対する、コミットされていない、処理中の作業単位をすべてキャンセルします。RM が作業単位をコミットした後だが、IMS に応答する前にエラーが発生した場合、その作業単位はコミットされます。

作業の処理中に RS が終了した場合は、RS 終了イベントにより RM 登録出口が実行されます。処理中の作業がある場合、その作業は未確定状態になる可能性があります。処理中の作業がコミットされていない場合は、RS によってキャンセルされます。COMMIT 後だが、RM に応答する前に RS が終了した場合、その作業単位はコミットされます。


未確定を解決するには、QUERY SHOW(DEFN) コマンドを使用して、処理中の作業の状況を判別する必要があります。IMSRSC リポジトリの作成または更新のタイム・スタンプ、または IMS のインポートのタイム・スタンプを使用して、処理中の作業の状況を判別することができます。作業単位がコミットされていない場合は、EXPORT、IMPORT、または DELETE DEFN コマンドを再発行する必要があります。

RM の終了により、RM が管理しているすべてのリポジトリから切断され、RS アドレス・スペースから登録解除されます。

関連概念:


 [IMSRSC リポジトリの概要 \(システム定義\)](#)


関連タスク:

 [CSL RM のシャットダウン \(オペレーションおよびオートメーション\)](#)

関連資料:

 [IMPORT DEFN SOURCE\(REPO | RDDS\) コマンド \(コマンド\)](#)

 [EXPORT コマンド \(コマンド\)](#)

 [QUERY コマンド \(コマンド\)](#)

第 14 章 IMS サービスの考慮事項

IMS にサービスを実装するために適切な計画を立てることは、IMS を使用可能で回復力のある状態に保つために極めて重要です。

サービス SMP/E SYSMOD のタイプ

IMS は、SMP/E フォーマットで、保守パッケージを提供します。

IMS の保守は、次の SMP/E SYSMOD の 3 タイプのいずれかでパッケージされます。

- プログラム一時修正 (PTF)

プログラム一時修正 (PTF) は通常、予防サービスと考えられます。PTF は問題のソリューションを格納しており、機械可読フォーマットで配布されます。PTF は、IMS リリースの問題用に提供される、ソリューションの最終手段ととらえられます。

モジュールの以前のレベルに置き換わるモジュールの場合、ソース変更はそのモジュールに対する累積デルタ・ソース変更になります。PTF に前提 PTF がある場合は、その PTF に組み込まれるソース変更は累積されたものではなく、その PTF に対して変更されたコードのみを反映しています。

- プログラム診断依頼書 (APAR)

プログラム診断依頼書 (APAR) 修正は、修正サービスととらえることができます。APAR は問題のソリューションを格納しており、機械可読フォーマットで配布されます。APAR は、問題に対する暫定ソリューション または一時的ソリューションとみなされます。最終的なソリューションは、対応する PTF か、APAR プロセスの終わりに作成された PTF です。APAR は、1 つ以上の PTF になることができます。

- USERMOD

IMS は、以下のような状態で USERMODS を提供します。

- APAR FIXTEST として。APAR に報告された問題が修正されたことを確認する、または APAR か PTF が使用可能になるまでのリリースとなる。
- 問題の回避策として。最後の APAR または PTF が使用可能になるまでのリリースとなる。
- 問題を理解する上で必要な詳細資料や情報 (ダンプのような) を得るためのトラップ (または特殊コード) として。

関連資料: SMP/E の詳細については、「SMP/E 解説書」を参照してください。

サービス **SYSMOD** のパッケージ化

IMS サービスのプロセスでは通常、APAR の完了直後にそれらを使用可能にします。これは通常、対応する 1 つの PTF または複数の PTF が完了する数週間前になります。APAR が完了し、PTF が使用可能になるまでの間に緊急の修正が必要な場合は、短期的なソリューションとして APAR を使用してください。

IMS に提供される APAR は、APAR とコードの従属関係にある SYSMODS のみを前提条件 (PRE、IFREQ など) として定義します。IMS APAR は、前提条件として、対応する APAR (PTF ではなく) をリストします。SMP/E により IMS APAR が処理されると、毎回回帰メッセージが表示される可能性があります。このメッセージが表示される場合は、それを分析し、実際に回帰が起こっていないことを確認する必要があります。

PTF は前提条件 (PRE、IFREQ など) として、同じエレメントに影響を与える以前の PTF をすべて含みます。APAR を処理する場合と違い、PTF を処理するには他の多くの SYSMODS を処理する必要がある場合があります。問題が発生し、緊急に解決する必要がある場合は、短期的なソリューションとして APAR を使用してください。これは、APAR には最低限の変更しか必要ないためです。ただし、最終修正には、必ず PTF を使用してください。

APAR を処理する場合は、通常回帰メッセージが出されます。このメッセージが出される場合は、それを分析し、実際に回帰が起こらないことを確認する必要があります。支援が必要であれば、IBM ソフトウェア・サポートに連絡してください。

PTF は、対応する APAR を取り替えます (SUP)。このため、PTF の処理前に APAR を除去する必要はありません。

重要: APAR および USERMOD 修正の処理に、SMP/E ACCEPT コマンドを使用しないでください。対応する 1 つの PTF または複数の PTF は、最終修正として処理してください。

IMS に提供される USERMOD は、USERMOD とコードの従属関係にある SYSMOD のみを前提条件 (PRE、IFREQ など) として定義します。USERMOD は前提条件として、PTF ではなく、対応する APAR をリストします。この意味では、USERMODS は APAR に似ています。SMP/E により IMS USERMODS が処理されると、毎回回帰メッセージが表示される可能性があります。このメッセージが表示される場合は、それを分析し、実際に回帰が起こっていないことを確認する必要があります。支援が必要であれば、IBM ソフトウェア・サポートに連絡してください。

IMS によって提供される USERMOD は、対応する APAR もしくは PTF によって置き換え (SUP) られません。最終修正が可能な場合は、システムから USERMOD を RESTORE してください。各 USERMOD ごとに出される ++HOLD 情報は、USERMOD を RESTORE するよう要求し、その方法を示します。

IMS サービスの入手

IMS サービスを入手するための推奨される方法は、推奨サービス・アップグレード (RSU) をオーダーすることです。RSU には、Consolidated Service Test (CST) に合格したすべての現行 PTF が含まれています。

CST は、z/OS プラットフォームに關与している製品のすべての現行 PTF を、まとめてテストする、z/OS プラットフォーム用の追加サービス・テスト環境です。これにより、IBM では、z/OS および主要なサブシステム用の PTF サービスを、1 つの RSU でまとめて推奨できます。

IMS サービスをオーダーするには、ShopzSeries (<https://www14.software.ibm.com/webapp/ShopzSeries/ShopzSeries.jsp>) にアクセスします。

ShopzSeries オプションの 1 つに、ご使用の SMP/E Consolidated Software Inventory (CSI) の現行の内容に基づいたカスタム・オーダーの作成があります。このオプションを使用する場合、ShopzSeries はご使用の環境内にある既存の機能、FMID、および PTF についての情報を必要とします。この情報は、SMP/E GIMXSID サービス・ルーチンを使用して生成できます。GIMXSID サービスは、指定された SMP/E CSI 内の指定されたターゲット・ゾーンおよびグローバル・ゾーンを検査して、ユーザーがすでにお持ちの機能およびサービスを判別します。ShopzSeries Web サイトには、サービス・オーダー処理の手引きとなる詳細な説明、サンプル JCL、およびビデオ・クリップが用意されています。

これまでと同様に、以下の経路から PTF を入手することもできます。ただし、これらの PTF は CST プロセスを経っていない場合があることに留意してください。

IBM ソフトウェア・サポート

特定の PTF を要求できます。それらは、ファイル転送プロトコル (FTP) サイトである IBMLINK からダウンロードできるほか、カートリッジでの郵送も可能です。

拡張サポート・オファリング (ESO)

ライセンス交付を受けたユーザーは、要求すれば ESO テープを入手できます。

カスタム・ビルド製品デリバリー・オファリング (CBPDO)

CBPDO サービス・テープは、お客様から要求があったときに作成されます。

ServerPac

ServerPac オーダーは、要求があったときに送付されます。製品のほかに、ServerPac オーダーには製品に取り込まれた PTF も含まれています。

ShopzSeries

サービスをより簡単にオーダーできるようにする Web ベースの生産性向上ツールです。サービスは、メールまたはインターネットを通じてユーザーに送付されます。

保守に関する推奨

以下のトピックで概説する推奨事項は、ご使用の環境で功を奏する保守戦略の開発を支援するように設計されています。

保守のインストールの準備状況の査定

保守をインストールする前、さらに言えば新規 IMS システムをインストールする前に、準備状況を判別する必要があります。これにはリスクを注意深く査定することも含まれます。これと同じ方針が、保守計画の作成についても適用されます。

保守のインストールの準備状況の査定には、ローカル・テスト環境の品質やビジネス・サイクルなど、数多くの要因が関与します。

- ローカル・テスト環境の品質

テスト環境の品質には、いくつかの要因が影響を及ぼします。考慮する必要のある問題は次のとおりです。

- RACF、DFP、または z/OS DFSMS などのソフトウェア製品が、テスト環境でも実動環境と同じように使用されており、レベルは、実動で使用されるレベルに一致するかどうか。
- Teleprocessing Network Simulator (TPNS) のように、ストレス・テストを実行してピーク時の実動アクティビティをシミュレートできるツールを持っているかどうか。
- テスト環境で使用するハードウェアが、どの程度まで実稼働環境に即したもののか。
- テスト環境で使用するアプリケーション・ソフトウェアが、どの程度まで実稼働環境に即したもののか。
- テスト結果が綿密にモニターされるか。

テスト環境の品質を評価するには、上記およびその他の要因を考慮に入れる必要があります。

- ビジネス・サイクル

重要なビジネス・サイクルの間に行えることはすべて行って、IMS が常に使用可能な状態にあるようにします。したがって、重要なビジネス・サイクルの間は、実動システムへの保守の実装は避けてください。

新規または移行済みの IMS システムの保守サービス・レベルの更新

IMS システムのインストール後は、新規システムの場合も既存の IMS システムの新規リリースの場合も、新システムのサービス・レベルを、許容可能な現行性のレベルにする必要があります。そのためには、サービス・レベルおよび PTF をインストールし、PE PTF を解決して、テスト・サイクルを実行します。

重要: IMS システム定義が原因で、APPLY のみの状態にある SYSMOD に回帰が起こる可能性があります。これを防ぐ方法については、253 ページの『IMS システム定義が原因で APPLY のみの状態の SYSMOD に回帰が起こるのを防ぐ』を参照してください。

注: /DIAGNOSE SNAP MODULE(modulename) コマンドを使用して、特定のモジュールの現行保守レベルを判別できます。

IMS 実動システムの基本実装サービス・レベルから始めようとしている場合は、以下のアクションが推奨されます。

1. IMS システムを実動用に実装する予定の 6 カ月前に、使用可能なすべての PTF をインストールします。
2. IMS システムを実動用に実装する予定の 3 カ月前に、HIPER APAR のすべての PTF 修正をインストールします。
3. PE を解決します。IMS システムにインストールするすべての PE PTF を評価し、それらのご使用の IMS 環境に影響を及ぼすかどうかを判別してください。影響がある場合は、その PE PTF の修正をインストールするか、その PE PTF を削除するか、または IBM ソフトウェア・サポートに連絡して支援を受けます。
4. IMS システムを実動用に実装する前に、3 カ月のテスト・サイクルを設けることを推奨します。

IMS 保守はテスト・サイクルの期間中もその後も絶えず配布されるので、できるだけ早く新規システムに、実動システムのサービス・レベルの継続的保守手順を組み込んでください。使用可能な手順の例については、244 ページの『保守の取得とインストール』を参照してください。

既存の IMS 実動システムのサービス・レベルの保守

IMS システムのインストールとテストが終わり、作動可能になったら、以下にリストする推奨事項に従って IMS システムのサービス・レベルの保守を行います。

- z/OS プラットフォーム上で実行するすべての製品に関する推奨事項
 - 年 4 回、RSU 保守をアップグレードする。
 - HIPER、PE、SEC/INT、および FIXCAT を定期的にモニターし、必要に応じて毎週または毎月、重要な修正をインストールする。
- IMS 実動システムに固有の推奨事項
 - 検出された重要なソフトウェアの問題に関する、十分にテストされた修正をインストールする。¹
 - ご使用の特定の IMS 環境に直接適用可能な、十分にテストされた重要な HIPER SYSMOD をインストールする。
 - ご使用の特定の IMS 環境に直接適用可能な PE PTF を解決する十分にテストされた重要な SYSMOD をインストールする。
 - 当該の IMS リリース・レベル用の IMS PSP バケット UPGRADE と、SUBSET FMID を調べる。これらのバケットには、重要な IMS 製品関連情報が頻繁に追加されます。バケットによってはアクションが必要になることもあります。

1. 修正の重要度は担当ショップが判別します。修正をインストールしなかった場合に予想される結果を担当ショップが許容できない場合、その修正は重要です。

保守の取得とインストール

以下の手順例は、ご使用の IMS インストールに固有の保守手順を実装するためのモデルとして使用できます。

この保守手順は、242 ページの『新規または移行済みの IMS システムの保守サービス・レベルの更新』および 243 ページの『既存の IMS 実動システムのサービス・レベルの保守』で概説された推奨に基づいています。

追加情報が記載されている類似の手順例については、通知 APAR II13024を参照してください。

重要: 適用 (APPLY) されるが受け入れ (ACCEPT) はされない SYSMOD は、システム定義プロセスにより回帰する場合があります。SYSMOD の回帰を避けるには、253 ページの『IMS システム定義が原因で APPLY のみの状態の SYSMOD に回帰が起こるのを防ぐ』を参照してください。

1. 現行のサービス・デリバリー方式 (Shopz など) を使用して、現行のサービスを取得します。

Shopz は、IBM の Web ベースの生産性向上ツールで、これを使用すると、System z ソフトウェア・プロダクト、プロダクト・アップグレード、およびシステム保守を簡単にオーダーできます。詳しくは、Shopz を参照してください。

2. 現行の z/OS 用拡張 HOLDDATA を取得して受け取ります。

最新の詳細情報については、Enhanced HOLDDATA z/OSを参照してください。

3. SMP/E を使用してサービスをインストールします。
 - SOURCEID(RSU $yymm$) を指定して、Consolidated Service Test (CST) を受け渡したサービスを組み込みます。詳しくは、Consolidated Service Test and the RSUを参照してください。
 - ご使用の IMS システムにインストールされている PE PTF のうち、対応フィックスがインストールされていないものをすべて評価して、それらの PE PTF が IMS 環境に影響を与えるかどうかを判別します。影響を与える場合は、その PE PTF のフィックスをインストールするか、PE PTF を削除するか、または IBM サポートに連絡して支援を受けます。
 - 処理中の PTF 内のシステム HOLD を解決します。
4. z Systems Security Portal から情報を取得します。z Systems Security Portal へのアクセス手順については、http://www.ibm.com/systems/z/solutions/security_subintegrity.htmlを参照してください。
5. 現行の z/OS 用拡張 HOLDDATA を再度取得して RECEIVE を実行し、前回の確認以降に変更された HIPER APAR および PE PTF エクスポージャーに確実に対応します。
6. SMP/E REPORT ERRSYSMOD コマンドを実行して、HIPER/PE/SEC エクスポージャーを識別します。出力を分析して、必要に応じて以下のアクションを実行します。
 - a. 適用できる SYSMODS を取得します。
 - b. 環境に適用できる SYSMOD を処理します。

IMS サポートの website サイトまたは PSP バケットを使用して、APAR に関する説明を入手してください。

すでにシステム上にある PE SYSMOD についての一連のアクションには次のものがあります。

- エラーが発生した PTF が受け入れ済みになっていない場合は、その PTF を除去します。
- 報告された PE 症状が重大でない場合は、PTF をそのまま残します。
- エクスポージャーが解決されるまではそのエクスポージャーが検出されないように、手順を設定します。
- 修正用の APAR/PTF フィックスが使用可能な場合は、それをインストールします。
- 当該の APAR がまだ公開されている場合は、報告された問題に対応する FIXTEST を IBM ソフトウェア・サポートに要求します。
- 報告された問題に対応する USERMOD コード・バイパスを IBM ソフトウェア・サポートに要求します。
- 必要に応じて、IBM ソフトウェア・サポートに連絡して支援を受けてください。

7. IMS PSP バケットを調べて、新規サービス情報を確認します。

8. 新しい保守レベルをテストします。

9. 新しい保守レベルを実装します。

上記の手順のステップ 4、5、6、7、および 8 を継続的に繰り返して、保守レベルを最新に維持します。最小限でも、テスト環境、開発環境、実稼働環境などの新規環境に IMS システムをマイグレーションする直前には、これらのステップを実行してください。

単一システム上の IMS サービスのインストール

IMS サービスは、以下の SMP/E メソッドを含むいくつかの方法でインストールできます。

サービスをインストールするための SMP/E メソッドには次のものがあります。

- RECEIVE/APPLY/ACCEPT (標準的なシーケンス)
- APPLY なしの ACCEPT (事前生成モード)
- APPLY 前の ACCEPT (システム定義が必要なサービス用)

重要:

- APAR または USERMOD を ACCEPT しないでください。
- サービスをインストールする前に、最新の HOLDDATA 情報を読んでいることを確認してください。

これらのプロセスについて不明な点がある場合は、処理を始める前に IBM ソフトウェア・サポートにお問い合わせください。

関連タスク:

252 ページの『IMSplex への IMS サービスのインストール』

RECEIVE/APPLY/ACCEPT (標準的なシーケンス)

この SMP/E メソッドは、サービスを処理するための推奨される標準的な方法です。

1. IMS 環境をバックアップします。
 - a. SMP/E データ・セット (Zones、SMPMTS、および SMPPTS など) をバックアップします。
 - b. IMS プロダクト・データ・セット (SDFSRESL および ADFSLOAD など) をバックアップします。
2. 該当サービス入手します。
3. パッケージに付随している資料を読みます。
 - ESO テープ資料
 - CBPDO のプログラム資料説明書補足
 - 予防サービス計画 (PSP)
4. SYSMODS と HOLDDATA の両方のパラメーターを設定して、SMP/E RECEIVE コマンドを実行します。
5. SMP/E APPLY CHECK GROUPEXTEND コマンドを実行します。

APPLY CHECK および APPLY プロセス中に、ターゲット・システムに含まれないモジュールに対して SMP/E メッセージ GIM43401 と GIM44402 を受け取ることができます。使用しない IMS 機能に関する部分を参照するメッセージについては、無視することができます。プログラミング例外 (PE) を解決し、サービスが要求レベルまで処理されているか確認します。支援が必要であれば、IBM ソフトウェア・サポートに連絡してください。

重要: IMS サービスには、インライン JCLIN 情報が含まれていることがよくあります。このタイプのサービスの場合、SMP/E は REDO パラメーターによる再 APPLY をお勧めしません。このタイプのサービスに REDO が使用されている場合 (NOJCLIN なしで)、SMP/E RESTORE 処理が正常に動作しない可能性があります。

6. APPLY CHECK 報告書をチェックし、必要であれば変更します。
7. SMP/E APPLY GROUPEXTEND コマンドを実行します。
8. 修正サービスをテストします。

APPLY のみの状態にあるサービスがある状況で IMS システム定義を行った場合、そのサービスに部分的または完全な回帰が起こる可能性があります。

推奨事項: IMS システム定義を行った後は毎回、APPLY のみの状態にあるすべての SYSMOD について、以下の SMP/E コマンドを発行してください。ここで、xxxx,xxxx は APPLY のみの状態にあるすべての SYSMOD のリストを示します (コンマかスペースで区切ります)。

```
APPLY S(xxxx,xxxx) REDO NOJCLIN BYPASS (...)
```

重要: SYSGEN を必要とする保留や DELETE のための保留を含む、APPLY のみの状態の SYSMOD には、特別な処理が必要な場合があります。

9. SMP/E ACCEPT CHECK GROUPEXTEND コマンドを実行します。
10. ACCEPT CHECK 報告書をチェックします。

11. SMP/E ACCEPT GROUPEXTEND コマンドを実行します。
12. RECEIVE コマンドから Enhanced Holddata の新規コピーをダウンロードします。
13. SMP/E REPORT ERRSYSMODS コマンドを実行します。
14. SMP/E REPORT ERRSYSMODS コマンドの出力を分析し、必要に応じて追加サービスを処理します。

APPLY なしの ACCEPT (事前生成モード)

この SMP/E メソッドは、サービスを処理するための事前生成の方法です。

重要: ここに記載されているのは、本書発行時の情報です。それ以後の情報や、詳細については、APAR 情報 II13024 を参照してください。

以下の手順を行うには、FMID が ACCEPT されたときに配布ゾーンで ACCJCLIN が設定されていたことが必要です。

1. IMS 環境をバックアップします。
 - a. SMP/E データ・セット (Zones、SMPMTS、および SMPPTS など) をバックアップします。
 - b. IMS プロダクト・データ・セット (SDFSRESL および ADFSLOAD など) をバックアップします。
2. 該当サービスを手に入れます。
3. パッケージに付随している資料を読みます。
 - ESO テープ資料
 - CBPDO のプログラム資料説明書補足
 - 予防サービス計画 (PSP)
4. SYSMODS と HOLDDATA の両方のパラメーターを設定して、SMP/E RECEIVE コマンドを実行します。
5. IMS 配布ゾーンおよび IMS ターゲット・ゾーンにあるすべての製品のすべての未解決サービスに対して SMP/E RESTORE または ACCEPT コマンドのいずれかを実行します。以下のサンプル SMP/E 制御ステートメントを使用して、この未解決のサービス (適用のみ行われた SYSMODS) を識別してください。

```
//SMPCNTL DD *
      SET BDY(targetzonename).
      LIST APAR PTF USERMOD NOACCEPT NOSUP.
```

6. SMP/E UNLOAD コマンドを使用して、ターゲット・ゾーン DDDEF をアンロードします。以下のサンプル SMP/E 制御ステートメントと JCL (ジョブ制御言語) を使用して、このタスクを完了してください。

```
//SMPCNTL DD *
      SET BDY(targetzonename).
      UNLOAD DDDEF.
//SMPPUNCH DD DSN=IMS.SMPUNLD,DISP=(,CATLG),
// UNIT=SYSDA,SPACE=(TRK,(5,1),RLSE),
// DCB=(RECFM=FB,BLKSIZE=16000,LRECL=80)
```

7. ターゲット・ゾーンに対して SMP/E LIST コマンドを実行して、OPTIONS 項目の名前を判別します。この OPTIONS 項目は、ステップ 10 で使用します。以下のサンプル SMP/E 制御ステートメントを使用して、このタスクを完了してください。

```
//SMPCNTL DD *
      SET BDY(targetzonename).
      LIST TARGETZONE.
```

8. 以下のデータ・セットをスクラッチして再割り当てします。

- SMPMTS
- SMPSTS
- SMPSCDS
- SMPLTS

注: SMPPTS、SMPSTS、および SMPSCDS データ・セットについては、スクラッチして再割り当てする代わりに SMP/E CLEANUP コマンドを使用できます。SMPLTS データ・セットの場合は、SMP/E CLEANUP コマンドを使用できません。このデータ・セットはスクラッチして再割り当てする必要があります。

以下のサンプル SMP/E 制御ステートメントを使用して、このタスクを完了してください。

```
//SMPCNTL DD *
      SET BDY(targetzonename).
      CLEANUP.
```

9. SMP/E ターゲット・ゾーンを削除します。

- a. ターゲット・ゾーンに対して SMP/E ZONEDELETE コマンドを実行します。以下のサンプル SMP/E 制御ステートメントを使用して、このタスクを完了してください。

```
//SMPCNTL DD *
      SET BDY(targetzonename).
      ZDEL TZONE(targetzonename).
```

- b. ターゲット CSI (VSAM クラスタ) に他の SMP/E ゾーンが存在しない場合は、パフォーマンスを向上させるために、ターゲット CSI に対して IDCAMS DELETE および DEFINE コマンドを実行します。

重要: 同じ CSI 内に複数のゾーンがターゲット・ゾーンとして含まれている場合は、このクラスタを削除および再定義しないでください。なぜなら、それらのゾーンに関する情報も失われてしまうからです。

10. 新しいターゲット・ゾーンを再初期設定します。

- a. IDCAMS REPRO コマンドを実行して、SYS1.MACLIB(GIMZPOOL) を新しい CSI にコピーします。

重要: ステップ 9 で説明したターゲット CSI の削除と再定義を行わなかった場合は、新しいターゲット・ゾーンに GIMZPOOL をコピーしないでください。

- b. 古い DLIB ゾーンと新しいターゲット・ゾーン間の関係を再構築します。以下のサンプル SMP/E 制御ステートメントを使用して、このタスクを完了してください。

```
//SMPCNTL DD *
  SET BDY(GLOBAL).
  UCLIN.
  ADD GZONE ZONEINDEX(
    (targetzonename,target.zone.cluster.name,TARGET)
  ).
  ENDUCL.

  SET BDY(targetzonename).
  UCLIN.
  ADD TARGETZONE(targetzonename)
  SREL(P115)
  RELATED(dlibzonename)
  OPTIONS(xxxxxx).
  ENDUCL.
```

注: この新しいターゲットが、正しい OPTIONS 項目を指すようにしてください。正しい OPTIONS 項目は、ステップ 7 (248 ページ) で作成された出力から判別できます。

- c. UCLIN を実行し、DDDEF を再びターゲット・ゾーンに追加します。このステップでは、ステップ 6 (247 ページ) で作成されたデータ・セットを入力として使用します。以下の SMP/E 制御ステートメントと JCL を使用して、このタスクを完了してください。

```
//SMPCNTL DD *
  SET BDY(targetzonename).
//          DD DSN=IMS.SMPUNLD,DISP=SHR
```

注: このステップでは、戻りコード 4 が予想されます。なぜなら、DDDEF は置き換えられるのではなく、追加されるからです。

重要: ステップ 11 で SMP/E を処理する前に、現行の拡張 HOLDDATA に対して RECEIVE を実行してください。これにより、SMP/E 処理の間に PE を解決できるようになります。最新の HOLDDATA を <http://service.boulder.ibm.com/390holddata.html> から入手できます。

11. 処理する PTF に対して SMP/E ACCEPT GROUPEXTEND BYPASS(APPLYCHECK) コマンドを実行します。以下のサンプル SMP/E 制御ステートメントを使用して、このタスクを完了してください。

```
//SMPCNTL DD *
  SET BDY(dlibzonename).
  ACCEPT GROUPEXTEND
  BYPASS(APPLYCHECK
    HOLDCLASS(ERREL,UCLREL)
    HOLDSYSTEM
  )
  SOURCEID (SMCREC,RSU08*,RSU090*,RSU0910,etc)
  PTFS.
```

重要: 現行の拡張 HOLDDATA を使用した SMP/E 処理が完了したら、SMP/E REPORT ERRSYSMODS コマンドを実行して、欠落した HIPER と PE エクスポージャーを確認してください。

12. CONTENT を指定した SMP/E ZONEMERGE コマンドを実行し、配布ゾーンを新しいターゲット・ゾーンにマージします。以下のサンプル SMP/E 制御ステートメントを使用して、このタスクを完了してください。

```
//SMPCNTL DD *
    SET BDY(targetzonename).
      ZONEMERGE(dlibzonename)
        INTO(targetzonename)
          CONTENT.
```

13. SMP/E GENERATE コマンドを実行し、ターゲット・ライブラリーの再ビルドに必要な JCL を作成します。

ヒント: このステップを行うには、IMS FMID が ACCEPT される前に配布ゾーンで ACCJCLIN が設定済みであったことが必要です。

以下のサンプル SMP/E 制御ステートメントと追加の JCL を使用して、このタスクを完了してください。ここで、DD 名 CNTL 用のデータ・セットには、サンプル JOB カードが含まれている、J という名前のメンバーが必要です。

```
//CNTL      DD DSN=yourpds,DISP=SHR
//SMPPUNCH DD DSN=IMS.GENERATE,
//          DISP=(,CATLG),UNIT=SYSDA,
//          SPACE=(CYL,(25,5),RLSE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=16000)
          SET BDY(targetzonename).
          GENERATE JOBCARD(CNTL,J) REPLACE.
```

14. ステップ 13 で作成された JCL を実行します。

注: 外部参照 (IEW2454W) が未解決であるために、SMPLTS ジョブの完了時には戻りコード 4 が発行されます。これ以外のジョブは、すべて戻りコード 0 とともに完了します。

15. IMS ALL タイプのシステム定義 (STAGE 1 および STAGE 2) を実行します。

重要: ステップ 16 を実行する前に、ステージ 2 の処理が完了していることを確認してください。

16. 入力として STAGE 2 JCL を指示している SMP/E JCLIN を実行します。
17. 受け入れ済みでないすべての IMS サービスに対して SMP/E APPLY コマンドを実行します。このサービスは、ステップ 5 (247 ページ) で確認済みです。
18. 他の製品用の受け入れ済みでないすべてのサービスに対して SMP/E APPLY コマンドを実行します。このサービスは、ステップ 5 (247 ページ) で確認済みです。
19. RECEIVE コマンドから Enhanced Holddata の新規コピーをダウンロードします。
20. SMP/E REPORT ERRSYSMODS コマンドを実行します。
21. SMP/E REPORT ERRSYSMODS コマンドの出力を分析し、必要に応じて追加サービスを処理します。
22. 新しいシステムをテストします。

APPLY の前に ACCEPT (システム定義の必要なサービス)

この方法は事前生成モードの一種で、同一の SMP/E ゾーンを共用する多くのプロダクトがあり、通常は ACCEPT BYPASS(APPLYCHECK) シーケンス (一般には、システム定義に影響する PTF) を必要とする PTF をインストールする必要がある場合に有用です。

この方法は、未解決のサービス (APPLY されたが ACCEPT されていないサービス) を持つ他のプロダクトを妨害することを回避します。

1. IMS 環境をバックアップします。
 - a. SMP/E データ・セット (Zones、SMPMTS、および SMPPTS など) をバックアップします。
 - b. IMS プロダクト・データ・セット (SDFSRESL および ADFSLOAD など) をバックアップします。
2. 該当サービスを入手します。
3. パッケージに付随している資料を読みます。
 - ESO テープ資料
 - 予防サービス計画 (PSP)
4. SYSMODS と HOLDDATA の両方のパラメーターを設定して、SMP/E RECEIVE コマンドを実行します。
5. SMP/E ゾーンを IMS と共用しているすべての製品用の未解決の APPLY サービスに対して、SMP/E ACCEPT または RESTORE コマンドを実行します。
6. SMP/E ACCEPT CHECK GROUPEXTEND BYPASS(APPLYCHECK) コマンドを実行します。
7. ACCEPT CHECK 報告書をチェックし、必要であれば変更します。
8. SMP/E ACCEPT GROUPEXTEND BYPASS (APPLYCHECK) コマンドを実行します。
9. IMS ALL タイプのシステム定義 STAGE 1 および STAGE 2 を実行します。
10. 入力として STAGE 2 JCL を指示している SMP/E JCLIN を実行します。
11. SMP/E APPLY CHECK GROUPEXTEND を実行します。

重要: REDO パラメーターは使用しないでください。
12. APPLY CHECK 報告書をチェックし、必要であれば変更します。
13. SMP/E APPLY GROUPEXTEND を実行します。

重要: REDO パラメーターは使用しないでください。
14. RECEIVE コマンドから Enhanced Holddata の新規コピーをダウンロードします。
15. SMP/E REPORT ERRSYSMODS コマンドを実行します。
16. SMP/E REPORT ERRSYSMODS コマンドの出力を分析し、必要に応じて追加サービスを処理します。
17. 新しいシステムをテストします。

IMSpdex への IMS サービスのインストール

IMSpdex 内に IMS サービスをインストールするのは、単一のシステムにサービスをインストールするのとよく似ています。

以下の考慮事項は、IMSpdex にサービスをインストールする計画を立てる際に適用されます。

- 複数の Resource Manager (RM) および Operations Manager (OM) を IMSpdex で同時に実行できます。

すべての IMS システムが 1 つの LPAR 上にある場合は、OM サービスを提供するために、1 つの OM が常に稼働している必要があります。

- ある特定の時点で IMSpdex の単一の論理区画 (LPAR) で実行できる Structured Call Interface (SCI) は 1 つのみです。
- 共通キュー・サーバー (CQS) アドレス・スペースに接続しているすべての CQS クライアントは、その CQS をシャットダウンする前に停止されていなければなりません。

推奨事項:

- 連続可用性のためには、IMSpdex 全体にサービスを同時にインストールしないでください。
- 1 つの論理区画 (LPAR) で複数の IMS システムを実行している場合は、それぞれ個別に接続を停止し、サービスを実装し、テストし、再びオンラインに戻すなど、IMSpdex 内の各 IMS システムを別個のシステムとして扱ってください。各システムを正常にオンラインに戻した後、次のシステムに移って処理を繰り返します。

より複雑な IMS システムにサービスを適用する前に、より複雑でない IMS システムにサービスを適用してください。

- 複数の LPAR を実行している場合は、一度に 1 つの LPAR にサービスをインストールします。

より複雑な LPAR にサービスを適用する前に、より複雑でない LPAR にサービスを適用してください。

- IMSpdex に IMS Connect が含まれている場合は、IMS Connect を開始する前に、すべての適切な IMS システムが使用可能になっていることを確認してください。そうしないと、IMS Connect により、「データ・ストア使用不可」というメッセージが出されます。

関連タスク:

245 ページの『単一システム上の IMS サービスのインストール』

インストールおよび保守に関する一般的な問題

以下のトピックで示すインストールおよび保守に関する一般的な問題のいくつかを認識して適切な処置を講じれば、より安定した IMS 環境を実現できます。

IMS システム定義が原因で APPLY のみの状態の SYSMOD に回帰が起こるのを防ぐ

保守が APPLY のみの状態にある時に IMS システム定義が実行されると、保守がもとに戻って (回帰して) することがあります。

以下の手順のいずれかを使用して、APPLY のみの状態の SYSMOD が IMS システム定義によって回帰するのを防ぐことができます。

1. システム定義の前に、APPLY のみの状態のすべての PTF に対して ACCEPT を実行する。
2. システム定義を実行し、APPLY のみの状態の SYSMOD を回帰させた後で SYSMOD を再処理する。
- 以下の SMP/E ステートメントを使用して、APPLY のみの状態の SYSMOD をリストして識別します。

```
SET BOUNDARY (targetzone).  
LIST APAR PTF USERMOD NOACCEPT NOSUP.
```

• 方法 1:

1. システム定義の前に、APPLY のみの状態のすべての PTF に対して ACCEPT を実行します。
2. すべての APAR および USERMOD をリストアします。
3. ステージ 1 およびステージ 2 のシステム定義を実行します。
4. 必要に応じて、APAR および USERMOD を再適用します。

• 方法 2:

1. ステージ 1 およびステージ 2 のシステム定義を実行します。
2. 事前に識別した SYSMOD を再処理 (APPLY) します。以下の例の SMP/E ステートメントを使用して、APPLY のみの状態の SYSMOD を再処理することができます。

重要: REDO の処理には、NOJCLIN パラメーターのみを使用してください。他のものを使用した場合、サービスを RESTORE できない可能性があります。

```
APPLY REDO NOJCLIN SELECT(  
    xxxxxxx  
    xxxxxxx  
    xxxxxxx  
)
```

上記の例の xxxxxxx フィールドは、システム定義を実行する前に APPLY のみの状態にあった各 SYSMOD のリストを表しています。

重要: APPLY のみの状態の SYSMOD では、システム定義または DELETE を要求する HOLD の特殊処理が必要になることもあります。そのような SYSMOD については、HOLD ステートメント内の指示に従ってください。

非システム定義ターゲット・ライブラリーをビルドするための JCL の生成

IMS のエレメントの中には、IMS システム定義プロセスに組み込まれていないものもあります。これらのエレメントは、SMP/E に対して識別され、FMID 用に APPLY 処理中にビルドされます。

SMP/E GENERATE コマンドを使用して、これらのコンポーネントをターゲット・ライブラリーで再ビルドするのに使用する JCL を作成できます。また、SMP/E GENERATE は、IMS 配布ゾーンにある IRLM のようなプロダクト用に JCL を作成することもできます。SMP/E GENERATE 処理は、FMID が ACCEPT された時に配布ゾーンで設定されていた SMP/E パラメーター ACCJCLIN に依存します。

IVP ダイアログの場合の保守の適用

IVP ダイアログ・プロセスに影響を与えるサービスには、特別な処理を実行しなければならない場合があります。

実行する必要があるアクションがあれば、SMP/E HOLDDATA は、必要なアクションを識別します。

HOLDDATA に識別された場合、以下のアクションが必要になる可能性があります。

- 表のマージ

いずれかのマスター表で行が追加、変更、または削除された場合、表のマージが必要となります。表のマージが行われたら、その変更を INSTALIB にあるユーザー表に反映する必要があります。ダイアログ処理が変更した変数については、デフォルト値が更新されません。

- 変数収集

新規および変更された変数のデフォルト値を、変更することができます。

- ファイル調整

ファイル調整を再実行し、新規の JOB または TASK について INSTALIB メンバーを追加できるほか、INSTALIB メンバーを新規または変更された変数の値で更新できます。

- 実行

IVP プロセスの一部を実行または再実行できます。

z/OS のアップグレード

IMS を実行中の z/OS システムをアップグレードする前に、z/OS インターフェースおよび VTAM インターフェースのすべての要件を考慮します。

IMS を実行中の z/OS システムをアップグレードするときには、必ず以下のことを行ってください。

1. z/OS に関する考慮事項を確認します。詳しくは、147 ページの『第 5 章 z/OS インターフェースの考慮事項』を参照してください。

2. VTAM に関する考慮事項を確認します。詳しくは、163 ページの『第 6 章 VTAM インターフェースの考慮事項』を参照してください。

注: IVP D シリーズのサンプルには、チャンネル間 (CTC) チャンネル終了付加ルーチンを除くすべての z/OS および VTAM インターフェースの例が含まれています。詳しくは、「IMS V15 インストール」を参照してください。

適切な **SYSLIB** 連結の確認

使用するマクロ・ライブラリーの連結順序は重要です。この順序が正しくないと、アセンブラー処理の間に予測不能な結果が生じる場合があります。使用する各ライブラリーが、正しい順番に連結されることを確認してください。

SMP/E APPLY の対象:

1. SMPMTS
2. IMS.SDFSMAC
3. MVS マクロ・ライブラリー

SMPMTS

配布ライブラリー内にもみ存在するマクロ用のターゲット・ライブラリー。このデータ・セットを使用すると、APPLY 処理でのアセンブリーにマクロの現行バージョンが使用されるようになります。

IMS.SDFSMAC

すべての IMS マクロ用のターゲット・ライブラリー。

MVS マクロ・ライブラリー

SYS1.MACLIB (AMACLIB)、SYS1.MODGEN (AMODGEN)、SYS1.TSOMAC (ATSOMAC)、および ASM.SASMMAC2 のうち、該当するものを組み合わせて構成されます。ASM.SASMMAC2 には 14 個のマクロが含まれていて、HLASM Toolkit の High Level Assembler に添付されています。

SMP/E ACCEPT の対象:

1. IMS.ADFSMAC
2. MVS マクロ・ライブラリー

注: SMP/E APPLY プロセスとの顕著な違いは、SMPMTS が使用されないことと、SMP/E がターゲット・ライブラリーではなく、配布ライブラリーを指していることです。SMPMTS には、受け入れ済みでないバージョンのマクロが入っていません。

IMS システム定義のステージ 1:

1. IMS.ADFSMAC

IMS システム定義のステージ 2:

1. IMS.ADFSMAC
2. MVS マクロ・ライブラリー

バインダー戻りコードの適切な解釈

一部のバインダー戻りコードは無視しても安全ですが、その他は無視できません。

以下の表に、各種のバインダー・プロセスからの許容される戻りコードをリストします。

表 16. バインダーからの許容される戻りコード

SMP/E 処理のタイプ	戻りコード	解釈の仕方
APPLY	0	未解決の外部参照を無視してはなりません。例外が SMPLTS にバインドされます。
ACCEPT	4	未解決の外部参照を無視しても安全です。
システム定義 STAGE 2	0	未解決の外部参照を無視してはなりません。

推奨事項: APPLY および ACCEPT 処理では、SMP/E で別のユーティリティー項目を指すようにしてください。

SMP/E を使用したサンプル出口ルーチンのアSEMBルとバインド

以下の例では、SMP/E にサンプル出口ルーチンのいずれかをアSEMBルおよびバインドさせる場合に使用できる技法を示します。

```
++ USERMOD (XYZUMOD) .
++ VER (P115)
   FMID(HMK1500) .
++ JCLIN.
//INJCLIN JOB . . . .
//LKED EXEC PGM=IEWL,
// PARM='(SIZE=(880K,64K)',RENT,REFR,NCAL,LET,XREF,LIST)
//ADFSLOAD DD DSN=IMS.ADFSLOAD,DISP=SHR
//SYSPUNCH DD DSN=IMS.OBJDSET,DISP=SHR
//SYSUT1 DD UNIT=(SYSDA,SEP=(SYSLSMOD,SYSLIN)),SPACE=(1024,(200,20))
//SYSPRINT DD SYSOUT=A
//SYSLSMOD DD DSN=IMS.SDFSRESL,DISP=SHR
//SYSLIN DD *
   INCLUDE ADFSLOAD(DFSCSI00)
   INCLUDE SYSPUNCH(DFSGMSG0)
   ENTRY DFSGMSG0
   NAME DFSGMSG0(R)
++ SRC (DFSGMSG0) SYSLIB(SDFSSMPL) DISTLIB(ADFSSMPL) .
DFSGMSG0 TITLE 'DFSGMSG0 -- GREETING MESSAGES user exit routine routine'
.....
.....
```

IMS の新しいバージョンへのマイグレーション

IMS の新しい版にマイグレーションする場合は、どの版からマイグレーションするかとはかかわりなく、実行する必要がある特定の作業があります。

IMS の新しい版にマイグレーションするときには、以下の作業を確実に行ってください。

1. これからマイグレーションする版の「リリース計画ガイド」の内容を確認します。特に、マイグレーションと共存に関する情報を確認してください。

2. ある版をスキップする場合は、スキップする版の「リリース計画ガイド」の内容を確認します。特に、マイグレーションと共存に関する情報を確認してください。
3. これからマイグレーションする版の PSP バケットの内容を確認します。
4. ある版をスキップする場合は、スキップする版の PSP バケットの内容を確認します。

第 15 章 共用キューの計画

このトピックでは、共用キューの使用のために理解しておくべき概念を概説し、共用キューに関連した計画および管理のための作業を説明します。

メッセージ・キューを管理するために、ご使用のシステムでは IMS キュー・マネージャーとメッセージ・キュー・データ・セットまたは共用キューを使用することができます。

制約事項: 高速機能適格システムで共用全機能メッセージ・キューを使用する場合は、EMH メッセージについての共用キューも定義する必要があります。

共用キュー環境での操作の前に、以下のトピックの情報をお読みにになり、共用キュー環境での操作と非共用キュー環境での操作の違いを理解してください。

共用キュー環境での CQS の役割

共通キュー・サーバー (CQS) は、キュー構造またはリソース構造などのカップリング・ファシリティー・リスト構造上のオブジェクトを管理する汎用サーバーです。CQS は 複数のクライアントのために共用キューからのデータ・オブジェクトを受信、保守、および配布します。

各クライアントは CQS と通信し共用キューにアクセスします。IMS は、その共用キューおよび共用リソースの両方の管理に CQS を使用する CQS クライアントの一例です。

CQS は、データ・オブジェクトのリポジトリとして z/OS カップリング・ファシリティーを使用します。カップリング・ファシリティー内のストレージは、構造と呼ばれる別のオブジェクトに分割されます。許可プログラムは構造を使用し、データ共用および高速シリアライゼーションを実装します。カップリング・ファシリティーは、リスト構造に従ってデータを保管し、配置します。キュー構造には、同じ名前を共用するデータ・オブジェクト (キューと認識されている) の集合が含まれます。

共用キュー環境での IMS の役割

共用キュー環境では、IMS は、端末用のフロントエンドとして、または処理用のバックエンドとして、あるいは同時にその両方として働きます。

フロントエンドとしての IMS

フロントエンドとして、IMS は端末リソースを管理し、コマンド、トランザクション、およびメッセージ通信などの通信トラフィックを開始します。メッセージは、一般的には端末で入力されたデータの結果です。フロントエンドとしての IMS は、データベース、プログラム、およびトランザクション・リソースが定義されている場合、バックエンドとしての役割も果たすことができます。

バックエンドとしての IMS

バックエンドとして、IMS はトランザクションおよびデータベース・リソースを管理し、アプリケーション・プログラムをスケジュールします。バックエンドとしての IMS は、端末リソースが定義されている場合、フロントエンドとしての役割も果たすことができます。

フロントエンドとバックエンドの両方としての IMS

フロントエンドとバックエンドの両方として、IMS は、ある IMS 端末から入力されたメッセージを、その IMS 上あるいは IMSplex 内の別の IMS 上のどちらでも処理することができます。

IMS がバックエンド・プロセッサとして働く場合には、ローカル、リモート MSC、およびリモート共用キューという 3 つの処理環境が存在します。

定義:

- ローカル処理が行われるのは、メッセージを処理する IMS 従属領域が、メッセージを受信した IMS 制御領域と同じ IMS に存在する場合です。
- リモート MSC 処理が行われるのは、メッセージを処理する IMS 従属領域が、メッセージを受信した IMS 制御領域とは異なる IMS に存在し、メッセージが MSC を使用してリモート IMS に送信される場合です。
- リモート共用キュー処理が行われるのは、メッセージを処理する IMS 従属領域が、メッセージを受信した IMS 制御領域とは異なる IMS に存在する場合です。ただし、この場合、リモート IMS が処理するメッセージは、共用キュー機能に含まれます。

IMS がキューに関するインタレストを登録する方法

IMS システムは、その IMS システムが処理できる作業のタイプに従って、共用キューに関するインタレストを登録および登録解除します。これにより、この IMS の CQS は、作業がキューに存在する場合に、この IMS に通知することができます。

作業のタイプには以下のものがあります。

- トランザクション
- LTERM
- MSC リソース (リモート LTERM、リモート・トランザクション、および MSNAME など)

一般に、IMS は、リソースが追加 (または開始) されたときにインタレストを登録し、リソースが削除 (または停止) されたときにインタレストを登録解除します。

キュー・タイプ

共用キュー環境では、様々なタイプのキューが管理されます。各キュー・タイプは様々なタイプの作業に使用されます。IMS は、ユーザーが定義した作業のタイプに基づいて、処理できるキュー・タイプにのみインタレストを登録します。

キュー・タイプと IMS がキュー・タイプで処理する作業を以下の表に示します。

表 17. 共用キュー環境で維持されるキュー・タイプ

キュー・タイプ	説明
トランザクション作動可能キュー	宛先がトランザクションであるメッセージの最初のキュー・バッファが入ります。
トランザクション・ステージング・キュー	IMS キュー・マネージャーが、単一のキュー・バッファを超えたメッセージの部分を保持するために使用する内部キュー。
トランザクション延期キュー	宛先が延期トランザクションであるメッセージの最初のキュー・バッファが入ります。
トランザクション逐次キュー	宛先が逐次トランザクション (TRANSACT マクロで SERIAL=YES として IMS に定義されたトランザクション) であるメッセージの最初のキュー・バッファが入ります。
LTERM 作動可能キュー	宛先が LTERM または MSNAME であるメッセージの最初のキュー・バッファが入ります。
LTERM ステージング・キュー	IMS キュー・マネージャーが、単一のキュー・バッファを超えたメッセージの部分を保持するために使用する内部キュー。
APPC 作動可能キュー	宛先が APPC 装置であるメッセージの最初のキュー・バッファが入ります。
REMOTE 作動可能キュー	宛先がリモート・トランザクションまたはリモート LTERM であるメッセージの最初のキュー・バッファが入ります。
OTMA 作動可能キュー	宛先が OTMA 装置であるメッセージの最初のキュー・バッファが入ります。

トランザクションに関するインタレストの登録および登録解除

IMS は、特定の処置が行われた場合に、インタレストをトランザクション・キューに登録したり、登録を解除したりします。

- IMS は、以下のイベントが発生すると、トランザクション・キューに関するインタレストに登録します。
 - IMS が初期設定される。
 - /MODIFY COMMIT コマンドが入力された後、またはグローバル・オンライン変更が可能な状態で INITIATE OLC PHASE(COMMIT) コマンドが入力された後、トランザクションがオンライン変更によって追加される。
 - オペレーターが /START TRAN コマンドを入力する。
 - オペレーターが UPD TRAN START(SCHD) コマンドを入力する。
 - CQS が終了して再始動された後に、IMS が CQS に再接続する。
- IMS は、以下のイベントが発生すると、トランザクション・キューに関するインタレストに登録解除します。
 - IMS がシャットダウンされる。
 - /MODIFY COMMIT コマンドの後、またはグローバル・オンライン変更が可能な状態で INITIATE OLC PHASE(COMMIT) コマンドの後、トランザクションがオンライン変更によって削除される。
 - /STOP TRAN コマンドが入力される。
 - /PSTOP TRAN コマンドが入力される。

- UPD TRAN STOP(SCHD) コマンドが入力される。
- IMS 高速機能の場合
 - IMS は、IFP 領域が処理するプログラム名に関するその領域の初期設定時に、プログラム・キューに関するインタレストを登録します。
 - IMS は、IFP 領域が終了すると、プログラム・キューに関するインタレストを登録解除します。

LTERM に関するインタレストの登録および登録解除

IMS は、特定の処置が行われた場合に、インタレストを LTERM キューに登録したり、登録を解除したりします。

- IMS は、以下のイベントが発生すると、LTERM キューに関するインタレストを登録します。
 - ユーザーが静的端末にログオンする。
 - ユーザーが動的端末にサインオンする。
 - LTERM がアクティブ・ユーザーまたはアクティブ・ノードに割り当てられる。
- IMS は、以下のイベントが発生すると、LTERM キューに関するインタレストを登録解除します。
 - ユーザーが静的端末からログオフする。
 - ユーザーが動的端末からサインオフする。
 - LTERM が非活動ユーザーまたは非活動ノードに割り当てられる。

MSC リソースに関するインタレストの登録

MSC 論理リンクが開始されると、IMS は、リンクと一緒に定義された MSNAME に関するインタレストを登録します。

同じ SYSID の対を使用して定義されたすべてのリモート・トランザクションも登録されます。

論理リンクが停止されると、IMS は、MSNAME とその関連するリモート・トランザクションに関するインタレストを登録解除します。

共用キュー環境でのメッセージ・フロー

共用キュー環境では、IMS が共用キューからメッセージを取り出すと、そのメッセージは、共用キュー上で、IMS がそのメッセージをアンロックするか削除するまで、ロックされます。IMS がメッセージのアンロックも共用キューからの削除も行わない場合 (例えば、IMS が異常終了した場合)、そのメッセージはロックされたままになります。

ロックされたメッセージは他の IMS システムによって処理できません。IMS が共用キュー上でメッセージをロックした後に、IMS をコールド・スタートすると、そのメッセージはロックされたままでコールド・キューに移されます。

RM がリソース構造で定義されている場合、IMS は、障害が起きている IMS 上の LTERM を宛先とするロック状態のメッセージをすべて解放します。メッセージは IMSplex 内の別の IMS にサインオンしているユーザーに対して使用可能になります。

定義: コールド・キューとは、クライアントのコールド・スタート後に、CQS が専用キューで検出したロックされたメッセージを入れるキュー・タイプです。メッセージは、削除されるかアンロックされるまでコールド・キューに残ります。

作業単位 (UOW) トラッキング

非共有キュー環境では、作業単位はリカバリー・トークンおよび作業単位 (UOW) によって追跡されます。共有キュー環境では、作業単位は UOW のみを使用して追跡されます。

UOW には以下のフィールドがあります。

- 発信元システム・メッセージ ID。メッセージを発信する IMS によって割り当てられるメッセージ ID です。
 - 発信 IMSID
 - タイム・スタンプ・トークン
- 処理システム・メッセージ ID。メッセージを処理する IMS によって割り当てられるメッセージ ID です。
 - 処理 IMSID
 - タイム・スタンプ・トークン

UOW は、メッセージを発信したシステムと (もしあれば) メッセージを処理するシステムの両方の ID を持っているので、元のメッセージに関連するすべてのメッセージは UOW (特に、UOW 内の発信システム・メッセージ ID) によって結合することができます。

端末自動ログオン

アプリケーション・プログラムは任意の IMS バックエンド・システムで実行できます。したがって、これらの任意のシステム上の同じユーザーまたは端末に出力を生成することができます。

待機しているすべてのユーザーに出力を送達するために、自動ログオン端末は IMSplex 内の IMS 間で頻繁に切り替えられます。

メッセージ通信およびプログラム間通信

メッセージ通信は、共有キュー環境でグローバルにキューに入れられます。プログラム間通信は、ローカル優先の最適化の対象として適格です。プログラム間通信メッセージは、ローカル優先の処理対象として選択されない場合はグローバルにキューに入れられます。

メッセージ通信の宛先がローカル IMS に定義されていない場合、宛先作成出口ルーチン (DFSINSX0) が呼び出されて宛先を識別します。

ETO がアクティブである間に、宛先が LTERM であることを出口ルーチンが示すと、この出口ルーチンは LTERM の動的ユーザー構造を作成して、メッセージをキューに入れます。

DFSINSX0 出口は、宛先がトランザクションであることを示すこともできます。IMS は、出口によって指定された環境とオプションに基づいて、トランザクションを作成します。

- DRD を使用しない共用キュー環境では、IMS は、キューイング専用トランザクションを作成して、メッセージをグローバルにキューに入れます。
- DRD が有効に設定された共用キュー環境では、IMS は、出口からの要求にしたがって、キューイング用またはスケジューリング用のトランザクションを作成します。

動的制御ブロック

該当するトランザクションを定義していない IMS システムで、ユーザーがトランザクションを入力できるようにするには、ユーザーは宛先作成出口ルーチン (DFSINSX0) に情報を提供して、IMS に動的スケジューラー・メッセージ・ブロック (SMB) を作成させることができます。

入力メッセージと出力メッセージは制御ブロックではなく共用キューに入れられません。

IMS キュー・マネージャー

IMS キュー・マネージャーは、IMS メッセージ・キューを管理します。キュー・マネージャー・データ・セット SHMSG および LMSG は、共用キュー環境ではメッセージのバックアップには使用されません。

しかし、IMS システム定義時に必要な変更を削減するために、MSGQUEUE マクロは依然として必要です。これにより、データ・セットは定義されますが、割り振りまたはオープンは行われません。

キュー・マネージャーは、多くの実記憶域キュー・バッファーを使用してメッセージを保管した後に、カップリング・ファシリティ上の共用キューにメッセージを置きます。ただし、メッセージが共用キューに置かれるのではなくローカル IMS に残る必要がある場合は、メッセージはローカル・キューに置かれます。ローカル・キューはキュー・マネージャーによっても管理されます。

共用キューに置かれたメッセージは回復可能です。ローカル・キュー上のメッセージは、メッセージ・キュー・データ・セットが使用されないため、最後の 2 つのチェックポイントのうち古い方が取られた後に、メッセージに関するログ・レコードが IMS ログに書き込まれた場合にのみカバリー可能です。

XRF 環境では、ローカル・キュー・マネージャー・データ・セット (QBLKSL、SHMSG、および LGMSG) が使用されます。XRF 代替サブシステムがトラッキング中、ローカル・メッセージはローカル・メッセージ・キュー・データ・セットに置かれます (アクティブ・サブシステムに関するメッセージは共用キューに置かれます)。XRF 代替サブシステムがアクティブ・サブシステムを引き継ぐと、ローカル・メッセージ・キュー・データ・セット内のすべてのメッセージが共用キューのメッセージと組み合わせられます。

MTO メッセージ

1 次マスター端末オペレーター (MTO) 用メッセージは、ローカルで保持され、IMS の再始動後にはリカバリーされません。2 次 MTO のメッセージは LOCAL=YES オプションを使用している共用キュー上に置かれます。

これらの 2 次 MTO メッセージは IMS 再始動後にリカバリーされ、CQSREAD によっては検索されません。

逐次アプリケーション処理

共用キューを使用しない IMSplex 環境では、アプリケーション・プログラムを逐次プログラムとして定義する (つまり、プログラムをメッセージ領域またはバッチ・メッセージ領域に同時にスケジュールすることができない) 場合、APPLCTN マクロの SCHDTYP= パラメーターを使用します。逐次プログラムとして定義されたアプリケーション・プログラムは、複数のメッセージ領域またはバッチ・メッセージ領域で同時に実行するようにスケジュールすることはできません。しかし、並列アプリケーション・プログラム (SCHDTYP=PARALLEL) は、同時に実行されるようスケジュールできます。

共用キューを持つ IMSplex 環境でも、アプリケーション・プログラムを逐次プログラムとして定義できます。アプリケーション・プログラムを逐次プログラムとして定義 (SCHDTYP=SERIAL) するほかに、次のことを行う必要があります。

- Resource Manager のアクティブ・インスタンスを定義する。
- CQS グローバル構造定義 IMS PROCLIB メンバー・データ・セット (CQSSGxxx) を更新して、RSRCSTRUCTURE パラメーターによりリソース構造を定義する。
- CSL RM 初期設定パラメーター PROCLIB メンバー・データの RSRCSTRUCTURE パラメーターを使用してリソース構造を識別する。

共用キューが使用可能で、かつリソース構造が定義されている RM がアクティブである場合、IMS は、IMSplex で逐次と定義されたアプリケーションのスケジューリングを直列化します。

関連資料:

- RM の定義および調整については、「IMS V15 システム定義」を参照してください。
- IMS PROCLIB データ・セットのメンバー CQSSGxxx については、「IMS V15 システム定義」を参照してください。

逐次トランザクション処理

逐次トランザクションのメッセージは、そのメッセージをキューに置く IMS によってのみ処理されます。IMS は、逐次トランザクション・メッセージのトランザクション逐次キューをカップリング・ファシリティ MSGQ 構造内に保持します。

AOI プログラムは、そのトランザクションを逐次として定義する必要があります (TRANSACTION マクロで SERIAL=YES を指定します)。

共用キュー環境での会話型トランザクション

共用キュー環境では、会話を開始する IMS に対する会話の状況はローカルです。

IMS が会話を開始すると、以下のイベントが発生します。

1. IMS が、会話型トランザクションを共用キューに置きます。
2. ローカルに定義されている SMB (または RSMB) あるいは出力作成ユーザー出口ルーチン (DFSINSX0) が、トランザクションを会話型として識別します。
3. トランザクションに関するインタレストを登録している任意の IMS がその処理を行えます。さらに、任意の IMS が、会話の任意のステップを処理することができます。
4. 1 つの IMS がメッセージを読み取り、メッセージを処理するアプリケーション・プログラムをスケジュールします。
5. アプリケーション・プログラムがメッセージを処理し、データを SPA に保管し、開始端末に応答します。

/EXIT コマンド、/HOLD コマンド、および /RELEASE コマンドは、会話を開始した IMS にのみ適用されます。

共用キュー環境の構成

複製構成と区分構成のどちらを使用しても、共用キュー環境を構成できます。

267 ページの図 28 および 269 ページの図 29 は、共用キュー環境を構成するための 2 つの方法を示しています。

複製構成

複製構成 では、ほとんど同じシステム定義を使用して同様の IMS システムを作成します。定義において、いくつかの変更 (MSC リンクなど) が必要です。システム定義時に IMS PROCLIB メンバー・データ・セット DFSDCxxx に生成される MTO 定義をオーバーライドすることができます。

関連情報: MTO 定義をオーバーライドする方法については、「IMS V15 システム定義」を参照してください。

すべての IMS システムの LTERM、データベース、およびトランザクションは同様に定義されます。この構成では、任意の IMS が特定のトランザクションを処理することができます。複製構成は自動作業負荷平衡化に適しています。

推奨事項: 複製構成を十分に利用するためには、データベースを IMSplex 全体で共用するようにしてください。

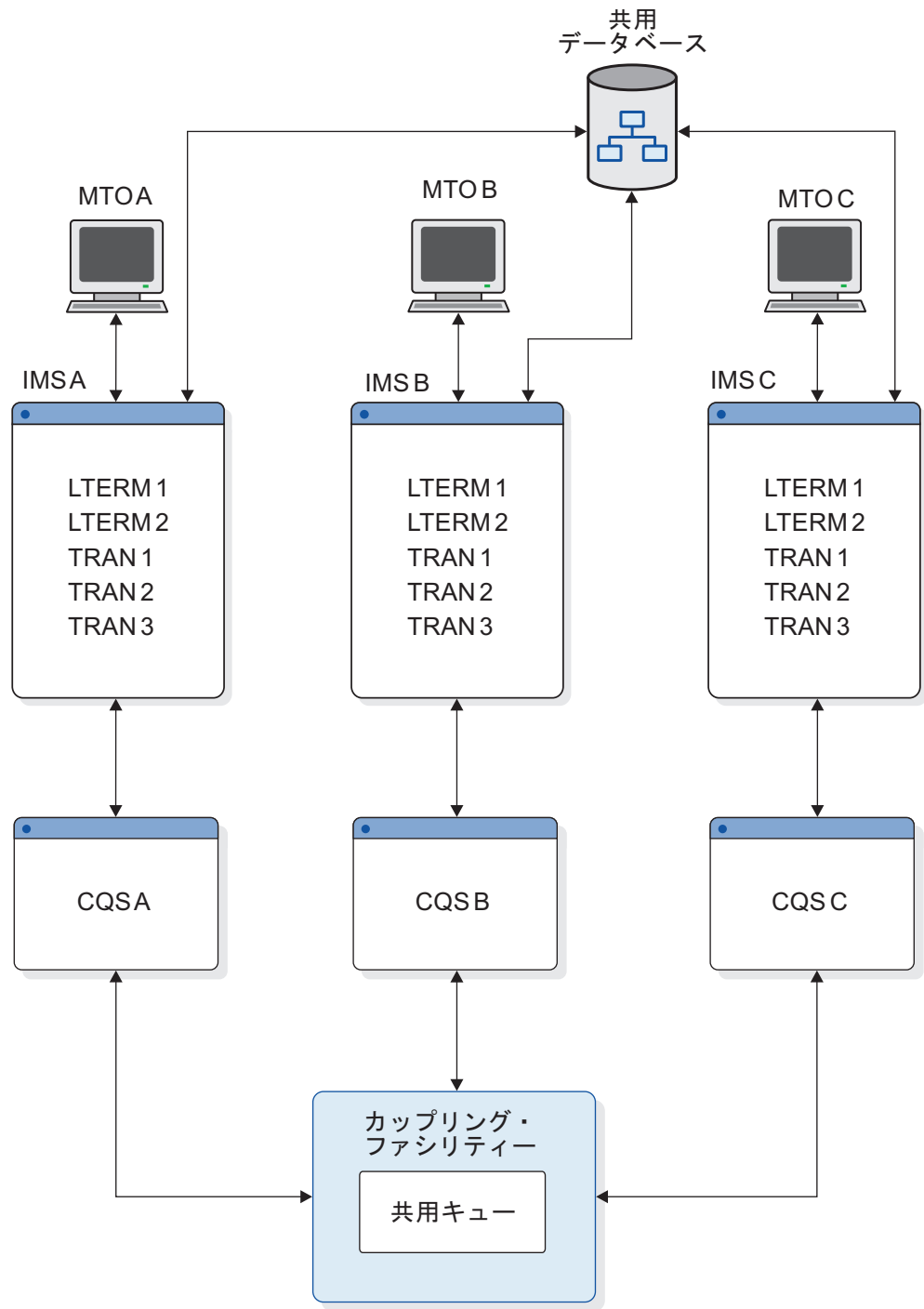


図 28. 共有キュー環境における複製構成

区分構成

区分構成は、VTAM のネットワーク責任をデータベース作業から分離します。各リソースは 1 つの IMS のみに対して定義されます。

複数のより小さな 9672 CMOS プロセッサがデータベース作業を管理するとき、1 つの ES/9000 9021 が IMS フロントエンドとして VTAM ネットワークを管理するように定義することができます。

区分構成は、MSC ネットワークを置き換えるものや追加するものとして有用です。

以下の図に示された区分構成では、IMS A はフロントエンドとして働き、LTERM 1、LTERM 2、および MTO 1 が定義されています。IMS B、IMS C、および IMS D はバックエンドとして働き、それぞれに別個のトランザクションおよびデータベースが定義されています。この構成のトランザクションは、それらが定義されている IMS システムでしか実行することができません。

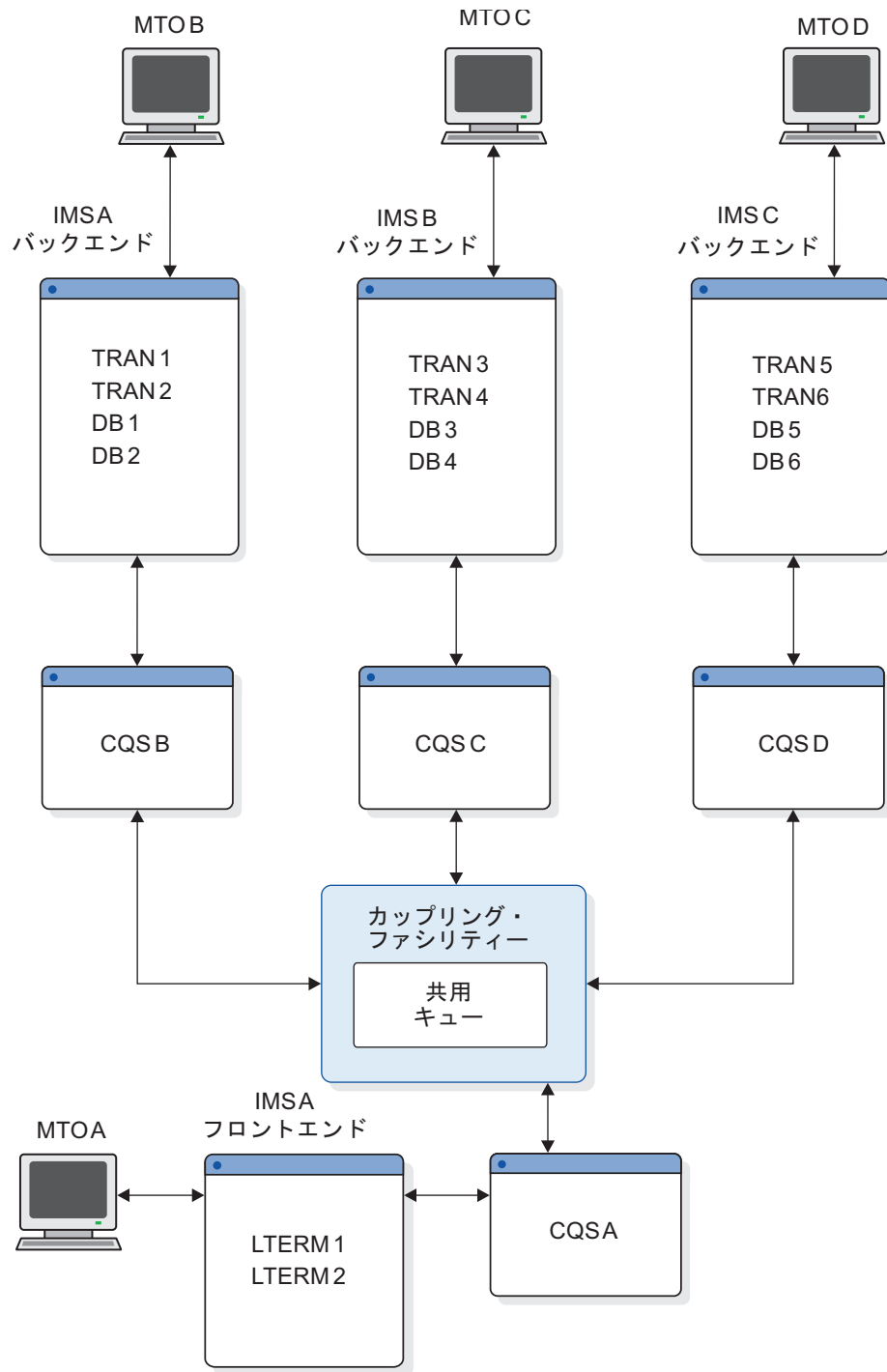


図 29. 共用キュー環境における区分構成

共用キューの使用可能化

共用キューを z/OS カップリング・ファシリティ・リソース管理 (CFRM) ポリシーで定義することにより、共用キューを使用可能にできます。

このトピックでは、CFRM ポリシーに含める必要のあるもの、CFRM ポリシーで作成した定義に準拠する必要のある他の定義について説明します。

CFRM ポリシーの定義

共用キューを使用可能にするには、すべてのメッセージ・キューの構造名および属性を含む CFRM ポリシーを定義する必要があります。高速機能を使用し、EMH キューを共用する場合は、EMHQ 構造の構造名と属性も含める必要があります。

メッセージ・キューおよび、すべてのオプションの EMH キューに関する CFRM ポリシーで定義する構造名が、以下の場所でも定義されていることを確認してください。

- 共用キュー IMS.PROCLIB メンバー・データ・セット (DFSSQxxx) のパラメーター MSGQ= (EMH キューを共用する場合は、さらにパラメーター EMHQ=)
- CQS ローカル構造定義 PROCLIB メンバー・データ・セット (CQSSLxxx) のパラメーター STRNAME=
- CQS グローバル構造定義 PROCLIB メンバー・データ・セット (CQSSGxxx) のパラメーター STRNAME= および OVFLWSTR=

推奨事項: 可能であれば、使用頻度の高い構造を別個のカップリング・ファシリティーに置いてください。例えば、IMS データ共用構造は、共通キュー構造とは別のカップリング・ファシリティーに置いてください。VSAM および Db2 for z/OS 構造を、さらに別のカップリング・ファシリティーに置いてください。使用頻度の低い構造 (RACF に関する構造など) は、スペースが使用可能な任意のカップリング・ファシリティーに置くことができます。

関連資料: CQS を使用するために必要な定義について詳しくは、「IMS V15 システム定義」を参照してください。

以下の例は、共用キューおよびリソース構造の定義を含む CFRM ポリシーの例を示しています。

```
//CFRMPLCY JOB MSGCLASS=A,REGION=2000K,CLASS=K
// MSGLEVEL=(1,1)
//*****
/* This JCL is used for configuration. INITSIZE is          *
/* used for the primary MSGQ and EMHQ structures.          *
//*****
/* 2 CF                                                    *
//*****
//POLICY EXEC PGM=IXCM2APU
//STEPLIB DD DSN=SYS1.MIGLIB,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSIN DD *

DATA TYPE(CFRM)
DEFINE POLICY
    NAME(CONFIG01)
    REPLACE(YES)

    CF NAME (CF01)
    TYPE(nnnnnn)
    MFG(aa)
    PLANT(nn)
    SEQUENCE(nnnnnnnnnnnn)
    PARTITION(n)
    CPCID(nn)
```



```

CF NAME (CF02)
TYPE(nnnnnn)
MFG(aa)
PLANT(nn)
SEQUENCE(nnnnnnnnnnn)
PARTITION(n)
CPCID(nn)
.
.
.
STRUCTURE NAME(QMSGIMS01)
SIZE(16000)
INITSIZE(8000)
| MINSIZE(8000)
| PREFLIST(CF01,CF02)
| REBUILDPERCENT(1)
| ALLOWAUTOALT(YES)
| FULLTHRESHOLD(60)
STRUCTURE NAME(QMSGIMS010FLW)
SIZE(8000)
| MINSIZE(8000)
| PREFLIST(CF01,CF02)
| REBUILDPERCENT(1)
| ALLOWAUTOALT(YES)
| FULLTHRESHOLD(60)
STRUCTURE NAME(QEMHIMS01)
SIZE(16000)
INITSIZE(10000)
| MINSIZE(10000)
| PREFLIST(CF01,CF02)
| REBUILDPERCENT(1)
| ALLOWAUTOALT(YES)
| FULLTHRESHOLD(60)
STRUCTURE NAME(QEMHIMS010FLW)
SIZE(8000)
| MINSIZE(8000)
| PREFLIST(CF01,CF02)
| REBUILDPERCENT(1)
| ALLOWAUTOALT(YES)
| FULLTHRESHOLD(60)
STRUCTURE NAME(MVSLOGQMSG01)
SIZE(16000)
INITSIZE(11000)
PREFLIST(CF01,CF02)
STRUCTURE NAME(MVSLOGQEMH01)
SIZE(4000)
PREFLIST(CF01,CF02)
REBUILDPERCENT(1)
STRUCTURE NAME(QRSCIMS01)
SIZE(16000)
INITSIZE(8000)
| MINSIZE(8000)
| ALLOWAUTOALT(YES)
| FULLTHRESHOLD(60)
| DUPLEX(ALLOWED)
| PREFLIST(CF01,CF02)
.
.
.

```

メッセージ・キュー・リスト構造の定義

共用キューを使用する前に、基本リスト構造 MSGQ を IMS に対して定義する必要があります。オプションでオーバーフロー構造を定義することができます。

例では、STRUCTURE NAME(QMSGIMS01) はメッセージ・キュー構造を定義し、STRUCTURE NAME(QMSGIMS010FLW) は QMSGIMS01 メッセージ・キューのオーバーフロー構造を定義しています。

リソース構造の定義

RM を使ってグローバル状況を共用するには、リソース構造を定義する必要があります。例では、STRUCTURE NAME(QRSCIMS01) はリソース構造を定義しています。RM によるグローバル・リソース情報の維持方法について詳しくは、201 ページの『CSL RM により管理される情報』を参照してください。

高速機能メッセージ・キュー・リスト構造の定義

高速機能が定義された IMS で MSGQ 基本リスト構造を定義する場合、オプションで共用 EMH キューの基本リスト構造を定義することができます。この構造は EMHQ 構造と呼ばれます。次にオプションとして EMHQ オーバーフロー構造を定義することができます。

EMHQ 構造を定義した場合、高速機能トランザクションを共用キュー・グループ内の IMS システムによって処理することができます。EMHQ 構造を定義しない場合、高速機能トランザクションがあれば、それはローカルのみで処理されます。

例では、STRUCTURE NAME(QEMHIMS01) は EMHQ 構造を定義し、STRUCTURE NAME(QEMHIMS010FLW) は QEMHIMS01 EMH キューのオーバーフロー構造を定義しています。

z/OS ログ・ストリームの定義

z/OS ログ・ストリームを定義する必要があります。定義を更新することにより、z/OS ログ・ストリームを定義します。

z/OS ログ・ストリームを定義する手順は、次のとおりです。

- GRSTNLxx PARMLIB メンバー内で、並列シスプレックス環境内のすべての IMS を、同じグローバル・リソースのシリアライゼーション複合システム内に存在するように指定してください。これによって、IMSplex 内のリソースのグローバル・シリアライゼーションが可能になります。
- メッセージ・キューと EMH キュー (共用する場合) に関する CFRM ポリシーにログ・ストリーム構造名と属性を定義します。

例: 270 ページの『CFRM ポリシーの定義』の例では、STRUCTURE NAME(MVSLOGQMSG01) はメッセージ・キューのログ・ストリーム構造名を定義しています。同様に、STRUCTURE NAME(MVSLOGQEMH01) は、高速機能の EMH キューのログ・ストリーム構造を定義しています。

- DFSMS をインストールし、SMS DATACLAS、STORCLAS、および MGMTCLAS 構成を変更してください。

- 在庫結合データ・セットをフォーマット設定し、ログ・ストリームおよび構造名を定義します。
- COUPLExx PARMLIB メンバーを更新します。
- ロガー在庫データ・セットを SYS1.PARMLIB(COUPLExx) 内に定義します。
- ログ・ストリームおよびカップリング・ファシリティ構造名をロガー在庫結合データ・セットに定義します。

関連情報: z/OS ログ・ストリームの定義方法について詳しくは、「IMS V15 システム定義」を参照してください。

CQS パラメーターの定義

共用キューを使用可能にするには、PROCLIB データ・セットの CQSIPxxx、CQSSLxxx、および CQSSGxxx メンバーを更新する必要があります。

共用キューを使用可能にするには、以下の CQS パラメーターを定義してください。

- PROCLIB メンバー・データ・セット CQSIPxxx 内の CQS 初期設定パラメーター。
- PROCLIB メンバー・データ・セット CQSSLxxx の CQS ローカル構造定義パラメーター (ローカル構造定義の場合)。上記の定義は単一の CQS にのみ適用されます。
- PROCLIB メンバー・データ・セット CQSSGxxx の CQS グローバル構造定義パラメーター (グローバル構造定義の場合)。

構造の対ごとに、上記の各パラメーターを 1 つずつ定義する必要があります。キューを共用するすべての CQS は、CQSSGxxx PROCLIB メンバー・データ・セット内で同じ値を指定されていなければなりません。値が異なる場合、最初の CQS だけが構造に接続され、他の CQS は接続されません。

- 実行パラメーター (オプション)。

ARMRST=、CQSGROUP=、SSN=、STRDEFG=、または STRDEFL= を指定しない場合、CQS は CQSIPxxx PROCLIB メンバー・データ・セットに指定されている値を検索します。

関連資料: CQS パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。

共用キューの IMS パラメーター

共用キューを使用可能にするには、IMS 制御領域パラメーターと IMS 始動パラメーターを定義する必要があります。

以下の IMS 制御領域実行パラメーターを定義します。

LGMSGSZ=

長メッセージのレコード長。

QBUF=

キュー・バッファー・プールの拡張上限。

QBUFHITH=

メッセージ・キュー・バッファを動的に拡張する高限界 % を表す 1 から 100 の範囲の値の 1 から 3 桁の数。デフォルト・パーセントは 80% です。

QBUFLWTH=

メッセージ・キュー・バッファを圧縮する低限界 % を表す 1 から 100 の範囲の値の 1 から 3 桁の数。デフォルト・パーセントは 50% です。

QBUFMAX=

キュー・バッファ・プールに入れることができるメッセージ・キュー・バッファの最大数。

QBUFCTX=

QBUFHITH パラメーター値に到達したときに動的に拡張される、最初に割り振られたメッセージ・キュー・バッファの % を表す 1 から 100 の範囲の値の 1 から 3 桁の数。デフォルト・パーセントは 20% です。

QBUFSZ=

キュー・バッファのサイズ。

SHAREDQ=

PROCLIB データ・セットの共用キュー・メンバーの接尾部。このパラメーターは PARMLIB メンバー DFSSQxxx にあります。

SHMSGSZ=

短メッセージのレコード長。

PROCLIB データ・セットのメンバー DFSSQxxx の以下の IMS 始動パラメーターを定義します。

CQS=

CQS プロシージャが入っている PROCLIB データ・セットのメンバー。
CQS= はオプションです。デフォルトは CQS=CQS です。

CQSSN=

CQS アドレス・スペース・サブシステムの名前。

EMHQ=

EMH キュー構造の名前。このステートメントが存在する場合は、共用キュー環境内に EMHQ 構造 (および、その他の関連した構造) が必要です。

MSGQ=

メッセージ・キュー構造の名前。

SQGROUP=

文字 DFS に連結される、z/OS システム間カップリング・ファシリティ
IMS 共用キュー・グループ名を表す 1 から 5 文字の ID。同じ構造セットを共用するすべての IMS システムは、同じ SQGROUP= 名を指定する必要があります。SQGROUP= 名は、CQSIPxxx PROCLIB メンバー・データ・セットに指定されている CQSGROUP= 名と同じにすることができます。

WAITRBLD=Y | N

EMHQ 構造に関するアクティビティを、CQS が構造再作成処理を完了するまで待つべきかどうかを指定します。WAITRBLD=NO は、EMHQ 構造に関

するアクティビティーを、CQS が構造を再構築する間、継続すべきことを指定します。WAITRBLD パラメーターはオプションです。デフォルトは WAITRBLD=N です。

関連情報: これらの IMS パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。

Common Queue Server の使用

IMS は、シスプレックス環境で共用キューを管理するために、CQS インターフェースを使用します。IMS は、CQS クライアントとして機能する各 IMS からこの CQS インターフェースを使用して、共用キューにアクセスします。

制約事項: 共用キュー・シスプレックスにおいて、複数のシステムで同じ LTERM 名を使用する場合には、すべての会話型モードおよび応答モードの応答メッセージは、発信側端末のシステムによって処理しなければなりません。これは、1 つの IMS で会話モードまたは応答モードの操作が進行中の間は、同じ LTERM 名を別の IMS 上の端末が同時にも逐次にも使用できないことを意味します。このトランザクション・マネージャー (TM) の制約事項は、共用キュー環境の高速機能と非高速機能の両方に適用されます。

CQS の開始

CQS は次の 2 つの方法のいずれかで活動化することができます。

- z/OS START コマンドを使用して、z/OS タスクとして
- z/OS バッチ・ジョブとして

さらに、IMS は、適切な場合には CQS を自動的に開始します。

CQS の再始動

CQS 構造にデータが入っているかどうかによって、CQS のウォーム・スタートまたはコールド・スタートを実行することができます。

- 構造が空であれば、CQS をコールド・スタートする必要があります。
- 構造にデータが入っていれば、CQS のウォーム・スタートまたはコールド・スタートを実行することができます。

CQS の再始動を完了すると、CQS 作動可能メッセージ (CQS0020I) が出されません。

CQS のウォーム・スタート

ウォーム・スタート時、CQS はチェックポイント・データ・セットを読み取って、最後のシステム・チェックポイントを表すログ・トークンを検出します。このログ・トークンを見つけると、CQS はウォーム・スタートを開始します。チェックポイント・データ・セットにログ・トークンが見つからないと、CQS は構造からログ・トークンを読み取ります。CQS は、ログ・トークンを検出した場合、WTOR を出すので、ユーザーはこのトークンの使用を確認することができます。

CQS のコールド・スタート

空の構造に接続した後に CQS がコールド・スタートした場合、CQS はその構造に関するすべてのログ・レコードを除去し、システム・チェックポイントを実行します。システム・チェックポイントの完了後、構造および CQS の再始動が完了します。

CQS ユーザー提供の出口ルーチンの使用

以下の出口ルーチンを使用すると、CQS アクティビティをモニターおよび変更することができます。

- CQS 初期設定/終了出口ルーチン
- CQS クライアント接続出口ルーチン
- CQS キュー・オーバーフロー出口ルーチン
- CQS 構造統計出口ルーチン
- CQS 構造イベント出口ルーチン

カップリング・ファシリティ構造のサイズ変更

カップリング・ファシリティでの構造の初期サイズは、CFRM ポリシー内の INITSIZE パラメーターの値によって決まります。CQS によって、構造のサイズを動的に再構成することができます。

最初の CQS が構造に接続する場合、その構造のサイズは INITSIZE に指定された値になります。この INITSIZE 値のサイズを確保するために十分なフリー・スペースが存在しない場合、構造のサイズは、カップリング・ファシリティで使用可能なスペースのサイズになります。

システム・チェックポイントの開始

再始動時に CQS 情報をリカバリーするためのシステム・チェックポイントとして、各 CQS はその固有の制御ブロックを z/OS ログに書き込みます。この再始動情報を検索するために、CQS は情報を CQS チェックポイント・データ・セットにも書き込みます。チェックポイントの進行中は、CQS はアクティビティを静止しません。

チェックポイント・データ・セット

CQS は、各構造ペアごとにチェックポイント・データ・セットを維持管理しています。CQS はチェックポイント・データ・セットに書き込み、それを再始動時に使用します。チェックポイント・データ・セットは CQS 初期設定時に動的に割り振られます。

システム・チェックポイント後の CQS の再始動方法

CQS 再始動時、CQS は最後のシステム・チェックポイントからログ・レコードを読み取り、進行中であった作業を再構築します。

構造チェックポイントの開始

メッセージ・キューを回復するために、CQS は構造チェックポイントを使用します。CQS は構造の対から構造リカバリー・データ・セット (SRDS) に共用キューをコピーします。CQS は、このコピー操作の一環を実行している間、アクティビ

ティーを静止します。次に、CQS は、各構造チェックポイントに続けてシステム・チェックポイントを実行します。次のいずれかの手順を使用して、構造チェックポイントを開始します。

- /CQCHKPT SHAREDQ コマンドを入力する。
- CQSCHKPT FUNC=CHKPTSTR CQS クライアント要求を発行する。
- 2 つの構造チェックポイント間のインターバルで、個々の CQS アドレス・スペースに書き込み可能なログ・レコードの最大数を指定する。この数値に到達した場合、CQS は自動的に構造チェックポイントを実行します。

構造リカバリー

z/OS によって構造を再作成することができます。構造再作成は、構造のコピーまたは構造リカバリーによって行います。

構造再作成

構造および構造上のメッセージをコピーまたは回復するには、1 つ以上の CQS を実行する必要があります。新しい構造を割り振ると、ポリシーの変更 (構造の位置など) が適用されます。

構造のコピー

構造再作成の開始時に少なくとも 1 つの CQS が構造にアクセスしている場合には、構造にアクセスしている CQS システムの 1 つが、その構造から新しい構造にすべてのメッセージ (リカバリー可能メッセージとリカバリー不能メッセージ) をコピーします。

構造リカバリー

構造再作成の開始時に、構造にアクセスしている CQS がない場合、構造は SRDS と z/OS ログからリカバリーされます。リカバリー不能メッセージ (高速機能入力メッセージなど) は失われます。

構造の削除

CQS が接続されていない構造は、削除することができます。構造の削除は、次の方法で行います。

1. 構造に接続されているすべての CQS をシャットダウンします。
2. 失敗したすべての持続接続を削除します。

重要: 構造に接続している間に CQS で障害が発生した場合は、障害発生時に処理していた作業をすべてクリーンアップできるように、CQS の再始動を許可します。このコマンドは、構造を削除しなければならないときに、障害が発生した接続を終了するために使用できます。このコマンドの使用法を誤ると、キューまたはリソースが失われる場合があります。

以下のコマンドを発行します。

```
SETXCF FORCE,CONNECTION,STRNAME=strname,CONNAME=ALL
```

3. 次のコマンドを発行します。

```
SETXCF FORCE,STRUCTURE,STRNAME=strname
```

このコマンドの *strname* 値が、PROCLIB メンバー・データ・セット CQSSGxxx の CQS グローバル構造定義パラメーターと PROCLIB メンバー・データ・セット CQSSLxxx の CQS ローカル構造定義パラメーターに指定されている *strname* の値と同じであることを確認してください。

メッセージ・キューの削除

構造上のメッセージ・キューを削除するには、構造を削除します。

共用キュー使用状況のモニター

キュー・スペース通知出口ルーチン (DFSQSSP0) を使用して IMS 共用メッセージ・キュー構造の使用状況をモニターすることができます。出口には、基本メッセージ・キュー構造およびオーバーフロー・メッセージ・キュー構造内の割り振り済みおよび使用中の項目およびエレメントの総数に関する情報が渡されます。この情報を使用して、メッセージ・キュー構造が満杯にならないようにすることができます。




CQS 構造に直接アクセスするプログラムを作成する場合、マクロ CQSPUT および CQSDEL でパラメーター FEEDBACK= および FEEDBACKLEN= を使用することにより、その構造の使用率に関する情報を受け取ることができます。

CQS ロギングおよび z/OS ロガー

z/OS システム・ロガーは、CQS が構造をリカバリーし再始動するために必要なすべての情報を記録します。CQS は、ログ・レコードを読み取るための、ファイル選択およびフォーマット設定印刷ユーティリティを提供します。

関連資料: ログ・ストリームの定義について詳しくは、「z/OS MVS Programming: Sysplex Services Guide」を参照してください。

関連資料:

-  IMS PROCLIB データ・セットの CQSSGxxx メンバー (システム定義)
-  IMS PROCLIB データ・セットの CQSSLxxx メンバー (システム定義)
-  ファイル選択およびフォーマット設定印刷ユーティリティ (DFSERA10) (システム・ユーティリティ)

キュー制御機能を使用した構造のモニターおよび再キューイング

キュー制御機能 (QCF) を 2 とおりの方法で使用して、メッセージをコピーおよび削除し、共用キューのメッセージ・カウントおよび経過時間を照会することができます。1 つは共用キュー・グループ内の IMS 上で実行する QCF BMP ジョブをサブミットする方法で、もう 1 つは QCF TCO/ISPF インターフェースを使用する方法です。

メッセージの削除は、クリーンアップまたは以後のメッセージの再挿入および処理に対して有用です。逸失メッセージを選択して再キューイングし再処理の共用キューに入れることができます。

QCF を使用したコールド・キューの再キューイング

IMS が異常終了し、その後コールド・スタートした場合、CQS は異常終了時に処理中であったメッセージを CQS 共用キュー構造のコールド・キュー上に置きます。QCF を使用してコールド・キュー構造上のこれらのメッセージを分析または削除することができます。また、QCF を使用して再処理のための処理キューの 1 つにメッセージを移動することもできます。

関連情報: QCF ツールについて詳しくは、「IMS Queue Control Facility for z/OS ユーザー・ガイド」を参照してください。

IMS コマンドによる CQS へのアクセス

これらの IMS コマンドは、ローカル IMS にのみ適用されます。コマンドおよびコマンド応答は共用キューに置かれないので、リカバリー可能ではありません。

以下の IMS コマンドを使用して、CQS と対話することができます。

/CQCHKPT SHAREDQ

CQS 構造チェックポイントを開始する。

/CQCHKPT SYSTEM

CQS システム・チェックポイントを開始する。

/CQQUERY STATISTICS

基本構造とオーバーフロー構造の統計を表示する。

/CQSET SHUTDOWN SHAREDQ

CQS シャットダウンで構造チェックポイントを取る状況を設定する。

/DEQUEUE

以下のいずれかから 1 つ以上のメッセージを除去する。

- アウトバウンド APPC キュー
- 高速機能プログラム
- LTERM
- MSNAME
- アウトバウンド OTMA キュー
- メッセージ・キュー
- リモート・トランザクションまたはリモート LTERM
- トランザクション

/DEQUEUE SUSPEND

1 つ以上の延期トランザクションをキューから除去する。

/DISPLAY CQS

IMS が追跡する CQS 情報を表示する。

/DISPLAY MODIFY

進行中のローカル作業を表示する。

/DISPLAY OVERFLOWQ

オーバーフロー・モードのキュー名のリストを表示する。

/DISPLAY QCNT

特定のリソース・タイプのグローバル・キュー情報を表示する。

/DISPLAY STRUCTURE

構造状況を表示する。

/DISPLAY TRACE TABLE QMGR

キュー・マネージャーのトレース・テーブル状況を表示する。

/DISPLAY TRACE TABLE SQTT

共用キュー・トレース・テーブル状況を表示する。

/TRACE SET OFF TABLE QMGR

キュー・マネージャー・アクティビティのトレースを停止する。

/TRACE SET ON TABLE QMGR

キュー・マネージャー・アクティビティのトレースを開始する。

/TRACE SET OFF TABLE SQTT

共用キュー・アクティビティのトレースを停止する。

/TRACE SET ON TABLE SQTT

共用キュー・アクティビティのトレースを開始する。

QUERY TRAN

共用キュー上のメッセージの情報を表示する。

QUEUE TRAN

共用キューに対しメッセージをエンキューおよびデキューする。

リスト構造

CQS は、カップリング・ファシリティー・リスト構造 を使用して、IMSplex 内で共用されているメッセージ・キューを管理します。基本リスト構造 はメッセージ・キューを保持します。オーバーフロー・リスト構造 を定義した場合は、基本リスト構造が事前定義しきい値に到達した後の追加のキューが保持されます。

高速機能トランザクションを共用していない IMS システムの場合、メッセージ・キュー基本構造とオプションのメッセージ・キュー・オーバーフロー構造のみを定義します。高速機能トランザクションを共用している IMS システムの場合、メッセージ・キューの基本構造 (およびオプションのオーバーフロー構造) と急送メッセージ・ハンドラー・キュー (EMHQ) の基本構造 (およびオプションのオーバーフロー構造) の両方を定義します。

281 ページの図 30 と 282 ページの図 31 は、IMS クライアント A および B の共用キューを示しています。

以下の図では、メッセージ・キュー・構造、メッセージ・キュー構造リカバリー・データ・セット (SRDS)、メッセージ・キュー・チェックポイント・データ・セット、およびメッセージ・キュー z/OS ログ・ストリームのみを使用しています。

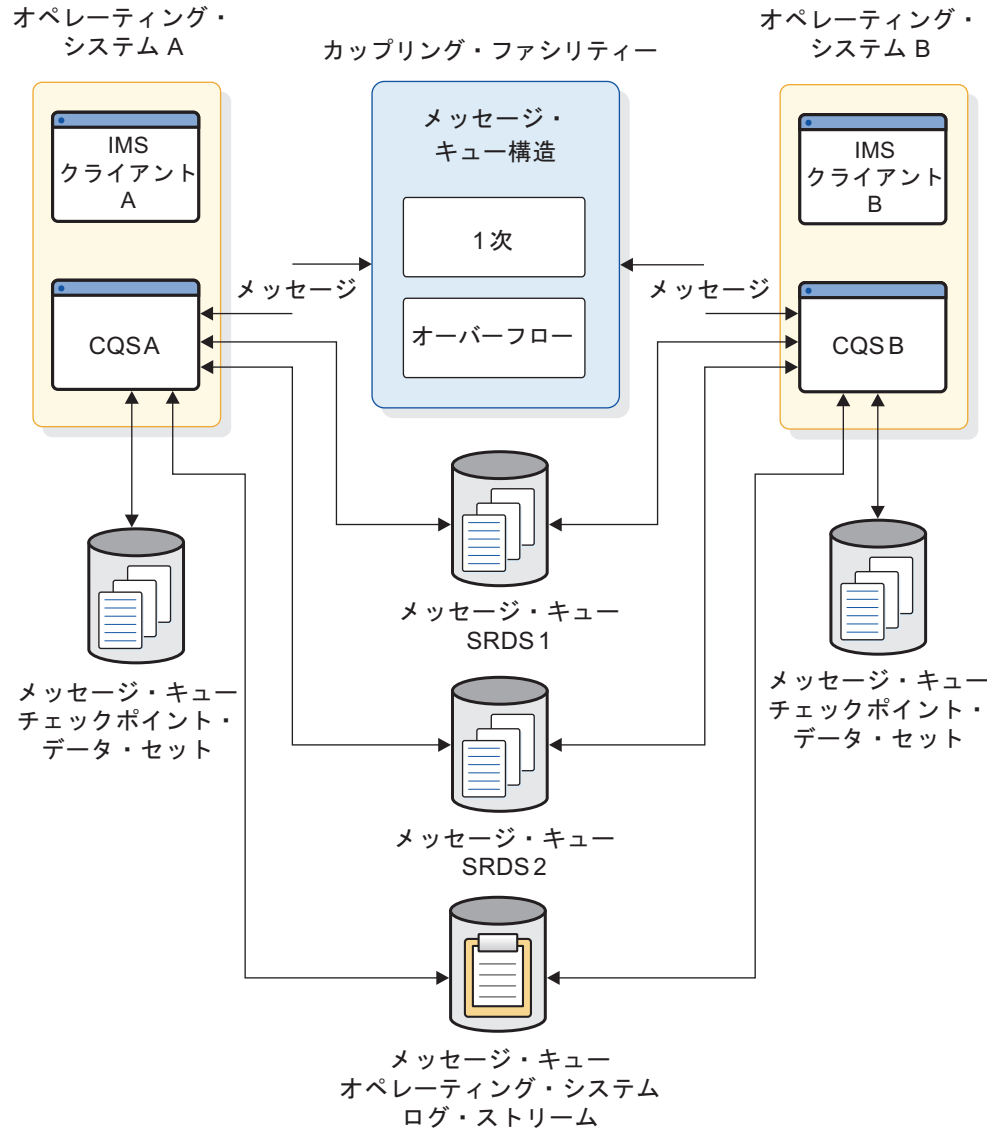


図 30. メッセージ・キュー (MSGQ) リスト構造がある共用キュー環境

以下の図に示す構成は、EMHQ 構造とそれに対応する SRDS、チェックポイント・データ・セット、および z/OS ログ・ストリームを示しています。このような構成の場合には、メッセージ・キューも含まれますが、この図では表示されていません。

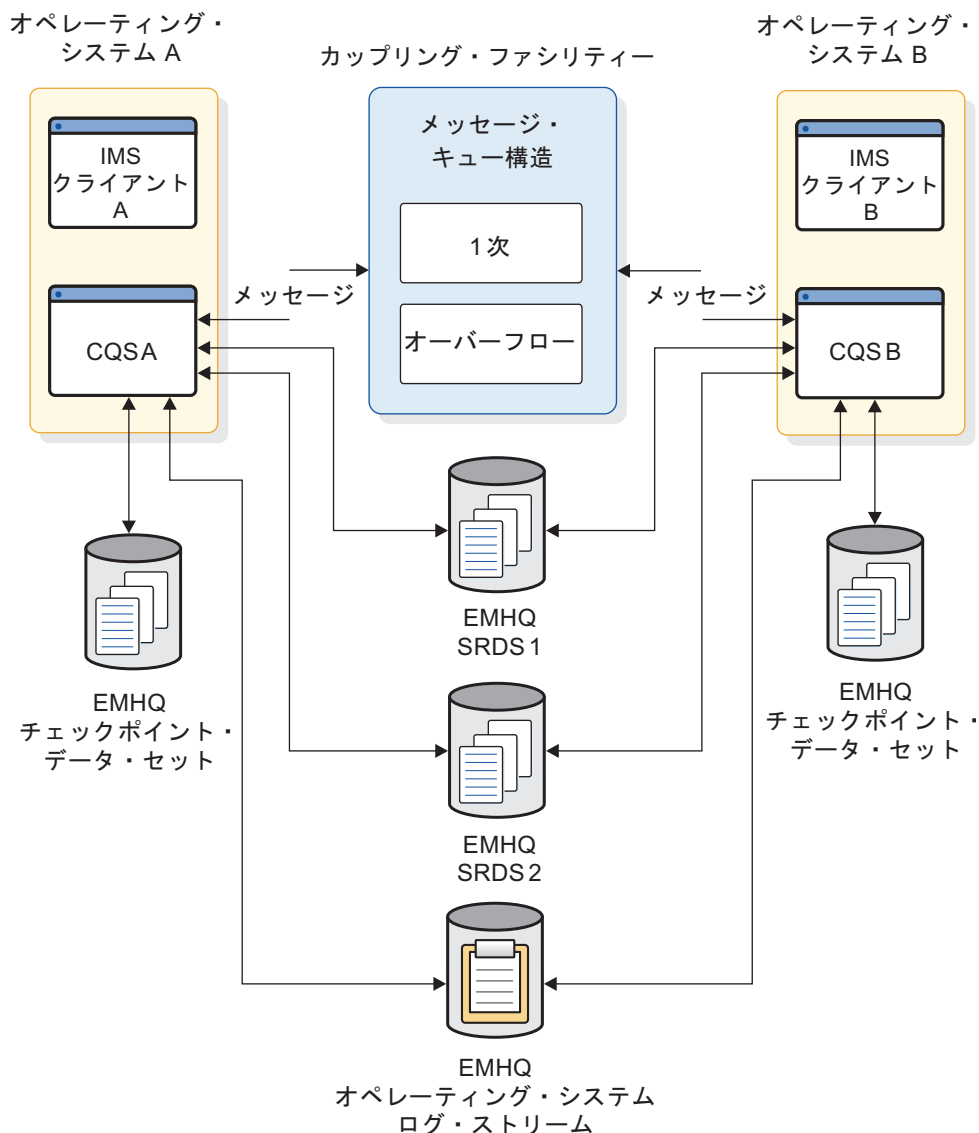


図 31. EMHQ リスト構造がある共用キュー環境

関連するプリンターの使用

共用キュー環境では、ユーザーがサインオンして関連するプリンター情報を指定すると、IMS フロントエンドは、各関連 LTERM についてのインタレストを登録します。アプリケーションがこれらの LTERM の 1 つへのメッセージを挿入すると、IMS フロントエンドは通知を受けて、プリンターの自動ログオンを開始します。

推奨事項: IMS バックエンド内のユーザーには自動ログオン・パラメーターを提供しないでください。提供すると、フロントエンドとバックエンドの両方が関連プリンターの自動ログオンを処理することになります。

共用キュー環境の共用プリンターでは、多くの場合、共用プリンター端末は IMSplex 内の IMS システム間で切り替えられます。これにより、出力を IMSplex 内の任意の IMS に送信して印刷することが可能になります。

IMS フロントエンド切り替えの計画

共用キュー環境で IMS フロントエンド切り替え (FES) を使用するには、以下の 2 つのイベントのいずれかが必要です。

IMS フロントエンド切り替えを完了するには、次のいずれかを行います。

- 入力メッセージを正しい LTERM に経路指定する FES 出口ルーチン (DFSFEBJ0) をコーディングする (FES 応答メッセージを正しく送達するため)。
- IMSplex 内の IMS システム間で ISC セッションを確立する (FES 入力メッセージと応答メッセージが任意の IMS によって確実に送受信されるようにするため)。

関連資料:

- FES 出口ルーチンについては、「IMS V15 出口ルーチン」を参照してください。
- ISC セッションの確立については、「IMS V15 コミュニケーションおよびコネクション」を参照してください。

シスプレックス全体にわたる処理に適しているかどうかの判別

共用キュー環境内の IMS メッセージには、シスプレックス全体にわたる処理に適しているものと、適していないものがあります。

定義: IMS メッセージ は、以下のステートメント内のいずれかです。

- トランザクションまたはトランザクションの応答
- メッセージ通信
- システム・メッセージ
- LU 6.2 または OTMA からのメッセージ
- コマンドまたはコマンドの応答

すべてのメッセージ (コマンドまたはコマンドの応答は含まない) は、共用キューに置かれます。

シスプレックス全体にわたる処理に適格なメッセージ

以下のタイプの通信に関するメッセージは、IMSplex 内の任意の IMS での処理に適格です。

- ローカルに定義されているか、宛先作成出口ルーチン (DFSINSX0) を使用して定義されているトランザクション
- シスプレックス処理オプション LOCAL FIRST または GLOBAL ONLY を指定した DBFHAGU0 出口ルーチンを使用して定義された高速機能トランザクション
- IMSplex 内でアクティブであるプログラムに関する高速機能メッセージ
- 以下の同期メッセージ・タイプ
 - APPC
 - OTMA
- LTERM、または宛先作成出口ルーチン (DFSINSX0) を使用するもの。

定義: LTERM に関するメッセージには、次のものがあります。

- トランザクションの応答
- メッセージ通信
- システム・メッセージ
- コマンドの応答

シスプレックス全体にわたる処理に適格ではないメッセージ

以下のタイプの通信に関するメッセージは、ローカルで処理する必要があるため、シスプレックス全体の処理には不適です。

- 逐次トランザクション (TRANSACT マクロの SERIAL=YES 指定によって定義されているトランザクション)
- コマンドおよびコマンドの応答

シスプレックス環境での APPC および OTMA メッセージの管理

APPC メッセージおよび OTMA メッセージは非同期または同期のいずれにでもできます。

非同期 APPC および OTMA メッセージ

シスプレックス環境で APPC メッセージおよび OTMA メッセージが確実に送達されるようにするには、共用キュー・グループ内のそれぞれのバックエンド IMS で APPC および OTMA を使用可能にします。あるバックエンド IMS で APPC または OTMA が使用可能にされていない場合、代替 PCB に挿入される非同期 APPC 出力または非同期 OTMA 出力はキューに入れられて、オペレーターが /STA APPC コマンドまたは /STA OTMA コマンドを出すまで送達されません。

プログラム間通信によって作成された非同期 APPC メッセージおよび非同期 OTMA メッセージは、共用キュー環境内のいずれの IMS 上でも実行することができます。

同期 APPC および OTMA メッセージ

同期 APPC および OTMA (送信後コミット) メッセージは、共用キュー環境内の任意の IMS 上で実行して、トランザクションの作業負荷を分散させることができます。入出力 PCB に挿入された同期 APPC または OTMA 出力は、フロントエンド IMS により送達する必要があります。そのため、どの IMS がトランザクションを実行しているかに関係なく、APPC または OTMA は同期出力をフロントエンド IMS に戻します。非会話型 I/O PCB 応答メッセージ (61 KB 未満) は、z/OS システム間カップリング・ファシリティ サービスを使用して、フロントエンド IMS に送信されます。会話型 I/O PCB 応答メッセージまたはすべての同期出力メッセージ (いずれも 61 KB より大) は、共用キュー、および XCF を使用して送信される特別な NOTIFY メッセージを使用して、フロントエンド IMS に送信されます。次の条件が真のとき、IMSplex は同期 APPC メッセージおよび同期 OTMA メッセージをサポートします。

- AOS=Y または AOS=B の場合、z/OS リソース・リカバリー・サービスはアクティブである。

- AOS=Y または AOS=B の場合、IMS 制御領域パラメーター RRS=Y が、IMSplex 内のすべての IMS システムに定義されている。
- AOS=Y、AOS=F、AOS=B、AOS=S、または AOS=X が DFSDCxxx PROCLIB メンバー・データ・セットに指定されている。

バックエンド IMS で実行されている同期 APPC または OTMA トランザクションが、代替 PCB に挿入される非同期 APPC または OTMA 出力になる場合は、バックエンド IMS で APPC または OTMA を使用可能にする必要があります。また、非同期出力は、バックエンド IMS から直接 APPC または OTMA クライアントに配信されます。


ローカル・トランザクションへの非会話型プログラム間通信によって開始された同期 APPC および OTMA トランザクションは、常時、そのプログラム間通信を開始したトランザクションと同じ IMS システムで実行されます。例えば、シスプレックス内のどのフロントエンドまたはバックエンド IMS システムでも実行できる同期 TRAN A が、同期 TRAN B と非同期 TRAN C を作成する場合、TRAN B と TRAN C は両方とも TRAN A と同じ IMS システムで実行されます。

APPC/OTMA SMQ Enablement がアクティブになっている共用キュー環境では、APPC/OTMA を介して IMS に入ってくるトランザクションは、IMSplex 内のどのメンバーも実行できます。ただし、トランザクションがプログラム間通信を実行した場合は、制御領域パラメーターの APPCASY=S または OTMAASY=S が指定されていない限り、トランザクションへの切り替えは、その切り替えが発生したのと同じ IMS メンバーによって処理される必要があります。

APPC/OTMA 共用キュー環境では、いつトランザクションがバックエンド IMS システムで処理され、複数プログラム間通信を実行するかは、トランザクションへの切り替えを同期的にスケジュールに入れられるかどうかに関係なく、GU IOPCB 時に決定されます。ただし、制御領域パラメーターの APPCASY=S または OTMAASY=S が指定されている場合は、どのトランザクションを同期的にスケジュールに入れるかの決定は、ISRT ALTPCB 時に行われます。

関連概念:

357 ページの『APPC 汎用リソース名の指定』

 [APPC/IMS および LU 6.2 装置の管理 \(コミュニケーションおよび接続\)](#)

IMS 送達不能キューの管理

IMS は、ローカル・ユーザー構造についてのみ送達不能キューを保持します。カップリング・ファシリティ上の共用キューについては、/DIS QCNT MSGAGE コマンドを使用して、潜在的な送達不能キューを表示してください。

表示出力を調べれば、特定のキューが送達不能キューであるかどうかを判別することができ、適切な処置を取ることができます。

急送メッセージ・ハンドラーの使用

共用キュー環境では、IMS 高速機能急送メッセージ・ハンドラー (EMH) も共用キューを使用することができます。IMS は、高速機能入力編集/経路指定出口ルーチン呼び出して、着信メッセージを高速機能で処理すべきかどうかを判別します。

高速機能でメッセージを処理する必要がある場合、ローカル IMS システムがメッセージを処理するか、あるいは IMS がメッセージを共用キューに置きます。

高速機能は、(PROCLIB の DFSSQxxx メンバー内の) MSGQ 構造と同じ方法で定義される EMHQ 構造を使用して、高速機能トランザクションを共用することができます。ただし、高速機能がインストールされた IMS システムでは、EMHQ 構造を持つことまたは高速機能トランザクションを共用することは必要ありません。EMHQ 構造を定義しない場合、高速機能トランザクションはすべてローカルで処理されます。

ある IMS システムが z/OS システム間カップリング・ファシリティー グループに結合するか、異常終了するか、あるいはシャットダウンした場合は、プログラム名テーブルを伴うメッセージを送信して、すべての共用 IMS システムに通知します。このメッセージの送信によって、すべての IMS システムが、特定プログラムにサービスを提供している IMS システムについて認識することができます。同様に、ある IMS システムでプログラムが開始または停止された場合、すべての共用 IMS システムに通知し、各 IMS システムは、各自のプログラム名テーブルを更新します。

プログラムにサービス提供するための使用可能なアクティブ領域が IMSplex 内に存在しない場合、そのプログラムに関するメッセージをキューに入れたすべての IMS システムが、メッセージ DFS2529I を入力端末に発行します。そうすると、このような端末はアンロックされます。このプログラムへの新しい入力はメッセージ DFS2533I を伴って拒否されます。

高速機能の入力メッセージは回復可能ではないので、CQS チェックポイント時に SRDS データ・セットに書き込まれません。EMHQ 構造は回復することができます。ただし、高速機能の入力メッセージは SRDS データ・セットに保持されません。高速機能の出力メッセージだけを EMH キューから回復することができます。EMHQ 構造再作成時に失われた入力メッセージからの出力を待っている端末ごとに、IMS はメッセージ DFS2766I を送信し、その端末をアンロックします。さらに、IMS は、失われたメッセージに関する X'5936' ログ・レコードを書き込みます。

EMHQ 構造はコピーすることができます。また、すべてのメッセージは、リカバリ可能かどうかに関係なく、古い構造から新しい構造にコピーされます。

EMHQ 再構築に関しては、再構築される EMHQ 構造が、再構築の進行中にアクセス可能であることを指定することができます。DFSSQxxx PROCLIB メンバー・データ・セット内で WAITRBLD= パラメーターを使用してください。

延期トランザクションの再キューイング

トランザクションは、延期されると、カップリング・ファシリティ内のトランザクション延期キューに置かれます。このようなメッセージを処理のために使用できるようにするには、`/DEQUEUE SUSPEND TRAN` コマンドを発行します。このコマンドを発行すると、トランザクションはトランザクション作動可能キューまたはトランザクション逐次キューに移されます。

トランザクション延期キューにトランザクションを置く IMS が、`/DEQUEUE SUSPEND TRAN` コマンドを発行する IMS と異なっても構いません。この場合、メッセージ (MSGQ) 構造を共用する他の IMS システムが `/DEQUEUE SUSPEND TRAN` コマンドを発行する必要があります。

関連情報: トランザクション延期キュー、トランザクション作動可能キュー、またはトランザクション逐次キューについては、260 ページの『キュー・タイプ』を参照してください。

AOI トランザクションのスケジューリング

トランザクションがローカルで実行されるようにするためには、AOI 出口ルーチンによってスケジュールされるすべての AOI トランザクションを逐次として定義しなければなりません。AOI トランザクションを共用キューに置くことによって、コマンドがその IMS に適用されるかどうかにかかわらず、任意の IMS がそのトランザクションを処理できるようになります。

共用キュー環境での MSC の計画

複数システム結合機能 (MSC) ネットワーク内の IMS システムは、共用キュー環境外の IMS システムに接続された共用キュー環境の一部とすることができます。

IMSplex 内の各 IMS は、以下のいずれかとして機能することができます。

- 端末または MSC リンクからのメッセージを受信するフロントエンド
- フロントエンドから受信したメッセージを処理するバックエンド
- MSC リンクによってメッセージを受信し、他の IMS システムにメッセージを送信する中間サブシステム

2 つの IMS システムに MSC リンクを定義することができますが、メッセージはこれらの IMS システムのうちの 1 つから共用キューに置かれ、IMSplex 内の他方の IMS による処理のために取り出されます。したがって、これらのメッセージは MSC リンクでは送信されません。

例外: このような IMS システムの 1 つが、共用キューが使用不能な状態で実行されている場合には、MSC リンクが使用されます。このような状態は、共用キュー環境へのマイグレーション時に発生する可能性があります。

共用キュー環境での MSC の使用を計画するには、以下の処置を行ってください。

- すべてのリモート IMS システムを MSNAME マクロ (SYSID および NAME) またはタイプ 2 の `CREATE MSNAME` コマンドで指定し、ローカル IMS 内で

固有の SYSID を割り当てます。これらのリモート MSNAME は停止されたままで、IMSplex 内の任意の IMS からリモート IMS へメッセージを経路指定するためにのみ使用されます。

- IMSplex 内の IMS ごとに、MSNAME マクロ またはタイプ 2 の CREATE MSNAME コマンドを使用して、MSC リモート・システム ID (SYSID) を定義します。
- ローカル IMS 内では、すべての SYSID を、IMSplex 内および MSC ネットワーク内のすべての IMS システムの間で固有にします。
- MSC リンク (バックエンド IMS) を持つ IMS に対してリモート・トランザクションと LTERM を定義します。また、それらをフロントエンド IMS に対しても定義します。定義しない場合、IMS は、宛先作成出口ルーチン (DFSINSX0) を呼び出して、宛先がローカルであるかリモートであるかを判別し、さらにそれがトランザクションであるか LTERM であるかを判別します。

MSC リンクがフロントエンド IMS に存在している場合、MSC を共用キュー環境内の APPC および OTMA トランザクションに使用することができます。会話はローカル IMS で保持されますが、リモート IMS には運ばれません。したがって、リモート・アプリケーション・プログラムは、SETO DL/I 呼び出し、または CPI-C verb SEND_ERROR および DEALLOCATE_ABEND を発行することはできません。また、リモート IMS は、別の APPC 会話を割り振る場合には、MPP 領域に関連するデフォルト・ユーザー ID を受け取り、セキュリティを检查するために元の APPC ユーザー ID を使用しません。

共用キュー環境でのパフォーマンスのチューニング

共用キュー環境用に定義した構造およびパラメーターによって、システムの実行方法が決まります。

システム・パフォーマンスを最適化するには、以下の構造およびパラメーターの調整を試みてください。

- IMS 実行パラメーターのサイズ
 - LGMSGSZ=
 - QBUF=
 - QBUFHITH=
 - QBUFLWTH=
 - QBUFMAX=
 - QBUFPCTX=
 - QBUFSZ=
 - SHMSGSZ=
- CQS ローカル構造定義パラメーター SYSCHKPT= の値
- CQS グローバル構造定義パラメーターの値
 - OVFLWMAX=
 - STRMIN=
- カップリング・ファシリティ上の共用キュー構造のサイズ
- カップリング・ファシリティ上の z/OS システム・ログ構造のサイズ

- z/OS システム・ログ構造をバックアップする z/OS システム・ログ・データ・セットの数
- 構造チェックポイントの頻度

共用キュー環境での高速機能トランザクションに関する推奨

共用高速機能トランザクションで、高速機能入力編集/経路指定出口ルーチン (DBFHAGU0) に LOCALONLY を指定する場合、EMH バッファ・プールのサイズを変更しないでください。非共用キュー環境の場合と同じバッファ・プール・スペースが、共用キュー環境でも必要になります。

ただし、共用高速機能トランザクションの DBFHAGU0 に LOCALFIRST を指定し、そのトランザクションをローカルで処理する場合、フロントエンド EMH バッファとバックエンド EMH バッファの両方がローカル IMS システムに必要となります。この結果、EMH バッファ・プールの使用量が増加します。

いずれの場合も、EMH バッファ使用量を分析し、必要に応じてスペースを調整してください。

関連情報: 共用キュー環境におけるパフォーマンス・チューニングの詳細については、538 ページの『共用キュー環境でのパフォーマンスの計画』を参照してください。

第 16 章 IMS 環境でのデータ共有

複数の IMS システムが並行して IMS データベースにアクセスすると、これらのシステムはデータを共有し、ユーザーはデータ共有環境にあることになります。データ共有環境では、それぞれがシスプレックス内の異なる z/OS システム上で稼働する、複数の IMS システムを管理します。このような構成では、より精巧な操作手順とリカバリー手順が必要になります。

ここでは、複数の IMS オンライン・システムまたはバッチ・システムに、共通データベース内にあるデータへの並行アクセスを許可する方法について説明します。IMS を使用してデータへのこのような共通アクセスを制御する場合は、データベース・リカバリー管理 (DBRC) によって提供されるデータ共有サポートを使用する必要があります。

データ共有の概要

IMS システムには、すべての宣言されたアプリケーション・プログラムにとって使用可能にできる、または共有できるデータを持つデータベースが含まれています。データベースへのアクセスは、アプリケーション・プログラムの PSB に定義されている特性です。データ共有サポートにより、別々の IMS システム内のアプリケーション・プログラムがデータベースに同時にアクセスすることが可能です。

IMS システムはロック管理を使用して、あるアプリケーション・プログラムによるセグメント・レベルでのデータベース変更が、他のアプリケーション・プログラムがそのセグメントのデータにアクセスする前に、完全にコミットされるようにします。

IMS システム間でのデータ共有は、シスプレックス環境と非シスプレックス環境の両方でサポートされています。

- シスプレックス・データ共有 とは、異なる z/OS オペレーティング・システム上にある IMS システム間のデータ共有のことです。IRLM では、データベースへのアクセスを制御するため、カップリング・ファシリティーを使用しています。
- 非シスプレックス・データ共有、別名ローカル・データ共有 とは、単一 z/OS オペレーティング・システム上にある IMS システム間のデータ共有のことです。カップリング・ファシリティーは、使用できますが、必要ではありません。

データ共有では、2 つのロック制御レベルを使用できます。

- データベース・レベル共有 では、アプリケーション・プログラムが更新を行っている間、データベース全体がロックされます。ロックを使用すると、データベース健全性を脅かす可能性のある並行データベース・アクセスやアプリケーション・プログラムのスケジューリングを防止できます。DEDB エリア・リソースでは、これはエリア・レベルの共有 と呼ばれます。
- ブロック・レベル共有 では、グローバル・ブロック・ロックを使用し、データベースへの並行アクセス時のデータベース健全性を維持できます。データ

ベース全体ではなく、ブロックがロックされます。複数のアプリケーション・プログラムが別々のブロックを更新する場合、それらのプログラムは同時にデータベースを更新することができます。

単一の z/OS オペレーティング・システム内、またはシスプレックス内で、複数の IMS オンライン・システムまたはバッチ・システムがデータベースを並行更新できます。データベース内では、リソースはブロック・レベルで確保されます。OSAM データベースでは、このブロックとは直接アクセス・ストレージ・デバイス上に格納される物理ブロックです。VSAM データベースおよび DEDB では、このブロックとは制御インターバル (CI) です。共用データに並行アクセスする複数の IMS オンライン・システムまたはバッチ・システムに対しては、データ保全性が保持されます。順次従属セグメントがサポートされます。DEDB に対するブロック・レベル共用は、IMS オンライン・システム間でのみ行われます。

データ・アクセスは保護されるので、次のようになります。

- 1 つの IMS システムがデータベースの更新を許可されると、許可されているその他の IMS システムは読み取り専用アクセスを持つこととなります。
- 許可されている複数の IMS システムが、読み取りアクセスまたは読み取り専用アクセスによってデータベースを並行スケジュールできます。

データ共用環境内の IMS システムは、オンラインにもバッチにもなります。エリア・レベル共用では、関与している IMS システムはオンラインでなければなりません。IMS ユーティリティは、データベース・レベル共用で動作するバッチ・システムと考えられます。

データ共用構成のサポートには、相違点があります。一般に、完全なデータベースは、1 つのデータ・リソースと見なされます。データベースが IMS オンライン・システム内で呼び出されたり、バッチ IMS システムとして呼び出される場合、個々のアプリケーション・プログラムがそれを処理できなければなりません。しかし、例えば、以下のような場合、データベースが使用不能になることもあります。

- データベースが 1 つの IMS システムのみにより使用されている。
- データベースにリカバリーが必要だとフラグが立てられている。
- バックアップ手順が進行中である。

しかし、高速処理データベース (DEDB) では、データ・リソースは分割されます。DEDB の場合は、個別のエリアがデータ・リソースの 1 つの単位と見なされます。データ共用を説明するこのトピックでは、「データベース」という用語は、特に断りのない場合、DEDB エリアを意味します。

高可用性ラージ・データベース (HALDB) では、データベースを 1 つ以上の区画に分割できます。HALDB データベースの場合は、個別の区画がデータ・リソースの 1 つの単位と見なされます。データ共用で言う「データベース」とは、特に断りが無い限り、HALDB 区画を意味します。

データ共用に関連する制約事項は次のとおりです。

- バッチ IMS サポートでは、MSDB と DEDB は使用できません。

- 高速機能を使用する IMS オンライン・システムのみが、DEDB を共用できます。
- データ共用サポートでは、MSDB および GSAM データベースは使用できません。

DBRC およびデータ共用サポート

1 つ以上の z/OS オペレーティング・システムでのシステムによるデータベースへの並行アクセスは、共用データベース・リカバリー管理 (DBRC) の RECON データ・セットで制御されます。IMS システムは DBRC へ自動サインオンしますが、これにより、現在共用アクセスに関与している IMS システム (およびユーティリティー) を DBRC が確実に認識できるようになります。

また、データベースへのアクセスがシステムに許可されるかどうかは、許可されている共用の程度、および RECON データ・セット内のその他の状況標識によって決まります。

データ保全性を維持するために、RECON データ・セット内の状況標識によってデータベースに対する並行アクセスおよびリカバリー・アクションが制御されます。ある特定のデータベースは DMB 番号 (共用しているすべての IMS システムにそのデータベースを一意的に識別させるもの) を持っている必要があるため、データ共用 IMSplex ではこの共通 RECON データ・セットが必要です。DBRC が RECON データ・セットに記録する DMB 番号は、データベースが DBRC に登録される順序と関係があります。複数の RECON データ・セットを使用すると、各 RECON データ・セットの同一の DMB 番号が異なるデータベースを表すことがあります。このような状態が発生すると、データベースが損傷を受けることがあります。

DBRC への登録

データ共用の対象となるデータベースは、RECON に登録する必要があります。登録されている個々のデータベースの現在の状況は、共用の対象となっているかどうか、および共用の有効範囲を反映しています。有効範囲の概念は、次のようないくつかの考え方から構成されています。

- アクセスのタイプ：読み取りか更新か
- データベース内で同時に複数のアクセスが発生できるかどうか
- アクセスを必要としている IMS が同一 z/OS オペレーティング・システム上にあるか、それとも異なる z/OS オペレーティング・システム上にあるか

個々の IMS データベースの共用レベルは、INIT.DB コマンドの SHARELVL の指定によって指定します。

データベースは、データベース・レベルまたはブロック・レベルで共用可能であり、あるいはまったく共用しないことも可能です。データベースの共用のタイプは、そのデータベースを使用するすべてのアプリケーション・プログラムのデータ保全性と可用性のニーズに基づいて、決定する必要があります。

複数の IMS システムが並行して作動し、データベース・レベルでデータを共用する場合、これらの IMS システムは同じ z/OS オペレーティング・システムに置くことも、異なる z/OS オペレーティング・システムに置くこともできます。

図 32 は、共用リソースとしてのデータベースを示し、エリア・レベルの共用を説明しています。DEDB を構成する 1 つ以上のエリアを、共用リソースにすることができます。また、RECON データ・セットは、オンライン IMS またはバッチ IMS の DBRC 制御部分による読み取り中および更新中の状態で、VSAM キー順データ・セット (KSDS) としても共用されます。IMS オンライン・システムでは、DBRC は初期設定時に制御領域によって自動的に始動され、別の領域で実行されます。バッチ IMS では、DBRC 制御は、データベース・レベルの共用の場合と同様に、バッチ領域内に含まれます。

RECON データ・セットは、次のものを追跡します。

- 各データベースに許可されている共用レベル
- プロセス権限が現在付与されているデータベースまたはエリア
- 関与している IMS システム
- それらのシステムすべての状況
- リカバリーという観点からのデータベースの状況

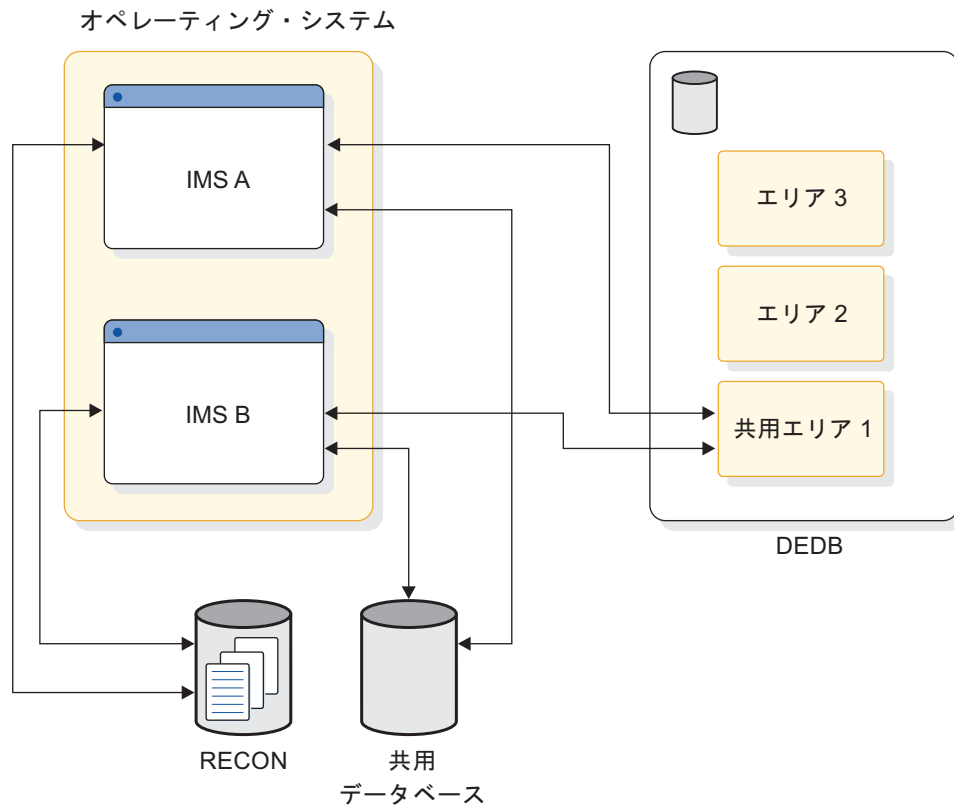


図 32. データベース・レベル共用

データ共用環境の命名規則

データ共用の使用を決定し、その他のデータベースで DBRC を使用して制御する場合は、データベース・データ・セットに対して確立したデータ・セット命名規則を検討する必要があります。

選択したリソース名を使用してデータベースがデータ共用に関与していることを示すことを考慮してください。また、共用データベースにアクセスするアプリケーション・プログラムも読み取り専用特性を示す PSB 名を持っています。

以下のリソースの命名規則を検討してください。

- データベース、およびその DD 名とデータ・セット名
- イメージ・コピーおよび変更累積データ・セット名
- オンライン・ログ・データ・セット (OLDS)
- システム・ログ・データ・セット (SLDS)
- PSB およびアプリケーション・プログラム名
- トランザクション・コード
- IMS システムおよび IRLM の名前の識別

データ共用サポートがない場合のデータベース保全性

データ共用サポートがない場合は、データベースの保全性に関する責任は管理と操作にあります。

データ保全性が保護されない状況として、次の 2 つがあります。

- 単一の z/OS オペレーティング・システムにおいて、データベース JCL がファイル属性指定 SHR を指定していると、複数の IMS オンライン・システムまたはバッチ・システムが同時実行した場合、保全性を維持することなしに、データベースへの並行アクセスが行われる可能性があります。
- 複数の z/OS オペレーティング・システムで複数の IMS システムが実行され、データベースが共用 DASD 上にある場合、これらの IMS システムはデータベースに並行アクセスする可能性があります。この並行アクセスは、ファイル属性指定 OLD が指定されていても、JCL で直接制御することはできません。

アプリケーションのデータ・アクセス方法

アプリケーション・プログラムは、更新アクセス、読み取りアクセス、または読み取り専用アクセスという 3 つの方法でデータにアクセスできます。

特定のデータベース・アクセスのためにプログラム連絡ブロック (PCB) を構成するステートメントのグループの一部として PROCOPT キーワード上にアクセスのタイプを示します。アプリケーション・プログラムのこれらの処理オプションは、プログラム仕様ブロック (PSB) 内で宣言され、データ・アクセスと変更に関するアプリケーション・プログラムの意図を示します。

アプリケーション・プログラムで挿入、削除、置換、またはそれらのアクションの組み合わせを実行する場合、そのアプリケーション・プログラムは更新アクセスを持ち、これは PROCOPT=A キーワードで指定されます。排他的アクセスを行うオンライン・アプリケーション・プログラムは、更新アクセスを持つものとしても解釈され、更新アクセスは、PROCOPT=E キーワードで指定されます。

データベースへのアクセスが必要であっても、データを更新しないアプリケーション・プログラムは、2 つの方法でデータベースにアクセスできます。それらのアプリケーション・プログラムは、保留中の変更がすべて、その変更を開始したアプリ

ケーション・プログラムによってコミット済みであるという保証のもとでデータにアクセスできます。このタイプのアクセスは読み取りアクセス であり、PROCOPT=G キーワードで指定されます。あるいは、アプリケーション・プログラムは、データ状況の保護を指定しない場合、コミット済みでないデータを読み取ることもできます。このタイプのアクセスは読み取り専用アクセス と呼ばれ、PROCOPT=GO キーワードで指定されます。

PROCOPT キーワード値について詳しくは、「IMS V15 アプリケーション・プログラミング」を参照してください。

IMS システムのデータベースの共用方法

このトピックでは、データベース・アクセスの確立方法、ブロック・レベルおよびデータベース・レベルでのデータ共用、および各共用レベルでのデータ・アクセス制御方法について説明します。

データベース・アクセスの確立

データベースへのアクセスを要求する IMS システムが、そのデータベースをどのように使用する計画であるかを指定するには、DATABASE マクロで ACCESS キーワード・パラメーターを使用します。

/START コマンドで ACCESS キーワードを使用して、データベースへのアクセスを要求するIMS システムがどのようにデータベースの使用法を計画するかを表示します。更新意図がある場合は、システム全体がそのデータベースに対する更新アクセスを必要とすることを意味します。データベース PCB が排他使用を必要とする場合は、システム全体がそのデータベースに対する更新アクセスを必要とします。更新アプリケーション・プログラムが更新アクセスを必要としない場合は、どのような種類の読み取りアクセスが必要であるかを確認しなければなりません。オンライン・システムがブロック・レベル共用を使用する場合は、通常、読み取りアクセスを宣言します。

アクセス・モードは、更新 (UP)、排他 (EX)、読み取り (RD)、または読み取り専用 (RO) のいずれかです。

IMS システムがデータベースの排他使用を要求すると、そのアクセスを排他的 アクセスとして宣言できます。排他的アクセスでは、IMS システムは、挿入、削除、置き換え、またはこれらを組み合わせたアクションを実行することができます。このため排他的アクセスでは、ブロック・レベルまたはデータベース・レベル構成でのデータ共用ができません。

IMS システムが挿入、削除、置き換え、またはこれらを組み合わせたアクションを実行する必要があり、データを共用しなければならない場合は、そのアクセスを更新 アクセスとして宣言できます。

データベースからの読み取りを計画しているが、データベースを更新しない IMS システムは、読み取りアクセスまたは読み取り専用アクセスのいずれかを使用できます。読み取りアクセス では、IMS システムは、保留中の変更が変更を要求したアプリケーション・プログラムによってコミットされているという保証のもとでデータにアクセスできます。

読み取り専用アクセス では、IMS システムは、コミットされていないデータを読み取ることができます。

IMS システムに対するデータベース・アクセスの指定方法は、バッチ・システムを実行しているか、それともオンライン・システムを実行しているかによって異なります。

データベース・イメージ・コピー・ユーティリティーのオンライン・バージョンを使用する予定の場合は、ACCESS=UP を指定する必要はありません。イメージ・データ・セットの作成中は、ユーティリティーのアクセスはある程度制限されますが、アクセスと許可は DBRC によって管理されます。

IMS バッチ・システムでのデータベースに対するアクセスの宣言

IMS バッチ・システムでは、データベース・アクセスは、アプリケーション・プログラムの PCB または SENSEG ステートメントに指定されている PROCOPT 値の最も高い値に直接対応しています。例えば、ある PCB の PROCOPT が G に設定されており、同一データベース用の別の PCB では PROCOPT が I に設定されている場合は、値 I が使用されます。つまり、そのデータベースに対するアクセスは更新アクセスです。

オンライン・システムでのデータベース・アクセスの宣言と変更

オンライン IMS では、システム定義中に DATABASE マクロの ACCESS キーワードでアクセスを指定します。アクセスはそれぞれの個別データベースに対して宣言され、そのデータベースに対してスケジュールされるすべてのアプリケーション・プログラムに適したアクセスを反映します。

関連資料: ACCESS キーワードの詳しい説明については、「IMS V15 システム定義」を参照してください。

オンライン IMS システムが個別データベースにアクセスする方法を動的に変更する場合、/START コマンドを使用します。これについては「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」で説明しています。

データベース・レベル共用における IRLM の使用

IMS がサービスをロックするために内部リソース・ロック・マネージャー (IRLM) を使用する場合、IRLM がすべてのロッキング・サービスを提供します。

IRLM が IMSCTRL マクロの IRLMNM パラメーターでアクティブにされなかった場合に、IRLM をアクティブにしたければ、実行 JCL に IRLM=Y をコーディングします。IRLMNM パラメーターが IMSCTRL マクロでも実行 JCL でも指定されず、IRLM=Y が実行 JCL で指定されると、IMSCTRL マクロは指定されたデフォルトの IRLM 名 ('IRLM') を使用します。

すべての IMS バッチ・インスタンスおよびオンライン・インスタンスが 1 つ以上の通信 IRLM をデータベース・レベル共用で使用すると、データベース拡張とバッファの無効が読み取り専用 IMS インスタンスに通知されます。これは、フル・ブロック・レベルのデータ共用の場合と同じですが、ただしロッキングのアクティビティは行われません。

データベース・レベル共用に IRLM を使用する場合は、保全性なしで読み取りを行うアプリケーション・プログラムは、有効ではあるが、恐らくコミットされていないデータをより頻繁に読み取ることになります。これによって保全性なしでの読み取りの問題が解決される保証はありませんが（「IMS V15 アプリケーション・プログラミング」を参照）、データベース・レベル共用は改善されます。変更されたブロックまたは制御インターバル (CI) のすべてが DASD に書き込まれる順序（それらが書き込まれる時期も含む）により、保全性なしで読み取りを行うアプリケーション・プログラムでは、ある実行と他の実行との間でやはり不整合が生じる場合があります。

IMS を使用すると、データベース・レベル共用とブロック・レベルのデータ共用の両方のレベルで、バッファの一貫性のためにカップリング・ファシリティを使用することができます。

ブロック・レベル共用

ブロック・レベル共用には内部リソース・ロック・マネージャ (IRLM) が必要です。各 z/OS オペレーティング・システムが、ブロック・レベル共用に関与している IMS オンライン・システムを含んでいる場合、IRLM コンポーネントはそれぞれの z/OS オペレーティング・システムに存在している必要があります。

IRLM は、データ共用に関与しているリソースへのアクセスを制御するロックに対するすべての要求を管理します。

それぞれの IRLM は、関連付けられている IMS システムからの要求に応答します。このために、保留中および待機中のロックの状況だけでなく、関与している IMS システムのレコードも保持します。

IMS オンライン・システムによって使用されている場合、IRLM はオンライン・システムのアプリケーション・プログラムによって並行アクセスされるデータベース・リソースを制御するロック管理も提供します。このようにして、IMS オンライン・システムと関連付けられている IRLM は、このシステムに対してほとんどすべてのロッキング・サービスを提供します。

IRLM は相互に通信して、さまざまなデータベース・ロック用に保持されているロックの状況を管理します。

IRLM を使用する場合は、ブロック・レベルのデータ共用に必要な適切な SHAREOPTIONS を指定して、DBRC 登録の SHARELVL=1 の VSAM データベースを VSAM に定義する必要があります。これには、VSAM の定義変更を必要とする場合もあります。

シスプレックス・データ共用内の複数の IRLM

複数の IMS システムが同一 z/OS オペレーティング・システムで実行されている場合は、複数の IRLM がブロック・レベルでのデータ共用の制御に関与できます。

このことが当てはまるのは、いくつかの IMS オンライン・システムのうちの 1 つが別の z/OS オペレーティング・システムにインストールされる前に、それらのシステムでデータ共用の使用についてテストされているような場合です。

複数の IRLM が単一の z/OS オペレーティング・システム上で実行されると、特殊なケースのシスプレックス・データ共有が発生します。この場合、グローバル共有を宣言して、個々の IRLM に固有の z/OS サブシステム名と IRLM を識別する番号 (ID) を指定します。

IRLM プロシージャの設定

活動化できるそれぞれの IRLM コンポーネントごとに、システム・コンソールのオペレーターが z/OS START コマンドを入力したときに呼び出されるプロシージャが必要です。

このようなプロシージャの例としては DXRJPROC があり、これは「IMS V15 システム定義」に記載されています。1 つのオペレーティング・システムで複数の IRLM プロシージャを使用する予定の場合は、その他のメンバーを IMS.PROCLIB または SYS1.PROCLIB に入れてください。その他の IRLM には、固有な ID が必要であり、パラメーター値の変更も必要です。プロシージャ名は、z/OS に対して確立されている IRLM サブシステム名と同じにすることはできません。

個々の IRLM コンポーネントに対して、プロシージャ・パラメーターは次のものを指定します。

- コンポーネントの識別名
- その制御の有効範囲
- シスプレックス・データ共有情報
- 障害状態での IRLM のアクション
- 仮想記憶域の量
- パフォーマンス要因

ブロック・レベル共有をサポートして実行する領域のための PARM1= および PARM2= 定位置パラメーターを、以下の表に示します。

表 18. IRLM 領域パラメーターのカテゴリと目的：

カテゴリー	パラメーター (PARM1 と PARM2)	目的
コンポーネント名	IRLMID	この IRLM の ID。
	IRLMNM	z/OS サブシステム名。
制御範囲	SCOPE	z/OS 間通信または z/OS 内制御。
データ共有	IRLMGRP	z/OS システム間カップリング・ファシリティー グループの名前。
	MAXUSRS	グループのユーザーの最大数。
	LOCKTAB	グループにより使用されるロック構造。

表 18. IRLM 領域パラメーターのカテゴリと目的 (続き):

カテゴリー	パラメーター (PARM1 と PARM2)	目的
ストレージ	MAXCSA	使用される z/OS 共通サービス域 (CSA) の最大量。このパラメーターは、IRLM 2.2 以上には適用されません。
	PC	専用ストレージ内の制御ブロックを指定します。IRLM 2.2 は、常に、PC=YES で実行します。
	PGPROT	共通ストレージに常駐する IRLM ロード・モジュールを MVS PAGE PROTECTED STORAGE に配置するか否かを決定します。デフォルトは YES です。
パフォーマンス	DEADLOK	デッドロック検出タイミング。
トレース	TRACE	IRLM 初期設定時にオンになるトレース・タイプ DBM、XCF、および SLM (EXP、INT、および XIT は常にオン)。

IRLM の識別

IRLMID パラメーターでは、IRLM を識別するための 10 進数 (1 から 255) を指定します。IRLMID は等価の 16 進数に変換されます。各 IRLM には固有の番号を割り当てる必要があります。IRLM エラーおよび状況メッセージには、メッセージを発行した IRLM のサブシステム名と ID が含まれています。

IRLMNM パラメーターは、この IRLM に割り当てられる 1 バイトから 4 バイトの z/OS サブシステム名を指定します。同一システム上で同時に複数の IRLM が使用されない限り、各オペレーティング・システムのサブシステム名として「IRLM」を使用できます。1 つのシステムで並行して IRLM が使用される場合は、サブシステム名は固有でなければなりません。

IRLM の有効範囲の指定

シスプレックス共用を使用する場合であっても、非シスプレックス共用を使用する場合であっても、どのような種類のデータ共用制御が IRLM コンポーネントに必要なかを指定する必要があります。SCOPE パラメーターでこれを行います。SCOPE=GLOBAL は、シスプレックス・データ共用が制御されることを指定します。これは、複数システム間のリソースの共用です。

SCOPE=LOCAL は、データ共用を非シスプレックス・データ共用に限定します。これは、すべてが単一 IRLM に識別されている IMS システム間でのリソースの共用です。カップリング・ファシリティが使用できない場合は、SCOPE=LOCAL が必要です。SCOPE=NODISCON は、シスプレックス・データ共用が特別な DISCONNECT 規則で制御されることを指定します。SCOPE=GLOBAL または NODISCON には、1 つのカップリング・ファシリティが定義されている必要があります。

IRLM デッドロック管理

IRLM は、デッドロック管理を提供します。DXRJPROC プロシージャで DEADLOK パラメーターを使用すると、IRLM がグローバル・デッドロック検出を開始する前に発生するローカル・デッドロック検出間隔とローカル・サイクルの数

を指定できます。ローカル・デッドロック検出間隔は、秒単位またはミリ秒単位で指定できます。z/OS コマンド MODIFY IRLMPROC を使用して、DEADLOK 値を動的に変更することもできます。

データ共有環境では、関与している IRLM が DEADLOK パラメーターを、指定された最大値と同期させます。このパラメーターを、データ共有グループ内の個々の IRLM の場合と同じにしなければなりません。

関連資料:

- DXRJPROC プロシージャおよび DEADLOK パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。
- コマンド MODIFY IRLMPROC について詳しくは、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

IRLM のためのシステム初期設定

IRLM 用にシステムを初期設定するには、IRLM を z/OS サブシステムとして定義する必要があります。

z/OS サブシステムとしての IRLM の定義

インストール・システムによっては、4 バイトのサブシステム名が割り当てられます。この名前として「IRLM」を使用できます。これは、システム・オペレーターがコンソールに送られたシステム・メッセージを認識する際に役立ちます。この名前は、z/OS システムに固有でなければなりません。1 つのオペレーティング・システムで複数の IRLM サブシステムを実行させる場合は、これらの名前は異ならなければなりません。

その他に各 IRLM を識別するのは、1 から 255 までの数字です。この識別は、IRLM 間で互いを識別するのに必要です。例えば、データを共有する 2 つの IRLM は、1 および 2 という ID を持つことができます。システム・コンソール・オペレーターに送られる状況およびエラー・メッセージには、IRLM サブシステム名と ID が含まれます。2 つの IRLM は、単一の z/OS で実行している場合も、または同じデータ共有グループのメンバーである別々の z/OS で実行している場合も、異なる ID 番号が必要です。

IRLM サブシステム名のもう 1 つの考慮事項は、始動プロシージャに使用されている名前と同じであってはならないということです。このプロシージャは SYS1.PROCLIB のメンバーであり、START コマンドでオペレーターが使用します。固有の START プロシージャが、各 IRLM サブシステムに対して存在していなければなりません。プロシージャ・ライブラリーのメンバーは、別の ID 番号および他の制御パラメーターによっても区別されます。

IRLM を実行するには、サブシステムを z/OS プログラム特性テーブル (PPT) に追加する必要があります。

サブシステムのシステム初期設定ルーチンは、ヌルでなければなりません。

関連資料:

- IRLM をサブシステムとして定義し、サブシステム名テーブルに項目を追加する方法について詳しくは、「z/OS MVS 初期設定およびチューニング ガイド」を参照してください。
- PPT への IRLM の追加について詳しくは、151 ページの『z/OS プログラム特性テーブルへの IRLM 項目の追加』を参照してください。

IRLM トレース出力印刷の許可

IRLM サブシステムで発生するアクティビティのトレースを行うには、z/OS コンポーネント・トレース機能、または IRLM プロシージャの DXRJPROC の TRACE=YES オプションを使用します。

CTRACE レコード

IRLM は、z/OS コンポーネント・トレース (CTRACE) 形式でトレース出力を作成します。これにより、IPCS CTRACE の形式設定、マージ、および検索のルーチンを使用して、バッファ・データをプロセスすることができます。ダンプ内と外部書き出しプログラム・データ・セット内の IRLM トレース・バッファは、どちらも IPCS を使用して形式設定することができます。IRLM トレース・フォーマット設定ロード・モジュール (DXRRLFTB) とバッファ検出ルーチン・ロード・モジュール (DXRRL186) は、IPCS に対して使用できなければなりません。

定様式ダンプ出力用に調整する。

IRLM は、z/OS の SDUMP 機能を使用します。ダンプ出力のオフライン印刷は、z/OS 印刷ダンプを使用して行われます。

関連資料:

- SDUMP の詳細については、「IMS Version 15 Diagnosis」を参照してください。
- IRLM プロシージャの DXRJPROC については、「IMS V15 システム定義」を参照してください。

シスプレックス・データ共有の IRLM の定義

シスプレックス・データ共有環境を実装する場合、IRLM が属するデータ共有グループ、そのグループが使用するロック構造、およびそのグループ内の最大ユーザー数を定義する必要があります。

データ共有グループの定義

データ共有グループに属する IRLM に関して、データ共有グループの名前とロック構造の名前を IRLM 始動プロシージャに指定する必要があります。このグループのすべての IRLM が、同一データを共有できます。グループの個々の IRLM は、以下の要件を満たしていなければなりません。

- 固有の IRLMID を持っている。
- GROUP パラメーターを使用して、同じデータ共有グループ名を指定する。
- LOCKTABL パラメーターを使用して、同じロック構造を指定する。

これらの定義を IRLM 始動プロシージャに指定できますが、推奨される方法は、CFNAMES 制御ステートメントを使用して定義することです。CFNAMES 制御ス

ステートメントは、シスプレックス・データ共用環境用にデータ共用グループを定義します。CFNAMES 制御ステートメントを使用しない場合は、ロック構造名は IRLM 始動プロシージャーから呼び出されます。

ロック構造の定義

シスプレックス・データ共用に関与している (GROUP=IRLMDS パラメーターで同じ z/OS システム間カップリング・ファシリティを名を指定する) IRLM それぞれに、同一のロック構造を指定しなければなりません。LOCKTABL パラメーターを使用して、ロック構造名を指定します。

データ共用グループのユーザー数の定義

データ共用グループのユーザーの最大数を指定する必要があります。MAXUSRS パラメーターで 2 から 32 の値を指定することによって、これを行います。

更新アクティビティーを行うデータベース・レベルでのデータ共用

このような共用では、1 つの IMS バッチ・システムまたはオンライン・システムが更新アクセスを持つことができます。他のすべての IMS システムは、読み取り専用アクセスを持ちます。

全機能データベースでは、コミットされていないデータを読み取ることができます。高速機能データベースでは、コミットされていないデータは存在しませんが、トランザクションに対する更新のすべてがすでに書き込まれているわけではありません。このため、Get Next プロセスが実行されると、無効なポインターが参照されることがあります。

通常のアプリケーション・プログラム分離で、更新システムの健全性は保護されます。しかし、IMS オンライン・システムのロック管理に IRLM を使用できます。以下の図は、更新アクセスを持つオンライン・システムと同一のデータへの読み取り専用アクセスを許可されている 2 つのバッチ・システムとのデータ共用を示しています。

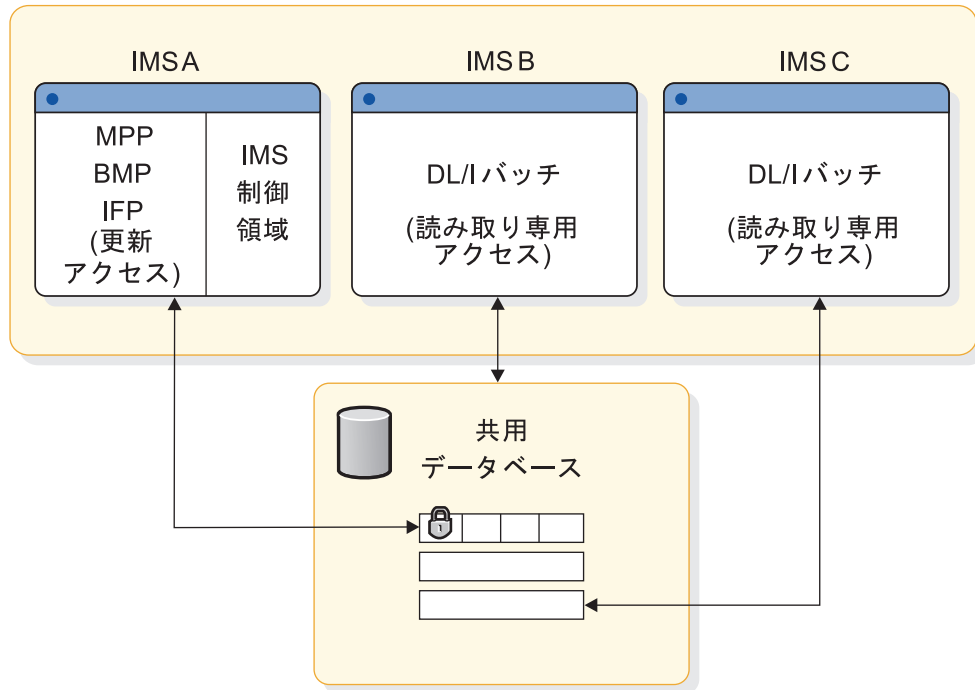


図 33. 更新アクセスによるデータベース・レベルでのデータ共有の例

複数の読み取りアクセスが行われるデータベース・レベルでのデータ共有

この種の共有により、権限を付与されている複数の IMS システムが、読み取りアクセスまたは読み取り専用アクセスでデータを同時に読み取ることができます。

構成例を以下の図に示します。この構成では更新は行えないので、データ保善性は損なわれません。

オペレーティング・システム

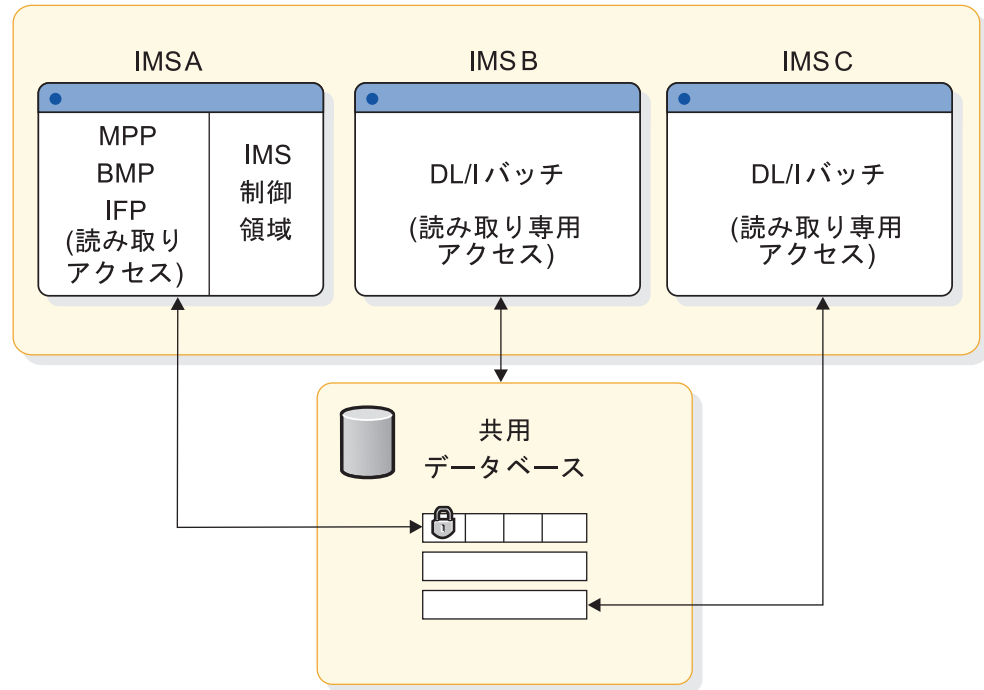


図 34. 読み取りアクセスによるデータベース・レベルでのデータ共有の例

ブロック・レベルでのデータ共有

全更新機能を持つオンライン IMS システムでは、システム相互間でデータを共有でき、他のバッチ IMS システムがコミット済みのデータの読み取りおよびデータの更新を行えるようになります。

このバッチ・システムでは、データ保全性が保護され、読み取り専用アクセスが使用できます。このような構成を以下の図に示します。

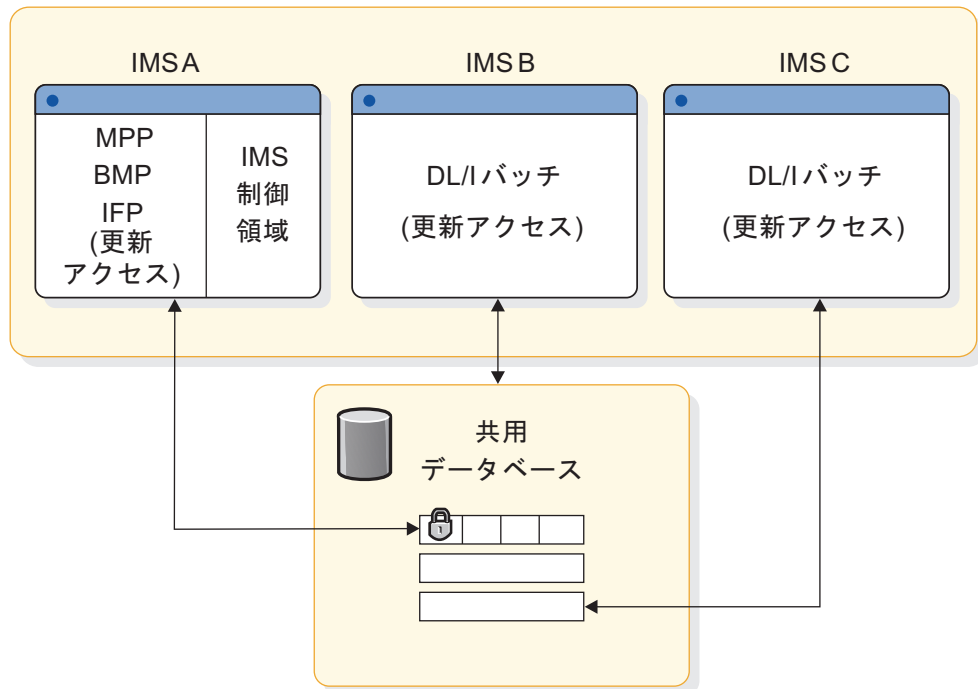


図 35. 更新アクセスによるブロック・レベルでのデータ共有の例

データを共有する IMS システムの調整

システム定義マクロを使用し、DBRC データ・セットおよびJCL を初期設定して、IMS システムを調整します。

ユーザーのデータ共有要件に合わせた IMS システムの調整

- システム定義マクロを使用して、次のことを行います。
 - IMS バッチ・システムに対する DBRC データ共有サポートを組み込む。
 - IRLM (ブロック・レベル共有に対して) を組み込む。
 - データベース・アクセス属性を宣言する。
- DBRC データ・セットと JCL を初期設定します。691 ページの『第 42 章 RECON データ・セットの初期化および保守』を参照してください。
- 実行 JCL と仮想記憶の使用法を調整します。308 ページの『実行 JCL の調整』を参照してください。

インストール・タスク

データ共有を使用する IMS システムを実行する前に、インストール関連のいくつかのタスクを完了しておく必要があります。

以下の表にタスクをリストし、各タスクがデータ共有に影響するかどうかを示し、完了しておく必要のある他のタスクを識別しています。

表 19. 追加のデータ共有アクティビティーを伴うインストール・ステップ

ステップ	インストール・タスク	データ共有なし でシステムに影 響する	データ共有あ りでシステム に影響する	追加タスク
1	システム・ライブラリーの作成 - IRLM	なし	あり	IRLM をシステム・ライブラリーに 追加する。
2	IMS データ・セットの割り振りと カタログ作成	あり	あり	RECON データ・セットを作成す る。 DBRC 制御を初期設定する。 DL/I 出口を調整する。
3	システム定義および JCL の作成	あり	あり	マクロおよび JCL を変更する。 システム定義を行う。
4	z/OS オペレーティング・システ ムの調整	なし	あり	IRLM サブシステムを追加する。
5	IMS パフォーマンス・オプション の調整	あり	あり	バッファを定義する。
6	DBDLIB の作成	なし	なし	なし
7	PSBLIB の作成	あり	あり	PROCOPT 値を検討する。
8	ACBLIB の作成	あり	あり	ACBGEN を実行する。
9	動的割り振りのための選択	あり	あり	オプションを検討する。
10	MFS ライブラリーの作成	なし	なし	なし
11	プログラム・ライブラリーの作成	あり	あり	呼び出しシーケンスを検討する。
12	初期データベース・ロードの実行	なし	なし	なし
13	セキュリティの確立	なし	なし	なし
14	IMS.MODSTAT の初期化	あり	あり	ACBLIB の使用、 DBDLIB および PSBLIB を調整する。
15	アクティブ・ライブラリーへのス テージング・ライブラリーのコピー	あり	あり	ACBLIB の使用を調整する。

データ共有環境におけるバッチ・システムのための DBRC および IRLM サポート

デフォルトでは、DBRC は、バッチ領域およびユーティリティ領域に対してデータ共有サポートを提供します。

デフォルトを変更するには、次のいずれかを行います。

- DFSIDEF0 をアセンブルして IMS 実行ライブラリーにバインドする (RMODE を指定する場合、RMODE=24 および AMODE=24 を指定する必要があります)
- DFSPBxxx IMS PROCLIB メンバー・データ・セットの値をオーバーライドする
- JCL の値をオーバーライドする

データ共用環境におけるバッチ・システムについてのデフォルトの IRLM サポート指定を定義するには、IMSCTRL マクロで IRLM= キーワードを使用します。IMSCTRL マクロで行われた指定は、バッチ JCL で IRLM= キーワードを使用することによりオーバーライドできます。

バッチ・システムでは、IRLM=Y または IRLM=N が指定できます。IRLM=N を指定した場合、次のようになります。

- IRLM は、バッチ・システムについてロッキングを実行しません。
- バッチ・システムは、ブロック・レベル共用に参加できません。

バッチ・システムが IRLM を使用しない場合、データベースの保全面性は DBRC により保証されます。DBRC は、データベースに対して現在許可されている他のすべての IMS システムおよびバッチ・ジョブが読み取り専用アクセスを持っている場合にのみデータベースに対する更新アクセスを持つバッチ・ジョブを許可します。

データ共用からのデータベースの除外

ACCESS=EX を指定したり、ACCESS= の値をブランクにしておく、指定されたデータベースは他の IMS システムによって並行アクセスされません。

共用レベル 0 を指定して、データベースを DBRC に登録してください。

他の IMS システムとデータを共用する必要のないデータベースは、管理者が定義する IMS オンライン・システムによって所有されます。DBRC を使用して、これらのデータベースへのアクセス許可を制御したくない場合は、これらのデータベースを登録しないでください。

実行 JCL の調整

データベースに対して更新意図があり、アクティブ DBRC で作動する任意のバッチ IMS に対しては、システム・ログは必要です。バッチ・システムが読み取りまたは読み取り専用アクセス意図だけを使用する場合は、システム・ログは必要ありません。

データ共用が含まれている構成を作成する場合は、以下のことを行う必要があります。

- システム・データ・セットを指定する。
- DISP=SHR でデータベースを指定する。
- 物理パスがデータに適合するように、データベース・データ・セットの配置および装置選択項目を指定する。
- IRLM プロシージャを作成する。

ACBLIB メンバー・オンライン変更プロセスは、OLCSTAT データ・セットを共用しているすべての IMS 間で調整されます。データ共用に関与しているすべてのシステムに共通になるように状況を調整しなければならないシステム・データ・セットは、次のとおりです。

- RECON データ・セット (RECON data set)
- DISP=SHR が指定されているデータベース・データ・セット
- IMS.ACBLIB、PSBLIB、および DBDLIB

- ランダム化ルーチンと DL/I 出口ルーチンを持つライブラリー

また、データ共用構成のために IRLM プロシージャを設定する必要もあります。データ共用環境のための IRLM プロシージャの作成については、299 ページの『IRLM プロシージャの設定』を参照してください。

z/OS オペレーティング・システムの調整

ブロック・レベル共用には、z/OS オペレーティング・システムの下で独立コンポーネントとしてインストールされている IRLM を使用する必要があるためいくつかの調整作業を行う必要があります。

次の作業を行う必要があります。

- VSAM データ・セット定義を調整する。
- z/OS システムを IRLM 用に調整する。
- シスプレックス全体でデータを共用している場合、IRLM を定義する。

共用オプションによる VSAM データ・セット定義の調整

データベース・アクセス方式が VSAM の場合は、どの程度の共用アクセスが必要かを、データ・セットの宣言によって VSAM に示します。

SHAREOPTIONS パラメーター値 (DFSMS DEFINE CLUSTER キーワードの) とそれらが指定する共用のタイプは、以下の表のとおりです。

表 20. SHAREOPTIONS パラメーターの指定

SHAREOPTIONS の値	共用のタイプ
(1,3)	共用なし、単一更新プログラム、または複数の読み取りアクセス
(2,3)	単一更新プログラムおよび複数の読み取りアクセス
(3,3)	複数の更新プログラム・アクセスおよび複数の読み取りアクセス

ブロック・レベル共用に関与し、VSAM アクセス方式を使用するデータベースの場合は、データ・セットの定義時に SHAREOPTIONS (3,3) パラメーターを指定しなければなりません。RECON データ・セットも KSDS としてアクセスされ、SHAREOPTIONS (3,3) を必要とします。

関連資料: 共用オプションを宣言してカタログする、DFSMS アクセス方式サービス・プログラムのステートメントの準備については、「z/OS DFSMS カタログのためのアクセス方式サービス・プログラム」を参照してください。

IRLM を使用する場合は、ブロック・レベルのデータ共用に必要な適切な SHAREOPTIONS を指定して、SHARELEVEL=1 の VSAM データベースを VSAM に定義する必要があります。これには、VSAM の定義変更を必要とする場合もあります。

IRLM のための z/OS システムの調整

IRLM コンポーネントには、以下のような他のシステム初期設定作業が必要です。

- z/OS サブシステムとしての IRLM の定義
- IRLM トレース出力を許可する。
- 定様式ダンプ出力用に調整する。
- シスプレックス・データ共有の IRLM の定義

これらの作業については、301 ページの『IRLM のためのシステム初期設定』で説明しています。

データ共有の開始および停止

データ共有オペレーションの開始および停止には、IRLM およびデータベース処理の開始および停止が含まれます。

IRLM の開始

IRLM を、それぞれが含まれるシステムのシステム・コンソールから、`START irlmproc,parms z/OS` コマンドを使用して開始します。

`irlmproc` は、IRLM プロシージャの固有名で、`parms` は、デフォルトのコマンドを指定変更する 1 つ以上の IRLM 制御パラメーターを表しています。

IRLM サブシステムを開始してから、バッチまたはオンラインの IMS を開始します。

IMS システムが IRLM に接続しようとしたときに、IRLM の初期設定が完了していない場合、IMS は DFS039A メッセージを出します。このメッセージは、IRLM がアクティブでないので、オペレーターに依頼して、メッセージに `retry`、`cancel`、または `dump` で応答するよう通知しています。

- IRLM を開始していなかった場合は、z/OS `START` コマンドを出して、IRLM がアクティブになるまで待ち、`RETRY` を応答します。すると、IMS は IRLM に接続を試行します。
- 単に IRLM の開始だけではなく、始動の調整に問題がある場合は、`CANCEL` を応答して、IMS システムを異常終了コード U0039 で終了させてください。

例: 次の環境があるとします。

- プロセッサ J で IMS システム (IMSJ) が IRLM (JRLM) を使用している。
- プロセッサ K で IMS システム (IMSK) が IRLM (KRLM) を使用している。

次に、以下のシナリオを想定します。

1. ハードウェア・エラーがプロセッサ K で発生した。
2. プロセッサ J で IMSK を再始動しようとし、データベース・バックアウトを実行する (しかし、IRLM 名前パラメーター `IRLMNM=KRLM` を変更しない)。

3. メッセージ DFS039A が KRLM が非アクティブであることを示す。

この場合、CANCEL を応答し、すぐに IMSK 始動を終了します。次に、IRLMNM= を JRLM に変更した後、プロセッサ J 上の IMSK を再始動します。

- DUMP と応答した場合、IMS は異常終了コード U0039 とストレージ・ダンプを出して終了します。

z/OS コンソール・オペレーターは、IRLM から出されるメッセージを注意して見る必要があります。これらのメッセージは、接頭部が DXR の 3 文字です。

IRLM の停止

IRLM は、IRLM を使用するすべての IMS システムが終了した場合のみ、正常に終了できます。IRLM を正常に停止するには、STOP *irlmproc* z/OS コマンドを使用します。

IRLM を異常終了するには、z/OS コマンド MODIFY *irlmproc,ABEND,NODUMP* を使用します。IRLM を異常終了し、ダンプを生成するには、z/OS コマンド MODIFY *irlmproc,ABEND* を使用します。

データベース処理の開始および停止

データ共有環境では、IMS タイプ 1 コマンドはローカルに、つまりローカル・システムのみに影響を及ぼすことも、またグローバルに、つまりシスプレックス内の共有 IMS システムすべてに影響を及ぼすこともできます。

UPDATE コマンドは、TSO SPOC などの自動化操作プログラムを使用して、個々の IMS システム宛てでも、IMSplex 内のすべての IMS システム宛てでも、経路を指定できます。

LOCAL キーワードを指定した (または GLOBAL キーワードが指定されていない) IMS コマンドは、これを入力したオンライン IMS システムのみに影響を与えます。GLOBAL キーワードを指定した IMS コマンドは、最初にこのコマンドを入力した IMS システムに効果を表し、これが成功すれば、次にシスプレックス内のその他の共有 IMS システムに影響を及ぼします。

ローカルまたはグローバルのどちらかで、/START DB コマンドまたは UPDATE DB START(ACCESS) コマンドを入力すると、DBRC は、各登録済みデータベースに、リカバリーまたはバックアウトの必要がないことを確認します。IRLM は、ブロック・レベル・データ共有環境内だけで、グローバル・コマンドと通信します。

以下の表は、登録されたデータベースに出されたときに、データ共有に影響のあるコマンドを要約したものです。データベースに入力されたコマンドは、全機能トランザクションに影響し、エリアに入力されたコマンドは、高速機能トランザクションに影響します。

表 21. IMS システム全体のデータベース制御コマンド

コマンド	DBRC によるアクション	ローカル IMS システムにおける影響	IMS システム共有における影響
ローカル			

表 21. IMS システム全体のデータベース制御コマンド (続き)

コマンド	DBRC によるアクション	ローカル IMS システムにおける影響	IMS システム共用における影響
/RMCHANGE	データベースを使用する IMS システムの共用レベルの変更が可能になる	ありません。	ありません。
/START AREA DATABASE LOCAL または UPDATE	リカバリーまたはバックアウトの必要がないデータベースまたはエリアを確認し、データベース開始を記録する	トランザクションにデータベースまたはエリアへのアクセスを許可し、また ACCESS= キーワードを使用すると、アクセス意図の変更が可能	ありません。
グローバル			
/DBDUMP AREA DATABASE GLOBAL	リカバリーまたはバックアウトの必要がないデータベースまたはエリアの確認、無許可のバッチ IMS システムへの戻りコードのセット、データベースのクローズの記録、およびアクセス・レベルの読み取りアクセスへの変更を行う	オンライン更新アクセスの防止	ローカルに入力されたかのようにコマンドを処理
/DBRECOVERY AREA DATABASE GLOBAL	以降の許可を禁止する	データベースの割り振りの防止	ローカルに入力されたかのようにコマンドを処理
/STOP AREA DATABASE GLOBAL	データベース状況を停止として記録する (以降の許可の制約)	トランザクションに対するデータベースまたはエリアのアクセスの停止	ローカルに入力されたかのようにコマンドを処理
/STOP ADS	エリア・データ・セットを使用不可として記録する	エリア・データ・セットの停止	ローカルに入力されたかのようにコマンドを処理
/START AREA DATABASE GLOBAL	データベースまたはエリアにリカバリーまたはバックアウトの必要がないことを確認し、以降の許可の禁止フラグ、および読み取り専用フラグをリセットし、データベース開始を記録し、およびアクセス・レベルの変更が可能になる	トランザクションにデータベースまたはエリアへのアクセスを許可する	ローカルに入力されたかのようにコマンドを処理

例: ある IMS システムで /START コマンドを GLOBAL キーワードを添えて入力して、いくつかのデータベース名を指定します。IRLM は、ローカル・システムには無効なデータベースの名前を削除してから、他の IMS システムにコマンドを伝送します。これですべての共用 IMS システムがコマンドを処理します。これらのオンライン・データ共用システムでは、DFS3334I メッセージの後に DFS3328I メッセージが表示され、グローバル /START コマンドが開始し、完了したことを知らせます。メッセージには、コマンドに組み込んだデータベース名が含まれます。

GLOBAL キーワードを省略した (または LOCAL キーワードを指定した) 場合、コマンドはローカル・オンライン IMS システムのみに適用され、その他の IMS システムによるアクセスには影響を及ぼしません。

/STOP コマンドまたは UPDATE コマンドをローカルに入力したときに、IMS と DBRC に相互作用は発生しません。

関連資料: CICS 環境でのこれらのコマンドのフォーマットについては、「CICS Transaction Server for z/OS CICS Supplied Transactions」を参照してください。

データ共有システムのモニター

データ共有をモニターするには、次の状況についての情報を入手します。IRLM、IMS システムおよびデータベース、RECON データ・セット、およびキャッシング・ファシリティ構造。

IRLM アクティビティの状況の入手

MODIFY z/OS コマンドを使用して、IRLM の状況を表示します。

該当システムまたは別の接続システム上の IRLM の状況を表示するには、z/OS コマンドを入力します。ここで、*irlmproc* は IRLM を開始するのに使用したプロシージャの名前で、*irlmx* は状況を表示したい IRLM の名前です。

```
MODIFY irlmproc,STATUS,irlmx
```

ALLD キーワードを使用して、データ共有グループ内の IRLM に対して識別された、すべての IMS の名前と状況が表示できます。または、ALLI キーワードを使用して、データ共有グループ内のすべての IRLM の名前と状況を表示できます。

IRLM アクティビティのトレースもできます。「IMS V15 オペレーションおよびオートメーション」の『IRLM アクティビティのトレース』を参照してください。

コンポーネントおよびリソースの表示

データ共有環境内のコンポーネントおよびリソースのモニターには、非共有環境の場合と同じ種類の手順が必要です。

以下の表は、さまざまな IMS リソースについての情報を取得するのに使用できる /DISPLAY コマンドのキーワードをリストにしています。

表 22. IMS リソースに関する情報を提供する /DISPLAY コマンド・キーワード

リソース	/DISPLAY コマンド・キーワード
アクティブ制御領域	ACTIVE REGION
アクティブ・ジョブ	ACTIVE
プログラム、トランザクション、および会話	CONVERSATION PROGRAM PSB STATUS PROGRAM STATUS TRANSACTION SYSID TRANSACTION TRANSACTION
データベース	DATABASE AREA STATUS DATABASE

表 22. IMS リソースに関する情報を提供する /DISPLAY コマンド・キーワード (続き)

リソース	/DISPLAY コマンド・キーワード
端末、回線、リンク、およびノード	ACTIVE DC ASSIGNMENT LINE ASSIGNMENT LINK ASSIGNMENT NODE LINE LINK LTERM MASTER MSNAME NODE PTERM STATUS LINE STATUS LINK STATUS LTERM STATUS MSNAME STATUS NODE STATUS PTERM
外部サブシステムおよび外部サブシステムへの接続	CCTL OASN SUBSYS SUBSYS
VTAM	TIMEOVER

例えば、/DISPLAY DB コマンドを /START DBコマンドの後に使用すると、データベースが開始済みであるかどうかを判別できます。

以下の表は、さまざまな IMS リソースについての情報を取得するのに使用できる QUERY コマンドのキーワードをリストにしています。

表 23. IMS リソースに関する情報を提供する QUERY コマンド・キーワード

リソース	QUERY コマンド・キーワード
アクティブ制御領域	MEMBER
IMSplex のメンバー	IMSPLEX
トランザクション	TRAN
データベース	DB AREA
グローバル・オンライン変更状況	OLC
言語環境プログラム (LE) 実行時オプション	LE
オンライン再編成 (OLR) 状況	OLREORG
RM リソース構造	STRUCTURE

表 23. IMS リソースに関する情報を提供する QUERY コマンド・キーワード (続き)

リソース	QUERY コマンド・キーワード
定義が作成または最後に更新されて以降、IMS RSC リポジトリにエクスポートされていないランタイム・リソースおよび記述子	DB SHOW(EXPORTNEEDED) DBDESC SHOW(EXPORTNEEDED) PGM SHOW(EXPORTNEEDED) PGMDESC SHOW(EXPORTNEEDED) RTC SHOW(EXPORTNEEDED) RTCDESC SHOW(EXPORTNEEDED) TRAN SHOW(EXPORTNEEDED) TRANDESC SHOW(EXPORTNEEDED)

例えば、QUERY DB コマンドを UPDATE DB START(ACCESS) コマンドの後に発行すると、データベースが開始済みであるかどうかを判別できます。

関連資料: /DISPLAY、/START、QUERY、および UPDATE コマンドについての詳細な説明は、「IMS V15 コマンド 第 1 巻: IMS コマンド A-M」および「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

RECON データ・セット内の情報のモニター

RECON データ・セットからの情報を毎日評価して、重要なすべてのデータ共用トランザクションについて、データ共用ワークロードの合計および応答時間の状況を理解しておいてください。

RECON データ・セットに記録されているイベントの履歴のトレースを使用して、共用データがアクティブであったときのインターバルを確認することができます。LIST コマンドまたは QUERY 要求を使用すると、イベントの包括的なリストを得ることができます。シスプレックス・データ共用では、すべてのオペレーティング・システムからのリストが必要です。IMS のサインオンおよびタイム・スタンプが付いたデータベース・データ・セットの OPEN/CLOSE などのイベントをトレースできます。

カップリング・ファシリティ上での構造のモニター

DISPLAY XCF,STRUCTURE および DISPLAY XCF,STRUCTURE,STRNAME= の z/OS オペレーター・コマンドは、特に構造アクティビティのモニターに便利です。

これらのコマンドを使用することで、カップリング・ファシリティ上での構造を表示し、リソース状況を判別し、障害の場合には、問題判別の情報を収集します。

これらのコマンドの詳細については、適切な z/OS 資料を参照してください。

DISPLAY XCF,STRUCTURE コマンド

このコマンドを使用して、アクティブ・ポリシーに定義した構造の状況を表示します。

以下の例は、このコマンドの出力例です。

```

IXC359I 11.09.26 DISPLAY XCF 376
STRNAME      ALLOCATION TIME  STATUS
CF01         02/17/96 17:03:49 ALLOCATED
CF02         --      --      NOT ALLOCATED
CF03         --      --      NOT ALLOCATED
CF04         --      --      NOT ALLOCATED
OSAMSESXI    02/17/96 17:02:54 ALLOCATED
                                     REBUILDING
                                     REBUILD PHASE: QUIESCE
VSAMSESXI    02/17/96 17:03:03 ALLOCATED

```

例えば次のとおりです。

STRNAME

構造の名前。

ALLOCATION TIME

構造がカップリング・ファシリティに割り振られた時間を示すタイム・スタンプ。

STATUS

構造の現行状況。構造は、ALLOCATED、NOT ALLOCATED、または ALLOCATED REBUILDING など、さまざまな状態があり得ます。

表示された情報は、適切な IRLM、OSAM、VSAM、共用 MSGQ、および共用 EMHQ の構造が、定義済みで割り振られているか判別する役に立ちます。そうでなければ、CFNAMES 制御ステートメントにある構造名が、表示されている構造名と一致するか調べてください。一致しない場合は、カップリング・ファシリティ・リソース・マネージャー (CFRM) ポリシーか、または CFNAMES 制御ステートメントのどちらかの名前を変更してください。

例には、6 つの構造が定義されています (CF01 から CF04、OSAMSESXI、および VSAMSESXI)。構造のうち 3 つは割り振られています (CF01、OSAMSESXI、および VSAMSESXI)。OSAM 構造の状況は、構造の障害の後、現在再作成されているところです。

DISPLAY XCF,STRUCTURE,STRNAME= コマンド

このコマンドを使用して、特定の構造についての詳細な情報を表示します。IMS にとって重要な構造は、IRLM、OSAM、VSAM、共用 MSGQ、および共用 EMHQ 構造です。

以下の例は、名前が OSAMSESXI の OSAM 構造を指定した、このコマンドからの出力例を示しています。

```

IXC360I 17.30.46 DISPLAY XCF 677
STRNAME: OSAMSESXI
STATUS: ALLOCATED
POLICY SIZE      : 2048 K
PREFERENCE LIST: CF02      CF01
EXCLUSION LIST IS EMPTY

ACTIVE STRUCTURE
-----
ALLOCATION TIME: 02/17/96 17:02:54
CFNAME         : CF02
COUPLING FACILITY: ND02...
                PARTITION: 0   CPCID: 00
ACTUAL SIZE    : 2048 K

```

```

STORAGE INCREMENT SIZE: 256 K
VERSION          : A8DAC970 15774B04
DISPOSITION     : DELETE
ACCESS TIME     : 0
MAX CONNECTIONS: 32
# CONNECTIONS   : 5

```

CONNECTION NAME	ID	VERSION	SYSNAME	JOBNAME	ASID	STATE
DLI11	04	00040001	MVS1	DLI11	0031	ACTIVE
DLI12	05	00050001	MVS1	DLI12	0033	ACTIVE
IMS1	01	00010013	MVS1	DLIOCSA8	0036	ACTIVE
IMS2	02	00020012	MVS2	DLIOCSB8	0035	ACTIVE
IMS3	03	00030011	MVS3	DLIOCSC8	0038	ACTIVE

出力は 3 部から成ります。最初の部分は、構造の状況とアクティブ・ポリシーについての情報を示しています。次の表示情報は、特に役に立ちます。

STRNAME

構造の名前で、コマンドに指定されたもの。

STATUS

構造の現行状況。構造は、ALLOCATED、NOT ALLOCATED、または ALLOCATED REBUILDING など、さまざまな状態があり得ます。

POLICY SIZE

ポリシーに指定された構造のサイズ。

PREFERENCE LIST

MVS が構造を割り振るカップリング・ファシリティを示します。z/OS はまずリストされた最初のカップリング・ファシリティ上に構造 (例の CF02) を割り振ろうとし、次の構造を (CF01)、というように割り振っていきます。

出力の 2 番目の部分は、より詳細な構造の状況情報を示します。

ALLOCATION TIME

構造が割り振られた時刻。

CFNAME

構造が割り振られていたカップリング・ファシリティの名前。

COUPLING FACILITY

構造が常駐するカップリング・ファシリティの名前。

ACTUAL SIZE

構造の実際のサイズ。

DISPOSITION

障害の状態にある構造にシステムが行う対処方法。IRLM 構造は常に後処理は KEEP です。OSAM および VSAM 構造は、常に後処理は DELETE です。

MAX CONNECTIONS

OSAM、VSAM、IRLM、共用 MSGQ、または共用 EMHQ 構造に接続できる接続の最大数。どの構造に対しても最大接続数 255 です。

CONNECTIONS

構造に接続された IMS システムの現行数。

出力の 2 番目の部分にあるその他の情報は、IMS シスプレックス・データ共有のモニターおよび問題判別に必要ありません。

出力の 3 番目の部分は、どの z/OS システムが構造に接続されているかを示し、それぞれの使用に関する情報を表します。

CONNECTION NAME

接続の名前。z/OS はテーブルの中の接続メッセージ・テキストを繰り返し、すべての接続に報告します。接続がなければ、接続テーブルは表示されません。

ID 接続 ID。

VERSION

接続される z/OS システムのバージョン番号。

SYSNAME

接続される z/OS システムの名前。

JOBNAME

接続に関連したジョブの名前。

ASID

接続に関連したアドレス・スペースの ID。

STATE

構造の状況。これは、次のいずれかです。

FAILED PERSISTENT

構造の後処理が KEEP で、構造への接続が脱落した場合、失敗した持続状況が発生します。

DISCONNECTING

IMS は構造からの切断の処理中です。

FAILING

IMS は異常終了の処理中です。

ACTIVE

IMS は接続されています。

ACTIVE &

IMS は接続されていますが、物理的に構造への接続が脱落しています。

ACTIVE OLD

構造は再作成されています。IMS は古い構造に接続されています。

ACTIVE &OLD

構造は再作成されています。IMS は古い構造に接続されますが、物理的にその構造への接続が脱落しています。

ACTIVE NEW,OLD

構造は再作成されています。IMS は古い構造と新しい構造の両方に接続されます。

ACTIVE NEW,&OLD

構造は再作成されています。IMS は古い構造と新しい構造の両方に接続されますが、物理的に古い構造への接続が脱落しています。

ACTIVE &NEW,OLD

構造は再作成されています。IMS は古い構造と新しい構造の両方に接続されますが、物理的に新しい構造への接続が脱落しています。

ACTIVE &NEW,&OLD

構造は再作成されています。IMS は古い構造と新しい構造の両方に接続されますが、物理的に両方の構造への接続が脱落しています。

DBRC を使用したデータベース割り振りおよびアクセスの制御

DBRC を使用することで、データ共用に含まれる IMS システムによるデータへのアクセスを制御できます。DBRC を使用すると、RECON データ・セット内の現行状況表示を修正、開始、および削除して、オンライン IMS システムのアクセス意図および登録されたデータベースの共用レベルを変更できます。

データ共用環境は、RECON データ・セット内に示されたデータベースおよび IMS システムの状況によって異なります。

/START コマンドまたは UPDATE コマンドの形式を使用して、アクセス意図標識を変更できます。

DBRC オンライン変更コマンドの 1 つ、/RMCHANGE の形式を使用して、共用レベル標識を修正できます。

許可の拒否

(データ共用環境での) DBRC が許可要求に応答するが、アプリケーション・プログラムに許可を獲得できない場合、次のことが発生します。

- データベースを識別するメッセージ DFS047A を受け取る。
- IMS はアプリケーション・プログラムの PSB をデータベース・アクセスなしでスケジュールする。IMS は、データベースにアクセスしようとした場合のみ、異常終了コード U3303 で BMP または MPP を異常終了する。
- IMS は、異常終了コード U0047 でバッチ領域またはユーティリティー領域を異常終了する。

データベース・アクセス意図の変更

以下のコマンドのいずれかを使用して、システム定義中に宣言するアクセス意図を変更します。ここで、*dbx* はデータベース名で、*xx* は新しいアクセス意図です。

- /START DATABASE *dbx* ACCESS=*xx*
- UPDATE DB START(ACCESS) SET(ACCTYPE(BRWS,EXCL,READ,UPD))

/START DATABASE *dbx* ACCESS=*xx* コマンドの場合、*xx* の値は次のとおりです。

- EX 排他使用
- UP 更新アクセス
- RD 読み取りアクセス
- RO 読み取り専用アクセス

このコマンドはローカルで、コマンドを入力した IMS システムだけに影響を及ぼします。GLOBAL キーワードは、ACCESS= キーワードと一緒に使用すると無効で

す。シスプレックスにまたがって共用データベースのアクセス・レベルを変更する必要がある場合、データベースを共用する各 IMS システムごとに、このコマンドを入力する必要があります。

UPDATE DB START(ACCESS) SET(ACCTYPE(BRWS,EXCL,READ,UPD)) コマンドは、ローカルでも、グローバルでもかまいません。

DEDB へのアクセス意図を変更するためには、DEDB 中の任意のエリアにアクセスするすべての PSB を停止する必要があります。DEDB にスケジュールされた入力待ち (WFI) トランザクションを持つ領域も停止する必要があります。

オンライン DBRC コマンド

/RMCHANGE コマンドは、DBRC ユーティリティー機能を制御する IMS コマンドのセットの 1 つで、IMS は対応するコマンドを直接 DBRC に出して、ユーティリティーに応答します。

以下の表は、これらのコマンドの推奨される使用法を示しています。

表 24. DBRC コマンドおよび機能

コマンド	推奨される使用法
/RMCHANGE	データベース共用レベルの変更。 その他のサブシステム許可の防止。 バックアウト要求状況の設定または除去。
/RMDELETE	データベース・レコードおよび関連した状況の除去。 無許可データベースでのサブシステムの削除。 エリア・データ・セット (ADS) レコードおよび関連した状況の除去。
/RMGENJCL	オンライン・リカバリー・ジョブの生成。
/RMINIT	新しいデータベースの登録。 エリア・データ・セットの制御の開始。
/RMLIST	データベースまたは ADS のリカバリー情報および許可情報の入手。
/RMNOTIFY	システム状況のリセット。

登録済みデータベース (名前は ORDERDB) の共用レベルをプロセッサ内ブロック・レベル・データ共用 (共用レベル 2) からプロセッサ間ブロック・レベル・データ共用 (共用レベル 3) に変更して、別のプロセッサにある IMS システムとデータを共用できるようにするには、次を入力します。

```
/RMCHANGE DBRC='DB DBD(ORDERDB) SHARELVL(3)'
```

DBRC は対応するコマンド入力 (CHANGE.DB DBD(ORDERDB) SHARELVL(3)) と一連の DBRC メッセージで応答します。

ジョブがデータベースにアクセスする間に、RECON データ・セットの中のデータベース・レコードの状況を変更する /RMCHANGE、/RMDELETE、またはその他のコマンドを入力しないでください。そのデータベースの RECON レコードを変更する前に、グローバル /DBRECOVERY DB コマンドまたは UPDATE DB STOP(ACCESS) コマンドを使用してデータベースを停止してください。そうしないと、IMS はそのコマンドをエラーで拒否します。

関連資料: これらのコマンドのすべての説明とその機能については、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

データベース可用性の改善

データ共用環境のデータベース可用性の改善には、オンライン変更の制御、データベースの再編成、およびデータベース・イメージ・コピーの作成を含みます。

オンライン変更の制御

共用環境で共用データベースをオンラインで修正する場合は、すべての共用 IMS システム全体で、/MODIFY コマンドのシーケンスを調整する必要があります。

1. そのデータベースを使用する IMS オンライン・システムで、修正するデータベースを停止する。
2. すべての共用 IMS システムでオンライン変更を実行する。
3. /MODIFY コマンド・シーケンスが影響を及ぼす、すべての IMS システムで完了した場合のみ、共用データベースを再始動する。

データ共用 IMS システムのいずれかが、物理的に ACBLIB、FORMAT、または MODBLKS ライブラリーのどれかを共用する場合にも、これらの IMS システムが同じ指定アクティブ・ライブラリー (例: IMS.ACBLIBA) を使用することを確認してください。

各システムは、IMS.MODSTAT データ・セットの中のアクティブ・ライブラリーを絶えずトラッキングしています。1 つの IMS システムだけでオンライン変更を実行すると、個々の IMS システムが異なる制御ブロックを使用することになり、データ共用環境の予期できないエラーの原因になる場合があります。

また、いずれかのライブラリーが共用されている場合は、オンライン変更コピー・ユーティリティー (DFSUOCU0) が非アクティブ・ライブラリーのみを更新するようにしてください。

データベースの再編成

HALDB オンライン再編成 (OLR) を使用している場合を除き、データベースを (データベース再編成ユーティリティーを使用して) 再編成する間は、そのデータベースの使用を許可された IMS システムがデータベースを変更しないように保護する必要があります。

データベースの保護には、次の 3 つの方法があります。

- /DBRECOVERY DB GLOBAL コマンドまたは UPDATE DB STOP(ACCESS) コマンドを出して、すべての IMS システム上のデータベースを閉じる。データベース再編成が完了した後に、各 IMS システムの MTO に、データベースが使用可能であることを知らせます。
- DBRC CHANGE.DB NOAUTH コマンドを出して、データベースがデータ共用に含まれる許可がないということを指定する。再編成の完了後に、CHANGE.DB AUTH コマンドを出してデータ共用を再開します。
- MTO プロシージャに工夫をして、共用 IMS システムでアプリケーション・アクティビティーを調整する。このプロシージャは MTO に指示して、アクテ

ィブでデータベースを使用している可能性のあるアプリケーションを停止させます。再編成の完了後、MTO は適切なオンライン IMS システムの中のアプリケーションを再始動できます。

再編成ユーティリティーは DBRC と通信します。応答で、DBRC は、再編成後でも、データベース・イメージ・コピーまたはリカバリー・グループが作成されるまで、データベースに対してそれ以上の処理を許可しません。

再編成ユーティリティーは、データベースに対する DBRC からの許可の取得に失敗した場合、実行されない場合があります。DBRC は、別の IMS システムにデータベースを更新する許可を与えていた場合は、許可を拒否します (これが、最初 /DBRECOVERY DB、UPDATE DB STOP(ACCESS)、または CHANGE.DB コマンドを出す必要がある理由です)。ユーティリティーの実行に DBRC が必要であることを知らせるメッセージ DFS044I を受け取り、ユーティリティーが異常終了した場合は、IMS システムが終了するのを待つか、またはオンライン IMS の場合は /DBRECOVERY DB コマンドまたは UPDATE DB STOP(ACCESS) コマンドを発行します。

関連資料: データベース・レベルの共用環境でデータベース・リカバリーを扱う操作手順例については、325 ページの『データベースの再編成』を参照してください。

オンライン・データベース・イメージ・コピーの作成

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) を BMP としてスケジュールすると、DBRC は制限付きアクセスの要件を管理します。ただし、イメージ・コピー BMP が稼働している IMS オンライン・システム以外に、IMS オンライン・システム、オンラインまたはバッチが、データベースの更新を許可されていないことを確認する必要があります。

イメージ・コピーをサポートする操作手順例については、326 ページの『共用データベース・イメージ・コピーの作成』を参照してください。

通常オペレーションの維持

データ共用は、通常の操作手順にさらに追加の手順が必要です。次のトピックは、データベース・レベルのデータ共用環境での、日常の調整および維持を行う方法を示すサンプル操作手順を説明します。

推奨事項: 外部時間参照 (ETR) 装置を使用し、これを協定世界時を使用するように設定してください。必要に応じて、それぞれのデータ共用 IMS システムごとに地方時を設定できます。ETR の使用の詳細については、「z/OS MVS 初期設定およびチューニング 解説書」を参照してください。

バッチ更新プログラムのスケジュールリング

次の初期構成および目標構成を想定し、この手順を使用してバッチ更新プログラムをスケジュールしてください。

- 初期構成
 - データベース x (dbx) が共用レベル 1 で登録されています。

- *dbx* は、オンライン IMS システム (IMSA) への更新アクセス権限が許可されています。
- 目標構成
 - *dbx* は、バッチ IMS システム (IMSB) への更新アクセスが許可されます。
 - *dbx* は IMSA の読み取り専用アクセスが許可されます。

次のステップに従って、バッチ更新プログラムをスケジュールしてください。

1. コマンド `CHANGE.DB DBD (dbx) NOAUTH` を使用して、他の IMS システムが *dbx* へのアクセス権限を獲得しないようにする。
2. 以下のいずれかのコマンドを使用して、*dbx* での IMSA のアクティビティを延期し、*dbx* をクローズして、IMSA の許可を解放する。
 - `/DBRECOVERY DB dbx NOFEOV`
 - `UPDATE DB STOP(ACCESS) OPTION(FEOV)`

AREA エリアを DB の代わりに指定して、DEDB エリアに `/DBRECOVERY` または `UPDATE` コマンドを使用することができます。
3. 以下のいずれかのコマンドを使用して、IMSA からの読み取り専用処理を許可する。
 - `/START DATABASE dbx ACCESS=RO`
 - `UPDATE DB NAME dbx START(ACCESS) SET(ACCTYPE(READ))`
4. コマンド `CHANGE.DB DBD (dbx) AUTH` を使用して、IMSB の初期設定を許可する。
5. IMSB を *dbx* への更新アクセスで開始する。

オンライン更新プログラムの復元

次の初期構成および目標構成を想定し、この手順を使用して初期構成を復元してください。

- 初期構成
 - *dbx* はオンライン IMS システム (IMSA) への読み取り専用アクセスを許可されています。
 - バッチ IMS システムは終了しています。
- 目標構成
 - *dbx* は共用レベル 1 で登録されます。
 - *dbx* は IMSA の更新アクセスが許可されます。

次のステップに従って、オンライン更新プログラムを復元します。

1. コマンド `CHANGE.DB DBD (dbx) NOAUTH` を使用して、他の IMS システムが *dbx* へのアクセス権限を獲得しないようにする。IMSA に強制的に *dbx* の許可を解放させると、別の IMS システムがこのデータベースを処理する許可を取得できません。NOAUTH キーワードを指定した `DBRC CHANGE.DB` コマンドは、データベースが処理に使用できないことを (RECON データ・セットのデータベース・レコードに) 示します。
2. IMSB を終了する。

3. 以下のいずれかのコマンドを使用して、*dbx* での IMSA のアクティビティを延期し、*dbx* をクローズして、IMSA の許可を解放する。
 - /DBRECOVERY DB *dbx* NOFEOV
 - UPDATE DB STOP(ACCESS) OPTION(FEOV)
4. 以下のいずれかのコマンドを使用して、IMSA に更新処理を許可する。
 - /START DB *dbx* ACCESS=UP
 - UPDATE DB START(ACCESS)
5. コマンド CHANGE.DB DB *dbx* AUTH を使用して、IMSA 内でのアクセスまたは更新を許可する。

この手順の初期構成については、322 ページの『バッチ更新プログラムのスケジューリング』で説明しています。

更新機能の転送

次の初期構成および目標構成を想定し、この手順を使用して、別の共用 IMS システムに更新機能を転送します。

- 初期構成
 - *dbx* は、オンライン IMS システム (IMSA) への更新アクセス権限が許可されています。
 - *dbx* は、別のオンライン IMS システム (IMSB) への読み取り専用アクセス権限が許可されています。
- 目標構成
 - *dbx* は IMSA の読み取り専用アクセスが許可されます。
 - *dbx* は IMSB の更新アクセスが許可されます。

次のステップに従って、更新の互換性を転送してください。

1. コマンド CHANGE.DB DBD(*dbx*) NOAUTHを使用して、他の IMS システムが *dbx* へのアクセス権限を獲得しないようにする。
2. 以下のいずれかのコマンドを使用してアクティビティを延期し、両 IMS システムで *dbx* をクローズする。
 - /DBRECOVERY DB *dbx* NOFEOV
 - UPDATE DB STOP(ACCESS)
3. 両 IMS システムで処理を許可する。IMSA で次のいずれかのコマンドを出す。
 - /START DB *dbx* ACCESS=RO
 - UPDATE DB START(ACCESS)
4. 次に、IMSB で次のいずれかのコマンドを出す。
 - /START DB *dbx* ACCESS=UP
 - UPDATE DB START(ACCESS)
5. コマンド CHANGE.DB DBD(*dbx*) AUTH を使用して、両 IMS システムでアクセスまたは更新を許可する。IMSA および IMSB には現在、互換性のあるデータベース許可がありますが、IMSA および IMSB が、データベースに依存する PSB でアプリケーション・プログラムをスケジュールするまで、データベースはその他の IMS システムから保護されません。したがって、その他すべての

更新プログラムを締め出すには、IMSB は許可を獲得する最初の IMS システムになり、データベースを更新アクセスで処理する必要があります。

データベースの再編成

次の初期構成および目標構成を想定し、この手順を使用してデータベースを再編成してください。

- 初期構成
 - *dbx* はデータベース・レベル共用で登録されています。
 - *dbx* は、オンライン IMS システム (IMSA) への更新アクセス権限が許可されています。
- 目標構成
 - *dbx* が正常に再編成されます。

次のステップに従って、データベースを再編成します。

1. コマンド `CHANGE.DB DBD(dbx) NOAUTH` を使用して、他の IMS システムがアクセス権限を獲得しないようにする。
2. 再編成ジョブを開始する。

保護された環境で、スキャン、アンロード、再ロード、または接頭部更新などの一連の再編成ジョブを実行する。

これらの再編成ユーティリティーは、その他の IMS システムから干渉を受けずに実行する必要があります。DBRC `CHANGE.DB NOAUTH` コマンドは、オンラインおよびバッチの IMS システムに対して、すべての再編成ユーティリティーを除いて、データベースへの許可を禁止します。

3. コマンド `CHANGE.DB DBD(dbx) AUTH` を使用して、オンライン・データベース・イメージ・コピー・ユーティリティー (DFSUICP0) の初期設定を許可する。このコマンドによって、その他のバッチまたはオンライン IMS システムは *dbx* へのアクセスを獲得できます。

関連資料: データベースを再編成する方法の説明については、321 ページの『データベースの再編成』を参照してください。

イメージ・コピーの排他制御での作成

次の初期構成および目標構成を前提として、このシナリオを使用してデータベース・データ・セットのイメージ・コピーを排他制御で作成してください。

- 初期構成
 - *dbx* はデータベース・レベル共用で登録されています。
 - *dbx* は、オンライン IMS システム (IMSA) への更新アクセス権限が許可されています。
 - *dbx* は、別のオンライン IMS システム (IMSB) への読み取り専用アクセス権限が許可されています。
- 目標構成
 - データベース *dbx* のイメージ・コピー (データ・セット *y*)。

次のステップに従って、イメージ・コピーを排他制御で作成してください。

1. 以下のコマンドを使用して、他の IMS システムがアクセスを獲得するのを防止する。CHANGE.DBDS DBD(*dbx*) DDN(*y*) ICON

CHANGE.DBDS コマンドを ICON キーワードを指定して使用し、イメージ・コピーが必要であることを (RECON データ・セットに) 示します。この指示は、イメージ・コピー・ユーティリティーを除くあらゆる IMS システムが、データベースへアクセスを獲得することを防止します。

2. 以下のいずれかのコマンドを使用して、*dbx* の IMS システム・アクティビティを延期し、すべての共用 IMS システムで *dbx* をクローズする。
 - /DBRECOVERY DB *dbx* NOFEOV
 - UPDATE DB STOP(ACCESS) OPTION(FEOV)
3. 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) を実行する。

このユーティリティーは、正常に完了すると、イメージ・コピーが必要であることを示す RECON データ・セット内の指示を除去して、DBRC が IMS システムにデータベースのアクセスを許可できるようにします。

4. 以下のいずれかのコマンドを使用して、*dbx* で IMS システム・アクティビティを再開する。
 - /START DB *dbx*
 - UPDATE DB START(ACCESS)

共用データベース・イメージ・コピーの作成

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、常に読み取りアクセスを要求するので、読み取り専用アクセスを要求する IMS システムとだけ、データベースを共用できます。イメージ・コピーを取得している間に、その他の IMS システムがデータベースを更新する必要がある場合、並行イメージ・コピー・オプションを使用します。

次の初期構成および目標構成を想定し、この手順を使用してデータベース・レベルの共用環境でイメージ・コピーを作成してください。

- 初期構成
 - *dbx* はデータベース・レベル共用で登録されています。
 - *dbx* は、オンライン IMS システム (IMSA) への更新アクセス権限が許可されています。
 - *dbx* は、バッチ IMS システム (IMSB) への読み取り専用アクセス権限が許可されています。
- 目標構成
 - IMSA でのオンライン・データベース・イメージ・コピー・ユーティリティー (DFSUICP0) は、読み取りアクセスを許可されます。
 - *dbx* は IMSA の読み取りアクセスが許可されます。
 - *dbx* は IMSB の読み取り専用アクセスが許可されます。

次のステップに従って、イメージ・コピーを作成してください。

1. 以下のコマンドを使用して、すべての IMS システムが更新アクセス権限を獲得しないようにする。CHANGE.DB DBD(*dbx*) READON

このコマンドは、IMS システムが更新意図でデータベースをアクセスできないことを (RECON データ・セット内に) 示します。

2. 以下のいずれかのコマンドを使用して、*dbx* の IMSA 更新アクティビティを延期し、よりレベルの低い読み取りアクセスを許可する。
 - /DBDUMP DB *dbx* NOFEOV
 - UPDATE DB STOP(UPDATES) OPTION(FEOV)
3. 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) を実行する。
4. 以下のいずれかのコマンドを使用して、IMS A で更新アクティビティを再開する。
 - CHANGE.DB DBD(*dbx*) READOFF
/START DATABASE *dbx* ACCESS=UP
 - CHANGE.DB DBD(*dbx*) READOFF
UPDATE DB NAME(*dbx*) START(ACCESS) SET(ACCTYPE(UPD))

READOFF キーワードを指定した CHANGE.DB コマンドは、IMS システムが更新意図でデータベースをアクセスできることを (RECON データ・セット内に) 示します。

データベースの共用レベルを下げる

次の初期構成および目標構成を前提として、このシナリオを使用してデータベースの共用レベルを下げてください。

- 初期構成
 - *dbx* は、オンライン IMS システム (IMSA) への更新アクセス権限が許可されています。
 - *dbx* は、別のオンライン IMS システム (IMSB) への読み取り専用アクセス権限が許可されています。
- 目標構成
 - *dbx* は IMSA の排他制御が許可されています。

次のステップに従って、共用レベルを下げます。

1. 以下のコマンドを使用して、すべての IMS システムがアクセス権限を獲得しないようにする。CHANGE.DB DBD(*dbx*) NOAUTH
2. 以下のコマンドのいずれかを使用して、両方の IMS システム内で *dbx* に対するアクティビティを中断し、*dbx* を閉じる。
 - /DBRECOVERY DB *dbx* NOFEOV
 - UPDATE DB STOP(ACCESS) OPTION(FEOV)
3. 以下のコマンドを使用して、IMS A 内の処理を許可する。

```
CHANGE.DB DBD(dbx) AUTH  
/START DATABASE dbx ACCESS=EX  
/START PGM  
/START TRAN
```

IMSA が、データベースに依存する PSB でアプリケーション・プログラムをスケジューリングするまで、データベースはその他の IMS システムから保護されません。したがって、その他すべての IMS システムを締め出すには、IMS A はデータベースを排他的アクセス権限で処理する許可を獲得する最初の IMS システムにならなければなりません。

IMSA はターゲット・データベースが許可されない場合のみ、アクセス意図の変更を受け入れます。

IMSA のアクセス意図を変更する代わりに、RECON データ・セットの中のデータベースの共用レベルを修正することによっても、同じ結果を得ることができます。ACCESS= キーワードを指定した /START DATABASE コマンドを省略し、ステップ 3 の前に以下のステップを実行します。

- DBRC に排他制御を強制することを要求する。

```
CHANGE.DB DBD(dbx) SHARELVL(0)
```

この場合は、DBRC は要求されたレベルでデータベース許可を授与し、IMSA がデータベースを排他的に所有します。

障害からのリカバリー

以下のトピックは、データ共用環境における障害からのリカバリーの計画に役立つ情報を提供します。

システム・ログの管理

共用データベースのリカバリーには、各共用 IMS システムからのログ・データが必要です。DBRC を使用した DBRC GENJCL.RECOV コマンドまたはデータベース・リカバリー・ユーティリティの実行は、ログがデータベースのためにマージされる必要がある場合は失敗します。

IMS データベース変更累積ユーティリティ (DFSUCUM0) のマージ機能を使用して、すべての共用 IMS システムからのログ・データをマージできます。

最初のログ・ボリュームの完全セットの確立を除いて、作業負荷の必要な頻度で、データベース変更累積ユーティリティ (DFSUCUM0) を実行できます。これを実行することにより、ログ入力を妥当なボリューム数に抑え、データベースをオフラインにせず、または IMS をダウンさせずに、変更レコードを適宜、累積できます。しかし、ユーティリティを実行することで、変更累積データ・セットのサイズを大きくできます。

IMS は、変更累積データ・セット上のログ・ボリュームのこれら一部のサブセットからのレコードを『予備レコード』として保管します。これらのレコードは、個々の更新特性を保存します。

データベース・リカバリー・ユーティリティ (DFSURDB0) を実行する前に、リカバリーに必要な DBDS にログ・ボリュームの完全セットを指定して、データベース変更累積ユーティリティを実行する必要があります。データベース変更累積ユーティリティをログの不完全なセットで実行した場合、変更累積出力データ・セットを次のデータベース変更累積ユーティリティの入力として指定してください

い。変更累積データ・セットがリカバリーが必要な DBDS に対して不完全なログ・セットを表している場合、そのデータ・セットは DBDS のリカバリーには無効です。

DBRC LIST.CAGRP コマンドは、変更累積グループの DBDS ごとのログのセットが完全かまたは不完全かを表示します。

変更累積データ・セットのサイズを最小化するために、いつでも有効なログ・ボリュームのセットで DFSUCUM0 を実行できます。データ共用複合システムの中にログの有効なセットを確立するには、以下のように行います。

1. すべてのバッチ IMS システムのオペレーションを延期する。
2. /DBRECOVERY DB コマンドまたは UPDATE DB STOP(ACCESS) コマンドを使用し、1 つのオンライン・サブシステムを除いたすべてのオンライン IMS システムに対してデータベース・アクセスを終了する。
3. 残りのオンライン IMS システム上で強制的にログの切り替えを行う。

リカバリー手順の計画

データ共用環境でのリカバリーは、標準の IMS リカバリーと同様です。

標準の IMS リカバリーとデータ共用環境でのリカバリーの両方に、以下の基本作業が含まれます。

- リカバリーのメカニズムの設定 (ロギング、チェックポイントの取得、レコードの保持)。
- リカバリーを必要とする状態の場合に従う操作手順の設定

非共用環境内のオンライン IMS システムのロギングおよびチェックポイントのメカニズムは、データ共用環境内でもアクティブです。これらには、以下のものが含まれます。

- システム・ログ・データ・セットと WADS および再始動データ・セットの使用
- アプリケーション・プログラム、システム、およびメッセージ・キュー・チェックポイント
- データベース・イメージ・コピーの作成

非共用環境とデータ共用環境との基本的な違いは、DBRC への依存の程度の差です。DBRC はデータ共用環境の制御を援助し、単にレコードを保持するだけではありません。

単一 IMS システムの場合のように、データ共用環境内の障害は、再始動はもちろん、オフライン (ユーティリティー・タイプの) リカバリーが必要な場合があります。ただし、データ共用環境内での障害後、障害が起こったシステムだけでなく、その他の共用システムでもリカバリー・アクションを考慮し、実行する必要があります。さらに、環境内の各 IRLM、DBRC、およびカップリング・ファシリティ構造は、別の潜在的な障害点、また、リカバリーおよび再始動手順に含める必要のある別のコンポーネントがあります。

メッセージ・キュー・データ・セット、(非共用キュー環境内の) ログ・データ・セット、およびシステム・データ・セットのリカバリーは、データ共用環境でのリカバリーと違いがありません。

- ログをリカバリーするには、ログ・リカバリー・ユーティリティ (DFSULTR0) を使用します。
- メッセージ・キュー・データ・セットをリカバリーするには、IMS 再始動で、IMS がキューを再作成することを要求します。キューの内容 (SNAPQ、DUMPQ、または PURGE) を記録した最後のシステム・チェックポイント時点からのシステム・ログ・データ・セットを提供する必要があります。
- メッセージ・キュー・データ・セット以外のシステム・データ・セットをリカバリーさせるには、最新のイメージ・コピーを更新します。データ共用を続けるために、RECON データ・セットをリカバリーする必要があります。

リカバリー機能

動的バックアウト、バッチ・バックアウト、および順方向リカバリーの機能は、データ共用環境と非共用環境で異なります。

動的バックアウト

非共用環境の場合のように、データ共用環境のオンライン IMS システムは、アプリケーション・プログラムが、失敗するか、またはロールバック呼び出しでバックアウトを要求すると、動的にアプリケーション・プログラムのコミットされていないデータベース変更をバックアウトし、コミットされていない出力メッセージを廃棄します。ただし、データ共用環境では、IRLM がロックし、RECON 状況表示が共用 IMS システムからのコミットされていない変更を保護することで、保全性を保証します。

アプリケーション・プログラムが失敗した後でも、システムおよびその他のアプリケーション・プログラムのオペレーションは妨害されずに続きます。

アプリケーション・プログラムが、ブロック・レベル・データ共用環境のバッチ更新プログラムの場合、IMS は次の状況のいずれでも動的に変更をバックアウトします。

- アプリケーション・プログラムがロールバック呼び出しを出す場合 (ROLL、ROLB、または ROLS)。
- バッチ領域が DASD に割り振られたシステム・ロギングを使用し、ユーザーが実行パラメーターに BKO=Y を指定し、アプリケーション・プログラムが ROLB または ROLS 呼び出しを出すか、DL/I が検出するエラーが発生する場合。
- ある種のリソース不足および複数のバッチ更新アプリケーション・プログラムを巻き込むデッドロックが発生する場合、IMS は 1 つ以上のバッチ・アプリケーション・プログラムに動的バックアウトを実行します。巻き込まれた領域が DASD ロギングを使用し、BKO=Y が指定されていれば、すべての実行が続きます。

これらの状況では、ユーザーがリカバリー・アクションを行う必要はありません。

オンラインまたはバッチの IMS システムどちらの場合でも、データベースがデータベース・レベルで共用されていると、データベースが動的にバックアウトされる前に、その他のアプリケーション・プログラムはコミットされていないデータを読み込む場合があります。バックアウトは、データベース内のアプリケーション・プログラムのデータの更新にのみ影響を及ぼし、このコミットされていないデータを

使用するその他のどのアプリケーション・プログラムの出力も影響を受けません。これは、データ共用環境内のデータベースのバックアウトの本来の性質というよりはむしろ、PROCOPT=GO 処理本来の性質です。

データベースの動的バックアウトが失敗した場合、バッチ・バックアウトを使用してデータベースをバックアウトする必要があります。次に、/START DB コマンドまたは UPDATE DB START(ACCESS) コマンドを使用して、データベースを再始動しなければなりません。

バッチ・バックアウト

失敗したアプリケーション・プログラムがブロック・レベル・データ共用に含まれるバッチ更新プログラムである場合、IMS はコミットされていないデータベース変更を常に動的にバックアウトするとはかぎりません。

非共用環境の場合と同じように、以下の状況があった場合は、データベースをリカバリーする必要があります。

- バッチ領域が DASD ロギングを使用していない。
- 実行パラメーターに BKO=Y を指定していない。
- 障害が DL/I が検出したエラーでない。

ただし、ブロック・レベル・データ共用環境では、すべてのバッチ更新アプリケーション・プログラムのデータベース変更をバッチ・バックアウト・ユーティリティー (DFSBB00) を使用してバックアウトする必要があります。

ブロック・レベル共用の間に、バッチ更新ジョブだけを含むアプリケーション・プログラム・デッドロックが発生した場合、IMS はこれらのジョブのどれか 1 つを異常終了コード U0777 で異常終了します。IMS が動的に変更をバックアウトできる場合は、その他のアプリケーション・プログラムは処理を継続します。バックアウトできない場合、IMS はその他のアプリケーション・プログラムを異常終了コード U3303 で異常終了させます。

動的バックアウトを行わなかったすべてのバッチ・ジョブを、そのジョブごとにバッチ・バックアウト・ユーティリティーを実行して、リカバリーする必要があります。これらのバッチ・ジョブが保持するロックは、その他のバッチ・ジョブまたはオンライン IMS システムのアクセスに影響を与える可能性があるため、すぐにバッチ・バックアウトを開始する必要があります。

制約事項: ブロック・レベルのデータ共用に含まれるバッチ IMS システムのリカバリーの場合、次のようになります。

- バッチ・バックアウト・ユーティリティーには、実行に IRLM が必要です。
- バッチ・バックアウト・ユーティリティーは、最後に正常終了したチェックポイントにバックアウトします。

順方向リカバリー

データ共用環境の中のデータベースのリカバリーは、非共用環境のリカバリーと似ています。どちらの環境でも、データベース・リカバリー・ユーティリティー (DFSURDB0) を、データベースの最新のイメージ・コピーとそのイメージ・コピーを作成してからの適切なログ・データ・セットと一緒に使用します。

ただし、ブロック・レベルのデータ共用環境では、さらに追加のステップが必要です。複数の IMS システムがデータベースを更新した可能性があるため、データベースを再構成するには、各 IMS システムからのログが必要です。さらに、これらの更新は同時に行われている場合が多いので、これらのログを順次にデータベース・リカバリー・ユーティリティに実行依頼できません。代わりに、まず、データベース変更累積ユーティリティ (DFSUCUM0) を実行してログをマージする必要があります。すべての関係するログを処理しているか確認してください。

変更累積を定期的に行っても、最新のログを以前のログとマージするよう、ユーティリティを実行してから、データベースをリカバリーする必要があります。

ただし、データベース変更累積ユーティリティ (DFSUCUM0) に、有効なログのセットを確立する必要はありません。バッチ IMS システムのオペレーションの延期も、1 つのオンライン IMS システムを除くすべてのデータベース・アクセスの終了も、または残りのオンライン IMS システムでのログの強制切り替えもせずに、いつでも DFSUCUM0 を実行できます。328 ページの『システム・ログの管理』を参照してください。

DBRC なしのリカバリー

DBRC が稼働していないときに、データベース・イメージ・コピーの作成などのリカバリー関連のアクションを行う場合、DBRC は状況の変更を認識していないために、問題が発生する場合があります。したがって、特に DBRC にこのような変更を通知する必要があります。

DBRC は変更の通知のために、複数のコマンドを用意しています。320 ページの『オンライン DBRC コマンド』を参照してください。

再始動が必要な場合があるので、例えば、アクティブである DBRC を使用しないユーティリティ制御機能 (UCF) を使用する必要があります。しかし、どのリカバリー関連の作業に UCF を使用する場合でも、後で特に、DBRC に状況の変更を通知する必要があります。

IMS の障害後の再始動

データ共用環境での IMS システムの再始動方法は、非共用環境の場合と同様です。

- IMS システムを正常にシャットダウンできていた場合は、IMS を /NRESTART コマンドを使用して正常に再始動する。
- IMS システムに障害があった場合は、/ERESTART コマンドを使用して、IMS を緊急時再始動を実行する。

ただし、データ共用環境では、関連した IRLM も停止または障害が起こった場合は、IRLM を再始動してから、ブロック・レベルのデータ共用環境で IMS を始動する必要があることを考慮してください。310 ページの『IRLM の開始』を参照してください。

DBRC の障害後の再始動

DBRC は IMS の制御下で稼働しているため、IMS が異常終了した場合、IMS は DBRC バッファの解放と、システム・ログのクローズを試行します。

DBRC 問題を訂正した後に、/ERESTART コマンドを使用して IMS を再始動してください。

IRLM を伴うリカバリー

データ共用構成の一部にある障害は、通常、その他の部分に障害を起こす原因にはなりません。

以下に例を示します。

- ハードウェア障害または z/OS オペレーティング・システムの異常終了のため、1 つの z/OS システムに障害が起こった場合でも、その他のシステム (IRLM とその他の IMS システム) は実行を継続します。この場合は、残りの稼働している IRLM は、z/OS によって障害を通知され、障害が起こったメンバーが所有するロックに、必要な保持ロック保護を提供します。
- IRLM に障害が起こった場合、すべてのアクティブ・アプリケーション・プログラムが U3303 異常終了コードを受け取り、バックアウトが完了できるように IMS がトランザクションのスケジューリングを延期する間、関連した IMS オンライン・システムは処理を継続します。バッチ IMS システムは、データベースの健全性を保護して終了します。残りの稼働している IRLM は、z/OS によって障害を通知され、次に、障害が起こった IRLM 上の IMS が所有するロックに、必要な保持ロック保護を提供します。

IMS DBCTL 環境で IRLM に障害が起こった場合には、未確定スレッドは次の DL/I 呼び出しで U3303 異常終了コードを受け取ります。未確定スレッドは、同期点処理のフェーズ 2 までアクティブのままです。フェーズ 2 コミットまたは打ち切りの処理後、スレッドは終了します。IMS DBCTL システムは、すべての未確定スレッドが終了した後に、IRLM への再接続を試行します。

- IMS システムに障害が起こった場合は、関連した IRLM およびその他の IMS システムは、障害が起こった IMS が所有するロックで、実行を続けます。このロックは、その IRLM が作成した保持ロックによって保護されています。
- DBRC に障害が起こった場合は、その IMS システムは異常終了しますが、関連した IRLM およびその他の IMS システムとその IRLM は処理を継続します。

オンライン IMS システムでのトランザクション・リカバリー

IMS がトランザクションを自動的にバックアウトするのは、IMS オンライン・システムが、直接または間接的にデータ共用に関係する問題のために、あるいはリソースが使用不可であるために、トランザクション処理を進めることができないと判断した場合、および TRANSACT システム定義マクロの中に SERIAL=NO を指定した場合 (またはデフォルトを受け入れた場合) です。

TRANSACT システム定義マクロの中に SERIAL=YES を指定した場合、IMS はトランザクションを先入れ先出し法で処理するトランザクション・キューの前に戻し、トランザクションを停止します (USTOP 状態)。

トランザクションが DL/I INIT STATUS GROUPA 呼び出しを使用しないのであれば、次の場合に、IMS はトランザクションを延期キューまたはトランザクション・キューに置きます。

- トランザクションが、失敗した IRLM に識別される IMS システムが保持するロックを要求した場合。
- トランザクションが、障害が起こった IMS システムが保持するロックを要求した場合。
- トランザクションが、IMS システムに許可されているレベルよりも高いレベルでデータベースにアクセスしようとしている場合。
- トランザクションが、停止またはロックされているデータベースを要求している場合。
- MTO は /DBRECOVERY DB コマンドまたは UPDATE DB STOP(ACCESS) コマンドを入力します。

IMS は、トランザクションが延期されていることを示すメッセージ DFS3324I を、延期しているそれぞれのトランザクションごとに送ります。

IMS はコールド・スタートを除くあらゆる再始動について延期キューを保管し、延期キュー上のすべてのトランザクションは再処理の対象となります。IMS がトランザクションを再処理すると、IMS はトランザクションをメッセージ・キューにリリースし、通常の方法でスケジュールします。

以下のいくつかの条件により、IMS は上記のトランザクションの再処理の試行を促されます。

- XRF テークオーバー後を含む、オンライン IMS システムの再始動。
- 再始動を実行する別の IMS システムからの通信、またはバックアウトを完了したバッチ IMS システムからの通信。
- IMS オンライン・システムの IRLM への再接続。
- MTO コマンド /DEQUEUE SUSPEND。
- データベースを開始するための MTO コマンド /START DB または UPDATE DB START(ACCESS)。

IMS は延期したトランザクションを再処理しようとするときに、メッセージ DFS3336I を出します。

単独で障害を起こした IRLM での再始動

データ共用構成で IRLM だけが障害を起こした場合は、関連付けられた IMS オンライン・システムがすべてのトランザクションのスケジュールリングを延期し、IMS が影響を受けるトランザクションのバックアウトを実行し、TRANSACT マクロに SERIAL=NO が指定されたか、または SERIAL=YES が指定されたかによって、IMS がトランザクションを延期キューまたはトランザクション・キューに置き、IMS が共用データベースを停止します。

IMS は、障害のために自動的に停止されているデータベースごとに、DFS2012I メッセージを出します。IMS は、IRLM と関連したバッチ IMS システムを異常終了

させます。バッチ・データベース・バックアウトが必要になるのは、データベース更新を実行し、動的バックアウトを使用しないこれらのバッチの実行だけに対してのみです。

障害の起こった IRLM を再始動した後に、RECONNECT キーワードを指定した z/OS MODIFY コマンドを IMS オンライン・システムごとに入力します。このコマンドは、データ共用環境を再確立し、延期キューに対する再処理を開始します。

関連資料: トランザクション・リカバリーに関する詳細については、333 ページの『オンライン IMS システムでのトランザクション・リカバリー』を参照してください。

シスプレックス・データ共用環境での再始動

シスプレックス・データ共用環境では、IRLM に z/OS の障害、IMS の障害、別の IRLM の障害、または物理的なロックング・リソースへの接続の消失が通知されます。

障害の影響を受けた IRLM は、データ共用グループから切断されます。データ共用グループ内の残りの IRLM は、まだ稼働している IRLM として参照されます。

IRLM は何らかの理由で強制的にデータ共用グループから外されると、その IRLM は、1 つの IRLM がデータ共用グループから切断されたことを示す DXR136I メッセージを出します。

まだ稼働している IRLM は、グループの状況が変わったこと、および 1 つの IRLM がデータ共用グループから切断されたことを示す DXR137I メッセージを出します。

オペレーターは、どちらのメッセージにもアクションを取る必要はありません。シスプレックス以外のデータ共用環境の場合のように、IRLM は、その他の IMS システムがアクセスしないように、ロックされたリソースを保護します。

通常再始動の手順は、IRLM およびこれに関連した IMS の再始動に使用できません。

シスプレックス環境でのデータ共用

ほとんどの部分について、非シスプレックスのデータ共用の概念は、シスプレックスのデータ共用にも適用されます。例外がありますが、それについては以下のトピックで説明しています。

以下のトピックでは、読者が z/OS シスプレックスの概念と用語に精通していることを前提としています。

シスプレックス・データ共用の概念と用語

複数の IMS システムが 2 つより多い z/OS イメージにまたがってデータを共用する場合、これをシスプレックス・データ共用 と呼びます。

シスプレックス・データ共用に関与している個々の z/OS イメージは、最低 1 つの IRLM、および対応する高速 DB リカバリー (FDBR) 領域を持っている必要があ

ります。各 IRLM は、カップリング・ファシリティに接続されています。カップリング・ファシリティは、338 ページの『カップリング・ファシリティ』で説明するように、データを共有する複数の IMS システムにまたがってデータ保全性を保守するために使用されます。

バッファ無効化

データベース・ブロックまたは CI が 1 つの IMS によって更新される場合は、IMS は他の IMS バッファ・プール内のこのブロックまたは CI のすべてのコピーに無効のマークが付いていることを確認しなければなりません。

これを行うために、更新 IMS が無効化呼び出しを発行します。この呼び出しは、バッファの内容が更新されたことと、バッファの他のすべてのコピーを無効化する必要があることをカップリング・ファシリティに通知します。すると、カップリング・ファシリティは、バッファに要求権を登録した個々の IMS に対してバッファを無効化します。このプロセスは、バッファ無効化と呼ばれます。IMS はバッファへのアクセスを試行する場合必ず、そのバッファがまだ有効であることを確認するためのチェックを行います。バッファが無効な場合、IMS は DASD からデータを再読み取りします。それによってデータ保全性が保持されます。

バッファ無効化は、すべての IMSplex データベース環境、つまり、DB/DC、DBCTL、DB バッチで機能します。シスプレックス環境では、IMS は VSAM、VSAM ハイパースペース、OSAM、および OSAM 順次バッファリング・バッファ一用のバッファ・プールをサポートします。

データ共有グループ

非シスプレックス・データ共有の場合のように、データ共有グループの概念が適用されます。以下の図は、データ共有グループの例を示しています。

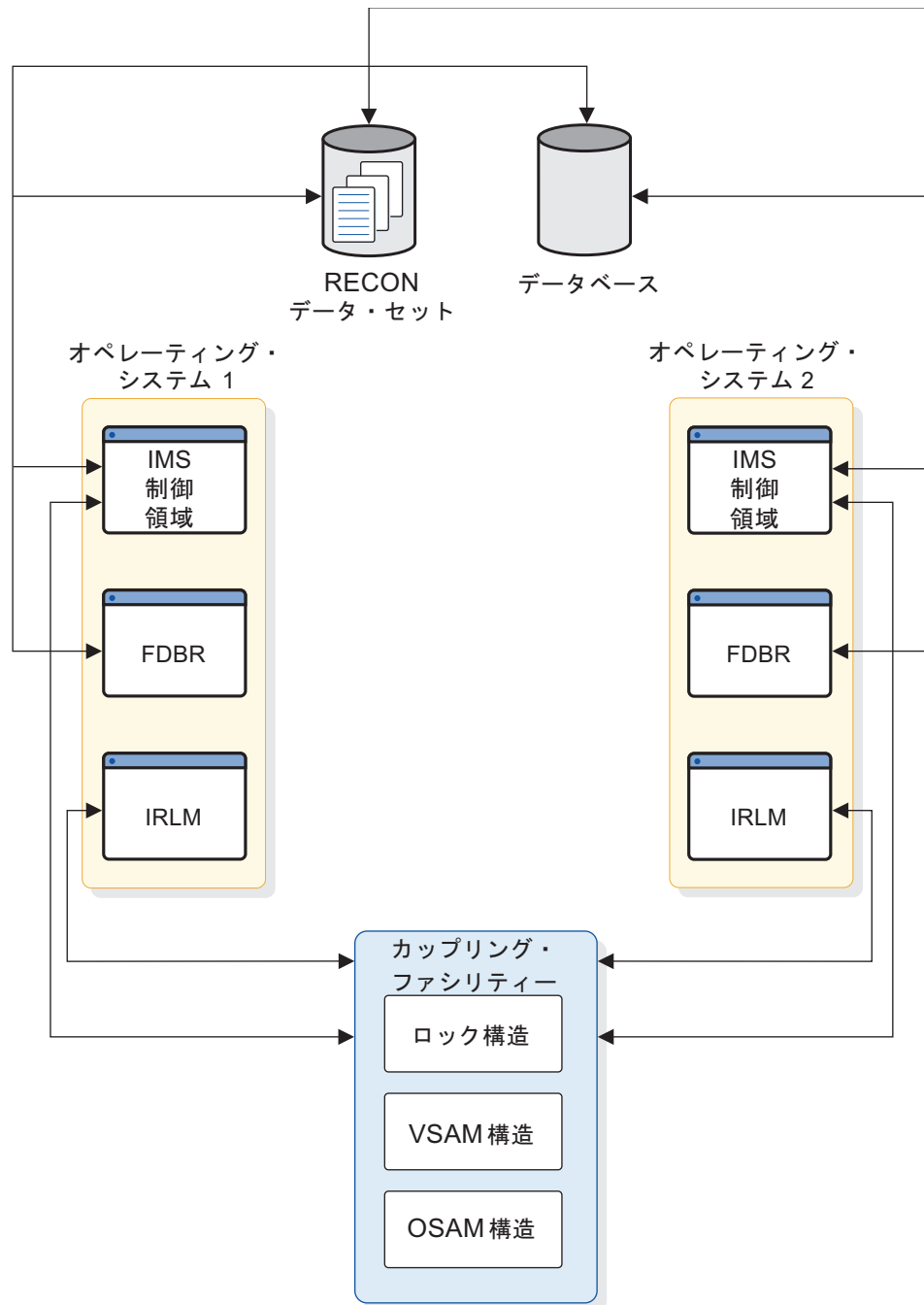


図 36. データ共有グループの例

データ共有グループは、2つのオペレーティング・システムを持っています。それぞれのオペレーティング・システムは、IMS 制御領域、FDBR、および IRLM を含んでいます。各オペレーティング・システムの IMS システムは以下のリソースを共有します。

- データベース
- RECON 二重コピー・データ・セット
- 1つ以上のカップリング・ファシリティ

- カップリング・ファシリティの単一 IRLM ロック・テーブル構造 (ロック構造)
- OSAM および VSAM バッファは、カップリング・ファシリティの構造 (以後、OSAM 構造または VSAM 構造と呼ぶ) を無効化します。

高速データベース・リカバリ領域は共用されません。

データ共有グループでの通信は、カップリング・ファシリティと接続されている IRLM によって行われます。カップリング・ファシリティなしには、有効なシスプレックス・データ共有環境は存在しません。IRLM は、非シスプレックス・データ共有の場合でも、グローバル・ロックを許可しません。IRLM は、シスプレックス・データ共有のためにカップリング・ファシリティを必要とします。カップリング・ファシリティが使用可能でない場合、IRLM は非シスプレックス・データ共有のみを許可します。

IMS は、カップリング・ファシリティの構造と接続します。シスプレックス・データ共有は、バッファ無効化のために、OSAM 構造および VSAM 構造を使用します。これは、各 IMS へのブロードキャスト (通知) を使用してバッファ無効化を行う非シスプレックス・データ共有とは異なります。シスプレックス・データ共有は、カップリング・ファシリティのロック構造を使用して、データ共有環境を確立して、制御します。

データ共有グループは、IMS PROCLIB データ・セットの DFSVSMxx メンバー内の CFNAMES 制御ステートメントを使用して定義されます。ロック、OSAM、および VSAM の各構造は、CFNAMES 制御ステートメントで指定されます。

カップリング・ファシリティ

337 ページの図 36 は、カップリング・ファシリティと、シスプレックス・データ共有に使用されるカップリング・ファシリティ内の 3 つの構造を示しています。

この図は単一のカップリング・ファシリティを示していますが、複数のカップリング・ファシリティを使用することもできます。バックアップ用としてさらにカップリング・ファシリティを定義することができます。あるいは、スループットを増大させるために、構造をいくつかのカップリング・ファシリティにわたって分割することができます。例えば、ロック構造をあるカップリング・ファシリティに置いて、OSAM 構造および VSAM 構造を別のカップリング・ファシリティに置くことができます。


OSAM 構造と VSAM 構造は、バッファ無効化に使用されます。カップリング・ファシリティに接続されている IMS によって読み取られた共有データのそれぞれのブロックごとに、OSAM または VSAM 構造内に項目が作成されます。個々の項目は、バッファ ID (z/OS にはリソース名として認識されている) 用のフィールドと 32 のスロットから構成されます。スロットは、IMS システムが項目のバッファに要求権を登録するためのものです。これによって、32 までの IMS システムがデータを共有できます。シスプレックス・データ共有の眼界である 32 の IMS システムというのは、構造に接続できる IMS システムの数であり、作動している IMS システムの数ではないことに注意してください。また、カップリング・ファシリティに対して実行できる総接続数 (255) でもありません。


ロック構造は、データ共用環境の確立に使用されます。データ共用グループでは、ロック構造に接続する最初の IMS システムが、その同じロック構造に後で接続する IMS システムのデータ共用環境を決定します。IRLM に通知する際に、IMS は CFNAMES 制御ステートメントに指定されているカップリング・ファシリティの構造名と、RECON ヘッダーの DBRC RECON 初期設定タイム・スタンプ (RIT) を渡します。通知操作は、IMS システムが同一の構造名と最初の IMS としての RIT を指定しないと失敗します。

構造に障害が起きると (または構成の手動での再作成を開始すると)、IMS は構造を再作成しようとします。IMS が再作成に失敗するか構造への接続が失われると、IMS はデータ共用トランザクションを静止して、データ共用を停止します。再作成が完了した後、IMS は構造に再接続しようとします。再接続が成功すると、IMS はデータ共用トランザクションのプロセスを続行します。IMS が再接続に失敗すると、データ共用は停止したままとなり、カップリング・ファシリティ・リソースが使用可能であることが再び通知されるまで、IMS は再接続を待機します。

注: IMS が構造再作成を試みたときにバッチ・ジョブがその構造へ接続された場合、IMS はそのバッチ・ジョブを終了し、異常終了 U3303 を発行します。

関連資料:

 zSeries Processor Resource/Systems Manager Planning Guide

 z/OS: カップリング・ファシリティの定義

XRF およびシスプレックス・データ共用

XRF は、代替システムの始動時にカップリング・ファシリティが使用可能で接続されている場合のみ、シスプレックス・データ共用環境で使用できます。

制約事項: XRF サブシステムに対し高速 DB リカバリーを使用することはできません。

シスプレックス・データ共用構成の例

以下の図は、シスプレックス・データ共用に使用できる構成を示しています。

DBRC RECON データ・セットおよび IMS データベースはこれらの図には示されていませんが、337 ページの図 36 に示されているように、RECON とデータベースは、シスプレックス・データ共用グループの IMS システムによって共用されます。

1 つのカップリング・ファシリティの 3 つの構造

次の構成は、IRLM を使用したシスプレックス・データ共用環境で稼働する複数の z/OS イメージを示しています。IMS は、それぞれの z/OS イメージで稼働しています。1 つのカップリング・ファシリティが使用されています。ロック、OSAM、および VSAM の各構造は、同一のカップリング・ファシリティに存在します。

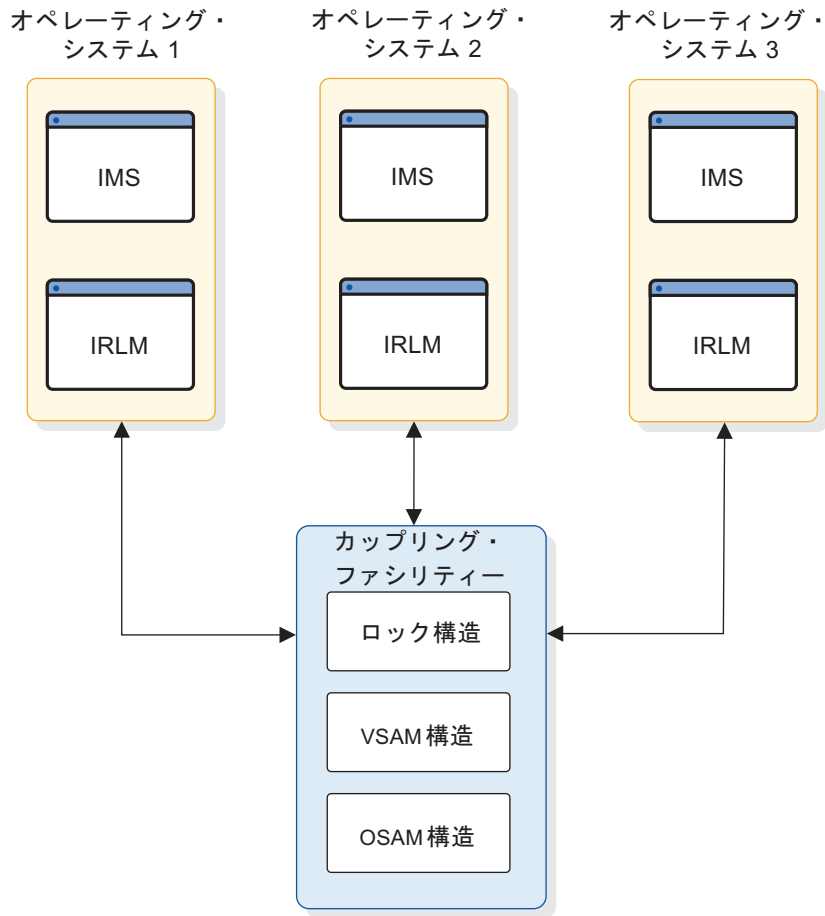


図 37. 1 つのカップリング・ファシリティ上に 3 つの構造を持つシスプレックス・データ共有構成

2 つのカップリング・ファシリティの 3 つの構造

次の構成は、IRLM を使用した IMSplex で稼働する複数の z/OS イメージを示しています。IMS は、それぞれの z/OS イメージで稼働しています。2 つのカップリング・ファシリティが使用されています。ロック構造はカップリング・ファシリティ 1 にあります。OSAM と VSAM の構造は、カップリング・ファシリティ 2 にあります。

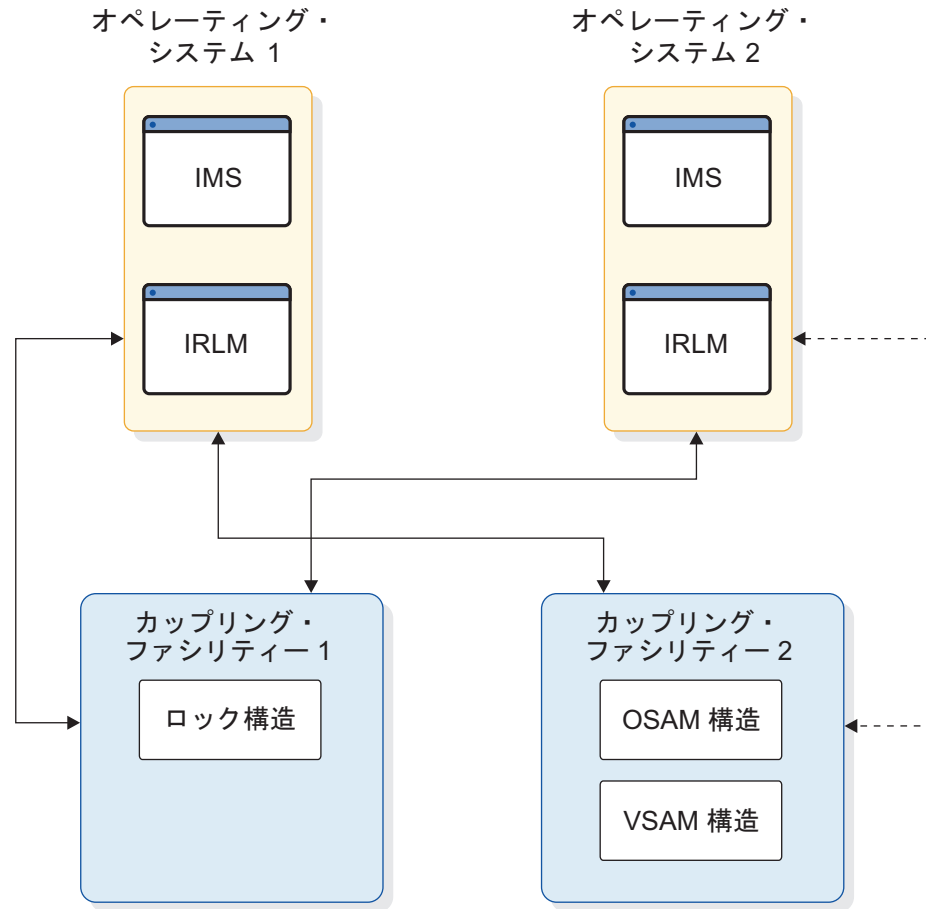


図 38. 2 つのカップリング・ファシリティ上に 3 つの構造を持つシスプレックス・データ共有構成

バックアップを持つ 1 つのカップリング・ファシリティの 3 つの構造

次の構成は、IRLM を使用した IMSplex で稼働する複数の z/OS イメージを示しています。IMS は、それぞれの z/OS イメージで稼働しています。2 つのカップリング・ファシリティが使用されています。ロック、OSAM および VSAM の各構造はカップリング・ファシリティ 1 にあります。カップリング・ファシリティ 2 はバックアップとして使用されており、カップリング・ファシリティ 1 で障害が発生した場合、これらの構造がカップリング・ファシリティ 2 に再作成できるようになっています。

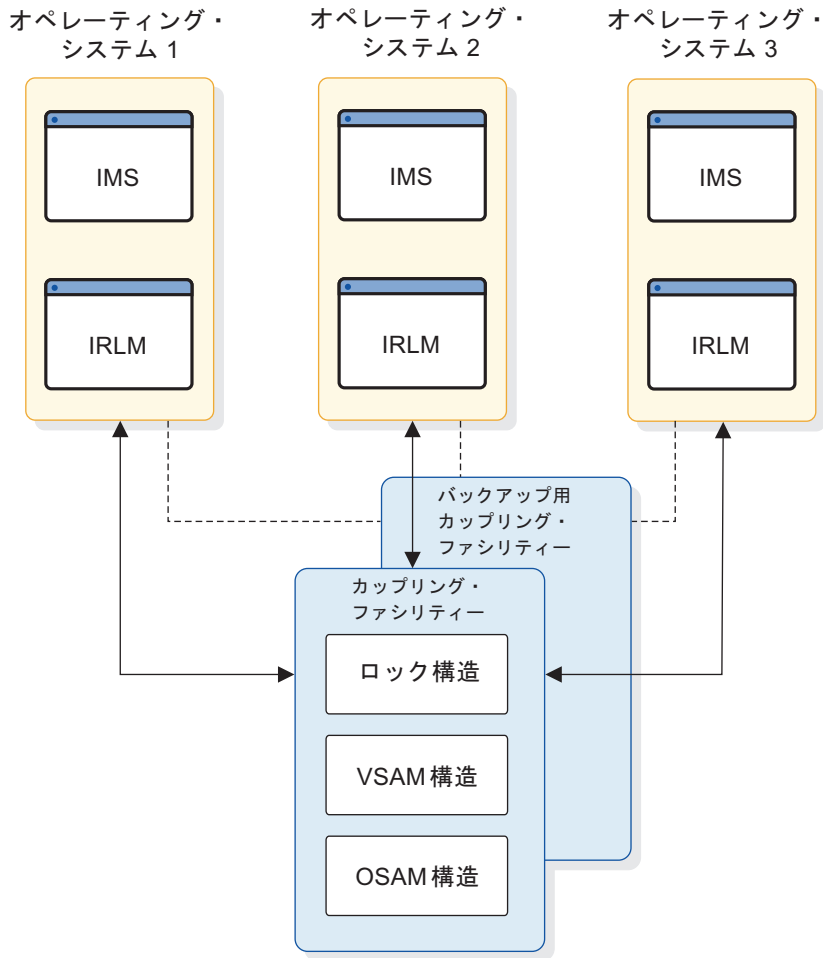


図 39. バックアップのある 1 つのカップリング・ファシリティ上に 3 つの構造を持つシスプレックス・データ共用構成

バックアップおよび XRF を持つ 1 つのカップリング・ファシリティの 3 つの構造

次の構成は、IRLM を使用した IMSplex で稼働する 6 つの z/OS イメージを示しています。3 つのアクティブ (XRF) IMS システムが稼働していて、それぞれのアクティブ IMS が代替システムを持っています。ロック、OSAM、および VSAM の各構造は、カップリング・ファシリティに存在します。

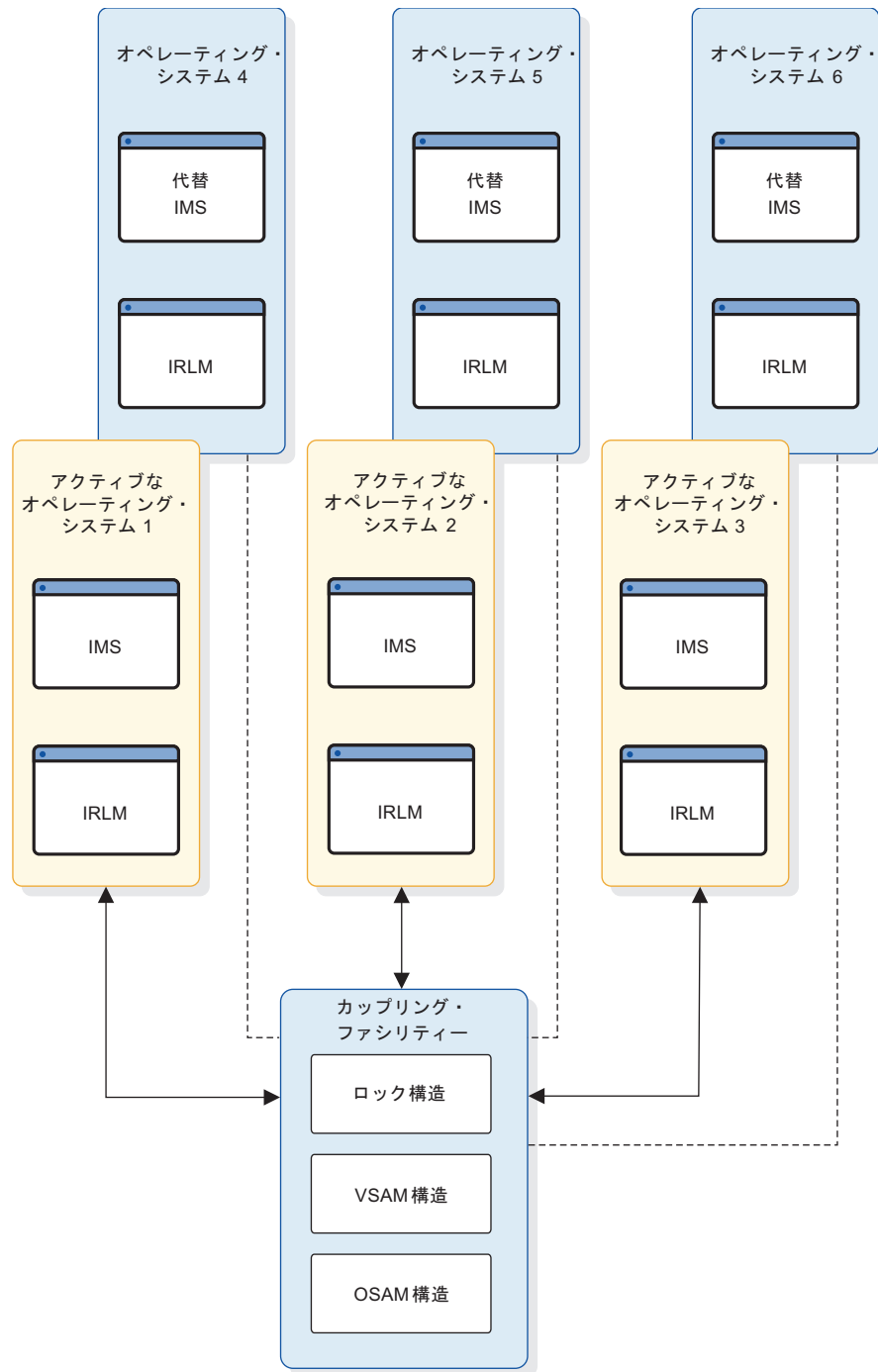


図 40. XRF 環境においてバックアップのある 1 つのカップリング・ファシリティ上に 3 つの構造を持つシブプレックス・データ共用構成

1 つのカップリング・ファシリティの 1 つの構造

次の構成は、IRLM を使用したデータ共用 IMSplex で稼働する複数の z/OS イメージを示しています。IMS は、それぞれの z/OS イメージで稼働しています。1 つのカップリング・ファシリティが使用されています。ロック構造が、そのカップリング・ファシリティにあります。OSAM 構造または VSAM 構造は、カップリング・ファシリティにありません。この構成では、バッファ無効化のための通

知プロトコルを使用するデータ共有 IMSplex が確立されます。

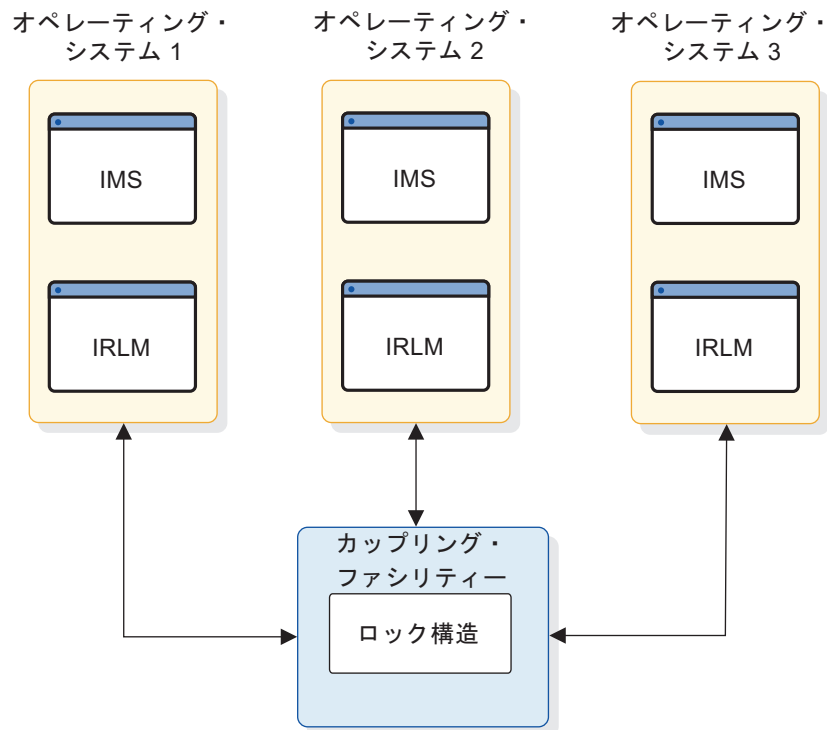


図 41. シスプレックス・データ共有となる 1 つのカップリング・ファシリティの 1 つの構造

シスプレックス・データ共有をどのような場合に使用するか

シスプレックス・データ共有環境にマイグレーションする 2 つの主要な理由は、コストと可用性に関係があります。z/OS シスプレックス・データ共有機能を使用すると、データベースの 1 つの論理ビューを保持しつつ複数の CPU へ作業負荷を分散させることができ、CMOS ベースのハードウェアを低コストで利用できる利点があります。これは、アプリケーション・ソフトウェアをまったく変更しないで実施できます。

シスプレックスの 1 つの CPU が失われると、作業負荷を残りの CPU に移動できるので、この環境での可用性は改善されます。

シスプレックス・データ共有のためのバッチ・ジョブ変更

OSAM または VSAM 構造に接続されている各 IMS は、各カップリング・ファシリティ構造への 1 つの接続またはスロットを使用します。したがって、データ共有モードで実行しているバッチ・ジョブはすべて、スロットを 1 つ使用します。

(アクティブのオンライン・ジョブとバッチ・ジョブの組み合わせの後に開始された、) 最大値を超えるジョブを同じデータ共有グループ内で実行することはできません。これらのバッチ・ジョブをバッチ指向の BMP ジョブに変換すれば、おそらくほとんどのジョブは実行可能になります。これは、BMP ジョブがオンライン領域のカップリング・ファシリティの接続下で実行されるためです。

バッチ・ジョブのバッチ指向 BMP ジョブへの変換については、「IMS V15 アプリケーション・プログラミング」を参照してください。

カップリング・ファシリティ構造のサイズの計算

カップリング・ファシリティ内の構造のサイズは、z/OS の計算式を使用して決定されます。この計算式で特定の IMS 値を算出しなければなりません。

推奨事項: 構造サイズの計算には、ピーク処理期間のトランザクション率を使用してください。そうすることにより、構造サイズのリセットを回避できます。サイズを大きく見積もりすぎても何ら問題はありませんが、小さく見積もりすぎると、異常終了することがあります。

ロック構造のサイズの計算

カップリング・ファシリティ・ロック構造のサイズは、トランザクション速度、更新比率、競合率、コミット頻度など、ご使用のシステムに固有の各種要因により異なります。したがって、見積もり、監視、および調整を必要としないロック構造の最適サイズを計算する方法はありません。

最も単純な方法は、256 MB または 512 MB など、2 のべき乗である適当なサイズから始めることです。そして、更新比率と競合率を監視します。ロック構造のサイズを調整して許容できるレベルまで競合を下げ、変更 (更新) ロックで使用可能なだけの十分な余裕があることを確認します。ロック構造が大きすぎても、パフォーマンスに不利な影響が出る可能性があります。

z/OS カップリング・ファシリティ構造サイズ決定支援ツール (CFSizer) も使用できます。CFSizer は、ユーザーから提供される入力データを基にした構造サイズを計算する、Web ベース・アプリケーションです。CFSizer を使用した場合は、更新比率および競合率を監視して、ロック構造のサイズを必要に応じて調整する必要があります。CFSizer ツールを使用するには、<http://www.ibm.com/systems/support/z/cfsizer/> にアクセスしてください。

初期値の設定後、IRLM メッセージ (メッセージによっては、構造のサイズを大きくするよう提案するものがあります) を使用して、ロック構造の使用をモニターしてください。現行のロック構造の 50% が使用されている場合は、IRLM メッセージの受信が開始されます。

カップリング・ファシリティのロック構造は 2 つの部分を含んでいます。最初の部分は、ロック・エントリ・テーブルであり、IMS 間読み取り/書き込みインタレストが特定のハッシュ・クラス (ロック・テーブル内の特定場所にハッシュされるリソース) にあるかどうかを判別するのに使用されます。2 番目の部分は、現在保留されている更新ロックのリストです (変更ロック・リストまたはレコード・リスト・テーブルと呼ぶこともあります)。ユーザーは、IRLM DXRJPROC プロシージャで、または IRLM MODIFY コマンドを使用して、これら 2 つのコンポーネント間でのロック構造ストレージの分割を制御できます。構造の分割比率をユーザーが指定しなければ、IRLM は、ロック・テーブル項目とレコード・リスト・ストレージ間の比率 1:1 でそれを分割しようとします。ロック構造の合計サイズは、十分に大きくして、ハッシュ・コンテンションの制限によるパフォーマンス上の問題を回避し、また MODIFY 項目 (RLE) を書き込むレコード・リスト・ストレージの

不足による障害を回避する必要があります。ロック・テーブル項目の数を適切に指定すれば、ハッシュ・コンテンションを回避するのに役立ちます。

IRLM は、レコード・テーブル項目の 10% を「完了必須」機能 (ロールバックやコミット処理など) のために予約するので、ストレージ不足によって IMS システム障害が発生することはありません。しかし、レコード・テーブルでストレージ不足が生じた場合、可用性への影響 (トランザクションが終了します)、応答時間への影響、およびスループットへの影響が起こり得ます。

ロック項目サイズの指定

インストール・パネル DSNTIPJ のフィールド LOCK ENTRY SIZE で、ロック・コンテンション制御情報 (すなわち、ロック・テーブル内の個々の項目) に必要なスペース量が決まります。ロック項目サイズ、および構造割り振りを引き起こすグループを結合させる最初の IRLM のロック・テーブル項目の数によって、グループ全体についてのロック・テーブルおよびロック・テーブル項目幅に必要なストレージ・サイズが決まります。デフォルトは 2 バイトであり、これはユーザーが必要とする数値と思われる。ただし、7 つ以上のメンバーからなるデータ共有グループを即時に作成する場合は異なってきます。各ロック項目を 2 バイトに制限すると、定義構造サイズから得られる LTE スペース量は最大になります。これは、偽コンテンションを回避するのに役立ちます。

ロック構造のストレージの見積もり

インストール計画の場合、ロック構造の初期サイズは、ユーザーが行う更新の量によって異なります。

推奨事項: IRLM DXRJPROC プロシージャで LTE= キーワードを指定しない場合、INITSIZE には 2 のべき乗の値を選んでください。そのようにすると、IRLM は、半分がロック・テーブル項目に、残りの半分がレコード・テーブル項目に使用されるようにカップリング・ファシリティのストレージを割り振ることができまます。合計サイズが 2 のべき乗でないときに 1:1 の分割が行われると、レコード・テーブル項目の深刻なスペース不足が起こり、IMS の障害、おそらく IRLM の障害が生じる可能性があります。(これが生じるのは、CONNECT で要求するロック・テーブル項目の数が 2 のべき乗でなければならないからです。) 構造が小さすぎるか、あるいはロック・テーブルの割り振りで残されたレコード・テーブル用のストレージが小さすぎる場合に、レコード・テーブルはストレージ不足になりやすくなります。

IRLM DXRJPROC プロシージャで LTE= パラメーターに値を指定する場合、あるいは IRLM MODIFY SET,LTE= コマンドを発行する場合、XES コンテンション率をモニターして、通常的环境に最も適した値を判断する必要があります。コンテンツン率が高いように思われる場合には、LTE= 値を増やして次の 2 のべき乗にしてください。その際に、ロック・テーブルのサイズを増やすと、構造サイズも同時に増やさない限り、それに応じてレコード・テーブルが減少することに留意してください。コンテンツン率がほとんどない場合に、レコード・テーブル項目にさらにストレージを使用したいときは、2 のべき乗単位で LTE= 値を減らしてください。ロック・テーブル項目の数を減らすときにはいつも、しばらくの期間、コンテンツン率をモニターすることをお勧めします。

構造の割り振りは CONNECT 時に行われるので、LTE= 値に行われた変更は、グループが終了するか、構造が強制されてグループが再始動されるか、あるいは REBUILD が実行されない限り、有効になりません。また、CONNECT する最初の IRLM の LTE= 値は、グループが使用するカップリング・ファシリティの属性を指示します。

ロック構造のサイズの自動変更

z/OS は、ストレージ・スペースを必要とする場合、カップリング・ファシリティ内のロック構造のサイズを自動的に拡大または縮小することができます。CRFM ポリシーの ALLOWAUTOALT パラメーターは、システムによって開始される変更(自動変更) がロック構造で許可されるかどうかを指定します。競合するリソースがカップリング・ファシリティを使用していない場合、ALLOWAUTOALT(YES) を指定して、ワークロードの増大に応じてロック構造を変更できるようにすることが望ましい可能性があります。ただし、このパラメーターが YES に設定されていると、他のリソースがロック構造を強制的に減らすことができるため、カップリング・ファシリティの使用中にロック・テーブルで IRLM ストレージ不足状態が発生する可能性があります。この場合、アプリケーションは U3307 で異常終了します。さらに、IMS 制御領域は、U0113 または U1027 の異常終了で終了する可能性があります。

ロック構造が安定している必要がある場合は、ALLOWAUTOALT(NO) を使用してください。

OSAM および VSAM 構造のサイズの計算

シスプレックス・データ共有では、各アクセス方式が必要とする構造のサイズは、データ共有グループ内の個々の IMS がアクセス方式に定義しているバッファの数によって異なります。すべての IMS 制御領域用のバッファと、データを共有していて IRLM に登録されているバッチ・ジョブ用のバッファをカウントしなければなりません。

OSAM バッファのカウンートを計算する公式を以下に示します。

OSAM buffer count = #osambfrs/IMS1 + #osambfrs/IMS2 + ... # osambfrs/IMS_n

VSAM バッファのカウンートを計算する公式を以下に示します。

VSAM buffer count = #vsambfrs/IMS1+ #vsambfrs/IMS2 + ... #vsambfrs/IMS_n

以下の 2 つの IMS システム (IMS1 および IMS2) を持つ VSAM 構造のサイズ変更の一例をここに示します。

- 次の指定に基づき、IMS1 は合計 700 個のバッファ (400 個の VSAM バッファに加えて 300 個のハイパースペース・バッファ) を持つ。

```
VS RBF=512,30
VS RBF=1024,20,I
VS RBF=1024,10,D
VS RBF=2048,40
VS RBF=4096,100,HS100,HSR
VS RBF=8192,100,HS100,HSR
VS RBF=16384,50,HS50,HSR
VS RBF=32768,50,HS50,HSR
```

- 次の指定に基づき、IMS2 は合計 400 個のバッファを持つ。

VSRBF=512,30
VSRBF=1024,20,I
VSRBF=1024,10,D
VSRBF=2048,40
VSRBF=4096,100
VSRBF=8192,100
VSRBF=16384,50
VSRBF=32768,50

両方の IMS システムに関するバッファの総数を加算することにより (700 + 400)、VSAM 構造のサイズ変更使用する数値 1100 が得られます。

推奨事項: OSAM または VSAM バッファ・プール定義が変更されたときは常に、OSAM 構造および VSAM 構造のサイズを検討してください。これらの構造のサイズが十分でない場合は、データ共用パフォーマンスが低下する可能性があります。

OSAM 構造は、キャッシング・データ用としてオプションで使用できます。データ・キャッシングを使用しない場合、この構造にはディレクトリ項目のみが含まれます。データ・キャッシングを使用する場合、この構造にはディレクトリ項目とデータ・エレメントが含まれます。この場合、構造のサイズはデータ・エレメントを許容できる大きさである必要があります。OSAM データは、複数の 2 KB データ・エレメントとして OSAM 構造に格納されています。データ・キャッシングを使用するには、ディレクトリとエレメントの比率を指定する必要があります。構造は、この比率でディレクトリ項目とデータ・エレメントに分割されます。この比率は、CFNAMES パラメーター・ステートメントの CFOSAM= キーワードで指定します。

CFNAMES パラメーター・ステートメントの CFOSAM= キーワードでこの比率を指定する方法についての詳細は、IMS V15 システム定義を参照してください。

要件: OSAM バッファ・カウントには、定義されている順次バッファリング・バッファをすべて入れなければなりません。VSAM バッファ・カウントには、定義されているハイパースペース・バッファをすべて入れなければなりません。

z/OS 公式

OSAM バッファ・カウントを計算した後、結果を z/OS キャッシュ構造サイズの計算式の TDEC フィールドに入れます。次に、VSAM バッファ・カウントにも同じことを行います。(計算は、各アクセス方式ごとに別々に行われます。) TDEC は、IMS バッファ (OSAM または VSAM) の数です。

OSAM データベース・カップリング・ファシリティ・キャッシングについては、IMS V15 システム定義を参照してください。

OSAM および VSAM 構造のサイズ変更

構造変更を使用すると、OSAM および VSAM キャッシュ構造のサイズ、または構造内の再配分オブジェクトのサイズを動的に変更できます。

また、構造が満杯になったときに、自動的に構造変更を実行して、構造サイズおよびエレメントと項目の比率を変更することもできます。構造変更をサポートするためには、カップリング・ファシリティのレベルが 1 でなければなりません。

構造変更について詳しくは、213 ページの『CQS における構造変更の使用』を参照してください。

シスプレックス・データ共有の障害からのリカバリー

シスプレックス・データ共有環境で、複数の障害が起こる可能性があります。

複数の障害が同時に発生した (例：同じカップリング・ファシリティーで IRLM 構造および OSAM 構造に障害が起こる) 場合、各コンポーネント (IRLM、IMS など) は、適切なりカバリー・アクションを行います。

カップリング・ファシリティーの接続の障害

さまざまな種類の問題が原因で、IMS およびカップリング・ファシリティーの間の接続が失われます。例えば接続は、物理リンク (光ファイバー・ケーブル) の切断、電源障害の後、ハードウェア・チェックによる停止、カップリング・ファシリティー・マイクロコード障害、および同様のイベントなどで失われます。

IMS およびカップリング・ファシリティーの間に複数のリンクがある場合、1 つのリンクが失われても IMS 上または IRLM には影響がありません。ただし、失われたリンクが最後のリンクの場合、そのリンクをリカバリーしないと、その IMS のデータ共有が再始動できません。処理はデータ共有グループ内のその他すべての IMS システムに対して継続されたままです。

IMS とカップリング・ファシリティーの間の最後の接続が失われた場合、以下のようになります。

- 接続が IMS および OSAM 構造または VSAM 構造の間にあった場合は、IMS はデータ共有からドロップアウトします。共有データにアクセスしようとするあらゆるトランザクションが U3303 異常終了コードで終了します。ただし、非共有データベースにアクセスするトランザクションは、処理を続行できます。
- 失敗した接続が IRLM と IRLM ロック構造の間にあった場合、IRLM はデータ共有グループから切断されます。共有データにアクセスしようとするあらゆるトランザクションが U3303 異常終了コードで終了します。

カップリング・ファシリティーおよび IMS の間の接続が再確立される時、z/OS は IMS に通知し、IMS はデータ共有を再開します。

以下の図は、カップリング・ファシリティーの接続の障害の結果としての、IMS、OSAM 構造、および VSAM 構造間の接続の逸失を示しています。

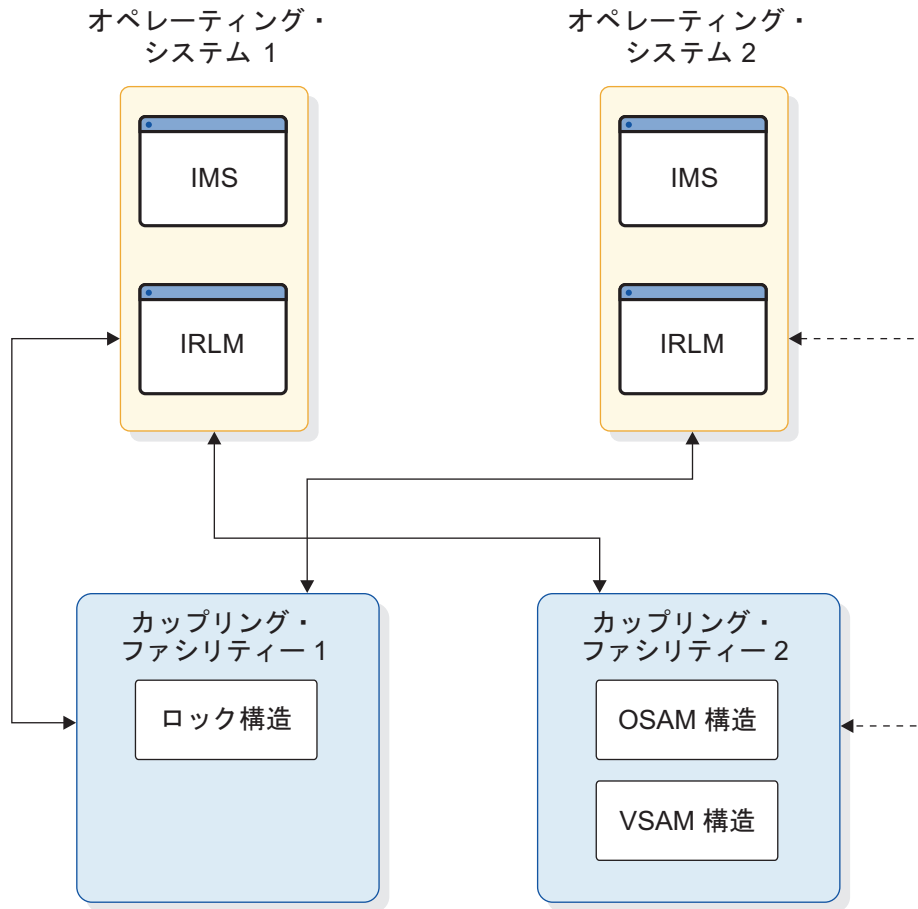


図 42. IMS と OSAM 構造および VSAM 構造間の接続の逸失

失われた接続は MVS2 とカップリング・ファシリティ CF02 の間のリンクです。最後のリンクに障害が起こると、次のようになります。

- IMS2 はデータ共用からドロップアウトし、MVS2 システム上でアクセスできるのは非共用データベースだけです。IMS1 および IMS3 はデータ共用を継続します。
- MVS2 上の共用データにアクセスしようとするトランザクションは、U3303 異常終了コードで終了します。

この状況では、DATA SHARING STOPPED メッセージを受け取ります。z/OS DISPLAY XCF,STRUCTURE,STRNAME= コマンドを入力して、何に障害が起こったか判別できます。以下の例にあるように、表示された出力はカップリング・ファシリティへの MVS2 リンクが失敗したことを示しています。MVS1 リンクおよび MVS3 リンクはまだアクティブです。

```
11:44:37.84 D XCF,STRUCTURE,STRNAME=OSAMSESXI
```

```
⋮
```

```
& AMPERSAND DENOTES CONNECTOR WHO LOST CONNECTIVITY TO STRUCTURE
CONNECTION NAME ID VERSION SYSNAME JOBNAME ASID STATE
-----
```


IMS1	03 0003000D MVS1	DLIOCSA8 0066 ACTIVE
&IMS2	01 0001001D MVS2	DLIOCSB8 0066 FAILING
IMS3	02 00020014 MVS3	DLIOCS8 0066 ACTIVE

オンライン環境では、OSAM 構造への MVS2 リンクが再確立され、MVS は IMS に通知し、データ共有が再開されます。バッチ環境では、リンクが再確立される際に、バッチ・ジョブを再実行依頼する必要があります。ユーザーは、IMS がデータ共有を再開したことを示す CF INITIALIZATION COMPLETE メッセージを受け取ります。DISPLAY XCF,STRUCTURE コマンドを出した場合、表示出力は MVS2 が接続されている (ACTIVE) ことを示します。

カップリング・ファシリティー構造の障害

IMS バッチ環境でカップリング・ファシリティー構造に障害が起きた場合は、すべてのバッチ・ジョブが終了します。DB/DC または DBCTL 環境では、データ共有グループのすべての IMS システムは、一時的にデータ共有を中断します。

構造に障害が起こった場合、何が起るかは、操作環境によって異なります。

バッチ環境

カップリング・ファシリティーに接続されていたすべての IMS バッチ・ジョブは、U3303 異常終了コードで終了します。

DB/DC または DBCTL 環境

データ共有グループのすべての IMS システムは、一時的にデータ共有を中断します。障害が起こった構造が OSAM または VSAM 構造であった場合、IMS は構造を再作成しようとします。再作成が正常終了すると、すべての IMS システムは新しい構造に接続し、データ共有が継続します。障害が起こった構造が IRLM 構造であった場合、すべての IRLM は IRLM 構造を再作成しようとします。再作成が正常終了すると、すべての IRLM は新しい構造に接続し、データ共有のロックが継続します。

例: 350 ページの図 42 は OSAM 構造の構成に障害が起こったことを示しています。カップリング・ファシリティー CF02 上の OSAM 構造に障害が起こると、以下が発生します。

- すべての IMS システムはデータ共有を延期します。この時点で構造の状況を表示すると (DISPLAY XCF,STRUCTURE,STRNAME=OSAMSESXI コマンド)、出力は以下の例に示されたように、構造が再作成の静止フェーズにあることを示しています。

```
08:50:59.29 D XCF,STRUCTURE,STRNAME=OSAMSESXI
08:51:13.32 IXC360I 08.50.59 DISPLAY XCF 143
STRNAME: OSAMSESXI
STATUS: REASON SPECIFIED WITH REBUILD START:
        STRUCTURE FAILURE
        REBUILD PHASE: QUIESCE
POLICY SIZE      : 2048 K
PREFERENCE LIST: CF02    CF01
EXCLUSION LIST IS EMPTY
```

⋮

```
* ASTERISK DENOTES CONNECTOR WITH OUTSTANDING REBUILD RESPONSE
CONNECTION NAME ID VERSION SYSNAME JOBNAME ASID STATE
-----
```

```

*IMS1          03 00030015 MVS1      DLIOCSA8 003A ACTIVE
*IMS2          01 00010026 MVS2      DLIOCSB8 0038 ACTIVE
*IMS3          02 0002001D MVS3      DLIOCSC8 0038 ACTIVE

```

- すべての IMS システムがデータ共用を延期すると、新しい構造が作成されます。次に、すべての IMS システムが新しい構造に接続され、データ共用が再開します。再作成後に新しい構造の状況を表示する場合、出力は以下の例に示されたものと同様になります。

```

08:52:39.78 D XCF,STRUCTURE,STRNAME=OSAMSESXI
08:52:40.43 IXC360I 08.52.39 DISPLAY XCF 168
STRNAME: OSAMSESXI
STATUS: ALLOCATED
POLICY SIZE      : 2048 K
PREFERENCE LIST: CF02      CF01
EXCLUSION LIST IS EMPTY

ACTIVE STRUCTURE
-----
ALLOCATION TIME: 03/25/94 08:51:27
CFNAME          : CF01
COUPLING FACILITY: ND01...
PARTITION: 0    CPCID: 00
ACTUAL SIZE     : 2048 K

```

⋮

```

CONNECTION NAME  ID VERSION  SYSNAME  JOBNAME  ASID STATE
-----
IMS1             03 00030015 MVS1     DLIOCSA8 003A ACTIVE
IMS2             01 00010026 MVS2     DLIOCSB8 0038 ACTIVE
IMS3             02 0002001D MVS3     DLIOCSC8 0038 ACTIVE

```

カップリング・ファシリティーの再作成での障害

カップリング・ファシリティーの再作成を z/OS SETXCF START,REBUILD コマンドを出して開始できます。

z/OS は以下への応答での再作成も開始できます。

- カップリング・ファシリティー構造の障害
- カップリング・ファシリティーを再作成するためのシステム障害管理 (SFM) のアクティブ・ポリシーがしきい値に達した場合、接続は失敗します。

再作成に障害が起こると、IMS はデータ共用を停止します。共用データにアクセスしようとするトランザクションは、U3303 異常終了コードで終了します。再作成の失敗が OSAM または VSAM 構造と関連していた場合、非共用データベースにアクセスするトランザクションは処理を続けることができます。障害が IRLM 構造と関連していた場合、IRLM はデータ共用グループから切断されます。

IMS 再始動中のカップリング・ファシリティーの接続の障害

カップリング・ファシリティーへのアクティブ・システムの接続が、IMS の再始動中に失敗した場合、再始動処理は継続します。

障害が XRF 代替サブシステム上にあり、代替サブシステムがカップリング・ファシリティーに接続できない場合、再始動の処理はできません。

IMS が IRLM に確認を試行しているときの障害

IMS がそれ自体を IRLM に確認できない場合、障害が発生します。障害が発生した場合、IMS は障害の理由と異常終了コードを示すメッセージを出します。

DISPLAY XCF,STRUCTURE コマンドおよび MODIFY irlmprocx,STATUS コマンドを使用して、取るべきアクションを判断するために情報を集めることができます。

この確認の失敗を除いて、IRLM の障害処理はシスプレックス以外のデータ共用と同じです。IMS:

- すべてのコミットされていないデータベース変更をバックアウトする。
- すべての共用データベースの許可を DBRC を介して除去する。システムは停止します。
- IRLM が再始動されるのを待つ。

システム・リカバリーが必要な z/OS およびその他の障害

システム・リカバリーが必要な z/OS の障害が起こるか、またはその他の障害が起こった場合 (例：停電)、IMS 処理はシスプレックス以外のデータ共用環境内での処理と同じです。

障害が起こったシステムの場合、次のように操作します。

- z/OS の再始動
- IRLM の再始動
- バッチ環境では、必要に応じてバッチ・ジョブのバックアウト
- IMS DB/DC または DBCTL 環境での、IMS の緊急時再始動 (/ERESTART コマンドの使用)

異なるシステムで z/OS および IRLM を再始動できます。また、バッチ・バックアウトおよび緊急時再始動は、データ共用グループ内のどの z/OS および IRLM にも実行できます。

障害が起こっていないシステムの場合:

- ロックが、障害が起こった IMS に対して保存されます。
- 障害が起こった IMS が再始動し、データベースがリカバリーされると、ロックが解放されます。
- 障害が起こった IMS のデータベース・バッファが無効にされます。
- データベース・エラー条件が、その他の IRLM に送られます。

複数の IMS システム障害後の IMS 再始動の同期化

データ共用環境では、複数の IMS システム障害が発生する可能性があります。こうした事態が発生した場合、障害の起きたすべての IMS システムの始動を同期化することが必要になる場合があります。また、災害時回復サイトのすべての IMS システムの始動の同期化も検討してください。

シスプレックスの障害、または災害時回復サイトにおける再始動が発生した場合、障害が起きた IMS システムが再始動されるまで、その障害が起きた IMS システムで保留されているデータベースまたは DEDB エリアに対しては、いずれの新しい許

可も付与されません。障害が起きた各 IMS システムが一度に 1 つずつ再始動して、通常の処理を始める場合、障害が起きた IMS システムにより許可されたデータベースまたは DEDB エリアに対する許可も失敗します。全機能データベースの場合、メッセージ DFS047 が戻りコード X'09' とともに出されます。DEDB エリアの場合、メッセージ DFS3709 が戻りコード X'09' とともに出されます。

障害が起きた IMS システムのすべてが再始動したら、これらのデータベースおよび DEDB エリアを手動で開始する必要があります。 /ERE コマンドの OPTION キーワードおよび SYNCPLEX パラメーターを使用して、障害が起きた IMS システムの再始動の直列化を行うことができます。障害が起きた各 IMS システムに対しコマンドを発行してください。

IMS 再始動処理完了後、要応答オペレーター宛 (WTOR) メッセージ DFS3067A が出されます。すべての IMS システムがこのメッセージを受け取る場合、オペレーターは各 WTOR に応答できます。通常の処理が再開し、データベースおよび DEDB エリアに対する許可を付与できるようになります。

OPTION SYNCPLEX キーワードおよびパラメーターは、/ERE コマンドでのみ有効です。

IMS 障害後のデータベースの許可

データ共用環境で 1 つの IMS システムに障害が起きた場合、その障害が起きた IMS システムによって保留されているデータベースまたは DEDB エリアの許可を試みている他の IMS システムに障害が起きます。

全機能データベースの場合、メッセージ DFS047 が戻りコード X'09' とともに出されます。高速機能 DEDB の場合、メッセージ DFS3709 が戻りコード X'09' とともに出されます。戻りコード X'09' は、データベースまたはエリアが、1 つ以上のオンライン IMS システムで前に許可されたことがあることを示します。IRLM 出口検査は、IMS システムが非アクティブであることを示します。そのため、識別されたデータベースまたは DEDB エリアは許可されません。

これが起きるのを防ぐため、IRLM パラメーター SCOPE=NODISCON を指定してください。これは IRLM がデータ共用環境にあり、システム間の共用が行われることを示します。データ共用グループに識別されている IMS がないときでも、IRLM は、データ共用グループに接続されたままです。

IMS システムに障害が起きたとき、IRLM は、その障害が起きた IMS の状況を記憶しています。障害が起きた IMS により保留されているデータベースに対する新規の許可がすべて許可されます。再始動時の IRLM 出口検査処理によって、障害が起きた IMS システムの状況が認識されて、新しい許可が付与されます。

第 17 章 VTAM 汎用リソース・グループの計画

このトピックでは、VTAM 汎用リソース・グループの使用に関連する計画および管理のための作業について説明します。

VTAM 汎用リソース・グループを使用するための要件

VTAM 汎用リソース・グループを使用してセッションを平衡化するには、いくつかの要件があります。

これらの要件には、以下のものが含まれます。

- シスプレックス環境。
- ISTGENERIC という名前の z/OS カップリング・ファシリティ構造が、シスプレックス環境に関するアクティブな CFRM ポリシーに定義されていること。
- VTAM APPN ノードと ISTGENERIC カップリング・ファシリティ構造との間の接続。
- 汎用リソース・グループに参加しているすべての IMS サブシステムが、シスプレックス環境内にあること。
- 汎用リソース・グループに参加しているそれぞれの IMS サブシステムが、同じ IMS 汎用リソース名を共用すること。
- LU 6.2 通信用の APPC 汎用リソース名が z/OS に対して定義されていること。この名前は、APPC/MVS APPCPMxx メンバー内の LUADD ステートメントの GRNAME パラメーターで定義してください。
- 汎用リソース・グループに属するすべての IMS システムが、等価の仕様を使用して定義されること。

関連資料: VTAM および ISTGENERIC カップリング・ファシリティ構造の定義に関する情報については、「z/OS Communications Server: SNA ネットワーク・インプリメンテーション・ガイド」を参照してください。

VTAM 汎用リソース・グループの制約事項

VTAM 汎用リソース・グループを作成する場合、システム・プログラマーは制約に注意する必要があります。

以下の制約事項があります。

- 特定の IMS を複数の VTAM 汎用リソース・グループのメンバーにすることはできません。
- GRNAME パラメーターに指定する汎用リソース名が XRF USERVAR の名前に一致しないようにしてください。

XRF および VTAM 汎用リソースについて詳しくは、358 ページの『XRF と VTAM 汎用リソース』を参照してください。

VTAM 汎用リソース類似性

特定の端末での作業を実行するために VTAM が汎用リソース・グループ内の特定の IMS サブシステムを選択すると、VTAM 汎用リソース (VGR) 類似性が確立され、IMS または VTAM によって端末の類似性がリセットされるまで、端末の後続のセッションはすべて同じ IMS サブシステムに接続されます。

IMS は、各セッションの状況リカバリー・モードに応じて、VGR 類似性管理をそのセッションの IMS または VTAM に割り当てます。IMS は、GRAFFIN= パラメーターを無視します。このパラメーターは、VGR では使用されなくなりました。複数システム結合機能 (MSC) リンクでは、ローカルな類似性のみ使用します。

注: ISC 端末および MSC リンクに関しては、VTAM および IMS は、すべての並列セッションが終了するまで VGR 類似性をリリースしません。

VGR 類似性管理

IMS と VTAM のいずれも VGR 類似性を管理できます。これらの 2 つの間の重要な相違点は、IMS 障害時に端末の類似性をリセットする能力にあります。端末または MSC リンクが、障害を起こした IMS との類似性を持つ場合、端末はその類似性がリセットされるまで作業を継続できません。

- IMS 管理型 - セッション (または IMS) 終了時に、IMS が VTAM に VGR 類似性削除のタイミングを指示します。IMS は、ローカル IMS システム上に端末の有効状況が存在しない限り、類似性を削除します。
- VTAM 管理型 - VTAM が、セッション (または IMS) 終了時に VGR 類似性を削除します。

VTAM が類似性を管理しているときに IMS が障害を起こすと、VTAM は、障害を起こした IMS との間で開かれたセッションを持つ端末または MSC リンクの類似性をリセットすることができます。それらの類似性がリセットされると、端末は汎用リソース・グループ内の別の IMS との新規セッションを開くことができます。

IMS が類似性を管理しているときに IMS が障害を起こすと、IMS は、端末と障害を起こした IMS との間の類似性をリセットできない場合があります。この場合、障害を起こした IMS が再始動されるまで、端末は新規セッションを開くことができません。MSC リンクの場合、類似性の設定時に、別の IMS 上のクローン・リンクにセッションを移動できません。

端末のタイプとその端末に設定した状況リカバリー・モードによって、IMS または VTAM のいずれかが VGR 類似性を管理するかが決定されます。

IMS 管理類似性

IMS は以下の場合の VGR 類似性を管理します。

- リカバリー・モード LOCAL を伴った静的端末。ローカル IMS システムに有効状況が存在する間は、端末には他の IMS システムへのアクセスが許可されてはならないので、IMS は VGR 類似性を設定します。IMS は、VGR 類似性が RM 類似性に一致していることを確認します。

- 状況リカバリー・モード LOCAL を伴った動的 STSN 端末。ローカル IMS システムに有効状況が存在する間は、端末には他の IMS システムへのアクセスが許可されてはならないので、IMS は VGR類似性を設定します。ユーザー名および状況リカバリー・モードはログオン時に提供される必要があるため、IMS は VGR 類似性が RM 類似性に一致していることを確認します。
- MSC リンクは、ローカル・モードおよび IMS 管理型類似性のみ使用し、RM をサポートしていません。

VTAM 汎用リソース・グループの作成

VTAM 汎用リソース・グループに参加するすべての IMS システムに同じ汎用リソース名を指定して、VTAM 汎用リソース・グループを作成します。

それぞれの IMS ごとに VTAM 汎用リソース名を指定するには次の 2 つの方法があります。

- IMS プロシージャおよび DCC プロシージャ内の GRSNAME= 始動パラメーター。
- IMS の制御領域を開始する時の /START VGR コマンドの GRSNAME パラメーター。

制約事項: GRSNAME パラメーターに指定する汎用リソース名が XRF USERVAR の名前に一致しないようにしてください。

関連資料:

- GRSNAME= 始動パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。
- /START VGR コマンドについて詳しくは、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

APPC 汎用リソース名の指定

APPC 汎用リソース名を指定することによって、LU 6.2 装置を汎用リソース・グループが提供するセッション平衡化に参加させることができます。APPC 汎用リソース名は、APPC/MVS APPCPMxx メンバー内の LUADD ステートメントの GRNAME パラメーター、または SET APPC z/OS オペレーター・コマンドを使用して指定してください。

制約事項: APPC 汎用リソース名は、GRSNAME 実行パラメーターで指定する名前と同じであってはなりません。

VTAM 汎用リソース・グループ内の IMS とのセッションの開始

端末セッションは、IMS または端末のいずれかから開始することができます。IMS によって開始されるセッションはすべて、汎用セッションです。すなわち、それらのセッションにおいては類似性が確立されます。MSC セッションは、/RSTART LINK コマンドにより開始されます。

端末からのセッションは、次の方法で開始することができます。

- GRSNAME 実行パラメーターを指定することによって、汎用リソース・グループ内の任意の IMS とのセッションを開始することができます。その結果、選択された IMS システムとの類似性が確立されます。
- システムの APPLID 名を指定することによって、特定の IMS とのセッションを開始することができます。類似性は確立されません。
- 汎用リソース・グループ内の IMS が MNPS を使用する XRF システムの一部でもある場合は、APPLID 名の代わりに MNPS 名を使用して、その特定の IMS とのセッションを開始することができます。類似性は確立されません。

関連資料:

- XRF および VTAM 汎用リソースについて詳しくは、『XRF と VTAM 汎用リソース』を参照してください。
- XRF システムおよび MNPS 名の指定について詳しくは、814 ページの『IMS 実行 JCL の調整』を参照してください。

XRF と VTAM 汎用リソース

拡張回復機能 (XRF) システムにより、セッション持続性と IMS リソースの可用性が保証されます。XRF システムには、2 つのタイプがあります。VTAM マルチノード持続セッション (MNPS) を使用する XRF システムと、USERVAR を使用する XRF システムです。MNPS を使用する XRF システムは VTAM 汎用リソース・グループに参加することができますが、USERVAR を使用する XRF システムは参加することができません。

MNPS を使用する XRF と VTAM 汎用リソース

汎用リソース・グループ内の MNPS を使用する XRF システムの場合、VTAM は汎用リソース名を APPLID 名ではなく MNPS を使用する XRF システムの MNPS 名に関連付けます。MNPS を使用する XRF システムの一部である IMS に直接ログオンする端末は、MNPS 名を使用する必要があります。

汎用リソース・グループ内のすべての IMS もまた MNPS を使用する XRF システムの一部でない限り、MNPS を使用する XRF システムを汎用リソース・グループに組み込んでも、汎用リソース名を使用してログオンする端末のセッション持続性は保証されません。

MNPS を使用する XRF を使っており、かつ 3745 コントローラーからマイグレーションする場合、パフォーマンス上の制限を受ける場合があります。

推奨事項: これらのパフォーマンス上の制限を最小化するには、以下を実行してください。

- XRF を継続して使用するために、3745 コントローラーの置き換えとして、Communication Controller for Linux on System z® (CCL) を使用する。
- XRF から並列シスプレックス環境が必要な VTAM 汎用リソース (VGR) へマイグレーションする。

USERVAR を使用する XRF と VTAM 汎用リソース

USERVAR を使用する XRF システムは汎用リソース・グループに参加することはできませんが、汎用リソース・グループと同じシスプレックス内に共存することができます。この場合、汎用リソース名と USERVAR は同じであってはなりません。

USERVAR を使用する XRF システムは、共用キュー環境に参加することができます。

ログオン・プロシージャの変更

汎用リソース名または IMS APPLID 名のいずれかを使用してログオンに関するオプションを提供することができるので、ログオン・プロシージャでこれらの名前のいずれかまたは両方を使用できるように設計してください。

APPLID フィールドのオーバーライド

各 IMS は、IMS COMM システム定義マクロの APPLID= キーワード、および VTAM APPL 定義ステートメントで指定したアプリケーション名を使用して VTAM に識別されます。

IMS 定義をできるだけ同じにするために、汎用リソース・グループ内のすべての IMS システムについて、COMM マクロの APPLID= キーワードに同じ名前を指定することができます。その後、APPLID1 実行パラメーターに固有のアプリケーション ID を指定することによって、この名前をオーバーライドする必要があります。

同様に、PASSWD= 実行パラメーターを使用して、COMM マクロの PASSWD フィールドをオーバーライドすることができます。

関連資料: APPLID1 実行パラメーターおよび PASSWD 実行パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。

類似性を終了する時期の判別

ログオフによってセッションを終了すると、ユーザーのノードと IMS VTAM 汎用リソース・メンバーの間の類似性も IMS によって終了されます。しかし、状態によっては、類似性が持続します。

これらの類似性を終了するには、いくつかの方法を使用することができます。

MSC および ISC 並列セッションの類似性は、1 つ以上のセッションがアクティブである、または少なくとも 1 つのセッションが有効状況を持ち、かつリカバリー・モードがローカルである限り維持されます。MSC セッションは、/PSTOP LINK コマンドがすべての並列リンクに対し発行されると、VTAM 類似性を終了します。MSC リカバリー・モードは、常にローカルです。

ISC セッションは、以下の場合に VTAM 類似性を終了します。

- /QUIESCE NODE コマンドがすべてのセッションに対し発行され (コールド終了)、かつリカバリー・モードがローカルまたはグローバルである。

- /CLOSE NODE コマンドがすべての並列セッションに対し発行され (ウォーム終了)、かつリカバリー・モードがグローバルである。

VTAM 類似性は、IMS が /CHE FREEZE|PURGE|DUMPQ コマンドにより正常にシャットダウンされ、すべてのセッションが正常終了した場合にも終了します。/CHE FREEZE|PURGE|DUMPQ LEAVEGR コマンドは、すべての類似性を無条件に終了します。シャットダウンが IMS (/CHE) によって開始されたか、セッション (/PSTOP、/CLOSE、または /QUIESCE) によって開始されたかに関係なく、同一 IMS システム間のいずれかの並列 MSC セッションまたは ISC セッションがアクティブのままか、あるいは終了されたが、まだ有効な状況を持っている場合、類似性は維持されます。

同一 IMS システム間 MSC と ISC を両方使用している並列セッションがある場合には、前述の類似性規則も同時に適用されます。つまり、いずれかのリンクまたはノードが有効状況を持つ場合、MSC および ISC 両方の並列セッションに対し類似性が維持されます。

すべての場合で IMS が類似性を終了するためには、VTAM ACB を /START DC コマンドで開く必要があります。

MSC の有効状況 (MSC リカバリー・モードは、常にローカル)

- リンクがアクティブ状況にある。
- リンクがエラー状況にある。

ISC の有効状況

- ノードがアクティブ状況にある。
- ノードが RM ローカル・リカバリー・モード (ノードに対し SRMDEF=Local を設定) を使用しており、ウォーム終了状況である。

IMS システムおよび特定のノードまたは MSC リンクとの類似性に関するリストを表示するには、/DISPLAY AFFIN コマンドを使用します。

関連資料: /DISPLAY AFFIN NODE または /DISPLAY AFFIN LINK コマンドについて詳しくは、「IMS V15 コマンド 第 1 巻: IMS コマンド A-M」を参照してください。

持続している類似性を終了する方法については、『持続している類似性の終了』を参照してください。

持続している類似性の終了

端末は、前のセッションから、特定の IMS との類似性が持続していると、セッション平衡化に参加できません。VTAMが VGR 類似性を管理している場合、類似性はセッション終了時に自動的にリセットされます。

IMS が VGR 類似性を管理している場合、以下の状態のそれぞれにおけるログオフまたは IMS 障害の後でも類似性は存続します。

- 端末は LOCAL 状況リカバリー・モードによって静的で、エンド・ユーザー有効状況 (会話、STSN、高速機能、全機能応答モード) を持っています。

- 端末は、LOCAL 状況リカバリー・モードおよび RCVYSTSN=YES を伴った、ETO SLUP または 3600 金融機関端末です。
- 端末は、静止していない、LOCAL 状況リカバリー・モードを伴った ISC 並列セッション端末です。
- 端末は ISC 並列セッション端末で、1 つ以上の並列セッションが静止していません。
- MSC リンクがエラー状況にあります。

以下のいずれかの方法を使用すれば、類似性を終了することができます。

- LEAVEGR キーワードを指定して /CHECKPOINT コマンドを入力する。
- ログオフ出口ルーチンまたはサインオフ出口ルーチンを使用して端末状況をリセットする。
- IMS サブシステムをコールド・スタートし、IMS の再始動時に GRSNAME を指定する。

関連資料:

- /CHECKPOINT コマンドおよび LEAVEGR キーワードについては、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。
- ログオフ出口ルーチンおよびサインオフ出口ルーチンについては、「IMS V15 出口ルーチン」を参照してください。

汎用リソース・グループからの IMS システムの除去

IMS を汎用リソース・グループから除去するために、/STOP VGRS コマンドを使用することができます。同様に、IMS を汎用リソース・グループに戻すには、/START VGRS コマンドを使用することができます。

端末状況のリセット

コマンド、出口ルーチン、またはパラメーターを使用して、端末状況をリセット (したがって類似性を終了) することができます。

以下のリストに、端末状況をリセットできる方法を示します。

- 許可コマンド
- ログオフ (DFSLGFX0) 出口ルーチン
- サインオフ (DFSSGFX0) 出口ルーチン
- RCVYFP、RCVYRESP、RCVYSTSN、および RCVYCONV パラメーター
- NONE 状況リカバリー・モード

ログオフ (DFSLGFX0) 出口ルーチンは、ログオフ時に端末の状況をリセットします。同様に、サインオフ (DFSSGFX0) 出口ルーチンを使用すると、サインオフ時にユーザーの状況をリセットすることができます。端末またはユーザーの状況をリセットすると、IMS は端末との類似性を終了するので、その端末は再度ログオンして、セッション平衡化に参加できるようになります。

関連資料:

- ログオフ出口ルーチンおよびサインオフ出口ルーチンの詳細については、「IMS V15 出口ルーチン」を参照してください。
- NONE 状況リカバリー・モードの詳細については、「IMS V15 コミュニケーションおよびコネクション」の『エンド・ユーザー有効状況に対する状況リカバリー・モード』を参照してください。
- RCVYFP、RCVYRESP、RCVYSTSN、および RCVYCONV パラメーターの詳細については、「IMS V15 システム定義」を参照してください。

類似性が持続している IMS 障害後のログオン

共用キュー環境で操作していて、ユーザーの端末が汎用リソース・グループ内の IMS にログオンしているとします。このIMS に障害が発生した場合、IMS とユーザーの端末の間の類似性が持続する可能性があります。

その結果、その端末上にユーザーを待つ出力メッセージが出されている可能性があります。以下の条件のすべてを満たせば、ユーザーのメッセージを入手するために、汎用リソース・グループ内の別の IMS にログオンすることができます。

- 汎用リソース・グループ内の別の IMS の APPLID 名を使用してログオンする。
- ユーザーの出力メッセージが、障害が発生した IMS 上でロックされていない。また、それらのメッセージが IMS 会話に対する応答または高速機能応答モード・トランザクションに対する応答のどちらでもない。
- リソース構造およびResource Managerを使用している場合、状況リカバリー・モードは LOCAL ではなく、障害が発生した IMS (RM 類似性なし) 上にはエンド・ユーザー有効状況は存在しません。

VTAM 汎用リソース・メンバー選択の制御

端末のログオン時に、端末のための汎用リソース・メンバーの VTAM による選択を制御することができます。選択処理を制御するには、VTAM 汎用リソース解決出口ルーチン (ISTEXCGR) または z/OS ワークロード・マネージャーのいずれかを使用します。

どちらの機能を使用するかは、ユーザーのアプリケーション・プログラムの要件およびインストール・システムの要件によって決まります。

VTAM が汎用リソース・メンバーを選択するために使用する基準は、次のとおりです

1. 既存の類似性。VTAM はカップリング・ファシリティ上での ISTGENERIC と呼ばれるリスト構造に類似性テーブルを保持します。
2. VTAM 汎用リソース解決出口ルーチン (ISTEXCGR)
3. z/OS ワークロード管理
4. 現行セッション・カウント

関連資料:

- VTAM 汎用リソース解決出口ルーチン (ISTEXCGR) について詳しくは、「z/OS Communications Server: SNA Customization」を参照してください。

- z/OS ワークロード管理について詳しくは、「z/OS MVS 計画: ワークロード管理」および「z/OS MVS プログラミング: ワークロード管理サービス」を参照してください。

IMSplex 間の整合性の保証

汎用リソース・グループでは、定義、オペレーターのコマンドとプロシージャー、および出口ルーチンにおける不整合によって問題が生じる可能性があります。

- 不整合によって、少なくとも、ログオンごとに異なる IMS システムとセッションしていることを認識しない端末ユーザーが混乱する可能性があります。
- 最悪の場合には、不整合によって処理が破壊される可能性があります。

制約事項: IMS と別のノード間の LU6.1 並列セッションは、すべてが VGR または非 VGR を使用する必要があります。LU6.1 パートナー間で確立された最初の並列セッションにより、実際の APPLID 名 (非 VGR) または GRSNAME (VGR) のどちらかによって、パートナー・ノードがどのように認識されるかが設定されるため、LU6.1 並列の VGR セッションおよび非 VGR セッション (ISC または MSC) を混合させることはできません。この条件は VTAM により記憶され、VTAM は後続の並列セッション開始時にフラグ (ISTNR IPL パラメーター領域内の INRIFLG0 = NIBNNAMS) を設定して、この名前の関連を示します。IMS はこのフラグを使用してセッションを確立します。不一致があると、セッション開始が失敗 (DFS3645 ログオン拒否など) したり、または IMS によるセッション類似性の管理が正しくなくなる場合があります。

IMS1 から VGR を使用する IMS2 へ、IMS1 から非 VGR を使用する CICS へ、または IMS1 から非 VGR を使用する IMS3 への同時利用など、非並列 LU6.1 セッション間では VGR および非 VGR を混合させることができます。

推奨事項: 汎用リソース・グループでの整合性を保証するには、以下のようにしてください。

- メンバーの IMS システムを定義する時は、等価の仕様を使用します。

例: ある IMS が ETO の組み込み (IMSCTRL マクロ上で ETOFEAT=YES) を指定する場合、汎用リソース・グループ内の他のすべての IMS システムも ETO の組み込みを指定する必要があります。

- 汎用リソース・メンバー間で実行パラメーターを整合して指定します。

例: 汎用リソース・グループ内のそれぞれの IMS システムが、ETO を使用可能にすること、または ETO を使用可能にしないことを指定しなければなりません。

- ETO を組み込む場合には、各 IMS 汎用リソース・グループ・メンバーの ETO 記述子に同じ情報を指定します。
- 汎用リソース・グループ内で LTERM とトランザクションに関して同じ命名規則を使用します。同じ LTERM 名を別の IMS システムの別のノードに割り当てると、予測できない結果が生じる可能性があります。

関連情報: 名前の一意性およびリソース・タイプの整合性を保証する方法について詳しくは、367 ページの『第 18 章 IMSplex 内のトランザクション・マネージャーのリソースの計画』を参照してください。

- ・ 汎用リソース・メンバー間ですべての端末定義の整合性を保持します。

例: IMS システム定義時に、LTERM 名を、汎用リソース・グループ内のすべての IMS システム上の同じ物理端末に対して整合して定義する必要があります。

- ・ 汎用リソース・グループ内の MSC リンクはクローンする必要があります。

例: 汎用リソース・グループは、同一の MSPLINK、MSLINK、および MSNAME 特性を持つ必要があります。

- ・ 汎用リソース・グループ内の各 IMS 上のマスター端末オペレーター (MTO) が、発行するコマンドに関して他の MTO に通知するためのプロシージャーを持っているようにします。単一点制御点 (SPOC) を使用している場合、コマンドは、そのコマンドを処理することができるすべての IMS システムに行くので、通知プロシージャーの必要はありません。

例: Resource Manager (RM) を使用しておらず、汎用リソース・グループ内のある IMS 上のユーザーを停止するために、ある MTO が /STOP コマンドを実行する場合、そのユーザーは /STOP が入力された IMS 上でのみ停止されます。ただし、RM およびリソース構造を持っている場合、RM はユーザーをグローバルに管理し、そのユーザーは IMSplex 内のすべての IMS システム上で停止されます。

関連資料: SPOC の使用について詳しくは、「IMS V15 オペレーションおよびオートメーション」の『TSO SPOC アプリケーションによる IMS の制御』を参照してください。

- ・ 汎用リソース・グループ間で出口ルーチン (特に、宛先作成出口ルーチン、OTMA 経路指定出口ルーチン、ログオンとサインオンの出口ルーチン、およびログオフとサインオフの出口ルーチン) を機能的に同様に作成します。

IMS ESTAE プロセスにおける類似性管理のバイパス

IMS を使用して汎用リソース類似性を管理する場合には、IMS が IMS ESTAE 出口で VTAM 汎用リソース論理を使用するか、使用しないか、またはバイパスするのかを制御することができます。IMS ESTAE プロセスを使用する場合は、DFSDCxxx IMS.PROCLIB メンバー・データ・セット内で端末セッションについては GRESTAE=Y、MSC リンク・セッションについては GRMESTAE=Y を指定します。

GRESTAE=Y または GRMESTAE=Y は、ACF/VTAM ACB のクローズ前に状況が存続していないノードの類似性を削除するために、IMS が既存の ESTAE 論理に従う必要があることを示します。GRESTAE=N または GRMESTAE=N は、IMS が ACF/VTAM ACB を即時にクローズして IMS 終了を促進し、すべてのノードの類似性を設定したままにすることを示します。

IMS ESTAE 出口の IMS VTAM 汎用リソース論理は、端末または MSC リンク状況が存続していなければ、汎用リソース類似性を削除しようとします。IMS ESTAE ルーチンは、VTAM 類似性削除強制オプション (ENDAFFNF) を使用します。このオプションでは、セッションを VTAM 端末から事前に切断する必要があります。

以下の表では、類似性が VTAM 管理あるいは IMS 管理のいずれであるか、およびどちらの GRESTAE オプションが DFSDCxxx IMS.PROCLIB メンバー・データ・セット内で指定されているかを基にして汎用リソース類似性を管理する方法を要約しています。

表 25. 汎用リソース類似性管理および GRESTAE オプション

類似性管理	GRESTAE/ GRMESTAE オ プション	汎用リソース類似性管理
IMS	Y	IMS が、ESTAE 処理期間を含め、汎用リソース類似性を管理します。 IMS は、IMS 障害時に発生する、ESTAE プロセス中に類似性をリセットします。
IMS	N	IMS が、ESTAE プロセス時を除いて汎用リソース類似性を管理します。 IMS は、IMS 障害時に発生する、ESTAE プロセスにおける類似性をリセットしません。
VTAM	Y または N	VTAM が、汎用リソース類似性を管理します。

第 18 章 IMSplex 内のトランザクション・マネージャーのリソースの計画

このトピックでは、IMSplex 内の IMS トランザクション・マネージャーのリソース管理に関連するリソース構造および Resource Manager (RM) の機能について詳しく説明します。

リソース構造および RM は、DB/DC 環境または DCCTL 環境内で、CSL 操作を使用して実行されることを意図しています。

関連資料:

- CSL による IMSplex の管理について詳しくは、165 ページの『第 7 章 CSL の管理』を参照してください。
- PROCLIB データ・セット の DFSDCxxx メンバーのリソース構造関連パラメーターの詳細については、「IMS V15 システム定義」を参照してください。
- ユーザー出口ルーチン DFSINSX0、DFSSGNX0、DFSINTX0、および DFSLGNX0 については、「IMS V15 出口ルーチン」を参照してください。
- リソース構造および RM に関連するコマンドのリストについては、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

リソース名の一意性

IMS は、リソース名がどの特定の時間においても IMSplex 内で必ず一回のみアクティブになることを保証します。IMSplex 内の IMS システムは、同じリソースを同時に活動化することはできません。Resource Manager (RM) がアクティブで、リソース構造がカップリング・ファシリティー内で定義されているときのみ、IMSplex はリソース名の一意性を自動的に制約します。

リソースがアクティブのとき、そのリソースは IMSplex 内の 1 つの IMS に所有されています。所有 IMS 上で非アクティブになるまでは、他の IMS が同じリソースを活動化することはできません。

名前の一意性は、VTAM LTERM、VTAM 単一セッション・ノード、ユーザー ID、およびユーザーに対してのみ強制されます。IMS では、BTAM 端末をサポートしていません。

名前の一意性は、SGN が M ではない場合のみ、ユーザー ID に対して制限されます。ユーザー ID 名の一意性の制限は自動ではありません。ただし、VTAM LTERM、VTAM 単一セッション・ノード、およびユーザーに対する名前の一意性は、RM およびリソース構造を定義している場合には、自動的に強制されます。

ISC 端末の場合、IMSplex 内のリソース共用を使用不可化するオプションが DFSINTX0 ユーザー出口ルーチンにある場合は、リソース名の一意性は静的 LU 6.1 セッションには強制されません。ISC TCP/IP 端末 (静的および動的) には、名前の一意性は強制されません。

リソース名の一意性の制約の使用不可化

DFSDCxxx PROCLIB メンバー・データ・セットで STM=NO と指定することによって、リソース名の制約、および一般には、TM リソース共有を使用不可に設定できます。

関連資料: IMS が特定のリソースに対して名前の一意性を強制する理由と、強制方法の詳細については、369 ページの『トランザクション・マネージャーのリソース』を参照してください。

リソース・タイプ整合性

リソース・タイプ整合性は、名前が、名前タイプと呼ばれるリソースのグループ内で固有であることを保証します。RM がアクティブでリソース構造がカップリング・ファシリティ内で定義されているとき、IMSplex はリソース・タイプの整合性を自動的に制約します。

IMS は、メッセージ宛先の名前タイプに対するリソース・タイプの整合性を強制します。IMS は、メッセージ宛先として定義されたリソースが IMSplex 全体の間で確実に整合性を持つようにします。

メッセージ宛先は以下のとおりです。

- LTERM
- APPC 記述子として定義された LTERM
- MSNAME
- IMS 定義のトランザクション
- CPI-C トランザクション

リソース・タイプ整合性の制約の使用不可化

リソース構造がある場合に、DFSDCxxx PROCLIB メンバー・データ・セット内で STM=NO と指定することによって、リソース・タイプ整合性の制約を使用不可にすることができます。STM=NO と指定すると、リソース・タイプ整合性は、IMS 定義のトランザクションおよび CPI-C トランザクションのみに強制されます。

関連資料: IMS が特定のリソースに対してタイプ整合性を強制する理由と、強制方法の詳細については、369 ページの『トランザクション・マネージャーのリソース』を参照してください。

グローバル呼び出し可能サービス

呼び出し可能サービスは、ユーザー提供の出口ルーチンがノード、LTERM、およびユーザーなどのリソースを検索することを可能にするために提供されます。呼び出し可能サービスはリソース構造内で共有されているグローバル・リソース情報を戻します。使用可能なグローバル情報が存在しない場合、ローカル情報がデフォルトで戻されます。

ローカル情報のみが必要な場合、DFSCCBLK マクロによって定義された CSCBLK DSECT 上の入力フラグ LOCAL を使用してください。

トランザクション・マネージャーのリソース

このトピックでは、以下のタイプの Transaction Manager (TM) リソースについて説明します。

TM リソース: APPC 記述子

IMS は、初期設定時または /STA LU62DESC の間に RM に対して APPC 記述子を定義し、メッセージ宛先に対するリソース・タイプ整合性を維持します。

IMS は、APPC 記述子名がトランザクション、MSNAME、または LTERM 名として使用されないことを保証します。記述子は、名前の一意性が強制されないので、複数の IMS システム上で定義することができます。

APPC 記述子の作成と削除

初期設定時または /STA LU62DESC の間は、IMS は、トランザクション、MSNAME、または、LTERM としてすでに定義された記述子をすべて無視し、警告メッセージを発行します。APPC 記述子は、記述子を定義したすべての IMS システムがウォームまたはコールドにかかわらず終了したときに RM から削除されます。記述子を定義しないシステムは終了する必要はありません。

TM リソース: VTAM LTERM

IMS は、メッセージの宛先のリソース・タイプ整合性、LTERM 名の一意性、および LTERM 状況のリカバリーのために、VTAM LTERM を RM に定義します。

LTERM 名の一意性

IMS は、VTAM LTERM 名に対する名前の一意性を強制し、IMSplex 内の出力セキュリティを保証します。ある LTERM がアクティブになった場合、IMS は、その LTERM が他のどの IMS 上でもアクティブにならず、他のどのユーザーまたは端末に対しても使用不可であることを保証します。

LU 6.1 セッションの場合は、DFSINTX0 ユーザー出口ルーチンの使用によって、LTERM 名の一意性を無視することができます。この出口ルーチンについて詳しくは、「IMS V15 出口ルーチン」を参照してください。

IMS では、BTAM 端末をサポートしていません。

LTERM の作成と削除

サインオン時またはログオン時に LTERM がアクティブになるとき、IMS はその LTERM を RM に定義します。名前がトランザクション、APPC 記述子、または MSNAME と同じ場合、あるいは LTERM が他の IMS 上でアクティブな場合、LTERM はアクティブにはなりません。動的ユーザーのサインオンは LTERM が割り当てられていなくても継続し (ただし LTERM が 1 つしか存在しない場合、サインオンはリジェクトされます)、静的端末へのログオンはリジェクトされます。IMS は、重要な状況に影響を与えるリカバリー可能コマンドが処理されるときにも LTERM を RM に定義します。

LTERM が非アクティブになり、重要な状況データが存在しない場合、IMS はその LTERM を削除します。LTERM が削除される状況には 3 つの場合があります。

- ログオフまたはサインオフ時
- 有効状況を削除するコマンドが処理されるとき
- IMS 障害後のリソース終結処理時で重要な状況が存在しないとき

TM リソース: MSNAME

リモート IMS システムを定義するため、および MSC ネットワーク内でリモートとローカルの IMS システム間の論理リンク・パスを定義するために、MSC ネットワークは MSNAME を使用します。MSNAME の場合は、Resource Manager (RM) は、メッセージ宛先の整合性のためにリソース・タイプ整合性の一貫性のみを強制しますが、リソース名の一意性は強制しません。

IMS を初期設定するとき、IMS は、RM にリモート SYSID を持つ各 MSNAME を定義します。リソース名の一意性が強制されないので、複数の IMS システムで同一の MSNAME を定義できます。

MSNAME TM リソースの作成と削除

MSNAME は、IMS 初期設定時に RM への TM リソースになります。RM はリソース構造で MSNAME を節約します。MSNAME は一度定義されるとリソース構造から削除されず、リソース構造自身が削除されない限りは再使用できません。

関連情報: MSNAME の追加情報および MSNAME の共用リソース・グループ内での使用方法については、「IMS V15 コミュニケーションおよびコネクション」を参照してください。

TM リソース: VTAM 端末ノード

IMS は RM にノードを定義して、単一セッション VTAM 端末に対する名前の一意性を強制し、ノード状況をリカバリーします。

ノード名の一意性

IMS は VTAM 端末の名前の一意性を強制して、一度にログオンできるのは 1 つの端末だけであることを保証し、データ保全性を保証します。これらの端末には、LU 6.1 (ISC) 並列セッションを除くすべての VTAM 端末が含まれます。

IMS は LU 6.1 並列セッションに対する名前の一意性を強制しませんが、データ保全性はユーザー名の一意性によって保証されます。LU 6.1 セッションの場合は、DFSINTX0 ユーザー出口ルーチンの使用によって、ノード名の一意性を無視することができます。この出口ルーチンについて詳しくは、「IMS V15 出口ルーチン」を参照してください。

注: IMSplex 内の複数のシステム上にログインしている端末へ非固有 ETO ノード名を割り当てるセッション・マネージャー・プロダクトは、RM とは正常に作業を行いません。IMS は、同じ非 ISC ノード名が複数のシステム内で一度にアクティブになることを防ぎます。

ノードの作成と削除

IMS は、ノードが端末ログオン時にアクティブになるときにそのノードを RM に定義します。ISC 並列セッションの場合を除いて、ノードが他の IMS システム上ですでにアクティブな場合、ログオンはリジェクトされます。IMS は、重要な状況を処理するリカバリー可能コマンドが処理されるときにもノードを RM に定義します。これらのコマンドには、グローバル MFSTEST、STOP、および TRACE を設定するコマンドが含まれます。

ノードが非アクティブになり、重要な状況データが存在しない場合、IMS はそのノードを削除します。ノードが削除される状況には 3 つの場合があります。

- ログオフ時
- 有効状況を削除するコマンドが処理される時
- IMS 障害後のリソース終結処理時で重要な状況が存在しないとき

TM リソース: トランザクション

IMS は、トランザクションを、タイプ 2 CREATE TRAN コマンドで動的に、またはアプリケーションが実行されるシステム内で静的に、あるいは APPC 会話により実行される CPI-C トランザクションとして動的に定義します。名前の一意性が制限されていないので、トランザクションは複数のシステムで同時に実行することができます。

TM リソース管理のコンテキスト内では、トランザクションは、他の TM リソースがトランザクション名を重複することの防止だけのためにリソースとして取り扱われます。たとえば DFSDCxxx PROCLIB メンバー・データ・セット内で STM=NO を指定して TM リソース共用を使用不可にする場合でも、RM はトランザクション名の重複防止を継続します。

トランザクション・リソース・タイプ整合性

IMS は、トランザクション名が MSNAME、APPC 記述子、または LTERM 名として使用されないことを保証します。名前が別のメッセージ宛先として定義されている場合、IMS は、その CPI-C トランザクションを無視します。その場合でも APPC は宛先を検証し、LTERM を宛先としては許可しません。

トランザクションの作成と削除

IMS を初期設定するとき、オンライン変更がトランザクションを追加するとき、またはトランザクションがタイプ 2 CREATE TRAN コマンドにより作成されるとき、IMS はトランザクションを RM に定義します。名前が別のリソースとして IMSplex 内ですでに定義されている場合、IMS はトランザクションを無視します。

IMS は、APPC 入力処理時に、名前が他のメッセージ宛先としてすでに定義されていないことを確認する検査を行った後に、CPI-C トランザクションを RM に示します。別のリソースとして定義されている場合、IMS は定義された宛先を使用し、エラーは生成されません。

IMS をコールド・スタートしても、トランザクションを削除しません。

TM リソース: ユーザー名

ユーザー名とユーザー ID は通常同じです。ただしユーザー出口と記述子はユーザー名をオーバーライドする場合があります。ユーザーとは、動的端末または並列セッション・サブプールにサインオンしたユーザーのことで、関連した作業および状況を持っています。ユーザー ID は、RACF などのセキュリティ製品によるセキュリティ許可のために端末にサインオンしたユーザーを識別します。

ユーザー名の一意性

IMS は、ユーザー名に対する名前の一意性を強制し、ユーザーのデータ保全性を保証します。ユーザーは、ユーザー・サインオン時または ISC セッション開始時に RM に定義されます。ユーザーがすでにアクティブな場合、IMS はサインオンまたはセッション開始をリジェクトします。

ISC セッションの場合は、DFSINTX0 ユーザー出口ルーチンの使用によって、ユーザー名の一意性を無視することができます。この出口ルーチンについて詳しくは、「IMS V15 出口ルーチン」を参照してください。

ユーザーの作成および削除

ユーザー・サインオン時にユーザーがアクティブになるとき、IMS はそのユーザーを RM に定義します。ユーザーが別のシステム上でアクティブな場合、サインオンはリジェクトされます。IMS は、重要な状況に影響を与えるリカバリー可能コマンドが処理されるときにもユーザーを RM に定義します。

ユーザーが非アクティブになり、重要なデータが存在しないとき、IMS は RM からユーザーを削除します。ユーザーが削除される状況には 3 つの場合があります。

- ユーザー・サインオフ時
- 有効状況を削除するコマンドが処理されるとき
- IMS 障害後のリソース終結処理時で重要な状況が存在しないとき

TM リソース: ユーザー ID

ユーザー ID は、RACF などのセキュリティ製品によるセキュリティ許可のために端末にサインオンしたユーザーを識別します。

通常、ユーザー ID とユーザー名は同一ですが、それらが同一ではない場合の例については、『TM リソース: ユーザー名』を参照してください。

ユーザー ID 名の一意性

ユーザー ID に対して名前の一意性を制限するかどうか決めることができます。デフォルトでは、ユーザー ID 名の一意性を制限します。1 つのユーザー ID で一度に複数の端末にサインオンできるようにする場合は (ただし、名前の一意性が強制されていない場合)、IMS 始動パラメーター SGN=M を使用してください。

ユーザー ID の作成および削除

ユーザーが端末にサインオンし、シングル・サインオン制約が要求されるとき、IMS はユーザー ID を RM に定義します。ユーザーがいったん IMSplex 内の VTAM 端末にサインオンすると、そのユーザーによるサインオン試行はリジェクト

されます。ユーザー ID が非アクティブになると、IMS はそのユーザー ID を RM から削除します。ユーザー ID が削除される状況には 2 つの場合があります。

- ユーザー・サインオフ時
- IMS 障害後のリソース終結処理時

RM およびリソース構造の IMS アクティビティーへの影響

このトピックでは、RM およびリソース構造がいくつかの IMS アクティビティーにどのように影響するかについて説明します。

会話

会話状況は IMSplex 内で共有されているので、その会話の ID は個別の動的ユーザーおよび静的ノードに対して一意的である必要があります。会話を一意的に識別するには、その会話を関連した入力 LTERM で限定してください。

会話状況を表示するには、ローカルおよび RM 会話情報に対する /DISPLAY CONVERSATION コマンドを使用してください。

初期設定

IMS は、RM およびリソース構造がアクティブのとき、始動パラメーター内のシングル・サインオン SGN=M の整合性を強制します。IMSplex 内で最初のアクティブな IMS は、IMSplex 全体に対する値を設定します。IMS が、異なる値をシングル・サインオンに使用して IMSplex に結合する場合、IMS は警告メッセージを発行し、IMSplex 値を使用します。すべての IMS システムが終了し、その後 IMSplex に再結合する (ウォーム・スタート) とき、この指定を変更することができます。/NRE または /ERE コマンドを使用して値をオーバーライドすることができます。

ウォーム・スタート時に RM がアクティブの場合、グローバル・サインオン値は、IMSplex に最初に結合する IMS の始動パラメーターから取得されます。RM がウォーム・スタート時に非アクティブの場合、サインオン値はチェックポイント・ログ・レコードから取得されます。

シャットダウン

IMS システムがシャットダウンするとき、システムが所有するすべてのリカバリー不能状況がリソース構造から削除されます。SCI は、IMS が正常シャットダウンまたは異常シャットダウンのいずれかによって終了することを、他の IMS システムに通知します。

IMS システムが異常シャットダウンした場合、IMSplex 内の他の IMS は終結処理 IMS になります。この IMS は、重要な状況を持たないリソースのリストを要求します。リソースが有効状況を持っていてリソース・リカバリー・モードが GLOBAL の場合、所有者はクリアされ、リソースはその他のシステムに対して使用可能になります。リカバリー・モードが LOCAL の場合、その障害システムに対する類似性が制限されます。

チェックポイント

端末およびユーザー状況は、アクティブの場合 RM によって保守されるので、ローカル DC 制御ブロックが非アクティブの場合は、IMS チェックポイント時に終結処理され削除されます。チェックポイントは GLOBAL 状況リカバリー・モードによりローカル・ブロック内のローカル状況を削除します。チェックポイントは LOCAL 状況リカバリー・モードによりローカル・ブロック内の状況を維持します。

これらのリカバリー・モードは、ウォーム・リスタートまたは緊急時再始動後の単純チェックポイント時に同じように扱われます。ただし、リソースが LOCAL 状況リカバリー・モードの場合、DFSSGNX0 および DFSLGNX0 は、障害のある IMS システムによって所有されるリソースを別の IMS が流用することを要求する場合があります。その場合、再始動する IMS は、再始動時にローカル状況を削除します。

第 19 章 IMS のセキュリティー

このトピックでは、IMS オンライン・システムのためのリソース・セキュリティーの設定に役立つ情報を提供し、保護できるリソースと、それらのリソースを保護するために使用可能な機能を識別し、設計に関する考慮事項を提供して、セキュリティーを活動化するのに必要な手順について説明します。

類似点と重複が数多くありますが、DB/DC 環境および DCCTL 環境のセキュリティーについては、DBCTL 環境のセキュリティーとは別に説明します。

関連概念:

174 ページの『ODBM とセキュリティー』

IMS でのデータ・セット暗号化のサポート

z/OS データ・セット暗号化は、z/OS 2.3 以上、および APAR OA50569 と従属 APAR がインストールされた後の z/OS 2.2 で使用できます。

z/OS データ・セット暗号化を使用して、DFSMS アクセス方式によってアクセスされるデータ・セットを暗号化することができます。データ・セットを、関連付けられた鍵ラベルを持つ SMS 管理の拡張フォーマット・データ・セットとして定義します。

データ・セットの鍵ラベルを指定するには、以下の方法を使用します。

- DFP RACF セグメントの **DATAKEY** パラメーターを使用して、鍵ラベルをデータ・セット名パターンに関連付ける SAF 規則を作成します。
- JCL、動的割り振り、または TSO ALLOCATE を使用して、鍵ラベルを指定します (**DSKEYLBL** パラメーター)。
- IDCAMS DEFINE コマンドで鍵ラベルを指定します (**KEYLABEL** パラメーター)。
- **DATACLAS** パラメーターと、それに関連する鍵ラベルを使用します。

既存のデータ・セットを、暗号化されるように鍵ラベルを使用して定義された新規の拡張フォーマット・データ・セットにコピーする必要があります。既存のデータ・セットは、単にその **DATACLAS** に付加された鍵ラベルがあること、または RACF 規則で鍵ラベルがデータ・セットに関連付けられていることだけを理由に暗号化されることはありません。

鍵ラベルへのアクセスは、データ・セットがオープンされたときに SAF アクセス規則を使用して検査されます。オープン操作が発生したアドレス・スペースのユーザー ID は、鍵ラベルの CSFKEYS クラスと突き合わせて検査されます。暗号化されたデータ・セットに対する読み取りおよび書き込みのために暗号鍵にアクセスできるようにするには、ユーザー ID が CSFKEYS クラスのリソース・キー・ラベルに対する READ 権限を持っている必要があります。

以下の表は、z/OS データ・セット暗号化をサポートするデータ・セットと、ユーザー ID がデータ・セットに関連付けられている鍵ラベルへのアクセス権限を必要とする IMS アドレス・スペースをリストしています。

注: 以下の表で、「ユーザー ID が鍵ラベルへのアクセス権限を必要とする IMS アドレス・スペース」の列は、すべてを示したものではありません。以下の表にあるデータ・セットの 1 つをオープンするすべてのプログラムおよびユーティリティーは、そのデータ・セットに関連付けられている鍵ラベルへのアクセス権限が必要です。

表 26. z/OS データ・セット暗号化をサポートする IMS データ・セット

データ・セットのタイプ	ユーザー ID が鍵ラベルへのアクセス権限を必要とする IMS アドレス・スペース
VSAM (HALDB、非 HALDB)	VSAM DB にアクセスする CTL、DLI、バッチ・ジョブ、およびユーティリティー
GSAM	IMS の BMP ジョブおよびバッチ・ジョブ
オンライン・ログ・データ・セット (DFSOLPnn、DFSOLSnn)	OLDS にアクセスする CTL (XRF 代替、FDBR 領域を含む)、ログ保存ユーティリティー、その他のユーティリティー、および RSR トランスポート・マネージャー
バッチ・ログ・データ・セット	バッチ・ログにアクセスする IMS バッチ・ジョブ、ユーティリティー、および RSR トランスポート・マネージャー
SLDS	SLDS にアクセスする CTL、ログ保存ユーティリティー、変更累積ユーティリティー、DB リカバリー・ユーティリティーとその他のユーティリティー、および RSR トランスポート・マネージャー
RLDS	ログ保存ユーティリティー、変更累積ユーティリティー、および DB リカバリー・ユーティリティー
変更累積データ・セット	変更累積ユーティリティーおよび DB リカバリー・ユーティリティー
イメージ・コピー・データ・セット	イメージ・コピー・ユーティリティーおよび DB リカバリー・ユーティリティー
CQS SRDS	CQS
IMS Connect レコーダー・トレース	IMS レコーダー・トレースを処理する IMS Connect および ユーティリティー
BPE トレース・データ・セット	BPE を使用するアドレス・スペース、BPE トレース・データを処理するユーティリティー (IPCS TSO ユーザーを含む)
高速機能トレース	従属領域
IMS 外部トレース・データ・セット	IMS 外部トレースを処理する CTL およびユーティリティー
z/OS ログ・ストリームのオフロード・データ・セットおよびステージング・データ・セット	z/OS ロガー・アドレス・スペース
IMS リポジトリ・データ・セット	リポジトリ・サーバー

表 26. z/OS データ・セット暗号化をサポートする IMS データ・セット (続き)

データ・セットのタイプ	ユーザー ID が鍵ラベルへのアクセス権限を必要とする IMS アドレス・スペース
RRDS	RRDS にアクセスする CTL およびユーティリティー
RECON データ・セット (RECON data set)	DBRC、DBRC を使用する IMS バッチ・ジョブ、および RECON データ・セットにアクセスするユーティリティーとツール
モニター・データ・セット	CTL、モニター・データ出力を処理するユーティリティー
CQS システム・チェックポイント・データ・セット	CQS
先行書き込みデータ・セット (WADS)	CTL (XRF 代替、FDBR 領域を含む)、ログ・リカバリー・ユーティリティー、および WADS にアクセスする他のユーティリティー

以下のデータ・セットは、標準外のアクセス方式を使用してアクセスされるか DFSMS が暗号化をサポートしていないために、暗号化できません。

- DEDB
- OSAM
- MSDB データ・セット (ダンプ、init、およびチェックポイントを含む)
- キュー・マネージャー・データ・セット (LGMSG、 SHMSG、 および QBLKS を含む)
- 再始動データ・セット (RDS)
- すべての PDS/PDSE タイプのデータ・セット (PSBLIB、 DBDLIB、 ACBLIB、 MODBLKS、 FMTLIB、 IMSTFMTx、 IMSDALIB、 プログラム・ライブラリー、 PROCLIB または構成データ・セット、 カタログ・ディレクトリー・データ・セット、 ステージング・データ・セット、 および BSDS を含む)
- スプール・データ・セット

バッチ・ログ・データ・セットの暗号化

新規バッチ・ジョブについては、いずれかの z/OS データ・セット暗号化方式を使用して、バッチ・ログ・データ・セットを鍵ラベルで定義します。実行が完了したバッチ・ジョブについては、IEBGENER ユーティリティーを使用して、作成されたログを元のログ・データ・セットから暗号化されたデータ・セットにコピーすることで、いずれのログでも暗号化できます。バッチ・ジョブの実行中は、暗号化されるバッチ・ログを動的に変更することはできません。

BPE トレース・データ・セットの暗号化

BPE トレース・レコードには通常は機密データは含まれていません。ただし、IMS バージョン 11 以降では、IMS Connect レコーダー・トレース (TYPE=RCTR) は BPE トレース機能を使用でき、レコーダー・トレースに機密データ (メッセージ)

が含まれることがあります。そのため、BPE トレース・データ・セットを IMS Connect レコーダー・トレースに使用できる場合は、それらのデータ・セットを暗号化することができます。

BPE トレース・データ・セットを暗号化するには、RACF 規則を使用するか、鍵ラベルが指定されている DATACLAS を使用します。BPE 構成 PROCLIB メンバー内の BPE EXTTRACE ステートメントは、DSKEYLBL パラメーターをサポートしません。

RACF 規則を使用する場合は、BPE トレース・データ・セット名を鍵ラベルに関連付ける規則を定義します。RACF が更新された後、新規に作成された BPE トレース・データ・セットは暗号化されます。

鍵ラベルが指定された DATACLAS を使用する場合は、以下のステップを実行します。

1. BPE 構成 PROCLIB メンバーを更新します。EXTTRACE ステートメントの **DATACLAS** パラメーターを、希望する鍵ラベルを持つデータ・クラスに変更または追加します。
2. 以下のコマンドを発行します。

```
F jobname,UPD TRTAB NAME(*) OPTION(REREAD)
```

これは BPE 構成 PROCLIB メンバーを再読み取りして、BPE トレース・データ・セットに関連付けられているデータ・クラスを更新します。BPE 外部トレースが現在アクティブである場合、現行データ・セットはクローズされ、更新された EXTTRACE パラメーターを使用して新規のデータ・セットがオープンされます。BPE 外部トレースがアクティブでない場合は、次回に BPE 外部トレースが使用されるときに新規の EXTTRACE パラメーターが使用されます。

変更累積データ・セットの暗号化

変更累積データ・セットに対して z/OS データ・セット暗号化を使用可能にすることができます。

DBRC 変更累積骨組み JCL を更新して、DSKEYLBL= によって鍵ラベルを指定するか、関連する変更累積データ・セットの鍵ラベルが付いた DATACLAS を指定します。あるいは、RACF を使用して、データ・セット名パターンによって鍵ラベルを変更累積データ・セットに関連付けます。DFSMS は、暗号化されるソート作業データ・セットをサポートしていません。

IMS Connect レコーダー・データ・セット (非 BPE トレース) の暗号化

IMS Connect レコーダー・データ・セットは、JCL 割り振り DD (IMS Connect 始動 JCL の HWSRCORDD) としてのみ指定できます。このデータ・セットを暗号化するには、鍵ラベルを使用して新規データ・セットを作成し、新規データ・セットを指すように IMS Connect JCL を更新し、IMS Connect を停止して再始動します。

CQS SRDS データ・セットの暗号化

暗号化された 2 つの新規 SRDS を現行の SRDS 以上のサイズで定義するには、IDCAMS を使用します。

ここでは、例として、元の SRDS の名前が SRDS1 および SRDS2 で、新しい SRDS の名前が SRDS1.NEW および SRDS2.NEW であるとします。

すべての CQS がダウンしている場合は、以下のステップを実行して CQS SRDS データ・セットを暗号化します。

1. IDCAMS REPRO を使用して、両方の SRDS を古いデータ・セットから新規の暗号化されたデータ・セットにコピーします。
2. コピーが正常に完了したら、以下のステップを実行します。
 - a. 古い SRDS を別の名前に (例えば、SRDS1 を SRDS1.OLD に、SRDS2 を SRDS2.OLD に) 変更します。
 - b. 新規の SRDS の名前を、元の SRDS データ・セット名に (SRDS1.NEW を SRDS1 に、SRDS2.NEW を SRDS2 に) 変更します。VSAM 基本データ・セット名と、それに関連付けられている .DATA データ・セットの両方 (例えば、SRDS1 と SRDS1.DATA) を名前変更する必要があります。
3. CQS を開始します。CQS が新規の SRDS に対する読み取りと書き込みを実行できることを検証するために、2 つの構造チェックポイントを取ります。

1 つ以上の CQS がアクティブである場合は、以下のステップを実行して、CQS SRDS データ・セットを暗号化します。

1. IDCAMS REPRO を使用して 1 つの SRDS (例えば、SRDS1) を古いデータ・セットから対応する新規の暗号化されたデータ・セットにコピーします。
2. コピーが正常に完了したら、以下のステップを実行します。
 - a. SRDS1 を SRDS1.OLD に名前変更します。
 - b. SRDS1.NEW を SRDS1 に名前変更します。VSAM 基本データ・セット名と、それに関連付けられている .DATA データ・セットの両方 (例えば、SRDS1 と SRDS1.DATA) を名前変更する必要があることに注意してください。
 - c. CQS が新規の暗号化されたデータ・セットにアクセスできることを確認するために、構造チェックポイントを取ります。新規の SRDS を CQS でオープンできない場合は、新旧の SRDS の名前を元の名前に変更し、2 つの構造チェックポイントを取ります。
3. 構造チェックポイントが成功した場合は、SRDS2 でステップ 1 とステップ 2 を繰り返します。アクティブな CQS を使用して SRDS のコピーと名前変更を行っている間は、決して CQS 構造チェックポイントを取らないでください。

IMS 外部トレース・データ・セットの暗号化

IMS 外部トレース・データ・セットに対して z/OS データ・セット暗号化を使用可能にすることができます。

通常、IMS 外部トレース・データ・セットには内部診断トレース・データが含まれており、お客様の機密データは含まれません。しかし、/DIAG コマンドを使用して IMS から内部ストレージおよび制御ブロックを取り込み、外部トレース・データ・

セットをターゲット出力として指定した場合は、機密データがトレース・データ・セットに取り込まれる可能性があります。

IMS 外部トレース・データ・セットの動的割り振りを使用している場合は、IMS の実行中に外部トレース・データ・セットを暗号化することができます。現在、IMS 外部トレースに書き込んでいないことを確認してください。次に、鍵ラベルを使用して、外部トレース・データ・セットの削除と再定義を行います。

JCL 割り振り外部トレース・データ・セット (制御領域開始 JCL 内の DD ステートメント DFSTRA01 および DFSTRA02) を使用している場合は、データ・セットの削除と暗号化済みとしての再作成を行うために IMS をシャットダウンします。

GSAM データベース・データ・セットの暗号化

既存と新規の GSAM データベース・データ・セットを暗号化するには、それぞれ異なる方法を使用します。

既存の GSAM データベース・データ・セットを暗号化する場合は、新規の暗号化されたデータ・セットを割り振って、元の GSAM データ・セットを新規データ・セットにコピーします。その後、新規の暗号化されたコピーを名前変更して使用します。

暗号化される新規の GSAM データベース・データ・セットを作成する場合は、そのデータ・セットを作成時に暗号化するように指定します。

イメージ・コピー・データ・セットの暗号化

イメージ・コピー・データ・セットに対して z/OS データ・セット暗号化を使用可能にすることができます。

DBRC イメージ・コピー骨組み JCL を更新して、DSKEYLBL= によって鍵ラベルを指定するか、関連するイメージ・コピー・データ・セットの鍵ラベルが付いた DATACLAS を指定します。あるいは、RACF を使用して、データ・セット名パターンによって鍵ラベルをイメージ・コピー・データ・セットに関連付けます。

オンライン・ログ・データ・セット (OLDS) の暗号化

変換中に IMS のシャットダウンと再始動を行える場合には、OLDS を暗号化できません。IMS がアクティブになっているときに IMS の停止と再始動を行わずに OLDS に対して暗号化を使用可能にする場合は、OLDS を一度に 1 つずつマイグレーションするか、または使用している OLDS の数が 50 以下であれば、暗号化されていない OLDS のセットを 1 つ、OLDS の新規セットにマイグレーションします。

変換中に IMS のシャットダウンと再始動を行える場合、OLDS の暗号化を使用可能にするには、以下のステップを実行します。OLDS を IMS 制御領域 JCL 内の DD ステートメントとして指定してある場合 (例えば、OLDS が動的に割り振られない場合) は、この方法を使用する必要があります。

1. 鍵ラベルを持つ暗号化された OLDS の新規セットを作成します (IMS が使用するセット内の既存の OLDS ごとに 1 つの OLDS)。新規の OLDS は、現行の OLDS と同じ属性を備えている必要があります。それらは拡張フォーマット・データ・セットでなければなりません。
2. OLDS 用の新規初期設定 (再初期設定) ボリュームのフォーマット設定 (システム定義)で説明されているように、新規の各 OLDS データ・セットを事前フォーマット設定します。これを行う 1 つの方法は、既存のフル OLDS を新規の暗号化された OLDS データ・セットの 1 つにコピーすることです。その後、新規に暗号化されたデータ・セットを他のすべての新規 OLDS にコピーします。これにより、新規 OLDS 内のすべてのブロックが確実に初期設定されます。
3. IMS がアクティブの場合は、/CHE FREEZE を発行して、IMS が必ず正常にシャットダウンされるようにしてください。すべてのログ保存ジョブが正常に完了していることを確認してください。
4. DSPURX00 ユーティリティーを使用して、すべての OLDS に対して次のコマンドを発行し、DBRC から PRIOLD/SECOLD 項目を除去します。

```
DELETE.LOG OLDS(DFSOLPxx) SSID(imsid)
```

IMS をシャットダウンした時にアクティブだった最後の OLDS に対して、次のように LASTCLOS パラメーターを指定します。

```
DELETE.LOG OLDS(DFSOLPxx) SSID(imsid) LASTCLOS
```

重複 OLDS が使用されている場合、このコマンドは DBRC から PRIOLD と SECOLD の両方の項目を除去します。

5. 元の OLDS データ・セットの名前をバックアップ名に変更し、新規の暗号化された OLDS の名前を元の OLDS データ・セット名に変更します。
6. IMS のウォーム・スタートを実行します (再始動に SLDS が使用されます)。
7. マイグレーションが完了し、すべてが正しく動作していることが確認されたら、古い OLDS データ・セットを削除します。

IMS がアクティブになっているときに IMS の停止と再始動を行わずに OLDS に対して暗号化を使用可能にする場合は、OLDS を一度に 1 つずつマイグレーションするか、または使用している OLDS の数が 50 以下であれば、暗号化されていない OLDS のセットを 1 つ、暗号化された OLDS の新規セットにマイグレーションします。どちらの方式でも、OLDS データ・セットに対して動的割り振りを使用する必要があります。これは、この手順で古いデータ・セットが削除または名前変更されるためです。制御領域に OLDS が割り振られている場合、それは失敗します。

IMS がアクティブである間に、一度に 1 つずつ OLDS をマイグレーションするには、以下のステップを実行します。

1. 鍵ラベルを持つ暗号化された OLDS の新規セットを作成します (IMS が使用するセット内の既存の OLDS ごとに 1 つの OLDS)。新規の OLDS は、現行の OLDS と同じ属性を備えている必要があります。それらは拡張フォーマット・データ・セットでなければなりません。
2. OLDS 用の新規初期設定 (再初期設定) ボリュームのフォーマット設定 (システム定義)で説明されているように、新規の各 OLDS データ・セットを事前フォーマット設定します。これを行う 1 つの方法は、既存のフル OLDS を新規に暗

号化された OLDS データ・セットの 1 つにコピーすることです。その後、新規の暗号化されたデータ・セットを他のすべての新規 OLDS にコピーします。これにより、新規 OLDS 内のすべてのブロックが確実に初期設定されます。

3. 現行の OLDS がいっぱいになって IMS が次の OLDS に切り替えるまで待つか、/SWI OLDS コマンドを発行して、強制的に即時切り替えを行います。その後、切り替え元の OLDS に対して、以下のステップを実行します。
 - a. 切り替え元の OLDS のログ保存が完了するまで待ちます。
 - b. 切り替え元の OLDS に対して、/STO OLDS nn コマンドを発行します。
 - c. 以下のいずれかのコマンド (DSPURX00 ユーティリティーまたはオンライン・コマンド) を使用して、PRIOLD/SECOLD 項目を削除することにより、DBRC から OLDS を削除します。

```
DELETE.LOG OLDS(DFSOLPxx) SSID(imsid)LASTCLOS または /RMD
DBRC='LOG OLDS(DFSOLPxx) SSID(imsid) LASTCLOS'
```

注: 重複 OLDS が使用されている場合、このコマンドは DBRC から PRIOLD と SECOLD の両方の項目を除去します。

- d. 古い OLDS データ・セットの名前を別の名前に変更します。OLDS 重複ロギングを使用している場合は、1 次と 2 次の両方の「切り替え元」OLDS でこれを行ってください。
 - e. 新規の暗号化された OLDS データ・セットの名前を古い OLDS データ・セット名に変更します。OLDS 重複ロギングを使用している場合は、1 次と 2 次の両方の OLDS でこれを行ってください。
 - f. /STAOLDS nn コマンドを発行して、暗号化された OLDS の使用を開始します。
4. すべての OLDS が暗号化された OLDS に変換されるまで、OLDS ごとにステップ 3 を繰り返します。
 5. マイグレーションが完了し、すべてが正しく動作していることが確認されたら、古い OLDS データ・セットを削除できます。

IMS がアクティブである間に、暗号化されていない OLDS のセットを暗号化された OLDS のセットにマイグレーションするには、以下のステップを実行します。

IMS には、使用できる OLDS が最大 100 個あり、それらの OLDS には 0 から 99 までの番号が付けられています。ある IMS サブシステムが現在 OLDS 0 から 20 を使用している場合は、それが元のセットと見なされます。ユーザーは、OLDS 21 から 41 の別のセットを「追加」セットとして定義できます。追加セット内の各 OLDS は、DFSVMxx IMS.PROCLIB メンバー内の OLDSDEF ステートメントで既に識別されている必要はありません。暗号化されていない元の OLDS と追加の OLDS のセットを暗号化された OLDS のセットにマイグレーションするには、以下のステップを実行します。

1. 鍵ラベルを使用して追加の OLDS の新規セットを作成します。新規の OLDS は、現行の OLDS と同じブロック・サイズにしてください。それらは拡張フォーマット・データ・セットでなければなりません。
2. OLDS 用の新規初期設定 (再初期設定) ボリュームのフォーマット設定 (システム定義)で説明されているように、新規の各 OLDS データ・セットを事前フォーマット設定します。これを行う 1 つの方法は、既存のフル OLDS を新規

の暗号化された OLDS データ・セットの 1 つにコピーすることです。その後、新規に暗号化されたデータ・セットを他のすべての新規 OLDS にコピーします。これにより、新規 OLDS 内のすべてのブロックが確実に初期設定されます。

3. 新規の追加 OLDS ごとに DFSMDA メンバーを作成します。重複 OLDS が使用されている場合は、各 2 次 OLDS の MDA メンバーも存在することを確認してください。OLDS 用の DFSMDA ステートメントの構文は、次のとおりです。

```
DFSMDA TYPE=OLDS,DSNAME=dsname,DDNAME=DFSOLxnn
```

DFSMDA ステートメントについて詳しくは、DFSMDA マクロ (システム定義)を参照してください。

4. 定義された追加の OLDS ごとに、/STA OLDS nn コマンドを発行します。
5. 新規に暗号化された「追加」の OLDS セットの最初のセットが IMS によって使用中になるまで、/SWI OLDS コマンドを繰り返し発行します。
6. 元の OLDS ごとに、/STO OLDS nn コマンドを発行します。
7. 以下のいずれかのコマンド (DSPURX00 ユーティリティーまたはオンライン・コマンド) を使用して、元の OLDS の各 PRIOLD/SECOLD 項目を DBRC から除去します。

```
DELETE.LOG OLDS(DFSOLPxx) SSID(imsid) または /RMD DBRC='LOG  
OLDS(DFSOLPxx) SSID(imsid)'.
```

以下のいずれかのコマンドを使用して、停止した最後の OLDS の LASTCLOS パラメーターを指定します。

```
DELETE.LOG OLDS(DFSOLPxx) SSID(imsid) LASTCLOS または /RMD  
DBRC='LOG OLDS(DFSOLPxx) SSID(imsid) LASTCLOS.
```

重複 OLDS が使用されている場合、このコマンドは DBRC から PRIOLD と SECOLD の両方の項目を除去します。

8. 削除または名前変更を行った後、KEYLABEL を使用して元の各 OLDS を再定義し、ステップ 2 のように事前フォーマット設定してください。
9. 元の OLDS ごとに、/STA OLDS nn コマンドを発行します。
10. すべての OLDS が暗号化されます。「追加」の OLDS セットを IMS による使用から除去する場合は、OLDS ごとに /STO OLDS x コマンドを発行できます。DBRC から削除しないでください。

SLDS/RLDS (IMS アーカイブ・ログ・データ・セット) の暗号化

SLDS/RLDS (IMS アーカイブ・ログ・データ・セット) に対して z/OS データ・セット暗号化を使用可能にすることができます。

DBRC ログ保存骨組み JCL を更新して、DSKEYLBL= によって鍵ラベルを指定するか、関連する SLDS/RLDS データ・セットの鍵ラベルが付いた DATACLAS を指定します。あるいは、RACF を使用して、データ・セット名パターンによって鍵ラベルをアーカイブ・ログ・データ・セットに関連付けます。新規に割り振られた SLDS および RLDS は暗号化されます。

全機能 VSAM データベース・データ・セット (非 HALDB または非 OLR 対応) の暗号化

全機能 VSAM データベース・データ・セット (非 HALDB または非 OLR 対応) に対して z/OS データ・セット暗号化を使用可能にすることができます。

以下のステップのほかに、別個のオンライン再編成製品を使用して、データベースをオフラインにせずにデータベース再編成を実行できる場合もあります。詳しくは、ご使用のオンライン再編成ツールの資料を参照してください。

1. 鍵ラベルを持つ新規のデータベース・データ・セットを事前割り振りします。
2. データベースをオフラインにします (例えば、/DBR DB, UPDATE DB STOP(ACCESS))。
3. 適切なアンロード・ユーティリティー (DFSURUL0 または DFSURGU0) を実行します。
- 4.

注: 接頭部の解決が必要な場合、実行されるソートでは、暗号化されていない一時作業データ・セットが作成されることに注意してください。DFSMS は、拡張フォーマットとしてのソート作業データ・セットをサポートしていません。適切な再ロード・ユーティリティー (DFSURRL0 または DFSURGL0) を実行して、データベースを暗号化されたデータ・セットに再ロードします。

5. 古いデータ・セットの名前をバックアップ名に変更します。
6. 新規データ・セットの名前を正しいデータベース・データ・セット名に変更します。
7. データベースをオンラインにします (例えば、/STA DB, UPDATE DB START(ACCESS))。

全機能 VSAM データベース・データ・セット (HALDB、OLR 対応) の暗号化

暗号化されていないデータ・セットを暗号化されたデータ・セットに再編成するには、HALDB オンライン再編成を使用します。非 HALDB VSAM データベース・データ・セットに対してオフライン・プロセスを使用することもできます。

1. 鍵ラベルを持つ新規の OLR ターゲット・データベース・データ・セットを事前割り振りします。
2. 1 つ以上の区画に対して IMS オンライン再編成を実行します。すべての区画が再編成されるまで、繰り返してください。

オフライン・プロセスを使用して z/OS データ・セット暗号化を有効にする方法については、全機能 VSAM データベース・データ・セット (非 HALDB または非 OLR 対応) の暗号化 (システム管理) を参照してください。

z/OS ログ・ストリームのオフロード・データ・セットおよびステージング・データ・セットの暗号化

ロガー・データ・セットを鍵ラベルに関連付けるか、ロガー・データ・セットによって使用される DATACLAS に鍵ラベルを追加するには、RACF 規則を使用しま

す。z/OS ロガーによって新規のオフロード・データ・セットまたはステージング・データ・セットが次回作成されるときに、暗号化が有効になります。

先行書き込みデータ・セット (WADS) の暗号化

WADS を暗号化するには、WADS を拡張フォーマットとして定義し、鍵ラベルに関連付けます。XRF システムおよび FDBR システムの場合は、WADS の暗号化を使用可能にする前に、両方のシステムに APAR PI84947 を適用する必要があります。

IMS が稼働状態で残っているとき、または再始動時に、WADS を暗号化することができます。IMS は最大 10 個の WADS を使用でき、これらの WADS には 0 から 9 までの番号が付いています。

暗号化されていない WADS から暗号化された WADS にマイグレーションする場合、以下の条件が満たされているときは、マイグレーションの前に IMS のリサイクルを行ってください。

- 暗号化された WADS と暗号化されていない WADS に異なるデータ・セット名が使用される。
- WADS データ・セット名に動的割り振り (DFSMDA メンバー) が使用されない。つまり、制御領域 JCL に WADS 用の DD ステートメントがある。

これは、オンライン制御領域に WADS が割り振られていることになり、IMS が稼働状態で残っているときに WADS を削除できなくなるからです。

再始動時のマイグレーション・プロシージャ

1. IMS が正常に終了したことを確認します。
2. すべての WADS を削除します。
3. すべての WADS を拡張フォーマットとして定義し、鍵ラベルを指定します。
4. オプション: 暗号化された WADS に異なるデータ・セット名を使用する場合は、MDA メンバーを更新するか、制御領域 JCL 内の DD ステートメントを変更します。
5. FMT WA または FMT ALL を指定した /NRESTART コマンドを発行し、IMS をコールド・スタートします。

IMS が稼働状態で残っている場合のマイグレーション・プロシージャ

1. /DISPLAY OLDS コマンドを発行して、どの WADS が使用中であるかを確認します。
2. 使用中でない WADS に対して /STOP WADS コマンドを発行します。
3. 停止しているすべての WADS を削除し、それらを拡張フォーマットとして定義し、鍵ラベルを指定します。
4. オプション: 暗号化された WADS に異なるデータ・セット名を使用する場合は、MDA メンバーを更新します。
5. 削除され定義された WADS ごとに、/START WADS コマンドを発行します。WADS は、/START WADS コマンド処理の一環としてフォーマット設定されます。
6. /SWITCH WADS コマンドを出す。

7. 切り替え元の WADS に対して、/STOP WADS コマンドを発行します。重複 WADS を使用する場合は、このコマンドを切り替え元の WADS ごとに発行します。
8. 切り替え元の WADS を削除し、それらを拡張フォーマットとして定義し、鍵ラベルを指定します。
9. オプション: 暗号化された WADS に異なるデータ・セット名を使用する場合は、MDA メンバーを更新します。
10. 削除され定義された WADS に対して、/START WADS コマンドを発行します。

DB/DC および DCCTL セキュリティーの概要

セキュリティ保護を開始するときに、リソースのセキュリティに責任を持つ人と、正当にそれらのリソースへのアクセスを必要とするユーザーとの間で、要件のバランスをとる必要があります。

リソース・セキュリティを担当する人は、危険にさらされる可能性があるリソースについて責任を負っているため、保護手段を独占するような容易なアクセスを許可すべきではありません。しかし、割り当てられた作業を実行するユーザーは、リソースへの簡便なアクセスを必要とします。ユーザーとセキュリティの専門家は、リソース・アクセスの容易さとそのリソースを保護することの複雑さの間でバランスのとれた方法を立案しなければなりません。

IMS システムには、2 つのタイプのセキュリティが存在します。管理者は、次の 1 つまたは両方に対処できます。

- ユーザーがアクセスできる種類のリソースを保護する。例えば、あるユーザーは、「部品」データベースにはアクセスできるが、「顧客注文」データベースへはアクセスできないなどの場合が考えられます。
- ユーザーがリソースへアクセスした後、そのリソースに対してユーザーができることについて保護を行う。例えば、あるユーザーは、ファイルの読み取りはできるが更新はできないという場合が考えられます。

保護できる DB/DC および DCCTL リソース

保護 IMS システムの設計のためにどのセキュリティ機能を使用するかを決める前に、システム内のどのリソースに保護が必要であるかを知っておかなければなりません。

言い換えれば、どのように保護するかを決める前に、何を保護するかを決める必要があります。

DB/DC 環境および DCCTL 環境では、以下のリソースを保護できます。

- IMS 制御領域
- IMS オンライン・システム
- IMS システム・データ・セット
- トランザクション
- コマンド
- プログラム仕様ブロック (PSB)

- オンライン・アプリケーション・プログラム
- データベース
- 従属領域
 - BMP (バッチ・メッセージ処理) 領域
 - IFP (IMS 高速機能) 領域
 - JBP (Java バッチ処理) 領域
 - JMP (Java メッセージ処理) 領域
 - MPP (メッセージ処理プログラム) 領域
- 端末
 - 論理端末 (LTERM)
 - 物理端末 (PTERM)
 - マスター端末

DB/DC および DCCTL システム定義時のセキュリティの定義

初期設定 EXEC パラメーターおよび 2 つのシステム定義マクロ (COMM および IMSGEN) を使用して、IMS セキュリティーの選択を行うことができます。これらの指定により、オンライン実行時にアクティブになるセキュリティのタイプを選択できます。

リソース・アクセス管理機能 (RACF) を使用して、どのリソースを保護したいかなど、その他のセキュリティを選択できます。IMS および DCC 始動プロシージャでセキュリティ・パラメーターを使用してセキュリティ選択項目を指定することもできます。

3 つのシステム定義マクロのどれにもセキュリティをまったく指定しない場合、IMS により、デフォルト・セキュリティと呼ばれる基本レベルのリソース・セキュリティが提供されます。これにより、以下を実現します。

- マスター端末以外の端末からの、特定のコマンドの入力が禁止されます。この基本セキュリティ機能は、IMS システム定義のステージ 2 の完了と同時にアクティブにされます。RACF を使用した入力アクセス・セキュリティを実装すると、IMS は、デフォルト・セキュリティの制限を除去します。静的端末の場合にサインオンが不要であれば、IMS は制御領域ユーザー ID を使用してコマンドの妥当性検査を行います。
- 静的に定義した端末だけに適用されます。ETO を使用して定義された端末は、同一レベルのデフォルト・セキュリティにより自動的に管理されます。コマンド許可出口ルーチン (DFSCCMD0) を変更して使用すると、IMS は、動的に定義された端末のデフォルト・セキュリティを除去します。

セキュリティの指定は、COMM マクロ、IMSGEN マクロ、または初期設定 EXEC パラメーターで行うことができます。セキュリティ関連の指定は、次の順序で階層的に受け入れられます。

1. DFSPBxxx PROCLIB メンバーに指定された初期設定 EXEC パラメーター
2. DFSDCxxx PROCLIB メンバーに指定された初期設定 EXEC パラメーター
3. COMM
4. IMSGEN

初期設定 EXEC パラメーターにコーディングされた指定は、COMM マクロおよび IMSGEN マクロにコーディングされたセキュリティ指定をオーバーライドしません。

DB/DC および DCCTL リソース用のセキュリティ機能

以下の表は、保護できるリソース、リソースの保護に使用できるセキュリティのタイプ、各タイプのセキュリティを実装するのに使用できるセキュリティ機能を要約しています。

使用するセキュリティを選択する際に、ご使用のシステムのセキュリティ標準と操作手順も考慮してください。

表 27. DB/DC および DCCTL リソースと、それらを保護するための機能：

リソース	セキュリティのタイプ	セキュリティ機能
IMS 制御領域および オンライン・システム	拡張リソース保護 (APPL リソース・クラスを使用)	RACF
システム・データ・ セット	z/OS パスワード保護	z/OS
	データ・セット保護 (VSAM)	RACF
データベース	セグメント・センシティブティ	PSBGEN RACF
	フィールド・センシティブティ	PSBGEN RACF
	パスワード・セキュリティ (/LOCK および /UNLOCK コマンド用)	RACF
PTERM	サインオン検査セキュリティ	出口ルーチン付きの RACF
	端末ユーザー・セキュリティ	RACF
	パスワード・セキュリティ (/LOCK および /UNLOCK コマンド用)	RACF
LTERM	パスワード・セキュリティ (/LOCK および /UNLOCK コマンド用)	RACF
	アプリケーション・グループ名セキュリティ	RACF
	リソース・アクセス・セキュリティ	RACF
ETO を使用して定 義された端末	サインオン検査セキュリティ	RACF および出口ルーチン
	入力アクセス・セキュリティ	RACF および出口ルーチン
LU 6.2 インバウン ド会話および IMS 管理アウトバウン ド会話	割り振り検証セキュリティ	RACF および出口ルーチン
	入力アクセス・セキュリティ	RACF および出口ルーチン
PSB	アプリケーション・グループ名セキュリティ	RACF
	リソース・アクセス・セキュリティ	RACF
	APSB SAF セキュリティ	RACF ¹

表 27. DB/DC および DCCTL リソースと、それらを保護するための機能 (続き):

リソース	セキュリティのタイプ	セキュリティ機能
トランザクション	アプリケーション・グループ名セキュリ ティ	RACF
	入力アクセス・セキュリティ	RACF
	リソース・アクセス・セキュリティ	RACF
	パスワード・セキュリティ (/LOCK および /UNLOCK コマンド用)	RACF
コマンド	デフォルト・セキュリティ	システム定義
	自動化操作プログラム (AO) コマンド用 のトランザクション・コマンド・セキュ リティー	RACF またはコマンド許可 のトランザクション・コマンド・セキュ リティー
	入力アクセス・セキュリティ	RACF
	タイプ 2 コマンド・セキュリティ	RACF
	DBRC コマンド許可 ²	RACF または出口ルーチン
タイプ 1 自動化操 作プログラム・イン ターフェース・アプ リケーション	トランザクション・コマンド・セキュ リティー	RACF またはコマンド許可 のトランザクション・コマンド・セキュ リティー
タイプ 2 自動化操 作プログラム・イン ターフェース・アプ リケーション	トランザクション・コマンド・セキュ リティー	RACF またはコマンド許可 のトランザクション・コマンド・セキュ リティー
オンライン・アプリ ケーション・プログ ラム	パスワード・セキュリティ (/LOCK および /UNLOCK コマンド用)	RACF
	拡張リソース保護 (APPL キーワードを 使用)	RACF
従属領域	アプリケーション・グループ名セキュ リティー	RACF
	APSB SAF セキュリティー	RACF
	リソース・アクセス・セキュリティ	RACF

注:

1. RACF を使用した APSB の保護は、CPI-C ドリブン・アプリケーションおよび ODBA アプリケーションにのみ適用されます。
2. DBRC コマンド許可は、DBRC コマンドにのみ適用される追加のコマンド・セキュリティー・オプションです。DBRC コマンドは、IMS システムでアクティブな、他のあらゆるコマンド・セキュリティー・オプションにも影響を受けます。

IMS DB/DC および DCCTL 用のセキュリティの設計

このトピックでは、さまざまなタイプの IMS セキュリティーの使用法を説明します。セキュリティ設計の個々の部分について考慮する場合、システムへのアクセスを獲得するためにユーザーが実行しなければならない物理処置を考慮してください。多くの場合、複数のタイプのセキュリティ検査が使用されます。

このトピックでは、セキュリティ用に次の制御点を想定しています。

- ユーザー識別
- マスター端末
- 自動化操作プログラム
- 領域の使用

このセクションでは、リソース・アクセス管理機能 (RACF) セキュリティー製品の使用も想定しています。これは、z/OS オペレーティング・システムと一緒にライセンス交付されます。RACF は IMS に対して外部です。

端末からのアクセスの制限

端末の制御点へのアクセスを制御するには、端末を使用するたびに、エンド・ユーザーに自分自身を識別させ、個々のユーザーが必要とするトランザクションとコマンドのセットを許可します。

静的端末、または SPQBname がノード名と同じである動的端末では、ユーザーがサインオンを許可されるのは、すべての会話が保留中の場合のみです。それ以外の場合は、ユーザーはアクティブな会話のトランザクションの使用を許可されません。

静的端末でアクティブな会話があり、ユーザーがそのトランザクションの使用を許可されていない場合、ユーザーはサインオンする前に /HOLD コマンドを入力して、すべての会話を保留状態にすることができます。そうするとユーザーはサインオンを許可されます。

SPQBname がノード名と同じである動的端末でアクティブな会話がある場合、アクティブな会話のトランザクションの使用を許可されたユーザーのみがサインオンを許可されます。動的端末の場合、サインオンの前に /HOLD コマンドを使用することはできません。

サインオン検査によるアクセス制御

サインオン検査セキュリティーでは、すべての端末または端末のサブセットで、/SIGN コマンドのパラメーターとしてユーザー ID の入力が必要です。

静的に定義された VTAM 端末に対してサインオン検査を実施するために、TYPE または TERMINAL システム定義マクロで OPTIONS= キーワードを SIGNON に設定してください。TERMINAL マクロでの指定は、TYPE マクロでの指定に優先します。TYPE マクロでの OPTIONS= のデフォルトは NOSIGNON です。

すべての静的端末にサインオンにしたい場合には、複数のシステム定義マクロに OPTIONS= キーワードを設定しないで済むように、DFSDCxxx PROCLIB メンバーに SIGNON=ALL を指定します。このためには、MTO、LU6.1、3284/3286、および SLU1 プリンター専用装置を除き、すべての端末にサインオンする必要があります。

サインオン検査は、RACF または出口ルーチン、あるいはその両方によって行われます。RACF を使用している場合、RACF は、ユーザー ID、パスワード、またはパスワード・フレーズ、およびグループについて検査します。出口ルーチンを使用してサインオン検査を実施している場合、出口ルーチンはユーザー ID とパスワードについて検査できます。

パスワードの検査を使用可能にするには、以下のようにします。





- 初期設定出口ルーチン (DFSINTX0) 上でパスワード検査バイトを設定する。
- DFS3656 メッセージに応答する。
- /SIGN ON VERIFY コマンドを発行する。

ユーザー ID とパスワードを検査するのに使用する出口ルーチンには、サインオン/オフ・セキュリティー出口ルーチン (DFSCSGN0) があります。ETO 端末については、サインオン出口ルーチン (DFSSGNX0) を使用することもできます。

DFSCSGN0 については、421 ページの『セキュリティー出口ルーチンの準備』で詳しく説明しています。

推奨事項: IMS システム定義 (DBC、DCC、または IMS のプロシージャーにおける PSWDC= パラメーター) で大/小文字混合パスワードのサポートを指定する場合は、パスワードを取り扱うすべての出口ルーチンも確実に大/小文字混合パスワードをサポートできるようにしてください。

関連資料:

-  /SIGN コマンド (コマンド)
-  IMS 環境で使用されるマクロ (システム定義)
-  IMS PROCLIB データ・セットの DFSDCxxx メンバー (システム定義)
-  トランザクション・マネージャー出口ルーチン (出口ルーチン)

RACF パスチケットの使用

RACF パスチケットは 1 回のみ使用できるパスワードであり、製品または機能からの要求により生成されます。パスチケットを RACF パスワードの代わりとして使用することができるため、ネットワーク経由で RACF パスワードを暗号化されていないテキストとして送信しなくてもすみます。

パスチケットにより、メインフレームのアプリケーション・ユーザー ID の認証を、RACF の代わりに、以下のいずれかに移行することも可能となります。

- ホスト・システム上で実行中の他の許可された機能
- ワークステーションのローカル・エリア・ネットワーク (LAN) 環境

パスチケットは、1 人のユーザーが特定のアプリケーションに 1 回のみ、およそ 10 分間しかアクセスできないため、再使用はできません。ほとんどのアプリケーションでは、特定のパスチケットを使用すると、その後 10 分間以内は同じユーザーが再び同じアプリケーションに対してこのパスチケットを使用することはできません。

IMS でパスワードではなくパスチケットが入った /SIGN ON コマンドを受け取った場合、アプリケーション名 (PTKTDATA プロファイル名) として IMSID を使用してパスチケットが作成されていないと、サインオン処理は失敗します。サインオン・コマンドを入力する他のシステムが IMSID を知らない場合もあるため、パスチケットを生成する際にアプリケーション名として他の名前の使用を認める柔軟性が与えられています。/SIGN ON コマンド内の新しいキーワード APPL により、エンド・ユーザーまたはプログラムは、パスチケットを作成する際に IMSID ではなく他の名前 (IMS VTAM アプリケーション名など) を指定することができます。

VTAM 汎用リソース (VGR) 環境下では、リモート・エンド・ユーザーは、どの IMS が接続に選択されるのかわかりません。DFSDCxxx PROCLIB メンバーは、総称グループ内のすべての IMS 用に、(IMSID の代わりとなる) システム全体のデフォルト・アプリケーション名を提供します。この名前をアプリケーション名の代わりに使用することにより、パスチケットの作成者は、IMS 端末セッションの受信側となる IMS システムがどれであるか知る必要がなくなります。

関連資料:

- パスチケットについて詳しくは、「z/OS Security Server RACF マクロおよびインターフェース」を参照してください。
- パスチケットを使用可能にする方法について詳しくは、「z/OS Security Server RACF セキュリティー管理者のガイド」を参照してください。

RACF パスチケットの作成と使用:

RACF の呼び出しを行うと、一般的なパスチケット生成アルゴリズムでは、特殊な入力情報を使用して固有のパスチケットを生成します。パスチケットは 8 文字の英数字ストリングであり、A から Z および 0 から 9 の英数字を含みます。

アルゴリズムでは、次の 4 つの入力値を使用してパスチケットを生成します。

- 1 つの RACF ホスト・ユーザー ID
- PTKTDATA クラスのプロファイル名 (アプリケーション ID とも呼ぶ)
- RACF 保護サインオン・アプリケーション・キー。このキーは PTKTDATA プロファイルに含まれます。
- ログオン機能を提供するアプリケーションからの必要な時刻および日付の入力データ

要件: RACF でパスチケットを正しく評価するためには、TOD クロックは現地時間ではなく GMT に設定しておくことが必要です。

トランザクションとコマンドの許可

このトピックでは、トランザクションとコマンドについて述べ、それらを許可する方法を説明します。

トランザクションの許可

トランザクション許可は、あるユーザー ID が特定のトランザクションの使用を許可されるかどうかを決めます。出口ルーチンまたは RACF、あるいはその両方を使用してトランザクション許可を実行することができます。インストール・システムのトランザクション許可方式として RACF を徐々に段階的に導入するには、トランザクション許可出口ルーチン (DFSCTRN0) を使用します。

このルーチンは、トランザクションが ETO 端末によって入力されていても、RACF によって保護されていない場合には、そのトランザクションをリジェクトできます。DFSCTRN0 と RACF の両方を使用した場合、DFSCTRN0 は、RACF がトランザクションを許可した後か、トランザクションが RACF に対して定義されていない場合にのみ、有効です。DFSCTRN0 については、421 ページの『セキュリティー出口ルーチンの準備』で説明します。

RACF に対して REVERIFY オプションを指定した場合、ユーザーは個々のトランザクション・コードと一緒にサインオン・パスワードを再入力しなければなりません。XRF 複合システムにおいてテークオーバーが発生した場合、REVERIFY はサポートされません。ユーザーは、クラス 1 端末を使用していない場合は、テークオーバー後に再度サインオンする必要があります。

DL/I 変更 (CHNG) 呼び出しを通して、または SPA 内のトランザクション・コードを変更することによってプログラム間通信を行っている場合は、RACF または出口ルーチン、またはその両方を使用してトランザクション許可を検査できます。トランザクション・コード状況が /SET、/LOCK、および /UNLOCK の各コマンドによって変更された場合にも、それと同じことが当てはまります。また、トランザクションと関連付けられているアプリケーション・プログラムによりデータベースが変更されると、その変更が特定のユーザーによって行われたことを識別するため、ユーザー ID が変更レコードと一緒に IMS システム・ログに記録されます。

IMS は、端末にサインオンしているオペレーターが現在コマンドまたはトランザクションを入力しているオペレーターであることを再検証することができるように、IMS プロシージャーに RVFY= パラメーターを提供しています。この再検証は、コマンドまたはトランザクション・プロファイルの APPLDATA フィールドに、'REVERIFY' という文字を入れることによって、RACF で行うことができます。以下に例を示します。

```
RDEFINE Txxx tran-name UACC(NONE) APPLDATA('REVERIFY')
```

トランザクションにパスワード保護が行われている場合は、ユーザーがこのトランザクション・コードを入力するたびに、IMS パスワードを入力するはずの場所に RACF パスワードを入力する必要があります。

コマンドの許可

コマンド許可出口ルーチン (DFSCCMD0) が IMS.SDFSRESL に組み込まれている場合、この出口ルーチンはコマンド許可検査を提供します。コマンド許可出口ルーチンは、RACF と一緒に機能させることもできますし、RACF を使用せずに単独で機能させることもできます。

コマンド許可出口ルーチンは、個々の IMS コマンドごとに呼び出されます。許可を行うために、デフォルト・セキュリティーまたは RACF が最初に呼び出されます。戻りコードはコマンド許可出口ルーチンへ渡されます。DFSCCMD0 は最終的な検証を行い、コマンド許可の成否を決定します。DFSCCMD0 を単独で使用して検証を行うこともできます。RACF が提供する以上に離散的なコマンド許可レベルを確立したい場合は、DFSCCMD0 の入力コマンド・バッファーを検査できます。DFSCCMD0 の入力コマンド・バッファーには、完全なコマンド・ストリームが入っています。

IMS.SDFSRESL に DFSCCMD0 が存在する場合、静的、ETO、および LU 6.2 を含むすべての装置タイプについて DFSCCMD0 が呼び出されます。

IMS は、端末にサインオンしているオペレーターが現在コマンドまたはトランザクションを入力しているオペレーターであることを再検証することができるように、IMS プロシージャーに RVFY= パラメーターを提供しています。この再検証は、コ

マンドまたはトランザクション・プロファイルの APPLDATA フィールドに、'REVERIFY' という文字を入れることによって、RACF で行うことができます。以下に例を示します。

```
RDEFINE Cxxx tran-name UACC(NONE) APPLDATA('REVERIFY')
```

トランザクションにパスワード保護が行われている場合は、ユーザーがこのトランザクション・コードを入力するたびに、IMS パスワードを入力するはずの場所に RACF パスワードを入力する必要があります。

/LOCK、/UNLOCK、および /SET コマンドを保護するパスワード

/LOCK、/UNLOCK、および /SET コマンドに対して、パスワード・セキュリティを実行して、LTERM、データベース、プログラム、およびトランザクションを保護できます。

以下のタスクを実行してパスワードを確認して、RACF または出口ルーチンを使用してパスワード・セキュリティを実行します。

1. 適切な RACF セキュリティー・クラス、すなわち、LIMS、PIMS、IIMS、または TIMS を使用して、保護するリソースを定義する。
2. リソースを定義する RACF セキュリティー・クラス内でリソースへのアクセスをもたせたいユーザーの ID を定義する。定義するユーザー ID は、/LOCK、/UNLOCK、または /SET コマンドが発行される端末へのサインオンに使用されるユーザー ID と同じものでなければなりません。
3. DFSDCxxx PROCLIB メンバーで、LOCKSEC=Y を指定する。
4. DBCTL のみの環境では、/LOCK コマンドはパスワード保護で保護されません。

RACF を使用した物理端末の保護

RACF は、特定の端末についてセキュリティを使用しないことから、事前定義済みユーザー・リストへ物理制御点を通じたアクセスを許可することまでを範囲とする端末ユーザー・セキュリティ機能を提供します。端末ユーザー・プロファイルは、IMS のすべての PTERM について作成できます。

以下の表は、端末ユーザー・プロファイルの例です。

表 28. 端末ユーザー・プロファイルの例：

User	物理端末				
	PTERMA	PTERMB	PTERMC	LTERMD	LTERME
USER 1	X				X
USER 2	X	X			
USER 3				X	X
USER 4					X
USER 5		X			X

ETO を通して定義されたユーザーは、サインオン検査を使用して IMS トランザクションまたはコマンドへのアクセスを取得することができます。動的端末のセキュリティは、RACF などのセキュリティ製品を通して、または出口ルーチンを通して定義しなければなりません。

RACF を使用したサインオン・パスワード

端末についてのサインオン検査セキュリティの実施に RACF が使用された場合、パスワードの検査は、サインオン検査出口ルーチンまたは RACF により、/SIGN コマンドの入力を使用して行われます。/SIGN ON コマンドで入力されたユーザー ID は、ユーザー・パスワードを必ず伴っている必要があり、他の省略可能なサインオン・データを伴うこともあります。RACF 再検証オプションが指定された場合、パスワードは、トランザクション・コードと一緒に再入力されたパスワードと比較できるように保管されます。


RACF を使用したサインオン・パスフレーズ

端末についてのサインオン検査セキュリティの実施に RACF が使用された場合、パスフレーズの検査は、サインオン検査出口ルーチンまたは RACF により、/SIGN コマンドの入力を使用して行われます。/SIGN PASSPHRASE コマンドまたは /SIGN PASSPHRASEQ コマンドで入力されたユーザー ID は、ユーザー・パスフレーズを必ず伴っている必要があり、他の省略可能なサインオン・データを伴うこともあります。RACF 再検証オプションは、パスフレーズではサポートされていません。

RACF パスワード保護

RACF パスワードはユーザーによって定義され、保守されます。RACF リソース・クラスがパスワードを使用して初期設定された後、ユーザーはその値を変更できます。サインオン検査が RACF でなく出口ルーチンによって提供された場合、ユーザー ID とパスワードのテーブルは、IMS 中核への別のバインドによって変更されなければなりません。ただし、ETO サインオン検査の場合は、ユーザー ID とパスワードのテーブルを IMS 中核への別のバインドによって変更する必要はありません。テーブルが出口ルーチン DFSINTX0 によってロードされた場合、またはテーブルが出口ルーチン DFSSGNX0 の一部である場合には、テーブルを更新するために IMS を再始動する必要はありません。

関連資料:

 /SIGN コマンド (コマンド)

ネットワーク・セキュリティ資格情報の伝搬

分散環境でユーザーによって入力されたネットワーク・セキュリティ資格情報を、IMS トランザクションのエンドツーエンド処理と関連付けることができます。IMS DL/I インターフェースの ICAL 呼び出しによって開始された同期コールアウト要求で渡されたネットワーク・セキュリティ資格情報を、分散環境のアプリケーションがサポートできるようにすることもできます。

分散ネットワーク・セキュリティ資格情報には、ネットワーク・ユーザー ID とネットワーク・セッション ID を含めることができます。

ネットワーク・ユーザー ID

ユーザーの分散 ID。ネットワーク・ユーザー ID の最大長は 246 バイトです。IMS TM リソース・アダプターのユーザーの場合、ネットワーク・ユーザー ID は X.500 シリーズの標準の識別名 (DN) です。

ネットワーク・セッション ID

分散ユーザーのセッション ID。ネットワーク・セッション ID の最大長は 254 バイトです。IMS TM リソース・アダプターのユーザーの場合、ネットワーク・セッション ID はドメイン・ネーム、レルム、またはレジストリ名です。

ネットワーク・セキュリティー資格情報が、以下のいずれかのユーザー・メッセージ出口を使用するアプリケーションから IMS に渡されるようにすることができます。

- HWSSMPL0
- HWSSMPL1
- HWSJAVA0

制約事項: DataPower[®]、IMS Connect API、および SOAP Gateway クライアントからの分散ネットワーク・セキュリティー資格情報は、IMS Connect によってサポートされません。

OTMA への入力メッセージにネットワーク・セキュリティー資格情報が含まれている場合は、その資格情報が、ICAL 呼び出しによって開始される同期コールアウト要求で IMS によって伝搬されるようにすることができます。IMS コールアウト要求内のネットワーク・セキュリティー資格情報を、分散環境内で RESUME TPIPE 呼び出しを発行するアプリケーションがサポートできるようにすることができます。

IMS OTMA 呼び出し可能インターフェース (OTMA C/I) の `otma_send_receivev` API と `otma_send_asyncx` API を使用して、ネットワーク・ユーザー ID とネットワーク・セッション ID を IMS に渡すことができます。それぞれの API では、最大 100 バイトのネットワーク・ユーザー ID と最大 100 バイトのネットワーク・セッション ID を IMS に渡すことができます。

トランザクション許可出口ルーチン (DFSCTRN0) を使用して、OTMA メッセージ接頭部内のネットワーク・セキュリティー資格情報のアドレスを渡すこともできます。

以下の OTMA ユーザー出口ルーチン (これらは OTMA メッセージ接頭部のセキュリティー・データ・セクションのアドレスを含んでいます) を使用して、ネットワーク・セキュリティー資格情報が IMS に渡された場合に OTMA 入力メッセージに入っている、その資格情報にアクセスすることができます。

- DFSYIOE0
- DFSYPRX0
- DFSYDRU0

分散ネットワーク・セキュリティー資格情報は、OTMA メッセージ接頭語のセキュリティー・データ・セクションに入れられて IMS に渡されるため、メッセージ接頭語に関する情報が入っているすべての IMS ログ・レコード (ログ・レコード X'01' および X'03' など) に分散ネットワーク・セキュリティー資格情報が入っています。

高速機能メッセージにネットワーク・セキュリティー資格情報が含まれていて、そのメッセージがローカル IMS システム上で高速機能の急送メッセージ・ハンドラー (EMH) によって処理される場合、その資格情報は X'5901' ログ・レコードに記録されます。

ネットワーク・セキュリティー資格情報が含まれる高速機能メッセージがフロントエンド IMS システムで共用キュー環境の EMH キュー (EMHQ) を使用して処理される場合、その資格情報は X'5911' ログ・レコードに入れます。処理を実行する IMS システムであるバックエンド IMS システムでは、資格情報は X'5901' ログ・レコードに入れます。

ユーザー作成 IMS Connect クライアント・アプリケーションのネットワーク・セキュリティー資格情報の伝搬

HWSSMPL0 または HWSSMPL1 のユーザー・メッセージ出口ルーチンを使用するユーザー作成 IMS Connect クライアント・アプリケーションから IMS にネットワーク・セキュリティー資格情報が渡されるようにすることができます。ネットワーク・セキュリティー資格情報が IMS に渡された後、その資格情報は、IMS ログ・レコードで監査され、IMS トランザクションに関連付けられます。

ネットワーク・セキュリティー資格情報が IMS へのインバウンド・メッセージで渡される場合、ICAL 呼び出しによってコールアウト要求が開始されるのであれば、その資格情報を IMS によって同期コールアウト・メッセージで渡すこともできます。

IMS がコールアウト要求を発行し、ユーザー作成 IMS Connect クライアント・アプリケーションで RESUME TPIPE 呼び出しを発行したときに、そのアプリケーションでネットワーク・セキュリティー資格情報をサポートする必要がある場合は、資格情報をサポートするように RESUME TPIPE 呼び出しを定義する必要があります。

ヒント: HWSSMPL0 または HWSSMPL1 のユーザー・メッセージ出口ルーチンを使用するアプリケーションから IMS にネットワーク・セキュリティー資格情報を渡す場合は、IMS Connect レコーダー・トレース機能に対して BPE 外部トレース機能を使用可能にすることを検討してください。ユーザー作成 IMS Connect クライアント・アプリケーションからのネットワーク・セキュリティー資格情報が IMS に渡される場合、IMS Connect との間でやり取りされる入出力メッセージのサイズが 670 バイトを超える可能性があり、メッセージ全体のデータを取り込むために BPE 外部トレース機能が必要になります。

1. IRM 拡張を使用して、HWSSMPL0 または HWSSMPL1 のユーザー・メッセージ出口ルーチンを使用するアプリケーションから IMS Connect にネットワーク・セキュリティー資格情報を渡します。*NETUID* の ID を持つ IRM 拡張を使用して、ネットワーク・ユーザー ID を渡します。この ID は 1 から 246 バイトの長さにすることができます。*NETSID* の ID を持つ IRM 拡張を使用して、ネットワーク・セッション ID を渡します。この ID は 1 から 256 バイトの長さにすることができます。ネットワーク・セキュリティー資格情報が IMS Connect に渡された後、HWSSMPL0 または HWSSMPL1 のユーザー・メッセージ出口ルーチンは、資格情報を組み込むために OTMA メッセージ接頭部のセキュリティー・データ・セクションを作成します。

2. OTMA メッセージ接頭部に組み込まれるネットワーク・セキュリティ情報の DSECT である HWSECDNDS、および HWSECARDS を生成するには、HWSOMPFX マクロで次のオプションの両方を指定します。

DSECT=

OTMA ヘッダーの各セクション用に DSECT を個別に生成します。ただし、HWSECDNDS DSECT および HWSECARDS DSECT は生成されません。

NETSEC_OPT=YES

DSECT= オプションも指定した場合、HWSECDNDS DSECT および HWSECARDS DSECT が生成されます。

3. HWSOMPFX マクロを使用するコードを変更して、OTMA ヘッダーをマップします。ネットワーク・セキュリティ資格情報がセキュリティ・データ・セクションに含まれている場合、セキュリティ・セクションのサイズは異なることがあります、セキュリティ・セクションより下にあるフィールドの位置も変わり、アクセスできなくなる可能性があります。OTMA ヘッダー内のデータにアクセスできるようにヘッダーのセクションを検査したり、変更したりするには、以下のステップを実行します。
- OMCTLPFL フィールドの値を検査して、メッセージの中に状態データ、セキュリティ・データ、ユーザー・データ、およびアプリケーション・データのセクションが存在しているかどうかを判別します。
 - OTMA ヘッダーの特定のセクションにアクセスするには、OMCTLLEN、OMHDRLEN、OMSECLN、および OMUSRLN の各フィールドの値を使用して (存在する場合)、メッセージの制御データ、状態データ、セキュリティ・データ、およびユーザー・データの各セクションをそれぞれスキップオーバーします。
4. HWSSMPL0 または HWSSMPL1 のユーザー・メッセージ出口ルーチンを使用するユーザー作成 IMS Connect クライアント・アプリケーションが IMS コールアウト要求内のネットワーク・セキュリティ資格情報をサポートできるようにするには、IRM 接頭部の以下のフィールド指定を使用して RESUME TPIPE 呼び出しを定義します。以下のフィールド指定が定義されていない場合、IMS は、コールアウト要求の OTMA メッセージ接頭部のセキュリティ・データ・セクションから分散ネットワーク・セキュリティ資格情報を除去します。


IRM_ARCH

X'05' (IRM_ARCH5)


IRM_F6

X'80' (IRM_F6_NWSE)


関連概念:

 RESUME TPIPE/受信プロトコル (コミュニケーションおよびコネクション)

関連タスク:

 RESUME TPIPE による同期コールアウト要求のリトリブ (コミュニケーションおよびコネクション)

関連資料:

 IRM 拡張のフォーマット (コミュニケーションおよびコネクション)

☞ クライアントからの入力メッセージおよび出口に渡される入力メッセージ (コミュニケーションおよびコネクション)

☞ メッセージ出口から戻される入力メッセージ (コミュニケーションおよびコネクション)

☞ クライアント出口からの出力構造 (コミュニケーションおよびコネクション)

☞ IMS Connect で使用される OTMA 状態データ・フィールド (コミュニケーションおよびコネクション)

☞ IMS Connect で使用される OTMA セキュリティー・データ・フィールド (コミュニケーションおよびコネクション)

☞ OTMA セキュリティー・データ・フィールドの説明 (コミュニケーションおよびコネクション)

☞ メッセージ出口からクライアントへの出力メッセージ (コミュニケーションおよびコネクション)

☞ 返されるネットワーク・セキュリティ・セグメントのフォーマット (コミュニケーションおよびコネクション)

IMS TM リソース・アダプター・クライアント・アプリケーションのネットワーク・セキュリティ資格情報の伝搬

HWSJAVA0 ユーザー・メッセージ出口ルーチンを使用する IMS TM リソース・アダプター・クライアント・アプリケーションから IMS にネットワーク・セキュリティ資格情報が渡されるようにすることができます。ネットワーク・セキュリティ資格情報が IMS TM リソース・アダプターから IMS に送信された後、その資格情報は、IMS ログ・レコードで監査され、IMS トランザクションに関連付けられます。

ご使用の IMS TM リソース・アダプター・クライアント・アプリケーションについて、以下の前提条件が使用可能になっていることを確認してください。

- 以下のいずれかのアプリケーション・サーバー:
 - WebSphere Application Server バージョン 8.0 以降
 - WebSphere Liberty バージョン 8.5.5.9 以降
- コンテナ管理のセキュリティ。
- 許可ユーザーを含んでいる外部ユーザー・アカウント・レジストリー (LDAP サーバーなど)。

IMS 従属領域で稼働する IMS アプリケーションが外部 Java EE アプリケーションに対する同期または非同期のコールアウト要求を行ったときに、IMS TM リソース・アダプターがネットワーク・セキュリティ資格情報をサポートできるようにすることもできます。

IMS TM リソース・アダプターによって抽出されるネットワーク・セッション ID は、デフォルトでは、使用された認証サーバーの IP アドレスとポートです。ネットワーク・セッション ID の最大長は 254 バイトです。

IMS TM リソース・アダプターによって抽出されるネットワーク・ユーザー ID は、認証サーバー内に定義されている識別名の 1 つです。ネットワーク・ユーザー ID の最大長は 246 バイトです。

以下の手順は、ネットワーク・セキュリティー資格情報を IMS TM リソース・アダプター・クライアント・アプリケーションから IMS に渡すことができるようにするために実行する必要がある手順の概要を説明したものです。

1. IMS TM リソース・アダプターと一緒に提供される Java Authentication and Authorization Service (JAAS) ログイン・モジュールをインストールし、構成します。JAAS ログイン・モジュールは、WebSphere Application Server または WebSphere Liberty にインストールできます。
2. JAAS ログイン・モジュールを、ご使用の Java アプリケーションにリンクします。アプリケーションを JAAS ログイン・モジュールにリンクした後、ユーザーは、外部ユーザー・アカウント・レジストリーによる認証のために IMS トランザクションを呼び出すときに、セキュリティー資格情報を入力する必要があります。外部ユーザー・アカウント・レジストリーは、WebSphere Application Server または WebSphere Liberty でサポートされる任意のユーザー・アカウント・レジストリー (LDAP サーバーなど) にすることができます。資格情報の認証が正常に完了した後、IMS TM リソース・アダプターは OTMA メッセージ接頭部のセキュリティー・データ・セクションを使用して、IMS Connect に分散資格情報を送信します。
3. IMS から IMS コールアウト・メッセージ内で送信されたネットワーク・セキュリティー資格情報を、IMS TM リソース・アダプター・クライアント・アプリケーションで受信する必要がある場合は、アプリケーションがその資格情報をサポートできるようにしてください。
 - IMS TM リソース・アダプター・クライアント・アプリケーションが同期コールアウト・メッセージ内のネットワーク・セキュリティー資格情報をサポートできるようにするには、IMSActivationSpec オブジェクトの resumeTpPipeNsc プロパティを true に設定します。
 - IMS TM リソース・アダプター・クライアント・アプリケーションが非同期コールアウト・メッセージ内のネットワーク・セキュリティー資格情報をサポートできるようにするには、IMSInteractionSpec オブジェクトに対する setResumeTpPipeNSC(int resumeTpPipeNSC) メソッドを呼び出し、setResumeTpPipeNSC プロパティの値を 1 に設定する必要があります。setResumeTpPipeNSC プロパティに 1 を設定すると、IMS TM リソース・アダプターは、ネットワーク・セキュリティー資格情報をコールアウト・メッセージに含めるよう指示するために、IMS へ送信される OTMA メッセージ接頭部にフラグ・バイトを設定します。

関連情報:

ネットワーク・セキュリティー資格情報と IMS TM リソース・アダプター
分散ネットワーク・セキュリティー資格情報に対応した WebSphere Application Server の構成

分散ネットワーク・セキュリティー資格情報に対応した WebSphere Liberty の構成

コールアウト・メッセージ内のネットワーク・セキュリティー資格情報の IMS TM リソース・アダプターによるサポートの使用可能化

メッセージ駆動型 Bean のための IMSActivationSpec プロパティの構成
RESUME TPIPE ネットワーク・セキュリティ資格情報 (resumeTpipeNSC)

マスター端末のセキュリティに関する考慮事項

マスター端末からのアクセスのセキュリティは重要です。MTO は、通常操作時にすべてのセキュリティ・プロファイルを修正できるので、2 次レベルの制御を使用して端末を保護することを考慮しなければなりません。

サインオン検査セキュリティは、その機能を提供します。第一の疑問は、セキュリティを修正する能力をどれくらいこの 2 次レベルの制御に与えるべきかということです。

デフォルト・セキュリティは、マスター端末を介したシステムのセキュリティ・プロファイルの修正を防止せず、また、防止する能力もありません。ただし、一部のコマンドが MTO から入力できないように制限したい場合があります。

DFSCCMD0 出口ルーチンを使用して、MTO から入力できるコマンドを制限することができます。

DFSCCMD0 出口ルーチンを使用して、MTO から入力できるコマンドを指定します。保護すべきコマンドの候補として、/ASSIGN、/CHANGE、および /DELETE があります。コールド・スタートまたはウォーム・スタートの場合、MTO は次のセキュリティを制御できます。

- サインオン検査セキュリティ
- トランザクション許可
- トランザクション・コマンド・セキュリティ
- コマンド許可

以下の表に示された指定を行うことにより、これらの許可を制御できます。

表 29. 初期設定 EXEC パラメーター

強制されるセキュリティ・オプション	初期設定 EXEC パラメーター
サインオン検査	SGN
トランザクション許可	TRN
端末セキュリティ	SGN
コマンド・セキュリティ	RCF

AO アプリケーション・プログラムのセキュリティ

自動化操作プログラム (AO) は、IMS オペレーター・コマンドのサブセットを発行できます。オペレーター・コマンドはセキュリティ・プロファイルを変更する可能性があるため、管理者は AO アプリケーション・プログラムが IMS リソースを変更する可能性がある重要なコマンドを実行できないようにする必要があります。

AO アプリケーション・プログラムには 2 つのタイプがあります。タイプ 1 は、CMD 呼び出しを使用して、DB/DC 環境および DCCTL 環境でコマンドを出します。タイプ 2 は、ICMD 呼び出しを使用して、DB/DC、DCCTL、および DBCTL

環境でコマンドを出します。CMD 呼び出しを使用して発行されるコマンドと、ICMD 呼び出しを使用して発行されるコマンドでは、セキュリティーを実行するためのオプションが異なります。

CMD 呼び出しと ICMD 呼び出しの両方について、AO セキュリティーを強制するセキュリティー機能を指定し、システム定義マクロおよび始動プロシージャーを、セキュリティー機能選択と連動するように調整する必要があります。

関連資料: AO アプリケーションについて詳しくは、「IMS V15 オペレーションおよびオートメーション」を参照してください。

タイプ 1 AO コマンド・セキュリティー

RACF またはその他の外部セキュリティー製品、コマンド許可出口ルーチン (DFSCCMD0)、あるいはその両方のセキュリティー機能を使用すると、CMD 呼び出しを通じてタイプ 1 AO アプリケーション・プログラムによって発行されるコマンドを保護できます。

IMS および DCC 始動プロシージャーで AOI1= キーワードを使用して、セキュリティー機能の指定を行います。

タイプ 1 AO アプリケーション・プログラムのセキュリティーを実行するセキュリティー機能を指定するには、始動プロシージャーで AOI1= 実行パラメーターを使用します。AOI1= を使用した指定は、TRANSACT マクロでの指定をオーバーライドします。

IMS を初期化するたびに、AOI1= 値を変更できます。AOI1= の指定は、チェックポイント・レコードに含まれないためです。

関連資料: AOI1= パラメーターと、システム定義マクロで AOI= キーワードを使用して行った指定とのそれぞれの関係について詳しくは、「IMS V15 システム定義」を参照してください。

タイプ 1 AO コマンドを発行するトランザクションの定義:

タイプ 1 AO コマンドの発行を許可されたトランザクションを定義する必要があります。これは、TRANSACT マクロで AOI= パラメーターを使用することによって定義できます。

AOI= パラメーターを使用して、RACF または DFSCCMD0、あるいはその両方をセキュリティー機能として使用する場合にセキュリティー検査に使用されるユーザー ID も指定できます。以下の ID はいずれも、ユーザー ID として使用できます。

- トランザクションを発行する端末でサインオンしたユーザーのユーザー ID
- トランザクション・コード
- コマンド名の最初の 3 文字

関連資料: AOI= パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。

タイプ 2 AO コマンド・セキュリティ

RACF またはその他の外部セキュリティ製品、コマンド許可出口ルーチン (DFSCCMD0)、あるいはその両方のセキュリティ機能を使用すると、ICMD 呼び出しを通じてタイプ 2 AO アプリケーション・プログラムによって発行される AO コマンドを保護できます。

IMS および DCC 始動プロシージャで AOIS= パラメーターを使用して、セキュリティ機能の指定を行います。

タイプ 2 AO コマンド・セキュリティ検査に使用するセキュリティ機能を指定するには、始動プロシージャで AOIS= 実行パラメーターを使用します。

AOIS= 実行パラメーターを使用して、以下のオプションのいずれかを指定できます。

- RACF がセキュリティ検査を実行する。
- DFSCCMD0 がセキュリティ検査を実行する。
- RACF と DFSCCMD0 の両方がセキュリティ検査を実行する。
- アプリケーションからのすべての ICMD 呼び出しを使用不可にする。
- ICMD 呼び出しに対するすべてのセキュリティ検査を使用不可にする。

IMS を初期化するたびに、AOIS= 値を変更できます。AOIS= 指定は、チェックポイント・レコードに含まれないためです。

関連資料: AOIS= キーワードの指定について詳しくは、「IMS V15 システム定義」を参照してください。

タイプ 2 AO コマンドを発行するトランザクションのためのユーザー ID の定義:

RACF と DFSCCMD0 がセキュリティ検査用にユーザー ID として使用するものを指定するには、TRANSACT マクロで AOI= キーワードを使用します。

以下の ID はいずれも、ユーザー ID として使用できます。

- トランザクションを発行する端末でサインオンしたユーザーのユーザー ID
- トランザクション・コード
- コマンド名の最初の 3 文字

特定の状況では、トランザクションの発行に使用されるユーザー ID は、セキュリティ検査の実行時に使用できないことがあります。このような場合、IMS は、他の ID をユーザー ID として代用します。IMS が ID を代用する状況および IMS が代用ユーザー ID として使用するものについては、405 ページの『AO アプリケーション・プログラムのユーザー ID の代替』を参照してください。

タイプ 1 AO コマンドと異なり、タイプ 2 AO コマンドの発行というトランザクションの機能を使用不可にすることはできません。AOI=NO はタイプ 1 AO コマンドにのみ適用されるもので、トランザクションがタイプ 2 コマンドを発行できないようにするものではありません。タイプ 2 AO コマンドの場合、AOI=NO は AOI=YES と同じです。

AO コマンド・セキュリティとシステム定義

TRANSACT マクロまたは初期設定 EXEC パラメーターで、AO コマンド・セキュリティに影響する指定を行います。

TRANSACT マクロの AOI= キーワードは、以下の 2 つの機能を果たします。

1. どのトランザクションがタイプ 1 AO コマンドを発行できるかを指定できます
2. セキュリティ検査に RACF またはコマンド許可出口ルーチン (DFSCCMD0) を使用する場合、トランザクションがタイプ 1 およびタイプ 2 の AO アプリケーション・プログラムを呼び出すときに、何をユーザー ID として指定するかを指定できます

タイプ 1 AO コマンドの場合、TRANSACT マクロに行った指定は、始動プロシージャーに AOI1= キーワードを使用することによってオーバーライドできます。

AOI=NO は、ICMD 呼び出しを使用するタイプ 2 AO アプリケーション・プログラムでは無視されます。タイプ 2 AO アプリケーション・プログラムを通じてコマンドを出すというトランザクションの機能を使用不可にすることはできません。

タイプ 1 とタイプ 2 両方のアプリケーション・プログラムの場合、AO コマンド許可に使用できるように、RACF および DFSCCMD0 に以下のユーザー ID を指定できます。

- トランザクションを発行している端末にサインオンしたユーザー ID。これは、AOI=YES で指定します。ユーザー ID は、セキュリティ検査に使用できるとは限らないため、IMS は、LTERM 名、PSB 名、または他の ID をユーザー ID として代用できます。詳しくは、405 ページの『AO アプリケーション・プログラムのユーザー ID の代替』を参照してください。
- トランザクション・コード。これは、AOI=TRAN で指定します。
- コマンド名の最初の 3 文字。これは、AOI=CMD で指定します。

AO コマンド・セキュリティの実施時にオプションを増やすために、セキュリティ検査に関与する 3 つの要素、すなわち、ユーザー ID、トランザクション、およびコマンドのいずれかを、許可されるユーザーとして定義できます。

AO セキュリティの場合、CMD 呼び出しおよび ICMD 呼び出しを介して発行されるコマンドは通常、リソースとして定義され、サインオンしたユーザーまたはトランザクション・コードのどちらかがユーザーとして定義されます。ただし、代わりに、コマンド名を、許可されるユーザーとして定義し、トランザクション・コードを、保護されるリソースとして定義することも選択できます。ユーザーとして定義される要素を変更することは、例えば、各種トランザクションの数が多いために、そのトランザクションすべてを、あるリソースとして定義されたコマンドの RACF セキュリティ・クラスに追加することが不可能な場合に役立ちます。

関連資料: TRANSACT マクロおよび AOI= キーワード・パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。

AO アプリケーション・プログラムのユーザー ID の代替

特定の状況では、IMS は、LTERM 名またはプログラム名など他の ID をユーザー ID として使用します。このことを予想して、RACF に対して代替ユーザー ID を定義している場合を除き、RACF は CMD 呼び出しまたは ICMD 呼び出しをリジェクトします。

以下の例は、AO アプリケーション・プログラムの実行環境におけるさまざまな状況で IMS がユーザー ID を使用する方法を示しています。これらの例は、TRANSACT マクロの AOI=YES という指定に基づいています。

MPP または IFP

メッセージ GU 呼び出しが完了した場合、ユーザーが ICMD を使用してコマンドを出すことができるかどうかを判別するためにユーザー ID が使用されます。ユーザー ID は、トランザクションが発行されたサインオフ端末の LTERM 名か、サインオン端末のユーザー ID です。GU が実行されなかった場合は、PSB 名が使用されます。

BMP メッセージ GU 呼び出しが完了した場合、ユーザーが ICMD を使用してコマンドを出すことができるかどうかを判別するためにユーザー ID が使用されます。ユーザー ID は、トランザクションが発行されたサインオフ端末の LTERM 名か、サインオン端末のユーザー ID です。GU が実行されないか、BMP がメッセージ・ドリブンでない場合は、JCL JOB ステートメントで指定された USER パラメーターの値が使用されます。USER パラメーターが指定されていない場合は、0000000 というユーザー ID が使用されます。


DRA THREAD

スケジュール要求用に PAPL で渡されるセキュリティー・トークンを使用して、ユーザーが ICMD を使用してコマンドを出すことができるかどうかを判別されます。

BMP 領域は、/START コマンドによって、または JCL ジョブによって開始されます。/START コマンドを使用する場合は、カタログ式プロシージャーが入っているライブラリーを保護するために RACF を使用してください。JCL を使用した場合、RACF リソース・アクセス・セキュリティー (RAS) を使用できます。BMP 領域へのアクセスを制御するには、JOB ステートメント上で RACF に対して固有のユーザー ID を定義します。RAS を使用している場合は、RACF セキュリティー・クラスにユーザー ID を入力します。

AO アプリケーション・プログラムが CMD 呼び出しまたは ICMD 呼び出しによりコマンドを実行する場合、パスワードは使用されません。

関連資料:

 AO アプリケーションからサポートされる IMS タイプ 1 コマンド (コマンド)

RACF を使用した AO コマンド・セキュリティーの実施

RACF は、RACF セキュリティー・クラスに格納されているユーザー ID およびリソース・プロファイルに基づいてセキュリティーを実行します。ユーザーのユーザー ID が適切な RACF セキュリティー・クラス内のリソースに関連付けられている場合、RACF は、ユーザーがそのリソースにアクセスすることを許します。

タイプ 1 とタイプ 2 両方の AO アプリケーション・プログラム・セキュリティーでは、RACF リソース・クラスで作成したユーザー ID およびリソースの定義を、TRANSACT マクロ内の AOI= キーワード・パラメーターでの指定と調整する必要があります。

ご使用のアプリケーション・プログラムが実行中の従属領域のタイプやその他の状況に応じて、代替ユーザー ID を定義する必要があります。それらの状況については、405 ページの『AO アプリケーション・プログラムのユーザー ID の代替』を参照してください。

AO アプリケーション・プログラム・セキュリティーのためのRACF 定義の例

タイプ 1 とタイプ 2 両方の AO アプリケーション・プログラムのセキュリティーに RACF を使用する場合、RACF 定義のコーディングに次の例が役立ちます。例では、ユーザー IMSUSER1 およびトランザクション APOL13 を定義します。使用する RACF 定義は、TRANSACT マクロの AOI= キーワードを使用して行う指定により異なります。

以下の例では、AOI=YES が指定されており、コマンド /STOP は保護リソースであり、このコマンドに対してユーザー IMSUSER1 が許可されます。IMSUSER1 は IMS 端末にサインオンして、トランザクション APOL13 を入力します。トランザクションは、/STOP コマンドを発行するために、CMD または ICMD 呼び出しを実行します (タイプ 2 AO セキュリティーの場合、AOI=YES という指定は必要ありません)。

```
ADDUSER IMSUSER1 PASSWORD(IUSER1PW) DFLTGRP(SYS1)
RDEFINE CIMS (STO) UACC(NONE)
PERMIT STO CLASS(CIMS) ID(IMSUSER1) ACCESS(UPDATE)
```

以下の例では、AOI=TRAN が指定されており、/STOP が保護リソースで、このコマンドに対してトランザクション APOL13 が許可されます。トランザクション APOL13 には、どのユーザーも入ることができます。トランザクションは、/STOP コマンドを発行するために CMD または ICMD 呼び出しを実行します。APOL13 トランザクションは、/STOP コマンドの発行を許可されます。

```
ADDUSER APOL13 NOPASSWORD DFLTGRP(SYS1)
RDEFINE CIMS (STO) UACC(NONE)
PERMIT STO CLASS(CIMS) ID(APOL13) ACCESS(UPDATE)
```

以下の例では、AOI=CMD が指定されており、トランザクション APOL13 が保護リソースで、そのトランザクションについては、ユーザー IMSUSER1 およびコマンド /STOP が許可されます。許可は、2 回検査されます。IMSUSER1 が端末でトランザクションを入力したときと、トランザクションが /STOP コマンドを発行するために CMD または ICMD 呼び出しを実行したときです。

```
ADDUSER IMSUSER1 PASSWORD(IUSER1PW) DFLTGRP(SYS1)
ADDUSER STO NOPASSWORD DFLTGRP(SYS1)
RDEFINE TIMS (APOL13) UACC(NONE)
PERMIT APOL13 CLASS(TIMES) ID(IMSUSER1) ACCESS(UPDATE)
PERMIT APOL13 CLASS(TIMES) ID(STO) ACCESS(UPDATE)
```

関連資料:

- AO アプリケーション・プログラム・セキュリティー目的でのユーザーの定義について詳しくは、404 ページの『AO コマンド・セキュリティーとシステム定義』を参照してください。
- RACF セキュリティー・クラスのプログラミングについて詳しくは、「z/OS Security Server RACF システム・プログラマーのガイド」を参照してください。
- RACF コマンドについて詳しくは、「z/OS Security Server RACF コマンド言語解説書」を参照してください。

コマンド許可出口ルーチンを使用した AO コマンド・セキュリティーの実施

コマンド許可検査を実行するために、AO コマンド処理中にコマンド許可出口ルーチン (DFSCCMD0) が呼び出されます。DFSCCMD0 を使用すると、CMD 呼び出しまたは ICMD 呼び出しによって発行されるコマンドをコマンド verb、キーワード、リソース名のいずれかのレベルで保護できます。

DFSCCMD0 は、IMS.SDFSRESL に含める必要があります。

DFSCCMD0 のパラメーター・リストで、次のことが識別されます。

- 誰がコマンドを発行したか
 - 端末
 - LU 6.2 アプリケーション
 - ICMD 呼び出し (コマンド許可にユーザー ID が使用された場合)
 - ICMD 呼び出し (コマンド許可に PSB 名が使用された場合)
- RACF (またはそれに相当するもの) が呼び出されたかどうか
 - SAF (システム許可機能) 戻りコード
 - RACF 戻りコード
 - RACF 理由コード
- セキュリティー・コード
 - X'80000000' RACF は呼び出されませんでした (AOIS=C)。
 - X'00000000' ユーザーは RACF に対してコマンドを出すことを許可されています。
 - X'00000004' RACF は使用可能ではありません。
 - X'00000008' ユーザーは RACF に対して定義されていません。
 - X'0000000C' コマンドは RACF によって保護されていません。
 - X'00000010' ユーザーはコマンドの発行を許可されていません。

関連資料: コマンド許可出口ルーチン (DFSCCMD0) の詳細については、「IMS V15 出口ルーチン」を参照してください。

時間制御操作のセキュリティー

時間制御操作 (TCO) のためにセキュリティーを実施する方法には 2 とおりあります。TCO スクリプトをロードできるユーザー (LTERM) を制限する方法と、ロードされた TCO スクリプトが発行できる IMS コマンドを制限する方法です。

TCO スクリプトをロードできる LTERM を制限するには、TCO CNT 編集出口ルーチン (DFSTCNT0) を、その名前を変えずに変更します。IMS は、TCO スクリプトがロードされるときに、必ず DFSTCNT0 を呼び出しますが、デフォルトでは、DFSTCNT0 は LTERM を制限しません。

TCO スクリプトが発行できる IMS コマンドを制限するには、始動プロシージャで RCF 実行パラメーターに A または S を、TCORACF 実行パラメーターに Y を指定します。IMS は RACF と、コマンド許可出口ルーチンが存在する場合は、この出口ルーチンを呼び出します。

最後に、スクリプトの始めと終わりに、それぞれ、/SIGN ON コマンドと /SIGN OFF コマンドを入れる必要があります。スクリプトの始めにサインオンしたユーザー ID は、スクリプトが発行するコマンドに対して許可されているかどうか検査されます。

あるいは、DFSDCxxx PROCLIB メンバー内で TCOUSID パラメーターまたは SIGNTCO パラメーターを指定して、再始動時に IMS が TCO 端末にサインオンするようにします。

関連概念:

➡ IMS 時間制御操作 (オペレーションおよびオートメーション)

関連資料:

➡ プロシージャの RCF= パラメーター (システム定義)

➡ プロシージャの TCORACF= パラメーター (システム定義)

➡ IMS PROCLIB データ・セットの DFSDCxxx メンバー (システム定義)

➡ コマンド許可出口ルーチン (DFSCCMD0) (出口ルーチン)

➡ 時間制御操作 (TCO) 通信名テーブル (CNT) 出口ルーチン (DFSTCNT0) (出口ルーチン)

高速機能アプリケーション・プログラムのセキュリティー

高速機能アプリケーション・プログラム用または高速機能データベースへアクセスする DL/I プログラム用にセキュリティー保護を設計する場合は、従属領域内の処理および高速機能端末のネットワークを保護することを考慮してください。

従属領域および端末は、以下のように保護します。

- 従属領域でスケジューリングされている高速機能アプリケーション・プログラムの PSB を保護することによる、従属領域内での処理の保護。これは、RAS セキュリティーと RACF IIMS セキュリティー・クラスを使用して行います。
- サインオン検査およびトランザクション許可を使用した、高速機能端末のネットワークの保護。

JMP アプリケーション・プログラムのセキュリティ

JMP 領域で稼働するアプリケーション・プログラムは、UNIX System Services にアクセスする必要があります。そのため、JMP アプリケーションを実行する場合は RACF (または同等の製品) および IMS サインオン検査セキュリティを使用しなければなりません。

JMP 領域で RACF を使用するには、ユーザー ID を RACF に定義します。このユーザー ID には OMVS セグメントが必要です。

関連資料: RACF でのユーザー ID の定義については、「z/OS UNIX システム・サービスの計画」を参照してください。

セキュリティと CPI-C ドリブン・アプリケーション・プログラム

z/OS システム許可機能 (SAF) を使用すると、CPI-C ドリブン・アプリケーション・プログラムからの APSB 呼び出しで指定されているすべての PSB を保護することができます。

APSB SAF のセキュリティが使用可能になっていると、IMS は SAF を呼び出して、CPI-C アプリケーションに関連付けられたユーザーの USERID に基づいて、AIMS または Axxxxxxx 一般リソース・クラス (ここで、xxxxxxx は RCLASS 初期設定 EXEC パラメーターで指定された値) に対して、APSB 呼び出しで指定された PSB を保護します。したがって、RACF (またはユーザー・インストール・システム出口ルーチン) により保護したい PSB は、AIMS または Axxxxxxx リソース・クラスに定義しておく必要があります。AIMS リソース・クラスには PSB を入れることができるので、AIMS リソース・クラスに指定するすべての PSB 名は固有である必要があります。また、IMS システム定義時に、RCLASS 初期設定 EXEC パラメーターで RCLASS=IMS|xxxxxxx を指定する必要もあります。

CPI-C ドリブン・アプリケーション・プログラムが APSB 呼び出しを行うと、PSB が AIMS リソース・クラスに対して定義されていない場合または AIMS リソース・クラスがアクティブでない場合、IMS は APSB SAF セキュリティをバイパスします。IMS は、APSB SAF セキュリティをバイパスした場合、RAS セキュリティを使用して PSB を許可しようと試みます。RAS セキュリティがアクティブな場合、IMS は、CPI-C ドリブン・アプリケーション・プログラムが PSB を使用する権限を持っているものとみなします。

関連資料: RACF および IMS について詳しくは、「z/OS Security Server RACF セキュリティ管理者のガイド」の IMS を参照してください。

ODBA アプリケーション・プログラムのセキュリティ

z/OS システム許可機能 (SAF) を使用して、ODBA アプリケーション・プログラムからの APSB 呼び出しで指定されているすべての PSB を保護することができます。

APSB SAF のセキュリティが使用可能になっていると、IMS は SAF を呼び出して、AIMS または Axxxxxxx 一般リソース・クラスに対して、APSB 呼び出しで指定された PSB を保護します。これは、それぞれタスク・レベル (TCB) またはア

ドレス・スペース・レベル (ASCB/ASXB) で ODBA アプリケーションに関連付けられたユーザーに基づいて行われます。したがって、RACF (またはご使用のインストール・システム出口) で保護する PSB は、AIMS または Axxxxxxx リソース・クラスに定義しておく必要があります。

AIMS リソース・クラスには PSB を入れることができるので、AIMS リソース・クラスに指定するすべての PSB 名は固有である必要があります。

また、IMS システム定義時に、RCLASS=IMS|xxxxxxx を初期設定 EXEC パラメータに指定する必要もあります。

関連資料: RACF および IMS について詳しくは、「z/OS Security Server RACF セキュリティー管理者のガイド」の『RACF および情報管理システム (IMS)』を参照してください。

RACF データ・スペースの使用

RACF (または同等の製品) のデータ・スペースがサポートされる場合 (RACF 2.1 またはそれ以降)、IMS は IMS コマンドとトランザクションに対する RACF プロファイルを、IMS 制御領域ではなくそのデータ・スペースにロードします。

RACF データ・スペースを使用すると、/MODIFY コマンドによる RACF の IMS オンライン変更サポートは無効になります。ただし、RACF データ・スペースが使用されていないときは、依然として IMS オンライン変更サポートは有効です。

RACF データ・スペースの使用時に RACF パラメーターが /MODIFY PREPARE コマンドで使用されると、メッセージ DFS3432 RACF PARAMETER INVALID IF RACF DATA SPACE USED が出力されます。RACF コマンド SETROPTS RACLIST (*classname*) REFRESH を使用して、IMS アプリケーションの作業を停止させずに RACF データ・スペースの RACF リソース・プロファイルをリフレッシュすることができます。

MSC および共用キュー環境でのセキュリティ

複数の IMS システムで構成される複数システム結合機能 (MSC) および共用キュー環境では、各 IMS システムは、端末や、同じ環境内の他の IMS システムからトランザクションを受け取ることができます。他の IMS システムのトランザクションについては、追加のセキュリティ対策を講じる必要があります。

MSC ネットワークにおいては、各 IMS システムを、ローカル IMS システムまたはリモート IMS システムと見なすことができます。ローカル IMS システムは、端末からトランザクションを受け取ります。リモート IMS システムは、MSC リンクを通じてローカル IMS システムからトランザクションを受け取ります。MSC 環境の各 IMS システムは、あるタイプのトランザクションについてローカルで、別のタイプのトランザクションについてリモートとすることができます。

同様に、共用キュー環境では、各 IMS システムは、フロントエンド IMS システムまたはバックエンド IMS システム、あるいはその両方として機能できます。フロントエンド IMS システムは、端末からトランザクションを受け取ります。バックエンド IMS システムは、共用キューを通じてフロントエンド IMS システムからト

ランザクションを受け取ります。共用キュー環境の各 IMS システムは、フロントエンド IMS システムとバックエンド IMS システムの両方として機能できます。

MSC および共用キュー環境では、各 IMS システムそれぞれについて別個にセキュリティを実行する必要があります。各 IMS システムのセキュリティ管理は、他の IMS システムのセキュリティ管理システムから独立しています。各 IMS システムは、他の IMS システムが実行するセキュリティ検査を認識しません。

関連資料: MSC および共用キューについて詳しくは、「IMS V15 コミュニケーションおよびコネクション」を参照してください。

ローカル MSC およびフロントエンド共用キュー・セキュリティ

ローカル MSC や、フロントエンド共用キュー IMS システムのために行う特別なセキュリティ指定はありませんが、IMS システムがリモート MSC またはバックエンド共用キュー IMS システムとしても機能する可能性がある場合は、必ず、適切にセキュリティを実施してください。そうでない場合は、DB/DC または DCCTL 環境内のスタンドアロンの IMS システムに対する場合と同様に、ローカルおよびフロントエンド IMS システムに対してセキュリティを実施してください。

リモート MSC およびバックエンド共用キュー・セキュリティ

リモート MSC およびバックエンド共用キュー IMS システムの場合、セキュリティ用に RACF を使用できます。CHNG 呼び出しと AUTH 呼び出し、IMS 会話型据え置きプログラム間通信 (入力端末と同じ IMS で行われる) では、許可検査を実行し、このランザクションを入力したユーザーに IMS リソースを使用する許可があるかどうかを判別します。

従属領域から RACF 許可を行うには、セキュリティ環境を最初に確立する必要があります。従属領域が入力端末と同じ IMS の一部であり、そのランザクションを入力したユーザーがまだサインオンしている場合、サインオン時に IMS 制御領域に作成されたセキュリティ環境が、許可呼び出しに使用されます。サインオンで作成されたセキュリティ環境は、次の状態では使用できません。

- 従属領域は入力端末と同じ IMS の一部だが、ユーザーがサインオフしている。
- 従属領域が別の IMS の一部で、MSC リンクによって入力端末を持つ IMS に接続されている。
- 従属領域が、IMS 共用メッセージ・キュー・サポートを伴うシスプレックスの他の IMS の一部になっている。

このような場合、RACF 許可検査を行うために、セキュリティ環境を動的に作成する必要があります。このセキュリティ環境の動的作成により、CHNG または AUTH 呼び出しを処理するのに必要な時間が増加します。動的に作成されたセキュリティ環境は、IMS がメッセージの処理を終わるまで (次の同期点または GET UNIQUE まで) 保持されます。IMS でセキュリティ環境を動的に作成する場合、セキュリティ環境構築ユーザー出口 (BSEX) を使用して制御を行います。

APPC または OTMA からの入力の場合は、セキュリティが FULL として定義されていると、そのセキュリティ環境は CHNG または AUTH 呼び出しの前にすでに作成されています。APPC/OTMA セキュリティが NONE として定義さ


れている場合は、RACF は呼び出されません。APPC/OTMA セキュリティーが CHECK として定義され、CHNG または AUTH 呼び出しが行われる場合は、セキュリティ環境を動的に作成する必要があります。

アプリケーション・プログラムの CHNG または AUTH 呼び出しにより、次の IMS 出口ルーチンが呼び出された場合、その呼び出しが、入力端末と同じ IMS の一部ではない IMS 従属領域から行われると、その CTB のアドレスはゼロです。

- コマンド許可出口ルーチン (DFSCCMD0)
- トランザクション許可出口ルーチン (DFSCTRNO)
- セキュリティー再検証出口ルーチン (DFSCTSE0)

共用キュー環境における LTERM に対するリソース・アクセス・セキュリティ (RAS) のセキュリティは、その IMS バックエンドに静的に定義されている LTERM に対してのみサポートされます。

関連資料:

 セキュリティー環境構築ユーザー出口 (BSEX) (出口ルーチン)

MSC リンクから受け取ったトランザクションのセキュリティ

MSC リンクから IMS システムが受け取るトランザクションのセキュリティを指定するときに、オプションがいくつかあります。デフォルトでは、RACF と出口ルーチンは直接経路指定トランザクションのセキュリティを検査し、MSC リンクから受け取った非直接経路指定トランザクションのセキュリティを検査します。

また、ご使用のシステムで MSC トランザクション・セキュリティが必要でない場合は、MSC リンクから受け取ったトランザクションに対するすべてのセキュリティ検査を使用不可にすることができます。

MSC リンクから受け取ったトランザクションのセキュリティを検査するのに RACF または出口ルーチンを使用するよう指定した場合は、これらのセキュリティ機能が使用するユーザー ID も指定する必要があります。

DFSDCxxx PROCLIB メンバーで MSCSEC= 始動パラメーターの定位置パラメーターを使用すると、このトピックで説明しているセキュリティ指定をすべて行うことができます。MSCSEC= での指定は、ご使用の IMS システムが MSC リンクからトランザクションを受け取ったときに実行するデフォルトのセキュリティを定義します。

MSCSEC= 始動パラメーターで行うセキュリティ指定は、有効範囲がローカルであり、その指定を行った IMS システムにのみ適用されます。

関連資料: DFSDCxxx PROCLIB メンバーの詳細については、「IMS V15 システム定義」を参照してください。

MSC トランザクション・タイプのセキュリティの指定:

DFSDCxxx PROCLIB メンバー内にある、MSCSEC= 始動パラメーターは、2 つの定位置パラメーターを含んでいます。1 つめの定位置パラメーターを使用すると、MSC トランザクション・タイプに基づいて、受信した IMS が、トランザクション

のセキュリティーを検査するために、受け取る側の IMS が RACF と出口ルーチン
を呼び出すかどうか、また呼び出す場合はその時期を指定することができます。

セキュリティー検査が実行される対象で、MSC リンク上で受け取ることができるト
ランザクションには、2 つのタイプがあります。直接経路指定トランザクションと
非直接経路指定トランザクションです。MSCSEC= という 1 つ目の定位置パラメー
ターを使用して、RACF と出口ルーチンにセキュリティーを検査させたいトランザ
クションのタイプを指定します。次のオプションがあります。

LRDIRECT

RACF と出口ルーチンは直接経路指定トランザクションのセキュリティーを検
査しますが、非直接経路指定トランザクションについてはセキュリティー検査は
行われません。これはデフォルトです。

LRNONDR

RACF と出口ルーチンは非直接経路指定トランザクションのセキュリティーを
検査しますが、直接経路指定トランザクションについてはセキュリティー検査は
行われません。

LRALL

RACF と出口ルーチンは、両方のタイプのトランザクションを検査します。

LRNONE

IMS システムは、どちらのタイプのトランザクションについてもセキュリテ
ィー検査を要求しません。

MSC リンクから受け取ったトランザクションのセキュリティーを検査するのに
RACF と出口ルーチンを使用する場合、受け取る側の IMS システムは、TM およ
び MSC メッセージ経路指定および制御ユーザー出口ルーチン (DFSMSCE0) を呼
び出した後で RACF と DFSCTRN0 を呼び出します。

MSC セキュリティーのためのユーザー ID の指定:

RACF または出口ルーチン、あるいはその両方を使用して MSC セキュリティー検
査を行う場合は、DFSDCxxx PROCLIB メンバーで MSCSEC= 始動パラメーター
の 2 つ目の定位置パラメーターを使用して、セキュリティー検査の基盤となるユー
ザー ID を指定します。

パラメーターのオプションは、次のとおりです。

CTL

RACF と出口ルーチンが、受け取る側の IMS 制御領域のユーザー ID を使用
することを指定します。これはデフォルトです。

MSN

RACF と出口ルーチンがユーザー ID として MSNAME を使用することを指定
します。

USR

RACF と出口ルーチンが入力端末のユーザー ID を使用することを指定しま
す。

注: RACF リソース・クラスで定義されたユーザー ID は固有のものでなければなりません。MSC セキュリティー用にユーザー ID を定義する場合は、既存のユーザー ID との競合に注意してください。

MSC セキュリティーと、TM および MSC メッセージ経路指定および制御出口ルーチン:

TM および MSC メッセージ経路指定および制御出口ルーチン (DFSMSCE0) は、主として、TM および MSC 用のメッセージの経路指定と関係していますが、この出口を使用して、セキュリティ検査に使用されるユーザー ID を指定することもできます。

ユーザー ID に関して DFSMSCE0 で行える指定は、DFSDCxxx PROCLIB メンバー内の MSCSEC= キーワードの 2 つ目の定位置パラメーターによって指定されるものと同じです。

IMS は、MSC リンク受信経路指定用に DFSMSCE0 を呼び出すときに、DFSDCxxx MSCSEC= パラメーターに定義された 2 つ目のセキュリティ指定を DFSMSCE0 に渡します。DFSMSCE0 のセキュリティ指定が所定のトランザクションについて IMS から受け取ったものと異なる場合、DFSMSCE0 はそのトランザクションについて IMS から受け取った指定をオーバーライドできます。DFSMSCE0 は、このトランザクションを転送することもできますし、取り消すこともできます。

IMS は、トランザクションを MSC リンクから受け取ったときに 4 つのエントリー・ポイントの 1 つを通じて DFSMSCE0 を呼び出しますが、次の 2 つのエントリー・ポイントはセキュリティ検査にのみ使用されます。

- 受け取る側の IMS システムにとってローカルな非直接経路指定トランザクション・メッセージのための LRTRAN
- 受け取る側の IMS システムにとってローカルな直接経路指定トランザクション・メッセージのための LRDIR

関連資料: TM および MSC メッセージ経路指定および制御出口ルーチン (DFSMSCE0) の詳細については、「IMS V15 出口ルーチン」を参照してください。

MSC セキュリティーとトランザクション許可出口ルーチン:

トランザクション許可出口ルーチン (DFSCTRN0) は、セキュリティ再検証出口ルーチン (DFSCCTSE0) およびサインオン/オフ・セキュリティ出口ルーチン (DFSCSGN0) と連動して、トランザクションを使用する権限について個々のユーザー ID を検査します。

DFSCTRN0 は、RACF が一緒に使用されても、使用されなくても、ユーザー ID がトランザクションの実行を許可されていることを検査できます。IMS システム定義で RACF オプションと DFSCTRN0 の両方が選択された場合、出口ルーチンは、RACF がトランザクションを検証した後で活動化されます。トランザクション要求が RACF によってリジェクトされた場合、出口ルーチンは呼び出されません。

IMS システム定義で RACF オプションを選択していない場合は、必要に応じてこの出口ルーチンを使用して、当該トランザクションに関するユーザーの許可およびパスワードを検査することができます。

関連資料: トランザクション許可出口ルーチン (DFSCTRN0) の詳細については、「IMS V15 出口ルーチン」を参照してください。

APPC/IMS セキュリティー

APPC/MVS は、トランザクション・コードまたは特定の IMS システムへアクセスするためのユーザー権限を検証しません。LU 6.2 装置を使用してトランザクションを実行するためのユーザーの権限の完全なセキュリティ検証を提供するには、RACF を使用する必要があります。

APPC/MVS は、セキュリティに RACF リソース・クラス APPCTP を使用します。これは、トランザクション許可検証用に RACF に対して定義されたすべての IMS トランザクションのプロファイルを保持します。トランザクション許可出口ルーチン (DFSCTRN0) はトランザクション用に呼び出され、コマンド許可出口ルーチン (DFSCCMD0) はコマンド用に呼び出されます。

APPC ローカル LU によるセキュリティ検査

アウトバウンド非同期メッセージングの RACF セキュリティー検査に基本論理装置 (LU) またはローカル LU のいずれかを使用できます。DFSDCxxx PROCLIB メンバー・データ・セットの APPCLU 開始パラメーターを使って、非同期メッセージ応答に対し基本 LU またはローカル LU のどちらを使用するかを指定してください。

IMS 15 共用キュー環境では、ローカル論理装置 (LU) をサポートできなかったメンバーでもローカル LU を許容します。ただし、こうしたローカル LU のリソース情報は失われます。IMSplex メンバーがローカル LU をサポートしていない場合、非同期メッセージの送信には基本 LU が使用されます。

APPCLU 開始パラメーターを使用する前に、IMSplex のすべてのメンバーを IMS 15 にマイグレーションさせることをお勧めします。APPCLU 開始パラメーターについては、「IMS V15 システム定義」を参照してください。

ETO 端末のセキュリティ

ETO は、動的 LTERM サポートを提供します。管理者は、ユーザー・サインオン・プロセスまたはアプリケーション ISRT プロセスに基づいて、端末セッションに対してローカル LTERM を動的に作成し、割り振ることができます。LTERM は、物理端末でなく特定のユーザー ID へ関連付けることができます。LTERM をユーザー ID へ関連付けることにより、正しくないユーザーが物理端末でメッセージを受け取るのを防止できます。

関連資料: ETO セキュリティーについて詳しくは、「IMS V15 コミュニケーションおよびコネクション」の『ETO を用いた高セキュリティ環境の計画』を参照してください。

初期設定出口ルーチン (DFSINTX0) は、IMS が ETO 記述子をロードする前に呼び出されますが、このルーチンを使用すると、ユーザー定義セキュリティ情報がロ

ードされ、セキュリティー出口ルーチンから使用できるようになります。
DFSINTX0 を使用して、IMS がサインオン時に新規パスワードを再検査するよう要求することもできます。

IMS は、DFSINTX0 を呼び出した後で、以下のセキュリティー出口ルーチンを、次の順序で検索して使用します。

1. ログオン出口ルーチン (DFSLGNX0)

ETO がインプリメントされている場合は、ログオン出口ルーチン (DFSLGNX0) を IMS.SDFSRESL に組み込むことができます。DFSLGNX0 はインストール・システム出口ルーチンです。DFSLGNX0 の数ある機能のなかでも、特に、次のことを行えるようにします。

- 指定された基準に基づくログオン試行の許可または不許可。
- IMS からサインオン出口ルーチン (DFSSGNX0) に渡したいユーザー・データの作成または変更。

2. サインオン出口ルーチン (DFSSGNX0)

ETO が実装されている場合は、サインオン出口ルーチン (DFSSGNX0) を IMS.SDFSRESL に組み込むことができます。DFSSGNX0 はユーザー出口ルーチンです。このルーチンは、数あるその機能のうちでも、特に、指定された任意の基準に基づいてサインオン試行を許可または不許可にできるようにします。

DFSSGNX0 出口ルーチンを使用すると、ノード・ユーザー記述子を動的に作成し、接尾部とノード名を使用して、ユーザー ID とは異なるユーザー構造名を作成することもできます。

関連資料: DFSSGNX0、DFSLGNX0、DFSINTX0について詳しくは、「IMS V15 出口ルーチン」を参照してください。

DB/DC および DCCTL 従属領域とそれぞれのリソースの保護

従属領域の保護は重要です。従属領域を保護する方法は、2 とおりあります。従属領域内でアプリケーション・プログラムを無許可でスケジューリングできないようにする方法と、アプリケーション・プログラムのスケジューリング後にそれらのプログラムがリソースを無許可で使用できないようにする方法です。

以下のリストは、従属領域とリソースの、それぞれ保護できる各種タイプを示しています。

- BMP 領域の場合:
 - トランザクション・コード
 - PSBs
 - 論理端末か、または IMSBATCH プロシージャの OUT= キーワードを使用して指定されたトランザクション・コード
- IFP 領域の場合: PSB
- JBP 領域の場合:
 - PSBs
 - 論理端末か、または DFSJBP プロシージャの OUT= キーワードを使用して指定されたトランザクション・コード

- JMP 領域の場合: トランザクション・コード
- MPP 領域の場合: トランザクション・コード

IMS は、従属領域を保護するためにリソース・アクセス・セキュリティー (RAS) をサポートしています。

リソース・アクセス・セキュリティーを使用した従属領域の保護

リソース・アクセス・セキュリティー (RAS) は、従属領域で実行中のアプリケーション・プログラムがリソース (トランザクション、PSB、または LTERM) の使用を許可されない限り、できないようにします。RAS は、従属領域でのアプリケーション・プログラムのスケジューリングを制限しません。

RAS によって保護されるリソースにアクセスするためのアプリケーション・プログラムの権限は、従属領域のユーザー ID に基づきます。従属領域のユーザー ID は、アプリケーション・プログラムが使用しようとするリソースの RACF セキュリティー・クラス・プロファイルで許可する必要があります。

RAS の指定は、IMS、DBC、および DCC 始動プロシージャーで ISIS= 実行パラメーターを使用して行えます。

関連資料:

- IMS、DBC、または DCC 始動プロシージャーについて詳しくは、「IMS V15 システム定義」を参照してください。

RAS セキュリティーのための RACF リソース・クラス

RAS は、RACF リソース・クラスを使用して、保護するリソースと、それらのリソースにアクセスできる従属領域のユーザー ID を定義します。RAS が使用する RACF リソース・クラスには、次のものがあります。

- PSB および PSB のグループの場合は IIMS および JIMS
- LTERM および LTERM のグループの場合は LIMS および MIMS
- トランザクションおよびトランザクションのグループの場合は TIMS および GIMS

これらのクラスのなかには、RACF によって事前定義されているものもありますが、そうでないものもあります。RACF が提供するクラスで必要なセキュリティー・クラスが見つからない場合は、RACF リソース・クラス・マクロ ICHERCDE を使用して、インストール先定義クラス記述子テーブル (CDT) を作成できます。

RACF セキュリティー・クラスには、RAS が保護するリソースの名前と、それらのリソースを使用できるユーザー ID が含まれています。従属領域内のアプリケーション・プログラムがリソースにアクセスしようとする時、RAS は、リソースのクラス・プロファイルを検査して、従属領域のユーザー ID がそのリソースについて許可されているかどうかを調べます。目的のユーザー ID が指定されている場合、RAS はアプリケーション・プログラムへのアクセスを認可します。ユーザー ID が指定されていない場合は、RAS はアクセスを拒否します。

関連資料:

- RACF セキュリティーとの連動について詳しくは、以下を参照してください。

- 422 ページの『RACF セキュリティー計画の準備』
- z/OS Security Server RACF セキュリティー管理者のガイド
- z/OS Security Server RACF マクロおよびインターフェース
- ICHERCDE マクロを使用した新しいクラスの定義について詳しくは、「z/OS Security Server RACF システム・プログラマーのガイド」を参照してください。

従属領域におけるリソース使用の許可

IMS 従属領域ではセキュリティーに、RACF (または同等の製品) を使用できます。CHNG 呼び出しおよび AUTH 呼び出しと、IMS 会話型据え置きプログラム間通信 (入力端末と同じ IMS で行われる) では、許可検査を実行して、このトランザクションを入力したユーザーが IMS リソースの使用を許可されているかどうかを判別します。

従属領域から RACF 許可を行うには、セキュリティー環境を確立する必要があります。従属領域が入力端末と同じ IMS の一部であり、そのトランザクションを入力したユーザーがまだサインオンしている場合、サインオン時に IMS 制御領域に作成されたセキュリティー環境が、許可呼び出しに使用されます。サインオンで作成されたセキュリティー環境は、次の状態では使用できません。

- 従属領域は入力端末と同じ IMS の一部だが、ユーザーがサインオフしている。
- 従属領域が別の IMS の一部で、MSC リンクによって入力端末を持つ IMS に接続されている。
- 従属領域が、IMS 共用メッセージ・キュー・サポートを伴うシスプレックスの他の IMS の一部になっている。

このような場合、RACF 許可検査を行うために、セキュリティー環境を動的に作成されなければなりません。このセキュリティー環境の動的作成により、CHNG または AUTH 呼び出しを処理するのに必要な時間が増加します。動的に作成されたセキュリティー環境は、IMS がメッセージの処理を終わるまで (次の同期点または GET UNIQUE まで) 保持されます。

APPC または OTMA からの入力の場合は、セキュリティーが FULL として定義されていると、そのセキュリティー環境は CHNG または AUTH 呼び出しの前にすでに作成されています。APPC または OTMA セキュリティーが CHECK として定義され、CHNG または AUTH 呼び出しが行われる場合は、セキュリティー環境を動的に作成する必要があります。

許可呼び出しを行うためのセキュリティー環境の作成で使用するユーザー ID は、以下の環境と基準に基づいています。

MPP または IFP

メッセージ GU 呼び出しが完了すると、入力メッセージのセキュリティー値を使用して許可を実行します。セキュリティー値は、トランザクションが発行されたサインオン端末のユーザー ID、またはサインオフ端末の LTERM 名です。GU が発行されていない場合は、PSB 名が使用されます。

BMP メッセージ GU 呼び出しが完了すると、入力メッセージのセキュリティー値を使用して許可を実行します。セキュリティー値は、トランザクションが

発行されたサインオン端末のユーザー ID、またはサインオフ端末の LTERM 名です。GU が発行されていないか、または BMP がメッセージ・ドリブンではない場合、JCL JOB ステートメントで指定された USER= パラメーターが使用されます。USER= パラメーターが指定されていない場合、ユーザー ID として 0000000 が使用されます。

アクセス機能環境エレメント (ACEE) が IMS 制御領域に存在せず、動的セキュリティー環境を動的に作成できない場合は、デフォルトのセキュリティー環境が使用されます。従属領域に PARDLI=1 が指定されているか、または IMS システムが LSO=Y で指定されている場合、デフォルトのセキュリティー環境は、IMS 制御領域に関連付けられたユーザー ID で作成された IMS 制御領域の環境になります。それ以外の場合、デフォルトのセキュリティー環境は、IMS 従属領域に関連付けられたユーザー ID で作成された IMS 従属領域の環境になります。

アプリケーション・プログラムの CHNG または AUTH 呼び出しにより、以下の IMS 出口ルーチンが呼び出された場合、その呼び出しが、入力端末と同じ IMS の一部ではない IMS 従属領域から行われると、その CTB のアドレスはゼロです。

- コマンド許可出口ルーチン (DFSCCMD0)
- トランザクション許可出口ルーチン (DFSCTRN0)
- セキュリティー再検証出口ルーチン (DFSCTSE0)

OTMA のセキュリティー

OTMA のセキュリティーは、RACF または同等のセキュリティー製品によって実施されます。

以下のものを使用して OTMA セキュリティーを提供できます。

- RACF セキュリティー・レベル
- システム実行プロシージャー、トランザクション・メッセージ接頭部、および /SECURE OTMA コマンド
- 非同期保留キューのメッセージを無許可の使用から保護する

OTMA は、セキュリティー検査を実行するだけでなく、ネットワーク・セキュリティー資格情報を IMS に渡すことができます。この資格情報には、分散環境のアプリケーションから入力されたネットワーク・ユーザー ID とネットワーク・セッション ID が含まれます。OTMA は、ネットワーク・セキュリティー資格情報が以下のいずれかのユーザー・メッセージ出口を使用するアプリケーションから入力された場合、その資格情報を OTMA メッセージ接頭部のセキュリティー・データ・セッションで IMS に渡します。

- HWSSMPL0
- HWSSMPL1
- HWSJAVA0

以下の OTMA 呼び出し可能インターフェース (C/I) API を使用して、IMS にネットワーク・セキュリティー資格情報を渡すこともできます。

- otma_send_receivey
- otma_send_asynxcx

IMS コールアウト要求の場合、以下の両方の条件が当てはまれば、ネットワーク・セキュリティ資格情報はコールアウト・メッセージの OTMA メッセージ接頭部に含まれています。

- ネットワーク・セキュリティ資格情報が、IMS への入力メッセージの OTMA メッセージ接頭部のセキュリティ・データ・セクションに入っている。
- RESUME TPIPE 呼び出しを発行するアプリケーションが、ネットワーク・セキュリティ資格情報をサポートしている。

関連概念:

➡ 分散ネットワーク・セキュリティ資格情報のサポートおよび OTMA (コミュニケーションおよびコネクション)

関連タスク:

➡ OTMA セキュリティー (コミュニケーションおよびコネクション)

➡ RESUME TPIPE による同期コールアウト要求のリトリブ (コミュニケーションおよびコネクション)

関連資料:

➡ otma_send_asyncx API (アプリケーション・プログラミング API)

➡ otma_send_receivey API (アプリケーション・プログラミング API)

➡ トランザクションおよびコールアウト・メッセージ (コミュニケーションおよびコネクション)

➡ T パイプ用の保留キューからの出力の再開 (コミュニケーションおよびコネクション)

➡ セキュリティー・データ・セクション (コミュニケーションおよびコネクション)

➡ OTMA セキュリティー・データ・フィールドの説明 (コミュニケーションおよびコネクション)

IMS Connect のセキュリティ

IMS Connect には、メッセージが IMS Connect とデータ・ストアそれぞれに到着すると同時に、メッセージに対して行われるセキュリティ検査を実装および変更するためのさまざまなオプションがあります。


IMS Connect は、IMS Connect 内のセキュリティ検査を行うための 2 つのオプションを提供します。1 つは、RACF を直接呼び出すように IMS Connect を構成する方法で、もう 1 つは、IMS Connect ユーザー・メッセージ出口ルーチンによってセキュリティ・ユーザー出口ルーチンを呼び出すようにする方法です。

IMS Connect が提供する追加のセキュリティ機能には、他に以下のものがあります。

- パスワード管理サポート
- データ・ストアに到着するメッセージ用のトラステッド・ユーザー種別
- OTMA アクセサー環境エレメント (ACEE) タイムアウト指定サポート

- IMS TM に接続するクライアントからの接続の場合、Secure Sockets Layer (SSL) サポート。IMS DB に接続するクライアントからの接続を、SSL を使用して保護するには、IBM z/OS Communications Server Application Transparent Transport Layer Security 機能 (AT-TLS) を使用できます。
- IMS との間で行われるセキュリティー資格情報の受け渡しのサポート。この資格情報には、分散環境でユーザーが入力するネットワーク・ユーザー ID およびネットワーク・セッション ID も含まれます。

関連概念:

 IMS Connect セキュリティー・サポート (コミュニケーションおよびコネクション)

DB/DC 環境および DCCTL 環境向けの IMS セキュリティーの活動化

このトピックには、IMS セキュリティー設計の活動化と、RACF、およびプログラム出入口ルーチンの使用方法に関するガイダンスが記載してあります。

セキュリティー出入口ルーチンの準備

許可の一部として、各セキュリティー出入口ルーチンを準備する必要があります。

- サインオン/オフ・セキュリティー出入口ルーチン (DFSCSGN0)

/SIGN ON/OFF セキュリティー出入口ルーチンは、インストール・システムではモジュール DFSCSGN0 としてコーディングしなければなりません。この出入口ルーチンは、有効なユーザー ID とそれぞれに関連付けられたパスワード、および RACF パスチケット (使用されている場合のみ) のテーブルにアクセスできる必要があります。このテーブルはアドレス可能となるように、モジュール DFSCSGN0、トランザクション許可出入口ルーチン (DFSCTRNO)、または IMS 中核のいずれかに常駐していなければなりません。出入口ルーチンは、個々の正常なサインオンを監視しなければなりません。/SIGN OFF コマンドが実行された場合、出入口ルーチンはそのユーザー ID を /SIGN ON 用に使用可能なものとしてマークしなければなりません。出入口ルーチンは、ロギング用のユーザー検証ストリングのデータ部分に情報を挿入できます。(レジスター内のアドレスはユーザー検証ストリングを指し示します。)

- トランザクション許可出入口ルーチン (DFSCTRNO)

トランザクション許可出入口ルーチン (DFSCTRNO) は、有効なユーザー ID、RACF パスチケット、パスワード、および各有効なユーザー ID と関連付けられたトランザクションのテーブルにアクセスできる必要があります。このテーブルはアドレス可能となるように、モジュール DFSCSGN0、/SIGN ON/OFF セキュリティー出入口ルーチン (DFSCSGN0)、または IMS 中核のいずれかに常駐していなければなりません。テーブルが中核にある場合、トランザクション許可出入口ルーチンとサインオン検査出入口ルーチンはそのテーブルを共用できます。

メッセージ編集ルーチンを使用した場合、メッセージを編集した後にセキュリティーが検査されます。

- コマンド許可出入口ルーチン (DFSCCMD0)

コマンド許可出口ルーチン (DFSCCMD0) は、有効なユーザー ID、パスワード、および有効な各ユーザー ID へ関連付けられたコマンドのテーブルへアクセスできなければなりません。このテーブルはアドレス可能となるように、モジュール DFSCCMD0、/SIGN ON/OFF セキュリティ出口ルーチン (DFSCSGN0)、または IMS 中核のいずれかに常駐していなければなりません。テーブルが中核にある場合、コマンド許可出口ルーチン、トランザクション許可出口ルーチン、およびサインオン検査出口ルーチンはそのテーブルを共用できます。

- リソース・アクセス・セキュリティ出口ルーチン (RASE) を使用している場合
リソース・アクセス・セキュリティ出口ルーチン (RASE) を使用して、RAS によって提供されるセキュリティ機能を増強または調整することができます。RASE ユーザー出口は、RACF への呼び出しの後で呼び出され、IMS 従属領域がアクセスしようとする IMS リソース (トランザクション、PSB、および LTERM) を使用することを IMS 従属領域に直接許可します。出口の解説書のリソース・アクセス・セキュリティ・ユーザー出口の説明に、IMS に対する RASE ユーザー出口の定義方法が記載されています。

関連タスク:

➡ トランザクション許可出口ルーチン (DFSCTRNO) の使用 (コミュニケーションおよびコネクション)

関連資料:

➡ リソース・アクセス・セキュリティ・ユーザー出口 (RASE) (出口ルーチン)

➡ コマンド許可出口ルーチン (DFSCCMD0) (出口ルーチン)

➡ サインオン/オフ・セキュリティ出口ルーチン (DFSCSGN0) (出口ルーチン)

RACF セキュリティ計画の準備

RACF のセキュリティ計画には、どのリソースを保護する必要があるかの定義、システム定義マクロ内でセキュリティ・オプションの指定、および保護するリソースを RACF に定義することが含まれます。

RACF を使用するセキュリティ計画を準備するには、以下のように行います。

1. 保護したいすべての IMS オンライン・リソースのリストを準備します。その際、対象となるリソース全体を概観できるようグループに分けて配置します。
2. リソース・グループを保護するセキュリティ機能を選択します。
3. 画面フォーマットを設計し、トランザクションおよびコマンドでのパスワード用の非表示フィールドを組み込みます。
4. SECURITY、COMM、および IMSGEN システム定義マクロでセキュリティ・オプションを指定します。
5. RACF に対して RACF リソース・クラス・プロファイルを定義します。
6. RACF にユーザー、グループ、データ・セットを追加します。
7. RACF に対してトランザクションとトランザクション・グループを定義します。

8. RACF に対してデータベース、セグメント、フィールド、その他のリソースとリソース・グループを定義します。
9. RACF に対してコマンドとコマンド・グループを定義します。
10. 拡張リソース保護ソース (APPL) を定義します。
11. IMS.PROCLIB の JCL プロシージャを修正します。

IMS セキュリティーは、各種の RACF リソース・クラスを使用します。それらのクラスは個々のリソースまたはリソース・グループを定義し、次のカテゴリーに分割されます。

APPC/IMS

APPC/IMS によって使用される APPC リソース・クラスは IMS に固有のものではなく、以下のものが含まれています。

APPCTP

RACF に対する LU 6.2 トランザクションのトランザクション・プロファイルを識別します。

APPCLU

セッションの会話セキュリティーを指定します。

APPCPORT

所定の LU (入力の APPC ポート) からのシステムへのアクセスを制御します。

関連資料: APPC/IMS と、APPC/IMS によって使用される RACF リソース・クラスについて詳しくは、「IMS V15 コミュニケーションおよびコネクション」の『APPC トランザクション・セキュリティー』のトピックを参照してください。

アプリケーション

アプリケーション・リソース・クラス APPL は、RACF に対して定義されたすべてのサブシステムのプロファイルを保持します。アプリケーション・グループ・リソース・クラス AIMS は、RACF に対して定義されたすべての APSP のプロファイルを保持します。IMS システムは、サインオン時のシステム・アクセス許可検査用の IMSID 名を使用して、このクラス内で (IMSCTRL マクロによって) 定義されます。

コマンド

コマンド・リソース・クラス CIMS は、コマンド許可検査用に RACF に対して定義されたすべてのコマンドのプロファイルを保持します。コマンド・グループ・リソース・クラス DIMS を使用すると、共通のアクセス権限プロファイルを持つ IMS コマンドをグループ化できます。コマンドは許可ユーザー ID 用に定義されます。

データベース

データベース・リソース・クラス PIMS は、許可検査用に RACF に対して定義されたすべてのデータベースのプロファイルを保持します。データベース・グループ・リソース・クラス QIMS を使用すると、共通のアクセス権限プロファイルを持つデータベース・リソースをグループ化できます。

フィールド

フィールド・リソース・クラス FIMS を使用すると、データベースのフィ

ワールドの RACF 許可検査を実行できます。フィールド・リソース・グループ・クラス HIMS を使用すると、RACF 許可検査用に共通アクセス・フィールドをグループ化できます。

LTERM

LTERM リソース・クラス LIMS は、LTERM 許可検査用に RACF に対して定義されたすべての IMS LTERM のプロファイルを保持します。

LTERM グループ・リソース・クラス MIMS を使用すると、共通のアクセス権限プロファイルをもつ IMS LTERM をグループ化できます。

その他

このリソース・クラス OIMS およびそのグループ・リソース・クラス WIMS は、他のどのクラスにも適合しないリソースに使用できます。この呼び出しは、AUTH 呼び出しとのインターフェースに使用できます。

PSB PSB リソース・クラス IIMS は、PSB 許可検査用に RACF に対して定義されたすべての IMS PSB のプロファイルを保持します。PSB グループ・リソース・クラス JIMS を使用すると、共通のアクセス権限プロファイルをもつ IMS PSB をグループ化できます。

セグメント

セグメント・リソース・クラス SIMS は、RACF に対して個々のセグメントを識別します。セグメント・グループ・リソース・クラス UIMS を使用すると、RACF 許可検査用に共通のアクセス権限プロファイルを持つセグメントをグループ化できます。

トランザクション

トランザクション・リソース・クラス TIMS は、トランザクション許可検査用に RACF に対して定義されたすべての IMS トランザクションのプロファイルを保持します。トランザクション・グループ・リソース・クラス GIMS を使用すると、共通のアクセス権限プロファイルをもつ IMS トランザクションをグループ化できます。

IMS システムが使用するリソース・クラスの名前を指定するには、RCLASS= 初期設定 EXEC パラメーターを使用します。

以下の表は、リソース・クラスの割り当てを示しています。

表 30. リソース・クラスの割り当て

リソース・クラス・タイプ	リソース・クラス名	
	RACF 定義名	ユーザー定義名
APPC/IMS	APPCTP、APPCLU、 APPCPORT、およびその他	IMS 固有のリソース・クラス はありません。
アプリケーション・リソース・クラス	APPL	これは、IMS 固有のリソース・クラスではありません。
アプリケーション・グループ名リソース・クラス	AIMS	Axxxxxxx
コマンド・リソース・クラス	CIMS	Cxxxxxxx
コマンド・グループ・リソース・クラス	DIMS	Dxxxxxxx

表 30. リソース・クラスの割り当て (続き)

リソース・クラス・タイプ	リソース・クラス名	
	RACF 定義名	ユーザー定義名
データベース・リソース・クラス	PIMS	Pxxxxxxx
データベース・グループ・リソース・クラス	QIMS	Qxxxxxxx
フィールド・リソース・クラス	FIMS	Fxxxxxxx
フィールド・グループ・リソース・クラス	HIMS	Hxxxxxxx
LTERM リソース・クラス	LIMS	Lxxxxxxx
LTERM グループ・リソース・クラス	MIMS	Mxxxxxxx
その他のリソース・クラス	OIMS	Oxxxxxxx
その他のグループ・リソース・クラス	WIMS	Wxxxxxxx
PSB リソース・クラス	IIMS	Ixxxxxxx
PSB グループ・リソース・クラス	JIMS	Jxxxxxxx
RESUME TPIPE クラス	RIMS	Rxxxxxxx
セグメント・リソース・クラス	SIMS	Sxxxxxxx
セグメント・グループ・リソース・クラス	UIMS	Uxxxxxxx
トランザクション・リソース・クラス	TIMS	Txxxxxxx
トランザクション・グループ・リソース・クラス	GIMS	Gxxxxxxx

RACF リソース・クラスは、RACF リソースのクラス記述子テーブル (CDT) に定義されます。最初は、RACF 定義リソース・クラス (424 ページの表 30 の 2 番目の列に示す) が CDT に事前定義されています。リソース・クラスを追加したり、またはユーザー定義名でリソース・クラスを定義するには、RACF リソース・クラス・マクロ ICHERCDE を使用して、インストール先定義 CDT を作成する必要があります。


要件: 大/小文字混合のパスワードを許可する予定 (かつセキュリティーに RACF を使用) の場合は、SETROPTS,PASSWORD(MIXEDCASE) コマンドを発行する必要があります。

関連資料:

- ICHERCDE マクロについて詳しくは、「z/OS Security Server RACF マクロおよびインターフェース」を参照してください。

- RACF リソース・クラス記述子テーブルの更新について詳しくは、「z/OS Security Server RACF システム・プログラマーのガイド」を参照してください。

関連資料:

 AUTH 呼び出し (アプリケーション・プログラミング API)

APSB SAF セキュリティーの使用可能化および使用不可化

APSB SAF セキュリティーは、次の方法のいずれかを使用して使用可能にすることができます。

- CPI-C アプリケーションの TP プロファイルの TP スケジューラー・セクションに RACF=FULL を指定し、IMS コマンド /SECURE APPC PROFILE を出します。コマンド /SECURE APPC PROFILE は、TP プロファイルに RACF=FULL の指定がある CPI-C アプリケーションに対してだけ APSB SAF セキュリティーを使用可能にします。このコマンドによって、他のすべての CPI-C アプリケーションの APSB SAF セキュリティーは使用不可になります。
- すべての CPI-C アプリケーションに対し APSB SAF セキュリティーを使用可能にするには、IMS コマンド /SECURE APPC FULL を出します。

APSB SAF セキュリティーを使用不可にするには、IMS コマンドの /SECURE APPC CHECK または /SECURE APPC NONE を出します。

RS カタログ・リポジトリおよび IMSRSC リポジトリへのアクセスの制限

Repository Server (RS) は、z/OS System Authorization Facility (SAF) の呼び出し可能サービスを使用して、RS カタログ・リポジトリおよび IMSRSC リポジトリへのアクセスを制限します。RACF などの、SAF が使用可能になった外部セキュリティ・マネージャーを使用して、RS セキュリティーを管理することができます。

アクセスを制限または許可できるリソースとしては以下のものがあります。

- リポジトリ
- RS カタログ・リポジトリ
- リポジトリ内のメンバー
- リポジトリに関連付けられた監査レベル

RACF RDEFINE コマンドで一般リソース・プロファイルを定義することによりリソースを保護します。

RACF PERMIT コマンドでリソース・プロファイルを定義しているユーザーにアクセスを付与します。

RS カタログ・リポジトリおよびユーザー・リポジトリへのアクセスを制限するには、以下のようにします。

1. ご使用のセキュリティ・データベースで以下を実行してください。
 - a. FACILITY などの既存のリソース・クラスを選択するか、RS のセキュリティ検査用のクラスを作成します。

b. 以下のようにして、このクラスの下に一般リソース・プロファイルを定義します。

- `FRPREP.repositoryname` の形式を使用して 1 つ以上のリソース・プロファイルを定義することにより、ユーザー・リポジトリへのアクセスを制御することができます。

例えば、`FRPREP.IMS_REPOS` を使用してリソース・プロファイルを定義することにより、`IMS_REPOS` という名前のリポジトリを保護できます。または、`FRPREP.*` を使用してリソース・プロファイルを定義することにより、すべてのリポジトリを保護することができます。

- `FRPBATCH` コマンドまたはその他の要求の一部として `RS` カタログ・リポジトリを変更またはアクセスできるユーザーに対するアクセス権を制御するには、`FRPREP.CATALOG` のリソース・プロファイルを定義して、許可ユーザーのみにアクセスを付与することができます。
- メンバー・レベル・セキュリティを定義することにより、各リソースまたはメンバーへのアクセスを制御することができます。メンバー・レベル・セキュリティは、許可されない呼び出し元によるアクセスに対して使用可能にされます。

メンバー・レベル・セキュリティは、ユーザー・リポジトリに定義されているセキュリティ・フィールドに基づいて使用可能にされます。保護する各メンバーのリソース・プロファイルを定義するには、以下の形式を使用できます。

`FRPMEM.repositoryname.product.type.membername`

推奨事項: `FACILITY` クラスの場合、プロファイル名の長さは、39 バイトのみに指定できます。メンバー・レベル・セキュリティが使用されている場合は、`RACF` クラス記述子テーブル (`ICHRRCDE`) に新しいクラスを追加し、新しいクラスで `RACF` ルーター・テーブル (`ICHRFR01`) を更新することにより、`RS` 用の新しい `RACF` クラスを作成します。

- `AUDIT` レベルを変更できるユーザーを制御するには、次の形式を使用してリソース・プロファイルを定義することができます。

`FRPAUD.repositoryname.product.type.TYPE`

`product` および `type` を指定する場合の考慮事項は、`CSL RM` アドレス・スペース内のリポジトリのセキュリティを設定する場合と同じです。

c. `RACF` 定義プロファイルを設定して、これらのプロファイルに適切なユーザーを許可します。 `RACF ADDGROUP` コマンドを使用することにより、同時にユーザー `ID` のグループを許可することができます。

以下は、プロファイルを定義し、適切なユーザーにアクセスを付与するために使用される `RACF` コマンドの例です。

リポジトリ:

```
RDEFINE XFACILIT FRPREP.REP01 UACC(NONE)
PERMIT FRPREP.REP01 CLASS(XFACILIT) ID(VIEWER1) ACCESS(READ)
PERMIT FRPREP.REP01 CLASS(XFACILIT) ID(ADMIN1) ACCESS(ALTER)
```

RS カタログ・リポジトリ:

```
RDEFINE XFACILIT FRPREP.CATALOG UACC(NONE)
PERMIT FRPREP.CATALOG CLASS(XFACILIT) ID(ADMIN1) ACCESS(ALTER)
```

リポジトリ内のメンバー:


```
RDEFINE XFACILIT FRPMEM.REP01.DFS.RSC.IMSPLEX1.TRAN.PART UACC(NONE)
PERMIT FRPMEM.REP01.DFS.RSC.IMSPLEX1.TRAN.PART CLASS(XFACILIT) ID(USER_UTIL10)
ACCESS(UPDATE)
PERMIT FRPMEM.*.*.*.*.* CLASS(XFACILIT) ID(USER_UTIL20) ACCESS(READ)
```

リポジトリに関連付けられた監査レベル:


```
RDEFINE XFACILIT FRPAUD.REP01.DFS.RSC.TYPE UACC(NONE)
PERMIT FRPAUD.REP01.DFS.RSC.TYPE CLASS(XFACILIT) ID(USER_ZOSMI) ACCESS(UPDATE)
```


2. IMS PROCLIB データ・セット内の FRPCFG メンバーで、SAF_CLASS パラメーターに、RACF 定義プロファイルに定義したクラス名と同じクラス名を指定します。


関連概念:

 [IMSRSC リポジトリの概要 \(システム定義\)](#)


関連資料:

 [FRPBATCH での ADD コマンド \(システム・プログラミング API\)](#)

 [Repository Server コマンド \(コマンド\)](#)

 [IMS PROCLIB データ・セットの FRPCFG メンバー \(システム定義\)](#)

関連情報:

 [FRPBATCH でのコマンド \(システム・プログラミング API\)](#)

CSL RM アドレス・スペースでの IMSRSC リポジトリのセキュリティの設定に関する考慮事項

Common Service Layer (CSL) Resource Manager (RM) のアドレス・スペースは、IMSRSC リポジトリ・サービスの許可呼び出し元です。

FRPREP.repository_name を使用してプロファイルを定義することにより、RM アドレス・スペースがリポジトリへのアクセスを許可されるかどうかを指定することができます。

RM がリポジトリへのアクセスを許可される場合は、RM へのすべての許可呼び出し元は、RM を介してリポジトリにアクセスでき、さらに、メンバー・レベル・セキュリティが定義されているかどうかに関係なく、リポジトリ内のすべてのメンバーにもアクセスできます。メンバー・レベル・セキュリティは、RM への許可呼び出し元には使用されません。

RM がリポジトリへのアクセスを許可される場合、非許可呼び出し元に対するリポジトリ内のメンバーへのアクセスは、メンバー・レベル・セキュリティが基になります。リポジトリに対してメンバー・レベル・セキュリティがセットアップされていない場合は、RM へのすべての非許可呼び出し元は、RM を介してリポジトリにアクセスでき、さらにリポジトリ内のすべてのメンバーにもアクセ

スできます。メンバー・レベル・セキュリティーが使用可能になっている場合、RM への非許可呼び出し元は、呼び出し元がメンバーへのアクセスを許可されている場合にのみ、RM を介してリポジトリにアクセスできます。

RM のメンバー・レベル・セキュリティーは、Repository Server (RS) に対して *product*、*type*、および *membername* に以下の値を使用して定義できます。

- *product*=DFS
- *type* = RSC
- *membername*=plexnamersctyperscname

plexname

リポジトリが定義されている CSL IMSplex の 8 バイトの名前

rsctype

8 バイトのリソース・タイプ。次のいずれかになります。

- DB
- DBDESC
- LTERM
- MSLINK
- MSNAME
- MSPLINK
- PGM
- PGMDDESC
- TRAN
- TRANDESC
- RTC
- RTCDESC


rscname

保護する 8 バイトのリソース名

例えば、リポジトリ IMS_REPOS にある IMSPLEX1 内の PART という名前のトランザクションを保護するには、

FRPMEM.IMS_REPOS.DFS.RSC.IMSPLEX1.TRAN.PART の規則を RACF に定義します。RM へのすべての許可呼び出し元は、トランザクション PART にアクセスできます。RM へのすべての非許可呼び出し元 (リポジトリの移植ユーティリティーなど) は、そのユーティリティーを実行するために発行されたジョブのユーザー ID が、FRPMEM.IMS_REPOS.DFS.RSC.IMSPLEX1.TRAN.PART のアクセス規則によって定義されたメンバー・レベル・セキュリティーにより RACF に定義されている場合にのみ、トランザクション PART にアクセスできます。

関連概念:

 [IMSRSC リポジトリの概要 \(システム定義\)](#)

235 ページの『IMSRSC リポジトリの CSL RM 管理』

438 ページの『DBCTL 環境のセキュリティーに関する考慮事項』

Repository Server の RACF 定義プロファイルの例

以下は、XFACILIT と呼ばれる RACF クラスについて、Repository Server (RS) を RACF に定義する方法の例です。


```
/* Define Resource Profiles */
/* FRPREP.CATALOG */
RDEFINE XFACILIT FRPREP.CATALOG UACC(NONE)
/* FRPREP.<repository name> */
RDEFINE XFACILIT FRPREP.* UACC(NONE)
/* FRPMEM.<repository name>.<product>.<type>.<plexname>.<rsctype>.<rscname> */
RDEFINE XFACILIT FRPMEM.*.*.*.*.* UACC(NONE)
/* FRPAUD.<repository name>.<product>.<type> */
RDEFINE XFACILIT FRPAUD.*.*.* UACC(NONE)

/* Define Groups */
/* View Data Group */
ADDGROUP FRPVIEW
/* View Edit Group */
ADDGROUP FRPEDIT
/* Operations Group */
ADDGROUP FRPOPER
/* Administration Group */
ADDGROUP FRPADMIN
/* Grant Access to groups as follows */
/* View Data Group */
PERMIT FRPREP.* CLASS(XFACILIT) ID(FRPVIEW) ACCESS(READ)
PERMIT FRPMEM.*.*.*.*.* CLASS(XFACILIT) ID(FRPVIEW) ACCESS(READ)
/* Edit Data Group */
PERMIT FRPREP.* CLASS(XFACILIT) ID(FRPEDIT) ACCESS(READ)
PERMIT FRPMEM.*.*.*.*.* CLASS(XFACILIT) ID(FRPEDIT) ACCESS(UPDATE)

/* Operations Group */
PERMIT FRPREP.CATALOG CLASS(XFACILIT) ID(FRPOPER) ACCESS(READ)
PERMIT FRPREP.* CLASS(XFACILIT) ID(FRPOPER) ACCESS(CONTROL)

/* Administration Group */
PERMIT FRPREP.CATALOG CLASS(XFACILIT) ID(FRPADMIN) ACCESS(READ)
PERMIT FRPREP.* CLASS(XFACILIT) ID(FRPADMIN) ACCESS(ALTER)
PERMIT FRPMEM.*.*.*.*.* CLASS(XFACILIT) ID(FRPADMIN) ACCESS(UPDATE)
PERMIT FRPAUD.*.*.* CLASS(XFACILIT) ID(FRPADMIN) ACCESS(UPDATE)
PERMIT FRPFLD.*.*.* CLASS(XFACILIT) ID(FRPADMIN) ACCESS(UPDATE)
PERMIT FRPHST.*.*.* CLASS(XFACILIT) ID(FRPADMIN) ACCESS(UPDATE)
/* Connect Users */
CONNECT <user1> GROUP(FRPVIEW)
CONNECT <user2> GROUP(FRPEDIT)
CONNECT <user3> GROUP(FRPOPER)
CONNECT <user4> GROUP(FRPADMIN)
```

関連概念:

 [IMSRSC リポジトリの概要 \(システム定義\)](#)

DB/DC および DCCTL 用のシステム始動時のセキュリティの制御

IMS および DCC 始動プロシージャの EXEC パラメーターは、現在の実行中に行われるセキュリティ検査の種類を制御する方法を提供します。これらのパラメーターは、システム定義マクロに指定されたさまざまなタイプのセキュリティ用のスイッチとして機能します。

また、それらはセキュリティ検査の選択をオーバーライドするために MTO がどのような柔軟性を持つかも決めます。これらのパラメーターの設定を、全体的なセキュリティ設計と操作手順の両方と調整する必要があります。セキュリティ用

の EXEC パラメーターは、RCLASS、SECCNT、TRN、SGN、RCF、ISIS、ASOT、ALOT、AOI1、AOIS、および TCORACF です。

IMS および DCC プロシーチャーのデフォルト値は、いずれも、セキュリティーを指定しません。管理者は、それらをリセットして、セキュリティーを使用可能にする必要があります。

セキュリティー機能とそれを指定するために使用する EXEC パラメーターを、以下の表に示します。表内の一部の EXEC パラメーターは、システム定義マクロで指定された関連パラメーターをオーバーライド (つまり、置換) します。

管理者はセキュリティー・テーブルのレベルと中核用の接尾部 ID を一致させなければなりません。MTO の操作上の制約事項については、401 ページの『マスター端末のセキュリティーに関する考慮事項』で説明しています。

関連資料: JCL パラメーターとそれらのパラメーター値については、「IMS V15 システム定義」を参照してください。

表 31. IMS セキュリティーを制御する EXEC パラメーター

セキュリティー機能の選択項目	EXEC パラメーター	セキュリティー選択項目のパラメーター値		注
		使用不可	使用可能	
IMS をリソース・クラスとして RACF に識別	RCLASS	RCLASS を指定しない	RCLASS に名前を指定し、さらに RCF= を指定	
MTO に通知するまでのセキュリティー違反の数	SECCNT	0	1、2、3	
トランザクション許可	TRN	N	Y、F	1 (432 ページ)、2 (432 ページ)、11 (432 ページ)
サインオン検査	SGN	N	D、E、F、M、W、X、Y、Z、G	1 (432 ページ)、2 (432 ページ)、3 (432 ページ)、8 (432 ページ)
トランザクション許可またはサインオン検査用の RACF セキュリティー	RCF	N	A、B、C、R、S、T、Y	4 (432 ページ)、5 (432 ページ)、8 (432 ページ)
RAS 従属領域セキュリティー	ISIS	0、1、2、N	A、C、R	6 (432 ページ)、9 (432 ページ)
自動サインオフ	ASOT	0 または 1440	10 から 1439	7 (432 ページ)





表 31. IMS セキュリティーを制御する EXEC パラメーター (続き)

セキュリティ機能の選択項目	EXEC パラメーター	セキュリティ選択項目のパラメーター値		注
		使用不可	使用可能	
自動ログオフ	ALOT	0 または 1440	10 から 1439	
CMD 呼び出しのためのセキュリティ検査	AOI1	N	A、C、R、S	8
ICMD 呼び出しのためのセキュリティ検査	AOIS	S	A、C、N、R	8
TCO 発行コマンドの RACF 検査	TCORACF	N	Y	8、10

注:

1. 値が N の場合、/NRESTART コマンドで MTO は、オプションで検査を呼び出すこともできます。
2. 値が Y の場合、セキュリティ機能は MTO によってオーバーライドされた場合以外はアクティブになります。
3. 値 M は、単一ユーザー ID 用の複数サインオンを示します。値 Z は Y + M に相当し、値 G は F + M に相当します。
4. RACF ライセンス・プログラムは、コマンド許可、トランザクション許可、サインオン検査のいずれかの出口ルーチンと結合して使用されます。
5. ヌル値を指定した場合は、システム定義で指定したデフォルトが選択されます。
6. ISIS キーワード・パラメーターを使用すると、必要な従属領域セキュリティのタイプ (RAS) と、セキュリティ・タイプが使用するセキュリティ機能 (RACF、および出口ルーチン) を選択できます。
7. ETO を使用して定義された端末では、最後の自動ログオン・ユーザーの最後のキューが完了した時点で、その自動ログオン・ユーザーは自動サインオフ・タイムアウト・インターバルを待たずに、ただちにサインオフします。
8. この指定はチェックポイント・レコードに含まれていないので、IMS を初期設定するたびに値を変更できます。
9. ISIS パラメーターの値 0、1、または 2 は互換性のために許容されます。内部的に、これらの値は ISIS=N に変換されます。
10. DFSDCxxx PROCLIB メンバーで TCOUSID または SIGNTCO が指定されている場合、TCORACF は Y に設定されます。
11. IMS システムのコールド・スタートを実行して、変更を有効にします。

関連資料:

-  IMS PROCLIB データ・セットの DFSDCxxx メンバー (システム定義)
-  DCC プロシージャ (システム定義)
-  IMS プロシージャ (システム定義)
-  /NRESTART コマンド (コマンド)

DB/DC 環境および DCCTL 環境におけるオンラインでのセキュリティー変更の実装

セキュリティー制約に RACF を使用する際に、オンラインによるセキュリティー定義の変更を行えます。

RACF セキュリティー定義をオンラインで変更するには、RACF データベースを更新します。次に、RACF コマンド SETROPTS RACLIST(*classname*) REFRESH を出します。

RACF は、*classname* に対応するクラスと、クラス記述子テーブル (CDT) 項目で同じ POSIT 値を共用するすべてのクラスをリフレッシュします。

関連資料: SETROPTS コマンドについて詳しくは、「z/OS Security Server RACF コマンド言語解説書」を参照してください。

DB/DC 環境および DCCTL 環境でのセキュリティー違反の制御

セキュリティー違反は、インストール・システムのセキュリティー管理の指針に従って処理されます。

IMS は、次のセキュリティー違反試行を IMS システム・ログに記録します。

- 無許可の端末からの入力メッセージ
- パスワードが必要なのに省略された
- 許可に必要なパスワードが正しくない
- パスワードのミススペル
- リジェクトされたサインオン
- アプリケーション・プログラムからの無許可の DL/I コマンド (CMD) 呼び出し

IMS は、メッセージを入力して違反をロギングした端末へメッセージを送信することによって無効な入力メッセージをリジェクトします。IMS システム・ログは、考えられるセキュリティー上の問題を調査するための監査証跡を提供します。IMS システム・ログ・セキュリティー違反は、X'10' ログ・レコード・タイプとして識別されます。ログは、ファイル選択・フォーマット設定印刷ユーティリティーを使用して印刷できます。

セキュリティー違反についての通知をただちに受け取れるように緊密なセキュリティーを設定することもできます。管理者は、セキュリティー違反が発生したときにマスター端末へメッセージを送信させることによって、マスター端末が違反について直ちに通知を受け取るよう準備できます。違反が発生したときにマスター端末が通知を受け取るようにするには、SECCNT 初期設定 EXEC パラメーターにゼロ以外の値を指定します。

ただし、大規模なネットワークでは、パスワード、トランザクション・コード、およびコマンドのつづりの誤りによって、きわめて多くの違反と違反通知が発生する可能性があります。管理者は、通知のしきい値を指定することにより、セキュリティー保護を回避しようとする本当の試みの証拠を提供する一方、オペレーター・エ

ラーが原因となった通知の数を減らすことができます。単一の端末からの違反の数が通知しきい値 (SECCNT 値で指定) に等しくなると、マスター端末に通知されません。

RACF がインストール済みである場合には、セキュリティー違反を記録する別の方法が使用できます。各リソース・アクセス違反ごとに、1 つの RACF タイプ 80 レコードを作成します。管理者は RACF 報告書作成プログラムを使用して、これらのレコードに基づいて報告書を作成できます。

関連資料:

- SECCNT 初期設定 EXEC パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。
- ファイル選択・フォーマット設定印刷ユーティリティーの詳細については、「IMS V15 システム・ユーティリティー」を参照してください。
- RACF 報告書作成プログラムを使用して RACF レコードの形式設定および印刷を行う方法について詳しくは、「z/OS Security Server RACF 監査担当者のガイド」を参照してください。

その他のアクセス管理方式に関する考慮事項

その他のアクセス方式を使用して、セキュリティー・ソフトウェアを強化します。

物理的セキュリティー

管理者は、システム・セキュリティーをサポートする物理的セキュリティー手段を考慮しなければなりません。

それらの手段としては、次のものがあります。

- コンピューター領域との間での被制御アクセス
- 特定の端末領域内での DP オペレーターと非オペレーターの許可
- テープ、ディスク、カード、ファイルなどのメディア用の別々に制御される領域
- コンピューター用紙と印刷出力の制御

物理的セキュリティーの必要性は動的であることが考えられるので、定期的な見直しや調整を行う価値があります。

DB/DC および DCCTL における表示バイパスおよびパスワード・マスキングの使用

IMS はパスワードを受け入れた後、それらのパスワードを端末装置表示メディアから消したり隠したりするソフトウェア機能を備えていません。しかし、メッセージ形式サービス (MFS) 機能を使用すると、非表示属性を付けたフィールドを定義できます (3270 ディスプレイ装置用)。IMS は、パスワードを、ログに記録する前にメッセージから除去します。

パスワードをトランザクション入力とコマンド入力の一部として使用する計画の場合は、非表示フィールドを組み込むための画面フォーマットを設計しなければなりません。この保護は、/SIGN コマンドの場合、特に重要です。DFS3649 サインオ


ン必須メッセージには、ACF/VTAM ディスプレイ端末でパスワードを入力するための非表示フィールドが組み込まれています。

ほとんどのキー駆動型端末は、文字を表示することなく入力できる機能 (バイパス機能と呼ばれる) を備えています。通常の場合、この機能を備えた端末は、表示モードまたはバイパス・モードで常時運用されます。セキュリティー要件をサポートするためにパスワードを隠す場合、この機能が必要となります。

バイパス機能は、パスワードだけでなくコマンド verb、コマンド、トランザクション・コード、およびテキストについての保護基準を運営上確立するのに使用できます。

パスワードや他の機密データを IMS ログ内で変更する必要がある場合、IMS はこのためのユーザー出口を提供します。

関連資料:

 ログ編集ユーザー出口 (LOGEDIT) (出口ルーチン)

リソースの保護

オンライン (DB/DC、DBCTL、DCCTL) 環境とバッチ環境の両方で、VSAM、OSAM、および高速機能の各データベースと同様に、IMS システム・ライブラリーとデータ・セットを保護できます。

IMS システム・ライブラリーとシステム・データ・セット

RACF を使用して、IMS システム・ライブラリーとシステム・データ・セットを保護することができます。IMS は RACF を呼び出し、リソースをオープンしようとするシステム・アドレス・スペース (制御領域、DLISAS、バッチのいずれか) へ関連付けられているユーザー ID が必要なアクセス許可を持っているかどうかを判別します。実際には、RACF はアクセスを許可した時点で、始動されたタスク・テーブルを通して、始動されたプロシージャー名 (IMS プロシージャーまたは DLISAS プロシージャー) へユーザー ID を関連付けます。JCL を使用して IMS を始動する場合、RACF ユーザー ID をそのパスワードと共にジョブ・カードに入れることができます。

ユーザー ID が許可を持っていない場合、アクセスは拒否されます。基本的な規則は、「DD カードを持つものは、すべて権限を持っていないなければならない」ということです。

関連資料: このプロセスについて詳しくは、「z/OS Security Server RACF セキュリティー管理者のガイド」を参照してください。

IMS プロシージャー

IMS プロシージャーが (十分な権限を持つ) RACF ユーザー ID へ関連付けられている場合、IMS 制御領域は RACF 保護データ・セットをオープンすることができます。関連付けが存在しない場合、IMS 制御領域は、要求された権限レベルに対して汎用アクセスを認めない RACF 保護データ・セットをオープンすることができません。

DLISAS プロシージャ

DLISAS プロシージャが RACF ユーザー ID へ関連付けられている場合、このプロシージャは IMS プロシージャ用に RACF ユーザー ID をオーバーライドします。関連付けが存在しない場合、IMS プロシージャへ関連付けられている RACF ユーザー ID が RACF アクセス検査に使用されます。

データベースの保護

管理者は、セグメント・レベルのセンシティブィー、フィールド・レベルのセンシティブィー、または RACF を使用して、高速機能 DEDB だけでなく VSAM および OSAM の全機能データベースを保護できます。

セグメント・レベルおよびフィールド・レベルのセンシティブィー：データベース定義、プログラム仕様ブロック、それらが常駐するライブラリーの内容に対する中央制御を通して、データへ保護属性の効果的なスキーマを割り当てることができます。ただし、PSB を通じたデータベース保護を全体として効果的にするには、(データベースへアクセスするコードを保護するため) PSB ライブラリーとアプリケーション・プログラム・ライブラリーも保護しなければなりません。

セグメント・レベルのセンシティブィー

フィールド・レベル・センシティブィーを使用していない場合、保護できるデータの最小単位はセグメントです。許可できる基本的な処置は、次のとおりです。

ありません。

セグメント・タイプへアクセスしません。

読み取り

セグメント・タイプは取り出すことだけができます。

次の追加処置の 1 つ以上を読み取りと組み合わせて許可することができます。

追加 セグメント・タイプの新しいオカレンスを挿入できます。

更新 セグメント・タイプの既存のオカレンスを置換できます。

Delete

セグメント・タイプの既存のオカレンスを削除できます。

PROCOPT キーワードのパラメーター値と PCB の指定方法は、許可を制御します。アクセス許可はプログラム・レベルで宣言されますが、許可の実施はトランザクション・コードか、データベースの個々の階層レベルに現れるようにすることができます。特定プログラムへ 1 つのトランザクション・コードだけが関連付けられている場合、アクセス許可はトランザクション・レベルで効果を持ちます。より高いレベルのセグメントの処理オプションとして、PSB で SENSEG ステートメントとキー・センシティブィーを使用することにより、個々の階層レベルでマスキングに効力を持たせることができます。

フィールド・レベル・センシティブィー

フィールド・レベルのセンシティブィーは、別の種類のデータベース・セキュリティを提供できます。データベース記述 (DBD) と PSB をコーディングして、セグメント内にあるフィールドの必須サブセットへのアクセス

を許可できます。フィールド・レベルのセンシティブティーを使用して、データベースの健全性を保証するため、フィールド・レベルの置換機能を制御することもできます。

関連資料:

- データベース・レベルのセキュリティーについては、「IMS V15 データベース管理」を参照してください。
- DBD および PSB の生成については、「IMS V15 システム・ユーティリティー」の『アプリケーション制御ブロック保守ユーティリティー』および『プログラム仕様ブロック (PSB) 生成ユーティリティー』を参照してください。

RACF を使用したデータベースの保護

実行する環境に応じて、DLISAS の RACF ユーザー ID が制御領域始動プロシージャを使用します。JCL を使用して IMS を始動する場合、RACF ユーザー ID をそのパスワードと共にジョブ・カードに入れることができます。

VSAM 全機能データベース

オンライン環境で、RACF ユーザー ID が DLISAS 始動プロシージャに関連付けられていると、その ID がアクセス検査に使用されます。RACF ユーザー ID が DLISAS 始動プロシージャに関連付けられていない場合、制御領域 RACF ユーザー ID が使用されます。(バッチ環境では、バッチ・ジョブのユーザー ID が使用されます。)
「CONTROL」のアクセス権限が必要です。データベースは、ICFCATALOG として定義されている必要があります。従来の「VSAM」カタログ・タイプを使用する場合は、制限があります。

OSAM 全機能データベース

オンライン環境では DLISAS の RACF ユーザー ID が使用され、バッチ環境ではバッチ・ジョブのユーザー ID が使用されます。

高速機能 DEDB

オンライン環境では、制御領域 RACF ユーザー ID が使用されます。(バッチ環境は存在しません。)

追加保護

RACF 内の DATABASE、FIELD、および SEGMENT のクラスにもデータベース・セキュリティーを実装できます。

関連資料: これらのリソース・クラスについて詳しくは、422 ページの『RACF セキュリティー計画の準備』を参照してください。

暗号化: アクセス制御に代わるもの

データへのアクセスを防止するのが困難であるか実際的でない場合には、ファイルのデータやネットワークで通信されているデータを暗号化によって保護することができます。IMS は一部のファイル暗号化機能を (例えば、IMS セグメント編集/圧縮出口ルーチンを通して) 提供しますが、通信暗号化機能は提供しません。

以下のリストに暗号化に関する追加情報を示します。

暗号サポートの使用

プログラム式暗号機能 (プログラム番号 5740-XY5) は、z/OS の下でファイルと通信の暗号化機能を提供します。物理階層データベースのファイル暗号化は、データベースが入っている物理ディスク・パックを通常のエリアから取り除くときに、未許可の個人がデータを見ることができないようにします。ファイル暗号化サポートは VSAM 物理データベースにも及びます。通信暗号化機能は、ACF/VTAM サポート端末をサポートします。

セグメント編集/圧縮出口ルーチンの使用 (DCCTL 以外)

このルーチンを使用して、データ暗号化を提供できます。IBM プログラム式暗号機能を出口ルーチンに組み込むと、プログラミングの手間を減らすことができます。この機能は、アセンブラー・マクロ呼び出しによって実行されます。セグメントは、データベース・バッファ・プールへ置かれる前に暗号化されます。IMS DBDGEN の SEGM 制御ステートメントは、この出口ルーチンの名前を指定するためにキーワードを組み込みます。

ICSF/CCA インターフェースの使用

ICSF/CCA API を IMS DB セグメント編集/圧縮出口で使うことができます。IMS では、プログラム式暗号機能 (PCF) インターフェースが、ICSF/CCA インターフェースを通じて透過的にサポートされています。PCF インターフェースのために作成されたプログラムは、変更せずに、ICSF/CCA インターフェースを通して実行できます。しかし、PCF プログラムで ICSF/CCA API を使用したい場合は、これらの PCF プログラムに変更が必要です。

ICSF/CCA インターフェースには、以下の 2 つの PCF 互換モードがあります。

- ICSF モード COMPAT(YES) は、プログラム式暗号機能インターフェースのために作成されたプログラムを変更せずに実行でき、呼び出しは直接 ICSF/CCA API に対して行われることを意味します。このモードでは、動的マスター・キーの変更にいくつか制限があります。
- ICSF モード COMPAT(NO) は、CCA API のためにコーディングされたプログラムのみが実行されることを意味します。

DBCTL 環境のセキュリティーに関する考慮事項

このトピックでは、IMS DBCTL 環境のセキュリティーを確立する方法を説明します。

関連概念:

428 ページの『CSL RM アドレス・スペースでの IMSRSC リポジトリのセキュリティーの設定に関する考慮事項』

保護できる DBCTL リソース

保護 IMS システムの設計のためにどのセキュリティー機能を使用するかを決める前に、システム内のどのリソースに保護が必要であるかを知っておかなければなりません。言い換えれば、どのように保護するかを決める前に、何を保護するかを決める必要があります。

DBCTL 環境では、以下のリソースを保護できます。

制御領域

オンライン・アプリケーション・プログラムが端末を通してデータベースを処理できるようにする IMS システム制御領域。

システム・データ・セット

IMS オンライン・システムの操作の基礎となるデータの集合。例えば、IMS.PROCLIB データ・セット。

従属領域

IMS オンライン・システムのストレージ域。ここでバッチまたはオンラインのアプリケーション・プログラムが実行されます。DBCTL 環境では、従属領域は、BMP 領域、JMP 領域、または CCTL 領域とすることができます。

PSB プログラム仕様ブロック。オンライン・アプリケーション・プログラムが使用する階層型データベース・グループと、論理メッセージの宛先を記述した制御ブロック。

BMP アプリケーション・プログラム

ユーザーの作業を実行する DBCTL 環境のプログラム。バッチ・メッセージ・プログラムは、領域が JCL によって始動された後、従属領域制御プログラムによってアクティブにされます。

データベース

ユーザーのアクティビティの基礎となるデータの集まり。プログラム連絡ブロック (PCB) を使用して、プログラムは、IMS 物理データベース設計によって記述されるとおりのデータベースの論理ビューを持ちます。

システム定義時に行われる DBCTL セキュリティーの選択

初期設定 EXEC パラメーターまたはシステム定義マクロ IMSGEN を使用して、IMS セキュリティーの選択を行います。リソース・アクセス管理機能 (RACF) を使用して、決定したセキュリティを実装することができます。

マクロまたは初期設定 EXEC パラメーターの指定を使用して、オンライン実行でアクティブにするセキュリティのタイプを選択します。リソース・アクセス管理機能 (RACF) を使用して、リソースを指名します。

DBCTL リソース用のセキュリティ機能およびセキュリティ・タイプ

特定の DBCTL リソースを保護するために使用するセキュリティ機能は、必要なセキュリティのタイプおよび、そのセキュリティのタイプに対して使用可能なセキュリティ機能によって決まります。DBCTL リソースに関するセキュリティ機能には、z/OS、RACF、および PSBGEN があります。

以下の表は、DBCTL リソースに対して使用可能なセキュリティのタイプとセキュリティ機能を要約したものです。

表 32. DBCTL リソース、セキュリティ・タイプ、およびセキュリティ機能：

リソース	セキュリティのタイプ	セキュリティ機能
システム・データ・セット	OS パスワード保護	z/OS
	データ・セット保護 (VSAM)	RACF

表 32. DBCTL リソース、セキュリティ・タイプ、およびセキュリティ機能 (続き):

リソース	セキュリティのタイプ	セキュリティ機能
データベース	セグメント・センシティブティ	PSBGEN
	フィールド・センシティブティ	PSBGEN
	パスワード・セキュリティ (/LOCK および /UNLOCK コマ ンド用)	RACF 注 ¹ を参照
PSB	リソース・アクセス・セキュリテ ィ	RACF
	APSB セキュリティー	RACF
BMP アプリケーション・ プログラム	パスワード・セキュリティ (/LOCK および /UNLOCK コマ ンド用)	RACF
	拡張リソース保護 (APPL キーワー ドを使用)	RACF
制御領域	拡張リソース保護 (APPL リソー ス・クラスを使用)	RACF
従属領域	RAS セキュリティー	RACF
	APSB セキュリティー	RACF
	リソース・アクセス・セキュリテ ィ	RACF

注:

1. /LOCK PGM および /LOCK DB コマンドに関する RACF 権限は、DBCTL のみのシステム (システム定義プロセス中の IMSCTRL マクロの SYSTEM パラメーターに DBCTL が含まれている、または IMS を DBC プロシージャで始動した場合) では検査されません。

DBCTL セキュリティーの設計に関する考慮事項

このトピックでは、IMS セキュリティーのさまざまな選択項目の使用法について説明します。セキュリティ設計の個々の部分について判断する場合は、システムへのアクセスを入手するためにエンド・ユーザーが実行しなければならない物理処置を考慮してください。多くの場合、複数のタイプのセキュリティ検査が使用されます。

このトピックでは、以下を想定しています。

- 制御点としてのユーザー ID
- 制御点としてのマスター端末
- RACF 保護の使用
- 制御点としての領域の使用

コマンド・キーワードによるパスワード保護の使用

コマンドが受け入れられる前に検証を提供するため、随伴するパスワードが必要となる場合があります。このパスワードは、コマンド verb の直後の括弧内に入力します。

従属 BMP、JBP、または CCTL 領域からのアクセスの制限

従属 BMP、JBP、および CCTL 領域は、保護が必要なリソースです。それらを保護するには、タスク開始 JCL によって無許可の従属メッセージ領域の始動を防止し、従属領域内での無許可のリソースの使用を防止します。

RAS を使用した DBCTL 従属領域の保護

RAS は、従属領域のユーザー ID に基づいて、リソースの使用を制限します。RAS は、セキュリティの制約に RACF を使用します。

アプリケーション・プログラムが実行中の従属領域のユーザー ID が PSB について許可されている場合、RAS はアプリケーション・プログラムがその PSB にアクセスできるようにします。PSB を使用するユーザー ID の権限は、IIMS または JIMS RACF セキュリティー・クラスのどちらかで定義されます。RAS は、従属領域でスケジューリングできるアプリケーション・プログラムを制限しません。

IIMS および JIMS クラスは、RACF によって事前定義されていますが、ご使用のリソースの RACF に付属していない場合は、RACF ICHERCDE マクロを使用して、それらを新しいクラスとして定義できます。

RAS は、初期設定 EXEC パラメーターで ISIS=R を指定することによって指定できます。

関連資料:

- ISIS= EXEC パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。
- RACF セキュリティーとの連動について詳しくは、以下を参照してください。
 - 422 ページの『RACF セキュリティー計画の準備』
 - *z/OS Security Server RACF セキュリティー管理者のガイド*
- ICHERCDE マクロについて詳しくは、「*z/OS Security Server RACF マクロおよびインターフェース*」を参照してください。
- RACF リソース・クラス記述子テーブルの更新について詳しくは、「*z/OS Security Server RACF システム・プログラマーのガイド*」を参照してください。

高速機能アプリケーション・プログラムのセキュリティに関する考慮事項

高速機能アプリケーション・プログラム用または高速機能データベースへアクセスする DL/I プログラム用にセキュリティを設計する場合は、従属領域内のプロセスを RAS セキュリティーによって保護できることを考慮してください。

ODBA アプリケーション・プログラムのセキュリティ

z/OS システム許可機能 (SAF) と、RACF などの外部セキュリティ製品を使用して、ODBA アプリケーション・プログラムからの APSB 呼び出しで指定されているすべての PSB を保護することができます。

ODBA アプリケーション・プログラムのセキュリティについて詳しくは、409 ページの『ODBA アプリケーション・プログラムのセキュリティ』を参照してください。

IMS DBCTL セキュリティーの活動化

このトピックでは、RACF およびプログラム出口ルーチンを使用して、IMS セキュリティー設計を活動化するためのステップについて、その指針を示します。

使用するセキュリティ機能に応じて、管理者は適切な作業を行う必要があります。

DBCTL における RACF セキュリティー計画を準備します。

RACF セキュリティー計画を実施するには、次のように行います。

1. 保護したいすべての IMS オンライン・リソースのリストを準備します。その際、対象となるリソース全体を概観できるようグループに分けて配置します。
2. リソース・グループを保護するセキュリティ機能を選択します。
3. IMSGEN マクロまたは初期設定 EXEC パラメーターをコーディングします。
4. RACF に対してリソース・クラス・プロファイルを記述します。
5. RACF にユーザー、グループ、データ・セットを追加します。
6. IMS.PROCLIB の JCL プロシージャーを修正します。

RACF リソース・クラスは IMS セキュリティー機能によって使用されます。PSB クラスは、PSB セキュリティーのプロファイルを保持します。RACF が事前定義済みリソース・クラスを提供します。あるいは、管理者が独自のものを定義できます。使用されるリソース・クラスの名前は、RCLASS= パラメーターで指定します。表 33 は、DBCTL に対するリソース・クラスの割り当てを示しています。

表 33. DBCTL に対するリソース・クラスの割り当て

リソース・クラス	リソース・クラス命名規則	
	RACF 定義名	ユーザー定義名
PSB リソース・クラス	IIMS	Ixxxxxxx
PSB グループ リソース・クラス (resource class)	JIMS	Jxxxxxxx

RACF リソース・クラスは、RACF のリソース・クラス記述子テーブル (CDT) に定義されます。最初は、IIMS、および JIMS リソース・クラスが CDT に事前定義されています。リソース・クラスを追加したり、ユーザー定義名でリソース・クラスを定義するには、RACF リソース・クラス・マクロ ICHERCDE を使用して、インストール先定義 CDT を作成する必要があります。

関連資料: RCLASS EXEC パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。

DBCTL セキュリティー用のシステム始動の制御

DBC プロシージャー用に生成された値は、すべて「セキュリティなし」を指定します。管理者は、それらをリセットして、セキュリティを使用可能にする必要があります。

制御領域用の ISIS 実行パラメーターには、現在の実行の過程で行われるセキュリティ検査の種類を制御する方法が含まれています。このパラメーターは、DBCTL オペレーターがセキュリティ検査の選択をどの程度の柔軟性を持ってオーバーライドするかを決めます。管理者は ISIS= キーワード・パラメーターの設定を、全体的なセキュリティ設計と操作手順の両方と調整しなければなりません。

ISIS キーワード・パラメーターを使用すると、必要な従属領域セキュリティのタイプ (RAS) と、そのセキュリティを実施するセキュリティ機能 (RACF、または出口ルーチン) を選択できます。選択は、ISIS= パラメーターを使用して指定します。ISIS= のパラメーター・オプションは、次のとおりです。

ISIS=0|N

RAS セキュリティを使用不可にします。

ISIS=1

RACF を使用して RAS セキュリティを指定します。ISIS=1 の指定は、ISIS=N の指定と同じ結果になります。

ISIS=2

出口ルーチンを使用して RAS セキュリティを指定します。ISIS=1 の指定は、ISIS=N の指定と同じ結果になります。

ISIS=R

RACF を使用して RAS を指定します。

ISIS=C

出口ルーチンを使用して RAS を指定します。

ISIS=A

RACF と出口ルーチンの両方を使用して RAS を指定します。

CCTL 用に RACF を使用する場合は、CCTL JOB ステートメントに USER 指定が含まれていなければなりません。CCTL 自体は、このセクションで説明した DBCTL セキュリティ検査とは別に、セキュリティ検査 (RACF またはそれ独自の) を提供できます。

オンラインでの DBCTL セキュリティ変更の実装

システム定義についてオンライン変更を使用する場合、RACF のセキュリティ定義を更新する必要があります。

システム定義へのオンライン変更の結果、以下のステップを実行して、セキュリティ指定を現行にしておく必要があります。

- パスワードを更新する
- パスワード・セキュリティ用にオンライン・システムにセキュリティ機能を追加する
- RACF データベースをリフレッシュする

RACF セキュリティ定義をオンラインで変更するには、RACF データベースを更新します。次に、RACF コマンド SETROPTS RACLIST(*classname*) REFRESH を出します。

DBCTL セキュリティー違反の制御

セキュリティ違反は、インストール・システムのセキュリティ管理の指針に従って処理されます。

IMS は、次のセキュリティ違反試行を IMS システム・ログに記録します。

- パスワードが必要なのに省略された
- 許可に必要なパスワードが正しくない
- パスワードのミススペル

これらのエラーのいずれかがあった場合、IMS はログに違反を記録します。IMS システム・ログは、考えられるセキュリティ上の問題を調査するための監査証跡を提供します。IMS システム・ログ・セキュリティ違反は、X'10' ログ・レコード・タイプとして識別されます。ログは、ファイル選択・フォーマット設定印刷ユーティリティを使用して印刷できます。

RACF がインストール済みである場合には、セキュリティ違反を記録する別の方法が使用できます。各リソース・アクセス違反ごとに、1 つの RACF タイプ 80 レコードを作成します。

関連資料:

- ファイル選択・フォーマット設定印刷ユーティリティの詳細については、「IMS V15 システム・ユーティリティ」を参照してください。
- RACF レコードのフォーマット設定および印刷を行うユーティリティについて詳しくは、「z/OS Security Server RACF 監査担当者のガイド」を参照してください。

第 20 章 IMS の制御

IMS の制御は、多数の作業からなっています。

それらの作業について、以下のトピックで説明します。

関連資料: 詳細については、「IMS V15 オペレーションおよびオートメーション」を参照してください。

システムのモニター

定期的なスケジュールでシステムの状況をモニターし、問題判別とパフォーマンスに関する情報を収集します。

例えば、追加のメッセージ領域を開始する必要の有無を判別する場合は、ピーク・ロード時のキューの状況をモニターします。

関連資料: モニターについて詳しくは、465 ページの『第 24 章 IMS モニター・データの収集および解釈』を参照してください。

分析のための IMS システム・ログ情報の処理

システム・ログ・データ・セットは、オンライン・システムが行う処理に関する統計の基本的なソースです。個々のログ・レコード・タイプには、さまざまな方法で分析することができるデータが入っています。

例えば、指定されたユーザー ID に関するすべてのアクティビティ、または IMS プールに関するすべてのアクティビティを選択およびフォーマット設定することができます。

IMS システム・ログ・ユーティリティーの使用

IMS は、システム・ログからのログ・レコードの抽出を支援する、いくつかのユーティリティーを備えています。これらのユーティリティーは、複数のログ・データ・セット全体に広がったデータの削減やマージの支援も行います。

以下のセクションでは、これらのユーティリティーについて説明します。

- ファイル選択・フォーマット設定印刷ユーティリティー

メッセージ・セグメントまたはデータベース変更アクティビティを詳細に検査したい場合は、ファイル選択・フォーマット設定印刷ユーティリティー (DFSERA10) を使用できます。このユーティリティーは、OLDS や SLDS、または CQS ログ・ストリームに含まれるログ・レコードの内容を印刷します。各ログ・レコードは、1 つまたは複数の 32 バイトのセグメントで表されます。印刷される出力では、各セグメントを文字フォーマットと 16 進フォーマットの両方で表示します。

- 高速機能ログ分析ユーティリティー

高速機能ログ分析ユーティリティ (DBFULTA0) は、IMS システム・ログに記録されたデータに基づいて、高速機能についての統計報告書を準備するときに使用します。

これらの報告書は、システムのインストール、チューニング、および障害追及に役立ちます。高速機能ログ分析ユーティリティは、IMS モニターやログ・トランザクション分析ユーティリティとは関連がありません。IMS モニターについての詳細は、480 ページの『IMS モニター』を参照してください。

- ログ・トランザクション分析ユーティリティ

IMS DB/DC または DCCTL 環境では、ログ・トランザクション分析ユーティリティ (DFSILTA0) を使用して、システム・ログ内のレコードに基づいて個々のトランザクションに関する情報を収集できます。このユーティリティで作成されたログ分析報告書には、合計応答時間、入力キューでの時間、処理時間、および出力キューでの時間など、多数のイベントが表の形で含まれます。

- 統計分析ユーティリティ

IMS DB/DC または DCCTL 環境においては、IMS 統計分析ユーティリティ (DFSISTS0) を使用していくつかの要約報告書を作成することができます。これらの報告書は、システムの実際のトランザクション・ロードや応答時間を獲得する場合に使用することができます。作成される統計は、入力システム・ログ・データ・セットによって異なります。

IMS データベース生産性向上ツール

IBM は、IMS データベース生産性向上ツールを多数提供しています。IMS ツールは、ご使用の IMS 環境用のデータベース・パフォーマンス拡張機能のセットです。

これらのツールは、ご使用の IMS ユーティリティのオペレーションの自動化と高速化に役立ちます。これらはまた、ご使用の IMS データベースの分析、管理、リカバリー、および修理を支援します。

関連資料: これらのツールの詳細については、DB2 および IMS ツールの Web サイト (<http://www.ibm.com/software/data/db2imstools/>) にアクセスしてください。

システムのリカバリー

DRD を使用する場合、なおかつリソース定義データ・セット (RDDS) からコールド・スタートを行う場合は、RDDS にリソース定義をエクスポートしてあることを確認してください。

動的に行われたすべてのリソース変更は、コールド・スタートを通して確実にリカバリーされるようにする必要があります。

IMS を実行しているとき、IMS システム・プログラマーまたはオペレーターは、システムをリカバリーする作業を完了する必要がある場合があります。

関連概念:

🔗 動的リソース定義の概要 (システム定義)

関連タスク:

🔗 リカバリー関連機能の実行 (オペレーションおよびオートメーション)

システム・リソースの変更および制御

IMS リソースの初期設定は、IMS システム定義プロセスの一環として行います。各種システム・リソースの変更は、MTO、およびその許可を受けた他のオペレーターが IMS コマンドを使用して行うことができます。

以下の IMS システム・リソースの作動状態は、タイプ 1 コマンドを使用して変更および制御できます。

- 従属領域
- 通信回線
- 端末
- トランザクション
- データベース
- ISC ユーザー (サブプール)
- ETO ユーザー
- MSC リソース
- セキュリティー・オプション
- 会話
- サブシステム

さまざまなタイプのリソースに類似の制御機能を実行するために、多数の IMS コマンドを使用することができます。

そのリソースを以下に示します。

- 通信回線、物理端末、またはノード・リソース
- 論理端末リソース
- 論理リンク・リソース
- 論理リンク・パス・リソース
- トランザクション・リソース
- トランザクション・クラス
- プログラム・リソース
- データベース・リソース
- サブシステム・リソース

「IMS V15 オペレーションおよびオートメーション」に、これらのコマンドとリソースの間関係が示されており、一連の具体的な疑問に対する回答が記載されています。例えば、コマンドが発行された後、リソースは以下のことができるかなどの疑問です。

- 入力を受信できるか

- 出力を送信できるか
- 出力メッセージのキューイングを実行できるか

以下の環境では、タイプ 2 コマンドを使用して IMS リソースを変更および制御することもできます。

- Operations Manager (OM) と Structured Call Interface (SCI) を含んでいるが、Resource Manager (RM) を含んでいない最小の共通サービス層を使用して構成されている単一 IMS の IMSplex。この構成は、拡張コマンド環境とも呼ばれます。
- 完全な CSL (OM、SCI、および RM) を使用して構成されている複数 IMS の IMSplex。

関連資料:

- システム定義プロセス (バッチおよび動的) および最小 CSL を使用した単一 IMS の構成の詳細については、「IMS V15 システム定義」を参照してください。
- IMS コマンドの詳細については、「IMS V15 コマンド 第 1 巻: IMS コマンド A-M」および「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

データ共有の制御

データ共有を制御する場合は、DBRC を使用し、データ共有システムをモニターする必要があります。

DBRC を使用したデータ共有の制御

DBRC を使用することで、データ共有に含まれる IMS サブシステムによるデータへのアクセスを制御できます。DBRC を使用すると、RECON データ・セット内の現行状況表示を修正、開始、および削除して、オンライン IMS サブシステムのアクセス意図および登録されたデータベースの共有レベルを変更できます。

データ共有環境は、RECON データ・セット内に示されたデータベースおよびサブシステムの状況によって異なります。

関連資料: DBRC をデータ共有にどのように関与させるかについて詳しくは、641 ページの『第 35 章 データ共有のサポート』を参照してください。

DBRC オンライン変更コマンドの 1 つ、/RMCHANGE の形式を使用して、共有レベル標識を修正できます。

データ共有システムのモニター

データ共有をモニターするためには、IRLM、IMS サブシステムおよびデータベース、RECON データ・セットの状況に関する情報を取得します。また、IMSplex 内のデータ共有については、データ共有に関与しているカップリング・ファシリティーの構造に関する情報も取得します。

操作:

- 使用しているシステムまたは別の接続したシステム上の IRLM の状況を表示するには、次の z/OS コマンドを入力します。

```
MODIFY irImproc,STATUS,irImx
```

関連資料: IRLM のコマンドのすべての説明については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

- データ共有環境でコンポーネントとリソースをモニターするには、非データ共有環境の場合と同じタイプの手順を使用します。モニターに関する詳細については、465 ページの『第 24 章 IMS モニター・データの収集および解釈』を参照してください。
- カップリング・ファシリティ上での構造をモニターするには、次の z/OS オペレーター・コマンドを使用します。
 - DISPLAY XCF,STRUCTURE: アクティブ・ポリシー内で定義されている構造の状況を表示するために使用します。
 - DISPLAY XCF,STRUCTURE,STRNAME=: 特定の構造の詳細を表示するために使用します。IMS にとって重要な構造は、IRLM、OSAM、VSAM、共有 MSGQ、および共有 EMHQ 構造です。

これらのコマンドを使用することで、カップリング・ファシリティ上での構造を表示し、リソース状況を判別し、障害の場合には、問題判別の情報を収集します。

関連資料: これらのコマンドについて詳しくは、「z/OS MVS システム・コマンド」を参照してください。

ログ・データ・セットの特性の制御

時折、ログ・データ・セットの特性を調整し、修正する必要があります。

例えば、以下の環境のような場合は、ログ・データ・セットの特性を調整し、修正する必要があります。

- モニター後
- システム使用可能性、保全性、またはオペレーター操作のユーザー要件の変更後

OLDS、WADS、SLDS、および RECON データ・セットの制御の詳細については、「IMS V15 オペレーションおよびオートメーション」を参照してください。

サブシステム接続および切断

IMS サブシステムから外部サブシステム (z/OS アドレス・スペース内で実行されている別のプログラム)、例えば DB2 などにアクセスするには、前もって IMS サブシステムをその別のサブシステムに接続しておく必要があります。IMS が別のサブシステムに接続できるのは、そのサブシステムが IMS.PROCLIB のサブシステム・メンバーで識別される場合のみです。

SSM EXEC パラメーターに、サブシステムのメンバー名を指定します。IMS への指定がされていて、IMS によってアクセスされているときは、IMS を停止せずにサブシステム・メンバー名を変更することはできません。

関連資料: SSM EXEC パラメーターについては、「IMS V15 システム定義」を参照してください。IMS と別のサブシステムとの接続について詳しくは、「IMS V15 オペレーションおよびオートメーション」を参照してください。

第 21 章 IMS の開始または再始動

IMS サブシステムの開始または再始動とは、IMS 制御領域を初期設定することを意味します。

操作環境を復元するためには、MTO は以下のものも確実に初期設定されるようにする必要があります。

- すべての IRLM
- すべての CQS
- あらゆる従属領域 (ユーティリティーも含む)
- すべての CSL アドレス・スペース (OM、RM、および SCI)、ただし CSL を使用している IMSplex として操作する場合
- VTAM へのあらゆる接続
- すべての通信回線および端末

ここで使用されている MTO という用語は、は、DB/DC 環境および DCCTL 環境の場合は IMS マスター端末オペレーターを指し、DBCTL 環境の場合はオペレーターを指します。表示されているコマンドはすべて、コマンド認識文字の最初がスラッシュ (/) であることを前提としています。

追加情報については、IMS V15 オペレーションおよびオートメーションを参照してください。

第 22 章 IMS のシャットダウン

データ通信や従属領域がまだ普通の方法で終了していない場合は、制御領域のシャットダウンに使用するコマンドによって、それらも強制的に終了し、かつ CQS サブシステムにシャットダウンを通知することもできます。

オンライン・システム全体をシャットダウンする共通のシーケンスを、以下に示します。

1. IMS DB/DC または DCCTL 環境では、データ通信を停止します。IMS DBCTL 環境では、CCTL を切断します。
2. 従属領域を停止します。
3. 動的リソース定義 (DRD) が使用可能で、AUTOEXPORT が使用可能でない場合、まだエクスポートされていない変更済みのランタイム・リソース定義および記述子定義をすべてエクスポートします。
4. 制御領域を停止します。
5. IMS DB/DC または DBCTL 環境では、IRLM を停止します。
6. 共用キュー環境では、CQS がシャットダウンされていなければ、それをシャットダウンします。
7. CSL を使用している IMSplex システムでは、CSL マネージャー・アドレス・スペース (OM、RM、および SCI) をシャットダウンします。

関連タスク:

 IMS のシャットダウン (オペレーションおよびオートメーション)

第 23 章 システムのテスト

システム管理者は、2 段階のテストを行う必要があります。第 1 段階のテストは、アプリケーション・プログラムが実動モードへのサービスインよりも前の時点で検査される場合に行われます。第 2 段階のテストでは、既存のアプリケーション・プログラムの変更または修正内容を評価して、アプリケーションの機能が退行していないこと確認し、新規機能または変更された機能を検査します。

複数のアプリケーション・プログラムがある場合は、あるアプリケーション・プログラムの修正が、エンド・ユーザーに提供されるサービス機能に影響しないことを確認してください。

さまざまなテスト段階、および各段階に対応する管理タスクと関連開発アクティビティを、以下の表に示します。

表 34. テスト段階に関連する管理タスク

テスト段階	管理タスク	関連開発アクティビティ
単体テスト	ボトルネックの識別	開発コードのテスト
機能テスト	操作手順の計画	明示的な機能のテスト
統合テスト	テスト・システム定義の作成	システムの構築
	テスト・データベースの計画	テスト・データベースの構築
コンポーネント・テスト	ネットワーク運用の検査	アプリケーション・プログラムの主要論理部分の検査
システム・テスト	操作とリカバリー手順のチェックアウト	完全実行可能システムの構築
	テスト・ツールの使用とモニターの調整	操作手順の妥当性検査
	ネットワークの作動可能性の確認	共存のテスト
パフォーマンスのテスト	シミュレーションの計画	クレームの検査とベンチマークの設定
ストレス・テスト	ピーク・ロードと応答基準の計画	大容量トラフィックと大容量プロセスのテスト
受け入れ検査	操作手順の完成	エンド・ユーザーを代表するユーザー連絡グループによるチェックアウト
保守テスト	制御ライブラリーとオンライン定義	アプリケーションと IMS サービスのチェックアウト
設計上の変更内容のテスト	管理タスクでの応答の計画	変更内容の検証の管理
レグレッション・テスト	モニターの計画	古い機能に影響が出ていないことの確認

テスト・システムの必要性

テスト・システムの開発には、各分野の担当者が参加する必要があります。

システムの実現の最終工程では、エンド・ユーザーへの十分なサービスの提供が主な課題となります。IMS システムは予想どおりに機能していますか？次の課題となるのは、1 つ以上のオンライン・アプリケーション・プログラムを変更した場合にすべてのエンド・ユーザーに対する保全性とサービス機能が保持されるということです。変更によって実稼働環境に重大な影響が現れますか？

インストール・システムでは、個別のテスト組織が存在することがあります。この場合、開発担当者は実行コードをテストしませんが、アプリケーション・プログラムを独自のテストに移す前に、このプログラムのデモンストレーションを行う必要があります。

業務上の要件を理解しているユーザー関連グループがエンド・ユーザーの代表となります。ユーザー関連グループは、アプリケーション・プログラムの妥当性と正確度を評価します。

テスト・システムの必要性は、次のような様々なシステムにより解決されます。

- 主要な開発作業中に運用可能な個別の IMS テスト環境
- 保守の変更とアプリケーション設計上の変更の進行検査に使用される個別の IMS テスト・システム
- 制御された変更内容をオンラインでテストできる実動システム。このテストでは、実動プロセスに影響を与えない時点でテスト用に初期化された領域が使用されます。

管理作業の 1 つとして、適切なオンライン IMS テスト・システムが作成されており、テスト・プロシージャと特殊なデータベース要件がきちんと記録されていることを確認することがあります。テスト・システムが保守システムとしても機能する場合、この確認は重要です。ただし、管理者の役割はもっと受動的であり、操作手順の確認のためのテストと、実動モードへの準備とシステム定義がすべて完了していることの確認のためのテストに参加するだけになります。

アプリケーション・パッケージが実動段階に入ると、開発担当者やテスト担当者の作業は、通常、プログラム保守担当者が行うこととなります。管理者は、保守作業が既存のオンライン IMS の設計に及ぼす影響を評価する必要があります。この作業についての詳細は、501 ページの『第 25 章 システム設計の変更』で説明します。

テスト・システムの設定

管理タスクの 1 つに、オンライン IMS システムへの変更内容を適用するためのプロシージャを定義することがあります。これには、テストに参加する担当者全員が実行する受け入れプロシージャを確立することが含まれます。

個別のテスト・システムを確立する際の主要な考慮事項は次のとおりです。

- データベースの保全性保護に関するインストール・システムでの方針
- アプリケーションを適切にテストされたものとして受け入れるための基準の存在
- 安定した実稼働環境の必要性

- 日常の問題を解決する際にどの程度のサポートを使用できるか
- アプリケーションの複雑度
- 現行のプログラムがどの程度正常に稼働できるか、または途中で発生する問題の迂回機能がどの程度機能するか

テスト・システムを継続実行機能として使用するかどうかを決定する上で考慮すべき項目は次のとおりです。

- 新規アプリケーション・プログラムまたは追加機能の実現作業が段階的に行われるか？
- テスト中に検出されたアプリケーションの問題を修正するために発生するオンライン・サービス割り込みに耐性があるかどうか、耐性がある場合はどの程度の耐性なのか？
- システムの変更内容または修正内容を受け入れるためのプロシージャは？
- IMS システム自体に対する保守をどのように行うか - テスト・システムを使用するか？
- アプリケーション・プログラムに対する訂正と IMS の問題修正を、要求に応じて組み込むか、または、合意された周期で一括して組み込むか？

テスト・データベースの設定

テスト・データベースには、システムを十分にテストできるだけのデータが入っていないければなりません。既存の機能の退行を防ぎ、データ使用量を節約するために、このデータによってアプリケーション・プログラムの主要論理部分をテストできます。

通常、テスト・トランザクション・ストリームはすでに判明しているデータベース状況に対して実行され、その実行結果は、あらかじめ予期された結果と比較されます。トランザクションの内容と、そのデータベース状況の予期された結果は、ユーザー関連グループと、データベース設計を担当するグループによってその正確度が分析されます。テスト・ケースが追加される場合は、さらに冗長度について検証されます。既存のトランザクションへテスト・ケースを追加するか、あるいはデータベースの他のデータをさらに必要とする追加トランザクションを定義します。この追加と定義を同時に実行することはできません。テスト・ケース・ストリームは、その処理内容や必要なデータが分かるように、適切に文書化されていなければなりません。

関連資料: テスト・データベースの作成について詳しくは、「IMS V15 データベース管理」を参照してください。

操作手順のテスト

テスト・システムを使用して、操作手順の正確度と使いやすさを確認することができます。

次の作業はシステム・テスト段階で行います。

- 準備段階の MTO プロシージャ、ラン・ブック、機能不良報告書の書式をテストできます。MTO はシステムを始動し、端末とノードのサブセットを接続して、再始動アクションを試行します。

- コマンド使用状況の簡単な監査を行って意見の相違について話し合い、オペレーターへの指示内容を改善します。
- オンライン実行時に、イメージ・コピーの取得、IMS モニターとトレース機能の呼び出し、アプリケーション・プログラムの異常終了に対する応答など、MTO の存在を必要とするイベントを調整します。
- リカバリー手順をテストします。オペレーターにデータベース・リカバリーを実行させて、システム・ロギングの制御およびリカバリーのユーティリティが正しく使用されていることを確認してください。DBRC のテスト用 RECON データ・セットを作成し、指定のチェックポイント位置までリカバリーする GENJCL ステップを実行することができます。

テスト・システムをアプリケーション・パッケージの変更内容のチェックアウトに使用している場合は、重要な操作変更も検証する必要があります。この検証には、MTO も監視員として参加することがあります。

IMS テスト環境でのモニター

テスト段階でのモニターの目的は、実動システムのモニターの目的とは多少異なります。

計画を立てておく必要のある 3 つの作業を次に示します。

- 実動モードに入る前に、発生する可能性のあるパフォーマンスの問題を検出して訂正する。

実際に使用するデータを使用すると、アプリケーション・プログラムに処理のパターン (特に DL/I 呼び出しの出現) を認識させることができます。モニター結果を、あらかじめ予期されたランザクシオン・プロファイルと比較してください。過度に発生する入出力イベントや長い入出力待機時間などのような項目によって、初期の段階でパフォーマンスの問題を明らかにすることができます。

- 操作手順のモニター部分をテストする。

出力の作成方法と、報告書の形式および内容について理解します。出力された報告書に適当にコメントを書き込むのではなく、モニターによる検出事項をまとめておく作業記録用紙を作成してください。

関連資料: 詳細なレポートの説明と、モニター戦略の考慮事項およびツールについて詳しくは、465 ページの『第 24 章 IMS モニター・データの収集および解釈』を参照してください。

- 新しいプログラムのプロセスの基本プロファイルに、モニター・ツールを使用する。

検出された事項について、開発担当者およびパフォーマンスの専門家と話し合ってください。パフォーマンスの問題に関する初期段階での警告事項を見つけるようにしてください。既存のオンライン IMS システムに新規アプリケーションを統合している場合は、追加アプリケーションの作業負荷による影響を事前に評価するようにしてください。

テスト段階では、スタンドアロン処理または調整処理を実行して、重大なランザクシオンの基本プロファイルを確立することができます。この場合、IMS モニター

の呼び出し要約報告書が役立ちます。 トランザクションのテスト結果を、実動システムで既に使用されている類似したトランザクションと比較することもできます。

DB/DC 環境でのモニター

DB/DC 環境では、IMSASAP II の報告書作成機能を使用して、アプリケーション・プログラムが発行した呼び出しの正確なシーケンスに関する情報を取得できます。

このツールは、オンライン IMS モニターを使用してモニター・トレース・レコードを作成します。モニター・ツールとしての使用法とインストール前提条件については、465 ページの『第 24 章 IMS モニター・データの収集および解釈』で説明します。

ネットワークの作動可能性の確認

実動モード向けにアプリケーション・パッケージを準備する場合は、オンライン IMS システムで使用できる予定のすべての端末または接続装置の状況について知っておいてください。

- 各装置と制御装置の詳細な実現計画を開発してください。この計画の項目には、次のようなものがあります。
 - 装置のタイプ、モデル、操作特性。
 - コンポーネント・セットの構成。
 - 装置の物理的な設置場所と設置担当者の名前。
 - オンライン IMS システムが装置を識別する名前。LTERM、回線番号、装置アドレス、ノード名など。
 - VTAM MODEENT マクロの PSERVIC パラメーター。
LUNAME、LUTYPE、TS プロファイル、32xx モデル、3270 画面サイズおよび NTO 装置タイプ
 - ETO で定義された端末とユーザーのログオン記述子、ユーザー記述子、MSC 記述子。
 - 適切な出口ルーチン。
 - 出力装置の場合は、給紙装置元と出力配布の配置。
 - すでに IMS の外部で運用中の場合は、テストのための装置の使用に関する制限事項。
 - 装置に対するシステム定義要件と、予定されている生成のタイミング。
 - 装置がプログラマブルの場合は、IMS の実行にローカルまたはホストに必要なプログラム。
- システムおよびネットワーク定義のための準備作業で作成された端末プロファイルを使用します。オンライン IMS システムではかなりの数の装置を操作できるため、この情報が必要になります。また、インストール担当者およびデータ通信の専門家（データ通信は通常 IMS 装置サポートの範囲外になります）と、ハードウェアについて話し合ってください。
- 静的に定義された VTAM 端末の状況について、システム定義の担当者と連絡をとってください。定義イベントは、通常、アプリケーション・パッケージを実装するマスター・プランの一部としてスケジュールされています。

DB/DC 環境および DCCTL 環境でのネットワーク・テスト

実動モードへの準備作業として、システム・テスト中にネットワークの各部分のオンライン・テストを調整します。IMS システムでアクティブになる各種端末タイプのオンライン・テストを監視してください。

クリティカルと想定されているトランザクションも監視してください。この監視結果は、リモート端末の操作手順とマスター端末のオペレーターへの指示を作成する場合に役立ちます。イベントの物理的なアクションとシーケンスに重点を置くことにより、これらの手順を明確にし、エンド・ユーザー向けにより完全な形にすることができるようになります。

端末操作の実際の手順の初期ドラフトを使用できない場合は、準備されたセッションのスクリプトを使用するのが最もよい方法です。テキストには、コマンドのテスト・シーケンスを記述したシナリオがオンライン・セッションとして含まれていなければなりません。

1. SNA 端末サブシステム・プログラムを伝送するための準備を行い、(実行可能であれば) ホストからこのプログラムを伝送する。
2. ACF/VTAM またはホスト・サブシステム・アプリケーション・プログラムを開始し、IMS を始動して VTAM との通信を確立する適切なコマンドを入力する。
3. プロセス装置にある SNA 端末サブシステム・プログラムを (可能であれば) 開始し、インテリジェント端末を IMS との通信ができるように準備する。
4. ホストとメッセージを交換し、端末から IMS トランザクションの実行依頼を行い、その実行結果を調べる。

DBCTL 環境でのテスト

CCTL で DBCTL 環境をテストしている場合、ネットワーク、端末、トランザクションは IMS ではなく CCTL が制御することに注意してください。

DCCTL 環境でのテスト

データベースへアクセスする必要のない DCCTL 機能はすべてテストすることができます。

例えば、データベース呼び出しが実行されない場合には、トランザクションのスケジューリングやコンポーネントのテストを行うことができます。テスト・システムに GSAM あるいは外部サブシステム (Db2 for z/OS など) がすでにインストールされている場合は、これらのシステムを必要とする接続やアプリケーションを検証することができます。

IMS テスト・エイド

以下のトピックでは、さまざまなテスト・エイドに関する情報を提供します。

関連資料: アプリケーション・プログラム・コードのテストのときに使用する、推奨されるプロシージャとユーティリティーについては、「IMS V15 アプリケーション・プログラミング」の『IMS アプリケーション・プログラムのテスト』を参照してください。

DB/DC 環境および DCCTL 環境におけるバッチ端末シミュレーターによるオンライン実行のシミュレーション

オンライン・アプリケーション・プログラムの開発がテスト段階まで進んでも、適切なオンライン IMS システムがその時点で使用可能でないことがよくあります。この段階では、バッチ端末シミュレーター (ライセンス・プログラム、プログラム番号 5655-J57) を使用できます。

バッチ端末シミュレーター (BTS) プログラムを使用すると、オンライン IMS システムがテスト・システムとして運用可能になる前の時点で、メッセージ処理領域の操作をシミュレートできます。このプログラムは、1 つ以上のアプリケーションを使用するバッチ IMS システムとして稼働します。BTS への入力はトランザクション・データで、これによりアプリケーション・プログラムが呼び出されます。データベース呼び出しはテスト・データベースに対して実行され、データ通信のための DL/I 呼び出しがシミュレートされます。

BTS を使用して次のような操作を行うことができます。

- 端末の入出力、およびオプションで、関係するデータベース・アクティビティのトレースを印刷する。DL/I 呼び出しの要約を、タイプ別に各 PCB に対して要求できます。
- 会話型トランザクションとプログラム間通信をシミュレートする。
- オンライン IMS システムのメッセージ・キューイング機能とアプリケーション・プログラム・スケジューリング機能をシミュレートする。トランザクション入力データをアプリケーション・プログラムの入力メッセージ構造に変換し、同様に出力メッセージを印刷レイアウトにマップする端末形式制御ルーチンを開発する。これは、IMS 3270 の入出力形式の場合に特に便利な機能です。
- バッチ・アプリケーションと通信アプリケーションの両方にデバッグ機能とトレース機能を (テスト段階で) 使用する。

IMS 制御プログラム、制御ブロック、ライブラリー、アプリケーション・プログラムを変更する必要はありません。

関連資料: BTS について詳しくは、「*IMS Batch Terminal Simulator for z/OS ユーザー・ガイドおよび解説*」を参照してください。

DB/DC 環境および DCCTL 環境での MFS 形式のオンライン・テスト

テスト・システムまたは実動システムのいずれかでアプリケーション・プログラムの変更内容をオンラインでテストする必要があり、さらに MFS がサポートする画面フォーマットがテスト・トランザクションの入力で使用される場合は、MFSTEST モードを使用することができます。

IMS を使用すると、MFS がサポートする個々の端末を MFSTEST モードにすることができます。このモードでは、実動形式ライブラリーのメッセージ形式ブロックを変更せずに、代替 MFS 形式ライブラリーから一時メッセージ形式を使用できます。形式ブロックは、実動ライブラリーに既に存在するメッセージ形式と同じ名前にすることもできます。

MFS パラメーターを指定した /TEST コマンドを入力すると、MFS がサポートする端末がテスト・モードになります。次に、リモート端末のオペレーターがトランザクション処理を行うために /FORMAT コマンドを入力すると、テスト形式の入ったライブラリー (IMS.TFORMAT) からメッセージ形式が選択されます。これらのメッセージ形式がこのライブラリーに入っていない場合は、アクティブ・メッセージ形式ライブラリー (IMS.FORMATA/B) から選択されます。/END コマンドを入力すると、端末はテスト・モードではなくなります。

MFS 形式をテストする場合に、/TRACE SET ON TRAP コマンドを使用して MFS エラーを捕そくし、分析することができます。このコマンドは、MFS 制御ブロックの誤った操作が原因で発生したエラーを分析する際に役立ちます。

MFSTEST モードのシステム定義要件

MFSTEST 形式を使用するには、IMSGEN マクロまたは COMM マクロのいずれかに適切なパラメーターを指定してください。

関連資料: IMSGEN および COMM マクロについての詳細は、「IMS V15 システム定義」を参照してください。

MFSTEST モードのオンライン実行要件

オンライン IMS 実動システムの実行中に MFS 形式のテストを実行する場合は、IMS.TFORMAT ライブラリーが定義されており、このライブラリーに適切なテスト・ブロックが入っていない限りなりません。

これらの形式は、既存の実動形式と同じ名前を付けることができます。IMS.TFORMAT ライブラリーを最初のデータ・セットとし、その後に対応する IMS.FORMATA/B ライブラリーを連結して指定した IMSTFMATA/B DD ステートメントが制御領域 JCL に記述されていなければなりません。

CIOP は、MFSTEST モードで操作する端末の形式ブロックをすべて保持するために使用されます。CIOP のストレージ・スペースは、システムの要求に応じて動的に割り振られます。

MFSTEST モードで実行すると、伝送とメッセージ形式化の両方が同じバッファーに格納されるため、パフォーマンスに影響が現れます。

テストのための動的リソース定義とオンライン変更の使用

動的リソース定義機能とオンライン変更機能を使用して、IMS オンライン・テスト・システムの実行中に選択的にテストを行うよう計画できます。テストを実行するときには、一時的な変更内容に合わせるためだけに変更テスト・システムを生成する必要はありません。

次の操作をサポートするテストを実行できます。

- 新規アプリケーションの追加
 - 新規トランザクション、プログラム、MFS 形式
 - 追加データベース
- 既存のアプリケーションの変更
 - データベース定義の変更

- PSB とプログラムの置換
- MFS 形式の更新
- 変更に応じて行う必要のある削除

IMS システムに対してまだ定義されていない端末または装置の使用をテストする際にこの方法を使用することはできません。

オンライン変更機能を使用する際には、セキュリティー機能の調整も行います。変更の計画では、テストの実行中に、トランザクションや既存の端末を使用するための適切な許可が確実に存在するようにしてください。

アプリケーションの一連の変更を計画して、テスト・システムでの IMS のオンライン操作中または実動サイクルで実行するテスト・パッケージを構成することもできます。オンライン変更は、システム全体の再始動割り込みを発生させずに、事前にテストされた変更内容をマイグレーションする場合に最も適した方法です。オンライン変更を計画する場合は、次の事項を考慮してください。

- オンライン・データ・リソースが適切に保護されていなければなりません。テストされるアプリケーション・プログラムが実動データを損傷する危険性について慎重に評価してください。
- 追加されたデータベースが IMS 制御領域の JCL に組み込まれていなければなりません。z/OS システムでの解決策としては、動的割り振りを使用する方法があります。
- データベース構造の変更内容を新しい ACB ビルドに反映することができますが、変更によって再編成が必要となる場合には、これは常に実用的な処置とはいえません。
- オンライン処理の状況により IMS.PGMLIB の更新を調整しなければなりません。エンド・ユーザーがこのアクティビティーについて認識していないと、置換されたプログラムが原因で問題が発生することがあります。ライブラリーの更新は、/MODIFY COMMIT コマンドの直前の期間に実行しなければならないことがあります。
- テスト実行をモニターすることにより、パフォーマンスに関する計画に有用な情報が得られますが、チューニング済みシステムに対するパフォーマンスの影響を評価しなければなりません。

/MODIFY PREPARE コマンドを入力すると、サービスイン操作が開始されます。オンライン変更の内容による影響を受けるリソースでアクティビティーが安定したら、/MODIFY COMMIT コマンドがサービスインを完了します。この時点でテストを開始できます。計画していたテストが完了したら、/MODIFY PREPARE、/MODIFY COMMIT コマンドをこのシーケンスで再度実行し、テスト中に非アクティブになった古い未変更のシステム・データ・セットを復元します。このテストを実行した担当者は、テストによる影響を取り消し、データベースの不適切な変更内容を取り消します。

DB/DC 環境および DCCTL 環境での SYSIN/SYSOUT を使用したプログラム・テスト

メッセージ処理プログラムをテストする方法の 1 つに、メッセージを含んだ入力データ・ストリームを入力する方法があります。

回線グループを READER として指定し、入力 SYSIN をローカル・カード読取装置に割り当てます。メッセージはメッセージ・キューへの GU 呼び出しによってアプリケーション・プログラムに渡されます。メッセージの処理中は、位置の編集またはロギングは行われません。メッセージの終わりまたはセグメンテーションの正確度は、入力ストリームによって判別されます。

同様に出力の場合も、LINEGRP マクロの UNITYPE キーワードを使用して、回線グループをプリンター、穿孔装置、テープ装置、DASD 装置として割り当てることができます。適切な装置特性に出力 LTERM を割り当てることができます。この場合、実際の装置を割り振る必要はありません。

LINEGRP マクロ・パラメーターが制御領域の DD ステートメントを作成し、出力レコードに対して適切なバッファー・サイズが定義されます。プリンター出力の場合は、48 文字セットへの変換、つまり小文字から大文字への変換が実行されます。その他のコードはすべてピリオドに変換されます。

DB/DC 環境および DCCTL 環境でのテレプロセシング・ネットワーク・シミュレーターを使用したネットワーク・テスト

通信網シミュレーター (TPNS) (プログラム番号 5662-262) は、端末/ネットワークのシミュレーション・ツールであり、システム・パフォーマンスや応答時間の判別、テレプロセシング・ネットワークの設計の評価、機能テスト、レグレッション・テスト・プロシージャの自動化に使用します。

キュー制御機能によるパフォーマンスとストレスのテスト

キュー制御機能 (QCF) を DB/DC 環境および DCCTL 環境で使用すると、アーカイブ・ログ・データ・セットに保管された実際の作業負荷ケースからメッセージを選択することができます。選択した比率でメッセージを再キューイングして、ストレスおよびパフォーマンスのテストを実行することができます。

関連資料: キュー制御機能について詳しくは、「*IMS Queue Control Facility for z/OS ユーザー・ガイド*」を参照してください。

第 24 章 IMS モニター・データの収集および解釈

モニターとは、IMS データを収集して解釈することです。このデータは、システムの毎日のニーズを理解し、変更を必要とする可能性がある領域を知るのに役立ちます。

次のような理由から、モニターは継続実行タスクでなければなりません。

- モニターにより、容量の計画および予測のための基本プロファイル、作業負荷の統計、データを確立できます。
- モニターにより提供される初期段階の警告と比較データによって、パフォーマンスの問題が発生するのを防ぐことができます。
- モニターでは、パフォーマンスの問題に対応して行ったチューニングが評価され、そのチューニングの効果が確認されます。

マルチシステム・ネットワークの場合、ネットワークの一部である IMS オンライン・システムの統計データとパフォーマンス・データを取得する計画を立てます。単一の IMS システムのパフォーマンス・データの生成に使用した次のモニター・ツールを使用することができます。


- IMS モニターは、複数のシステムで同時に実行できます。IMS モニターによる各 IMS システムに関する報告書を取得して、プロセスに関する分析を調整します。

ヒント: IMS モニター・ユーザー出口は、IMS モニター・データにリアルタイムでアクセスするためのフレームワークと出口点を提供します。

- DB/DC および DCCTL システムの場合:
 - IMS 統計分析ユーティリティ (DFSISTS0) は、各システムのトランザクション・トラフィックの要約を生成します。再度、複合ピクチャーの統計を結合します。
 - IMS ログ・トランザクション分析ユーティリティ (DFSILTA0) では、複数システムを通してのトランザクションをトレースし、さまざまなアクティブ物理リンクを使用してトラフィックを調べることができます。

全体として、連続したモニターと履歴レコードからの結論は、エンド・ユーザーの苦情に応え、プロジェクトのチューニングのための初期方向を知るための適切な開始点となります。

関連資料:

 IMS モニター・ユーザー出口 (IMSMON) (出口ルーチン)

モニター手順の確立

使用できるモニター方法がいくつかあります。

次の作業を行うことができます。

- オンライン実行全体の実際の作業負荷を計算する。この作業には、継続的なトラッキングと定期的なトラッキングの両方が含まれます。全体の作業負荷または選択した代表的なトランザクションを追跡できます。
- ピーク・ロード時でのサンプル・スナップショットと通常の状態でのサンプル・スナップショットをとる。次の 2 つの理由から、ピーク期間のモニターは常に有用です。
 - ボトルネックと応答時間の問題は、ピーク・ボリュームではさらに顕著になります。
 - 現行のピーク・ロードにより、将来の平均値がどのくらいになるか分かりません。
- 文書化されたパフォーマンス基準を持つクリティカル・トランザクションまたはプログラムをモニターする。
- 作業負荷の平衡、平均化、リソースの分散を管理する z/OS ワークロード・マネージャーを使用する。

前もってモニター手順の計画を立てておいてください。プロシージャでは、使用するツール、分析方法、このようなアクティビティーによる操作の範囲、アクティビティーの実行頻度が説明されている必要があります。

使用する方法に関係なく、次の操作を行う必要があります。

- パフォーマンス基準の作成
- モニター、データ収集、分析のマスター・プランの作成

パフォーマンス目標の確立

オンライン IMS システムの設計では、パフォーマンス目標はそれぞれに固有のものであります。パフォーマンス目標の確立は主要な作業であり、IMS 全体の作業負荷に関するデータを収集する必要があります。

作業負荷を定義し、必要となるリソースの見積もりを行った後、目標とする応答と達成可能であると想定した応答とを調整する必要があります。システムのパフォーマンスをモニターして、これらの目標が達成されるかどうかを判別します。パフォーマンスの目標は次の項目を基に設定します。

- 目標とする受け入れ可能な最大応答時間
- トランザクション当たりの平均および最大リソース要求 (または作業負荷)
- 予測されるトランザクションのボリュームと実際のトランザクションのボリューム

パフォーマンス目標の確立は繰り返し行われる処理で、次の作業が含まれます。

1. ユーザーを基準にしたパフォーマンスの目標と優先順位を設定する。

これらの目標や優先順位は、エンド・ユーザーによるシステム提供サービスの受け止め方が元になっています。IMS では、これらの目標には端末のエンド・ユーザーにとっての期待応答時間を示します。

応答時間の目標を確立する場合、この目標はシステム内時間 (最後の入力メッセージ・セグメントの入力から最初の応答セグメントまでの経過時間) だけでなく、IMS とアプリケーション・プログラムの予想プロセス量も反映していな

ればなりません。応答時間に関して、平均、90 %、あるいはワーストケースについて、基準を定義するかどうかを検討してください。どれを定義するかは、インストール・システムの監査制御機能と特定のトランザクションの特性によって異なります。

2. ユーザーを基準とした目標に対するパフォーマンスの測定方法とユーザーへの報告方法を決定する。

この作業には、測定結果のデータとユーザーに対して表示されるデータとの体系的な相違点の識別も含まれます。内部 (IMS が認識する) と外部 (エンド・ユーザーが認識する) での応答時間の測定単位の相違について調査してください。この調査では、次のツールを使用できます。

- IMS 統計分析ユーティリティーによって作成されるトランザクション応答報告書。この報告書には、内部応答時間についてトランザクション・タイプ別に次のデータが出力されています。
 - 最長応答時間と最短応答時間
 - 応答時間分布における、20 %、50 %、75 %、および 90 % のもの
 - インストール先作成プログラム。このプログラムは IMS ログ分析ユーティリティーの出力を分析します。その他のプログラムも同種類の情報を提供します。これらのプログラムは、通常、インストール・システムの要件を満たすように調整されています。
3. 現在の作業負荷の理解と記録する。

全体の作業量をカテゴリーに分類し、各カテゴリー別に、次の項目を含む作業負荷プロファイル (通常は見積もり) を作成する必要があります。

- トランザクション・カテゴリーの定義 (例えば、トランザクション・タイプやトランザクション・グループなど)。トランザクション・カテゴリーには次の 2 つの特性があります。
 - IMS の作業負荷。通常トランザクション・プロファイルによって記録されています。適切に設計された IMS オンライン・システムでは、大部分のトランザクションが 1 つの機能を実行し、それぞれ識別可能な作業負荷プロファイルを持っています。プロセスの後半では、共通プロファイルを持つトランザクション・タイプは便宜上 1 つに合併されます。
 - トランザクション・ボリューム。記録されている作業負荷がすでに操作可能な場合、IMS 統計分析ユーティリティーからトランザクションの合計ボリュームを取得することができます。(アプリケーション・アカウント報告書でプログラム名の中にトランザクション・カウントが出力されます。) その他の場合は、トランザクションのボリュームが見積もられます。
- カテゴリーの相対優先順位。優先順位が変更される期間も含まれます。
- 作業のリソース要件
 - オペレーティング・システムが管理する物理リソース (実記憶域、DASD 入出力、端末入出力)
 - サブシステムが管理する論理リソース (制御ブロック、ラッチ、バッファ、領域の数など)

トランザクションのリソース要求の基本プロファイルを取得する場合、専用マシンで IMS を開始し、トランザクションをいくつか実行して初期設定を行いバッファ・プールを使用します。次に IMS モニターを開始してトランザクション実行のサンプルを測定します。

基本トランザクション・プロファイルを使用して、トランザクションの作業負荷を検討し、その作業負荷を軽減できるかどうかを調べることができます。このような設計上の変更を行うと、システムに大きな影響を及ぼし、その結果、システム全体での競合が発生します。基本プロファイルを実稼働環境のトランザクション・プロファイルと比較できます。

4. 取得したリソース要件とボリューム情報を、各作業カテゴリ別のシステムを基準にした目標に変換する。

この変換では、サポートされるトランザクション率 (ピーク期間を含む) と保管される内部応答時間プロファイルに関するステートメントも使用されます。

5. システムを基準にした目標が妥当であることを確認する。

システムの初期化とシステム操作のモニターを実行した後、作業負荷の測定値に基づいて、(使用可能なハードウェアで) 目標が妥当なものであるかどうかを判断する必要があります。使用した見積値と測定値が大幅に異なる場合は、作業負荷を記した資料とシステムを基準にした目標をそれに応じて検討し直すか、システムをチューニングします。

6. パフォーマンス目標を確立する。

パフォーマンス目標の確立は、z/OS ワークロード・マネージャーを使用するための前提条件です。パフォーマンス目標を確立する際に収集した情報の多くを、ワークロード管理を計画する際の考慮事項として使用できます。

ワークロード管理の計画

z/OS が提供するワークロード管理機能は、作業負荷の平衡、平均化、競合する作業負荷へのリソースの配分を行う際に役立ちます。

z/OS は、z/OS ワークロード・マネージャー (WLM) が提供するパネル操作アプリケーションで作業負荷の処理方法が指定されると、自動的にこのサポートを提供します。

ワークロード・マネージャーと IMS

z/OS WLM を使用して、トランザクションのパフォーマンス目標と、トランザクションとアドレス・スペースの相対的重要度を z/OS に対して定義します。すると、WLM は、定義を使用して、z/OS によって制御されるリソースの割り振り方法を決定します。

トランザクションのパフォーマンス目標と業務の重要度:

パフォーマンス目標値には、目標とするトランザクション平均応答時間や、応答時間内の特定のトランザクションの完了率を指定することができます。

業務の重要度とは、インストール・システムの環境でのトランザクションのクリティカルの程度を表す優先順位です。1 が最高優先順位です。

IMS によってトランザクションがスケジューリングされた後で、z/OS WLM は、各トランザクション用に定義したパフォーマンス目標を使用して、この目標を達成するためにプロセッサ・サイクルやストレージなどのリソースをどのように割り当てるかを決定します。システム・リソースに対する競合が発生すると、z/OS ワークロード・マネージャーはトランザクションに割り当てられている業務の重要度を使用して、z/OS WLM が制御するリソースをどのトランザクションが優先して使用するかを判別します。パフォーマンス目標値と業務の重要度はサービス定義で定義されます。

関連資料: サービス定義の詳細については、470 ページの『サービス定義の確立』を参照してください。

IMS アドレス・スペースへの相対的重要度の割り当て:

IMS トランザクションのパフォーマンス目標値を設定後、他のすべての IMS アドレス・スペースと比較して各 IMS アドレス・スペースの重要度を WLM サービス・クラスに指定します。WLM は、この指定値を使用して、z/OS システム内のすべてのアドレス・スペースのプロセッサ・ディスパッチング優先順位を決定します。

以下の図で、WLM サービス・クラスを定義するための WLM ISPF ドリブン・パネルは、2 という相対的重要度を IMS 制御領域アドレス・スペースに割り当てます。

```

                                MODIFY A SERVICE CLASS                                ROW 1 TO 00000100
COMMAND =====> _____ 00000200
                                00000300
SERVICE CLASS NAME . . . . . : CNTL 00000400
DESCRIPTION . . . . . : IMS CTL RGN SERVICE CLASS 00000500
WORKLOAD NAME . . . . . : IMS (NAME OR ?) 00000600
BASE RESOURCE GROUP . . . . . : _____ (NAME OR ?) 00000700
CPU CRITICAL . . . . . : YES (YES OR NO) 00000800
                                00000900
SPECIFY BASE GOAL INFORMATION. ACTION CODES: I=INSERT NEW PERIOD, 00001000
E=EDIT PERIOD, D=DELETE PERIOD. 00001100
                                00001200
      ---PERIOD--- -----GOAL----- 00001300
ACTION # DURATION IMP. DESCRIPTION 00001400
  _ 1 2 EXECUTION VELOCITY OF 70 00001500
                                00001600
                                00001700

```

図 43. WLM ISPF パネル「Modify a Service Class」を使用した重要度の割り当て

以下の表は、いくつかの IMS アドレス・スペースの推奨相対的優先順位を実現するための WLM 重要度値の割り当て方法を提案しています。表の最初の列の N は、1 から 5 までの値を表します。1 が最高優先順位です。

表 35. IMS アドレス・スペースへの WLM 重要度の割り当て

WLM 重要度	IMS アドレス・スペース
N	IRLM、DBRC
N + 1	IMS 制御領域、CQS、OM、RM、SCI、IMS Connect
N + 2	DL/I 分離アドレス・スペース

表 35. IMS アドレス・スペースへの WLM 重要度の割り当て (続き)

WLM 重要度	IMS アドレス・スペース
N + 3	従属領域

表に示されているとおりに IMS アドレス・スペースを優先順位付けることにより、従属領域が処理を終了するまで待たないとそれらが必要とするプロセッサ・リソースを受け取れないという重大な機能のリスクを最小限に抑えます。そのような重大な機能には、ロッキング、DB 許可、IMSplex 全体のコマンド処理、メッセージング、DB のオープン・アクティビティーおよびクローズ・アクティビティーがあります。

関連資料: ディスパッチング優先順位に関する追加情報については、520 ページの『ディスパッチング優先順位』を参照してください。

サービス定義の確立

サービス定義には、ワークロード管理プロセスを実行するのに必要な情報がすべて含まれています。ワークロード管理に装備されているオンラインのパネル操作アプリケーションを使用してサービス定義を確立することができます。

サービス定義の確立に必要な情報の多くは、466 ページの『パフォーマンス目標の確立』で説明したパフォーマンスの目標に含まれています。サービス定義に含まれている重要な情報に、種別の規則があります。

種別の規則は、作業区分修飾子とサービス・クラスで構成されています。サービス定義別に種別の規則セットが 1 つあります。ただし、各サービス定義は複数のサービス・クラスを指定できます。ワークロード管理提供機能を使用して、作業区分修飾子を 1 つ以上指定し、種別の規則を確立し、サービス・クラスを定義します。作業区分修飾子は、着信トランザクションを特定のサービス・クラスと関連付けます。このサービス・クラスは、パフォーマンスの基準 (パフォーマンス目標値と業務の重要度など) が類似しているトランザクションのグループを表します。作業区分修飾子として、サブシステム・タイプ、IMS トランザクションの名前、IMS トランザクション・クラス、入力する LTERM 名、ユーザー ID のいずれかを指定できます。いずれかの値を使用するか、複数の値を組み合わせ使用して、サービス・クラスをトランザクションに割り当てることができます。サービス・クラスには、パフォーマンスの基準が定義されています。トランザクションにサービス・クラスが割り当てられると、z/OS ワークロード・マネージャーはサービス・クラスに定義されているパフォーマンス基準に応じてトランザクションを処理します。

次の例では、種別の規則で作業区分修飾子としてトランザクション名が使用されています。DEPOSIT という名前の着信トランザクションはすべてサービス・クラス IMSHIGH に割り当てられます。

- 作業区分修飾子:
 - サブシステム名: IMS
 - トランザクション名: DEPOSIT
- サービス・クラス:
 - パフォーマンスの目標: 1 秒未満の応答時間
 - 業務の重要度: 1

IMSHIGH サービス・クラスのパフォーマンス・ゴールは、1 秒未満の応答時間となっています。業務の重要度は 1 で、これにより、DEPOSIT トランザクションが IMS によってスケジューリングされた後、システム・リソースに対する競合が発生した場合にこれらの DEPOSIT トランザクションの優先度が高くなります。

推奨事項: 業務上の重要度とゴールに応じて、複数の WLM サービス・クラスを指定してください。次に、WLM の方針に沿って IMS トランザクションをクラス分けします。このクラス分けでは、IMS トランザクションのクラスが、各トランザクションの応答時間のゴールにとって最適な WLM サービスを行うための作業区分修飾子として使用されます。

さまざまな IMS トランザクションにさまざまな WLM サービス種別を割り当てることにより、すべての IMS トランザクションで応答時間のゴールを達成する最適な機会が得られる形で、WLM によって z/OS リソースを管理するようにします。

1 つの WLM サービス・クラスをすべての IMS トランザクションに割り当てた場合、WLM では制御領域も含めすべての IMS アドレス・スペースを同等に扱います。つまり、WLM では、このすべてのアドレス・スペースに同じディスパッチング優先順位を割り当てることになります。トランザクションの一部が応答時間のゴールに到達していない場合、WLM はどのアドレス・スペースの優先順位を高くすべきか判断できない (本来ならば IMS 制御領域が優先) ので、WLM ではどのトランザクションの優先順位も変更しません。

ワークロード管理機能へのマイグレーション

ワークロード管理機能は、互換モードとゴール・モードの 2 つの操作モードを備えています。互換モードは現在のパフォーマンス管理の方式で、ゴール・モードはワークロード管理の方式です。

マイグレーション作業を容易にするために、互換モードで稼働中にサービス定義を確立することができます。サービス定義の内容に満足し、ワークロード管理機能へのマイグレーション手順を完了したら、ゴール・モードに切り替えることができます。

関連情報: サービス定義のセットアップとワークロード管理の使用の詳細については、「z/OS MVS 計画: ワークロード管理」を参照してください。

z/OS WLM 変更状態パフォーマンス・ブロック・サービス・コードの解釈

WLM 変更状態パフォーマンス・ブロック (PB) サービス機能を使用して、トランザクションの現在の状態を表示します。

IMS の場合、PB サービス・コードは、次のように解釈されます。

状態 説明

アクティブ

トランザクションはアプリケーション・プログラムの実行中

解放状態

WLM. はこの状態を報告しない

アイドル状態

トランザクションは作業を待機中

待機状態 - 入出力

IMS は入出力の待機中 (IMS は入出力を開始した)

待機状態 - ロック

IMS はロック要求の待機中

WLM を使用したモバイル・ワークロードのモニター

IMS は、モバイル・アプリケーションによって処理されたり、モバイル・デバイスから発信されたりするワークロード・トランザクションを識別できます。調整済みの請求方法をモバイル・ワークロード・トランザクションに使用できます。

モバイル・ワークロードの分類規則の定義:

モバイル・ワークロードのレポート作成を簡素化するために、分類規則の *Reporting Attribute* オプションを使用して、モバイルと非モバイルのワークロードを区別することができます。

Reporting Attribute オプションには、以下のいずれかの値を使用できます。

NONE

すべての作業に適用可能。これはデフォルトです。

MOBILE

モバイル作業にのみ適用可能。

CATEGORYA

作業の最初の汎用サブセット用に予約済み。これは、将来の使用に備えて提供されています。

CATEGORYB

作業の 2 番目の汎用サブセット用に予約済み。これは、将来の使用に備えて提供されています。

モバイル作業を別個のモバイル・レポート用に識別するには、モバイル作業を MOBILE として定義します。

関連概念:

470 ページの『サービス定義の確立』

WLM を使用したモバイル・ワークロードの測定:

IMS は、MWP を使用してモバイル・ワークロードを追跡するために、LPAR レベル、サブシステム・レベル、および CPU レベルでのトラッキングという 3 つのメカニズムをサポートします。それぞれのトラッキング・メカニズムに別々の構成が必要です。

以下のいずれかのメカニズムを使用するように IMS システムを構成できます。

- モバイル専用のワークロードに LPAR を個別に使用する。すべての定義プログラムは、その LPAR 上で稼働する General Capacity Processor (GCP) CPU をモバイル CPU として報告できます。

- モバイル専用ワークロード用に、定義プログラムのサブシステムを個別に構築する。この場合、定義プログラムは、サブシステム上で稼働する GCP CPU をモバイル CPU として報告することができます。
- モバイル・ワークロードと非モバイル・ワークロードに同じサブシステムを使用するが、モバイル CPU に対して異なる CPU レポートを作成する。

CPU レベルのトラッキング

モバイル・ワークロードが CPU レベルで追跡される場合、IMS は TCP/IP ポート番号を論理端末 (LTERM) オーバーライド名または OTMA トランザクション・パイプ (TPIPE) 名と一緒にワークロード・マネージャー (WLM) に報告して、モバイル・ワークロードと非モバイル・ワークロードを区別します。

LTERM オーバーライド名は、Open Transaction Manager Access (OTMA) での IMS アプリケーション・プログラムの入出力 PCB の LTERM フィールドにある既定の値をオーバーライドするために設定されます。LTERM オーバーライド名が提供されなかった場合、OTMA は、代わりに、事前定義された TPIPE 名を入出力 PCB LTERM フィールドに配置します。

ご使用の TCP/IP ポート構成に応じて、以下のいずれかの値でワークロード・タイプを区別できます。

TCP/IP ポート番号

各 TCP/IP ポートが、モバイル・ワークロードまたは非モバイル・ワークロードのどちらか 1 つのワークロード・タイプのトランザクションだけを処理するよう、IMS Connect を構成してください。

LTERM オーバーライド名または TPIPE 名

モバイル・トランザクションと非モバイル・トランザクションの両方が同じポートに到着する場合は、LTERM オーバーライド名 (存在する場合) または TPIPE 名を使用してください。

モバイル・ワークロードの価格設定に適切なモバイル作業を指定するには、分類規則を使用します。どのようなポート番号、LTERM オーバーライド名、または TPIPE 名をモバイル・トランザクションに関連付けるかを明示的に定義するには、*Reporting Attribute* を MOBILE に設定します。指定した値に基づいて、IWM4CLSY マクロはサービス・クラスを返します。

モバイル・ワークロードのレポート作成の分類規則をセットアップする方法について詳しくは、472 ページの『モバイル・ワークロードの分類規則の定義』を参照してください。

ワークロード・レコードの検査


抽出された z/OS システム管理機能 (SMF) レコードに対して ERBSCAN コマンドおよび ERBSHOW コマンドを実行することにより、SMF レコード内のワークロード実行レコードを検査することができます。

X'56FA' ログ・レコードを調べて、ワークロード・タイプを判別することもできます。X'56FA' ログ・レコードには、トランザクション・ワークロードに関する統計情報が含まれています。ログ・レコードはワークロードがモバイルであるかどうか


を直接示すものではありませんが、ジョブ名、プログラム名、トランザクション・コード、トランザクション・クラスなどの情報から、ワークロード・タイプを判別できます。

WLM パネルを使用してトランザクションを分類している場合は、X'56FA' ログ・レコードを調べてワークロード・タイプを判別する必要はありません。

関連概念:

 論理端末 (LTERM) (コミュニケーションおよびコネクション)

関連資料:

 TCPIP ステートメント (システム定義)

毎月のモバイル・ワークロード・レポートの生成:

毎月のレポート作成に使用されるモバイル・ワークロード・トランザクションのソース・データの収集と保持は、ユーザーが行います。IMS は、サブキャパシティー・レポートング・ツール (SCRT) を使用するデータ収集とレポート作成の 2 つの方法をサポートしています。

モバイル・ワークロード・トランザクション・レポートは、モバイル・トランザクションを処理するすべてのマシンの LPAR ごとの時間別に要約された、各モバイル・トランザクション・プログラムの汎用プロセッサ CPU 秒から構成されている必要があります。モバイル・ワークロード・レポートを生成するには、モバイル CPU 使用量レポートが毎月サブミットされる必要があります。

IMS には、モバイル・ワークロード・トランザクションを処理している各 MWP 定義プログラムが使用する CPU 時間を識別するために、データ収集およびレポート作成のための 2 つの方法が用意されています。すなわち、WLM 分類規則を使用するか、元の CSV ファイルを使用します。

注: 各地域の法律とポリシーによっては、監査目的のために、請求対象期間後の一定の期間、SMF レコードとモバイル・ワークロード・トランザクション・データを保持することが必要になる場合があります。

WLM 分類規則を使用したレポートの生成

WLM 分類規則を使用して、モバイル・ワークロード・トランザクションを識別できます。WLM モバイル価格設定サポートに使用されるパラメーター (CPUTIME、TIMEONCP、および OFFLOADONCP) は、WLM が、事前定義されたインターバル中にモバイル・ワークロード・トランザクション CPU 時間を LPAR 内に累積できるようにします。CPU 時間は、SMF タイプ 70 レコードに保管されます。各サービス・クラスまたはレポート・クラスのワークロード・アクティビティー情報は、SMF タイプ 72 サブタイプ 3 レコードに保管されます。

WLM 分類規則を使用してレポートを生成するには、以下の手順を実行します。

1. 新規の分類規則を定義して、モバイル・ワークロード・トランザクションを追跡します。
2. SMF タイプ 72、タイプ 70、およびタイプ 89 のレコードを毎月、SCRT にエクスポートしてロードします。

3. サブキャパシティー・レポート作成期間ごとに、SCRT バージョン 23 リリース 13.0 以降を実行し、結果を IBM に送信します。

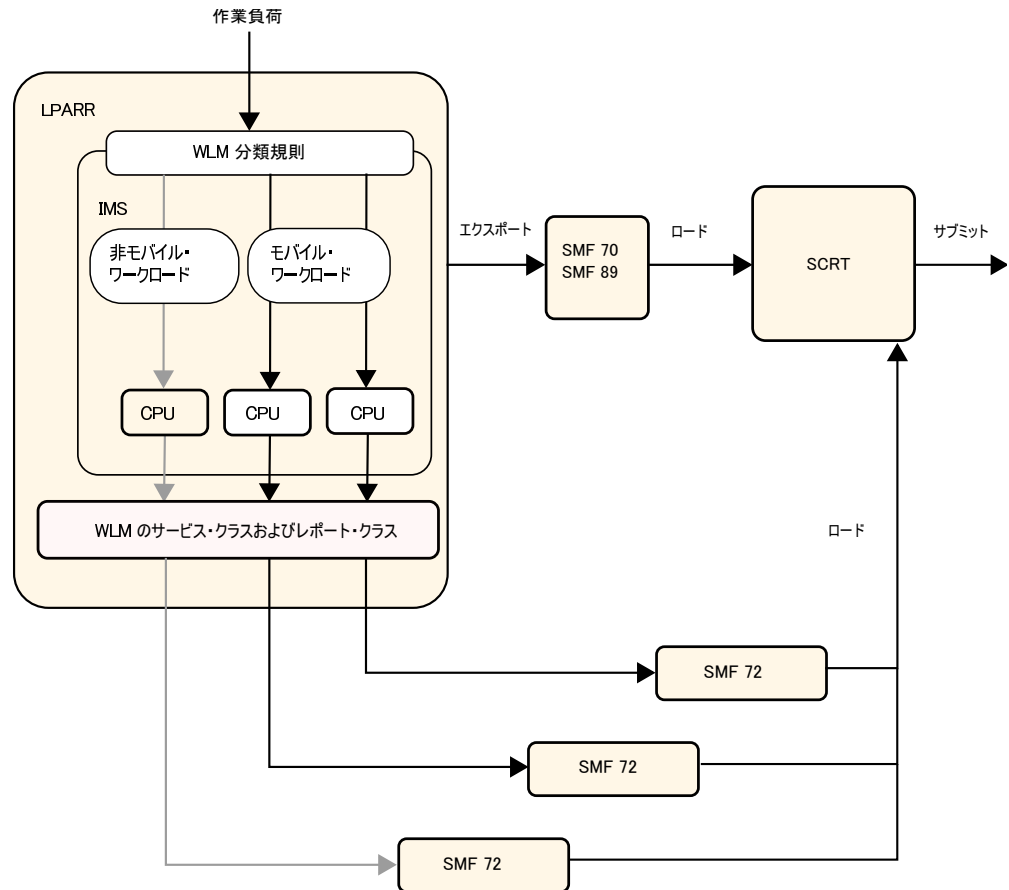


図 44. WLM 分類規則を使用したワークロード・レポートの生成

CSV ファイルを使用したレポートの生成

モバイル・ワークロード・トランザクションの WLM 分類規則を使用できない場合は、お客様の責任で各サブキャパシティー・レポート作成期間ごとにモバイル・ワークロード・トランザクション・データを前処理し、SCRT にロードするために事前定義されたフォーマットにします。データは、モバイル・ワークロード・トランザクションを処理するすべてのマシンの LPAR ごとの時間別に要約された、モバイル・ワークロード・トランザクションの汎用プロセッサ CPU 秒から構成されている必要があります。

データ収集に元の CSV ファイル方式を使用する MWP 定義プログラムの場合、以下の手順に従ってレポートを生成します。

1. プログラムごとに、LPAR ごとの毎時ベースでモバイル・ワークロード・トランザクション (CPU 秒を含む) を追跡します。
2. IBM Transaction Analysis Workbench for z/OS の MWP コマンドを使用して、コンマ区切り値 (CSV) ファイルを生成します。この CSV ファイルには、各プログラムの月ごとのモバイル・ワークロード・トランザクション CPU 使用量が入っている必要があります。

3. 結果のデータ・ファイルを、SMF タイプ 70、および SMF タイプ 89 のレコードと一緒に SCRT に毎月ロードします。
4. サブキャパシティー・レポート作成期間ごとに、SCRT バージョン 23 リリース 13.0 以降を実行し、結果を IBM に送信します。

モニター・アクティビティーおよび技法の決定

パフォーマンスのモニターと分析に関するマスター・プランを作成するには、動的、日次、詳細のいずれのモニターが最も要求に合っているかを判断し、それに応じてプランを作成する必要があります。

パフォーマンスのモニターと分析に関するマスター・プランを作成する場合には、次の項目を確立する必要があります。

- モニター・アクティビティーの基本スケジュール

操作手順とモニターを調整して、毎日行うデータ収集または明細データ収集の命令にオンライン・イベントのフィードバックを組み込んでください。

- モニターに使用するツール

データ収集に使用するツールは、動的モニター機能、毎日の統計収集機能、詳細モニター機能を装備している必要があります。

- 実行する分析の種類

モニター出力からどのデータが抽出されるデータを記録して、データのソースと使用法を識別してください。モニター・ツールが提供する定様式の報告書はデータのボリュームを編成するには役立ちますが、データ抽出やデータの削減を実行する際に利用できるワークシートを設計する必要があります。

- 検出結果の検証を行う担当者のリスト

モニター・データの分析結果と結論は、ユーザー連絡グループとシステム・パフォーマンスの専門家にも通知してください。

- チューニング推奨に基づくオンライン IMS システムの設計への変更の実装方法

この変更の実装方法を、インストール・システムでの実稼働環境の変更頻度の基準とテストの基準にあわせて調整してください。変更の管理については、501 ページの『第 25 章 システム設計の変更』で説明します。

モニター・アクティビティーについて、次の 3 つの広いレベルで計画を立ててください。

- 動的

短期間でもパフォーマンス目標からかなり逸脱したものを検出するため、システムの操作を継続的に監視します。

このレベルのモニターでは、/DISPLAY または QUERY コマンドの出力を、エンド・ユーザーへのフィードバックとともに使用すると役立ちます。リソース測定機能 (RMF™) II は、プロセッサ、チャンネル、入出力装置の使用状況に関する情報を収集するために使用できます。

MTO は、オンライン IMS システムの動作に関する重要な情報源となります。MTO フィードバックでは、IMS モニター実行中の一連の状態が重要になります。この情報は、モニター・データの妥当性を確立する際に役立ちます。

/DISPLAY または QUERY コマンドを使用して取得できる状況情報を使用して、オンライン実行時のプロセス状況を取得する準備ができます。状況情報には、キューのレベル、アクティブ領域、アクティブ端末、会話型トランザクションの数とタイプが含まれます。このような状況情報は、MTO が呼び出す自動化操作プログラムを使用すると取得できます。メッセージまたは BMP 領域のスケジューリング前、ネットワークの一部のシャットダウン時、ピーク・ロード時など、実動サイクルで事前に設定されている特定の時点で、トランザクションのプロセス状況とシステム・リソース・レベルの測定値を記録することができます。

ページング率を動的にモニターするには、次の方法のいずれかを使用します。

システム全体のページング率

RMF モニター II ページング報告書には、サンプリング・インターバル別のページング・アクティビティのスナップショットが出力されます。

ユーザー当たりのページング率

RMF モニター II アドレス・スペース・リソース・データ報告書には、各 RMF 測定インターバルのアドレス・スペースごとの割り振り済みフレームとページ不在のカウントが出力されます。ページ不在カウントは累積カウントであり、ローカル・アドレス・スペースのページ不在 (ページ入出力ではない) と CSA/LPA も含まれます。

IMS で一時的な遅延が発生していることがモニターにより検出された場合、TSO またはバッチのイニシエーターを停止できることがあります。

プロセッサ・リソースをモニターするには、次の機能を使用してください。

システム使用状況の合計

RMF CPU アクティビティ報告書には、RMF の報告書作成インターバル (通常 10 から 30 分) の WAIT TIME PERCENTAGE が出力されます。

プロセッサ・オーバー・コミットメント

RMF モニター II 実記憶域/プロセッサ/SRM 報告書には、各 RMF サンプル・インターバルのプロセッサ使用率の割合が出力されます。アドレス・スペースがディスパッチできる状態になっているが、インターバルの間はプロセッサ・サイクルを待機する必要がある場合はこの値は 101 % になります。

- 目次

主要なシステム・パラメーターを毎日調べて記録します。日ごとの平均値とピーク期間 (通常 1 時間) の平均値を記録します。主要なパフォーマンス目標とこの平均値を比較して、適切でない傾向を見つけだします。

通常、このデータはイベントのカウンと全体でのタイミングのカウンとで構成されています。場合によっては、入力キューイングやプログラム実行などの経過時間のように、IMS システム全体で時間測定情報の平均値を算出することがあります。

IMS システム・ログをオフライン・プロセスへの入力として使用して、日次の統計または定期的な統計を出すことができます。このレベルのモニターには IMS ログ・トランザクション分析と IMS 統計分析の 2 つのユーティリティーが適しています。これは、この 2 つのユーティリティーを使用してもオンライン・システムのプロセス負荷が増えることがないためです。

動的 RMF ページング・アクティビティー報告書の分析で、ページング入出力が増加したことが示されている場合、これは対応する期間内に作業負荷で何らかの変更が行われたことと関連しています。チューニング処置を決定するため、通常さらに詳細な分析が必要となります。

RMF CPU アクティビティー報告書の分析で、遅い応答時間内の CPU 使用率が断続的に高いことを示す場合、IMS メッセージ領域によるトランザクションの経過時間とプロセッサ時間を調べてください。優先順位の低い領域で経過時間/プロセッサの比率がかなり高くなる場合は、他の非 IMS 作業に対してディスパッチング優先順位を高くしてください。

関連資料: すべての領域に均等に影響を及ぼす場合は、パス長さを削減する方法について、534 ページの『パス長さの最小化』を参照してください。必要に応じて競合する作業の優先順位を調整し、IMS メッセージ領域よりも低くなるようにしてください。

プロセッサ・オーバー・コミットメントのサブシステムによる使用状況をモニターするには、RMF モニター II アドレス・スペース状態データ報告書を使用します。この報告書には、RMF 測定インターバルごとに各アドレス・スペースが使用するプロセッサ装置が出力されます。IMS の制御領域とメッセージ領域は、ジョブ名によって識別できます。

- 詳細

システム指向の目標と作業負荷プロファイルに対するパフォーマンスを分析するため、システム操作の詳細な統計を定期的に収集します。

このレベルでは、データの量が非常に多くなります。これには通常、イベントや作表の (発生) シーケンスも含まれます。タイミングは詳細レベルで報告されません。

このレベルのモニターでは、IMS モニターおよび汎用トレース機能 (GTF) などのような特殊なトレース・ツールが役立ちます。これらのツールはオンライン処理の詳細なサンプルを収集し、従属領域でのアクティビティー、端末とメッセージ・キューの非同期プロセス、バッファ・プールの使用状況、システム・データ・セットの入出力を識別します。

これらのモニター・ツールの使用法に関する追加情報については、このセクション内の別のトピックに含まれています。モニター機能とツールを使用してパフォーマンスの問題を検出する方法については、539 ページの『パフォーマンスの問題の識別と訂正』で説明します。

次の方法を使用して、ページング率の詳細モニターを実行します。

ユーザー当たりのページング率

毎日行う分析では不十分な場合は、GTF トレースの実行について計画を立ててください。複数のページ・データ・セットを使用している場合は、専用ページングとグローバル・ページングを識別することができます。GTF 詳細トレース報告書を調べて、ページ不在による経過時間の遅延 (特に制御領域) を算出し、ページングによる IMS への影響を評価します。GTF ページ不在要約報告書を使用して、ページ不在による影響を受ける IMS の領域またはシステム・コードを検出できます。この報告書には、ページ不在のタイプも出力されます。

このタイプのデータを分析して、実記憶域使用のチューニングを行うかどうかを判断します。

関連資料: 実記憶域の使用法をチューニングする場合に考慮すべき他の要素については、531 ページの『IMS が制御する入出力とページングとのトレードオフ』を参照してください。

スケジュール/終了処理時の NOT-IWAIT 時間の検討

IMS モニター領域要約報告書は、スケジュール時間または DL/I NOT-IWAIT 時間を調べてディスパッチング優先順位に関する問題を最初に評価する際に役立ちます。DL/I NOT-IWAIT 時間とは、IWAIT 時間として見なされない経過時間です。NOT-IWAIT 時間が増加する原因として、次のものが考えられます。

- ページングの遅延
- 優先順位の高いタスクのディスパッチ

発生するページングが最も少ない場合、領域のスケジュールと終了処理中に発生する経過待機時間の部分は、システム間で一貫しています。この経過待機時間は、IWAIT 時間として見なされません。この時間は IMS モニター領域要約報告書に記録されて、ヘッディング NOT-IWAIT TIME の下に作表されます。DL/I 呼び出しの NOT-IWAIT 時間には動的ロギング入出力の遅延も含まれます。優先順位の高いタスクのページングまたはディスパッチが原因で発生した遅延により、NOT-IWAIT 時間が増加します。

NOT-IWAIT TIME の合計平均値が極端な場合は、IMS に対してマシン・リソースが十分でない可能性があります。優先順位の高いタスクが存在していない場合は、IMS スケジューラー・コード、制御ブロック、PSB プールのページング率が高いことが原因となっている可能性があります。

SVC マッピング要約は、SVC が発行された位置を判別するのに役立ちます。

関連資料: パス長さの削減について詳しくは、534 ページの『パス長さの最小化』を参照してください。

詳細モニター用ツール

詳細データの収集に使用できるモニター用ツールの多くは、一般的な診断にも使用することができます。データの収集と分析のために IMS が提供する基本的なツールは、オンライン・サブシステムをモニターできる IMS モニターです。

SLDS が駆動する独立型の IMS DB バッチ・システムの場合は、DB モニターを使用してください。DB モニターは、IMS バッチ・ジョブの全実行を通じてアクティブにすることも、システム・コンソールからそれを停止して、再始動することもできます。

また、IMS Performance Analyzer、プログラム分離およびロック・トレース、および外部トレース機能も使用できます。

関連概念:

951 ページの『第 50 章 //DFSSTAT 報告書』

➡ IMS Performance Analyzer for z/OS の概要

関連タスク:

➡ パフォーマンス関連データの収集 (オペレーションおよびオートメーション)

IMS モニター

IMS モニターは、オンライン IMS サブシステムの稼働中にデータを収集します。これは、すべてのディスパッチ・イベントに関する情報を収集し、その情報を IMS モニター・レコードの形で順次データ・セット内に配置します。IMS モニター・データ・セットを指定するには、IMS 制御領域 JCL の中で IMSMON DD ステートメントを使用します。

/TRACE コマンドを使用して IMS モニターを活動化する際に、IMS がデータをこのデータ・セットに追加します。IMS MTO は、IMS モニターを開始したり、停止したりして、システムのスナップショットを随時入手することができます。ただし、IMS モニターによりシステム・オーバーヘッドが増加し、かなりの量のデータが生成されます。

IMS モニター出力の制御

IMS モニターの実行は短い間隔で行い、そのオペレーションを慎重に制御することを考えてください。間隔が短いほど、統計全体にわたる平均化を防ぐことにもなり、システム内の問題の識別を速めることができます。IMS モニターの出力は、以下の項目によって制約することができます。

- モニターするアクティビティのタイプ
- データベースまたは区画またはエリア
- 従属領域
- 時間間隔

IMS モニターの出力データ・セット

IMS モニター出力は、テープまたは DASD データ・セットのいずれかにすることができます。DASD を使用すると、磁気テープ・ドライブをオンライン・システム

に割り振る必要がなくなります。IMS モニターを頻繁に使用したい場合は、DASD データ・セットに永続的にスペースを割り振るほうが便利な場合があります。1 つの技法としては IMSMON DD ステートメントで DISP=SHR をコーディングする方法があります。こうすると、IMS モニターの実行が終了するたびに報告書を作成することができます。

IMS モニターを活動化すると、そのつど既存のデータが上書きされるので、報告書作成についてはオペレーターと調整する必要があります。磁気テープ・データ・セットの場合は、この上書きは発生しませんが、新しいボリュームをマウントする必要があります。ボリュームは巻き戻され、IMS モニターを開始するごとに取り付け要求が出されます。

推奨事項:

- IMS モニター・データ・セットは、カタログを作成しないでください。IMS モニターは、データ・セットのカタログが作成されていない場合は、IMS が稼働している間に複数の出力ボリュームを作成することができます。

IMS モニター・データ・セットが IMS によって動的に割り振られるようにするには、IMS 制御領域 JCL に IMSMON DD ステートメントを組み込まないでください。

- IMS が IMS モニター・テープ・データ・セットを動的に割り振られるようにしてください。磁気テープ・ドライブは、動的に割り振られたデータ・セットに対して、制御領域用に永続的には予約されません。

IMS モニターのトレース

モニター要件を設定した後、IMS モニターのアクティビティの有効範囲を制限できる場合があります。効力範囲を制限すると、システム・スループットに対する IMS モニターの影響を減らせる利点があります。しかし、有用なデータの収集を犠牲にしてまで、これを行うべきではありません。

/TRACE コマンドで特定のキーワードを使用することによって、トレースするイベントの特定のタイプを制御することができます。例えば、回線アクティビティ、スケジューリングおよび終了イベント、アプリケーション・プログラムとメッセージ・キュー間のアクティビティ、アプリケーション・プログラムとデータベース間のアクティビティ、またはすべてのアクティビティをモニターすることができます。また、以下に対するモニターを制限することもできます。

- 特定のデータベース、区画、またはエリア
- 特定の従属領域
- 指定された時間間隔




IMS モニター報告書

IMS Performance Analyzer for z/OS または IMS モニター報告書印刷ユーティリティ (DFSUTR20) を使用することにより、IMS モニターの出力に基づいた報告書を取得できます。


IMS モニター報告書印刷ユーティリティは、IMS が作成した生データ（高速機能データを除く）を要約して、フォーマットし、情報を一連の報告書で表します。DL/I 呼び出しや表形式の頻度分布に関する報告書を、抑止することができません。

IMS モニターされるイベントの期間は、モニターの開始と停止の項目によって決められます。報告書作成に異なる時間枠を選択することはできません。時間が測定されるイベントの多くは連続的に取り込まれるのではなく、IMS モニターの開始時と停止時にだけ取り込まれるからです。したがって、IMS 制御域を停止するための何らかのアクションを実行する前に、IMS モニターが停止していることを確認してください。

関連概念:

-  IMS モニター (データベース管理)
-  システムのモニター (オペレーションおよびオートメーション)
-  IMS Performance Analyzer for z/OS の概要

関連資料:

-  IMS モニター報告書印刷ユーティリティ (DFSUTR20) (システム・ユーティリティ)

z/OS 汎用トレース機能 (GTF)

z/OS 汎用トレース機能 (GTF) は、幅広いシステム・レベル・イベントを記録するために使用します。このトレース・アクティビティは、MODIFY コマンドを使用して z/OS システム・コンソールから制御することができます。


出力は、汎用フォーマット設定ユーティリティが使用する順次データ・セットにスプールされます。トレース・レコードを編集し、そのデータを希望どおりに表示するために、このフォーマット設定ユーティリティが呼び出す、出口ルーチンを作成することができます。

以下の GTFPARS 報告書は、IMS では特に有益です。

- ジョブ要約報告書: この報告書は、IMS メッセージ領域のそれぞれと制御領域に関して、SVC カウント、モジュールのロードと、EXCP、SIO、および入出力のタイミングのリストを作成します。
- 詳細トレース報告書は、以下の情報に関するリストを作成します。
 - IMS メッセージ領域: プログラム管理アクティビティの詳細分析。この分析を使用すると、イベント・シーケンスを詳細に調べることができます。
 - IMS 制御領域: 制御領域で処理されるデータベース入出力、およびネットワークに関連したアクティビティ。
 - 並列に稼働するすべての IMS 領域: 領域間の競合 (例えば、プログラム・ライブラリーへのアクセスに関する競合)、および制御領域から MPP へのフローの制御。
- z/OS 内のマスター・アドレス・スペースのジョブ要約報告書: この報告書は、システム・ページング・アクティビティを示します。

- z/OS 内のマスター・アドレス・スペースの詳細トレース報告書: この報告書は、ページングに起因する遅延を詳しく調査するために使用できます。
- システム要約報告書: この報告書は、すべての IMS データ・セットの入出力サブシステム分析に使用できます。

関連概念:


 z/OS: 汎用トレース機能 (GTF)

z/OS コンポーネント・トレース (CTRACE) サービス


IRLM は、z/OS コンポーネント・トレース (CTRACE) サービスを使用して IRLM アクティビティをトレースします。トレース出力は z/OS CTRACE フォーマットであるため、IPCS CTRACE のフォーマット、マージ、および位置指定ルーチンを使用して、バッファ・データを処理することができます。


IRLM 診断トレースの開始、停止、または修正を行うときは、z/OS TRACE CT コマンドを使用します。このコマンドは、z/OS マスター・コンソールからのみ入力できます。このコマンドの入力には、しかるべきレベルの z/OS 権限が必要です。IRLM は、このコマンドで使用可能なすべてのオプションをサポートしているわけではありません。IRLM プロシージャに TRACE=YES を指定して、IRLM トレースを開始することもできます。

関連タスク:

 IRLM アクティビティのトレース (オペレーションおよびオートメーション)

関連資料:

 TRACE CT コマンド (コマンド)

 z/OS: MVS 対話式問題管理システム (IPCS) の CTRACE サブコマンド

 z/OS: TRACE コマンド

プログラム分離およびロック・トレースの入手

IMS DB/DC または DBCTL 環境においては、プログラム分離トレース報告書ユーティリティ (DFSPIRPO) が作成した出力を調べることによって、データベース・セグメントの競合を検出することができます。

ユーティリティのソース・データを得るには、/TRACE SET ON PI OPTION ALL コマンドを出します。ソース・データの収集を停止するときは、/TRACE SET OFF PI コマンドを出します。ユーティリティの制御ステートメントは、指定した日付に相対する開始または停止時刻を選択することができます。

プログラム分離機能のトレースにより、追加ログ・レコードが作成される場合があります。これらのレコードには、データベース更新、チェックポイント、およびメッセージ処理イベントの結果としての同期点間で、プログラム分離機能が出したエンキューあるいはデキュー要求が含まれます。

プログラム分離トレース報告書ユーティリティは、待ち時間を必要としたこれらのイベントを報告するだけです。この報告書は、データ管理ブロック (DMB) 名、

データベース制御ブロック (DCB) 番号、相対バイト・アドレス (RBA)、プログラム仕様ブロック (PSB) 名、およびトランザクション・コードを識別します。ユーティリティーは、すべてのアクティビティーを RBA 番号別にソートします (報告書内では ID で表示)。報告書は、トレース・インターバル間の待ちを必要としたエンキューの経過時間と、ID、DCB、および DMB ごとのエンキューの合計数をリストします。要求元 PSB またはトランザクションは、同じセグメントを待つ次のエンキューの保持 PSB またはトランザクションと考えられます。サンプル報告書を以下の例に示します。この報告書では、高速機能に関しては、経過待ち時間は記録されません。

PROGRAM ISOLATION TRACE REPORT PAGE 1

DATE: 08/10/04
TIME: 16:36 TO 16:37

DCB *** REQUESTING	*** ELAPSED	**** HOLDING ***	ID	TOTAL	DCB TOTAL	DMB	TOTAL			
DMB NAME	NUM	ID	TRAN AND PSB NAMES	TIME	TIME	TRAN AND PSB NAMES	ENQ'S	ENQ'S	ENQ'S	
TABLEDBQ	1	0022D020	DE1Q	PROGDE1Q	16:36:54	0:00.061	DE2Q	PROGDE2Q		
		003BE00C	DE2Q	PROGDE2Q	16:36:51	0:00.027	DE1Q	PROGDE1Q		1
		007D901C	DE2Q	PROGDE2Q	16:36:34	0:00.036	DE1Q	PROGDE1Q		1
		008EF014	DE2Q	PROGDE2Q	16:36:49	0:00.038	DE1Q	PROGDE1Q		1
		0090401C	DE1Q	PROGDE1Q	16:36:50	0:00.072	DE2Q	PROGDE2Q		1
		00A06010	DE2Q	PROGDE2Q	16:36:38	0:00.046	DE1Q	PROGDE1Q		1
		00A1401C	DE1Q	PROGDE1Q	16:36:50	0:00.008	DE2Q	PROGDE2Q		1
										1 7 7
TABLEDBR	1	002A901C	DE2R	PROGDE2R	16:36:40	0:00.034	DE1R	PROGDE1R		
		0045801C	DE2R	PROGDE2R	16:36:41	0:00.043	DE1R	PROGDE1R		1
		0072F024	DE1R	PROGDE1R	16:36:30	0:00.053	DE2R	PROGDE2R		1

ファイル選択・フォーマット設定印刷ユーティリティーを使用すると、以下の方法で、ログ・レコード内のトレース・テーブルや PI 項目を選択し、印刷することができます。

- OPTION ステートメントを、PRINT パラメーターと COND=E および EXITR=DFSERA40 キーワード・パラメーターで指定する。出力は、順次にフォーマットされたプログラム分離 (PI) トレース・レコードを含む報告書です。
- IMS トレース・テーブル・レコード・フォーマット設定・印刷モジュール (DFSERA60) を使用して、トレースを含んでいるログ・レコードのみを選択する。このモジュールは、ファイル選択・フォーマット設定印刷ユーティリティーからタイプ X'67FA' のログ・レコードを受け取り、それらのレコードを SYSPRINT データ・セット上にフォーマット設定する出口ルーチンです。

OPTION ステートメントを、PRINT パラメーターと COND=E および EXITR=DFSERA60 キーワード・パラメーターで指定する。出力は、PI トレース項目、DL/I トレース項目、およびロック・トレース項目 (これらの項目を順次に表示するようにフォーマットされた) を含む報告書です。見出しの説明については、マクロ IDLIVSAM TRACENT のアセンブリー・リストを参照してください。

IMS トレース・テーブル・レコード・フォーマット設定・印刷モジュールからの出力報告書を使用して、以下に関する詳細を知ることができます。


- 制御レベル (LEV) の欄は、読み取り専用、共用、排他制御、および単一更新アクティビティーを表示します。
- 戻りコード (RC) 欄は、DFSFXC10 または IRLM からの戻りコードを示します。呼び出し元が要求したリソースを待つ必要があったかどうか、あるいはトランザクションがデッドロック状態の原因であったかどうかを判別することができます。
- PST 通知コード (PC) 欄は、待機の原因を表示します。この項目が X'60' の場合、デッドロックが発生しました。


ファイル選択・フォーマット設定印刷ユーティリティーに追加の OPTION ステートメントを指定して、デッドロックを確認するレコードのみを印刷することによって、検査のレコード数を削減することができます。

IMS は、動的バックアウトを使用してデッドロック状態を自動的に解決します。しかし、デッドロックの検出が重要であるため、以降のデッドロックを起こさせないためにアプリケーションの設計を修正することができます。

プログラム分離トレース報告書の利点は、特定のセグメントまたはセグメント範囲について競合が発生した場所を示す点です。また、その報告書は、データベース内で競合するトランザクションも示します。さらには、応答時間の遅延を物語っている可能性がある、長い待ち時間も示します。セグメントの競合を処理する 1 つの方法として、データベースの設計を変更して、フィールドのいくつかを 1 つの追加セグメント・タイプに分離することも可能です。

関連資料:

 プログラム分離トレース報告ユーティリティー (DFSPIRPO) (データベース・ユーティリティー)

 プログラム分離トレース・レコード・フォーマット設定・印刷モジュール (DFSERA40) (システム・ユーティリティー)

IMS トレース機能

IMS トレース・テーブルを内部的にあるいは外部トレース・データ・セットに書き込むときは、IMS トレース機能を使用することができます。

IMS は、この外部トレース・データ・セットを、以下のように DASD またはテープのいずれかに書き込むことができます。

- DASD データ・セットは、JCL によって割り当てることも、動的に割り振ることもできます。
- テープ・データ・セットは、動的に割り振る必要があります。

トレース・テーブルを OLDS に書き込むこともできますが、これは、逆に OLDS のパフォーマンスに不利な影響を与える可能性があります。外部トレース・データ・セットは OLDS から独立しているため、OLDS が使用できなくても、トレース・テーブルを外部トレース・データ・セットに書き込むことができます。

トレースの状況を表示するには、/DISPLAY TRACE コマンドを使用します。このコマンドは、有効な IMS トレースの状況、および使用中の外部トレース・データ・セットがあればその状況を、実際に判別するために使用できます。

OTMA メッセージ・ルーティングのパフォーマンスの分析

OTMA タイプ 2 コマンド QUERY OTMATI を使用して、OTMA メッセージ・ルーティングのパフォーマンスを分析できます。

拡張回復機能 (XRF) 環境では、アクティブ・システムと代替システムで QUERY OTMATI コマンドを発行できます。

パラメーターを指定せずに QUERY OTMATI コマンドを発行すると、IMS では以下が表示されます。

- IMSplex 内の各 IMS インスタンスのワークロードと、ターゲット・メンバー (tmember) 名およびトランザクション・パイプ (tpipe) 名
- 総ワークロードのサブセットとしてキュー内に格納されているアクティブ・メッセージの総数
- トランザクション・インスタンス・ブロック (TIB) が存在する時間の長さを入力メッセージの関連 ID

この情報を使用して、OTMA 入力メッセージの処理に問題があるかどうかを特定できます。

QUERY OTMATI コマンドに SHOW パラメーターを指定すると、キュー内のアクティブ・メッセージの総数は表示されません。代わりに、他の特定のパラメーターによってフィルターに掛けられて、個々のアクティブ・メッセージが表示されます。例えば、経過時間が 5 秒以上であるトランザクション・コードの数を表示する場合は、以下のコマンドを発行します。

```
QUERY OTMATI MSGAG(5) SHOW(TRAN)
```

このコマンドを発行すると、個々の TIB と、経過時間が 5 秒以上であるトランザクション・コードの数が表示されます。この情報から、ストレージが制御ブロックで埋め尽くされるなど、修正または回避の可能なメッセージ・キュー内のすべての潜在的な問題を特定することができます。または、IMS を再始動せずに、IMSplex 内のアウトバウンド・トランザクションのワークロードのバランスを取ることができます。

MSC ネットワークでのパフォーマンス情報の調整

複数システム結合機能 (MSC) の一部となっている各システムに関する IMS システム・ログには、それぞれのシステムで発生したイベントだけが記録されています。リンクで受信されたトラフィックのロギングも記録されます。

チェックポイント・インターバルを記録したシステム・ログ記録に、結合されているすべてのシステムのシステム識別を追加することもできます。メッセージ・キューにあって処理されていないトランザクションがあることが分かっているので、これは報告書の内容を解釈する際に役立ちます。さらに、リモート・ソースからの追加トランザクションのロードを予期できます。分析プロシージャには、他のシステムのトランザクションによって起動されたプロセスを分離する方法を組み込んでください。

システム間プロセスを含む通常のアクティビティをモニターするための要件を満たすために、マスター端末のオペレーター間での IMS モニターと他のトレースの

スケジュールを調整してください。モニター期間を完全に同一にする必要はありません。ただし大幅に異なる場合、報告書要約を平均してしまうと、システム間メッセージによって起動されたプロセスの影響を解釈することが困難になることがあります。

関連資料: MSC 報告書の解釈の詳細については、899 ページの『MSC の IMS モニター報告書』を参照してください。

DB/DC 環境および DCCTL 環境での高速機能システムのモニター

メッセージ・ドリブン高速機能アプリケーション・プログラムを含む IMS オンライン・システムをモニターする際に、特に重要となるのは、迅速な応答と高いトランザクション率とのバランスです。高速機能を使用すると、パフォーマンス・データがシステム・ログ情報の一部になります。

また、IMS モニターを使用して高速機能のアクティビティを監視することができます。IMS 高速機能ログ分析ユーティリティを使用して、システム・ログ・レコードから統計報告書を生成することができます。このユーティリティは高速機能のトランザクション・ロードの要約、例外的な応答時間を強調する報告書、システムで時間内に発生するキー・イベント間の経過時間の分析を提供します。

モニター方法、パフォーマンス・プロファイル、分析プロシージャの設定というシステム管理作業は、高速機能環境で行う必要があります。

関連資料: IMS モニターまたは IMS 高速機能ログ分析ユーティリティについて詳しくは、「IMS V15 システム・ユーティリティ」を参照してください。

DB/DC 環境および DCCTL 環境でのトランザクション・フロー

1 つのトランザクションを実行すると、イベントが特定のシーケンスで発生します。メッセージ関連プロセスは IMS 内で非同期に実行されます。つまり、このプロセスは従属領域のプロセスとは関連していません。

このようなプロセスの例として、メッセージ・トラフィック、編集、書式設定、リカバリー関連メッセージのエンキューがあります。これらのプロセスはいずれも、アプリケーション・プログラムによる別のトランザクションのプロセスと並行して実行できます。アプリケーション・プログラムのスケジューリングから終了までのイベントは、PST に関連付けられており、同期イベントとして見なすことができます。

オンライン IMSIMS システムが、混合トランザクションを並行処理する際に発生するアクティビティの全体像を理解するには、以下の図を参照してください。この図ではイベントの発生シーケンスを示しています。各イベントについては、図の下にある注記で説明します。

IMS システムを測定する際の基準となる作業単位はトランザクション (または、入力メッセージの入力から、これに応じた 1 つ以上の出力メッセージの受信までの 1 つの会話型反復処理) です。作業単位のフローを表す方法として、以下の図に示すように、このフローを、すべてのトランザクションが通過しなければならない 3 つの「じょうご」で表現してみます。この図の中央には、トランザクションの応答時

間に影響する主要なイベントが番号付きで示されています。図の左側には関連するメッセージが、右側にはアプリケーションに関連するメッセージが記載されています。矢印は、1つのトランザクション・フローをトレースしています。(この図では、複数の経過時間に分散されるページング・エレメントやシステム・チェックポイントのプロセスは示されていません。)

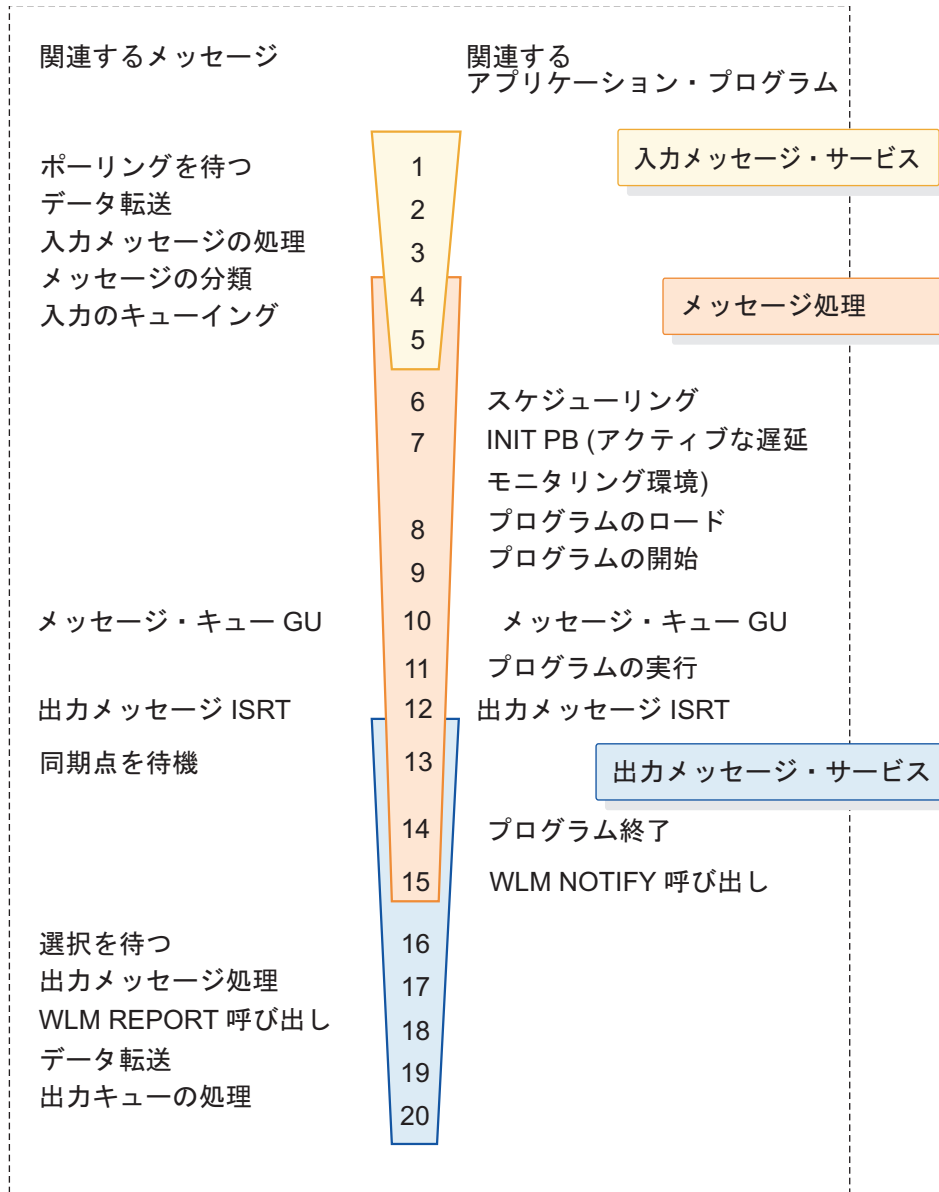


図 45. IMS におけるトランザクション・フローでのイベントのプロセス

図に関する注:

1. ポーリングの待機

Enter キーを押してから、ポーリングを受信する (これによりチャネル・プログラムによってデータが読み込まれる) までの時間。伝送路上にある様々な制御装置が原因で、多少遅延することがあります。これは、各制御装置が、1つの装置が送信可能になるまで順番にポーリングされるためです。伝送路で他の

端末へ (あるいは他の端末から) 伝送されるデータによって遅延が発生する場合もあります。また、この時点では、IMS 通信により伝送路上の端末からの入力メッセージが処理されている可能性があり、ポーリングは再開していません。ポーリング待機時間を測定するには、何らかのハードウェア・モニター・プログラムが必要になります。

2. データ転送

この時間には、マルチブロック入力メッセージに対する伝搬遅延とモデム送受反転時間が含まれます。伝送されたデータの量が分かっているならば、データ転送時間を見積もることができます。

3. 入力メッセージ処理

入力メッセージが HIOP で使用可能になった時点で、IMS によるトランザクションの管理が開始されます。メッセージがこのプールに入っている時間、MFS によるプロセスにかかる時間、メッセージ・キュー・バッファへの移動にかかる時間は、応答時間に影響します。形式ライブラリーへのそれぞれのトランザクション入出力は、メッセージ・キューに影響を及ぼします。応答時間を判別する場合の主要な考慮事項は、それぞれのプールが、入力キューイングへと続くトランザクションの現在のボリュームを格納できるくらい大きいかどうかです。特にメッセージ・キュー・プールが小さ過ぎると、メッセージ・キュー・データ・セットのオーバーフローが発生します。

4. メッセージの種別

これは、入力メッセージの WLM サービス種別を入手する z/OS WLM 呼び出しです。

5. 入力キューイング

入力キューに入っている時間、またはメッセージ領域が使用可能になるまでメッセージ・キュー・バッファで待機する時間です。頻繁に使用されるシステムでは、この時間が応答時間の大部分を占めます。使用可能な領域にスケジューリングされたプログラムのパターンと領域占有率は、重要なので注意してモニターしてください。

6. スケジューリング

クラス・スケジューリングのため、トランザクションがキューに入っている間、領域がアイドル状態になります。スケジューリング・パラメーターを指定すると、次のようになることがあります。

- PSB の競合またはメッセージ・クラスの優先順位が原因で、スケジューリングが終了する。
- 処理内容の競合が発生したために、スケジューリングが終了する。
- 意図リスト、PSB、または DMB 用の IMS.ACBLIB への入出力によるスケジューリングの拡張を行う。
- PSB または DMB プールでプール・スペース障害が発生する。

7. PB 呼び出しの開始 (遅延モニター環境の活動化)

メッセージが従属領域に入ると、そのメッセージに対して WLM 遅延モニター環境がアクティブになります。WLM PB は、サービスの種別、およびトラ

ンザクション名、メッセージ到着時刻、プログラム実行開始時刻 (現在時刻)、ユーザー ID などによって初期化されます。

8. プログラム・ロード

イベント 9 を参照してください。

9. プログラムの初期設定

スケジューリング後、アプリケーション・プログラムが始動可能になる前に、次のようなプロセス・イベントが発生します。

- 従属領域の内容監視
- プログラム・ライブラリーとそのディレクトリーの検索
- プログラム・ライブラリーからのプログラム取り出し
- メッセージ・キューへの最初の DL/I 呼び出しが行われるまでのプログラムの初期設定

モニターのために、上記のアクティビティーにかかった合計の時間を入手することができます。入出力の数は定期的にチェックする必要があります。

10. メッセージ・キュー GU

これは、メッセージ・キューへの GU 呼び出しです。このイベントは、システム・ログに記録され、プログラムの 1 つのスケジューリングで複数のメッセージが処理される場合の反復プロセスの開始点として使用されるため、測定ポイントとして選択されています。

11. プログラムの実行

出力メッセージ挿入への最初のメッセージ呼び出しからのプログラム実行時間です。これは各トランザクションの基本統計値です。実行される作業に関連してこの時間を検討することが重要になります。

- スケジュール当たりで処理されるトランザクションの数
- トランザクション当たりの DL/I 呼び出しの数とタイプ
- トランザクション当たりの入出力の数

経過時間をプロセッサ時間と入出力に分解することにより、重要なリソースを使用するトランザクションの判別が容易になります。

12. 出力メッセージの挿入

これにより、出力応答の非同期プロセスが開始されます。処理される出力メッセージ要求は「じょうご」へと流れますが、アプリケーション・プログラムは他の入力メッセージを処理しているか、あるいはクローズ・プロセスとプログラム終了プロセスを実行しています。

13. 同期点の待機

プログラムから出力メッセージが発行されると、このメッセージはプログラムが同期点に達するまで一時宛先でエンキューされます。MODE=MULT として指定されているプログラムでは、プログラムが 1 つのスケジューリングで複数のトランザクションを実行する場合に、出力の伝送で長期に渡る遅延が発生することがあります。プログラムが終了するまでは、前の出力メッセージが伝送用に解放されることはありません。プログラムで障害が発生すると、現在のト

ランザクションはすべてバックアウトされます。実際、メッセージがデキューされて一時宛先から永久宛先に移動する時には、LIFO シーケンスが使用されます。MODE=SNGL の場合は、同期点 (メッセージ・キューへの次の GU が実行される時点) の待機は通常無視できます。

14. プログラム終了処理

13 (490 ページ) のイベントで説明した MODE=MULT の場合については、プログラム終了処理時に同期点が発生します。データベース更新がデータベース・バッファ・プールから除去され、待機中の出力メッセージが解放されます。

MODE=SNGL の場合は、同期点は直前のメッセージ・キューへの GU 呼び出し (QC 状況コードで指定された GU) で発生します。また、最後のメッセージ・キューへの GU 呼び出しの後にアプリケーション・プログラムがデータベースを更新した場合を除いては、終了プロセス時にデータベース・コミット処理は発生しません。

15. WLM 通知呼び出し

これは、アプリケーション・プログラムが実行を終了したことを WLM に通知します。PB と現在時刻が WLM に渡されます。

16. 選択の待機

これは、入力時のポーリングの待機に似ていますが、出力メッセージがエンキューされる時点で伝送路上に処理中のポーリングがない場合、出力メッセージがポーリング・サイクルの完了まで待機する必要がない点が異なります。ただし、伝送路上の別の端末へのデータ伝送中に待機しなければならないことがあります。頻繁に使用されるシステムでは、出力キューで待機する時間が応答時間の大部分を占めることがあります。

17. 出力メッセージの処理

このアクティビティは 3 (489 ページ) のイベントに似ています。

18. WLM Report 呼び出し

これは、応答が送信中であることを WLM に通知します。IMS は、入力メッセージ到着時刻、サービス種別、現在時刻 (出力送信時刻) を渡します。

19. データ転送

このアクティビティは 2 (489 ページ) のイベントに似ています。

20. 出力キューの処理

送信された出力メッセージは、受信されたことが端末によって確認された後、デキューされます。ページ出力の場合、出力メッセージの受信は別の入力が発生するか、または端末から PA2 が入力されることで確認されます。

関連概念:

7 ページの『DB/DC 環境』

DB/DC 環境および DCCTL 環境での IMS モニター

IMS 提供の基本的なモニター・ツールが IMS モニターです。このツールは、DB/DC 環境ではこの制御プログラムの必須部分です。バッチ環境で IMS モニターに相当するツールは、DB モニターです。

IMS モニターは DB/DC 環境の稼働中にデータを収集します。すべてのディスパッチ・イベントに関する情報がモニター・レコードの形で収集され、順次データ・セットに入れられます。IMS モニターのデータ・セットは制御領域 JCL の IMSMON DD ステートメントに指定されます。/TRACE コマンドで IMS モニターを活動化すると、データがこのデータ・セットに追加されます。MTO は、システムの状況を認識しながらモニターの開始および停止を行って、複数のスナップショットを取得することができます。

関連資料: IMS モニター報告書の読み方については、837 ページの『第 5 部 IMS 報告書の使用』を参照してください。

トランザクション・レベル統計のモニター

IMS は、トランザクションに関する統計データを OLDS の X'56FA' ログ・レコードに記録します。このログ・レコードを使用して、一般的な業務目的のために、統計データの収集、分析、および解釈を行うことができます。

この統計情報は、X'07' ログ・レコードへ記録される情報のサブセットにすぎません。X'07' ログ・レコードからの特定の有効範囲 (リカバリー単位など) 内のトランザクションが収集されます。トランザクションに関する統計情報は別個に X'56FA' ログ・レコードにも記録されるため、X'07' ログ・レコードから抽出するよりも簡単に、この情報にアクセスできます。

IMS は非メッセージ・ドリブン・アプリケーションについて、以下のいずれかの場合に、リカバリー単位内のトランザクション (アプリケーション・プログラムによって同期点間で行われた作業) に関する統計情報を、ログ・レコード X'56FA' に記録します。

- キーワード `MODE=SNGL` が `TRANSACT` マクロ内で指定されている場合
- `CMTMODE(SNGL)` パラメーターが `CREATE TRAN` または `UPDATE TRAN` コマンドで指定されている場合

IMS はメッセージ・ドリブン・アプリケーションについて、以下のいずれかの場合に、各メッセージの後のトランザクションに関する統計情報をログ・レコード X'56FA' に記録します。

- キーワード `MODE=MULT` が `TRANSACT` マクロ内で指定されている場合
- `CMTMODE(MULT)` パラメーターが `CREATE TRAN` または `UPDATE TRAN` コマンドで指定されている場合

このロギングは、非メッセージ・ドリブン・アプリケーションの場合はプログラム単位で、またメッセージ・ドリブン・アプリケーションの場合はトランザクション単位で、使用可能または使用不可にすることができます。このロギングをトランザクション単位で使用可能または使用不可にするには、以下のコマンドを使用します。

- 既存のトランザクションの場合は、UPDATE TRAN または UPDATE TRANDESC コマンドを発行して、新しい TRANSTAT() パラメーターを指定します。
- 新規トランザクションの場合は、CREATE TRAN または CREATE TRANDESC コマンドを発行して、新しい TRANSTAT() パラメーターを指定します。

このロギングをプログラム単位で使用可能または使用不可にするには、以下のようになります。

- 既存のアプリケーション・プログラムの場合は、UPDATE PGM または UPDATE PGMDESC コマンドを発行して、新しい TRANSTAT() パラメーターを指定します。
- 新規アプリケーション・プログラムの場合は、CREATE PGM または CREATE PGMDESC コマンドを発行して、新しい TRANSTAT() パラメーターを指定します。

このロギングをシステム定義時にグローバルに使用可能または使用不可にするには、新しい DFSDFxxx PROCLIB メンバーの診断統計セクションで、新しいパラメーター TRANSTAT= Y | N を指定します。この設定は、システム定義プロセスを使用して作成されるすべてのトランザクションおよびアプリケーション・プログラムに適用されます。

IMS のコールド・スタートの後にオンライン変更プロセスまたは宛先作成出口ルーチン (DFSINSX0) を使用して作成されるトランザクションまたはアプリケーション・プログラムは、DFSDFxxx PROCLIB メンバーの DIAGNOSTICS_STATISTICS セクションに指定されている TRANSTAT= パラメーター設定を継承します。

あるトランザクションまたはアプリケーション・プログラムについて、このロギングが使用可能に設定されているかどうかを判別するには、以下のいずれかのタイプ 2 コマンドを発行します。

- QUERY TRAN
- QUERY TRANDESC
- QUERY PGM
- QUERY PGMDESC

DBCTL 環境でのモニター手順

このトピックでは、DBCTL 環境でのモニター手順の確立方法について説明します。最初に、DB/DC 環境でのモニターとは、通常、トランザクションのモニターを意味することを考慮してください。

エンド・ユーザーが端末から入力したトランザクションは、DB/DC 環境で処理され、その結果がエンド・ユーザーに戻されます。測定対象となるトランザクションの特性には、合計応答時間やリソース競合の数と期間などがあります。

DBCTL 環境では、トランザクションはなく、端末のエンド・ユーザーがいません。ただし、CCTL 端末のユーザーが入力した CCTL トランザクションの代行処理は行います。DBCTL でのモニターを実行すると、CCTL トランザクションが DBCTL 環境のデータベースにアクセスする時に発生する DBCTL プロセスに関する

るデータが作成されます。このアクセスは、データベース・リソース・アダプター (DRA) 要求を行っている CCTL によって行われます。

CCTL トランザクションに相当する DRA 要求の最も一般的なシーケンスは、次のようになります。

- DBCTL 環境で PSB をスケジュールする SCHED 要求
- データベース呼び出しを行う DL/I 要求
- データベース更新をコミットして PSB を解放する同期点要求である COMMTERM

この要求を含む DBCTL プロセスは、リカバリー単位 (UOR) と呼ばれます。

DBCTL が提供する UOR モニター・データには、次のようなものがあります。

- UOR が存在している期間の合計時間
- PSB のスケジューリングまでの待機時間
- データベース呼び出しでの入出力アクティビティー

この情報は DB/DC モニター・データと類似しています。DB/DC モニター・データと同一のこともあります。ただし、DBCTL 環境では、UOR データは CCTL トランザクションの全体プロセスの一部だけにしか相当しません。CCTL トランザクションのパフォーマンスについて全体的な情報を入手するには、CCTL モニター・データも含めてください。

このトピックでは、「トランザクション」という用語は CCTL トランザクションを指します。UOR に該当する場合は、特に UOR と記述します。

CCTL の管理者は、トランザクションのパフォーマンスをモニターする方法を決定しなければなりません。このトピックでは、いくつかのモニター方法と、IMS 機能がその中でどのように機能するかについて説明します。

使用できるモニター方法がいくつかあります。次の作業を行うことができます。

- オンライン実行全体にかかる実際の作業負荷を計算する。この処理は継続的に行われるか、または承認された時点で行われます。作業負荷の合計量または選択された代表的なトランザクションを追跡することができます。
- ピーク・ロード時でのサンプル・スナップショットと通常の状態でのサンプル・スナップショットをとる。次の 2 つの理由から、ピーク期間のモニターは常に有用です。
 - ボトルネックと応答時間の問題は、ピーク・ボリュームではさらに顕著になります。
 - 現行のピーク・ロードにより、将来の平均値がどのくらいになるか分かりません。
- 文書化されたパフォーマンス基準を持つクリティカル・トランザクションまたはプログラムをモニターする。

前もってモニター手順の計画を立てておいてください。プロシージャでは、使用するツール、分析方法、このようなアクティビティーによる操作の範囲、アクティビティーの実行頻度が説明されている必要があります。

次の作業が必要です。

- パフォーマンス基準の作成
- モニター、データ収集、分析のマスター・プランの作成

DBCTL 環境でのパフォーマンス目標の確立

オンライン DBCTL/CCTL システムでは、パフォーマンス目標はそれぞれに固有のもので、パフォーマンス目標の確立は主要なタスクであり、DBCTL/CCTL の作業負荷全体のデータの収集を必要とします。作業負荷を定義し、必要となるリソースの見積もりを行った後、目標とする応答と達成可能であると想定した応答とを調整する必要があります。システムのパフォーマンスをモニターして、これらの目標が達成されるかどうかを判別します。

パフォーマンスの目標は次の項目を基に設定します。

- 目標とする受け入れ可能な最大応答時間
- トランザクション当たりの平均および最大リソース要求 (または作業負荷)
- 予測されるトランザクションのボリュームと実際のトランザクションのボリューム

パフォーマンス目標の確立は繰り返し行われる処理で、次の作業が含まれます。

- CCTL ユーザーを基準にしたパフォーマンスの目標と優先順位の設定

これらの目標や優先順位は、システム提供のサービス機能を CCTL のエンド・ユーザーが認知する方法が元になっています。これらの目標は端末のエンド・ユーザーが確認する予想応答時間を示します。

応答時間の目標を確立する場合、この目標はシステム内時間 (最後の入力メッセージ・セグメントの入力から最初の応答セグメントまでの経過時間) だけでなく、IMS、CCTL、アプリケーション・プログラムの期待プロセス量も反映していなければなりません。応答時間に関して、平均、90 %、あるいはワーストケースについて、基準を定義するかどうかを検討してください。どれを定義するかは、インストール・システムの監査制御機能と特定のトランザクションの特性によって異なります。

- ユーザーを基準とした目標に対するパフォーマンスの測定方法とユーザーへの報告方法の決定

UOR データと CCTL トランザクション・データ結合して完全なトランザクション・プロファイルを作成できますが、この場合には次の点に注意してください。

- CCTL でトランザクションあたり 1 つの UOR だけしか許可されていない場合、1 対 1 の比率を使用するとモニター・データの結合が容易になります。この比率により、追跡も容易になります。これは、CCTL がトランザクション ID を SCHED 要求に入れることができるためです。DBCTL はこの ID をモニター・ログ・レコードに記述し、モニター報告書に出力します。
- CCTL が、1 つのトランザクション内で複数の UOR を使用することを許可している場合、各 UOR のデータを追加することによって合計 UOR データが作成されます。

UOR を終了させた要求の完了時には、IMS モニターのデータに加えて UOR データも CCTL に提供されます。このデータは上記のいずれの場合にも使用できます。CCTL は、トランザクションに関連する UOR 統計をリアルタイムで分析することもできます。

同じ UOR データが、UOR の状況と UOR の終わりを記録する IMS システム・ログの一部にもなっています。

- 現在の作業負荷の理解と記録

全体の作業量をカテゴリーに分類し、各カテゴリー別に、次の項目を含む作業負荷プロファイル (通常は見積もり) を作成する必要があります。

- トランザクション・カテゴリーの定義 (例えば、トランザクション・タイプやトランザクション・グループなど)。トランザクション・カテゴリーには次の 2 つの特性があります。
 - トランザクション・プロファイル。大部分のトランザクションは 1 つの機能を実行し、識別可能な作業負荷プロファイルを持っています。プロセスの後半では、共通プロファイルを持つトランザクション・タイプは便宜上 1 つに合併されます。
 - トランザクション・ボリューム。記録されている作業負荷がすでに操作可能な場合、トランザクションの合計ボリュームを取得することができます。その他の場合は、トランザクションのボリュームが見積もられます。
- カテゴリーの相対優先順位。優先順位が変更される期間も含まれます。
- 作業のリソース要件
 - オペレーティング・システムが管理する物理リソース (実記憶域、DASD 入出力、端末入出力)
 - サブシステムが管理する IMS 論理リソース (制御ブロック、ラッチ、バッファ、データベース・リソース・アダプター (DRA) スレッドの数など)
 - 必要な CCTL リソース

トランザクションのリソース要求の基本プロファイルを取得する場合、専用マシンで DBCTL/CCTL を開始し、トランザクションをいくつか実行して初期設定とバッファ・プールの使用を完了します。次に IMS モニターを開始してトランザクション実行のサンプルを測定します。

基本トランザクション・プロファイルを使用して、トランザクション作業負荷を検討し、その作業負荷を軽減できるかどうかを調べることができます。このような設計上の変更を行うと、システムに大きな影響を及ぼし、その結果、システム全体での競合が発生します。基本プロファイルを実稼働環境のトランザクション・プロファイルと比較することができます。

- 取得したリソース要件とボリューム情報を、各作業カテゴリー別のシステムを基準にした目標に変換する。

この変換には、サポートされるトランザクション率 (ピーク期間を含む) と保管される内部応答時間プロファイルに関するステートメントも含まれます。

- システムを基準にした目標が妥当であることを確認する。

システムの初期化とシステム操作のモニターを実行した後、作業負荷の測定値に基づいて、(使用可能なハードウェアで) 目標が妥当なものであるかどうかを判断する必要があります。使用した見積値と測定値が大幅に異なる場合は、作業負荷を記した資料とシステムを基準にした目標をそれに応じて検討し直すか、システムをチューニングします。

DBCTL 環境でのモニター・アクティビティーおよび技法の決定

パフォーマンスのモニターと分析に関するマスター・プランを作成するには、動的、日次、詳細のいずれのモニターが最も要求に合っているかを判断し、それに応じてプランを作成する必要があります。

パフォーマンスのモニターと分析に関するマスター・プランを作成する場合には、次の項目を確立する必要があります。

- モニター・アクティビティーの基本スケジュール

操作手順とモニターを調整して、毎日行うデータ収集または明細データ収集の命令にオンライン・イベントのフィードバックを組み込んでください。

- モニターに使用するツール

データ収集に使用するツールは、動的モニター機能、毎日の統計収集機能、詳細モニター機能を装備している必要があります。

- 実行する分析の種類

モニター出力からどのデータが抽出されるデータを記録して、データのソースと使用法を識別してください。モニター・ツールが提供する定様式の報告書はデータのボリュームを編成するには役立ちますが、データ抽出やデータの削減を実行する際に利用できるワークシートを設計する必要があります。

- 検出結果の検証を行う担当者のリスト

モニター・データの分析結果と結論は、ユーザー連絡グループとシステム・パフォーマンスの専門家にも通知してください。

- チューニング推奨による DBCTL 環境と CCTL 環境の設計への変更の実装方法

この変更のインプリメント方法を、インストール・システムでの実稼働環境の変更頻度の基準とテストの基準にあわせて調整してください。変更の管理については、501 ページの『第 25 章 システム設計の変更』で説明します。

モニター・アクティビティーについて、次の 3 つの広いレベルで計画を立ててください。

- 動的

短期間でもパフォーマンス目標からかなり逸脱したものを検出するため、システムの操作を継続的に監視します。

このトピックでは、動的モニターに関連する IMS 機能と一般的な概念について説明します。CCTL に類似した機能 (DISPLAY コマンドなど) がある場合は、それらの機能も使用してください。

このレベルのモニターでは、/DISPLAY または QUERY コマンドの出力を、エンド・ユーザーへのフィードバックとともに使用すると役立ちます。リソース測定機能 (RMF) II は、プロセッサ、チャンネル、入出力装置の使用状況に関する情報を収集するために使用できます。

MTO は、オンライン IMS システムの動作に関する重要な情報源となります。MTO フィードバックでは、IMS モニター実行中の一連の状態が重要になります。この情報は、モニター・データの妥当性を確立する際に役立ちます。

/DISPLAY または QUERY コマンドを使用して取得できる状況情報を使用して、オンライン実行時のプロセス状況を取得する準備ができます。この状況には、プール・レベルとアクティブ領域を含めることもできます。メッセージまたは BMP 領域のスケジューリング前、ネットワークの一部分のシャットダウン時、ピーク・ロード時など、実動サイクルで事前に設定されている特定の時点で、トランザクションのプロセス状況とシステム・リソース・レベルの測定値を記録することができます。

- 目次

主要なシステム・パラメーターを毎日調べて記録します。日ごとの平均値とピーク期間 (通常 1 時間) の平均値を記録します。主要なパフォーマンス目標とこの平均値を比較して、適切でない傾向を見つけだします。

IMS システム・ログをオフライン・プロセスへの入力として使用して、日次の統計または定期的な統計をとることもできます。07 レベルと 08 レベルのログ・レコードには、UOR データが含まれています。

- 詳細

システム指向の目標と作業負荷プロファイルに対するパフォーマンスを分析するため、システム操作の詳細な統計を定期的に収集します。

このレベルでは、データの量が非常に多くなります。これには通常、イベントや作表の (発生) シーケンスも含まれます。タイミングは詳細レベルで報告されません。

このレベルのモニターでは、IMS モニターおよび汎用トレース機能 (GTF) などのような特殊なトレース・ツールが役立ちます。これらのトレース機能により、オンライン処理の詳細なサンプルを収集して、CCTL スレッド、バッファ・プールの使用状況、システム・データ・セット入出力を区別することができます。

関連資料:

- IMS モニターの使用方法については、「IMS V15 オペレーションおよびオートメーション」を参照してください。
- モニター方法とパフォーマンス問題の検出に使用されるツールについては、539 ページの『パフォーマンスの問題の識別と訂正』で説明します。

汎用パフォーマンス分析報告プログラム (GPAR) (プログラム番号 5798-CPR) のシステムは、報告書作成プログラム (IBM プログラムまたはユーザー作成プログラム) の基盤として設計されています。このプログラムは、IMS システム・ログ、IMS モニター・テープ、GTF トレースなどの順次アクティビティ・トレ

ースを要約する際に役立ちます。このプログラムには、ユーザーがパフォーマンス・データ・ログまたは VSAM 以外の順次データ・セットから作成したグラフを印刷する機能も備えています。GPAR GPAR は、ASAP II、VTAMPARS、GTFPARS の前提プログラムです。

第 25 章 システム設計の変更

このトピックでは、IMS のシステム設計で変更を要する場合に必要なとなる計画作業について概説します。

変更が必要となる原因として、次のことが考えられます。

- 新規アプリケーションの組み込み
- アプリケーション・パッケージの段階的実現
- すでに実動しているアプリケーション設計に対する変更
- ETO が使用されていない場合のネットワークに対する変更
- 現行レベルでの IMS システムの保守と、アプリケーション・プログラムの問題に対する修正の組み込み
- システム制御パラメーターのチューニング
- データベース・データ・セットの再分散
- ストレージ・デバイスの変更

これらの変更すべてが IMS システム定義に影響を与え、新たな中核と制御ブロックの作成が必要になります。上記の最後の 2 つの変更に必要なのは、JCL (ジョブ制御言語) の変更とデータベース再編成だけです。しかし、影響が広範囲に及ぶ可能性もあります。どのような設計変更でも、既存のシステムとすべてのエンド・ユーザーのニーズを守るために、変更の実施を管理することが重要です。実動システムへのほとんどの変更の実装は、事前のテストと、妥当性検査期間の入念なモニターにより最適に処理されます。

IMS オンライン・システムに対する変更を管理する作業には、次のアクティビティがあります。

- IMS システム・データ・セットの再割り振りと、実行ライブラリーの更新
- 適切なシステム定義作業の実行
- セキュリティー配置の検討、RACF ユーザー、グループ、およびリソース・プロファイルへの変更の追加
- 操作プロシージャのマニュアルの変更
- リカバリー・プロシージャのマニュアルの変更
- テスト方法と、実動モードへのサービスインのタイミングの認識
- パフォーマンスの影響の分析と予測の修正
- IMS モニターの使用の調整や、実施前後のシステムのモニター
- 比較パフォーマンス・データの収集

アプリケーションの変更の評価

既存の実動 IMS オンライン・システムの設計とチューニングには多くの人手と多くのマシン・タイムが必要になるので、変更の影響を最小限に抑え、それにより生じる可能性のある結果をできる限り識別する必要があります。

以下の表には、アプリケーション・プログラム (またはシステムの制御) の変更による影響がどのように評価できるかが要約してあります。このチェックリストは、影響を受ける IMS のエリアを示し、他にどのような管理上の責任エリアが影響を受けるかを示しています。

アプリケーション・プログラムの設計変更が通知されたり、段階的実現をトラッキングしている場合は、必要な物理システム定義の変更に関する情報を文書から取り出さなければなりません。また、定義した環境および動作制御が影響を受けるかどうかを評価できるようにするには、適切な詳細情報も必要です。

表の最初の列は変更の種類を示し、2 番目の列は文書から分離する変更イベントを示しています。例えば、アプリケーション・システムはネットワークの物理部分が停止した後に報告書を生成するために、バッチ・メッセージ・プログラムを追加します。次の情報を得ることができます。

プログラム

名前と PSB (プログラム仕様ブロック)、プログラム・サイズ、オーバーレイの使用

データベース

すでに定義済みの読み取り専用アクセス

トランザクション

ありません。

メッセージ形式

ありません。

出力 スプール出力プロシージャの名前

端末 ありません。

ネットワーク制御

ネットワークの可用性に何らかの変更があるかどうか

スケジューリング

BMP 領域: 時間、頻度、限界

出口ルーチン

ありません。

チューニング

N/A

セキュリティー

RAS 制御なし

これらの情報を使用して、APPLCTN マクロを定義し、システム定義 (CTLBLKS) を実行し、BMP 領域の JCL (ジョブ制御言語) を調整します。次に、その BMP スケジューリング指示を操作手順 (スプール出力印刷ステップを含む) に追加します。代替の方法としては、MODBLKS システム定義を行って実動モードへのサービスインのためにオンライン変更を使用するか、動的リソース定義機能を使用することが考えられます。

表 36. アプリケーション変更管理の評価

変更タイプ	変更イベント	影響を受ける領域	影響と処置
プログラム	コーディング修正	IMS.PGMLIB	文書、変更のログをとる
	データベース・アクセス	IMS.PGMLIB IMS.PSBLIB IMS.ACBLIB	DL/I 呼び出しのモニターとオンライン変更の使用
	追加	IMS.PGMLIB IMS.PSBLIB IMS.ACBLIB APPLCTN macro	プールとリソースのコンテンツ、およびオンライン変更の使用
データベース	アクセス	IMS.ACBLIB	モニター
	構造	IMS.ACBLIB	データベース再編成
	追加	JCL、IMS.ACBLIB DATABASE マクロ	操作手順とリカバリー手順、およびオンライン変更の使用
トランザクション	内容の変更	TRANSACT マクロ	メッセージ・キューの再計算、モニター
	追加	TRANSACT マクロ	アルゴリズムのスケジューリング、オンライン変更の使用
	作業負荷	メッセージ・キュー	ブロック化の再計算
	メッセージ形式	IMS.FORMAT	バッファとモニター、およびオンライン変更の使用
出力	z/OS ファイル	装置用 JCL	操作手順
	スプール印刷	LINEGRP マクロ、LINE マクロ	操作手順
	LTERM	TERMINAL マクロ	代替宛先、操作変更
端末	VTAM	システム定義マクロ	VTAM 生成、操作手順
	ETO	IMS.PROCLIB	ETO 生成
	その他	システム定義マクロ	オンライン用 JCL、操作手順
	ネットワーク制御	システム定義マクロ	VTAM 初期設定、MTO 手順
	スケジューリング	TRANSACT マクロ	メッセージ・クラス・アルゴリズム、操作手順
	出口ルーチン	DL/I	IMS.SDFSRESL
メッセージ編集		IMS.USERLIB	異常終了に対処する
セキュリティー		IMS.USERLIB	セキュリティーのニーズを満たすようにする
ETO		IMS.SDFSRESL	ETO のモニター
ユーザー記述子		IMS.PROCLIB	

表 36. アプリケーション変更管理の評価 (続き)

変更タイプ	変更イベント	影響を受ける領域	影響と処置
チューニング	JCL パラメーター	IMS.PROCLIB	オペレーターの変更の調整
	セキュリティー	EXEC パラメーター	操作の調整
	IMS.PROCLIB パラメーター	DFSDCxxx	操作の調整
	サインオン出口ルーチン	RACF、出口ルーチン	パスワードの検証および中核との調整
	端末	RACF、出口ルーチン	パスワードの検証および中核との調整
	トランザクション	RACF、出口ルーチン	パスワードの検証および中核との調整
	RAS	RACF および出口ルーチン	パスワードの検証および中核との調整

関連資料: DRD コマンドの使用について詳しくは、「IMS V15 システム定義」の『動的リソース定義プロセス』を参照してください。

アクティブ IMS システムへの変更されたアプリケーションの紹介

変更されたアプリケーション・プログラムをアクティブな IMS システムに導入する場合、一部の領域タイプには、UPDATE PGM START(REFRESH) コマンドが有用です。

UPDATE PGM START(REFRESH) コマンドを使用すると、プログラムがスケジュールされているすべての領域を検出し、それらを手動で停止することなく、アプリケーション・プログラムの変更を容易にロールアウトすることができます。

UPDATE PGM START(REFRESH) コマンドは、以下の領域タイプでスケジュールされたプログラムについてサポートされます。

- プログラムがスケジュールされており、プログラムが DFSMPLxx PROCLIB メンバーによってプリロードされていない MPP 疑似入力待ち (PWFI) 領域
- 指定されたプログラム名がスケジュールされている JMP PWFI 領域
- プログラムがスケジュールされており、WFI=YES として定義されたトランザクションを実行している MPP、JMP、およびメッセージ・ドリブン BMP の各領域

UPDATE PGM START(REFRESH) コマンドは、以下の領域タイプについてはサポートされません。

- プログラムが DFSMPLxx PROCLIB メンバーによってロードされた MPP 領域
- IFP 領域
- JBP 領域
- 非メッセージ・ドリブン BMP 領域

変更されたアプリケーション・プログラムの更新済みコピーがこれらの領域タイプに確実にロードされるようにするには、以下のステップを実行します。

1. /DISPLAY PGM コマンドを発行して、変更しようとするアプリケーション・プログラムを使用する関連トランザクションを識別します。
2. /DISPLAY TRAN コマンドを発行して、ステップ 1 で識別された各トランザクションのスケジューリング・クラスを識別します。
3. /DISPLAY ACT コマンドを発行して、ステップ 2 で識別されたスケジューリング・クラスを処理できる従属領域を識別します。
4. /STOP PGM コマンドを発行して、変更しようとするアプリケーション・プログラムを停止します。
5. プログラムをリフレッシュします。
6. ステップ 3 で識別されたすべての従属領域を停止して、再始動します。これらの領域を一度に停止して再始動すると、残りのアクティブな領域は、このプロセス中に停止されたプログラムと関連付けられていないトランザクションを処理できます。
7. 変更されたアプリケーション・プログラムを開始します。

非アクティブ ACB ライブラリーのオンラインでのサイズ変更

ACB ライブラリーのサイズ調整が必要になる場合があります。変更をオンラインにする方法はデータ・セットの割り振り方法 (JCL で割り振られているか、それとも動的に DFSMDA マクロを使用して割り振られているか) によって異なります。

割り振りに JCL が使用された場合は、IMS をダウンさせ、再始動して変更を認識させる必要があります。このタスクを行う際に IMS をダウンさせず済ませたい場合は、アクティブと非アクティブの両方の ACBLIB データ・セットに動的割り振りを使用し、その後、オンライン変更プロセスを実行して、サイズ変更した ACB ライブラリーをオンラインにします。

例

迅速なアプリケーション開発サイクルのために、ACB ライブラリーを頻繁に拡張しなければならない場合があります。アクティブと非アクティブの両方の ACB ライブラリーが DFSMDA マクロを使用して動的に割り振られている場合は、ACBLIB をサイズ変更するために IMS を停止させる必要はありません。

ACB ライブラリーの静的割り振りから動的割り振りへのマイグレーション

非アクティブ ACB ライブラリーを動的に割り振るために、DFSMDA メンバーを作成した後、IMS を再始動して、新しい DFSMDA メンバーを使用します。

DFSMDA マクロを使用した ACBLIB の動的割り振りにマイグレーションするには、以下の手順を実行します。

1. ACBLIBA および ACBLIBB データ・セット用の DFSMDA メンバーを作成します。
2. IMS および DL/I JCL プロシージャから、IMSACBA および IMSACBB DD ステートメントを除去します。

3. 新しい DFSMDA メンバーが常駐するデータ・セットを指す DD ステートメントを追加します。この DD ステートメントは、IMS STEPLIB 連結と IMSDALIB 連結のどちらに入っているかまいません。
4. IMS を停止してから、DFSMDA メンバーを使用して再始動します。

この時点で、IMS の停止と再始動を行わなくても以下のタスクを実行できるようになります。

- 非アクティブ ACB ライブラリー・データ・セットのサイズを大きくします。
- 非アクティブ ACB ライブラリー内のエラーを修正します。
- 追加データ・セットを非アクティブ ACBLIB 連結に追加します。

システム定義の変更の計画の立案

マクロ仕様に反映される IMS オンライン・システム設計は、変更されることがあります。初期インストール時の一度の実行でこの設計が決定されることはめったにありません。

ハードウェア変更が原因ではなく、アプリケーションの作業負荷および設計の変更が原因の場合は、実リソースの割り振りや端末接続が変わります。多くの場合、進行中のパフォーマンス・モニター・タスクによって、オンライン・システム設計の要素が変更されます。

システム定義プロセスの管理

システム定義に対する変更を計画する場合は、通常、一連のマクロ変更を調整しなければなりません。IMS システム定義ステージ 1 入力は、制御点として示されません。

マクロに対する変更のパラメーター値の正確度を検討して、その値の導出方法についての適切な説明を追加します。提案された変更の性質、変更の理由、適切な許可を記録する管理文書を持っていないとできません。実行する処置または計画している処置を項目ごとに記入することによって、その記録を完成することができます。

また、システム定義ステージ 2 プロセスのタイミングも調整しなければなりません。必要な JCL の変更が加えられた制御プログラムの新規バージョンは、運用のサービスインに結び付いていないとできません。

1 つ以上のマクロ・ステートメント上のパラメーターを再指定するには、全システム定義または完全なステージ 2 ジョブ・ストリームの処理は必要ありません。JCL ステートメントに指定されている他の値でパラメーター値をオーバーライドすることにより、多くのチューニング変更を行うことができます。マクロ内容にもっと大規模な変更が発生するまで、新しい値を制御プログラムの不変部分にするのを延期することができます。このようなオーバーライドはすべて、PROCLIB の内容およびオペレーターの指示で調整する必要があることを考慮に入れてください。

必要なシステム定義タイプの判別

全システム定義を実行しなければならないか、部分的なシステム定義の実行でよいかは、変更されたマクロ・ステートメントによって決まります。しかし、特定のキーワードを変更すると、その場合のみ、全部の生成が必要になります。

ステージ 2 の実行中に行われるプロセスの量は非常に多いので、選択した生成タイプの経済的側面を可能な限り生かすようにすべきです。

関連資料: 選択するシステム定義タイプの概要については、「IMS V15 システム定義」を参照してください。

DB/DC 環境および DCCTL 環境でのネットワーク定義の変更

付随するバッファ・ストレージのチューニングを行っている IMS オンライン・システムに接続されている端末は、頻繁に変更できます。静的 VTAM 端末をシステムに導入する場合は、中核を再生成しなければなりません。(VTAM サポートを以前に定義していない場合は、オンライン・システム定義が必要です。) 同時に、特定の端末タイプに対するサポートを削除できます。

VTAM 端末を導入して ETO を使用する場合は、中核を再生成する必要はありません。既存の記述子でサポートされている ETO 端末を追加すると、IMS は適切な ETO ログオン記述子を使用して端末を定義します。

端末の削除を検討するときに、端末が物理的に取り外され、メッセージを LTERM にルート指定できない場合は、定義を所定の位置に置いておく方が効率的です。後で定義を変更するときに、削除することができます。

初めて MSC を追加する場合あるいは、物理リンクを追加する場合は、オンライン・システム定義が必要です。MSC MTM、TCP/IP、または VTAM 接続を削除する場合は、中核のみを再生成する必要があります。リンク名またはシステム識別名を変更する場合は、再生成された制御ブロックを再バインドしてください。

関連タスク:

➡ ログオン記述子の作成 (コミュニケーションおよびコネクション)

➡ 拡張端末オプション (ETO) (コミュニケーションおよびコネクション)

システム・チューニングの変更

進行中のモニター、およびエンド・ユーザー・フィードバックについての認識を通して、発生する可能性のあるチューニング作業を確認しておかなければなりません。

パフォーマンス関連のチューニング作業のツールと方法については、465 ページの『第 24 章 IMS モニター・データの収集および解釈』と 515 ページの『第 26 章 システムのチューニング』で説明しています。これらのセクションは、パフォーマンス管理が次のことからなる反復作業であるということを前提にしています。

データ収集

モニター機能とトレース機能の使用

データ整理

報告書処理プログラムと手作業プロシージャの使用

データ分析

問題標識の検出

変更の実施

解決策の提案とシステムの変更

この計画には、チューニングと保守の項が含まれていなければなりません。この項の項目を他の必要な変更と組み合わせて、余分なシステム定義プロセスを省き、操作プロシージャの変更の頻度を減らすことができます。全体的な設計変更ログによって、将来の実施が検討されている提案だけではなく、分析および承認されている提案を常に把握しておくことができます。このような管理文書は、矛盾する変更を見つけるのにも役立ちます。

IMS オンライン・システム設計に影響を与える変更に関する文書には、変更理由と、システムのどのパラメーターまたはコンポーネントを変更する必要があるかについてのチェックリストが含まれていなければなりません。つまり、操作、データベース保守、リカバリー、セキュリティー、調整のモニターなどに関する項がなければなりません。

リソース使用状況の変更

特定の変更を加えることにより、仮想記憶域のさらに効率的な使用を促進し、既存システムの入出力を最小にすることができます。

これらの変更には、以下のものが含まれます。

- 制御領域に対する実行パラメーターの再指定
- メッセージ・キューに対する DASD ストレージの割り振り
- データベース再編成とデータ・セットの分散
- 64 ビット・ストレージへの ACB の配置

アプリケーションおよびデータベース設計の変更

アプリケーション・パッケージのパフォーマンス調査の結果として、オンライン・システム設計に対する変更が必要になることがあります。

これらの変更は、IMS.PGMLIB でのアプリケーション・プログラムの単純な更新から、IMS.DBDLIB および IMS.ACBLIB での DBD および ACB の再定義に及ぶ可能性があります。IMS.PGMLIB でアプリケーション・プログラムを変更した場合は、正しいディレクトリー項目を BLDL リストに格納するために、メッセージ処理領域を停止して再始動するか、あるいは適用可能であれば UPDATE PGM START(REFRESH) コマンドを使用する必要があります。このコマンドをアプリケーション・プログラムの変更で使用できる場合は、UPDATE PGM コマンドの説明を参照してください。

関連資料:

 UPDATE PGM コマンド (コマンド)

DB/DC 環境および DCCTL 環境での通信設計の変更

通信装置と IMS との間の伝送速度の障害を検出した場合の対応は、ネットワーク定義の変更の場合と同じ計画に含まれています。

オンライン・システム定義の変更の管理

アプリケーションの追加要求、または、現行の一連のプログラム、トランザクション、およびデータベースの使用法の変更要求では、オンライン・ユーザーに対する中断が生じる可能性があります。完全なシステム定義と IMS の再始動を強制する必要はありません。要求を検査して、オンライン変更または動的リソース定義 (DRD) プロセスで要求を満たすことができるかどうかを調べることができます。

IMS ネットワークに対する変更や現行 IMS システム定義に定義されているような静的端末の使用法の変更を必要としない要求の場合は、IMS システムをオンラインで実行中に、オンライン変更プロセスを使用して変更を行うことができます。IMS システム・データ・セットの準備を、「IMS V15 システム定義」の『オンライン変更のための IMS システム・データ・セット』の説明に従って管理してください。端末ユーザーおよびオンライン・システム操作への影響を評価することも重要です。

関連資料:

- DRD を使用したシステム定義の変更について詳しくは、「IMS V15 システム定義」を参照してください。
- オンライン変更機能の概要については、570 ページの『ローカル・オンライン変更機能の概要』を参照してください。

システム修正変更がオンライン変更を使用できるかどうかの判断

オンライン変更で保守できる変更のチェックリストを、以下の表に示しています。

このリストは、ステージ 1 システム定義マクロに与える影響に基づいています。これは、ステージ 1 入力制御点と見なされるからです。

表 37. オンライン変更で実施できるシステム定義リソースの変更

システム定義マクロ	実施できるオンライン変更
APPLCTN	PSB (アプリケーション) とその属性の追加 属性の変更 PSB の削除
DATABASE	データベースとその属性の追加 属性の変更 データベースの削除
RTCODE	宛先コードと照会属性の追加 宛先コードの削除
TRANSACT	トランザクションとその属性の追加 属性の変更 トランザクションの削除

表に示されている変更には、MODBLKS 生成が必要です。また、これに対応したセキュリティの変更のためには、RACF セキュリティー・プロファイルを更新して、セキュリティを必要とする新規 IMS リソースを反映させる必要もあります。

また、システム定義プリプロセッサを使用すると、変更準備が容易になります。リソース名を追加または変更する場合に、プリプロセッサは無効な名前や重複している名前を検出でき、システム定義が正しく実行されることを保証するのに役立ちます。変更修正の制限については、『オンライン変更を計画する際の考慮事項』に説明があります。端末ネットワークに影響を与える変更修正は行えません。

オンライン変更を計画する際の考慮事項

一連の変更をオンライン変更で対処できるかどうかを評価する場合には、いくつかの制限を考慮に入れなければなりません。一般に、ステージ 1 処理で実行されるチェックでは、オンラインで実施できない変更を行ったかどうか、ユーザーにはわかりません。

次の影響を考慮する必要があります。

- APPLCTN マクロ

メッセージ・クラスが PGMTYPE パラメーターの一部として割り当てられる場合、そのクラスは、現在そのシステムに定義されているメッセージ・クラスの最大数を超えることはできません。

高速機能トランザクションの場合は、システム内で高速機能がアクティブになっていなければなりません。

別の IMS システムへのトランザクションのルーティングには、MODBLKS の生成に対してシステム名 (SYSID パラメーター) と MSC の使用が新規に定義されていないことが必要です。

RESIDENT 特性および DOPT 特性に変更を加えることができる場合でも、RESIDENT として定義されている PSB は、次の再始動後まで非常駐として機能します。これは、IMS システムの初期設定時に PSB を常駐にするアクションが発生するからです。

常駐 PSB のスケジューリング属性を変更すると、次の IMS の再始動まで、PSB は非常駐になります。

BMP プログラムがメッセージ処理プログラムになると、メッセージ・スケジューリングを制御する TRANSACT マクロで定義されたトランザクション特性は、次の再始動後まで有効になりません。しかし、MTO は /ASSIGN コマンドを使用して、特定のトランザクションに適したメッセージ・クラスと処理優先順位を指定できます。これにより、このトランザクションは通常メッセージ・スケジューリングに適したものになります。

UPDATE PGM コマンドを使用して、状況と属性の両方を更新できます。APPLCTN マクロの特性に対して行った一部の変更はオンライン変更処理の一部

として実装されず、IMS オンライン・システムの次回のコールド・スタート時のみ有効になります。それらの特性は、以下のとおりです。

TRANSTAT

- DB/DC 環境および DBCTL 環境での DATABASE マクロ

RESIDENT 特性を追加できる場合でも、DMB をデータベース常駐と関連付けるプロセスは、IMS オンライン・システムの次の再始動後まで有効になりません。

ACCESS パラメーターに対する変更は、オンライン変更の一部ではありません。この変更は、/START DATABASE コマンドで処理できます。別の方法として、UPDATE DB START(ACCESS) コマンドを使用することもできます。

MSDB には、どのような種類の変更も加えることはできません。

- RTCODE マクロ

このマクロ・ステートメントの追加または指定の変更は、システムで高速機能がアクティブである場合のみ可能です。既存の高速機能ユーザー入力編集ルーチンが、すべての追加された宛先コードを処理できることを確認してください。

- TRANSACT マクロ

/ASSIGN、/MSASSIGN、/START、/STOP などのコマンドを使用して、このマクロによって指定した特性のいくつかを制御できます。また、UPDATE TRAN を使用して状況と属性の両方を更新することもできます。TRANSACT マクロの特性に対して行った一部の変更はオンライン変更処理の一部として実装されず、IMS オンライン・システムの次回のコールド・スタート時のみ有効になります。それらの特性は、以下のとおりです。

PRTY
PROCLIM
PARLIM
SEGNO
SEGSIZE
SYSID
TRANSTAT

高速機能利用可能として指定されたトランザクションに対しては、現行システムで高速機能がアクティブになっている必要があります。

トランザクションを別の IMS システムにルーティングするには、現行システムで MSC 機能がアクティブになっている必要があります。現行システムで以前に定義されなかったシステム名 (SYSID パラメーター) は採用できません。

トランザクションに対して指定される編集出口ルーチンは、すでに現行 IMS オンライン・システムの一部になっていなければなりません。

- ページ固定

IMS の次の再始動までは、追加された制御ブロックに対してさらにページ固定が行われることはありません。

- EMHB サイズ

トランザクション固有の EMHB サイズの追加または変更にはオンライン変更を使用する場合は、新しい EMHB サイズが EPSESRT サイズよりも大きくないことを確認してください。EPSESRT サイズの決定は、初期設定中にしか行えません。

通常のトランザクション処理中に、IMS は入力メッセージのサイズを EMHB 長および EPSESRT 長と照合します。入力メッセージが EMHB 長または EPSESRT 長を超えていると、メッセージ DFS0444 を出力してリジェクトされます。

容量計画の実行

他のトピックでは、IMS オンライン設計の処理要件の短期または中期の変更計画について述べていました。しかし、設計の長期にわたる妥当性を調べなければならない場合もあります。

例えば、作業負荷の傾向を推定する場合に、十分な計算能力が使用可能かどうかなどです。

以下を使用することができます。

- 個別のトランザクション・プロファイル
- 全体的な処理プロファイル
- プロセッサ使用率 (パーセント) の算定

パフォーマンス・モニターの結果として、トランザクション・プロファイルを作成できます。これらのプロファイルの一部は、傾向分析プロファイルになる場合もあります。平均到着速度とピーク到着速度、つまり、現在予測される実際のサンプル、および将来の予測を判別しなければなりません。全体的なプロセスの概要は、IMS モニター実行プロファイル報告書から得られます。

容量の制限を調査する 1 つの方法は、トランザクションの容量を推定することです。いくつかのモニター・ポイントに対するプロセッサ使用率から、傾向を把握することができます。パフォーマンス測定からの一般的な観測は、マシン・サイクルの増加が IMS 専用システムのトランザクション (秒単位) の増加とほぼ比例することを示しています。使用率に対するトランザクション負荷の増加を示すいくつかの点をプロットすると、使用率としてのトランザクション数の限界が 100% に近づくことがわかると推定することができます。

また、予測最大作業負荷が、実際の上限よりも下である使用率に対応しているかどうかを調べることができます。作業負荷は、追加のトランザクションを伴う現行の全体的なプロファイルを評価することによって推定できます。

このような予測を行うときに、十分な仮想記憶域と、増加するトランザクション負荷を処理するのに十分な入出力構成が必要です。その他の使用可能な入出力リソースと仮想記憶リソースについて十分に考慮する必要があります。これらのエレメントの競合は、プロセッサ使用率のレベルよりも重大な影響をパフォーマンスに与えるからです。

z/OS 下で動作する大型システムを管理する場合には、CSA に対する要件を減らしたい場合があります。DL/I 分離アドレス・スペース、および、必要に応じてロッ

ク管理のための IRLM 用ローカル・ストレージを使用する計画を立てることができません。予測した作業負荷を使用して、さまざまな構成を持つ仮想記憶域を推定することもあります。

第 26 章 システムのチューニング

パラメーターを変更して、IMS オンライン・システムの効率化を図ることができます。

このような目的でパラメーターを変更すると、IMS は次のように効率化されます。

- パフォーマンス基準を達成する。
- リソースの有効に使用する。

465 ページの『第 24 章 IMS モニター・データの収集および解釈』では、パフォーマンス目標の設定とモニターについて説明します。これらの作業は、IMS オンライン・システムのアクティビティをチューニングする前に行う必要のある前提作業です。このセクションで定義される用語については、後で再度説明することはありません。分析に必要なデータは、次の 1 つ以上のツールの報告書作成機能のレベルによって異なります。

- ログ・トランザクション分析ユーティリティ (DFSILTA0)
- 統計分析ユーティリティ (DFSISTS0)
- IMS モニター
- GPAR で実行する報告書作成プログラム: IMSASAP II
- リソース測定機能 (RMF II)

このセクションでは、IMS オンライン・システムの一般的なチューニング方法を説明しています。このセクションのトピックでは、パフォーマンス管理について説明した後、チューニングについて次の 2 段階に分けて説明します。

- z/OS および IMS パラメーターの初期設定
- パフォーマンスの問題の識別と訂正

それぞれの作業ごとに、対象となる主要分野が識別されています。それぞれの領域で行う可能性のあるチューニング作業について検討してください。これはチューニング方法の概要であり、パフォーマンスに関する正式な研究結果に代わるものではありません。ただし、この章の内容を使用すると、ほとんどの主要な問題について明記できるようになります。

システムのチューニングには、z/OS およびサブシステムのパラメーターの初期設定と、チューニングに関連する問題の識別と訂正が含まれます。この 2 つのトピックは、初期段階の問題が発生するのを防ぐための手引きと、進行中に発生する問題の解決方法として読むことができます。ただし、まず最初に、これらの項目が、パフォーマンス管理の他の項目とどのように関係しているかについて理解してください。

チューニング作業とシステムの再定義は、IMS のパフォーマンス管理の作業の一部であり、さらに以下の図に示すように、IMS のパフォーマンス管理アクティビティ・サイクルの一部です。システムのパフォーマンスについて判断する場合には、設計の決定と予測も行います。

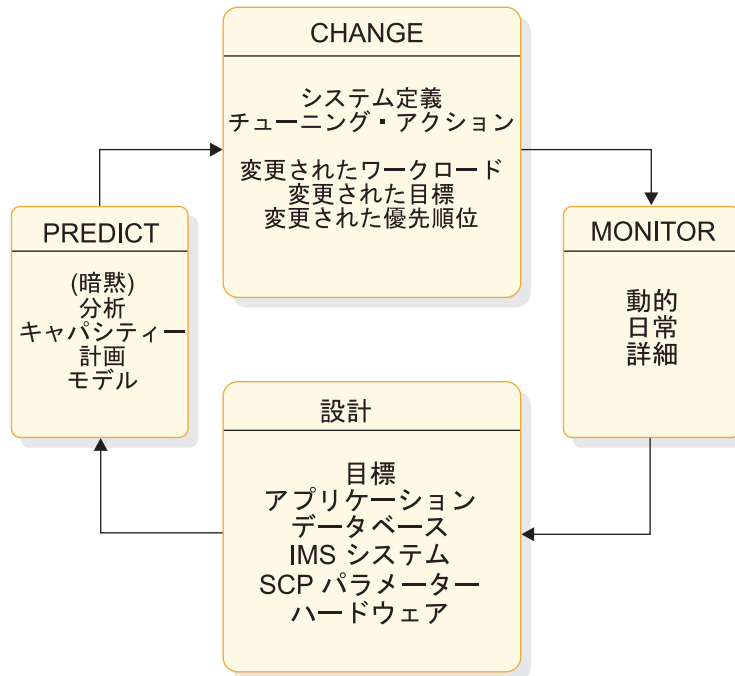


図 46. IMS パフォーマンス管理アクティビティ・サイクル

関連資料:

- MSC リンクのモニターおよびチューニングについては、「IMS V15 コミュニケーションおよびコネクション」を参照してください。
- 高速機能のチューニングについては、「IMS V15 データベース管理」を参照してください。
- 共用キュー環境のチューニングについては、288 ページの『共用キュー環境でのパフォーマンスのチューニング』を参照してください。

パフォーマンスの管理

IMS のパフォーマンス管理作業は、一連の目標を達成するために設計された一連のアクティビティからなります。

目標は、以下のものからなります。

1. 各サブシステム (IMS、TSO、バッチ) のパフォーマンス目標とその優先順位を確立する。
2. 定義されたパフォーマンス目標を達成するために、MVS とサブシステム・パラメーターを初期設定する。
3. 全体的なシステムの操作をモニターし、パフォーマンスの目標が達成されていることを確認する。

モニターには、以下の 3 つの広いレベルがあります。

- 動的モニター。確立した目標と実際のパフォーマンスとが離れているかどうかを確認します。

- 主要なパフォーマンスの目標に関する日次モニター。パフォーマンスの傾向や、発生する可能性のあるパフォーマンスの問題の初期段階の兆候をモニターします。
- 定期的に行われる詳細モニター。このモニター結果から、パフォーマンスについて検討し、チューニング要件を決定します。

これらのモニター・レベルについて詳しくは、465 ページの『第 24 章 IMS モニター・データの収集および解釈』を参照してください。

4. モニターした問題を識別して、この問題を訂正する。この際に、ユーザーへの通常のサービスへの悪影響を最小限に抑える。
5. 容量計画または傾向分析を行って、近い将来に計画した負荷が原因でパフォーマンスの問題が発生しないようにする。

パフォーマンス目標値およびモニター操作の確立については、445 ページの『システムのモニター』で扱います。容量計画または傾向分析の実行については、512 ページの『容量計画の実行』、および「IMS V15 オペレーションおよびオートメーション」の『IMS Performance Analyzer for z/OS の使用』に説明があります。

変更管理機能

パフォーマンス管理とは、実際は変更の管理です。IMS システムで変更が数回しか行われない場合は、パフォーマンスは安定しています。しかし、実際には継続的に変更が行われる環境の方が一般的です。このような環境では

IMS システムは新しいアプリケーション・プログラムの追加に合わせて調整され、以下の変更に対応しなければなりません。

- アプリケーションの設計
- トランザクションのボリューム
- データベースのボリューム
- 混在する作業負荷
- ユーザーの数
- ユーザーの優先順位
- IMS 以外の作業の量
- ハードウェアの構成

IMS チューニング機能は、効率的なトランザクション処理のためにリソースを割り振ろうとします。

さまざまな手法を使用して、運用中のシステムをチューニングすることができます。例えば、次のような方法を使用できます。

- 新しいメッセージ領域を開始する。
- トランザクション PARLIM または PROCLIM、あるいはクラスの割り当てを変更する。
- メッセージ領域のクラス割り当てを更新する。

これらの作業には、領域、プール・スペース、制御ブロック領域などの IMS リソースが十分あるかどうかを確認する作業も含まれています。IMS の構成は、インス

ツール・システムでの物理リソースの制約事項に一致している必要があります。通常、仮想記憶域を使用すると、IMS によって制御される一部の入出力アクティビティーが制限され、この結果、DASD 装置に格納されているデータに対する競合が減少します。チューニング作業は重大な状況を管理する作業ではありませんが、変更や作業負荷の傾向にあわせて継続的に行われる補償作業です。

チューニング作業の体系的な実行方法については、539 ページの『パフォーマンスの問題の識別と訂正』で説明します。

設計変数

IMS システムのパフォーマンスは、様々な設計と作業負荷との相互関係によって異なります。

パフォーマンスは次のようなものから影響を受けます。

- IMS システム定義
- IMS の実行時オプションとパラメーター
- アプリケーションの設計
- データベースの設計
- オペレーティング・システムの構成
- ハードウェアの構成

上記の項目については、『チューニングのための z/OS および IMS パラメーターの初期設定』で説明します。設計段階では、システムの稼働時のモニター結果に基づいて決定した設計とチューニングに関する変更事項の確認も行われます。

予測のタイプ

予測には、暗黙の予測と明示的な予測の 2 つがあります。

暗黙的

チューニングの変更としてオンライン IMS の設計が変更される場合、パフォーマンスの改善が暗黙に予測されます。

明示的

予測を行うために何らかのアクションが行われると、分析モデルまたはシミュレーション・モデルが使用され、容量計画が作成されます。

チューニングのための z/OS および IMS パラメーターの初期設定

IMS パラメーターと z/OS パラメーターの初期設定ステップには、パフォーマンスに影響を与える可能性のある操作が含まれています。

これらのパラメーターを設定する場合は、システムの他の点についても考慮してください。例えば、これらのパラメーターの推奨値に矛盾するかまたはこれよりも優先する設計要件または機能要件がある場合は、代わりに使用する値を評価する必要があります。

z/OS および IMS パラメーターの初期設定で行うアクティビティーについて、以下のトピックで説明します。

z/OS ディスパッチング優先順位の割り当て

システム全体について、調整済みのディスパッチング優先順位の方式を選択してください。この方式は、並行して実行されるすべてのサブシステムの相対的な価値を反映していなければなりません。

IMS 制御領域は、システム・タスクとして保護キー 7 で動作します。これにより、一部の監視機能での実行命令の数が減少します。

ジョブ・ディスパッチング優先順位

ジョブ・ディスパッチング優先順位により、使用可能なマシン・サイクルに割り当てられるジョブのシーケンスが指定されます。制御領域よりも高いディスパッチング優先順位を持つジョブが存在すると、干渉が発生します。VTAM または TCAM (JES や z/OS のマスター・スケジューラーと同様) は、通常、高い優先順位で実行されますが、IMS 制御領域と従属領域については他の作業 (主に TSO およびバッチ) の順位を考慮に入れて検討する必要があります。

従属領域のディスパッチング優先順位

制御領域と従属領域の両方が優先順位処理を取得することが重要になります。IMS の並列 DL/I 機能により、大部分の DL/I 呼び出しプロセスは従属領域で実行されます。割り当てられたディスパッチング優先順位があまりに低いと、アプリケーション・プログラムの迅速な保守ができなくなります。通常、BMP 領域の優先順位は MPP の優先順位よりも低く設定します。ただし、BMP 領域の優先順位は MPP の優先順位の直後に指定してください。これは、BMP と MPP が、プログラム分離エンキューなど、同一の制御領域リソースに対して競合するためです。通常、TSO の『第 1 番目の期間』の作業は IMS 制御領域よりも低くなりますが、従属領域よりは高くなります。TSO の『第 2 番目の期間』の作業とバッチ処理は通常 IMS 従属領域よりも低くなります。

z/OS でのパフォーマンス・グループ番号の設定

各 IMS 領域に、固有なパフォーマンス・グループ番号を割り当てることができます。これにより、Resource Measurement Facility™ (RMF) 報告統計を領域ごとに別々に取得できます。

DL/I 分離アドレス・スペース・オプションを使用する場合は、制御アドレス・スペースと DL/I アドレス・スペースを同一の RMF パフォーマンス・グループに含めることがあります。

DBCTL 環境での優先順位の設定

DBCTL 環境では、CCTL 領域の優先順位は DBCTL 領域よりも高くなければなりません。つまり、CCTL 領域の優先順位はどのバッチ処理領域よりも高くなければなりません。

CICS は、DBCTL 領域よりも高い優先順位で実行できます。ご使用のシステムで使用可能なワークロードおよびリソースに合わせて、DBCTL および CICS 領域の優先順位を調整してください。

ディスパッチング優先順位

ディスパッチング優先順位は、z/OS ワークロード・マネージャー (WLM) によって設定され、一部は、WLM ISPF パネルで各 IMS アドレス・スペースに割り当てる相対的重要度に基づいています。

関連資料: WLM 重要度の割り当てについて詳しくは、468 ページの『ワークロード・マネージャーと IMS』を参照してください。

次の図は、WLM が認識しているすべてのアドレス・スペースに割り当てられた、相対的ディスパッチング優先順位の WLM ISPF 表示の例を示しています。各アドレス・スペースのディスパッチング優先順位は、列 DP に 16 進値として示されます。X'FF' が最高優先順位です。アドレス・スペースが持つ値が小さいほど、その優先順位は低くなります。

NP	JOBNAME	StepName	ProcStep	JobID	Owner	C	Pos	DP	Real	Paging	00000100
	ALLOCAS	ALLOCAS				NS	FF	806	0.00	00000110	
	IOSAS	IOSAS	IEFPROC			NS	FF	256	0.00	00000120	
	IXGLOGR	IXGLOGR	IEFPROC			NS	FD	803	0.00	00000130	
	SMS	SMS	IEFPROC			NS	FE	329	0.00	00000140	
	SMF	SMF	IEFPROC			NS	FF	363	0.00	00000150	
	JES2AUX	JES2AUX				NS	F9	120	0.00	00000160	
	IRL2194	IRL2194	IRLM	STC00312	+++++++	NS	FD	283	0.00	00000170	
	NETMONC4	NETMONC4	SYSTEM	STC00005	+++++++	NS	FB	240	0.00	00000180	
	JES2	JES2	IEFPROC			NS	F9	3095	0.00	00000190	
	VTAM94	VTAM94	IEFPROC	STC00002	+++++++	NS	FD	2492	0.00	00000191	
	SOFIMSP	SOFIMSP		STC00006	+++++++	LO	FF	226	0.00	00000192	
	MPC4A04	REGION		JOB00323	USRT008	A NS	F3	385	0.00	00000200	
	MPC4A05	REGION		JOB00324	USRT008	A NS	F3	385	0.00	00000300	
	MPC4A06	REGION		JOB00325	USRT008	A NS	F3	385	0.00	00000400	
	MPC4A07	REGION		JOB00326	USRT008	A NS	F3	385	0.00	00000500	
	MPC4A08	REGION		JOB00327	USRT008	A NS	F3	385	0.00	00000600	
	SCIC94	SCIC94	IEFPROC	STC00313	+++++++	NS	FD	460	0.00	00000700	
	OMC94	OMC94	IEFPROC	STC00314	+++++++	NS	FD	442	0.00	00000800	
	CQSC94	CQSC94	IEFPROC	STC00315	+++++++	NS	F7	1052	0.00	00000900	
	RMC94	RMC94	IEFPROC	STC00316	+++++++	NS	FD	507	0.00	00001000	
	I91CTL4	I91CTL4	IEFPROC	STC00317	+++++++	NS	F7	12T	0.00	00001100	
	I91DLSC4	I91DLSC4	IEFPROC	STC00318	+++++++	NS	F5	27T	0.00	00001200	
	I91DBRC4	I91DBRC4	IEFPROC	STC00319	+++++++	NS	FD	617	0.00	00001300	
	BPX0INIT	BPX0INIT	BPX0INIT			IN	FF	162	0.00	00001400	
									0.00	00001500	

図 47. WLM ISPF パネルに表示される相対的ディスパッチング優先順位 (列 DP)

パフォーマンスに対する IMS オプションの選択

パフォーマンスに関する考慮点は、多くのパラメーターと関連しています。これらのパラメーターは、システム定義用に選択されているもの、実行時オプションに選択されているもの、設計仕様の決定事項の一部であるものなどがあります。

以下のリストで、多数のオプションについて説明します。

- IMS の最適化とアプリケーション・モジュールのロード

仮想記憶域の使用が制限されていない場合は、可能であれば、IMS 再入可能モジュール、領域コントローラーおよびプログラム・コントローラー、頻繁に使用する

る高水準言語モジュール、再入可能アプリケーション・コード、オーバーレイ監視プログラムにはページング可能リンク・パック域 (PLPA) を使用してください。

IMS プロシージャで実行時パラメーター SRCH=1 を使用すると、STEPLIB または JOBLIB の前にジョブ・パック域とリンク・パック域が検索されます。

- プログラム・ライブラリーに関する考慮事項

実動プログラムが格納されているプログラム・ライブラリーを、メッセージ領域の STEPLIB 連結の初めに指定してください。IMS.PGMLIB のプログラムのシーケンスは、使用頻度に基づいて降順で指定してください。

プログラム・ライブラリーのシステム検索順序は、プログラム・ロードで効率が下がる原因になることがあります。STEPLIB または JOBLIB は、最初に検索されるプログラム・ライブラリーです。これらのいずれも使用しない場合は、IMS.PGMLIB を SYS1.PARMLIB の LNKLSTnn メンバーの先頭に指定してください。IMS.PGMLIB の位置は IMS.SDFSRESL よりもかなり重要です。その理由は IMS.PGMLIB の内容がシステム初期設定時にのみ必要というより、アプリケーション・プログラムのスケジューリングでも引き続き必要になるためです。検索時間とロード時間は重要です。これは、プログラムのスケジューリング時にこれらの時間がパフォーマンス低下の原因となるためです。ライブラリーのトラック・サイズいっぱいまでプログラムをブロック化してください。リンケージ・エディターの DC オプション (下位互換のために使用される) によって、プログラム・ブロックのサイズが 1024 バイトになります。このオプションを使用せずに、再度バインドを行ってください。

- 従属領域の BLDL リスト

項目のリストは、プログラムのディレクトリー索引が含まれている従属領域に保管されています。このリストには最もアクティブで最後に使用された項目を基準に保管しているため、プログラム・ディレクトリーへの入出力は減少します。このリストの項目を持つプログラムは、使用頻度の低いプログラムに比べて、スケジューリングから最初の DL/I 呼び出しまでの経過時間が短くなります。メッセージ処理領域のデフォルトの項目数 20 をオーバーライドするには、DFSMPR プロシージャの EXEC ステートメントに DBLDL パラメーターを指定します。メッセージ処理領域の最大項目数は 9999 です。

新規プログラムまたは変更されたプログラムのテストに使用される領域の場合は、DBLDL パラメーターを 0 に設定して、実行のためのロード時に必ずプログラムの最新バージョンがロードされるようにします。DOPT パラメーターを指定すると、高速スケジュール変更が使用不能になります。

- アプリケーション制御ブロックの配置

ACBLIB データ・セットへの読み取り入出力の量を減らすために、DFSDFxxx PROCLIB メンバーの DATABASE セクションに ACBIN64=ggg を指定することにより、ACB メンバーを 64 ビット・ストレージにロードできます。指定する値 (ggg) は、PSB および DMB ACB メンバーに割り振る 64 ビット・ストレージの量 (ギガバイト単位) です。

64 ビット・ストレージ機能の ACB メンバーを使用可能に設定した場合、ACB メンバーはアプリケーション・スケジューリング時に、ACBLIB データ・セットからでなく、64 ビット・ストレージから取り出されます。

- アプリケーション・プログラムの制御

TRANSACT マクロ・ステートメントに `MODE=SNGL` を指定して、メッセージ・キューの入出力アクティビティを減らし、応答時間を短くします。

時間制限により制御されるバッチ・メッセージ・プログラムを実行する場合には、IMSBATCH プロシージャのパラメーター `STIMER=2` を選択します。これにより、`STIMER/TTIMER` マクロが、`STIMER=1` を指定した DL/I 呼び出しごとではなく、スケジューリング別に 1 回だけ発行されます。同様に、メッセージ処理領域を制御する場合には、DFSMPR プロシージャの定位置パラメーター `STIMER` に値 2 を設定します。

IMSBATCH プロシージャでは、パラメーター値 `SPIE=0` と `TEST=0` を指定して、実動プログラムで不必要な SVC が削除されるようにします。DFSMPR プロシージャでは、これに相当する定位置パラメーターは `SPIE` と `VALCK` です。

PL/I の場合、最新リリースの最適化コンパイラを使用して、初期設定と終了のオーバーヘッドを削減します。

- チェックポイント頻度の設定

チェックポイントの頻度を調整して、チェックポイントが 10 分から 20 分に 1 回の割合よりも頻繁にとられないようにしてください。この頻度は、以下のいずれかの方法を使用して制御します。チェックポイント頻度を指定しなかった場合、IMS はデフォルト値の 500,000 を使用します。

- IMS PROCLIB データ・セットの DFSPBxxx メンバーで `CPLOG=` パラメーターを指定する
- JCL で `CPLOG=` パラメーターを指定する
- `/CHANGE CPLOG` コマンドを発行する

- メッセージ形式オプション

複数の DFLD を持つ DOF に `FILL=NULL` または `FILL=PT` を指定して、ブランク文字の伝送を最小限に減らします。

- キューイング・オプションの設定

優先順位処理を実行しない場合は、IOS キューイング・オプションを使用して、優先順位処理をオフに設定します。IOS オプションを設定すると、IMS はキュー内のすべてのジョブを『先入れ先出し』順で実行します。

- IMS リソースのページ固定

IMS 仮想記憶域が常に実記憶域によってバックアップされるようにする場合は、IMS.PROCLIB のメンバー DFSFIXxx にページ固定要求のリストを指定することができます。ページ固定は、IMS 制御領域の初期設定時に実行されます。キュー・マネージャーのバッファの長期にわたるページ固定を要求するには、制御領域 JCL の EXEC ステートメントに `EXVR` パラメーターを指定します。

バッファ・プールをチューニングして、システムのページング率を上げたり、仮想記憶域を制限することなく、入出力アクティビティを削減または最小化できるだけの十分な大きさにします。バッファ・プールのページ固定に関する考慮事項については、531 ページの『IMS バッファ・プールのページ固定』で説明します。

- DB/DC 環境および DCCTL 環境での DREF ストレージ用の IMS リソースの定義

IMS 仮想記憶域を補助記憶装置にマイグレーションしないという要求を指定することができます。IMS PROCLIB メンバーの DFSDRFxx メンバーに DREF 要求を指定することにより、仮想記憶域が拡張記憶域を超えて移動されることはなくなります。拡張ストレージが使用できない場合、ストレージはページ固定になります。

関連資料: DREF 要求について詳しくは、「IMS V15 システム定義」を参照してください。

- MVS のページ固定に関する考慮事項

IMS システムが頻繁に参照する MVS モジュールは、初期プログラム・ロードでページ固定される必要があります。

特定のモジュールをページ固定することにより、タイマー機能が起きるたびに『ページ固定および待機』 SVC が保管されます。これらのモジュールは、なるべく隣接するようにまとめて配置して、ページング可能リンク・パック域 (PLPA) のページ数を最小化してください。

関連資料: MVS モジュールについて詳しくは、「z/OS MVS 初期設定およびチューニング ガイド」を参照してください。

IMS リソース (バッファ・プールを除く) に対する競合の回避

並行領域のインターリーブ・プロセスが、IMS リソースの競合の主な原因になります。混在するトランザクションを処理する一連の領域は、プロセスを並行して実行し、順次実行のマルチタスク化を行うことができます。

次のリストに、スケジューリングおよび領域の効率化のパターンに関係するいくつかの分野を示します。

- DB/DC 環境および DCCTL 環境でのクラス・スケジューリングの使用

クラス・スケジューリングの使用について判断する際に、次の点を考慮する必要があります。

- クラスを使用して、優先順位が低く、実行時間の長いトランザクションを優先順位の高い作業から分離する。
- クラス構造を調整して、優先順位の高いクラスの作業が複数の領域でスケジューリングされる可能性を高くする (ただし、プログラム分離の競合など、何らかの理由でこのようなスケジューリングが望ましくない場合を除く)。マルチサーバー・システムでは、通常シングル・サーバー・システムよりも待機時間が短くなります。

- トランザクションが入力キューで待機している間に、メッセージ領域がアイドル状態にならないようにするため、(優先順位の低い) 追加クラスを 1 次クラスまたは領域に割り当てられているクラスよりも下のメッセージ領域に割り当てる。クラスの作業が常に処理を待つ状態になるかどうか分からない場合は、1 つのクラスだけをサービスする領域がないようにする。
- トランザクションのボリュームが非常に多い場合は、メッセージ領域を占有して、プログラムを入力待ちプログラムとしてクラス分けし、入力キューイング時間を削減し、スケジューリングとプログラム・ロードが行われないうにする。

- DB/DC 環境および DCCTL 環境での処理限界カウント

トランザクション・クラスと優先順位方式について検討します。TRANSACT マクロで PROCLIM パラメーターを設定して、他のトランザクションが待機中でメッセージ領域でのプロセスが可能な場合に、どのタイプのトランザクションでもメッセージ領域を占有することがないようにします。競合するトランザクションの応答時間が重要でない場合は、PROCLIM に 5 より大きい値だけを指定してください。この場合、優先順位の低いトランザクションには PROCLIM=1 を設定します。入力待ちトランザクションには、PROCLIM に適度に高い値 (あるいは最大値) を設定して、アプリケーション・プログラムの再スケジューリングが行われないうにする。疑似 WFI と高速スケジュール変更機能を使用して、プロセスのオーバーヘッドを除去します。高速スケジュール変更機能を実行することにより、領域が処理できるスケジュール当たりのメッセージ数が処理限界よりも多くなるため、アプリケーション・プログラムの不必要な再スケジューリングや再ロードが実行されなくなります。

注: 処理限界カウントが 0 のトランザクションに対してスケジューリングされた領域は、高速スケジュール変更を実行できないか、または疑似 WFI になりません。

- DB/DC 環境および DCCTL 環境での並列スケジューリング

順次にスケジューリングできるボリュームの小さいトランザクションには、並列スケジューリングを使用しないでください。並列プロセスにメッセージ領域を使用すると、プログラムの各スケジュールごとに複数のトランザクションが処理される領域のプロセス効率が下がることがあります。並列スケジュールは、単一メッセージ処理領域が「あふれて」しまうため、着信率の一時的なピークによってキューイングが過度に実行されるトランザクションに対して予約されていなければなりません。このような状況では、そのトランザクション・タイプの平均サービス時間と平均応答時間の目標に合った PARLIM レベルを使用してください。

- DB/DC 環境および DCCTL 環境での IMS メッセージ処理領域

IMS システムを運用する時に使用するメッセージ処理領域の数が多過ぎても少な過ぎても、パフォーマンスとトランザクションの応答時間に影響が出ます。多くの各種トランザクションを使用している環境では、占有レベルが非常に高い領域を注意深くモニターして、このような領域が原因で IMS メッセージ・キュー上にキューが作成されないことを確認してください。

使用されているトランザクションの種類が少ない環境では、各スケジュールの複数のトランザクションを処理する領域に、アプリケーション・プログラムが入っていることが多いため、通常高い占有レベルを維持できます。

占有レベルが高いことは、トランザクションの混在、トランザクションの特性、クラス・スケジュールの方式、応答時間の目標に応じてその意味が異なってきます。一部のトランザクションでは、キューイングの遅延が長くても受け入れられることがあります。一方、他のトランザクションでは、ユーザーを満足させるために、キューイングの遅延を短くしなければなりません。

これとは対照的に、占有率の低い並行メッセージ処理領域は、バッファ・プール・スペース、実記憶域、IMS システム・データ・セット、データベース・レコードに対する領域間の競合を不必要に増加させることがあります (これはプログラム分離エンキューのために発生します)。

疑似入力待ち (PWFI) を MPP 領域始動プロシージャに指定することで、プロセス・オーバーヘッドを一部除去することができます。メッセージを処理する必要がない場合でも、PWFI メッセージ処理領域はアプリケーション・プログラムを終了せず、スケジューリングされたままの状態になります。これにより、不必要な終了と再スケジューリングが行われなくなります。

メッセージ処理領域の数を調整して、応答時間の目標を達成できる最小の数にしてください。領域の数を増やすのは、増加する作業負荷を目標の値で処理できない場合や、メッセージ・クラス構造、高速スケジュール変更、PWFI、および領域制御パラメーター (PARLIM、PROCLIM、WFI を含む) の変更で、応答時間の増加が指定されていない場合だけにしてください。従属領域は最大 999 まで指定できますが、最小数の領域で応答時間の要求に対応するようにしてください。

入力待ち領域を除いた必要な領域の数は、以下の例に示す式で見積もることができます。

$$\text{領域数} = \frac{((\text{着信率}) * (\text{領域当たりの処理時間}))}{(\text{望ましい領域占有率のパーセンテージ})}$$

着信率は、あるインターバルで受信したトランザクションの合計数を、そのインターバルを秒単位で表した値で割った値です。領域当たりの処理時間は、IMS モニター領域要約報告書から分かる、スケジュールと終了、スケジューリングから最初の DL/I 呼び出しまでの経過時間、実行経過時間の各平均時間の合計値として定義されます。

入力待ち処理専用の領域の占有率は 100 % であるため、この計算にはこの領域は含めないでください。

メッセージ処理領域の数を確立するだけでなく、次の確立によって、これらの領域へスケジューリングされるトランザクションを制御します。

- クラスと優先順位構造の選択
- TRANSACT マクロの PROCLIM パラメーターと PARLIM パラメーター
- DB/DC 環境および DBCTL 環境の CCTL 領域

CCTL アプリケーション・プログラムのパフォーマンスは、データベース・リソース・アダプター (DRA) リソースに対する競合の影響を受けることがあります。特に、許可されている DRA スレッドの最大数が PSB スケジュール要求を作成する CCTL アプリケーション・プログラムの数より少ない場合、一部の要求は、DRA スレッドが使用可能になるまで待機します。

- DB/DC 環境および DCCTL 環境の保管域セット

特定のトランザクション率について領域の数とその占有度を考慮する場合、入力データ伝送が遅れないことを確認してください。遅延の要因の 1 つに、並行端末入出力の数の制限があります。オンライン EXEC ステートメントの SAV パラメーターを使用して、この数の制限を調整することができます (最大 999 の並行入出力が可能です)。このパラメーターは、端末入出力の「動的保管域セット」の数と言われます。これは、これらの領域の 1 つがアクティブな各端末入出力を制御するためです。

- DB/DC 環境および DBCTL 環境のデータベース競合

データベースが少数のルートまたはルート・アンカー・ポイントを持たないようにしてください。データベースに対して複数のトランザクションが同時にアクセスすると、プログラム分離 (PI) の競合が発生する可能性があります。データベースの制御レコードまたは合計処理の実行を維持するレコードを使用すると、PI のエンキューが発生します。

データベースが少数のルートを持たないようにできない場合は、クラス・スケジュールを使用してこのように競合しているトランザクションをすべて単一の領域に割り振り、トランザクションを強制的に順次処理します。

IMS バッファ・プールのチューニング

全体的なシステム・パフォーマンスを改善するために、IMS バッファ・プールを構成することができます。

メッセージ・キュー・プールのチューニング

メッセージ・キュー・プールは、メッセージ・キュー・データ・セットのバッファ・プールとして機能します。不要な入出力アクティビティを削減するように、メッセージ・キュー・プールを最適化してください。

トランザクションの処理時にメッセージ・キュー入出力が発生する場合は、メッセージ・キュー・データ・セットは IMS モニター領域 IWAIT 報告書に IWAIT 項目として検出されます。短メッセージ・キュー・バッファと長メッセージ・キュー・バッファの LRECL サイズが相互に正しい比率にない場合、プールのサイズが十分あっても、混在するトランザクションによってバッファの 1 つ以上のセクションが使用されるため、入出力が発生します。

バッファの数は MSGQUEUE マクロに指定されています。EXEC ステートメントのメッセージ・キュー・バッファ (QBUF) パラメーターによって、実行時にこの数をオーバーライドできます。1 つのチューニング方法として、モニター・データ上でメッセージ・キュー入出力とマークされた IWAIT インスタンスの増加が確認できるまでバッファ数を減らしていきます。これで、QBUF パラメーター値に 1 ページ・サイズ単位を加算した値を使用できます。

メッセージ・キューの処理に関する特定の要求を識別します。このような要求には、次のようなものがあります。

- 大規模複数セグメントの入出力メッセージ
- セグメント挿入プロセス間の拡張プロセス
- リモート印刷出力

- 端末オペレーターによるページング
- 複数端末からの出力
- プログラム間通信回数の増加
- BMP の優先順位 0 のメッセージ
- 会話型トランザクションの SPA サイズ

SPA サイズは、適切に設計されたアプリケーション・プログラムの処理の入力として使用される単一セグメント・メッセージの効率を低減し、逆にメッセージ・キューのパフォーマンスに影響を及ぼします。


OTMA メッセージ接頭語のセキュリティー・データ・セクションにネットワーク・セキュリティー資格情報が含まれている場合、メッセージ・キュー・プールの使用量が増加します。


メッセージ・キュー LRECL のサイズには SPA サイズも含めておかなければなりません。これは、SPA がメッセージの最初のセグメントとしてこのキューに入れられるためです。SPA セグメントは、IMS.LGMSG データ・セットに書き込む場合には複数の論理レコードに分割できます。あるいは、LRECL サイズに大きい値を指定できます。どちらを選択するかは、大きな SPA サイズを必要とするトランザクションの相対数によって異なります。

ログ・トランザクション分析ユーティリティを使用して、入出力キュー時間をモニターできます。IMS 統計分析ユーティリティは回線と端末の報告書を作成します。この報告書は平均送信サイズと平均受信サイズを示しますが、長メッセージと短メッセージの個別の統計は示しません。各トランザクションおよび端末の平均メッセージ長とカウントは、システム・ログ・タイプ X'40' チェックポイント・レコードに含まれます。

関連資料:

 統計分析ユーティリティ (DFSISTS0) (システム・ユーティリティ)

 ログ・トランザクション分析ユーティリティ (DFSILTA0) (システム・ユーティリティ)

 IMS PROCLIB データ・セットの DFSFIXnn メンバー (システム定義)

メッセージ形式プールのチューニング

メッセージ形式バッファ・プールは、使用頻度の高いトランザクションのメッセージ形式ブロックを保持します。これを最適化すると、共通トランザクションの入出力の負荷が削減されます。

メッセージ形式バッファ・プールのストレージを取得するには、もっとも参照頻度が低い形式ブロックを維持しているスペースを解放して、着信率の高いトランザクションのブロックが通常使用可能になるようにしてください。形式ブロックの入出力を最小限に抑えるためには、最も使用頻度の高いトランザクションが使用する入力形式ブロックと出力形式ブロックのペアを保持するのに十分なスペースを、プール内に維持することを計画してください。



取り出し要求エレメント (FRE) を使用すると、プールの効率的な使用に影響します。各 FRE は、プール内の 1 つの形式ブロックを制御します。すべての形式プロ

ックの長さが 1000 バイトで 10 の FRE が指定されている場合、最大プール・スペースは 10000 バイトになります。FRE の割り当ては、システム定義中に BUFPOOLS マクロで行われますが、制御領域の EXEC ステートメントの FRE パラメーターを使用してオーバーライドできます。/DISPLAY POOL MFP コマンドを使用すると、フリー・スペースの容量を全スペースの容量と比較することができます。プールのサイズに対してフリー・スペースの割合が一貫して高い場合は、FRE の割り当てが原因で割り振り済みストレージを完全に使用できないようになっていないかどうか確認してください。

メッセージ形式バッファ・プールの使用状況を改善するためのもう一つの方法は、アプリケーション・プログラムが形式を共用できるようにするというものです。例えば、汎用メッセージ入力形式は、共通の DIF/MID ブロックを、入力メッセージ・フィールド編集出口ルーチンと入力メッセージ・セグメント編集出口ルーチンによって実行される後続の編集処理と共用することができます。

パフォーマンス関連の機能は、メッセージ形式サービス (MFS) で使用可能です。MFS サービス・ユーティリティを使用すると、形式ブロックのすべてのサブセットまたは選択されたサブセットの位置を指し示す直接ポインターのディレクトリを生成できます。この索引は、メッセージ形式バッファ・プールに常駐して、アクティブ IMS.FORMATA/B ライブラリー・ディレクトリ内の形式ブロックの位置を検索するのに必要な入出力を最小限にとどめます。(各索引項目には 14 バイト必要です。) 参照頻度の高いブロックには索引のストレージを割り振るか、またはブロックごとの入出力を減らすかのトレードオフを評価してください。

関連資料:

-  MFS サービス・ユーティリティ (DFSUTSA0) (システム・ユーティリティ)
-  IMS PROCLIB データ・セットの DFSFIXnn メンバー (システム定義)

PSB、PSBW、および DMB プールのチューニング

プログラム仕様ブロック (PSB) プール、PSB 作業 (PSBW) プール、およびデータ管理ブロック (DMB) プールのこれらのガイドラインにより、プール・スペースの障害を解決できます。

PSB プールと意図リストに関する考慮事項

PSB プールでは、スケジュールされたアプリケーション・プログラムの PSB がすべて保持されます。PSB は、他の PSB をロードするのにスペースが必要となるまでプールに存在します。すべてのスペースを使い果たした状態になると、参照される頻度の最も低い非アクティブ PSB が解放されます。常駐しており、並行スケジュールされていない PSB は、この規則には当てはまりません。このような PSB はプール割り振りから解放されず、通常、参照頻度の高い (通常、プリロード済み) プログラム用に指定されています。常駐の PSB は、IMS.PROCLIB データ・セット内のメンバー DFSFIXxx 用にコーディングされているモジュール固定リストにロード・リスト域 DFSPSBRs を入れることにより、ページ固定することができます。

DL/I アドレス・スペース・オプションが使用されている場合は、z/OS 共通域 (DFSPSBRs) と DL/I アドレス・スペース (DFSDLIRS) にそれぞれ 1 つずつ、2 つの常駐 PSB プールが存在します。

同様に、DFSFIXxx メンバーに DFSINTRS を組み込むことにより、常駐意図リストをページ固定することができます。意図リストを常駐リストにしてください。

PSB 作業プールのストレージ

各アクティブ PSB が PSB 作業プールの一部を必要としますが、全体のサイズを最大サイズよりも大きくする必要はありません。

大きなサイズを指定して開始してから、使用されている最大容量よりも大きい特定の値まで減らす方法を使用している場合は、プール・スペースの障害が発生した時点でスケジューリングが停止します。

DB/DC 環境および DBCTL 環境の DMB プールに関する考慮事項

DMB プールを、IMS データベースの DMB がすべて格納できる十分な大きさにしてください。DMB をプールにロードするときには必ず、当該データベースがオープンされており、スペースを解放するために他のデータベースはクローズされていなければなりません。このアクティビティにより、DL/I 呼び出しに必要な経過時間が増加します。この場合、OPEN と CLOSE を使用するよりも、制限されたページ不在を使用する方が適切です。アクティブなデータベースの DMB を常駐ブロックにしてください。ページ固定する対象としては、このような DMB が適切です。DMB プールのすべてのスペースを使っていない場合は、DMB に対する IWAIT が発生するまでサイズを減らしてください。IMS モニター領域 IWAIT 報告書に DMB= 項目が出力された場合、DMB プールのサイズが小さすぎることを示します。この時点で、BUFPOOLS マクロの DMB パラメーターの指定値を大きくすることができます。


PSB プール、PSB 作業プール、および DMB プールのスペースの障害

プール・スペースの障害は、いずれかのプール内のスペースが、並行して実行されている従属領域の現行数を維持するのに十分でない場合に発生します。各アプリケーション・プログラムは、その PSB、PSBW、意図リスト、および DMB をスケジュールに入れる必要があります。すべてのプールが、プール・スペース障害を発生させないのに十分な大きさであることを確認してください。この場合の原則は、DMB、PSB、または PSBW のセット (適宜) の一部である最大ブロック (2R+1) を収めるのに十分なスペースを確保することです。ここで R は、アクティブな従属領域の数です。このガイダンスでは、プール・スペース内のフラグメント化が考慮されています。不要な PCB を PSB から除去することにより、プール・スペースの所要量を削減できます。

関連概念:

944 ページの『IMS 内部リソースの使用』

関連資料:

 IMS PROCLIB データ・セットの DFSFIXnn メンバー (システム定義)

データベース・バッファ・プールのチューニング

IMS データベース・バッファ・プールに十分なサイズのバッファを割り振って、プールに以前入れられたデータを再度読み取る必要のあるアプリケーション・プログラムによる入出力を発生させないようにします。サブプールのサイズを慎重

に選択し、データベースのブロック・サイズとデータベースの参照頻度の突き合わせを行うことにより、不要な入出力が最小化されます。

高速機能バッファ・プールの場合、バッファ・プール・サイズをシステム定義プロセスの一部として明示的に指定するか (DBBF パラメーター)、高速機能 64 ビット・バッファ・マネージャーを使用して、高速機能データベース・バッファ・サブプールの数とサイズを制御することができます。高速機能バッファ・マネージャーを使用可能にするには、PROCLIB データ・セットの DFSDFxxx メンバーの FASTPATH セクションに FPBP64=Y を指定します。高速機能バッファ・プール・マネージャーを使用している場合、DEDB バッファ・プールは 64 ビット・ストレージに配置されますが、主記憶データベース (MSDB)、順次従属 (SDEP) セグメント、システム・サービス、およびバッファ・ヘッダー用のバッファ・プールは 31 ビット・ストレージで管理されます。

複数の OSAM サブプールを、特定のバッファ・サイズを持つよう定義して、特定のデータ・セットを特定のサブプールに出力することができます。ブロック・サイズが類似しているかまたはまったく等しい多くの OSAM データ・セットがある場合は、大きい 1 つのサブプールを複数のサブプールに置き換えるとパフォーマンスが改善されます。小さいサブプールを複数使用すると、セグメントを検索するためのプール・スキャンが減少します。また、この方法を使用して、多数のデータベース・セグメントにアクセスする優先順位の低いトランザクション (または BMP) がプール・スペースを占有するのを防ぐことができます。

ヒント: OSAM データベースの順次処理に必要な時間は、順次バッファリング (SB) の使用により、かなり削減できます。

ユーザーは、複数の VSAM ローカル共用リソース・プールを定義することができます。制御インターバル・サイズが類似しているかまたはまったく等しい多くの VSAM データ・セットがある場合は、大きい 1 つのサブプールを、同じサイズのバッファの複数のサブプールに置き換えるとパフォーマンスが改善されます。VSAM データ・セットに対してサイズが同一の複数のサブプールを生成すると、OSAM 複数サブプール・サポートと同様の効果が得られます。

複数のローカル共用リソース・プールを使用して、複数の同一サイズの VSAM サブプールを指定します。複数の共用リソース・プールを作成してから、他のローカル共用リソース・プールの VSAM サブプールと同じサイズの VSAM サブプールをそれぞれの共用リソース・プールに入れます。これで、特定のデータベースのデータ・セットを共用リソース・プールに割り当てることにより、このデータ・セットを特定のサブプールに割り当てることができます。このデータ・セットは、データ・セットの制御インターバル・サイズに基づいて、割り当てられた共用リソース・プールの中の特定のサブプールに送られます。

1 つの VSAM ローカル共用リソース・プール内に、VSAM KSDS 索引とデータ・コンポーネント用の個別のサブプールを生成することができます。索引とデータ・コンポーネントは、バッファを共用したり、同一サブプール内のバッファに対して競合することがないため、これらのサブプールを使用すると効果的です。

複数の VSAM ローカル共用リソース・プールを使用すると、ハイパースペース・バッファリングを使用する利点が広がります。ハイパースペース・バッファリングを使用し、仮想記憶域の他に拡張記憶域を使用して、割り振る 4 KB および 4 KB

の倍数のバッファの数を増やします。複数ローカル共有リソース・プールおよびハイパースペース・バッファリングを使用すると、特定の参照パターンがあるデータ・セット（例えば、1 次索引データ・セット）を、ハイパースペースで強化されたサブプールに分離することができます。これによって、データベースの処理に必要な VSAM 読み取り入出力アクティビティーが減ります。

High Availability Large Database (HALDB) を使用している場合、HALDB 区画データ・セットを特定の OSAM または VSAM サブプールに送ることもできます。

関連概念:

- ➡ OSAM 順次バッファリング (データベース管理)
- ➡ ハイパースペース・バッファリング (データベース管理)
- ➡ 高速機能バッファのパフォーマンスに関する考慮事項 (データベース管理)

関連資料:

- ➡ IMS PROCLIB データ・セットの DFSVSMxx メンバー (システム定義)

IMS が制御する入出力とページングとのトレードオフ

IMS によって制御される入出力とページングのいずれかを選択する場合には、いくつかのトレードオフを検討してください。

一般的なバッファ・プールに関する考慮事項

IMS 内の小さいチューニング済みプールは、ページングを最小限に抑える傾向があります。さらに、ページングは IMS システムによって制御されるわけではないため、ストレージが制限されている環境ではそれが著しくエスカレートします。

一方、IMS バッファ・プールの入出力は制御されるため、見積もりを出し、これらの見積もりと比較して行うことができます。通常、プロセッサでは、ページ不在とページ入出力操作を使用すると、BSAM または IMS OSAM の入出力を使用するよりもコストがかかります。

したがって、この点で IMS を最適化するには、プール・サイズの調整が必要になりますが、同時にページングとプールのスキャンのパス長さという点でのコストについて留意しておいてください。

IMS バッファ・プールのページ固定

IMS バッファ・プールの初期最適化については、526 ページの『IMS バッファ・プールのチューニング』で説明しています。これらの指針をシステムに適用して、入出力操作全体（プール入出力およびページング入出力）の最小値を達成します。つまり、1 つのプール内での場合と同様に、すべてのプールでの入出力とストレージとのトレードオフについて考慮する必要があります。例えば、大きな PSB プールの最大サイズを半分にすると、入出力がわずかに増えますが、その余分なストレージをメッセージ・キューまたは形式バッファ・プールへ渡すと、入出力をさらに節約できる場合があります。

プールをチューニングする方法は、実記憶域の環境によって異なります。通常、最大プール・サイズを増やすと、ページングも増えます。つまり、プール・サイズを減らすとこれらの領域のページングも減ります。

IMS プールがページングの対象になっており、IMS がプロセッサの使用を共用している場合、メッセージとデータベースのアクセスに必要な有効なバッファのページ固定を検討できます。これは、トランザクション率が非常に低く、プールがページ固定されていないと IMS がページアウトされる非専用 IMS システムで便利な方法です。頻繁にロードされる IMS システムでは、データ通信プールをページ固定することで応答時間が改善されます。これは、データ通信プールに最長期間を超えてもデータが保持されているためです。

ページ固定できる主要 IMS プールは次のとおりです。

- メッセージ・キュー・プール (QBUF)
- DMB プール (DLDP)
- PSB プール (DLMP)
- メッセージ形式バッファ・プール (MFBP)
- データベース・サブプール


DL/I アドレス・スペース・オプションを使用している場合は、z/OS 共通域の DLMP と DL/I アドレス・スペースの DPSB という 2 つの PSB プールが存在します。


IMS.PROCLIB の DFSFIXxx メンバーに組み込まれている制御ステートメントにリソースを指定して、IMS に、データベース・サブプールを除くすべてのこのような領域をページ固定するよう指示します。これらの制御ステートメントで使用されているプールに対応する名前は、上記の一覧では、括弧に囲まれて示されています。

データベース・サブプールのページ固定を行うには、IMS.PROCLIB の DFSVSMxx メンバーの制御ステートメントを使用します。IOBF ステートメントによって、OSAM サブプールが制御されます。VSAM プールのページ固定は、OPTIONS 制御ステートメントの VSAMFIX キーワードで指定します。

通常、プールの入出力がパフォーマンスに及ぼす影響は、プールを介したページ不在よりも少なくなります。ストレージが制約となっている場合は、ページングの対象となる大きなプールを使用するよりも、サイズの小さい固定プールを使用してください。ただし、ページ固定によってパフォーマンスが改善されることはなく、システムが IMS 専用になっている場合は、ページ固定によって全体のページング率が高くなることがあります。これは、ページ固定によって、システムの残りの部分が使用できるページ・プールが小さくなってしまうためです。ストレージの必要量を減らすために何らかの処置が取られない限り、高いページング率が発生することになります。

関連資料:

 IMS PROCLIB データ・セットの DFSFIXnn メンバー (システム定義)

 IMS PROCLIB データ・セットの DFSVSMxx メンバー (システム定義)

IMS ログ・バッファ

システム・ログ入出力を最小限に抑えるために IMS オンライン・ログ・データ・セット (OLDS) のブロック・サイズとバッファの数を選択する場合、一般的なバッファ・プールに関する考慮事項と同様の概念が適用されます。

ログ・バッファは、バッファ・スペースを獲得するために IMS がログ書き込みするのを待たなくても済むだけの数があれば十分です。先行書き込みデータ・セット (WADS) により、いっぱいになったログ・バッファだけが OLDS に書き込まれるようになります。ログ先行書き込み機能は、ログ・バッファの切り捨てを行いません。OLDS のブロック・サイズを定義する際には、4 K の倍数を使用すると、ログ・バッファが実記憶域では 2 ギガバイト境界の上に配置されるが、仮想記憶域では 2 ギガバイト境界の下に配置されることに留意してください。

一部のバッファ・スペースは動的バックアウトに使用されます。バックアウト・レコードが出力バッファに存在している場合は、これらのレコードはバッファから直接使用されます。そうでない場合は、バッファが出力バッファのセットから除去され、必要なログ・ブロックを OLDS から読み取るためにこのバッファが使用されます。バックアウトが終了すると、バッファが戻されて出力バッファとして使用されます。ログ・バッファは長期にわたってページ固定されます。

関連資料: 一般的なバッファ・プールに関する考慮事項について詳しくは、531 ページの『一般的なバッファ・プールに関する考慮事項』を参照してください。

z/OS でのライブラリー・ルックアサイド機能の使用

z/OS プログラム取り出し機能による IMS プログラムのロードの代替方法として、z/OS サービスのライブラリー・ルックアサイド機能 (LLA) の使用を指定できます。ロード済みプログラム・ライブラリーを動的に管理する方法として LLA を使用することをお勧めします。

この機能は、IMS アプリケーション・プログラムの場合、ロード・モジュールをデータ・スペースに保管するため、DASD の場合よりも効率的にモジュールを検索できます。LLA は、モジュールのアクセス頻度をモニターし、データ・スペースのロード状況を再調整して、モジュールが要求された場合にデータ・スペース内にモジュールが入っている確率を最適化します。

DB/DC 環境および DCCTL 環境でのアプリケーション・プログラム・プリロードのためのガイドライン

プリロード・オプションは通常、仮想記憶域をあまり使用しない、高アクティビティのプログラムに使用されます。IMS.PROCLIB データ・セットのメンバー DFSMPLxx にプリロードするモジュールの名前を指定します。

メッセージ処理領域の開始時に、DFSMPR プロシージャの EXEC ステートメントに接尾部 xx をパラメーターとして指定します。すると、プリロードされたアプリケーション・プログラムには、FETCH プログラムを使用せずに直接ブランチするようになります。プリロード・プログラムのスケジュールから最初の DL/I 呼び出し経過時間は、FETCH プログラム使用時の経過時間よりも短くなければなりません。ただし、ページ不在を順次実行することにより、アプリケーション・プログラムの経過時間が長くなる場合があります。

制約事項: DLIBATCH プロシージャーまたは DBBBATCH プロシージャーを使用して実行されるバッチ領域には、アプリケーション・プログラムをプリロードしないでください。このような環境でのアプリケーション・プログラムのプリロードには利点がありません。

最も効果的なプログラム・プリロードの実装の指針を次に示します。

- よく使用される PL/I、VS Pascal、COBOL サブルーチンとアプリケーション・プログラム・サブルーチンを、各従属領域にプリロードする必要があります。これらのサブルーチンは再入可能なため、ページング可能リンク・パック域 (PLPA) に入れておき、実記憶域には 1 つのコピーだけが常駐するようにしてください。サブルーチンの使用状況を領域プリロード・リストと突き合わせて、適切なモジュールがプリロードされていることを確認してください。
- アプリケーション・プログラムが領域トランザクション・ボリュームの高い割合を占める原因となっている場合に、このアプリケーション・プログラムをプリロードすることが効果的です。プリロードが最も効果的なのは、プリロードしたプログラムのトランザクション着信率が、プログラムの作業セットを実記憶域で維持するのに適切な値の場合です。システムが実記憶域の競合によって制約されている場合は、プリロードによってページング率が上がることもあるため、サブルーチンと高ボリュームのアプリケーション・プログラムだけをプリロードしてください。

これらのサブルーチンやアプリケーション・プログラムのプリロードを決定したら、さらに高ボリューム・トランザクションのクラス・スケジュールについて検討してください。トランザクション着信率によっては、1 つまたは 2 つの領域へのプリロードとトランザクションのクラス・スケジュールが効果的な場合があります。

システム・パフォーマンスを改善するためにプリロード・オプションを使用するかどうかは、実記憶域の可用性と対象となるトランザクション着信率に大幅に依存しています。

- アプリケーション・プログラム・オーバーレイは、場合によってはプリロードの有効な代替手段となります。汎用アプリケーション・プログラムが、入力トランザクション・データを識別し、トランザクション・サブコードに対してアプリケーション・プログラム・コードの一部だけしか実行しないようにコーディングされている場合は、プログラム・オーバーレイ入出力がプログラム全体のロードまたはページングよりも効率的なことがあります。主記憶域に作業セットを維持するのに十分なトランザクション着信率の場合は、プリロードの方が効果的な代替方法になります。

DFSMPLEXx PROCLIB メンバーに指定された疑似入力待ち (PWFI) プログラムおよび入力待ち (WFI) プログラムは、UPDATE PGM START(REFRESH) コマンドを使用して動的にリフレッシュすることはできません。

パス長さの最小化

トランザクションを処理するために実行されるマシン命令 (システム・サービス機能、IMS サービス機能、およびアプリケーション・プログラム自体を含む) の合計数は、スループットに直接関連します。実行された命令の累計を、パス長さ といいます。パフォーマンス用に選択されたオプションはパス長さに影響します。

520 ページの『パフォーマンスに対する IMS オプションの選択』で説明した処置はすべて、パス長さの最小化の原因となっています。DL/I 呼び出しイメージ・キャプチャー機能や /TRACE コマンドで呼び出される他のトレース機能などのトレース機能を通常は使用しないようにしてください。これは、IMS.PROCLIB の DFSVSMxx メンバーの OPTION ステートメントにパラメーターとして指定します。

事前に計画した 10 分から 20 分のインターバル以外の時点では、IMS モニター (DFSMNTR0) を実行しないでください。

実記憶域が制約されているシステムでは、パス長さを減らすより効果的な方法として、ページングを最小限に抑える方法があります。最小プールは、読み取り前にページングする必要のあるディレクトリーやバッファのコストのかかるスキャンを除去することにより、ページングを最小化する際に役立ちます。仮想記憶域の必要量が削減されると、次のようになります。

- 最小 PSB プールを使用すると、バッファ検索処理が最小限に抑えられます。
- チューニングされたデータベース・プールを使用すると、バッファ検索処理が最小限に抑えられます。大きなデータベース・プールを使用しても、パス長さの点でコストがかかり、入出力は削減されません。
- チューニングされたメッセージ・キュー・プールを使用すると、バッファ検索処理が最小限に抑えられます。大きなプールを使用すると、IMS メッセージ・キュー入出力は減りますが、この場合キューの 1 回の操作当たりのプロセッサ・サイクルが高くなります。
- メッセージ・キュー・プールと同じことがメッセージ形式バッファ・プールにも当てはまります。

DB/DC 環境および DCCTL 環境での通信ネットワークに関する考慮事項

応答時間に影響するパフォーマンス要因は、NCP およびネットワークに関する考慮事項、またはホスト・トランザクション処理に関する考慮事項に関連している場合があります。

- NCP とネットワークに関する考慮事項

ネットワーク内で、遅延の発生する可能性のあるポイントについて検討して、ボトルネックの発生を防いでください。遅延は次のような状況で発生します。

- クラスタまたは回線のキュー内
- 伝送を待機している時とポーリングされるのを待機している時
- 回線エラー発生時
- NCP 内
- VTAM 内
- IMS がマクロを受信するのを待機している時
- Within IMS 内
- 出力の伝送を待機している時

この他に発生する可能性のある問題としては、VTAM が大量の出力伝送を容易に処理できなくなることがあります。TERMINAL マクロ・ステートメントの OUTBUF キーワードの値と ETO ログオン記述子を確認してください。

IMS に入力スレッドが残っていないために入力が遅れることがあります。

- ホスト・トランザクション処理に関する考慮事項

通信ネットワークに対する処理を行う VTAM 入出力バッファのサイズの変更と補助記憶装置の使用とでは、どちらを選択してもそれぞれに長所と短所があることに注意してください。IMS 通信トラフィックは、TSO または他のサブシステムとこれらのリソースを共用することがあります。

VTAM によって許可されたパス・オプションを使用して、要求パラメーター・リスト (RPL) と RPL で指定された入出力領域の妥当性チェックをバイパスし、SEND コマンドと RECEIVE コマンドの処理量を削減します。これを実施するには、制御領域の EXEC ステートメントに VAUT=1 をパラメーターとして指定します。

IMS トランザクションは、可能であれば応答モードとして指定してください。このように指定することで、伝送回数と回線反転の回数が減り、回線の使用率が下がります。

IMS システム・データ・セットの配置に関するガイドライン

作業負荷にあわせて IMS システム・データ・セットを調整するだけでなく、データ・セットの使用に対する過度の競合が発生しないようにする必要があります。使用頻度の高いデータ・セットを正しく配置することで、ボトルネックの発生を防ぐことができます。

IMS オンライン・データ・セットを配置する際に考慮する必要がある指針を次に示します。

- 使用頻度の低い DASD ボリュームを、IMS.PGMLIB、オンライン・ログ・データ・セット、アクティブ IMS.FORMATA/B ライブラリーおよび IMS.ACBLIBA/B ライブラリー、メッセージ・キューなどの使用頻度の高いライブラリーやデータ・セットから分離します。
- 使用頻度の高い複数のデータ・セットが同一の装置上にある場合は、これらのデータ・セットをなるべく接する形で配置してください。保管データのフラグメント化を防ぐため、スペース割り振りでは、連続ストレージとシリンダー単位での割り振りが必要となります。
- 専用としてマウントされた IMS に対してボリュームを使用可能にし、オペレーティング・システムの一部データ・セットがこれらのボリュームに割り振られないようにします。
- オンライン IMS システム・データ・セットを共用 DASD ボリュームに入れないうでください。装置と制御装置の両方に対する競合を防ぎます。可能であれば、基本 OLDS と 2 次 OLDS は個別の制御装置とチャンネルに配置してください。
- プログラム取り出し入出力を最適化するために、PGMLIB を使用頻度を降順に配置します。フル・トラック・ブロック化を使用します。これにより、入出力開始 (SIO) 操作とシーク・タイムが最小限に抑えられます。

- 固定ヘッド装置を使用する場合は、先行書き込みデータ・セット (WADS) とデータベース索引データ・セットを固定ヘッド部分に配置します。
- プログラム・ライブラリーの競合が問題となっている場合は、コピーを複数作成します。各コピーはメッセージ領域の 1 つのサブセット用です。実動プログラム・ライブラリーとテスト・プログラム・ライブラリーを混合しないでください。
- 更新対象の IMS 共用データ・セットの場合は、同一のシステムまたは複数のシステムからの並列入出力を許可できる DASD 装置上にデータ・セットを配置してください。

入出力サブシステムの構成

RMF チャンネル・アクティビティ報告書と直接アクセス装置アクティビティ報告書を使用して、入出力サブシステムでのキューイングの遅延を監視します。物理チャンネルと論理チャンネルで負荷を平衡化することで、このような遅延を最小化できるように入出力サブシステムを構成してください。

DB/DC 環境および DCCTL 環境でのアプリケーション最適化

ACBLIB データ・セットへの読み取り入出力の量を減らすためには、DFSDFxxx PROCLIB メンバーの DATABASE セクションに ACBIN64=ggg を指定することにより、ACB メンバーを 64 ビット・ストレージにロードすることを考慮してください。アプリケーションがスケジュールに入れられると、ACB メンバーは、ACBLIB データ・セットからでなく、64 ビット・ストレージから取り出されます。

可能であれば、メッセージ処理プログラムでオーバーレイが使用されないようにしてください。ただし、アプリケーション・プログラムがトランザクション処理の一部で大きなコード・ブロックを排他的に使用する場合を除きます。

DL/I でない機能 (STIMER および TTIMER、内容監視など) を実行するアプリケーション・プログラムは使用しないでください。通常、これらのアプリケーション・プログラムは、慎重に制御しない限りパフォーマンスが非常に低くなります。

実行時間の長い冗長なアプリケーション・プログラムをオンラインで開始する必要がある場合は、このアプリケーションを、ユーザーに応答し、プログラム間メッセージをバッチ処理部分に送信する対話式処理部分と、優先順位の低い領域でも実行できるバッチ処理部分に分割します。

パフォーマンスがクリティカルなアプリケーションは、再設計を検討してください。再設計を検討しない場合は、全体的に作成し直す前に、トランザクションをそのまま処理できるようにシステムをチューニングする計画を立ててください。プログラム・ロード時間と実行時間を改善するため、パフォーマンスがクリティカルなアプリケーションを高水準言語からアセンブラに変換することを考慮してください。

プログラムを異常終了させるのではなく、ロールバック (ROLB) 呼び出しを常に使用します。これにより、オペレーターが介在しなくてもメッセージ領域が使用可能な状態で維持され、他のトランザクションに対するメッセージ領域が使用可能にな

る際に発生する遅延が短くなります。(アプリケーション・プログラムを異常終了させると、スケジュールされるのを待機している他のトランザクションに対して領域が使用不能のままになります。)

DL/I に関する考慮事項

DL/I データベースを設計するときは、以下の事項を考慮してください。

- 可能であれば常に DL/I パス呼び出しを使用して、DL/I パス長さとプログラム実行時間を短縮してください。
- 修飾された DL/I 呼び出しを使用できる場合はこの呼び出しを使用して、所定のセグメントを取得するために必要な呼び出しの数を最小限に抑えてください。
- アプリケーション・プログラムが常に単一セグメント入力メッセージを検索する場合は、入出力 PCB に対して GN 呼び出しを発行しないようにしてください。
- アプリケーション・プログラムが戻って、完了時に入出力 PCB に別の GU を発行することを確認してください。これにより、同じ種類の別のトランザクションが入力キューにある場合、このトランザクションを処理できるようになります。さらに、終了とスケジュール変更の処理のオーバーヘッドも発生しなくなります。
- 出力を複数の TP-PCB に送信する場合は、CHNG 呼び出しを使用するのではなく、複数の代替 PCB を定義します。
- 可能であれば、データベース編成として HDAM (HALDB を使用している場合は PHDAM) を選択します。これは、HDAM (または PHDAM) が適切にチューニングされていれば、通常、パス長さは短くなり、DL/I 呼び出し当たりの入出力が減るためです。

トランザクション処理の特性と、アプリケーション・プログラムとデータベース間の対話について評価する場合は、「IMS V15 データベース管理」と「IMS V15 アプリケーション・プログラミング」に記載されているパフォーマンスに関する設計の考慮事項に留意してください。

共用キュー環境でのパフォーマンスの計画

共用キュー環境では、各トランザクションのパフォーマンスは、非共用キュー環境の場合よりも劣る可能性があります。ただし、並列性が強化され、作業負荷のバランスがよくなるので、全体的なシステムのパフォーマンスとスループットは向上するはずですが、入力トランザクションは発行元のローカル IMS システムによって処理されるので、それぞれの IMS システムの従属領域の数の制御により IMS のパフォーマンスに影響を与えることができます。

以下はシステムのパフォーマンスに影響を与えます。

- IMS 実行パラメーター: QBUF=、QBUFMAX=、LGMSGSZ= および SHMSGSZ=
- CQS ローカル構造定義パラメーター SYSCHKPT=
- CQS グローバル構造定義パラメーター: OBJAVGSZ=、OVFLWMAX=、および STRMIN=
- カップリング・ファシリティの共用キュー構造のサイズ

- カップリング・ファシリティの z/OS システム・ログ構造
- z/OS システムのログ構造をバックアップする z/OS システムのログ・データ・セットの数
- 構造チェックポイントの頻度

すべてのログ・レコードのサイズは 8 バイトずつ増加し、キュー・マネージャーと EMH ログ・レコードのサイズは 32 バイト増加します。

SPA プールはもはや使用されず、これにより、各チェックポイントでログに記録されるデータの量は減り、使用可能なストレージの量は増加します。SPA は、現在は、その入力メッセージと出力メッセージで保持されます。

高速機能システムの場合、高速機能入力編集ルーター (DBFHAGU0) の使用を注意深く検討する必要があります。「ローカルのみ」として定義されているメッセージは、その他のすべてのメッセージ（「ローカル第一」および「グローバルのみ」）よりも優先順位が高いからです。

パフォーマンスの問題の識別と訂正

一般的な方法としては、一連のタスクを実行することにより、発生している問題の原因を判別します。

問題の原因を判別するには、以下のタスクを実行します。

1. 問題を以下のいずれかのカテゴリに切り分ける。
 - ページング率が高いことが原因で、一部またはすべての作業負荷の処理速度が遅くなっている。
 - 全体の作業負荷の一部のサブセットによるプロセッサの使用率が高いため、別の（サブセット）でディスパッチの問題が発生した。
 - DASD サブシステムで入出力の競合が発生したため、作業負荷の一部のサブセットの処理速度が遅くなっている。
 - 通信サブシステムに関連するリソースの使用状況、またはこのようなりソースに対して発生した競合が原因で、入力メッセージまたは出力メッセージの伝送速度が遅くなっている。
 - IMS 内で、物理リソース（プロセッサ、入出力装置、記憶装置など）の使用状況または論理リソース（プール、領域、制御ブロック、ラッチなど）に対する競合が、次のいずれかの一部またはすべてのトランザクションのパフォーマンスに悪影響を及ぼしている。
 - 入力メッセージまたは出力メッセージの処理。入力キューイングとメッセージ形式サービスも含まれます。
 - プログラムのスケジューリングと終了
 - プログラムのロードと初期設定
 - プログラムの実行
2. 問題の領域を切り分けた後、必要であればさらに詳しく調べて、以下を判別する。
 - 問題の詳細な特性
 - トランザクションまたは他のカテゴリによる主要な違反

- 問題を緩和する可能性の最も高いチューニング作業
3. 適切なチューニング操作を実行して、問題の再発を防止する。

ページング率の検討

オペレーティング・システムは、制御領域または従属領域が占有するアドレス・スペースに実記憶域を割り振ります。IMS は、実記憶域に対する要求が多過ぎるために、十分なマシン・サイクルを取得しないことがあります。ページング率と、ページ不在によるプロセスの遅延頻度を監視してください。RMF II の出力を使用して、ピーク IMS ロードでのページング率を検討することができます。

IMS オンライン・システムでページ不在が発生すると、ページング・アクティビティの所要時間の分だけトランザクションのシステム内時間 (time-in-system) が増加します。ページ不在が必ず発生する場合は、ページ不在を従属領域内のアプリケーション・プログラムに制限することが最良の解決策です。制御領域でページ不在が発生すると、制御領域サービスを待機している従属領域が保留状態になります。IMS にはいくつかのページ固定オプションがあります。これらについては、531 ページの『IMS バッファ・プールのページ固定』で説明します。ただし、実記憶域が制限されている場合、通常ページ固定によってページングが増加するため、このような場合にページ固定を使用することはお勧めしません。基本となるチューニング方法は、仮想記憶域と実記憶域の必要量を減らすことです。IMS の一部を固定すると、固定されていない領域のページングにまで悪影響を及ぼします。全体的なページング率も下がる可能性がある専用 IMS 環境では、この方法は特に重要です。

ページング率の動的モニター、日次モニター、または詳細モニターを行うことができます。ページング率のモニターについて詳しくは、476 ページの『モニター・アクティビティおよび技法の決定』を参照してください。

ストレージのオーバー・コミットメントが増加した原因を識別したら、次のいずれかまたはすべての作業を行う必要があります。

- IMS 制御の入出力に対する IMS ページング入出力のトレードオフを割り出す。
- オーバー・コミットメントを削減 (同時ユーザーを削減) するか、または他のユーザーと相対的に IMS 優先順位を上げて IMS ページング入出力を減らす。
- 入出力競合の削減、ページ・データ・セットの分割、またはページ・データ・セットをより高速な装置に格納することによりページ入出力速度を改善する。
- 非専用 IMS 環境では、IMS の特定部分を固定し、競合する他の作業を犠牲にして IMS ページングを削減する。

プロセッサ・リソースに関する問題の検出

多数のジョブやサブシステムを並行して実行している環境では、IMS がマシン・サイクルに対して競合します。IMS バッチ実行や TSO などのようなジョブやサブシステムのディスパッチング優先順位がオンライン IMS システムよりも高い場合は、これらのジョブやサブシステムの作業負荷は、IMS のパフォーマンスに著しく影響します。

注: ご使用のリソースに対して、動的モニター、日次モニター、または詳細モニターを実行してください。このモニターの実行方法について詳しくは、476 ページの『モニター・アクティビティーおよび技法の決定』を参照してください。

入出力リソース競合を除去するためのチューニング

パフォーマンスの問題の原因が入出力の競合であることがあります。入出力装置、DASD 記憶装置アクセス、チャンネル使用状況、制御装置の競合についてのチューニング・アクティビティーは、システム関連装置だけで行うわけではありません。IMS の場合、入出力アクティビティーを削減すると競合が減るため、入出力アクティビティーの量が重要な要因になります。

537 ページの『入出力サブシステムの構成』と、548 ページの『プログラム実行時間』に記載されているデータベース入出力の分析に関する考慮事項を参照してください。

入出力サブシステムの動的モニター

修正作業では通常装置やデータ・セットの再構成が必要となるため、通常、入出力サブシステムは動的にモニターされません。このような再構成は、計画と実行の段階で多少時間がかかります。ただし、RMF を実行している場合は、RMF チャンネル・アクティビティー報告書と直接アクセス装置アクティビティー報告書を使用して、入出力時間、チャンネル使用状況、装置の使用状況に関する各自の目標値をチェックすることができます。

直接アクションは実行できない可能性があります。ただし、データ・セットを移動したり、装置の再構成を行って、次に IMS を再始動する前に問題を修正することは可能です。

日次モニター

入出力構成とデータ・セットの配置を理解して、RMF 直接アクセス装置アクティビティー報告書とチャンネル・アクティビティー報告書を詳しく調べると、536 ページの『IMS システム・データ・セットの配置に関するガイドライン』および『入出力サブシステムの動的モニター』で述べた方針に沿ったチューニング・アクションのヒントを得られる場合があります。

詳細モニター

追加データが必要な場合は、GTF を実行して、ポリューム・マップ、シーク・ヒストグラム、システム/ジョブ要約 (EXCP-SIO-IO データ) 報告書を取得する計画を立ててください。これらの報告書は、入出力アクティビティーを詳細に分析する場合に使用できます。

DB/DC 環境および DCCTL 環境での通信サブシステムの競合

伝送装置や回線の種類が多いため、通信サブシステムで発生する問題は、それぞれ別個に調べる必要があります。

535 ページの『DB/DC 環境および DCCTL 環境での通信ネットワークに関する考慮事項』で説明したように、遅延の大部分の原因は、ネットワーク・ハードウェア

アとネットワーク・ソフトウェアにある可能性があります。ネットワークで発生した可能性のある問題を調査する場合には、様々なツールが必要となります。

動的モニター

通信サブシステムのパフォーマンスを動的にモニターすることはできません。

日次モニター

応答時間の明細データが、IMS 出力キュー時間が長く、場合によってその長さが異なるということを示す場合、ボトルネックは、IMS 出力メッセージ処理または回線に存在する可能性があります。適切な準備期間をおかずに TP ネットワークを再構成することは通常はまずないため、IMS モニター・トレース機能と IMS 回線トレース機能を使用してこれらの IMS 回線について調査する計画を立ててください。

詳細モニター

VTAMPARS は、GTF SMS レコードの分析と VTAM バッファ・サイズのチューニングに使用できます。DFSILTA0 と IMSPA 伝送ログ・データを、回線または PTERM 別にソートすると、出力キューの回線関連の問題を 1 回線または回線グループ単位に分離できます。

IMS モニターは、IMS 大容量入出力プール (HIOP) のチューニングを必要とする回線関連の IWAIT を示します。HIOP は動的に割り振られますが、バッファ・サイズの定義と、拡張する上限の両方を調整する必要があります。通信での競合に關係のある IMS モニターの出力は次のとおりです。

- 通信要約報告書には、VTAM ノードを使用したトランザクションの平均経過時間が出力されます。この時間は、メッセージ形式バッファ・プール内でのアクティビティーの影響を受けます。

この報告書には、平均 NON-IWAIT 時間も出力されます。この時間には、『DB/DC 環境および DCCTL 環境の保管域セット』で説明されているように、装置のボトルネックや、入力スレッドを取得する際に発生した問題が反映されている場合があります。スレッドが使用不能だった時間のカウンタは、「REPORTS」というヘッディングの付いた別の報告書に出力されます。'Total Times ECBS waited for SAPS = nnn' という行に合計時間が出力されます。モニター・インターバルに、ネットワークの初期設定にかかった時間が含まれていると、この数値は高くなります。

- 回線機能報告書には、465 ページの『第 24 章 IMS モニター・データの収集および解釈』で説明したように、各アクティブ・ノード別にデータ伝送アクティビティーが出力されます。TRANSMISSION BLKSIZE の値は、IMS によって OUTBUF バッファの VTAM に渡された文字数を示します。また、これは VTAM バッファの過負荷を示すことがあります。TURNAROUND インターバルは、トランザクション入力を分散させることができる非アクティブ回線を識別する場合に使用できます。
- 通信 IWAIT 報告書には、各ノードの IWAIT に関する詳細情報が出力されません。

VTAM アクティビティーをトレースするには、GTF (バッファ・トレース機能も含む) を使用して、VTAM バッファにデータが入力された時と VTAM バッファ

ーからこのデータが出た時の VTAM バッファの内容を調べます。また、ストレージ管理トレース機能を使用して、VTAM バッファの全体的な使用状況を反映することができます。

最後に、ネットワーク問題のチューニングでは、次の 1 つ以上の作業を行う必要があります。

- 伝送されるデータの量を減らす。(520 ページの『パフォーマンスに対する IMS オプションの選択』を参照してください。)
- VTAM または NCP のオプションとパラメーターを調整する。
- ネットワークを再構成して、マルチドロップ回線でのボトルネックを除去する。
- 装置エラーが原因で発生したハードウェアの遅延を分離して訂正する。

DB/DC 環境および DCCTL 環境での IMS メッセージ処理

このトピックでは、IMS システムでのメッセージ処理とメッセージ・キューに関連して起こりうるパフォーマンスの問題を示します。

動的モニター

MFS 統計および キュー・プール統計は、このタイプの問題を示します。プール・サイズを動的に増やすことはできません。プール・サイズは、次の IMS 始動時に増やしてください。通常、このようなパフォーマンスの問題は、一度チューニングした後は定期的なモニターだけが必要です。

日次モニター

IMS 出力メッセージのプロセスが問題になっている場合は、IMS 入力メッセージのプロセスも同じ遅延の影響を受けます。IMS PA リソース使用報告書と DC キュー・マネージャー・トレース報告書を使用して、IMS メッセージ・キューの長い出力キュー時間または可変長の出力キュー時間と、MFS (メッセージ形式バッファ・プール) 統計を調べます。

さらに、DB/DC 環境で出力キュー時間が長い場合は、IMS PA DC キュー・マネージャー・トレース報告書で出力メッセージの長さを調べて、IMS モニターを実行し通信 IWAIT 報告書を調べます。

詳細モニター

詳細モニターを実行する場合は、次の機能を使用してください。

メッセージ・キューの処理

DC キュー・マネージャー・トレース報告書を使用して IMS キュー・マネージャーを詳細に調べる場合に、IMS PA を使用できます。DFSUTR20 と IMSASAP は、通信関連の IWAIT の発生頻度と期間の統計を出力します。

IMS 制御領域の GTFPARS ジョブ要約報告書と詳細トレース報告書を使用して、MFS とメッセージ・キュー入出力のパフォーマンスを調べることができます。

メッセージ・キュー・データ・セットのチューニング

IMS.QBLKS、IMS.SHMSG、IMS.LGMSG の 3 つのデータ・セットは、メッセージ・キューのディレクトリー、短メッセージのセグメント、長メッセ

ージのセグメントを含んでいます。稼働率が高く、端末の入出力と従来型 SPA プールに制約のあるオンライン・システムでは、これらのデータ・セットが最もアクティブでクリティカルなデータ・セットになります。この 3 つのデータ・セットを異なる装置に格納して、シーク時間を最小限に短縮します。短メッセージと長メッセージで入出力アクティビティーが平衡かどうかを確認してください。発生する可能性のある装置アクセス時間の差について考慮しなければならないことがあります。

この時間差について考慮するには、IMS モニターの通信 IWAIT 報告書を調べます。項目は行番号を付けて編成されており、これらの項目によって、高い頻度と長い入出力経過時間が分かります。報告書のそれぞれの行には IWAIT を発生させた DD 名が示されているので、IMS.QBLKS、IMS.LGMSG、および IMS.SHMSG に属する行を識別することができます。領域 IWAIT 報告書でも同様にメッセージ・キューの入出力が示されており、領域の IWAIT 時間の合計が出力されます。

長メッセージ・キューと短メッセージ・キューはそれぞれ 10 まで定義できます。

関連資料: このオプションについて詳しくは、「IMS V15 システム定義」を参照してください。

MFS 使用状況

IMS 制御領域についての GTFPARS ジョブ要約報告書で、FORMATA/B に対する EXCP-SIO-IO 時間測定情報に関するデータを提供します。IMS モニターでは、形式プール・スペースの IWAIT、ディレクトリーの IWAIT、またはブロック・ロードの IWAIT を通信 IWAIT 報告書に出力します。

メッセージ形式ライブラリーの最適化

アクティブ IMS.FORMAT A/B データ・セットは、オンライン実行時に使用される MFS 形式ブロックすべてのソースであり、通常は使用中のデータ・セットです。

IMS モニター通信 IWAIT 報告書を使用して、このデータ・セットについて頻度と長い入出力経過時間を調べることができます。報告書で、I/O=DIR=fn または I/O=BLK=fn (fn は形式ブロック名を表します) で示される行には、形式バッファのサイズを増やした結果発生するディレクトリー検索とブロック取り出しの入出力の IWAIT 時間が出力されます。このプール・サイズは、システム定義時に BUFPOOLS マクロの FORMAT キーワードで指定されます。EXEC ステートメントの FBP パラメーターを使用して、オンライン実行用にこの値をオーバーライドできます。

ディレクトリーへの入出力を減らすには、ブロックのトラック・アドレスの索引 (または頻繁に参照されるブロックのみのトラック・アドレスの索引) を作成して、この索引を MFS プールに保持します。索引が作成されるのは、ユーティリティー制御ステートメントに INDEX を指定して MFSRVC プロシージャを実行するときです。また、形式ブロックまたはメッセージ・ブロックの名前を索引に組み込むこともできます。

DASD に格納されている MFS 制御ブロックのサイズが、トラック・サイズまたは z/OS ブロック・サイズよりも大きいと、アクティブ・ライブラリー・データ・セットの複数読み込みが発生します。

DB/DC 環境および DCCTL 環境での入力キューイングとスケジューリング/終了

パフォーマンス目標を達成できない場合に発生するパフォーマンスの問題があります。この問題は、作業負荷の増加あるいはアプリケーション・プログラムのスケジューリングの遅延によって入力キューが作成されるときに発生します。

動的モニター

このトピックでは、IMS 入力キューイング、IMS 領域の占有状況、IMS スケジューリング、同期点処理に関する動的モニター情報を取得する方法について説明します。

IMS 入力キューイング

入力キュー時間は使用できません。ただし、IMS /DISPLAY コマンドを QUEUE パラメーターを指定して実行すると、入力キューのサイズをモニターできます。ACTIVE パラメーターを使用して、パフォーマンスを追跡し、スケジューリングと領域占有におけるボトルネックを検出できます。

IMS 領域占有

動的な報告書では使用できません。IMS コマンド /DISPLAY ACTIVE を使用して領域の使用状況をモニターします。

IMS のスケジューリング

518 ページの『チューニングのための z/OS および IMS パラメーターの初期設定』で説明したように、PSB プールと DMB プールはすでにチューニング済みである必要があります。すべての領域がすでに使用されている場合、必要な追加領域を開始してください。一部の領域がアイドル状態になっている場合は、クラスを修正するか、または 1 つのトランザクションで領域が過負荷の場合には PARLIM を修正します。

関連資料: 詳しくは、523 ページの『IMS リソース (バッファー・プールを除く) に対する競合の回避』を参照してください。

同期点処理

同期点の待機が原因で発生した問題は、動的に分離することはできません。

日次モニター

このトピックでは、トランザクション応答時間、応答時間の明細、領域占有状況、プログラムのスケジュール、プログラム終了時の同期点処理に関する日次モニター情報の取得方法について説明します。

トランザクション応答時間

選択されたトランザクションの合計 (内部) 応答時間を抽出してください。これはシステム・パフォーマンスを示す値として使用できます。

関連資料: 応答時間のソースについて詳しくは、466 ページの『パフォーマンス目標の確立』を参照してください。

応答時間の明細

同様のトランザクションについて、IMS PA または DFSILTA0 から応答時間の明細 (入力キュー、切り替えキュー、プロセス、出力キュー) を抽出してください。

領域占有

IMS PA リソース可用性報告書を使用して、DB/DC 環境内での従属領域の相対的な使用状況を示す情報を取得できます。また、IMS モニター領域要約報告書に示されている領域占有率の要約を直接使用することもできます。

プログラム・スケジューリング

応答時間の分析データが、入力キュー時間が長いことや、状況によって異なることを示す場合は、領域占有状況の数値を調べてください。すべてのメッセージ領域が高い占有率になっている場合は、別のメッセージ領域が必要になります。あるいは、プログラム・ロード時間とプログラム実行時間を短縮して、占有率を下げることがもできます。IMS メッセージ領域の一部またはすべてが使用中でない場合、トランザクション別とクラス別の IMS PA トランザクション経過報告書を分析すると、たいていは、あるトランザクションまたはクラスが他のトランザクションやクラスよりも決定的な影響を与えていることが分かります。このような場合、クラスの指定と領域への割り振りについて検討してください。PROCLIM と PARLIM についても検討してください。

関連資料: スケジューリング・オプションについては、520 ページの『パフォーマンスに対する IMS オプションの選択』を参照してください。

入力待ちとして実行されるプログラムは、実行時に領域に 100 % の時間入っていても、占有率 100 % として表示されません。占有率が 0 になっている場合は、オペレーターのプロシージャーを検討し直して、キューの表示に基づいて、メッセージ領域の数を管理する指示を出します。

IMS PA トランザクション経過報告書のグラフィック要約は、測定期間全体にわたって入力時間で入力キューイング時間を分析する際に役立ちます。この要約は、長い入力キュー時間の原因が、トランザクション・ボリュームの一時的なピークか、またはもっと断続的な問題なのかを判別する際に役立ちます。DFSILTA0 も同じ目的で使用できますが、この出力はグラフィックではなく数値になります。

プログラム終了時の同期点処理

IMS PA を使用して応答時間明細データを作成している場合、プログラム間通信キュー時間が長い場合は、同期点処理の完了を待機していることが原因で遅延が発生しています。MODE=MULT を使用するとこのような状況が発生します。このタイプのトランザクションのうち、プログラム・スケジューリングごとにキューから解放される複数のトランザクションについて、IMS PA CPU 使用報告書を調べてください。通常、このような遅延を発生させないようにするには MODE=SNGL を使用します。

詳細モニター

スケジュールと同期点処理に関する詳細モニター情報を調べてください。

スケジューリング

IMS モニターは、PSB と DMB の入出力カウントと経過時間に関する報告

書を生成します。IMS 制御領域に関する GTFPARS ジョブ要約報告書と詳細トレース報告書は、これらの入出力イベントを詳細に調べる場合にも使用できます。ただし、スケジューリングの問題は領域の可用性とメッセージ・クラス PROCLIM、領域クラス割り当てに関連していることがよくあります。これらについては、523 ページの『IMS リソース (バッファ・プールを除く) に対する競合の回避』に説明があります。

同期点処理

同期点の待機が原因で発生した遅延の場合は、通常詳細な調査を必要としません。

プログラムのロードと初期設定

このトピックでは、プログラムのロードと初期設定の動的モニター、日次モニター、詳細モニターについて説明します。

動的モニターと日次モニター

プログラムのロードまたは初期化に使用するリソースは、通常の日次モニター・データから動的に分離したり、訂正することはできません。通常、パフォーマンスのこのような点は、一度チューニングした後は、定期的なモニター以外の処置は必要ありません。

IMS.PGMLIB データ・セットは、従属領域の JCL に指定され、アプリケーション・プログラムを含んでいます。IMS モニター領域要約報告書とプログラム要約報告書に出力されるスケジューリングから最初の DL/I 呼び出しまでの経過時間には、プログラムを領域に入れるための入出力時間も含まれます。この入出力時間には、JOB LIB (最初に検索される) と STEPLIB のシステム検索の時間も含まれます。JOB LIB と STEPLIB がプログラム・ライブラリー用に使用されていない場合は、IMS.PGMLIB が SYS1.PARMLIB の LNKLSTnn メンバー内の先頭になければなりません。プログラム・ライブラリーは常に全トラック・サイズでブロック化されなければなりません。ライブラリーは、従属領域で必要になるメンバーだけを含むように作成できるため、従属領域の STEPLIB に入れることをお勧めします。

実記憶域の可用性による制約を受けない環境では、プログラムのロードとプリロードのトレードオフについて考慮してください。IMS モニター領域要約報告書に出力されている、スケジューリングから最初の DL/I 呼び出しまでの経過時間を使用して、プリロードによって節約される可能性のある時間を評価してください。

z/OS ライブラリー・ルックアサイド機能 (LLA) は、ロード済みプログラム・ライブラリーを動的に管理するメソッドです。

関連資料: LLA について詳しくは、533 ページの『z/OS でのライブラリー・ルックアサイド機能の使用』を参照してください。

詳細モニター

IMS メッセージ領域に関する GTFPARS ジョブ要約報告書と詳細トレース報告書には、ユーザーのアプリケーション・プログラムを検出してロードし、高水準言語の初期設定モジュールを実行するために発行される SVC と入出力操作のカウンタとシーケンスに関する包括的なデータが出力されます。

プログラム・ライブラリー・オプションと、初期設定に関する考慮事項については、520 ページの『パフォーマンスに対する IMS オプションの選択』を参照してください。

プログラム実行時間

IMS 内部応答時間の動的モニター、プログラム実行の日次モニター、および詳細モニターを行うことにより、プログラム実行時間に関する情報を取得できます。

IMS 内部応答時間の動的モニター

IMS 内部応答時間は、動的な報告書には出力されません。ただし、多くのインストール・システムでは GHU、ISRT、REPL、DLET シーケンスなどのように、一定の数 (通常は少数) の DL/I 呼び出しを常に実行する特別なトランザクションを作成しています。IMS MTO は、このようなトランザクションを使用することにより、IMS システムの応答を動的にモニターできます。これは正確な測定結果にはなりませんが、ユーザーがその時点でどのような処理を行っているかを示します。

プログラム実行の日次モニター

IMS PA または DFSILTA0 の使用可能な応答時間明細データによって実行時間が長いことが示される場合、その原因は次のいずれかに関連している可能性があります。

- プログラムのロード時間

日次モニターで実行時間の増加が示された場合は、入出力サブシステムに問題が生じている可能性があります。

- プログラムの初期設定とアプリケーション・コード

チューニングを行っても、実行時間は大幅に変化しません。

- DL/I 呼び出しと入出力

IMS PA 内部リソース使用報告書に、データベース入出力の統計が出力されます。この統計値が高い場合は、データベース・バッファのサイズと数が出力されている GTFPARS データ・セット概要報告書を検討してください。揮発性データベースを定期的に再編成して、OSAM または VSAM のオーバーフロー入出力を減らしてください。この点に関する詳細な調査を行うには、IMS モニターを実行して呼び出し要約報告書および領域 IWAIT 報告書を取得してください。

- その他の IMS リソースの遅延

IMS PA 内の IMS 内部リソース使用報告書は、プログラム分離アクティビティで競合が多数発生しているかどうかを示します。競合が多数発生している場合は、IMS PI トレースを実行して分析する計画を立てる必要があります。

- 従属領域の変更

従属領域の経過時間が長い理由と、各トランザクションの経過時間が長いことを示すアプリケーション・プログラムの経過時間が長い理由を検討してください。検討する際に、IMS モニター・プログラム要約報告書を使用できます。ディスパッチング優先順位が適切であるものと想定して、スケジューリングから最初の DL/I 呼び出しまでの経過時間が長いものを検索できます。プログラムのロード

とページングに関する要因を考慮しない場合は、インストール・システムにおいて従属領域に変更が行われたかどうかを確認する必要があります。

z/OS データ・セットへのアクセスまたは他の z/OS サービスの使用をサポートするために変更が行われた場合は、これらの変更が適切であるかどうかとその効果を評価する必要があります。このような変更を行う理由の 1 つに、オーバーレイ・プロセスまたは初期設定が多過ぎることがあります。

詳細モニター

IMS メッセージ領域の GTFPARS ジョブ要約報告書と詳細トレース報告書は、アプリケーション・プログラムのシステム関連アクティビティを詳細に調べる場合の最適な手段です。IMSASAP プログラム・トレース報告書には、DB/DC 環境でのデータベース・アクセスに関連するアプリケーション・アクティビティの詳細な分析結果が出力されます。IMS PA データベース・トレース報告書には、DB/DC 環境でのデータベース更新アクティビティの分析が出力されます。DFSPIRPO は、長い遅延の原因として PI ロックアウトが考えられる場合に、この遅延について調べる時に使用します。

プログラム実行時間の主要部分として、入出力操作をとまなうデータベース検索にかかった時間があります。各データベース・セグメント・タイプへの各種 DL/I 呼び出しの頻度と、IMS IWAITS の発生頻度が呼び出し要約報告書に出力されます。プログラム実行時間が長くなった理由としてデータベース関連の遅延が考えられる場合は、その遅延の原因となった特定の呼び出しとデータベースの組み合わせを識別する際に、この報告書が役立ちます。入出力待機時間は、IMS モニター領域 IWAIT 報告書に出力されます。

IWAIT 時間が 100 ミリ秒よりも長くなる場合、その原因は通常、共用 DASD からの干渉またはリソースのオーバー・コミットメントです。VSAM を使用しているデータベースでは、短い IWAIT がよく発生します。これは、VSAM が管理する入出力バッファ制御により、物理入出力を行わずに要求を満たすことができるためです。レコードのチェーニングを過度に行ったためにトラックのオーバーフローが発生することも原因として考えられます。ストレージ・デバイスのデータ転送速度による効果は限られています。

データベースの設計上の考慮事項に加えて、入出力 IWAIT 時間を短縮する主要な方法としては、次の 2 つがあります。

- バッファを追加して、入出力の頻度を削減する。この作業では、不適切な IMS バッファ割り振りを識別する必要があります。
- 入出力リソースの使用状況を調べて、データ・セットの分散に問題があるかどうか判別する。これは、GTF トレースと RMF トレースから取得した詳細な出力を使用する、時間がかかる分析です。

DL/I 構造とデータベース編成の変更は、さらに大規模な作業となります。

関連資料: DL/I 構造の変更またはデータベース編成の変更についての詳細は、「IMS V15 データベース管理」を参照してください。

第 27 章 オンライン変更の実行

IMS は、オンライン IMS システム内のリソースを変更するためのオプションを数種類提供しています。

オンライン IMS システム内のリソースを変更するためのオプションには、次のものが含まれます。

- 動的リソース定義が使用可能になっている場合は、IMS タイプ 2 コマンド CREATE、UPDATE、および DELETE。IMPORT コマンドを使用して、IMSRSC リポジトリまたはリソース定義データ・セット (RDDS) のいずれかに格納された定義からリソースをアクティブにすることもできます。
- ACB の IMS 管理が使用可能な場合は、IMPORT コマンド。このコマンドにより、DDL またはいずれかの IMS カタログ・データ取り込みユーティリティーを使用して事前に IMS カタログに格納された新規定義または変更済み定義から、DBD および PSB リソースをアクティブにします。
- データベース変更機能
- オンライン変更機能

IMPORT コマンド (DBD および PSB の変更用)

ACB の IMS 管理が使用可能な場合は、IMPORT DEFN SOURCE(CATALOG) コマンドを発行することで、IMS カタログ内のデータベース定義 (DBD) およびプログラム・ビュー定義 (PSB) をオンライン IMS システムでアクティブにすることができます。指定した MANAGEDACBS 制御ステートメントと共に DDL またはいずれかのデータ取り込みユーティリティーを使用して、DBD および PSB を IMS カタログに実行依頼する必要があります。

動的リソース定義 (DRD)

DRD を使用すると、オンライン変更プロセスを使用せずに、特定のランタイム・リソース定義をオンラインで変更できます。(例えば IMS.ACBLIB 内のリソースなど) 他のリソースをオンラインで変更するには、やはりオンライン変更プロセスを使用する必要があります。DRD を使用すると、オンライン変更プロセスを使用する代わりに、タイプ 2 コマンドを発行してランタイム・リソース定義を動的に作成、更新、および削除するか、以前に出力作成出口ルーチンと呼ばれていた拡張宛先作成出口ルーチン (DFSINSX0) を使用して、トランザクション (また、必要であれば、トランザクションに関連したプログラム) を作成できます。

以下のようにする必要があります。

- コマンドは、変更を適用したいシステムへ送られ、そのシステム上で実行される
- DRD コマンドで動的に加えた変更は、コールド・スタート後にリカバリーされる
- リソースおよび記述子の定義は RDDS へエクスポートされる

DRD を使用すると、システム定義またはオンライン変更プロセスを使用せずに、以下の IMS リソース定義およびそれらの記述子定義を動的に作成、更新、照会、および削除できます。

- アプリケーション・プログラム
- データベース
- 高速機能宛先コード
- トランザクション

DRD はタイプ 2 コマンドを使用するので、DRD を使用できるのは IMS が以下のように構成されている場合に限られます。

- Operations Manager (OM) と Structured Call Interface (SCI) を含んでいるが、Resource Manager (RM) を含んでいない最小の共通サービス層を使用して構成されている単一 IMS の IMSplex。この構成は、拡張コマンド環境とも呼ばれます。
- 完全な CSL (OM、SCI、および RM) を使用して構成されている複数 IMS の IMSplex。

データベース変更機能

DEDB データベースのオンライン変更機能は、HALDB データベースのオンライン変更機能とは異なります。

DEDB 変更ユーティリティを使用して、オンライン DEDB のデータベースおよび領域を変更します。

INITIATE OLREORG コマンドの ALTER オプションを使用して、オンライン HALDB データベースおよび区画を変更します。

変更機能を使用してデータベースに対して実行できる変更は、特定のタイプのもののみです。実行できる変更のタイプは、変更しているデータベースが DEDB か HALDB かによって異なります。

オンライン変更機能

オンライン変更機能には以下の 2 つのバリエーションがあります。

ローカル・オンライン変更

ある IMS 用の IMS リソースに対してのオンライン変更を可能にする。


グローバル・オンライン変更

IMSplex 内の IMS システムすべてに渡る IMS リソースに対してのオンライン変更を調整する。

このセクションの以降のサブトピックでは、DRD を使用不可にした場合の IMS のオンライン変更機能について説明します。

関連概念:

 ACB の IMS 管理 (システム定義)

 DEDB 変更ユーティリティを使用したオンライン DEDB データベースの定義の変更 (データベース管理)

🔗 動的リソース定義の概要 (システム定義)

関連タスク:

🔗 オンライン・データベースの変更 (データベース管理)

🔗 オンライン HALDB データベースの定義の変更 (データベース管理)

関連資料:

🔗 INITIATE OLREORG コマンド (コマンド)

🔗 DEDB 変更ユーティリティ (DBFUDA00) (データベース・ユーティリティ)

管理対象 ACB 環境でのオンライン変更

ACB の IMS 管理が使用可能な場合、オンライン IMS システムでアクティブ ACB を変更するには、IMPORT DEFN SOURCE(CATALOG) コマンドを使用します。IMS によって管理されている ACB を活動化するために、オンライン変更機能を使用することはできません。

保留中の変更を含んでいる ACB は、ユーザーが IMPORT DEFN SOURCE(CATALOG) コマンドを発行するまで、IMS カタログのステージング・データ・セット内に存在する必要があります。

IMPORT DEFN SOURCE(CATALOG) コマンドが発行されると、IMS は以下のアクションを実行します。

1. IMS カタログのディレクトリー・データ・セット内の ACB の追加または置き換えを行います。
2. IMS カタログのステージング・データ・セットから ACB を削除します。
3. リソースの古いコピーをすべてメモリーから削除し、DMB と PSB をそれぞれのプールからキャストアウトします。
4. IMS カタログが複数の IMS システムによって共有されている場合は、IMSplex 全体で変更を調整します。

IMPORT DEFN SOURCE(CATALOG) コマンドを発行すると、IMS は、オンライン変更機能に似ている内部プロセスを使用して、オンライン IMS システムでリソースを活動化します。そのため、インポート処理中に QUERY MEMBER コマンドを発行すると、状況には、オンライン変更で使用されているのと同じ準備とコミットのフェーズが反映されます。エラーが発生した場合は、メッセージにもこれらのフェーズが反映されることがあります。

行おうとする変更のタイプ、および PSB、DBD、またはその両方を更新するかどうかに応じて、リソースを使用する前に、さらにいくつかのステップを実行することが必要になる場合があります。例えば、特定のタイプのデータベース変更のためにデータベースの再編成が必要になったり、データ・セットの割り振りが必要になったり、DBRC へのデータベースの登録が必要になったり、新規データベースを作成する場合は IMS タイプ 2 の CREATE DB コマンドの発行が必要になったりします。

関連概念:

☞ ACB の IMS 管理 (システム定義)

関連資料:

☞ IMPORT DEFN SOURCE(CATALOG) コマンド (コマンド)

☞ IMS カタログ・ユーティリティー (システム・ユーティリティー)

IMS が ACB を管理する場合のリソースの活動化の概要

IMS がアプリケーション制御ブロック (ACB) を管理する場合、オンライン IMS システム内の新規データベース (DBD) またはプログラム・ビュー (PSB) の活動化は、変更が IMS にサブミットされた時点で自動的に実行できるほか、変更がサブミットされて IMS カタログのステージング・データ・セットに保管された後、手動で行うこともできます。

データベース・リソースが活動化された と見なされるのは、そのリソースの ACB が IMS ディレクトリー・データ・セットにインポートされ、そのリソースがオンライン・システムによってメモリー内にロードされた場合、またはオンライン・システムがリソースをスケジュールした次回にロードされる場合です。ただし、活動化されたデータベース・リソースを使用するには、事前に、さらにいくつかのステップを実行することが必要になる場合もあります。例えば、新規データベースの場合は、データベース・データ・セットを割り振ったり、RECON データ・セット内にデータベースを定義したり、データベースを初期設定したり、IMS タイプ 2 コマンド CREATE DB または CREATE PGM コマンドを使用して、データベースまたはプログラムをオンライン・システムに対して指定したりすることが必要になる場合があります。変更されたデータベースでは、データベースの再編成が必要な場合もあります。

DDL CREATE ステートメントを使用して新規データベースを定義する場合は、DFSDFxxx PROCLIB メンバーの <CATALOG> セクションに AUTOCREATE=YES または AUTOIMPORT(CREATE) が指定されていれば、IMS は自動的にデータベースを活動化することができます。DDL を使用して DEDB データベースを作成する場合は、DFSDFxxx PROCLIB メンバーの <DDL> セクションで DEDB エリア・データ・セットを自動的に割り振るためのオプションを指定することができます。

IMS 生成ユーティリティーを使用してデータベース・リソースを定義および生成し、ACB Generation and Catalog Populate ユーティリティー (DFS3UACB) または IMS Catalog Populate ユーティリティー (DFS3PU00) のいずれかを使用してリソースを IMS にサブミットする場合は、いずれかのオンライン・システムが IMS カタログを使用していると、ほとんどのリソースを自動的に活動化することはできません。ご使用の z/OS システムが PDSE データ・セットの拡張共用をサポートしており、いずれかの Populate ユーティリティーの実行時に MANAGEDACBS=(UPDATE,SHARE) を指定した場合は、DOPT PSB を自動的に活動化することができます。

DDL を使用した場合も、いずれかの Populate ユーティリティーを使用した場合も、即時に活動化されなかったリソースは IMS カタログのステージング・データ・セット内に保管され、IMPORT DEFN SOURCE(CATALOG) コマンドが発行されることで活動化されるまで、保留中状態で待機します。

リソースを活動化する前に必ず、活動化のためのすべての前提条件ステップを完了しておくか、少なくとも計画しておいてください。






IMS が DDL ステートメントの受信時に自動的にリソースを活動化する場合も、後でユーザーが `IMPORT DEFN SOURCE(CATALOG)` コマンドを発行して活動化する場合も、IMS は同じ内部インポート 処理を使用して、IMS ディレクトリー・データ・セットおよび関係する IMS システムすべてのオンライン・メモリーに対してリソースを更新します。IMS が使用する内部インポート処理は、ACB、DBD、および PSB ライブラリーを使用して ACB を管理するオンライン IMS システムでリソースを活動化するために使用する必要がある、手動のオンライン変更処理と同様です。

この内部インポート処理には、いくつかの準備とコミットのフェーズが含まれており、これらはオンライン変更機能の準備とコミットのフェーズと同等です。

IMS インポート処理の状況を判別するには (特に、活動化中にエラーが発生したかどうか)、`QUERY MEMBER TYPE(IMS) SHOW(STATUS)` コマンドを発行します。

IMS カタログが複数の IMS システムによって共有されている場合は、IMS カタログが共有されている場合の活動化 (システム管理)を参照してください。

関連資料:

-  `QUERY MEMBER` コマンド (コマンド)
-  `IMPORT DEFN SOURCE(CATALOG)` コマンド (コマンド)
-  `DFSDFxxx` メンバーの `CATALOG` および `CATALOGxxxx` セクション (システム定義)
-  `DFSDFxxx` メンバーの `DDL` セクション (システム定義)
-  `IMS Catalog Populate` ユーティリティー (`DFS3PU00`) (システム・ユーティリティー)

IMS が ACB を管理する場合の変更されたリソースの活動化

IMS が ACB を管理する場合に、変更されたアプリケーション・プログラム (PSB) およびデータベース (DBD) をオンライン IMS システムに導入するには、更新された定義を IMS にサブミットして、影響を受けるリソースを停止し、`IMPORT DEFN SOURCE(CATALOG)` コマンドを発行します。

DDL または IMS 生成ユーティリティー・マクロ命令のいずれかを使用して、変更されたリソースを定義できます。

以下の手順は、ステージング・データ・セット内に保留中の変更として保管されている変更を活動化するためのものです。この手順は、単一のオンライン IMS システム内の変更を活動化するか、複数の IMS システムがリソースを共有している場合は、IMSplex 内の変更全体を活動化します。

IMSplex 内にある IMS システムのサブセット内のアプリケーション・プログラムの変更のみを活動化するには、561 ページの『IMSplex 内のシステムのサブセットでの PSB の活動化』を参照してください。

動的オプション (DOPT) PSB のみを活動化するには、IMS 管理の ACB 環境での DOPT PSB の管理 (システム・ユーティリティ) を参照してください。

IMS が ACB を管理する場合に、オンライン IMS システム内の変更された PSB、DBD、またはその両方を活動化するには、以下のようにします。

- DDL または IMS 生成ユーティリティ・マクロを使用して、定義を変更します。
 - DDL を使用した定義の変更について詳しくは、DDL を使用したデータベースおよびプログラム・ビューの定義 (データベース管理) を参照してください。
 - IMS 生成ユーティリティを使用した定義の変更について詳しくは、IMS 生成ユーティリティを使用した IMS データベースの設計 (データベース管理) を参照してください。
- 変更した定義を IMS にサブミットします。
 - DDL を使用する場合は、IMS Enterprise Suite Explorer for Development または SQL バッチ・ユーティリティ (データベース・ユーティリティ) のいずれかを使用して、DDL ステートメントを IMS にサブミットします。
 - 生成ユーティリティ・マクロ・ステートメントを使用する場合は、DBD および PSB 生成ユーティリティを実行した後、ACB を生成し、MANAGEDACBS=STAGE 制御ステートメントを指定して ACB Generation and Catalog Populate ユーティリティ (DFS3UACB) (システム・ユーティリティ) を使用することによって、同じジョブ・ステップ内で IMS に変更をサブミットします。

DDL を使用する場合、IMS は自動的に活動化されなかった変更済みの DBD 定義および PSB 定義を保留中の変更として IMS カタログのステージング・データ・セット内に保持します。

- 必要であれば、データベースを再編成します。
- 変更によって影響を受ける既存のデータベースおよびプログラム・リソースを、すべて停止します。
- IMPORT DEFN SOURCE(CATALOG) コマンド (コマンド) を発行します。

IMPORT コマンドで OPTION(DELPENDERR) を指定すると、インポート処理の準備フェーズでエラーが発生した場合に、IMS にステージング・データ・セットを自動的にクリーンアップさせることができます。エラーの後にステージング・データ・セットをクリーンアップしておく、IMPORT コマンドを次回に発行したときに、さらに問題が起きるのを回避するのに役立ちます。

コマンド・マスターの IMS システムで有効になっている ACBSHR= パラメーターの値が、IMSplex 内の他の IMS システムが IMPORT DEFN SOURCE(CATALOG) コマンドを処理するかどうかに影響を与える可能性があります。

- コマンド・マスターで ACBSHR=Y が指定されている場合、IMPORT DEFN SOURCE(CATALOG) コマンドは、ACBSHR=Y を指定する IMSplex 内の

すべての IMS システム、および ACBSHR=N を指定する IMS システム (このコマンドの発行時に ROUTE パラメーターで指定されている場合) によって処理されます。ACBSHR=N を指定する IMS システムの場合は、これらの IMS システムについて IMS カタログ・インスタンス内に保留中の ACB の変更が存在する場合のみ、IMPORT コマンドは正常に実行されます。

- コマンド・マスターで ACBSHR=N が指定されている場合、このコマンドはコマンド・マスターによって処理され、さらに、同じく ACBSHR=N を指定している (ROUTE パラメーターに指定された) IMS システムによって処理されます。このコマンドは、ACBSHR=Y を指定する IMS システムではリジェクトされます。

コマンド・マスターの役割を果たす IMS システムは、IMPORT DEFN コマンドが経路指定された IMS システムから IMS によって選択されます。

多数のリソースを活動化する場合、IMPORT DEFN SOURCE(CATALOG) コマンド (コマンド) コマンドは完了までにかなりの時間を要することがあります。IMPORT コマンドの処理状況を確認するには、QUERY MEMBER コマンド (コマンド) を発行します。

6. すべての IMS システムで、影響を受けたリソースを再始動します。

関連概念:

559 ページの『IMS カタログが共用されている場合の活動化』

関連タスク:

『IMS が ACB を管理する場合の新規リソースの活動化』

IMS が ACB を管理する場合の新規リソースの活動化

IMS が ACB を管理する場合に、新規のプログラム・ビュー (PSB) およびデータベース (DBD) をオンライン IMS システムに導入するには、更新された定義を IMS にサブミットして、リソースが自動的に活動化されない場合には IMPORT DEFN SOURCE(CATALOG) コマンドを発行します。

DDL または IMS 生成ユーティリティー・マクロ命令のいずれかを使用して、変更されたリソースを定義できます。

DDL を使用する場合は、DFSDFxxx PROCLIB メンバーの CATALOG セクションで AUTOIMPORT(CREATE) パラメーターを指定することにより、新規の DBD および関連する PSB が自動的に活動化されるように IMS を構成できます。IMS は必要な制御ブロックを作成し、活動化します。必須のデータ・セットがまだ存在していない場合は、それらを作成する必要があります。

DDL を使用して高速機能 DEDB データベースを定義し、DFSDFxxx PROCLIB メンバーの CATALOG セクションで AUTOCREATE=YES を指定すると、IMS は DBD および PSB の制御ブロックの作成と活動化だけでなく、DEDB エリアのデータ・セットの作成も自動的に行います。これらのデータ・セットの属性は、DFSDFxxx PROCLIB メンバーの DDL セクションで定義できます。AUTOCREATE=YES を指定した場合は、AUTOIMPORT(CREATE) を指定する必要はありません。

以下の手順は、同じ IMS カタログを共用するすべての IMS システムの新規リソースを活動化します。

IMS が ACB を管理する場合に、オンライン IMS システム内の新規の PSB、DBD、またはその両方を活動化するには、以下のようになります。

- DDL または IMS 生成ユーティリティー・マクロを使用して、リソースを定義します。
 - DDL を使用したデータベースおよびプログラム・ビューの定義については、DDL を使用したデータベースおよびプログラム・ビューの定義 (データベース管理) を参照してください。
 - IMS 生成ユーティリティーを使用した DBD および PSB の定義については詳しくは、IMS 生成ユーティリティーを使用した IMS データベースの設計 (データベース管理) を参照してください。
- 定義を IMS にサブミットします。
 - DDL を使用する場合は、IMS Enterprise Suite Explorer for Development または SQL バッチ・ユーティリティー (データベース・ユーティリティー) のいずれかを使用して、DDL ステートメントを IMS にサブミットします。
 - 生成ユーティリティー・マクロ・ステートメントを使用する場合は、DBD および PSB 生成ユーティリティーを実行した後、ACB を生成し、MANAGEDACBS=STAGE 制御ステートメントを指定して ACB Generation and Catalog Populate ユーティリティー (DFS3UACB) (システム・ユーティリティー) を使用することによって、同じジョブ・ステップ内で IMS に変更をサブミットします。

DDL を使用して新規データベースを作成する場合、IMS システム内で有効になっている自動活動化オプションに応じて、新規 DBD を自動的に活動化することができます。それ以外の場合、IMS は新しい DBD と PSB の ACB を、保留中の変更として IMS カタログのステージング・データ・セット内に保持します。

- 新規データベースを作成する場合は必ず、以下のステップが完了しているか、計画されているか、IMS によって自動的に処理されることを確認してください。
 - データベース・データ・セットの作成。
 - 新規データベースにアクセスするための PSB の作成。
 - データベースおよびアプリケーション・プログラムに対するオンライン IMS システム内のランタイム属性制御ブロック (DDIR および PDIR) の作成。
- IMPORT DEFN SOURCE(CATALOG) コマンド (コマンド) を発行します。

IMPORT コマンドで OPTION(DELPENDERR) を指定すると、インポート処理の準備フェーズでエラーが発生した場合に、IMS にステージング・データ・セットを自動的にクリーンアップさせることができます。エラーの後にステージング・データ・セットをクリーンアップしておくこと、IMPORT コマンドを次回に発行したときに、さらに問題が起きるのを回避するのに役立ちます。

コマンド・マスターの IMS システムで有効になっている ACBSHR= パラメーターの値が、IMSplex 内の他の IMS システムが IMPORT DEFN SOURCE(CATALOG) コマンドを処理するかどうかに影響を与える可能性があります。

- コマンド・マスターで ACBSHR=Y が指定されている場合、IMPORT DEFN SOURCE(CATALOG) コマンドは、ACBSHR=Y を指定する IMSplex 内のすべての IMS システム、および ACBSHR=N を指定する IMS システム (このコマンドの発行時に ROUTE パラメーターで指定されている場合) によって処理されます。ACBSHR=N を指定する IMS システムの場合は、これらの IMS システムについて IMS カタログ・インスタンス内に保留中の ACB の変更が存在する場合のみ、IMPORT コマンドは正常に実行されます。
- コマンド・マスターで ACBSHR=N が指定されている場合、このコマンドはコマンド・マスターによって処理され、さらに、同じく ACBSHR=N を指定している (ROUTE パラメーターに指定された) IMS システムによって処理されます。このコマンドは、ACBSHR=Y を指定する IMS システムではリジェクトされます。

コマンド・マスターの役割を果たす IMS システムは、IMPORT DEFN コマンドが経路指定された IMS システムから IMS によって選択されます。

多数のリソースを活動化する場合、IMPORT DEFN SOURCE(CATALOG) コマンド (コマンド) コマンドは完了までかなりの時間を要することがあります。IMPORT コマンドの処理状況を確認するには、QUERY MEMBER コマンド (コマンド) を発行します。

5. すべての IMS システムで、影響を受けたリソースを再始動します。

関連概念:

『IMS カタログが共用されている場合の活動化』

関連タスク:

555 ページの『IMS が ACB を管理する場合の変更されたリソースの活動化』

IMS カタログが共用されている場合の活動化

ACB を管理する複数の IMS システムが IMS カタログを共用する場合、IMSplex 内の IMS システム間でのリソース活動化の調整の制御は制限されます。

IMS は、IMS によって管理される ACB を共用する IMS システム間全体で、リソースの変更を自動的に調整します。IMS は内部的に、グローバル・オンライン変更プロセスが ACBLIB データ・セット内の変更を活動化するために使用するプロセスと同様のグローバル調整プロセスを使用します。

IMS は、各システムでの ACBSHR= パラメーターの指定によって、どの IMS システムが IMS カタログ内の ACB を共用しているかを判別します。ACBSHR=Y を指定しているシステムのみが、リソース活動化のグローバル調整に含まれます。ACBSHR=N を指定している IMS システムは含まれません。ACBSHR= パラメーターは、DFSDFxxx メンバーの COMMON_SERVICE_LAYER セクション、または PROCLIB データ・セットの DFSCGxxx メンバーで指定されます。

管理対象 ACB 環境では、1 つの IMS システムが、リソース活動化のグローバル調整でマスターとして機能します。マスターとして機能するには、IMS システムで ACBSHR=Y が指定されている必要があります。マスターは、IMS カタログ・ディレクトリー・データ・セットを更新し、IMS カタログを共用する IMSplex 内の他の IMS システムに対して、リソース定義を IMS ディレクトリーからリフレッシュす

るように通知します。IMPORT DEFN SOURCE(CATALOG) によってリソース活動化がトリガーされた場合、マスターは、ACBSHR=Y が指定されているすべての IMS システムを (コマンドがそれらのシステムに明示的に経路指定されていない場合でも) リソース活動化のグローバル調整に含めます。

リソースがグローバルに活動化されている状態で、共用 IMS システムの 1 つがダウンした場合、IMS は、ダウンしたシステムを記録しておきます。リソースがまだ活動化中に IMS システムが稼働状態になった場合、再始動中の IMS システムはメッセージ DFS4373I を発行し、処理が完了するまで待ってから続行します。コマンドの処理が完了した後に IMS システムが稼働状態に戻った場合、IMS システムは、IMS システムがダウンしている間に IMS カタログが更新されたことを示すメッセージを発行します。IMS システムがウォーム・リスタートまたは緊急時再始動によって再始動されている場合、IMS システムは、IMS カタログから ACB をリフレッシュします。

管理対象 ACB 環境での変更された PSB の活動化のステージング

PSB のみを変更する場合、OPTION(UPDATEPSB) キーワードを指定して IMPORT DEFN SOURCE(CATALOG) コマンドを使用することで、変更した PSB の活動化を IMSplex 全体でステージングすることができます。PSB の活動化をステージングすることにより、変更された PSB の活動化とテストを 1 つの IMS システム上で実行する一方、他の IMS システム上で元の PSB を使用した処理を続行して、可用性を維持することができます。OPTION(UPDATEPSB) キーワードが指定されたときに IMS カタログのステージング・データ・セット内に DBD があると、IMPORT DEFN SOURCE(CATALOG) は拒否されます。

データ共用環境でのリソースの活動化中のエラー・シナリオ

リソースの活動化中にマスターが異常終了した場合、データ共用グループ内の他の IMS システムが、活動化処理のテークオーバーおよびクリーンアップをローカルで試行します。

IMS カタログを共用する他の IMS システムのうち 1 つのみが、新規のマスターとしてテークオーバーを行い、IMS カタログのディレクトリー・データ・セットに対する更新をバックアウトするかコミットすることができます。新規のマスター IMS も異常終了した場合や、正常に引き継ぎプロセスを完了できなかった場合は、IMPORT DEFN SOURCE(CATALOG) コマンドまたは DDL ステートメントを再サブミットして、ディレクトリー・データ・セットをクリーンアップする必要があります。

インポート処理中に IMS システムが異常終了した場合は、QUERY MEMBER TYPE(IMS) SHOW(STATUS) コマンドを発行して、インポート処理とテークオーバー試行の両方の状況を判別することができます。

関連タスク:

561 ページの『IMSplex 内のシステムのサブセットでの PSB の活動化』

関連資料:

 QUERY MEMBER コマンド (コマンド)

 IMPORT DEFN SOURCE(CATALOG) コマンド (コマンド)

➡ DFSDFxxx メンバーの CATALOG および CATALOGxxxx セクション (システム定義)

➡ DFSDFxxx メンバーの COMMON_SERVICE_LAYER セクション (システム定義)

IMSplex 内のシステムのサブセットでの PSB の活動化

IMSplex 環境で PSB のみを変更する場合は、IMPORT DEFN SOURCE(CATALOG) コマンドの UPDATEPSB オプションを使用して、IMS システムのサブセットで PSB を活動化することができます。後で、このコマンドの REFRESHPSB オプションを使用することにより、IMSplex 内の他の IMS システムで PSB を活動化します。

システムのサブセットのみで PSB を活動化すると、IMS システムのサブセット上で PSB 変更をテストしながら、IMSplex 内の他のシステムで元の PSB を使用して、プログラムの実行を続けることができます。

IMPORT コマンドを発行するときは、FOR(IMSID()) キーワードで、UPDATEPSB オプションまたは REFRESHPSB オプションを処理する IMS システムのサブセットを指定します。ただし、このコマンドを IMSplex 内の特定の IMS システムに最初に経路指定する場合は、コマンドを最初にサブミットするときに、TSO SPOC またはその他のコマンド・インターフェースでルーティング要件を指定する必要があります。

UPDATEPSB および REFRESHPSB のオプションでは、コマンドをそれぞれ別の方法で経路指定する必要があります。UPDATEPSB オプションが指定される場合は、FOR(IMSID()) キーワードで指定されている 1 つ以上の IMS システムにコマンドを経路指定する必要があります。REFRESHPSB オプションが指定される場合は、FOR(IMSID()) キーワードで指定されていない 1 つ以上の IMS システムにコマンドを経路指定する必要があります。

UPDATEPSB オプションは、IMS システムがオンライン・メモリーに保管する PSB に使用します。例えば、64 ビット・ストレージ・プールにキャッシュされた常駐および非常駐の PSB として定義されている PSB です。動的オプション (DOPT) PSB および IMS システムのオンライン 64 ビット・ストレージ・プールにキャッシュされていない非常駐 PSB の場合、UPDATEPSB オプションの効果は、OPTION(UPDATEPSB) を指定せずに IMPORT DEFN SOURCE(CATALOG) コマンドを発行する場合と変わりません。


UPDATEPSB オプションが指定されたときに FOR(IMSID()) キーワードで指定されていない IMS システムは、IMPORT コマンドを処理しません。そのため、変更された PSB 定義が IMS ディレクトリーにインポートされた後、メモリー内の PSB 定義を再ロードしないでください。ただし、コマンド処理の完了後に、コマンドに指定されていなかった IMS システムで他の理由により IMS ディレクトリーからメモリー内の PSB 定義を再ロードする必要がある場合は、変更された PSB 定義はその IMS システムでアクティブな PSB 定義になります。

通常、UPDATEPSB オプションから除外された IMS システムは、IMS システムに対して IMPORT DEFN SOURCE(CATALOG) OPTION(REFRESHPSB) コマンド

が発行されるか、IMS システムが再始動されるまで、元のメモリー内の PSB 定義を使用し続けます。UPDATEPSB オプションが指定されたときに再ロードされなかったメモリー内の PSB 定義は、IMPORT コマンド応答で最新表示の保留中 (REFP) 状況として示されます。

IMS システムで PSB 変更を活動化して、PSB の最新表示の保留中状況を解除するには、FOR(IMSID()) キーワードで指定された IMS システムの ID を指定して IMPORT DEFN SOURCE(CATALOG) OPTION(REFRESHPSB) コマンドを発行します。REFRESHPSB オプションにより、IMS システムで最新表示の保留中状況になっているすべての PSB 定義が再ロードされます。

関連概念:

 IMS カタログが共有されている場合の活動化 (システム管理)

関連資料:

 IMPORT DEFN SOURCE(CATALOG) コマンド (コマンド)

UPDATEPSB によるシステムの初期サブセットでの変更された PSB の活動化

変更された PSB を IMSplex で初めてオンライン IMS システムのサブセットのみに対して導入し、活動化するには、OPTION(UPDATEPSB) キーワードを使用します。

IMS カタログのステージング・データ・セットに、IMSplex 内で活動化されるのを待っているデータベース定義 (DBD) が含まれていないことを確認してください。ステージング・データ・セットに保留中の DBD 変更が含まれている場合、IMPORT DEFN コマンドは、UPDATEPSB オプションや REFRESHPSB オプションが指定されていると失敗します。

IMSplex 内の IMS システムのサブセットで PSB を活動化するには、以下のようにします。

1. 変更された PSB 定義を、IMS カタログのステージング・データ・セットにロードします。
2. PSB を活動化するターゲット IMS システムを避けて、ワークロードを経路指定します。
3. ターゲット IMS システム内の影響を受けるプログラム・リソースを停止します。
4. TSO SPOC または OM API では、以下のコマンドを、FOR(IMSID()) キーワードで指定されている 1 つ以上の IMS システムへ明示的に経路指定します。
IMPORT DEFN SOURCE(CATALOG) OPTION(UPDATEPSB)
FOR(IMSID())。

UPDATEPSB オプションのコマンド・マスターとして使用される IMS システムは、FOR(IMSID()) キーワードで指定された IMS システムの 1 つでなければなりません。指定されたシステムの 1 つへコマンドを経路指定すると、それらのシステムの 1 つが必ずコマンド・マスターとして機能します。


コマンドの処理時に、FOR (IMSID()) キーワードで指定された IMS システム内で、変更された PSB が活動化されます。FOR(IMSID()) キーワードで指定さ

れなかった IMS システムでは、元の PSB 定義が依然として使用されますが、PSB はリフレッシュ保留中 (REFP) の状況を示します。

5. ターゲット IMS システム内の影響を受けたリソースを再始動します。
6. ターゲット IMS システム内の新しいプログラム定義をテストします。
7. テストが完了したら、ターゲット IMS システム上で新規の PSB を使用してワークロードを再開します。

変更した PSB を IMSplex 内の他のシステムで活動化する準備ができたなら、『REFRESHPSB によるシステムの後続のサブセットでの変更された PSB の活動化』の手順に従ってください。

関連概念:

 IMS カタログが共有されている場合の活動化 (システム管理)

関連資料:

 IMPORT DEFN SOURCE(CATALOG) コマンド (コマンド)

REFRESHPSB によるシステムの後続のサブセットでの変更された PSB の活動化

変更された PSB が UPDATEPSB オプションによって IMSplex 内の IMS システムの初期サブセットで活動化された後、IMSplex 内の残りのシステムで変更された PSB を活動化するには、IMPORT DEFN SOURCE(CATALOG) コマンドの REFRESHPSB オプションを使用します。

REFRESHPSB オプションは、UPDATEPSB オプションの後にのみ使用してください。

IMS カタログのステージング・データ・セットに、IMSplex 内で活動化されるのを待っているデータベース定義 (DBD) が含まれていないことを確認してください。ステージング・データ・セットに保留中の DBD 変更が含まれている場合、IMPORT DEFN コマンドは、UPDATEPSB オプションや REFRESHPSB オプションが指定されていると失敗します。

REFRESHPSB オプションを指定すると、FOR(IMSID()) キーワードで指定されたすべての IMS システムは、オンライン・メモリー内でリフレッシュ保留中 (REFP) 状況にあるすべての PSB を再ロードします。IMS システムは、REFRESHPSB オプションを使用して IMPORT DEFN コマンドを処理した後、次回に PSB がスケジュールされたときに、PSB を再ロードします。


1. REFRESHPSB オプションのターゲット IMS システムを避けて、ワークロードを経路指定します。
2. ターゲット IMS システム上の影響を受けるリソースを停止します。
3. TSO SPOC または OM API では、特に以下のコマンドを、FOR(IMSID()) キーワードで指定されていない 1 つ以上の IMS システムに経路指定します。
IMPORT DEFN SOURCE(CATALOG) OPTION(REFRESHPSB)
FOR(IMSID())。

REFRESHPSB オプションのコマンド・マスターとして使用される IMS システムは、FOR(IMSID()) キーワードで指定されていない IMS システムの 1 つで

なければなりません。FOR(IMSID()) キーワードで指定されていない IMS システムへコマンドを経路指定すると、必ずターゲット・システム以外の IMS システムがコマンド・マスターとして機能します。

4. リフレッシュのターゲット IMS システム上にある、影響を受けたリソースを開始します。
5. リフレッシュのターゲット IMS システム上で、新規のプログラム定義をテストします。
6. テストが完了したら、リフレッシュのターゲット IMS システム上で新規の PSB を使用してワークロードを再開します。

関連概念:

 IMS カタログが共有されている場合の活動化 (システム管理)

関連資料:

 IMPORT DEFN SOURCE(CATALOG) コマンド (コマンド)

IMSplex 内のシステムのサブセットで変更された PSB を活動化する例

以下の一連の図は、変更された PSB を IMSplex 内のすべてのシステムにわたって徐々に活動化する例を示しています。

この IMSplex は、5 つの IMS システム、ディレクトリー・データ・セットとステージング・データ・セットを備えた 1 つの IMS カタログ、および IMS 共通サービス層 (CSL) の Operations Manager (OM) から構成されています。すべての IMS システムは、ACBSHR=Y の指定で示されているように、IMS カタログおよびディレクトリー・データ・セット内の ACB を共有します。

この例では、PSB1 が更新されます。当初のバージョンの PSB1 は、図で *ts01* によって表されるタイム・スタンプを備えています。更新されたバージョンの PSB1 は、それより後の *ts02* によって表されるタイム・スタンプを備えています。

一連の図のうち、最初の図では、タイム・スタンプが *ts01* である元のバージョンの PSB1 が、IMSplex 内の 3 つの IMS システム (IMS1、IMS3、および IMS5) で稼働しています。IMS2 はダウンしており、IMS4 は PSB1 を非常駐として定義しており、この時点ではバッファ内にも 64 ビット・ストレージ内にもコピーを持っていません。タイム・スタンプが *ts02* である新バージョンの PSB1 は、ステージング・データ・セット内および IMS カタログ内に置かれた後に活動化される準備ができています。

IMSplex

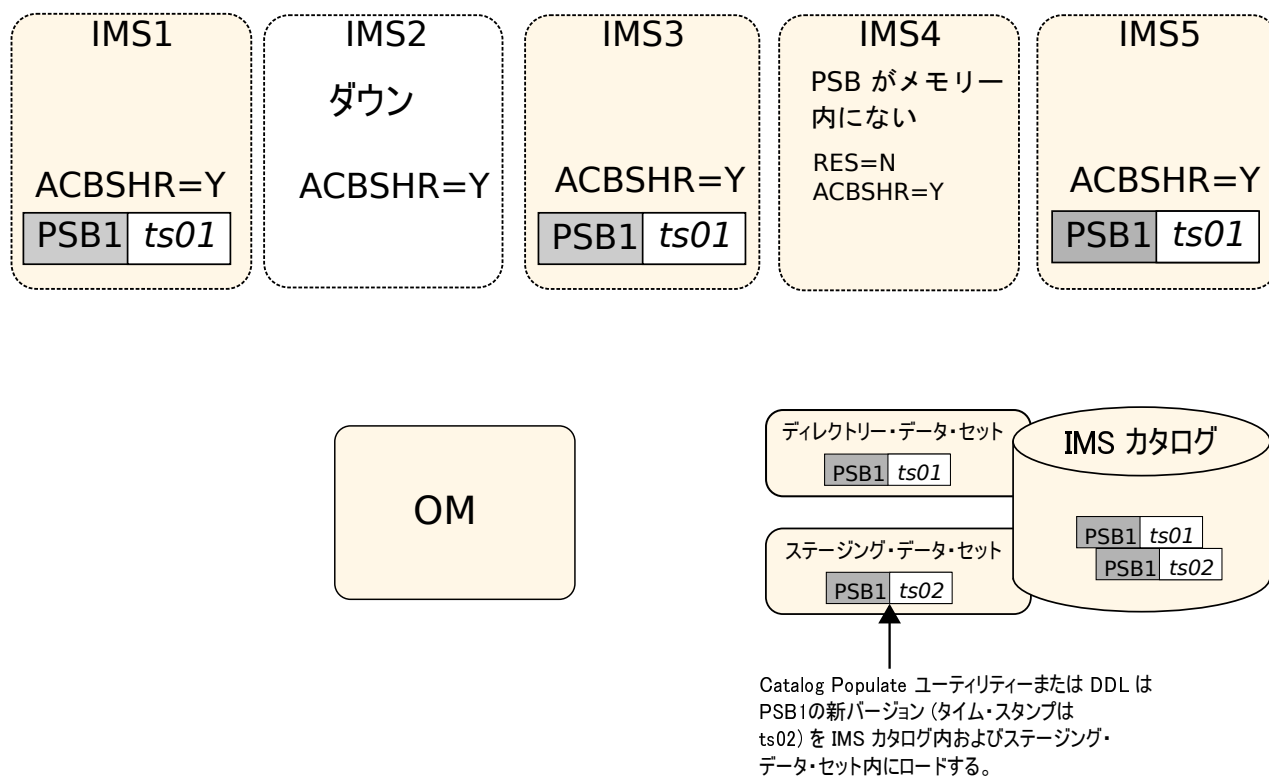


図 48. 新バージョンが活動化される前の IMSplex 内の PSB1 の初期状態

次の図では、タイム・スタンプが ts02 である新バージョンの PSB1 を IMS1 上でのみ活動化するために、コマンド `IMPORT DEFN SOURCE(CATALOG) OPTION(UPDATEPSB) FOR(IMSID(IMS1))` が発行されます。PSB1 を IMS1 上でのみ活動化するために、このコマンドは明示的に IMS1 へ経路指定されると同時に、FOR キーワードで指定されています。ただし、PSB1 は IMS4 上でも活動化されると見なされます。IMS4 は PSB1 を非常駐として定義しており、現時点で PSB1 のコピーをメモリー内に持っていないためです。IMS4 は、次回に PSB がスケジュールされたときに、IMS ディレクトリーから PSB1 を再ロードします。

IMS3 および IMS5 はコマンドの FOR キーワードに含まれていないため、PSB1 をリフレッシュせず、メモリー内にある元のバージョンを引き続き使用します。IMPORT DEFN コマンドの出力は、IMS3 および IMS5 の PSB1 についてリフレッシュ保留中であることを示します。

IMSplex

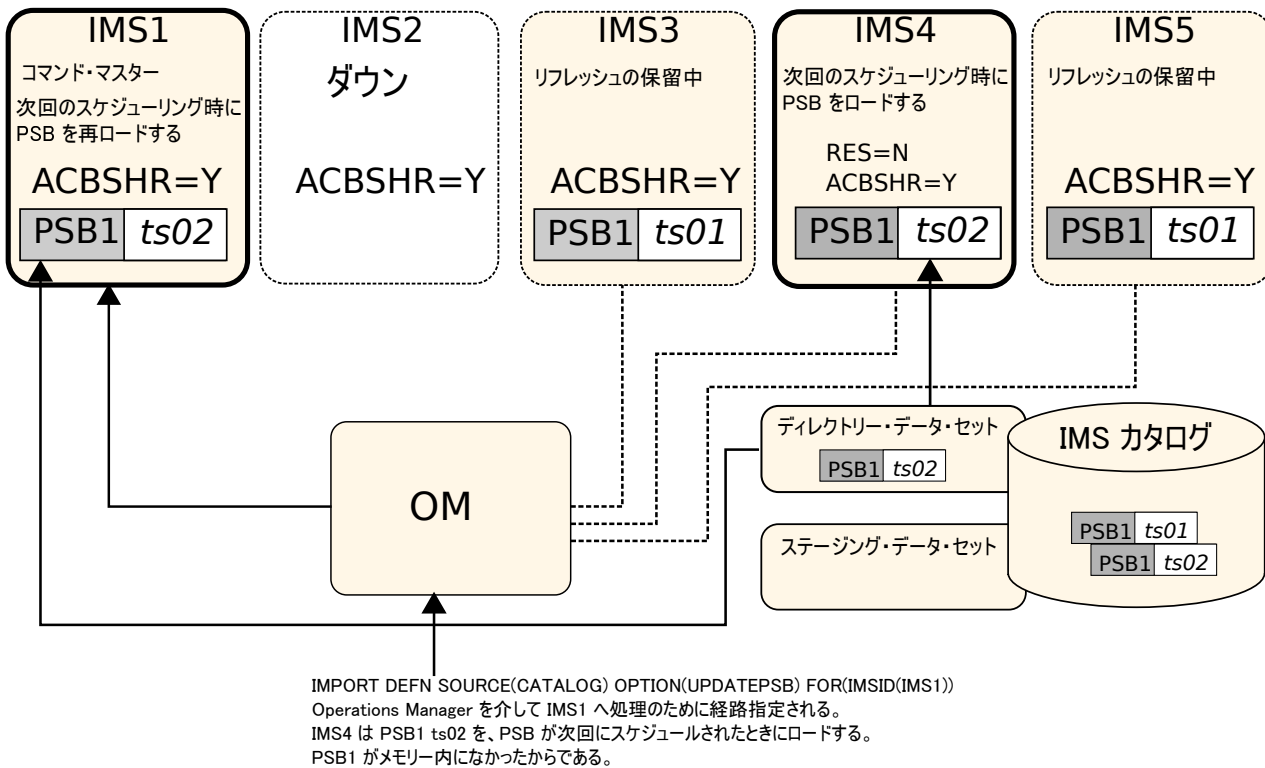
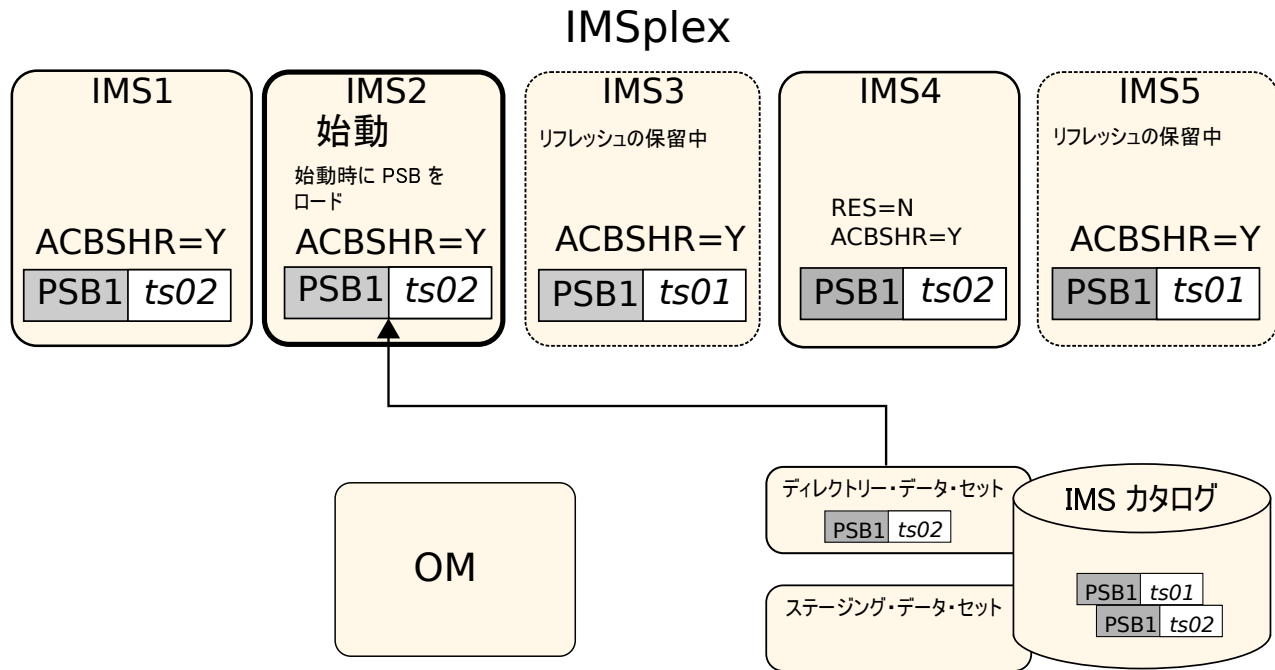


図 49. 新バージョンの PSB1 が IMS1 上で活動化される。IMS2 はスケジューリング時に新バージョンをロードする。

次の図では、IMPORT DEFN SOURCE(CATALOG) OPTION(UPDATEPSB) FOR(IMSID(IMS1)) コマンド時にダウンしていた IMS2 が再始動します。IMS2 はコマンドの影響を受けず、再始動時に IMS ディレクトリーから新バージョンの PSB1 をロードします。



IMS2はコマンドが発行された後に始動し、
ディレクトリー・データ・セットを読み取り、PSB1 ts02 をロードする

図 50. IMS2 が再始動時に PSB1 をロードする

次の図では、タイム・スタンプが *ts02* である新バージョンの PSB1 を IMS3 上でのみ活動化するために、コマンド `IMPORT DEFN SOURCE(CATALOG) OPTION(REFRESHPSB) FOR(IMSID(IMS3))` が発行されます。PSB1 を IMS3 上で活動化するために、FOR キーワードでは IMS3 が指定されますが、コマンドは明示的に IMS1 へ経路指定されます。REFRESHPSB オプションを指定する場合、FOR キーワードで指定される IMS システムをコマンド・マスターとして使用することはできません。

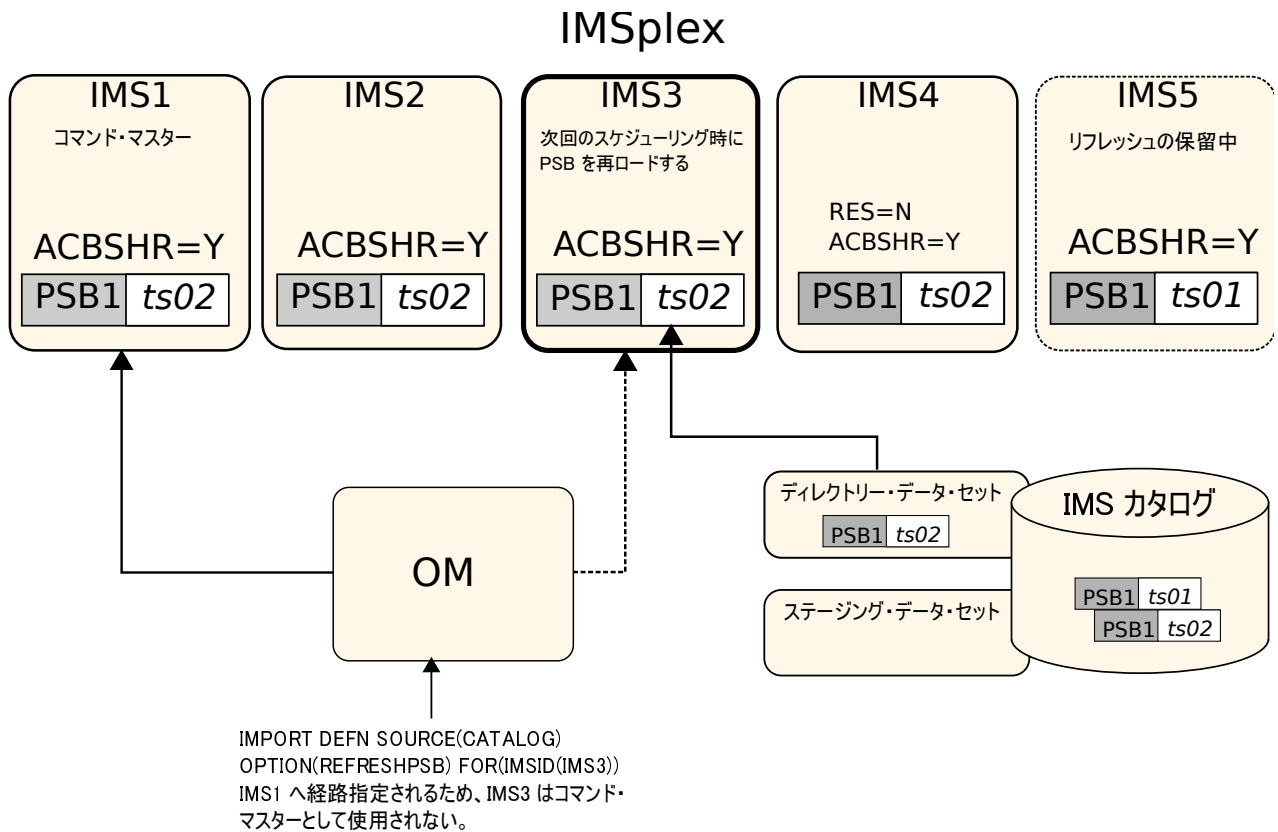
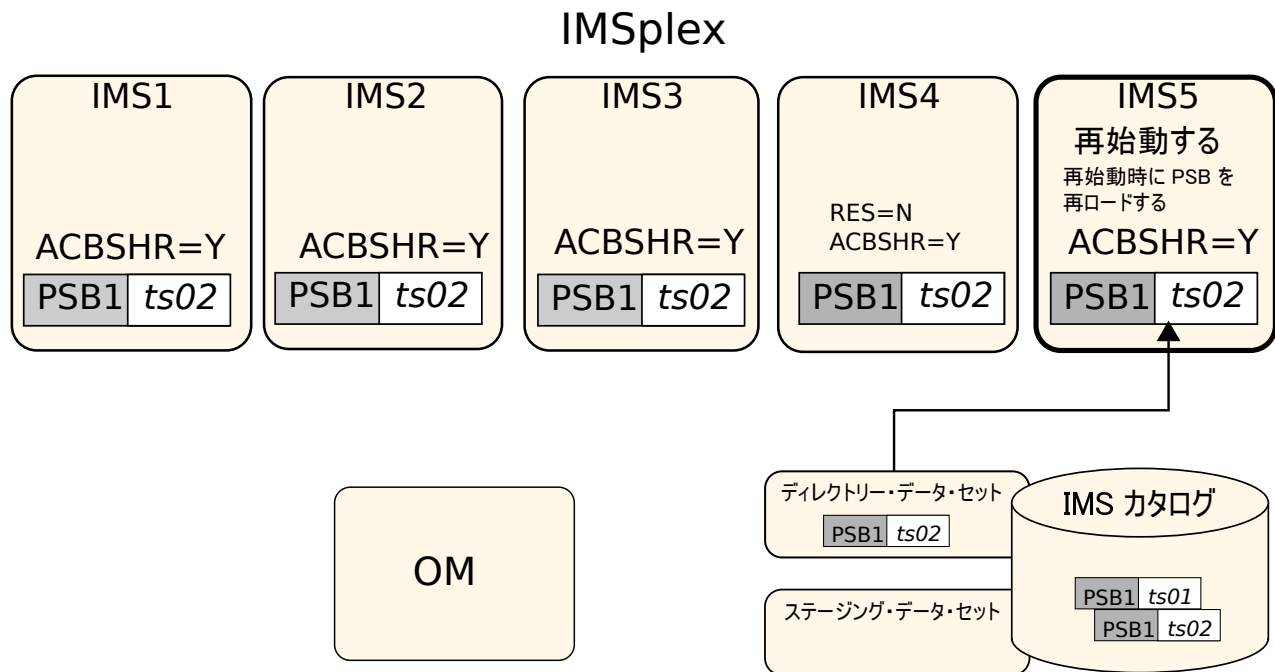


図 51. REFRESHPSB オプションが IMS3 上で新バージョンの PSB1 を活動化する

次の図では、IMS5 が再始動します。再始動前に、IMS5 は元のバージョンの PSB1 (これはリフレッシュ保留中状態だった) をロードしていました。しかし、再始動時にリフレッシュ保留中状態は除去され、新バージョンの PSB1 が IMS ディレクトリーからロードされます。



IMS5 は再始動され、再始動時にディレクトリー・データ・セットを読み取り、PSB1 ts02 をロードする。

図 52. IMS5 が再始動し、新バージョンの PSB1 をロードする

オンライン変更機能

オンライン変更機能には、ローカル・オンライン変更とグローバル・オンライン変更の 2 種類があります。

1 つの IMS について IMS リソースのオンライン変更を行うには、ローカル・オンライン変更機能を使用します。

IMSpdex 内のすべての IMS システムにわたって IMS リソースのオンライン変更を調整するには、グローバル・オンライン変更機能を使用します。

注: ACB の IMS 管理が使用可能になっている場合、オンライン変更機能は ACB の変更をサポートしません。代わりに、保留中の ACB 変更は、IMPORT DEFN SOURCE(CATALOG) コマンドを使用して、オンライン IMS システム内で活動化されます。詳しくは、管理対象 ACB 環境でのオンライン変更 (システム管理) を参照してください。

関連概念:

56 ページの『ACB ライブラリー・メンバー・オンライン変更』

関連タスク:

➡ オンライン変更機能を使用したデータベース変更の活動化 (データベース管理)

ローカル・オンライン変更機能の概要

ユーザーの IMS システムを止めることなしに、IMS データベース、プログラム、トランザクション、および MFSフォーマットをオンラインで追加、削除、置き換えができることは、連続稼働に向けての重要なステップです。IMS リソースの追加、削除、または変更に伴って、これらのリソース用にセットアップされた制御ブロックに対する変更も必要となります。

IMS.FORMAT、IMS.ACBLIB、または IMS.MODBLKS データ・セット内の制御ブロック・メンバーに、変更を適用できます。変更は、これらのデータ・セットに個別に適用するか、データ・セットの組み合わせに対して適用できます。

制約事項: リソース定義の動的定義が使用可能になっている場合、オンライン変更機能は以下のタイプのリソースに対してサポートされません。

- ACB の IMS 管理が使用可能である場合のデータベース (DBD) およびプログラム・ビュー (PSB) リソース。
- 動的リソース定義が使用可能である場合の MODBLKS リソース。

IMS オンライン変更機能を使用するには、以下のライブラリーのそれぞれについて 3 つのコピーを作成する必要があります。

- IMS.MODBLKS : データベース、プログラム、トランザクション、宛先コードのオンライン変更をサポートする制御ブロックが入っているライブラリー
- IMS.ACBLIB : データベース記述子とプログラム記述子が入っているライブラリー
- IMS.FORMAT : MFS 言語およびサービス・ユーティリティーが生成した MFS マップが入っているライブラリー

上にリストしたライブラリーは、IMS オフライン機能専用であり、ステージング・ライブラリーと呼ばれます。それぞれのライブラリーごとにコピーが作成され、接尾部に A および B がついたデータ・セット名 (例えば、IMS.FORMATA および IMS.FORMATB) を持つデータ・セットを生成します。各ライブラリーのこれら 2 つのコピーが、IMS オンライン・システムにより使用されます。

IMS IVP 処理の完了時には、ステージング・ライブラリーと IMS A ライブラリーは同一のものになります。この時点では、A ライブラリーのことを「アクティブ・ライブラリー」と呼びます。IMS は、これらのライブラリーから実行情報を取り出します。B ライブラリーはこの時点では使用されず、「非アクティブ・ライブラリー」と呼ばれます。

以下の図に、オンラインでご使用のシステムを変更する場合に、ライブラリーがどのように使用されるか示します。

1. 変更をステージング・ライブラリーに適用します。
2. その後、ステージング・ライブラリーは、オンライン変更ユーティリティーを使用して非アクティブ (B) ライブラリーへコピーされます。
3. 以下のためのオペレーター・コマンドが発行されます。
 - a. 変更によって影響を受けるリソースを照会し、変更が正しいかどうかを確認します。

- b. B ライブラリーがアクティブ・ライブラリーになるようにします。古いアクティブ (A) ライブラリーは非アクティブ・ライブラリーになります。

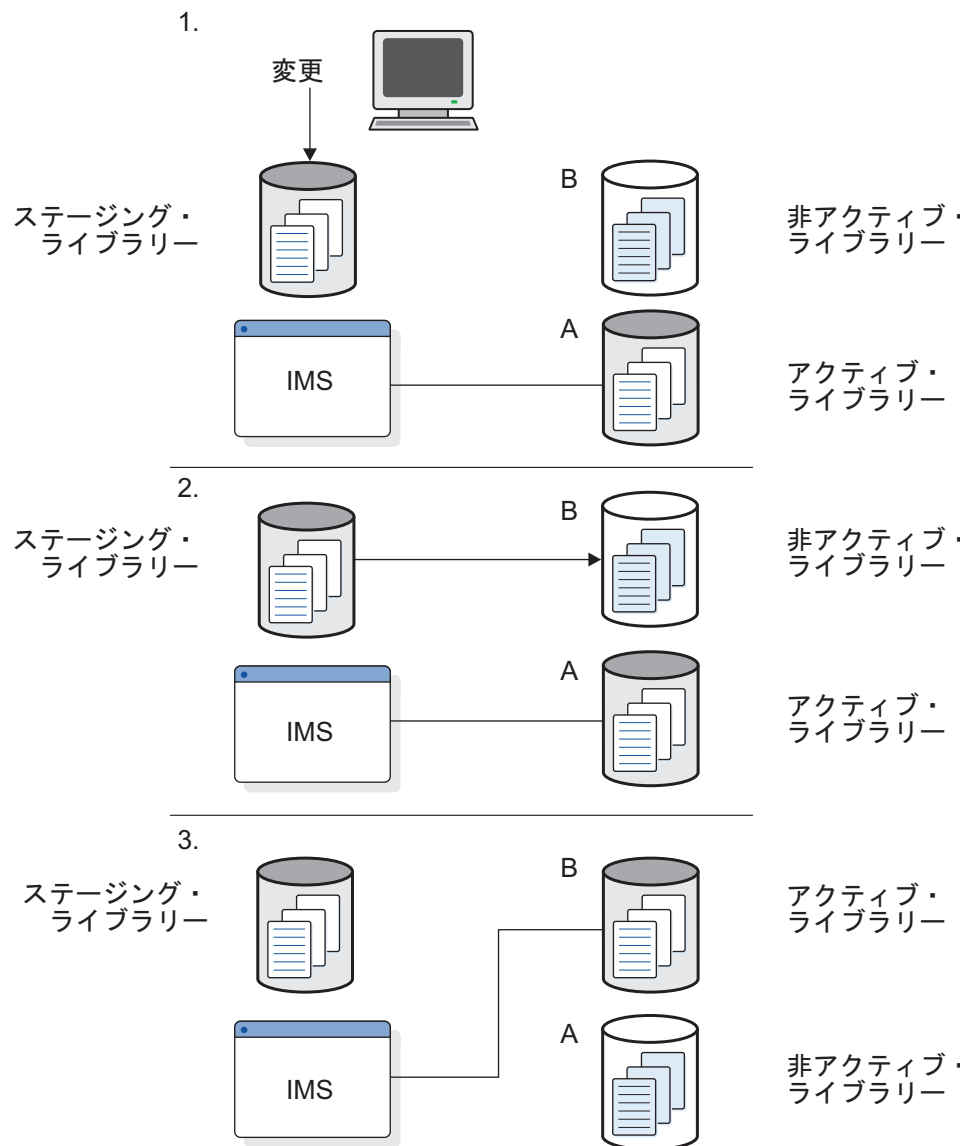


図 53. システムをオンラインで変更するときライブラリーが使用される様子

上記のプロセスは、必要に応じて繰り返されます。

グローバル・オンライン変更プロセスのバリエーションを使用すると、完全なライブラリー切り替えを行う代わりに ACB ライブラリーの個々のメンバーを変更できます。グローバル・オンライン変更プロセスのすべてのバリエーションの詳細については、577 ページの『グローバル・オンライン変更機能の概要』を参照してください。

このトピックに出てくる IMS リソースのいずれかを追加、置き換え、または削除することを選択したとき、ユーザーは以下のいずれかを行って、変更をオフラインのステージング・ライブラリーに適用します。

- MODBLKS タイプのシステム定義：アプリケーション、プログラム、全機能データベース、DEDB、宛先コード、またはトランザクションを追加、変更、または削除した場合

MODBLKS システム定義は、オンラインで追加や変更ができるリソース用の制御ブロック・メンバーを生成します。これらの制御ブロックは、ライブラリー IMS.MODBLKS に保管され、ユーザーの IMS システムに対するオンライン変更が要求された時に、IMS 制御領域、および複数システム結合検査ユーティリティーで使用されます。

- ACB 保守ユーティリティー (DFSUACB0)：データベースまたはプログラムの追加または変更を行った場合
- MFS 言語およびサービス・ユーティリティー：MFS 形式定義の追加または変更を行った場合

一連のコマンド (ローカル・オンライン変更には /MODIFY、グローバル・オンライン変更には INITIATE OLC) が出され、以前の非アクティブ・ライブラリーがアクティブ・ライブラリーになると、以前のアクティブ・ライブラリーは非アクティブ・ライブラリーになります。これらは、次のオンライン変更シーケンスによって上書きされるまで、破棄されません。バックアップおよびリカバリーが必要な場合、あるいはオンライン変更の実行中に正しくない定義が発生した場合は、非アクティブ・ライブラリーに戻すことができます。

さらに、IMS は、ユーザーに代わって、現在どのライブラリーのセットがアクティブになっているかをモニターします。ローカル・オンライン変更が可能な場合、この情報は状況データ・セット IMS.MODSTAT に保持されています。グローバル・オンライン変更が可能な場合、この情報は IMSPLEX.OLCSTAT データ・セットに保持されています。

オンライン変更が正常に完了した後で、この変更はすべてのタイプの IMS 再始動に有効となります。さらに、ユーザーの SMP 制御データ・セットにある新システム定義の内容を記録するためにユーザーの MODBLKS システム定義によって生成された、ステージ 1 出力ストリームに対して SMP/E JCLIN を実行することにより、新規のリソースは容易に保守できます。これにより、ユーザーの IMS システムに適用されたすべての保守が、現行のアクティブ IMS システムに確実に適用されます。マイグレーション/再呼び出しシステムを使用して、オンライン変更データ・セットを管理してはなりません。このシステムは、本来割り振られていたボリューム以外のボリュームに、データ・セットを再呼び出しする可能性があります。もしユーザーがこれを管理した場合、IMS はウォーム・スタートやシステムの緊急開始を行えなくなる可能性があります。

オンライン変更機能をサポートする IMS 環境

グローバル・オンライン変更は、ローカル・オンライン変更をサポートする、すべての IMS 環境内でサポートされているわけではありません。MODSTAT/MODSTAT2 データ・セットを要求する環境は、グローバル・オンライン変更をサポートしません。FDBR、XRF 代替、および DBCTL スタンバイは、グローバル・オンライン・プロセスに関与しませんが、ログ・レコードに基づいてオンライン変更を追跡します。

以下の表を使用して、ご使用の環境でグローバル・オンライン変更がサポートされているかどうか判断してください。

表 38. オンライン変更をサポートする IMS 環境

環境	ローカル・オンライン変更のサポート	グローバル・オンライン変更のサポート
DB/DC	あり	あり
DBCTL	あり	あり
DBCTL スタンバイ	なし	なし
DCCTL	あり	あり
FDBR	なし	なし
XRF アクティブ	あり	あり
XRF 代替	なし	なし

オンライン変更に適応できるコマンド

ローカル・オンライン変更操作を実行するには、/MODIFY PREPARE、/DISPLAY MODIFY、および /MODIFY COMMIT タイプ 1 IMS コマンドを使用します。

また、Structured Call Interface (SCI) と Operations Manager (OM) を含んでおり、Resource Manager (RM) を含んでいない最小の共通サービス層を備えた単一 IMS の IMSplex として IMS をセットアップしてある場合は、タイプ 2 IMS コマンドを使用して、ローカル・オンライン変更操作を行うこともできます。この IMS 構成は、拡張コマンド環境と呼ばれます。最小 CSL (RM なし) での IMS の定義については、「IMS V15 システム定義」を参照してください。単一 IMS の IMSplex でのグローバル・オンライン変更操作の実行 (ACB ライブラリー・メンバー・オンライン変更の実行を含む) については、577 ページの『グローバル・オンライン変更機能の概要』を参照してください。

オンライン変更を行う場合は、非アクティブ・ライブラリー (加えたい変更を含んでいるライブラリー) をアクティブにするか、ステージング・ライブラリーからメンバーをアクティブ・ライブラリーにコピーします。以下のコマンドを以下の順序で使用してください。

1. /MODIFY PREPARE

/MODIFY PREPARE コマンドを使用して、IMS をローカル・オンライン変更用に準備し、どのリソースを追加、変更、または削除するかを指定します。このコマンドを出すと、IMS は、変更または削除されるデータベース・プログラムまたはトランザクションへのメッセージを、端末がキューイングするのを防ぎます。

/MODIFY PREPARE の後に追加のコマンドを実行して、そのコミットが失敗した原因となった、進行中の作業を表示および解決する必要がある場合があります。

2. /DISPLAY MODIFY

/DISPLAY MODIFY コマンドを使用して、作業が保留中のリソース (変更または削除するリソース) のリストを生成します。作業保留中のときは、変更は受け入れられません。このコマンドは、/MODIFY COMMIT コマンドを出す前に、保留中の作業がないことを確認するのに使用します。

3. /MODIFY COMMIT

/MODIFY COMMIT コマンドを使用して、前の /MODIFY PREPARE コマンドで指定した変更を適用します。このコマンドが正常に終了すると、修正は、すべての IMS の再始動を通して維持されます。

/MODIFY PREPARE または /MODIFY COMMIT コマンドが失敗し、別の時期までオンライン変更を再試行したくない場合は、/MODIFY ABORT コマンドを使用して IMS 上のローカル・オンライン変更を中止し、状況を前の /MODIFY PREPARE または /MODIFY COMMIT コマンドを発行する以前の状況にリセットしてください。

システム定義のオンライン変更

IMS サブシステムに変更を加えるときは、必ずしもすべての IMS システム定義を再実行する必要はありません。変更を調べて、オンライン変更が行えるかどうかを確認します。

要求に、IMS ネットワークまたは静的端末への変更が含まれていない場合は、おそらくオンライン変更が可能です。

IMS オンライン・システムに変更を加える手順は、以下のように要約できます。

1. 必要があれば、必要なシステム定義ステージ 1 の入力変更を行う。
2. 必要があれば、MODBLKS システム定義を行う。
3. 必要があれば、MFS 言語ユーティリティーおよび MFS サービス・ユーティリティーを実行する。
4. 必要があれば、DBD、PSB、および ACB を生成する。
5. 必要があれば、オンライン変更コピー・ユーティリティーを使用して、IMS.MODBLKS、IMS.ACBLIB、および IMS.FORMAT データ・セットの適切なコピーを生成する。
6. 適当なオンライン変更コマンド・シーケンスを実行し、オンライン変更を適切な時期に行う。それらの変更は、ライブラリー切り替え OLC 用に非アクティブ・ライブラリーからコピーする必要があります。
7. 必要な変更が行われていたか検査する。

オンライン変更を取り消すか、バックアウトする必要がある場合は、非アクティブなライブラリーを変更しないようにして、オンライン変更コマンドを繰り返し行ってください。

プログラムおよびトランザクションの変更

IMS DB/DC または DCCTL 環境においては、オンライン変更 PREPARE コマンドは、変更するリソースのすべての着信メッセージおよびトランザクションを停止しますが、すでにキューに入っているメッセージには影響を与えません。オンライン変更 COMMIT コマンドは、直接影響を受けた変更または削除するリソースのメッセージがスケジュール（デキュー）されるまでは有効になりませんので、メッセージがスケジュールされるのを待つが、関連トランザクションを停止する必要があります。

正当な理由から、スケジュールするメッセージ (有効になるオンライン変更 COMMIT コマンドの) を待ちたくない場合は、 /STOP または /PSTOP コマンドを使用して、関連トランザクションを停止してから、オンライン変更 COMMIT コマンドを実行します。

推奨事項: 変更は、リソースに関連するアクティビティーが少ないとき、つまり処理するトランザクションが少ないときに行うようにプランします。アクティビティーが少ないときは、メッセージがスケジュールされるのを長く待つ必要はなく、逆に待ちたくない場合も、多数のトランザクションを停止させる必要はありません。

入力待ちプログラムがスケジュールに入れられている場合はオンライン変更 COMMIT コマンドが実行されないため、メッセージ・ドリブンの高速機能プログラム、あるいは変更または削除されるリソースを直接参照する入力待ちプログラムは、すべて停止する必要があります。

また、オンライン変更 COMMIT コマンドが完了する前に、変更または削除するリソースを含むアクティブな会話は、すべて完了していなければなりません。

オンラインでのセキュリティ変更の実装

IMSがアクティブであるときは、RACFセキュリティ定義に対する変更は、完全にIMSの外側で行われます。

以下の一般的な手順は、RACF に対する変更方法を説明したものです。

1. RACF データベースで、RACFコマンドを出して、IMS リソース (例えば、トランザクションおよびコマンド) を記述する RACFセキュリティ・プロファイルを更新する。
2. RACFコマンド SETROPTS RACLIST(classname) REFRESH を出して、RACF データ・スペース内の該当セキュリティ・クラスのプロファイルを最新表示する。

MFS フォーマットの変更

オンライン変更 COMMIT コマンドを使用すると、関連アクティビティーを停止しないで MFS フォーマットを変更することができます (関連する MFS フォーマット・ブロックを変更する間、メッセージをキューおよびスケジュールすることができます)。

しかし、問題が発生しないように、MFS フォーマットの変更とメッセージ・キューイングを調整する必要があります。

IMS DB/DC および DCCTL 環境では、MFS フォーマットを変更するとき、オンライン変更 PREPARE コマンドは関連メッセージのキューイングを停止しません。

DEDB のオンライン変更の実行

IMS DB/DC および DBCTL 環境では、高速機能の高速処理データベース (DEDB) に対するデータベース・レベルとエリア・レベルの両方の変更を行うことができます。

データベース・レベルの変更は、DEDB の構造に影響し、エリアの追加または削除、セグメント・タイプの追加、またはランダムマイザー・ルーチンの変更などの変

更が含まれます。エリア・レベルの変更には、オーバーフロー部分 (IOVF と DOVF) および制御インターバル (CI) サイズなど、エリアのサイズの増減が含まれます。

データベース・レベルの変更の場合は、DEDB のすべてのエリアを停止する必要があります。エリア・レベルを変更する場合は、/DBRECOVERY AREA コマンドを使用してエリアを停止する必要があります。

変更された DEDB にアクセスする IFP および MPP 領域は、オンライン変更の間停止する必要はありません。

関連資料: DEDB をオンラインで追加、修正、または削除する場合に関する詳細については、「IMS V15 データベース管理」を参照してください。

HALDB のオンライン変更の実行

HALDB 区画での IMS オンライン変更には、以下の制約事項が適用されます。

- ALL または MODBLKS システム定義中に、IMS のコールド・スタートを実行せずに、現在、HALDB 区画として定義されているデータベースを再定義することはできません。コールド・スタートは、データベースをオンラインで削除する場合でも必要です。すなわち、HALDB 区画を削除する場合、IMS コールド・スタートなしで ALL または MODBLKS オンライン変更中に追加する予定のデータベースに、このデータベース名を再利用することはできません。
- ALL または MODBLKS システム定義中に定義されたデータベースを、IMS のコールド・スタートを実行せずに HALDB 区画に変換することはできません。コールド・スタートは、データベースをオンラインで削除する場合でも必要です。IMSコールド・スタート後、そのデータベースを HALDB 区画として再定義する必要があります。

これらの制約事項は、HALDB マスター・データベースには適用されません。

FDBR とローカル・オンライン変更

FDBR は、x'70' ログ・レコードを介してオンライン変更を追跡します。

オンライン変更の実行後に FDBR がシャットダウンまたは再始動された場合に問題が生じないように、またはオンライン変更後に起こったその他の状況による問題を回避するために、オンライン変更が完了した後で FDBR がシャットダウンまたは再始動される前に、1 つ以上の IMS システム・チェックポイントを取ってください。これにより、最後のオンライン変更の後で取られたチェックポイントから FDBR が再始動するようになります。

例えば、オンライン変更の後で FDBR を再始動する前に、アクティブ・システム上で /CHECKPOINT コマンドを発行して、単純チェックポイントを取ります。DFS3804I メッセージから、最後の再始動チェックポイントのタイム・スタンプがオンライン変更後であることを確認します。

XRF 複合システム内でのローカル・オンライン変更の実行

IMS DB/DC および DCCTL XRF 環境では、ステージング・ライブラリーの準備に加えて、非アクティブ・ライブラリーを準備する必要があります。

アクティブ IMS が /MODIFY COMMIT コマンドを処理する間に XRF のテークオーバーが発生すると、変更は、行われる場合も、行われない場合もあります。オンライン変更の終わりに、IMS はログ・レコードを書き込みます。XRF 代替サブシステムは、このログを読み取り、オンライン変更を行って、アクティブ・サブシステム上の変更と突き合わせます。代替サブシステムで発生した変更をチェックするときは、/DISPLAY DB コマンドまたは QUERY DB コマンドを使用します。

オンライン変更のパフォーマンスの考慮事項

オンライン変更を使用して IMS.MODBLKS データ・セットからデータベースを削除すると、意図リストがロード済みでない MODBLKS 内のすべてのアプリケーション・プログラムについて、ACB ライブラリーから意図リストを読み取るというパフォーマンス上のオーバーヘッドが生じます。

これは、コミットを失敗させる原因となる可能性のある作業を防止するため、キューイングするのを停止するトランザクションを決定するのに必要です。このパフォーマンス・オーバーヘッドは、意図リストがロードされない、MODBLKS に定義されているアプリケーション・プログラムの数に比例します。

推奨事項: MODBLKS からデータベースを削除する必要がある場合は、ローカル・オンライン変更またはグローバル・オンライン変更を使用する代わりに、IMS コールド・スタートを実行することを検討してください。

グローバル・オンライン変更機能の概要

グローバル・オンライン変更は、IMSplex 内のすべての IMS システムに対するリソースを変更する IMS機能です。ユーザーまたは Operations Manager (OM) によって指定されたマスター IMS 制御領域は、Resource Manager (RM) を使用して、オンライン変更のフェーズと IMSplex 内のその他の IMS システムとの調整を行います。

制約事項: リソース定義の動的定義が使用可能になっている場合、オンライン変更機能は以下のタイプのリソースに対してサポートされません。

- ACB の IMS 管理が使用可能である場合のデータベース (DBD) およびプログラム・ビュー (PSB) リソース。
- 動的リソース定義が使用可能である場合の MODBLKS リソース。

メンバー・オンライン変更の場合、INIT OLC TYPE(ACBMBR) NAME(XXX) で指定されたメンバー名および関連するメンバーが、ステージング・ライブラリーからアクティブ・ライブラリーにコピーされます。ステージング・ライブラリーには、コピーされないメンバーが含まれる場合もあります。IMS.ACBLIB の個々のメンバーを追加または変更し、IMSplex を静止せずに、またアクティブな ACB ライブラリーのフル・リフレッシュも必要とせずに、これらのメンバーをオンラインにすることができます。

ライブラリー交換オンライン変更の場合 (ローカルまたはグローバルに関わらず)、アクティブなライブラリーと非アクティブなライブラリーが交換されます。

オンラインでの IMS.ACBLIB メンバーの変更または追加について詳しくは、591 ページの『オンラインでの IMS.ACBLIB メンバーの変更または追加』を参照してください。

RM を使用しない IMSplex を実行している場合は、DFSCGxxx IMS.PROCLIB メンバーに CSL グローバル・パラメーター RMENV=N を指定できます。このパラメーターが指定されていると、IMSでは RM は不要であるので、RM サービスを使用しませんが、タイプ 2 コマンド・サポートは使用できます。OLC=GLOBAL の場合は、INITIATE OLC コマンドを使用して、ローカル・オンライン変更を開始してください。OLCSTAT データ・セットを定義する必要があります。このデータ・セットは、IMSシステム間で共用できません。

グローバル・オンライン変更を使用して、マスター IMS 制御領域は、IMSplex 内のすべての IMS システムにある以下のリソースのオンライン変更を調整します。

- データベース (IMS.ACBLIB 内の DMB)
- データベース・ディレクトリー (IMS.MODBLKS 内の DDIR)
- MFS フォーマット (IMS.FMTLIB)
- プログラム (IMS.ACBLIB 内の PSB タイプの ACB)
- プログラム・ディレクトリー (IMS.MODBLKS 内の PDIR)
- トランザクション (IMS.MODBLKS 内の SMB)

グローバル・オンライン変更に適用できるコマンド

グローバル・オンライン変更を行うには、コマンドを発行して、非アクティブ・ライブラリー (その変更が含まれているライブラリー) をアクティブにする、またはメンバーをステージング・ライブラリーからアクティブ・ライブラリーにコピーします。

以下のコマンドは、非アクティブ・ライブラリーをアクティブにするか、ステージング・ライブラリーをアクティブ・ライブラリーにコピーします。

1. /DISPLAY MODIFY

このコマンドは、保留中の作業を持つリソース (変更または削除を行う) のリストを作成します。作業保留中のときは、変更は受け入れられません。このコマンドは、INITIATE OLC PHASE(PREPARE) コマンドを出した後、および INITIATE OLC PHASE(COMMIT) コマンドを出す前に、保留中の作業がないことを確認するのに使用します。

2. INITIATE OLC PHASE(PREPARE)

このコマンドには、どのリソースを追加、変更、または削除するかを指定することで行うグローバル・オンライン変更に関する、IMSplex 内のすべての IMS システムが準備されています。

このコマンドは、RM を使用しない (RMENV=N) 環境で実行すると、コマンド・マスターである IMS、またはこのコマンドが送付された単一の IMS 上のみでオンライン変更を行います。

3. INITIATE OLC PHASE(COMMIT)

このコマンドは、IMSplex 内のすべての IMS システム上のグローバル・オンライン変更をコミットします。

このコマンドは、RM を使用しない (RMENV=N) 環境で実行すると、このコマンドを処理する IMS 上でローカル・オンライン変更を行います。

/INITIATE OLC PHASE(PREPARE) の後に追加のコマンドを実行して、そのコミットが失敗した原因となった、進行中の作業を表示および解決する必要がある場合があります。これらのコマンドには、以下が含まれます。

TERMINATE OLC

このコマンドは、IMSplex 内のすべての IMS システム上のグローバル・オンライン変更を打ち切ります。

このコマンドは、RM を使用しない (RMENV=N) 環境で実行すると、このコマンドを処理する IMS 上でローカル・オンライン変更を終了します。

QUERY MEMBER

このコマンドは、適用できる場合であれば、ローカルまたはグローバル・オンライン変更状況を含む、IMSplex 内の IMS システムに関する状況または属性を表示します。

QUERY OLC

このコマンドは、OLCSTAT データ・セット・コンテンツ、または TYPE(ACBMBR) OLC で変更されているリソースに関する情報を提供します。

システム定義のオンライン変更

IMS サブシステムに変更を加えるときは、必ずしもすべての IMS システム定義を再実行する必要はありません。変更を調べて、オンライン変更が行えるかどうかを確認します。

要求に、IMS ネットワークまたは静的端末への変更が含まれていない場合は、おそらくオンライン変更が可能です。

IMS オンライン・システムに変更を加える手順は、以下のとおりです。

1. 必要があれば、必要なシステム定義ステージ 1 の入力変更を行う。
2. 必要があれば、MODBLKS システム定義を行う。
3. 必要があれば、MFS 言語ユーティリティーおよび MFS サービス・ユーティリティーを実行する。
4. 必要があれば、DBDGEN、PSBGEN、および ACBGEN を実行する。
5. 必要があれば、オンライン変更コピー・ユーティリティーを使用して、IMS.MODBLKS、IMS.ACBLIB、および IMS.FORMAT データ・セットの適切なコピーを生成する。
6. 適当なオンライン変更コマンド・シーケンスを実行し、オンライン変更を行う。それらの変更は、ライブラリー切り替え OLC 用に非アクティブ・ライブラリーからコピーする必要があります。
7. 必要な変更が行われていたか検査する。

オンライン変更を取り消すか、バックアウトする必要がある場合は、非アクティブなライブラリーを変更しないようにして、オンライン変更コマンドを繰り返し行ってください。

プログラムおよびトランザクションの変更

IMS DB/DC または DCCTL 環境においては、オンライン変更 PREPARE コマンドは、変更するリソースのすべての着信メッセージおよびトランザクションを停止しますが、すでにキューに入っているメッセージには影響を与えません。

オンライン変更 COMMIT コマンドは、直接影響を受けた変更または削除するリソースのメッセージがスケジュール (デキュー) されるまでは有効になりませんので、メッセージがスケジュールされるのを待つか、関連トランザクションを停止する必要があります。

スケジュールするメッセージ (有効になるオンライン変更 COMMIT コマンドの) を待ちたくない場合は、 /STOP または /PSTOP コマンドを使用して、関連トランザクションを停止してから、オンライン変更 COMMIT コマンドを実行します。

推奨事項: 変更は、リソースに関連するアクティビティーが少ないとき、つまり処理するトランザクションが少ないときに行うようにプランします。アクティビティーが少ないときは、メッセージがスケジュールされるのを長く待つ必要はなく、逆に待ちたくない場合も、多数のトランザクションを停止させる必要はありません。

入力待ちプログラムがスケジュールされた場合は、オンライン変更 COMMIT コマンドは実行しないため、メッセージ・ドリブンの高速機能プログラムや、変更または削除するリソースを参照する入力待ちプログラムは、すべて停止する必要があります。

また、オンライン変更 COMMIT コマンドが完了する前に、変更または削除するリソースを含むアクティブな会話は、すべて完了していなければなりません。

オンラインでのセキュリティ変更の実装

IMS がアクティブであるときは、RACF セキュリティ定義に対する変更は、完全に IMS の外側で行う必要があります。

以下の一般的な手順は、RACF に対する変更方法を説明したものです。

1. RACF データベースで、RACFコマンドを出して、IMS リソース (例えば、トランザクションおよびコマンド) を記述する RACFセキュリティ・プロファイルを更新する。
2. RACFコマンド SETROPTS RACLIST(classname) REFRESH を出して、RACF データ・スペース内の該当セキュリティ・クラスのプロファイルを最新表示する。

MFS フォーマットの変更

IMS DB/DC および DCCTL 環境では、MFS フォーマットを変更するとき、オンライン変更 PREPARE コマンドは関連メッセージのキューイングを停止しません。

オンライン変更 COMMIT コマンドを使用すると、関連アクティビティーを停止しないで MFS フォーマットを変更することができます (関連する MFS フォーマット・ブロックを変更する間、メッセージをキューおよびスケジュールすることができます)。ただし、問題が起きないように、MFS フォーマットおよびメッセージ・キューイングの変更を調整する必要があります。

DEDB のオンライン変更の実行

IMS DB/DC および DBCTL 環境では、高速機能の高速処理データベース (DEDB) に対するデータベース・レベルとエリア・レベルの両方の変更を行うことができま

す。データベース・レベルの変更は、DEDB の構造に影響し、エリアの追加または削除、セグメント・タイプの追加、またはランダムマイザー・ルーチンの変更などの変更が含まれます。

エリア・レベルの変更には、オーバーフロー部分 (IOVF と DOVF) および制御インターバル (CI) サイズなど、エリアのサイズの増減が含まれます。

データベース・レベルの変更の場合は、DEDB のすべてのエリアを停止する必要があります。エリア・レベルを変更する場合は、UPDATE AREA STOP(ACCESS) コマンドを使用してエリアを停止する必要があります。

変更された DEDB にアクセスする IFP および MPP 領域は、オンライン変更の間停止する必要はありません。

関連資料: DEDB をオンラインで追加、修正、または削除する場合に関する詳細については、「IMS V15 データベース管理」を参照してください。

HALDB のオンライン変更の実行

HALDB 区画での IMS オンライン変更には、以下の制約事項が適用されます。

- ALL または MODBLKS システム定義中に、IMS のコールド・スタートを実行せずに、現在、HALDB 区画として定義されているデータベースを再定義することはできません。コールド・スタートは、データベースをオンラインで削除する場合でも必要です。すなわち、HALDB 区画を削除する場合、IMS コールド・スタートなしで ALL または MODBLKS オンライン変更中に追加する予定のデータベースに、このデータベース名を再利用することはできません。
- ALL または MODBLKS システム定義中に定義されたデータベースを、IMS のコールド・スタートを実行せずに HALDB 区画に変換することはできません。コールド・スタートは、データベースをオンラインで削除する場合でも必要です。IMSコールド・スタート後、そのデータベースを HALDB 区画として再定義する必要があります。

これらの制約事項は、HALDB マスター・データベースには適用されません。

ランタイム・リソース定義の変更

DRD を使用して、IMS リソースおよびその記述子を動的に作成、更新、および削除できます。ただし、ACB メンバーの変更には、ACB メンバー・オンライン変更機能を使用する必要があります。

ACB メンバーは、CREATE DB コマンドまたは CREATE PGM コマンドの一部として、またはオンライン変更を使って、データベースまたはプログラム・リソースと関連付けることができます。IMSplex 内の ACB メンバーを関連付けるには、次の 2 つの方法があります。

- ACBLIB 内のすべての ACB を作成した後、DRD を使用して データベースおよびプログラム・リソースを ACB に関連付けます。
- DRD コマンドを使用してデータベースおよびプログラム・リソースを作成した後、オンライン変更を行ってリソースを ACB に関連付けます。

データベースおよびプログラム・リソース用の CREATE コマンドでは、無償提供の ACB メンバーが ACBLIB に存在する必要はありません。作成中のリソースに対

し ACB メンバーが存在している場合、CREATE コマンド発行時にリソースと ACB メンバー間の関連付けが作成され、オンライン変更は必要ありません。

DRD コマンドによる **ACB** メンバーの関連付け:

ACB 生成を実行してから、CREATE DB または CREATE PGM コマンドを発行して、新しいデータベースまたはプログラム・リソースを作成します。

ACB メンバーを DRD CREATE コマンドの一部として関連付けるには、次のステップを実行します。

1. ACB 生成を実行し、ステージング ACB ライブラリー内に実行可能 ACBLIB メンバーを作成します。
2. INITIATE OLC コマンドを発行して、その ACB メンバーをアクティブな ACBLIB にします。
3. CREATE DB または CREATE PGM コマンドを発行して、新しいデータベースまたはプログラム・リソースを作成します。CREATE コマンドにより、データベースは自動的に ACBLIB に関連付けられます。

ACB メンバー・オンライン変更機能による **ACB** メンバーの関連付け:

ACB 生成を実行してから、INITIATE OLC コマンドを発行して、ACB メンバーをデータベースまたはプログラム・リソースに関連付けます。

ACB メンバー・オンライン変更機能を使って ACB メンバーを関連付けるには、次のステップを実行します。

1. CREATE DB コマンドを発行して、新しいデータベースを作成します。
2. ACB 生成を実行し、ステージング ACB ライブラリー内に実行可能 ACBLIB メンバーを作成します。
3. TYPE(ACBMBR) パラメーターを指定して INITIATE OLC コマンドを発行して、更新したメンバーまたは新規メンバーをステージング・ライブラリーからアクティブ・ライブラリーにコピーし、ACB メンバーをデータベースまたはプログラム・リソースに関連付けます。

ACB オンライン変更機能による **ACB** メンバー:

ACB 生成を実行してから、INITIATE OLC コマンドを発行して、ACB メンバーをデータベースまたはプログラム・リソースに関連付けます。

ACB オンライン変更機能を使って ACB メンバーを関連付けるには、次のステップを実行します。

1. CREATE DB コマンドを発行して、新しいデータベースを作成します。
2. ACB 生成を実行し、ステージング ACB ライブラリー内に実行可能 ACBLIB メンバーを作成します。
3. INITIATE OLC コマンドを発行して、ライブラリー交換オンライン変更を行い、ACB メンバーをデータベースまたはプログラム・リソースに関連付けます。

FDBR とグローバル・オンライン変更

FDBR は、x'70' ログ・レコードを介してオンライン変更を追跡します。

オンライン変更の実行後に FDBR がシャットダウンまたは再始動された場合に問題が生じないように、またはオンライン変更後に起こったその他の状況による問題を回避するために、オンライン変更が完了した後で FDBR がシャットダウンまたは再始動される前に、1 つ以上の IMS システム・チェックポイントを取ってください。これにより、最後のオンライン変更の後で取られたチェックポイントから FDBR が再始動するようになります。

例えば、オンライン変更の後で FDBR を再始動する前に、アクティブ・システム上で /CHECKPOINT コマンドを発行して、単純チェックポイントを取ります。DFS3804I メッセージから、最後の再始動チェックポイントのタイム・スタンプがオンライン変更後であることを確認します。

XRF 複合システム内でのローカル・オンライン変更の実行

IMS DB/DC および DCCTL XRF 環境では、ステー징・ライブラリーの準備に加えて、非アクティブ・ライブラリーを準備する必要があります。

アクティブ IMS が INIT OLC PHASE(COMMIT) コマンドを処理する間に XRF のテークオーバーが発生すると、変更は、行われる場合も、行われない場合もあります。オンライン変更の終わりに、IMS はログ・レコードを書き込みます。XRF 代替サブシステムは、このログを読み取り、オンライン変更を行って、アクティブ・サブシステム上の変更と突き合わせます。代替サブシステムで発生した変更をチェックするときは、/DISPLAY DB コマンドまたは QUERY DB コマンドを使用します。

XRF を使用するシスプレックスのグローバル・オンライン変更作成

XRF 代替は、グローバル・オンライン変更に直接は参加しません。代替は、XRF アクティブ・システムによって書き込みされた X'70' および X'22' ログ・レコードをいったん受信したら、間接的に参加します。

QUERY MEMBER コマンドは、X'70' および X'22' ログ・レコードを処理している間に、XRF 代替に関するオンライン変更状況を表示する場合があります。

アクティブ・システムをテークオーバーしている間に、XRF 代替は、その IMSID とともに OLCSTAT データを更新します。テークオーバーが完了した後、XRF 代替は、IMS アクティブ・システムのように作動し、グローバル・オンライン変更に参加することができます。必要があれば、異常終了した XRF アクティブ・システムは、新規の XRF 代替としてバックアップすることが可能です。

MODSTAT および MODSTAT2 データ・セットは使用されません。ご使用の IMS 制御領域 JCL 内で MODSTAT および MODSTAT2 データ・セット DD ステートメントを残すこともできますが、無視されます。

XRF 環境のグローバル・オンライン変更には、以下の制限があります。

- XRF アクティブ・システムが障害を起こしたか、またはシャットダウンした (/SWITCH SYSTEM ACTIVE または /CHE FREEZE) ときに、XRF 代替システムがテークオーバーする場合、IMS は、障害またはシャットダウン XRF アクティブ・システムの IMSID を OLCSTAT データ・セットからテークオーバー処理中に削除して、前のアクティブ・システムがダウンしている場合でもグローバル・オンライン変更が作動できるようにします。テークオーバー後に、障害またはシャットダウン XRF アクティブ・システムを新しい XRF 代替システムと

して起動してグローバル・オンライン変更を開始した場合、新しい XRF 代替システムはオンライン変更コマンドを直接処理せず、後で、オンライン変更ログ・レコードを検出したときにオンライン変更を実行します。

- XRF アクティブ・システムが障害を起こし、かつ XRF 代替システムが、XRF アクティブ・システムのグローバル・オンライン変更状態に追いつく前にテークオーバーしようとする場合、XRF 代替システムは U2801 RC=X'0011' とともに異常終了します。XRF 代替は、OLCSTAT データ・セットとは同期しません。IMSplex 内のその他の IMS システムが OLCSTAT データ・セットと同期している場合があるため、XRF 代替はテークオーバーを許可されません。

XRF テークオーバーおよび DBCTL 待機緊急時再始動後のグローバル・オンライン変更

XRF テークオーバー中、前のアクティブ・システムの IMSID が OLCSTAT から除去され、新しいアクティブ・システムの IMSID が OLCSTAT に追加されます。アクティブ・システムの IMSID 除去中にエラーがあった場合、テークオーバーは完了しますが、IMSID が OLCSTAT 内に存在します (前の XRF アクティブ IMSID と新しい XRF アクティブ IMSID)。

RMENV=N および OLC=GLOBAL が指定された DBCTL 待機の緊急時再始動後に、OLCSTAT データ・セットに 2 つの IMSID が存在します (前の DBCTL アクティブ IMSID と新しい DBCTL アクティブ IMSID)。

FUNC=DEL を指定したグローバル・オンライン変更ユーティリティ (DFSUOLC0) を実行して、前のアクティブ XRF または前の DBCTL アクティブ IMS システムの IMSID を OLCSTAT データ・セットから削除できます。それ以外の場合、OPTION(FRCABND,FRCNRML) を指定しないと、グローバル・オンライン変更を開始するために次に発行した INITIATE OLC PHASE(PREPARE) コマンドは失敗します。

関連資料: XRF および DBCTL 待機について詳しくは、743 ページの『XRF の概要』を参照してください。

オンライン変更のパフォーマンスの考慮事項

オンライン変更機能の用途に応じて、異なるパフォーマンスの考慮事項が適用されます。

MODBLKS データ・セットからのデータベースの削除

オンライン変更を使用して MODBLKS からデータベースを削除すると、MODBLKS にあって意図リストがロードされていないすべてのアプリケーション・プログラムについて ACBLIB から意図リストを読み取る、パフォーマンス・オーバーヘッドを生じることになります。

これは、コミットを失敗させる原因となる可能性のある作業を防止するため、キューイングするのを停止するトランザクションを決定するのに必要です。このパフォーマンス・オーバーヘッドは、意図リストがロードされない、MODBLKS に定義されているアプリケーション・プログラムの数に比例します。

推奨事項: MODBLKS からデータベースを削除する必要がある場合は、ローカル・オンライン変更またはグローバル・オンライン変更を使用する代わりに、IMS コールド・スタートを実行することを検討してください。

ACB メンバーの変更

デフォルトで、DBD が INITIATE OLC PHASE(PREPARE) TYPE(ACBMBR) コマンドの NAME キーワードで指定される場合、ACB メンバー・オンライン変更 (OLC) 機能により、関連した PSB および外部参照 DBD がすべて、ステージング ACB ライブラリーからアクティブ ACB ライブラリーに自動的にコピーされます。PSB および DBD の数によっては、コピー・プロセスに必要な時間量が大き場合があります。

多くの場合、INITIATE OLC PHASE(PREPARE) TYPE(ACBMBR) コマンドで OPTION(NAMEONLY) を指定することによって、ACB メンバー・オンライン変更機能に必要な時間量を大幅に短縮することができます。NAMEONLY オプションは、ACB メンバー・オンライン変更機能の処理を、コマンドの NAME() キーワードで指定されている DBD と PSB のみに制限します。

関連資料:

 INITIATE OLC コマンド (コマンド)

シスプレックスのローカル・オンライン変更の実行

IMS は、シスプレックス全体のローカル・オンライン変更を自動的に調整しません。グローバル・オンライン変更を使用し、シスプレックス全体のオンライン変更を調整してください。このトピックでは、シスプレックス全体のローカル・オンライン変更を手動で調整する方法について説明します。

推奨事項: シスプレックス全体で IMS サブシステムを複製し、ライブラリー (ACBLIB、MODBLKS、FORMAT) を共有してください。MODSTAT ID は、シスプレックス全体で同じでなければなりません。

シスプレックス環境で IMS オンライン変更を行う準備は、以下のように行います。

1. 自動化操作 (AO) を行うアプリケーション・プログラムを使用する。AO アプリケーション・プログラムは、以下のことが行えなければなりません。
 - 変更または削除したいリソースを停止する。
 - 個々の IMS サブシステムのオンライン変更例外条件を処理する。
 - /MODIFY PREPARE または /MODIFY ABORT コマンドを出す。

IMS が出すオンライン変更メッセージを探し、推奨アクションを取る、AO 出口ルーチンを設計します。オンライン変更の問題があった場合にそれを解決するためのコマンドを発行するように、AO アプリケーション・プログラムを設計するか、自動化ツールを使用します。

2. アクティブ、非アクティブ、およびステージングとしてライブラリーを定義する。
3. アクティブおよび非アクティブ・ライブラリーを共有する。

オンラインでの変更を可能にしたいすべての IMS サブシステムは、ステージング・ライブラリーも共用しなければなりません。

4. 各 IMS を定義する (その独自の MODSTAT データ・セットによって)。

各 IMS は、それぞれ独自の MODSTAT データ・セット (アクティブおよび非アクティブな ACBLIB、FMTLIB、および MODBLKS 定義が同じ) を持つ必要があります。

5. 各 IMS を始動する。

オンライン変更を行うには、次のようにします。

1. 変更のステージング・ライブラリーにシステム定義を行う。
2. オンライン変更コピー・ユーティリティを実行する
3. IMS サブシステムごとに /MODIFY PREPARE を出す。

このコマンドは、IMS オペレーターか AO アプリケーション・プログラムのいずれかから出すことができます。コマンドがシズプレックス内の IMS サブシステムで失敗した場合は、オペレーターまたは AO アプリケーション・プログラムは、/MODIFY PREPARE が成功したすべての IMS サブシステムに /MODIFY ABORT コマンドを出す必要があります。

4. オプション: 変更または削除するすべてのリソースを停止する。

このステップを使用すると、個々の IMS サブシステムでオンライン変更が成功する機会が著しく増加します。

5. 処理中の作業があれば解決する。
6. /DISPLAY MODIFY コマンドを出す。

COMMIT フェーズを試みる前に、/DISPLAY MODIFY コマンドを使用して、変更または削除されるリソースに関する進行中の作業を表示してください。

COMMIT が成功した場合、IMS がダウンしている間にグローバル・オンライン変更が起こらない限り、変更がすべての IMS 再始動にわたって存続します。

7. IMS サブシステムごとに /MODIFY ABORT|COMMIT を出す。
8. 変更または追加されたリソースを開始する。

グローバル・オンライン変更機能の使用可能化

グローバル・オンライン変更ユーティリティ (DFSUOLC) を使用して、OLCSTAT データ・セットを初期設定し、グローバル・オンライン変更に対して IMSplex を使用可能にします。

グローバル・オンライン変更に対して IMSplex を使用可能にするには、以下が必要です。

1. グローバル・オンライン変更ユーティリティ (DFSUOLC0) を実行し、OLCSTAT データ・セットを初期設定する。

グローバル・オンライン変更ユーティリティは、最初の IMS を初めてコールド・スタートする前に一度実行する必要があります。OLCSTAT データ・セットは、ローカル・オンライン変更によって使用される MODSTAT データ・セットに類似しています。

重要:

- 誤って有効 OLCSTAT データ・セットの内容を破棄しないよう、注意してグローバル・オンライン変更ユーティリティを使用してください。有効 OLCSTAT データ・セットの内容を破棄した場合は、OLCSTAT データ・セットが再初期設定されるまで、グローバル・オンライン変更および追加の IMS システムの初期設定は失敗します。

OLCSTAT データ・セットの損失を処理する OLCSTAT データ・セット・リカバリー手順を確立する必要があります。グローバル・オンライン変更が正しく行われるたびに、変更 ID、アクティブ・オンライン変更ライブラリ接尾部、およびオンライン変更ライブラリと現行の IMS システムのリストを記録してください。OLCSTAT データ・セットが破棄された場合は、保管されたデータを使用してグローバル・オンライン変更ユーティリティの初期設定機能を実行し、OLCSTAT データ・セットを再初期設定してください。

関連資料: グローバル・オンライン変更ユーティリティに関する詳細については、「IMS V15 システム・ユーティリティ」を参照してください。

- 多重システム・シスプレックスを構築すると、同時に自動的にグローバル・リソースの逐次化システムを構築することになります。グローバル・オンライン変更では、複数システムから OLCSTAT データ・セットへのアクセスをシリアライズするのに、グローバル・リソースのシリアライゼーションを利用します。複数の z/OS システムを持つシスプレックス内でグローバル・オンライン変更を使用する場合、MVS parmlib 内に GRSRNL=EXCLUDE パラメーターを定義してはいけません。GRSRNL=EXCLUDE を定義すると、OLCSTAT データ・セットのシリアライゼーションがグローバルではなく、ローカルとなります。
 - RM を使用せずに実行する IMSplex (CSL グローバル・パラメーター RMENV=N および OLC=GLOBAL を指定) の場合、別の IMS サブシステムが、RM を使用しない IMS サブシステムからの OLCSTAT データ・セットを使用しないようにする必要があります。
2. IMSplex 内の各 IMS について、以下のステップを実行してください。
- a. MODSTAT DD および MODSTAT2 DD ステートメントを IMS 制御領域 JCL から除去する。
 - b. グローバル・オンライン変更に関連する、または DFSDFxxx メンバーの CG セクション内の DFSCGxxx IMS.PROCLIB メンバー・パラメーターを定義する。

OLC=GLOBAL 指定し、グローバル・オンライン変更を使用可能にする。

OLCSTAT= をオンライン変更状況データ・セット名とともに指定する。IMSplex 内のすべての IMS システムは、同じ OLCSTAT データ・セットを指定する必要があります。
 - c. IMS をシャットダウンする。
 - d. IMS のコールド・スタート

グローバル・オンライン変更の開始

INITIATE OLC (オンライン変更の開始) コマンドは、グローバル・オンライン変更をサポートするために提供されます。ここで、オンライン変更は、IMSplex 内の IMS システムにわたって調整されます。INITIATE OLC は、IMSplex 全体のグローバル・オンライン変更に適用される点を除いて、/MODIFY PREPARE および /MODIFY COMMIT に似ています。OM は、INITIATE OLC コマンドを IMSplex 内の 1 つの IMS に送信します。

構文、環境、および使用に関する詳細については、「IMS V15 コマンド 第 1 巻: IMS コマンド A-M」を参照してください。

注: このコマンドは、RM を使用しない (RMENV=N) 環境で実行すると、このコマンドを処理する IMS 上でローカル・オンライン変更を行います。

INITIATE OLC は、IMSplex 全体にわたる IMS リソースのグローバル・オンライン変更を準備またはコミットします。

ローカル・オンライン変更が使用可能なとき、INITIATE OLC はサポートされません。コマンドが送られる IMS がグローバル・オンライン変更をサポートしていない場合、INITIATE OLC PHASE(PREPARE) コマンドはリジェクトされます。INITIATE OLC PHASE(PREPARE) コマンドがリジェクトされ、グローバル・オンライン変更をサポートする IMS が存在する場合、グローバル・オンライン変更をサポートする IMS にコマンドを送ってください。

正しいオンライン変更コマンドの順序は、次のとおりです。

1. INITIATE OLC PHASE(PREPARE)
2. /DISPLAY MODIFY COMMIT フェーズを試みる前に、/DISPLAY MODIFY コマンドを使用して、変更または削除されるリソースに関する進行中の作業を表示してください。COMMIT が成功した場合、IMS がダウンしている間にグローバル・オンライン変更が起こらない限り、変更がすべての IMS 再始動にわたって存続します。
3. INITIATE OLC PHASE(COMMIT)

INITIATE OLC PHASE(PREPARE) が、FRCABND または FRCCNRML キーワードなしで指定されていて、かつ、オンライン変更がコミットされる前に 1 つ以上の IMS がダウンしているか、ダウンしたためにコマンドが失敗した場合、オンライン変更を打ち切ってやり直す必要があります。オンライン変更を打ち切るには、TERMINATE OLC コマンドを出します。

INITIATE OLC PHASE(PREPARE) が FRCNRML キーワード付きで指定されていて、コマンドがいずれかの IMS に対して失敗した場合、準備が失敗したこれらの IMS システムをシャットダウンした後、INITIATE OLC PHASE(COMMIT) コマンドを使用して先に進むことができます。それ以外の場合は、オンライン変更を打ち切り、やり直す必要があります。

INITIATE OLC PHASE(PREPARE) が FRCABND キーワード付きで指定されていて、IMS に対してコマンドが失敗した場合、準備が失敗した IMS システムをキャンセルした後で、INITIATE OLC PHASE(COMMIT) コマンドを使用して先に進むことができます。または、オンライン変更を打ち切り、やり直してください。

OLCSTAT データ・セットが更新される前に、INITIATE OLC PHASE(COMMIT) コマンドがいずれかの IMS に対して失敗した場合、エラーを訂正してコミットを再試行するか、あるいはTERMINATE OLC コマンドを使用して、オンライン変更を打ち切ることができます。

OLCSTAT データ・セットが更新された後で、いずれかの IMS に対して INITIATE OLC PHASE(COMMIT) コマンドが失敗した場合は、エラーを訂正しコミットを再試行することができます。オンライン変更は打ち切ることはできません。

IMS がオンライン変更中に異常終了し、INITIATE OLC PHASE(PREPARE) コマンドが FRCABND 付きで指定されなかった場合、TERMINATE OLC コマンドを発行し、オンライン変更を打ち切ってください。このケースでは、INITIATE OLC PHASE(COMMIT) コマンドは許可されません。オンライン変更中に IMS が異常終了し、INITIATE OLC PHASE(PREPARE) コマンドが FRCABND 付きで指定された場合、INITIATE OLC PHASE(COMMIT) コマンドは許可されます。

コマンドがリソースを変更する場合、OM インターフェースからのタイプ 1 コマンドおよび タイプ 2 コマンドは、コマンドがリソースを変更する場合、コミット・フェーズ中にリジェクトされます。リソースを変更するコマンドは、そのリソースのオンライン変更を妨害する可能性があります。コマンド (例えば、/DISPLAY または QUERY) がリソースを表示する場合は、OM インターフェースからのタイプ 1 コマンドおよび タイプ 2 コマンドは、コミット・フェーズ中に許可されます。システム・コンソールまたは IMS 端末から入力されたタイプ 1 コマンドは、オンライン変更コミット・フェーズ中にキューに入れられます。これらのコマンドは、オンライン変更がコミットされた後、あるいは打ち切られた後で実行されます。

OM API を通じてのみINITIATE を指定することができます。

INITIATE は、XRF 代替、および FDR システム上では無効です。

グローバル・オンライン変更に関与している各 IMS は、ローカル・オンライン変更の場合のように、マスター端末またはシステム・コンソールに対して、同じ同期オンライン変更メッセージを発行しません。OM コマンド応答には、DFS3499 メッセージの内容のような、ローカル・オンライン変更に対して表示されるオンライン変更メッセージに相当する情報が含まれています。

グローバル・オンライン変更に関与している各 IMS は、DFS3400、DFS3445、および DFS3498 などの非同期オンライン変更メッセージをシステム・コンソールに対して出す場合があります。これらのメッセージの詳細については、「IMS V15 メッセージおよびコード 第 1 巻: DFS メッセージ」を参照してください。

OM コマンドのデフォルト・タイムアウト 300 秒 (5 分) は、オンライン変更フェーズが完了するには十分な時間ではない場合があります。インストールのニーズに基づいて、タイムアウト値をコマンドに指定しなければならない場合があります。


INITIATE OLC 戻りコードと理由コード:

戻りコードおよび理由コードは、INITIATE OLC コマンドによって OM に戻されます。OM から経路指定されるすべての IMS コマンドの戻りコードおよび理由コードは、IMS.SDFSMAC データ・セット内の DFSCMDRR マクロで定義されます。

INITIATE OLC コマンド・マスターは、ほとんどの場合、オンライン変更フェーズをまずローカルで実行します。オンライン変更フェーズがローカルで失敗した場合、コマンド・マスターは、通常、その他の IMS システムへのオンライン変更フェーズの送信をスキップし、オンライン変更フェーズが試行されなかったことを示す完了コードをそれぞれの IMS に設定し、コマンド処理を終了します。ただし、進行中の作業が原因で、INITIATE OLC PHASE(COMMIT) コマンドがローカル IMS で失敗した場合、コマンド・マスターは引き続き他の IMS システムにコミット・フェーズ 1 を送信します。進行中の作業がスムーズに完了するように、IMSplex のすべての IMS システムに対して進行中の作業を報告することが目的です。

混合 IMSplex では、特定のバージョンのオンライン変更をサポートする IMS システムとサポートしない IMS システムが混在していることがあります。コマンドが成功したと考えられるためには、少なくとも IMSplex 内の 1 つの IMS は、オンライン変更フェーズを正常に実行しなければなりません。IMSplex 内のすべての IMS がオンライン変更の同じバージョンをサポートしない場合、コマンドの理由コードは、すべての IMS システムがオンライン変更フェーズを実行しなかったことを示します。IMSplex のいずれの IMS にも適用されない INITIATE OLC PHASE(PREPARE) コマンドのバージョンを入力した場合、TERMINATE OLC コマンドを使用してオンライン変更を終了する必要があります。

関連資料:

 OM へのコマンドの戻りコードおよび理由コード (メッセージおよびコード)

INITIATE OLC コマンドの取り扱いエラー:

INITIATE OLC コマンドは、さまざまなオンライン変更状態における IMSplex 内の 1 つ以上の IMS システムを終わらせるエラーを引き起こします。エラーを訂正してください。QUERY MEMBER および QUERY OLC コマンドを実行し、TERMINATE OLC コマンドを実行してオンライン変更を終了するか、INITIATE OLC コマンドを再び試行するかを決めるのに役立ててください。

グローバル・オンライン変更を試行する前に、QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID) コマンドを実行し、現行の変更 ID を取得してください。INITIATE OLC コマンドが失敗した場合は、QUERY OLC LIBRARY(OLCSTAT) SHOW(MODID) コマンドを再び実行し、変更 ID が同じかどうかを確認してください。変更 ID が 1 つ増えていた場合、オンライン変更は正常に完了したと考えられます。

INITIATE OLC コマンドが失敗した場合、QUERY MEMBER TYPE(IMS) SHOW(STATUS) コマンドを実行し、IMSplex 内のすべての IMS システムのオンライン変更状態を表示してください。QUERY MEMBER TYPE(IMS) SHOW(STATUS) 出力を評価して、何をするかを判断してください。QUERY

MEMBER TYPE(IMS) SHOW(STATUS) 出力および実行するアクションに関する詳細については、593 ページの『IMSplex 内の IMS システムに関する状況および属性情報の表示』を参照してください。

オンラインでの **IMS.ACBLIB** メンバーの変更または追加

IMSplex 環境では、ACB メンバー・オンライン変更 (OLC) 機能を使用して、ACB ライブラリーの個々のメンバーの追加または変更、あるいは ACB ライブラリー全体の変更を行い、さらに IMSplex を静止せずに、またアクティブな ACB ライブラリーのリフレッシュも行わずに、新規メンバーまたは変更したメンバーをオンラインにすることができます。

メンバー OLC の影響を受けないメンバーは静止されません。

ACB ライブラリーにオンライン変更を加えられるのは、以下の IMSplex 環境です。

- 単一の IMS システムで構成される IMSplex。この構成では、IMS は、OM と SCI からなる CSL を持つ OLCSTAT データ・セットを使用する必要があります。RM は必要ではありません。
- 複数の IMS システムで構成される IMSplex。この構成では、各 IMS システムが、同じ共用 OLCSTAT データ・セット、または各自のローカル OLCSTAT データ・セットを使用する必要があります。CSL は OM、SCI、および RM で構成されていなければなりません。リソース構造をお勧めしますが、必須ではありません。

ACBLIB メンバーに対しグローバル・オンライン変更を行えるのは、IMSplex 中のみです。

IMSplex 内で ACBLIB メンバーを共用にするのか、専用にするのかを指定できます。以下に、ACBLIB メンバー・オンライン変更プロセスを OLCSTAT データ・セットを共用しているすべての IMS システム間で調整する例を中心に説明します。

- OM が、コマンド・マスター IMS となる IMS システムを選択します。
- コマンド・マスター IMS が、RM を使って OLCSTAT データ・セットを共用している他の IMS システムとメンバー OLC 処理の調整を行います。
- DFSCGxxx または DFSDFxxx PROCLIB メンバーにおいて、ACBSHR= を Y または N に指定する必要があります。
 - Y は、OLCSTAT 内のすべての IMS システムが同じアクティブおよび非アクティブな ACBLIB を使用していることを示します。
 - N は、OLCSTAT 内の各 IMS が各自専用のアクティブおよび非アクティブな ACBLIB を使用していることを示します。
- OLCSTAT データ・セットを共用している IMSplex 内のすべての IMS システムは、ACBSHR= の値を指定する必要があります。
- メンバー OLC では、共用されているかどうかにかかわらず、すべてのアクティブな ACBLIB が更新されます。

ACB メンバー・オンライン変更では、IMS タイプ 2 コマンドのみ使用されます。ACB ライブラリー・メンバー・オンライン変更の実行に関わるコマンドは、次のとおりです。

- INITIATE OLC PHASE(PREPARE)
- INITIATE OLC PHASE(COMMIT)
- TERMINATE OLC
- QUERY MEMBER TYPE(IMS)
- QUERY OLC SHOW(RSCLIST)

デフォルトで、DBD が INITIATE OLC PHASE(PREPARE) TYPE(ACBMBR) コマンドの NAME キーワードで指定される場合、ACB メンバー・オンライン変更 (OLC) 機能により、関連した PSB および外部参照 DBD がすべて、ステージング ACB ライブラリーからアクティブ ACB ライブラリーに自動的にコピーされます。PSB および DBD の数によっては、コピー・プロセスに必要な時間量が大きい場合があります。

多くの場合、INITIATE OLC PHASE(PREPARE) TYPE(ACBMBR) コマンドで OPTION(NAMEONLY) を指定することによって、ACB メンバー・オンライン変更機能に必要な時間量を大幅に短縮することができます。NAMEONLY オプションは、ACB メンバー・オンライン変更機能の処理を、コマンドの NAME() キーワードで指定されている DBD と PSB のみに制限します。

変更されたリソースの以前のバージョンにフォールバックするには、完全なライブラリー切り替えとともに、完全なオンライン変更プロセスを行ってください。

ACB メンバー・オンライン変更を行うには、次のようにします。

1. ローカル MODSTAT 定義を除去し、DFSCGxxx PROCLIB メンバー内に OLCSTAT データ・セットを定義することにより、オンライン変更機能を使って IMS システムを定義します。
2. PSB 生成を実行して、PSBLIB メンバーを生成または更新します。
3. DBD 生成を実行して、DBDLIB メンバーを生成または更新します。
4. ACB 生成を実行し、ステージング ACB ライブラリー内に実行可能 ACBLIB メンバーを作成します。

ステージング ACB ライブラリーは、アクティブな ACB ライブラリーのクローンかサブセットとすることができます。

5. オプションとして、フォールバック用にアクティブ ACBLIB メンバーのバックアップ・コピーを作成します。これはオンライン変更コピー・ユーティリティ (DFSUOCU0) を使って行えます。
6. TYPE(ACBMBR) パラメーターを指定して INITIATE OLC コマンドを発行して、更新したメンバーまたは新規メンバーをステージング・ライブラリーからアクティブ・ライブラリーにコピーします。

使用している ACB が 64 ビット・ストレージ内にある場合、オンライン変更プロセスによって影響を受けた ACB メンバーは非常駐プールから除去され、64 ビット・プールから削除されます。次回に新規または更新済みの ACB メンバーを必要とするアプリケーションがスケジュールされたとき、それらのメンバーは実行用に ACB ライブラリーから読み取られて非常駐プールにコピーされ、64 ビット・プールにもコピーされます。

変更されたリソースの以前のバージョンにフォールバックするには、ライブラリーの完全なライブラリー切り替えとともに、完全なオンライン変更プロセスを行ってください。

関連資料:

 INITIATE OLC コマンド (コマンド)

IMSplex 内の IMS システムに関する状況および属性情報の表示

QUERY MEMBER コマンドは、オンライン変更状況を含む、IMSplex 内の IMS システムに関する状況または属性情報を表示します。

OM API を通してのみ、QUERY MEMBER を設定することができます。構文、環境、および使用に関する詳細については、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

OLCSTAT データ・セットに関する情報の表示

QUERY OLC LIBRARY コマンドは、OLCSTAT データ・セットの内容に関する情報を提供します。構文、環境、および使用に関する詳細については、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

オンライン変更の終了

TERMINATE OLC コマンドは、IMSplex のグローバル・オンライン変更の打ち切りをサポートします。TERMINATE OLC コマンドは、TERM OLC と短くすることもできます。

グローバル・オンライン変更を打ち切る TERMINATE OLC コマンドは、グローバル・コマンドがグローバル・オンライン変更に参加している IMSplex 内のすべての IMS に適用される点を除いて、/MODIFY ABORT に似ています。

OM は TERMINATE OLC コマンドを IMSplex 内の IMS に送信します。

TERMINATE OLC を使用して、オンライン変更が INITIATE OLC PHASE(COMMIT) と正常にコミットする前に、INITIATE OLC PHASE(PREPARE) によって開始された IMSplex-wide グローバル・オンライン変更を打ち切ることができます。

TERMINATE OLC 使用して、OLCSTAT データ・セットが更新される前に INITIATE OLC PHASE(COMMIT) 障害が起こった後、オンライン変更を打ち切ってください。コミット処理が OLCSTAT データ・セットを更新した後では、オンライン変更は成功したとみなされ、打ち切ることはできません。

ローカル・オンライン変更が使用可能になっている場合は、TERMINATE OLC はサポートされません。コマンドが送られた IMS がグローバル・オンライン変更をサポートしない場合、TERMINATE OLC コマンドはリジェクトされます。そのような事態が起き、グローバル・オンライン変更をサポートする IMS が存在する場合は、ユーザーはグローバル・オンライン変更をサポートする特定の IMS にこのコマンドを経路指定する必要があります。

OM API を通してのみ、TERMINATE OLC を指定することができます。

TERMINATE OLC コマンドは、DB/DC、DBCTL、および DCCTL 環境によって処理することができます。

TERMINATE OLC コマンドは、XRF 代替システムおよび FDR システムでは無効です。

300 秒 (5 分) の OM コマンド・タイムアウト・デフォルトは、オンライン変更フェーズがコンプリートするには十分な時間ではない場合があります。インストールのニーズに基づいて、タイムアウト値をコマンドに指定しなければならない場合があります。

このコマンドのコマンド構文は XML で定義され、OM と通信する自動化プログラムに対して使用可能です。コマンド構文、環境、および使用に関する詳細については、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

グローバル・オンライン変更コマンドの取扱エラー

グローバル・オンライン変更コマンドがエラーを引き起こした場合、実行するアクションは、オンライン変更エラーが発生した後の IMSplex 内の IMS システムの状態によって異なります。

以下の表は、オンライン変更コマンドがエラーを引き起こしたときに実行することができるアクションについて説明しています。

表 39. オンライン変更コマンドの取扱エラー

IMSplex 状態	実行するユーザー処置
オンライン変更コマンド・エラー後の準備失敗の後、1 つ以上の IMS システムが準備完了状態	TERMINATE OLC コマンドを出す。コマンドを再実行する前に問題を訂正する。
すべての IMS システムが準備完了状態	INIT OLC PHASE(COMMIT) または TERMINATE OLC コマンドを実行する。
準備完了状態の IMS システムとコミット・フェーズ 1 完了状態の IMS システムの混在	問題を訂正して、コミットを再試行する、または TERMINATE OLC コマンドを実行する。
OLCSTAT データ・セットが更新される前の、すべての IMS システムがコミット・フェーズ 1 完了状態。	問題を訂正して、コミットを再試行する、または TERMINATE OLC コマンドを実行する。
OLCSTAT データ・セットが更新された後の、すべての IMS システムがコミット・フェーズ 1 完了状態。	問題を訂正し、コミットを再試行する。オンライン変更がコミットされ、打ち切ることはできません。
コミット・フェーズ 1 完了状態の IMS システムとコミット・フェーズ 2 完了状態の IMS システムの混在	問題を訂正し、コミットを再試行する。
すべての IMS システムがコミット・フェーズ 2 完了状態	問題を訂正し、コミットを再試行する。
コミット・フェーズ 2 完了状態の IMS システムとオンライン変更状態ではない IMS システムの混在	問題を訂正し、コミットを再試行する。

グローバル・オンライン変更シナリオ

このトピックのシナリオでは、オンライン変更コマンドの処理中に起こる可能性のあるイベントについて説明し、それらのコマンドに対して、IMS システムがコマンドの現行の状態に基づいてどう応答するかについて説明します。

シナリオ 1: すべての IMS システムが **MODBLKS=OLC** を使用する場合の正常なグローバル・オンライン変更:

このシナリオは、正常なグローバル・オンライン変更コマンドのシーケンスの結果として TSO SPOC に戻ってきた出力を表示します。

1. QUERY OLC LIBRARY(OLCSTAT) コマンドが実行され、ACBLIBA、FMTLIBA、および MODBLKSA といったアクティブ・オンライン変更ライブラリーを表示します。このコマンドはまた、現行の変更 ID が 1 で、OLCSTAT データ・セット名が IMSTESTL.IMS.OLCSTAT であることを表示します。これらのオンライン変更ライブラリーを使用中の IMS システムとして、SYS3、IMS2、および IMS3 があります。これらの IMS システムは、ウォーム・スタートまたは緊急時再始動を実行することが許可されます。
2. INIT OLC PHASE(PREPARE) TYPE(MODBLKS) OPTION (FRCABND,FRCNRML) が実行され、MODBLKS オンライン変更のために IMS システムが準備されます。IMS3 は出力行のすべてを作成するための準備マスターです。FRCABND および FRCNRML オプションは、オンライン変更コマンドが実行されたときに指定されているため、ダウンしている IMS システムはすべてオンライン変更を実行することを許可されています。IMS2、SYS3、および IMS3 は OLCSTAT データ・セットにあるので、オンライン変更に参加します。
3. INIT OLC PHASE(COMMIT) が実行され、オンライン変更がコミットされます。IMS2、IMS3、および SYS3 のすべてがオンライン変更を正常に完了します。コミット・マスター IMS3 は、新規のオンライン変更ライブラリー MODBLKSB、ACBLIBA、および FMTLIBA と、新規の変更 ID 2 を報告します。
4. 別の QRY OLC LIBRARY(OLCSTAT) SHOW(ALL) コマンドが実行され、新規のオンライン変更ライブラリーおよび新規の変更 ID 2 が表示されます。

次の出力例は、このシナリオで説明されている 4 つのコマンドに対する 4 つの応答を示しています。

Response for: QRY OLC LIBRARY(OLCSTAT) SHOW(ALL)

MbrName	CC	Library	ACBLIB	FMTLIB	MODBLKS	Modid	DSName	LastOLC	MbrList
IMS3	0	OLCSTAT	A	A	A	01	IMSTESTL.IMS.OLCSTAT		SYS3,IMS2,IMS3

Response for: INIT OLC PHASE(PREPARE) TYPE(MODBLKS) OPTION(FRCABND,FRCNRML)

MbrName	Member	CC	ACBLIB	FMTLIB	MODBLKS	Modid
IMS3	IMS2	0				
IMS3	SYS3	0				
IMS3	IMS3	0	A	A	A	1

Response for: INIT OLC PHASE(COMMIT)

MbrName	Member	CC	ACBLIB	FMTLIB	MODBLKS	Modid
IMS3	IMS2	0				
IMS3	SYS3	0				
IMS3	IMS3	0	A	A	B	2

Response for: QRY OLC LIBRARY(OLCSTAT) SHOW(ALL)

MbrName	CC	Library	ACBLIB	FMTLIB	MODBLKS	Modid	DSName	LastOLC	MbrList
IMS3	0	OLCSTAT	A	A	B	02	IMSTESTL.IMS.OLCSTAT	MODBLKS	SYS3,IMS2,IMS3

シナリオ 1a: 1 つの IMS が MODBLKS=DYN を使用する場合は正常なグローバル・オンライン変更:

DRD を使用する IMS は、グローバル・オンライン変更に参加しません。グローバル・オンライン変更に参加するこの他の IMS システムで変更しているリソースを変更するには、当該 IMS 上で DRD を使用する必要があります。

このシナリオでは、以下の属性の 4 つの IMS システムを持つ IMSplex に対しグローバル・オンライン変更を行います。

- オンライン変更プロセスを MODBLKS データ・セット内のリソースに対し使用する 3 つの IMS システム。これらは、DFSDFxxx IMS.PROCLIB メンバーの DYNAMIC_RESOURCES セクション内に MODBLKS=OLC を指定しています。
- 動的リソース定義 (DRD) を使用する 1 つの IMS。これは、DFSDFxxx IMS.PROCLIB メンバーの DYNAMIC_RESOURCES セクション内に MODBLKS=DYN を指定しています。

以下のリストに、正常なグローバル・オンライン変更コマンドの順序を示します。

1. QUERY OLC LIBRARY(OLCSTAT) コマンドが実行され、ACBLIBA、FMTLIBA、および MODBLKSA といったアクティブ・オンライン変更ライブラリーを表示します。このコマンドはまた、現行の変更 ID が 1 で、OLCSTAT データ・セット名が IMSTESTL.IMS.OLCSTAT であることを表示します。これらのオンライン変更ライブラリーを使用中の IMS システムとして、SYS3、IMS2、および IMS3 があります。これらの IMS システムは、ウォーム・スタートまたは緊急時再始動を実行することが許可されます。
2. INIT OLC PHASE(PREPARE) TYPE(MODBLKS) OPTION (FRCABND,FRCNRML) が実行され、MODBLKS オンライン変更のために IMS システムが準備されます。IMS3 は出力行のすべてを作成するための準備マスターです。FRCABND および FRCNRML オプションは、オンライン変更コマンドが実行されたときに指定されているため、ダウンしている IMS システムはすべてオンライン変更を実行することを許可されています。IMS2、SYS3、および IMS3 は OLCSTAT データ・セットにあるので、オンライン変更に参加します。
3. INIT OLC PHASE(COMMIT) が実行され、オンライン変更がコミットされます。IMS2、IMS3、および SYS3 のすべてがオンライン変更を正常に完了します。コミット・マスター IMS3 は、新規のオンライン変更ライブラリー MODBLKSB、ACBLIBA、および FMTLIBA と、新規の変更 ID 2 を報告します。
4. 別の QRY OLC LIBRARY(OLCSTAT) SHOW(ALL) コマンドが実行され、新規のオンライン変更ライブラリーおよび新規の変更 ID 2 が表示されます。

次の出力例は、このシナリオで説明されている 4 つのコマンドに対する 4 つの応答を示しています。

```
Response for: QRY OLC LIBRARY(OLCSTAT) SHOW(ALL)
MbrName CC Library ACBLIB FMTLIB MODBLKS Modid DSName LastOLC MbrList
IMS3 0 OLCSTAT A A A 01 IMSTESTL.IMS.OLCSTAT SYS3,IMS2,IMS3
```

```
Response for: INIT OLC PHASE(PREPARE) TYPE(MODBLKS) OPTION(FRCABND,FRCNRML)
MbrName Member CC ACBLIB FMTLIB MODBLKS Modid
IMS3 IMS2 0
IMS3 SYS3 0
IMS3 IMS3 0 A A A 1
```

```
Response for: INIT OLC PHASE(COMMIT)
MbrName Member CC ACBLIB FMTLIB MODBLKS Modid
IMS3 IMS2 0
IMS3 SYS3 0
IMS3 IMS3 0 A A B 2
```

```
Response for: QRY OLC LIBRARY(OLCSTAT) SHOW(ALL)
MbrName CC Library ACBLIB FMTLIB MODBLKS Modid DSName LastOLC MbrList
IMS3 0 OLCSTAT A A B 02 IMSTESTL.IMS.OLCSTAT MODBLKS SYS3,IMS2,IMS3
```

シナリオ 2: **IMS** がダウンしたため、オンライン変更準備コマンドがリジェクトされた:

このオンライン変更シナリオでは、INITIATE OLC PHASE(PREPARE) TYPE(ALL) コマンドを発行し、IMS A に送っています。2 つの IMS システムがダウンしているのに (IMS B が異常終了、IMS C が正常にシャットダウン)、コマンドが発行されたときに FRCABND または FRCNRML オプションが指定されていなかったため、INITIATE OLC PHASE(PREPARE) コマンドはリジェクトされます。

以下の表に、このシナリオの実行中のコマンド・シーケンスおよび戻されるコードを示します。

表 40. シナリオ 2: 失敗したオンライン変更準備コマンドのコマンド・シーケンスおよびコード

アクションまたはイベント	IMS A 完了 コード	IMS B 完了 コード	IMS C 完了 コード	コマンド戻 りコード	コマンド理 由コード
INIT OLC PHASE(PREPARE) TYPE(ALL)	2	B2	B2	C	3004
INIT OLC PHASE(PREPARE) TYPE(ALL) OPTION (FRCABND,FRCNRML)	0	B2	B2	4	100C
INIT OLC PHASE(COMMIT)	0	B2	B2	4	100C
INIT OLC PHASE(PREPARE) TYPE(ALL)	0			0	0
INIT OLC PHASE(COMMIT)	0			0	0

- INITIATE OLC PHASE(PREPARE) TYPE(ALL) コマンドが実行されます。IMS A はコマンド・マスターです。IMS A は、IMS B および IMS C がダウンしていること、および FRCABND または FRCNRML キーワードが設定されていないことを検出するので、コマンドはリジェクトされます。IMS A は、完了コード 2 (未完了) をそれ自体に戻します。IMS A はまた、完了コード B2 (IMS 状態エラー) を IMS B および IMS C を戻します。IMS A は、全般戻りコード C、および全般理由コード 3004 (IMS システムすべてで失敗) を設定します。

2. FRCABND および FRCNRML とともに、別の INITIATE OLC PHASE(PREPARE) TYPE(ALL) コマンドが実行されました。これらのオプションは、IMS システムがダウンしていてもオンライン変更を強制します。IMS A は、再びコマンド・マスターとなります。IMS A は、IMS B および IMS C がダウンしていることを検出しますが、これは許容されると判断されます。IMS A は、IMS B および IMS C に対する B2 完了コード(IMS 状態エラー)を設定します。IMS B および IMS C はダウンしているため、準備フェーズを実行しません。IMS A は準備フェーズをローカルで実行し、それ自身に完了コード 0 を戻します。それから IMS A は全般戻りコード 4 および理由コード100C (受け入れ可能な理由により、複数の IMSシステムで失敗)を設定します。
3. INIT OLC PHASE(COMMIT) コマンドが実行され、コミットが継続します。IMS A は、新規情報とともに OLCSTAT データ・セットを更新します。IMS A には、それ自身のレコードが組み込まれていますが、IMS B および IMS C のレコードは省略されます。IMS B および IMS C には、オンライン変更ライブラリーが有効ではありません。IMS B および IMS C は、最後のオンライン変更 MODBLKS が組み込まれているので、コールド・スタートする必要があります。MODBLKS オンライン変更の間に IMS がダウンしている場合、コールド・スタートする必要があります。
4. 別の INITIATE OLC PHASE(PREPARE) コマンドが実行されます。IMS B および IMS C は、オンライン変更ライブラリーが有効ではないため、参加しません。IMS A のみグローバル・オンライン変更に参加します。
5. INIT OLC PHASE(COMMIT) コマンドが実行されます。IMS B および IMS C は、オンライン変更ライブラリーが有効ではないため、参加しません。IMS A のみグローバル・オンライン変更をコミットします。IMS A はそれ自身に対する完了コード 0、全般戻りコード、および理由コード 0 を設定します。

シナリオ 3: コミット・フェーズ 1 が進行中の作業により失敗:

このオンライン変更シナリオでは、コミット・フェーズ 1 は進行中の作業により失敗します。ただし、INITIATE OLC PHASE(COMMIT) コマンドは最終的にすべての IMS システム上で継続され、結果として 0 の全般戻りコードおよび理由コードとなります。

以下の表に、このシナリオの実行中のコマンド・シーケンスおよび戻されるコードを示します。

表 41. シナリオ 3: 失敗コミット・フェーズ 1 のコマンド・シーケンスおよびコード

アクションまたはイベント	IMS A 完了 コード	IMS B 完了 コード	IMS C 完了 コード	コマンド戻 りコード	コマンド理 由コード
INIT OLC PHASE(PREPARE) TYPE(ALL)	0	0	0	0	0
INIT OLC PHASE(COMMIT)	B1	B1	4	C	3004
INIT OLC PHASE(COMMIT)	B1	4	4	C	3004
/DISPLAY MODIFY およびその他のコマ ンド	なし	なし	なし	なし	なし
INIT OLC PHASE(COMMIT)	0	0	0	0	0

1. INITIATE OLC PHASE(PREPARE) コマンドが実行され、すべての IMS システム上で継続されます。IMS A はコマンド・マスターです。それぞれの IMS は、準備フェーズをローカルで設定し、完了コード 0 を指定します。IMS A は全般戻りコードおよび理由コード 0 を設定します。
2. INITIATE OLC PHASE(COMMIT) コマンドが実行され、IMS A がコマンド・マスターになります。IMS A コミット・フェーズ 1 をローカルで試行しますが、進行中の作業のため、コミット・フェーズは失敗します。IMS A は、それ自身に対して、完了コード B1 (リソース状態) を設定します。IMS A は処理を継続し、コミット・フェーズ 1 を、進行中の作業を報告する必要があるその他の IMS システムに送信します。IMS C は正常にコミット・フェーズ 1 を完了しますが、IMS B は進行中の作業を持っています。IMS A は、IMS C に関する完了コード 4 (未完了) を設定し、この時点でエラーによりコミット処理が終了されます。
3. 別の INITIATE OLC PHASE(COMMIT) コマンドが実行され、IMS A がふたたびマスターになります。IMS A は再びコミット・フェーズ 1 をローカルで試行しますが、進行中の作業のため、失敗します。IMS A はコミット・フェーズ 1 を IMS B および IMS C に送信します。IMS B はコミット・フェーズ 1 を正常に完了します。IMS C は、すでに正しい状態にあることを示す完了コードを戻します。IMS A は、エラーによりコミットを終了し、IMS B および IMS C に関する完了コードを 4 (未完了) に設定します。IMS A は、全般戻りコード C および理由コード 3004 (すべての IMS システムで失敗) を設定します。
4. /DISPLAY MODIFY およびその他のコマンドが実行され、コミットを失敗させる可能性のある、進行中の作業を解決します。
5. 最後のINITIATE OLC PHASE(COMMIT) コマンドが実行されます。すべての IMS システムは、コミット・フェーズ 1、2、および 3 を正常に完了し、それ自体の完了コード 0 を設定します。全般戻りコードおよび理由コードは 0 です。IMS A は、0 という全般戻りコードおよび理由コードを設定します。

シナリオ 4: タイムアウト・エラーにより、オンライン変更準備コマンドが失敗した :

このシナリオでは、オンライン変更準備コマンドは、タイムアウト・エラーによって失敗します。1 つの IMS が準備フェーズを受信しないため、応答を待機している間にコマンドがタイムアウトになります。準備失敗の後でオンライン変更を打ち切る必要があります。

以下の表に、このシナリオの実行中のコマンド・シーケンスおよび戻されるコードを示します。

表 42. シナリオ 4: タイムアウト・エラーによって失敗したオンライン変更準備コマンドのコマンド・シーケンスおよびコード

アクションまたはイベント	IMS A 完了 コード	IMS B 完了 コード	IMS C 完了 コード	コマンド戻 りコード	コマンド理 由コード
INIT OLC PHASE(PREPARE) TYPE(MODBLKS)	0	0	91	C	3000
INIT OLC PHASE(PREPARE) TYPE(MODBLKS)				10	4110

表 42. シナリオ 4: タイムアウト・エラーによって失敗したオンライン変更準備コマンドのコマンド・シーケンスおよびコード (続き)

アクションまたはイベント	IMS A 完了 コード	IMS B 完了 コード	IMS C 完了 コード	コマンド戻 りコード	コマンド理 由コード
INIT OLC PHASE(COMMIT)	2	2	B2	C	3004
TERMINATE OLC	0	0	3	4	100C
INIT OLC PHASE(PREPARE) TYPE(MODBLKS)	0	0	0	0	0
INIT OLC PHASE(COMMIT)	0	0	0	0	0

- このシナリオでは、IMS A はコマンド・マスターで、INITIATE OLC PHASE(PREPARE) TYPE(MODBLKS) コマンドが実行され、IMS A がコマンド・マスターになります。SCI は、IMS C が準備フェーズを受信する前にダウンします。IMS A および IMS B は準備フェーズをローカルで正常に完了し、それ自身に対する完了コード 0 を設定します。RM プロセス・ステップ応答はタイムアウトし、RM はプロセス・ステップの結果を IMS A に戻します。IMS A は、IMS C に対する完了コード 91 (タイムアウト) を設定します。IMS A は、タイムアウト・エラーにより、準備コマンドをリジェクトします。失敗の後で、IMS A および IMS B はオンライン変更準備状態にとどまり、IMS C はオンライン変更状態ではありません。IMS A は、全般戻りコード C および理由コード 3000 (一部の IMS システムで準備が作動) を設定します。IMS C が準備をせず、準備失敗が起こったために、オンライン変更を打ち切る必要があります。

この状態で、QUERY MEMBER TYPE(IMS) を実行し、IMS システムの状態を確認して、何をすべきかを判断してください。

- 別のINIT OLC PHASE(PREPARE) TYPE(MODBLKS) コマンドが実行されます。ただし、準備失敗の後、オンライン変更を打ち切る必要があるため、この準備コマンドも失敗します。コマンド・マスター IMS A は、すでに準備状態にあることを検出し、戻りコード 10 および理由コード 4110 とともにコマンドをリジェクトします。(INIT OLC コマンドは、コマンド・マスターのオンライン変更状態に適用しないので、このコマンドはリジェクトされます。)
- INIT OLC PHASE(COMMIT) コマンドが実行されます。準備失敗の後、誤ってオンライン変更のコミットが試行されます。IMS A は、もう一度コマンド・マスターになります。IMS A はコミット・フェーズ 1 をローカルで実行し、コミット・フェーズ 1 を IMS B および IMS C に送信します。IMS A および IMS B はコミット・フェーズ 1 をローカルで正常に実行します。IMS C は、オンライン変更準備状態ではないので、コミット・フェーズ 1 をリジェクトします。IMS C は、完了コード B2 (IMS 状態エラー) を戻します。IMS A はエラーを検出し、IMS A および IMS B に対して、完了コード 2 (未完了) を設定します。IMS A はまた、完了コード B2 (IMS 状態エラー) を戻します。IMS A はコミット処理を終了します。コマンドは、戻りコード C および理由コード 3004 を伴って失敗します (IMS システムがコマンドを正常に完了せず)。
- TERMINATE OLC コマンドが実行され、グローバル・オンライン変更が打ち切られます。IMS A および IMS B が打ち切りを正常に完了します。IMS C はオンライン変更状態にないので、完了コード 3 を戻します (IMS がすでに、要求されたオンライン変更状態にある)。コマンド・マスター IMS A は、これを

受け入れ可能として扱い、全般戻りコード 4 および理由コード 100C を戻します (正常なのに適用できない IMS システムあり)。

5. INIT OLC PHASE(PREPARE) TYPE(MODBLKS) コマンドが実行されます。準備が継続されます。
6. INIT OLC PHASE(COMMIT) コマンドが実行されます。コミットが継続されます。

コールド・スタートの実行

ある環境の下、例えば、IMS がオンライン変更中に障害を起こした場合には、コールド・スタートの実行が必要になることがあります。コールド・スタートを実行するための処理は、IMS がローカルあるいはグローバル・オンライン変更を使用するかによって異なります。

コールド・スタートに関する一般情報については、「IMS V15 オペレーションおよびオートメーション」の『コールド・スタート』を参照してください。

ローカル・オンライン変更およびコールド・スタート

ローカル・オンライン変更の使用は、すべてのタイプの IMS システム定義に影響します。IMS オンライン・サブシステムは、制御ブロックを IMS.MODBLKS ステージング・ライブラリーに入れます。コールド・スタートを行う前に、MODBLKS ライブラリーを IMS.MODBLKSA または IMS.MODBLKSB のいずれかのデータ・セットにコピーする必要があります。

これらのデータ・セットを非アクティブ・ライブラリーにコピーする場合は、IMS.MODSTAT データ・セットを更新して、IMS.MODBLKS データ・セット DD 名を変更します。アクティブ・ライブラリーにコピーする場合は、問題が生じると、変更前の最後のシステム環境に戻ることはできません。

グローバル・オンライン変更およびコールド・スタート

重要: 典型的な例として、グローバル・オンライン変更中にダウンした IMS は、バックアップするときにコールド・スタートする必要があります。再始動の失敗の原因になったり、後で重大エラーをもたらす可能性があるため、再始動が現行のオンライン変更ライブラリーに対するログ・レコードを処理しないようにするために、コールド・スタートが必要です。

IMSplex 内の最初の IMS が初めてコールド・スタートをする前に、OLCSTAT データ・セットを初期設定する必要があります。OLCSTAT を初期設定するには、グローバル・オンライン変更ユーティリティー (DFSUOLC0) を使用してください。

重要: 誤って有効 OLCSTAT データ・セットの内容を破棄しないよう、注意してグローバル・オンライン変更ユーティリティーを使用してください。有効 OLCSTAT データ・セットの内容を破棄した場合は、OLCSTAT データ・セットが再初期設定されるまで、グローバル・オンライン変更および追加の IMS システムの初期設定は失敗します。

OLCSTAT データ・セットの損失を処理する OLCSTAT データ・セット・リカバリー手順を確立する必要があります。グローバル・オンライン変更が正しく行われるたびに、変更 ID、アクティブ・オンライン変更ライブラリー接尾部、およびオンラ

イン変更ライブラリーと現行の IMS システムのリストを記録してください。
 OLCSTAT データ・セットが破棄された場合は、保管されたデータを使用してグローバル・オンライン変更ユーティリティーの初期設定機能を実行し、OLCSTAT データ・セットを再初期設定してください。

関連資料: グローバル・オンライン変更ユーティリティーに関する詳細については、「IMS V15 システム・ユーティリティー」を参照してください。

いつコールド・スタートを実行するかの判別

IMS が最後のグローバル・オンライン変更時にダウンし、その再始動のタイプが最後のオンライン変更のタイプと矛盾する場合、または IMS が複数のグローバル・オンライン変更でダウンした場合は、コールド・スタートを行います。

最後のグローバル・オンライン変更の間に IMS がダウンしたときは、ウォーム・スタートを実行すると、再始動は最後のグローバル・オンライン変更と競合しません。

特定のオンライン変更タイプに対して許可された再始動コマンドのタイプについての詳細は、601 ページの『コールド・スタートの実行』を参照してください。

表 43. 特定のオンライン変更タイプに対して許可された再始動コマンド

オンライン変更のタイプ	許可された再始動コマンド
ACBLIB	/NRE CHECKPOINT 0 /ERE COLDBASE
ALL	/NRE CHECKPOINT 0
FORMAT	/NRE CHECKPOINT /NRE /ERE /ERE COLDCOMM /ERE COLDBASE
MODBLKS	/NRE CHECKPOINT 0

再始動が失敗した後のクリーンアップ

グローバル・オンライン変更の間に IMS がダウンし、IMS にコールド・スタートが必要なために再始動コマンドがリジェクトされる場合は、コールド・スタートが正しく行われるようにするために 1 つ以上のクリーンアップ・ステップの実行が必要な場合があります。

再始動の試行が失敗した後にクリーンアップを実行するには、以下のようにします。

1. DBRC LIST.RECON コマンドを実行し、IMS サブシステム・レコードまたはオープン OLDS を検査してください。
2. サブシステム・レコードがある場合は、DBRC DELETE.SUBSYS コマンドを実行して、その IMS の DBRC サブシステム・レコードをクリーンアップします。

3. オープンしているすべての OLDS に対して、DBRC GENJCL.CLOSE コマンドを実行します。このコマンドにより、DBRC は RECON データ・セットを読み取り、その IMS に対してどの OLDS をクローズするかを決定できるようになります。

第 28 章 IMS スプール API による出力の印刷

このトピックでは、IMS DL/I スプール・アプリケーション・プログラミング・インターフェース (API) を使用するための設計上および操作上の推奨を示し、IMS アプリケーション・プログラムから拡張印刷機能へのアクセスを増加させるための IMS スプール API の使用法について詳しく説明します。

IMS は、アプリケーション・プログラムが JES と直接インターフェースして、JES スプールに印刷データ・セットを作成できるようにする DL/I アプリケーション・プログラミング・インターフェースの拡張機能を提供します。これらの印刷データ・セットは、アプリケーション・プログラムのニーズを満たすために、印刷マネージャーおよびスプール・サーバー機能で処理できるようになります。

設計および操作上の考慮事項

以下のトピックでは、IMS スプール API の IMS サポートを紹介します。

固有の IMS 端末サポート

IMS は、IMS システム生成プロセスおよび ETO によって IMS 制御領域と関連付けられている一連の端末およびプリンターをサポートします。

ほとんどの環境では、IMS がサポートしている端末に送られるメッセージは、中間記憶装置用の IMS メッセージ・キューに入れられます。端末がメッセージを受け取ることができると、IMS は IMS メッセージ・キューからメッセージを検索して、メッセージを装置に送ります。メッセージの伝送中に VTAM セッションが失われたり、伝送エラーが発生すると、IMS はそのメッセージを IMS メッセージ・キューに戻し、後で伝送できるようにします。IMS メッセージ・キューはリカバリー可能リソースなので、IMS はメッセージ保全性を提供し、IMS メッセージ・キューと関連付けられているメディア障害からリカバリーすることができます。

以下の図は、IMS 環境を単純化した図です。この図は、IMS 制御領域、IMS メッセージ・キュー、およびメッセージ処理領域によって制御されている端末装置を示しています。DL/I プリプロセス・ルーチンと、IMS 制御領域のサービスは、端末とデータベースの両方にサービスを提供します。プリンターに出力メッセージを送信するアプリケーション・プログラムは、DL/I CHNG 呼び出しによってプリンターを識別し、DL/I ISRT 呼び出しによってプリンターへメッセージを挿入することができます。IMS は、これらのメッセージがアプリケーション・プログラムによってコミットされるまで、IMS メッセージ・キューに入れます。呼び出しがコミットされた後、IMS は指定されたプリンターにメッセージを送達します。

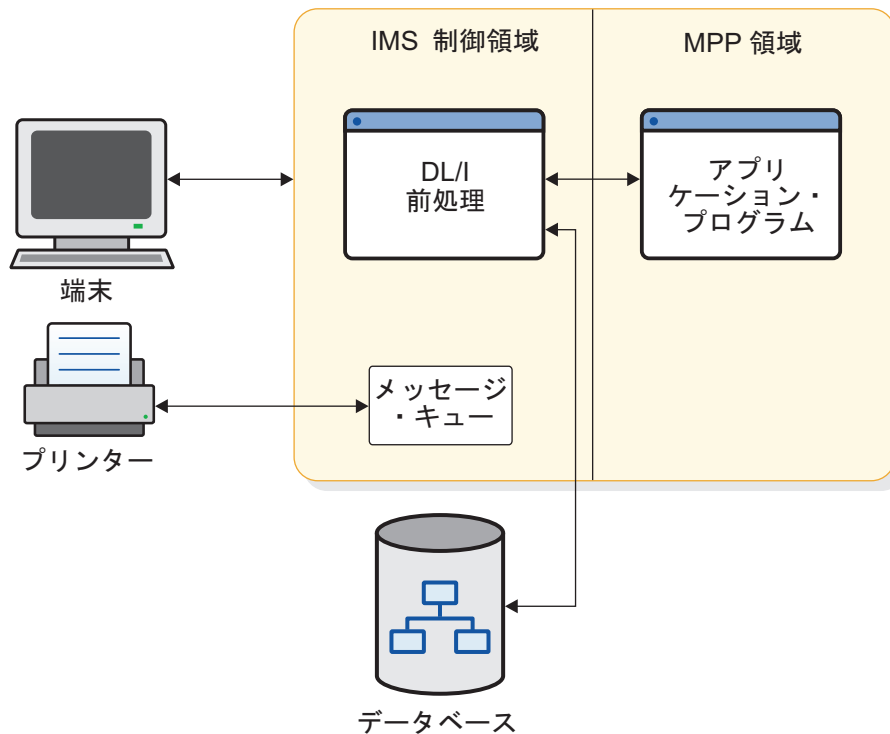


図 54. スプール API なしのIMS 環境

アプリケーション要件

IMS アプリケーション・プログラムは、IMS 固有プリンターや、高品質テキスト、特殊フォント、線、グラフなどを印刷できるプリンターにメッセージを送信できます。これらのプリンター用にデータ・ストリームを作成してプリンターにデータを送信するアプリケーション・プログラムは、ジョブ・アプリケーション・サブシステムの IMS スプール API を使用します。IMS スプール API は、データ・セット・クラス、リモート宛先、サイズ (最大 32 KB のセグメント・サイズが通常使用される)、その他の特性に基づいて、IBM の高機能印刷 (Advanced Function Printing) (AFP) サービスまたは OEM プリント・サーバーに印刷データ・セットを渡します。

IMS アプリケーション領域から JES 印刷データ・セットを作成するアプリケーションには、適正なパフォーマンス、平易なインターフェース、およびプログラム使用法の柔軟性が必要です。IMS スプール API インターフェースの拡張機能は、これらのニーズへの対応を試みるものです。DL/I アプリケーション・インターフェースは、JES 印刷データ・セットの作成をサポートします。これらの JES 印刷データ・セットの作成手法は、メッセージを固有の IMS プリンターに送信するのに使用される手法と類似しています。

IMS、いくつかの z/OS サービスを使用して、IMS スプール API をサポートします。これらの z/OS サービスによって、IMS およびそのアプリケーション・プログラムが以前には扱う必要がなかったパフォーマンスや可用性の考慮事項 (データ・セット・オプションの解析のためのその他のプロセッサ要件から、JES サブシステムによる同期点サポートの欠如に至るまで) が発生するようになります。以下の図に示す IMS スプール API 環境は、これらの問題に対処できる設計になっています。

印刷データ・セットは非常に大きい場合があり、また、メッセージ・キューを中間記憶装置として使用すると、メッセージ保安全性は強化されずに、プロセスのオーバーヘッドが追加されるので、ほとんどの IMS スプール API の実行は、IMS 従属領域で行われます。しかし、この非同期メッセージ手法は、メッセージ送達の問題を発生させる関連アプリケーション・プログラム・エラーの問題も提起します。

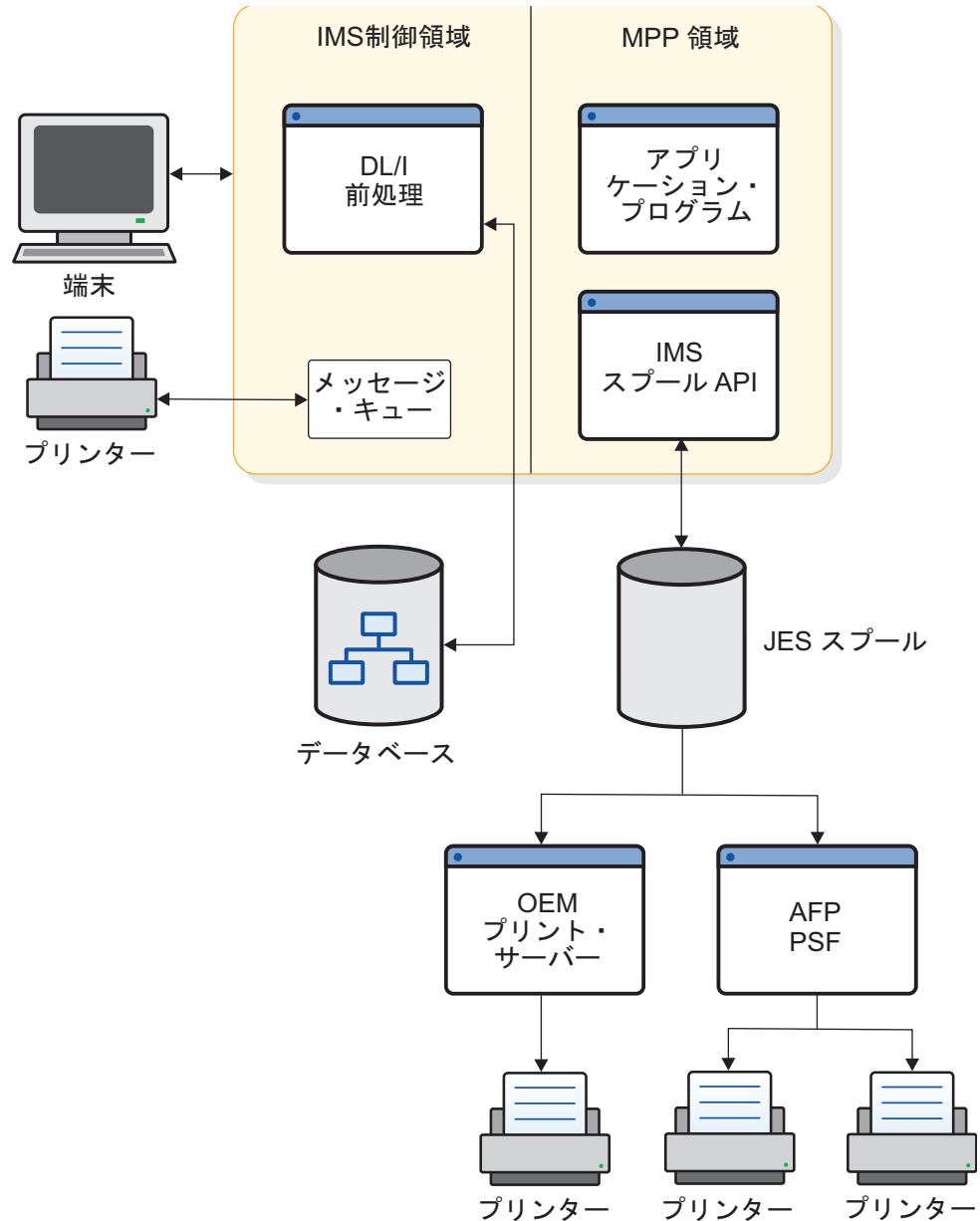


図 55. スプール API を伴う IMS 環境

注: IMS スプール API 上に JES 印刷データ・セットを作成するためのサポートを、IMS システム生成プロセスで UNITYPE=SPOOL として定義されるスプール回線に対する IMS 端末サポートと間違えないようにしてください。スプール回線に対する端末サポートは、JES 印刷データ・セットに対するサポートとはまったく関係ありません。

データ・マネージャーとしての IMS スプール API

IMS スプール API は、z/OS システムでのジョブの入力、実行、および出力を制御します。IMS スプール API は JOB またはタイム・シェアリング (TSO) ユーザー境界にあるので、個々の印刷データ・セットが論理メッセージであり、ジョブ境界の一部でもなければ、ジョブ境界と関係付けられてもいない IMS 環境に対して、JES は特殊な考慮事項を発生させます。

IMS と関連付けられているデータ・マネージャーは、DL/I アプリケーション・プログラミング・インターフェースを提供します。このインターフェースは、データの一部バッファリングに同期点処理を提供します。同期点処理は、作業単位が正常終了した場合には作業単位に対して変更をコミットし、作業単位が異常終了した場合には変更をバックアウトします。IMS スプール API は同期点処理を使用しませんが、異常終了した作業単位によって行われた変更の影響をバックアウトすることができます。

IMS スプール API は、IMS Transaction Manager や IMS Database Manager が行うようなデータのリカバリーは行いません。IMS スプール API はスプール・データ・セットに変更を記録しないので、スプール装置がデータ・エラーを検出すると、スプール上のデータが再始動時に失われることがあります。

印刷データ・セットの特性

JCL ステートメントに SYSOUT= を指定して印刷データ・セットを割り振るか、または SVC 99 に対する MVS 動的割り振りサポートを使用して、印刷データ・セットを動的に割り振ることによって、JES 印刷データ・セットを作成できます。

印刷データ・セットが割り振られるときに、印刷データ・セット特性を定義するか、またはそのデフォルトを使用することができます。

これらの印刷データ・セット特性の例としては、FORMS、COPIES、DESTINATION、WRITER などがあります。これらの出力特性のいくつかを DD カードに指定することも、OUTPUT JCL ステートメントで定義されている一連の印刷データ・セット特性と印刷データ・セットを関連付けることもできます。JES は、動的出力または SVC 99 プロセスとして知られているシステム・サービスを使用して、これらの印刷データ・セット記述子を動的に定義する方法を提供します。動的出力プロセスを動的割り振りとともに使用すると、システム出力または SYSOUT データ・セットを、動的に作成された一連の OUTPUT 記述子と関連付けることができます。

次の 3 つの方法を使用して、IMS スプール API の印刷データ・セット記述子を作成できます。それぞれについて、以下のトピックで説明します。

関連資料: 印刷データ・セットの動的割り振りのための z/OS サービスの詳細については、「z/OS MVS Programming: Authorized Assembler Services Guide」を参照してください。

変更 (CHNG) 呼び出し

ほとんどのアプリケーション・プログラムでは、変更 (CHNG) 呼び出しインターフェースを使用して印刷データ・セット記述子を提供できます。IMS スプール API は、この既存のテクノロジーを使用します。

CHNG 呼び出しは、IMS スプール API に対して 2 つのオプション・パラメーターをサポートします。

- オプション・リスト・パラメーターは、オプションのリストを含んでいる文字ストリングをポイントします。IAFP キーワードは、CHNG 呼び出しを IMS スプール API サービスに対する要求として識別します。
- フィードバック域パラメーターは、オプション・リストに誤りが見つかったときに、IMS がエラー情報をアプリケーション・プログラムに返すのに使用できるエリアを指しています。

アプリケーションが、同じ印刷オプションで複数のデータ・セットを作成する場合や、入力待ち (WFI) である場合は、SETO 呼び出しを使用して、解析の影響を少なくしてください。

CHNG 呼び出しは、次のいずれかの方法でデータ・セット特性を提供します。

- この呼び出しは、PRTO オプションを使用することにより、データ・セット・オプションを直接提供することができます。

CHNG 呼び出しで PRTO オプションが使用されると、IMS は z/OS サービス (SJF) をアクティブにして印刷オプションを検証し、動的出力のために z/OS サービスを呼び出して、印刷データ・セットの割り振り時に使用される出力記述子を作成します。これは、アプリケーション・プログラムが印刷データ・セット特性を提供する最も簡単な方法です。PRTO オプションを使用して CHNG 呼び出しを行うたびに解析が発生するので、PRTO オプションの使用は、ほとんどのオーバーヘッドに影響を及ぼします。

複数の CHNG 呼び出しにわたって事前作成されているテキスト単位を使用しても、アプリケーションのパフォーマンスがさほど改善されない場合には、PRTO オプションを使用してください。

- この呼び出しは、TXTU オプションを使用して、動的記述子のために事前作成されている一連のテキスト単位を参照できます。

テキスト単位を再利用できるアプリケーション・プログラムは、TXTU オプションを使用することにより、より良いパフォーマンスを達成できます。アプリケーション・プログラムが動的出力に必要なテキスト単位を管理できる場合は、TXTU オプションを使用して、多くの印刷データ・セットに対する解析を回避してください。

アプリケーション・プログラムは、アプリケーション域内にテキスト単位を必要な形式で作成し、それらのテキスト単位を IMS に渡します。

アプリケーション・プログラムは、SETO 呼び出しによって、IMS に印刷オプションを提供し、テキスト単位作成用の作業域を提供できます。IMS は TXTU オプションを使用して解析とテキスト単位の作成を行い、SETO 呼び出し成功後

の動的出力に必要なテキスト単位が、渡された作業域に含まれるようにすることができます。この作業域にはアドレス依存型情報が含まれているので、再配置はしないでください。

- この呼び出しは、OUTN オプションを使用することにより、従属領域の JCL の OUTPUT JCL ステートメントを参照できます。

これは、アプリケーション・プログラムで使用できる場合に、印刷データ・セット情報を提供する簡単な方法です。出力 JCL ステートメントは、CHNG 呼び出し時にオプション・リストの OUTN オプションによって参照されます。OUTN オプションが使用されると、IMS は動的割り振り時に出力 JCL ステートメントを参照し、その結果、IMS スプール API は、出力 JCL ステートメントから印刷データ・セット特性を取得します。

オプションの設定 (SETO) 呼び出し

オプションの設定 (SETO) 呼び出しを使用して、印刷データ・セット記述子を作成できます。SETO 呼び出しは、入力待ち (WFI) 実行を持つアプリケーション・プログラムには特に便利です。SETO 呼び出しは、データ・セットに対して解析および出力記述子のテキスト単位の作成を行うのに必要なオーバーヘッドを減少させます。

アプリケーション・プログラムが、プログラムのスケジューリング中に一連の記述子を再使用する必要がある場合は、アプリケーション・プログラムは SETO 呼び出しによって、IMS スプール API に印刷データ・セット特性を提供します。SETO 呼び出しは出力オプションを解析し、アプリケーション・プログラムによって与えられた作業域に動的出力テキスト単位を作成します。アプリケーション・プログラムに事前作成されているテキスト単位が与えられると、これらのテキスト単位は、TXTU パラメーターを用いた CHNG 呼び出しで使用されます。解析およびテキスト単位作成のオーバーヘッドを受けることなく、データ・セットに対して印刷特性を提供できます。

他のプログラミング技法を使用してテキスト単位を事前作成できる場合は、SETO オプションを使用してテキスト単位を事前作成する必要はありません。

出力 DD ステートメント

出力 JCL ステートメントを従属領域 JCL に追加することにより、印刷データ・セット記述子を提供できます。アプリケーション・プログラムは、OUTN パラメーターを指定した CHNG 呼び出しを使用して、従属領域 JCL の出力 JCL ステートメントの名前を渡すことができます。印刷データ・セットが動的に割り振られると、IMS は出力 JCL ステートメントを使用できます。

出力ステートメントが従属領域の JCL に存在しない場合は、最初の挿入が行われるときに、動的割り振りは失敗します。アプリケーション・プログラムは、挿入 (ISRT) が失敗したことを示す状況コード AX を受け取ります。

IMS スプール API へのデータの書き込み

挿入 (ISRT) 呼び出しまたは削除 (PURG) 呼び出しを使用して、IMS スプール API にデータを書き込むことができます。

ROLL 呼び出しまたは ROLB 呼び出しを使用して、印刷データ・セットをバックアウトすることができます。

印刷データ・セットのバックアウトには、SETS 呼び出しまたは ROLS 呼び出しを使用しないでください。これらの呼び出しは、IMS スプール API によってサポートされていません。

挿入呼び出し

標準の挿入 (ISRT) 呼び出しは、BSAM を使用して IMS スプール API にデータを書き込みます。スプールに書き込めるデータの長さは、最大 32760 バイトなので、BSAM 書き込みはアプリケーション・プログラムのバッファ領域から直接行われません。

IMS は、ユーザーの入出力域が 16 MB 境界よりも上にあることを検出すると、BSAM 書き込みを行う前に、それより下の作業域にユーザーのアプリケーション・プログラム・データを移動します。

アプリケーション域がその境界よりも下にすでにある場合は、入出力域から直接書き込みを行えます。できれば、入出力域は 16 MB 境界より下に置くべきです。ただし、入出力域をそれより下に置く必要があることをパフォーマンスが示していないければ、その必要はありません。アプリケーション・プログラムのバッファを最初に移動しないで書き込むことにより、IMS は大きなデータ・ブロックの移動に関連する問題の発生を防止します。BSAM を使用する場合は、BSAM ブロック記述子ワード (BDW) が含まれるように、入出力域の形式を変更してください。

関連資料: 入出力域形式についての詳細は、「IMS V15 アプリケーション・プログラミング」を参照してください。

PURG 呼び出し

高速代替 PCB とともに削除 (PURG) 呼び出しを使用して、印刷データ・セットを IMS スプール API に解放し、即時に印刷できるようにします。PURG 呼び出しが発行されると、印刷データ・セットはクローズされ、割り振り解除されます。EXPRESS=YES とマークされていない代替 PCB に対して PURG 呼び出しが発行されると、印刷データ・セットに対しては何のアクションも行われません。

入出力域を使用してもしなくても、PURG 呼び出しを発行できます。

- 入出力域を使用して PURG 呼び出しが発行されると、2 つの呼び出しであるかのように処理されます。最初の機能は PURG 要求を処理し、2 番目の機能は入出力域によって提供されたデータを挿入します。
- EXPRESS=YES とマークされていない代替 PCB に対して PURG 呼び出しが発行されると、削除機能は無視されます。入出力域が含まれていると、データは印刷データ・セット内に置かれます。

PURG 呼び出しは、EXPRESS=YES として生成されている代替 PCB とともに使用すると、IMS スプール API と多少異なります。

- 現行の印刷データ・セットはクローズされ、割り振り解除されて、印刷のために JES に送信されます。

- 代替 PCB には有効な宛先が含まれていないことを示す状況コード A3 がアプリケーション・プログラムに返されます。

PURG 呼び出しの結果については、以下の表を参照してください。

表 44. PURG 呼び出しの結果

PURG 呼び出しの		
タイプ	高速 PCB	非高速 PCB
入出力域を使用する	印刷データ・セットは、JES による印刷のために解放されます。PURG 呼び出しは、状況コード A3 を受け取りません。	PURG 呼び出しは、ISRT 呼び出しと同様に機能します。
入出力域を使用しない	印刷データ・セットは、JES による印刷のために解放されます。PURG 呼び出しは、ブランクの状況コードを受け取ります。	アクションはとられません。

ROLL および ROLB 呼び出し

ROLL および ROLB 呼び出しは、IMS スプール API 機能を使用して、アプリケーション・プログラムによって発行できます。これらの呼び出しの結果、高速代替 PCB に対する PURG 呼び出しによって IMS スプール API に解放されなかった印刷データ・セットのバックアウト (データ・セット削除) が行われます。

SETS、SETU、および ROLS 呼び出し

IMS スプール API は、SETS、SETU、または ROLS 呼び出しをサポートしません。これらの呼び出しを発行しても、IMS スプール API は何のアクションも行わず、SETS、SETU、または ROLS 呼び出しがアプリケーション・プログラムによって発行されたことを示す特別な状況コードも返されません。

SETS、SETU、および ROLS 呼び出しの使用には、制約事項はありません。これらの呼び出しは、印刷データ・セットのプロセス以外でアプリケーション・プログラムが使用できるからです。

特殊な考慮事項 — 許可される記述子

IMS スプール API によって許可される出力記述子は、パラメーターの妥当性検査が行われるときに IMS が実行されている z/OS システムの TSO OUTDES コマンドでサポートされる出力記述子です。

関連資料: 出力記述子について詳しくは、z/OS TSO/E コマンド解説書を参照してください。

IMS スプール API で使用する初期設定パラメーターは、印刷データ・セット特性を定義するのに使用するパラメーターに影響を与えます。

印刷データ・セットの制御

アプリケーション・プログラムは、IMS がそのアプリケーション・プログラムのために作成する IMS スプール API ファイルを管理しています。この管理は、アプリ

ケーション・プログラムによって生成されたデータ・ストリームにも同様に適用されます。IMS はこのデータの影響を受けないからです。

データ・ストリーム自体の中に制御情報を組み込むこともでき、アプリケーション・プログラム DL/I CHNG 呼び出しで IMS スプール API ファイル・オプションを指定することもできます。これらのプロセス・オプションは、データ・セット・ベースで指定され、TSO OUTDES ステートメントに指定されます。

IMS スプール API は、出力スケジューリングに基づいて印刷データを送ります。一般に、スケジューリングは、出力クラス (CLASS パラメーター)、プリント出力先 (DEST パラメーター)、FORM、および PRMODE によって決定されます。

IMS は透過送達サービスです。IMS は、データ・セット処理オプションに基づいて IMS スプール API ファイルを動的に作成し、そのファイルにメッセージを書き込んで、IMS スプール API データ・セット・サーバー (例えば PSF) で処理できるように割り振り解除します。

印刷データ・セットの後処理が適切に行われるようにするために、例えば次のような状況下では z/OS のオペレーターが介入する必要があります。

- IMS 緊急時再始動
- 従属領域の異常終了
- 印刷データ・セットに対する動的割り振り解除の失敗

このような状況の下では、z/OS オペレーターは、未確定のメッセージを示す IMS スプール API データ・セットを適切に処理しなければなりません。アプリケーション・プログラムがこれらのデータ・セットの適切な後処理を要求すると、は通知メッセージを表示して、z/OS オペレーターを援助します。

インストール・システムにおける印刷不能データ・セットの操作手順を検討してください。多くのインストール・システムでは、所定の日数後に印刷データ・セットを削除するジョブを行います。これらのジョブによって、印刷しなければならない IMS 印刷データ・セットが削除されてしまうことがないことを確認してください。

高速 代替 PCB

IMS スプール API は、高速代替 PCB の使用をサポートします。EXPRESS=YES を指定して生成された代替 PCB に対して削除 (PURG) 呼び出しを発行すると、印刷データ・セットはクローズされて、割り振りは解除されます。これは、IMS 作業単位の終了前の JES プロセスに有効です。

関連資料: 高速代替 PCB を使用した PURG 呼び出しの使い方については、611 ページの『PURG 呼び出し』を参照してください。

XRF 環境

IMS スプール API は、特殊なサポートを持たない XRF システム内で動作します。XRF のテークオーバー中に IMS スプール API は、障害と正常リスタートが発生したかのような影響を受けます。テークオーバー発生時に部分的な印刷データ・セットが存在し、主システムで IMS 異常終了プロセスが実行されない場合は、その部分的な印刷データ・セットを、IMS スプール API が印刷のために使用できるようにすることができます。

関連資料: これらの部分的な印刷データ・セットのための適切なプロシージャーについては、「IMS V15 アプリケーション・プログラミング」を参照してください。

割り振りエラーについて

IMS S スプール API インターフェースが印刷データ・セットを動的に割り振るのは、データが実際にデータ・セットに挿入された後だけです。これによって、オーバーヘッドが軽減され、異常終了トランザクションに対する終結処置が簡略化されます。

CHNG または SETO 呼び出しで提供された誤ったデータ・セット印刷オプションによって発生した動的割り振り中に、エラーが発生することもあります。データ・セット印刷オプションは、CHNG および SETO 呼び出しの処理中に解析できません。宛先の妥当性検査は、動的割り振り中にのみ行えます。

最初の挿入がデータ・セットに対して行われるときに、印刷オプションのいずれかが誤っていて、動的割り振りが失敗すると、ISRT 呼び出しが状況コード AX を受け取ります。IMS スプール API コードは、以下のものをアプリケーション・プログラムに提供します。

- 状況コード
- エラー・メッセージ DFS0013E
- 診断ログ・レコード (67D0)

エラー・メッセージは、アクティブにされるサービスのタイプと、そのエラーの原因を示す戻りコードおよび理由コードを示します。例えば、よく発生する障害は、理由コード 046C (リモート・ワークステーションがジョブ入力サブシステムに対し定義されていない) によって示されます。IMSTEMP の宛先が JES に定義されていない場合に、無効な宛先を選択したり、保全性オプション 2 (選択不能な宛先) を選択すると、この理由コードが表示されます。IMS が印刷データ・セットの割り振りを解除するときに、呼び出しの宛先名パラメーターで無効な宛先を指定すると、動的割り振りエラーが発生します。

エラー・メッセージによって示されるサービスには、次のようなものがあります。

- DYN** z/OS 動的割り振り (SVC99)
- OPN** z/OS データ・セットのオープン
- OUT** z/OS 動的出力記述子の作成 (SVC109)
- UNA** z/OS 動的割り振り解除 (SVC99)
- WRT** z/OS BSAM 書き込み

関連資料: サービスが動的割り振りまたは動的割り振り解除用である場合、あるいは動的出力記述子作成である場合は、「z/OS MVS Programming: Authorized Assembler Services Guide」を参照してください。

第 29 章 IMS 内から外部システムへのアクセス

IMS アプリケーション・プログラムは、外部システムにアクセスして、それらの外部システム上に存在するデータを使用したり、IMS 環境の外部に存在するビジネス・ロジックと対話したりできます。

外部サブシステム・データを使用する IMS アプリケーション・プログラムは、以下のいずれか、または両方の機能を使用できます。

- 外部サブシステム接続機能 (ESAF)。この機能は、JMP、JBP、MPP、BMP、および IFP アプリケーション・プログラムで使用でき、Db2 for z/OS データベースや IBM MQ などの外部サブシステム・データにアクセスできます。
- DB2 リカバリー可能リソース・マネージャー・サービス接続機能 (DB2RRMS)。この機能は、Db2 for z/OS データベースにアクセスする JMP および JBP アプリケーションで使用できます。

1 つの SSM PROCLIB メンバー内で、ESAF と DB2RRMS の両方のアクセスを指定できます。

IMS 環境の外部に存在するビジネス・ロジックとの対話を必要とする IMS アプリケーション・プログラムは、IMS コールアウト機能を使用して、それを実現できます。それらのコールアウト要求は、同期でも非同期でもかまいません。コールアウト要求のターゲットには、以下のものが可能です。

- IMS Enterprise Suite SOAP Gateway を通過した後の、任意の汎用 Web サービス。
- IMS TM Resource Adapter を通過した後の、メッセージ・ドリブン Bean、Enterprise JavaBeans コンポーネント、Java EE アプリケーション、または Web サービス。
- TCP/IP および IMS Connect プロトコルを使用してコールアウト要求を検索する、ユーザー作成の IMS Connect TCP/IP アプリケーションまたはベンダー提供のソリューション。

関連資料:

- 外部サブシステムのインストールと IMS への定義の詳細については、以下を参照してください。
 - 「IMS V15 コミュニケーションおよびコネクション」の『複数の外部サブシステムへのアクセス』。
 - 「IMS V15 システム定義」の『EXEC パラメーター SSM= の指定』。
- コールアウト機能のソフトウェア要件の詳細については、「IMS V15 リリース計画」を参照してください。アプリケーション・プログラムでコールアウト機能を使用する方法の詳細については、「IMS V15 アプリケーション・プログラミング」を参照してください。

第 3 部 DBRC 管理

このトピックでは、IMS データベース・リカバリー管理機能 (DBRC) の概要を紹介し、IMS と DBRC の間の相互作用について説明します。

第 30 章 DBRC の概要

IMS データベースのリカバリーをより容易にする IMS Database Manager の機能である DBRC は、データベース・リカバリーに必要な情報の保守、リカバリー制御ステートメントの生成、リカバリー入力の検証、データベース・データ・セットごとに別個の変更ログの保守、および複数の IMS サブシステムによる IMS データベースおよびエリアの共用のサポートを行います。

DBRC は IMS に不可欠の部分です。IMS は、多数の項目の情報を記録および管理に関して DBRC に依存しています。DBRC はこの情報を一連の VSAM データ・セットの中に保持します。これらのデータ・セットを総称して、リカバリー管理 (RECON) データ・セットと呼ばれ、RECON データ・セットの情報に基づき、進め方に関して IMS にアドバイスします。

具体的に、DBRC は次の項目を実行します。

- ログイン・プロセスに関連する情報を記録および管理して、IMS システムとデータベースの保全性を確保できるように支援する。
- 再始動にどのログを使用するかを IMS に通知して、IMS の再始動処理を支援する。
- データベース許可情報を記録および管理して、IMS がデータ共用環境内のデータベースへのアクセスを許可または禁止できるように支援する。
- 次のことを実行してデータベースとログのリカバリーを可能にする。
 - ログ内の情報の使用と保全性を制御する。
 - RECON データ・セット内のデータベースとログの情報を記録および保守する。
 - さまざまな IMS ユーティリティー・プログラム用のジョブ制御言語 (JCL) を生成および検査する。
- サブシステムが拡張回復機能 (XRF) に対応するかどうかを (RECON データ・セット内で) 識別することにより、XRF をサポートする。
- 以下によって IMSplex をサポートする。
 - いずれかの DBRC が RECON データ・セットの再構成を行っているときに、そのことを同じ DBRC 共用グループ内の同じ IMSplex 内にあるすべての DBRC に通知する。
 - RECON データ・セットへの並列アクセスを許可する。

DBRC は、IMS オンライン・サブシステムの再始動およびリカバリーに必要なログ・データを管理する上でも重要な役割を果たします。

以下のような多くの IMS 構成で、DBRC が必要です。

- オンライン構成、すなわち、DB/DC、DCCTL、または DBCTL
- IMSplex などのデータ共用環境
- 拡張回復機能 (XRF) を使用する構成

重要: DBRC は IMS バッチ・ジョブおよび一部のオフライン・ユーティリティーでは必要ありません。ただし、登録済みデータベースをアクセスするバッチ・ジョブおよびユーティリティーが DBRC なしでの実行を許可された場合には、データベースのリカバリー可能性と保全性が失われる恐れがあります。ご使用の構成で DBRC を使用する必要がない場合 (例えば、データを共有しないバッチ環境など) でも、DBRC を使用してリカバリーを監視し、データベースを保護すれば、リカバリー処理を単純化することができます。

関連資料:

- 「IMS V15 オペレーションおよびオートメーション」では、DBRC を使用する場合と使用しない場合のリカバリー手順について詳しく説明しています。
- 635 ページの『第 33 章 DBRC はリカバリーにどのように役立つか』で、DBRC がデータベース・リカバリーにどのように役立つかについてさらに詳しく説明しています。

関連概念:

- 8 ページの『データベース・リカバリー管理機能』
- 129 ページの『OLDS の特性の変更』
- 134 ページの『データベース・リカバリー管理機能およびリカバリー』

DBRC コンポーネント

DBRC には、RECON データ・セット、データベース・リカバリー管理ユーティリティー (DSPURX00)、および骨組み JCL が組み込まれています。

RECON データ・セット (RECON data set)

DBRC は RECON データ・セットにリカバリー関連情報を保管します。

推奨事項: DBRC をインストールするときは、RECON データ・セットとして 3 つの VSAM KSDS を定義してください。最初の 2 つのデータ・セットはアクティブ・データ・セットで、3 番目のデータ・セットはスペアです。2 番目のアクティブ・データ・セットは 1 番目のアクティブ・データ・セットのコピーです。ほとんどの場合、この 2 つのアクティブ・データ・セットは、単一のデータ・セット (RECON データ・セット) と見なすことができます。

関連資料: これらのデータ・セットの詳細については、691 ページの『第 42 章 RECON データ・セットの初期化および保守』に説明があります。

データベース・リカバリー管理ユーティリティー (DSPURX00)

DBRC コマンドを実行するには、データベース・リカバリー管理ユーティリティーを使用します。

DBRC コマンドを使用すれば、以下のタスクをすべて実行することができます。

- RECON データ・セット内の情報をリストする
- RECON データ・セット内の情報を更新する
- RECON データ・セット内の情報を使用して、IMS ユーティリティー用のジョブを生成する

骨組み JCL

DBRC は、一部のリカバリー・ユーティリティ用の入力データを生成するため、入力モデル (またはテンプレート) として区分データ・セット (PDS) のメンバーを使用します。これらの PDS メンバーは IMS と一緒に配布され、骨組み JCL と呼ばれます。


DBRC は、骨組み JCL、RECON データ・セットからの情報、および GENJCL コマンドからの命令を使用して、一部のリカバリー・ユーティリティを正しく実行するために必要な JCL と制御ステートメントを生成します。骨組み JCL は、ご使用のシステムのシステム構成を反映するように変更してください。

関連資料: 骨組み JCL の詳細については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

IMS は、骨組み JCL ステートメントが入っている PDS メンバーを提供します。これらの PDS メンバーは、基幹 JCL 実行メンバーと呼ばれ、IMS.SDFSISRC ターゲット・ライブラリーに入っています。

IMS Installation Verification Program (IVP) は、骨組み JCL 実行メンバーをカスタマイズし、カスタマイズされたメンバーを IMS PROCLIB ライブラリーに配置します。DBRC は、これらの IMS PROCLIB メンバーを使用して、IMS V15 データベース・ユーティリティ にリストされている IMS ユーティリティ用のジョブ (JCL および制御ステートメント) を生成します。その中には JOB ステートメントを生成する骨組み JCL 実行メンバーの JOBJCL もあります。

関連資料:

 データベース・リカバリー管理ユーティリティ (DSPURX00) (システム・ユーティリティ)

DBRC タスク

DBRC によって自動的に実行されるいくつかのタスクと、ユーザーが DBRC に対するコマンドまたは要求によって開始する少数の追加タスクが存在します。

DBRC は自動的に以下のタスクを実行します。

- IMS のログに関する情報を記録および管理する
- RECON データ・セット内のリカバリー情報を記録する
- データベース・ユーティリティに正しい入力データがあるかどうかを検査する
- DBRC に登録されたデータベースのリカバリーを制御する
- 共用データベースの制御とシリアライゼーションに関するアクセス許可情報を制御する

DBRC にコマンドまたは要求を渡して開始すると、以下の作業およびこれ以外の作業を実行できます。

- DBRC アプリケーション・プログラミング・インターフェース (API) を開始または停止する
- RECON データ・セット内のリカバリー情報を記録する

- さまざまな IMS ユーティリティー用の JCL を生成し、ユーザー定義の出力を生成する
- RECON データ・セット内の一般情報をリストする
- RECON データ・セットから特定の情報を収集する

関連資料:

- DBRC のロギング・サポートの追加情報については、625 ページの『第 31 章 ログ情報の記録および制御』を参照してください。
- DBRC のデータ共用サポートの追加情報については、641 ページの『第 35 章 データ共用のサポート』を参照してください。
- DBRC コマンドの詳細情報については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。
- DBRC API 要求の詳細情報については、「IMS V15 システム・プログラミング API」を参照してください。

DBRC コマンドおよび API 要求

DBRC からのサービスを取得するには、DBRC コマンドまたは DBRC アプリケーション・プログラミング・インターフェース (API) 要求を使用します。

DBRC コマンド

RECON データ・セット内の情報のリストを生成し、RECON データ・セット内の情報の追加、変更、および削除を行い、データベース・リカバリーの際に使用する IMS ユーティリティーの実行に必要な JCL と制御ステートメントを生成するには、DBRC コマンドを使用します。

以下に、DBRC バッチ・コマンドのリストを各コマンドの用途とともに示します。

- BACKUP.RECON コマンドは、RECON データ・セットのバックアップ・コピーを作成する際に使用します。
- CLEANUP.RECON コマンドは、古いか有効期限が切れたりリカバリー関連情報およびログ情報を、RECON データ・セットから削除する際に使用します。
- CHANGE コマンドは、RECON データ・セット内の情報を変更する際に使用します。
- DELETE コマンドは、RECON データ・セットから情報を削除する際に使用します。
- GENJCL コマンドは、各種の IMS リカバリー・ユーティリティー用のジョブを生成する際に使用します。
- INIT コマンドは、RECON データ・セットに次のような変更を加える際に使用します。
 - DB グループを作成する
 - DBDS グループを作成する
 - リカバリー・グループを作成する
 - データベースを登録する
 - サポートされているイメージ・コピー・ユーティリティーをあとで実行するときに使用できる、イメージ・コピー・レコードを作成する。

- あとで変更累積ユーティリティーで使用できる変更累積データ・セットを作成する。
- 高速機能エリアに属するエリア・データ・セット (ADS) を定義する項目を作成する。
- CA グループを作成する
- DBDS または DEDB エリアを登録する
- HALDB 区画を登録する
- RECON データ・セットを初期化する
- LIST コマンドは、RECON データ・セットのすべての、または選択された部分のフォーマット済み印刷出力を作成する際に使用します。
- NOTIFY コマンドは、RECON データ・セットに、通常はそこに自動的に書き込まれる情報を追加する際に使用します。

DBRC コマンドは、データベース・リカバリー管理ユーティリティー (DSPURX00) を実行するための JCL にそれらを組み込むことにより発行できます。このユーティリティーは、バッチ環境で、または TSO フォアグラウンド・プログラムとして実行している間に、上記のコマンドを処理できます。

DBRC コマンド (LIST コマンドは例外として) を COMMAND API 要求のパラメーターとして発行することもできます (IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンドを参照)。

一部の DBRC コマンドは、オンライン IMS 環境から実行することができます (/RMxxxxxx コマンド)。オンライン DBRC コマンドには、次のものがあります。


- /RMCHANGE
- /RMDELETE
- /RMGENJCL
- /RMINIT
- /RMLIST
- /RMNOTIFY

DBRC API 要求

DBRC API 要求は、DBRC からサービスを受けるために使用します。

関連資料: DBRC アプリケーション・プログラミング・インターフェースと DBRC API 要求の詳細については、「IMS V15 システム・プログラミング API」を参照してください。

関連資料:

 [DBRC コマンド \(コマンド\)](#)

第 31 章 ログ情報の記録および制御

IMS は、その作動中、常に IMS ログに自らのアクティビティを記録します。DBRC の初期化を完了すると、DBRC はロギング・アクティビティの情報の記録および制御により、RECON データ・セット内の IMS ロギング・プロセスに参入します。

IMS ログには以下の情報が含まれます。

- IMS の始動とシャットダウン
- データベースに行われた変更
- 受信されたトランザクション要求と送信された応答
- アプリケーション・プログラムの初期化と終了
- アプリケーション・プログラムのチェックポイント
- IMS システム・チェックポイント

IMS ログは、次のデータ・セットで構成されています。

- 先行書き込みデータ・セット (WADS)
- オンライン・ログ・データ・セット (OLDS)
- システム・ログ・データ・セット (SLDS)
- リカバリー・ログ・データ・セット (RLDS)

以下の図では、IMS が生成するログ・データ・セットと再始動データ・セット (RDS)、および DBRC が作成し、保守する RECON データ・セットを示します。

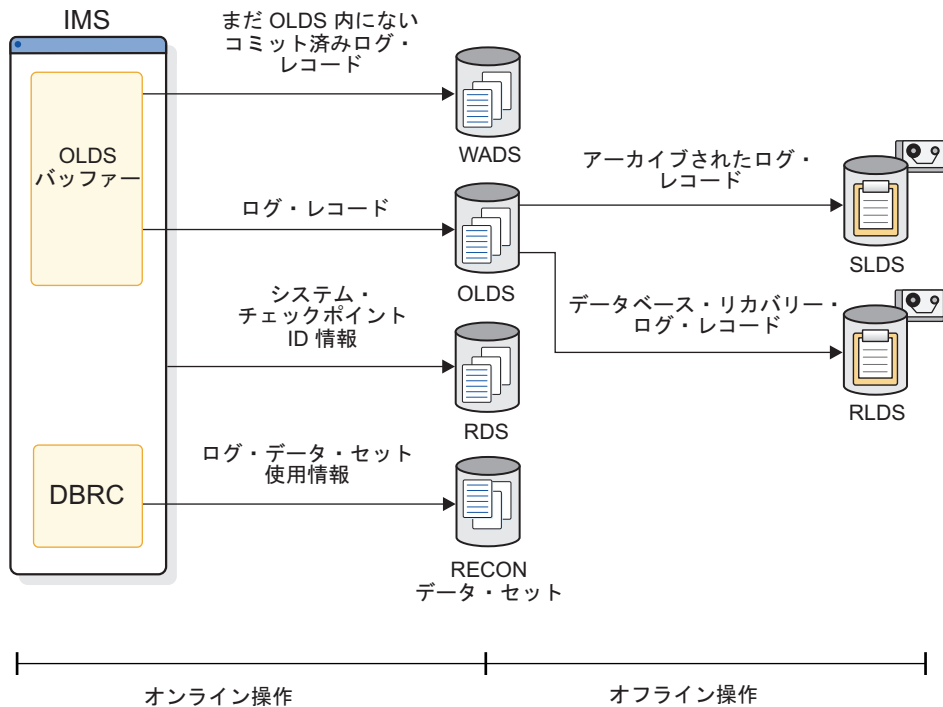


図 56. リカバリーと再始動に関して生成されるログ

関連資料:

- 108 ページの『ロギング』 (IMS ロギング・プロセスの詳細な解説の場合)
- 628 ページの『DBRC ログ関連のコマンド』

DBRC の初期化を完了すると、DBRC はロギング・アクティビティーの情報の記録および制御により、IMS ロギング・プロセスに参入します。この情報は、RECON データ・セットに記録されます。データベース・データ・セット (DBDS) のリカバリーを DBRC に制御させる場合は、その DBDS を DBRC に登録する必要があります。

DBRC は次の項目を含む、多数の項目を自動的に RECON に記録します。

- ログ・データ・セットの情報
- データベース・データ・セットの情報
- 次のようなイベントの情報
 - データベース更新
 - データベース許可
 - データベース・イメージ・コピーの作成
 - データベースの再編成
 - データベースのリカバリー
 - OLDS のアーカイブおよび対応する SLDS と RLDS の作成
 - ログ・リカバリー・ユーティリティーの実行
 - サブシステムのサインオン

IMS は上記の情報を、自らの再始動とデータベース・リカバリー・ジョブ (データベースが DBRC に登録済みの場合) のために使用します。さらに DBRC は、OLDS のアーカイブ要件を追跡し、要求があれば、ジョブをアーカイブするための JCL を生成して実行依頼します。

また、DBRC は、付加されたすべてのサブシステムにリカバリー単位の管理も提供します。DBRC は、バッチ・バックアウト、動的バックアウト、部分バックアウト、および再始動のために、このようなリカバリー単位に関する情報を提供します。

バッチ・システムによって生成されたログの場合は、DBRC を使用する必要はありません。バッチ・ジョブで DBRC を使用する利点は、DBRC が、バッチ・ジョブで作成されるすべてのログ・データ・セットを記録し、ダミーまたはヌルのログ・データ・セットが指定された場合にバッチ更新ジョブが実行されないようにする点です。

重要: 登録済みデータベースが DBRC による制御なしに更新された場合、DBRC はそのデータベースのリカバリーを正しく制御することができず、データベースの安全性が危険にさらされることがあります。

RECON データ・セットのログ・レコードの変更

RECON データ・セットのログ・レコード内の OLDS、RLDS、または SLDS に関する情報を変更するには、CHANGE.PRILOG または CHANGE.SECLOG コマンドを使用します。

データ・セットにエラーが発生した時点を示すとき、またはボリューム通し番号が変更されたことを示すときは、このログ・レコードを更新してください。通常は、これらのコマンドを使用する必要はありません。

OLDS のアーカイブ

ログ保存ユーティリティ (DFSUARCO) を実行して OLDS を SLDS にアーカイブし、IMS が OLDS を再利用できるようにします。アーカイブすべき頻度は、サブシステムの負荷および OLDS に書き込まれるログ・レコード数によって異なります。

ログ保存ユーティリティは常に SLDS を生成します。この SLDS には、データベース・リカバリーおよびオンライン IMS 再始動の両方に必要なすべてのログ・レコードが入っています。

ログ保存ユーティリティに、SLDS に加えて RLDS も生成するように要求することができます。RLDS には、データベース・リカバリーに必要なログ・レコードしか入っていません。

RLDS を要求すると、出力 RLDS データ・セットに関する情報は RECON データ・セット内の PRILOG レコードに記録され、SLDS データ・セットに関する情報は PRISLD レコードに記録されます。RLDS を要求しない場合は、SLDS データ・セットに関する同じ情報が PRILOG レコードと PRISLD レコードの両方に記録されます。

2 次 OLDS が存在するか、または単一の OLDS から重複ログを作成するように要求すると、2 次ログ出力に関する情報が対応する SECLOG レコードと SECSLD レコードに記録されます。

重要: IMS バッチ・ジョブからの出力であるログ・データ・セットは厳密に言えば SLDS ですが、そのログ・データ・セットに関する情報は PRILOG レコードと SECLOG レコードに記録されます。

GENJCL.ARCHIVE コマンドを実行してログ保存ユーティリティを実行します。これによって、DBRC は満杯の OLDS を判別し、適切な JCL を生成します。

関連資料: ARCHDEF ステートメントおよび自動アーカイブの詳細については、「IMS V15 システム定義」のメンバー DFSDFxxx を参照してください。

推奨事項: 自動アーカイブを使用するか自分でアーカイブを呼び出すかに関係なく、アーカイブ・ジョブは必ず、できるだけ早く実行してください。オンライン・サブシステムは、OLDS がアーカイブされた後でのみ、これを再利用します。アーカイブ・ジョブが実行されず、すべての OLDS が満杯になる場合、オンライン・サブシステムは待機します。アーカイブ・ジョブを迅速に実行する方法の 1 つは、かなり高い優先順位で実行されるイニシエーター使用し、他の多数のユーザーが使わないものを使用することです。これにより、アーカイブ・ジョブが内部読み取りプログラムのキューにあまり長く留まらないようにすることができます。

DBRC が RECON データ・セット内の OLDS にエラーのマークを付けた場合は、GENJCL 機能はその OLDS のアーカイブを実行依頼しません。1 対の OLDS の一方が破壊されているか使用できない場合は、RECON データ・セット内でその OLDS にエラーのマークを付けるようにすることができます。

以下に、ログ・レコードのアーカイブ処理についての詳細を知ることができる参照先を示します。

関連資料:

- ログ・レコードの自動、手動およびカスタム・アーカイブの詳細については、「IMS V15 オペレーションおよびオートメーション」を参照してください。
- エントリー・ポイントの指定およびログ保存ユーティリティの実行の詳細については、「IMS V15 システム・ユーティリティ」を参照してください。
- ログ保存およびロガー出口ルーチンの詳細については、「IMS V15 出口ルーチン」を参照してください。

DBRC ログ関連のコマンド

DBRC ログ関連機能を実行するコマンドは、操作手順において使用できます。

以下のコマンドは、DBRC ログ関連機能を実行します。

- CHANGE.PRILOG
- CHANGE.RECON
- CHANGE.SECLOG
- DELETE.LOG
- GENJCL.ARCHIVE

- GENJCL.CLOSE
- LIST.LOG
- NOTIFY.PRILOG
- NOTIFY.SECLOG

LIST.LOG コマンドに加え、ログ照会または OLDS 照会 API 要求を使用して RECON データ・セットからログ関連情報を検索することもできます。COMMAND API 要求を使用して、RECON データ・セット内の情報の追加、変更、および削除を行うこともできます。

関連資料: DBRC コマンドについて詳しくは、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

第 32 章 データベースのリカバリー

IMS データベースのリカバリー処理には、次の 3 つの基本ステップが含まれます。ただし、処理の詳細はリカバリーされるデータベースのタイプによって異なります。

1. データベースを最新のイメージ・コピーに復元する。

重要: HALDB オンライン再編成を使用している場合は、すべてのデータベース変更がログ・データ・セット上に存在するので、このステップをスキップしてください。

2. ログ・データ・セット (または変更累積データ・セット) を使用して、イメージ・コピーが作成された後にデータベースに加えられた変更を復元する。

定義: 変更累積とは、リカバリーに関連しないレコードを除去し、単一セグメントに対する複数の変更を 1 つの変更としてマージすることにより、1 つ以上の IMS ログ・データ・セットの圧縮バージョンを作成する処理のことです。変更累積の詳細については、671 ページの『ログ・レコードの変更累積』を参照してください。

3. 不完全な変更はいずれもバックアウトする。

リカバリーが正常に完了すると、DBRC はリカバリーに関する情報を RECON データ・セットに記録します。タイム・スタンプ・リカバリーを実行した場合、DBRC は省略された変更の範囲に関する情報を記録します。

要件: データベース割り振り間隔中にタイム・スタンプ・リカバリーを実行した場合は、即時にイメージ・コピーを作成して、それ以降のリカバリーの試みのために有効な開始点を確保する必要があります。こうすると、DBRC は、後続のリカバリーで変更が再適用されるのを防ぎます。

以下の図では、単純なデータベース・リカバリーを示します。



図 57. IMS データベース・リカバリー・プロセス

データベース・リカバリーの情報は、以下のソースのいずれか、またはすべてによりもたらされます。

- データベースのイメージ・コピー
- データベース再編成データ・セット
- ログ・データ・セット (SLDS および RLDS)

- 変更累積データ・セット

DBRC を使用してこれらの情報源をすべて追跡することができるため、データベース・リカバリーのタスクが大幅に単純化されます。

関連資料: リカバリー処理の詳細については、103 ページの『リカバリー作業の理解』を参照してください。

RECON データ・セットにリカバリー可能データベースを登録する場合、DBRC は、データベース変更レコードを含むログ・データ・セットとデータベースとの関連を記録します。

DBRC はさらに次の情報を記録します。

- データベース・イメージ・コピー
- 再編成 (DEDB オンライン再編成を除く)
- リカバリー
- 変更累積
- バックアウト

DBRC は、この情報を RECON に記録するので、データベース・リカバリー実行用の JCL を生成することができます。ユーザーが GENJCL コマンドを使用して JCL を生成するか、自分で JCL を用意するかに関係なく、DBRC は RECON データ・セット内の情報を使用して入力に必要なデータ・セットを正確に判別します。データベース・リカバリー・ユーティリティーは、JCL が正しいことを DBRC が検証した場合にのみ実行されます。

タイム・スタンプ・リカバリーを実行することによって、ユーザーは入力から一定時間後にログに記録された変更をすべて省略することができます。タイム・スタンプ・リカバリーとは、データベースから省略された変更をバックアウトするのと同じことです。

ほとんどのタイム・スタンプ・リカバリーでは、正常に実行するために DBRC が必要です。タイム・スタンプ・リカバリーの要求で DBRC を使用すると、DBRC は正しいログを選択し、実行時には、ユーザーの要求を正しく実行するためには入力の処理をどこで停止すればよいかについて、データベース・リカバリー・ユーティリティーと連絡を取ります。

以下の図では DBRC がデータベース・リカバリー・ユーティリティーとどのように連動するかを示します。

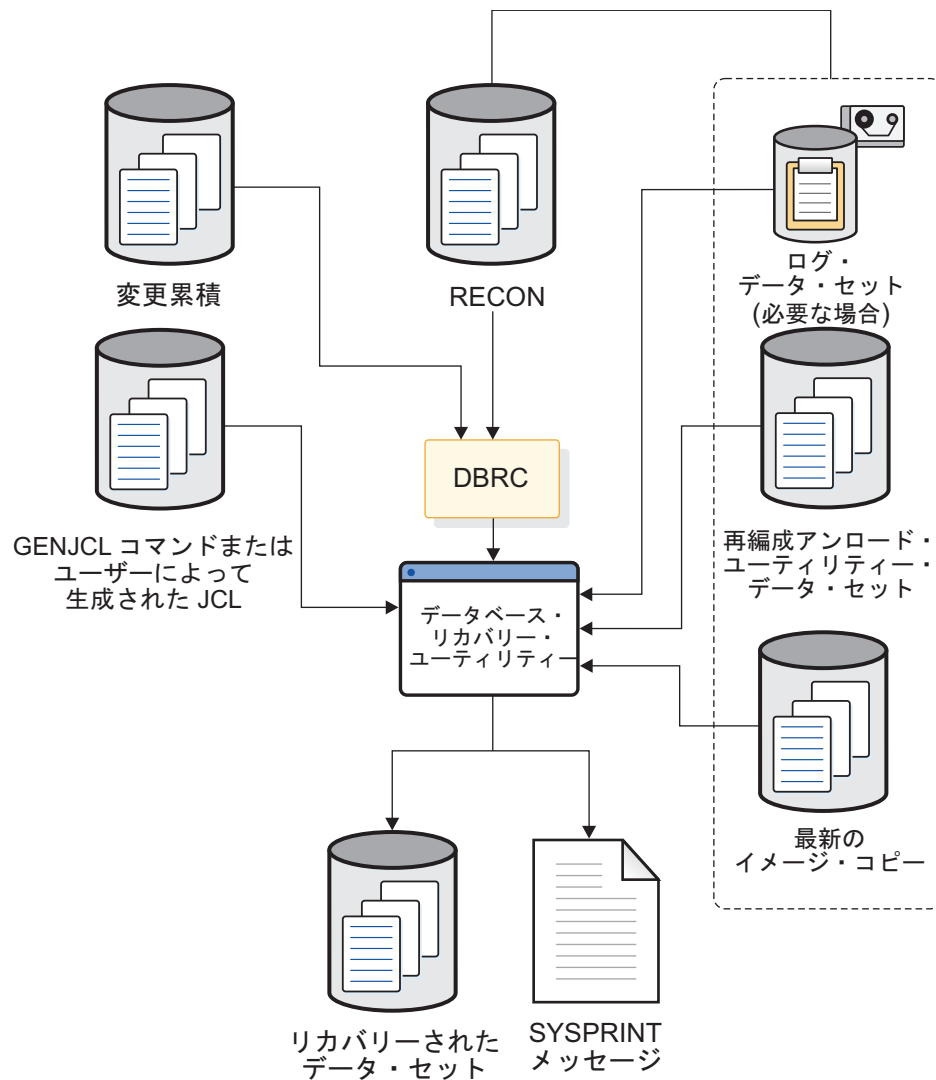


図 58. DBRC はデータベース・リカバリー・ユーティリティとどのように連動するか

推奨事項: DBRC は段階的に実装し、最初は、ごく一部のリカバリー可能データベースのみを RECON データ・セットに定義してください。これにより、DBRC の使用経験を積むことによって、バックアップ、リカバリー、および操作手順で必要な変更の評価、作成、およびテストを行う機会が得られます。

第 33 章 DBRC はリカバリーにどのように役立つか

DBRC はリカバリー・ツールおよびユーティリティーによって呼び出され、データベース・リカバリーに必要なリソースの情報を提供します。これらのリソースに、イメージ・コピー、ログ、変更累積、およびデータベース・データ・セットに関する情報を組み込むことができます。

リカバリー JCL の生成

GENJCL.RECOV コマンドまたは /RMGENJCL コマンドを使用して、登録済みデータベース・データ・セットのリカバリーを行うために必要な JCL を生成できます。

RECON データ・セットに記録された情報を使用して、DBRC は次のことを実行します。

1. 最新のイメージ・コピーのロードに使用するイメージ・コピー・データ・セットを選択する
2. イメージ・コピーが作成された後にログに記録されたすべての変更を適用するための入力となる変更累積とログ・データ・セットを選択する

変更累積入力が必要 (データ共用のために) であるにもかかわらずそれが存在しないか使用できない場合、DBRC はその事実をユーザーに通知し、GENJCL.RECOV コマンドは失敗します。

GENJCL.USER コマンドはユーザー定義の出力を生成でき、その出力に JCL を組み込むことができます。GENJCL.USER コマンドをサポートする骨組み JCL 実行メンバーは提供されていません。GENJCL.USER コマンドを入力する場合は、そのコマンドをサポートするメンバーを用意する必要があります。

GENJCL コマンドまたは /RMGENJCL コマンドを発行した場合、DBRC は骨組み JCL を読み取り、RECON データ・セット内に記録されている情報に基づいてシンボリック・パラメーターを実際の値に置き換え、適切な JCL を構築します。例えば、データベースをリカバリーする JCL を生成するように DBRC に要求すると、DBRC はライブラリーから骨組み JCL メンバーを検索し、必要であれば、最新のイメージ・コピー、変更累積、およびログ・データ・セットの情報をを用いて JCL を完成します。データベース処理の JCL を DBRC が生成するためには、そのデータベースが登録されている必要があります。

GENJCL コマンドを使用してリカバリーに必要な JCL と制御ステートメントを生成すると、データベースのリカバリーに要する時間と労力を大幅に削減できます。また、GENJCL コマンドを使用して、多くのリカバリー・エラーの原因を取り除くこともできます。これまではデータベースのリカバリーの際に、どの入力データ・セットをどのような順序でデータベース・リカバリー・ユーティリティーに渡すべきかを判別するのに、多くの時間を費やす可能性がありました。

変更累積データ・セットまたは PRILOG レコード (RECON データ・セット内の) データ・セットが使用可能な場合は、DBRC は、SLDS ではなくこれらをリカバリー用に選択します。これによって、データベース変更累積を定期的に行う場合には、データベース・リカバリーがより迅速に行えるようになります。DBRC は、どのログ・データ・セットが必要であるかを認識し、IMS が確実にすべてのボリュームを正しい順序で処理するようにします。また、DBRC はデータベース・リカバリー用に最新のイメージ・コピーの選択も行います。

DBRC は、可能であれば常に変更累積データ・セットを使用して、データベース・リカバリー・ユーティリティーに最適の入力データを選択します。データベース変更累積ユーティリティーを使用していない場合、またはそのユーティリティーが一部のログ・データ・セットを処理しなかった場合、DBRC は、RLDS、SLDS、または RLDS と SLDS の両方の項目が入っている可能性がある、PRILOG (または SECLOG) レコードから必要なログ・データ・セットを選択します。

関連資料:

- IMS PROCLIB メンバー、DBRC プロシージャー、JCLOUT および JCLPDS DD ステートメントの調整アクションについての詳細は、「IMS V15 システム定義」を参照してください。
- ユーザー独自の骨組み JCL のカスタマイズおよび IMS が提供する骨組み JCL の内容の詳細については、IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンドを参照してください。

推奨事項: 高速処理データベース (DEDB) の可用性を向上するために、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、オンライン・エリアで使用できる追加のコピーを用意してください。このユーティリティーはリカバリー用のバックアップ・コピーは提供しません。DEDB エリア・データ・セット作成ユーティリティーは、その入力の一部として RECON データ・セットを使用します。

ユーティリティー JCL の妥当性検査

リカバリー・ユーティリティーが登録済みデータベース・データ・セットを処理する場合、このユーティリティーはその入力を妥当性検査のために DBRC に提示します。ユーザーあるいは DBRC のいずれがリカバリー JCL を作成したかに関係なく、DBRC はユーティリティーへの入力 JCL が正しいかどうかを (RECON データ・セットの現在の情報に従って) 検査します。GENJCL コマンドで JCL を作成した場合であっても、割り込んできたイベントがユーティリティーの実行前に入力 JCL を無効にしてしまう可能性もあります。

DBRC は以下の IMS ユーティリティーおよびサービスによって呼び出され、入力を検証し、その結果を記録します。

- 索引/ILDS 再作成ユーティリティー (DFSPREC0)
- データベース・イメージ・コピー・ユーティリティー (DFSUDMP0)
- データベース・イメージ・コピー 2 ユーティリティー (DFSUDMT0)
- 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)
- データベース変更累積ユーティリティー (DFSUCUM0)
- バッチ・バックアウト・ユーティリティー (DFSBB000)

- データベース・リカバリー・ユーティリティ (DFSURDB0)
- ログ・リカバリー・ユーティリティ (DFSULTR0)
- ログ保存ユーティリティ (DFSUARC0)
- HALDB オンライン再編成
- HD 再編成アンロード・ユーティリティ (DFSURGU0)
- HD 再編成再ロード・ユーティリティ (DFSURGL0)
- HISAM 再編成アンロード・ユーティリティ (DFSURUL0)
- HISAM 再編成再ロード・ユーティリティ (DFSURRL0)
- データベース接頭部更新ユーティリティ (DFSURGP0)
- DEDB エリア・データ・セット作成ユーティリティ (DBFUMRI0)
- /RECOVER コマンド

以下の図では、上記のユーティリティを実行する際の DBRC の役割を示します。

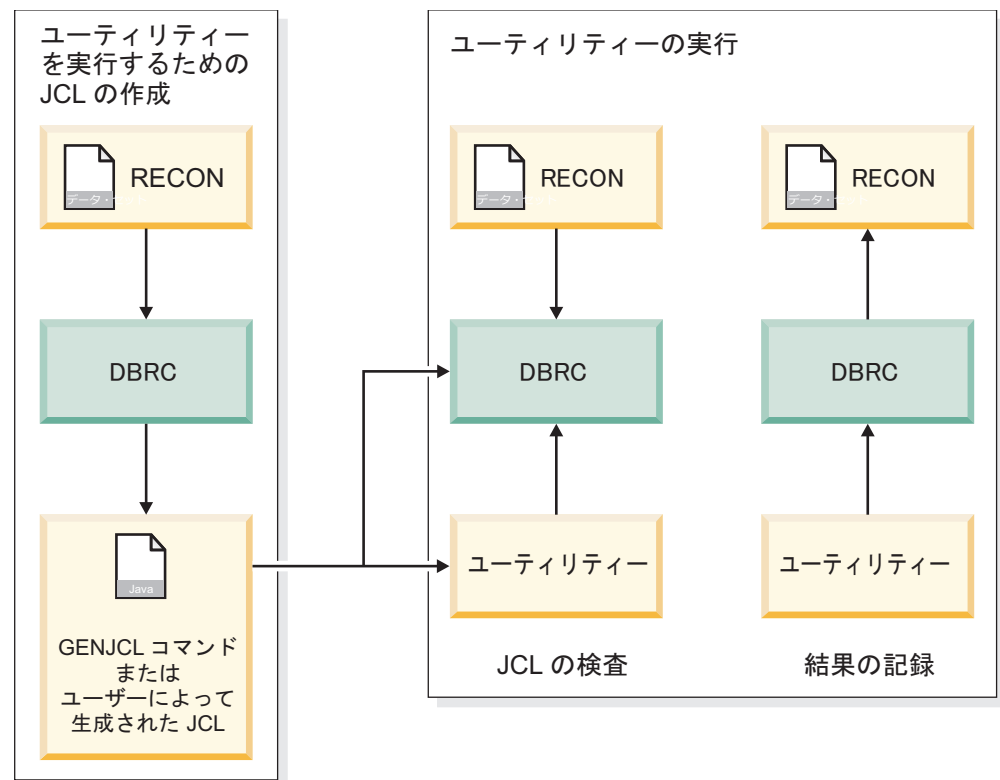


図 59. ユーティリティの実行における DBRC の役割

関連資料: 詳しくは、以下を参照してください。

- IMS データベース・ユーティリティについては、「IMS V15 データベース・ユーティリティ」を参照してください。
- IMS ログ・ユーティリティについては、「IMS V15 システム・ユーティリティ」を参照してください。
- RECON データ・セット内に BACKOUT レコードを手動で作成する NOTIFY.BKOUT コマンドについては、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

- /RECOVER コマンドについては、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

バッチ・バックアウト・ユーティリティー (DFSBB000) を実行すると、DBRC は特定のバックアウト・ジョブに必要なログの完全セットを判別します。さらに、DBRC は、バックアウト・ジョブと再始動ジョブが容易に連係できるように、ログに関する情報を管理しています。

例外: DBRC は HD および HISAM 再編成ユーティリティーの JCL 入力の検査をしますが、RECON データ・セット内におけるそれらのユーティリティーの実行に関する情報は記録します。

第 34 章 データベースのオープンと更新に関する情報の記録

IMS がデータベースをオープンした後で、DBRC は RECON データ・セット初期設定トークン (RIT) と、各 DBDS に関連した拡張エラー・キュー・エレメント (EEQE) を返します。RIT により、IMS は、データベースが DBRC なしで使用されていたかどうか、あるいは別の RECON データ・セットによって制御されていたかどうかを判別することができます。

関連資料: EEQE の情報については、「IMS V15 データベース管理」を参照してください。

DBDS とエリアに対する変更が行われると、DBRC はそれらの変更を RECON データ・セットに記録します。DBRC はその後、この情報を使用して、所定の DBDS またはエリアに対する変更レコードが入っている可能性のあるログ・データ・セットを判別します。

RECON データ・セットに登録されている DBDS が更新のために初めてオープンされる時 (または割り振られる時) に、IMS は DBRC に、ALLOC レコードを作成するように指示します。DEDB エリアの場合、そのエリアが更新のために初めてオープンされる時に、ALLOC レコードが作成されます。ALLOC レコードは、DBDS またはエリアを示し、最初の更新のときのタイム・スタンプおよび対応する PRILOG のオープンのタイム・スタンプを含みます。

DBRC は、ALLOC レコードを作成するときに、変更の時点にアクティブである PRILOG の LOGALL レコードに、変更される DBDS またはエリアの名前も入力します。

オンライン IMS サブシステムのオペレーター・コンソールから /DBRECOVERY コマンドを使用して DBDS またはエリアの割り振りを解除 (クローズ) すると、DBRC は、ALLOC レコードに割り振り解除タイム・スタンプを書き込みます。割り振り解除時刻が記録されない場合は、DBRC は、関連したログのクローズ時刻を割り振り解除時刻として使用します。このように RECON データ・セットには、所定のログ・データ・セットに変更レコードが存在する可能性がある DBDS またはエリアの名前のリスト (LOGALL レコード)、特定の DBDS またはエリアについて変更が存在する可能性がある時刻範囲のリスト (ALLOC レコード)、およびそれらの変更を含むログのリストが含まれています。

第 35 章 データ共有のサポート

データ共有を行うには、データベースが DBRC に登録されている必要があります。DBRC は、サブシステムが要求されたタスクを実行する権限を持っているか、さらに他のサブシステムが現在、データベースを予約していないか、チェックします。

関連資料: データ共有の詳細については、291 ページの『第 16 章 IMS 環境でのデータ共有』および「IMS V15 オペレーションおよびオートメーション」を参照してください。

データ共有のレベル

DBRC は、次の 2 つのレベルの IMS データ共有をサポートします。データベース・レベルおよびブロック・レベルです。

データベース・レベル

データベースまたは DEDB エリア全体が、一度に 1 つの IMS システムが更新アクセスのできるリソースです。エリア・リソースの場合、これはエリア・レベルの共有 と呼ばれることもあります。

ブロック・レベル

データベースまたは DEDB エリアを、複数の IMS サブシステムが並行してアクセスすることができます。データ保全本性は、共有データにアクセスする IMS サブシステムに対して維持されます。データベースまたはエリア内では、リソースはブロック・レベルで予約されます。

定義: OSAM データベースの場合、ブロックとは DASD に保管される物理データ・ブロックのことです。VSAM データベースおよび DEDB では、このブロックとは制御インターバル (CI) です。

RECON データ・セット内のデータ共有情報の記録

DBRC は、すべての登録済みデータベースについて、RECON データ・セット内にデータ共有情報を記録します。

以下のデータ共有情報は、RECON データ・セット内に記録されます。

- 各データベースに許可されている共有レベル
- 現在、処理が許可されているデータベースまたはエリアの名前
- 関与している IMS サブシステムの名前
- IMS サブシステムの状況
- リカバリーの視点からのデータベースの状況

DBRC による共用レベルの割り当て

データベースまたは DEDB エリアの共用レベルによって、アクセス要求が付与されるかどうかが決まります。DBRC により、4 つの共用レベルのうち 1 つを設定することができます。

以下の共用レベルは、INIT.DB コマンドを使用して定義され、CHANGE.DB コマンドを使用して変更されます。

SHARELVL 0

データベースの共用は行われません。データベースは、一度に 1 つの IMS システムで使用を許可されます。SHARELVL 0 は、/START コマンドで ACCESS=EX を指定することと同じです。

SHARELVL 1

共用はデータベース・レベルで行われます。一度に 1 つの IMS システムが更新を許可されますが、共用システムはいずれも、読み取り専用処理のみが許可されます。それ以外の場合は、データ共用は複数の読み取り用です。

SHARELVL 2

ブロック・レベルの共用ですが、単一 IRLM および単一 z/OS の有効範囲内のみです。共用には、1 つのデータベースを共用する IMS サブシステムが同じ RECON データ・セットを使用することが必要です。複数の IMS システムに更新処理または読み取り処理を許可することができます。

SHARELVL 3

複数レベル上の複数の IMS サブシステムによるブロック・レベルでの共用です。複数の IMS システムに非排他的アクセスを許可することができます。IMS サブシステムは、異なる IRLM を使用する複数の z/OS イメージ上で使用できます。

注:

IRLM サポートなしで実行されているバッチ・ジョブが SHARELVL 3 データベースにアクセスする場合にデータ共用環境内のデータベースの保全性を確保するために、DBRC は、データベースに対して DBRC によって現在許可されている他のすべての IMS システムとバッチ・ジョブに読み取り専用アクセス権限がある場合に限り、更新アクセス権限を持つバッチ・ジョブを許可します。

IMSCTRL マクロまたは DBBATCH プロシージャで IRLM=N が指定されている場合、データベースに対して現在 DBRC によって許可されている他のすべての IMS システムおよびバッチ・ジョブが読み取り専用アクセス権限を持つ場合に限り、DBRC はデータベースに対する更新アクセス権限をバッチ・ジョブに許可します。

バッチ IRLM=N ジョブが更新許可を持つ場合、オンライン・システムまたは他のバッチ・ジョブの許可は、そのシステムまたはジョブが読み取り専用アクセス許可を持っていない限り失敗します。

第 36 章 IMSplex のサポート

IMSplex 内でのみ使用可能な 1 つの DBRC 機能は、IMSplex 内で RECON データ・セットの入出力エラーが発生したときに、スペア・データ・セットが使用可能であれば、エラーを認識した DBRC のインスタンスが良好な RECON データ・セットをスペアにコピーし、スペアをアクティブにし、元の RECON データ・セットを割り振り解除することです。

処理のこの時点で、入出力エラーを通知した DBRC は IMSplex 内の他の DBRC に対し、再構成について自動的に通知できます。その後、元の RECON データ・セットの割り振りが解除されると、このデータ・セットを削除し、スペアの RECON データ・セットとして再定義できます。

RECON データ・セットには、順次モードまたは並列モードのいずれかでアクセスできます。順次モードでは、RECON データ・セットは一度に 1 つの DBRC インスタンスだけで使用できます。DBRC は更新が進行中である間、z/OS RESERVE マクロを使用して RECON データ・セット全体をロックするからです。並列モードでは、DBRC は DFSMS のトランザクション VSAM 機能を使用して、RECON データ・セットのレコード・レベルの共用を可能にします。単一 DBRC 要求用の RECON への (IMS、コマンド、または API からの) アクセスは、すべて、トランザクション VSAM によって 1 つ以上のトランザクションとして処理されます。

関連資料: 並列モードでの RECON データ・セットへのアクセスについて詳しくは、692 ページの『並列モードでの RECON データ・セットへのアクセス』を参照してください。

第 37 章 DBRC の IMS に対しての定義

DBRC の IMS に対する定義は、IMS システム定義プロセスの一環として行います。このトピックでは、DBRC 領域を作成し、実行するための要件について概説します。

関連資料:

- RECON データ・セットの作成および割り振りについては 691 ページの『第 42 章 RECON データ・セットの初期化および保守』を参照してください。
- データベースの登録については、647 ページの『第 38 章 データベースとデータベース・データ・セットの登録』を参照してください。
- 詳しくは、「IMS V15 インストール」、「IMS V15 リリース計画」、および「IMS V15 システム定義」を参照してください。
 - IMS のインストール手順および要件についての詳しい説明
 - DBRC プロシージャーおよびそのパラメーターの情報
 - DFSIDEF0 モジュールの情報
- INIT.RECON コマンドの詳細については、IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンドを参照してください。

IMS オンライン・システムでは必ず DBRC を使用します。これをオーバーライドすることはできません。

IMS バッチ・ジョブで DBRC を使用できるようにするかどうかを選択できますが、データ共用など、一部の機能は DBRC なしでは使用できません。

バッチおよびユーティリティー領域は、初期設定の実行中に DFSIDEF0 モジュールを使用できます。DFSIDEF0 では、DBRC= キーワードを YES、NO、または FORCE に設定できます。ADFSSMPL ライブラリーに入れて出荷される DFSIDEF0 モジュールには、バッチおよびユーティリティー領域のデフォルト DBRC=YES が入っています。この値は、DFSIDEF マクロでコーディングされます。

1. DBRCNM= パラメーターを使用して、IMS にカタログ式 DBRC プロシージャーの作成を要求します。

システム定義時に DFSPBIMS、DFSPBDBC、および DFSPBDCC メンバーの中で指定された DBRC プロシージャー名をオーバーライドすることができます。

2. バッチ・プロシージャーで DBRC を使用するかどうかを決定します。

DBRC をバッチ・プロシージャーで使用する場合は、これを DBRC= EXEC パラメーターの中で指定します。この EXEC パラメーターはオンライン IMS では無視されます。

DBRCNM= パラメーターを使用すると、オンライン IMS 実行用の DBRC プロシージャー名をオーバーライドすることができます。

3. DBRC のロード・モジュールを、IMS ロード・モジュールの通常のロード・ライブラリー検索シーケンスにあるロード・ライブラリー (例えば IMS.SDFSRESL) に入れてください。
4. システム定義時に、DBRC プロシージャを組み込みます。

IMS は、制御領域の初期設定の際に、z/OS START コマンドを実行することにより、DBRC プロシージャを自動的に開始します。このプロシージャでは、DBRC 領域用のパラメーターを指定します。オンライン IMS サブシステムの場合、DBRC は独自のアドレス・スペース内で実行されます。

IMS 15 では、DBRC プロシージャは SMP/E 処理中に IMS.SDFSPROC データ・セット内に作成され、SYS1.PROCLIB へコピーされる必要があります。メンバー名は、IMSCTRL マクロの DBRCNM パラメーター、または該当する EXEC プロシージャで指定した名前と一致している必要があります。

DBRCNM が複数の場所に指定されている場合、または DBRCNM が明示的に指定されていない場合は、以下の優先順位が適用されます。

- DBRCNM=DBRC はデフォルトです。
- IMSCTRL マクロの DBRCNM=*name* は、デフォルトをオーバーライドします。
- IMS.PROCLIB の DFSPBxxx メンバーで定義された DBRCNM=*name* は、IMSCTRL マクロの設定をオーバーライドします。
- JCL EXEC パラメーターで定義された DBRCNM=*name* は、IMS.PROCLIB メンバーの設定をオーバーライドします。

5. RECON データ・セットを初期化します。

アクセス方式サービス・プログラム (IDCAMS) の DEFINE CLUSTER コマンドを使用して RECON データ・セットを作成し、次に INIT.RECON コマンドを使用して、その RECON データ・セットを DBRC で使用可能として初期化します。

データベースを登録する予定がない場合は、データ・セットの初期化のために実行すべきコマンドは INIT.RECON コマンドのみです。

第 38 章 データベースとデータベース・データ・セットの登録

RECON データ・セットは、リカバリー DBRC が管理すべきそれぞれのデータベースごとに DB レコードを持つ必要があります。

DBRC によってデータベース・リカバリーを制御する場合は、そのデータベースを RECON データ・セットに登録する必要があります。これにより、DBRC は、データベース更新と、更新されたログ・レコードが入っている、対応するログ・データ・セットについての情報を記録します。また、DBRC は、イメージ・コピー・データ・セットと変更累積データ・セットの作成も記録し、登録済みのデータベースに影響を与えるデータベースのリカバリーと再編成も記録します。

非 HALDB (High Availability Large Databases) の場合は、INIT.DB コマンドおよび INIT.DBDS コマンドを使用してデータベースを RECON データ・セットに登録し、それらをリカバリー可能またはリカバリー不能として定義します。HALDB の場合は、INIT.DB および INIT.PART コマンドか、または HALDB 区画定義ユーティリティを使用できます。

登録したそれぞれの非 HALDB データベースごとに INIT.DBDS コマンドを出して、そのデータ・セットまたは DEDB エリアをすべて登録します。DEDB の場合は、INIT.ADS コマンドを使用して各エリア内のデータ・セットを示します。1 つのエリアに最大 7 個のエリア・データ・セット (ADS) を入れることができます。

要件: INIT.DB または INIT.PART コマンド (HALDB を登録する場合)、または任意の INIT.DBDS コマンドを実行するユーティリティ・ジョブには、コマンドの実行対象である HALDB DBD (または非 HALDB DBDS) のエントリーが入っている IMS.DBDLIB データ・セットの DD 名が必要です。非 HALDB コマンドの場合、IMS.DBDLIB の DD 名は必要ありません。

DBRC から見ると、HALDB は HALDB マスター (TYPE=HALDB) と HALDB 区画 (TYPE=PART) で構成されています。

RECON データ・セット内の HALDB に関する情報を更新または削除するには、HALDB 区画定義ユーティリティまたは DBRC コマンドの CHANGE.DB、CHANGE.DBDS、CHANGE.PART、DELETE.PART、または DELETE.DB を使用することができます。

関連資料:

- HALDB の概要および HALDB の作成方法についての詳細は、「IMS V15 データベース管理」を参照してください。
- HALDB 区画定義ユーティリティについては、「IMS V15 データベース・ユーティリティ」を参照してください。
- CHANGE コマンドおよび INIT コマンドの詳細については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

第 39 章 リカバリーの計画

非データ共用環境およびデータ共用環境でのリカバリーの計画には、2 つの基本タスクが含まれます。リカバリー・メカニズム (ロギング、チェックポイントの取得、レコードの保持) のセットアップと、リカバリーを必要とする状態に合わせた操作手順のセットアップです。

リカバリー・メカニズムの設定

非共用環境でのオンライン IMS サブシステムのロギングおよびチェックポイント・メカニズムは、データ共用環境でもアクティブです。

以下のものが含まれます。

- システム・ログ・データ・セットと WADS および再始動データ・セットの使用
- プログラム、システム、およびメッセージ・キュー・チェックポイント
- データベース・イメージ・コピーの作成

非共用環境とデータ共用環境の間の主な違いは、DBRC への依存の程度にあります。DBRC はデータ共用環境の制御を援助し、単にレコードを保持するだけではありません。

具体的に、非データ共用環境では以下の項目に違いがあります。

- データベースの再構成 (順方向リカバリー)
- プログラム・レベル (動的) のバックアウト
- データベース・バックアウト
- IMS 再始動 (組み込みリカバリーによる)

関連資料: 非データ共用環境とデータ共用環境におけるリカバリー手順の詳細については、「IMS V15 オペレーションおよびオートメーション」を参照してください。

リカバリー機能

動的バックアウトおよびバッチ・バックアウト・サポートは、非データ共用環境と共用環境では動作が異なります。

- 動的バックアウト

非データ共用環境およびデータ共用環境におけるオンライン IMS サブシステムは、アプリケーション・プログラムでコミットされていないデータベースの変更をバックアウトし、以下のいずれかの条件の下ではそのコミットされていない出力メッセージを廃棄します。

- プログラムが失敗する。
- プログラムがロールバック呼び出しでバックアウトを要求する。

ただし、データ共用環境では、IRLM ロックおよび RECON データ・セット内の状況表示によって、コミットされていない変更が共用サブシステムから保護され、保全性が保証されます。システムおよびその他のプログラムの操作は、中断されずに続行します。

関連資料: 動的バックアウトの詳細については、「IMS V15 データベース管理」を参照してください。

- バッチ・バックアウト・サポート

DBRC は、IMS 再始動動的バックアウトおよびバッチ・バックアウト・ユーティリティーとインターフェースを持って、バックアウトが必要なデータベースへのアクセスを制御することにより、データベースの保全性を改善しています。DBRC は BACKOUT レコードを作成して、バックアウトが必要なそれぞれのデータベースごとに各リカバリー単位 (UOR) を追跡します。DBRC は、バッチ・バックアウト・ユーティリティーへの入力であるログを検査します。

BACKOUT レコードはオンライン・サブシステム用に作成されます。動的バックアウトが使用され、それが失敗したのではない限り、DL/I バッチ・サブシステム用には BACKOUT レコードは作成されません。

COLDSYS パラメーターを指定して緊急時再始動を実行する (/ERESTART COLDSYS) 前に、COLDSTART または ACTIVE パラメーターを指定してバッチ・バックアウトを実行します。バッチ・バックアウトの完了後、DBRC は経過中で未確定のすべての UOR について、バックアウトの候補として BACKOUT レコードを作成します。次の IMS 再始動により、候補の UOR は「バックアウト必要」状況に促進され、DBRC は RECON データ・セットのデータベース・レコードに BACKOUT NEEDED=ON を設定します。

動的バックアウトが失敗すると、動的バックアウトの失敗を示す UOR とともに BACKOUT レコードが作成され、DBRC は RECON データ・セットのデータベース・レコードに BACKOUT NEEDED=ON を設定します。

データベースが正常にバックアウトされると、DBRC は (RECON データ・セットの) データベース・レコードのバックアウト・フラグを適宜リセットし、BACKOUT レコードを更新します。すべての UOR のバックアウト処理が完了すると、DBRC は RECON データ・セットから BACKOUT レコードを削除します。

DBCTL の場合、未解決の未確定 UOR をバックアウトしないようにした場合は、DELETE.BKOUT コマンドまたは CHANGE.BKOUT コマンドのいずれかを使用して、RECON データ・セットの BACKOUT レコードからその未解決の未確定 UOR を除去します。

DBRC コマンドは、必要であれば、BACKOUT レコードを更新する際に使用できます。バックアウト・レコードを手作業で変更することは、最小限にとどめておいてください。

重要: DELETE.BKOUT または CHANGE.BKOUT コマンドを使用する場合、データベースの保全性を損なうおそれがあるため、細心の注意を払ってください。

DBRC は、バッチ・バックアウト・ユーティリティーへの入力として使用されるログの妥当性を検査します。DBRC は入力ログ・ボリュームについて、そのログ・ボリュームの順序が正しいか、必要なログがすべて用意されているか、および正しくクローズされたかを確認する検査を行います。バッチ・バックアウト・ユーティリティーの SYSIN ステートメントに ACTIVE または COLDSTART ステートメントが組み込まれている場合、DBRC は追加の検査を行って、再始動に関連するすべてのボリュームが含まれているか確認します。また、DL/I バッチ・ログの場合は、正しいボリュームが最後のバッチ実行から来ていることを確認する検査も行われます。

重要: BYPASS LOGVER がバッチ・バックアウト・ユーティリティーの SYSIN ステートメントに組み込まれている場合、DBRC は入力ログを検査せず、バッチ・バックアウト・ユーティリティーは UOR について DBRC に通知しません。バックアウト処理が正常に完了すると、既存の UOR は更新されません。

DBRC のバックアウト・サポートによって、バックアウトの使用によく起こるエラーが原因の損傷からデータベースが保護されますが、以下のリストでは、これ以外にデータベース損傷のもととなるいくつかの可能性を紹介します。

- IMS サブシステムが異常終了し、経過中で未確定の UOR が (バッチ・バックアウトの実行によって) DBRC に識別される前に /ERE COLDSYS コマンドが実行されると、そのような経過中で未確定の UOR に関連したデータベースは、COLDSTART ステートメントまたは ACTIVE ステートメントを使用して最初のバッチ・バックアウトが実行されるまで、誤った更新から保護されません。
- バッチ・ジョブの複数の実行からのログを (DISP=MOD を指定することによって)、同じログ・データ・セットに組み込むと、ログ検査は信頼性の低いものになります。
- RECON データ・セットは DBRC コマンドを使用して変更できます。これらのコマンドを不注意に使用すると、以下のようなエラーにつながる可能性があります。
 - 行ってはならないバックアウトが許可される
 - バックアウトの必要があるデータベースへ他のサブシステムのアクセスが許可される
- 入力ログが無効、あるいはバックアウトが不要であることを RECON データ・セットの情報が示している場合、バッチ・バックアウト BYPASS LOGVER 制御ステートメントによりバックアウトを行うことができます。この制御ステートメントを不注意に使用すると、行ってはならないバックアウトを行う原因になります。
- 未登録のデータベースは、バックアウトが必要な場合でも、使用されないように保護されることはありません。
- タイム・スタンプ・リカバリーを (このログの前の指定時刻までデータベースをリカバリーするために) 実行した後に、正常に終了した (IRLM を使用しなかった) ジョブをバックアウトすると、ログ検査は信頼性の低いものになります。バックアウトされるログはこのサブシステム名 (SSID) での前回のログであるため、DBRC は、そのログを入力として使用できるという誤った確認をしてしまいます。

関連資料: UOR、データベース・バックアウト処理、およびバッチ・バックアウト・ユーティリティーの詳細については、「IMS V15 データベース管理」および「IMS V15 データベース・ユーティリティー」を参照してください。

- 順方向リカバリー

データ共用環境でのデータベースのリカバリー処理と、非共用環境でのデータベースのリカバリーとは似ているところがあります。どちらの環境でも、データベース・リカバリーには DBRC を使用し、リカバリー処理への入力データとして次のものを使用します。

- 破損したデータベースの最新のイメージ・コピー

注: HALDB オンライン再編成をリカバリーの開始点に使用する場合は、すべてのデータベース変更がログ・データ・セット (単数または複数) に記録されるため、イメージ・コピーは必要ありません。

- リカバリー・ログ・データ・セット (RLDS) またはそのコピーが作成された後に使用されたすべての関連システム・ログ・データ・セット

推奨事項: リカバリー処理への入力データとしては RLDS を使用するようにしてください。RLDS にはデータベース変更ログ・レコードしか含まれていないため、RLDS は、関連システム・ログ・データ・セットより効率的な、リカバリー処理への入力データです。

順方向リカバリー用にデータベース・リカバリー管理ユーティリティーを使用していて、ブロック・レベルのデータ共用が使用されている場合、変更累積の追加ステップが必要になる場合があります。

関連資料: 順方向リカバリーの詳細な解説については、「IMS V15 データベース管理」を参照してください。

DBRC なしのリカバリー

DBRC が実行中でないときにリカバリー関連のアクション (データベース・イメージ・コピーの作成など) をオフラインで実行すると、DBRC に状況の変更が伝えられないため、問題が生じることがあります。

したがって、DBRC にはそのような変更を特に通知しておく必要があります。DBRC は、このためにいくつかのコマンドを用意しています。

ユーティリティー制御機能 (UCF) は再始動が必要なため、DBRC をアクティブにせずに使用しなければならない場合があります。リカバリー関連の作業で UCF を使用する場合は、状況変更について、後で DBRC に通知する必要があります。

関連資料: DBRC を使用しないリカバリーと、それに関連するコマンドの詳細については、「IMS V15 データベース管理」を参照してください。

IMS の障害後の再始動

データ共用環境における IMS サブシステムの再始動は、非データ共用環境と同じ方法で行います。

- IMS を正常にシャットダウンできる場合は (/CHECKPOINT PURGE、/CHECKPOINT DUMPQ、または /CHECKPOINT FREEZE コマンドを使用するチェックポイント・シャットダウン)、 /NRESTART コマンドを使用して IMS を通常どおり再始動します。
- IMS に障害が発生した場合は、/ERESTART コマンドを使用して緊急時再始動を実行する必要があります。

データ共用環境では、関連した IRLM も停止または失敗した場合、ブロック・レベル環境で IMS を始動する前に、まず IRLM を始動する必要があります。

関連資料: IMS の障害後の再始動の詳細については、「IMS V15 オペレーションおよびオートメーション」を参照してください。


DBRC の障害後の再始動

IMS は、DBRC が IMS の制御下で実行されているため、DBRC に障害が発生したかどうかを認識できます。DBRC に障害が生じると、IMS は異常終了し、そのバッファのフラッシュを試み、システム・ログをクローズします。DBRC の問題を訂正した後で、/ERESTART コマンドを実行して IMS を再始動することができます。

IRLM 構成を伴うリカバリー

DBRC は、IRLM 状況情報を RECON データ・セット内のサブシステム・レコードに記録します。

関連資料:

 RECON 内のフィールドのリスト (レコード・タイプ別)(コマンド)

バッチ・バックアウト

COLDSYS パラメーターを指定して緊急時再始動を実行する (/ERESTART COLDSYS) 前に、COLDSTART または ACTIVE パラメーターを指定してバッチ・バックアウト・ユーティリティを実行します。

(バックアウトを必要とする) 登録済みデータベースのバックアウトが完了するまで、そのデータベースがアクセスされないように保護するため、DBRC にログの名前を与えます。

DB レベルの制御の場合、未解決で未確定の UOR をバックアウトしない場合には、DELETE.BKOUT コマンドを使用して、BACKOUT レコードからその UOR を除去してください。

重要: DELETE.BKOUT コマンドの使用には細心の注意が必要です。このコマンドは、RECON データ・セットからサブシステムのバックアウト情報をすべて削除します。

第 40 章 DBRC システムの考慮事項

DBRC は、データベース関連プロセスおよびログ関連プロセスを制御します。

これらの処理には、以下のものが含まれます。

- データベースのバックアップ・コピーの作成
- DB 変更累積データ・セットの作成
- データベースのリカバリー
- バックアウトを必要とするデータベースの保護
- OLDS のアーカイブ

DBRC 使用上の一般的な考慮事項

DBRC の使用時には、いくつかの考慮事項を認識しておく必要があります。

DBRC を使用する際は、以下の考慮事項に留意してください。

- DBRC は主記憶データベース (MSDB) をサポートしていません。
- DBRC は GSAM データベースの処理上は何の役割も果たさないため、GSAM データベースを登録する理由はありません。
- DBRC を使用し、更新アクセスを行うバッチ・ジョブの場合は、ロギングが必須です。
- 現在使用中のデータ・セットに関する RECON データ・セット情報 (例えば、DBDS やログ・データ・セットについての) を更新してはなりません。
- 時刻 (TOD) 刻時機構の値をホスト・プロセッサに正確に設定するようにしてください。さもないと、予測できない結果が RECON データ・セットで生じる恐れがあります。

データ・セット命名規則

多くの場合、イメージ・コピーおよび変更累積データ・セットで z/OS カタログ機能を使用するかどうかはオプションです。その理由は、DBRC は、常にこれらのデータ・セットに関するボリューム通し番号の情報を RECON データ・セットに記録するためです。イメージ・コピー・データ・セットと変更累積データ・セットをカタログする場合は、これらに固有のデータ・セット名を付ける必要があります。

要件: データベース・イメージ・コピー 2 ユーティリティの DFSMSdss 高速複製コピー機能を使用して取ったイメージ・コピーは、z/OS カタログ機能を使用する必要があります。

DBRC はデータ・セット命名規則を備えており、将来使用するために定義するような、イメージ・コピー・データ・セット (標準イメージ・コピーのほか、HSSP イメージ・コピーと並行イメージ・コピーも) および変更累積データ・セットに固有のデータ・セット名を生成する際に役立ちます。

この規則を常時使用していれば、データ・セット名が固有であることは保証されま
す。この規則を時々しか使用しない場合には、命名規則に従わなかったこと、およ
び RECON データ・セットに重複データ・セット名が存在する可能性があることを
示すメッセージがジョブ・ステップの終了時に送られてきます。DBRC は、引用符
で囲まれたデータ・セット名が命名規則に従っていないものと見なします。したが
って、DBRC は、引用符で囲まれたデータ・セット名は検査しません。

将来使用するために、RECON データ・セットにリカバリー・ユーティリティの
1 つにデータ・セットを作成するレコードを追加し、このデータ・セット命名規則
を使用するときは、完全修飾データ・セット名を指定しても、単に省略形の
high-level-qualifier..low-level-qualifier* (例えば、ALPHA1*.OMEGA) を指
定しても、いずれでもかまいません。命名規則に従ってデータ・セット名を指定す
る場合は、DBRC のいずれの INIT、CHANGE、DELETE、または
NOTIFY.REORG コマンドでも、このような省略名を使用することができます。
DBRC は、RECON データ・セットにアクセスする前に、省略名をもとの完全修飾
形式に拡張します。

以下のトピックでは、追加のデータ・セット命名規則について説明します。

イメージ・コピー・データ・セットの命名規則

DFSMSdss 高速複製機能を使用しないイメージ・コピー・データ・セットの命名規
則は、以下のとおりです。*high-level-qualifier.dbdname.ddname.IC.low-level-
qualifier*

high-level-qualifier

1 から 8 文字の長さの英数字からなる文字ストリング。先頭文字は英字で
なければなりません。

dbdname

DBDS のデータベース名であり、このデータベースのイメージ・コピー・デ
ータ・セットが RECON データ・セットに記録されます。

ddname

DBDS のデータ・セット DD 名であり、このデータベースのイメージ・コ
ピー・データ・セットが RECON データ・セットに記録されます。

IC これがイメージ・コピー・データ・セットであることを示しています。

low-level-qualifier

1 から 8 文字の長さの英数字からなる文字ストリング。それぞれの DBDS
ごとに固有の文字ストリングを選択し、先頭文字は英字でなければなりません。

DFSMSdss 高速複製機能を使用するイメージ・コピー・データ・セットの命名規則
は、以下の 2 つのフォーマットのどちらでもかまいません。

high-level-qualifier.dbdname.ddname.Dyyddd.Thhmmss

high-level-qualifier.dbdname.ddname.

high-level-qualifier

1 から 25 文字の長さの英数字からなる文字ストリング。先頭文字は英字で
なければなりません。この文字ストリングは、複数のデータ・セット修飾子
を表すことができ、その場合、各修飾子は 1 文字から 8 文字までの長さの

英数字で、ピリオドで区切ります。各修飾子の 1 文字目は英字であることが必要で、ストリングの最後の文字はピリオドであってはなりません。

dbdname

データベースの DBD 名であり、このデータベースのイメージ・コピー・データ・セットが RECON データ・セットに記録されます。

ddname

DBDS のデータ・セット DD 名であり、このデータベースのイメージ・コピー・データ・セットが RECON データ・セットに記録されます。

yyyddd

日付です。このストリングは、文字ストリング「D」で始める必要があります。

hhmmss

タイム・スタンプです。このストリングは、文字ストリング「T」で始める必要があります。

関連資料: データベース・イメージ・コピー 2 ユーティリティ (DFSUDMT0) で DFSMSdss 高速複製機能を使用する詳細については、「IMS V15 データベース・ユーティリティ」のこのユーティリティに関する説明を参照してください。

重複イメージ・コピー・データ・セットの命名規則

重複イメージ・コピー・データ・セットの形式は、次のとおりです。

high-level-qualifier.dbdname.ddname.IC2.low-qualifier

IC2 フィールドが重複イメージ・コピー・データ・セットであることを示す点を除き、この形式はイメージ・コピー・データ・セットの規則と同じです。

変更累積データ・セットの命名規則

変更累積データ・セットの形式は次のとおりです。*high-level-qualifier.cagrname.CA.low-level-qualifier*

high-level-qualifier

1 から 8 文字の長さの英数字からなる文字ストリング。先頭文字は英字でなければなりません。

cagrname

CA グループの名前であり、これに対して変更累積データ・セットを作成しています。

CA これが変更累積データ・セットであることを示しています。

low-level-qualifier

1 から 8 文字の長さの英数字からなる文字ストリングで、それぞれの CA グループごとに固有でなければなりません。先頭文字は英字でなければなりません。

データベースのバックアップ・コピー

IMS は通常のシステム・チェックポイントを取る場合、DL/I の内部制御情報を記録します (高速機能の場合は、IMS はバッファおよび MSDB も記録します) が、データベースの外部の内容は記録しません。データベースが破損すると、最後のシステム・チェックポイントを調べても役に立ちません。ログを見ればどのような変更が行われたかは分かりますが、データベースのバックアップ・コピーがないと、リカバリーは不可能です。

推奨事項: データベースを最初にロードした後、データベースのバックアップ・コピーを取り、さらに新しいバックアップ・コピーを定期的に取り、リカバリーの際に処理すべきログ変更レコードが少なく済むため、リカバリーに要する時間が削減されます。

IMS では、次のような方法でバックアップ・コピーを取ることができます。

- データベース・イメージ・コピー・ユーティリティ (DFSUDMP0) はオフラインで実行され、アクセス方式サービス・プログラム (AMS) を使用してコピーを作成します。
- データベース・イメージ・コピー 2 ユーティリティ (DFSUDMT0) は、オフラインで実行され、DFSMSdss 並行コピーまたは DFSMSdss 高速複製を呼び出してコピーを作成します。
- 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 (DFSUICP0) はオンラインで実行され、IMS サービスを使用してコピーを作成します。
- HISAM 再編成アンロード・ユーティリティ (DFSURUL0) からの出力であるアンロードされたデータは、バックアップ・コピーとして使用することができます。
- HSSP 処理は DEDB エリアのイメージ・コピーを作成することもできます。

これらのユーティリティを実行する場合に、ユーティリティは (インストール・パラメーターに応じて) DBRC を呼び出して、RECON データ・セット内の重要な情報を更新することができます。サポートされているイメージ・コピー・ユーティリティを使用すると、DBRC は登録済みデータベースのイメージ・コピーを記録します。GENJCL.IC または GENJCL.OIC コマンドを入力した場合、DBRC はこのユーティリティ用の JCL も生成します。

また、オペレーティング・システムによって提供される各種のユーティリティを使用してバックアップ・コピーを作成することもできます。ただし、これらのユーティリティは DBRC と対話していないため、標準外イメージ・コピーを DBRC に通知するには特定のアクションを取る必要があります。

関連資料: DEDB エリアの HSSP 処理の詳細については、「IMS V15 データベース管理」を参照してください。

イメージ・コピー・ユーティリティ (DFSUDMP0、DFSUDMT0、DFSUICP0)

IMS では、データベースに変更が加えられる前後のデータベースの「写真を撮る」ことができます。この「写真」にあたるものをイメージ・コピーといいます。この用語は、コピーが現状のままのイメージであるという事実を示しています。つま

り、イメージ・コピー・ユーティリティーは、データベースをコピーするときに、データベースの物理形式を変えません。イメージ・コピーはデータのバックアップ・コピーであり、データベース・リカバリーの処理速度を向上させるのに役立ちます。

データベース・イメージ・コピー・ユーティリティー (DFSUDMP0)、データベース・イメージ・コピー 2 ユーティリティー (DFSUDMT0)、およびオンライン・データベース・イメージ・コピー・ユーティリティー (DFSUICP0) は、データベースのイメージ・コピーを作成します。すべてのイメージ・コピー・ユーティリティーは、データ・セットまたは DEDB エリアに対して操作します。そのため、データベースが複数のデータ・セットまたはエリアで構成されている場合は、ユーティリティーに複数の指定を必ず与えるようにしてください。サポートされているいずれかのイメージ・コピー・ユーティリティーで、1 つのイメージ・コピー・データ・セットと 1 つの重複イメージ・コピー・データ・セットの両方をユーティリティーの 1 回の実行で作成するように要求することができます。

推奨事項: 1 つのデータベースに属しているすべてのデータ・セットまたはエリアは、一度でコピーしてください。データベースを前の状態にリセットするために複数のリカバリーを実行する場合は、そのデータベースおよび論理的に関連するすべてのデータベース (アプリケーション・プロセスによって関連付けられているものも含めて) に属するすべてのデータ・セットを同一点までリカバリーして、保全性の問題が生じないようにしてください。

それぞれのイメージ・コピー・ユーティリティーは、データベースおよびエリアをオフラインにせずにバックアップ・コピーを作成するオプションを備えています。この機能を使用して、データベースの可用性を高めることができます。データベースが並行更新処理で使用可能な間に IMS アプリケーションが取ったイメージ・コピーは、並行イメージ・コピーまたはファジー・イメージ・コピーと呼ばれます。アクティブ・アプリケーションによってすでにデータベースに加えられた変更は、データ・セットに物理的に書き込まれていなかったことが考えられるため、コピーから欠落している可能性があります。ただし、これらの変更はログには書き込まれています。この場合、すべての変更が適用されることを確実にするために、ログ上で前の時点に戻る必要があります。どの程度戻るかは、データベースのタイプおよびどのイメージ・コピー・ユーティリティーが使用されたかによって異なります。

並行コピー・イメージ・コピーも高速複製イメージ・コピーも使用しない場合は、データベースをオフラインにするか「読み取り」アクセス専用とし、一貫性のある、つまり「クリーンな」イメージ・コピーが取られるようにする必要があります。詳しくは、663 ページの『並行イメージ・コピー』を参照してください。

データベースが更新のためにアクセスされていないときにイメージ・コピーが作成された場合は、コピーの実行時以降にログに記録された変更のみが必要になります。

これらのユーティリティーを使用する場合、出力イメージ・コピーを 1 つから 4 つ作成することができるオプションがあります。データベース・イメージ・コピー 2 ユーティリティーだけが、3 つまたは 4 つの出力コピーを作成することができ、その最初の 2 つの出力コピーのみが RECON データ・セットに記録されます。データベース・イメージ・コピー 2 ユーティリティーで DFSMSdss 高速複製機能を使用した場合は、1 つの出力コピーだけが作成されます。

複数コピー作成の利点は、一方のコピーで入出力エラーが起こった場合に、ユーティリティーは他方のコピーで最後まで継続できるということです。また、一方のコピーを読み取ることができない場合は、もう一方のコピーを使用してリカバリーを実行することもできます。複数コピーを作成するかどうか決める際のトレードオフは、追加のコピーを書き込むのに要する時間のために、パフォーマンスが低下する可能性があるという点です。

DBRC は、3 つのイメージ・コピー・ユーティリティーで同じ様に機能します。DFSMSdss 高速複製機能を使用しない場合、イメージ・コピー・データ・セットの事前定義および再使用に関する規則が 3 つすべてに適用されます。各ユーティリティーは DBRC を呼び出して、以下のことを行います。

- ユーティリティーへの入力を検査する (DBRC は、入力が有効な場合にのみユーティリティーの実行を許可します)。
- ユーティリティーが作成したイメージ・コピー・データ・セットに関する情報を RECON データ・セットに記録する。

RECON データ・セット内のイメージ・コピー・レコードは同一フォーマットを持ち、対応するイメージ・コピー・データ・セットが、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いずれによって作成されたかに関係ありません。

次の 2 つの異なるコマンドが、イメージ・コピー・ジョブを作成します。

- GENJCL.IC。オフライン・ユーティリティーのデータベース・イメージ・コピーおよびデータベース・イメージ・コピー 2 の場合
- GENJCL.OIC。オンライン・ユーティリティーのオンライン・データベース・イメージ・コピーの場合

ロギングなしでバッチ・ジョブを実行する場合は、その直後にイメージ・コピーを取ります。これらのバッチ・ジョブを、後続のリカバリーの一部として、データベース・リカバリー・ユーティリティーと組み合わせて再実行することに頼らないでください。バッチ処理は物理的に反復可能であるとの保証はないため、この組み合わせによってデータベースが損傷する恐れがあります。

データベース・イメージ・コピー (DFSUDMP0)

データベース・イメージ・コピーはオフラインで実行されます。これは、データベースがオンラインのままでもイメージ・コピーを作成できる、並行イメージ・コピー (CIC) オプションをサポートします。CIC オプションを使用すると、OSAM データ・セットおよび VSAM 入力シーケンス・データ・セット (ESDS) のイメージ・コピーを作成できます。データベース・イメージ・コピーは、VSAM キー・シーケンス・データ・セット (KSDS) のイメージ・コピーの作成はサポートしません。

データベース・イメージ・コピー (DFSUDMP0) ユーティリティーを使用して、次のデータベースのデータ・セット・コピーを作成します。

- HISAM データベース
- HIDAM データベース
- HDAM データベース

- PHDAM (区分 HDAM データベース)
- PHIDAM (区分 HIDAM) データベース
- PSINDEX (区分副次索引) データベース
- DEDB エリア

並行イメージ・コピーを要求するには、GENJCL.IC コマンドで CIC キーワードを使用します。もしくは、イメージ・コピー・ジョブの EXEC ステートメントで CIC パラメーターを指定することもできます。このユーティリティーでは DBRC を使用する必要があり、DBRC に登録されているデータベースの並行イメージ・コピーしか取ることができません。

データベース・イメージ・コピー・ユーティリティーを実行して一貫性のあるイメージ・コピーを取る (CIC オプションは指定しない) とき、DBRC は必須ではありませんがお勧めします。DBRC は、ユーティリティーの実行中にデータベースまたはエリアに対する更新アクティビティーが行われていないことを確認します。DBRC を使用せずにこのユーティリティーを実行する場合は、データベースまたはエリアに対して更新が行われていないことを確認する必要があります。例えば、/DBDUMP コマンドまたは /STOP AREA コマンドを出して、以前システムで更新を行ったトランザクションがデータベースまたはエリアを更新しないようにすることができます。

データベース・イメージ・コピー 2 (DFSUDMT0)

データベース・イメージ・コピー 2 ユーティリティー (DFSUDMT0) は、DFSMSDss 並行コピー機能または DFSMSDss 高速複製機能を使用してイメージ・コピーを取ります。

並行コピー・オプションを使用した場合、DFSUDMT0 は DFSMSDss を呼び出して入力データ・セットをダンプします。この出力は、RECON データ・セットに記録され、他のいずれかのイメージ・コピー・ユーティリティーからの出力と同じ方法で、データベース・リカバリーに使用することができます。このユーティリティーは、HISAM、HIDAM、HDAM、PHDAM、PHIDAM、および PSINDEX データベースと、DEDB エリアに対して使用することができます。このユーティリティーがコピーするデータベース・データ・セットは、並行コピー・オプションをサポートしているハードウェア上にある必要があります。

オプションの COMPRESS パラメーターは、DFSMSDss 並行コピーで「圧縮」オプションを呼び出します。この圧縮オプションによって、イメージ・コピーの保持に必要なストレージ・スペースを削減できます。ただし、圧縮オプションを使用すると、コピー操作を実行するために必要な CPU 時間が増えます。

CPU 使用量を軽減して圧縮の向上を図るために、DFSUDMT0 は、COMPRESS ZCOMPRESS(PREFERRED) オプションを指定した DFSMDdss から圧縮を要求して、z/Enterprise Data Compression (zEDC) サービスが z/OS システム内で使用可能になっていれば、zEDC を圧縮に使用するようにします。zEDC が使用可能になっていない場合、または zEDC サービスに対する要求が何らかの理由で拒否された場合は、標準 DFSMSDss 圧縮ルーチンが使用されます。

注: COMPRESS パラメーターは、DFSMSDss 高速複製には使用できません。

高速複製オプションを使用した場合、DFSUDMT0 は DFSMSdss を呼び出して入力データ・セットをコピーします。この出力は、RECON データ・セットに記録され、他のいずれかのイメージ・コピー・ユーティリティからの出力と同じ方法で、データベース・リカバリーに使用することができます。このユーティリティは、HISAM、HIDAM、HDAM、PHDAM、PHIDAM、および PSINDEX データベースと、DEDB エリアに対して使用することができます。このユーティリティによってコピーされるデータベース・データ・セットは、高速複製オプションをサポートしているハードウェア上に存在する必要があります。

データベース・イメージ・コピー 2 ユーティリティは、一貫性のあるイメージ・コピーを取る際に、データベースの可用性を高めます。データベースで更新処理ができなくなるのは、DFSMSdss が並行コピー・セッションまたは高速複製セッションを確立する間の、非常に短い時間だけで済みます。その後、イメージ・コピー・データ・セットが実際に書き込まれる間に、更新処理を再開することができます。この更新はイメージ・コピーには含まれません。

また、このユーティリティは、IMS アプリケーションがデータベースを更新している最中に、そのデータベースをコピーすることもできます。この場合に作成されるイメージ・コピーは、「ファジー」イメージ・コピーです。データベース・イメージ・コピー 2 ユーティリティは、データベースがオンラインになっている間に、サポートされているすべてのタイプ (VSAM KSDS を含む) のデータ・セットをコピーすることができます。VSAM KSDS をオンラインの間にコピーするためには、VSAM KSDS が SMS で管理されており、BWO(TYPEIMS) パラメーターを使用して IDCAMS によって割り振られている必要があります。

要件: データベース・イメージ・コピー 2 ユーティリティは DBRC を使用する必要があります。コピーされるデータベースおよびエリアは DBRC に登録済みでなければなりません。このユーティリティは、並行コピーを使用して最大 4 つまでのコピーを作成できます。高速複製を使用した場合は、1 つのコピーだけを作成できます。1 次コピーと 2 次コピーに関する情報だけが、RECON データ・セットに記録されます。

データベース・イメージ・コピー 2 ユーティリティからの並行コピー・イメージ・コピーの出力は DFSMSdss ダンプ・フォーマットであり、他のイメージ・コピー・ユーティリティの出力フォーマットとは異なります。しかしこのフォーマットは、データベース・リカバリーへの入力として直接使用できます。イメージ・コピーの情報は、イメージ・コピー・タイプ SMSVIC または SMSNOCIC を指定して RECON データ・セットに記録されます。

データベース・イメージ・コピー 2 ユーティリティの高速複製呼び出しからのイメージ・コピー出力は、DFSMSdss コピー・フォーマットになっており、これは他のイメージ・コピー・ユーティリティの出力フォーマットとは異なります。DFSMSdss ダンプ・フォーマットと同様に、これはデータベース・リカバリーへの入力として、直接使用できます。イメージ・コピーの情報は、イメージ・コピー・タイプ SMSONLC または SMSOFFLC を指定して RECON データ・セットに記録されます。

要件: データベース・リカバリー・ユーティリティはリカバリー用に SMSVIC、SMSNOCIC、SMSONLC、または SMSOFFLC イメージ・コピー・データ・セットを使用する場合は、DBRC を使用する必要があります。

オンライン・データベース・イメージ・コピー (DFSUICP0)

オンライン・データベース・イメージ・コピー (DFSUICP0) は、オンライン IMS および DBCTL 環境で、BMP プログラムとして実行されます。

このユーティリティは、HISAM、HIDAM、HDAM、PHDAM、PHIDAM、および PSINDEX データベースでのみ使用することができます。エリアはこれによってサポートされていません。このユーティリティによってコピーされるデータベースは、ユーティリティを実行している IMS サブシステムで更新用に使用することができます。その他の IMS サブシステムでは、このデータベースへの並行更新アクセスを行うことはできません。

このユーティリティの実行中にコピーされるデータベースが更新されると、ファジー・イメージ・コピーが作成されます。このイメージ・コピーによるリカバリーには、オンライン・データベース・イメージ・コピー・ユーティリティが開始された時に使用したログから始まるすべてのログが必要となります。

並行イメージ・コピー

IMS は、データベースをオフラインにすることなく、そのデータベースのイメージ・コピーを取る機能を備えています。このことは、イメージ・コピーを取っている間にデータベースの更新が可能であり、イメージ・コピーにその更新の一部または全部が反映される場合もあれば、まったく反映されない場合もあるということを意味しています。

このイメージ・コピーは「ファジー」と呼ばれます。その理由は、このコピーは、ある特定の時点というよりも、むしろ、ある一定の期間にわたるデータベースの状態を表しているためです。また、更新処理が生じている最中にコピーが取られるため、「並行イメージ・コピー」とも呼ばれます。

IMS アプリケーションがデータベースを更新中にイメージ・コピーを取るというこの機能により、データベースの可用性を高めることができます。オフライン・イメージ・コピー・ユーティリティであるデータベース・イメージ・コピーとデータベース・イメージ・コピー 2 は、並行イメージ・コピーを取るオプションを備えています。並行イメージ・コピーは、データベース・イメージ・コピー 2 の並行コピー・オプションと高速複製オプションのどちらによっても作成できます。オンライン・データベース・イメージ・コピー・ユーティリティによってコピーされるデータベースは、そのユーティリティを実行中の IMS サブシステム (他の IMS サブシステムは不可) によって、並行して更新することができます。HSSP 処理によって作成されるイメージ・コピーも「ファジー」コピーです。これは、HSSP の実行中でも、エリアは更新処理で使用できるためです。

一貫性のあるイメージ・コピーがデータベース・リカバリーに入力される時、リカバリーはそのイメージ・コピーのジョブ完了以降のログのみを必要とします。並行イメージ・コピーには、イメージ・コピー処理の開始以前、またはその実行中に行われた更新は含まれていない可能性があります。したがって、並行イメージ・コピーがリカバリーに入力される場合は、イメージ・コピー処理の開始以前からのログをデータベース・リカバリーに与える必要がある可能性があります。

制約事項: 並行イメージ・コピーを作成するには、次の制約事項が適用されます。

- 並行イメージ・コピー用には、DBRC に登録済みのデータベースのみが適格です。
- データベース・イメージ・コピー・ユーティリティ (DBRC を使用しないで一貫性のあるイメージ・コピーを取ることができる) は、並行イメージ・コピーを作成するとき DBRC を使用する必要があります。
- リカバリー不能データベースは、イメージ・コピーを完了するためのログ・データがないため、並行イメージ・コピーには適格ではありません。

関連概念:

131 ページの『データベースのバックアップ・コピー』

将来の利用と再利用のためのイメージ・コピー・データ・セットの作成

REUSE パラメーターを使用して、イメージ・コピー・データ・セットを定義し、将来、利用するためにそれを RECON データ・セットに記録できるように DBRC に伝えます。

イメージ・コピー・データ・セットは INIT.IC コマンドで定義します。DBRC は、GENJCL.IC コマンドを処理する際に、イメージ・コピー・ユーティリティが使用するイメージ・コピー・データ・セットを 1 つ選択します。

制約事項: DFSMSdss 高速複製 (データベース・イメージ・コピー 2 ユーティリティによって呼び出されるもの) は、REUSE パラメーターをサポートしていません。

データベース・イメージ・コピー・ユーティリティが使用可能なイメージ・コピー・データ・セットを使用する場合、DBRC は、データベース・イメージ・コピー・ユーティリティがそのイメージ・コピー・データ・セットを使用して実行したときのタイム・スタンプで RECON データ・セット内のそのレコードを更新します。

NOREUSE パラメーターを指定すると、イメージ・コピー・データ・セットを事前定義することはできません (これがデフォルトです)。GENJCL.IC コマンドの処理で使用される骨組み JCL メンバーで、あるいは自分で作成した JCL で、このユーティリティ用の出力データ・セット名を与える必要があります。NOREUSE を指定すると、DBRC は、出力イメージ・コピー・データ・セットの装置タイプを動的に設定します。DBRC は、これを、INIT.RECON および CHANGE.RECON コマンドで指定されたときの装置のデフォルト装置タイプに設定します。DFSMSdss 並行コピーを 2 つを超えて必要とする場合は、NOREUSE を指定します (並行コピーは最大 4 つ取ることができます)。

REUSE を指定しない場合は、イメージ・コピー・ユーティリティを実行するたびに、DBRC は、GENMAX および RECOVPD の両方の値を超える最も古いイメージ・レコードを削除します。イメージ・コピー・データ・セットそのものは削除されません。RECON データ・セット内のそのレコードのみが削除されます。DBRC はもはやそのデータ・セットの存在を認知していないため、削除するか、もしくは自分で追跡し続ける必要があります。

DEDB エリアの HSSP のイメージ・コピー・オプションを使用している場合、そのエリアは REUSE パラメーターを指定して定義する必要があり、また事前定義するデータ・セットはカタログしておく必要があります。

管理されるイメージ・コピー数の制御

INIT.DBDS および CHANGE.DBDS コマンドを使用して、DBRC がレコードを維持すべき DBDS またはエリアのイメージ・コピーの数を指定します。

GENMAX パラメーターは、識別された DBDS または DEDB エリア用に DBRC が維持するリカバリーの世代 (イメージ) の最大数を指定します。重複イメージ・コピー・データ・セットは、この数には含まれません。

データを特定の期間維持する場合は、RECOVPD パラメーターを使用します。

関連資料:

- RECOVPD パラメーターの詳細については、『イメージ・コピー・データ・セットおよび GENMAX のリカバリー期間』を参照してください。
- REUSE、NOREUSE、GENMAX、および RECOVPD の詳細については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」の INIT.DBDS コマンドを参照してください。

イメージ・コピー・データ・セットおよび GENMAX のリカバリー期間

DBDS を RECON データ・セットに登録するとき、DBRC が記録するイメージ・コピーの最大世代数を、INIT.DBDS コマンドの GENMAX パラメーターで指定します。この数を超えると、DBRC は、最も古いイメージ・コピーに関連する情報を廃棄します。

例えば、毎日イメージ・コピーを取り、バックレベルのイメージ・コピーを 4 日間保持する場合は、GENMAX に 4 を指定します。ただし、緊急事態によっては、さらに頻繁にバックアップ・コピーを取ることができます (例えば 1 日に 3 回など)。

DBRC が以前のコピーに関係のある情報を破棄しないようにするためには、オプションのリカバリー期間パラメーター (RECOVPD) を指定することによって、情報を保持したい日数を指示してください。GENMAX の限界に達しても最も古いイメージ・コピー・レコードの RECOVPD がまだ経過していない場合は、DBRC は警告メッセージ (DSP0065I) は出しますが、レコードは廃棄しません。DSP0065I 警告メッセージが頻繁に出る場合は、CHANGE.DBDS コマンドを使用して GENMAX または RECOVPD の値を調整する必要があるかもしれません。

リカバリー期間 とは、情報が RECON データ・セットの中で保守される最低限の期間のことです。例えば、DBDS や DEDB エリアのリカバリー期間が 14 日の場合は、DBRC は十分なリカバリー世代情報を最低限 14 日間は保守します。

DBDS または DEDB エリアに GENMAX と RECOVPD の両方が指定されている場合、DBRC は、イメージ・コピー・データ・セットを再利用すべきか、削除すべきかを決める際に、その両方を考慮します。

以下の表では、REUSE パラメーターを指定して定義された DBDS またはエリアに GENMAX と RECOVPD の両方が指定されている場合の、GENJCL.IC 処理の結果を示します。

表 45. GENMAX および RECOVPD を REUSE で指定した場合の GENJCL.IC 処理の結果

イメージ・データ・セットの数	使用中のイメージ・コピーの数	最も古いイメージ・コピーの経過時間	GENJCL の結果
=GENMAX	=GENMAX	<RECOVPD	失敗 (DSP0063I)
>GENMAX	=GENMAX	<RECOVPD	使用可能な IC DS を使用 (DSP0065I)
=GENMAX	=GENMAX	>RECOVPD	最も古い IC DS を再利用
>GENMAX	=GENMAX	>RECOVPD	使用可能な IC DS を使用

以下の表では、NOREUSE パラメーターを指定して定義された DBDS またはエリアに GENMAX と RECOVPD の両方が指定されている場合の、イメージ・コピー・ユーティリティの実行結果を示します。

表 46. GENMAX および RECOVPD を NOREUSE で指定した場合の GENJCL.IC 処理の結果

イメージ・コピーの数	最も古いイメージ・コピーの経過時間	ユーティリティ EOI の結果
=GENMAX	>RECOVPD	最も古いイメージ・コピーを削除
=GENMAX	<RECOVPD	削除しない (DSP0064I)
<GENMAX	該当なし	削除しない

CHANGE.DBDS コマンドを出して、既存の値より小さい GENMAX および RECOVPD の値を新たに指定すると、リカバリー期間を過ぎた使用済みのイメージ・コピー・データ・セットはいずれも、残りのイメージ・コピー・データ・セットの数が指定された GENMAX の値と等しくなるまで削除されます。

DELETE.IC コマンドを出すと、RECOVPD または GENMAX には関係なく、指定したイメージ・コピー・データ・セットのレコードはいずれも削除されます。

関連資料:

 CHANGE.DBDS コマンド (コマンド)

変更累積データ・セットのリカバリー期間と GRPMAX

CA グループを RECON データ・セットに登録するとき、INIT.CAGRP コマンドの GRPMAX キーワードを使用して、記録する DBRC の変更累積世代の最大数を指定します。この数を超えると、DBRC は、最も古い変更累積実行レコードに関連する情報を破棄します。

例えば、変更累積を毎日実行し、30 日間のバックレベルの変更累積を保持したい場合は、GRPMAX 値 30 を指定します。ただし、場合によっては、変更累積をより頻繁に実行することがあります。

以前の変更累積に関連する情報を DBRC が破棄しないようにするためには、オプションのリカバリー期間キーワード (RECOVPD) を指定して、情報を保持する日数を示します。GRPMAX 限界に達しても、最も古い変更累積レコードの RECOVPD の有効期限が切れていない場合、DBRC は情報メッセージ (DSP1232I) を出し、レコードは破棄しません。DSP1232I メッセージが頻繁に出される場合は、CHANGE.CAGRP コマンドを使用して GRPMAX 値または RECOVPD 値を調整する必要がある可能性があります。

リカバリー期間とは、情報が RECON データ・セットの中で保守される最低限の期間のことです。例えば、CA グループのリカバリー期間が 30 日の場合、DBRC は十分な変更累積世代情報を少なくとも 30 日間保持します。DBRC は、変更累積実行レコードの停止時刻を使用して、リカバリー期間の有効期限が切れたかどうかを判別します。

CA グループに対して GRPMAX と RECOVPD の両方のキーワード値が指定されている場合、DBRC は両方の値を使用して、変更累積実行データ・セットを再使用するか削除するかを判別します。

以下の表は、REUSE キーワードで定義されている CA グループに GRPMAX と RECOVPD の両方の値が指定されているときに変更累積ユーティリティを実行した場合の結果を示しています。

表 47. REUSE とともに GRPMAX および RECOVPD が指定されている場合の変更累積ユーティリティの実行結果


変更累積データ・セットの数	使用中の変更累積データ・セットの数	最も古い変更累積データ・セットの経過時間	GENJCL の結果
= GRPMAX	= GRPMAX	<RECOVPD	失敗 (DSP1229A)
> GRPMAX	= GRPMAX	<RECOVPD	使用可能な CA データ・セットが使用される (DSP1232I)
= GRPMAX	= GRPMAX	>RECOVPD	最も古い CA データ・セットが再使用される
> GRPMAX	= GRPMAX	>RECOVPD	使用可能な CA データ・セットが使用される

以下の表は、NOREUSE キーワードで定義されている CA グループに GRPMAX と RECOVPD の両方の値が指定されている場合の変更累積ユーティリティの実行結果を示しています。

表 48. NOREUSE とともに GRPMAX および RECOVPD が指定されている場合の変更累積ユーティリティの実行結果

変更累積データ・セットの数	最も古い変更累積データ・セットの経過時間	ユーティリティのジョブ終了の結果
= GRPMAX	>RECOVPD	最も古い CA データ・セットを削除する
= GRPMAX	<RECOVPD	削除しない (DSP1233I)
< GRPMAX	N/A	削除しない

関連資料:

 INIT.CAGRP コマンド (コマンド)

イメージ・コピー・データ・セットの再利用

DBRC によって、古いイメージ・コピー・データ・セットを再利用することができます。INIT.DBDS コマンドの REUSE パラメーターを使用すると、将来の利用のためにイメージ・コピー・データ・セットを定義することができるほか、DBRC がイメージ・コピー・データ・セットを再利用できるようになります。イメージ・コピー・データ・セットの再利用とは、同じ名前、同じボリューム、そして同じ物理スペースが新しいイメージ・コピーで使用されることを意味します。

制約事項: DFSMSdss 高速複製 (データベース・イメージ・コピー 2 ユーティリティによって呼び出されるもの) は、REUSE パラメーターをサポートしていません。したがって、このトピックの情報は高速複製イメージ・コピーには当てはまりません。

IMS イメージ・コピー・ユーティリティの 1 つを実行する際に次の条件がすべて満たされているとき、REUSE 属性を持つ DBDS またはエリアの最も古いイメージ・コピー・データ・セットが自動的に再利用されます。

- GENMAX の現在の値と同じ数のイメージ・コピー・データ・セットが RECON データ・セットに記録されている。GENMAX の現在の値を見るには、LIST.DBDS コマンドを使用します。
- 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・ユーティリティが、この DBDS のすべてのイメージ・コピー・データ・セットを使用した。使用できるイメージ・コピー・データ・セットがありません。
- 最も古いイメージ・コピーがリカバリー期間を超えている。
- IMS イメージ・コピー・ユーティリティの 1 つを実行する際、使用可能なイメージ・コピーがあればそれが自動的に使用される。GENMAX にすでに到達していて、しかも最も古いイメージ・コピーがリカバリー期間を過ぎているか、リカバリー期間がない場合には、最も古いイメージ・コピーのデータ・セットが使用可能になります。

GENJCL.IC コマンドを使用してデータベース・イメージ・コピー・ユーティリティのジョブを生成する場合、再利用すべきイメージ・コピー・データ・セットが自動的に選択されます。イメージ・コピー・データ・セットの数が GENMAX の値より小さく、イメージ・コピー・データ・セットがすべて使用されてしまった場合は、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、DBDS またはエリアに対してさらにイメージ・コピー・データ・セットを定義する必要があります。リカバリー期間を使用する場合は、イメージ・コピー・データ・セットの数を GENMAX 値より大きくする必要があります。

データベース・イメージ・コピー 2 ユーティリティでは、最大 4 つの出力イメージ・コピー・データ・セットを作成することができます。ただし、REUSE として定義された DBDS の場合、GENJCL.IC コマンドを実行すると、DBRC が生成する JCL の出力コピーは 1 つまたは 2 つのみです (定義できるイメージ・コピー・デ

ータ・セットのコピーの数は 1 つまたは 2 つのみのため)。REUSE として定義された DBDS を処理するためにデータベース・イメージ・コピー 2 ユーティリティで GENJCL.IC コマンドを使用し、2 つより多い出力コピーが必要な場合は、ジョブを実行する前に生成された JCL を変更してください。

DBRC は、イメージ・コピーの出力には必ず最も古い DASD データ・セットを使用するように JCL を生成します。DBRC はテープ・ボリュームでも同じ機能を備えています。ただし、既存のテープ・ライブラリー技法を調べて、矛盾がないことを確認する必要があります。

HSSP イメージ・コピー

高速順次処理 (HSSP) イメージ・コピー (IC) 処理を行うには、データベースを DBRC に登録し、さらに定義済み IC データ・セットをリサイクルするために、1 つのエリアを REUSE 属性で登録する必要があります。HSSP によって作成されるイメージ・コピーは、並行イメージ・コピーです。

HISAM コピー (DFSURUL0 および DFSURRL0)

HISAM 再編成アンロード・ユーティリティ (DFSURUL0) を使用してデータベースのバックアップ・コピーを作成すると、1 回のパスで HISAM データベース全体を処理することができます (イメージ・コピー・ユーティリティは、データ・セットまたはエリアを 1 つずつ処理します)。このアンロード・ユーティリティ (DFSURUL0) は、データベースをコピーしながら同時に再編成も行います。

アンロード・ユーティリティ (DFSURUL0) はデータベースを再編成するため、通常のオンライン操作を再開する前に HISAM 再編成再ロード・ユーティリティ (DFSURRL0) を使用して、データ・セットを再ロードする必要があります。ログインは、アンロードと再ロードの間に行われますが、もとのデータ・セット編成を反映します。

HISAM ユーティリティを使用してバックアップ・コピーを作成する場合は、直ちに再ロードしてください。さもないと、実際のデータベースとバックアップ・データベースがミスマッチになります。

再ロード・ユーティリティ (DFSURRL0) は DBRC に通知します。アンロード・ユーティリティは、各データ・セットの再編成されたコピーを作成します。次に再ロード・ユーティリティは、再編成されたコピーから各データ・セットを再ロードし、DBRC を通じて、それぞれのデータ・セットごとに REORG および IC レコードを RECON データ・セットに作成します。IC レコードは、アンロード・ユーティリティからの出力のデータ・セットと、再ロード・ユーティリティへの入力のデータ・セットを指します。データベースが再編成された後、DBDS は、そのデータ・セットのイメージ・コピーが作成された場合にのみ許可されます。

アンロード操作と再ロード操作の間に、データベースの更新が行われなければならない必要があります。アンロード操作後で再ロード操作の前に行われたデータベースの更新は、再ロード操作によって消去されます。さらに、ログに記録された変更レコードはもとの編成を反映しているため、このログ・レコードを使用して後でリカバリーを行うと、データベースに損傷を与えます。

次の方式のいずれかを使用すれば、再編成時に共用データベースへのアクセスを防ぐことができます。

- オンライン IMS サブシステムから、再編成されるデータベースに対してグローバル /DBRECOVERY コマンドを出す。これにより、再編成とリカバリー以外の、後続の許可はすべて防止されます。再編成中にリカバリー・ユーティリティを実行しないようにしてください。
- CHANGE.DB コマンドの NOAUTH パラメーターを指定することにより、手作業で RECON データ・セットを更新する。これにより、再編成およびリカバリー・ユーティリティへの許可を除いて、後続の許可はすべて防止されます。再編成中にリカバリー・ユーティリティが実行されていないことを確認してください。再編成処理の完了後、再編成された直後のデータベースに対して CHANGE.DB コマンドの AUTH パラメーターを指定することにより、手作業で RECON データ・セットを更新します。

標準外イメージ・コピー・データ・セット

データベース・イメージ・コピー・ユーティリティ以外の方法でも DBDS のバックアップ・コピーを作成することができます。

例えば、DBDS のあるボリュームのコピーを作成することができます。DBRC は、このような標準外イメージ・コピー・データ・セットの存在を RECON データ・セットに自動的に記録しません。NOTIFY.UIC コマンドを使用して、このようなデータ・セットが存在することを RECON データ・セットに通知してください。この情報が RECON データ・セットに記録されない場合、DBRC は、DBDS に加えられた変更について、その後の情報の解釈を誤る可能性があります。REUSE オプションを指定して定義されている DBDS に対して、NOTIFY.UIC コマンドを出すことはできません。

DBDS または DEDB エリアを標準外イメージ・コピー・データ・セットによってリカバリーする前に、以下のステップを実行します。

1. /DBR (NOFEOV の指定なし) を使用してデータベースをクローズする。標準外イメージ・コピー (UIC) からデータ・セットをロードし、(RCVTIME を指定した NOTIFY.RECOV を出すことによって) そのイベントを RECON データ・セットに記録する。
2. (GENJCL.RECOV コマンドに USEDDBS または USEAREA を指定して DBRC を実行するか、または DBRC JCL の DFSDUMP DD DUMMY ステートメントを実行することによって) UIC 以降に作成されたログからの変更レコードを適用する。

ステップ 2 ではイメージ・コピーが使用されないため、DBRC ではリカバリーの範囲外の変更を含むログの処理はできません。リカバリーの範囲は、タイム・スタンプ・リカバリー・レコード RECOV TO (イメージ・コピー時刻) および RUNTIME 値によって定義されます。

標準外イメージ・コピー・データ・セットによる DBDS または DEDB エリアのリカバリーの詳細については、「IMS V15 データベース管理」を参照してください。

バックアップ・コピーの頻度および保存

一般に、頻繁にコピーをとれば、それだけリカバリーに要する時間は少なくてすみます。古いコピーが時間をさかのぼって保存されていればいるほど、過去にさかのぼってリカバリーすることができます。(プログラム・ロジック・エラーは何週間も発見されない場合があるということを思い出してください。) 逆に、新しいコピーの作成にはそのつど作業が必要となり、古いコピーを保管すればそれだけ追加のリソースを使用することになります。

データベースのバックアップ・コピーを作成する際は、次の点を考慮してください。

- 新しいコピーはどれほどの頻度で作成すべきか
- 古い (バックレベル) コピーはどれほどの期間保持すべきか

これらの質問に対する的確な回答はありません。

確固とした指針となるのは、以下に示すことのみです。

- データベースが複数のデータ・セットから構成されている場合は、すべてのデータ・セットを同時にコピーするようにしてください。
- データベースを再編成する場合は、そのデータベースの新しいバックアップ・コピーを直ちに作成してください。(これは、オンライン DEDB または HISAM 再編成の場合は不要です。)
- 新しいデータベースを作成する場合は、そのデータベースのバックアップ・コピーを直ちに作成してください。
- タイム・スタンプ・リカバリーを実行する場合は、後でリカバリーに使用するためのバックアップ・コピーを作成してください。

ログ・レコードの変更累積

IMS を実行するにしたがって、保管される SLDS または RLDS の数は増加していきます。このような保管ボリュームを使用して逸失または損傷したデータベースをリカバリーすることができますが、保管ボリュームを未加工の形で用いるのでは非効率です。

これは、以下の理由から非効率です。

- SLDS または RLDS にはそれぞれ、システム全体のアクティビティのレコードと、全データベースのすべてのデータ・セットが含まれています。ところが、データベースをリカバリーする場合は、単一のデータ・セットについてのみリカバリーを行います。したがって、SLDS または RLDS にある多くのものは不要です。
- SLDS または RLDS では、1 つのレコードに加えられた変更をそれぞれ発生順に記録します。データ・セットを最後にバックアップ・コピーしてから、その後レコードが 100 回変更されたとすると、SLDS または RLDS にはそのような記録が 100 個入ることになります。しかし、リカバリーの際、ユーザーが関心するのは、データ・セットが破損した時点でデータが持っていた値だけです。それ以前の 99 回の変更は無関係です。

データベース・リカバリー・ユーティリティでは、変更レコードを発生順に入力する必要がありますが、データベースが共用されている場合は、ユーティリティへの入力ができなくなるような形で変更レコードが別々のログ・データ・セットに分散されている可能性があります。このログ・データが正しくマージされなかった場合には、DBRC GENJCL.RECOV コマンドまたは DB リカバリー・ユーティリティの実行が失敗します。そのような障害が生じると、変更累積を実行する必要があることを通知するメッセージが出されます。

累積した SLDS または RLDS (変更累積) の圧縮

データベース変更累積ユーティリティは、累積したログ・データ・セット全体をマージし、圧縮して、あらかじめソートする方法を提供します。

このユーティリティは、以下のことを行います。

- 異なるサブシステムからの更新をマージする
- データベースのリカバリーに関連したログ・レコードだけを取り出す
- それらのレコードをデータベース内のデータ・セット別にソートする
- データ・セットごとに変更された各部分の最新の状態だけを保管する

以下の図では、データベース変更累積ユーティリティが、1 つの IMS システムの異なる 3 つの SLDS (A、B、および C) からの関連ログ・データ (a、b、および c) を、1 つの変更累積データ・セット・レコードにマージする方法を示します。

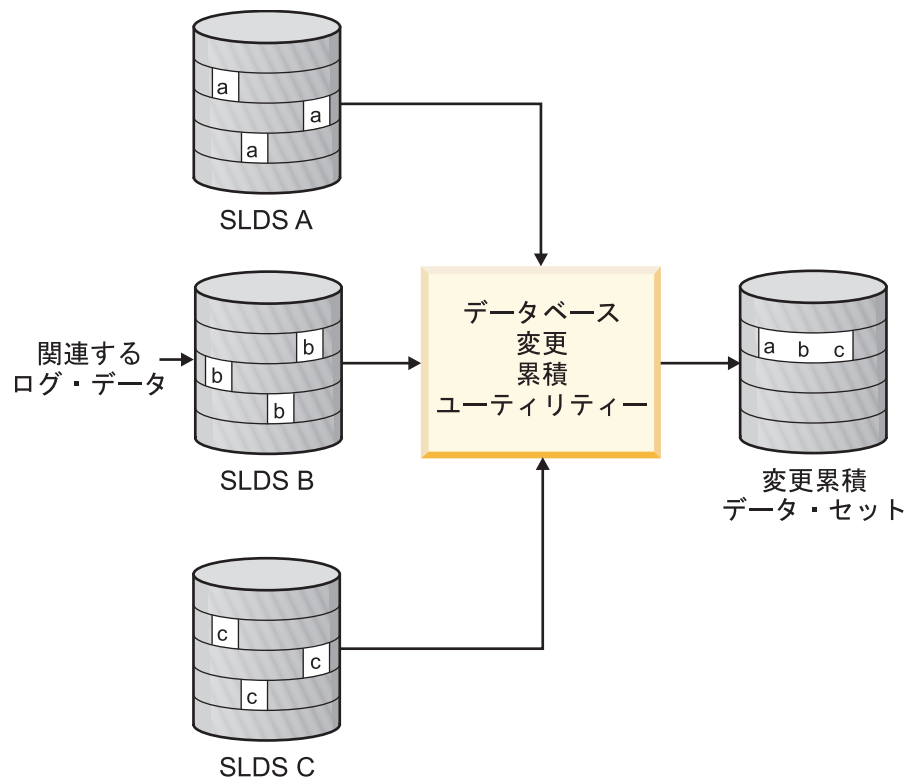


図 60. 変更累積による異なるデータ・ストリームからの再実行データの処理

どのようなときに変更累積が必要か

当該時間枠に、1 つのシステムだけがデータベースを更新した場合、データベース変更累積ユーティリティを実行する必要はありません。このユーティリティを定期的に使用すると、必要となるデータベース・リカバリーはいずれも処理が速まります。その代わりに、リカバリーが必要になった場合にのみデータベース変更累積ユーティリティを実行することもできます (DB リカバリー・ユーティリティの実行の直前に)。

各種のログに非並行データ・セット更新情報が存在する場合は、変更累積が不要となります。ログが DB リカバリー・ユーティリティに順次に入力される場合は、データベースの変更が正しい順序で受け取られます。

各種のログに並行データ・セット更新情報が存在する場合は、変更累積が必要となります。変更レコードが正しい順序で見られるような方法で、ログを DB リカバリー・ユーティリティに入力することはできません。

関連資料: データベース変更累積ユーティリティを実行する際の詳細な説明については、「IMS V15 データベース・ユーティリティ」を参照してください。

データベース変更累積ユーティリティへの入力

保管された SLDS および RLDS に加えて、直前の変更累積データ・セットとその他の IMS ログ・ボリュームを入力として使用することができます。このユーティリティは、累積された変更を新しい変更累積データ・セットに書き込みます。

すべてのログ・ボリュームまたはログ・ボリュームのサブセットを、データベース変更累積ユーティリティへの入力として指定することができます。ログ・ボリュームのサブセットを指定した場合、DBRC は、そのサブセットがそれぞれの DBDS またはエリアごとに完全であるかどうかを判別します。次の条件がすべて満たされる場合、ログ・ボリュームのサブセットは DBDS またはエリアに対して完全であると言えます。

- サブセット内の最初のボリュームが一番古いボリュームであり、DBDS に変更が行われている可能性があるが、その変更は最新の変更累積または最新のイメージ・コピーには組み込まれなかった。
- 残りのボリュームは順になっている。
- データ共用環境では、更新のあったすべての更新サブシステムからのログも、DBDS のいずれのオープン・データ・ストリームからのログも組み込まれている。

DBRC LIST.CAGRP コマンドは、変更累積グループのそれぞれの DBDS ごとのログ・サブセットが完全であるかどうかを示します。

ログ・ボリュームのサブセットが完全あるいは不完全のいずれの場合でも、後でデータベース変更累積ユーティリティを実行するときに、変更累積データ・セットを入力として使用することができます。ただし、変更累積データ・セットを DB リカバリー・ユーティリティへの入力として使用できるのは、それが完全なログ・サブセットを表している場合のみです。

推奨事項: DBRC を使用している場合は、GENJCL.CA コマンドを使用して変更累積処理を実行してください。このコマンドにより正しい JCL が作成され、未処理の

ログ・データ・セットがすべて組み込まれます。その代わりに独自の JCL を使用する場合は、実行する前に変更累積の指定を検査してください。

変更累積レコードの有効な開始点を作成するためには、指定されたデータベース・データ・セットのイメージ・コピーが必要であり、それを RECON データ・セットに示す必要があります。

有効な最後のイメージ・コピー以降の変更はすべて、このユーティリティによって収集されます。最後のイメージ・コピー以降にタイム・スタンプ・リカバリーが行われた場合、作成された変更累積は将来のリカバリーに無効になります。GENJCL.CA によって、あるいはユーティリティの実行によって生成されるエラー・メッセージはありません。

いつでも、有効なログ・サブセットを用いて変更累積ユーティリティを実行し、データを最小限に削減することができます。

変更累積ユーティリティへの入力として最新のイメージ・コピーが使用され、そのイメージ・コピーが並行イメージ・コピーの場合、アクティブ・アプリケーションによってすでにデータベースに加えられた変更は、データ・セットに物理的に書き込まれなかったことが考えられるため、コピーから欠落している可能性があります。ただし、これらの変更はログには書き込まれています。この場合、すべての変更が適用されることを確実にするために、ログ上で前の時点に戻る必要があります。どの程度戻るかは、データベースのタイプおよびどのイメージ・コピー・ユーティリティが使用されたかによって異なります。

変更累積ユーティリティを開始するために選択された時刻指定は、除去時刻 と呼ばれます。

関連資料: データベース変更累積ユーティリティの詳細については、「IMS V15 データベース・ユーティリティ」を参照してください。

変更累積グループ

データベース変更累積ユーティリティを使用して、リカバリー可能な、登録済みデータベースの一部または全部を変更累積 (CA) グループとして処理できます。例えば、DBDS を次のグループに編成することができます。

- アプリケーション関連データベース
- 物理データベース・クラスター
- 論理データベース・クラスター
- 揮発性 (クリティカル・データ) データベース

CA (変更累積) グループを作成した後、グループのメンバーを追加または削除することができます。データベースは 1 つの CA グループでのみ、メンバーになることができます。ある CA グループから別の CA グループにメンバーを移動するためには、まずその現行 CA グループからメンバーを削除し、その後、メンバーを新規 CA グループに追加する必要があります。

推奨事項: CA グループ・メンバーのイメージ・コピーを作成する場合は、先にそのメンバーを新しい CA グループに移動してください。変更累積ユーティリティーは、最新の良好なイメージ・コピーに基づいて、リカバリーに必要なメンバーのすべてのログを処理します。

変更累積グループには 1 から 32767 のメンバーを入れることができますが、そのメンバーは登録済みの DBDS またはエリアであることが必要です。

INIT.CAGRP を使用して変更累積グループを作成します。変更累積グループを変更する場合は、CHANGE.CAGRP を使用します。DBDS またはエリアは 1 つの変更累積グループにしか属することができません。

DBRC は、変更累積グループによってのみ変更累積ユーティリティーをサポートします。同様に、GENJCL.CA は変更累積グループについてのみ JCL を生成します。DBRC 検査出口ルーチンは、変更累積グループ全体について検査します。

DBRC=YES が有効であっても、変更累積グループに属していない DBDS について変更累積ユーティリティーを実行することができます。ただし、DBRC はユーティリティーへの入力を検査せず、その出力の記録も行いません。

将来の利用のための変更累積データ・セットの定義

INIT.CAGRP コマンドで REUSE キーワードを使用すると、データ・セットの「プール」を定義して、変更累積ユーティリティーからの出力を受け取ることを DBRC に通知することができます。INIT.CA コマンドを使用してデータ・セットを定義します。定義できる変更累積データ・セットの最大数は、グループを定義した INIT.CAGRP コマンドの GRPMAX パラメーターの値までです。

REUSE パラメーターを指定して変更累積グループを定義し、さらに GENJCL.CA コマンドを使用してデータベース変更累積ユーティリティーのジョブも生成すると、データ・セットの再利用が生じる場合があります。この変更累積グループで使用できる変更累積データ・セットがすべて使用されてしまい、変更累積データ・セットの最大数に達した場合、データベース変更累積ユーティリティーがこのグループで次に実行されるときに、最も古い変更レコードの入っている変更累積データ・セットが再利用されます。変更累積データ・セットを再利用するということは、そのデータ・セット名、ボリューム、および物理スペースが、空の変更累積データ・セット用のものであるかのように使用されるということを意味します。

NOREUSE を指定して変更累積グループを定義した場合:

- このユーティリティーの出力データ・セット名を、GENJCL.CA コマンドで使用した骨組み JCL メンバーか、もしくはユーザーが作成した JCL で指定する必要があります。
- 変更累積ユーティリティーを実行する際、GRPMAX で指定されたデータ・セットの数に達していると、DBRC は、最も古い変更レコードが入っている (RECON データ・セットからの) 変更累積データ・セットのレコードを削除します。データ・セットそのものは消去されません。データ・セットを保持したい場合は、DBRC はもはやその存在を認識していないため、そのデータ・セットを手作業でスクラッチするか、モニターする必要があります。

関連資料: 変更累積 JCL の指定の詳細については、「IMS V15 データベース・ユーティリティー」および「IMS V15 コマンド 第 1 巻: IMS コマンド A-M」を参照してください。

DBDS グループの考慮事項

DBDS グループは、DBDS または DEDB エリアの、名前の付いた集合です。DBRC は DBDS グループごとに各種の操作を実行することができるため、グループの全メンバーについてコマンドを繰り返し出す必要はありません。

DBDS グループは以下のコマンドで指定することができます。

- GENJCL.IC
- GENJCL.OIC
- GENJCL.RECEIVE
- GENJCL.RECOV
- GENJCL.USER
- LIST.DBDS
- LIST.HISTORY

コマンドで DBDS グループが指定されると、DBRC はその DBDS グループのそれぞれのメンバーごとにそのコマンドを呼び出します。例えば、給与計算のような特定のアプリケーション用に DBDS グループを持っているとします。例えば、タイム・スタンプ・リカバリーを実行する場合、データベースの特定アプリケーションのすべての DBDS を同一点までリカバリーする必要があります。GENJCL.RECOV コマンドで DBDS グループを指定すれば、このコマンドを 1 回呼び出すだけで、すべての DBDS をリカバリーすることができます。

CA グループを DBDS グループとして指定することもできます。その場合、DBRC は CA グループのそれぞれのメンバーごとにこのコマンドを実行します。

DBDS グループは必要な数だけ定義することができます。1 つのグループ内の DBDS は最大 32767 です。グループ内の DBDS はすべて RECON データ・セットに登録する必要があります。1 つの DBDS を複数の DBDS グループに所属させることができます。

データベースは、GENJCL および LIST コマンドの暗黙の DBDS グループです。1 つのデータベースの DBDS またはエリアからなる DBDS グループは定義する必要はありません。データベース全体で操作するには、データベース名を指定し、DD 名は省略します。

次のコマンドは、DBDS グループの定義に影響を与えます。

- INIT.DBDSGRP
- CHANGE.DBDSGRP
- DELETE.DBDS
- DELETE.DBDSGRP
- LIST.DBDSGRP

DBDS グループには ILDS (間接リスト・データ・セット) および索引データ・セットを含めることができます。

関連資料: これらのデータ・セットが GENJCL コマンドに及ぼす影響については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

DB グループ

DB グループは、データベースまたは DEDB エリアの名前の付いた集合です。DB グループ名は /START、/STOP、/DBRECOVERY、および /RECOVER コマンドで指定することができます。それぞれのデータベースまたはエリアごとにこれらのコマンドを別々に出す必要はありません。

これらのコマンドで DB グループ名を指定すると、コマンドを出す回数を大幅に減らすことができます。DATAGROUP キーワードを使用して、DB グループ名を指定します。

DB グループには HALDB マスター名および HALDB 区画名も含めることができます。HALDB マスター名および 1 つ以上のその区画を持つ DB グループを使用してコマンドを出した場合の影響については、あらかじめ認識しておいてください。詳しくは、IMS V15 データベース管理を参照してください。

DB グループは必要な数だけ定義することができます。1 つのグループ内のデータベースまたはエリアは最大 32767 です。データベースやエリアは複数の DB グループに所属することが可能であり、RECON データ・セットに登録する必要はありません。

推奨事項: DBDS グループを DB グループとして使用できる場合であっても、可能な限りデータベース・グループを使用してください。DBDS グループを DB グループとして処理すると、必然的にオーバーヘッドが増大します。

DB グループは DBDS グループの形式であるため、DBDS グループ・レコードを使用して RECON データ・セットに保管されます。次のコマンドは、DB グループの定義に影響を与えます。

- INIT.DBDSGRP
- CHANGE.DBDSGRP
- DELETE.DBDSGRP
- LIST.DBDSGRP

DBRC グループ

ある IMSplex 環境内で複数の DBRC が同じ RECON データ・セットにアクセスする場合、それらの DBRC は 1 つの DBRC グループと見なされます。

このグループは、以下のとおりです。

- DBRC グループ ID によって識別される。
- 順次モードまたは並列モードで共用 RECON データ・セットにアクセスできる。

- IMS V10 以上でサポートされる。

DBRC グループ ID は、CHANGE.RECON コマンドで指定できます。 DBRCGRP は、QUERY IMSPLEX コマンドでサブタイプ列に表示できます。

制約事項: 同じ IMSplex 内で異なる RECON データ・セットを使用する DBRC は、並列 RECON データ・セット・アクセスを使用している場合、異なる DBRC グループ ID を使用する必要があります。

推奨事項: IMSplex 内で順次 RECON アクセスを使用している場合、同じ IMSplex 内で異なる RECON データ・セットを使用する DBRC は、同じ DBRC グループ ID を使用できますが、それは推奨されません。1 つの RECON データ・セットがエラーを検出した場合、同じ DBRC グループ ID を持つ IMSplex 内のすべての DBRC インスタンスによって自動 RECON 損失通知処理が実行されるため、エラーを起こした RECON データ・セットを使用していない DBRC インスタンスに、処理オーバーヘッドが追加されます。異なる DBRC グループ ID を使用すれば、この追加の処理オーバーヘッドが回避されます。

HALDB オンライン再編成に関する DBRC の考慮事項

次のトピックでは、HALDB オンライン再編成に関する DBRC の考慮事項について説明します。

次の DBRC コマンドは HALDB オンライン再編成をサポートします。

- CHANGE.DB
- GENJCL.IC および GENJCL.OIC

GENJCL.IC コマンドと GENJCL.OIC コマンドで事前に JCL を生成し、その JCL を即時には実行しないでおくことが可能です。このため、DBDS のアクティブ・セットがその一時的な期間中に変更される可能性があります。

GENJCL では、コピーされる HALDB DBDS 用の DD ステートメントは生成されなくなっています。これは、データベース・イメージ・コピー・ユーティリティーとデータベース・イメージ・コピー 2 ユーティリティーでは動的割り振りが推奨されているためです。SYSIN 制御ステートメントが、どの DBDS、すなわち、データ・セットの A から J セットまたは M から V セットのいずれをコピーするかを示します。これにより、イメージ・コピー・ユーティリティーはデータ・セットの対応するアクティブ・セット (A から J または M から V) をコピーします。

- INIT.DB
- NOTIFY.IC、NOTIFY.REORG、および NOTIFY.UIC

HALDB オンライン再編成カーソルがアクティブの場合 (OLREORG CURSOR ACTIVE =YES)、または終了されている場合は、イメージ・コピーを取ることはできません。HALDB オンライン再編成カーソルがアクティブの間に NOTIFY.IC コマンドと NOTIFY.UIC コマンドでイメージ・コピーの記録を試みると、これらのコマンドは失敗します。

HALDB オンライン再編成では、いくつかのレコードに依存して、オンライン再編成に関連する情報とアクティビティをトラッキングします。このセクションでは、HALDB オンライン再編成の影響を受けるいくつかのレコードについて説明します。

- HALDB マスター・レコード
- 区画 DB レコード

HALDB 区画の DB レコードには、HALDB オンライン再編成の情報が入っています。HALDB 区画の DB レコード内にあるフィールドの完全な説明については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」の『レコード・タイプ別の RECON リストに存在するフィールド (Fields present in a listing of a RECON by record type)』を参照してください。DB レコードに含まれる内容は、次のとおりです。

- データベースがオンライン再編成可能かどうかを示すフィールド (ONLINEREORG CAPABLE=)。このフィールドは HALDB マスター DB レコードの中にもあります。次の場合、このフィールドはリセットできません。
 - オンライン再編成がアクティブである
 - M から V および Y データ・セットのいずれかがアクティブである
 - HALDB のいずれかの区画が許可されている

これらの条件が存在せず、フィールドをリセットできる場合、RECON データ・セット内の M から V および Y データ・セットの情報はすべて削除されます。

- HALDB オンライン再編成カーソルがアクティブかどうかを示すフィールド (OLREORG CURSOR ACTIVE=)。
- HALDB オンライン再編成を現在実行中の IMS の SSID (OLRIMSID=)。
- 区分に対してアクティブな DBDS セット (1 つ以上) を示すフィールド (ACTIVE DBDS=)。

- DBDS レコード

M から V および Y の DBDS レコードは、対応する A から J、L、および X の DBDS レコードと同じ属性を指定して、RECON データ・セット内で定義されます。M から V および Y の DBDS レコードは、A から J および X の DBDS と同じ CAGROUP に追加されます。同様に、M から V および Y の DBDS レコードは、A から J の DBDS レコードがある各 DBDS グループに追加されます。

属性の類似点に関する例外には、DD 名、データ・セット名、および DSID があります。M から V および Y データ・セットの DSID は対応する A から J、L、および X データ・セットと同じですが、例外として、その X'80' ビットがオンに設定されています。RECON データ・セットをリストする目的で、X'80' ビットは無視されます。

以下の表では、A から J、L、および X データ・セットと、M から V および Y データ・セットに関連したデータ・セットの ID (DSID) DCB 番号を示します。

表 49. A から J, L, および X データ・セットと、M から V および Y データ・セットの DSID DCB 番号

DSID DCB 番号	A から J, L, および X データ・セット		M から V および Y データ・セット	
	PHIDAM	PHDAM	PHIDAM	PHDAM
1	A	A	M	M
2				
3	L	L		
4		B		N
5	X	C	Y	O
6	B	D	N	P
7	C	E	O	Q
8	D	F	P	R
9	E	G	Q	S
10	F	H	R	T
11	G	I	S	U
12	H	J	T	V
13	I		U	
14	J		V	

• REORG レコード

HALDB 区画 DBDS の REORG レコードには、HALDB オンライン再編成の停止時刻が含まれます。データベースのオフライン再編成を実行するには、/DBR コマンドを実行してそのデータベースをオフラインに変更する必要があります。このため、オンライン再編成の開始時刻のみが重要になります。HALDB オンライン再編成の場合はデータベースがオンラインなので、処理の所要時間が重要です。オンライン再編成の停止時刻は REORG レコードに記録され (STOP=timestamp)、それがオンライン再編成であることと、このオンライン再編成をリカバリーへの入力として使用できるかどうかを示す情報がレコードと一緒に記録されます。

関連資料: REORG レコード内に存在するフィールドの詳細については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

REORG レコードは、A から J, L, および X の DBDS と、M から V および Y の DBDS の両方の下に記録されます。出力データ・セットだった DBDS の REORG レコードは、そのデータ・セットをリカバリーの入力として使用できることを示します。入力データ・セットだった DBDS の REORG レコードは、オンライン再編成の完了後にそのデータ・セットをリカバリーに使用できないことを示します。

オンライン再編成に影響されるリカバリー関連レコードの保守

リカバリーの目的で、A から J まで、および M から V までのデータ・セットは、RECON データ・セット内で 1 つの DBDS 論理セットとして扱われます。

これら 2 つの DBDS ヘッダーのリカバリー情報は、1 つの論理 DBDS であるかのように保守されます。例えば、GENMAX、RECOVPD、および CAGRP の値は同じです。アクティブ DBDS に行われた変更はすべて、非アクティブ DBDS にも行われます。

各区分データ・セット間には 1 対 1 の関係が存在します。例えば、A は M と相互関連し、B は N と相互関連し、C は O と相互関連する、などのようになります。また、M から V データ・セットは、それに対応する A から J データ・セットと同じ属性を指定して RECON データ・セット内で定義されます。オンライン再編成の開始前にどのセットがアクティブだったかに応じて、これらのデータ・セット (A から J、L および X、または M から V および Y) のいずれに対しても、任意の時点でオンライン再編成を実行することができます。このため、データ・セット A と M はオンライン再編成の実行中は 1 つの論理データ・セットとなります。

イメージ・コピーが取られた後、DBRC は RECON データ・セットから無関係のリカバリー関連レコード (IMAGE、ALLOC、RECOV、および REORG) をすべて除去します。これらの無関係のレコードは、その DBDS のリカバリー生成の最大数 (GENMAX) 設定とオプションのリカバリー期間 (RECOVPD) 設定によって決定されます。

例えば、データ・セット A (および M) の GENMAX 値が 2 であり、データ・セット A に 2 つのイメージ・コピーがあるとします。データ・セット M がアクティブ・データ・セットになり、1 つのイメージ・コピーが取られると、イメージ・コピーのクリーンアップ処理によって、すべての無関係のリカバリー・レコードを含め、最も古いイメージ・コピーがデータ・セット A から削除されます。データ・セット M のイメージ・コピーがもう 1 つ取られると、データ・セット A のイメージ・コピーとリカバリー・レコードの残りが削除されます。データ・セット A のイメージ・コピーとリカバリー・レコードが削除されると、データ・セット A はリカバリー不能になるので注意してください。

オンライン再編成処理

オンライン再編成が開始されると、その再編成を担当する IMS の ID が区分に割り当てられます。

カーソル・アクティブ状況が RECON データ・セットに記録され、A から J データ・セットと M から V データ・セット間でコピーが始まると、以下の変更が発生します。

- 区分の DB レコードが、オンライン再編成がアクティブである (OLREORG CURSOR ACTIVE =YES) ことを示します。
- アクティブ DBDS の状況は、両方のデータ・セットがアクティブ (ACTIVE DBDS=A-J および M-V) であることを示します。
- オンライン再編成の開始時刻を示す REORG レコードが書き込まれます。停止時刻はゼロです。
- 出力データ・セットに関連する REORG レコードは、このデータ・セットがリカバリーの入力として使用できることを示すようになります。

- 入力データ・セットに関連する REORG レコードは、このデータ・セットがリカバリーに使用できることを示しません。DBRC はこのレコードを使用して、オンライン再編成の完了後、このデータ・セットをリカバリーできないことを確認します。
- M から V および Y データ・セットがまだ存在していない場合は、これらのデータ・セットが RECON データ・セット内で定義されます。
- A から J、L、および X の DBDS に変更累積グループが割り当てられている場合、M から V および Y の DBDS はそれと同じ変更累積グループに追加されます。
- M から V および Y の DBDS は、A から J、L、および X の DBDS があるいずれかの DBDS グループに追加されます。

これらの変更が行われた後にオンライン再編成は完了するはずですが、完了しない場合は、オフライン再編成を実行して、HALDB オンライン再編成設定をリセットする必要があります。

オンライン再編成は、すでに区分に対して実行中である場合は開始できません。ただし、オンライン再編成が終了している場合、すなわち、IMS ID が割り当てられていなければ、別の IMS がそのオンライン再編成を引き継ぐことができます。

再編成に割り当てられた IMS システムに対し、次のいずれかのオプションによりシステムが終了したときに所有権を解除するよう指示することができます。

- INITIATE OLREORG コマンドまたは UPDATE OLREORG コマンドの OPTION(REL)
- IMS.PROCLIB データ・セットの DFSDFxxx メンバーの DATABASE セクション内の RELOLROWNER=Y。

その場合は、別の IMS システムが、中断された再編成を再開することができます。

オンライン再編成処理が正常に完了すると、以下の変更が発生します。

- HALDB OLR 停止時刻を組み込むように REORG レコードが更新されます。
- 区分の DB レコードが変更されて、オンライン再編成がアクティブでない (OLREORG CURSOR ACTIVE = NO) ことを示します。
- アクティブ DBDS の状況が、新たに再編成されたデータ・セットに切り替えられます。
- オンライン再編成の IMS ID がクリアされます。

推奨事項: オンライン再編成が正常に完了したらできるだけ早く、HALDB 区分 DBDS のイメージ・コピーを作成してください。

制約事項: オンライン再編成カーソルがアクティブ (OLREORG CURSOR ACTIVE =YES) の間は、DBDS のイメージ・コピーを取ることはできません。この状態では、DBRC はイメージ・コピーが取れないようにします。

第 41 章 DBRC セキュリティー

RECON データ・セットは、IMS データベースの保全性を保つために重要です。したがって、DBRC コマンドのサブセットを実行する必要があるユーザーに対しては、そのサブセットのみへのアクセスを許可することを考慮してください。

オンライン領域では、IMS は出されるコマンドに対するいくつかの許可機能を提供します。IMS は、デフォルトで、マスター端末オペレーター (MTO) に /RMxxxxxx コマンド (/RMLIST を除く) を制限します。

ユーザーは、RACF (または同等のセキュリティー・プロダクト)、ユーザー出口ルーチン、またはこの両方を通じて、DBRC コマンド許可の制御を設定することができます。例えば、ログオン ID SMITH は、INIT.DB コマンドを、DBD PAYROLL に対しては実行できるが、DBD CUSTOMER に対しては実行できない、というアクセス許可を持つ、という場合があります。

HALDB 区画定義ユーティリティーは、RECON データ・セットでの IMS HALDB とそれらの区画の定義を管理できるようにする ISPF アプリケーションであり、以下のコマンドと同等の機能を提供します。

- INIT.DB
- INIT.PART
- CHANGE.DB
- CHANGE.PART
- CHANGE.DBDS
- DELETE.DB
- DELETE.PART
- LIST.DB

HALDB 区画定義ユーティリティーに対する許可は、これらのコマンドに定義されたものと同じリソースを使用して制御されます。ただし、1 つの例外は、CHANGE.PART リソースが CHANGE.DBDS リソースの代わりに使用されることです。

関連資料: HALDB 区画定義ユーティリティーの詳細については、「IMS V15 データベース・ユーティリティー」を参照してください。

DBRC コマンドおよび API 要求のセキュリティー

DBRC コマンドおよび DBRC API 要求の許可制御は、RACF (または同等のセキュリティー製品)、出口ルーチン、またはその両方を使用して確立することができます。また、必要に応じて、RECON データ・セットの DBRC セキュリティーを使用不可にすることもできます。

RACF による DBRC コマンドと API 要求の許可

RACF による DBRC コマンドおよび API 要求許可サポートをセットアップするには、すべての DBRC コマンドと API 要求を対象とするリソース・プロファイル (明示的または総称) を定義して、それらのプロファイルに対する適切なユーザー・アクセスを許可します。

リソース名は、コマンドに直接に対応し、高位修飾子 (HLQ) と DBRC コマンドの最高 3 つのエレメントで構成されます。リソース名のモデルは次のような形式をしています。

- 高位修飾子 (例えば、SAFHLQ)
- コマンド verb (例えば、NOTIFY)
- コマンド修飾子 (例えば、PRILOG または IC)
- コマンド修飾子 (例えば、dbname または OLDS)

DBRC は、コマンド許可サポート用のリソース名のリストを使用します。686 ページの『コマンド許可のためのリソース名』を参照してください。

使用する RECON データ・セットのセットに基づいて、異なるプロファイルを使用することができますが、RECON データ・セットごとに 1 つの HLQ 名しか使用できません。例えば、実動 RECON データ・セットに対する HLQ 名を PRODRECN とし、テスト RECON データ・セットに対する HLQ 名を TESTRECN とします。

定義についてのその他の考慮事項は次のとおりです。

- DBRC コマンドを保護するリソース・プロファイルは、FACILITY リソース・クラスで定義する必要があります。
- アクセス機構環境エレメント (ACEE) の ACEEUSRI フィールドにあるユーザー ID は、すべての許可検査に使用され、ユーザーは少なくともリソース・プロファイルへの読み取りアクセスを持ちます。
- DBRC コマンド許可を使用可能または使用不可にするには、CMDAUTH キーワードが指定された CHANGE.RECON および INIT.RECON コマンドを使用します。コマンド許可を使用可能にするときは、HLQ 名が必要です。

DBRC コマンドおよび API 要求のセキュリティー・プロファイルの許可

特定の DBRC コマンドと要求を保護するためにシステムにセキュリティー・リソース・プロファイルを定義することができ、各コマンドまたは要求に対する適切なユーザー・アクセスを許可することができます。

セキュリティー・リソース名は、DBRC コマンドまたは要求に直接対応し、高位修飾子 (HLQ) と、そのコマンドまたは要求の 3 つまでのエレメントから構成されます。例えば、データ・セキュリティーに RACF を使用しており、IMS システムを識別するために IMSA を HLQ として使用するとします。

DELETE.SUBSYS コマンドへのアクセスを制限するには、次のようにします。

1. すべての DBRC コマンドへのアクセスを許可するには、次の RACF コマンドを発行します。

```
RDEFINE FACILITY IMSA.** OWNER(...) UACC(READ)
```

2. DELETE.SUBSYS コマンドを ID パラメーターでリストしたユーザー ID だけが使用できるように制限するには、次の 2 つの RACF コマンドを発行します。

```
RDEFINE FACILITY IMSA.DELETE.SUBSYS.* OWNER(...) UACC(NONE)
```

```
PERMIT IMSA.DELETE.SUBSYS.* CLASS(FACILITY) ID(...) ACC(READ)
```

これで、コマンドの保護が有効になります。

3. キャッシュを消去し、RACF をリフレッシュしてデータベースから新しい値を読み取るには、次のコマンドを発行します。

```
SETROPTS REFRESH GENERIC(FACILITY) RACLIST(FACILITY)
```

4. RECON データ・セットへのアクセスを指定するには、次の DBRC コマンドを発行します。例えば、セキュリティ検査に接頭部 IMSA を使用するには、次のようにします。

```
CHANGE.RECON CMDAUTH(SAF,IMSA)
```

DSPDCAX0 出口ルーチンによる DBRC コマンドおよび API 要求の許可

DBRC コマンド許可出口ルーチン (DSPDCAX0) を使用すると、ユーザーが特定のコマンドまたは API 要求を実行することを許可されているかをチェックすることができます。

この出口を使用する場合は、INIT.RECON または CHANGE.RECON コマンドの使用時に、該当する選択を行います。詳しくは、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

DBRC コマンドを許可するために DSPDCAX0 出口ルーチンが指定されている場合 (つまり、サブパラメーター EXIT または BOTH が INIT.RECON または CHANGE.RECON コマンドの CMDAUTH キーワードに指定されている場合) には、DSPDCAX0 が必須です。DSPDCAX0 出口ルーチンは許可ライブラリーか、または LINKLST に指定されていなければなりません。DSPDCAX0 出口ルーチンが、連結された STEPLIB または JOBLIB にある場合は、DSPDCAX0 出口ルーチンを含むデータ・セットだけが許可されなければなりません。DSPDCAX0 出口ルーチンが LINKLST にある場合は、許可検査は行われません。

関連資料: DBRC コマンド許可出口 (DSPDCAX0) の詳細については、「IMS V15 出口ルーチン」を参照してください。

DSPDCAX0 出口ルーチンと RACF の両方を使用した DBRC コマンドおよび API 要求の許可

コマンド許可を与えるために、DBRC コマンド許可出口ルーチン (DSPDCAX0) を RACF と一緒に使用することができます。

その場合は、セキュリティ・プロダクトを先に呼び出します。戻りコードと理由コードは出口ルーチンに渡されます。出口ルーチンが出す戻りコードにより、コマ

ンド許可が成功したか失敗したかが最終的に判別されます。つまり、出口ルーチンはセキュリティー・プロダクトの結果をオーバーライドします。そのため、セキュリティー・プロダクトの呼び出しが失敗したときに DBRC メッセージは塗り消されます。

繰り返しますが、INIT.RECON または CHANGE.RECON コマンドの CMDAUTH キーワードに EXIT または BOTH が指定されている場合は、DSPDCAX0 が必須の出口ルーチンです。また、許可ライブラリーまたは LINKLST に DSPDCAX0 が含まれていなければなりません。DSPDCAX0 が、連結された STEPLIB または JOBLIB に含まれている場合は、DSPDCAX0 を含むデータ・セットだけが許可されなければなりません。DSPDCAX0 が LINKLST に含まれている場合は、許可検査は行われません。

関連資料: DBRC コマンド許可出口 (DSPDCAX0) ルーチンの詳細については、「IMS V15 出口ルーチン」を参照してください。

コマンド許可のためのリソース名

DBRC は、コマンド許可サポート用のリソース名のリストを使用します。

以下の表は、保護できるリソースのリストです。

表 50. コマンド許可のためのリソース名

verb	修飾子	クォリファイヤー
AUTH	適用外	.dbname
BACKUP	.RECON	
CHANGE	.ADS	.dbname
CHANGE	.BKOUT	.ssid
CHANGE	.CA	.grpname
CHANGE	.CAGRP	.grpname
CHANGE	.DB	.dbname
CHANGE	.DB	.ALL
CHANGE	.DBDS	.dbname
CHANGE	.DBDSGRP	.grpname
CHANGE	.IC	.dbname
CHANGE	.PART	.dbname
CHANGE	.PRILOG	.OLDS
CHANGE	.PRILOG	.RLDS
CHANGE	.PRILOG	.SLDS
CHANGE	.RECON	
CHANGE	.RECON	.CATDS
CHANGE	.RECON	.NOCATDS
CHANGE	.RECON	.CHECK17
CHANGE	.RECON	.CHECK44
CHANGE	.RECON	.NOCHECK
CHANGE	.RECON	.CMDAUTH

表 50. コマンド許可のためのリソース名 (続き)

verb	修飾子	クォリファイアー
CHANGE	.RECON	.DASDUNIT
CHANGE	.RECON	.DUAL
CHANGE	.RECON	.REPLACE
CHANGE	.RECON	.FORCER
CHANGE	.RECON	.NOFORCER
CHANGE	.RECON	.IMSPLEX
CHANGE	.RECON	.NOPLEX
CHANGE	.RECON	.LISTDL
CHANGE	.RECON	.NOLISTDL
CHANGE	.RECON	.LOGALERT
CHANGE	.RECON	.LOGRET
CHANGE	.RECON	.MINVERS
CHANGE	.RECON	.SIZALERT
CHANGE	.RECON	.SSID
CHANGE	.RECON	.STARTNEW
CHANGE	.RECON	.NONEW
CHANGE	.RECON	.TAPEUNIT
CHANGE	.RECON	.TIMEZONE
CHANGE	.RECON	.TIMEZIN
CHANGE	.RECON	.TIMEFMT
CHANGE	.RECON	.TRACEON
CHANGE	.RECON	.TRACEOFF
CHANGE	.RECON	.UPGRADE
CHANGE	.SUBSYS	.ssid
CHANGE	.SUBSYS	.ALL
CHANGE	.UIC	.dbname
CLEANUP	.RECON	
CLEANUP	.RECON	CAGRANGE
CLEANUP	.RECON	CAONLY
CLEANUP	.RECON	LASTCA
CLEANUP	.RECON	.RETPRD
CLEANUP	.RECON	.TIME
CLEANUP	.RECON	.DBRANGE
CLEANUP	.RECON	.DBONLY
CLEANUP	.RECON	.LASTIC
CLEANUP	.RECON	.LISTDL
CLEANUP	.RECON	.NOLISTDL
DELETE	.ADS	.dbname
DELETE	.ALLOC	.dbname
DELETE	.BKOUT	.ssid

表 50. コマンド許可のためのリソース名 (続き)

verb	修飾子	クォリファイア
DELETE	.CA	.grpname
DELETE	.CAGRP	.grpname
DELETE	.DB	.dbname
DELETE	.DBDS	.dbname
DELETE	.DBDSGRP	.grpname
DELETE	.IC	.dbname
DELETE	.LOG	.INACTIVE
DELETE	.LOG	.OLDS
DELETE	.LOG	.STARTIME
DELETE	.LOG	.TOTIME
DELETE	.PART	.dbname
DELETE	.RECOV	.dbname
DELETE	.REORG	.dbname
DELETE	.SUBSYS	.ssid
DELETE	.UIC	.dbname
GENJCL	.ARCHIVE	.ssid
GENJCL	.CA	.grpname
GENJCL	.CLOSE	.ssid
GENJCL	.IC	.dbname
GENJCL	.IC	.grpname
GENJCL	.OIC	.dbname
GENJCL	.OIC	.grpname
GENJCL	.RECEIVE	.dbname
GENJCL	.RECEIVE	.grpname
GENJCL	.RECOV	.dbname
GENJCL	.RECOV	.grpname
GENJCL	.USER	.mbrname
GENJCL	.USER	.dbname
GENJCL	.USER	.grpname
INIT	.ADS	.dbname
INIT	.CA	.grpname
INIT	.CAGRP	.grpname
INIT	.DB	.dbname
INIT	.DBDS	.dbname
INIT	.DBDSGRP	.grpname
INIT	.IC	.dbname
INIT	.PART	.dbname
INIT	.RECON	
LIST	.BKOUT	.ssid
LIST	.BKOUT	.ALL

表 50. コマンド許可のためのリソース名 (続き)

verb	修飾子	クォリファイア
LIST	.CAGRP	.grpname
LIST	.CAGRP	.ALL
LIST	.DB	.dbname
LIST	.DB	.ALL
LIST	.DB	.TYPEIMS
LIST	.DB	.TYPEFP
LIST	.DB	.TYPHALDB
LIST	.DBDS	.dbname
LIST	.DBDS	.grpname
LIST	.DBDSGRP	.grpname
LIST	.DBDSGRP	.ALL
LIST	.HISTORY	.dbname
LIST	.HISTORY	.grpname
LIST	.LOG	.ALL
LIST	.LOG	.ALLOLDS
LIST	.LOG	.OLDS
LIST	.LOG	.STARTIME
LIST	.RECON	
LIST	.RECON	.STATUS
LIST	.SUBSYS	.ONLINE
LIST	.SUBSYS	.ssid
LIST	.SUBSYS	.ALL
LIST	.SUBSYS	.BATCH
NOTIFY	.ALLOC	.dbname
NOTIFY	.BKOUT	.ssid
NOTIFY	.CA	.grpname
NOTIFY	.IC	.dbname
NOTIFY	.PRILOG	.OLDS
NOTIFY	.PRILOG	.RLDS
NOTIFY	.PRILOG	.SLDS
NOTIFY	.PRILOG	.DPROP
NOTIFY	.RECOV	.dbname
NOTIFY	.REORG	.dbname
NOTIFY	.SECLOG	.OLDS
NOTIFY	.SECLOG	.RLDS
NOTIFY	.SECLOG	.SLDS
NOTIFY	.SECLOG	.DPROP
NOTIFY	.SUBSYS	.ssid
NOTIFY	.UIC	.dbname
REPAIR	.RECON	

表 50. コマンド許可のためのリソース名 (続き)

verb	修飾子	クォリファイア
STDBRC	適用外	.ssid

コマンド許可の詳細については、683 ページの『第 41 章 DBRC セキュリティー』を参照してください。

RECON データ・セットの DBRC セキュリティーのオーバーライド

テストの目的で、または IBM ソフトウェア・サポートに送付し、元の RECON データ・セットの問題の再現に役立てるために、実動 RECON データ・セットのコピーを作成できます。

RECON データ・セットのコピーは、元の RECON データ・セットのセキュリティ設定を継承します。現行のセキュリティ設定で必要な許可のレベルを持たない誰かが、RECON データ・セットのコピーにアクセスする必要がある場合は、そのコピーに対する DBRC セキュリティーを使用不可にしてください。

元の RECON データ・セットのセキュリティ設定を継承しない RECON データ・セットのコピーを作成するには、以下のようにします。

1. *rcnqual* オプションを指定した CHANGE.RECON コマンドを発行します。
rcnqual 値が、RECON COPY1 データ・セット名のサブストリングでない場合、コマンドはセキュリティをバイパスします。ただし、*rcnqual* 値が RECON COPY データ・セット名のサブストリングの場合は、*rcnqual* の値の後にアスタリスク (*) が続いており、この値がデータ・セット名の先頭でない場合にのみ、コマンドはセキュリティをバイパスします。例えば、*rcnqual*=IMSTEST* で、IMSTESTL.RECON1 が RCNCOPY.IMSTESTL.RECON1 にコピーされた場合、コピーされたデータ・セット名には *rcnqual* の値が含まれていますが、その値で始まっていないため、セキュリティはバイパスされます。*rcnqual*=IMSTEST の場合は、コピーされたデータ・セットに対するセキュリティはバイパスされません。
2. *rcnqual* オプションで指定したストリングを含んでいない名前で、実動 RECON データ・セットのコピーを作成します。これらの RECON データ・セットのコピーには、コマンド許可のないユーザーでもアクセスできます。

コピーを使用しようとした場合、システムは RECON データ・セット内に保管されているストリングを RECON COPY1 データ・セット名と比較します。それらのストリングが一致せず、コピー内のストリングが RECON データ・セットのサブストリングでないことが確認されると、メッセージ DSP1211W が発行されます。このメッセージは、現行のコマンド許可レベルが実施されないことをユーザーに通知します。

メッセージ DSP1211W が発行された後、コマンド許可のないデータ・セットにより、RECON データ・セットのコピーと対照してテストまたはデバッグを完了することができます。メッセージ DSP1211W は、後続のすべてのコマンドの後にも、警告メッセージとして発行されます。

第 42 章 RECON データ・セットの初期化および保守

DBRC は、リカバリー関連情報を、REcovery CONtrol (RECON) データ・セットと呼ばれる 1 対のキー順データ・セット (KSDS) に記録します。DBRC は、2 つの RECON データ・セットを使用して、可用性とリカバリー可能性を向上します。これらのデータ・セットには同一の情報が含まれます。これらのデータ・セットは RECON1 および RECON2 という DD 名で識別されます。

RECON データ・セットを 2 つだけ定義し、操作中にその 1 つにエラーが起こった場合、現行ジョブは残りの 1 つを使用して続行します。新規ジョブは、アクティブ RECON データ・セットが 1 つだけでも開始できるように RECON が設定されていない限り、開始できません。二重モードで操作を続行する場合は、3 番目の RECON データ・セット (RECON3) を定義することができます。DBRC は、2 つのアクティブ RECON データ・セットのいずれかにエラーが発生しない限り、このスペア・データ・セットは使用しません。これにより、DBRC は正常な RECON をスペア・データ・セット (RECON3) にコピーし、次にこれがアクティブになります (こうして RECON 二重モード操作が維持されます)。

以下の図では、推奨される 3 つの RECON データ・セット操作構成を示します。

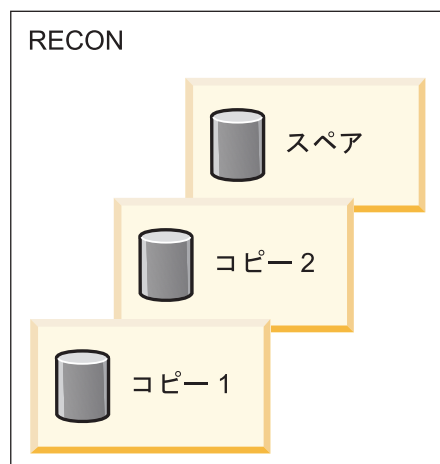


図 61. スペアを使用する DBRC 二重 RECON セットアップ

RECON データ・セットは、DBRC と IMS の両方にとってクリティカル・リソースです。RECON データ・セットが両方とも破損すると、DBRC は、データベースの保全性を弱めるというよりも、異常終了してしまいます。IMS は、実行可能な RECON データ・セットなしではトランザクション処理を継続できないため、IMS もやはり異常終了します。

RECON データ・セットの計画の考慮事項

VSAM SHAREOPTION(3,3) を指定して RECON データ・セットを定義します。理想的には、それぞれのデータ・セットは別々の装置、別々のチャネル、別々の制御装置、および別々のカタログ上にあるべきです。また、これらの RECON データ・セットは異なるサイズにしてください。

各 RECON データ・セットは、32 バイトのキーを持つ VSAM KSDS です。

推奨事項:

- アクセス方式サービス・プログラム (AMS) を使用して RECON データ・セットを定義する場合は、次のようにします。
 - すべての RECON データ・セットで同一サイズの索引制御インターバル (CI) と、同一サイズのデータ CI を使用する。
- 指定されたデータ CI サイズが指定された索引 CI サイズを少なくとも 2048 バイトは超えていることを確認してください。超えていない場合は、DBRC のパフォーマンスは著しく低下します。
- RECON データ・セットがいっぱいになるのを防ぐ安全機能として、2 次割り振りを使用してください。

関連概念:

699 ページの『RECON データ・セットのスペース問題の回避』

RECON データ・セットの最初の初期化

RECON データ・セットを割り振ってから、INIT.RECON コマンドを使用して RECON データ・セットを初期化してください。

このコマンドは、空の、初期化されていない RECON データ・セットに対してのみ有効です。初期設定ジョブが失敗した場合は、データ・セットを削除し、再定義してから、そのジョブを再実行します。

RECON データ・セットの競合問題の回避

RECON データ・セットの最大の可用性を得るには、デッドロック状態をなくし、複数の RECON データ・セットが同時に利用不能になる状態を最小化する必要があります。

可用性を最大にするには、それぞれの RECON データ・セットが次のようになっている必要があります。

- スペース割り振りが異なる。スベア・データ・セットは、少なくとも最大の RECON データ・セットと同じ大きさでなければなりません。
- 別々の装置上にある。
- 別々のチャネル上にある。
- 別々のユーザー・カタログにある。

並列モードでの RECON データ・セットへのアクセス

データ・セットのデッドロック状態を回避するために、RECON データ・セットに並列モードでアクセスできます。

並列モードに関連したトピックでは、以下の用語が使用されます。

DBRC グループ (DBRC group)

IMSplex 環境で単一の RECON データ・セットを共有する 1 つ以上の DBRC インスタンス。

DBRC グループ ID

DBRC グループの ID

DBRC 要求 (DBRC request)

IMS、DBRC アプリケーション・プログラム、およびジョブによって DBRC に要求される個々のアクション (コマンド・プロセス、照会、および更新)。

DBRC は、DFSMS のトランザクション VSAM 機能を使用して、RECON データ・セットへの並列アクセスを提供します。トランザクション VSAM は以下のものを使用します。

- z/OS リソース・リカバリー・サービスによって提供されている 2 フェーズ・コミットおよびバックアウトのサービス。これにより、トランザクションによって行われたすべての変更の確定、またはトランザクションによって行われたすべての変更のバックアウトが確実に実行されます。
- バッファのキャッシング、ロッキング、およびロギング・サービスのためのカップリング・ファシリティ (CF)。

以下のコマンドを使用して、RECON データ・セットのアクセス・モードを順次または並列に設定します。

- 新規 RECON データ・セットの場合は、INIT.RECON ACCESS(SERIAL または PARALLEL) コマンドを使用します。
- 既存の RECON データ・セットの場合は、CHANGE.RECON ACCESS(SERIAL または PARALLEL) コマンドを使用します。

並列 RECON データ・セット・アクセスを使用可能にするには、以下の両方の条件が真であることが必要です。

- RECON データ・セットにアクセスするすべての DBRC インスタンスが、固有の DBRC グループ ID を持つ DBRC グループに属している。
- RECON データ・セットにアクセスするすべての DBRC インスタンスが、Structured Call Interface (SCI) に登録されている。これは、RECON 損失通知が関与する状況によく似ています。DBRC はグループ ID をメンバー・サブタイプとして使用して、SCI に登録します。

並列 RECON データ・セット・アクセスの計画:

並列 RECON データ・セット・アクセスの準備のために、Structured Call Interface およびトランザクション VSAM をセットアップします。

並列 RECON アクセスを準備するには、以下の作業を行う必要があります。

- Structured Call Interface (SCI) をセットアップします。
 - 共通サービス層 (CSL) を定義し、IMS システムおよびバッチ・ジョブが RECON データ・セットにアクセスする元になる各 z/OS イメージ上に、Structured Call Interface が存在するように IMSplex 環境を構成します。

- RECON データ・セットの正しい IMSplex 名を指定するために、DBRC SCI 登録出口ルーチン DSPSCIX0 を必要に応じて調整します。出口ルーチンが IMS DSPSCIX0 拡張インターフェースで機能するようにコーディングされていることを確認します。
- トランザクション VSAM を以下のようにしてセットアップします。
 - トランザクション VSAM をサポートするように、SYS1.PARMLIB 内の IGDSMSxx メンバーを更新します。

推奨事項: ローカルとグローバルの両方の検出について検出時間が 2 秒未満になるように、IGDSMSxx で DEADLOCK_DETECTION を指定してください。

- トランザクション VSAM 処理を使用可能にするように、SMS 構成をセットアップします。
- RECON データ・セットが SMS で管理されていない場合は、SMS で管理されるように変換します。
- 各 RECON データ・セットのストレージ・クラスを指定して、そのデータ・セットをカップリング・ファシリティ (CF) キャッシュ構造に割り当てます。
- SMS 構成用に少なくとも 1 つの CF キャッシュ構造を定義します。
- z/OS に対して、CF ロック構造 IGWLOCK00 を定義します。
- トランザクション VSAM のインスタンスごとに、1 次システム・ログ・ストリームを定義します。
- トランザクション VSAM のインスタンスごとに、2 次システム・ログ・ストリームを定義します。

関連資料: SMS の構成とトランザクション VSAM 用の CF 構造のセットアップについては、「z/OS DFSMS Storage Administration Reference (for DFSMSdftp, DFSMSdss, and DFSMSHsm)」を参照してください。トランザクション VSAM 用の SYS1.PARMLIB の変更およびログ・ストリーム定義の詳細については、「z/OS TSO/E カスタマイズ」を参照してください。

- RECON 入出力出口ルーチン (DSPCEXT0) を使用する場合は、この出口ルーチンが IMS 拡張 RECON 入出力出口インターフェースで機能することを確認してください。
- バッチ・コマンドか DBRC API を使用して CHANGE.RECON.ACCESS コマンドを発行するユーザーが、以下の権限を持っていることを確認してください。
 - DEFINE CLUSTER コマンドの LOG パラメーターを保守するためのデータ・セット ALTER 権限
 - 指定したデータ・セットと RACF クラス・プロファイル STGADMIN.IGWSHCDS.REPAIR に対する UPDATE 権限
- 読み取り専用ジョブに RACF クラス・プロファイル STGADMIN.IGWSHCDS.REPAIR に対する READ 権限があることを確認します。
- オンライン DBRC 領域で実行されるジョブを除いて、読み取り専用でないジョブはすべて、クラス・プロファイル STGADMIN.IGWSHCDS.REPAIR に対する

UPDATE 権限を備えている必要があります。UPDATE 権限が許可されていないと、これらの領域は、シャント済み入出力が関係するエラーからリカバリーできない場合があります。

- RECON データ・セットを共用するすべての DBRC インスタンスが並列 RECON アクセス機能および自動 RECON 損失通知機能に同じ IMSplex 名を使用するようにします。

同じ IMSplex 内で自動 RECON 損失通知機能と並列 RECON アクセス機能をアクティブにする場合は、CHANGE.RECON IMSPLEX() コマンドを使用して、IMSplex 内のすべての DBRC が、RECON データ・セット内で指定されているのと同じ IMSplex 名を使用するようにしなければなりません。

CHANGE.RECON IMSPLEX() コマンドを発行する前に DBRC SCI 登録出口ルーチン (DSPSCIX0) を使用するか、IMSplex EXEC パラメーターを使用した場合は、メッセージ DSP1136A が発行され、後続のジョブは RECON データ・セットを使用できないために失敗します。

順次アクセス・モードでの RECON データ・セットのデッドロック状態の回避

デッドロックをなくすためには (RECON データ・セット用に予約された実装ハードウェアを使用する場合)、RECON データ・セットをそれぞれのカタログ内にカタログされた唯一のオブジェクトとし、そのカタログと同じ装置上に置く必要があります。RECON データ・セットがアクセスされると、結果としてその RECON データ・セット上にエンキューが生じ、続いてカタログ上にエンキューが生じます。RECON データ・セットとカタログが同一のデバイス上にある場合は、そのデバイスを逆順でエンキューしている別のジョブとの競合の可能性を取り除くことができます。

複数のプロセッサ間で共用される RECON データ・セットの配置には、特別な配慮をしてください。物理的にオープンしている間に、DBRC は RECON1、RECON2、および RECON3 を予約します。DBRC はどれが使用可能か、ならびにどれがコピー 1、コピー 2、およびスペアかを判別します。その後 DBRC は、スペア (もし存在すれば) および使用不可の RECON データ・セットをすべてクローズし、デキューします。そこで、DBRC の使用中はほとんどの時間、2 つの RECON データ・セットが予約されています。DBRC は常に、この 2 つの RECON データ・セットを RECON1、RECON2 の順序で予約します。RECON1 と RECON2 がジョブの間中一貫して指定されていれば、DBRC がデッドロックを検出することはありません。

ただし、マルチボリュームを予約している他のジョブでは、そのいずれかのボリュームにも RECON データ・セットが含まれている場合、デッドロックを引き起こすことがあります。

推奨事項: 競合を除去し、リカバリー可能性を促進するために、RECON データ・セット用に実装ハードウェアの予約を使用している場合、RECON データ・セットをそれぞれの装置で唯一のタイプのデータ・セットとすることをお勧めします。

デッドロックを回避するためには、RECON データ・セットへのアクセスを制御する必要があります。RECON データ・セットへのアクセスを制御するには、次に説明する RESERVE、GRS、OBTAIN、および DEQ の各 z/OS マクロを使用しま

す。z/OS マクロ、DFP レコード管理サービス、および RECON データ・セットのシリアライゼーション・ストラテジーについては、以下のリストで説明します。

- RESERVE

DBRC は z/OS RESERVE マクロを出して、各 RECON データ・セットへのアクセスをシリアライズします。DBRC はその処理が完了するまで RECON データ・セットを予約したままにします。DBRC は、調べたり変更したりする必要がある RECON レコードが多ければ多いほど、より長い間 RECON データ・セットを保持します。RESERVE マクロは、リソース (共用 DASD ボリューム上のデータ・セット) があるボリュームの制御を取得して、他システムのジョブがそのボリューム全体のいずれのデータ・セットも使用できないようにして、そのリソースへのアクセスをシリアライズします。この予約は、大分類名 DSPURI01 の下で行われ、有効範囲は SYSTEMS です。

バッチ・ジョブは、まず別のリソース名でシリアライズしてから、RECON データ・セットのための RESERVE を発行します。バッチのリソース名は DSPURI02 で、その有効範囲は SYSTEM です。各ジョブは、タスクの要求の位置と、その要求が排他制御または共用制御のいずれであったかを基にして、リソースの制御を取得します。キューの配列は、タスクの優先順位によるものではありません。バッチ・ジョブのこのシリアライゼーションの効果は、IMS オンライン領域が複数のバッチ・ジョブの完了を待たずに RECON データ・セットにアクセスできる点に現れています。

- グローバル・リソース・シリアライゼーション (GRS)

GRS マクロは、RESERVE 要求を ENQ 要求に変換する方式を提供します。ENQ、DEQ、および RESERVE マクロは、リソースをそのシンボル名で識別します。シンボル名は、以下の 3 つの部分から構成されます。

- 大分類名 (qname)
- 小分類名 (rname)
- 有効範囲 (STEP、SYSTEM、または SYSTEMS のいずれか)

例えば、ENQ または DEQ マクロでは、リソースのシンボル名は APPL01,MASTER,SYSTEM のようになります。大分類名 (qname) が APPL01、小分類名 (rname) が MASTER、そして有効範囲が SYSTEM です。

アプリケーションが ENQ、DEQ、および RESERVE マクロを使用してリソースをシリアライズする場合、グローバル・リソース・シリアライゼーションは、リソース名リスト (RNL) と ENQ、DEQ、または RESERVE マクロ上の有効範囲を使用して、リソースがローカル・リソースまたはグローバル・リソースのいずれであるかを判別します。グローバル・リソース・シリアライゼーションは、各リソースをそのシンボル名全体で識別します。例えば、A,B,SYSTEMS と指定されているリソースは、A,B,SYSTEM または A,B,STEP とは異なるリソースと見なされます。これは、それぞれのリソースの有効範囲が異なるためです。アプリケーションに変更を加えることなく、リソースがユーザーの望むとおりに扱われるようにするために、グローバル・リソース・シリアライゼーションは以下の 3 つのリソース名リスト (RNL) を備えています。

- SYSTEMS EXCLUSION RNL

SYSTEMS EXCLUSION RNL には、グローバル・リソース・シリアライゼーションにローカル・リソースとして扱わせる、有効範囲 SYSTEMS を指定して要求されるリソースのリストが含まれます。

- RESERVE CONVERSION RNL

RESERVE CONVERSION RNL には、グローバル・リソース・シリアライゼーションに RESERVE を抑制させる、RESERVE マクロで要求されるリソースのリストが含まれます。

- SYSTEM INCLUSION RNL

SYSTEM INCLUSION RNL には、グローバル・リソース・シリアライゼーションにグローバル・リソースとして扱わせる、有効範囲 SYSTEM を指定して要求されるリソースのリストが含まれます。

関連資料: GRS と、z/OS RESERVE、DEQ、および ENQ の各マクロについての詳細は、下記の資料を参照してください。

- z/OS MVS Programming: Assembler Services Reference Vol 1
- z/OS MVS 計画: グローバル・リソース逐次化

• OBTAIN

DBRC は FORMAT-4 DSCB (VTOC) に対する VSAM DADSM (直接アクセス装置ストレージ管理) の OBTAIN 要求を使用して入出力を強制し、それにより DBRC が実際に RECON データ・セットをマルチシステム環境で予約できるようにします。

OBTAIN マクロは、同期 RESERVE (SYNCHRES) オプションが実施されている場合は出されません。同期 RESERVE 機能は、グローバル・リソース・シリアライゼーション ENQ を付与する前に、装置のためにハードウェア RESERVE を取得するかどうかをインストールの際に指定できるようにします。

SYNCHRES オプションは、parmlib メンバー GRSCNFxx かまたは SETGRS オペレーター・コマンドのどちらかを通じて活動化することができます。GRSCNFxx の GRSDEF ステートメントには、SYNCHRES (YES | NO) パラメーターが含まれます。SYNCHRES のデフォルト値は、YES です。通常のシステム操作のなかで、オペレーターは SETGRS コマンドを出して SYNCHRES の設定値を変更することができます。SYNCHRES を活動化するには、SETGRS SYNCHRES=YES を出し、非活動化するには SETGRS SYNCHRES=NO を出します。

• DEQ

DBRC は z/OS DEQ マクロを使用して RECON データ・セットを解放します。

• DFP レコード管理サービス

DBRC は、VSAM サービスを使用して RECON データ・セット・レコードの検索、操作、および保管を行います。これらのレコードは、32 バイトのレコード・キーを持っています。

• RECON データ・セットのシリアライゼーション・ストラテジー

推奨事項:

- GRS スター型構成では、RECON データ・セットにアクセスするすべてのシステムがシスプレックス (または GRSPlex) の中にある場合は、RESERVE CONVERSION RNL を DSPURI01 に対して実装する必要があります。

注: RECON データ・セットがシスプレックス外部のシステムによってアクセスされる場合は、その予約を変換しないでください。その代わりに、必ず SYSTEMS EXCLUSION RNL をインプリメントしてください。

RECON データ・セットの QNAME の DSPURI01 を変換リストに追加することにより、GRS RNL CONVERSION を実装すると、ハードウェアの予約は取り消され、GRS エンキューで置き換えられて、それが他のすべての共用 z/OS システムに通信されます。

RECON データ・セットが予約済みの間も、同じ DASD ボリューム上の他のデータ・セットは使用することができます。これが RNL 変換を実行することの利点です。GRS RNL 変換は CPU およびストレージを使用するため、システム・パフォーマンスに良い影響を及ぼします。パフォーマンス (使用される CPU 時間が最小、使用ストレージが最小で、経過時間が最小という観点で) は、GRS STAR 構成でこのオプションを使用する場合が最高です。

この方式を実装するには、次のステップに従ってください。

1. RECON データ・セット QNAME を RESERVE CONVERSION RNL に追加する。例えば、次のように指定します。
RNLDEF RNL(CON) TYPE(GENERIC) QNAME(DSPURI01)
2. VSAM QNAME の SYSZVVDS および SYSIGGV2 の配置を慎重に考慮する。

関連資料: GRS RNL CONVERSION の実装の詳細については、「z/OS MVS 計画: グローバル・リソース逐次化」を参照してください。

- GRS リング構成では、RECON がハードウェアの予約によってシリアライズされるように、SYSTEMS EXCLUSION RNL を DSPURI01 に対してインプリメントする必要があります。

GRS SYSTEMS EXCLUSION RNL を実装した場合、GRS はグローバル・シリアライゼーションを行わず、RESERVE マクロが実行されます。これは、RECON データ・セットが、他の z/OS システムで必要とされる他のデータ・セットを含んでいない DASD ボリュームに配置されている場合には正しく機能します。

この方式を実装するには、次のステップに従ってください。

1. RECON データ・セット QNAME を SYSTEMS EXCLUSION RNL に追加する。例えば、次のように指定します。
RNLDEF RNL(EXCL) TYPE(GENERIC) QNAME(DSPURI01)
2. VSAM QNAME の SYSZVVDS および SYSIGGV2 の配置を慎重に考慮する。

関連資料: 「z/OS MVS 計画: グローバル・リソース逐次化」に、VSAM QNAME に関する詳しい情報が記載されています。

RECON データ・セットの割り振り

オンラインおよびバッチの DBRC ジョブの両方の場合とも、RECON1、RECON2、および RECON3 データ・セットに JCL を割り当てるか、あるいは DBRC にデータ・セットを動的に割り振りさせることができます。

RECON データ・セットを動的に割り振る場合は、RECON1、RECON2、および RECON3 の DD ステートメントは省略します。

IMS DFSMDA 動的割り振りマクロの TYPE=RECON マクロ・ステートメントを使用することによって、IMS.SDFSRESL 内の 3 つの動的割り振りパラメーター・リストを設定します。複数のプロセッサが同じ RECON データ・セットにアクセスする場合は、動的割り振りパラメーターに関する IMS.SDFSRESL 情報を、すべてのプロセッサで同期させます。

関連資料: DFSMDA マクロについては、「IMS V15 システム定義」を参照してください。

DBRC は、常に DISP=SHR で RECON データ・セットを割り振ります。

JCL 割り振りと動的割り振りはいずれも RECON データ・セットを割り振る有効な方式ですが、JCL 割り振りは制御されたテストでのみ使用してください。

推奨事項: 実動システムおよび他のすべてのテスト環境または開発環境で、動的割り振りを使用してください。

RECON データ・セットの動的割り振りの主な利点は、次のとおりです。

- すべての DBRC ジョブは、自動的に正しく、かつ現行の RECON データ・セットを使用しており、JCL ステートメントが古いままになることはありません。
- エラーが起こった場合、オンライン IMS システムをシャットダウンしなくても、RECON データ・セットを再編成し、復元することができます。

要件: RECON データ・セットに並列モードでアクセスする場合は、それらのデータ・セットが DFSMS によって管理されている必要があります。

RECON データ・セットのスペース問題の回避

それぞれの RECON データ・セットに異なる量のスペースを割り振り、一方がいっぱいになっても、スペアの置き換えを用意する間に、システムがもう一方の RECON データ・セットを使用して続行できるようにします。RECON データ・セットにスペースを定義するときに、そのデータ・セットに 2 次エクステントを割り振ります。

オンライン操作中に一方の RECON データ・セットがいっぱいになると、IMS はその RECON の割り振りを解除します。DBRC は、スペアの RECON データ・セットがあれば、それに有効な RECON データ・セットをコピーし、再編成することによって対応します。スペアの RECON データ・セットがない場合は、システムは単一 RECON モードで実行します。

すべてのアクティブ・サブシステムが障害を起こした RECON データ・セットの割り振りを解除した場合は、オフラインで AMS を使用してそれを削除し、再定義することができます。廃棄された RECON データ・セットの置き換えについては、723 ページの

723 ページの『廃棄された RECON データ・セットの置き換え』を参照してください。単一モードの場合で、スペアの RECON データ・セットが使用可能であれば、次回に DBRC が RECON データ・セットにアクセスする際、DBRC は自動的に二重 RECON モードに入ります。CHANGE.RECON コマンドで、DUAL または REPLACE オプションを指定して入力する必要はありません。

RECON データ・セットの作成

RECON データ・セットは DEFINE CLUSTER コマンドを使用して作成します。

RECON VSAM KSDS を定義する際にお勧めするキーワードを、次のリストで説明します。すべてのキーワードに関する情報は、「z/OS DFSMS カタログのためのアクセス方式サービス・プログラム」に記載されています。

CONTROLINTERVALSIZE

このキーワードで使用される値は、DBRC が VSAM および内部バッファー用に使用するストレージの合計量に影響します。DBRC は、VSAM のローカル共有リソース (LSR) オプションを使用して RECON データ・セットを処理します。DBRC が作成する索引およびデータ・バッファーの数をデフォルトに設定できる場合、RECON バッファーに使用されるストレージの量は、次のとおりです。

$$(60 \times \text{index_ci_size}) + (120 \times \text{data_ci_size})$$

この量のストレージが使用されるのは、索引とデータの CI サイズがすべての RECON で同じ場合です。DSPBUFFS バッファー・サイズ指定機能を使用して、DBRC がオンラインまたはバッチ環境で使用する索引またはデータ・バッファーのデフォルトの数を変更することもできます。

DBRC は RECON データ・セットごとに 1 セットずつの内部バッファーを使用して、小さいセグメントから論理レコードを作成します。これらのバッファーのサイズは、初期には、CI サイズまたは VSAM の最大レコード・サイズ (LRECL) のどちらか小さい方によって決まります。これらのバッファーの最初の割り振りのあと、論理レコード・バッファーは、RECON データ・セットの最大論理レコードが大きくなるにつれて、時間の経過とともに大きくなります。バッファーの最初の割り振りだけは VSAM データ・セット定義で決められるので、内部バッファーに割り振られるストレージが、CI サイズまたはレコード・サイズの変更によって根本的に影響されることはありません。

DBRC は所有するレコードをセグメントに分割します。各セグメントは、常に単一の制御インターバルより小さく、VSAM からは完全な物理レコードとして見られます。VSAM レコード・スパンは使用されません。セグメント化により、VSAM RECORDSIZE パラメーターに関係なく、論理 RECON レコードを 16 MB の大きさにすることができます。

推奨事項: 最初に、最小 CI サイズを最小の 8 KB に設定してください。使用できる CI サイズは、RECORDSIZE で選択した値に影響されます。また、最小のデータ CI サイズが、最大の索引 CI サイズを少なくとも 2048 バイトは超えているようにします。超えていない場合は、DBRC のパフォーマンスは著しく低下します。

関連資料: バッファー・サイズ指定機能 (DSPBUFFS) の使用の詳細については、「IMS V15 出口ルーチン」を参照してください。

CYLINDERS

クラスターに割り振るスペースの量を指定します。

FREESPACE

デフォルト値 FREESPACE(0 0) を使用してはなりません。RECON データ・セットに初期情報を入力する際に、フリー・スペースとして高いパーセンテージ (例えば、70%) の制御インターバルを指定する必要があります。後で、アクセス方式サービス・プログラム (AMS) の ALTER コマンドで、そのパーセンテージを下げるすることができます。

INDEXED

クラスターがキー・シーケンス・データ用に定義されていることを指定します。

KEYS

KEYS(32 0) は必須です。

LOG

並列 RECON アクセスを使用する場合は、AMS DEFINE CLUSTER コマンドの LOG パラメーターを DBRC に管理させます。並列 RECON アクセスが使用可能な場合、DBRC は LOG パラメーターの値を UNDO (データ・セットがリカバリー可能であることを意味します) に設定し、RECON ヘッダー内の ACCESS= フィールドを PARALLEL に設定します。並列 RECON アクセスが使用不可の場合、DBRC は LOG パラメーターの値を NONE に設定し、RECON ヘッダー内の ACCESS= フィールドを SERIAL に設定します。RECON データ・セットを削除して再定義するときは常に、LOG 値に適切な値を指定してください。例えば、並列アクセスを使用する場合は、LOG(UNDO) を指定してください。

NAME

クラスターのエントリー名を定義します (必須)。

NONSPANNED

RECORDSIZE を参照してください。

RECORDSIZE

論理 RECON レコードの長さは 16MB まで許容されますが、DBRC は常に、長さが CI サイズ未満の物理 VSAM レコードを書き込みます。

推奨事項: 最大レコード・サイズは最大で CI サイズから 7 バイトを引いた値と同等になるように設定し、NONSPANNED が指定されていることを確認してください。例えば、CI サイズが 8192 バイトとして定義されている場合は、RECORDSIZE (4086,8185) が適切です。

SHAREOPTIONS

SHAREOPTIONS(3 3) を指定する必要があります。最初の値は単一ホスト・プロセッサでは必須です。複数ホスト・プロセッサでは両方の値が必須です。

SPEED

これは、初期ロードが高速になるので、推奨されます。

NOWRITECHECK

WRITECHECK の使用は避けてください。RECON データ・セットの入出力パフォーマンスを低下させる恐れがあります。二重 RECON データ・セットを使用すれば、WRITECHECK を使用する必要はありません。

推奨事項: オペレーター・プロンプトが頻繁に生じる結果となるため、許可キーワードは使用しないでください。

RECON データ・セットのセキュリティー上の考慮事項

RECON データ・セットを更新するジョブには、RECON データ・セットに対する UPDATE 権限が必要です。

読み取り専用アクセスを必要とするジョブには、READ 権限を使用してください。並列アクセスの場合、CHANGE.RECON ACCESS コマンドを発行するジョブには、ALTER 権限が必要です。並列アクセスを使用する場合には、オンライン IMS DBRC 領域に ALTER 権限が必要です。

RECON データ・セットの保全性の保持の詳細については、683 ページの『第 41 章 DBRC セキュリティー』を参照してください。

RECON データ・セットへの初期アクセス

ジョブが RECON データ・セットを読み取る必要がある場合は、RECON データ・セットがオープンされていなければなりません。RECON データ・セットが 3 つ存在する場合、DBRC は、どの 2 つがアクティブ RECON データ・セットで、いずれがスペア・データ・セットかを判別します。

以下の表は、この判別がどのように行われるかを示しています。表の列には、次の意味があります。

- 「DD ステートメント」列の RECONA、RECONB、および RECONC は、順序不定で RECON1、RECON2、および RECON3 DD ステートメントを表します。
- 「データ・セットの状況」列は、オープン中のデータ・セットの状況を示します。
- 「データ・セットの使用」列は、DBRC がデータ・セットをどのように割り当てるかを示します。

表 51. アクセスされる RECON データ・セットの判別:

事例	DD ステートメント	データ・セットの状況	データ・セットの使用	DBRC 選択基準
1	RECONA	作成モード	コピー 1	INIT.RECON コマンドが指定されるまで選択されない
	RECONB	作成モード	コピー 2	INIT.RECON コマンドが指定されるまで選択されない
	RECONC	作成モード	スペア	
2	RECONA	RECON	コピー 1	RECON の現行コピー
	RECONB	作成モード	コピー 2	コピー 1 から作成
	RECONC	作成モード	スペア	
3	RECONA	RECON	コピー 1	RECON の現行コピー
	RECONB	RECON	コピー 2	RECON の現行コピー
	RECONC	作成モード	スペア	

表 51. アクセスされる RECON データ・セットの判別 (続き):

事例	DD ステートメント	データ・セットの状況	データ・セットの使用	DBRC 選択基準
4	RECONA	RECON	コピー 1	RECON の現行コピー
	RECONB	RECON	未使用	RECONA より古いコピー
	RECONC	作成モード	コピー 2	コピー 1 から作成
5	RECONA	RECON	コピー 1	RECON の現行コピー
	RECONB	RECON	未使用	RECONA より古いコピー
	RECONC	RECON	未使用	RECONA より古いコピー
6	RECONA	RECON	コピー 1	RECON の現行コピー
	RECONB	RECON	コピー 2	RECON の現行コピー
	RECONC	RECON	未使用	RECONA より古いコピー
7	RECONA	RECON	コピー 1	RECON の現行コピー
	RECONB	作成モード	コピー 2	コピー 1 から作成
8	RECONA	作成モード	コピー 1	INIT.RECON コマンドが指定されるまで選択されない
	RECONB	作成モード	コピー 2	INIT.RECON コマンドが指定されるまで選択されない
9	RECONA	RECON	コピー 1	RECON の現行コピー
	RECONB	RECON	コピー 2	RECON の現行コピー
10	RECONA	RECON	コピー 1	RECON の現行コピー
	RECONB	RECON	未使用	RECONA より古いコピー
11	RECONA	作成モード	ありません。	処理を中止する
12	RECONA	RECON	コピー 1	RECON の現行コピー

ケース 4 は、2 つの RECON データ・セットが使用可能の状態ですが、1 つはもう期限切れの状態です。DBRC は、期限切れの RECON データ・セットは使用しません。その代わりに、最新の RECON データ・セットをスペア・データ・セットにコピーします。

ケース 5、10、および 12 の場合、1 つ RECON データ・セットのみが使用可能です。INIT.RECON または CHANGE.RECON コマンドの STARTNEW パラメーターを指定した場合は、処理は 1 つの RECON データ・セットで続きます。それ以外の場合は、処理は終了します。別の DBRC インスタンスがアクティブで、並列 RECON アクセスが使用可能の場合、新規 DBRC は、STARTNEW パラメーターに関係なく、既存の DBRC と (名前もアクティブ・カウントも) 同じデータ・セットにアクセスできなければなりません。新規 DBRC が同じデータ・セットにアクセスしない場合、オンライン DBRC 領域については WTOR が発行され、オンラインでない DBRC 領域については処理が終了します。

RECON データ・セット内のレコード

RECON データ・セットには多くのタイプのレコードが含まれます。一部のレコード (ヘッダー・レコードなど) は、RECON データ・セットの処理を主に制御するため存在します。その他のレコードは、DBDS のリカバリーに使用される各種のデータ・セットを定義するためのものです。さらにその他のレコードは、DBDS の使用に関連するイベントを記録するためのものです。

以下の図では、RECON レコード・タイプ間の主要な関係を示します。これらの関係については、以下の図の後の各トピックで詳しく説明します。

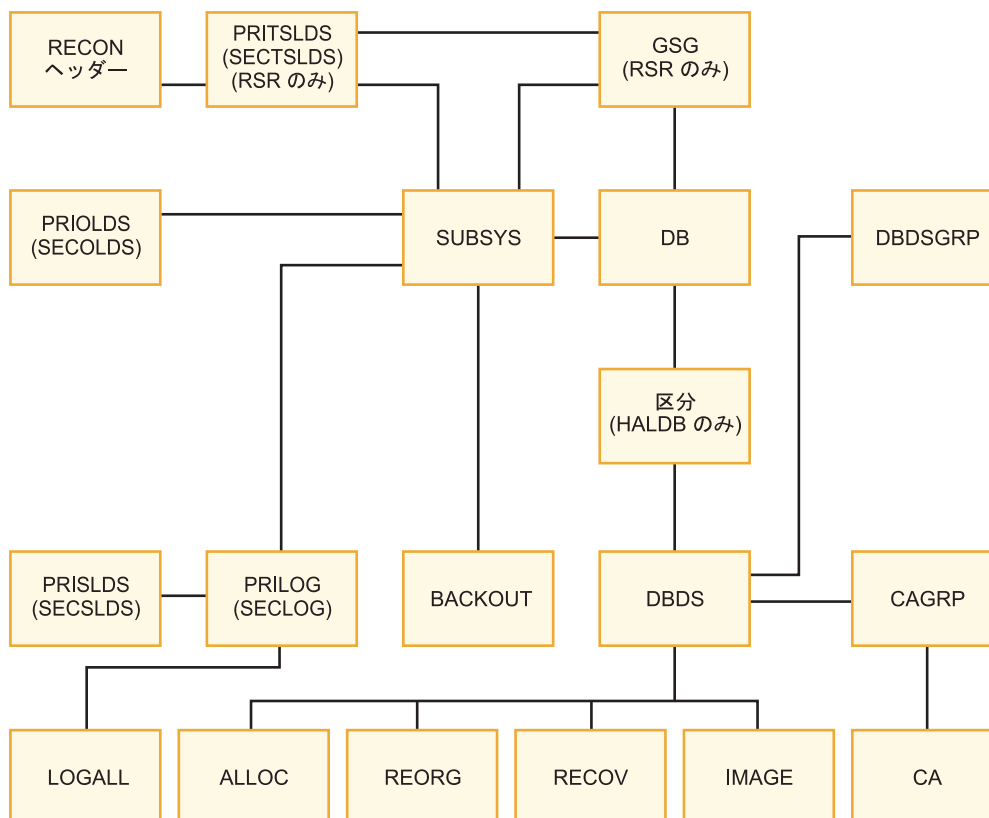


図 62. 主要な RECON レコード・タイプおよびその関係

関連資料:

- レコードの内容の詳細については、IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンドを参照してください。
- RECON キーについては、「IMS Version 15 Diagnosis」の『Database Recovery Control service aids』を参照してください。

RECON ヘッダー・レコード

RECON データ・セットには、DBRC がその処理に使用する制御情報を含む、2 つのレコード・タイプがあります。

- RECON ヘッダー

RECON ヘッダー・レコードには、DBRC が作成し、使用する内部制御情報に加えて、INIT.RECON および CHANGE.RECON コマンドのパラメーターでユーザーが提供する情報が含まれます。

- RECON ヘッダー拡張

RECON ヘッダー拡張レコードには、DBRC が RECON データ・セットを処理するために使用する RECON データ・セット構成データおよび状態データが含まれます。

ログ・データ・セット・レコード

ログ・データ・セットには 3 つのタイプがあります。それぞれのタイプには、1 次ログ・レコード、2 次ログ・レコード、中間 1 次ログ・レコード、および中間 2 次ログ・レコードがあります。

それぞれのログ・データ・セットごとのログ・データ・セット・レコードは、次のとおりです。

- リカバリー・ログ・データ・セット
 - PRILOG (1 次ログ・レコード)
 - SECLOG (2 次ログ・レコード)
 - IPRI (中間 1 次ログ・レコード)
 - ISEC (中間 2 次ログ・レコード)
- システム・ログ・データ・セット
 - PRISLD (1 次システム・ログ・データ・セット)
 - SECSLD (2 次システム・ログ・データ・セット)
 - IPRISL (中間 1 次システム・ログ・データ・セット)
 - ISECSL (中間 2 次システム・ログ・データ・セット)
- オンライン・ログ・データ・セット
 - PRIOLDS (1 次オンライン・ログ・データ・セット)
 - SECOLDS (2 次オンライン・ログ・データ・セット)
 - IPRIOL (中間 1 次オンライン・ログ・データ・セット)
 - ISECOL (中間 2 次オンライン・ログ・データ・セット)

ログ・レコードは、*PRILOG* ファミリー と呼ばれる集合に加えられます。

PRILOG ファミリーは、*PRILOG* のほか、所定の時間枠と IMS サブシステム用に、*SECLOG*、*PRISLD*、および *SECSLD* の 1 つ以上から構成されます。この集合内のレコードはすべて、開始時刻と終了時刻が同じであり、通常、マッチング用のデータ・セット・エントリーを持っています。この集合のすべてのメンバーに同じ *LOGALL* レコードが適用されます。

DBRC は、オンライン IMS が最初の *OLDS* をオープンすると必ず、*PRILOG* および *PRISLD* レコードを作成し、*OLDS* がアーカイブされるたびに、これらのレコードを更新します。二重アーカイブを使用する場合は、DBRC は、最初の *OLDS* がアーカイブされると *SECLOG* および *SECSLD* レコードを作成し、*OLDS* がアーカイブされるたびに、これらのレコードを更新します。

関連資料: DBRC およびオンライン・ログのアーカイブの詳細については 627 ページの『OLDS のアーカイブ』を参照してください。

IMS バッチ・ジョブからのログ・データ・セット出力は、たとえそれが技術的には SLDS であっても、PRILOG/SECLOG レコードに記録されます。これらのレコードは出力ログがオープンされると必ず作成され、ボリュームの切り替えが行われると更新されます。

さらに、ログ・リカバリー処理中に、DBRC は、ログ・リカバリー・ユーティリティーが実行されると、必ず、中間 1 次ログ・データ・セットごとに IPRISL または IPIROL レコードを作成し、中間 2 次ログ・データ・セットごとに中間 2 次ログ・レコードを作成します。中間ログ・レコードは、ログ・リカバリー・ユーティリティーの DUP 機能の中間処理結果を反映するために使用される内部レコードです。

関連資料: 中間 1 次および中間 2 次ログ・データ・セットの詳細については、「IMS V15 システム・ユーティリティー」の『ログ・リカバリー・ユーティリティー (DFSULTR0)』を参照してください。

通常の DBRC 操作の一部ではありませんが、次のコマンドを使用してログ・レコードを作成することができます (例えば、テスト環境のセットアップや RECON データ・セットの修復の目的で)。

- NOTIFY.PRILOG
- NOTIFY.SECLOG

データベース・リカバリー・レコード

ログ・データ・セット・レコードのほかに、RECON データ・セットには、内部にデータベース・リカバリー情報が入っているレコードが含まれています。

関連資料: RECON データ・セット内のレコードのリストと、RECON データ・セット・レコード内のフィールドを説明した表については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

バックアウト・レコード (BACKOUT)

BACKOUT レコードには、タイム・スタンプ、関連 PSB 名、リカバリー・トークン、およびデータベース名を含むリカバリー単位についての情報が含まれます。リカバリー不能データベースの名前は BACKOUT レコードには保管されません。

変更累積グループ・レコード (CAGRP)

CAGRP レコードは変更累積 (CA) グループを識別します。データベース変更累積ユーティリティーが 1 回実行される間に、DBDS の変更レコードが累積されますが、このレコードにはそのような DBDS の名前が最大 32,767 個まで入ります。(DBRC がリカバリーを制御している) それぞれの DBDS は、データベース変更累積ユーティリティーがその変更を累積できるようにするために、ただ 1 つの CA グループのメンバーになることができます。CAGRP 名の指定は、RECON データ・セットに CAGRP レコードを作成するために使用する INIT.CAGRP コマンドで行います。

CAGRP レコードには、区画データ・セットのメンバーの名前が入っています。このメンバーには、この CA グループに対してデータベース変更累積ユーティリティーを実行するための JCL を生成するために使用される骨組み JCL が含まれています。また、CAGRP レコードには、このグループに対応する変更累積データ・セットが再利用できるかどうかを指定する標識も含まれています。これには、このグループに対して維持される変更累積データ・セットの最大数とリカバリー期間の標識が含まれています。

変更累積実行レコード (CA)

CA レコードには変更累積データ・セットについての情報が含まれ、その状態は使用可能または使用中のいずれかです。それぞれの CAGRP レコードごとに、最大 1024 の CA レコードを入れることができます。

使用可能 CA レコードは、INIT.CA コマンドによって作成されます。この CA レコードには、データベース変更累積ユーティリティーをその後で実行したときの出力に使用するデータ・セットの、ボリューム通し番号とデータ・セット名が含まれます。使用可能 CA レコードは、REUSE パラメーターを指定して定義される CA グループについてのみ作成することができます。

使用中変更累積レコードは、データベース変更累積ユーティリティーの実行によって作成されます。このレコードは、データベース変更累積ユーティリティーの実行時に使用された、以前の使用可能 CA レコードでも、JCL から得た情報を含む新規レコードでも構いません。使用中変更累積レコードには、以下の情報が含まれます。

- データ・セット名
- ボリューム通し番号
- 変更累積ユーティリティーの実行時間
- データベース変更累積ユーティリティーが最後に処理したログ・ボリュームの停止時刻、あるいは、CA がログのサブセットを処理した場合は、次の実行で最初に処理するログの開始時刻

CA レコードを作成するには NOTIFY.CA コマンドを使用します。

データ・グループ・レコード (DBGRP、DBDSGRP、RECOVGRP)

DBDSGRP レコードを使用して、以下のタイプの名前の付いたグループを定義することができます。DBDSGRP、DBGRP、および RECOVGRP

名前の付いたグループは、次のとおりです。

- DBDSGRP。DBDS および DEDB エリアのグループ
- DBGRP。/STA、/STO、/DBR の各コマンドの DATAGROUP パラメーター内に指定することのできる、DL/I DB および DEDB エリアのグループ。
- RECOVGRP。データベース・リカバリーの目的で論理的に関連付けられている、DL/I DB および DEDB エリアのグループ。

すべてのグループは固有の名前を持つ必要があります。例えば、DBDSGRP は DBGRP と同じ名前を持つことはできません。

INIT.DBDSGRP、CHANGE.DBDSGRP、DELETE.DBDSGRP、および LIST.DBDSGRP コマンドを使用して、データ・グループの 3 つのタイプすべてを取り扱うことができます。

推奨事項: 任意のタイプのグループも /STA、/STO、および /DBR コマンドの DATAGROUP パラメーターで名前を付けることができますが、DBDSGRP を使用するのとは非効率的なため、お勧めできません。

関連資料: DBDS グループの詳細については、676 ページの『DBDS グループの考慮事項』を参照してください。

データベース・レコード (DB)

DBRC は、DL/I、高速機能、および HALDB (高可用性ラージ・データベース) データベース・レコードを別々に扱います。

これらのレコードのタイプ、その取り扱い、および内容について、以下で説明します。

DL/I データベース・レコード

DB レコードは、登録済みで、そのリカバリーが DBRC の制御下にあるデータベースを識別します。このレコードには、以下のように、データベースについての情報と関連するリカバリー情報が入っています。

- データベース名
- データベース・タイプ
- 共用レベルのデータベース
- データベースを使用するサブシステムのリスト
- 拡張エラー・キュー・エレメント (EEQE) カウンター
- データベースを許可された最初のサブシステムの IRLM ID (IRLM を使用する場合)

DBDS レコードは、そのリカバリーを DBRC が制御することになっている DBDS を識別します。このレコードには、以下のように、DBDS についての情報 (そのデータ・セット編成など) と関連するリカバリー情報が入っています。

- DBDS が属する CA グループの名前
- この DBDS のために維持されるイメージ・コピー・データ・セットの最大数
- イメージ・コピー・データ・セットを再利用すべきかどうかの指示
- この DBDS のためにイメージ・コピー・データ・セットが維持される期間
- 暗黙の基幹 JCL デフォルト・メンバーの名前
- 拡張エラー・キュー・エレメント (EEQE)
- この DBDS に対して実行されるユーティリティー用の JCL を生成するために使用される、基幹 JCL の区画データ・セットのメンバーの名前

DL/I データベースおよび DBDS を記述するために、DBRC は、以下の図に示すような論理的に関連したレコードを保守します。

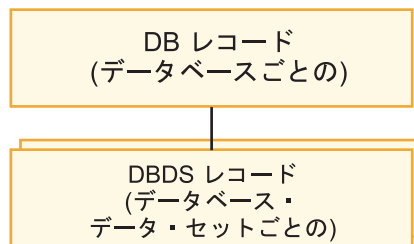


図 63. DBRC DLI レコード

HALDB レコード

HALDB は、RECON データ・セットの観点からは、HALDB マスター (TYPE=HALDB) と 1 つ以上の HALDB 区画 (TYPE=PART) で構成されます。

以下の図では、HALDB を表す RECON データ・セット・レコードの関係を示します。

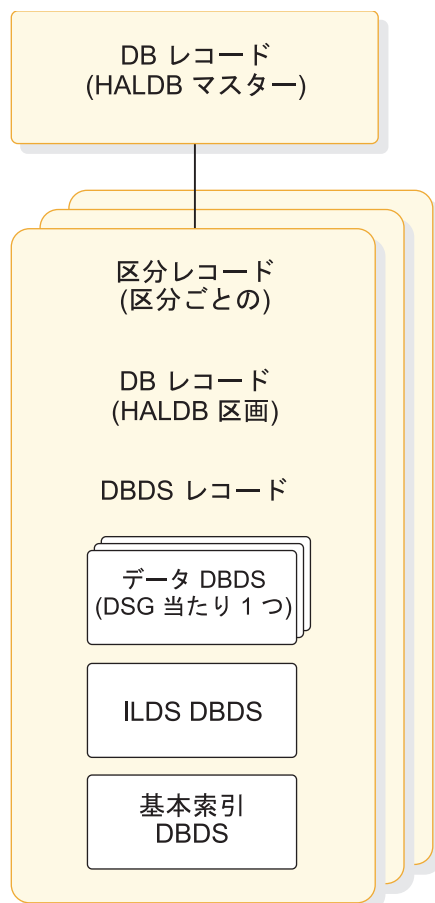


図 64. RECON データ・セット内の HALDB のレコード構造

HALDB マスター (DSPDBHRC)

RECON データ・セットは、DB ヘッダー・レコードを使用して HALDB 全体に関する情報を保管します。DB ヘッダー・レコードには以下のものがあります。

- HALDB マスター名
- グローバル DMB 番号
- 現在の変更バージョン番号
- TYPE=HALDB
- データ共用レベル
- リカバリー・グループ
- HALDB 区画選択出口ルーチン名
- データベースおよびデータ・セット編成
- OSAM データ・セットの最大データ・セット・サイズ
- 定義済みの最新の区画の区画 ID
- HALDB のリカバリー可能性
- HALDB 内の区画の数
- オンライン再編成の可能性
- データ・セット・グループ・メンバーの数

TYPE=HALDB DB レコードは HALDB に関する情報を保管します。アプリケーションは HALDB マスター・レベルで DB アクティビティを管理し、DBRC は HALDB 区画レベルで DB アクティビティを管理します。サブシステムは HALDB 区画を許可しますが、HALDB マスターは許可しません。

HALDB 区画

HALDB の各区画は、以下の RECON データ・セット・レコードで構成されます。

- 区画レコード (DSPPTNRC)

DSPPTNRC には、個々の区画に適用される情報が含まれます。HALDB 区画定義ユーティリティは、区画レコード情報を表示します。LIST コマンドは区画レコードを表示しません。

- 区画 DB レコード (DSPDBHRC)

DSPDBHRC は区画レベルで HALDB にアクセスします。TYPE=IMS DB レコードと同様に、HALDB 区画レコードの DB レコードはすべての共用情報とリカバリー情報を記録します。区画名は、このレコードにデータベース名フィールドを設定します。TYPE=PART がこのレコードに定義されています。以下のフィールドは、HALDB 全体にわたって、それぞれの区画ごとに同じ設定を持っています。

- グローバル DMB 番号
 - データ共用レベル
 - 区画のリカバリー可能性
 - HALDB マスター名
 - オンライン再編成の可能性
- 区画 DBDS レコード (DSPDSHRC)

DBDS には、HALDB の編成によって、それぞれの HALDB 区画ごとにデータ、索引、および ILDS データ・セットの 3 つのタイプがあります。複数のデータ DBDS が存在することはありますが、他のタイプはそれぞれ 1 つだけです。リカバリーすることができるのは、データ DBDS だけです。その他の DBDS は、HALDB 索引/ILDS 再作成ユーティリティー (DFSPREC0) を使用して再作成されます。

関連資料: DBDS レコードのデータ・セットおよび DDN 命名規則については、「IMS V15 データベース管理」を参照してください。

高速機能データベース・レコード

DEDB、エリア、およびエリア・データ・セット(ADS) を記述するために、DBRC は以下の図に示すような論理レコード構造を備えています。

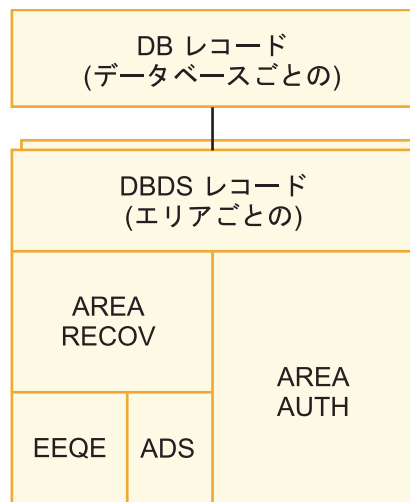


図 65. DBRC 高速機能データベース・レコード

DBRC は DB および DBDS レコードを使用して DL/I データベースと DEDB の両方を記述します。ただし、DBRC は高速機能 DBDS レコードに ADS リストを追加して、各 ADS についての情報を提供します。各 DEDB には複数のエリアが含まれ、その各エリアに最大 7 つの ADS が含まれます。

高速機能 DB レコードには DL/I DB レコード内の情報と同様の情報が含まれます。ただし、異なる点は、高速機能 DB レコードは DEDB を記述し、また許可サブシステムのリストは含みません。高速機能の場合、このリストは DBDS レコードの中にあります。このレコードはエリア許可レコードとエリア・リカバリー・レコードから構成されています。高速機能 DB レコードはこのリストに DBDS レコードとして表示されます。

関連資料: 「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」のサンプルの RECON リストを参照してください。

エリアが RECON データ・セットに登録されると、次の確認が行われます。

- DEDB 内のエリア名は固有である
- エリア内の ADS DD 名は固有である
- 1 エリアに 7 つを超えない ADS が定義されている

イメージ・コピー・レコード (IMAGE)

IMAGE レコードにはイメージ・コピー・データ・セットについての情報が含まれ、その状態は使用可能 または使用中 のいずれかです。

使用可能 IMAGE レコードは、INIT.IC コマンドによって作成されます。このレコードは、イメージ・コピー・ユーティリティーのその後の実行からの出力に使用されるデータ・セットを記述します。使用可能 IMAGE レコードを作成できるのは、REUSE パラメーターを指定して定義される DBDS についてのみです。

使用中 IMAGE レコードは、次のように、イメージ・コピー・ユーティリティーの実行によって作成されます。

- DBDS またはエリアが REUSE を指定して定義される場合、使用中 IMAGE レコードは、以前の使用可能 IMAGE レコードか、または再利用される使用中レコードです。
- DBDS またはエリアが NOREUSE を指定して定義される場合、それは JCL から取得したデータ・セット記述を持つ新規レコードです。

使用中 IMAGE レコードは、データ・セット記述に加えて、コピー操作のタイプを識別し、コピー操作の実行時間を含み、コピーのタイプによっては、コピー操作の停止時刻も含まれます。

イメージ・コピー・ユーティリティーに、イメージ・コピー・データ・セットおよび重複イメージ・コピー・データ・セットを作成するように要求すると、DBRC は、その両方についての情報を同じイメージ・コピー・レコードに記録します。最初のレコードは IC1 に指定され、重複レコードは IC2 に指定されます。

標準外イメージ・コピー・データ・セット (イメージ・コピー・ユーティリティーが作成しなかったもの) を作成する場合は、NOTIFY.UIC コマンドを使用して、その存在を RECON データ・セットに記録する必要があります。

高速機能 DEDB 変更ユーティリティーで変更された各エリアに、標準外イメージ・コピー (ユーザー・イメージ・コピーとも呼ばれる) が自動的に作成されます。これは、シャドウ・イメージ・コピー・データ・セット (IDS) を使用して行われます。

関連資料: IMAGE COPY レコードの詳細については、658 ページの『イメージ・コピー・ユーティリティー (DFSUDMP0、DFSUDMT0、DFSUICP0)』を参照してください。

再編成レコード (REORG)

DBRC は、ユーザーが以下のいずれかの方法を使用して登録済みデータベースまたは区画のデータ・セットを再編成するたびに、REORG レコードを作成します。

- DEDB 変更ユーティリティー (ALTRAREA 機能のみ)
- HALDB オンライン再編成
- HALDB 変更再編成


- HISAM 再編成再ロード・ユーティリティ
- HD 再編成再ロード・ユーティリティ

HALDB オンライン再編成の場合、REORG レコードは REORG のタイプがオンラインであることを示し、停止時刻を組み込み、そして再編成されたデータ・セットをリカバリーに使用できるかどうかを示します。DBRC は、ユーザーが再編成するデータベース・データ・セットごとに、RECON データ・セット内に REORG レコードを作成します。


HALDB 変更機能によって作成される REORG レコードは、そのレコードが HALDB 変更用に作成されたことを示すフラグが追加されることを除いて、HALDB オンライン再編成の REORG レコードと同じです。

同様に、高速機能 DEDB 変更ユーティリティによって変更されるエリアごとに、REORG レコードが作成されます。


関連概念:


 HALDB のオンライン再編成 (データベース管理)

関連タスク:

 オンライン HALDB データベースの定義の変更 (データベース管理)

関連資料:

 HD 再編成再ロード・ユーティリティ (DFSURGL0) (データベース・ユーティリティ)

 HISAM 再編成再ロード・ユーティリティ (DFSURRL0) (データベース・ユーティリティ)

ログ割り振りレコード (LOGALL)

LOGALL レコードは、対応するログ・データ・セットがオープンされている間に変更された登録済み DBDS を識別します。

DBRC は、それぞれの PRILOG レコードごとにログ割り振り (LOGALL) レコードを作成します。

データベース割り振りレコード (ALLOC)

高速機能の場合、DBRC は、エリアが更新のためにオープン状況に置かれているときに、ALLOC レコードを作成します。DL/I の場合、DBRC は、IMS の実行中に初めて、あるいは /DBRECOVERY コマンドを入力した後初めて IMS が DBDS を更新するときに、ALLOC レコードを作成します。このレコードには、その作成のタイム・スタンプと、対応するログのオープンのタイム・スタンプが含まれます。このタイム・スタンプによって、リカバリーに必要なデータベース変更レコードを含むログ・データ・セットが識別されます。DBDS がその後、割り振りを解除されると、DBRC は割り振り解除のタイム・スタンプを ALLOC レコードに追加します。

割り振りレコードが最も古いイメージ・コピー・レコードより古い場合、および DBRC ではその割り振りレコードがデータベースのリカバリーでは必要としなくなった場合、DBRC は自動的にその割り振りレコードを削除します。DBRC は、

ALLOC レコードを削除すると、関連した LOGALL レコードを変更します。これは、DBRC イメージ・コピー・ユーティリティーの出口ルーチン処理の一部です。DBDS に対する ALLOC レコードのこの自動削除は、以下のいずれかの条件の下では起こりません。

- ALLOC レコードが割り振り解除時刻を持っていない。

割り振り解除時刻は、データベースまたはエリアが /DBR コマンドを実行した場合に記録されます。それ以外の場合は、ALLOC レコードは暗黙の割り振り解除時刻としてログ・クローズ時刻を使用します。

- ALLOC レコードに関連した PRILOG レコードがオープン (STOPTIME がゼロになっています) になっているが、エラーのマークが付いていない。

このような場合は、DELETE.ALLOC コマンドを使用して、不要な ALLOC レコードを RECON データ・セットから削除することができます。DBRC は、オンラインで並行イメージ・コピー・ユーティリティーの出口ルーチンを処理しているときに割り振りタイム・スタンプを自動更新し、適時に割り振りタイム・スタンプを前進させます。

リカバリー・レコード (RECOV)

DBRC は、データベース・リカバリー・ユーティリティーを実行して DBDS をリカバリーするたびに、RECOV レコードを作成します。

RECOV レコードは、次の 2 つのタイプのリカバリーのいずれかを示します。

- DBDS の完全リカバリー。この場合、RECOV レコードにはリカバリーのタイム・スタンプが含まれます。
- タイム・スタンプ・リカバリー。この場合、RECOV レコードには、データベース・リカバリー・ユーティリティーの実行のタイム・スタンプと、DBDS がリカバリーされた時点のタイム・スタンプが含まれます。

サブシステム・レコード (SSYS)

RECON データ・セットは、サブシステム (SSYS) レコードを使用してデータ共有情報を記述します。

SSYS レコードは、IMS サブシステムが DBRC にサインオンするときに作成されます。この SSYS レコードには、次のように、サブシステムについての情報と、関連するリカバリー情報が入っています。

- サブシステムの名前およびタイプ (オンラインまたはバッチ)
- IRLM ID
- 異常終了フラグおよびリカバリー処理開始フラグ
- 許可されたデータベースのリスト
- サブシステム・エントリーと該当するログ・レコードを関係付けるタイム・スタンプ
- これが BPE ベースのサブシステムかどうかを示すフラグ。

RECON データ・セット内のレコードのマッピング

DBRC で RECON レコードのフォーマットを定義している DSECTS を、ADFSMAC と SDFS MAC 内で見つけるには、SDFS MAC=ALL を使用してジョブの生成を実行します。

メンバー名は、次のとおりです。

メンバー (DSECT) 名

DSECT がマップする RECON レコード

DSPALLRC

ALLOC レコード

DSPBKORC

BACKOUT レコード

DSPCAGRC

変更累積グループ (CAGRP) レコード

DSPCHGRC

変更累積 (CA) レコード

DSPDBHRC

データベース (DB) レコード

DSPDGRC

DBDSGRP レコード

DSPDSHRC

DBDS レコード

DSPIMGRC

イメージ・コピー (IC) レコード

DSPLGARC

LOGALL レコード

DSPLOGRC

PRILOG / SECLOG / PRISLD / SECSLD レコード

DSPOLDRC

PRIOLD/SECOLD レコード

DSPPTNRC

HALDB 区画レコード

DSPRCNRC

HEADER レコード

DSPRCR1

HEADEREXT レコード

DSPRCVRC

RECOV レコード

DSPRRGRC

REORG レコード

DSPRDTRC

RECON DMB テーブル・レコード

DSPSSRC

SUBSYS レコード

RECON データ・セットの保守

データ保全性と許容レベルのパフォーマンスを維持するために、RECON データ・セットで定期的に保守を実行する必要があります。

以下のトピックでは、関係するメインタスクについて説明します。

RECON データ・セットのバックアップ

RECON データ・セットは頻繁にバックアップしてください。それらはクリティカル・リソースです。RECON レコードで何らかの保守 (データベースの登録、変更累積グループの追加や削除など) を行なった後は、必ず RECON データ・セットのバックアップ・コピーを作成してください。

BACKUP.RECON コマンドを使用してバックアップを行ないます。このコマンドは、バックアップ保全性を保証するのに必要な RESERVE コマンド (バックアップ処理時に装置を予約する) を実行します。次に AMS REPRO コマンドを呼び出して、データ・セットをコピーします。BACKUP.RECON コマンドは RECON データ・セットの最初のコピーのみをコピーします。そのパラメーターは、コピーを 1 部作成するか 2 部作成するかを決めます。並列アクセス・モードでは、予約コマンドは実行されません。代わりに、RECON アクティビティは、アクティブなすべての DBRC インスタンスについて静止します。

並列アクセスが使用可能な場合、ユーザーには、DFSMSdss COPY または DUMP を使用してトランザクション VSAM (TVS) RECON のバックアップ・コピーを取るオプションがあります。DFSMSdss COPY または DUMP を使用した場合、TVS は一貫性のある、つまり「鮮明な」バックアップ・コピーが生成されるように自動的にデータ・セットを静止させます。

関連資料: BACKUP.RECON コマンドの追加情報については、「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」を参照してください。

不要な RECON レコードの削除

次の方法により、不要な RECON レコードを削除することができます。

関連概念:

665 ページの『イメージ・コピー・データ・セットおよび GENMAX のリカバリ期間』

無関係のレコードの自動削除

通常、データベース関連のレコードの場合、多くのレコード保守を実行する必要はありません。

RECON データ・セットはイメージ・コピーについて通知されると、最も古い使用中 IMAGE レコードを削除または再利用し、その後の IMAGE レコードが最も古

い IMAGE レコードになります。(現在) 最も古い IMAGE レコードより前の開始時刻を持つ RECOV レコードと REORG レコード、および最も古い時刻より前の DEALLOC 時刻を持つ ALLOC レコードは現在関係ないため、RECON データ・セットから削除されます。これが、イメージ・コピーのクリーンアップ処理です。

無関係の IMAGE レコードが RECON データ・セットから削除されると、すべてのアクティブ ALLOC レコードは、DBDS またはエリアの最も古いイメージ・コピーに基づいて、リカバリーに必要な最初のログ・ボリュームの時刻に更新されます。クリーンアップ処理は、無関係の ALLOC レコードが削除すると、それに関連した LOGALL レコードの状態を変更します。LOGALL レコードに関連付けられた ALLOC レコードがすべて削除されると (これは多くのデータベースの多数のイメージ・コピーにわたって生じる可能性があります)、その LOGALL レコードに関連付けられた PRILOG レコードは非アクティブになります。

PRILOG の圧縮

オープン PRILOG レコードの圧縮とは、PRILOG レコードのすべての非アクティブ・データ・セット・エントリーを削除することです。

データ・セット・エントリーは、以下の基準のいずれよりも古くなった場合に、非アクティブであると定義されます。

- ログの保存期間
- そのログ上で更新されるデータベースの最も古い割り振り (ALLOC)
- オンライン IMS の最も早い再始動チェックポイント

PRILOG レコード圧縮は、非アクティブ・データ・セット・エントリーを、ログ上の最も古い ALLOC まで、あるいはデータ・セット・エントリーの最初のギャップまで削除します。ギャップは、OLDS がまだアーカイブされていない場合に生じます。

非アクティブ・データ・セット・エントリーは、アクティブ PRILOG から削除されると、開始時刻がログの開始時刻と同じで、停止時刻が最後に削除された非アクティブ・データ・セット・エントリーの停止時刻と同じ、1 つのダミー・データ・セット・エントリーに圧縮されます。

PRILOG 圧縮は、以下の 2 つの方式を使用して行うことができます。

自動 PRILOG 圧縮

PRILOG レコード圧縮は、OLDS がアーカイブされた後で、自動的に試行されます。

圧縮には、PRILOG レコード内の非アクティブ・データ・セット・エントリーをすべて削除することも含まれます。該当する場合は、SECLOG、PRISLD、および SECSLD レコード内の対応するエントリーも削除されます。

手作業による PRILOG 圧縮

DELETE.LOG INACTIVE コマンドを使用すれば、PRILOG レコード圧縮を手作業で開始することができます。このコマンドは、アクティブ PRILOG レコードから非アクティブ・データ・セット・エントリーを削除し、非アクティブ PRILOG レコードの全体を削除します。

PRILOG が圧縮されない場合は、下記のいずれかの理由でメッセージ DSP1150I を受け取ります。

- EARLIEST CHECKPOINT

このタイム・スタンプが PRILOG の最初の DSN エントリー内にある場合は、圧縮は不可能です。多くの場合、問題の原因となっている、再始動に必要な最も古いチェックポイントが、メッセージ・キューを再作成するチェックポイントとなっています。その場合は、SNAPQ チェックポイントをとってください。この SNAPQ チェックポイントを記録する OLDS がアーカイブされると、最も古いチェックポイントのタイム・スタンプが更新されます。

- EARLIEST ALLOC TIME

LOGALL レコードにリストされているデータベースが、最も古い割り振りタイム・スタンプ順になっています。LOGALL レコードにリストされている最も古い割り振り時刻が、リストの最初のデータベースに関連づけられます。

並行イメージ・コピー (CIC) およびオンライン・イメージ・コピー (OIC) は、割り振り時刻を更新して、リカバリーに必要な最も古いログを反映するようにしています。ログ上の、CIC を開始する前のチェックポイント ID およびチェックポイント・カウントを使用して、新しい割り振り時刻をどこに設定するかを決定します。割り振り時刻を進めていない場合は、チェックポイント ID およびカウントが DSN エントリーに記録されていることを確認してください。

LOGALL レコード内のデータベースに対する古い割り振りが、圧縮が失敗する原因になることがあります。定期的にデータベースのイメージ・コピーをとって、古い割り振りが削除されるようにしてください。

- LOG RETENTION TIME

DBRC が RECON データ・セット内にログ・レコードを保持する最小時間を示します。

手作業によるログ・レコードの削除

DELETE.LOG INACTIVE または CLEANUP.RECON コマンドを使用して、ログ・レコードを手作業で削除できます。

DBRC は、ログ・データ・セットを記述する RECON レコード (PRILOG および SECLOG レコード) を自動的に削除することはありません。この設計により、ログ・データ・セットに関連したどの RECON レコードを削除するかを制御することができます。リカバリーには不要となった PRILOG および SECLOG レコードは、定期的に削除する必要があります。

PRILOG レコードの現在のサイズが、PRILOG レコードのリストに印刷されます。

非アクティブ PRILOG レコード、SECLOG レコード、およびそれに関連した LOGALL レコードを削除するには、DELETE.LOG INACTIVE コマンドを使用します。

よくある問題として、異常終了したジョブ (そのログをクローズせずに) によって作成された、古いオープン (停止時刻がゼロでない) PRILOG レコードの問題があります。こうしたオープン・ログは、LIST.LOG コマンドに OPEN パラメーターを





指定して実行すれば、検出することができます。DELETE.LOG コマンドで STARTIME パラメーターを指定して使用すると、古い、不要なオープン PRILOG ファミリー・レコードを除去することができます。

あるいは、CLEANUP.RECON コマンドを使用して、古いログ情報や有効期限が切れたログ情報、CA データ・セット、およびリカバリーに関連した情報 (イメージ・コピー、割り振り、再編成、およびリカバリーのレコード) を RECON データ・セットから無条件に削除します。指定されたタイム・スタンプまたは保存期間を使用してリカバリー関連データおよび非アクティブ・ログが削除され、オープン・ログと、この時点より後のクローズ時刻にクローズされるログの両方に対して、PRILOG 圧縮が強制実行されます。

削除によって空いたスペースが VSAM で再利用されない場合があるため、ログ・レコードを削除しても、RECON データ・セットがいっぱいになるのを防止することにはなりません。ただし、RECON データ・セットをバックアップまたは再編成する前にログ・レコードを削除した場合は、バックアップまたは再編成中にそのスペースを再利用することができます。

重要: 必要なりカバリー情報とログ情報が RECON データ・セットから削除されるのを避けるために、最初に RECON データ・セットのコピーに対して CLEANUP.RECON コマンドを実行し、結果を検証してください。結果からは、このジョブに必要な時間の長さが分かります。その後、この情報を使用して、このコマンドをアクティブな RECON データ・セットに対して実行する時期を判別できます。

関連資料:

-  CLEANUP.RECON コマンド (コマンド)
-  DELETE.LOG コマンド (OLDS 用) (コマンド)
-  DELETE.LOG コマンド (RLDS および SLDS 用) (コマンド)
-  LIST.LOG コマンド (PRILOG ファミリー用) (コマンド)

RECON データ・セットの再編成

RECON データ・セットは定期的に再編成する必要があります。RECON データ・セット内のレコード・キーの多くは、日時が組み込まれています。DBRC が IMS ログとデータベース・アクティビティを記録することによって、CI と CA の分割が起こり、その結果パフォーマンスが低下することがあります。

さらに、不要なレコードを削除しても、空いたスペースを VSAM が再利用するとは限らないため、RECON データ・セットがいっぱいにならないようにすることはできません。

RECON データ・セットを再編成するための CHANGE.RECON の使用

CHANGE.RECON コマンドを使用すると、RECON データ・セットの動的割り振りを使用している場合は、RECON データ・セットをオンラインで再編成することができます。その場合、スペアの RECON データ・セットが使用できるようになっている必要があります。

この状態で、REPLACE オプションを指定して CHANGE.RECON コマンドを実行できます。その結果、DBRC は、(CHANGE コマンドで指定された) アクティブ RECON データ・セットをスペア・データ・セットにコピーし、再編成します。VSAM は CI と CA の分割をすべて削除し、元の FREESPACE 属性を復元します。

CHANGE コマンドも、古い RECON データ・セット (再編成を必要としたもの) の割り振りを解除します。ただし、このデータ・セットを削除し、再定義するには、その前に、このデータ・セットを使用している他のすべてのサブシステムがこれを割り振り解除するのを待つ必要があります。データ・セットをその元の名前と同じ名前前で再定義すると、オンライン・システムでスペア・データ・セットとして使用することができます。この処理を繰り返すと、2 番目のアクティブ RECON データ・セットを再編成することができます。データ・セットを割り当てるために JCL を使用する場合は、動的割り振り解除は起こりません。

動的割り振りを使用しないか、スペア RECON データ・セットが使用できない場合、再編成を行うには、RECON データ・セットにアクセスするオンライン・サブシステムおよび他のすべてのサブシステムがその RECON データ・セットの割り振りを解除するまで待つ必要があります。

再編成前の RECON データ・セットのバックアップ

再編成の前と後には、以下の手順を使用して RECON データ・セットをバックアップしてください。

1. RECON データ・セットを別のデータ・セット名をもつ一時データ・セットにコピーする。
2. コピーしたデータ・セットが破壊されていないか検査する。
3. 元のデータ・セットを削除する。
4. 同じデータ・セット名を使用して、元のデータ・セットを再定義する。
5. 一時データ・セットを、これらの元のデータ・セットにコピーし直す。

RECON データ・セットの再編成の手順

ログをクリーンアップして RECON データ・セットを再編成するには、この手順に従います。この処理は、DBRC が二重モードで、RECON1 (コピー 1 として)、RECON2 (コピー 2 として)、およびスペアを使用して行なわれていることを前提としています。

1. LIST.LOG OPEN コマンドを実行し、オープン・ログをすべて識別する。識別されたログのうち、オープンすべきもの、クローズまたは削除すべきものを決める。クローズまたは削除すべきログはいずれもクローズする。
2. DELETE.LOG INACTIVE コマンドを実行し、非アクティブまたは未使用のログをすべて削除する。これで、RECON データ・セットの再編成の準備ができました。
3. RECON1 の REPLACE オプションを指定して CHANGE.RECON を実行する。これにより DBRC は次のことを実行します。
 - a. メッセージ DSP0380I を出して RECON データ・セットの再構成の開始を通知する

- b. メッセージ DSP0388I を出して再構成時にアクティブなサブシステムを識別する
 - c. メッセージ DSP0381I を出して処理が完了したことを通知する
4. 廃棄された RECON データ・セットを置き換える。詳しくは、723 ページの『廃棄された RECON データ・セットの置き換え』を参照してください。

これで RECON データ・セットが再編成され、ログがクリーンな状態になりました。

損傷した RECON データ・セットの置き換え

残っている唯一の現行 RECON データ・セットで入出力エラーが発生した場合、DBRC はジョブを停止します。それ以外の入出力エラーが検出されなければ、現在その RECON データ・セットを使用している他のジョブはいずれも実行を継続します。

RECON データ・セットの 1 つで入出力エラーが起こり、RECON データ・セットが 2 つ存在する場合、DBRC はスペア・データ・セットを見付けようとします。スペアが使用可能な場合、DBRC は入出力エラーのない RECON データ・セットをスペア RECON データ・セットにコピーします。次に、DBRC はそのスペアをコピー 2 RECON データ・セットとして設定します。

推奨事項: スペア RECON データ・セットがエラーのあった RECON データ・セットと置き換わった後、できるだけ早く廃棄された RECON データ・セットを再定義してください。入出力エラーのある RECON を直ちに置き換えれば、RECON データ・セットがすべて失われたことによるサブシステム障害は発生しません。詳しくは、723 ページの『廃棄された RECON データ・セットの置き換え』を参照してください。

DBRC がスペア RECON データ・セットを見付けることができず、INIT.RECON コマンドの STARTNEW パラメーターが指定されている場合、DBRC は 1 つの RECON データ・セットで処理を続けます。それ以外の場合、DBRC は現行ジョブを完了しますが、スペア RECON データ・セットが定義されるまで、新規ジョブを開始しません。

RECON データ・セットのリカバリー

RECON データ・セットのリカバリーのステップは、RECON データ・セットのセットアップ方法と状態によって異なります。

以下の段落では、RECON データ・セットのリカバリーを行うためのさまざまなシナリオについて説明します。

スペア **RECON** データ・セットが使用可能な場合

RECON データ・セットで入出力エラーが発生し、スペア・データ・セットが使用可能である場合、DBRC はエラーのない RECON データ・セットをスペアにコピーし、次にそのスペアを活動化します。

ただし、RECON データ・セットのエラーを分析したい場合は、廃棄された RECON データ・セットを削除し再定義する前に、後の問題診断用にそのコピーを作成してください。

スペア RECON データ・セットが使用できない場合

スペア RECON データ・セットが使用できない場合は、現在実行中のすべてのジョブが、RECON データ・セットを単一モードで使用して処理を続行します。INIT.RECON または CHANGE.RECON コマンドに STARTNEW パラメーターが指定されている場合、DBRC は、1 つの RECON データ・セットだけで新しいジョブを開始することを許可します。ただし、システムの保全性を危うくするので、この方法はお勧めできません。

1 組の RECON データ・セットの中のデータ・セットの 1 つが DBRC によって使用不可になった場合は、使用できない RECON データ・セットの割り振りを解除し、新しいスペアを割り振る必要があります。

両方の RECON データ・セットが使用可能でない場合

両方の RECON データ・セットが使用できなくなることはあまり考えられません。しかし、もし両方の RECON データ・セットが使用不可になった場合は、この手順に従ってください。

1. RECON データ・セットへのアクセスを必要とするすべてのジョブを停止する。
2. オプション: ユーザーが両方の RECON データ・セットにアクセスできる場合は、AMS REPRO コマンドを使用して RECON データ・セットをバックアップする。
3. AMS ユーティリティーを使用して RECON データ・セットを削除し、再定義する。
4. AMS REPRO コマンドを使用して RECON データ・セットの 1 つを復元する。
5. AMS REPRO コマンドを使用して、もう一方の RECON データ・セットを初めから復元する。
6. LIST.RECON コマンドを使用して RECON データ・セットの 1 つをリストする。リストを調べて、ステップ 2 でバックアップを行った後に更新された DBDS を評価してください。どの DBDS が更新されたかを判別できない場合は、すべてが更新されたと想定してください。
7. INVALID パラメーターを指定して CHANGE.IC コマンドを使用してステップ 6 ですべての該当する DBDS について、エラーのあるすべてのイメージ・コピー・レコードにマークを付ける。
8. ステップ 6 で該当するすべての DBDS のイメージ・コピーを作成する。
9. BACKUP.RECON コマンドを使用して RECON データ・セットのバックアップ・コピーを作成する。

これで RECON データ・セットは復元され、各データベースと再同期されました。

過度に多くのデータベースを制御することがなければ、次の手順に従ったほうが簡単かもしれません。

1. RECON データ・セットへのアクセスを必要とするすべてのジョブを停止する。
2. 新規 RECON データ・セットを定義する。
3. それらの RECON データ・セットを初期化する。

4. 環境を登録する。最近初期化されたが未使用の RECON データ・セットのバックアップ・コピーを常に使用可能状態にしておきます。
5. すべてのデータベースのイメージ・コピーを取る。

最後に、通常の操作に進む前に、新規 RECON データ・セットをクリーンアップします。例えば、オープンしている、古くなった OLDS をすべて、NOTIFY.PRILOG コマンドによってクローズします。

廃棄された RECON データ・セットの置き換え

DBRC が、RECON データ・セットが廃棄されていることを検出するのは、何らかのアクティビティーが生じて、DBRC がその RECON データ・セットにアクセスすることになった場合のみです。オンラインでもバッチでも、複数の IMS サブシステムをユーザーが持っている場合は常に、ユーザーは DBRC の複数インスタンスを持つことができます。DBRC のすべてのインスタンスが、変更が生じたことを検出し、それらのインスタンスが廃棄データ・セットの割り振りを解除するまで、ユーザーは廃棄された RECON データ・セットを削除し、再定義することはできません。

DBRC は、再構成時にアクティブなサブシステムをメッセージ DSP0388I にリストします。このメッセージにより、RECON データ・セットの状況変更を検出する際にユーザーの支援が必要と思われるサブシステムを識別することができます。

入出力エラーが発生した後で、または CHANGE.RECON REPLACE コマンドとともに RECON データ・セットを再定義する場合は、この手順に従います。

1. すべてのバッチ・ジョブを、DBRC を使用して終了できるようにする。
2. 自動損失通知 (Automatic Loss Notification) をアクティブにしていない場合は、すべてのオンライン・サブシステムで LIST.RECON STATUS を実行する。このコマンドを実行すると、オンライン・サブシステムは同一のコピー 1 およびコピー 2 RECON データ・セットを取得し、廃棄された RECON データ・セットの割り振りを解除します。自動損失通知をアクティブにしている場合、すべてのサブシステムは廃棄された RECON データ・セットの割り振り解除を自動的に通知されます。RECON 損失通知について詳しくは、725 ページの『RECON 損失の通知』を参照してください。
3. AMS DELETE コマンドを使用して廃棄された RECON データ・セットを削除する。
4. AMS DEFINE コマンドを使用して、空の VSAM KSDS データ・セットとして RECON データ・セットを再作成する。RECON データ・セットを最初に作成する際に使用したものと同一手順を使用します。700 ページの『RECON データ・セットの作成』を参照してください。

RECON データ・セットの修復

RECON データ・セットで発生する可能性のある特定の問題を修正して、データ保全性を確保することができます。

通常、DBRC の制御外で作成される RECON データ・セットでは、特定の問題が発生する可能性があります。ほとんどの場合、問題は、DMB テーブル・レコードと、RECON ヘッダー、データベース、および高速機能エリアのレコード内の

DMB 番号の矛盾に関連しています。REPAIR.RECON コマンドを使用すると、DMB テーブル・レコードの再作成または作成、およびレコード内の矛盾する DMB 番号の修正が可能です。

REPAIR.RECONコマンドで DMBTABLE キーワードを使用すると、DMB テーブル・レコード全体と、DMB テーブル・レコードの作成または再作成に使用される DMB 番号が含まれているすべてのレコードが処理されます。以下の修復を行うには、このコマンドを使用してください。

- 矛盾する DMB 番号の訂正。これには、以下の項目の確認が必要です。
 - RECON ヘッダー、データベース、および高速機能エリアのレコード内の上位ビットがオンであること
 - 区画データベース・レコード内の DMB 番号が HALDB マスター・レコード内の DMB 番号に一致すること
 - 高速機能エリア・レコード内の DMB 番号が DEDB レコード内の DMB 番号に一致すること
- RECON リスト内の登録済みデータベースの数の更新
- DMB テーブル・レコードの再作成、または DMB テーブル・レコードが存在しない場合はその作成

REPAIR.RECON コマンドで DMBNUM キーワードを指定すると、矛盾する DMB 番号の検出と修正も行われますが、適用範囲を、すべてのデータベース・レコード、特定のタイプのデータベース・レコード (HALDB、高速機能、または IMS)、特定のデータベース・レコード、または RECON ヘッダーに制限することができます。

- ALL キーワードが指定されると、すべてのデータベース・レコードが処理され、RECON ヘッダー、データベース、および高速機能エリアのレコードで上位ビットがオンになり、区画データベース・レコード内の DMB 番号は HALDB マスター・レコード内の DMB 番号に一致し、さらに高速機能エリア・レコード内の DMB 番号は DEDB レコード内の DMB 番号に一致するようになります。
- RECON キーワードと一緒に RESET キーワードが指定されると、DMB テーブル・レコードが存在する場合、DMB# フィールドおよび LAST USED# フィールドは 32767 に設定されます。次のデータベースが RECON データ・セットに登録され、これらのフィールドが 32767 に設定されると、次の割り当てに使用可能な DMB 番号の DMB テーブル・レコードでの検索は、最後に追加された DMB 番号からではなく、1 から始まります。
- CHECK キーワードは、DMB 番号を修正することなく、それらとの矛盾が存在するかどうかを判別します。UPDATE キーワードは、DMB 番号のエラーの検出と訂正の両方を行います。

DBRC は、データベースまたはエリアが無許可である場合に限り、DMB 番号の矛盾を修正するようレコードを変更することができます。コマンドが処理されると、許可されたすべてのデータベースまたはエリアに対してメッセージが発行されます。識別されたデータベースまたはエリアを未許可にしてから、コマンドを再実行します。

推奨事項:

- 読み取られて変更されることがあるレコードの数が原因で、REPAIR.RECON コマンドは、RECON データ・セットの可用性に影響する可能性があります。最初に、RECON データ・セットのバックアップ・コピーに対して REPAIR.RECON を実行して、この影響を判断します。次に、実際の RECON データ・セットへのアクセスがごくわずかなときに、このコマンドを実行します。
REPAIR.RECON DMBNUM では、最初に、バックアップ・コピーに対して CHECK キーワードを指定してコマンドを実行し、修正が必要な DMB 番号があった場合は、実際の RECON に対して UPDATE キーワードを指定してコマンドを実行してください。
- RECON のアップグレード前に RECON データ・セットの保全性を確保して、コマンドを実行するように、REPAIR.RECON DMBNUM CHECKUP を、1 年に少なくとも 1 回実行されるようにスケジュールします。エラーが検出された場合は、UPDATE キーワードを使用してコマンドを再実行して、エラーを修正してください。

関連資料:

 REPAIR.RECON コマンド (コマンド)

RECON 損失の通知

同じ IMSplex を使用して、RECON データ・セットに割り振られているすべての DBRC インスタンスには、エラーが発生すると、IMS Structured Call Interface (SCI) を通じてエラーが通知されます。次に、廃棄された RECON に割り振られている DBRC データ・セットがすべて割り振り解除されます。

並列 RECON アクセスでは、DBRC は再構成プロセスの間、他の DBRC インスタンスが RECON データ・セットにアクセスできないようにする必要があります。その時点では、他のすべての DBRC が並列モードで再アクセスします。

DBRC は、IMSplex 名を、DBRC SCI 登録出口ルーチン (DSPSCIX0) か、または EXEC パラメーター IMSPLEX から取得します。異なる IMSplex を使用するか、または IMSplex を使用せずに RECON データ・セットへのアクセスを行うと、リジェクトされ、メッセージ DSP1136A が出されます。それ以外の場合は、RECON 損失の通知が自動的に呼び出されます。

RECON のセットに関連付けられた IMSplex を変更したい場合は、CHANGE.RECON コマンドで IMSPLEX または NOPLEX パラメーターを使用します。DBRC インスタンスがアクティブの場合、IMSplex から RECON への関連付けを変更するには、以下のようにします。

1. 現行 IMSplex 上のすべての DBRC アクティビティが停止するまで待つ。
2. IMSplex 名を変更するデータベース・リカバリー管理ユーティリティ・ジョブを実行依頼する。

並列アクセスが使用可能であり、CHANGE.RECON NOPLEX コマンドを使用して RECON データ・セットから IMSplex 名を除去したい場合は、最初に、CHANGE.RECON ACCESS(SERIAL) コマンドを使用して並列アクセスを使用不可にする必要があります。

3. DSPSCIX0 出口ルーチンの IMSplex 名を変更する。
4. 新規の IMSplex SCI が作動可能であることを確認する。

関連資料:

- IMSplex の入門情報については、20 ページの『IMSplex の概要』を参照してください。
- DBRC SCI 登録出口ルーチン (DSPSCIX0) の詳細については、「IMS V15 出口ルーチン」を参照してください。
- IMSPLEX EXEC パラメーターの詳細については、「IMS V15 システム定義」を参照してください。

RECON データ・セットに行われた変更のトラッキング

管理上の目的で、RECON データ・セットに行われた変更のログを維持することをお勧めします。ユーザー独自の出口ルーチンを提供することが可能であり、この出口ルーチンは、RECON レコードが変更または読み取られるたびに呼び出されます。

この出口ルーチンは RECON 入出力出口ルーチン (DSPCEXT0) とも呼ばれ、これにより、RECON データ・セットに対する変更をジャーナル形式で追跡することができます。

関連資料: RECON 入出力出口ルーチンの詳細については、「IMS V15 出口ルーチン」を参照してください。

第 43 章 DBRC のヒント

このトピックでは、頻繁に使用される手順について、以下のようなトピックで作業本位の説明を行いません。

タイム・スタンプを元の現地時間で出力するための RECON データ・セットの変更

RECON データ・セットのリスト作成でタイム・スタンプ用のオフセット値がリストされない場合は、レコードが作成されたときのオフセット値が現行オフセット値と異なっていた可能性があります。このようなレコードを変更または削除するために DBRC コマンドを使用すると、「レコードが検出されない」状態になります。なぜなら、コマンドのタイム・スタンプ・パラメーターにオフセットが含まれていない場合は、現行オフセットが適用されるからです。

作成されたときの元の現地時間でタイム・スタンプを出力するよう、RECON データ・セットを変更するには、以下のどちらかを使用します。

```
CHANGE.RECON TIMEFMT(0,0,P,4,PERM)
```

または

```
CHANGE.RECON TIMEFMT(0,0,P,2,PERM)
```

これにより、RECON データ・セットは、レコードが当初作成されたタイム・スタンプを、適切なオフセットが適用された区切り付きの 4 桁または 2 桁の年表記のフォーマットで出力するよう、永続的に変更されます。リストされたタイム・スタンプは、以下に示すように、コマンドの中で使用することができます。

```
CHANGE.PRIOLOG STARTIME('2007.063 16:11:58.123456 -08:00') -  
DSSTART('2007.063 16:11:58.123456 -08:00') ERROR RLDS
```

RECON データ・セット内の最後の SLDS 停止時刻の検出

以下の手順を使用して、GENJCL.USER コマンドを使用し、RECON データ・セット内の最後の SLDS 停止時刻を検出します。この手順は、IMS がまだ稼働しているときに使用でき、すでに PRISLD がクローズされている場合でも使用できます。

1. JCL 実行メンバーの骨組みを作成する。

次に示すのは、SELSLDS という名前のメンバーの例です。

```
%SELECT SLDS(%USSID, LAST)  
%ENDSEL  
SLSDDD BEG=%SLDSTIM  
          END=%SLDETIM  
          VOL=%SLDVOLS  
          UNT=%SLDUNIT  
          DSN=%SLSDSN
```

2. GENJCL.USER コマンドを発行して、一緒に実行される実行メンバーを指定する。

以下は GENJCL.USER コマンドのサンプルで、このコマンドがメンバー SELSLDS を指定して実行されることを示します。

```
GENJCL.USER MEMBER(SELSDS) NOJOB USERKEYS((%USSID,'IMS1')) -  
JCLOUT(SYSPRINT)
```

以下に示すのは、SLDS のリストです。

```
PRISLD                                RECORD SIZE=    1104  
START = 07.064 16:58:06.000000      *  SSID=IMS1    VERSION=10.1  
STOP  = 07.064 17:00:04.656844      #DSN=6  
GSGNAME=**NULL**  
FIRST RECORD ID= 0000000000000001    PRILOG TOKEN= 0  
  
DSN=IMSVS.SLDSP.IMS1.D07064.T1658060.V00      UNIT=SYSDA  
START = 07.064 16:58:06.000000      FIRST DS LSN= 0000000000000001  
STOP  = 07.064 16:58:16.108103      LAST  DS LSN= 0000000000000128  
FILE SEQ=0001 #VOLUMES=0001  
CHECKPOINT TYPES=80: SIMPLE=Y SNAPQ=N DUMPQ=N PURGE=N FREEZE=N  
  
VOLSER=000000 STOPTIME = 07.064 16:58:16.108103  
CKPTCT=1  CHKPT ID = 00.000 00:00:00.000000  
LOCK SEQUENCE#= 000000000000  
  
DSN=IMSVS.SLDSP.IMS1.D07064.T1658161.V00      UNIT=SYSDA  
START = 07.064 16:58:16.108103      FIRST DS LSN= 0000000000000129  
STOP  = 07.064 16:59:00.886236      LAST  DS LSN= 0000000000000BE9  
FILE SEQ=0001 #VOLUMES=0001  
CHECKPOINT TYPES=00: SIMPLE=N SNAPQ=N DUMPQ=N PURGE=N FREEZE=N  
  
VOLSER=000000 STOPTIME = 07.064 16:59:00.886236  
CKPTCT=0  CHKPT ID = 07.064 16:58:06.603513  
LOCK SEQUENCE#= 000000000000  
  
DSN=IMSVS.SLDSP.IMS1.D07064.T1659008.V00      UNIT=SYSDA  
START = 07.064 16:59:00.886236      FIRST DS LSN= 0000000000000BEA  
STOP  = 07.064 16:59:15.330532      LAST  DS LSN= 000000000000111A  
FILE SEQ=0001 #VOLUMES=0001  
CHECKPOINT TYPES=00: SIMPLE=N SNAPQ=N DUMPQ=N PURGE=N FREEZE=N  
  
VOLSER=000000 STOPTIME = 07.064 16:59:15.330532  
CKPTCT=0  CHKPT ID = 07.064 16:58:06.603513  
LOCK SEQUENCE#= 000000000000  
  
DSN=IMSVS.SLDSP.IMS1.D07064.T1659153.V00      UNIT=SYSDA  
START = 07.064 16:59:15.330532      FIRST DS LSN= 000000000000111B  
STOP  = 07.064 16:59:28.219762      LAST  DS LSN= 00000000000016A5  
FILE SEQ=0001 #VOLUMES=0001  
CHECKPOINT TYPES=00: SIMPLE=N SNAPQ=N DUMPQ=N PURGE=N FREEZE=N  
  
VOLSER=000000 STOPTIME = 07.064 16:59:28.219762  
CKPTCT=0  CHKPT ID = 07.064 16:58:06.603513  
LOCK SEQUENCE#= 000000000000  
  
07.064 17:00:18.768831                LISTING OF RECON                PAGE 0005  
DSN=IMSVS.SLDSP.IMS1.D07064.T1659282.V01      UNIT=SYSDA  
START = 07.064 16:59:28.219762      FIRST DS LSN= 00000000000016A6  
STOP  = 07.064 17:00:04.440567      LAST  DS LSN= 0000000000001907  
FILE SEQ=0001 #VOLUMES=0001  
CHECKPOINT TYPES=08: SIMPLE=N SNAPQ=N DUMPQ=N PURGE=N FREEZE=Y  
  
VOLSER=000000 STOPTIME = 07.064 17:00:04.440567  
CKPTCT=1  CHKPT ID = 07.064 16:58:06.603513  
LOCK SEQUENCE#= 000000000000  
  
DSN=IMSVS.SLDSP.IMS1.D07064.T1659282.V02      UNIT=SYSDA  
START = 07.064 17:00:04.440567      FIRST DS LSN= 0000000000001908
```

```

STOP = 07.064 17:00:04.656844          LAST DS LSN= 00000000000019A0
FILE SEQ=0001      #VOLUMES=0001
CHECKPOINT TYPES=00: SIMPLE=N SNAPQ=N DUMPQ=N PURGE=N FREEZE=N

VOLSER=000000 STOPTIME = 07.064 17:00:04.656844
CKPTCT=0      CHKPT ID = 07.064 17:00:04.417809
LOCK SEQUENCE#= 000000000000

```

以下に示すのは、前述した GENJCL.USER コマンドのサンプルを実行した結果の出力です。

```

SLDSDD BEG=07064170004440567-0800
        END=07064170004656844-0800
        VOL=000000
        UNT=SYSDA
        DSN=IMSVS.SLDSP.IMS1.D07064.T1659282.V02

```

GENMAX の調整 (その値に達したか値が大きすぎる場合)

イメージ・コピー・ユーティリティを実行する前に、検査終了処理により、イメージ・コピー・データ・セットの再利用中に以下の状態が真であると判別される場合があります。そのような場合は、GENMAX 値が大きすぎるか小さすぎます。いずれの場合でも、コマンドを発行して問題を修正する必要があります。

- 状態 1: GENMAX 値が小さすぎる
 - GENMAX 値に達した。
 - リカバリー期間内であるため、最も古いイメージ・コピーを再利用できない。
 - 使用可能なイメージ・コピーが存在しない。
 - メッセージ DSP0063I が表示された。
 - GENMAX を大きくして追加のイメージ・コピー・データ・セットを定義する。
 - リカバリー期間を変更して、最も古いイメージ・コピー・データ・セットを使用できるようにする。

CHANGE.DBDS コマンドを実行して GENMAX および INIT.IC パラメーターを増加し、追加のイメージ・コピー・データ・セットを定義します。

1. 以下のジョブの中の 1 つに類似した JCL ジョブを作成します。これらの例は、DBD 名が THISDB、エリア名が AREA1、直前の GENMAX 値が 2 であると想定しています。

```

//CHNGDBDS JOB
//SYSIN DD *
        CHANGE.DBDS DBD(THISDB) AREA(AREA1) -
                    GENMAX(4)
/*//INITIC JOB
//SYSIN DD *
        INIT.IC DBD(THISDB) DDN(DDN1) -
                ICDSN(IMS.*.NEWICDSN)
        INIT.IC DBD(THISDB) DDN(DDN1) -
                ICDSN(IMS.*.NEWICDSN2)
/*

```

CHANGE.DBDS RECOVPD を実行して、リカバリー期間を短縮します。

```
//CHNGDBDS JOB
//SYSIN DD *
      CHANGE.DBDS DBD(THISDB) AREA(AREA1) RECOVPD(10)
/*
```

2. ジョブを実行します。
 3. JES ログを検査して、ジョブが正常に実行されたことを確認します。
 4. LIST コマンドを実行して新しい GENMAX 値を調べます (オプション)。
 5. イメージ・コピー・ジョブを実行します。
- 状態 2: GENMAX 値が大きすぎる
 - GENMAX に達していないため、使用中イメージ・コピー・データ・セットを使用することができない。
 - 使用できる使用可能イメージ・コピー・データ・セットが存在しない。
 - メッセージ DSP0084I が表示された。
 - リカバリー期間が定義されていない。
 - INIT.IC コマンドを実行して、識別されたデータベース・データ・セットまたはエリア・データ・セットに新しいイメージ・コピー・データ・セットを定義します。

あるいは、以下の作業を行うこともできます。

- CHANGE.DBDS コマンドを実行して GENMAX 値を小さくします。

重要: CHANGE.DBDS コマンドを使用して GENMAX を小さくした場合、新しい GENMAX 値は、最も古いイメージ・コピーを削除できるかどうかに関係なく記録されます。それらのコピーはリカバリー期間内であるからです。

関連資料:

 CHANGE.DBDS コマンド (コマンド)

GRPMAX の調整 (その値に達したか値が大きすぎる場合)

変更累積ユーティリティを実行する前に、DBRC は、変更累積データ・セットが再利用されており、GRPMAX 値の変更が必要であると判別する場合があります。コマンドを発行してこの問題を修正する必要があります。

状態 1 - GRPMAX 値が小さすぎる

- GRPMAX 値に達した。
- 最も古い変更累積データ・セットは、リカバリー期間内にあるため再利用できない。
- 使用可能な変更累積データ・セットがない。
- メッセージ DSP1229A が表示された。

状態 1 は、次のいずれかのアクションによって解決されます。

- GRPMAX 値を増加し、追加の変更累積データ・セットを定義する。
- 最も古い変更累積データ・セットを使用できるように、リカバリー期間を変更する。

GRPMAX 値を増加し、追加の変更累積データ・セットを定義するには、以下の例に従って CHANGE.CAGRP コマンドと INIT.CA コマンドを発行してください。これらの例では、CA グループ名が CAGRP1 で、前の GRPMAX 値が 10 だったと想定しています。

```
//CHNGCAG JOB

//SYSIN DD *
    CHANGE.CAGRP GRPNAME(CAGRP1) -
        GRPMAX(12)
/*

//INITIC JOB

//SYSIN DD *
    INIT.CA GRPNAME(CAGRP1) -
        CADSN(IMS.*.NEWCADSN1)
    INIT.CA GRPNAME(CAGRP1) -
        CADSN(IMS.*.NEWCADSN2)
/*
```

最も古い変更累積データ・セットを使用できるようにリカバリー期間を変更するには、以下に示すように、RECOVPD キーワードを指定して CHANGE.CAGRP コマンドを実行してください。

```
//CHNGCAG JOB
//SYSIN DD *
    CHANGE.CAGRP GRPNAME(CAGRP1) RECOVPD(10)
```

状態 2 - GRPMAX 値が大きすぎる

- GRPMAX 値に達していないため、使用中の変更累積データ・セットを使用できない。
- 使用可能な変更累積データ・セットがない。
- メッセージ DSP0085I が表示された。
- リカバリー期間が定義されていない。

状態 2 は、次のいずれかのアクションによって解決されます。

- INIT.CA コマンドを実行して、識別された CA グループの新しい変更累積データ・セットを定義する。
- CHANGE.CAGRP コマンドを実行して、GRPMAX 値を小さくする。

重要: CHANGE.CAGRP コマンドを使用して GRPMAX 値を小さくすると、最も古い変更累積データ・セットはリカバリー期間内にあるため、それらを削除できるかどうかに関係なく GRPMAX 値が記録されます。

関連資料:

 CHANGE.CAGRP コマンド (コマンド)

オープン・オンライン PRILOG レコードのクローズ

タイム・スタンプ・リカバリー中に、DBRC が、リカバリーには不要なオープン割り振りがあることを報告した場合は、個々の割り振りを削除するより、オープンしている PRILOG レコードをクローズするほうが簡単な場合があります。オープンしている PRILOG レコードをクローズするには、NOTIFY.PRILOG コマンドを使用します。

オープン PRILOG レコードがオンライン IMS サブシステムで検出され、その PRILOG レコードが現在実行されている IMS 用ではない場合、それは、このオンライン IMS の最後の OLDS がまだアーカイブされていないことを示しています。OLDS がもう使用不能であり、オープン PRILOG レコードをクローズする必要がある場合は、以下のコマンドを使用して、PRILOG 内にダミーの DSN エントリーを作成することができます。

```

/* This command creates dummy dataset */
NOTIFY.PRILOG DSN(DUMMY) STARTIME('2010.072 10:23:53.979935 -07:00') -
    FIRSTREC(00) RLDS SSID(IMS1) VOLSER(VOL001)

/* This command closes PRILOG record-Stoptime of 0's becomes runtime */
NOTIFY.PRILOG RUNTIME('2010.072 11:23:53.980001 -07:00') -
    STARTIME('2010.072 10:23:53.979935 -07:00') -
    LASTREC(500) RLDS SSID(IMS1)

/* This command changes last PRILOG, DUMMY, to ERROR */
CHANGE.PRILOG STARTIME('2010.072 10:23:53.979935 -07:00') ERROR -
    DSSTART ('2010.072 10:24:43.286431 -07:00')

```

以下の例では、NOTIFY.PRILOG コマンドを実行して PRILOG レコードをクローズする前に、RECON データ・セットに現れる PRILOG レコードです。

```

PRILOG                                RECORD SIZE=      464
START = 2010.072 10:23:53.979935 -07:00*  SSID=IMS1  VERSION=15.1
STOP  = 0000.000 00:00:00.000000 -07:00    #DSN=2
GSGNAME=**NULL**
FIRST RECORD ID= 0000000000000001          PRILOG TOKEN= 0
EARLIEST CHECKPOINT = 2010.072 10:23:54.517056 -07:00

DSN=IMSVS.RLDSP.IMS1.D07072.T1023539.V00          UNIT=SYSDA
START = 2010.072 10:23:53.979935 -07:00  FIRST DS LSN= 0000000000000001
STOP  = 2010.072 10:24:32.697834 -07:00  LAST  DS LSN= 00000000000006D1
FILE SEQ=0001    #VOLUMES=0001

VOLSER=000000 STOPTIME = 2010.072 10:24:32.697834 -07:00
CKPTCT=1    CHKPT ID = 2010.072 10:23:54.517056 -07:00
LOCK SEQUENCE#= 000000000000

DSN=IMSVS.RLDSP.IMS1.D07072.T1024326.V00          UNIT=SYSDA
START = 2010.072 10:24:32.697834 -07:00  FIRST DS LSN= 00000000000000D2
STOP  = 2010.072 10:24:43.286431 -07:00  LAST  DS LSN= 0000000000000831
FILE SEQ=0001    #VOLUMES=0001

VOLSER=000000 STOPTIME = 2010.072 10:24:43.286431 -07:00
CKPTCT=0    CHKPT ID = 2010.072 10:23:54.517056 -07:00
LOCK SEQUENCE#= 000000000000

LOGALL
START  = 2010.072 10:23:53.979935 -07:00*
EARLIEST ALLOC TIME = 2010.072 10:24:23.704405 -07:00
DBDS ALLOC=1          -DBD-    -DDN-    -ALLOC-
                    DBVHDJ05 CJVHDG1E  1

```

以下の例では、NOTIFY.PRILOG コマンドを実行して PRILOG レコードをクローズした後に、RECON データ・セットに現れる PRILOG レコードです (違いが強調表示されています)。

```

PRILOG                                RECORD SIZE=      624
START = 2010.072 10:23:53.979935 -07:00*  SSID=IMS1  VERSION=15.1
STOP  = 2010.072 11:23:53.980001 -07:00    #DSN=3
GSGNAME=**NULL**
FIRST RECORD ID= 0000000000000001          PRILOG TOKEN= 0
EARLIEST CHECKPOINT = 2010.072 10:23:54.517056 -07:00

```

```
DSN=IMSVS.RLDSP.IMS1.D07072.T1023539.V00          UNIT=SYSDA
START = 2010.072 10:23:53.979935 -07:00  FIRST DS LSN= 0000000000000001
STOP  = 2010.072 10:24:32.697834 -07:00  LAST  DS LSN= 000000000000006D1
FILE SEQ=0001      #VOLUMES=0001
```

```
VOLSER=000000 STOPTIME = 2010.072 10:24:32.697834 -07:00
CKPTCT=1      CHKPT ID = 2010.072 10:23:54.517056 -07:00
LOCK SEQUENCE#= 000000000000
```

```
DSN=IMSVS.RLDSP.IMS1.D07072.T1024326.V00          UNIT=SYSDA
START = 2010.072 10:24:32.697834 -07:00  FIRST DS LSN= 000000000000006D2
STOP  = 2010.072 10:24:43.286431 -07:00  LAST  DS LSN= 00000000000000831
FILE SEQ=0001      #VOLUMES=0001
```



```
VOLSER=000000 STOPTIME = 2010.072 10:24:43.286431 -07:00
CKPTCT=0      CHKPT ID = 2010.072 10:23:54.517056 -07:00
LOCK SEQUENCE#= 000000000000
```

```
DSN=DUMMY          ERROR UNIT=3400
START = 2010.072 10:24:43.286431 -07:00  FIRST DS LSN= 0000000000000000
STOP  = 2010.072 11:23:53.980001 -07:00  LAST  DS LSN= 00000000000001F4
FILE SEQ=0001      #VOLUMES=0001
```

```
VOLSER=VOL001 STOPTIME = 2010.072 11:23:53.980001 -07:00
CKPTCT=0      CHKPT ID = 0000.000 00:00:00.000000 -07:00
LOCK SEQUENCE#= 30E100000000
```

```
LOGALL
START = 2010.072 10:23:53.979935 -07:00*
EARLIEST ALLOC TIME = 2007.072 10:24:23.704405 -07:00
DBDS ALLOC=1          -DBD-      -DDN-      -ALLOC-
                    DBVHDJ05 CJVHGD1E 1
```

関連資料:

-  NOTIFY.PRILOG コマンド (RLDS 用) (コマンド)
-  LIST.LOG コマンド (PRILOG ファミリー用) (コマンド)

サブシステム・レコード (SSYS) の処理

データベースの許可が失敗し、RECON データ・セット内のデータベースおよびすべてのサブシステム・レコードをリストすると、サブシステム名 (SSID) がどのように作成されたのか、不思議に思われるかもしれません。

以下のリストでは、SSID のさまざまな側面について説明します。

- RECON サブシステム・レコードにおける SSID の命名規則

SSYS レコードは、サインオン呼び出しが DBRC に出されたときに作成されます。SSID は固有でなければなりません。サブシステム名を作成する際の通常の規則は、次のとおりです。

IMS および DBCTL サブシステム = IMSCTRL SYSGEN マクロの IMSID 値
(4 文字)

BATCH および UTILITY サブシステム = JOBNAME

XRF = IMS システムの RENAME (リカバリー・サービス・エレメント)
(アクティブおよび代替)

例外は次のとおりです。

ONLINE IMAGE COPY - XXXTZZZZ ここで、

XXXX is the DMB number of the database + DCB number translated
to 0-9,A-Z
ZZZZ は制御領域 IMSID です。

オンライン・イメージ・コピーは、RECON データ・セット内に独自のサブシステム・レコードを作成する唯一の BMP です。

- バッチ・バックアウト

バッチ・バックアウトがバッチ・サブシステムによって作成されたログを処理する場合、使用される SSID はバックアウトされるサブシステムのジョブ名です。バッチ・バックアウト・ユーティリティー・ジョブのジョブ名は使用されません。

バッチ・バックアウトがオンライン・サブシステムによって作成されたログを処理する場合は、バックアウト・ユーティリティー・ジョブのジョブ名が SSID として使用されます。

- サブシステム・レコードの削除

RECON データ・セット内の特定のサブシステム・レコードが不要になった場合は、次のコマンドを使用して SSYS を削除することができます。

```
CHANGE.SUBSYS SSID(XXXX) STARTRCV  
CHANGE.SUBSYS SSID(XXXX) ENDRECOV <---(removes auth from DB)  
DELETE.SUBSYS SSID(XXXX)
```

- サブシステム・レコード・サイズ

サブシステム・レコードのサイズは増加します。サブシステム・レコードが増大して、最大レコード・サイズの 16M を超えると、IMS は異常終了します。

DB/AREA レコードから SSYS 間の許可の不整合を除去する方法

データベース許可を変更すると、あるいは許可が失敗すると、DB またはエリアは、存在していないサブシステムに許可され、バックアウトは不要であることを、LIST.DB コマンドが示しています。

以下のコマンドを使用して、許可を取り除きます。また、LIST.SUBSYS は、データベース/エリア名は許可されていると示しているのに、LIST.DB ではそれが示されていない場合は、次のコマンドを使用して、サブシステム・レコード内のリストからデータベース/エリア名を除去することもできます。

```
CHANGE.DB DBD(dbname) SSID(XXXX) UNAUTH (または)  
CHANGE.DB DBD(dbname) AREA(areaname) SSID(XXXX) UNAUTH
```

ログの変更累積プロセスの再始動

変更累積が誤ってログの処理を停止し、問題の原因が、ARC STARTED 状態のままの OLDS までトレースされます。アーカイブ・ジョブが実行された CPU で z/OS の IPL が行われ、OLDS の状況はこの状態のままになっています。

GENJCL.ARCHIVE ジョブには、ARC STARTED 状態の OLDS は含まれていません。その OLDS を作成したオンライン IMS は、この CPU 上では実行されていません。

IMS の再始動時に、オンライン IMS がアーカイブ・ジョブと同じ CPU で実行される場合、IMS は、ARC STARTED 状況にあるすべての OLDS の状況を ARC NEEDED にリセットします。OLDS は次のアーカイブ・ジョブに組み込まれます。この問題を回避するためには、アーカイブ・ジョブを、対応する IMS システムと同じ CPU で実行してください。

手操作による介入が必要な場合は、以下のコマンドを使用して、OLDS の状況をアーカイブが必要 (ARNEEDED) に変更します。

```
CHANGE.PRILOG OLDS(DFSOLPXX) ARNEEDED SSID(XXXX)
```

処理対象がないとされた場合の変更累積作業の再始動

変更累積が、処理すべきログが見付からない旨のメッセージを出しました。リストの検討により、処理が必要なログが多数あることが判明します。

DBRC は、有効なイメージ・コピー時刻以降、それぞれのDBDS ごとに割り振りを満足させる必要があるログをすべて選択します。DBRC が処理したログを追跡するための手段として、ログ・ボリューム開始時刻順にログが並べられます。DBRC は使用可能なすべてのログを処理し、エラーのあるログやオープン・ログを検出した場合、あるいはアーカイブされていない OLDS を検出した場合はログ・ボリュームのリストを打ち切ります。検出された状態 (オープン・ログなど) を示すメッセージが出されます。

最後に実行した変更累積レコードが、ログの SUBSET がそのレコードへの最後の入力であったことを示している場合、STOPTIME は使用不能な最初のログ・ボリュームの開始時刻を反映します。RECON データ・セットのリストの中から、このログ・ボリュームのタイム・スタンプを見つけてください。この開始時刻を持つログ・ボリュームが存在しない場合は、アーカイブされていない OLDS があります。

次の時間軸は、このオープン OLDS 条件を示しています。

```
T1--T2---T3--T4-T5-----T6-----T7-----T8----T9---
```

```
|---A1---D1---|-----A4-----// (open)  
Log1 (DSN1)    (OPEN OLDS)
```

```
  |--A2--|  |--A3-|  
  Log2   Log3
```

GENJCL.CA が T7 で実行される場合、これは Log1(DSN1) を含み、T4 の STOPTIME を持つ SUBSET です。T4 は使用不能な最初のログの開始時刻です。この JCL の実行後、GENJCL.CA を実行しようとする、必ず、2 つのメッセージが出されます。1 つは、アーカイブされていない OLDS が存在するというもので、もう 1 つは、オープン OLDS がクローズされ、アーカイブされるまで、処理すべきログはないというものです。

ログ・データ・セットを移動する

DBRC は IMS ログの情報を RECON データ・セットに記録します。この情報には、ログ・データ・セット名 (DSN)、その開始時刻と終了時刻、およびボリューム通し番号 (VOLSER) が含まれます。

ログの破損を回避する最善の方法は、カタログ式ログ・データ・セットおよび DBRC CATDS オプションを使用することです。

非カタログ式ログ・データ・セットの場合は、CHANGE.PRILOG または CHANGE.SECLOG コマンドを使用して、DBRC に変更を通知します。

関連資料: 「IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンド」の CHANGE.PRILOG および CHANGE.SECLOG コマンドを参照してください。

データ・セットをカタログする

CHANGE.RECON または INIT.RECON コマンドで CATDS または NOCATDS キーワードを使用して、イメージ・コピー、変更累積、およびログ・データ・セットがカタログまたは SMS が管理されているかどうかを示すことができます。

CHANGE.RECON および INIT.RECON コマンドで CATDS または NOCATDS のいずれかを指定すると、それに応じて RECON データ・セットのヘッダー・レコードが更新されます。カタログに関してこの 2 つのコマンドで唯一異なるのは、INIT.RECON は IMS 領域に必要な RECON データ・セット (3) のそれぞれのセットごとに 1 回しか実行できない点です。

これらのデータ・セットがカタログされるように指定するには、次のように入力します。

```
//CHGRECON JOB
:
//SYSIN DD *
    CHANGE.RECON CATDS
/*
```

または、次のように指定できます。

```
//INITRCON JOB
:
//SYSIN DD *
    INIT.RECON CATDS
/*
```

CHANGE.RECON コマンドで CATDS が有効であるようにするには、データ・セットをカタログ済みとして初期化しておく必要があります。

CHANGE.RECON または INIT.RECON コマンドで NOCATDS を指定すると、これらのデータ・セットは、そのカタログ状況とは無関係に、カタログ済みとしては取り扱われません。

これらのデータ・セットがカタログ済みとして取り扱われないように指定するには、次のように入力します。

```
//CHGRECON JOB
:
//SYSIN DD *
    CHANGE.RECON NOCATDS
/*
```

または、次のように指定できます。

```
//INITRCON JOB
:
//SYSIN DD *
      INIT.RECON NOCATDS
/*
```

関連資料: 詳しくは、IMS V15 コマンド 第 3 巻: IMS コンポーネントおよび z/OS コマンドを参照してください。

テスト環境で複数コールド・スタートを行なう

テスト環境でしばしば、IMS のコールド・スタートを行いたい場合があります。IMS をコールド・スタートするには、最後の OLDS をクローズする必要があります。

OLDS をクローズする必要がない場合は、次のコマンドを使用して OLDS をクローズし、RECON データ・セットでその OLDS にアーカイブ済みのマークを付けて、再利用できるようにすることが可能です。

```
NOTIFY.PRILOG STARTIME(%OLDOTIM) RUNTIME(070682226022-0800) -
      LASTREC(670) OLDS(DFSOLP02) SSID(IMS1)
CHANGE.PRILOG OLDS(DFSOLP02) ARCHIVED SSID(IMS1)
```

PRIOLD レコードのそれぞれの OLDS ごとのエントリー (例えば、DFSOLP00、DSPOLP01、および DSPOLD02) は、OLDS が使用されるときに構築されます (RECON データ・セット内にまだそれらのエントリーが存在していない場合)。また、OLDS を RECON データ・セットから削除する必要がある場合は、次のコマンドを使用することができます。

```
DELETE.LOG OLDS(DFSOLP00) SSID(IMS1)
DELETE.LOG OLDS(DFSOLP01) SSID(IMS1)
DELETE.LOG OLDS(DFSOLP02) SSID(IMS1) LASTCLOS
```

IMS によって最後に使用された OLDS を削除するには、LASTCLOS が必要であることに注意してください。最後の DD 名エントリーが除去されると、PRIOLD レコードも削除されます。

コマンドを実行する前の PRIOLD レコードは、次のとおりです。

```
PRIOLD
  SSID=IMS1          # DD ENTRIES=3
  EARLIEST CHECKPOINT = 07.068 20:55:58.622862

  DDNAME=DFSOLP00   DSN=IMSTESTL.IMS01.OLDSP0
  START = 07.068 20:55:57.984378          FIRST DS LSN= 0000000000000001
  STOP  = 07.068 20:56:39.570062          LAST  DS LSN= 00000000000006D1
  LOCK SEQUENCE# = 000000000000
  STATUS=ARC COMPLT                                FEOV=YES  AVAIL
  PRILOG TIME=07.068 20:55:57.984378          ARCHIVE JOB NAME=JT205639
  VERSION=10.1

  DDNAME=DFSOLP01   DSN=IMSTESTL.IMS01.OLDSP1
  START = 07.068 20:56:39.570062          FIRST DS LSN= 00000000000006D2
  STOP  = 07.068 20:56:53.689006          LAST  DS LSN= 0000000000000831
  LOCK SEQUENCE# = 000000000000
  STATUS=ARC COMPLT                                FEOV=YES  AVAIL
  PRILOG TIME=07.068 20:55:57.984378          ARCHIVE JOB NAME=JT205653
  VERSION=10.1
```

```

DDNAME=DFSOLP02 DSN=IMSTESTL.IMS01.OLDSP2
START = 07.068 20:56:53.689006 FIRST DS LSN= 0000000000000832
STOP = 00.000 00:00:00.000000 LAST DS LSN= 0000000000000000
LOCK SEQUENCE# = 000000000000
STATUS=ACTIVE FEOV=NO AVAIL
PRILOG TIME=07.068 20:55:57.984378
VERSION=10.1

```

コマンドを実行して、DFSOLP02 をクローズし、アーカイブ済みのマークを付けた後の PRIOLD レコードは、次のようになります (違いは強調表示されています)。

```

PRIOLD
SSID=IMS1 # DD ENTRIES=3
EARLIEST CHECKPOINT = 07.068 20:55:58.622862

DDNAME=DFSOLP00 DSN=IMSTESTL.IMS01.OLDSP0
START = 07.068 20:55:57.984378 FIRST DS LSN= 0000000000000001
STOP = 07.068 20:56:39.570062 LAST DS LSN= 00000000000006D1
LOCK SEQUENCE# = 000000000000
STATUS=ARC COMPLT FEOV=YES AVAIL
PRILOG TIME=07.068 20:55:57.984378 ARCHIVE JOB NAME=JT205639
VERSION=10.1

DDNAME=DFSOLP01 DSN=IMSTESTL.IMS01.OLDSP1
START = 07.068 20:56:39.570062 FIRST DS LSN= 00000000000006D2
STOP = 07.068 20:56:53.689006 LAST DS LSN= 0000000000000831
LOCK SEQUENCE# = 000000000000
STATUS=ARC COMPLT FEOV=YES AVAIL
PRILOG TIME=07.068 20:55:57.984378 ARCHIVE JOB NAME=JT205653
VERSION=10.1

DDNAME=DFSOLP02 DSN=IMSTESTL.IMS01.OLDSP2
START = 07.068 20:56:53.689006 FIRST DS LSN= 0000000000000832
STOP = 07.068 22:26:02.200000 LAST DS LSN= 000000000000029E
LOCK SEQUENCE# = 000000000000
STATUS=ARC COMPLT FEOV=NO AVAIL
PRILOG TIME=07.068 20:55:57.984378
VERSION=10.1

```

潜在的な RECON データ・セット・エンキュー問題の削減

以下のリストでは、RECON データ・セットのエンキュー問題の原因となる可能性があるいくつかのシナリオを説明します。それらの問題を回避するためのヒントも記載されています。

- 共用 DASD 環境のシナリオ

共用 DASD 環境での操作では、RECON データ・セットのエンキュー問題で最も一般的な原因は、RECON データ・セットと同じボリューム上にある、それ自体の ICF カタログ内にそれぞれの RECON データ・セットをカタログするという勧告に従わなかったことによるものです。

順次アクセスを使用しており、しかも各 RECON データ・セットを RECON データ・セットと同じボリューム上にあるそのデータ・セット自体の ICF カタログにカタログしており、それでも問題が解消しない場合は、GRS、(または同等の) RESERVE 変換リストを検査して、SYSIGGV2 および DSPURI01 QNAME の処理方法を判別してください。この 2 つの組み合わせによっては、デッドロックを生じることがあります。

- 非共用 DASD 環境のシナリオ

非共有 DASD 環境で操作中に、問題が生じる場合、その原因は、デッドロックにあるのではなく、競合や低いパフォーマンスにあると考えられます。以下、この状態にあるときに調査すべきことをいくつか示します。

- ジョブが RECON データ・セットを保持している時間を最小限にする。

この時間を最小限にする 1 つの方法は、RECON データ・セットにアクセスする際に DBRC が使用する LSR バッファ・プールを調整することです。オンラインとバッチに使用する値が入っている CSECT、DSPBUFFS (順次アクセスのみ) があります。これをザップして値を変更する方法については、「IMS V15 出口ルーチン」の『バッファ・サイズ指定機能 (DSPBUFFS)』を参照してください。デフォルトは、ユーザーの使い方からすれば低い値かもしれませんが。並列アクセスの場合は、代わりに IDGSMSEX RLS_MAX_POOL_SIZE 定義を調べてください。

- RECON データ・セットがあるボリュームの他の内容を分析する。他の入出力アクティビティからの干渉 (およびその逆の干渉) を防ぐため、これらのボリュームを分離することを考慮してください。RECON データ・セットをハイパフォーマンス・キャッシュ DASD、あるいはことによると固体素子 DASD に置くことを考慮してください。

関連資料: 詳しくは、695 ページの『順次アクセス・モードでの RECON データ・セットのデッドロック状態の回避』を参照してください。

第 4 部 IMS システム・リカバリー

拡張回復機能 (XRF) または IMSplex を使用すると、障害が発生したときに、エンド・ユーザー・サービスを迅速に再開できるようになり、オンライン・トランザクション処理の継続を維持することが可能になります。


第 44 章 拡張回復機能の概要

拡張回復機能 (XRF) 複合システムを使用すると、障害が発生したときに、エンド・ユーザー・サービスを迅速に再開できるようになり、オンライン・トランザクション処理の継続を維持することが可能になります。

関連概念:

11 ページの『拡張回復機能の実行』

135 ページの『XRF 複合システムでのリカバリー』

 拡張回復機能の使用による IMS のリカバリー (オペレーションおよびオートメーション)

XRF の概要

XRF は、インストール・システムが使用できる既存のプログラム内の一連の機能であり、選択されたエンド・ユーザーに対して高水準の可用性、信頼性、およびサービスの継続性を達成します。XRF は、オンライン・トランザクション処理の継続性に対するニーズに応えます。

XRF を構成するために IMS とともに機能する、必要な一連の IBM ソフトウェア・プロダクトは次のとおりです。

- z/OS
- DFSMS
- VTAM
- USERVAR を使用する XRF 複合システムのためのネットワーク制御プログラム (NCP)
- USERVAR を使用する XRF 複合システムのためのシステム・サポート・プログラム (SSP)

XRF の高可用性へのアプローチは、次の 2 つの前提に基づきます。

- 多くの IMS のインストール・システムでは、計画停止と計画外の停止の両方を最小限にしようと努めます。これらのインストール・システムでは、優先順位の高い作業に対してサービスを向上させるために、余分にリソースを割り当てようとします。
- ある環境で障害が発生させる欠陥が、別の環境で必ずしも障害が発生させるとは限りません。

XRF によって停止が発生しなくなるわけではありません。計画外の停止 (ハードウェアやソフトウェアの障害など) がすべてなくなっても、計画停止 (保守、構成変更、マイグレーションなど) は引き続き行われます。XRF によって連続可用性が提供されるわけではありませんが、計画停止および計画外の停止がエンド・ユーザーに与える影響を少なくします。

IMSplex 内の XRF

XRF は、ローカル・システム上で使用されるときと同じ方法で、IMSplex 内で使用することが可能です。XRF 代替は、IMSplex 内で定義し、共通サービス層 (CSL) へのアクセス権限を持つ必要があります。

Structured Call Interface (SCI) は、XRF 代替が常駐する z/OS イメージ上に常駐する必要があります。

XRF システムは、その IMSID を使用して、Operations Manager (OM) で IMS コマンドへの登録を行います。XRF 代替システムまたはアクティブ・システムにコマンドが経路指定されると、IMSID を ROUTE() で使用する必要があります。

所有リソースの場合、XRF システムは、リソース所有者として RSENAME を使用して、そのリソースを Resource Manager (RM) に定義します。XRF 代替システムのテークオーバーで、リソースの所有権をテークオーバーします。

XRF を使用して全 IMSplex をリカバリーすることはできませんが、IMSplex 内の個別 IMS をリカバリーすることは可能です。

HALDB OLR は XRF 代替システムでは開始できません。HALDB OLR サポートについて詳しくは、「IMS V15 データベース管理」を参照してください。

XRF の利点を生かせるインストール・システムのタイプ

データ処理インストール・システムの中で、サービスの中断に特に影響されやすいインストール・システムには、金融機関、信用調査会社、製造工程コントロール・センターなどがあります。このようなインストール・システムでは、サービスの低下によって、エンド・ユーザーの不満と事業上の損失をもたらす場合があります。

対話型作業負荷アプリケーションおよび重要なビジネス・アプリケーションが発達したために、可用性の問題が多く、IMS のインストール・システムでの主要な関心事になりました。例えば、シフトの切り替えの間に業務に就いているかまたは就いていない従業員をログに記録するのに IMS を使用するインストール・システムでは、毎就業日に 2 回の短期間の高可用性を必要とします。IMS を使用してトランザクションを処理する銀行では、午前 6 時から午後 10 時まで高可用性が求められます。1 週間に 7 日、1 日 24 時間絶え間ないプロセスが必要なインストール・システムでは、1、2 時間の停止も認められません。XRF はこのようなインストール・システムでの停止をなくすことはできませんが、エンド・ユーザーへのサービスの中断のほとんどを除去できます。

一般に、XRF の利点を生かせるインストール・システムには、次のものがあります。

- 単一の動作環境内にある複数の大型システム
- IMS を使用している多数のエンド・ユーザー
- エンド・ユーザーに対する非常に高い可用性に必要な要件

このようなインストール・システムは、通常、よく系統立てられ、厳しく管理された環境、つまり、XRF をインストールするのに必要な効率的なインストール管理が存在する環境で動作します。

XRF 概念と用語

IMS システム障害に対して、エンド・ユーザー・サービスをほんの短時間しか中断させない高可用性を達成するのは困難な場合があります。再始動を行う前に、終了プロセスを実行し、障害の原因を判別しなければなりません。領域を再始動して、端末セッションを再確立しなければならない場合は、再始動自体、非常に時間がかかることがあります。

XRF 複合システムを使用すれば、障害が発生したときにエンド・ユーザー・サービスを迅速に再開できます。この構成は、優先プロセッサで通常動作する主サブシステム (アクティブ IMS と呼ばれる) と、同じプロセッサまたは異なるプロセッサでアクティブ IMS のアクティビティを同時にトラッキングする代替サブシステム (代替 IMS と呼ばれる) からなります。

アクティブ IMS は XRF がない場合と同じ方法で IMS 作業負荷を処理します。通常のプロセスおよびロギング以上の作業は行いません。アクティブ IMS は、ログによって、自らのアクティビティを代替 IMS に知らせます。ログ、再始動データ・セット (RDS)、ISC リンクを使用した監視メカニズムの組み合わせにより、アクティブ IMS の問題が代替 IMS に警告されます。アクティブ IMS は、代替 IMS のアクティビティを認識しません。

代替 IMS は、アクティブ IMS で入力された IMS トランザクションを処理しませんが、障害の兆候がないか、アクティブ IMS をモニターします。代替 IMS は、アクティブ IMS のリソース変更を記録し、アクティブ IMS の状況と一致するようにその状況を連続的に変更して、制御ブロックとバッファを更新します。この代替 IMS はテークオーバーの準備ができていて状態にあるので、従属領域が始動され、データ・セットが割り振られてオープンされ、セッションが確立されるのを待たずに、作業をアクティブ IMS から代替 IMS へ迅速にシフトできます。アクティブ IMS から代替 IMS への作業負荷のこのようなシフトは、テークオーバー と呼ばれます。テークオーバーが発生した後、新しい代替 IMS を確立できます。

IMS 障害が発生したり、z/OS オペレーティング・システムまたは中央演算処理装置複合システム (CPC) 全体に影響を及ぼす障害が発生すると、代替 IMS がアクティブ IMS の作業負荷を引き受けます。エンド・ユーザーが適切な端末を使用している場合は、テークオーバーの発生時に、サービスはほとんど中断されません。このときの影響は、単一システム・イメージの場合と同様です。エンド・ユーザーはどの IMS が使用中であるかを知る必要はなく、IMS ホスト・システムでの切り替えが行われたことさえ知らないかもしれません。

アクティブ IMS が異常終了すると、テークオーバーが発生する場合があります。また、異常終了以外にも、IMS インストール時によく発生する次のような停止の原因によって、テークオーバーが発生することもあります。

- 監視で検出できる IMS 障害
- 監視で検出できる z/OS 障害、ループ、または待ち状態
- CPC の障害
- TPEND 出口を引き起こす結果となる VTAM 障害
- STATUS 出口を引き起こす結果となる内部リソース・ロック・マネージャー (IRLM) の障害

また、XRF を呼び出し、エンド・ユーザーへのサービスの停止を最小限にして、実稼働環境へ計画変更を導入することもできます。

主要なサービス・エレメントの 1 つは、中央演算処理装置複合システム (CPC) です。CPC は、主記憶装置、1 つ以上の中央処理装置、タイマー、およびチャネルから構成される、ハードウェアの物理的集合です。CPC は、単一オペレーティング・システムの制御下で稼働します。これには、ユニプロセッサの場合もあり、マルチプロセッサ (双頭プロセッサも含む) の場合もあります。

XRF 複合システムが USERVAR を使用するときに必要な、3705、3725、または 3745 通信コントローラーについて記載する場合、ここでは、37x5 通信コントローラー という用語を使用します。

注: 3745 コントローラーからマイグレーションしようとするインストール・システムには、以下のオプションがあります。

- XRF を継続して使用するために、3745 コントローラーの置き換えとして、Communication Controller for Linux on System z (CCL) を使用する。
- XRF から、並列シスプレックス環境を必要とする VTAM 汎用リソース (VGR) へマイグレーションを行う。
- IMS マルチノード持続セッション (MNPS) の XRF のサポートを使用する。パフォーマンスの制約があるため、このソリューションはお勧めできません。

推奨事項: 3745 コントローラーの永続的な置き換えソリューションとして、CCL または VGR のいずれかを使用してください。

以下の用語は、XRF に固有のものであります。

リカバリー可能サービス・エレメント (RSE) (recoverable service element (RSE))

XRF 複合システム内の 1 つの単位として機能するアクティブ IMS システムおよび代替 IMS システム。

RSE 内の 2 つの IMS システムは、1 つの RSENAME と、XRF の端末セッションを管理するために選択した方法に応じて、MNPS ACB 名または USERVAR テーブルを共有します。RSE 内の どちらかの IMS システム (操作制限に従って) アクティブ IMS システムとして始動し、もう一方を代替 IMS システムとして始動します。どちらも、z/OS の可用性マネージャー・コンポーネントに対して自らを識別します。

従属サービス・エレメント (DSE) (dependent service element (DSE))

代替 IMS システム内に対応部分をもっているが、それ自身のものについてはテークオーバーをトリガーできないアクティブ IMS システムのエレメント。

DSE は、IMS に依存して、障害の発生を認識し、テークオーバーを要求します。CPC、z/OS、VTAM、および IRLM は DSE です。

テークオーバー (takeover)

アクティブ IMS システムから代替 IMS システムへの作業負荷のシフト。テークオーバーには、次の 2 種類があります。

計画テークオーバー (planned takeover)

アクティブ IMS システムの保守を許可するための、代替 IMS システムへの IMS 作業負荷の意図的なシフト。

プラン外テークオーバー

アクティブ IMS システムや、そのアクティブ IMS システムと関連付けられた CPC、z/OS オペレーティング・システム、または VTAM の障害による、代替 IMS への IMS 作業負荷のシフト。

DASD サービス・エレメント

XRF 複合システムと関連付けられた DASD。

DASD サービス・エレメントは、リカバリー可能でも、従属でもありません。アクティブ IMS システムと代替 IMS システムの両方で共用されます。DASD サービス・エレメントで重大な障害が発生すると、エンド・ユーザーへのサービスが停止します。XRF 複合システムが USERVAR を使用する場合、これは、37x5 通信コントローラーについてもあてはまります。

XRF 複合システム

インストール・システムで XRF を使用するには、XRF 複合システムと呼ばれる適切な構成を構築しなければなりません。XRF 複合システムは、ハードウェア (CPC や DASD など) と、ライセンス・プログラムから構成されています。

XRF で必要とされるハードウェアおよびライセンス・プログラムを以下にリストします。

- 独立動作が可能な 2 つの CPC (例えば、2 つの IBM eServer™ zSeries 990 CPC)
- 各 CPC で実行する z/OS(DFSMS を含む)、IMS、および VTAM
- アクティブと代替の両方の IMS システムで共用されている IMS システム・ログ (オンライン・データ・セット (OLDS) と先行書き込みデータ・セット (WADS))
- アクティブ IMS システムと代替 IMS システムからのアクセス・パスを持つデータベース
- USERVAR を使用する XRF 複合システムの場合、境界ノードにある SNA 端末を制御する個々の IBM 37x5 通信コントローラーで実行されるネットワーク制御プログラム (NCP)
- いずれかの IMS システムと通信できるリモート端末

アクティブ IMS は、XRF が IRLM を必要としない点を除き、XRF がない場合と同様に、優先順位の高い IMS 作業負荷を処理します。代替 IMS は、IMS トランザクションを処理しませんが、アクティブ IMS をモニターして、障害の兆候を検出します。代替 IMS のタスクは、アクティブ IMS の進行状況を追跡し、作業をアクティブ IMS から代替 IMS に迅速にシフトできるように、自身の制御ブロックを更新することです。

XRF テークオーバーで、XRF は、端末に割り当てられたクラスに応じて、それぞれ別個に取り扱います。XRF システムでは、以下の基準に沿って、クラス 1、クラス 2、またはクラス 3 として端末を分類します。

- 端末の種類

- 端末を制御するテレプロセシング・アクセス方式
- 代替 IMS システムへの端末の接続方法
- システム・プログラマーによる端末の定義方法

クラス 1 端末は、接続がなくなる、すなわちデータ処理が XRF テークオーバーで中断されることはないと考えられます。しかし、端末セッション管理に使用するのが MNPS なのか、それとも USERVAR なのかによって、XRF がクラス 1 端末を管理する方法は異なります。

USERVAR 方式の XRF 端末セッション管理では、代替 IMS は、クラス 1 端末にオープン・バックアップ・セッションを絶えず持続し、テークオーバーの場合、端末をそのセッションに切り替えます。MNPS 方式では、代替 IMS は、バックアップ・セッションを維持しません。その代わりに、テークオーバー時に、代替 IMS は、MNPS ACB の新規インスタンスを開きます。すると、クラス 1 端末のセッションは、代替 IMS 上の新しい MNPS ACB を通じて転送されます。各方式の違いは、エンド・ユーザーには影響ありません。

その他のすべての端末では、テークオーバーが発生すると、セッションが中断されます。端末によっては、テークオーバーの発生時に IMS がサービスを即時に再確立しようとするものもあります。このような端末は、クラス 2 端末です。その他の端末、例えば、テークオーバー発生時にオペレーターが手動で代替 IMS システムに切り替えなければならない端末では、テークオーバー時に XRF サポートを使用することができません。このような端末は、クラス 3 端末です。すべての ETO および LU 6.2 装置はクラス 3 です。このような端末のユーザーは、オペレーターに IMS システムとのセッションの再確立を依頼します。

関連資料: 端末クラスについて詳しくは、800 ページの『XRF 複合システムの端末』を参照してください。

適用しなければならないすべてのプログラム一時修正 (PTF) が特定されます。両方のシステムが、同一の IMS リリースになっていなければなりません。予定テークオーバーは、ある IMS リリースから別の IMS リリースへのマイグレーションには使用できません。

代替 IMS は XRF 関連のタスクを実行しなければなりません、代替 IMS 内の CPC、z/OS、および VTAM は、他の作業のプロセスにも使用できます。テークオーバー時には非 XRF 作業に対してサービスが低下することを念頭に置いて、作業負荷を慎重に計画してください。例えば、代替 IMS で TSO を実行すると、テークオーバー時に z/OS が TSO ユーザー・アドレス・スペースをスワップアウトします。

複合システムのどちらの IMS システムのオペレーターも、XRF 複合システムにもう一方の IMS を含めるために、自身の関与範囲および責任範囲を拡大しなければなりません。どちらかの IMS が状況を変えた場合は必ず、両方のオペレーターがそれについて知っていなければなりません。

どちらのオペレーターも、以下に示す、XRF 複合システム内の IMS システムの振る舞い特性を把握している必要があります。

- アクティブ IMS システムは、IMS 作業負荷を処理します。

- 代替 IMS システムは、アクティブ IMS の作業負荷をテークオーバーする準備ができていない状態を維持します。
- アクティブ CPC と代替 CPC はどちらも、IMS 以外の作業を実行させることができます。

XRF のバリエーション: MNPS と USERVAR

XRF 複合システムは、VTAM マルチノード持続セッション (MNPS) および MNPS ACB の使用と VTAM USERVAR テーブルの使用という 2 つの方法のいずれかにより、テークオーバー中に端末セッションを管理します。XRF 複合システムを設計するときに、セッション切り替えの管理に MNPS を使用するか、USERVAR を使用するかを選択する必要があります。

端末ユーザーの場合、MNPS を使用する XRF 複合システムと USERVAR を使用する XRF 複合システムとに明白な違いはありません。しかし、各タイプの XRF 複合システムはテークオーバー中にそれぞれ異なる方法で端末セッションの切り替えを管理します。USERVAR を使用する XRF 複合システムにはハードウェアとソフトウェアが必要ですが、MNPS を使用する XRF 複合システムでは必要ありません。端末セッションの管理方法を除き、これら 2 つのタイプの XRF 複合システムは、機能的には同じです。

USERVAR を使用する XRF 複合システムのハードウェアおよびソフトウェア要件には、37x5 通信コントローラーによる NCP の実行が含まれます。MNPS を使用する XRF 複合システムでは、追加のハードウェアまたはソフトウェアは必要ありません。

USERVAR を使用する XRF 複合システムでは、セッション切り替えが機能するために、管理者がログオン・メッセージ、USERVAR 名、および APPLID 名を定義する必要があります。MNPS を使用する XRF 複合システムでは、XRF 複合システム内の各 IMS システム上で MNPS ACB の定義が必要です。端末ユーザーは、MNPS 名を使用してログオンします。

MNPS を使用する XRF 複合システムでは、バックアップ・セッションは必要ありません。USERVAR を使用する XRF 複合システムでは、代替 IMS システムは、クラス 1 として定義された端末についてアクティブ IMS システムでオープンされているすべてのセッションについてバックアップ・セッションを保持します。

MNPS を使用する XRF 複合システムでは、代替 IMS システムはバックアップ・セッションを作成しません。その代わりに、テークオーバーが発生した場合、VTAM が、アクティブ IMS でオープン・セッションをもつ、クラス 1 として定義されたすべての端末について持続セッションを保持します。代替 IMS システムが新規 MNPS ACB をオープンした後で、VTAM は、その新規 MNPS ACB を通じてクラス 1 セッションを経路指定します。

各タイプの XRF 複合システムの指定について詳しくは、814 ページの『XRF のためのシステムの準備』を参照してください。

USERVAR を使用する XRF 複合システムでのバックアップ・セッション

USERVAR を使用する XRF 複合システムでは、代替 IMS システムは、アクティブ IMS システムでオープンなすべてのクラス 1 端末セッションのバックアップ・セッションをオープンするため、テークオーバー中の端末の切り替えに必要な時間を削減できます。

クラス 1 端末ユーザーが XRF IMS にログオンすると、アクティブ IMS はこの新しいセッションをログに記録します。代替 IMS は、このレコードを読み取って、その端末に対してバックアップ・セッションを初期設定します。バックアップ・セッションが成立している限り、データはバックアップ・セッション上を流れません。ユーザーがアクティブ IMS からログオフすると、代替 IMS はバックアップ・セッションを終了させます。

テークオーバー時に、代替 IMS は、アクティブ IMS と通信している各クラス 1 端末を、1 次セッションから事前オープンされているバックアップ・セッションへ切り替えることを、VTAM を通して NCP へ要求します。NCP は、これらの要求に応答します。テークオーバーの透過性は、テークオーバー時に端末上で行われている処理によって異なります。ユーザーはテークオーバーを認識しないか、あるいは、最後のトランザクションを再入力するように依頼されます。

DBCTL 機能

DBCTL 環境だけでは、すべての XRF 機能は提供されません。しかし、DBCTL 環境によって、待機代替 DBCTL 環境を持つことができます。

この代替 IMS は、アクティブ IMS を追跡しません。管理者にできることは、この代替システムを完全に初期設定され、引き継ぎの準備ができた状態にすることだけです。この時点で、代替 IMS は、事前初期設定済み DBCTL 環境と呼ばれます。アクティブ IMS で障害が発生した場合、再始動コマンドを代替 IMS に送信できます。

DBCTL サービスは IMS DB に組み込まれているため、DB/DC 環境では、CCTL に DBCTL サービスを提供できます。その DB/DC 環境で XRF を実行すると、提供された DBCTL サービスはすべての XRF 機能を持つサービスになります。オペレーターまたは CCTL は、切り替えコマンドを実行して、テークオーバーを開始することができます。

XRF テークオーバー (XRF takeover)

テークオーバーを説明する最も良い方法は、例を挙げて説明する方法です。高可用性を必要とする典型的なインストール・システム、つまり、多くのリモート端末からの製品データを処理する製造コントロール・センターについて考えてみます。プロセスには、IMS データベースを更新すること、および、アクティビティーを IMS システム・ログに記録することも含まれます。それに付随するタスクは、給与計算の実行、企業報告書の作成、定期保守など、優先順位が低いことが多い作業の実行です。このインストール・システムは、テレプロセッシング・アクセス方式として VTAM を使用する z/OS を実行します。

この例の障害は、IMS 制御領域の異常終了です。

XRF がない場合の IMS 障害

IMS システムが 1 つで、XRF を持たないインストール・システムでは、IMS 制御領域が異常終了すると、端末でのすべてのサービスが中断されます。システムが使用できない時間が長過ぎると、製造ラインはシャットダウンします。オペレーター、システム・プログラマー、またはこの両者は、以下を行わなければなりません。

1. 診断ダンプを待つ。
2. 異常終了の原因を判別する。
3. IMS を再始動する。
4. 関連付けられている従属領域を再始動する。
5. 端末とのセッションを再確立する。

停止の種類とインストール・システムのサイズによって、これらの作業には、1 時間から数時間かかる場合があります。停止以前の状態に IMS をリカバリーするには、さらに 30 分が必要です。当初の問題は微細なものであっても、リカバリーには大きな犠牲が強いられます。運用スタッフがリカバリー・ステップを実行している間、エンド・ユーザーは、約 2 時間もサービスを利用できない場合があります。処理能力を取り戻し、ダウン期間中のデータベースとメッセージ・キューの保全性を保証することが、このインストール・システムには必要です。

XRF がある場合の IMS 障害

XRF をインストールしているインストール・システムは、上述のインストール・システムとは物理的に異なります。以下の図は、XRF 複合システムの 2 つのシステムを示しています。個々のシステムには、CPC と 3 つのライセンス・プログラム z/OS (DFP を含む)、IMS、および VTAM が含まれています。

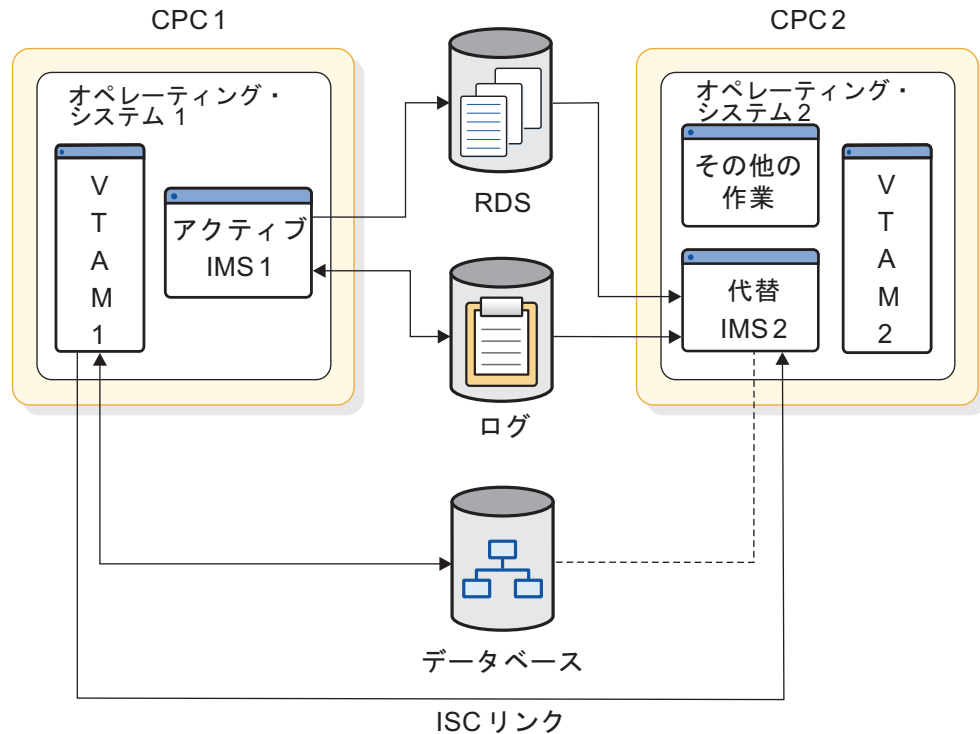


図 66. テークオーバー前の XRF 複合システム

IMS 1 は、リモート端末からの製造データ（優先順位の高い作業）を処理します。データベースを更新し、そのアクティビティを IMS システム・ログに記録します。

IMS 2 は、IMS システム・ログのレコードをモニターすることにより、アクティブ・サブシステムを追跡します。アクティブ IMS の環境と同じ環境を保持するために、IMS 2 は代替 IMS にある多数の制御ブロックとメッセージ・キューを更新し、それらをアクティブ IMS に反映します。また、XRF 複合システムが USERVAR を使用する場合、IMS 2 は、アクティブ IMS にログオンするクラス 1 端末ユーザーに対して、バックアップ・セッションもオープンします。このアクティビティによって使用されない CPC 容量およびストレージは、企業の報告書プログラムをサポートします。

IMS 1 が異常終了すると、テークオーバーが開始されます。XRF が要求を処理する実記憶域の量次第で、オペレーティング・システム 2 は企業の報告書プログラムをスワップアウトします。IMS 2 は実動作業負荷を代替 IMS へシフトし、XRF 複合システムが MNPS を使用する場合は、MNPS ACB の新しいインスタンスをオープンします。この時点で、IMS 2 はクラス 1 およびクラス 2 端末へのサービスを開始するので、障害の発生した IMS 1 での問題を判別できます。

IMS 2 は、データをリカバリーして、クラス 1 端末でのセッションを切り替える一方で、オペレーティング・システム 1 と協同でして、IMS 1 が IMS システム・ログとデータベースに書き込まないようにします。IMS 2 はログを分離して、テークオーバーを続行します。同時に、オペレーティング・システム 1 は、入出力防止を実行します。すなわち、IMS 1 からデータベースへのすべての新しい入出力要求は確実に実行せずに戻されます。z/OS は、データベース・データ・セットへの既存の入出力要求をすべて完了または取り消すと、その旨をオペレーターに通知しま

す。以下の図は、テークオーバー中の 2 つのシステムを示しています。

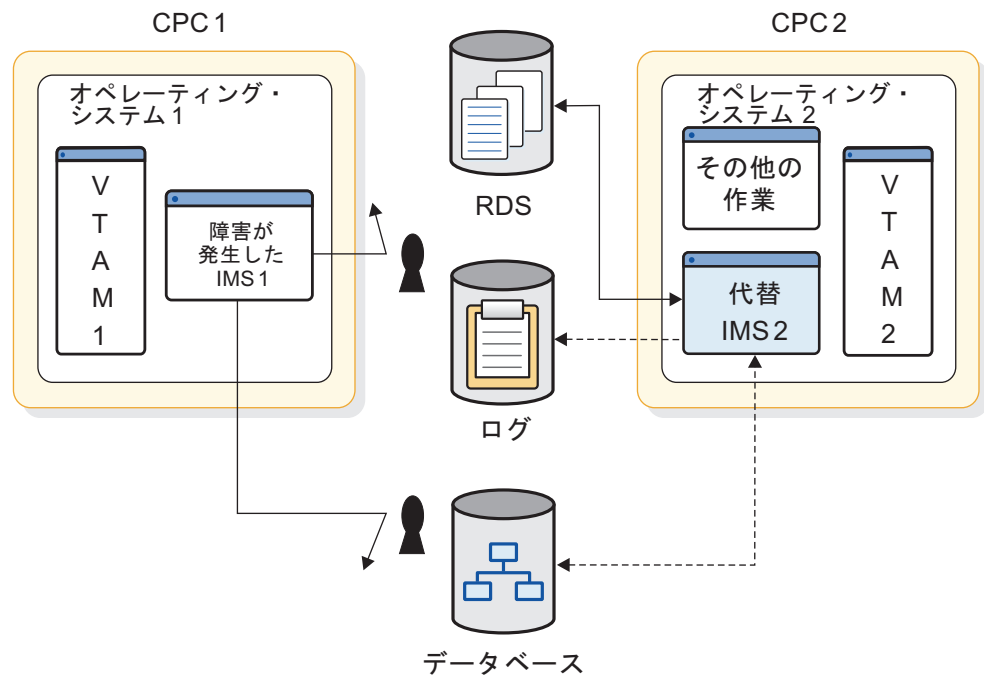


図 67. テークオーバー中の XRF 複合システム

クラス 1 端末のすべてのユーザーが IMS 2 と通信し、トランザクションを入力し、IMS アプリケーションから応答を受け取ることができると、テークオーバーは完了します。障害が発生している IMS 1 がデータベースに書き込めないとわかると、IMS 2 はデータベースの保護を停止します。以下の図は、テークオーバー後の XRF 複合システムを示しています。

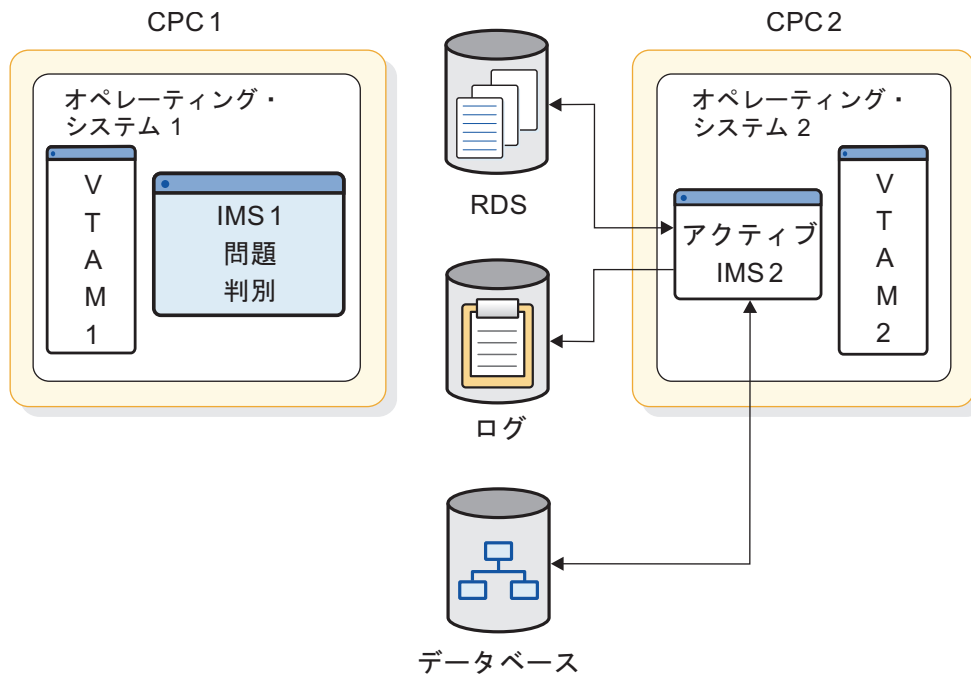


図 68. テークオーバー後の XRF 複合システム

XRF は、インストール・システムに関する問題判別を行いません。しかし、ダンプおよび再始動中に端末ユーザーがサービスを受けていない間に非 XRF のインストール・システムに存在する圧迫を減少させます。それでもまだ、インストール・システムの要員は、障害が発生したアクティブ・サブシステムの問題の原因を判別しなければなりません。障害が発生したアクティブ・サブシステムでダンプ・アクティビティが完了すると、オペレーターは、オペレーティング・システム 2、IMS 2、VTAM 2、および CPC 2 用の代替 IMS として、オペレーティング・システム 1、IMS 1、VTAM 1、および CPC 1 を稼働状態に戻すことができます。ダンプ・アクティビティが長い場合は、オペレーターは別のシステムを代替システムとして立ち上げます。IMS ユーザーの要求を処理しているシステムの代替 IMS が存在してはじめて、XRF 複合システムが存在します。

テークオーバー中に、IMS および z/OS はシステム・オペレーターにメッセージを送信して、テークオーバーの進行を通知します。テークオーバーの原因とインストール・システムのリソースによって、オペレーターが行なわなければならない作業がいくつかあります。

関連資料:

- z/OS、IMS、VTAM、および NCP が XRF テークオーバー処理でどのような機能を提供しているかについては、758 ページの『XRF プロセスでのコンポーネントの役割』を参照してください。
- テークオーバー時に XRF がオペレーターに送るメッセージと、オペレーターの適切な応答については、786 ページの『XRF プロセスのテークオーバーの後処理フェーズ』を参照してください。

テークオーバー条件

テークオーバー条件とは、代替 IMS の IMS にテークオーバーを要求させるイベントです。XRF を使用するインストール・システムでは、IMS 制御領域の異常終了が常にテークオーバー条件となります。IMS の調整時にシステム・プログラマーが指定するパラメーターによって、以下の障害がテークオーバー条件になるかどうかが決まります。

- z/OS の障害、ループ、または待ち状態
- CPC 障害
- VTAM 障害
- IRLM 障害

推奨事項: IMS が SVC ダンプを処理している間に望ましくないテークオーバーが行われないようにするために、以下の条件を必ず満たすようにしてください。

- SVC ダンプが完了できるほど XRF テークオーバー・インターバルが十分に長い。
- システム・データ・セットには、SVC ダンプのデータを取り込めるだけの十分なスペースがある。

IMS インストール・システムでの障害がすべて、テークオーバーが必要な状態として適格なわけではありません。XRF は、重複させていないサービス・エレメントの障害で発生した停止には対処しません。例えば、XRF は次の障害には対応しません。

- CPC と通信コントローラーまたは DASD との間の通信を中断させるチャンネル障害またはリンク障害
- 通信コントローラー、NCP、回線、および端末などの通信ネットワークの障害
- システム間障害、例えば JES3 または CTC に起因する障害
- IMS データベースの損失または損傷
- 複合システム内の両方の CPC に影響を及ぼす電源障害
- データベースなどのデータ・セットを示すユーザー・カタログの障害

各インストール先で、テークオーバーに関する計画決定を下さなければなりません。テークオーバーを開始する条件、オペレーターの介入の程度、どの端末セッションがテークオーバー時に自動的にリカバリーするか、アクティブ IMS の問題を知るために代替 IMS がどの方式を使用するかなどを決定しなければなりません。

プランされたテークオーバー

インストール・システムは、2 つのシステムに対するある種の変更をスケジュールするのにテークオーバーを利用できます。システムが代替システムのとときに、オペレーターはそのシステムをダウンさせ、システム・プログラマーが更新を行う間(短時間)、アクティブ IMS をバックアップ・サポートがない状態にしておくことができます。次にオペレーターは、所定の位置のプログラム・コードまたはハードウェアを保守して代替 IMS を再始動したり、所定の位置の構成変更をして z/OS を IPL します。アクティブ IMS に対して変更を加えるために、オペレーターはテークオーバーを開始して、前のアクティブ IMS でプロセスを繰り返します。

XRF 要件

XRF 複合システムの作成には、プロセス、ストレージ、通信などのリソースがさらに必要です。

必要なりソースの数は、以下によって決まります。

- インストール・システムがすでに持っているリソースはどのようなリソースか
- CPC の使用程度

以下のリストでは、XRF に対するソフトウェア要件、ハードウェア要件、および操作要件について説明します。必須レベルの要素については、「IMS V15 リリース計画」を参照してください。

- XRF ライセンス・プログラムの要件

XRF を使用するには、IMS はアクティブ IMS システムと代替 IMS システムに同じシステム定義を指定しなければなりません。XRF を操作可能にするには、必須レベルの z/OS、DFSMS、SSP および VTAM をアクティブと代替の両方の IMS システムにインストールしておく必要があります (ただし、これらのライセンス・プログラムを同時にインストールする必要はありません)。XRF 複合システムが USERVAR を使用する場合は、NCP もインストールしておく必要があります。

- XRF のハードウェアの要件

XRF は、以下の IBM ハードウェア製品で機能します。

- CPC

XRF は、z/OS によりサポートされる CPC で稼働します。CPC は異なるモデルでもかまいません。

- 37x5 通信コントローラー

XRF 複合システムが USERVAR を使用する場合、XRF では、テークオーバー時にセッションを自動的に切り替えるために、クラス 1 端末、クラス 2 端末、および中間ルーティング・ノード (IRN) を 37x5 通信コントローラーに接続することが必要です。

- 端末

ご使用のシステム内の IMS を使用しているすべての端末が、XRF 複合システムで操作できます。743 ページの『XRF の概要』に説明されているように、サポートのレベルはさまざまです。

X.25 通信プロトコルを使用している端末は、IMS で使用できます。X.25 端末は、IMS における端末の定義、および X.25 特性に基づいて、クラス 1、クラス 2、またはクラス 3 の XRF サポートを持つことができます。X.25 NCP パケット交換インターフェース (NPSI) は、IMS とのセッションが確立されている適格 SNA 端末について XRF 端末の切り替えに参加します。適格 SNA 端末は、以下を使用して X.25 ネットワークに接続されます。

- 統合 X.25 アダプター
- ネットワーク・インターフェース・アダプター (5973-L02)

- 外部 SDLC/HDLC PAD

適格な SNA 端末だけが、クラス 1 XRF サポートを受けることができます。

X.25 端末の接続について詳しくは、以下を参照してください。

- X.25 NCP Packet Switching Interface General Information Version 3
- NCP X.25 Planning and Installation

XRF 端末の切り替えは、SNA 端末への切り替えに限定されています。XRF は他の NPSI 機能 (以下の機能も含む) により SNA ホストと通信する非 SNA 装置をサポートしません。

- 非 SNA 装置に対するプロトコル変換 (PCNE)
- パケット組み立て/分解機能 (PAD)
- X.25 トランスポート拡張機能への一般アクセス (GATE)
- X.25 トランスポート拡張機能への占有アクセス (DATE)

- XRF の操作および管理要件

XRF のサポートのために要員の追加は必要ありませんが、システム・プログラマーやオペレーターに要求される作業が増えます。z/OS、IMS、およびネットワーク・オペレーターは、XRF プロセスを理解し、テークオーバー発生の可能性に注意を怠らず、テークオーバー時に相互に通信できる準備ができていなければなりません。アクティブ IMS の管理に責任があるオペレーターが、代替 IMS のオペレーターと容易に通信できるようにシステム・コンソールを編成する必要があります。

システム・プログラマーは、XRF 関連のパラメーターおよびマクロを使用して、z/OS、IMS、VTAM、および NCP を初期設定しなければなりません。ユーザー・アプリケーションは、IMS 標準に準拠していれば、XRF 複合システムで稼働し、XRF によって可用性が改善されます。システム・プログラマーは、XRF 複合システムの標準診断エイドを使用できます。

インストール・システムに XRF をインストールする前に、テークオーバー中のデータ・セットの中断を最小限にとどめるための準備をしなければなりません。以下を確認します。

- 必要なすべてのデータがアクティブ・システムと代替システムの両方に重複している (つまり、z/OS データ・セット、不揮発性アプリケーション、およびサブシステム・データが重複している)。
- アプリケーション・データベースへのアクセスとログが、アクティブ・システムおよび代替システムで共用されている DASD によって行われる。

非常駐 ACB を 64 ビット・ストレージに保持すると決めた場合は、待機システムの 64 ビット・ストレージ・プールのサイズが、アクティブ・システムの 64 ビット・ストレージ・プールのサイズと同じになるようにして、XRF 待機システムを同じ方法でセットアップする必要があります。

XRF を使用すると、インストール・システムのリソースにさらに必要になります。しかし、XRF がどのように機能するかを理解し、慎重に計画を立てれば、こ

のオーバーヘッドを軽減することができます。インストール・システムが IMS サービスの長期に渡る計画割り込みおよび計画外割り込みによって重大な影響を受ける場合は、このような努力をしてみる価値があります。

- XRF に対する IMSplex の要件

共通サービス層 (CSL) を備えた IMSplex では、CSL にアクセスできる代替 IMS を定義する必要があります。XRF 代替システムは、固有の IMSID を使用して IMSplex に定義する必要があります。代替 IMS のオペレーティング・システムには、構造化呼び出しインターフェース (SCI) を定義する必要があります。Operations Manager (OM) を使用して代替およびアクティブの IMS システムにコマンドを経路指定するには、ROUTE() パラメーターに IMSID を指定してください。テークオーバーの後、その新規のアクティブ IMS は、Resource Manager (RM) に定義されているリソースの所有権を引き継ぎます。

RM のない IMSplex でオンライン変更機能を使用する場合、アクティブ IMS システムが、OLCSTAT データ・セットを排他的に所有している必要があります。OLCSTAT データ・セットには、アクティブ IMS システムのみの IMSID を含めることができます。

テークオーバーの後で代替 IMS システムがアクティブ IMS システムになった場合、代替 IMS システムの IMSID に一致するように OLCSTAT データ・セット内の IMSID を変更しないと、INITIATE OLC コマンドを発行できません。オンライン変更ユーティリティ (DFSUOLC0) を使用して OLCSTAT データ・セット内の IMSID を変更できます。

関連概念:

35 ページの『アプリケーション・プログラムのシステム・サポート』

XRF プロセスでのコンポーネントの役割

XRF プロセスでは、IMS、z/OS、およびいくつかの z/OS エレメントが共同して動作します。

それぞれが実行する機能に精通することにより、発生する問題の診断だけでなく、XRF の計画、インストール、および操作が容易になります。

XRF プロセスに与える IMS の利点

IMS ユーザーが XRF サービスの最大の受益者であるように、IMS は XRF プロセスで重要な役割を果たします。アクティブ IMS での IMS プロセスのほとんどは、IMS 端末ユーザーの要求に応えるもので、代替システムでの IMS プロセスは、すべて XRF と関係しています。代替 IMS システムは、ユーザーからのログオン要求の受け入れを拒否します。

以下のリストは、テークオーバー前とテークオーバー中の 2 つの IMS システムのアクティビティを要約しています。

- アクティブ IMS および代替 IMS は、IMS システム・ログを使用して、あるいは、代替 IMS がアクティブ IMS から受け取る簡単な信号によって、継続的な通信を確立し、維持します。監視メカニズムが信号を提供します。
- アクティブ IMS は次のことを行います。

- IMS ユーザーの要求を処理します。
- ISC リンクと再始動データ・セット (RDS) を使用して監視信号を送信します。
- 端末に対してセッションをオープンおよびクローズします。
- データベースをオープンおよびクローズします。
- アクティビティーをログに記録します。
- 代替 IMS は次のことを行います。
 - 通常の始動プロシージャーに応答しますが、テークオーバー後までトランザクションを処理しません。
 - 通常操作中
 - ログからの情報を使用して、制御ブロックを更新します。
 - アクティブ IMS がオープンするのと同じデータベースを割り振ってオープンします。
 - XRF 複合システムが USERVAR を使用する場合、XRF IMS にログオンする、あるいは XRF IMS からログオフするクラス 1 端末用に、バックアップ・セッションをオープン、あるいはクローズします。
 - アクティブ IMS で有用な作業が中止されたことがモニターによってわかった場合は、監視メカニズムを使用してアクティブ IMS を監視し、テークオーバーを開始します。
 - テークオーバー中
 - 他のライセンス・プログラムと一緒に、アクティブ IMS の作業負荷のテークオーバーを行います。IMS システム・ログを管理することにより、作業負荷を取得します。
 - 障害が発生している アクティブ IMS のログへの書き込みを防止します。
 - 経過中トランザクション、つまり、アクティブ IMS が部分的にしか処理しなかったトランザクションをリカバリーします。
 - XRF 複合システムが MNPS を使用する場合、MNPS ACB の重複インスタンスをオープンします。
 - 新しいトランザクションの実行を開始します。
 - セッション・リカバリーをクラス 1 端末およびクラス 2 端末で開始します。本書では、「セッション・リカバリー」とは、IMS がクラス 1 端末上でセッションの切り替えを試みるか、クラス 2 端末上でサービスの再確立を試みることです。

リカバリー可能サービス・エレメントとしての IMS

XRF 複合システムの 2 つの IMS は、複合システム内に特殊な状況を持っています。IMS、z/OS、VTAM、および CPC での障害はテークオーバーを発生させる場合があります、したがって、リカバリー可能ですが、XRF は、IMS のみがリカバリー可能サービス・エレメント (RSE) であると見なします。z/OS、VTAM、および CPC は従属サービス・エレメント (DSE) です。これらは、IMS に依存して障害を認識し、テークオーバーを要求します。RSE 名は、テークオーバー時にオペレーターが受け取るメッセージに示されます。例えば、IMS1 と IMS2 という名前の 2 つの IMS が、IMSPROD という名前の RSE を形成しています。この複合システムでテ

ークオーバーが発生した後、SUBSYSTEM IMS2 IS NOW THE ACTIVE ELEMENT OF RSE IMSPROD (メッセージ AVM007I) という z/OS メッセージがオペレーターに送られます。

関連資料: RSE 名について詳しくは、818 ページの『XRF のための IMS.PROCLIB メンバーの定義』の RSENAME を参照してください。

監視メカニズム

通常操作中、代替 IMS は IMS システム・ログを読み取ることにより、アクティブ IMS をトラッキングします。アクティブ IMS は、そのアクティビティをログに書き込み、代替システムは、これらのレコードを検査して固有の制御ブロックを更新します。このように、代替システムの制御ブロックは、アクティブ IMS の制御ブロックにそれらのレコードを反映し、代替 IMS は、テークオーバーの準備ができている状態を維持します。

アクティブ IMS は、システム・ログに以下の障害を記録します。

- IMS 制御領域の異常終了
- IMS で TPEND 出口につながる VTAM 障害
- IMS で STATUS 出口につながる IRLM 障害

エラー条件によっては、IMS がこれらの障害をシステム・ログに記録するのを防ぐことがあります。このような条件の例としては、IMS が OLDS バッファを除去できない、などが挙げられます。

このログは、他の障害について警告しません。このような理由から、監視メカニズムがアクティブ IMS から定期信号を提供します。代替 IMS は、これらの信号をモニターします。信号が発信されないと、代替 IMS はテークオーバーと見なします。これらの信号がないということは、次のことを示しています。

- z/OS 障害
- z/OS のループまたは待ち状態
- CPC の障害

代替 IMS は、これらの信号を以下の図に示されている 3 つの方法で受信できます。

1. アクティブ IMS は、アクティブ IMS と代替 IMS サブシステム間の ISC リンクを使用してメッセージを送信します。
2. アクティブ IMS はタイム・スタンプを RDS に置きます。
3. アクティブ IMS は、IMS システム・ログに新しいレコードを追加し続けます。

IMS.PROCLIB のメンバー DFSHSBxx のパラメーターを使用して、ご使用のシステムでどの監視方式を作動させるかを選択します。以下の図は、3 つの監視メカニズムを示しています。ISC リンクは、VTAM CTC でも、37x5 通信コントローラーによる共用回線でもかまいません。

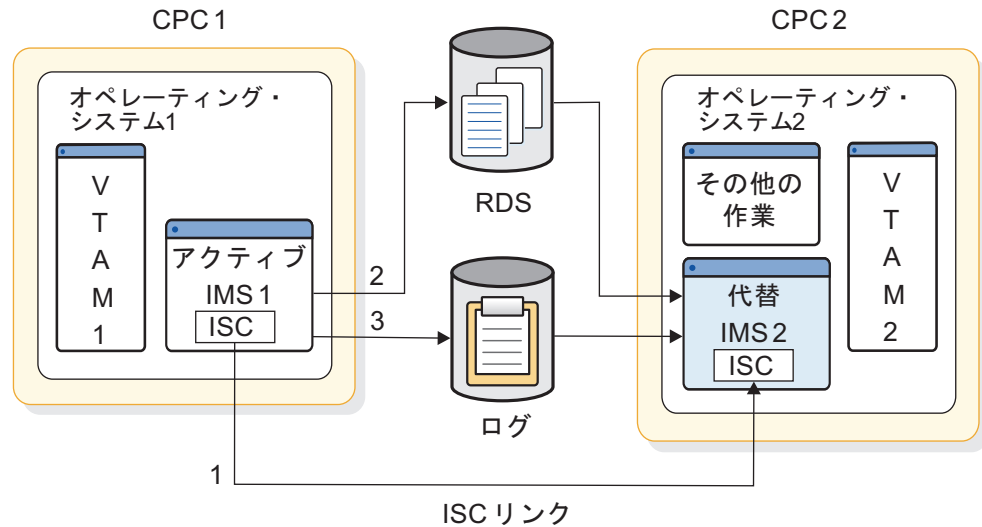


図 69. 監視オプション

ISC リンクおよび RDS 上の信号は、定期信号です。IMS システム・ログを監視メカニズムとして使用するために、代替 IMS はログ上の新しいレコードを定期的にチェックします。

ISC リンクおよび RDS データ・セット上で信号の受信に失敗すると、テークオーバーが発生します。ログ上に新しいレコードがないことはありません。しかし、代替 IMS は、ログ・アクティビティーを認識すると、RDS または ISC リンクからのテークオーバーの指示をオーバーライドします。

IMS.PROCLIB の DFSHSBxx メンバーで定義するパラメーターで、以下を指定します。

- システムで使用したい監視メカニズムはどれか
- アクティブ IMS が ISC リンクまたは RDS 上の信号を送信する頻度
- 代替 IMS がログの新規レコードをチェックする頻度
- 指定した時間内にどの信号が送信されないことを、テークオーバー条件とするか

768 ページの『XRF の監視の確立』では、これらのパラメーターをコーディングする方法について説明しています。オペレーターは、IMS /STOP コマンドと /START コマンドを使用して、監視を動的に停止または開始できます。また、/CHANGE コマンドを使用して、監視のタイミング・インターバルを変更できます。

z/OS および z/OS エLEMENTの利点

z/OS コンポーネントおよび他の製品は、XRF 処理にアクティブに関与します。これらのコンポーネントおよび製品には、可用性マネージャー (AVM)、データ機能記憶管理サブシステム (DFSMS)、仮想記憶通信アクセス方式 (VTAM)、ネットワーク制御プログラム (NCP)、システム・サポート・プログラム (SSP)、および Tivoli NetView for z/OS などがあります。

可用性マネージャー

z/OS は、その可用性マネージャー (AVM) のエレメントによって、アクティブ IMS および代替 IMS に環境を提供し、テークオーバー中にサービスを提供します。特に、AVM は次のことを行います。

- 障害が発生している IMS システムがデータベースにアクセスしないようにする入出力の防止を提供します。
- テークオーバー中に XRF 複合システムの状態を説明する接頭部 AVM が付いているメッセージを送信します。

これらのメッセージには、接頭部 AVM が付いています。これらのメッセージは、XRF 複合システムの状態を説明し、オペレーターがテークオーバーに正しく対応できるようにします。

z/OS のシステム・リソース・マネージャー (SRM) コンポーネントは、代替 IMS によるテークオーバー開始時の新しい作業負荷の受け入れを促進します。このとき、データベースおよびセッションのリカバリーのために代替 IMS が必要とする実記憶域が一時的に増加します。z/OS の SRM コンポーネントは、代替 IMS によるストレージに対するニーズについての分析をスピードアップします。この頻繁なチェックによって、SRM はこのニーズに迅速に応えることができます。

入出力防止 は、障害が発生している IMS によるデータベースへの書き込みを停止するために AVM が行うアクションです。IMS がテークオーバーを要求していることをアクティブ z/OS が認識すると、入出力防止が開始されます。このとき、アクティブ IMS はデータベースへの更新をスケジュールしています。データベースの健全性を保持するために、アクティブ z/OS の AVM は、現行の入出力操作が完了し、z/OS がアクティブ IMS からデータベースへの追加の入出力要求を受け入れないことを保証します。データベース・データ・セットが安全であることを確認すると、AVM は I/O PREVENTION IS COMPLETE (メッセージ AVM006E) というメッセージを発行します。

当然のことながら、作動していないときには、z/OS は入出力を防止できません。z/OS が入出力を防止できない場合は、障害が発生しているアクティブ z/OS のオペレーターが、手操作によって、アクティブ IMS にデータベースへの書き込みを行わせないようにしなければなりません。

関連資料: 操作手順の説明は、「IMS V15 オペレーションおよびオートメーション」を参照してください。

障害が発生しているアクティブ IMS のオペレーターは、IMS がこれ以上データベースを変更しないことを確認すると、代替 IMS のオペレーターに知らせます。そして、代替 IMS のオペレーターは、REPLY "GO" WHEN I/O PREVENTION COMPLETES (メッセージ AVM006E) というメッセージに「GO」と応答します。この処置 (あるいは /UNLOCK SYSTEM コマンド) により、代替 IMS によってデータベースの健全性を保証するための作業が完了します。

システム・プログラマーおよびオペレーターは、障害が発生しているアクティブ IMS の IMS がデータベースにアクセスするのを防止することが重要であることを理解する必要があります。両方の IMS システムがデータベースに同時に書き込むことができると、データベースの健全性が失われます。

関連資料: テークオーバー手順の確立については、「IMS V15 オペレーションおよびオートメーション」を参照してください。

データ機能記憶管理サブシステム (DFSMS)

XRF は、入出力防止、VSAM、およびメディア・マネージャー用に DFSMSを必要とします。しかし、DFSMS はメッセージを作成することも、既存のプロシーチャーを変更することはありません。

仮想記憶通信アクセス方式 (VTAM)

VTAM の利点は、XRF 複合システムが MNPS を使用するか、USERVAR を使用するかによって、いくらか異なりますが、端末ユーザーにとっては、明白な違いはありません。

XRF プロセスに対する VTAM の利点には、次のものがあります。

- VTAM は、ユーザーがシングル・ログオン・ログオン・メッセージを使用して XRF 複合システム内の IMS システムにログオンできるようにします。ユーザーは、どの IMS システムが現在アクティブになっているかを知る必要がありません。

XRF を使用すると、複合システム内の 2 つの IMS システムは、端末ユーザーからは単一の IMS システムのように見えます。しかし、VTAM 側から見ると、これらのサブシステムは固有のアプリケーションです。VTAM により、端末ユーザーは、管理者が選択するログオン・メッセージを使用して、XRF IMS にログオンできます。また、端末ユーザーは、ログオン・メッセージを指定する LOGON APPLID コマンドを使用してログオンできます。

- VTAM は、代替 IMS システムが、テークオーバー中にアクティブ IMS にログオンしたクラス 1 端末についてセッションを維持できるようにします。MNPS を使用する XRF 複合システムでは、VTAM は、代替 IMS システム上で MNPS ACB の新しいインスタンスを通じてクラス 1 端末のセッションを転送します。USERVAR を使用する XRF 複合システムでは、VTAM は、代替 IMS システム上で NCP がクラス 1 端末をそれぞれのバックアップ・セッションに切り替えられるようにします。

XRF 複合システムでの VTAM の動作は、USERVAR が使用されるか MNPS が使用されるかに応じて異なります。

USERVAR を使用する XRF 複合システムでの VTAM の動作

VTAM は、解釈テーブルを検査して、ログオン・メッセージに対応する変数を取得します。VTAM は次に、USERVAR テーブルでこの変数に対応する VTAM アプリケーション名を調べます。このアプリケーション名によって、この時点でときに通信する端末に対して IMS システムを識別します。この IMS システムは、端末のセッション・パートナーです。

この 2 つのテーブルとテーブルの項目について確実に理解するようにしてください。各テーブルには 2 つの列が含まれています。解釈テーブルには、ログオン・メッセージ列と USERVAR 変数列があります。USERVAR テーブルには、USERVAR 変数列と VTAM アプリケーション名列があります。VTAM は、USERVAR 変数を使用して、ログオン・メッセージをセッション・パートナーのアプリケーション名と関連付けます。この変数は、

解釈テーブルと USERVAR テーブルの両方の USERVAR 変数列にあります。以下の図では、解釈テーブルと USERVAR テーブルについて図解しています。解釈テーブルのログオン・メッセージ列にある 2 つの項目によって、ユーザーは、ログオン・メッセージ IMSP または IMSA を使用して XRF IMS にログオンできます。

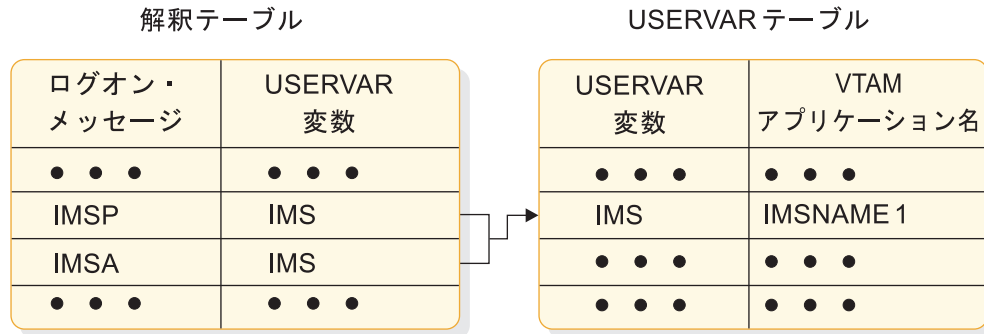


図 70. 解釈テーブルと USERVAR テーブル

解釈テーブルは、VTAM の初期設定時にユーザーが作成する静的テーブルです。LOGCHAR マクロを使用して、このテーブルに項目を入れます。USERVAR テーブルの項目は、初期確立時に確立されますが、動的に変更できます。IMS は MODIFY USERVAR コマンドを実行して、最初に項目を USERVAR テーブルに入れます。オペレーターまたはアプリケーション・プログラムは、この同じコマンドを使用して、ユーザー管理の USERVAR を削除し、VTAM が USERVAR を自動的に管理することを指定したり、あるいは、USERVAR テーブルの項目を変更します。例えば、MODIFY USERVAR コマンドでは、テークオーバー時に、すべての端末の新しいセッション・パートナーを反映するように、USERVAR テーブルのアプリケーション名を変更します。

関連資料: VTAM USERVAR テーブルについて詳しくは、827 ページの『VTAM USERVAR テーブルの定義』を参照してください。

以下の図の上部では、端末を所有している VTAM 用の解釈テーブルと USERVAR テーブルを示しています。この場合、代替 IMS では VTAM です。ユーザーがログオン・メッセージ IMSP でログオンすると、VTAM は解釈テーブルを検索して、IMSP を取得し、対応する USERVAR IMS を見つけます。次に VTAM は、USERVAR テーブルを検索して、IMS に対応するアプリケーション名を取得し、IMS システムの IMS1 を使用して、セッションをオープンします。

以下の図の下部は、テークオーバー後、IMS2 の要求時に、VTAM2 が USERVAR テーブルのアプリケーション名を変更したことを示しています。ここで、端末のユーザーが IMSP を指定すると、VTAM はユーザーを IMS2 に接続します。

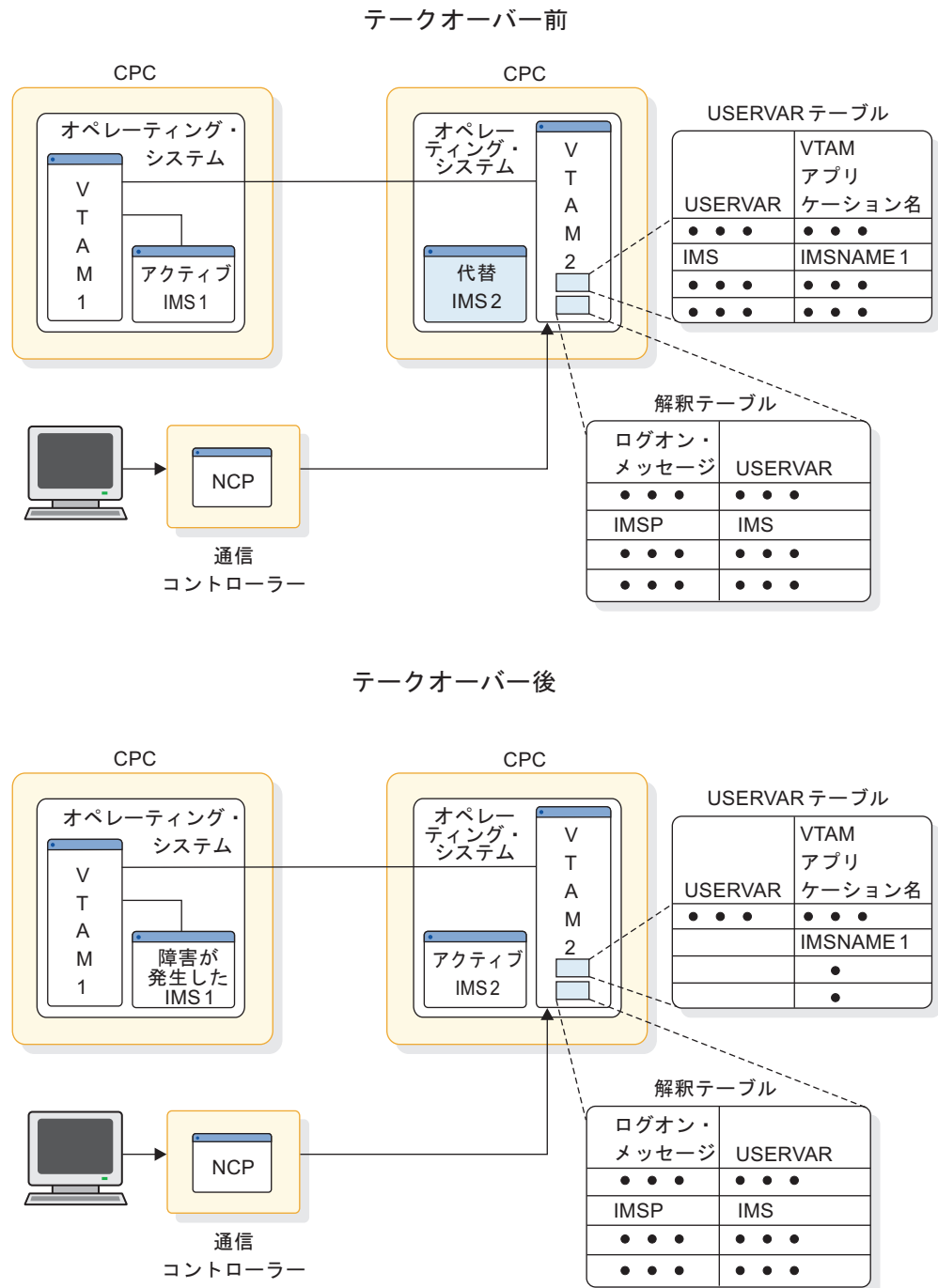


図 71. USERVAR を使用する XRF 複合システムでのログオンの VTAM 処理

現在の IMS 標準に準拠している既存のユーザー・プログラムはすべて、XRF 複合システムで実行し続けます。USERVAR が VTAM 管理の場合は、新しい VTAM APPLID への変更は自動的に行われます。以前に使用したログオンを依然として使用しているプログラムを持っていて、それが VTAM 管理でない場合は、ログオンがアクティブ IMS の VTAM APPLID と一致しないので、変更する必要があります。

ユーザー管理 USERVAR を持つアプリケーション・プログラムを変更するには、次のようにします。

- INQUIRE USERVAR を使用してアクティブ IMS と通信し、現行のアクティブ IMS の実名を判別します。
- BIND データ内の PLU/SLU 名によって影響を受けやすい端末プログラムの場合は、IMS が BIND データに USERVAR を付加します。
- 新しい VTAM SENSE コードをテストしなければなりません。

IMS は、INVALID LOGON REQUEST IN THE BACKUP SYSTEM (メッセージ 3862I) を発行して、ユーザーが代替 IMS にログオンするのを防止します。代替 IMS と通信できるのは、IMS マスター端末および 2 次端末、システム・コンソール、ISC 監視リンクだけです。このメッセージの BACKUP という用語は、XRF 複合システムの代替 IMS のことを意味します。IMS メッセージでは、BACKUP という用語は、クラス 1 端末上でのバックアップ・セッションを指す場合もあります。

MNPS を使用する XRF 複合システムでの **VTAM** の動作

MNPS を使用する XRF 複合システムでは、端末ユーザーは MNPS ACB 名を使用してログオンします。この名前は、アクティブ IMS と代替 IMS の両方の MNPS ACB について同一ですが、存在できる MNPS ACB は一度に 1 つだけです。テークオーバー以前には、アクティブ IMS の MNPS ACB だけが存在します。テークオーバーの後では、代替 IMS のものだけが存在します。どの存在する MNPS ACB に関係なく、端末ユーザーが入力するログオン情報は同じです。

VTAM は、MNPS ACB により、すべての端末セッションを経路指定します。テークオーバーの場合、VTAM は、アクティブ IMS がシャットダウンされ、代替 IMS の MNPS ACB がオープンされるまで、クラス 1 端末のすべてのセッションを持続セッションとして維持します。代替 IMS 上の MNPS ACB がオープンされると、VTAM はこの新しい ACB を使用してすべての端末セッションを経路指定します。

以下の図は、MNPS を使用する XRF 複合システムでの端末セッションの VTAM 所有権と、VTAM がテークオーバーより前にセッションを管理する方法の例を示しています。この場合、VTAM 1 は CPC 1 上にアクティブ IMS を所有しています。CPC 2 にログオンすると、VTAM 2 は、端末のセッションを VTAM 1 に経路指定します。こうして、端末のセッションが MNPS ACB とアクティブ IMS に接続されます。

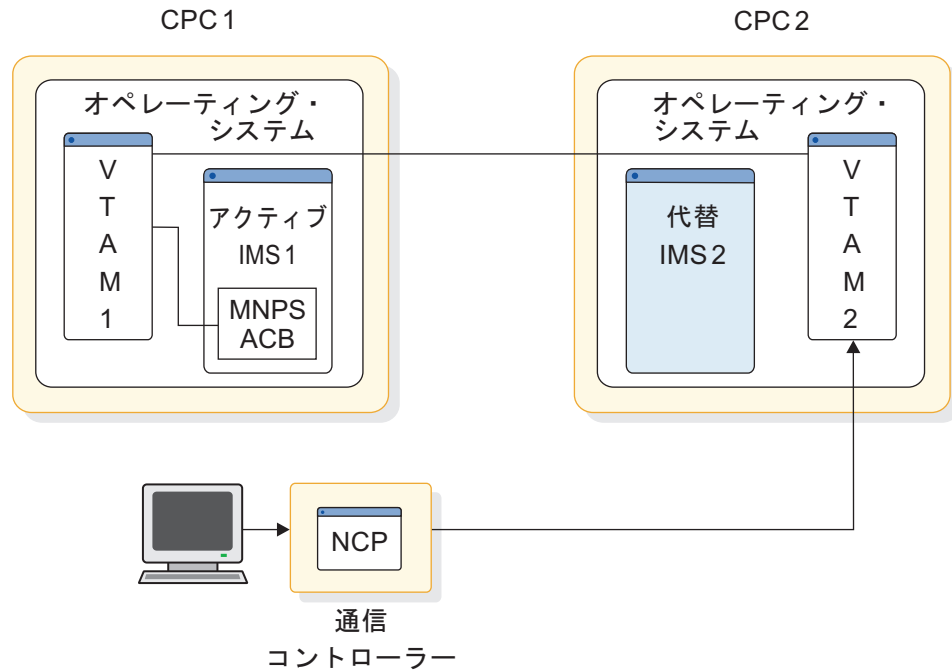


図 72. MNPS を使用する XRF 複合システム・データのログオンの VTAM 処理 (第 1 部)

以下の図は、VTAM がテークオーバー後に端末セッションをどのように転送するかを示しています。ここでは、代替 IMS によって、CPC 2 上で MNPS ACB の新しいインスタンスがオープンされています。VTAM 2 は、今度は、VTAM 1 が実行し続けた場合でも、セッションを VTAM 1 に渡さずに、端末セッションを MNPS ACB に直接接続します。端末ユーザーは、この切り替えに気が付きません。

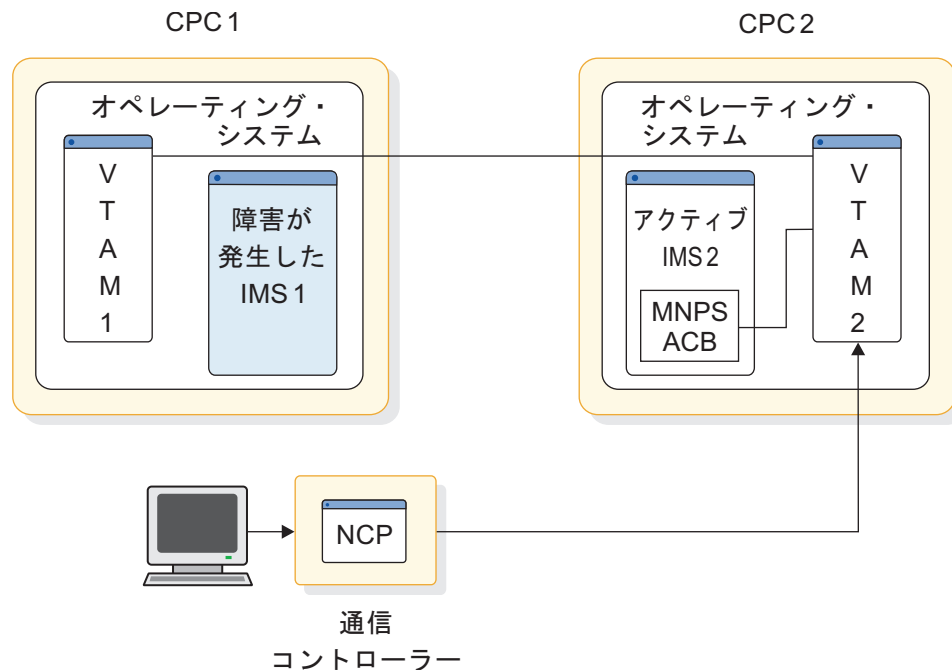


図 73. MNPS を使用する XRF 複合システム・データのログオンの VTAM 処理 (第 2 部)

z/OS バージョン 1.6 以上では、MNPS を使用する XRF 複合システムは、VTAM 持続セッション強制テークオーバー・サポートを利用します。強制テークオーバー・サポートは、アクティブ IMS で障害が発生したが、VTAM にはまだアクティブであるように見えるときに、代替 IMS システムがテークオーバーできるようにします。

アクティブ IMS システムは、FORCETKO が指定された VTAM SETLOGON OPTCD=PERSIST マクロを実行することにより、強制テークオーバーを許可します。これにより、VTAM は、代替 IMS システムが XRF テークオーバー処理中に MNPS ACB をオープンできるようにします。

ネットワーク制御プログラム (NCP)

USERVAR を使用する XRF 複合システムで、NCP は、バックアップ・セッション用の制御ブロックを保持し、代替 IMS がセッションの切り替えを要求したときに切り替えを実行します。MNPS を使用する XRF 複合システムでは NCP の役割はありません。

システム・サポート・プログラム (SSP)

USERVAR を使用する XRF 複合システムで、SSP は NCP を定義し、生成します。SSP はまた、37x5 通信コントローラーへのプログラムのロードも行い、37x5 通信コントローラーの内容を診断目的でダンプしてホスト CPC に戻します。SSP はメッセージを作成することはありませんし、既存のプロシーチャーを変更することはありません。MNPS を使用する XRF 複合システムでは、SSP の役割はありません。

USERVAR を使用する Tivoli NetView for z/OS

インストール・システムでは、ライセンス・プログラムの Tivoli NetView for z/OS のサービスが使用されていることがあります。XRF に対して変更が加えられていなくても、Tivoli NetView for z/OS は VTAM アプリケーション名の変更を XRF 複合システム内外の VTAM に知らせることができます。ネットワークの VTAM が、USERVAR 管理拡張機能を持つバージョン 4 でない場合は、USERVAR 変更をその VTAM に伝搬するために、Tivoli NetView for z/OS CLIST を使用しなければなりません。VTAM アプリケーション名のこのような変更は、初期設定時 (IMS 再始動完了時または /START DC コマンドのプロセス完了時) または、テークオーバー時に発生します。インストール・システムに Tivoli NetView for z/OS がない場合、または VTAM がバージョン 4 より以前である場合は、ネットワークの個々のシステムのオペレーターは、MODIFY コマンドを実行して、VTAM アプリケーション名を更新しなければなりません。

XRF の監視の確立

5 つのパラメーター SURV、LNK、LOG、RDS、および SWITCH を使用して、XRF の監視を確立します。

以下の表では、計画決定、対応するパラメーター、パラメーターのオプションについて説明しています。監視パラメーターとオプションについては、以下を参照してください。

- 2つのシステム間のISCリンクとしてのLNK
- IMSシステム・ログとしてのLOG
- RDSとしての再始動データ・セット(RDS)

表 52. パラメーターの監視オプション

計画決定	パラメーター	オプション
どのような監視メカニズムをXRF複合システムで作動させるか(ステップ1)	SURV	LNK、LOG、RDS
監視メカニズムとしてLNKを指定した場合は、信号間のインターバルはどの程度の長さにするか(ステップ2)	LNK	間隔
LNKがテークオーバー条件である場合は、テークオーバーを考慮するまでに代替IMSが信号を待つ時間はどの程度にするか(ステップ4)	LNK	Timeout
監視メカニズムとしてLOGを指定した場合は、信号間のインターバルはどの程度の長さにするか(ステップ2)	LOG	間隔
LOGがテークオーバー条件である場合は、テークオーバーを考慮するまでに代替IMSが信号を待つ時間はどの程度にするか(ステップ4)	LOG	Timeout
監視メカニズムとしてRDSを指定した場合は、信号間のインターバルはどの程度の長さにするか(ステップ2)	RDS	間隔
RDSがテークオーバー条件である場合は、テークオーバーを考慮するまでに代替IMSが信号を待つ時間はどの程度にするか(ステップ4)	RDS	Timeout
特定の監視信号が送信されない場合に、代替IMSにテークオーバーを要求させるべきか(ステップ3)	SWITCH	LNK,LOG,RDS

監視を確立するには、次の4つのステップを実行します。

1. 複合システムに対してどの監視メカニズムを使用するかを決定します。

監視メカニズムを確立するには、SURVパラメーターを使用します。3つのメカニズムLNK、LOG、およびRDSを組み合わせて指定できます。ただし、監視は重要なので、この3つのメカニズムをすべてを使用することをお勧めします。

代替IMSは、これら3つの監視メカニズムをさまざまな方法で使用します。代替IMSがISCリンクまたはRDSの信号の受信に失敗することは、テークオーバーの原因になる可能性があります。代替IMSがログの新しいレコードの受信に失敗しても、テークオーバーの原因にはなりません。むしろ、IMSはLOGオプションを使用して、RDSまたはISCリンクからの信号の欠如に基づいて代替IMSが行う決定を確認またはオーバーライドします。例えば、SURV=(LNK,LOG)を指定して、代替IMSがISCリンクの信号の受信を停止するとします。ISCリンクを使用した信号送信で発生する障害はテークオーバーを意味しますが、代替IMSはログ・レコードを十分に受信し続ける限り、テークオーバーを要求しません。この場合、IMSは、ISCリンク自体で障害が発生しているため、信頼できる標識ではないと見なします。

USERVAR を使用する XRF 複合システムでは、ISC リンクがまだ確立されていない場合、クラス 1 端末を制御する 37x5 通信コントローラーを使用して ISC リンクを確立してください。

監視メカニズムおよび内部パラメーターは、マスター端末から発行されたコマンドによって動的に変更できます。

関連資料: 監視メカニズムについて詳しくは、819 ページの『DFSHSBxx の XRF パラメーター』を参照してください。

2. 使用する各監視メカニズムに対してインターバル値 (代替 IMS が信号を受信する頻度) を決定します。

どの監視メカニズムを複合システムで作動させるかを確立したら、代替 IMS がその監視メカニズムを使用して信号を受信する頻度を決定しなければなりません。

LNK パラメーターと RDS パラメーターをコーディングして、選択したそれぞれの監視メカニズムの LNK と RDS に対して、信号間のインターバルを確立します。

LNK および RDS に対するタイミングの設定で考慮しなければならないことは、次の 2 つです。

- ISC リンクを使用した通信の速度
- 代替 z/OS のパフォーマンスのオーバーヘッド。代替 z/OS の作業を減らす必要がある場合は、タイミング・インターバルを高く設定してください。

LOG パラメーターをコーディングして、代替 IMS がログの新しいレコードをチェックする頻度を確立します。

インターバル値の詳細については、819 ページの『DFSHSBxx の XRF パラメーター』のトピックを参照してください。

関連資料: IMS には、2 つのシステムに対するインターバル値のコーディングについて特有の規則と推奨があります。これらの規則については、「IMS V15 システム定義」を参照してください。

3. これらの信号の一部または全部が送信されないことをテークオーバー条件にするかどうかを決定します。

SWITCH パラメーターを使用して、監視信号の欠如をテークオーバーの基準として確立します。代替 IMS にある一定のイベントを警告する際に重要な監視メカニズムを、SWITCH パラメーターで指定します。例えば、IRLM を使用してアクティブ IMS でローカル・ロックを管理しているインストール・システム、または、データ共用を使用しているインストール・システムは、テークオーバーの基準として IRLM 障害を指定します。大規模な VTAM ネットワークを持つインストール・システムでは、通常は、VTAM の障害によって IMS TPEND 出口が引き起こされたときに、テークオーバーが発生するように指定します。大規模な非VTAM ネットワークや小規模な VTAM ネットワークを持つインストール・システムでは、通常は、テークオーバーの基準として VTAM の障害を指定しません。SWITCH パラメーターの詳細については、819 ページの『DFSHSBxx の XRF パラメーター』のトピックを参照してください。

4. 3 (770 ページ) の答えが「はい」である場合は、タイムアウト値 (テークオーバーを考慮するまで、代替 IMS が監視信号を待つ時間の長さ) を決定します。

どの監視信号の欠如がテークオーバー条件かを指定した後、テークオーバーを要求するまでに代替 IMS が信号を待つ時間を決定できます。LNK、LOG、および RDS パラメーターでタイムアウト値を指定します。デフォルトのインターバルは、LNK パラメーターの場合は 9 秒、LOG と RDS の場合は 3 秒です。

タイムアウト値が低すぎると、非 XRF 環境で透過である z/OS リカバリー・ルーチンがテークオーバーを発生させる場合があります。このようなルーチンの例として、代替 CPU リカバリー (ACR) ルーチンがあります。これらの不要なテークオーバーが発生しやすいのは、大規模な CPC または IMS 作業負荷が大きい場合です。

XRF プロセスのフェーズ

XRF 複合システムを構成している 2 つのシステムには、6 つの識別可能なフェーズがあります。

XRF のフェーズは次のとおりです。

- 初期設定
- 同期化
- トラッキング
- テークオーバー
- テークオーバーの後処理
- 終了

代替 IMS 用コンソールでシステム状況をチェックすることによって、いつでも現行のフェーズを判別できます。

個々のサブシステムが各フェーズに関与しますが、アクティブ IMS は、同期化、トラッキング、テークオーバーなどのフェーズでは XRF アクティビティーをほとんど行いません。このトピックのサブトピックでは、6 つのそれぞれのフェーズごとのアクティブ・サブシステムと代替サブシステムのアクティビティーについて説明します。

XRF プロセスの初期設定フェーズ

オペレーターが START AVM コマンドを使用するか、あるいは /NRESTART または /ERESTART コマンドを使用して、z/OS 開始タスクとして AVM を立ち上げると、初期設定フェーズが始まります。

START コマンドは、モジュールおよび制御ブロックをロードして、オプションで、アクティブ・システムと代替 IMS システムを接続する ISC リンクでセッションをオープンします。

以下の図は、XRF 初期設定フェーズを示しています。

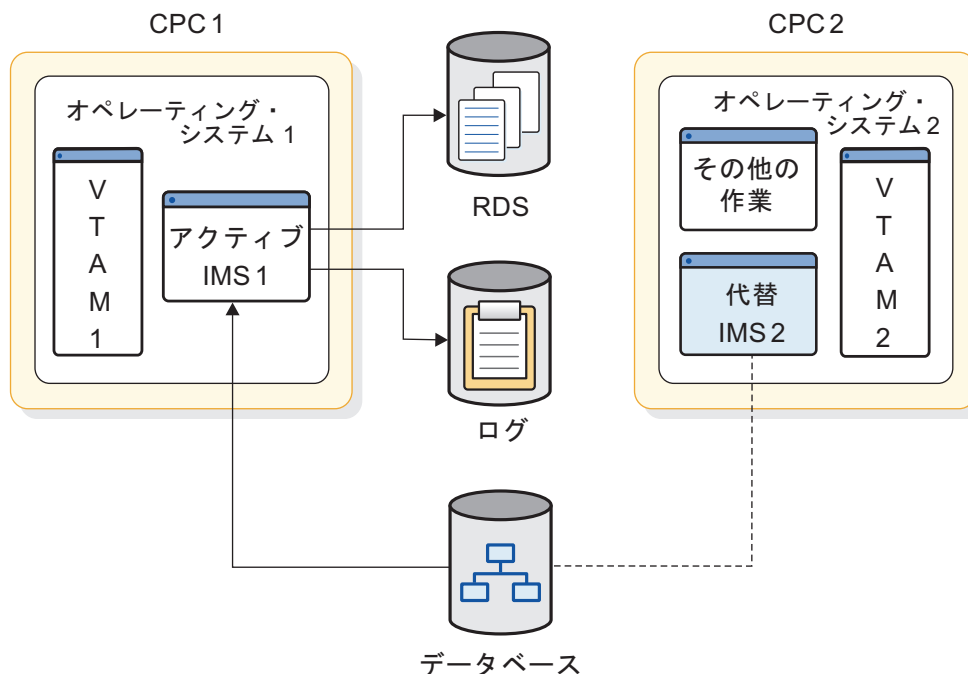


図 74. XRF 初期設定フェーズ

アクティブ IMS のオペレーターは、通常の再始動または緊急時再始動のときのように、z/OS START コマンドを使用して、アクティブ IMS を立ち上げます。アクティブ IMS が起動していると、代替 IMS のオペレーターは、いつでも代替 IMS を立ち上げることができます。これらのコマンドは、代替 IMS のオペレーターがオプション AUTO=N を START IMS コマンドに追加して、IMS コマンド /ERESTART BACKUP を発行する場合を除いて、どちらのシステムに対しても同様です。

重要: オペレーター・エラーによって、システム保全性またはデータベース保全性が失われることがあります。

代替 IMS は、テークオーバーが発生するまでは緊急時再始動に似た状態にあります。テークオーバー準備のために、代替 IMS は、アクティブ IMS とデータベースを共用します。代替 IMS はデータベースをオープンしますが、テークオーバーが終わるまでそのデータベースにはアクセスしません。

通常の IMS 初期設定機能の他にも、IMS は XRF 複合システムを設定するために以下のことを行います。

1. 新しい DFSHSBxx IMS PROCLIB メンバーを処理します。詳しくは、819 ページの『DFSHSBxx の XRF パラメーター』を参照してください。
2. 新しい XRF/IMS モジュールをロードして、新しいプールおよび作業域を取得します。新しいモジュールはほとんど、16 MB 境界より下の専用制御領域にあります。
3. 818 ページの『XRF のための IMS.PROCLIB メンバーの定義』で説明しているとおり、RSENAME キーワードを使用して、z/OS 可用性マネージャーに両方のサブシステムを認識させます。

4. RSE 名を使用して、アクティブ・サブシステムと代替サブシステムの両方に DBRC 環境を確立します。テークオーバーが発生すると、アクティブ IMS の許可が、代替 IMS に自動的に渡されます。
5. VTAM にアクティブ IMS システムの APPLID を知らせます。

関連資料 オペレーターがアクティブおよび代替 IMS システムを立ち上げる方法について詳しくは、「IMS V15 オペレーションおよびオートメーション」を参照してください。

XRF プロセスの同期フェーズ

同期化フェーズでは、代替 IMS は、以下の図に示されているとおり、アクティブ IMS のイメージになります。環境が完全に初期設定されると、このフェーズが始まります。

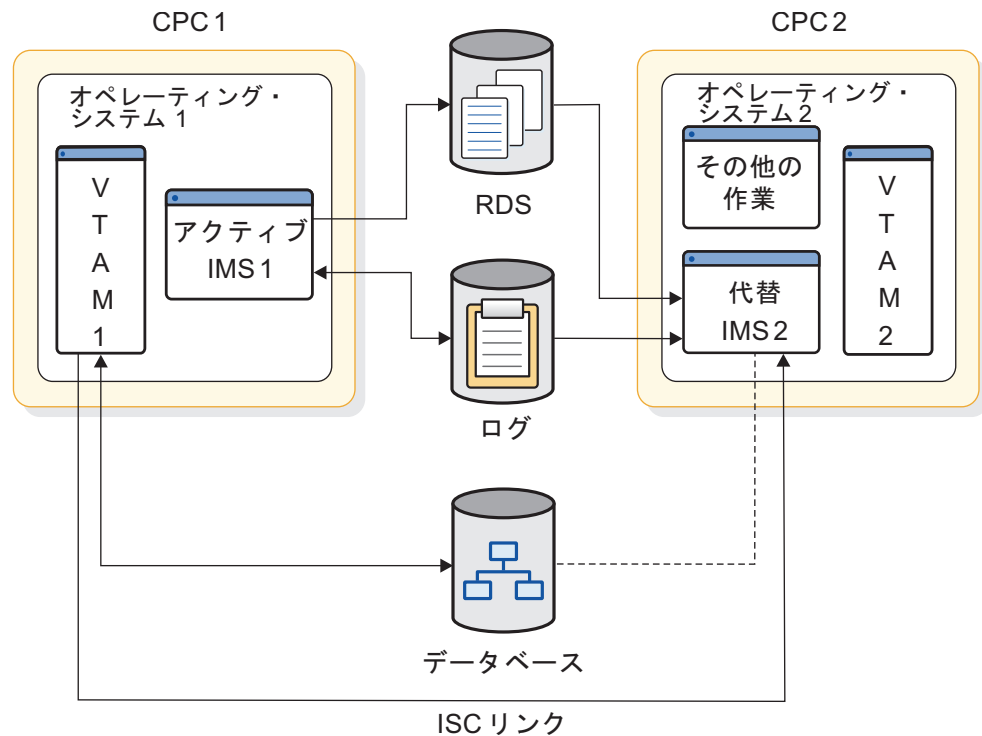


図 75. XRF 同期化フェーズ

代替 IMS に制御ブロックを作成するプロセスは、非常に簡単です。代替 IMS は、最後のチェックポイントが SNAPQ であったかどうかを判別します。最後のチェックポイントが SNAPQ ではなく、かつ、ISC リンクがある場合には、代替 IMS は、アクティブ IMS が SNAPQ チェックポイントを取得することを要求します。SNAPQ ログ・レコードは、両方のシステムの IMS システム定義が一致していることを確認するためと、代替 IMS 上でアクティブ IMS の初期「スナップショット」を作成するために使用されます。

ISC リンクがない場合には、代替 IMS はメッセージを発行して、アクティブ IMS のマスター端末オペレーターに SNAPQ チェックポイントを要求するように依頼します。

代替 IMS は IMS システム・ログを開いて、SNAPQ チェックポイントを検索し、ログ・データ・セットからの読み取りを行います。

ISC リンクが存在する場合は、チェックポイントがログに書き込まれている間に、代替 IMS にメッセージが送信されます。ISC リンクが存在しない場合、代替 IMS は、SNAPQ の完了について再始動データ・セット (RDS) をモニターします。

IMS は、ダンプの内容を使用して、以下のことを行います。

- 代替 IMS は、アクティブ IMS が使用している MODSTAT および RDS を判別します。
- 代替 IMS は、アクティブ IMS によってアクセスされるデータベースとエリアの事前割り振りと事前オープンをしようとします。
- MSDB は、SNAPQ チェックポイント・レコードから代替 IMS によってロードされます。別個の MSDB チェックポイント・データ・セットが代替 IMS で使用可能でなければなりません。
- XRF 複合システムが USERVAR を使用する場合、アクティブ IMS でオープンされている端末のバックアップ・セッションは、BACKUP オプションで定義されているそれらのクラス 1 端末用の代替 IMS で開始されます。
- セキュリティー制御ブロックは、セキュリティに RACF を使用している場合、RACF データ・セットから代替 IMS にロードされます。
- DBRC サブシステム・レコードは、代替サブシステムが存在していることを示すために更新されます。これによって、アクティブ IMS の障害が発生しても、DBRC 'SIGNOFF ABNORMAL' メッセージが抑制されます。
- アクティブ IMS の従属領域の状況は、代替 IMS で追跡されます。

代替 IMS がログに書き込めるのは、テークオーバー後だけです。

オペレーターは、代替 IMS の従属領域を始動して、その領域をテークオーバー準備状態にします。

代替 IMS が SNAPQ チェックポイントを処理すると、同期化フェーズは完了します。代替 IMS の制御ブロックがアクティブ IMS の制御ブロックと同期化されると、代替 IMS は、アクティブ IMS のプロセスをテークオーバーすることができます。この時点から、予定テークオーバーを要求できます。

XRF プロセスのトラッキング・フェーズ

ほとんどの時間、XRF 複合システムはトラッキング・フェーズにあります。アクティブ IMS は、IMS ユーザーの要求を処理してログに書き込み、初期設定フェーズで設定された監視メカニズムを使用して信号を送信します。代替 IMS は、テークオーバー時に代替 IMS が行わなければならない作業を減らすために、アクティブ IMS をトラッキングします。

トラッキング・フェーズは、773 ページの『XRF プロセスの同期フェーズ』に示されている XRF 同期フェーズに類似しています。

代替 IMS のトラッキング・アクティビティーには、次のものがあります。

- ログを読み取る。
- 制御ブロックを更新するためにログ情報を使用する。

- XRF 複合システムが USERVAR を使用する場合、クラス 1 端末に対して、バックアップ・セッションをオープンおよびクローズする。
- 監視メカニズムとログ・レコードをチェックして、アクティブ IMS に障害の兆候がないかどうかを調べる。

トラッキング・アクティビティーは、代替 IMS の非格納リソースをほとんど使用しません。実記憶域の使用は、IMS ストレージをどのようにページ固定するか、そして、サブシステムがアクティブの場合だけ有効なページ固定オプションを持つ DFSFIXxx のメンバーを定義するかどうかによって決まります。

代替 IMS の制御ブロックの更新

代替 IMS は、OLDS を読み取って、以下のものをトラッキングします (一方の OLDS が失われた場合に、もう一方の OLDS を使用してトラッキングを続けられるよう、重複 OLDS を使用してください)。

- 通信ネットワークの状況

緊急時再始動の処理中に行われる通常の端末状況のトラッキングだけではなく、代替 IMS は端末の接続状況をモニターします。これは、初めは SNAPQ チェックポイントのレコードを基に行われますが、その後は、アクティブ IMS によって書き込まれた端末の接続状況変更ログ・レコードを使用します。USERVAR を使用する XRF 複合システム内のクラス 1 端末の場合、バックアップ・セッションは代替 IMS によって保守されます。クラス 1 端末セッションがアクティブ IMS で終了すると、バックアップ・セッションも代替 IMS で終了します。ユーザーが、代替 IMS にログオンすることはできません。端末タイプについては、800 ページの『XRF 複合システムの端末』を参照してください。

- アプリケーション・プログラムとトランザクションの状況

アプリケーション・プログラムがアクティブ IMS でスケジュールされると、代替 IMS は制御ブロック構造を作成して、従属領域の状況を追跡します。

オペレーターは、代替 IMS の従属領域を事前開始して、代替 IMS がテークオーバーをさらに迅速に完了できるようにします。

- データベースの状況

アクティブ IMS でデータベースおよびエリアが割り振られてオープンされると、代替 IMS はこれらの同じデータベースとデータ・セット・エリアを事前割り振りして事前オープンします。すべてのデータベースおよびエリアに対する IMS マクロ定義は、SHARED として定義しなければなりません。代替 IMS は、すべての DBRC データベース権限の追跡も行います。

推奨事項: すべてのデータベースとエリアを DBRC に登録してください。代替 IMS はまた、すべてのロックの状況を追跡して、コミットされていないデータベース変更があるかどうか調べます。

- メッセージ・キューの状況

SNAPQ チェックポイント・レコードから、代替 IMS にメッセージ・キューが作成されます。メッセージ・キューは、OLDS レコードによって、代替 IMS で更新されます。ローカル・メッセージ・キューは、オペレーターと通信するために代替 IMS が使用します。IMS システムの定義時に、2 つのマスター端末と 2

つの 2 次マスター端末が指定されます。アクティブ IMS のマスター端末は、アクティブ IMS を制御し、代替 IMS のマスター端末は、代替 IMS を制御します。

- MFS プールの状況

MFS ブロックは、アクティブ IMS にロードされて解放されると、代替 IMS にプリロードされて解放されます。

- 従属領域の状況

代替 IMS は、従属領域の状況を反映するように、制御ブロックを初期設定して保持します。代替 IMS は、コミットされていないプロセスのバックアウトがテークオーバー時に発生するように、領域内のすべてのアクティビティをモニターします。

この情報によって、代替 IMS は制御ブロックを更新し、アクティブ IMS を複製する環境を保守します。さらに、これは、アクティブ IMS が IMS データベースとデータ域を割り振ってオープンすることをログが示すと、それらを事前割り振りして、事前オープンします。また、MFS 制御ブロック、バックアウトに使用するためのプログラム仕様ブロック (PSB)、およびデータ管理ブロック (DMB) のロードも行います。

アクティブ IMS の監視

760 ページの『監視メカニズム』に記載してあるように、代替 IMS はアクティブ IMS のテークオーバー条件の一部を示す作業を監視メカニズムに依存しています。監視メカニズムに再始動データ・セット (RDS) が含まれている場合は、アクティブ IMS は定期的に RDS にタイム・スタンプを入れます。監視メカニズムに ISC リンクが含まれている場合は、アクティブ IMS は ISC リンクを使用して定期信号を送信します。監視メカニズムにログが含まれている場合は、代替 IMS は定期的にチェックを行い、ログ・レコードが引き続き到着することを確認します。

XRF プロセスのテークオーバー・フェーズ

代替 IMS は、以下の 3 つの理由のいずれかにより、テークオーバーを要求します。

- アクティブ IMS でのプロセスを継続できないほど重大な問題が発生している。このような問題には、次のようなものがあります。
 - IMS の異常終了
 - z/OS の障害、ループ、または待ち状態
 - CPC 障害

これらの問題は、少なくとも、再始動データ・セット (RDS) および ISC リンクを含む基本監視メカニズムを確立しているという前提に基づいています。監視メカニズムがアクティブ IMS で障害を検出する場合、またはオペレーターが /SWITCH SYSTEM FORCE コマンドを実行する場合、代替 IMS はログを確保して、アクティブ IMS がさらにプロセスを行うことを防止しテークオーバーを続行します。オペレーターは、元のアクティブ IMS を終了するか、CPC をリセットして、データベースがさらに更新されるのを防止しなければなりません。

アクティブ IMS が異常終了すると、アクティブ IMS の AVM は入出力防止を呼び出して、データベースがさらに更新されるのを防止します。

- オペレーターが予定テークオーバーを開始している。

予定テークオーバーを開始するために、アクティブ IMS または代替 IMS の IMS オペレーターは、/SWITCH SYSTEM コマンドを実行して、テークオーバーを発生させます。IMS は、すべての従属領域に対して、すべての経過中メッセージが完了するまで待って、異常終了します。元のアクティブ IMS の AVM は入出力防止を実行します。代替 IMS に対してはバックアウトを実行する必要はありません。予定テークオーバーが使用される例としては、システム・プログラマーがアクティブ IMS で保守を行いたい場合などがあります。785 ページの『予定テークオーバーの実際の用途』には、予定テークオーバーのその他の使用例が示してあります。

- DFSHSBxx PROCLIB メンバーで設定した他の基準が、テークオーバーのトリガーになっている。

760 ページの『監視メカニズム』で説明されているように、監視メカニズムから信号が送信されない場合にテークオーバーを行うように指定することができます。例えば、10 秒間 ISC リンクを通して信号が送信されないことがテークオーバー条件であることを代替 IMS に知らせることができます。

テークオーバー基準として設定できるその他の 2 つの障害は、次のとおりです。

- IRLM の異常終了

次の 2 つの理由から、複合システムには IRLM があります。

- アクティブ IMS の IRLM は、XRF IMS が複合システム外の IMS システムとブロック・レベルで共用しているデータベースへのアクセスを制御します。この場合、XRF 複合システムは、2 つの IRLM を、それぞれアクティブ・サブシステムと代替サブシステムに持ちます。
- IRLM は、ローカル・ロックを管理しています。

どちらの場合にも、アクティブ IMS の IRLM が終了し、その時点で IMS STATUS 出口ルーチンが呼び出されたら、テークオーバーを発生させたい場合があります。

- VTAM の障害

VTAM が IMS TPEND 出口ルーチンを呼び出した場合に、代替 IMS にテークオーバーを要求するように指示することができます。

VTAM および IRLM で障害が発生すると、インストール・システムでパフォーマンスが低下するため、この障害が発生した場合には、テークオーバーを回避しなければならないことがあります。

IMS.PROCLIB のメンバー DFSHSBxx に追加するパラメーターを使用して、監視信号が送信されないこと、または、VTAM 障害や IRLM 障害が発生したことをテークオーバー条件として指定します。768 ページの『XRF の監視の確立』で、これらのパラメーターについて説明しています。

代替 IMS は、RDS および ISC リンク上で受け取る信号を検査して、ログで受け取るレコードを読み取り、オペレーターのコマンドをモニターします。最終的にはテークオーバーを要求します。この時点で、テークオーバーが開始されます。IMS がどのように調整されているかによって、テークオーバーが自動的に行われる場合と、オペレーターがある作業を実行するのを待ってから、テークオーバーが行われる場合があります。テークオーバー・フェーズを、以下の図に示します。

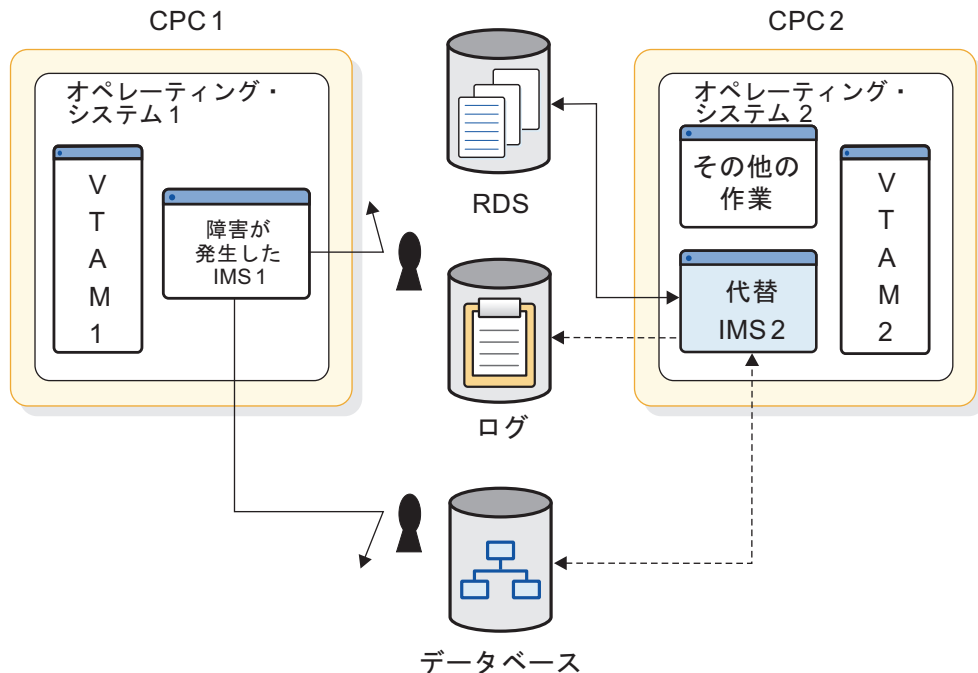


図 76. XRF テークオーバー・フェーズ

テークオーバー中に、代替 IMS は同期化フェーズとトラッキング・フェーズで収集したデータを使用して、自分自身がアクティブ IMS に移行します。また、元のアクティブ・サブシステムでの作業が停止していることも確認します。このフェーズ中の XRF の主な目的は、データベースの保全性を保証することです。このフェーズの終わりに、代替 IMS は新しいアクティブ IMS になります。テークオーバーが開始されると、取り消すことはできません。完了するまで続行されます。

テークオーバーの開始

テークオーバー時に障害が発生したアクティブ IMS の状態は不確定なので、代替 IMS は作業のほとんどをテークオーバー時に行います。代替 z/OS は、代替 IMS で処理されていた非 XRF 作業負荷をどう扱うかを決定しなければなりません。798 ページの『代替 IMS での非 XRF 作業負荷』を参照してください。

このサブトピックに記載のイベントは、IMS が異常終了したり、アクティブ IMS のオペレーターが /SWITCH SYSTEM コマンドを実行した場合、テークオーバー中に発生します。他のイベントによって引き起こされるテークオーバー処理については、783 ページの『入出力防止のないテークオーバー処理』を参照してください。

テークオーバー中に、アクティブ IMS は以下のことを行おうとします。

- 障害を認識して、終了メッセージをログに書き込みます。着信メッセージのプロセスを停止します。
- アクティブ・サブシステムと代替サブシステムが別の CPC に存在する場合は、アクティブ IMS の OLDS および WADS に対して RESERVE を発行します。

推奨事項: これらの装置に他のデータ・セットを配置すると、RESERVE の発行後にアクセス不能になりますので配置しないでください。テークオーバーの開始時点で、これらのデータ・セットにアクセスすると、テークオーバーが停止されます。

アクティブ IMS システムと代替 IMS システムが同じ CPC に存在する場合は、アクティビティーが異なります。代替 IMS は、アクティブ IMS を ABTERM して、アクティブ IMS 上の IMS リソース終結処理ルーチンが、ログへの入出力がアクティブ IMS で終了した旨を代替 IMS に通知するまで待つ必要があります。

- IMS がログにアクセスできなくなったら、WADS を使用して OLDS をクローズします。DBRC はテークオーバーを障害と見なさないので、代替 IMS は以前に割り振られた OLDS の次のセットに切り替わります。
- 代替 IMS のためのリソースを取得しようとする z/OS システム・リソース・マネージャー (SRM) を呼び出します。データベースおよびセッションのリカバリーのために代替 IMS が必要とする実記憶域が一時的に多くなるので、SRM は代替 IMS による新しい作業負荷の受け入れを促進します。
- テークオーバー中に、不完全なメッセージを除去します。代替 IMS にメッセージ・キューがすでに作成されていると、テークオーバー時に不完全なメッセージが発生します。
- リカバリー不能なメッセージの削除を延期します。
- 障害発生時にプロセス中であつたりリカバリー可能メッセージを再エンキューします (バックアウトしなくてよい場合)。バックアウトが必要な場合は、これらのメッセージは、各バックアウトの完了時に自動的に再エンキューされます。
- IMS マスター端末のオペレーターが、テークオーバーのために重大なメッセージを失うようなことがないように、ローカル・メッセージ・キューのメッセージをマージします。

アクティブ IMS は、OLDS バッファーを除去できない場合以外は、上述のタスクを実行します。

AVM が、アクティブ IMS で動作している場合は、入出力防止を実行します。通常、入出力防止は、数秒以内の入出力防止要求で完了します。761 ページの『z/OS および z/OS エレメントの利点』で、障害が発生している IMS によるデータベースへの書き込みを停止するために AVM が行うアクションについて説明しています。AVM が動作していない場合は、アクティブ IMS のオペレーターは、CPC をリセットするか、または IMS を終了して、入出力防止を手動で実行しなければなりません。入出力防止が完了したら、アクティブ IMS のオペレーターは、代替 IMS のオペレーターに完了を通知する必要があり、また、代替 IMS のオペレーターは、メッセージ AVM005A に対して「GO」と応答する必要があります。

この通知前であっても、代替 IMS はデータベースを保護しながら、テークオーバーを続行します。代替 IMS は、現行の OLDS および WADS を含む DASD を確保することによって、重要な共用リソースを取得します。

ログを確保した後、代替 IMS は以下の 3 つのアクティビティを同時に実行します。

- メッセージ・キューおよびデータベースのデータをリカバリーする。
- ネットワーク変更を実行する。
- 新しいトランザクション、および、再処理されているトランザクションを実行する。

以下のトピックでは、この 3 つの並列アクティビティについて説明します。

- 『メッセージ・キューおよびデータベース・データのリカバリー』
- 781 ページの『ネットワーク変更の開始』
- 782 ページの『新しいトランザクションおよび再処理されたトランザクションの実行』

メッセージ・キューおよびデータベース・データのリカバリー

アクティブ IMS がプロセスを開始しなかったメッセージを処理するために、代替 IMS は、メッセージ・キューのコピーが完了していることを確認します。それにより、これらのメッセージを処理して、データベースを更新できます。

経過中トランザクションをリカバリーするために、代替 IMS は、アクティブ IMS が開始したが完了しなかったすべてのトランザクションをデータベースからバックアウトし、トランザクションを再処理します。また、高速機能データベースの健全性を保証するために、順方向リカバリーを実行します。

代替 IMS は、障害が発生しているアクティブ IMS がデータベースに書き込めなくなったことを確認するまで、障害が発生しているアクティブ IMS によって上書きされる可能性のあるデータ・ブロックまたは VSAM 制御インターバルに対する変更をデータベースへ書き込むのを延期することによって、入出力許容を実行します。代替 IMS は、変更をストレージに一時的に保持するので、この変更へのすべての参照は、データベースではなく、そのストレージに対して行われます。状況表示に MODE というヘッディングが付いた I/O TOLERATION という語が表示されます。障害が発生しているアクティブ IMS で AVM がアクティブになっていない場合は、アクティブ・コンソールのオペレーターが、IMS を取り消すか、またはシステムをリセットして、入出力操作を防止しなければなりません。

代替 IMS サブシステムは、入出力許容の一部として以下のアクションを実行します。

- 障害時に更新されたデータベース・ブロックまたは制御インターバルを、拡張エラー・キュー・エレメント (EEQE) を使用して識別し、タグを付けます。
- これらのレコードを仮想バッファに保持して、ユーザーが入出力許容を終了するまで、動的バックアウトおよびトランザクション処理の間にアクセスできるようにします。また、EEQE を DBRC に伝達して、共用サブシステムに通知します。

- 経過中で不完全な更新をバックアウトし、スケジュール変更して、プロセスできるようにします。
- テークオーバー時に拡張されていたデータベースの使用を停止します。

IMS は、XRF 複合システム外で IMS システムと共用している (IRLM のサービスを使用) 任意のデータへのアクセスを制限します。特に、他の IMS システムが、経過中トランザクションに影響されるデータのブロックにアクセスするのを防止します。

メッセージ AVM005A: REPLY "GO" WHEN I/O PREVENTION COMPLETES に対し、代替 IMS のオペレーターが「GO」と応答する (または、/UNLOCK SYSTEM コマンドを実行する) と次のようになります。

- IMS は、ストレージに保持していた経過中データに対する変更でデータベースを更新します。
- IMS は、データベースのすべての部分を XRF 複合システム外の IMS システムで使用できるようにします。
- 入出力許容は終了します。

IMS コマンド /UNLOCK SYSTEM は、「GO」と応答した場合と同じアクションを実行します。

ネットワーク変更の開始

テークオーバー中にネットワークに対して行われる変更は、XRF 複合システムが MNPS を使用するのか、USERVAR を使用するのかによって異なります。

MNPS を使用する XRF 複合システムでのネットワークの変更

XRF テークオーバーの場合、代替 IMS システムは新しい MNPS ACB をオープンします。この新しい MNPS ACB は、障害のある IMS システムと同じ MNPS ACB を使用するため、障害のある IMS システムの MNPS ACB をクローズしないと、新しい MNPS ACB は開始できません。VTAM は、新しい MNPS ACB がオープンするまで、すべてのクラス 1 端末セッションを持続セッションとしてオープンしたまま維持します。新しい MNPS ACB がオープンすると、VTAM は、その新しい MNPS ACB を通じてすべての端末セッションを経路指定します。

テークオーバー時に、端末セッションは、次の 3 つの方法のいずれかで処理されます。クラス 1 端末のセッションは新しい MNPS ACB に切り替わり、クラス 2 端末のセッションは短時間クローズした後で再確立され、クラス 3 端末のセッションは停止します。クラス 3 では、サービスは XRF が存在しない場合とほとんど同じです。800 ページの『XRF 複合システムの端末』では、クラス 1、クラス 2、クラス 3 の各端末のさまざまな特性を説明しています。

代替 IMS へのセッションの切り替え速度、または再確立の速度は、代替 IMS の CPC のタイプ、端末の切り替え優先順位、ネットワークのサイズによって決まります。

IMS は、クラス 2 端末 (MSC 物理リンクによって XRF IMS と通信する端末など) とのセッションを再確立しようとします。

USERVAR を使用する XRF 複合システムでのネットワークの変更

XRF テークオーバー後にさらに端末ユーザーが代替 IMS にログオンできるようにするためには、VTAM は USERVAR テーブルのアプリケーション・プログラム名を変更しなければなりません。代替 IMS は MODIFY USERVAR コマンドを実行して、それ自身の USERVAR テーブルの項目を変更します。このプロシージャが自動化されていない場合は、XRF IMS と通信する他のすべての VTAM のネットワーク・オペレーターは、MODIFY コマンドを実行して、USERVAR テーブルのアプリケーション・プログラム名を変更しなければなりません。このプロシージャは、Tivoli NetView for z/OS によって自動化できます。代替 IMS が MODIFY コマンドを実行すると、Tivoli NetView for z/OS はセッションが確立されている他の Tivoli NetView for z/OS に、アプリケーション・プログラム名の変更を知らせます。

テークオーバー時に、端末で 3 つの明らかな変更が発生します。クラス 1 端末のセッションはバックアップ・セッションに切り替わり、クラス 2 端末のセッションは再確立され、クラス 3 端末のセッションは停止します。クラス 3 では、サービスは XRF が存在しない場合とほとんど同じです。

800 ページの『XRF 複合システムの端末』では、クラス 1、クラス 2、クラス 3 の各端末のさまざまな特性を説明しています。

代替 IMS は、NCP がクラス 1 端末の個々の 1 次セッションを、事前オープンされているバックアップ・セッションに切り替えることを要求します。NCP は、すでにオープンされているバックアップ・セッションがないセッションを切り替えることはできません。アクティブ IMS の障害によって、障害が発生している IMS とのセッションが終了されなかった場合は、NCP が 1 次セッションを終了します。

バックアップ・セッションへのセッションの切り替え速度、または再確立の速度は、代替 IMS の CPC のタイプ、端末の切り替え優先順位、ネットワークのサイズ、切り替えを実行する NCP の数によって決まります。

IMS は、クラス 2 端末 (MSC 物理リンクによって XRF IMS と通信する端末など) とのセッションを再確立しようとします。

新しいトランザクションおよび再処理されたトランザクションの実行

代替 IMS はアクティブ IMS からログを分離して、メッセージ・キュー内のコミットされていないメッセージを削除します。そして、リカバリーを実行するのに必要なロックを再取得して、新しいトランザクションおよび再処理されたトランザクションのプロセスを開始します。IMS は端末のセッションをリカバリーして、共用データベースを使用できるようにするので、新しいトランザクションのプロセスが増えます。

テークオーバー時に、非共用データベースを即時に使用できます。データベースのリカバリー管理 (DBRC) の再確認が完了した後、共用全機能データベースが使用可能になります。順方向リカバリーおよび DBRC の再確認が完了した後、および、単純チェックポイントがとられた後、共用高速処理データベース (DEDB) が使用できるようになります。

入出力防止のないテークオーバー処理

780 ページの『メッセージ・キューおよびデータベース・データのリカバリー』、781 ページの『ネットワーク変更の開始』、782 ページの『新しいトランザクションおよび再処理されたトランザクションの実行』の各トピックでは、IMS の異常終了および /SWITCH SYSTEM ACTIVE コマンドで発生するテークオーバー処理について説明します。その他のすべてのテークオーバーの原因では、AVM は入出力防止を実行しません。アクティブ IMS の IMS オペレーターは、アクティブ IMS がデータベースへの書き込みができないことを代替 IMS の IMS オペレーターに通知する前に、2 つの処置のいずれかを実行しなければなりません。

- CPC または z/OS で障害が発生した場合は、オペレーターは CPC をリセットしなければなりません。
- 他のすべての障害の原因では、オペレーターは IMS を終了し、入出力防止を呼び出さなければなりません。

アクティブ z/OS オペレーティング・システムと代替 z/OS オペレーティング・システム間では通信は行われないため、アクティブ z/OS オペレーティング・システムのオペレーターは、代替 z/OS オペレーティング・システムのオペレーターにデータベースが安全であることを通知する必要があります。

テークオーバー時の IRLM プロセス

以下の図は、データベースを別の XRF 複合システムと共用する XRF 複合システムを示しています。基本と 2 次とは、IMS パートナー間の関係を示しています。共用グループ内のすべての IRLM 間にコミュニケーションが存在します。1 つの複合システムは、以下のもので構成されています。

- IRLM1 で動作するアクティブ IMS である IMS1
- IRLM2 で動作する代替 IMS である IMS2

2 番目の複合システムは、以下のものから構成されています。

- IRLM3 で動作するアクティブ IMS である IMS3
- IRLM4 で動作する代替 IMS である IMS4

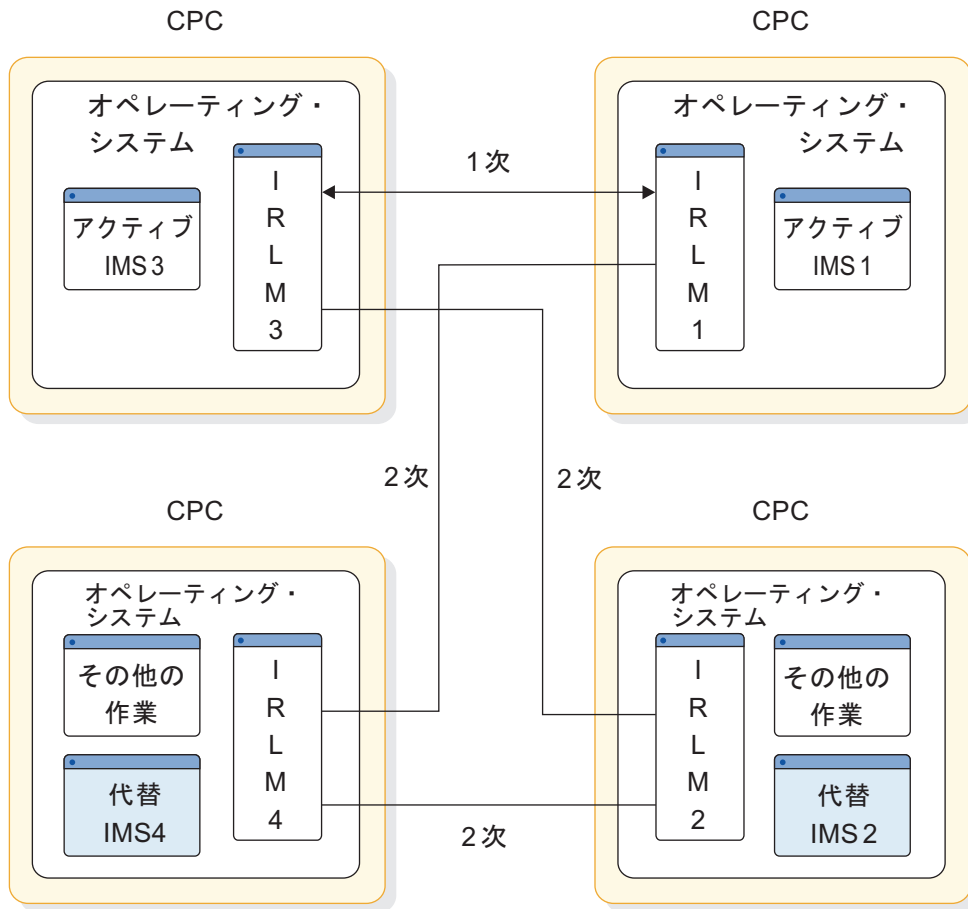


図 77. 2 つのデータ共有 XRF 複合システム

オペレーターがこれらの複合システムを始動するときに、IMS は、以下の共用関係を初期設定します。

- IMS1 および IMS3 間の基本共用パートナー
- 次の二者間の 2 次パートナー
 - IMS3 と IMS2
 - IMS1 と IMS4
 - IMS2 と IMS4

IMS1 の障害によって発生したテークオーバー時に、IMS2 は IRLM2 にテークオーバーを通知して、オペレーターに IRLM TAKEOVER ISSUED というメッセージを送ります。IRLM2 は IRLM3 にテークオーバーの準備をするよう通知し、IMS2 は IMS1 が保持していたロックを取得します。IMS2 でテークオーバー処理が完了すると、IMS2 は IMS1 として識別されるようになります。これにより、IMS3 と IMS2 (別名 IMS1) は基本共用パートナーになります。IMS2 はメッセージ DFS3887 を発行します。

テークオーバーが完了すると、IMS1 は IRLM1 上の代替 IMS として再始動できます。

ブロック・レベルのデータ共有が、IRLM オペレーターのテークオーバーの後処理プロシージャに追加されます。オペレーターは、代替 IMS として IMS1 を立ち上げる前に、IRLM1 を再始動して、下記の両者間に 2 つの新しい 2 次 IRLM セッションを確立しなければなりません。

- IRLM3 と IRLM1
- IRLM4 と IRLM1

関連資料 他の複合システムでも、テークオーバーの後に同様なプロシージャが行われます。これらのプロシージャについては、「IMS V15 オペレーションおよびオートメーション」を参照してください。

XRF IMS は、ローカル・ロック・マネージャーとして IRLM を使用します。この場合、IRLM は相互通信しません。テークオーバー時に、代替 IMS は自身のロックを取得して、テークオーバー・メッセージを発行します。

IRLM は、APPL2 パラメーターと APPL3 パラメーターだけを使用して、定義できます。2 つのアクティブ IMS システムが 2 次セッションを持つ 2 つの異なる IRLM に通知すると、2 次セッションが 1 次セッションにプロモートされます。プロモートされたセッションは、テークオーバーによって再構成を強制されるまで、あるいは、有効なデータ共有状態が別の IRLM によって発生するまで、1 次セッションのままです。1 次セッションがセッション障害や IRLM の損失によって中断されても、別の 1 次セッションが作成されていないければ、そのセッションを再確立することができ、このセッションが自動的に 1 次セッションになります。

セッションの両端は等しく定義しなければならず、個々のセッションは固有でなければなりません。つまり、2 つの APPL2 セッションを定義することはできません。APPLS パラメーターは、テークオーバーが発生するまで、自動セッション判別をオーバーライドします。

テークオーバー発生後に 1 次セッションが 2 次セッションに格下げされるときには、オペレーターの介入は必要ありません。

予定テークオーバーの実際の用途

XRF を使用して予定テークオーバーを行うことにより、以前には夜間または週末に行わなければならなかった作業を日中に行うなど、柔軟な対応が可能になります。アクティブ IMS のオペレーターが /SWITCH SYSTEM ACTIVE コマンドを実行して、アクティブ IMS の作業負荷が代替 IMS に移ると、予定テークオーバーが始まります。代替 IMS がテークオーバーを実行する前に、アクティブ IMS は静止しようとしています。従属領域は、同期点に達することができます。オペレーターが元のアクティブ IMS をダウンさせた後、システム・プログラマーは、元のアクティブ IMS で以下の作業を実行できます。

- バックアップ・プロシージャのテスト
- コードの保守
- 予防保守
- ハードウェア保守
- ライブラリーの保守
- ハードウェア構成変更

計画した作業を実行した後、そのシステムを XRF 複合システムの代替 IMS として立ち上げます。

予定テークオーバーは、慎重に使用しなければなりません。ハードウェアの初期設定と IMS の初期設定は、両方の環境で同時に発生しなければなりません。例えば、アクティブ IMS と代替 IMS は同じシステム定義を持っていないなければならないので、システムが代替システムであるときに、IMS システム定義を変更するためにテークオーバーを要求することはできません。オンライン変更を使用してシステム定義変更を行えない場合は、両方のサブシステムの安全なシャットダウンを行い、その後に IMS のコールド・スタートを行う必要があります。

一般に、エンド・ユーザーの作業を中断させずに、ほとんどのプログラム一時修正 (PTF) を実施できます。例外は、両方のシステムのサービスを同時に実施するために XRF 複合システムを終了しなければならない場合です。このようなケースは、PTF で識別されます。IMS サービスでは、保守に非 XRF 環境での IMS コールド・スタートや、IMS ログ・レコード・フォーマットの変更が必要ない場合は、どちらかのシステムの動作中に PTF を行うことができます。

IMS サービスの需要が増加または減少した場合に、予定テークオーバーを使用して、作業負荷を別の環境にシフトします。例えば、アクティビティーがピークの昼間に、ある CPC で実動作業を実行できます。アクティビティーが低い夜間に、作業負荷を別の CPC にシフトできます。毎日このような 2 つのテークオーバーを行うことは、インストール・システムでの XRF の効果的な使用方法です。

XRF プロセスのテークオーバーの後処理フェーズ

テークオーバーの後処理フェーズは、テークオーバー・フェーズの完了とオーバーラップします。新しいアクティブ IMS が新しいユーザー・トランザクションを処理するとき、IMS は引き続き DL/I バックアウト、高速機能順方向リカバリー、ネットワーク切り替えを実行します。入出力防止が完了して、元のアクティブ IMS が終了したことが新しいアクティブ IMS に通知された後、オペレーターは別の代替 IMS を立ち上げることができます。

新しいアクティブ・サブシステムで以下のイベントが同時に発生します。

- 標準高速機能 DEDB 順方向リカバリーが完了します。IMS は、書き込まれていないがコミットされている変更を含む VSAM CI に対するロックを再取得します。これらの変更のそれぞれに対して、EEQE が作成されます。
- DL/I 再始動データベースのバックアウトが完了します。代替 IMS は、再始動中にクローズされるのではなく、以前のアクティブ IMS からデータベースに対する DBRC 許可を継承します。
- 新しいアクティブ IMS が、DFSHSBxx の DEFERFIX=xx パラメーターで識別された非高速機能オプションをページ固定します。
- 新しいアクティブ IMS が、それがアクティブ IMS であることを z/OS 可用性マネージャーに通知します。障害が発生したアクティブ IMS のオペレーターは、入出力防止が完了したら、代替 IMS のオペレーターに通知しなければならず、それによって、代替 IMS に対する入出力許容を中断できます。
- サービスがクラス 3 端末へ戻されます。

テークオーバーが完了すると、以下の図に示されているように、代替 IMS は、新しいアクティブ IMS になります。

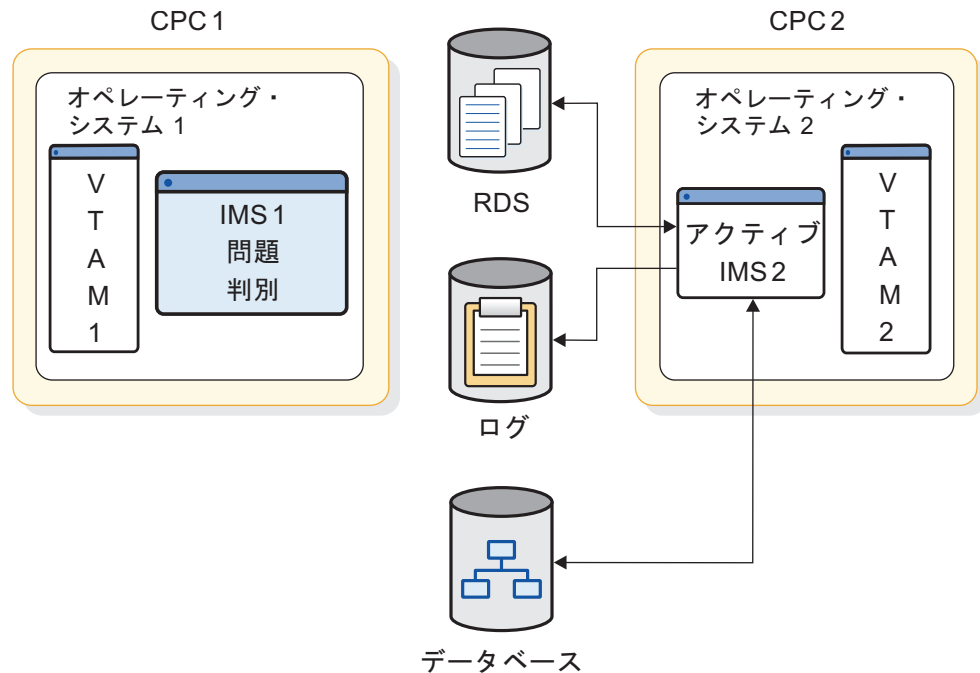


図 78. XRF テークオーバーの後処理フェーズ

テークオーバーの後処理フェーズ中は、新しいアクティブ IMS には、代替環境はありません。新しいアクティブ IMS には、XRF の利点を使用せずにエンド・ユーザーにサービスを提供する責任があります。

障害が発生した IMS システムは、問題判別アクティビティを実行します。これらが完了すると、オペレーターはその障害が発生した IMS を新しい代替 IMS として立ち上げます。

問題判別アクティビティに、IMS データ域の長時間にわたるダンプが必要な場合は、別のシステムを XRF 複合システム内の新しい代替 IMS として立ち上げる場合があります。

オペレーターが新しい代替 IMS、つまり、新しい XRF 複合システムを確立すると、テークオーバーの後処理フェーズは終了します。

障害が発生したアクティブ IMS を新しいアクティブ IMS として復帰させる

複合システムをこの配置のままにしておかないで、元に復帰させたい場合があります。例えば、障害が発生した CPC の方が、もう一方の CPC よりも、実行速度または記憶容量が優れている場合は、できるだけ迅速にそれをアクティブ IMS として復元しなければならないことがあります。障害が発生したアクティブ IMS を新しいアクティブ IMS として再初期設定することはできません。復帰は、段階的に行わなければならないなりません。まず、代替 IMS として立ち上げてから、テークオーバーを手動で要求します。/SWITCH SYSTEM コマンドによって開始される、このような作業負荷のシフトは、予定テークオーバーのもう一つの例です。

XRF IMS が、そのデータベースを他の IMS システムと共用すると、テークオーバーは、IMS オペレーターによる作業が必要な複雑なプロセスになります。オペレーターの手順については、783 ページの『テークオーバー時の IRLM プロセス』で説明しています。

別のシステムを代替 IMS として立ち上げる

新しいアクティブ IMS が代替 IMS のない状態にいる時間を制限するために、オペレーターは代替 IMS として別のシステムを立ち上げることができます。入出力防止が完了するとすぐに、新しい代替 IMS を初期設定することができます。新しいアクティブ IMS はメッセージ DFS994I を発行して、最初のチェックポイントを取得したことを知らせます。このとき、その別のシステムのオペレーターは、/ERESTART BACKUP コマンドを実行して、代替 IMS を始動することができます。

管理者は、DASD および通信コントローラーも含め、すべての装置へのパスが存在することを確認する責任があります。

XRF プロセスの終了フェーズ

サービスからアクティブ IMS を削除するという意味で、テークオーバーは終了と言えますが、また一方で、オペレーターがどちらか一方の IMS システムまたは両方の IMS システムを停止すると、別個の XRF フェーズとしての終了が発生します。

以下の図に示されている終了フェーズは、2 つの別々の独立した環境に XRF 複合システムを戻し、代替 IMS におけるすべてのアクティビティを停止します。

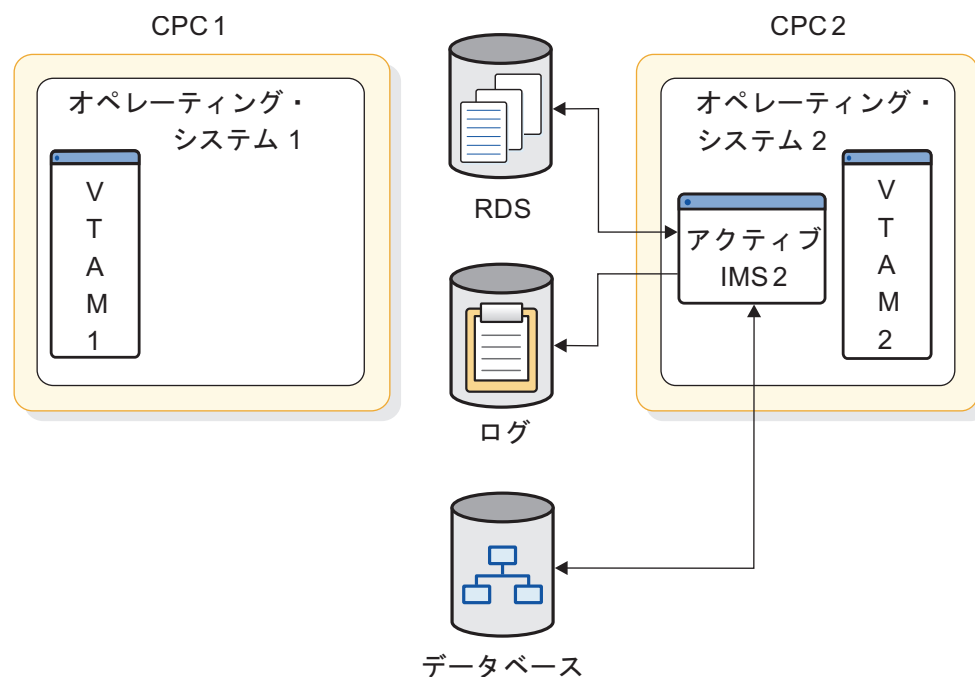


図 79. XRF 終了フェーズ

アクティブ IMS の 終了のタイプには次の 3 つがあります。

通常シャットダウン

XRF 複合システムの通常シャットダウン (/CHECKPOINT FREEZE) で、IMS チェックポイント処理が発生します。これが完了すると、IMS は X'06' ログ・レコードを書き込んで、代替 IMS は自動的に終了します。アクティブ IMS は、プロセスが入出力防止によって終了していることを AVM に通知します。

予定テークオーバー

アクティブ IMS に対して SWITCH SYSTEM コマンドが入力されると、従属領域の処理の静止がアクティブ IMS で実行され、代替サブシステムが処理をテークオーバーすることを示す X'06' ログ・レコードが書き込まれます。アクティブ IMS は、通常のテークオーバーが発生している AVM と通信します。

異常終了

アクティブ・サブシステムの異常終了では、ログ・バッファを削除して、X'06' ログ・レコードを書き込もうとします。アクティブ IMS が、可用性マネージャーから異常終了します。

代替 IMS の終了のタイプには次の 2 つがあります。

通常シャットダウン

オペレーターは、代替 IMS に対して /STOP BACKUP を入力するか、あるいは、アクティブ IMS に対して /CHECKPOINT FREEZE または /CHECKPOINT DUMPQ を入力することにより、代替 IMS を停止できます。どちらの場合にも、代替 IMS は AVM をそのままにして、入出力防止を指定します。

異常終了

IMS ESTAE の終了が入力され、AVM から代替サブシステムが異常終了します。

XRF フェーズのサイクル

XRF の 6 つのフェーズが 1 つのサイクルを形成しています。オペレーターが新しい代替 IMS を立ち上げると、初期設定フェーズから、複合システム内の 2 つのシステムの役割は全く逆になります。

XRF 複合システムの編成

XRF 複合システムは以下のように複数の方法で編成できます。方法ごとに利点と欠点があります。例えば、2 つの IMS システムが定義されている 1 つの CPC では、XRF 複合システムは IMS を IMS 障害から保護しますが、ご使用のシステムを z/OS 障害やハードウェア障害から保護しません。

1 つの CPC を持つ 1 つの XRF 複合システム

2 つの IMS が定義されている 1 つの CPC に、XRF 複合システムを 1 つ設定できます。しかし、XRF IMS の利点がフルに発揮されるのは、z/OS、VTAM、および CPC だけではなく、IMS も複製される場合だけです。CPC が 1 つしか使用されない場合は、IMS だけが代替を持ちます。z/OS、VTAM および CPC の障害が、両方の IMS システムに影響します。

これは、次のものから構成されます。

- z/OS および VTAM を実行する 1 つの CPC
- 2 つの IMS システム (アクティブ IMS システムと代替 IMS システム)
- アクティブ IMS システムと代替 IMS システムで共用されている IMS システム・ログ
- アクティブ IMS システムと代替 IMS システムで共用されているデータベース
- SNA プロトコルを使用し、いずれかの IMS と通信できるリモート端末
- XRF 複合システムが USERVAR を使用する場合、SNA 端末が接続されている個々の 37x5 通信コントローラーで動作する NCP

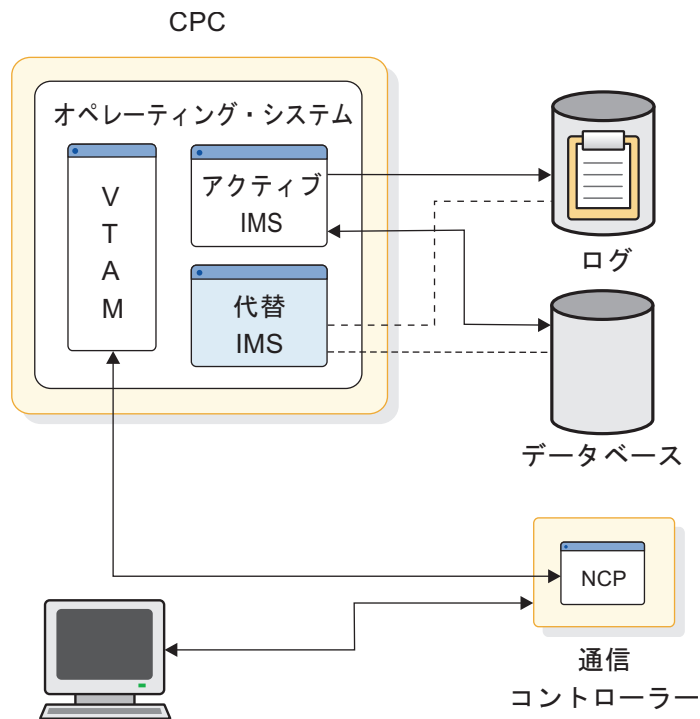


図 80. 1 つの CPC の 1 つの XRF 複合システム

この配置は、テスト環境としてしかお勧めしません。また、CPC の処理速度が十分にあり、XRF のサポートに十分な実記憶域および仮想記憶域がシステムにある場合のみお勧めします。

1 つの CPC XRF 複合システムでの処理

1 つの CPC で実行される XRF 複合システムでは、XRF プロセスは 2 つの CPC で作動する XRF とは若干異なります。1 つの重要な相違点は、テークオーバー中の AVM の役割です。CPC が 2 つある XRF 複合システムでは、代替 IMS が IMS ログに RESERVE コマンドを実行して、アクティブ IMS がデータベースにアクセスするのを防止します。CPC が 1 つの複合システムでは、アクティブ IMS によるデータベースの変更は、このアクションでは防止されません。入出力防止によって、データベース・データ・セットへの未解決の入出力要求だけでなく、システム・ログへの未解決の入出力要求も静止されます。入出力防止が完了すると、オペレーターは AVM006A に「GO」と応答して、テークオーバーを完了すること

ができます。AVM が入出力防止を実行できない場合は、代替 IMS は、障害が発生しているアクティブ IMS の終了を待ってから、テークオーバーを完了します。

CPC が 1 つの複合システムでの計画に関する考慮事項

CPC が 1 つの XRF 複合システムでは、2 つの z/OS オペレーティング・システムが複合システムにあるかのように、AVM からのメッセージがすべて、1 つの z/OS コンソールに表示されます。

テークオーバー時に AVM が入出力防止の実行に失敗すると、IMS オペレーターは CPC をリセットして、アクティブ IMS がデータベースへ書き込むのを防止することはできません。このアクションを行うと、XRF 複合システムが終了します。CPC が 1 つの複合システムでは、オペレーターは入出力防止が完了していると思わず前に、IMS を終了させ、アクティブ IMS のすべてのアドレス・スペースが終了していることを確認しなければなりません。

2 つの CPC を持つ 1 つの XRF 複合システム

以下に図示してあるこの構成の XRF 複合システムが、最も一般的な構成です。

この構成では、以下のコンポーネントが必要です。

- 2 つの CPC、2 つのオペレーティング・システム、および 2 つの VTAM
- 1 つのアクティブ・システムと 1 つの代替システムという 2 つの IMS システム
- アクティブ IMS システムと代替 IMS システムで共用されているデータベース
- アクティブ IMS システムと代替 IMS システムで共用されている IMS システム・ログ
- SNA プロトコルを使用し、いずれかの IMS と通信できるリモート端末
- XRF 複合システムが USERVAR を使用する場合、SNA 端末が接続されている個々の 37x5 通信コントローラーで動作する NCP

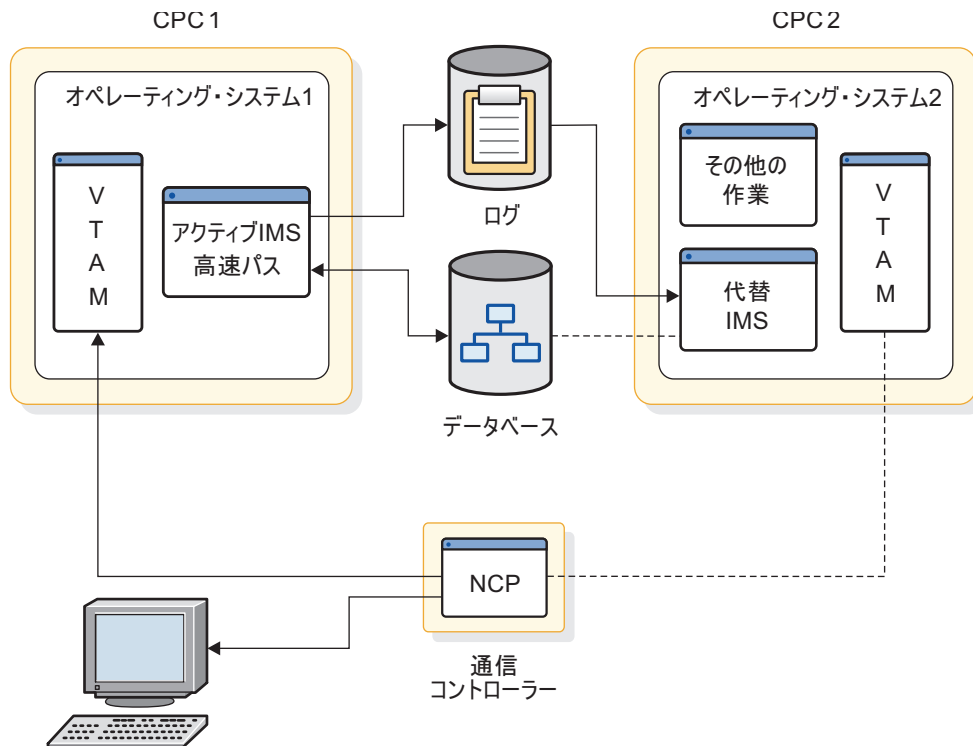


図 81. 2 つの CPC の 1 つの XRF 複合システム

2 つの CPC を持つ 1 つの XRF 複合システムと XRF のない IMS

この構成における XRF 複合システムは、以下の図で示すように、データベースを共用するアクティブ IMS システムと代替 IMS システムの両方を必要とします。

以下の図に示すように、この構成には次の項目が必要です。

- 3 つの CPC、3 つの z/OS、および 3 つの VTAM (CPC、z/OS、および VTAM の各 2 つのみが XRF 内にある)
- 3 つの IMS システム。2 つはアクティブ IMS で、もう 1 つは代替 IMS
- アクティブ IMS システムと代替 IMS システムの両方で共用されているデータベース
- 個々のアクティブ IMS システムのための IMS システム・ログと、代替 IMS との 1 つの共用システム・ログ
- SNA プロトコルを使用し、どちらか一方または両方のアクティブ IMS システムと通信するリモート端末
- XRF 複合システムが USERVAR を使用する場合、CPC に接続されている 37x5 通信コントローラーで動作する NCP

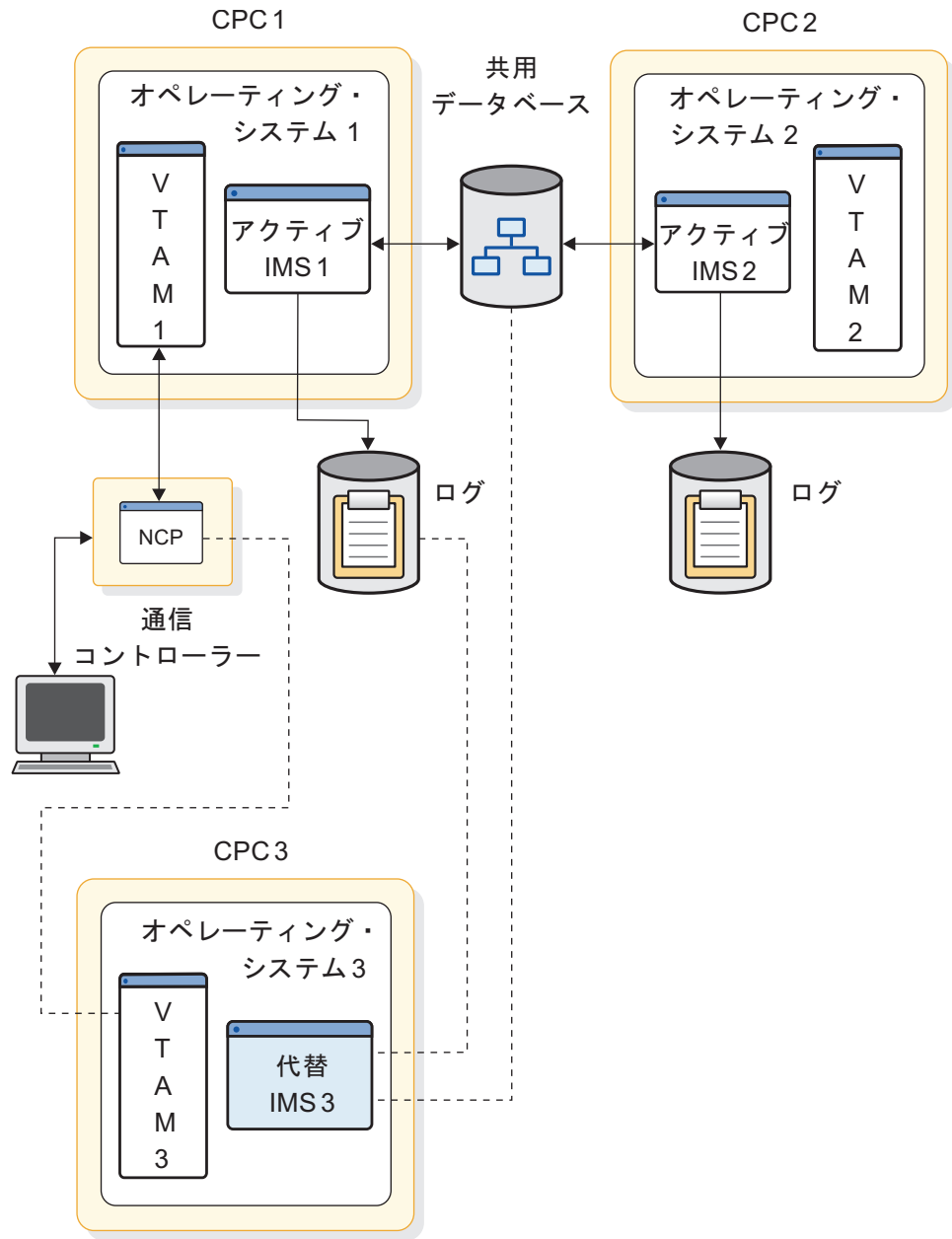


図 82. 2 つの CPC 内の 1 つの XRF 複合システムと XRF のない IMS

3 つの CPC を持つ 2 つの XRF 複合システム

2 つの代替サブシステムを動作させる CPC で十分な実記憶域と仮想記憶域が使用できる場合は、以下の図に示すように 2 つの XRF 複合システムを 3 つの CPC で作動させることができます。

この構成には、4 つの CPC の 2 つの XRF 複合システムよりも有利な点があります。その利点とは、代替 IMS がどちらも単一の PC で動作するので、VTAM 1 つ、z/OS 1 つ、CPC 1 つが必要なくなることです。これは、次のものから構成されます。

- 3 つの CPC、3 つの z/OS、および 3 つの VTAM
- 2 つのアクティブ IMS システムと 2 つの代替 IMS システムから構成される 4 つの IMS システム
- 個々のアクティブ・サブシステムのためのものであり、その代替 IMS によって共有されている IMS システム・ログ
- 個々の複合システムのアクティブ IMS と代替 IMS で共有されているデータベース
- SNA プロトコルを使用し、どちらか一方または両方のアクティブ IMS システムと通信するリモート端末
- XRF 複合システムが USERVAR を使用する場合は、CPC に接続されている各 37x5 通信コントローラーで動作する NCP

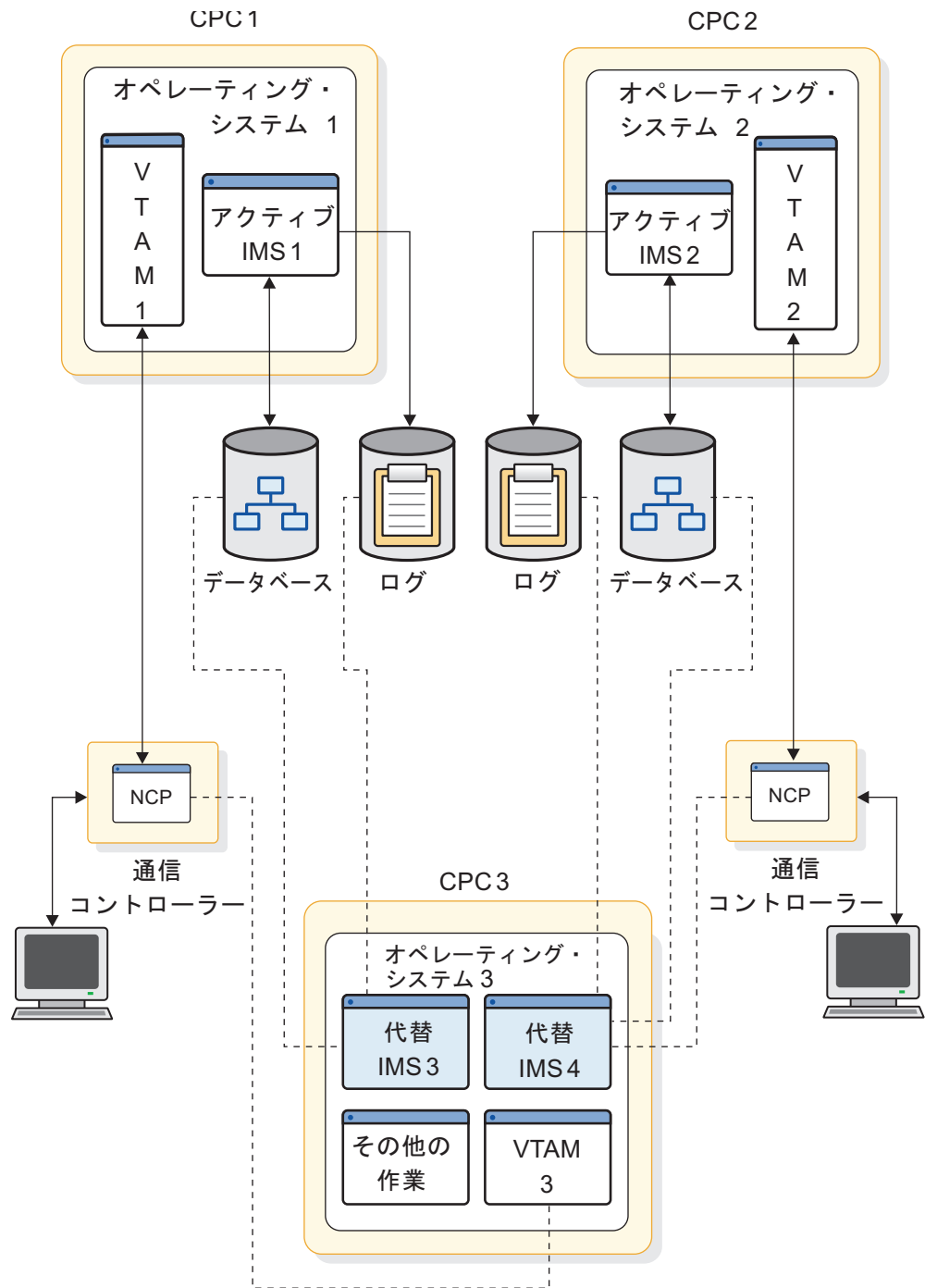


図 83. 3 つの CPC の 2 つの XRF 複合システム

個々のサブシステムのユーザーに対して、アクティブ・サブシステムと代替サブシステムの両方へのパスが存在しなければなりません。個々のデータベースは、関連付けられているアクティブ CPC と代替 CPC への共用アクセス（テークオーバー時のオンライン・アクセスの場合でも、切り替え可能アクセスの場合でも）を持っていなければなりません。

4 つの CPC を持つ 2 つの XRF 複合システム

データ共有を行っている 2 つの XRF 複合システムは、すべての CPC に IRLM がなければなりません。

これは、次のものから構成されます。

- 4 つの CPC、4 つの z/OS、4 つの VTAM、および 4 つの IRLM
- それぞれが優先 CPC (IMS1 と IMS2) にある 2 つの IMS アクティブ・サブシステム
- それぞれが優先 CPC (IMS3 と IMS4) にある 2 つの IMS 代替サブシステム
- アクティブ・サブシステムと代替サブシステムで共有されている 2 つの IMS システム・ログ
- アクティブ・サブシステムと代替サブシステムで共有されているデータベース
- SNA プロトコルを使用し、どちらか一方または両方のアクティブ IMS システムと通信するリモート端末
- XRF 複合システムが USERVAR を使用する場合は、CPC に接続されている各 37x5 通信コントローラーで動作する NCP

以下の図は、は、4 つの CPC の 2 つの XRF 複合システムを示しています。

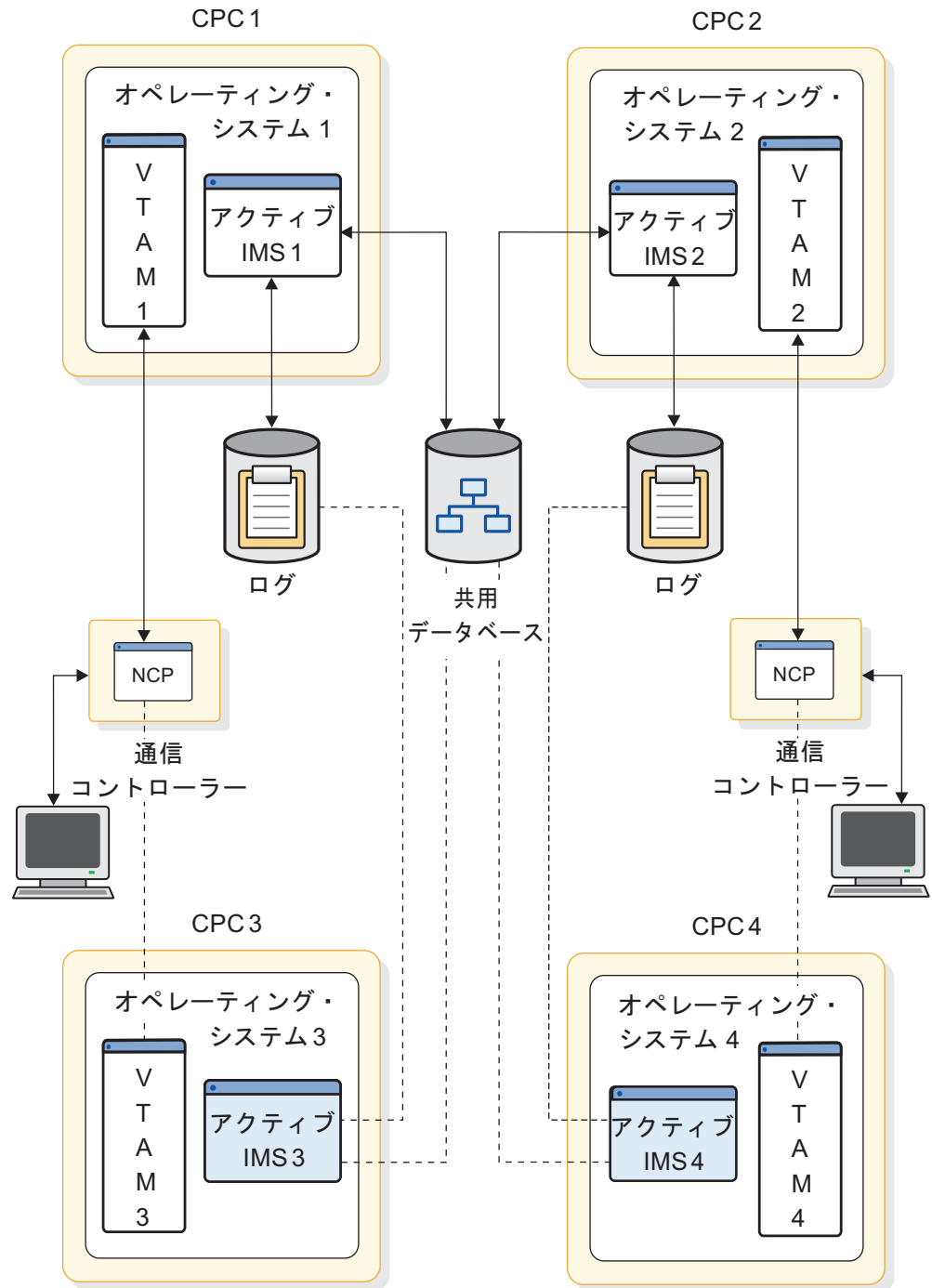


図 84. 4 つの CPC の 2 つの XRF 複合システム

共通アクセス・パスが、RECON データ・セット用のすべての IMS システムとすべての共用データベースの所定の位置にあります。

XRF 複合システムの計画

テークオーバーを強制する条件、オペレーターの介入の程度、テークオーバー時にどの端末セッションが自動的にリカバリーするか、アクティブ IMS の問題を認識するために代替 IMS がどのような方法を使用するかなどのテークオーバーに関する計画決定を行わなければなりません。

XRF の制限事項

IMS インストール・システムに障害が発生すると必ず、テークオーバーが実行されるわけではありません。

例えば、XRF は以下の障害によって発生した停止には対応しません。

- CPC と通信コントローラーとの間の通信を中断させるチャンネル障害またはリンク障害
- 制御装置、NCP、回線、端末などの通信ネットワークの障害
- システム間障害、例えば、JES または CTC によって発生した障害
- アプリケーション・プログラムの何らかの障害
- IMS データベースまたはログの消失または損傷
- 複合システム内の両方の CPC に影響を及ぼす電源障害
- オペレーターの何らかのミス

関連資料: XRF 複合システム内の z/OS 再始動マネージャー (ARM) については、812 ページの『z/OS 計画に関する考慮事項』を参照してください。

代替 IMS での非 XRF 作業負荷

テークオーバーを処理することにより、代替 IMS の作業が著しく増加します。インストール・システムでは、代替 IMS の非 XRF 作業負荷のプロセスをスローダウンするか、または停止させる準備をしなければなりません。作業をスワップできる場合は、z/OS は作業を一時的にスワップアウトします。z/OS が作業負荷をスワップインすると、IMS 作業負荷がどの程度 CPC を必要としているかによって、作業負荷は低下モードで動作します。

代替 IMS のジョブを取り消す場合は、テークオーバー完了後に、一度に 1 つのジョブに対して CANCEL コマンドを実行します。CANCEL コマンドのプロセスの方がテークオーバーのプロセスよりも優先されるため、テークオーバーが完了するのを待たずにこのコマンドを発行すると、テークオーバーのプロセスがスローダウンします。

代替 IMS に残る非 XRF の作業は、以下のように、リソースに関して XRF 関係の作業と競合します。

- テークオーバー時に、スワップできない作業がテークオーバー処理と競合します。システム・リソース・マネージャー (SRM) は、代替 IMS に対し、アクティブ IMS から作業負荷をテークオーバーできるようにするためにリソースを提供しようとしています。SRM は、代替 IMS システムでのストレージに関するニーズの分析を高速化します。テークオーバーによって、スワップできない作業に対して高いページング・アクティビティが生じますが、応答時間は長くなります。ただし、プロセスは継続します。

- テークオーバー後、スワップできない作業は、IMS 対話型作業負荷と競合します。非 XRF ユーザーの場合の応答時間は、インストール・システム・パフォーマンス仕様 (IPS) の設定をどのようにコーディングしたか、そして、スワップできない作業がどの程度の実記憶域を必要とするかによって決まります。

代替 IMS 上で TSO を実行する場合は、IMS よりも優先順位を低くしなければなりません。実記憶域が必要な場合、z/OS はテークオーバー時に TSO ユーザー・アドレス・スペースをスワップアウトします。これらのアドレス・スペースは、新しいアクティブ IMS が安定するまで、スワップアウトされたままです。その期間は、インストール用プロファイルによって異なります。IMS が優先するように、代替 IMS には、アクティブ IMS の場合と同じ IPS 設定値と SRM パラメーターを指定しなければなりません。テークオーバーが発生すると、IMS 作業負荷が優先されます。

システム間連絡リンクの使用

XRF は、アクティブ IMS と代替 IMS 間のシステム間連絡リンク (ISC) セッションを効果的に活用できます。

このセッションが有利な理由は次のとおりです。

- 監視メカニズムとして、アクティブ IMS の障害を代替 IMS に警告します。
- 同期化フェーズ中に、オペレーターの介入なしに、アクティブ IMS が SNAPQ チェックポイントを取得できます。

ISC リンクは通常の VTAM APPL 間セッションであり、以下のいずれかの経路を指定できます。

1. チャネル間プロトコル (CTC または 3088)
2. 両方のシステムへのチャネル・パスを持つ 37x5 通信コントローラー
3. テレプロセッシング (TP) 回線による別個の 37x5 通信コントローラー

以下の図は、3 種類の ISC リンクを示しています。番号は、前述のリストの番号に対応しています。一般に、経路上にあるハードウェア装置が多くなればなるほど、通信速度が遅くなり、障害が発生する可能性が高くなります。監視および他の通信目的で選択できる経路の中で最も信頼性の高い経路を選択してください。

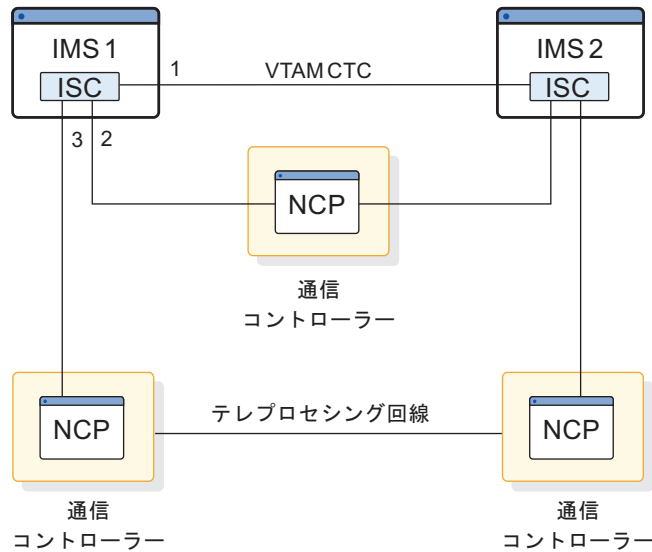


図 85. ISC リンク・オプション

XRF 複合システムの端末

XRF は、現在 IMS ユーザーに対応しているすべての端末をサポートします。このサポートは、簡単に得ることができます。XRF IMS では、4 つまたは 5 つのシステム定義マクロ・ステートメントまたは ETO ログオン記述子でキーワードを指定する必要があります。

端末の優先順位をデフォルトから切り替えたくない場合、または、端末タイプ・セッションを変更したくない場合は、システム定義マクロを変更する必要はありません。

XRF IMS は、インストール・システムの端末に、その端末の特性で可能な最良のサービスを提供します。このトピックでは、それらの特性について説明します。

特性の説明で、「端末」という用語は、プログラマブル・コントローラーおよびそれに接続されているオペレーター端末、プリンター、補助制御装置などを含む端末システムのコンポーネントを示すのに使用されています。また、リモート・サブシステムと MSC リンクも示しています。

クラス 1 端末

XRF を十分に活用するためには、複合システムのほとんどの端末をクラス 1 端末にする必要があります。クラス 1 端末とは、次のような端末です。

- SNA プロトコルを使用する。
- XRF をサポートする VTAM および NCP で制御される。
- XRF 複合システムが USERVAR を使用する場合は、37x5 通信コントローラーに接続される
- IMS システム定義で TYPE マクロの UNITYPE キーワードに、以下のいずれかとして定義できる。
 - SLUTYPE1

- SLUTYPE2
- SLUTYPEP
- FINANCE

BACKUP= キーワードを使用して、クラス 1 として端末を定義します。
 BACKUP= キーワードは、TERMINAL マクロ、他のシステム定義マクロ、および ETO ログオン記述子にあります。

クラス 1 端末は、BACKUP=(n, YES) で定義されます。ここで、n は 1 から 7 までの優先順位設定です。優先順位設定は、IMS に端末を再接続するよう指示し、IMS が再接続するときの順序を示します。YES パラメーターは、端末上の作業が中断されないように端末上のデータ・トラフィックを追跡するよう VTAM に指示します。これは、USERVAR のみを使用する XRF 複合システムは、代替 IMS でバックアップ・セッションを作成するためです。

MNPS を使用する XRF 複合システムでテークオーバーが発生した場合、代替 IMS システムはクラス 1 端末上のすべてのデータ・トラフィックを一時停止し、代替 IMS システムが固有の MNPS ACB をオープンするまで、クラス 1 端末の持続セッションを維持します。代替 IMS は、次に、その新しい MNPS ACB を使用してクラス 1 端末を再接続します。テークオーバーの前に進行中だったデータ・トラフィックはすべて再開されます。アクティブ IMS の MNPS ACB をクローズしないと、代替 IMS 上で新しいものをオープンすることはできません。

USERVAR を使用する XRF 複合システムでは、クラス 1 端末は、代替 IMS システム上にバックアップ・セッションを持っています。これらのセッションは、アクティブ IMS のセッションが開始されると、トラッキング・フェーズの過程で確立されます。端末は、セッションを失うことなく、テークオーバー後に自動的に切り替えられます。

端末に対して高い優先順位を設定しても、テークオーバー時にセッション・リカバリーが先に行われるとは限りません。リソースの競合および回線の伝送速度のために、リカバリー時間は予測不能な場合があるためです。

一般に、テークオーバー時の透過性の度合いは、以下によって決まります。

- 端末のタイプ
- テークオーバー時の端末のアクティビティー
- 端末に割り当てられているセッション・リカバリー優先順位

例えば、テークオーバー時に進行中の出力または入力がなく、BACKUP=7 優先順位を持つ FINANCE 端末が、最も透過性の高いリカバリーを持っています。

ユーザーから見ると、MNPS を使用するクラス 1 端末セッションは、テークオーバー中に影響を受けずにいますが、XRF から見ると、クラス 1 端末セッションは、テークオーバーの開始時に、障害のある IMS に接続されます。いったんテークオーバーが始まると、VTAM はクラス 1 端末セッションを持続セッションとしてオープンしたままに保持し、元の MNPS ACB はクローズし、代替 IMS は独自の MNPS ACB を開きます。すると、VTAM はクラス 1 端末セッションをその新しい MNPS ACB に再接続します。VTAM は、クラス 1 端末のセッション状態とデータ・トラフィックを絶えずトラッキングします。テークオーバー処理中、

VTAM は、テークオーバー前にクラス 1 端末との間で転送中だった情報を一時停止します。テークオーバーが完了すると、VTAM は、情報の転送を再開します。

(USERVAR を使用する端末の) 端末ユーザーから見ると、IMS とのセッションが 1 つだけ存在しています。NCP から見ると、2 つのホスト・セッションが存在しています。実際のメッセージは、論理装置 (LU) とアクティブ IMS との間で受け渡しされます。代替 IMS は、IMS ログをモニターすることにより、メッセージ・アクティビティの状況を追跡します。アクティブ IMS がクラス 1 端末との端末セッションを確立または終了すると、代替 IMS はバックアップ・セッションを確立または終了します。アクティブ IMS で障害が発生すると、代替 IMS は、事前に確立されているバックアップ・セッションのモードを BACKUP から ACTIVE に変更することにより、LU の観点からは制御を失うことなく端末セッションを引き継ぎます。セッションが切り替えられると、NCP は代替 IMS に端末の状況のビューを送信します。IMS は、これを自身の状況のレコードと比較して、どのようなリカバリー・アクション (ある場合) をとるべきかを判断します。REVERIFY オペランドを RACF とともに使用する場合は、ユーザーは再びサインオンしなければなりません。

クラス 2 端末

クラス 2 端末は、代替 IMS でのバックアップ・セッション・サポートを提供しませんが、テークオーバー後に自動的に再始動されます。代替 IMS でのセッションの再確立後のリカバリー手順を含む端末とのインターフェースは、XRF 複合システムが MNPS を使用する場合でも、USERVAR を使用する場合でも、同じです。

BACKUP= キーワードを使用して、クラス 2 として端末を定義します。

BACKUP= キーワードは、TERMINAL マクロ、他のシステム定義マクロ、および ETO ログオン記述子にあります。

クラス 2 端末は、BACKUP=(*n*, NO) で定義されます。ここで、*n* は 1 から 7 までの優先順位設定です。優先順位設定は、IMS に、テークオーバー後に端末を再接続するよう指示し、IMS が再接続するときの順序を示します。NO パラメーターは、VTAM が端末上のデータ・トラフィックを追跡しないことを指定します。

アクティブ IMS でクラス 2 端末セッションが確立されると、代替 IMS はセッションの開始または終了をログ・レコードを使用して追跡します。アクティブ IMS が異常終了した時、代替 IMS は障害が起こる前にアクティブであったネットワーク・リソースで新規セッションの設定を試みます。

クラス 2 端末は、XRF から十分なサポートを受けます。実際、テークオーバー時に、IMS はすべてのクラス 1 端末セッションが切り替えられる前に、一部のクラス 2 端末でサービスを再確立できる場合があります。

IMS が使用可能なパスがない端末とのセッションを再確立しようとする時、失敗します。まず、オペレーターが新しいアクティブ IMS に端末を切り替えなければ、IMS はローカル接続端末でセッションを再確立することができません。この種の端末が複合システムに多数ある場合は、テークオーバーが実際に進行する前に、テークオーバー時の制御をオペレーターに与えることを考慮することができます。このときに、オペレーターは切り替えを行って、これらの端末のセッションをより迅速

にリカバリーできるようにすることができます。管理者は、IMS.PROCLIB のメンバー DFSHSBxx の AUTO パラメーターを使用して、この制御をオペレーターに与えます。

IMS はクラス 2 端末に対する新しいセッションを確立しようとしませんが、端末の所有者または所有者とのリンクに障害が起これば、新しいセッションを確立する試みも失敗します。

クラス 2 の資格を有する端末は、次のとおりです。

- VTAM を介して IMS XRF と通信する複数システム結合機能 (MSC) および ISC のサブシステム。
- 両方の CPC にローカル接続されている共用 DASD のスプール回線グループ
- VTAM で管理されている非 SNA 3270 端末
- ネットワーク端末オプション (NTO) ライセンス・プログラムで制御されている装置
- ローカル接続装置
- SNA プロトコルを使用しない端末
- 以下を除く、クラス 1 端末の資格を有する端末
 - 37x5 通信コントローラーで制御されていない端末
 - XRF をサポートしているか、または、システム定義マクロまたは ETO ログオン記述子に BACKUP=(n,NO) が定義されている NCP または VTAM で制御されていない端末

XRF 複合システムのプログラマブル・コントローラーと通信するアプリケーション端末がある場合は、通信部分にコードを追加して、使用可能な最高レベルの透過性を達成する必要があります。また、ISC 端末に対しては、ISC 端末を XRF IMS に接続するリンクに対してセッションを定義しなければなりません。

関連資料 プログラムが追加する必要のあるコードについては、「IMS V15 コミュニケーションおよびコネクション」を参照してください。

クラス 3 端末

クラス 3 端末は、代替 IMS にバックアップ・セッション・サポートを持っていません。これらの端末セッションは、MTO またはユーザー・ログオンによって、テークオーバー後に新しいアクティブ IMS で手動で再始動しなければなりません。

クラス 3 に適格な端末には、クラス 3 のサービスに適格なすべての端末 (すべての LU 6.2 装置など) と、切り替えを行わないように指示されているクラス 1 およびクラス 2 端末があります。これは、システム定義マクロまたは ETO ログオン記述子に BACKUP=NO を指定することによって行われます。

クラス 3 端末は、アクティブ IMS システムと代替 IMS システムの両方に物理的に接続されている必要があります。この端末が両方の IMS システムに物理的に接続されていない場合、セッションの再確立はできません。

端末サポートの指定

IMS がサポートする端末はすべて、XRF 複合システムで使用できます。端末特性は、XRF 複合システムで端末が持っているサポートのタイプを、クラス 1 が最高のサポートを持ち、クラス 3 は何もサポートしないと判断します。

LU 6.2 装置は、常にクラス 3 です。ETO 端末およびユーザーは、どのようなレベルのサポートにも適格です。

関連資料: XRF における LU 6.2 および ETO については、「IMS V15 コミュニケーションおよびコネクション」の『端末サポートの指定』を参照してください。

以下の表に、端末クラスに対する各種装置の適格性を要約しています。

表 53. 各種装置のクラス適格性

装置	クラス 1 に適格か	クラス 2 に適格か	クラス 3 に適格か
SLUTYPE1 (3286,...)	あり	あり	あり
SLUTYPE2 (3278,3279,...)	あり	あり	あり
SLUTYPEP (4700,8100,SERIES 1,...)	あり	あり	あり
FINANCE (3601)	あり	あり	あり
LUTYPE6 (ISC)	なし	あり	あり
MSC (VTAM)	なし	あり	あり
MSC (TCP/IP)	なし	あり	あり
NTO (専用回線)	なし	あり	あり
3270 (VTAM)	なし	あり	あり
スプール回線グループ	なし	あり	あり
ローカル接続装置	なし	あり	あり

表にリストされている優先順位または端末タイプを変更するには、TERMINAL マクロまたは ETO ログオン記述子に以下の BACKUP キーワードのいずれかを指定します。

- クラス 1、クラス 2、クラス 3 のいずれかのサービスに適格な装置の場合は、次のように指定します。
 - BACKUP=(1-7, YES): クラス 1 サービスの場合
 - BACKUP=(1-7, NO): クラス 2 サービスの場合
 - BACKUP=NO: クラス 3 サービスの場合
- クラス 2 または 3 のサービスにのみ適格な装置の場合は、次のように指定します。
 - BACKUP=(1-7, NO): クラス 2 サービスの場合
 - BACKUP=NO: クラス 3 サービスの場合

システム定義ステートメントのいくつかのマクロでの BACKUP= キーワードは、端末が受けるサポートを指定します。インストール・システムでは、確立されたバックアップ・セッションと端末の切り替え優先順位を持っている端末を決定します。レベル 1 は優先順位が最低で、レベル 7 は最高です。VTAM 要求は IMS によって優先順位が決定されますが、ペーシング、回線速度、輻輳 (ふくそう) などのネットワークの予測不能な要因のために、任意の順序で完了する場合があります。

BACKUP=(1-7,YES) で定義された端末は、障害時にセッション制御や未完了データの喪失なしで自動的に引き継がれます。これは、クラス 1 サポートになります。

USERVAR を使用する XRF 複合システムでは、端末バックアップ・セッションは、BACKUP=(1-7,YES) で定義された端末のトラッキング・フェーズ中に確立されます。代替 IMS システムで確立できる端末バックアップ・セッションの最大数は、NCP BUILD ステートメントに BACKUP=n オプションを使用して指定されます。この場合、n は端末バックアップ・セッションの最大数です。IMS は、端末バックアップ・セッションの最大数に達すると、追加のクラス 1 端末ユーザーへのログオンを拒否します。代替 IMS のバックアップ・セッションの最大数を慎重に決定し、適切な数を指定してください。

BACKUP=(1-7,NO) の場合は、この端末はテークオーバー後に自動的に再始動されますが、バックアップ・セッションは確立されません。これは、クラス 2 サポートを強制します。

クラス 1 とクラス 2 の両方の端末では、デフォルトの切り替え優先順位は 4 です。

BACKUP=NO の場合、この端末は、テークオーバー後に手動で再始動しなければなりません。これは、クラス 3 サポートを強制します。

テークオーバーが端末ユーザーに与える影響

ユーザーがテークオーバーを認識するかどうかは、端末のクラス、IMS がセッション・リカバリーを要求する優先順位、端末を制御するアクセス方式、障害発生時の入力メッセージまたは出力メッセージの状況などによって異なります。主要なネットワークの目的は、できるだけ早く端末のサービスをリカバリーすることです。

クラス 1 およびクラス 2 の端末のセッションがリカバリーしたとき、ユーザーは動作の一時停止に気付く場合があります。クラス 1 端末のユーザーがメッセージを入力したばかりの場合、代替 IMS は、ユーザーがメッセージを再入力するか、またはメッセージ DFS3861I を再発行することを要求します。どちらの場合にも、ユーザーは画面を消去して、最後のメッセージを再入力しなければなりません。

関連資料 FINANCE 端末および SLUTYPEP 端末のユーザーがこのメッセージを受け取ったことを確認する方法については、「IMS V15 コミュニケーションおよびコネクション」を参照してください。

IMS は、セッションが再接続されたことを示すメッセージ DFS2469W、DFS3649、または DFS3650 を発行して、セッションがリカバリーされたほとんどのクラス 2 端末ユーザーに通知します。FINANCE 端末、SLUTYPEP 端末、および ISC 端末は、これらのメッセージを受け取りません。

関連資料 これらのメッセージの詳細については、IMS V15 メッセージおよびコード 第 1 巻: DFS メッセージ を参照してください。

IMS オペレーターは、テークオーバー時に障害が発生した端末でサービスを手動でリカバリーする必要があります。

クラス 1 端末ユーザー (SLUTYPE2 ユーザーを除く) の場合のテークオーバー

ユーザーがトランザクションとトランザクションの間で、最後のデータ入力に対する応答を受け取り、次のトランザクションをまだ開始していない場合は、テークオーバーが発生したことで、端末が切り替えられたことを示すものは何もありません。入力または出力中、あるいは、トランザクションが経過中のときに障害が発生すると、ユーザーはテークオーバーが発生したことに気付きます。

入力

Queued Recoverable

リカバリー可能トランザクションの入力メッセージは、キューに入れられると、テークオーバー後に自動的にスケジュールされ、ユーザーは端末の切り替えに気付きません。

Not Yet Queued Recoverable

リカバリー可能トランザクションの入力メッセージが障害発生時にキューに入れられないと、このメッセージは失われます。これには、同期点に達していない高速機能トランザクションが含まれます。

会話型

トランザクションが会話型であった場合は、代替 IMS は例外応答 (SSENSE=X'0826', USENSE=X'0F15') メッセージを送信します。端末がシステム・メッセージを受信できる場合は、DFS3861I も送信されます。RTO は会話のために /HOLD コマンドおよび /RELEASE コマンドを実行でき、最終出力の最初のページを再び受信します。

非会話型

トランザクションが会話型ではなく、最後の出力メッセージが MFS でなかった場合は、例外応答とシステム・メッセージが送信され (上記の会話型を参照)、RTO は新しいトランザクションで再開します。

Queued Nonrecoverable

トランザクションがリカバリー不能で、キューに入れられている場合は、このメッセージは失われます。

Not Yet Queued Nonrecoverable

トランザクションがリカバリー不能で、キューに入れられていない場合は、このメッセージは失われます。代替 IMS は例外応答 (SSENSE=X'0826', USENSE=X'0F15') とメッセージ DFS3861I を送信します。

経過中

Recoverable

障害発生時にトランザクションが処理されている場合は、IMS は更新をバックアウトして、トランザクションをスケジュール変更し、トランザクション処理完了後に適切な出力メッセージを送信します。ユーザーがテークオーバーに気付かないのは、トランザクションのスケジューリングに遅延がない場合のみです。それ以外の場合は、端末ユーザーは応答時間の遅延を経験します。ユーザーは、データを再入力する必要はありません。

Nonrecoverable

トランザクションがリカバリー不能トランザクションや、同期点に達していない高速機能トランザクションである場合は、このトランザクションは失われます。例外応答 (SENSE=X'0826', USENSE=X'0F15') とメッセージ DFS386I が送信されます。

出力

Recoverable

障害発生時の出力メッセージがリカバリー可能な場合、代替 IMS は、次のセグメントからメッセージの送信を続行し、端末ユーザーはテークオーバーに気付きません。

Nonrecoverable

障害発生時の出力メッセージがリカバリー不能な場合は、代替 IMS は現在のメッセージをリセットし、端末がシステム・メッセージを受信できる場合は、メッセージ DFS386I を送信します。

Unknown Output

代替 IMS が最後の出力メッセージの内容を認識しない場合 (最後の出力がログに記録されないシステム・メッセージであった場合) は、IMS は現在の出力メッセージを取り消し、端末がシステム・メッセージを受信できる場合は、メッセージ DFS386I を送信します。

クラス 1 SLUTYPE2 端末ユーザーの場合のテークオーバー

SLUTYPE2 装置の端末ユーザーは、テークオーバー・アクティビティに必ず気付きます。システム・メッセージを受信するように端末が定義されている場合は、テークオーバー後にメッセージ DFS386I テキストを受け取ります。障害発生時の入出力アクティビティは、常にリセットされます。

入力

Queued Recoverable

リカバリー可能トランザクションの入力メッセージは、キューに入れられている場合は、テークオーバー後に自動的にスケジュールされます。端末が DFS386I メッセージを受信するように定義されている場合は、端末ユーザーは PA1 または PA2 キーを押して、出力を受け取らなければなりません。

Not Queued Recoverable

リカバリー可能トランザクションの入力メッセージが障害発生時に

キューに入れられないと、このメッセージは失われます。これには、同期点に達していない高速機能トランザクションが含まれます。

1. 会話型

トランザクションが会話型の場合は、/HOLD コマンドおよび/RELEASE コマンドを入力することにより、端末ユーザーは最終出力の最初のページを受け取ることができます。

2. 非会話型

トランザクションが非会話型の場合は、端末ユーザーは新しいトランザクションを開始しなければなりません。/FORMAT コマンドが必要です。

Nonrecoverable

障害発生時に、リカバリー不能トランザクションまたは結果の応答が、障害発生時にシステム内にある場合は、そのトランザクションまたはメッセージは失われます。

経過中

Recoverable

障害発生時にトランザクションが処理されていると、IMS は更新をバックアウトして、トランザクションをスケジュール変更します。端末ユーザーは、PA1 または PA2 キーを押して、出力メッセージを受け取らなければなりません。テークオーバーの直後にトランザクションをスケジュールできないと、ユーザーは応答時間の遅延を経験します。ユーザーは、トランザクションを再入力する必要はありません。

Nonrecoverable

トランザクションがリカバリー不能トランザクションや、同期点に達していない高速機能トランザクションである場合は、このトランザクションは失われます。

出力

MFS Logical Paging

端末ユーザーが論理ページによってページングしている場合は、ユーザーは PA1 または PA2 キーを押して、検討されている最後の一連の論理ページの最初の画面を表示しなければなりません。

Single MFS Page or Non-MFS Output-Not Dequeued

IMS が出力を送信したときから、IMS が受信の肯定応答を受け取ったとき (メッセージはデキューされない) までの間に障害が発生した場合は、最終出力が端末に再び送信されます。

Single MFS Page or Non-MFS Output-Dequeued

出力が送信されて、肯定応答が受信された後に障害が発生すると、ユーザーはトランザクションを再び開始しなければなりません。/FORMAT コマンドが必要です。

MFS Multipage Output without Operator Logical Paging-Last Page Send and Not Dequeued

IMS が出力を送信したときから、IMS が肯定応答を受け取ったとき (メッセージはデキューされない) までの間に障害が発生した場合は、最終出力の最初のページが再び送信されます。

Last Page Sent-Dequeued

出力が送信されて、肯定応答が受信された後に障害が発生すると、ユーザーはトランザクションを再び開始しなければなりません。/FORMAT コマンドが必要です。

Last Page Not Sent

障害発生時に最後のページが送信されないと、IMS は出力の最初のページを再び送信します。端末ユーザーは、PA1 キーでページングします。

クラス 2 端末ユーザーの場合のテークオーバー

クラス 2 端末ユーザーには、緊急時再始動後の非 XRF 環境で提供されるサポートと同じサポートが提供されます。これらのサポートの相違点は、ログオン・シーケンスがテークオーバーによって自動的に処理されるという点です。FINANCE 端末、SLU P 端末、および ISC 端末のユーザーは、テークオーバー時に自動的にサインオンされます。その他の ETO ユーザーは、テークオーバー後に再びサインオンが必要な場合があります。

3270 VTAM

テークオーバー後に、IMS は代替 IMS に新しいセッションを確立しようとします。

LUTYPE6 (ISC)

テークオーバー後に、IMS は代替 IMS に新しい ISC セッションを確立しようとします。

MSC (VTAM)

テークオーバー後に、IMS は障害発生時にアクティブだったすべての VTAM 物理リンクを再確立しようとします。

NTO テークオーバー後に、IMS は代替 IMS に新しいセッションを確立しようとします。

ローカル接続装置

これらの装置は、手動で切り替えなければなりません。

LU 6.2 装置

進行中の会話は、再配置しなければなりません。

リモート装置

ツイン接続されている VTAM 端末、または、ローカル間 NCP 接続されている VTAM 端末は、自動的に切り替えられます。端末が新しいアクティブ IMS に物理的に接続されていない場合は、切り替えは失敗します。

所有している VTAM に障害が起こった場合には、オペレーターが端末を新しいアクティブ IMS に切り替えなければなりません。装置の切り替えも、2914/3814 を使用して行えます。

スプール回線グループ

テークオーバー後に、IMS は新しいスプール・データ・セットをオープンします。

端末切り替えに VTAM 所有権が与える影響

端末は、アクティブ CPC、代替 CPC、コミュニケーション管理構成 (CMC) の CPC にある VTAM によって所有できます。XRF 複合システム外のホストに端末を所有させると、最も安定した所有権が得られます。これは、ホストはテークオーバーの影響を受けず、テークオーバーは CMC アクティビティに影響されないからです。

以下の表は、テークオーバー後の端末のセッション状況、ログオン機能、ログオフ機能の概要です。これは、アクティブ IMS の VTAM が機能していないことを前提としています。

表 54. テークオーバー中およびテークオーバー後の所有 VTAM の端末機能：

System	アクティブ			代替			CMC またはその他のホスト		
	1	2	3	1	2	3	1	2	3
セッションの継続	あり	なし	なし	あり	なし	なし	あり	なし	なし
LOGOFF 使用可能	あり	N/A	なし	あり	N/A	なし	あり	N/A	N/A
LOGOFF 使用可能 ¹	あり	あり	あり	あり	あり	あり	N/A	N/A	N/A
LOGON 使用可能	なし	なし	なし	あり	あり	あり	あり	あり	あり
LOGON 使用可能 ¹	あり	あり	あり	あり	あり	あり	N/A	N/A	N/A

注:

1. 別のシステムが端末所有権を獲得した後

クラス 1 端末の VTAM 所有権についての計画

クラス 1 端末の所有権をどこに設定するかを決定するときは、どこに所有権があれば、オペレーターの負担が最小になるか、クラス 1 端末ユーザーに対するサービスの中断が最小になるか、CPC にかかる負担を最小にできるかを考えてください。

決定を下す前に、所有権に関する VTAM 障害の意味について考慮してください。まず初めに、XRF にとって特別な意味を持つ VTAM についての情報を検討してください。

端末ユーザーが IMS にログオンする場合、あるいは IMS からログオフする場合には、所有権が重要です。個々の VTAM には 1 つのシステム・サービス制御点 (SSCP) があり、所有している端末に対するセッション開始およびセッション終了を行います。したがって、所有している VTAM で障害が発生すると、VTAM の一部の機能は使用できなくなります。IMS にログオンされているクラス 1 端末のサービスは、継続します。しかし、そのドメイン内のクラス 1 端末のユーザーがログオンしようとする、VTAM は要求を受け入れることができません。ネットワーク・オペレーターは、VTAM の再始動時に端末の所有権を再設定しなければなりません。あるいは、ネットワーク・オペレーターは、所有権を別の VTAM に移行することができます。

テークオーバー時に、クラス 1 端末の所有権は代替 IMS システムの VTAM へは移行しません。所有権をアクティブ VTAM に割り当てて、アクティブ IMS で障害が発生した場合、クラス 1 端末は代替 IMS システムに切り替わりますが、障害が発生した IMS システムの VTAM は所有権を維持します。障害が発生した IMS システムの VTAM は、障害が発生した IMS システムがサービスに復帰したが、代替 IMS システムとしてのみ機能する場合でも、所有権を維持します。

端末所有権について決定する際は、次の 3 つのオプションを考慮してください。クラス 1 端末の所有者としては、以下のものが可能です。

- アクティブ・システムの VTAM
- 代替システムの VTAM
- 通信管理構成 (CMC) の VTAM

管理者は、クラス 1 端末の所有権を CMC の別の VTAM に割り当ててください。XRF 複合システム外の CPC に所有 VTAM を置くと、以下のような利点があります。

- 通常操作中のアクティブ・システムの競合が少なくなります。
- テークオーバー中の代替システムの競合が少なくなります。
- XRF 複合システムの障害が起きても、クラス 1 端末を所有している VTAM が終了しません。
- SNA 交換 (つまりダイヤルアップ) 回線上の NCP に接続した端末をクラス 1 端末にすることができます。

CMC を持っていない場合は、クラス 1 端末の所有権を複合システム内の VTAM に割り当ててください。また、VTAM を再始動しない場合は、テークオーバー後に所有権を変更しないでください。

関連概念:

 [z/OS: SNA 操作の概要](#)

USERVAR を使用する XRF についての NCP 計画に関する考慮事項

ネットワーク制御プログラム (NCP) についての基本的な計画作業は、それぞれの 37x5 通信コントローラーに接続するクラス 1 端末の数を決定することです。個々の NCP に対して、BUILD 定義ステートメントの BACKUP オペランドで、この数を指定します。このアクションによって、NCP は SSP を使用して、クラス 1 端末用の 1 次セッションおよびバックアップ・セッションに適した数の制御ブロックを生成することができます。

クラス 1 端末に接続した 37x5 通信コントローラーを IMS 作業専用にする必要はありません。また、ゲートウェイを含むその他のネットワーク・トラフィックをサポートすることもできます。クラス 1 端末をサポートするゲートウェイ 37x5 通信コントローラーのストレージ要件を計画するときには、この種の 37x5 通信コントローラーが、XRF 端末もゲートウェイもない 37x5 通信コントローラーより多くのストレージを必要とすることに留意してください。

関連資料: NCP ストレージについて詳しくは、「z/OS Communications Server: SNA ネットワーク・インプリメンテーション・ガイド」を参照してください。

パフォーマンスの考慮事項

XRF 機能を持つ構成を評価する場合は、一般の可用性基準と応答時間目標とを区別しなければなりません。構成を慎重に計画することによって、代替アクセス・パスを提供し、単一の障害点の発生回数を減らすことができます。

トータルでシステム応答特性を考える場合は、アクティブの各システムに対する応答を評価します。IMS、ページ固定、および領域使用率の初期パラメーターは、CPC と予期される作業負荷に合わせてサイズが決定されます。

応答時間の目標

特殊な状態を考慮するには、どちらかの CPC の作業負荷に対して確立した目標を調整する必要があります。例えば、計画停止から正常なテークオーバーが発生する状態を、統計から除外してください。重要なトランザクションの目標を設定する場合は、通常、テークオーバー処理の後には、応答が安定化するのに要する期間が続くことを覚えておいてください。エンド・ユーザーは、若干の遅延を経験します。

容量計画

XRF 複合システムの仮想記憶域の必要量を計画する際に重要な問題は、所定の IMS ユーザー・セットに許容できるサポートを提供するのに必要なリソースの量です。同一の代替 CPC で複数の IMS 代替サブシステムが実行されることになっている場合は、見積もりを慎重に行わなければなりません。

z/OS 計画に関する考慮事項

XRF を持たない z/OS システムは、単一のインストール・システムの z/OS XRF システムと共存できます。特に、z/OS XRF と非 XRF システムは、グローバル・リソース・シリアライゼーション (GRS) リング、および、単一の JES2 複合システムまたは JES3 複合システムを構成することができます。z/OS システムが 1 つの場合でも、XRF IMS サブシステムは別の IMS を含む非 XRF ジョブまたはサブシステムと共存できます。

以下のリストで z/OS 計画に関する考慮事項の一部について説明しています。

XRF 複合システム内での z/OS 自動リスタート・マネージャー

XRF 環境では、代替 IMS システムは、トラッキング・フェーズを開始するときに z/OS 自動リスタート・マネージャー (ARM) の関連を発行します。これによって、バックアップの引き継ぎ後は、アクティブ IMS システムが ARM によって自動的に再始動しないことが保証されます。バックアップの引き継ぎの後で アクティブ IMS が自動的に再始動してしまうと、IMS ログ (OLDS) とメッセージ・キューは破壊されることになります。

IMS が再始動を完了する前に異常終了すると、異常終了した IMS は ARM から登録が解除され、自動的な再始動は行われません。XRF トラッキングは、XRF バックアップの再始動が完了していると考えられるからです。

グローバル・リソース・シリアライゼーションに関する考慮事項

XRF には、グローバル・リソース・シリアライゼーションは必要ありません。

ん。しかし、XRF 複合システムの 1 つの CPC がグローバル・リソース・シリアライゼーション・リングにある場合は、複合システムのもう一方の CPC は同じリングになければなりません。XRF を持つ z/OS と XRF を持たない z/OS は、単一のグローバル・リソース・シリアライゼーション・リングを構成することができます。

XRF および XRF テークオーバーは、グローバル・リソース・シリアライゼーションのための通常のプロシージャーには影響を与えません。しかし、2 つのシステムがグローバル・リソース・シリアライゼーション・リングにあると、ユーザーは XRF とグローバル・リソース・シリアライゼーションとの関係に気付きます。

アクティブ IMS システムが実行している CPC または z/OS オペレーティング・システムで障害が発生すると、代替 IMS システムがテークオーバーを要求します。グローバル・リソース・シリアライゼーションからのメッセージは、リングが破壊されていることを示します。システムが再始動 (SYS1.PARMLIB の GRSCNFxx メンバーの RESTART(YES) パラメーターによって、自動的に始動するか、オペレーターが VARY GRS,RESTART コマンドを実行して始動) すると、破壊された GRS リングをグローバル・リソース・シリアライゼーションが再作成します。障害が発生したシステムが所有しているリソースを解放するために、オペレーターは VARY GRS,PURGE コマンドを実行します。オペレーターが z/OS を IPL して、IMS を XRF 複合システムの新しい代替サブシステムとして再始動すると、システムはリングを再結合します。

代替システムで CPC または z/OS が障害を起しても、アクティブ IMS のプロセスは正常に継続します。オペレーターは VARY GRS,PURGE コマンドを入力して、障害が発生したアクティブ IMS が所有しているリソースを解放し、リングの別の CPC に代替 IMS を再配置しなければなりません。

すべての IMS データ・セット名を、グローバル・リソース・シリアライゼーションの SYSTEMS 排他リソース名リスト (RNL) に入れてください。インストール・システムでの命名規則によっては、すべての IMS データ・セット名を 1 つの総称項目を使ってリストに入れることができます。SYSTEMS 排他 RNL の名前をリストしないと、グローバル・リソース・シリアライゼーションは、それらのデータ・セットへのアクセスや、DBRC および IRLM が IMS データ・セットにすでに提供しているサービスをシリアライゼーションします。

DBRC RECON または OLDS および WADS 名を RESERVE 変換 RNL に入れないでください。これらを入れると、IMS が予約を発行したときに、物理予約が DASD に送られます。

CTC で障害が発生した場合に、インストール・システムが GRS を使用し続けられるように、バックアップ CTC をお勧めします。バックアップ CTC がないと、すべてのグローバル ENQ 要求は、システム間の通信をオペレーターが再確立するまで中断されたままになります。

JES に関する考慮事項

XRF を持つ z/OS と XRF を持たない z/OS は、どちらも JES2 または JES3 複合システムを構成できます。アクティブ IMS システムと代替 IMS システムは、2 つの JES3 グローバル・システム内に置くことができま

す。JES3 のサービスを使用する場合は、すべての IMS データ・セットとデータベースを RESDSN ステートメントに入れてください。以上のことを行わず、JES3 グローバル・システムで障害が発生した場合は、JES3 グローバル・システムが使用可能になるまで、アクティブ IMS はデータ・セットとデータベースを割り振ることができません。

JES3 を立ち上げるときには、IMS 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。

RACF に関する考慮事項

代替 IMS はテークオーバー時に RACF データ・セットのセキュリティー情報を必要とするので、アクティブ IMS システムと代替 IMS サブシステムで共用されている DASD に RACF データ・セットを置いてください。単一障害ポイントを回避するために、RACF バックアップ機能を使用して、これらのデータ・セットの 2 番目のコピーも共用 DASD に保持します。

RACF RDEFINE APPLDATA コマンドの REVERIFY オペランドは、テークオーバー中に無効にされます。インストール・システムがこのオペランドを使用すると、ユーザーはテークオーバー後に再びサインオンして、パスワードを新しいアクティブ IMS に通知しなければなりません。

XRF のためのシステムの準備

IMS システム定義の過程で、XRF 複合システムの一部になるように IMS を定義します。

そのためには、次のようにします。

- IMS 実行 JCL の調整
- システム定義マクロ・ステートメントをコーディングして、以下を行います。
 - 生成されたシステムの一部になるように XRF を確立します
 - VTAM アプリケーション名およびパスワードを指定します
 - IMS マスター端末および 2 次端末を定義します
 - 個々の端末をカスタマイズします
 - オプションで、端末と MSC リンクが使用可能になる順序に影響を与える優先順位を宣言します
- IMS.PROCLIB メンバーを使用して、システム・データ・セットを初期設定します

IMS 実行 JCL の調整

XRF 複合システムの個々の CPC は、連続して実行される IMS ジョブを持っています。z/OS START コマンドは、2 つの異なるプロシージャーを使用して、アクティブ IMS と代替 IMS を初期設定します。

制御領域 EXEC ステートメントの XRF 関連パラメーターは、以下のとおりです。

HSBID=

一方のプロシージャーでは HSBID=1 として指定され、もう一方のプロシージャーでは HSBID=2 として指定されます。HSBID=null と指定すると、XRF 機

能を非アクティブにします。テークオーバー後の代替 IMS として別の IMS を立ち上げる場合は、障害が発生したアクティブ IMS と同じ値を HSBID= で指定してください。

HSBMR=

IMS.PROCLIB に含まれている DFSHSBxx メンバーに対応する 2 文字の接尾部として指定されます。このメンバーは、制御情報とテークオーバー基準が指定される場所にあります。

MNPS=

MNPS を使用するものとして XRF を定義し、MNPS ACB の名前を指定します。EXEC ステートメントの MNPS= キーワードは、DFSHSBxx PROCLIB メンバーの MNPS= キーワードおよび USERVAR= キーワードをオーバーライドします。

MNPSPW=

MNPS= キーワードに指定された MNPS ACB 名と一緒に使用されるパスワードを定義します。これはオプション・キーワードで、DFSHSBxx PROCLIB メンバーでの指定をオーバーライドします。VTAM がパスワードを検査するよう構成されている場合、VTAM が検査するのはこのパスワードです。これが指定されず、VTAM でこのパラメーターを予想している場合、MNPS ACB は開かれません。

USERVAR=

USERVAR 名を指定します。MNPS ACB 名が MNPS= キーワードを使用して定義されている場合、このキーワードは無視されます。

XRF に対する IMS システム定義マクロ・ステートメントのコーディング

IMS システムを定義する場合は、既存の生成デックにあるマクロ・ステートメントのキーワードを指定します。

XRF キーワードは、以下のシステム定義マクロに現れます。

COMM
CTLUNIT
IMSCTRL
LINE
LINEFRP
MSLINK
MSPLINK
STATION
TERMINAL
TYPE

一部のリソース (MSC リンクなど) では、リソースを動的に定義することができます。この場合、対応するシステム定義マクロはありません。

システム定義マクロ・ステートメントの構文については、「IMS V15 システム定義」を参照してください。

XRF には、以下のキーワードをコーディングすることが必要です。

- IMS に対して XRF サポートを確立する
- 2 つの VTAM アプリケーション名を提供する
- ACB セキュリティーのために 2 つのパスワードを指定する
- マスターと 2 次の IMS 端末を定義する

テークオーバー時に何が発生するかを定義する他のキーワードにデフォルトを使用できます。デフォルトを使用すると、テークオーバー時に IMS サポートの変更が可能であり、テークオーバー時にセッションをリカバリーするのに IMS が使用する優先順位を変更できます。端末を定義するマクロおよび ETO 記述子に XRF がデフォルトを使用できるようにすると、IMS が端末に提供できる最良のリカバリーを達成できます。テークオーバー時にセッションをリカバリーするための IMS に対する優先順位のデフォルトは 4 で、最高の優先順位は 7 です。

VTAM アプリケーション名およびパスワードの指定

アクティブ IMS システムおよび代替 IMS システムの VTAM アプリケーション名とパスワードを指定するには、COMM マクロで APPLID= キーワードと PASSWORD= キーワードを使用します。

- COMM マクロの APPLID= キーワードでアクティブ IMS システムおよび代替 IMS システムの VTAM アプリケーション名を指定します。

APPLID= キーワードは、VTAM が IMS システムの認識に使用するアプリケーション名 (APPLID 名) を指定します。APPLID 名は、ACB に格納されます。

APPLID 名は、XRF 複合システムが MNPS を使用するのか、USERVAR を使用するのかによって使い方が異なります。

MNPS を使用する XRF 複合システムの場合

APPLID 名は、XRF 複合システム内の各 IMS を VTAM に対して固有に定義します。APPLID 名を使用して、XRF 複合システム内の特定の IMS システムに直接ログオンすることができます。APPLID 名および ACB には、MNPS を使用する XRF 複合システムでのテークオーバー中の端末切り替えでは役割がありません。

USERVAR を使用する XRF 複合システムの場合

- VTAM は USERVAR テーブルに APPLID 名を入れます。一度に USERVAR テーブルに入れられる APPLID 名は 1 つだけでは、アクティブ IMS システムに適用される名前です。テークオーバー時に、IMS は、それを代替 IMS システムの APPLID 名と置き換えるように VTAM に指示します。
 - APPLID 名は、VTAM がクラス 1 端末についてバックアップ・セッションをサポートすることを許可する VTAM APPL 定義ステートメントのラベルと同じでなければなりません。
- COMM マクロの PASSWORD= キーワードでアクティブ IMS システムと代替 IMS システムの VTAM パスワードを指定します。

COMM マクロの PASSWORD= キーワードは、VTAM が ACB で検査する 2 つのパスワードを定義します。

例えば、パスワード IMSP1 と IMSP2 で、VTAM アプリケーション名 IMSNAME1 と IMSNAME2 指定するには、次のようにします。

```
COMM APPLID=(IMSNAME1,IMSNAME2),PASSWORD=(IMSP1,IMSP2)
```

IMSマスター端末および 2 次端末の定義

アクティブ IMS サブシステムと代替 IMS サブシステムはどちらも、常にアクティブである IMS 基本マスター端末と 2 次マスター端末を持っています。これらを定義するには、TERMINAL マクロで XRF 関連のキーワードを指定します。

VTAM が IMS マスター端末を制御している場合は、NAME キーワードを指定します。

TERMINAL マクロの NAME キーワードは、VTAM マスター端末または 2 次端末に対して別のノード名を定義します。これは、両方の名前が COMM マクロの APPLID キーワードと一致する場合に、アクティブ・サブシステムと代替サブシステム間の IMS 管理の ISC リンクを定義するのにも使用されます。

個々の端末のカスタマイズ

複合システムの端末の数とは無関係に、それらの端末に対して XRF サポートを得るには、キーワードをさらに指定する必要はありません。デフォルトを受け入れると、現在の端末で可能な最良のリカバリーを IMS が判別します。

例えば、以下の基準をすべて満たす端末が、クラス 1 端末として自動的に定義されます。

- UNITYPE キーワードに SLUTYPE2 として定義されている。
- 37x5 通信コントローラーに接続されている。
- XRF をサポートする NCP および VTAM で制御されている。

UNITYPE=SLUTYPE2 として定義され、37x5 通信コントローラーに接続されている任意の端末を、IMS は自動的にクラス 2 と見なします。

以下の表は、個々の端末をカスタマイズするためにコーディングするキーワードを示しています。

表 55. VTAM 制御端末用の XRF システム定義マクロのキーワード

マクロ	キーワード
COMM	APPLID
	PASSWORD
TYPE	BACKUP
TERMINAL	BACKUP
	NAME
NAME	

TYPE マクロ、TERMINAL マクロ、および ETO 記述子の BACKUP キーワードは、クラス 1 とクラス 2 の端末のセッション・リカバリーを指定し、IMS がそれらの端末のサービスをリカバリーする優先順位を設定します。BACKUP キーワードのデフォルトは (4,YES) です。これは、クラス 1 の資格のある端末が、テークオー

バー時に切り替え優先順位 4 で自動的にバックアップ・セッションを持つことを意味します。クラス 2 の資格のある端末は、テークオーバー時に優先順位 4 で新しいセッションが自動的に確立されます。

何らかの理由でクラス 1 またはクラス 2 サポートをクラス 3 に変更する場合は、BACKUP=NO を指定してください。

クラス 1 またはクラス 2 の端末のサービスのリカバリーには、BACKUP キーワードを使用して、4 以外の優先順位を確立してください。BACKUP=n を指定してください。この n は、1 (最も遅いリカバリー) から 7 (最も早いリカバリー) までの数字です。

これらの優先順位は、IMS によって使用されますが、ネットワーク接続要求は、内部 VTAM プロセスのために、別の順序で完了されます。

クラス 1 サポートをクラス 2 サポートに変更するには、BACKUP=n,NO をコーディングしてください。この n は、端末に対するセッション・リカバリー優先順位です。

例えば、次のようにします。

- TYPE マクロで定義されているすべての端末を、セッション・リカバリー優先順位が 7 のクラス 2 端末にしたい場合は、次のように指定します。

```
TYPE . . .BACKUP=(7,NO)
```

- テークオーバー時にクラス 1 端末のセッションを IMS にリカバリーさせたくない場合は、次のように指定します。

```
TERMINAL . . .,BACKUP=NO
```

MSC リンクの定義

XRF IMS が MSC リンクを用いて別の IMS システムと通信している場合は、IMS がこれらのリンクで新しいセッションを確立する優先順位を変更できます。

MSPLINK 定義の BACKUP キーワードは、MSC 物理リンクの自動リカバリーを制御し、テークオーバー時にこれらの物理リンクで新しいセッションを確立する優先順位を設定します。これらのリンクを使用して新しいセッションを確立するのに IMS が使用する優先順位を高くすることを考えてください。デフォルトは BACKUP=4 です。

MSLINK 定義の BACKUP キーワードを使用すると、論理リンク定義に対して MSPLINK 定義で指定した BACKUP オプションをオーバーライドできます。デフォルトは BACKUP=4 です。

物理リンクは動的に定義できます。論理リンクの定義で BACKUP を指定すると、関連付けられた物理リンクの定義に指定された切り替えオプションはすべてオーバーライドされます。

XRF のための IMS.PROCLIB メンバーの定義

PROCLIB データ・セット・メンバー DFSDCxxx、DFSHSBxx、DFSFIXxx、および DFSVSMxx で XRF のキーワード値を指定します。

以下の PROCLIB メンバーには、XRF システムを定義するのに使用できるパラメーターが含まれています。

- DFSDCxxx

DFSDCxxx PROCLIB メンバーは、PSTIMER パラメーターを保持します。PSTIMER パラメーターは、代替 IMS が新しい MNPS ACB をオープンするのを待っているときに VTAM に持続セッションをオープンさせておく時間の長さを指定します。この時間が満了するまでに新しい MNPS ACB が開かれない場合、セッションはクローズされます。デフォルト値は 3600 秒です。

- DFSHSBxx

DFSHSBxx メンバーは、z/OS 可用性マネージャー、DBRC、および IRLM に認識されている、XRF 複合システムの名前を指定します。また、複合システムに対する監視メカニズムおよびテークオーバー条件も指定します。

推奨事項: アクティブ・サブシステムと代替サブシステムの両方に同じメンバーを使用してください。

DFSHSBxx メンバー内の XRF のパラメーターについては、『DFSHSBxx の XRF パラメーター』を参照してください。

個々のサブシステムは、連続動作する IMS を持っています。EXEC ステートメントの HSBID パラメーターを使用して、どの IMS がアクティブ IMS であるかを最初に宣言します。HSBID パラメーター HSBID=1 または HSBID=2 は、COMM ステートメントの各 IMS APPLID、VTAM マスター端末の NODE を指します。HSBID は、各サブシステムに関連付けられているすべてのメッセージ・キュー・データ・セットも識別します。

また、IMS.PROCLIB に含まれている DFSHSBxx メンバーに対応する接尾部も、HSBMBR パラメーターを使用して指定する必要があります。

DFSHSBxx の XRF パラメーター

XRF 用に IMS システムを調整する場合、IMS.PROCLIB の DFSHSBxx メンバーに IMS 制御ステートメントを指定する必要があります。制御ステートメントは、多くのパラメーターから構成されています。

アクティブ IMS 用に DFSHSBxx のコピーが 1 つと、代替 IMS 用に DFSHSBxx のコピーが 1 つあります。2 つのコピーは同一である必要はありませんが、パラメーター RSENAME と、MNPS または USERVAR が各コピーで同じでなければなりません。IMS.PROCLIB プロシージャ・ステートメントの HSMBR=xx は、DFSHSBxx メンバーを定義します。

IMS はオペレーターが DFSHSBxx の監視オプションを動的に変更するのに使用できるコマンドを提供しますが、DFSHSBxx の他のパラメーターを変更するのは、z/OS が新しい IMS を始動するときだけです。

IMS.PROCLIB メンバーの DFSHSBxx に指定する以下の操作パラメーターは、パフォーマンスを向上させます。

- SURV

- RSENAME
- MNPS
- MNPSPW
- USERVAR
- SWITCH
- ALARM
- AUTO
- KEYEVENT
- DEFERFIX
- LNK
- LOG
- 再始動データ・セット (RDS)

SURV=ALL, LNK, LOG, RDS, NO

XRF IMS は監視メカニズムを提供して、アクティブ IMS での障害を代替 IMS に通知します。監視メカニズムがないと、アクティブ IMS で障害が発生しても、代替 IMS には何も警告されません。SURV パラメーターは、XRF 複合システムに対してどのような監視メカニズムが有効になるかを指定します。

SURV パラメーターのオプションは、次のとおりです。

オプション

結果

- ALL** ALL パラメーターは、3 つの監視メカニズム (LNK、LOG、および RDS) をすべて使用することを示します。
- LNK** LNK パラメーターは、LNK パラメーターのインターバル値によって指定した頻度で、アクティブ IMS が ISC リンクを使って代替 IMS に信号を送信することを示します。ISC リンクは、アクティブ IMS が示している SNAPQ チェックポイントを同期化中に代替 IMS から要求するのにも使用されます。
- LOG** LOG パラメーターは、代替 IMS が LOG パラメーターのインターバル値が示しているのと同じ頻度で OLDS の新しいレコードをチェックすることを示します。このチェックは、代替 IMS による OLDS へのアクセスだけでなく、アクティブ IMS で発生している処理を反映する代替 IMS のシステム状況を維持するためにも行われます。
- RDS** RDS パラメーターは、RDS パラメーターのインターバル値が示しているのと同じ頻度で、アクティブ IMS が共用再始動データ・セット (RDS) にタイム・スタンプを入れることを示します。代替 IMS は RDS を検査して、タイム・スタンプが書き込まれていることを確認します。
- NO** このパラメーターは、監視メカニズムがアクティブでないことを示します。

LNK、LOG、RDS のどのような組み合わせでも指定できますが、この 3 つをすべて使用することをお勧めします。マスター端末から発行するコマンドによって、監視パラメーターおよびインターバル・パラメーターを動的に変更できません。

3 つの監視メカニズム (LNK、LOG、および RDS) は、それぞれインターバル値とタイムアウト値とともに指定されます。アクティブ IMS では、インターバル値は、監視メカニズムが信号を送信したり、データ・セットを更新する頻度を示します。代替 IMS では、インターバルは、信号またはタイム・スタンプが受信される頻度を示します。タイムアウト値を超過すると、代替 IMS はテークオーバーを開始します。これらの時間値は、おおよその値にすぎません。実際の値は、若干大きくなります。

これらの監視パラメーターは、次のことを決定します。

- どのような監視メカニズムを複合システムで使用するか
- 代替 IMS が監視メカニズム (LNK、LOG、RDS) から信号を受信する頻度
- 監視メカニズムが欠如していることが原因でテークオーバーが発生するかどうか

代替 IMS は、これら 3 つの監視メカニズムをさまざまな方法で使用します。ISC リンク (LNK) は、LOG オプションまたは RDS オプションと一緒に使用すると最も効果的な低オーバーヘッドの監視メカニズムです。ISC リンクで障害が発生したり、RDS が書き込みエラーを発生させると、アクティブ IMS は、障害を示すためにログ・レコードを書き込みます。代替 IMS が ISC リンクまたは RDS の信号の受信に失敗することは、障害の原因になる可能性があります。代替 IMS がログの新しいレコードの受信に失敗しても、障害の原因にはなりません。例えば、SURV=LNK,LOG を指定して、代替 IMS が ISC リンクの信号の受信を停止するとします。ISC リンクの信号の障害はテークオーバーを示しますが、ログがレコードを十分に送信し続ける限り、IMS はテークオーバーを要求しません。この場合、IMS は、ISC リンク自体で障害が発生しているので、信頼できる標識ではないと見なします。

ログの読み取りと、代替 IMS による監視メカニズムとしての LOG を混同しないでください。XRF IMS は、代替 IMS がログを読み取ることを要求します。XRF IMS は、ログの情報に従って、制御ブロックを更新します。SURV=LOG を指定した場合のみ、代替 IMS はログを定期的にチェックして、アクティブ IMS が新しいレコードを送信していることを確認します。代替 IMS が新しいレコードの受信に失敗しても、テークオーバーは発生しません。IMS は LOG オプションを使用して、RDS または ISC リンクからの信号の欠如に基づいて代替 IMS が行う決定を承認または禁止します。LOG オプションを唯一の監視メカニズムとして使用しないでください。このオプションは、LNK または RDS が失敗したときにシステムが実際に機能していないことを確認するのに使用すると最も効果的です。LOG レコードがないということは、指定したインターバルの間、作業を行う必要がないということを単に意味します。

アクティブ IMS は 2 つの異なる速度でログへの書き込みを行い、代替 IMS はログを読み取ります。代替 IMS は、アクティブ IMS に追いつかれなければ、ログを連続的に読み取ります。代替 IMS がアクティブ IMS に追いつかれた場合は、LOG パラメーターのインターバル値が、代替 IMS がログを読み取る頻度を制御します。したがって、このインターバルが入出力アクティビティーのレベルに影響を与えることがあります。代替 IMS の作業を減らす 1 つの方

法は、LOG パラメーターで指定するインターバル値を大きくすることです。インターバルを小さく設定しすぎると、データ・セットへの入出力アクティビティが増え、そのことがアクティブ IMS のパフォーマンスに影響を与える場合があります。

RDS は重要な監視ツールです。ISC リンクで障害が発生したり、システムが非アクティブで、ログ・レコードが書き込まれないと、RDS は更新され続けます。例えば、SURV=LNK,RDS を指定して、z/OS が待ち状態に入ると、ISC リンクおよび RDS は定期的な信号の送信を停止します。これによって、代替 IMS に障害を警告します。監視メカニズムを指定しないと、待ち状態であるという警告を代替 IMS は受け取りません。その場合には、オペレーターがこの状態に気付いて、テークオーバーを要求する必要があります。以下の表は、ある一定のイベントについて知るために、代替 IMS が RDS、IMS リンク、およびログ・レコードをどのように使用するかを示しています。

表 56. アクティブ IMS 内のイベントの代替 IMS への通知

イベント	ログ・レコードが代替 IMS に通知するか	代替システムに通知する監視オプション
IMS 異常終了	あり	ありません。
z/OS 異常終了	なし	LNK、RDS
z/OS ループ ¹	なし	LNK、RDS
z/OS 待ち状態 ¹	なし	LNK、RDS
CPC 障害	なし	LNK、RDS
TPEND 出口を引き起こす VTAM 障害	あり	ありません。
VTAM 待ち状態	なし	LNK、RDS
IRLM 障害	あり	ありません。

注:

1. 監視で検出できるループおよび待ち状態

監視メカニズムとして LNK および LOG を使用するには、初期制御ステートメントを指定するだけで、他には何も必要ありませんが、ISC リンクがまだ所定の場所がない場合があります。その場合には、クラス 1 端末を制御する 37x5 通信コントローラーで、ISC リンクを確立しなければなりません。

SURV パラメーターを指定しないと、IMS は 3 つのオプション (LNK (ISC リンク)、RDS、および LOG) をすべて使用します。

アクティブ IMS からの信号の頻度のデフォルトは、ISC リンクでは 3 秒で、RDS では 1 秒です。LNK および RDS に対するタイミングの設定で考慮すべき要因は、ISC リンクを使用した通信の速度と、代替 IMS の z/OS のパフォーマンス・オーバーヘッドの 2 つです。

代替 IMS がログの新しいレコードについてチェックする頻度のデフォルトは、1 秒です。

RSENAME=

リカバリー可能サービス・エレメント (RSE) は、アクティブ・サブシステムと代替サブシステムから構成されています。VTAM、z/OS、および CPC の障害が原因でテークオーバーが発生する場合がありますが、XRF 複合システムのこ

これらのエレメントは、障害の認識とテークオーバーの開始を IMS に依存します。RSENAME は、1 から 8 文字からなる ID です。IMS は、この ID によって、DBRC、IRLM、および z/OS 可用性マネージャーに認識されます。DBRC データベースのサインオンと許可付与は、RSENAME キーワードを使用して行われます。RSENAME は、アクティブ・サブシステムと代替サブシステムを AVM に通知するのに使用される ID です。また、AVM のメッセージで表示される名前でもあります。XRF には、RSENAME が必要です。

MNPS=

MNPS パラメーターは、MNPS を使用する XRF 複合システム内の IMS システムの MNPS ACB を定義します。この名前は、アクティブ IMS システムと代替 IMS システム両方のプロシーチャーで同じでなければなりません。ユーザーは、XRF 複合システムにログオンするときに、この名前を IMS ログオン名として入力します。テークオーバーが発生すると、アクティブ IMS 上の MNPS ACB のインスタンスがクローズされ、代替 IMS は、同じ MNPS ACB 名を使用して新しいインスタンスを開きます。このキーワードを使用して行われた指定は、すべての USERVAR 指定をオーバーライドします。

MNPSPW=

このキーワードは、XRF 複合システム内の IMS システムの MNPS ACB のためにパスワードを定義するのに使用します。各 IMS システムは、MNPS ACB を開いたときにこのパスワードを VTAM に提供します。パスワードの有無を検査するように VTAM を設定することも必要です。VTAM がパスワードを予想しており、管理者がパスワードを指定していない場合、MNPS ACB は開かれません。

USERVAR=

USERVAR パラメーターは、USERVAR を使用する XRF 複合システム内で IMS の USERVAR を定義します。USERVAR は、USERVAR テーブルと解釈テーブルの両方に格納されます。USERVAR は、USERVAR テーブルではアクティブ IMS システムの APPLID にリンクされます。USERVAR は解釈テーブルではログオン・メッセージにリンクされます。XRF 複合システムにログオンするユーザーが知っている必要があるのは、アクティブ IMS システムに接続されるログオン・メッセージだけです。テークオーバーが発生すると、IMS は、新しいアクティブ IMS システムの APPLID をアクティブ IMS システムの USERVAR テーブルに入れます。

MNPS パラメーターに値が入力された場合、USERVAR パラメーターはオーバーライドされます。

SWITCH=

次のリストには、SWITCH パラメーターのオプションが記述され、代替 IMS がテークオーバーを考慮するときに、そのオプションをどのように使用するかについて説明されています。

オプション

代替 **IMS** がテークオーバーを考慮する場合

IRLM IRLM で障害が発生したことをログ・レコードが示している。

LNK 指定されている時間が経過しても、ISC リンクの信号が送信されない。

LOG 指定されている時間が経過しても、ログに新しいレコードが現れない。

RDS 指定されている時間が経過しても、RDS リンクのタイム・スタンプが現れない。

TPEND

VTAM が IMS TPEND 出口を起動してから、障害を起こしたことをログ・レコードが示している。

VTAM の障害をテークオーバーの原因として確立するには、SWITCH パラメーターの TPEND オプションを指定します。

ローカル・ロックの管理、または、XRF 複合システムが他の IMS システムと共用しているリソースを管理するのに IRLM を使用する場合は、IRLM の障害をテークオーバーの原因にしたい場合があります。SWITCH パラメーターの IRLM オプションを指定してください。IRLM でテークオーバーを発生させたい場合は、AUTO=YES を指定してください。

代替 IMS にある一定の障害を警告する際に重要である SWITCH パラメーターで、監視メカニズムを指定してください。これらのオプションを組み合わせることもできますし、複数回使用することもできます。例えば、SWITCH=(LNK,LOG) を指定すると、指定されている時間が経過しても ISC 信号が送信されず、新しいログ・レコードが現れない場合に、IMS はテークオーバーを考慮します。これらのイベントのいずれか 1 つだけの場合は、IMS はテークオーバーを考慮しません。

テークオーバーを考慮するまでに、どのくらいの時間信号を待つかを代替 IMS に知らせる LNK、LOG、および RDS パラメーターのタイムアウト値を指定してください。ISC リンクではデフォルトのタイムアウトは 9 秒で、LOG と RDS では 3 秒です。これらの時間はおよそその値にすぎず、実際の値は若干大きくなります。

SWITCH パラメーターのデフォルトは、次のとおりです。

SWITCH=(LNK,LOG,RDS),(TPEND),(IRLM)

VTAM が障害を起こして、その結果、IMS TPEND 出口になる場合、あるいは、IRLM が障害を起こして、その結果、IMS STATUS 出口になる場合、または、3 つのすべての監視メカニズムが障害を代替 IMS に警告する場合に、SWITCH パラメーターは、テークオーバーを考慮するように代替 IMS に指示します。

タイムアウト・インターバルを指定する場合は、以下の例外条件を考慮してください。

- ソフトウェア障害またはハードウェア障害が原因で、システムは 30 秒間から 60 秒間、IMS 作業のプロセスを停止します。

ある一定のソフトウェア障害またはハードウェア障害によって、最大 1 分間継続する待ち状態またはループが発生します。この種の障害の例は、ソフトウェア・リカバリー・ルーチン、ハードウェア・スピン・ループ、ハードウェア・リカバリー手順などです。タイムアウト・インターバルが、LNK の場合のデフォルトが 9 秒、RDS の場合のデフォルトが 3 秒、LOG の場合のデフォルトが 1 秒に設定されると、これらの条件によってテークオーバーが発生します。障害が頻発する場合は、原因を特定して、必要な変更を行わなければならないかもしれません。修正を行う前に、以下の手段をとることができます。

- LNK、LOG、および RDS パラメーター・ステートメントのタイムアウト・インターバルを長くして、その条件でテークオーバーが発生しないようにします。
- 条件自体を修正し、可能であればテークオーバーを回避するために条件に与える時間を長くし、AUTO パラメーターで NO を指定して、テークオーバー時にオペレーターが管理できるようにします。
- SLIP トラップは、トラップ条件が満たされた場合に再始動可能な待ち状態に入ることを指定します。これによってアクティブ IMS は、代替 IMS がテークオーバーするのに十分長い時間停止します。再始動可能な待ち状態によりテークオーバーが発生します。この状態では、テークオーバーは適切であるように見えます。トラップが発生すると、その後には、長時間の分析期間と、ほとんどの場合は、独立型ダンプが続きます。
- オペレーターが CPC を停止します。

オペレーターが CPC を停止したいときには、監視は停止されなければなりません。オペレーターが CPC を停止するのは、SDUMP または使用不可になったコンソールが原因で、新しい JES を入力するときです。オペレーターがアクティブ CPC を停止して、監視を停止しなかった場合は、監視メカニズムからの信号がなくなるために、代替 IMS がテークオーバーを要求します。テークオーバーが発生させたくない場合は、監視を中止するか、AUTO=NO を指定して、テークオーバーが進行する前にオペレーターに制御権を与えます。

テークオーバーは、一連の時間依存型パラメーターが満たされると発生する場合があるので、監視インターバルの設定は、重要な管理タスクです。また、それらのインターバルが、アクティブ IMS システムによって実行されるすべての作業負荷に適していない場合もあります。

監視インターバルに影響する要因は、次のとおりです。

- CPC の能力またはタイプ (例えば、UP または MP)
- 操作プロシージャおよび応答の目標
- タイムアウトを超過した場合のリカバリー/遅延状態
- 重要な作業に対しては、計画外の停止に対する相対的な許容度

ALARM=NO | YES

このパラメーターは、テークオーバーが開始されるときにサービス・プロセッサがオペレーターに通知するように要求します。デフォルトは NO です。

代替 IMS によるアクティブ IMS の自動テークオーバーが発生した場合、YES が指定されていると、代替 IMS のサービス・プロセッサ・アラームが鳴ります。システムを手動で切り替えるためにオペレーターの介入が必要な場合にも、このアラームが鳴ります。

AUTO=YES | NO

AUTO は、IMS がテークオーバーの自動進行を要求したり、オペレーターが介入した場合に、テークオーバーを自動的に進行させるかどうかを確立します。AUTO=YES を指定すると、テークオーバーが (監視メカニズムによって検出されたホストの障害に基づいて) 自動的に発生します。AUTO=NO を指定すると、オペレーターは手動でシステムを切り替えなければなりません。テークオーバーが指示されたことを、監視メカニズムによって代替 IMS が検出すると、メ

メッセージ DFS3869 が代替 IMS のマスター・コンソールに表示されます。AUTO=NO が指定されている場合は、オペレーターはテークオーバーを延期して、その間にアクティブ IMS の問題の解決に努めることができます。アクティブ IMS の問題が修正されている間、代替 IMS に対して通常のトラッキングが継続されます。問題が修正された後、またはオペレーターが切り替えを開始した後、テークオーバー・メッセージはアクティブ IMS に送信されなくなります。

オペレーターの介入によって、NCP が XRF 端末上のセッションを切り替えて、IMS が端末でセッションを再確立するまでの時間は確実に延長されます。また、オペレーターの介入によって、作業負荷プロセスの中断も発生します。しかし、インストール・システムでは、テークオーバーが進行する前にオペレーターがある一定のタスクを実施することを望む場合があります。そのタスクは次のとおりです。

- テークオーバー決定が有効であるかどうか確認する。
- 端末または DASD を手動で切り替える。
- 代替 IMS のジョブを静止する。

代替 IMS での作業の静止を制御することが重要な場合は、オペレーターの介入なしにはテークオーバーが進行しないように要求してください。これによって、テークオーバー要求からテークオーバー発生までの間に、ジョブを取り消す時間がオペレーターに与えられます。

オペレーターにテークオーバーを管理させたくない場合は、AUTO=YES を指定します。デフォルトは AUTO=NO です。

KEYEVENT=MSG | NONE

KEYEVENT は、XRF に対するすべての IMS メッセージを IMS オペレーターが受け取るかどうかを決定します。テークオーバー・メッセージは表示されなければなりません、ある一定のメッセージは表示されないようにすることができます。KEYEVENT=MSG を指定すると、すべてのメッセージが表示されます。KEYEVENT=NONE を指定すると、オプション・メッセージ DFS3882 から DFS3888 が表示されません。デフォルトは NONE です。

DEFERFIX=xx

このパラメーターは、テークオーバーまで代替 IMS の非高速機能リソースのページ固定が延期されることになっている場合に処理されなければならない DFSFIXxx PROCLIB メンバーを示します。このとき、IMS は高速機能のページ固定オプションを無視します。代替 IMS に対しては、それらの DL/I ブロック、および、既存の DFSFIXxx IMS.PROCLIB メンバーでページの固定化ができるルーチンだけが、ページ固定されます。

DFSFIXxx

DFSFIXxx PROCLIB メンバーは、IMS システムに対してページの固定化値を指定します。XRF 複合システムでは、2 つのメンバーを使用できます。一方のメンバーは、アクティブ IMS サブシステムと代替 IMS システムの両方に対して、初期設定時に何がページ固定されるかを指定します。このメンバーは、始動プロシージャの FIX=xx パラメーターによって示されます。もう 1 つのメンバーは、IMS.PROCLIB の DFSHSBxx メンバーの DEFERFIX = xx パラメーターによって

示されます。アクティブ IMS では、このメンバーで要求されたページ固定は、初期設定時に発生します。代替 IMS では、このメンバーで要求されたページ固定は、テークオーバー時に発生します。

初期設定中に代替 IMS で行われるページの固定化によって、テークオーバーは高速化されますが、トラッキング・フェーズを通して、代替 IMS に実記憶域がさらに必要になります。

DFSVSMxx

DFSVSMxx PROCLIB メンバーは、ログ・データ・セット定義情報を含み、OLDS と WADS の配置、OLDS に対して使用されるバッファの数、OLDS の操作のモード (単一または二重) を指定します。OLDS の消失が発生して、バックアップが使用できない場合、XRF 複合システムで障害が発生するので、重複 OLDS の使用をお勧めします。代替 IMS は、自身に対して定義されているログ・データ・セットと、アクティブ IMS によって使用されるデータ・セットを割り振るので、操作のモードとバッファ定義は、両方のサブシステムで同じでなければなりません。アクティブ IMS システムと代替 IMS サブシステムは、同じメンバーを使用しなければなりません。

VTAM USERVAR テーブルの定義

以下の例は、IMS システム定義パラメーターと、拡張回復機能 (XRF) と一緒に使用される VTAM 解釈 USERVAR テーブル・パラメーターとの関係を示しています。

関連資料:

- USERVAR テーブルについて詳しくは、「USERVAR を使用する XRF 複合システムでの VTAM の動作」を参照してください。
- IMS システム定義パラメーターについて詳しくは、「IMS V15 システム定義」を参照してください。

以下の作業を行う必要があります。

- IMS 生成時の COMM マクロの APPLID パラメーターおよび PASSWORD パラメーターで、両方のシステムの APPLID およびパスワードを指定しなければなりません。例えば、次のように指定します。

```
IMSCTRL HSB=YES COMM APPLID=(IMS1,IMS2),PASSWORD=(IMSP1,IMSP2)
```

- IMS システムの開始前に、USERVAR パラメーターで総称アプリケーション ID を指定する IMS-PROCLIB の DFSHSBxx メンバーを設定しなければなりません。例えば、次のように指定します。

```
USERVAR=IMS
```

- 総称アプリケーション ID は、ユーザー変数として VTAM 解釈テーブルに登録しなければなりません。例えば、次のように指定します。

```
LOGCHAR APPLID=(USERVAR,IMS),SEQUENCE='IMS'
```

- COMM マクロの APPLID パラメーターで指定される APPLID は、VTAM 定義ライブラリーに SPO (2 次プログラム式オペレーター機能) または PPO (基本プログラム式オペレーター機能) として定義しなければなりません。例えば、次のように指定します。

```
VBUILD TYPE=APPL IMS1 APPL AUTH=(ACQ,SPO),PRTCT=IMSP1,HAVAIL=YES
IMS2 APPL AUTH=(ACQ,SPO),PRTCT=IMSP2,HAVAIL=YES
```

XRF 構成における IMS データ・セットの配置

別の CPC の代替 IMS とともに動作している IMS は、動作的には単一システムとして管理されますが、IMS システム・データ・セットの重複について、詳細な計画を立てる必要があります。

データ・セットの配置に関する主要な 3 つの考慮事項は、次のとおりです。

- トラッキングおよびテークオーバーの際のデータ・セットの可用性

XRF 複合システムは、時には同一のデータ・セットあるいはそれらの同一のコピーにアクセスしなければならない、2 つのシステムから構成されています。したがって、IMS では、2 つのシステムによって共有されている DASD にいくつかのデータ・セットを割り振る必要があります。

推奨事項: 他のデータ・セットを共有 DASD に配置してください。ただし、切り替え装置によって一部のデータ・セットを切り替えたり、それらのデータ・セットの別個のコピーを保持したりすることができます。

- 単一障害ポイントの予防

IMS では、2 つのシステムに対して、いくつかのデータ・セットの別個のコピーを保持する (そして、常に同期化する) 必要があります。

推奨事項: 他のデータ・セットの別個のコピーを保持してください。

- 1 つの IMS システムからのデータ・セットのアクセス可能性

推奨事項: ローカル (つまり、非共有) DASD の 1 つのシステムに固有のデータ・セットを保持してください。

テークオーバー時に最良のパフォーマンスを達成するために、共有入出力と非共有入出力を相互に別々に保持します。例えば、IMS データベース・データ・セットを、z/OS データ・セットとは別のボリューム、制御装置、チャンネルに配置します。

システム・ログの配置は重要です。テークオーバー処理の重要なステップは、障害が発生しているアクティブ IMS からデータベースを分離することです。アクティブ IMS と代替 IMS が別の CPC で動作する場合は、OLDS および WADS が存在している DASD 装置を予約することにより、代替 IMS は、障害が発生しているアクティブ IMS が OLDS および WADS にアクセスするのを防止できます。したがって、OLDS および WADS が含まれている DASD に他のデータ・セットを配置しないでください。

すべての IMS データベースとエリア・データ・セットを動的に割り振らなければなりません。それらが存在していない場合は、すべての IMS のデータベースおよびエリアに対して DFSMDA メンバーを生成しなければなりません。

関連資料 すべての IMS データベースおよびエリアのための DFSMDA メンバーの生成について詳しくは、IMS V15 システム定義を参照してください。IMS 全機能データベース名および DEDB エリア名はすべて、固有の名前でなければなりません。

このトピックに記載されている要件および推奨のいくつかは、データ・セットの保護に関するものです。リソースによって可能な範囲まで、必ずデータ・セットを保護してください。以下のリストは、これらのデータ・セットの必須および推奨される配置をまとめたものです。

以下のリストは、データベース A (アクティブ IMS に関連したもの) 内のデータ・セットおよびデータの推奨配置を示しています。

- IMS.LGMSG
- IMS.LGMSG L
- IMS.SHMSG
- IMS.SHMSG L
- IMS.QBLKS
- IMS.QBLKS L
- IMS.MSDBDUMP
- SYSDUMP データ
- SYSABEND データ

以下のリストは、データベース B (代替 IMS に関連したもの) 内のデータ・セットおよびデータの推奨配置を示しています。

- IMS.LGMSG
- IMS.LGMSG L
- IMS.MSDBDUMP
- IMS.QBLKS
- IMS.QBLKS L
- IMS.SHMSG
- IMS.SHMSG L
- SYSDUMP データ
- SYSABEND データ

以下のリストは、データベース C および D (アクティブおよび代替 IMS システム両方に関連したもの) 内のデータ・セットおよびデータの推奨配置を示しています。

- IMS.ACBLIBA
- IMS.ACBLIBB
- IMS.FORMATA
- IMS.FORMATB
- IMS.JOBS
- IMS.MODBLKSA
- IMS.MODBLKSB

- IMS.PGMLIB
- IMS.PROCLIB
- IMS.SDFSRESL
- IMS.TFORMATA
- IMS.TFORMATB

XRF で必要な追加のデータ・セット

以下の図は、上記リストの情報を図示したものであり、4つのデータベース (A から D)、アクティブおよび代替 IMS システム、共用データ・セットの間の関連を示しています。

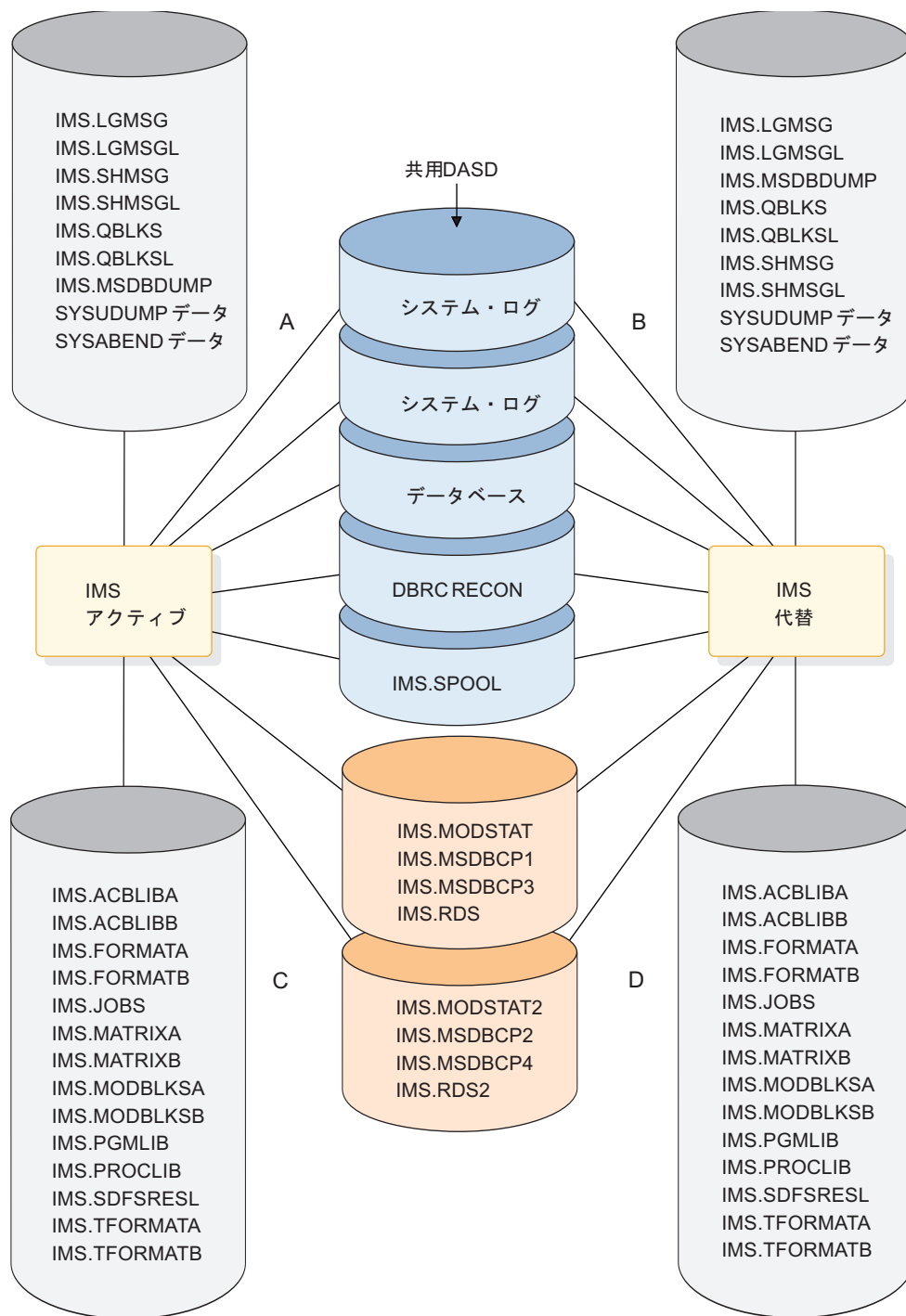


図 86. 推奨されるデータ・セットの配置

この図にリストしてある追加のデータ・セットは、XRF 複合システムの制御領域 JCL にあります。

表 57. XRF で必要な追加のデータ・セット

データ・セット	用途
IMS.RDS2	別個の再始動データ・セット
IMS.LGMSG	ローカル・メッセージ・キュー

表 57. XRF で必要な追加のデータ・セット (続き)

データ・セット	用途
IMS.SHMSGSL	ローカル・メッセージ・キュー
IMS.QBLKSL	ローカル・メッセージ・キュー
IMS.MODSTAT2	別個の MODSTAT データ・セット
IMS.MSDBCP3	代替 MSDB チェックポイント
IMS.MSDBCP4	代替 MSDB チェックポイント

XRF 複合システムで共用しなければならないデータ・セット

データベースを構成している重要なデータ・セットだけではなく、それぞれのシステム・データ・セットに 1 つずつコピーを用意し、アクティブ・サブシステムと代替サブシステムで共用する必要があります。これらのデータ・セットは、以下の DD 名を持っています。

- RECON1、RECON2、RECON3
- DFSOLPnn
- DFSOLSnn
- DFSWADSnn
- IMS スプール・データ・セット
- IMS.MODSTAT
- IMS.MODSTAT2
- IMS.RDS
- IMS.RDS2
- IMS.MSDBCP1
- IMS.MSDBCP2
- IMS.MSDBCP3
- IMS.MSDBCP4

推奨事項: 効率性を高め、操作を容易にするためには、2 つの CPC によって共用されている DASD に DL/I データベースを配置します。しかし、IBM 3814 などの切り替え装置や、IBM 3880 磁気ディスク制御機構のチャンネル切り替えなどを使用してこれらのデータ・セットを切り替えることを選択することもできます。

また、2 つの CPC によって共用されている OLDS の 2 つのコピーを持つこともお勧めします。OLDS のコピーが 1 つしかなく、永久入出力エラーまたはハードウェアの電源障害が原因でそのコピーが物理的に失われた場合には、トラッキングを続行できません。この場合には、新しいアクティブ IMS の障害が発生します。

831 ページの図 86 は、共用 DASD に置く必要のあるデータ・セットを示しています。この図では、これらは、アクティブ IMS と代替 IMS との間に置かれている DASD にあります。システム・ログ、再始動データ・セット (RDS)、MODSTAT、および MSDB データ・セットが共用 DASD だけでなく、別個のコピーも共用 DASD にあることに注意してください。アクティブ IMS はこれらのコピーを保持します。

XRF 複合システムで共用されないデータ・セット

最良のパフォーマンスを得るために、1 つのシステムに固有の情報が入っているデータ・セットをローカル非共用 DASD に置き、これらのデータ・セットを別々のカタログに定義します。別々に保持するデータ・セットには、1 つのシステムに固有の情報、例えば、障害からのリカバリーを試行するときにシステムが必要とするデータなどが含まれています。831 ページの図 86 は、A および B というラベルの DASD 上のこれらのデータ・セットを示しています。これらは、2 つのシステムで共用されている DASD にも置くこともできます。

ある一定のデータ・セットを、共用されない別個のライブラリー・コピーとして保持してください。これは、障害点を減らすためです。このようなシステム・データ・セットは、次のとおりです。

- IMS.ACBLIBA(B)
- IMS.FORMATA(B)
- IMS.TFORMATA(B)
- IMS.MODBLKSA(B)
- IMS.JOBS
- IMS.PROCLIB
- IMS.SDFSRESL

重複させなければならないデータ・セット

一部のデータ・セットは、別個のデータ・セットでもなければなりません。このようなシステム・データ・セットは、次のとおりです。

- IMS.MSDBDUMP
- IMS.LGMSG
- IMS.SHMSG
- IMS.QBLKS
- IMS.LGMSGL
- IMS.SHMSGL
- IMS.QBLKSL
- IMS スプール・データ・セット
- SYSUDUMP および SYSABEND データ・セット

XRF 複合システム内でのオンライン変更の実行

XRF 複合システムでオンライン変更ユーティリティーを使用することもできます。このユーティリティーを使用して、IMS データ・セットの別個にされている重複コピーを保持する場合は、データ・セットの両方のコピーに対して同一の変更を行ってください、次に、/MODIFY PREPARE、/DISPLAY MODIFY、および /MODIFY COMMIT コマンドをアクティブ IMS で発行してください。

第 45 章 IMSplex でのリカバリー

IMSplex でのリカバリーは、メンバー IMS システム、Repository Server (RS)、および Common Service Layer (CSL) に対して異なる方法で実行されます。

以下のトピックでは、各タイプのリカバリーの概要を紹介します。

IMSplex での IMS システムのリカバリー

IMSplex 内で障害が発生した IMS システムのリカバリー手順は、スタンドアロンの IMS システムのリカバリー手順と同じです。例えば、z/OS の障害またはハードウェアの電源障害のために IMS システムが終了した場合の手順は、IMS システムが IMSplex 内のメンバーであってもなくても同じです。z/OS の IPL を実行し、/ERESTART コマンドを使用して IMS システムを再始動する必要があります。

IMSplex での CSL のリカバリー

CSL 全体で障害が起こった場合は、それぞれの CSL マネージャーを再始動する必要があります。IMSplex に 1 つの OM しか定義されていないときに OM の障害が起こった場合、タイプ 2 コマンド・インターフェース機能を失うことになり、SPOC を介してコマンドを入力できなくなります。それでもタイプ 1 コマンドは、サポートするソースから入力可能です。OM の再始動は、OM 実行パラメーターに定義されている z/OS 自動リスタート・マネージャー (ARM) を使用するか、開始済みプロシージャーを指定するか、あるいは JCL を実行依頼することにより行うことができます。

同様に、RM または SCI の障害が起こった場合、z/OS 自動リスタート・マネージャーを使用するか、開始済みプロシージャーを指定するか、あるいは JCL を実行依頼することにより、それぞれを再始動する必要があります。RM が使用できない場合、IMSplex は引き続き機能しますが、RM のすべての機能は RM が再始動されるまで終了したままです。すべての RM、リソース構造、カップリング・ファシリティ、または CQS の障害が起こるか、あるいはそれらが IMSplex 内でアクセス不能になった場合、グローバル・リソース管理はそれらがすべて再始動されるまで機能しません。すべての SCI で障害が起こった場合、通信がすべてダウンするので、CSL は操作不能になります。SCI の障害は、自動 RECON 損失通知にも影響を与えます。CSL を作動状態に戻すには、SCI を再始動する必要があります。

IMSplex での RS のリカバリー

IMSRSC リポジトリーは、定義されているリポジトリーを管理する 1 つのマスター RS によって管理されます。シスプレックス内には 1 つ以上の従属 RS を定義できます。



従属 RS は、マスター RS がシャットダウンしたことを識別するまで初期化状態で待機します。シャットダウンした時点で、すべての従属 RS は始動プロセスを完了して、マスター RS になろうとします。1 つの従属 RS が新しいマスター RS になります。それ以外は従属 RS のままになります。


マスター RS と従属 RS は、同じ z/OS システム間カップリング・ファシリティ (XCF) グループに属している必要があります。別の XCF グループで RS を始動した場合、それらは互いに独立しています。


従属 RS がマスター RS になろうとすると、FRP2003I メッセージが出されます。従属 RS が正常にマスター RS になると、FRP2002I メッセージと FRP2025I メッセージが出されます。

従属 RS の実装は、マスター RS が終了したときにできるだけ迅速に新しい接続を受け入れられるよう設計されています。ただし、従属 RS はマスター RS のシャドウを生成しません。RS クライアントは、登録中に、RS に登録出口を渡します。登録出口は、マスター RS がダウンしたときに RS 使用不可イベントによって駆動され、従属 RS がマスターになったときに使用可能イベントによって駆動されます。新しいマスターが使用可能になると、クライアントは新しい RS に登録し、使用中のすべてのリポジトリに再接続する必要があります。



関連概念:

-  [IMSRSC リポジトリの概要 \(システム定義\)](#)
-  [従属リポジトリ・サーバーの開始 \(オペレーションおよびオートメーション\)](#)

 [IMSRSC リポジトリおよび RS カタログ・リポジトリ・データ・セット \(システム定義\)](#)

 [IMSRSC リポジトリのオープン \(オペレーションおよびオートメーション\)](#)

関連情報:

-  [FRP2003I \(メッセージおよびコード\)](#)
-  [FRP2002I \(メッセージおよびコード\)](#)


第 5 部 IMS 報告書の使用

これらのトピックでは、IMS 報告書およびその使用方法について説明します。

第 46 章 DB モニター報告書

これらのトピックでは、DB モニター報告書とその使用方法について説明します。

関連概念:

 VSAM バッファアのモニター (データベース管理)

VSAM バッファア・プール報告書

VSAM バッファア・プール報告書は、特定の VSAM サブプールに関する処理情報を提供します。この報告書は、指定されたサブプールごとに 1 つ作成されます。

VSAM ローカル共用リソース・プール内のサブプールの場合は、報告書は、サブプール・バッファア・サイズに基づいて昇順で作成されます。索引サブプールとデータ・サブプールが共に、特定の共用リソース・プールに存在するときは、索引サブプール報告書は、データ・サブプール報告書に続いて作成されます。サブプールがいくつかの異なる共用リソース・プール内にあるときの報告書は、常に、共用リソース・プールが指定された順に作成されます。共用リソース・プールとサブプールは、IMS の初期設定時に処理される制御ステートメントで指定します。バッチ・システムでは、制御ステートメントは DFSVSAMP データ・セットに置かれます。オンライン・システムでは、DFSVMnn というメンバー名で IMS.PROCLIB データ・セットに置かれます。

HSAM または SHSAM データベースの場合は、どちらのデータベースもアクセス方式に VSAM を使用できないので、VSAM バッファア・プール報告書は何も意味がありません。

VSAM バッファア・プール報告書の使用方法

VSAM バッファア・プール報告書の主要な用途は、サブプールに対する読み書きのために入出力操作が必要であった回数を算出することです。

入出力操作の回数を減らすことができるかどうかを判別するには、バッファア・プールのサイズを大きくしてみます。また、バックグラウンド書き込みをオンにしておきます。あるいは、入出力操作の回数が時間の経過とともに増加するときは、データベースを再編成する必要があります。

バッファア・プールに対する読み書きに必要な入出力操作の回数を求めるには、次の式を使用してください。

入出力操作の総数は、次に挙げるものの合計です。

1. CI をデータベースからバッファアに読み込む必要があった回数 (この報告書の NUMBER OF CONTR INT READ FROM EXTERNAL STORAGE フィールド)
2. バッファアをデータベースに書き込む必要があった回数 (この報告書の NUMBER OF VSAM WRITES INITIATED BY IMS フィールド)

- 新しい CI をバッファに読み込めるようにするために、バッファをデータベースに書き込む必要があった回数 (この報告書の NUMBER OF VSAM WRITES TO MAKE SPACE IN THE POOL フィールド)

VSAM バッファ・プール報告書のフィールド

以下の例は、VSAM バッファ・プール報告書の例です。

VSAM BUFFER POOL			
		FIX INDEX/BLOCK/DATA	N/Y/N
		SHARED RESOURCE POOL ID	VPL/1
		SHARED RESOURCE POOL TYPE	D
		SUBPOOL ID	1
		SUBPOOL BUFFER SIZE	2048
		NUMBER HIPERSPACE BUFFERS	0
		TOTAL BUFFERS IN SUBPOOL	4
	17:08:15	17:10:16	
	START TRACE	END TRACE	DIFFERENCE
NUMBER OF RETRIEVE BY RBA CALLS RECEIVED BY BUF HNDLR	0	0	0
NUMBER OF RETRIEVE BY KEY CALLS	0	0	0
NUMBER OF LOGICAL RECORDS INSERTED INTO ESDS	0	0	0
NUMBER OF LOGICAL RECORDS INSERTED INTO KSDS	0	0	0
NUMBER OF LOGICAL RECORDS ALTERED IN THIS SUBPOOL	0	0	0
NUMBER OF TIMES BACKGROUND WRITE FUNCTION INVOKED	0	0	0
NUMBER OF SYNCHRONIZATION CALLS RECEIVED	18566	27923	9357
NUMBER OF WRITE ERROR BUFFERS CURRENTLY IN THE SUBPOOL	0	0	0
LARGEST NUMBER OF WRITE ERRORS IN THE SUBPOOL	0	0	0
NUMBER OF VSAM GET CALLS ISSUED	0	0	0
NUMBER OF VSAM SCHBFR CALLS ISSUED	0	0	0
NUMBER OF TIMES CONTR INT REQUESTED ALREADY IN POOL	229956	349375	119419
NUMBER OF CONTR INT READ FROM EXTERNAL STORAGE	1078	1229	151
NUMBER OF VSAM WRITES INITIATED BY IMS	33	64	31
NUMBER OF VSAM WRITES TO MAKE SPACE IN THE POOL	0	0	0
NUMBER OF VSAM READS FROM HIPERSPACE BUFFERS	0	0	0
NUMBER OF VSAM WRITES FROM HIPERSPACE BUFFERS	0	0	0
NUMBER OF FAILED VSAM READS FROM HIPERSPACE BUFFERS	0	0	0
NUMBER OF FAILED VSAM WRITES FROM HIPERSPACE BUFFERS	0	0	0

報告書の各フィールドの意味は次のとおりです。

FIX INDEX/BLOCK/DATA

このフィールドには、このサブプールの索引バッファ/データ・バッファ接頭部/データ・バッファに関する固定オプションが示されます。

SHARED RESOURCE POOL ID

このフィールドには、サブプールの定義時に指定された 4 文字の VSAM ローカル共用リソース・プール ID です。この ID は、DFSVSAMP または IMS.PROCLIB データ・セットの制御ステートメントで指定します。

SHARED RESOURCE POOL TYPE

このフィールドには、このサブプールに置かれているのが、索引バッファ (I) であるか、データ・バッファ (D) であるかが示されます。

SUBPOOL ID

このフィールドは、サブプール ID を示します。DFSVSAMP または IMS.PROCLIB データ・セットの制御ステートメントで指定された VSAM ローカル共用リソース・プール内の異なるデータベース・バッファ・サイズごとに、固有のサブプール ID が割り当てられています。

SUBPOOL BUFFER SIZE

このフィールドは、このサブプール内のバッファのサイズをバイト単位で示します。異なるサイズのバッファが指定できます。1 つのサブプールに、同じサイズのバッファがすべて入れられます。すべてのサブプールを 1 つにまと

めたものが、バッファ・プールです。バッファ・サイズは、DFSVSAMP または IMS.PROCLIB データ・セットの制御ステートメントで指定します。例えば、次のように指定したとします。

```
//DFSVSAMP DD input for VSAM and OSAM buffers and options
:
POOLID=BB
VSRBF=4096,4,D,HSR,10
VSRBF=8192,4,D,HSO,20
POOLID=FF
VSRBF=4096,4,I
VSRBF=8192,4,D
:
/*
```

これで 4 つのサブプールと、以下の例に示す 4 つの VSAM バッファ・プール報告書が得られます。各報告書表示の見出しは、例に示すとおりです。

```
SHARED RESOURCE POOL ID:   BB
SHARED RESOURCE POOL TYPE:  D
SUBPOOL ID:                 1
SUBPOOL BUFFER SIZE:        4096
HIPERSPACE BUFFERS:         10
TOTAL BUFFERS IN SUBPOOL:   4

SHARED RESOURCE POOL ID:   BB
SHARED RESOURCE POOL TYPE:  D
SUBPOOL ID:                 2
SUBPOOL BUFFER SIZE:        8192
HIPERSPACE BUFFERS:         20
TOTAL BUFFERS IN SUBPOOL:   4

SHARED RESOURCE POOL ID:   FF
SHARED RESOURCE POOL TYPE:  D
SUBPOOL ID:                 1
SUBPOOL BUFFER SIZE:        8192
HIPERSPACE BUFFERS:         0
TOTAL BUFFERS IN SUBPOOL:   4

SHARED RESOURCE POOL ID:   FF
SHARED RESOURCE POOL TYPE:  I
SUBPOOL ID:                 2
SUBPOOL BUFFER SIZE:        4096
HIPERSPACE BUFFERS:         0
TOTAL BUFFERS IN SUBPOOL:   4
```

NUMBER OF HIPERSPACE BUFFERS

このフィールドには、サブプールの定義時に指定されたハイパースペース・バッファの数が示されます。

TOTAL BUFFERS IN SUBPOOL

このフィールドは、指定されたサブプール内にいくつのバッファがあるかを示します。

START TRACE、END TRACE、および DIFFERENCE

START TRACE フィールドと END TRACE フィールドには、DB モニター・プログラムが最後に 始動された時刻と、終了された時刻が示されます。この時刻は時刻機構によって生成されます。時刻は次のように示されます。

Clock time = hh.mm.ss

ここで、

hh 0 時から 23 時まで

mm 分

ss 秒

DB モニター・プログラムがバッチ実行の全期間を通じてオンになっていたときは、トレース開始時刻とトレース終了時刻は、バッチ実行が開始された時刻と終了された時刻です。DB モニター・プログラムが同じバッチ実行で 2 回以上オン/オフにされたときは、トレース開始時刻とトレース終了時刻は、モニターが最後に 始動された時刻と停止された時刻です。

START TRACE フィールドと END TRACE フィールドの数字は、バッチ実行の全期間に累積された数です。モニターが一度だけオン/オフにされたときは、START TRACE の数はゼロになっています。モニターが 2 回以上オン/オフにされたときは、START TRACE の数は、モニターが最後にオンにされたときの累積数であり、STOP TRACE の数は最後にオフにされたときの累積数です。

DIFFERENCE 欄は、START TRACE フィールドと END TRACE フィールドの累積数の差を示します。モニターが 1 回だけオン/オフにされたときは、DIFFERENCE 欄には、END TRACE 欄と同じ数が示されます。

NUMBER OF RETRIEVE BY RBA CALLS

このフィールドは、相対バイト・アドレス (RBA) による検索呼び出しがこのサブプールに対して実行された回数を示します。RBA による検索呼び出しは、DL/I が内部で出す呼び出しです。あるセグメントを検索するときにとどっていく直接アドレス・ポインターごとに、RBA による検索呼び出しが 1 回ずつ出されます。例えば、HDAM または PHDAM データベースに置かれている従属セグメントに対する GN 呼び出しでは、その従属セグメントを検索するために一連の RBA 呼び出しが使用され、その従属セグメントがたどっていく直接アドレス・ポインターごとに 1 回の呼び出しが出されます。

RBA による検索呼び出しが使用されたときの正確な検索順序を知るには、バッファ・ハンドラー・トレースをオンにし、SNAP 呼び出しを使用してトレース・レコードを見ることによって、その順序を記録することができます。バッファ・ハンドラー・トレースをオンにするには、DFSVSAMP または DFSVSMnn データ・セットの OPTIONS ステートメントの DL/I= オペランドを使用します。SNAP 呼び出しはアプリケーション・プログラムから出すこともできますが、DFSDDLT0 テスト・プログラムを使用することによって出すこともできます。

アプリケーション・プログラムから呼び出しを 1 回出すと、RBA による検索呼び出しが 2 つ以上生成される場合があります。RBA による検索呼び出しは、入出力操作を必要とする場合と、必要としない場合があります。このフィールドの数は、セグメントにアクセスするときの入出力操作の回数を反映していないので、この数から VSAM のパフォーマンスを判断しないでください。

NUMBER OF RETRIEVE BY KEY CALLS

このフィールドは、キーによる検索呼び出しがこのサブプールに対して実行された回数を示します。キーによる検索呼び出しは、DL/I が内部で出す呼び出しです。この呼び出しは、キーを修飾として使用して KSDS を検索するために出されます (この場合、キーは X に等しいか、それより大きい)。例えば、HIDAM データベースに置かれたルート・セグメントで GU 呼び出しが出されると、DL/I は、要求されたルート・セグメントを指している索引セグメントにアクセスするために、キーによる検索呼び出しを出します。

アプリケーション・プログラムから呼び出しを 1 回出すと、キーによる検索呼び出しが 2 つ以上生成される場合があります。キーによる検索呼び出しは、入出力操作を必要とする場合と、必要としない場合とがあります。このフィールドの数は、セグメントにアクセスするときの入出力操作の回数を反映していないので、この数から VSAM のパフォーマンスを判断しないでください。

NUMBER OF LOGICAL RECORDS INSERTED INTO ESDS

このフィールドは、以前空だったが、今はセグメントが含まれている ESDS 内の論理レコード数を示します。従属セグメントが HISAM または HIDAM データベースに置かれた ESDS に挿入される時、既に他のセグメントが入っている論理レコードにその従属セグメントが収まらない場合があります。このような場合は、そのセグメントは新しい ESDS 論理レコードに入れられます。従属セグメントが HISAM データベースに置かれた ESDS 論理レコードに挿入される時、同じ論理レコードに入っている他のセグメントを新しい ESDS 論理レコードに移して、挿入するセグメントのためのスペースを作る必要が出てくる場合があります。

報告書ごとにこのフィールドを調べてください。そうすれば、割り振られた 1 次スペース内の論理レコードのなくなる時期を判別できます。2 次スペースは 1 次スペースの近くにないことがあるので、2 次スペースから論理レコードを使用するのを避けることが、最良の方法です。2 つの区域のスペース間が離れていると、シーク時間が余分にかかるので、パフォーマンスが低下します。一般に、2 次スペースを使用する必要が起こる前に、データベースを再編成しておくのが最良の方法です。

NUMBER OF LOGICAL RECORDS INSERTED INTO KSDS

このフィールドは、以前空だったが、今はセグメントが含まれている KSDS 内の論理レコード数を示します。HISAM データベースでは、ルート・セグメントが挿入される時、新しい論理レコードが使用されます。HIDAM 索引データベースでは、ルート・セグメントが挿入される時作成される索引セグメントには、新しい論理レコードが使用されます。副次索引データベースでは、新しいポインター・セグメントが挿入される時、新しい論理レコードが使用されます。

報告書ごとにこのフィールドを調べてください。そうすれば、割り振られた 1 次スペース内の論理レコードのなくなる時期を判別できます。2 次スペースは 1 次スペースの近くにないことがあるので、2 次スペースから論理レコードを使用するのを避けることが、最良の方法です。2 つの区域のスペース間が離れていると、シーク時間が余分にかかるので、パフォーマンスが低下します。一般に、2 次スペースを使用する必要が起こる前に、データベースを再編成しておくのが最良の方法です。

NUMBER OF LOGICAL RECORDS ALTERED IN THIS SUBPOOL

このフィールドは、バッファ・プールに入っている間に変更済みのマークが付けられた論理レコードの数を示します。論理レコードでセグメントの挿入または置き換えが行われると、バッファ内でその論理レコードには、それがデータベースに書き戻されるまで変更のマークが付けられています。

NUMBER OF TIMES BACKGROUND WRITE FUNCTION INVOKED

バックグラウンド書き込み機能を使用することが指定されている場合、このフィールドには、その機能が使用された回数が見られます。バックグラウンド書き込み機能は、変更されたデータが入っているバッファを、時々、データベースに書き戻します。このようにするのは、アプリケーション・プログラムがバッファ

ーを必要とするとき、使用できるようにしておくためです。バックグラウンド書き込み機能を使用しないと、アプリケーション・プログラムは、変更データが既に入っているバッファーにデータを読み込もうとした場合に、そのバッファーの内容がデータベースに書き戻されるまで待つ必要があります。バックグラウンド書き込みが呼び出される回数は、ある特定のローカル共用リソース・プールについて、モニターが特定の期間実行されている間に作成されるどのサブプール報告書の場合も同じです。バックグラウンド書き込み機能が一度呼び出されると、ローカル共用リソース・プール内のすべてのサブプールからバッファーが書き込まれます。

バックグラウンド書き込みは、DFSVSAMP または DFSVSMnn データ・セットの OPTIONS ステートメントの BGWRT= オペランドに指定します。

NUMBER OF SYNCHRONIZATION CALLS RECEIVED

このフィールドは、モニターがオンになっていた間に、ローカル共用リソース・プール内のすべての変更済みバッファーを書き込む要求が出された回数を示します。この要求を行う代表的な例として、CHKP 呼び出しと STAT 呼び出しがあります。

NUMBER OF PERM WRT ERROR BUFFS NOW IN THE SUBPOOL

このフィールドは、バッファーをデータベースへの書き込み中に永続入出力エラーが起こったために、現在ストレージ内で「凍結」されているバッファーの数を示します。VSAM 書き込み操作が原因で永続入出力エラーが起こったときは、その影響を受けたバッファーは、データ・セットがクローズされるまで、あるいはオンライン・システムでは、システムがシャットダウンされるまで、ストレージに凍結されます。データ・セットがクローズされるか、システムがシャットダウンされると、これらのバッファーはログに書き込まれ、このフィールドの値は再びゼロになります。

この報告を受ける前に、オペレーターが、影響を受けたデータ・セットのデータベース・リカバリーを行っていたかどうかを確かめてください。

LARGEST NUMB OF PERM ERR BUFFS EVER IN THE SUBPOOL

このフィールドは、前のフィールドで説明した状態が起こったときにストレージ内で凍結していた バッファーの数を示します。

NUMBER OF VSAM GET CALLS ISSUED

このフィールドは、VSAM GET 呼び出しが出された回数を示します。VSAM GET 呼び出しは、DL/I が内部で出す呼び出しです。GET 呼び出しは、バッファー・プールにあるデータで間に合う場合と、データをバッファー・プールに読み込む必要がある場合があります。このフィールドの数は、セグメントにアクセスするのに必要な入出力操作の回数を反映していないので、この数から VSAM のパフォーマンスを判断しないでください。

NUMBER OF VSAM SCHBFR CALLS ISSUED

このフィールドは、HD スペース管理ルーチンが、セグメントを挿入するスペースを探すための呼び出しを実行した回数を示します。

モニター報告書ごとにこのフィールドの数が大きくなっていくということは、新しいセグメントを保管するためのスペースが最も望ましい位置にないことを意味します。最終的には、パフォーマンスを向上させるために、データベースを再編成する必要があります。再編成では、データベース・スペースに影響を及ぼすオペランド (DBD ステートメントの RMNAME= キーワードの BYTES オペラ

ンドと、DATA SET ステートメントの FRSPC= キーワードの fbff と fspf オペランド) に特別な注意を払う必要があります。

NUMBER OF TIMES CONT INT REQUESTED ALREADY IN POOL

このフィールドは、既にバッファに入っている CI で論理レコードが検出された回数を示します。このようなときは、必要とするセグメントにアクセスするための入出力操作は必要ありません。

パフォーマンスを向上させるには、割り振っておいたバッファの数を増やしてください。バッファの数を増やした場合、このフィールドに注目して、その数が大きくなったかを調べれば、パフォーマンスが向上したことが分かります。

NUMBER OF CONTR INT READ FROM EXTERNAL STORAGE

このフィールドは、既にバッファに入っている CI で論理レコードが検出されなかった回数を示します。このようなときは、その論理レコードが入っている CI をバッファ・プールに読み込むための入出力操作が必要になります。行われる入出力操作が少なければ少ないほど、パフォーマンスは向上するので、指定しておいたバッファの数を増やすことにより、このフィールドの数がどのように変化するかを調べることができます。指定するバッファ数が増えるにつれて、バッファ・プール内の CI (および論理レコード) の数が増えます。ただし、この処理には損益分岐点があります。つまり、損益分岐点では、指定するバッファが多くなり過ぎるため、CI をバッファに読み込む時間よりも、バッファを検索し、保持しておく時間の方が長くなります。

バッファの数は、DFSVSAMP または DFSVSMnn データ・セットの制御ステートメントで指定します。

NUMBER OF VSAM WRITES INITIATED BY IMS

このフィールドは、データベースにデータを書き込むための書き込み要求を DL/I が出した回数を示します。書き込み操作の要求は、次に挙げるようなときに出されます。

- データ・セットがクローズされる時。クローズするデータ・セットで変更されたデータが入っているデータベース・バッファがデータベースに書き込まれます。
- アプリケーション・プログラム処理の途中で異常終了が起こった時。変更されたデータが入っているデータベース・バッファがデータベースに書き込まれます。
- バックグラウンド書き込み機能が呼び出された時。変更されたデータが入っているデータベース・バッファが選択的にデータベースに書き込まれます。
- チェックポイント呼び出しが出された時。変更されたデータベース・バッファがすべてデータベースに書き込まれます。

NUMBER OF VSAM WRITES TO MAKE SPACE IN THE POOL

このフィールドは、変更されたデータを持つ論理レコードが入っているバッファに CI を読み込まなければならなかった回数を示します。このようなときは、バッファ (ここには、変更されたデータが入っているため) をデータベースに書き戻しておかないと、新しい CI をそのバッファに読み込むことができません。このことは、アプリケーション・プログラムが、書き込み操作が行われている間、待たなければならないことを意味します。

パフォーマンスを最良にするためには、このフィールドがゼロに近い数になるようにしてください。一般に、バッチ処理のときにバックグラウンド書き込み機能をオンにしておき、割り振っておいたバッファの数を調整すれば、そのようにすることができます。

NUMBER OF VSAM READS FROM HIPERSPACE BUFFERS

このフィールドは、ハイパースペース・バッファからの正常な VSAM 読み取り (MOVEPAGE および NON-MOVEPAGE) の総数を示します。

NUMBER OF VSAM WRITES FROM HIPERSPACE BUFFERS

このフィールドは、ハイパースペース・バッファへの正常な VSAM 書き込み (MOVEPAGE および NON-MOVEPAGE) の総数を示します。

NUMBER OF FAILED VSAM READS FROM HIPERSPACE BUFFERS

このフィールドは、ハイパースペースからの VSAM 読み取り要求が失敗したために DASD からの読み取りが行われた回数を示します。

NUMBER OF FAILED VSAM WRITES FROM HIPERSPACE BUFFERS

このフィールドは、ハイパースペースへの VSAM WRITE 要求が失敗したために DASD への書き込みが行われた回数を示します。

VSAM 統計報告書

VSAM 統計報告書は、特定のアプリケーション・プログラム PCB、アプリケーション・プログラムが使用しているデータ・セット、およびアプリケーション・プログラムが発行した DL/I 呼び出しのタイプに基づいています。

HSAM または SHSAM データベースの場合は、どちらのデータベースもアクセス方式として VSAM を使用できないので、VSAM 統計報告書は何も意味がありません。

VSAM 統計報告書の使用方法

VSAM 統計報告書からは、アプリケーション・プログラムの中のどの呼び出しが、非常に多くの入出力操作を必要としているかを知ることができます。それが分かれば、入出力操作を減らすようにデータベースかアプリケーション・プログラムのどちらかをチューニングすることにより、パフォーマンスを向上させることができます。

この報告書の次のフィールドには、実際に行われた入出力アクティビティが示されるので、最も重要なフィールドとして注目すべきです。

- READS
- USR WTS
- NUR WTS

これらのフィールドの平均値を減らすことができれば、パフォーマンスを向上させることができます。

入出力操作を減らすようにチューニングを行うとき、数多く出された呼び出しに最大限の注意を払ってください。2000 回実行される呼び出しで 1 秒ずつ節約した方が、10 回出される呼び出しで 5 秒ずつ節約するよりも大きい利益が得られます

(2000 秒対 50 秒)。DL/I 呼び出し要約報告書 (857 ページの『DL/I 呼び出し要約報告書』の項を参照) には、各 DL/I アプリケーション・プログラム呼び出しが出された回数が見られます。

VSAM 統計報告書のフィールド

以下の例は、VSAM 統計報告書の例です。

```

IMS MONITOR   ****VSAM STATISTICS****   TRACE START 1989 076 12:42:54   TRACE STOP 1989 076 12:43:07   PAGE 0012

```

PCBNAME	DDNAME	DL/I FUNC	VSAM IWAITS	RET RBA	RET KEY	ISRT ESDS	ISRT KSDS	BFR ALT	BKG WTS	SYN PTS	GETS	SCHBFR	FOUND	READS	USR WTS	NUR WTS
DLVNTZ02	DBHVSAM2	STAT	1	1.00	0.00	0.00	0.00	1.00	0.00	1.00	1.00	0.00	1.00	0.00	1.00	0.00
	DD TOTAL		1	1.00	0.00	0.00	0.00	1.00	0.00	1.00	1.00	0.00	1.00	0.00	1.00	0.00
	HIDAM	STAT	4	307.25	36.50	0.50	0.00	35.00	0.00	0.25	120.25	0.00	157.00	0.25	0.75	0.00
	DD TOTAL		4	307.25	36.50	0.50	0.00	35.00	0.00	0.25	120.25	0.00	157.00	0.25	0.75	0.00
	XDLBT04I	GU	2	15.00	4.50	0.00	0.00	0.00	0.00	0.00	17.50	0.00	21.50	1.00	0.00	0.00
	DD TOTAL		2	15.00	4.50	0.00	0.00	0.00	0.00	0.00	17.50	0.00	21.50	1.00	0.00	0.00
PCB TOTAL																
			7	180.00	22.14	0.28	0.00	20.14	0.00	0.28	73.85	0.00	96.00	0.42	0.57	0.00
DLVNTZ02	HIDAM	GN	1	15.00	3.00	0.00	0.00	0.00	0.00	0.00	18.00	0.00	20.00	1.00	0.00	0.00
	GU		1	1.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00	0.00
	DD TOTAL		2	8.00	1.50	0.00	0.00	0.00	0.00	0.00	9.50	0.00	10.00	1.00	0.00	0.00
	DBHVSAM1	GU	8	0.00	0.12	12.75	520.00	7.50	0.00	0.00	0.12	0.00	0.00	0.87	0.00	0.00
	DD TOTAL		8	0.00	0.12	12.75	520.00	7.50	0.00	0.00	0.12	0.00	0.00	0.87	0.00	0.00
	XDLBT04I	GU	6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00
	DD TOTAL		6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00
	DBHVSAM2	GU	3	0.66	0.00	0.00	0.00	0.00	0.00	0.00	0.66	0.00	0.00	1.00	0.00	0.00
	DD TOTAL		3	0.66	0.00	0.00	0.00	0.00	0.00	0.00	0.66	0.00	0.00	1.00	0.00	0.00
PCB TOTAL																
			19	0.94	0.21	5.36	482.10	3.15	0.00	0.00	1.15	0.00	1.05	0.94	0.00	0.00
BATCH TOTAL																
			26	49.15	6.11	3.84	83.07	3.11	0.00	0.07	20.73	0.00	26.61	0.80	0.15	0.00

報告書の各フィールドの意味は次のとおりです。

TRACE START および TRACE STOP

TRACE START フィールドと TRACE STOP フィールドには、DB モニター・プログラムが最後に始動された時刻と、停止された時刻が示されます。この時刻は時刻機構によって生成されます。時刻は次のように示されます。

Clock time = hh.mm.ss

ここで、

hh 0 時から 23 時まで

mm 分

ss 秒

DB モニターがバッチ実行の全期間にわたってオンになっていたときは、トレース開始時刻とトレース終了時刻は、バッチ実行が開始された時刻と停止された時刻です。DB モニターが同じバッチ実行で 2 回以上オン/オフにされたときは、トレース開始時刻とトレース終了時刻は、モニターが最後に始動された時刻と停止された時刻です。

PCBNAME

このフィールドは、報告書が情報を提供している PCB の名前を示します。各ア

アプリケーション・プログラムは、1 つ以上の PCB を持っていることに注意してください。 PCB 名が各アプリケーション・プログラムに固有のものである場合、報告書に記載されている情報がどのアプリケーション・プログラムに関するものであるかを知るためにのみ、このフィールドを使用できます。

DDNAME

このフィールドは、アプリケーション・プログラムが使用しているデータ・セットの名前を示します。1 つの報告書内で、アプリケーション・プログラム (PCB) による複数のデータ・セットへのアクセスが示される場合があります。コンパイルされた統計は、データ・セット別にリストされます。

DL/I FUNC

このフィールドは、アプリケーション・プログラムが実行した DL/I 呼び出しのタイプを示します。

VSAM IWAITS

このフィールドは、特定のデータ・セットに対する DL/I 呼び出しのタイプ別に、IMS が処理を続行できるまで待機しなけりなならなかつた回数を示します。IMS が待たなければならぬケースでは、ほとんどの場合、入出力操作が行われるのを待つこととなります。つまり、データがデータベースからバッファへ読み取られるか、あるいはバッファからデータベースに書き戻されるまで待ちます。

報告書の残りのフィールドの各欄の数字は、いずれも平均値です。これらの欄に示されるのは、特定の回数ではなく、あるアクティビティが行われた回数の平均値です。平均値は待ちに関するものです。これらの数字には、切り捨てが行われています。例えば、値が 0.019 であると、0.01 と印刷されます。

RET RBA

このフィールドは、相対バイト・アドレス (RBA) による検索呼び出しがこのサブプールに対して実行された回数を示します。RBA による検索呼び出しは、DL/I が内部で出す呼び出しです。あるセグメントを検索するときにとどっていく直接アドレス・ポインターごとに、RBA による検索呼び出しが 1 回ずつ出されます。例えば、HDAM または PHDAM データベースに置かれている従属セグメントに対する GN 呼び出しでは、その従属セグメントを検索するために一連の RBA 呼び出しが使用され、その従属セグメントがたどっていく直接アドレス・ポインターごとに 1 回の呼び出しが出されます。

アプリケーション・プログラムから呼び出しを 1 回出すと、RBA による検索呼び出しが 2 つ以上生成される場合があります。RBA による検索呼び出しは、入出力操作を必要とする場合と、必要としない場合とがあります。このフィールドの数は、セグメントにアクセスするときの入出力操作の回数を反映していませんので、この数から VSAM のパフォーマンスを判断しないでください。

RET KEY

このフィールドは、キーによる検索呼び出しがこのサブプールに対して実行された回数を示します。キーによる検索呼び出しは、DL/I が内部で出す呼び出しです。この呼び出しは、キーを修飾として使用して KSDS を検索するために出されます (この場合、キーは X に等しいか、それより大きい)。例えば、HIDAM または PHIDAM データベースに置かれたルート・セグメントで GU 呼び出しが出されると、DL/I は、要求されたルート・セグメントを指している索引セグメントにアクセスするために、キーによる検索呼び出しを出します。

アプリケーション・プログラムから呼び出しを 1 回出すと、キーによる検索呼び出しが 2 つ以上生成される場合があります。キーによる検索呼び出しは、入出力操作を必要とする場合と、必要としない場合とがあります。このフィールドの数は、セグメントにアクセスするときの入出力操作の回数を反映していないので、この数から VSAM のパフォーマンスを判断しないでください。

ISRT ESDS

このフィールドは、以前空だったが、今はセグメントが含まれている ESDS 内の論理レコード数を示します。従属セグメントが HISAM、HIDAM、または PHIDAM データベースに置かれた ESDS に挿入されるとき、既に他のセグメントが入っている論理レコードにその従属セグメントが収まらない場合があります。このような場合は、そのセグメントは新しい ESDS 論理レコードに入れられます。従属セグメントが HISAM データベースに置かれた ESDS 論理レコードに挿入されるとき、同じ論理レコードに入っている他のセグメントを新しい ESDS 論理レコードに移して、挿入するセグメントのためのスペースを作る必要が出てくる場合があります。

報告書ごとにこのフィールドを調べてください。そうすれば、割り振られた 1 次スペース内の論理レコードのなくなる時期を判別できます。2 次スペースは 1 次スペースの近くにないことがあるので、2 次スペースから論理レコードを使用するのを避けることが、最良の方法です。

ISRT KSDS

このフィールドは、以前空だったが、今はセグメントが含まれている KSDS 内の論理レコード数を示します。HISAM データベースでは、ルート・セグメントが挿入されるとき、新しい論理レコードが使用されます。HIDAM および PHIDAM 索引データベースでは、ルート・セグメントが挿入されるときに作成される索引セグメントには、新しい論理レコードが使用されます。

報告書ごとにこのフィールドを調べてください。そうすれば、割り振られた 1 次スペース内の論理レコードのなくなる時期を判別できます。2 次スペースは 1 次スペースの近くにないことがあるので、2 次スペースから論理レコードを使用するのを避けることが、最良の方法です。2 つの区域のスペース間が離れていると、シーク時間が余分にかかるので、パフォーマンスが低下します。一般に、2 次スペースを使用する必要が起る前に、データベースを再編成しておくのが最良の方法です。

BFR ALT

このフィールドは、バッファ・プールに入っている間に変更済みのマークが付けられた論理レコードの数を示します。論理レコードでセグメントの挿入または置き換えが行われると、バッファ内でその論理レコードには、それがデータベースに書き戻されるまで変更のマークが付けられています。

BKG WTS

バックグラウンド書き込み機能を使用することが指定されている場合、このフィールドには、その機能が使用された回数が見られます。バックグラウンド書き込み機能は、変更されたデータが入っているバッファを、時々、データベースに書き戻します。このようにするのは、アプリケーション・プログラムがバッファを必要とするとき、使用できるようにしておくためです。バックグラウンド書き込み機能を使用しないと、アプリケーション・プログラムは、変更データが既に入っているバッファにデータを読み込もうとした場合に、そのバッファの内容がデータベースに書き戻されるまで待つ必要があります。バックグラウン

ド書き込みが呼び出される回数は、モニターが特定の期間実行されている間に作成されるどのサブプール報告書の場合も同じです。これは、バックグラウンド書き込み機能が一度呼び出されると、すべてのサブプールからバッファーが書き込まれるためです。

バックグラウンド書き込みは、DFSVSAMP または DFSVSMnn データ・セットの OPTIONS ステートメントの BGWRT= オペランドに指定します。

SYN PTS

このフィールドには、モニターがオンになっている間に、チェックポイント呼び出しが DL/I プログラムで出された回数が見られます。

GETS

このフィールドは、VSAM GET 呼び出しが出された回数を示します。VSAM GET 呼び出しは、DL/I が内部で出す呼び出しです。GET 呼び出しは、バッファー・プールにあるデータで間に合う場合と、データをバッファー・プールに読み込む必要がある場合があります。このフィールドの数は、セグメントにアクセスするのに必要な入出力操作の回数を反映していないので、この数から VSAM のパフォーマンスを判断しないでください。

SCHBFR

このフィールドは、HD スペース管理ルーチンが、セグメントを挿入するスペースを探すための呼び出しを実行した回数を示します。

モニター報告書ごとにこのフィールドの数が大きくなっていくということは、新しいセグメントを保管するためのスペースが最も望ましい位置にないことを意味します。最終的には、パフォーマンスを向上させるために、データベースを再編成する必要が出てきます。再編成では、データベース・スペースに影響を及ぼすオペランド (DBD ステートメントの RMNAME= キーワードの BYTES オペランドと、DATA SET ステートメントの FRSPC= キーワードの fbff と fspf オペランド) に特別な注意を払う必要があります。

FOUND

このフィールドは、既にバッファーに入っている CI で論理レコードが検出された回数を示します。このようなときは、必要とするセグメントにアクセスするための入出力操作は必要ありません。

パフォーマンスを向上させるには、割り振っておいたバッファーの数を増やしてください。バッファーの数を増やした場合、このフィールドに注目して、その数が大きくなったかを調べれば、パフォーマンスが向上したことがわかります。

READS

このフィールドは、既にバッファーに入っている CI で論理レコードが検出されなかった回数を示します。このようなときは、その論理レコードが入っている CI をバッファー・プールに読み込むための入出力操作が必要になります。行われる入出力操作が少なければ少ないほど、パフォーマンスは向上するので、指定しておいたバッファーの数を増やすことにより、このフィールドの数がどのように変化するかを調べることができます。指定するバッファー数が増えるにつれて、バッファー・プール内の CI (および論理レコード) の数が増えます。ただし、この処理には損益分岐点があります。つまり、損益分岐点では、指定するバッファーが多くなり過ぎるため、CI をバッファーに読み込む時間よりも、バッファーを検索し、保持しておく時間の方が長くなります。

バッファの数は、DFSVSAMP または DFSVSMnn データ・セットの制御ステートメントで指定します。

USR WTS

ユーザー書き込みを示すこのフィールドには、データをデータベースに書き込むために DL/I から書き込み要求が出された回数が見られます。書き込み操作の要求は、次に挙げるようなときに出されます。

- データ・セットがクローズされる時。クローズするデータ・セットで変更されたデータが入っているデータベース・バッファがデータベースに書き込まれます。
- アプリケーション・プログラム処理の途中で異常終了が起こった時。変更されたデータが入っているデータベース・バッファがデータベースに書き込まれます。
- バックグラウンド書き込み機能が呼び出された時。変更されたデータが入っているデータベース・バッファが選択的にデータベースに書き込まれます。
- チェックポイント呼び出しが出された時。変更されたデータベース・バッファがすべてデータベースに書き込まれます。

NUR WTS

非ユーザー書き込みを示すこのフィールドには、変更されたデータを持つ論理レコードが入っているバッファに CI を読み込む必要があった回数が見られます。このようなときは、バッファ（ここでは、変更されたデータが入っているため）をデータベースに書き戻しておかないと、新しい CI をそのバッファに読み込むことができません。このことは、アプリケーション・プログラムが、書き込み操作が行われている間、待たなければならないことを意味します。

パフォーマンスを最良にするためには、このフィールドがゼロに近い数になるようにしてください。一般に、バッチ処理のときにバックグラウンド書き込み機能をオンにしておき、割り振っておいたバッファの数を調整すれば、そのようにすることができます。

SCRS

このフィールドは、ハイパースペース・バッファからの正常な VSAM 読み取り (MOVEPAGE および NON-MOVEPAGE) の総数を示します。

SCWS

このフィールドは、ハイパースペース・バッファへの正常な VSAM 書き込み (MOVEPAGE および NON-MOVEPAGE) の総数を示します。

SCRF

このフィールドは、ハイパースペースからの VSAM 読み取り要求が失敗したために DASD からの読み取りが行われた回数を見られます。

SCWF

このフィールドは、ハイパースペースへの VSAM 書き込み要求が失敗したために、DASD への書き込みが行われた回数を見られます。

DD TOTAL

このフィールドには、特定のデータ・セットについて、あるアクティビティーが行われた総平均回数が見られます（ただし、VSAM CALLS フィールドの場合は、総回数が見られます）。

PCB TOTAL

このフィールドには、特定の PCB について、あるアクティビティが行われた総平均回数が見られます (ただし、VSAM CALLS フィールドの場合は、総回数が示されます)。

BATCH TOTAL

このフィールドには、モニターが稼働していた時間の間に、あるアクティビティが行われた総平均回数が見られます (ただし、VSAM CALLS フィールドの場合は、総回数が示されます)。

データベース・バッファ・プール報告書

データベース・バッファ・プール報告書からは、処理時の OSAM サブプールに関する情報を知ることができます。バッファ・プール内のすべてのサブプールについての報告書が1つ作成されます。

VSAM 報告書は、バッファ・プール内のサブプールごとに作成される点で、この報告書とは異なります。

HSAM、SHSAM、SHISAM、または GSAM データベースの場合は、どのデータベースもアクセス方式として OSAM を使用できないので、データベース・バッファ・プール報告書は何も意味がありません。

データベース・バッファ・プール報告書の使用方法

データベース・バッファ・プール報告書の主要な用途は、OSAM バッファ・プールに対する読み書きのために入出力操作が必要であった回数を算出することです。

入出力操作の回数を減らすことができるかどうかを判断するには、バッファ・プールのサイズを大きくしてみます。あるいは、この回数が時間の経過とともに増加するときは、データベースを再編成する必要があります。

バッファ・プールに対する読み書きに必要な入出力操作の回数を求めるには、次の式を使用してください。

入出力操作の総数は、次に挙げるものの合計です。

1. データベースから読み取られたブロック

OSAM 特定要求を満たすために読み取られたブロックの数 (出された読み取り要求の数)

2. データベースに書き込まれたブロック

バッファ・スチール処理と除去処理のために書き込まれたブロックの数 (書き込まれたブロックの数)

データベース・バッファ・プール報告書のフィールド

以下の図は、データベース・バッファ・プール報告書の例です。

DATA BASE BUFFER POOL

FIX PREFIX/BUFFERS	Y/Y
SUBPOOL ID	004K
SUBPOOL BUFFER SIZE	4096

	TOTAL BUFFERS IN SUBPOOL		1000
	17:08:15	17:10:16	
NUMBER OF LOCATE-TYPE CALLS	1117674	1676213	558539
NUMBER OF REQUESTS TO CREATE NEW BLOCKS	0	0	0
NUMBER OF BUFFER ALTER CALLS	215874	322936	107062
NUMBER OF PURGE CALLS	25077	37454	12377
NUMBER OF LOCATE-TYPE CALLS, DATA ALREADY IN OSAM POOL	870306	1301187	430881
NUMBER OF BUFFERS SEARCHED BY ALL LOCATE-TYPE CALLS	1258247	1886843	628596
NUMBER OF READ I/O REQUESTS	238165	360260	122095
NUMBER OF SINGLE BLOCK WRITES BY BUFFER STEAL ROUTINE	0	0	0
NUMBER OF BLOCKS WRITTEN BY PURGE	95057	142413	47356
NUMBER OF LOCATE CALLS WAITED DUE TO BUSY ID	780	1297	517
NUMBER OF LOCATE CALLS WAITED DUE TO BUFFER BUSY WRT	0	0	0
NUMBER OF LOCATE CALLS WAITED DUE TO BUFFER BUSY READ	0	0	0
NUMBER OF BUFFER STEAL/PURGE WAITED FOR OWNERSHIP RLSE	178	261	83
NUMBER OF BUFFER STEAL REQUESTS WAITED FOR BUFFERS	0	0	0
TOTAL NUMBER OF I/O ERRORS FOR THIS SUBPOOL	0	0	0
NUMBER OF BUFFERS LOCKED DUE TO WRITE ERRORS	0	0	0

報告書の各フィールドの意味は次のとおりです。

FIX PREFIX/BUFFERS

このフィールドには、このサブプールのバッファ接頭部/データ・バッファに関する固定オプションが示されます。

SUBPOOL ID

このフィールドは、サブプールの定義時に指定された 4 文字のプール ID です。

SUBPOOL BUFFER SIZE

このフィールドには、このサブプール内のバッファのサイズがバイト数で示されます。

TOTAL BUFFERS IN SUBPOOL

このフィールドには、このサブプール内のバッファの総数が示されます。

次の行には、トレース開始時刻とトレース終了時刻を示す時刻項目があります。

START TRACE フィールドと END TRACE フィールドには、DB モニター・プログラムが最後に開始された時刻と、終了された時刻が示されます。

NUMBER OF LOCATE-TYPE CALLS

このフィールドには、このサブプールの探索タイプ呼び出しの数が示されます。

NUMBER OF REQUESTS TO CREATE NEW BLOCKS

このフィールドには、ブロックにセグメントが初めて挿入された回数が見られます。ブロックに初めてセグメントが挿入されると、該当ブロックには変更のマークが付けられます。また、このブロックは最終的に、データベースに書き戻されなければなりません。

NUMBER OF BUFFER ALTER CALLS

このフィールドには、このサブプールのバッファ変更呼び出しの数が示されます。このカウントには、NEW BLOCK 呼び出しと BYTALT 呼び出しの数が含まれます。

NUMBER OF PURGE CALLS

このフィールドには、このサブプールでの除去要求の数が示されます。

NUMBER OF LOCATE-TYPE CALLS, DATA ALREADY IN SUBPOOL

このフィールドには、データが既に OSAM プールに置かれていた場合の、このサブプールの探索タイプ呼び出しの数が示されます。

NUMBER OF BUFFERS SEARCHED BY ALL LOCATE-TYPE CALLS

このフィールドには、このサブプールですべての検索タイプ呼び出しで検索されたバッファの数が示されます。

NUMBER OF READ I/O REQUESTS

このフィールドには、このサブプールでの読み取り入出力要求の数が示されま

NUMBER OF SINGLE BLOCK WRITES BY BUFFER STEAL ROUTINE

このフィールドには、このサブプールでバッファ・スチール・ルーチンによって開始された単一ブロック書き込みの数が示されます。

NUMBER OF BLOCKS WRITTEN BY PURGE

このフィールドには、このサブプールで除去によって書き込まれたブロックの数が示されます。

TOTAL NUMBER OF I/O ERRORS FOR THIS SUBPOOL

このフィールドには、このサブプールでの入出力エラーの総数が示されます。

NUMBER OF BUFFERS LOCKED DUE TO WRITE ERRORS

このフィールドには、バッファをデータベースに書き込んでいるときに永続入出力エラーが起こったため、現在ストレージに「凍結」されているバッファの数が示されます。書き込み操作が原因で永続入出力エラーが起こったときは、その影響を受けたバッファは、データ・セットがクローズされるまで、あるいはオンライン・システムでは、システムがシャットダウンされるまで、ストレージに凍結されます。データ・セットがクローズされるか、オンライン・システムがシャットダウンされると、これらのバッファはログに書き込まれ、この値は再び 0 になります。

NUMBER OF LOCATE CALLS WAITED DUE TO BUSY ID

このフィールドには、このサブプールで ID が使用中のため待たされた探索呼び出しの数が示されます。

NUMBER OF LOCATE CALLS WAITED DUE TO BUFFER BUSY WRT

このフィールドには、このサブプールでバッファが書き込み中であつたため待たされた探索呼び出しの数が示されます。

NUMBER OF LOCATE CALLS WAITED DUE TO BUFFER BUSY READ

このフィールドには、このサブプールでバッファが読み取り中であつたため待たされた探索呼び出しの数が示されます。

NUMBER OF BUFFER STEAL/PURGE WAITED DUE TO BUFFER BUSY READ

このフィールドには、このサブプールで所有権が解放されるまで待たされた探索呼び出しの数が示されます。

NUMBER OF BUFFER STEAL REQUESTS WAITED FOR BUFFERS

このフィールドには、このサブプールでスチールすべきバッファが残っていないかつたため待たされたバッファ・スチール要求の数が示されます。

プログラム入出力報告書

プログラム入出力報告書には、IMS がリソースの使用待ちまたはイベントの完了待ちにあつたため、非アクティブ (どの IMS コードも実行されていない状態) であつた時間が示されます。この時間は IWAIT 時間と呼ばれ、実際的には、入出力操作が行われている間、IMS が待たされていた時間と見なすことができます。

DB モニターが実行されるたびに 1 つの報告書が作成され、その報告書には、IWAIT 時間が PCB 名とデータ・セット名別に記載されています。

プログラム入出力報告書に記載されている時間の単位は、いずれもマイクロ秒です。マイクロ秒とは、100 万分の 1 秒です (つまり、7050 マイクロ秒は 0.007050 秒です)。

プログラム入出力報告書の使用方法

プログラム入出力報告書を使用すると、IWAIT 時間と特定の PCB およびデータ・セットとの相関関係を知ることができます。その結果、IWAIT の数が比較的多くなる原因となっているデータベースを特定できます。

IWAITS は、PCB 名とデータ・セット名別に示されるため、多数の IWAIT を調べていくと、特定のアプリケーション・プログラムを突き止めることができます。パフォーマンスのために調整を行う際には、そのアプリケーション・プログラムに特に注意するようにしてください。次のトピックで説明する DL/I 呼び出し要約報告書を使用すると、アプリケーション・プログラムの中のどの DL/I 呼び出しが原因で IWAIT が多数になるのかが分かります。

プログラム入出力報告書のフィールド

以下の例は、プログラム入出力報告書の例を示しています。

```

IMS MONITOR  ****PROGRAM I/O****          TRACE START 1989 076 12:42:54  TRACE STOP 1989 076 12:43:07  PAGE 0008
                .....IWAIT TIME.....
PCB NAME      IWAITS      TOTAL      MEAN      MAXIMUM  DDNAME  MODULE  DISTR.
                NO.
-----
DLVNTZ02      1          35853     35853     35853    DBHVSAM2  VBH      127
                4          257649    64412     196028   HIDAM     VBH      128
                2          79222     39611     62452    XDLBT04I  VBH      129
PCB TOTAL
                7          372724    53246
DLVNTZX2      2          57645     28822     40686    HIDAM     VBH      130
                8          176622    22077     46141    DBHVSAM1  VBH      131
                6          105340    17556     27843    XDLBT04I  VBH      132
                3          65296     21765     23458    DBHVSAM2  VBH      133
PCB TOTAL
                19         404903    21310
BATCH TOTAL
                26         777627    29908
    
```

報告書の各フィールドの意味は次のとおりです。

TRACE START および TRACE STOP

TRACE START フィールドと TRACE STOP フィールドには、DB モニター・プログラムが最後に始動された時刻と、停止された時刻が示されます。この時刻は時刻機構によって生成されます。時刻は次のように示されます。

Clock time = hh.mm.ss

ここで、

hh 0 時から 23 時まで

mm 分

ss 秒

DB モニター・プログラムがバッチ実行の全期間にわたってオンになっていたときは、トレース開始時刻とトレース終了時刻は、バッチ実行が開始された時刻と停止された時刻です。DB モニター・プログラムが同じバッチ実行で 2 回以上

オン/オフにされたときは、トレース開始時刻とトレース終了時刻は、モニターが最後に始動された時刻と停止された時刻です。

PCBNAME

このフィールドは、報告書が情報を提供している PCB の名前を示します。各アプリケーション・プログラムは、1 つ以上の PCB を持っていることに注意してください。PCB 名がアプリケーション・プログラムに固有のものである場合、報告書に記載されている情報がどのアプリケーション・プログラムに関するものであるかを知るためにのみ、このフィールドを使用できます。

IWAITS

このフィールドは、IMS が非アクティブだった回数を示します。

IWAIT Time

このフィールドは、IMS が非アクティブだった経過時間を示します。

TOTAL

IMS が待ち状態にあった合計時間

MEAN

IMS が待ち状態にあった IWAIT の平均時間

MAXIMUM

IMS が待ち状態にあった 1 回の最長時間

DDNAME

このフィールドは、アプリケーション・プログラムが使用しているデータ・セットの名前を示します。1 つの報告書内で、アプリケーション・プログラム (PCB) による複数のデータ・セットへのアクセスが示される場合があります。コンパイルされた統計は、データ・セット別にリストされます。

MODULE

このフィールドは、IMS が待機する原因となった (DL/I 呼び出しに対する応答である) 内部呼び出しを実行したモジュールを示します。

DBH モジュール DFSDBHR0

DLE モジュール DFSDDL0

VBH モジュール DFSVSM0

DISTR. NUMBER

この分布番号は、分布付録報告書と一緒に使用される参照番号です。その意味については、860 ページの『分布付録報告書』を参照してください。

PCB TOTAL

特定の PCB に関して、このフィールドは以下のことを示します。

IWAITS

IMS が非アクティブであった総回数

TOTAL

IMS が待ち状態にあった合計時間

MEAN

IWAIT ごとの平均時間

BATCH TOTAL

この DB モニターの実行に関して、このフィールドは以下のことを示します。

IWAITS

IMS が非アクティブであった総回数

TOTAL

IMS が待ち状態にあった合計時間

MEAN

IWAIT ごとの平均時間

DL/I 呼び出し要約報告書

DB モニターが実行されるたびに 1 つの DL/I 呼び出し要約報告書が作成され、その報告書には、時間が PCB 名と DL/I 呼び出しのタイプ別に記載されています。

DL/I 呼び出し要約報告書からは、次の 2 つのことが分かります。

- IMS がリソースの使用待ちまたはイベントの完了待ちにあったため、非アクティブ (どの IMS コードも実行されていない状態) であった時間。この時間は IWAIT 時間と呼ばれ、実際的には、入出力操作が行われている間、IMS が待たされていた時間と見なすことができます。
- IMS が非アクティブであった経過時間のうち、それが実際には IWAIT 時間であった時間。

DL/I 呼び出し要約報告書に記載されている時間の単位は、いずれもマイクロ秒です。マイクロ秒とは、100 万分の 1 秒です (つまり、7050 マイクロ秒は 0.007050 秒です)。

DL/I 呼び出し要約報告書の使用方法

DL/I 呼び出し要約報告書の主要な用途は、多数の IWAIT の原因となっている DL/I 呼び出しを突き止めることです。

IWAITS/CALL フィールドの数が相対的に多いときは、その理由を調べる必要があります。報告書に記載されている数は、特定の DL/I 呼び出し、セグメント、および PCB と関係があるので、IWAIT をたどっていくと、アプリケーション・プログラムの特定の部分を突き止めることができます。さらに、分布付録報告書を使用すると、出された DL/I 呼び出しのうち、IWAITS/CALL フィールドの平均値をゆがめている呼び出しの個数が分かります。DL/I 呼び出し要約報告書の情報のこれ以外の使い方については、860 ページの『分布付録報告書』を参照してください。

入出力操作の回数を減らすようにチューニングを行うとき、数多く出された呼び出しに最大限の注意を払ってください。2000 回実行される呼び出しで 1 秒ずつ節約した方が、10 回出される呼び出しで 5 秒ずつ節約するよりも大きい利益が得られます (2000 秒対 50 秒)。

DL/I 呼び出し要約報告書のフィールド

以下の例は、DL/I 呼び出し要約報告書の例を示しています。

IMS MONITOR *****DL/I CALL SUMMARY*****				TRACE START 1989 076 12:42:54				TRACE STOP 1989 076 2:43:07 PAGE 0009			
CALL	LEV	STAT	DL/I	IWAITS/	IWAITS/	..ELAPSED	TIME...	.NOT	IWAIT	TIME..	DISTRIB.
PCB NAME	FUNC	NO.SEGMENT	CODE	CALLS	CALLS	MEAN	MAXIMUM	MEAN	MAXIMUM	NUMBER	
DLVNTZ02	STAT	(00)	GA	1	1	1.00	968281	968281	932428	932428	1 A,B,C

GN	(00)	GB	7	0	0.00	5703	29519	5703	29519	4	A,B,C
GN	(03)K1Z		6	0	0.00	165	253	165	253	5	A,B,C
GN	(02)K8K5	GA	8	0	0.00	263	888	263	888	6	A,B,C
GN	(03)K1Z	GK	3	0	0.00	6844	20272	6844	20272	7	A,B,C
GN	(03)K6Y		3	0	0.00	140	173	140	173	8	A,B,C
GN	(03)K5YK1	GA	3	0	0.00	197	219	197	219	9	A,B,C
GN	(04)K1Y	GK	5	0	0.00	306	892	306	892	10	A,B,C
GN	(04)K42		5	0	0.00	121	173	121	173	11	A,B,C
GN	(03)K5XK2	GK	5	0	0.00	9951	41900	9951	41900	12	A,B,C
GN	(03)K6		7	0	0.00	1292	7219	1292	7219	13	A,B,C
GN	(02)K5	GA	9	0	0.00	160	220	160	220	14	A,B,C
GN	(04)K1Y		7	0	0.00	12008	45991	12008	45991	15	A,B,C
GN	(03)K5XK2		4	0	0.00	6333	20917	6333	20917	16	A,B,C
GN	(02)K5		3	0	0.00	126	138	126	138	17	A,B,C
GN	(01)K1	GA	5	0	0.00	6773	23625	6773	23625	18	A,B,C
GN	(03)K5YK1		10	0	0.00	9444	30320	9444	30320	19	A,B,C
GN	(04)K6X	GK	4	0	0.00	141	165	141	165	20	A,B,C
GN	(04)K1X	GK	4	0	0.00	3278	12578	3278	12578	21	A,B,C
GN	(04)K4		4	0	0.00	423	1342	423	1342	22	A,B,C
GN	(03)K3K5		6	0	0.00	1360	6982	1360	6982	23	A,B,C
GN	(02)K2		4	0	0.00	325	910	325	910	24	A,B,C
GN	(03)K3K5	GA	2	0	0.00	189	196	189	196	25	A,B,C
GN	(04)K1X		2	0	0.00	134	142	134	142	26	A,B,C
GU	(01)K1		9	0	0.00	3387	26518	3387	26518	27	A,B,C
GN	(02)K8K5		3	0	0.00	179	207	179	207	48	A,B,C
GN	(02)K5	GE	1	0	0.00	116	116	116	116	49	A,B,C
GU	(03)K5YK1		8	0	0.00	7752	26664	7752	26664	50	A,B,C
GNP	(02)K5	GE	2	0	0.00	108	120	108	120	51	A,B,C
GNP	(03)K5YK1		5	0	0.00	7873	37362	7873	37362	52	A,B,C
GU	(04)K1Y		1	0	0.00	669	669	669	669	56	A,B,C
GU	(04)K4		4	0	0.00	11572	44675	11572	44675	59	A,B,C
GU	(02)K5		6	0	0.00	9025	45363	9025	45363	61	A,B,C
GNP	(02)K2	GE	1	0	0.00	87	87	87	87	62	A,B,C
GNP	(03)K3K5		2	0	0.00	165	167	165	167	63	A,B,C
GU	(02)K2		3	0	0.00	431	566	431	566	64	A,B,C
GU	(03)K5XK2		5	0	0.00	13570	52617	13570	52617	65	A,B,C
GU	(04)K6X		3	0	0.00	4892	13242	4892	13242	66	A,B,C
GU	(03)K6		2	0	0.00	1247	1810	1247	1810	67	A,B,C
GU	(04)K42		3	0	0.00	998	1390	998	1390	68	A,B,C
DLET	(04)K42		2	0	0.00	46374	49018	46374	49018	72	A,B,C
DLET	(04)K6X		2	0	0.00	108321	118802	108321	118802	73	A,B,C
DLET	(02)K2		1	0	0.00	44789	44789	44789	44789	74	A,B,C
DLET	(03)K3K5		3	0	0.00	87567	121071	87567	121071	75	A,B,C
GU	(02)J5	GE	1	0	0.00	1644	1644	1644	1644	111	A,B,C
DLET	(03)J6		1	0	0.00	35633	35633	35633	35633	112	A,B,C
GU	(02)J9	GE	1	0	0.00	705	705	705	705	113	A,B,C
DLET	(04)J7J9		1	0	0.00	245921	245921	245921	245921	114	A,B,C
REPL	(03)J7PJ6	DA	1	0	0.00	225	225	225	225	115	A,B,C
BATCH	TOTAL										
			392	22	0.05	22310		20983			

報告書の各フィールドの意味は次のとおりです。

TRACE START および TRACE STOP

TRACE START フィールドと TRACE STOP フィールドには、DB モニター・プログラムが最後に始動された時刻と、停止された時刻が示されます。この時刻は時刻機構によって生成されます。時刻は次のように示されます。

Clock time = hh.mm.ss

ここで、

hh 0 時から 23 時まで

mm 分

ss 秒

DB モニターがバッチ実行の全期間にわたってオンになっていたときは、トレース開始時刻とトレース終了時刻は、バッチ実行が開始された時刻と停止された時刻です。DB モニターが同じバッチ実行で 2 回以上オン/オフにされたときは、トレース開始時刻とトレース終了時刻は、モニターが最後に始動された時刻と停止された時刻です。

PCBNAME

このフィールドは、報告書が情報を提供している PCB の名前を示します。各 A

アプリケーション・プログラムは、1 つ以上の PCB を持っていることに注意してください。 PCB 名がアプリケーション・プログラムに固有のものである場合、報告書に記載されている情報がどのアプリケーション・プログラムに関するものであるかを知るためにのみ、このフィールドを使用できます。

CALL FUNC

このフィールドは、アプリケーション・プログラムが実行した DL/I 呼び出しのタイプを示します。

LEV NO.

このフィールドは、DL/I 呼び出しを実行するためにデータベース・レコードの階層でアクセスされたレベルを示します。このフィールドがゼロのときは、位置が設定されていないため、レベルが設定されなかったことを意味します。これは通常、何らかの理由で DL/I 呼び出しが処理できないときに起こります。

SEGMENT

このフィールドは、DL/I 呼び出しによってアクセスされたセグメントの 8 文字の名前を示します。

STAT CODE

このフィールドは、呼び出しの後に返された状況コードを示します (状況コードがブランクでなかった場合)。

DL/I Calls

このフィールドは、この特定の DL/I 呼び出しが発行され、前の 5 つの欄にリストされた 5 つの固有特性を持っていた回数を示します。

IWAITS

このフィールドは、IMS が非アクティブだった回数を示します。

IWAITS/CALL

このフィールドは、DL/I 呼び出し別に、IMS が非アクティブだった平均回数を示します。

ELAPSED TIME

このフィールドは、DL/I 呼び出し別に、呼び出しの経過時間を示します。

MEAN

DL/I 呼び出しごとの平均経過時間

MAXIMUM

1 つの DL/I 呼び出しの最長経過時間

NOT IWAIT TIME

このフィールドは、DL/I 呼び出し別に、経過時間から IWAIT 時間を差し引いた時間を示します。

MEAN

NOT IWAIT TIME の平均時間

MAXIMUM

1 つの NOT IWAIT TIME の最長時間

NOT IWAIT TIME には、IMS 領域で実行されている優先度の高いタスクが消費した時間があれば、その時間が含まれています。NOT IWAIT TIME は、

IMS データベース領域が優先度の高いタスクであって、優先度の低いタスクが割り込みを引き起こしていなければ、総プロセッサ時間にほぼ等しくなります。

DISTRIB. NUMBER

この分布番号は、分布付録報告書と一緒に使用される参照番号です。その意味については、『分布付録報告書』を参照してください。

C、A、および B

IWAITS/CALL、ELAPSED TIME、および NOT IWAIT TIME 欄の上に示されているこれらの文字は、分布付録報告書と一緒に使用される参照文字です。説明については、『分布付録報告書』を参照してください。

BATCH TOTAL

この DB モニターの実行に関して、このフィールドは以下のことを示します。

DL/I Calls

DL/I 呼び出しの総数

IWAITS

IMS が非アクティブであった総回数

IWAITS/CALL

DL/I 呼び出しごとの IWAITS の平均数

分布付録報告書

分布付録報告書を使用するのは、通常でないイベントの組み合わせが発生したと考えられ、プログラム入出力報告書や DL/I 呼び出し要約報告書の時間や総計といった情報だけでは問題の原因が突き止められない場合です。

分布付録報告書には、時間または総計が生成された特定イベントが集められ、一定の範囲にわたるイベントの分布が示されます。これは、プログラム入出力報告書と DL/I 呼び出し要約報告書から取り出した特定イベントについて行われます。

分布付録報告書がどのようなときに役立つかについては、以下の例を参照してください。DL/I 呼び出し要約報告書に、以下の例に示す情報行があったとします。

PCB NAME	CALL FUNC	LEV NO. SEGMENT	DL/I CALLS	(C) IWAITS CALL	...	DISTRIB. NUMBER
DHVBT203	DLET	(01)A1111111	11	8.63		11C

IWAITS/CALL フィールドの上に示されている (C) は、そのイベントが分布付録報告書に詳しく記載されていることを意味します。DISTRIB. NUMBER フィールドは、このイベントが分布付録報告書で、具体的に言えば、行 11 C で分析されていることを示しています。この例では、IWAITS/CALL フィールドの 8.63 という値は、DL/I 呼び出し要約報告書の他の IWAITS/CALL の数字に比べて、非常に大きいものになっています。詳細については、分布付録報告書の行 11 C を調べてください。この行は、次のようになっています。

```
11 C ....0....0....1....2....3....4....5....6....7....8....INF
      0      0      0      0      0      2      4      1      3      1
```

行 11 C の横に並んでいる数字 (0 から INF まで) は、事前に定義された範囲です。行 11 C の下に並んでいる数字はイベントの分布です。 IWAITS/CALL の総計が大きい理由 (PCB DHVBT203 でセグメント A1111111 に対して出された DLET 呼び出し) についての詳しい情報を探してください。行 11 C の下に示されている数字の分布は次のことを示しています。すなわち、出された 11 個の DLET 呼び出しのうち、

- 2 つの呼び出しが 5 つの IWAIT を起こした
- 4 つの呼び出しが 6 つの IWAIT を起こした
- 1 つの呼び出しが 7 つの IWAIT を起こした
- 3 つの呼び出しが 8 つの IWAIT を起こした
- 1 つの呼び出しが 9 つ以上の IWAIT を起こした

1 つの呼び出しが 9 つ以上の IWAIT を起こしているため、この呼び出しが報告書の IWAITS/CALL フィールドの平均値をゆがめている可能性があります。1 回の DL/I 呼び出し当たりの数が相対的に多い IWAIT を除くようデータベースをチューニングし、9 つ以上の IWAITS を要求した呼び出しを調べてください。この例では、分布間の間隔は 1 です。他の項目 (例えば、以下の例の報告書の行 4B) については、間隔が 1 よりも大幅に大きくなっています。そのような場合、データは、16000 から 32000 までの範囲内に呼び出しが 1 回あるものと解釈できます (行 4B の場合)。

以下の例は、分布付録報告書です。この報告書を見ると、次のことに注意してください。

- 分布付録報告書の行は、常に数字 (55、56 など) または数字と文字 (3A、3B など) によって示されます。数字は、プログラム入出力報告書または DL/I 呼び出し要約報告書の DISTRIBUTION NUMBER フィールドに使用されている数字です。文字は、どのイベントが分布されているかを示すために DL/I 呼び出し要約報告書で使用されています。
- 行の横にある数字は、イベントを分布させることのできる事前定義された範囲です。これらの範囲は、イベントに適したものになっています。例えば、範囲 0、1、2、3 などは、DL/I 呼び出しについての IWAIT の総計を分布させるのに適しています。(この例では、0、1、2、3、といった IWAIT を必要とする呼び出しがいくつあったかを知る必要がありました)。プログラム入出力報告書と DL/I 呼び出し要約報告書の他のイベントはすべて、時間であって総計ではないため、これらのイベントには、0、1000、2000 といった範囲が適しています。0、1000、2000 といった数字の単位は、マイクロ秒です。マイクロ秒とは、100 万分の 1 秒です (つまり、2000 マイクロ秒は 0.002000 秒です)。範囲にはデフォルト値がありますが、定義し直すことができます。詳しくは、863 ページの『デフォルトの範囲の再定義』を参照してください。
- 行の下の数字は、常に、分布されるイベントの数を表す数字です。

```

IMS MONITOR  ****DISTRIBUTION APPENDIX****  RACE START 1989 076 12:42:54  TRACE STOP 1989 076 12:43:07 PAGE 0014
# 1A.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF
      0          0          0          0          0          0          0          0          0          0          0          1
1B.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF
      0          0          0          0          0          0          0          0          0          0          0          1
1C.....0.....0.....1.....2.....3.....4.....5.....6.....7.....8.....INF
      0          1          0          0          0          0          0          0          0          0          0          0
# 4A.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF
      5          0          0          1          0          0          1          0          0          0          0          0
4B.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF

```

	5	0	0	1	0	1	0	0	0	0	0	0	
	4C.....0.....0.....1.....2.....3.....4.....5.....6.....7.....8.....INF												
#	7	0	0	0	0	0	0	0	0	0	0	0	
	5A.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF												
	6	0	0	0	0	0	0	0	0	0	0	0	
	5B.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF												
	6	0	0	0	0	0	0	0	0	0	0	0	
	5C.....0.....0.....1.....2.....3.....4.....5.....6.....7.....8.....INF												
#	6	0	0	0	0	0	0	0	0	0	0	0	
	6A.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF												
	8	0	0	0	0	0	0	0	0	0	0	0	
	6B.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF												
	8	0	0	0	0	0	0	0	0	0	0	0	
	6C.....0.....0.....1.....2.....3.....4.....5.....6.....7.....8.....INF												
#	8	0	0	0	0	0	0	0	0	0	0	0	
	7A.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF												
	2	0	0	0	0	1	0	0	0	0	0	0	
	7B.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF												
	2	0	0	0	0	1	0	0	0	0	0	0	
	7C.....0.....0.....1.....2.....3.....4.....5.....6.....7.....8.....INF												
#	3	0	0	0	0	0	0	0	0	0	0	0	
	8A.....0.....1000.....2000.....4000.....8000.....16000.....32000.....64000.....128000.....256000.....INF												
	3	0	0	0	0	0	0	0	0	0	0	0	
⋮													
#	131.....0.....2000.....8000.....24000.....50000.....100000.....150000.....200000.....250000.....300000.....INF												
	0	1	5	2	0	0	0	0	0	0	0	0	
#	132.....0.....2000.....8000.....24000.....50000.....100000.....150000.....200000.....250000.....300000.....INF												
	0	0	5	1	0	0	0	0	0	0	0	0	
#	133.....0.....2000.....8000.....24000.....50000.....100000.....150000.....200000.....250000.....300000.....INF												
	0	0	3	0	0	0	0	0	0	0	0	0	

報告書の上部の TRACE START フィールドと TRACE STOP フィールドには、DB モニターが最後に始動された時刻と、停止された時刻が示されます。この時刻は時刻機構によって生成されます。時刻は次のように示されます。

Clock time = hh.mm.ss

ただし、次のとおりです。

hh 0 時から 23 時まで

mm 分

ss 秒

DB モニターがバッチ実行の全期間オンになっていたときは、トレース開始時刻とトレース終了時刻は、バッチ実行が開始された時刻と停止された時刻です。DB モニターが同じバッチ実行で 2 回以上オン/オフにされたときは、トレース開始時刻とトレース終了時刻は、モニターが最後に始動された時刻と停止された時刻です。

分布付録報告書の作成方法

分布付録報告書は、DB モニターが稼働しているときに自動的に作成されるものではありません。分布付録報告書を手にするには、分析制御データ・セットに DIS 入力制御ステートメントを入れておく必要があります。

分布できるイベントとそのデフォルトの範囲

OSAM の場合の IWAIT 時間など、特定のイベントは、プログラム入出力報告書と DL/I 呼び出し要約報告書の中に分布できます。

以下の表は、プログラム入出力報告書と DL/I 呼び出し要約報告書の中で分布させることができるイベントを示しています。各イベントは ID を持っています。

表 58. 分布できるイベントとその ID :

分布できるイベント	ID
プログラム入出力報告書	
OSAM の場合の IWAIT 時間	D23
VSAM の場合の IWAIT 時間	D24
HSAM の場合の IWAIT 時間	D34
DL/I 呼び出し当たりの経過時間	D11
DL/I 呼び出し当たりの NOT IWAIT 時間	D12
DL/I 呼び出し当たりの IWAIT の数	D13

以下の表は、デフォルトの範囲が使用されたとき、事前に定義された各範囲のセットがどのような構成になっているかを、この ID を使用して示しています。表に示すように、範囲内の最初の数値は常にゼロにデフォルト設定され、最後の数は常に無限大 (INF) にデフォルト設定されます。

表 59. デフォルトを使用したときの事前定義の範囲

ID	デフォルトの範囲
D11,D12	0,1000,2000,4000,8000,16000,32000,64000,128000,256000,INF
D13	0,0,1,2,3,4,5,6,7,8,INF
D23,D24	0,2000,8000,24000,50000,100000,150000,200000,250000,300000,INF
D34	0,2000,4000,8000,16000,32000,64000,96000,128000,160000,INF

デフォルトの範囲の再定義

デフォルトの範囲は再定義することができます。そのためには、変更するデフォルトの範囲ごとに、入力制御ステートメントを分析制御データ・セットに入れておく必要があります。

関連資料: 分析制御データ・セットについては、「IMS V15 データベース・ユーティリティ」の『IMS モニター報告書印刷ユーティリティ (DFSUTR20)』に説明してあります。

デフォルトの範囲を変更する場合は、制御ステートメントを Dn n1、n2、.. の形式で指定します。

- Dn は、分布するイベントの ID です (例えば、D23 は、プログラム入出力報告書中の OSAM イベントの場合の IWAIT 時間を表す ID です)。
- n1 は特定のデフォルト値です。10 個のデフォルト値すなわちバケットをそれぞれ再定義することができます。例えば、プログラム入出力報告書の OSAM の IWAIT 時間は、次のように再定義することができます。

```
D23 0,500,1000,1500,2000,4000,,,100000,500000
```

この結果、7 番目と 8 番目の範囲はそのデフォルト値 (150000 と 200000) のままになり、最後の範囲 (INF) もそのデフォルト値のままになります。

モニター・オーバーヘッド報告書

この報告書には、DB モニターを実行するために必要なオーバーヘッドが記載されます。DB モニターは IMS の実行と並行して稼働するので、モニターがリソースを使用すると、モニターが稼働していないときと比べて、パフォーマンスが若干低下します。

モニター・オーバーヘッド報告書のフィールド

以下の例は、モニター・オーバーヘッド報告書の例を示しています。

```
IMS MONITOR ****MONITOR OVERHEAD****      TRACE START 1989 076 12:42:54      TRACE STOP 1989 076 12:43:07 PAGE 0013
MONITOR OVERHEAD DATA
```

M
M

```
13144 MILLISECONDS, TRACE INTERVAL
550 MILLISECONDS, MONITOR MODULE TIME
853 MONITOR RECORDS WERE PRODUCED
645 MICROSECONDS PER MONITOR ENTRY
```

報告書の各フィールドの意味は次のとおりです。

TRACE START および TRACE STOP

TRACE START フィールドと TRACE STOP フィールドには、DB モニター・プログラムが最後に始動された時刻と、停止された時刻が示されます。この時刻は時刻機構によって生成されます。時刻は次のように示されます。

Clock time = hh.mm.ss

ここで、

hh 0 時から 23 時まで

mm 分

ss 秒

DB モニターがバッチ実行の全期間オンになっていたときは、トレース開始時刻とトレース終了時刻は、バッチ実行が開始された時刻と停止された時刻です。

DB モニターが同じバッチ実行で 2 回以上オン/オフにされたときは、トレース開始時刻とトレース終了時刻は、モニターが最後に始動された時刻と停止された時刻です。

MILLISECONDS, TRACE INTERVAL

このフィールドは、モニターがオンになっていた時間を示します。

MILLISECONDS, MONITOR MODULE TIME

このフィールドは、モニターがオンになっていた間、モニターのコードが実際に実行されていた時間を示します。(この時間には、どの IMS コードも実行できないため、このフィールドには、モニターを使用するために必要な正味のオーバーヘッドが示されます)。

MONITOR RECORDS WERE PRODUCED

このフィールドは、モニターが書き込んだレコード数を示します。

MICROSECONDS PER MONITOR ENTRY

このフィールドは、モニターがレコードを書き込むために要した平均時間を示します。

第 47 章 IMS モニター報告書

IMS モニターの出力は、全体的に見た IMS DB/DC 環境を表します。報告書のサブセットは、データベース呼び出しとバッファリングを取り扱います。

バッチ・アプリケーション・パフォーマンスに関してこのデータを解釈すること、およびデータベース設計を検査することは、データベース管理に類する別個の作業と考えられます。

トランザクションのフローと IMS モニターのイベント

IMS モニターが時間測定情報を収集する、イベント、システム・アクティビティ、およびストレージ (バッファ・プールまたはデータ・セット) の使用方法に関する全体的な状況については、以下の表を参照してください。

左端の欄は、上から下に向かって処理イベントの順序を示し、それぞれのイベントは、表の後の注に示した IMS モニター報告書項目に関連しています。

表 60. トランザクションのフローと IMS モニターのイベントの説明：

フロー	イベント	アクティビティ	プール	データ・セット
1	ポーリングを待機	待機	N/A	N/A
2	データ転送	N/A	N/A	N/A
3	入力メッセージ処理	装置	CIOP	(回線用)
		MFS	MFP	IMS.FORMATx
		エンキュー	QBUF	メッセージ・キュー
4	入力キューイング	待機	N/A	N/A
5	スケジューリング	スケジューリング	QBUF	メッセージ・キュー
		PSB ロード	PSB、PSBW、DPSB	IMS.ACBLIBx
		DMB ロード	DMBP	IMS.ACBLIBx
6	プログラム・ロード	プログラム・ロード		IMS.PGMLIB
7	プログラム初期設定	初期化	N/A	N/A
8	メッセージ・キュー GU	プライム GU 呼び出し	N/A	N/A
9	プログラム実行	DC 呼び出し	QBUF	メッセージ・キュー
	DL/I の経過	DB 呼び出し	OSAM/VSAM	N/A
	待機の経過	DB 入出力	OSAM/VSAM	データベース
		SPA 挿入	CWAP	IMS.SPA

表 60. トランザクションのフローと IMS モニターのイベントの説明 (続き):

フロー	イベント	アクティビティ	プール	データ・セット
10	出力メッセージ挿入	DC ISRT 呼び出し	QBUF	メッセージ・キュー
		DC GU 呼び出し	QBUF	メッセージ・キュー
		同期点 (MODE=SNGL)	OSAM/VSAM	データベース
11	同期点を待機	(MODE=MULT の場合のみ)	N/A	N/A
12	プログラム終了	プログラムの終了	OSAM/VSAM	データベース
13	選択を待機	待機	QBUF	メッセージ・キュー
14	出力メッセージ処理	メッセージ送信	QBUF	メッセージ・キュー
		MFS 装置	MFP CIOP	IMS.FORMATx (回線用)
15	データ転送	N/A	N/A	N/A
16	出力キュー処理	デキュー	QBUF	メッセージ・キュー

注:

1. IMS モニターは、ポーリングの待ち時間を記録しません。
2. すべてのメッセージについて、伝送されたデータに関する回線アクティビティと、IMS 通信サブタスク・アクティビティが記録されます。入力メッセージ・アクティビティと出力メッセージ・アクティビティとは区別されません。
3. 入力メッセージ処理時に、装置依存処理および通信入出力プール (CIOP) の使用が記録されます。メッセージ形式サービス (MFS) が必要な場合は、メッセージ形式バッファ・プール、およびアクティブ IMS.FORMATA/B ライブラリーに対するすべての入出力が記録されます。その後、入力メッセージは、メッセージ・キュー・バッファ (QBUF) に入れられ、メッセージ・キュー・データ・セットへの入出力が可能になります。IMS モニターの報告では、このアクティビティと出力メッセージ処理が区別されません。
4. IMS モニターは、入力キューでの待ち時間を記録しません。
5. IMS は、ある領域内に処理プログラムをスケジュールし、このアクションの一部として、メッセージをプログラムに提示できるようにメッセージ・キューまたは QBUF プールにアクセスします。この時点で、必要な PSB および物理データベース・ブロックは、PSB プールおよび DMB プールで使用可能にされます。それらがプール内にまだ存在していない場合は、アクティブ IMS.ACBLIBA/B データ・セットから検索されます。これらのイベントは、IMS モニター報告書の SCHEDULING AND TERMINATION に要約されません。

6. 次に、アプリケーション・プログラムは IMS.PGMLIB から領域にロードされます。あるいは、その領域内に既にある場合には初期設定されます。この処理は、IMS モニター報告書の SCHEDULE TO FIRST CALL の一部に含まれます。
7. アプリケーション・プログラムが実行する初期設定は、この時間枠内に含まれ、最初のメッセージ・キュー呼び出しまたはデータベース呼び出しの時間までの間に行われます。この処理は、1 つのスケジューリング・イベントで複数のトランザクションが処理されるときには繰り返されないため、スケジューリング・プロセスの一部と見なされます。この処理は、IMS モニター報告書の SCHEDULE TO FIRST CALL の一部として記録されます。
8. 最初のメッセージ・セグメントを入手するための最初の DL/I GU 呼び出しを実行するイベントは、単独では記録されません。アプリケーションにこのメッセージを与えるための処理 (プライム処理) は、SCHEDULE TO FIRST CALL に含まれます。
9. プログラム経過イベントは、最初の呼び出しから処理プログラムの終了まで測定されます。合計経過時間が、IMS モニター報告書に ELAPSED EXECUTION として記録されます。
 - 各 DL/I 呼び出しは、DC または DB のいずれであっても、メッセージ・キューまたはデータベース・データ・セットの使用、およびそれぞれのプールとともに個別に記録されます。各外部サブシステム呼び出しは個別に記録されます。これらのイベントは、IMS モニター報告書の CALLS の部分に記録されます。
 - DL/I 呼び出しで入出力アクティビティーが行われた場合、データ待ちに費やされる時間が WAIT 時間として、および入出力の回数として記録されます。プログラムのデータベース処理意図と他の更新アクティビティーが競合状態にあるときは、これも WAIT 時間の原因となります。外部サブシステム呼び出しごとに、外部サブシステム処理に費やされる時間が、個別に WAIT 時間として記録されます。処理がある意図のために延期されるときは、経過時間は IMS モニター報告書の項目 IDLE FOR INTENT に記録されます。入力待ちとして指定された領域に入力メッセージがないために処理が延期される場合は、次の入力メッセージがくるまでの待ち時間は、経過時間間隔および待ち時間から除外されます。入力待ち時間は、プログラム入出力報告書の **WFI 項目のみに示されます。
 - メッセージ処理が会話型トランザクションの一部分である場合は、通信作業域プール (CWAP) と IMS.SPA データ・セット内でのアクティビティーが記録されます。
10. 処理されたばかりのメッセージに応答するための DL/I ISRT 呼び出しが、トランザクションについて記録されます。

処理が続行されるときには、別の入力メッセージを得るためのメッセージ・キューに対する GU やチェックポイント呼び出しなどの同期点が取られます。プログラム処理をコミットするための、データベースおよびメッセージ・キューの入出力が記録されます。IMS または処理プログラムによるチェックポイント操作は、個別に記録されます。

11. 同期前の複数メッセージについての処理の場合 (MODE=MULT)、同期点待ちがある可能性があります。IMS モニターはこの待ち時間を記録しません。

12. アプリケーション・プログラムからの終了後の処理 (プログラム終了イベント) は、初期スケジューリング時間に含まれます。これは、SCHEDULING AND TERMINATION の一部です。
13. 出力メッセージは、選択が端末に伝送されるのを待ちます。IMS モニターは、出力キューでの待ち時間を記録しません。
14. アプリケーション・プログラムを出て終了した後、出力メッセージ処理イベントが記録されます。これには、メッセージ形式および装置従属処理、ならびに出力メッセージの送信が含まれます。IMS モニターの報告では、このアクティビティと入力メッセージ処理が区別されません。
15. すべてのメッセージについて、伝送されたデータに関する回線アクティビティと、IMS 通信サブタスク・アクティビティが記録されます。出力メッセージ・アクティビティと入力メッセージ・アクティビティとは区別されません。
16. キューから出力メッセージを除去する処理が記録されます。

IMS モニター・トレース・イベントの間隔

IMS モニター・トレースの間隔は、マスター端末のオペレーターが開始コマンドの入力と停止コマンドの入力の間に /TRACE コマンドを使用することによって境界が設定されます。IMS モニターは、指定した間隔の満了によって停止することもあります。

オンライン IMS イベントは、IMSMON データ・セットに入れられる IMS モニター・レコードに記録されます。イベント・タイミングは、従属領域アクティビティに関連付けられます。以下の図に、時間測定が行われるイベントの間隔の境界を示します。

モニター・トレース間隔には、以下の間隔が含まれます。

- スケジューリングと終了
 - メッセージなし
 - ブロック・ローダー使用中
 - 意図障害 (排他的意図およびデータ共用) およびスケジュール障害 (PSB 使用中およびスペース障害)
 - スケジューリングと終了の経過
 - 待ちなし
 - ACBLIB 待ち
 - DB フラッシュ待ち
 - DB CLOSE 待ち
- 領域占有 (「スケジュールリングと終了の経過」全体とオーバーラップします)
 - スケジュールから最初の呼び出しまで
 - 実行の経過

領域の待ちなし時間は、待ち時間には含まれない経過時間のことです。ページング、またはプロセッサが優先順位の高いタスクにディスパッチされるために発生する遅延は、待ちなし時間の増加の原因となります。

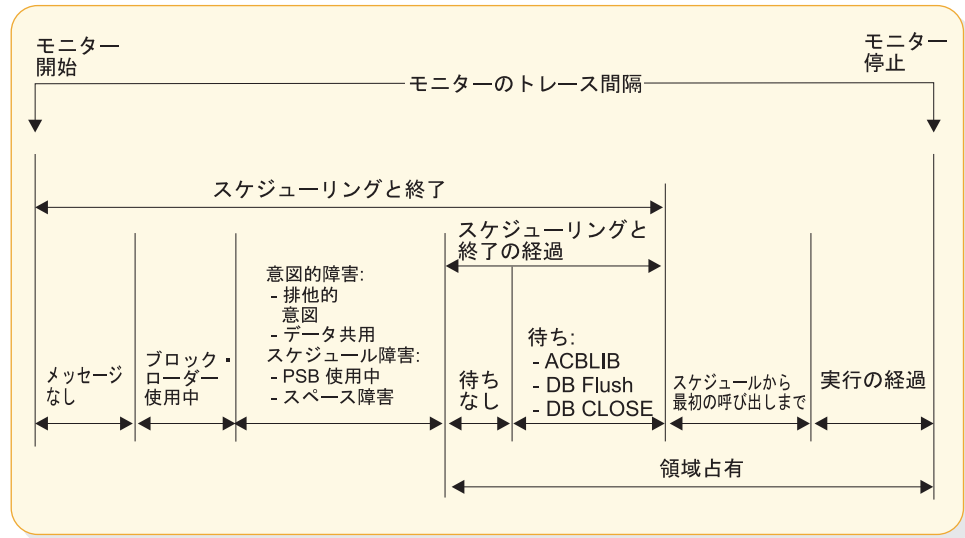


図 87. IMS モニター・トレース・イベントの間隔

IMS モニター報告書の概要

IMS モニターが収集するデータから得られる報告書のリストと、その報告書に含まれている主要なパフォーマンス・データを以下の表に示します。

関連資料: 「DC」が付けられている報告書については、925 ページの『第 49 章 DCCTL の IMS モニター報告書』を参照してください。

リストで「MSC」が付いている報告書は、MSC がアクティブなときのみ作成されます。MSC 報告書とその解釈は、以下の表に示してあります。

以下の表にリストした報告書の順序は、IMS モニター報告書印刷プログラムからの出力の順序と一致しています。モニター・スナップショットの期間または制約には、個々の報告書に必要な特定のイベントが含まれていないことがあります。その場合には、報告書の見出しまたはデータの一部のみが作成されます。

表 61. IMS モニター報告書の出力順序および情報

報告書名	主な内容
システム構成	モニター実行の文書
メッセージ・キュー・プール	トランザクションごとのバッファリングとメッセージ入出力
データベース・バッファ・プール (DB)	トランザクションごとの DB 呼び出しと入出力のカウンタ
VSAM バッファ・プール (DB)	挿入と入出力のカウンタ
メッセージ形式バッファ・プール	取り出したブロックと入出力のカウンタ
ラッチ競合統計	IMS 内部処理
一般的な待ち時間イベント	SNAPQ の待ち時間
領域およびジョブ名	モニター実行の文書

表 61. IMS モニター報告書の出力順序および情報 (続き)

報告書名	主な内容
領域要約	経過時間と DL/I 呼び出しのカウンタ
領域待機	待ち時間
領域別プログラム	領域使用の経過時間
プログラム要約	プログラム全体の統計
プログラム入出力 (DB)	待ち時間/PCB
通信要約	回線の経過時間
通信待機	回線別の待ち時間
回線機能	送信されたブロックのカウンタとサイズ
MSC トラフィック (MSC)	トランザクションのカウンタと経路指定
MSC 要約 (MSC)	宛先別のトランザクションのカウンタ
MSC キューイング要約 (MSC)	リンク別のカウンタとキューイング時間
トランザクション・キューイング	キューの負荷の統計
報告書	スペース障害とデッドロックのカウンタ
実行プロファイル	モニター実行の文書
呼び出し要約 (DB)	呼び出しカウンタと時間測定情報/セグメント・タイプ
分布付録	イベントの度数分布

IMS モニター報告書のデータ項目のほとんどは、経過時間です。通常、それらはマイクロ秒単位で表されます。1876534 という項目は、1.876534 秒または 1876 ミリ秒を表します。この規則に従っていない時間については、報告書の中で計測単位が表示されています。

さらに、OCCURRENCES の見出しの下に、イベントの数と、バイト数を表す数字がいくつか示されます。

モニター実行の文書化

トレース間隔ごとに、実行された処理について、いくつかの一般報告書または要約報告書があります。これらの報告書は、ご使用の IMS モニター実行の文書の一部として使用できます。

トレースが実行されたときの条件は、できるだけ正確に記録することが重要です。文書には、トレースの前後に /DISPLAY コマンドで何回か得られたシステム状況情報、予期されているアプリケーション・プログラム・アクティビティのプロファイル、および必要な処理イベントを含めることができます。トレース間隔は、代表的な処理ロードを表すべきであり、偏った、あるいは不適当なヒストリー・レコードであってはなりません。

システム構成報告書データへの追加

SYSTEM CONFIGURATION という表題の付けられた最初の一般報告書は、BUFFER POOL STATISTICS というページ見出しのもとにあります。これは、IMS

および z/OS システムの修正レベルを示しています。システム構成の出力は、以下のセクションに例示してあります。

モニターのトレース間隔の記録

ほとんどの IMS モニター報告書の見出しには、トレースの開始時刻と停止時刻が示されています。その形式は、年 日 (ユリウス暦) HH:MM:SS です。トレース間隔の全長は、MONITOR OVERHEAD DATA という表題の下にミリ秒単位で示されています。続く行には、IMSMON データ・セットに入れられたトレース・レコードの数が示されています。モニターのトレース間隔の記録例は、以下の例に示してあります。

```
***IMS MONITOR*** BUFFER POOL STATISTICS TRACE START 2009 180 5:55:15 TRACE STOP 2009 180 5:59:49 PAGE 0001
```

SYSTEM CONFIGURATION

```
SYSTEM CONFIGURATION : MVS/ESA
IMS VERSION           : 11
RELEASE LEVEL         : 1
MODIFICATION NUMBER   : 0
```

モニター実行プロファイルの完了

処理率が、実行プロファイル報告書の最後の部分にまとめて示されています。この統計は、モニター間隔に関して、トランザクション・スループット、および DL/I と入出力アクティビティの度合いを要約したものです。以下の例は、実行プロファイル報告書の例です。

実行プロファイル報告書の下の部分に、以下のいくつかの率が表示されます。

各領域ごとの、DL/I 経過時間に対するプログラム経過時間の率

DL/I 処理時の、待ち時間に対する DL/I 経過時間の率

他のサブシステム呼び出し経過時間に対するプログラム経過時間の率

他のサブシステム呼び出し経過時間に対する DL/I 経過時間の率

各従属領域には、領域 1 から始まるシーケンス番号が付き、これで識別されます。

```
IMS MONITOR **RUN PROFILE** TRACE START 2009 180 5:55:15 TRACE STOP 2009 180 5:59:49 PAGE 0184
TRACE ELAPSED TIME IN SECONDS.....274.6
TOTAL NUMBER OF MESSAGES DEQUEUED.....1403
TOTAL NUMBER OF SCHEDULES.....173
NUMBER OF TRANSACTIONS PER SECOND.....5.1
TOTAL NUMBER OF DL/I CALLS ISSUED.....18632
NUMBER OF DL/I CALLS PER TRANSACTION.....13.2
NUMBER OF OSAM BUFFER POOL I/O'S.....11236, 8.0 PER TRANSACTION
NUMBER OF MESSAGE QUEUE POOL I/O'S.....0, 0.0 PER TRANSACTION
NUMBER OF FORMAT BUFFER POOL I/O'S.....0, 0.0 PER TRANSACTION
RATIO OF PROGRAM ELAPSED TO DL/I ELAPSED:
REGION 1: 1.09
REGION 2: 1.09
REGION 3: 1.00
REGION 4: 1.02
REGION 5: 1.01
REGION 6: 1.00
REGION 7: 1.00
REGION 8: 1.00
REGION 9: 1.17
REGION 10: 1.00
REGION 11: 1.00
REGION 49: 1.03
REGION 50: 1.19
RATIO OF DL/I ELAPSED TO DL/I IWAIT:
REGION 1: 325.65
REGION 2: 73.49
REGION 4: 100.35
```

REGION 5: 85.76
REGION 6: 82.99
REGION 47: 95.64
REGION 48: 45.93
REGION 49: 9.22

領域およびジョブ名報告書を使用すると、領域と z/OS ジョブ名を突き合わせる
ことができます。ジョブ名は、トレースが開始される前にオペレーターが開始したす
べての従属領域の EXEC ステートメントのステップ名に相当します。領域ジョブ名
は、GENERAL REPORTS という見出しの付いたモニター出力ページに表示されま
す。885 ページの『チェックポイントの影響の検出』に示されているとおりで
す。

一部の汎用処理率は、いくつかのバッファ・プール統計報告書の最後の部分に示
されます。それらの率を、トレース間隔の文書化プロファイルに含めることができ
ます。それらの率は、1 つのアプリケーションまたはシステム・リソースに固有の
ものではなく、一連のモニター実行全体のバリエーションの標識として使用するこ
とができます。

次の率が示されます。

- (OSAM 読み取り数 + OSAM 書き込み数 + すべての待機) の合計数を、トラン
ザクションの合計数で除算したもの。

この率は、メッセージ・キュー・プール報告書からのもので、トランザクション
1 つ当たりの、メッセージ・キューイング機能の処理に必要な物理入出力アクテ
ィビティを示します。

- (OSAM 読み取り数 + OSAM 書き込み数 + BISAM 読み取り数) の合計数を、
トランザクションの合計数で除算したもの。

この率は、データベース・バッファ・プール報告書からのもので、トランザク
ション 1 つ当たりの、データベース・バッファリング機能の処理に必要な物理
入出力アクティビティを示します。

- (事前取り出し入出力数 + 即時取り出し入出力数 + ディレクトリー入出力数) の
合計数を、トランザクションの合計数で除算したもの。

この比率は、メッセージ形式バッファ・プール報告書から得られ、MFS 機能
の処理に必要な物理入出力アクティビティをトランザクション別に示します。

IMS モニター報告書オカレンスの検査

IMS モニター報告書印刷プログラムの出力を調べる際に、報告書の見出しがあつて
も、それは必ずしも該当するデータがリストされていることを意味するものではあ
りません。

システム定義オプションおよびユーティリティー制御ステートメントにより、出力
の内容は次のような影響を受けます。

- 制御ステートメントで DLI を指定しないと、呼び出し要約報告書は出力されま
せん。

- 出力には、制御ステートメントで DIS または DISTRIBUTION を指定していない限り、1 組の分布報告書は含まれません。多数の報告書に現れる DISTRIBUTION NUMBER という表題の欄には、分布報告書に含まれている項目の相互参照が含まれています。
- 制御ステートメントで ONLY DLI を指定すると、呼び出し要約報告書だけが出力されません。

多数の要約報告書では、開始値と終了値の差を計算するためにシステム状況が必要であり、しかもこの状況は /TRACE SET OFF 処理の間に得られるため、IMS モニターの実行を IMS 制御領域の終了前に終わらせる必要があります。トレースが正しく停止しなかった場合は、次のメッセージが出されます。

```
NO QUEUE BUFFER POOL TRACES AT END TIME ON MONITOR LOG TAPE
****QUEUE BUFFER POOL REPORT CANCELLED****
```

同様に、他の要約報告書も作成されません。

BUFFER POOL STATISTICS という表題の一連の報告書には、VSAM BUFFER POOL セクションは含まれません。ただし、IMS.ACBLIB 内のデータベースが VSAM アクセス方式を使用している場合は除きます。VSAM が使用されていなければ、次のメッセージが出されます。

```
NO VSAM BUFFER POOL TRACES ON MONITOR LOG TAPE
****VSAM BUFFER POOL REPORT CANCELLED****
```

MESSAGE FORMAT BUFFER POOL のセクションは、メッセージ形式サービス (MFS) を使用する装置がシステム定義に指定されている場合のみ、含まれます。

IMS モニターが、特定の IMS モニター報告書またはその報告書の 1 セクションを作成するために使用するソース・データをトレース間隔中に記録していなかった場合は、報告書には見出しのみが表示されます。

従属領域内のアクティビティのモニター

IMS モニターは、トレース間隔中にアクティブである、/TRACE コマンドで識別されたすべての従属領域について、時間測定情報を収集します。各イベントごとの経過時間の合計、検出された最大の単独の時間、および平均時間をこのプログラムは記録します。

時間測定値を表示する主要な報告書は 3 つあります。それらの報告書とその内容のリストは次のとおりです。

- 領域要約報告書 (REGION SUMMARY REPORT)
 - スケジューリングと終了
 - スケジュールの終わりから最初の呼び出しまで
 - 以下について表示される、個別要約付きの実行経過
 - DL/I 呼び出し
 - 外部サブシステム・サービスとコマンド呼び出し
 - 外部サブシステムのデータベース・アクセス呼び出し
 - チェックポイント処理
 - 領域占有

- 領域待機 (REGION WAIT)
 - スケジューリングと終了時の待機
 - DL/I 呼び出し時の待機
 - 外部サブシステム呼び出し時の待機
 - チェックポイント時の待機
- 領域別プログラム (PROGRAMS BY REGION)
 - 実行の経過
 - スケジュールの終わりから最初の呼び出しまで

これら 3 つの報告書は、以下の例に示してあります。

従属領域のアクティビティーは、次の 5 つのカテゴリーに分けられます。

- スケジューリングと終了の経過時間

スケジューリング処理には、アクティブ IMS.ACBLIBA/B データ・セットからのブロック・ロード、および PSB の所有権の獲得などの多数の準備イベントが含まれます。アプリケーション・プログラムの終了後に領域アクティビティーを終わらせるのに必要な時間も含まれます。

- スケジュールの終わりから最初の呼び出しまでの経過時間

この時間は、制御プログラム・サービスの先頭にマークを付ける初期呼び出し (メッセージ・キュー、データベース、または外部サブシステムへの呼び出し) の前に行われる、アプリケーション・プログラムの初期設定およびハウスキーピング用に確保されます。これは、複数トランザクションが単一のスケジューリングで処理されるときに、反復して行われたい処理の時間です。

- プログラム経過時間 (すべての呼び出しを含む)

この時間には、アプリケーション・プログラムの主要な処理が含まれ、最初の呼び出しから、プログラムへの戻りまたは終了までの時間が測定されます。

- DL/I 呼び出しの実行に要した経過時間

この時間は、すべての DL/I 呼び出しの時間を含みます。各 DL/I 呼び出しイベントは、アプリケーション・プログラムへの戻りの呼び出し時刻から測定されます。

- 外部サブシステム呼び出しの実行に要した経過時間

この時間は、すべての外部サブシステム呼び出しの時間を含みます。各外部サブシステム・イベントは、呼び出し時刻から IMS への戻りまで測定されます。

以下の例は、領域要約報告書の例を示しています。

IMS MONITOR		****REGION SUMMARY****			TRACE START 1993 130 5:55:15		TRACE STOP 1993 130 5:59:49		PAGE 0011
		(A)			(B)				
OCCURRENCES	ELAPSED TIME.....			NOT IWAIT TIME(ELAPSED-IWAIT)			DISTRIBUTION	
		TOTAL	MEAN	MAXIMUM	TOTAL	MEAN	MAXIMUM	NUMBER	
SCHEDULING AND TERMINATION		-----	-----	-----	-----	-----	-----	-----	
**REGION	5	5	4146	829	948	4146	829	948	287A,B
**REGION	6	7	6028	861	1067	6028	861	1067	214A,B
**REGION	8	8	6847	855	1098	6847	855	1098	129A,B
**REGION	10	7	9664	1380	3668	9664	1380	3668	272A,B
**REGION	47	6	5482	913	1021	5482	913	1021	145A,B
**REGION	49	3	2612	870	917	2612	870	917	443A,B
**TOTALS		123	126042	1024		126042	1024		

SCHEDULE TO FIRST CALL

**REGION	1	1	15479797	15479797	15479797					555
**REGION	2	1	22376350	22376350	22376350					564
**REGION	3	1	15169488	15169488	15169488					578
**REGION	4	1	48146258	48146258	48146258					584
**REGION	48	1	795351	795351	795351					592
**REGION	49	4	2960425	740106	2951746					442
**REGION	50	1	15713464	15713464	15713464					575
**TOTALS		168	514286738	3061230						

ELAPSED EXECUTION

**REGION	1	1	290146255	290146255	290146255					1
**REGION	2	1	252290108	252290108	252290108					2
**REGION	3	1	259496970	259496970	259496970					3
**REGION	4	1	322812716	322812716	322812716					4
**REGION	48	1	273871107	273871107	273871107					48
**REGION	49	4	271703421	67925855	155176058					49
**REGION	50	1	290379922	290379922	290379922					50
**TOTALS		173	14238540145	82303700						

DL/I CALLS

**REGION	1	60	264626241	4410437	88981490	263813671	4396894	88970053	0.76	247A,B,C
**REGION	2	223	230505269	1033655	61048758	227368742	1019590	61011153	0.73	237A,B,C
**REGION	3	29	257704383	8886358	69000514	257704383	8886358	69000514	0.00	98A,B,C
**REGION	4	792	313735347	396130	52439653	310609035	392183	52439653	0.22	180A,B,C
**REGION	49	592	262886317	444064	30202068	234394017	395935	30159782	2.46	177A,B,C
**REGION	50	36	242591451	6738651	48651260	242591451	6738651	48651260	0.00	289A,B,C
**TOTALS		18632	12386905286	664818		12024562411	645371		0.97	

IDLE FOR INTENT

NONE

CHECKPOINT

NONE

REGION OCCUPANCY

**REGION	1	100.0%
**REGION	2	100.0%
**REGION	3	100.0%
**REGION	4	100.0%
**REGION	48	100.0%
**REGION	49	100.0%
**REGION	50	100.0%

以下の例は、領域待機報告書の例を示しています。

IMS MONITOR	****REGION IWAIT****	TRACE START	1993 130	5:55:15	TRACE STOP	1993 130	5:59:49	PAGE 0023
**REGION	5 OCCURRENCES	TOTAL	MEAN	MAXIMUM	FUNCTION	MODULE	DISTRIBUTION NUMBER	
SCHEDULING + TERMINATION								
SUB-TOTAL								
TOTAL								
DL/I CALLS								
	11	181816	16528	24375	DD=IMMSTR2A	DBH	117	
	8	112831	14103	17846	DD=IMMSTR1A	DBH	118	
	5	85460	17092	33717	DD=IMMSTR3A	DBH	119	
	5	58420	11684	14643	DD=IMINDEXA	VBH	120	
	1	4160	4160	4160	INT=DDLTRN24	BLR-64BIT	35	
	1	3623	3623	3623	PSB=BMPFPE02	BLR-64BIT	36	
	12	173866	14488	22152	DD=PRODCNTA	VBH	121	
	3	100576	33525	68373	DD=IMMSTR2B	DBH	428	
	1	17921	17921	17921	DD=IMMSTR3B	DBH	429	
	1	17195	17195	17195	DD=IMMSTR1B	DBH	430	
	1	13577	13577	13577	DD=IMINDEXB	VBH	431	
	3	49928	16642	20396	DD=PRODCNTB	VBH	432	
	4	10973	2743	2787	DD=ITEMACTB	DBH	453	
	2	37680	18840	27664	DD=IAINDEXB	VBH	454	
	49	1500067	30613	138284	DD=INVENTRA	DBH	472	
	23	345595	15025	27613	DD=VENDORA	VBH	473	
	1	342952	342952	342952	PI=VENDORA...	1	498	
	1	14612	14612	14612	PI=VNSINDXA...	1	499	
	6	69203	11533	19492	DD=VNSINDXA	VBH	500	
TOTAL		136	3132672	23034				

以下の例は、領域別プログラム報告書の例を示しています。

IMS MONITOR	****PROGRAMS BY REGION****	TRACE START	1993 130	5:55:15	TRACE STOP	1993 130	5:59:49	PAGE 0069
		(A)		(B)				
**REGION	1	OCCURRENCES	ELAPSED EXECUTION TIME	SCHEDULING END TO FIRST CALL	DISTRIBUTION NUMBER			
		TOTAL	MEAN	MAXIMUM	TOTAL	MEAN	MAXIMUM	

PROGSC6D	1	290146255	290146255	290146255	15479797	15479797	15479797	885A,B
REGION TOTALS	1	290146255	290146255		15479797	15479797		
**REGION	2							
PROGIT8C	1	252290108	252290108	252290108	22376350	22376350	22376350	889A,B
REGION TOTALS	1	252290108	252290108		22376350	22376350		
**REGION	3							
PROGTS1C	1	259496970	259496970	259496970	15169488	15169488	15169488	893A,B
REGION TOTALS	1	259496970	259496970		15169488	15169488		
**REGION	4							
PROGSP3D	1	322812716	322812716	322812716	48146258	48146258	48146258	897A,B
REGION TOTALS	1	322812716	322812716		48146258	48146258		
**REGION	5							
PROGSP3A	2	62893103	31446551	40693590	5435	2717	2862	901A,B
PROGTS1B	1	61794787	61794787	61794787	2790	2790	2790	1271A,B
PROGSP3B	1	18294458	18294458	18294458	3104	3104	3104	1350A,B
PROGIT2B	1	36095342	36095342	36095342	2731	2731	2731	1363A,B
PROGSC2A	1	93902771	93902771	93902771	1667791	1667791	1667791	1401A,B
REGION TOTALS	6	272980461	45496743		1681851	280308		
**REGION	6							
PROGIT1B	2	39000315	19500157	23703429	5286	2643	2801	905A,B
PROGTS1B	1	34293636	34293636	34293636	3136	3136	3136	1207A,B
PROGSP3A	1	51887767	51887767	51887767	2534	2534	2534	1278A,B
PROGSP3B	2	67375031	33687515	40291430	17210570	8605285	17213287	1328A,B
PROGIT8A	1	69132416	69132416	69132416	3291	3291	3291	1359A,B
PROGSC4A	1	30165017	30165017	30165017	2571	2571	2571	1433A,B
REGION TOTALS	8	291854182	36481772		17193752	2149219		
**REGION	7							
PROGSC2B	1	269618583	269618583	269618583	5047875	5047875	5047875	909A,B
REGION TOTALS	1	269618583	269618583		5047875	5047875		
**REGION	8							
PROGIT8A	1	5181039	5181039	5181039	2928	2928	2928	913A,B
PROGSP3A	1	27304257	27304257	27304257	3350	3350	3350	1132A,B
PROGSC4B	1	37286872	37286872	37286872	3009	3009	3009	1255A,B
PROGIT2A	1	36902995	36902995	36902995	2850	2850	2850	1298A,B
PROGIT1B	1	30407479	30407479	30407479	2565	2565	2565	1336A,B
PROGIT1A	3	109875360	36625120	45190114	4279008	1426336	4272096	1357A,B
PROGIT8B	1	23405220	23405220	23405220	2679	2679	2679	1395A,B
REGION TOTALS	9	270363222	30040358		4296389	477376		

データベース処理意図の競合の検出

IMS モニターは、既にスケジュールされている別のアプリケーション・プログラムによって排他所有されているデータベースが更新されるのを待機していて、領域がアイドル状態になっている時間間隔を記録します。

DL/I 呼び出しの後に、IDLE FOR INTENT に合計アイドル時間、最大アイドル時間、および平均アイドル時間が表示されます。その領域のプログラムのスケジュールリングが失敗したときの経過時間は、その領域の要約行の時間に含まれます。

データベースの所有権が解放されている場合であっても、領域のスケジュールが失敗することがあります。意図障害によって処理が保留された回数は、「INTENT FAILURE SUMMARY」という表題の下に別個に記録されます。この報告書を、以下の例に示します。この報告書からは、セグメント・タイプおよび問題のデータベース名についての排他意図のために、どの PSB が競合状態にあるかが分かります。

INTENT FAILURE SUMMARY		
PSBNAME	DMBNAME	OCCURRENCES
SSTPSBNM	SSTDMBNM	1
TOTAL		1

チェックポイントの影響の検査

領域要約報告書の領域ごとの要約の最後にあるチェックポイント行に、モニター間隔中にシステム・チェックポイントがとられた回数、経過時間、および待ちなし時間が表示されます。

チェックポイント処理は、制御プログラムによって開始することができ、システム・ログに入れられるレコード数によって決まる指定の頻度で実行されます。他のチェックポイントは、オペレーター・コマンドによって取ることができます。

チェックポイント処理時の待ち時間は、領域待機報告書の最初の領域要約の終わりに示されます。DD 名とモジュール・コードの組み合わせごとに遅延を検出することができます。この部分の代表的な項目は、メッセージ・キュー・データ・セットおよび再始動データ・セットに関するものです。ストレージ待ちが原因である場合には、FUNCTION 欄の下の項目は、STG.= とその後続くプール ID になります。

領域占有の測定

領域アクティビティの尺度となるのは、領域占有の割合 (%) です。これは、一般的には、トレース間隔に対する、領域で処理を実行するときの経過時間の率です。

領域占有時間には、メッセージがないときの時間、ブロック・ロードが遅れたときの時間、あるいは PSB を使用できないときの時間は含まれません。領域要約報告書の最後のセクションには、時間測定が行われたイベントが収集されたすべてのアクティブ領域がリストされ、計算された領域占有百分率が示されます。

アプリケーション・プログラムの経過時間のモニター

IMS モニターは、各トランザクションの経過時間およびアプリケーション・プログラムのスケジューリングの経過時間の測定値を記録できます。このことは、他のプログラムが並行して実行されている間に、モニター間隔中に行われます。

経過時間は、最初の DL/I 呼び出し (または他の呼び出し) の開始からそのプログラムの終了までの時間から計算されます。アプリケーション・コードで費やされる時間と DL/I 処理で費やされる時間を区別することができます。以下の図は、イベントの間隔を示しています。

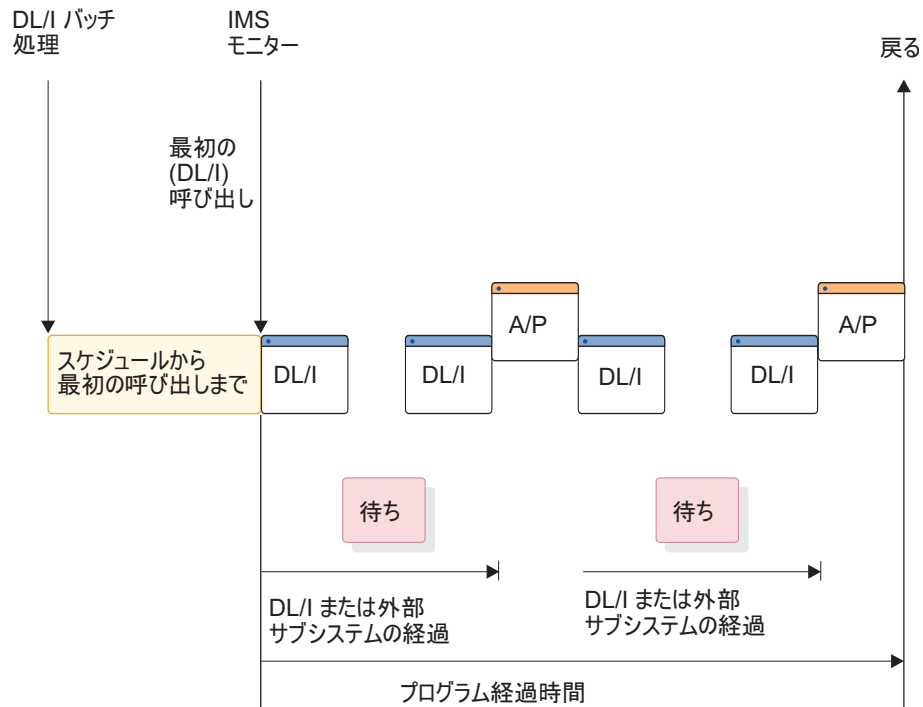


図 88. アプリケーション・コードと DL/I 処理の時間イベントの間隔

DL/I 呼び出しの経過時間内の、セグメント・データを入手するための待ち時間は、別個に記録されます。同様に、外部サブシステム呼び出しの経過時間内の、外部サブシステムでの処理時間は、待ち時間として別個に記録されます。入力待ちとして指定された領域の場合は、入力メッセージを待つのにかかった時間は、報告書に示される値から除外されます。アプリケーション処理 (A/P) 時間には、プログラム・オブジェクト・コードによって費やされるマシン・サイクルのほかに、多数の種類の副次的サービス (サブルーチンのロード、z/OS データ・セットへの入出力、およびオーバーレイ処理など) の時間が含まれます。ディスパッチされるのをプログラムが待機している場合、または実記憶域を使用するためにプログラムでページングが必要な場合には、これらに起因する遅れもアプリケーション・プログラム処理時間に含まれます。プログラムはスケジュールごとに多数のトランザクションを実行できるので、スケジュールから最初の呼び出しまでの経過時間は別個に記録されます。この時間には、アプリケーション・プログラムが実行する初期設定が含まれるとともに、プログラムのロードの時間が含まれます。

経過時間は、プログラム要約報告書に示されます。以下の例は、報告書の例です。プログラムは、報告書内の各行にある PSB 名で識別されます。それぞれの行に、測定された間隔中の各 PSB のアクティビティーの要約が示されています。すなわち、スケジュール数、DL/I 呼び出し数、完了した (デキューされた) トランザクション数、および DL/I 呼び出し、入出力呼び出し、外部サブシステム処理の待ち数の合計数が示されます。報告書の各行には、次のものについて計算された平均時間が示されています。

- スケジュール当たりの経過時間
- スケジュール当たりのプロセッサ時間
- スケジュール当たりの、スケジュールから最初の DL/I 呼び出しまでの経過時間
- トランザクション当たりの経過時間

この報告書には、以下のものも表示されます。

- トランザクション当たりの呼び出しの頻度
- DL/I 呼び出し当たりの入出力待ち回数
- 外部サブシステム呼び出し当たりの待ち回数
- スケジュール当たりの、デキューされたトランザクション数

TOTALS 行は、モニター間隔中にアクティブだった PSB のすべてのアクティビティを合計したものです。(PSB DUMMY 行は、スケジュール時に領域が停止したために不完全になったスケジュールリング、または疑似異常終了が起きたプログラムについて調整する行です。)

IMS MONITOR		****PROGRAM SUMMARY****				TRACE START 1993 130 5:55:15				TRACE STOP 1993 130 5:59:49				PAGE 0075
						(A).....(B).....				(A).....(B).....				
PSBNAME	NO. SCHEDS.	TRANS. DEQ.	CALLS /TRAN	CALLS /TRAN	I/O IWAITS	I/O IWAITS /CALL	TRAN. DEQD. /SCH.	CPU TIME /SCHED.	DISTR. NO.	ELAPSED TIME /SCHED.	SCHED. TO 1ST CALL /SCHED.	DISTR. NO.	ELAPSED TIME /TRANS.	
PROGSC6D	1	13	60	4.6	46	0.7	13.0	10010	884A,B	290146255	15479797	886A,B	22318942	
PROGIT8C	3	17	225	13.2	166	0.7	5.6	90592	888A,B	256617508	73283259	890A,B	45285442	
PROGTS1C	2	25	47	1.8	0	0.0	12.5	10010	892A,B	239190808	7586234	894A,B	19135264	
PROGSP3D	1	23	792	34.4	182	0.2	23.0	10010	896A,B	322812716	48146258	898A,B	14035335	
PROGSP3A	13	36	1246	34.6	267	0.2	2.7	49782	900A,B	32801812	2228611	902A,B	11845098	
PROGIT1B	11	21	99	4.7	0	0.0	1.9	6341	904A,B	23212388	2036217	906A,B	12158870	
PROGSC2B	7	155	3068	19.7	1845	0.6	22.1	346112	908A,B	93655514	789390	910A,B	4229603	
PROGIT8A	12	28	434	15.5	293	0.6	2.3	34350	912A,B	30196795	1745815	914A,B	12941483	
PROGSP2C	1	10	179	17.9	205	1.1	10.0	10010	916A,B	221024429	53642029	918A,B	22102442	
PROGTS1B	8	20	54	2.7	0	0.0	2.5	5447	920A,B	39943245	2895	922A,B	15977298	
PROGSP3C	1	14	468	33.4	117	0.2	14.0	10010	924A,B	310644485	35978027	926A,B	22188891	
PROGIT1C	1	9	32	3.5	0	0.0	9.0	10010	930A,B	304892631	30226173	932A,B	33876959	
PROGSC2C	1	9	160	17.7	101	0.6	9.0	10010	934A,B	296909110	22242652	936A,B	32989901	
PROGIT2B	8	21	393	18.7	63	0.1	2.6	21703	938A,B	35126671	1798496	940A,B	13381589	
PROGIT2C	6	17	211	12.4	39	0.1	2.8	13312	942A,B	288883508	50698467	944A,B	101958885	
PROGTS1D	2	26	50	1.9	0	0.0	13.0	10010	950A,B	284944505	10613350	952A,B	21918808	
PROGSP3B	8	22	770	35.0	169	0.2	2.7	35737	954A,B	38016279	2149158	956A,B	13824101	
PROGIT1A	11	24	106	4.4	0	0.0	2.1	7925	958A,B	30883486	1935855	960A,B	14154931	
PROGSC4A	9	163	1775	10.8	5101	2.8	18.1	235921	963A,B	62172947	3011199	965A,B	3432862	
PROGSC6C	1	10	44	4.4	38	0.8	10.0	10010	967A,B	228098334	46568124	969A,B	22809833	
PROGSP2B	11	28	557	19.8	604	1.0	2.5	35069	971A,B	33309266	1181831	973A,B	13085783	
PROGIT8D	1	12	175	14.5	133	0.7	12.0	10010	975A,B	253392289	21274169	977A,B	21116024	
PROGSC4C	1	10	98	9.8	349	3.5	10.0	10010	979A,B	248736332	25930126	981A,B	24873633	
PROGSC6A	7	157	789	5.0	457	0.5	22.4	11703	983A,B	73936039	115979	985A,B	3296511	
PROGIT2A	7	22	430	19.5	71	0.1	3.1	28529	987A,B	37905001	2982	989A,B	12060682	
PROGSC2D	1	15	280	18.6	180	0.6	15.0	10010	991A,B	316194222	41527764	993A,B	21079614	
PROGSP2A	6	25	490	19.6	548	1.1	4.1	43177	995A,B	58277945	2467506	997A,B	13986707	
PROGSC2A	5	121	2363	19.5	1420	0.6	24.2	276187	1001A,B	88906184	6022954	1003A,B	3673809	
PROGIT2D	1	20	361	18.0	62	0.1	20.0	10010	1005A,B	386092737	111426279	1007A,B	19304636	
PROGSC4B	10	131	1421	10.8	4115	2.8	13.1	617016	1011A,B	53826667	2632409	1013A,B	4108905	
PROGSC4D	1	19	197	10.3	668	3.3	19.0	10010	1020A,B	227999124	46667334	1022A,B	11999953	
PROGSP2D	1	13	240	18.4	291	1.2	13.0	10010	1025A,B	327602445	52935987	1027A,B	52500188	
PROGSC6B	5	140	694	4.9	395	0.5	28.0	16884	1032A,B	78994223	3290769	1034A,B	2821222	
PROGIT1D	1	10	36	3.6	0	0.0	10.0	10010	1041A,B	290379922	15713464	1043A,B	29037992	
PROGIT8B	8	17	288	16.9	190	0.6	2.1	33436	1259A,B	35223857	2902	1261A,B	16575932	
**TOTALS	173	1403	18632	13.2	18115	0.9	8.1	90328		82303700	2972755		10148638	

呼び出し要約報告書を使用すれば、プログラムごとの呼び出し処理の詳細を調べることができます。この報告書は、呼び出しタイプ別に項目が分けられており、モニター間隔について要約されています。複数ページの出力からの抜き出しを、以下の例に示してあります。入出力 PCB を使用した呼び出しが最初に示されて、小計が与えられています。次に、各データベース PCB および各外部サブシステムに対する、各タイプの呼び出しの合計がリストされています。PSB TOTAL 行は、各プログラムのデータの最後を表します。

IMS MONITOR		****CALL SUMMARY****				TRACE START 1993 130 5:55:15				TRACE STOP 1993 130 5:59:49				PAGE 0186
						(C)				(B)				
PSB NAME	PCB NAME	CALL FUNC	LEV NO. SEGMENT	STAT CODE	CALLS	IWAITS	IWAITS/ CALL	..ELAPSED MEAN	TIME... MAXIMUM	.NOT MEAN	IWAIT TIME.. MAXIMUM	DISTRIB. NUMBER		
PROGSC6B	I/O PCB	ISRT ()	()		138	0	0.00	372	1240	372	1240	598A,B,C		
		GU ()	()		134	133	0.99	2600917	20974615	2587532	20962866	602A,B,C		
		(GU) ()	()		3	0	0.00	15	16		16	716A,B,C		
		ASRT ()	()		3	0	0.00	330	333	330	333	869A,B,C		
		GU ()	()	QC	2	1	0.50	17639806	21219588	17634776	21209529	870A,B,C		
		I/O PCB	SUBTOTAL											

			280	134	0.47	1370910		1364469		
INVENTRB	DLET	(03)IN060SUP	138	0	0.00	813	1289	813	1289	599A,B,C
	GNP	(03)IN060SUP	138	7	0.05	2112	112589	1047	112589	600A,B,C
	GU	(01)IN010PAR	138	254	1.84	29511	75356	1195	19229	601A,B,C
	DL/I	PCB SUBTOTAL								
			414	261	0.63	10812		1018		
		PSB TOTAL								
PROGSC2A	I/O PCB		694	395	0.56	559555		551114		
	ISRT	()	118	0	0.00	381	1496	381	1496	603A,B,C
	GU	()	114	284	2.49	3304809	21784513	3164423	21664181	632A,B,C
	(GU)	()	2	0	0.00	17	18	17	18	781A,B,C
	ASRT	()	3	0	0.00	367	444	367	444	871A,B,C
	GU	()	2	5	2.50	19931897	20045206	19799530	19925277	872A,B,C
		I/O PCB SUBTOTAL								
			239	289	1.20	1743339		1675270		
LOGVENDA	REPL	(03)IN040SLQ	118	0	0.00	268	804	268	804	604A,B,C
	GNP	(03)IN040SLQ	118	5	0.04	899	16995	218	305	605A,B,C
	REPL	(02)VN030PAR	826	0	0.00	805	1578	805	1578	606A,B,C
	GNP	(02)VN030PAR	826	873	1.05	19321	94521	456	1363	607A,B,C
	REPL	(01)VN020REO	118	58	0.49	8879	48076	832	1682	623A,B,C
	GU	(01)VN020REO	118	195	1.65	31688	360775	1300	1746	625A,B,C
	DL/I	PCB SUBTOTAL								
			2124	1131	0.53	10145		636		
		PSB TOTAL								
PROGSC2D	I/O PCB		2363	1420	0.60	185445		170013		
	ISRT	()	14	0	0.00	377	621	377	621	608A,B,C
	GU	()	14	36	2.57	22360408	52048566	22221852	51901313	634A,B,C
		I/O PCB SUBTOTAL								
			28	36	1.28	11180393		11111115		
LOGVENDD	REPL	(03)IN040SLQ	14	0	0.00	263	328	263	328	609A,B,C
	GNP	(03)IN040SLQ	14	1	0.07	1407	16889	223	307	610A,B,C
	REPL	(02)VN030PAR	98	0	0.00	820	1015	820	1015	611A,B,C
APOL1	I/O PCB									
	ASRT	(..)	3							
	(GU)	(..)	3							
	INQY	(..)	2							
		I/O PCB SUBTOTAL								
			8							
OTMDEST1	ICAL	(..)	4							
	DL/I	PCB SUBTOTAL								
			4							
		PSB TOTAL								
			12							

アプリケーション・プログラム DL/I 呼び出しのための入出力のモニター

IMS モニター報告書は、モニター間隔中の入出力オカレンスの合計数と、実行されたアプリケーション・プログラムごとに入出力オカレンスにかかった合計時間を示します。

プログラム入出力報告書は、モニター間隔中にアクティブだったすべての PSB についてこれら 2 つの合計を示し、プログラムが使用する各 PCB によって引き起こされた入出力待ち時間の詳細な内訳を含んでいます。

この報告書は、アプリケーション・プログラムの処理中に起きた競合を示します。競合の各タイプおよび起きた回数は、入出力 PCB またはデータベース PCB ごとに記録されます。この報告書は、合計待ち時間、最大待ち時間、および平均時間を示します。1 つの PSB のもとにある PCB ごとに、そして各 PSB のもとにあるすべての PCB について小計が示されます。

DDN/FUNC 欄には、データ・セット DD 名がリストされます。MODULE 欄では、競合のソースを示すためにコードを使用しています。競合のタイプおよびコードを次に示します。

- メッセージ処理

コード 競合

DBH メッセージ・キューでの OSAM 入出力

MFS MFS 形式ライブラリー・ディレクトリー

- PMM** メッセージ形式バッファ・プール・スペースまたは制御ブロック入出力
- QMG** メッセージ・キュー管理
- スケジューリング
 - コード 競合
 - BLR** ACBLIB からのロード/読み取り
 - MSC** MPP 領域の初期設定
 - SMN** 仮想記憶管理
 - データベースのアクセス
 - コード 競合
 - DBH** OSAM 入出力
 - DLE** DL/I 機能
 - VBH** VSAM インターフェース
 - (物理セグメント・コード)
 - プログラム分離

外部サブシステム呼び出しの場合には、処理の完了にかかる経過時間は待ち時間と見なされます。DDN/FUNC 欄では、次に示す外部サブシステム呼び出し機能を示します。

- 外部サブシステム
 - コード サブシステム呼び出し機能
 - AB0** ABORT
 - CT0** スレッドの作成
 - D50** 識別またはスレッドの終了、サインオフ
 - D80** INIT
 - I30** 識別、コマンド、エコー、終了
 - I30** 識別、サブシステムの終了
 - I50** INIT
 - I60** 疑わしいリソース解決
 - PR0** サブシステム作動不能
 - P10** コミット準備 (フェーズ 1)
 - P20** コミット続行 (フェーズ 2)
 - SO0** サインオン出口ルーチン
 - SI0** 識別出口ルーチン

以下の例は、報告書の例を示しています。

```

IMS MONITOR  ****PROGRAM I/O****          TRACE START 1993 022  14:00:18  TRACE STOP 1993 022  14:02:20  PAGE 0088
.....IWAIT TIME.....
PSBNAME  PCB NAME  IWAIT  TOTAL  MEAN  MAXIMUM  DDN/FUNC  MODULE
-----  -

```

PROGHR1A I/O PCB	122	2341116	19189	70795	HOTELDBA	DBH
	34	24177936	711115	3950160	**W F I	
	40	23652665	591316	2668917	**W F I	
	5	67613	13522	21214	SHMSG	QMG
	4	110363	27590	60486	QBLKS	QMG

PCB TOTAL

PSB TOTAL	131	2519092	19229			
___	305	6725063	20049			

PROGDE1A TRMNALDA	20	624677	31233	68252	TRMNALDA	VBH
	1	275811	275811	275811	PI TRMNALDA....	

PCB TOTAL

I/O PCB	21	900488	42880			
	16	488812	30550	79980	TRMNALDA	VBH
	1	16118	16118	16118	SHMSG	QMG

PCB TOTAL

TABLEDBA	17	504930	29701			
	16	290471	18154	33254	TABLEDA	DBH

PCB TOTAL

PSB TOTAL	16	290471	18154			
___	54	1695889	31405			

PROGHR2B HOTELDBB	8	698384	87298	184475	HOTELDBB	DBH
	4	5820650	1455162	1455278	PI HOSINDEXB....	
	4	4481024	1120256	1209075	PI HOTELDBB....	
	2	260817	130408	232750	HOSINDEXB	VBH
	7	106623	15231	16410	HOSINDEXB	VBH
	1	15366	15366	15366	HOTELDBD	DBH

PCB TOTAL

PSB TOTAL	26	11382864	437802			
___	26	11382864	437802			

PROGHR2A HOTELDBA	17	655801	38576	366108	HOSINDEXA	VBH
	73	1836721	25160	82141	HOTELDBA	DBH
	2	54663	27331	41975	HOTELDBD	DBH
	1	9887	9887	9887	HOTELDBC	DBH
	2	851042	845635	845635	HOSINDOA	VBH

PCB TOTAL

I/O PCB	95	3408114	35874			
	20	575847	28792	74227	HOTELDBA	DBH
	21	370390	17637	43153	HOSINDEXA	VBH

IMS MONITOR ****PROGRAM I/O**** TRACE START 1993 022 14:00:18 TRACE STOP 1993 022 14:02:20 PAGE 0089

.....IWAIT TIME.....

PSBNAME	PCB NAME	IWAITS	TOTAL	MEAN	MAXIMUM	DDN/FUNC	MODULE
PROGHR2A	I/O PCB	5	4654544	930908	2020043	**W F I	
		8	32796604	4099575	9328891	**W F I	

PCB TOTAL

PSB TOTAL	41	946237	23078			
___	136	4354351	32017			

PROGPS2A LOGIMA	89	2046670	22996	73593	IMMSTR3A	VBH
	612	53886417	88049	185674	IMMSTR1A	VBH
	3	44906	14968	20788	IMINDEXA	VBH

PCB TOTAL

	704	55977993	79514			
	469	11742900	25038	170337	COMPOSDA	DBH
	329	8198418	24919	91422	CPINDEXA	VBH

PCB TOTAL

I/O PCB	798	19941318	24989			
	3	47511	15837	20806	SHMSG	QMG

PCB TOTAL

___	3	47511	15837			
-----	---	-------	-------	--	--	--

PSB TOTAL	1505	75966822	50476			
PROGSC6C I/O PCB	52	2698602	51896	473763	INVENTRC	VBH
	4	70921	17730	34241	SHMSG	QMG
	3	50699	16899	24724	QBLKS	QMG
PCB TOTAL	59	2820222	47800			
	55	2666884	48488	210752	INVENTRC	VBH
	50	797587	15951	41706	ININDEXC	VBH
	1	119253	119253	119253	PI INVENTRC...	1
	1	8634	8634	8634	INVENTRB	VBH
	2	83947	41973	53936	INVENTRA	VBH
PCB TOTAL	109	3676305	33727			
PSB TOTAL	168	6496527	38669			
PROGHR2D I/O PCB	21	2285296	108823	199223	HOTELDBD	DBH
	28	762370	27227	111860	HOSINDXD	VBH
	1	11685	11685	11685	SHMSG	QMG
PCB TOTAL	50	3059351	61187			
HOTELDBD	96	6279107	65407	139032	HOTELDBD	DBH

MONITOR ****PROGRAM I/O**** TRACE START 1993 022 14:00:18 TRACE STOP 1993 022 14:02:20 PAGE 0090							
.....IWAIT TIME.....							
PSBNAME	PCB NAME	IWAITS	TOTAL	MEAN	MAXIMUM	DDN/FUNC	MODULE
PROGHR2D	HOTELDBD	31	2130585	68728	769130	HOSINDXD	VBH
		3	115999	38666	56394	HOTELDBA	DBH
		2	69833	34916	43470	HOTELDBC	DBH
		2	41430	20715	28020	HOSINDOD	VBH
		4	5515374	1378843	1458884	PI HOSINDXD....	
		4	3997017	999254	1026228	PI HOTELDBD....	
PCB TOTAL		142	18149345	127812			
PSB TOTAL		192	21208696	110461			

入出力 PCB の呼び出しのための入出力待ちは、PSB の最初の項目としてグループ化されています。DL/I 呼び出しの場合、入出力が行われたデータ・セットは DDN/FUNC 見出しの下に示され、モジュール・コードはどのようなタイプの競合が待機の原因になったかを示します。外部サブシステム呼び出しの場合は、機能が DDN/FUNC 見出しの下に示され、モジュール・コードにより呼び出し項目の原因が示されます。

LGMSG および SHMSG 以外の名前が、入出力 PCB の DDN/FUNC 欄の中に現れることがあります。例えば、データベース・バッファの書き込みを引き起こすアプリケーション・プログラム (入出力 PCB を使用) によって出されたチェックポイント呼び出しがあります。

プログラムが入力待ちとして指定されており、次のメッセージの入力を待たなければならぬ場合は、待機項目が DDN/FUNC の見出しの下に **WFI で示され、MODULE 欄には何の項目も示されません。次の入力メッセージを待つのにかかる時間は、待ち時間のもとに示されます。**WFI 項目は、参考情報として示されるのみで、それらの値は統計の計算には使用されません。

データベース内の同じ物理セグメントの競合は、プログラム分離のための待機の原因になります。これは、PCB 行の DDN/FUNC 欄の Pldmb 項目 (ここで、dmb

は物理データ・セットの DMB) によって示されます。MODULE 欄では、DBD 生成で割り当てられた物理セグメント・コードを使用してセグメント・タイプが識別されます。

アプリケーションが、アクセス方式として VSAM を使用してデータベースにアクセスしているときは、DL/I 呼び出しでは一般に入出力待ちになりません。MODULE 欄の項目 VBH は、VSAM のインターフェースが起きて入出力待ちがあったことを示します。

関係ないと思われる項目が、データベース PCB の DDN/FUNC 欄の下に表示されることがあります。その一例としては、検索されるデータ用の余地を作るためにバッファの除去が行われるような、データベース (DB-A) への検索呼び出しがあります。バッファの内容が別のデータベース (DB-B) に属するデータを含んでいると、報告書の入出力項目では、DB-B の DD 名が、DB-A への PCB アクセスに関して競合関係にあるものとして示されます。

MFS アクティビティのモニター

メッセージ形式バッファ・プール報告書からは、メッセージ形式バッファ・プールの使用の管理に関して行われるすべてのアクティビティの要約を知ることができます。

この報告書を、以下の例に示します。このデータは、トレース間隔の開始時および終了時のカウントとそれらの差を示します。

メッセージのフォーマット設定が行われるときに、適切なメッセージ・ブロックがメッセージ形式バッファ・プール内に存在していなければなりません。そのブロックは、入力の場合には DIF/MID の対、出力の場合には DOF/MOD の対です。これらのブロックがバッファにまだない場合には、アクティブ

IMS.FORMATA/B ライブラリーへの入出力を行う必要があります。ブロック検索では、プール内にある索引を使用した先行ディレクトリー探索 (つまり、直接的な検索) も行われることがあります。

多数のカウント値により、内部イベント管理の詳細が明らかにされます。ブロックのディレクトリー項目がない場合は、余分のディレクトリー検索 I/O を意味します。

I M S M O N I T O R BUFFER POOL STATISTICS TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0007

MESSAGE FORMAT BUFFER POOL

	5:55:15 START TRACE	5:59:49 END TRACE	DIFFERENCE
NUMBER OF P/F REQUESTS	0	0	0
NUMBER OF I/F REQUESTS	18	20	2
NUMBER OF I/F I/O'S	2	2	0
NUMBER OF TIMES POOL COMPRESS WOULD BE SUCCESSFUL	0	0	0
NUMBER OF DIRECTORY I/O OPERATIONS	2	2	0
NUMBER OF TIMES BLOCK WASHED FOR FRE	0	0	0
NUMBER OF TIMES P/F REQUEST IGNORED	0	0	0
NUMBER OF F/B REQUESTS	18	20	2
NUMBER OF TIMES F/B REQUEST IGNORED	0	0	0
NUMBER OF TIMES I/F ON F/B QUEUE	16	18	2
NUMBER OF TIMES I/F ON I/F QUEUE	0	0	0
NUMBER OF TIMES F/B ON I/F QUEUE	18	20	2
NUMBER OF TIMES P/F ON I/F QUEUE	0	0	0
NUMBER OF TIMES P/F ON F/B QUEUE	0	0	0
NUMBER OF TIMES THERE WAS NO DIR ENTR FOR A BLOCK	0	0	0

```

NUMBER OF TIMES I/O ERRORS POINT OR READ MACRO          0          0          0
NUMBER OF IMMEDIATE I/O REQUESTS WAITED DUE TO MAXIMUM I/O 0          0          0
NUMBER OF REQUESTS SATISFIED BY INDEX/DYNAMIC DIRECTORY 0          0          0

```

QUOTIENT : IMMEDIATE FETCH I/O'S + DIRECTORY I/O'S OPERATIONS = 0.00

TOTAL NUMBER OF TRANSACTIONS

メッセージ・キュー処理のモニター

トランザクション処理の効率に直接影響する重要なリソースは、メッセージ・キュー・プール、およびメッセージ・キューの入出力管理です。メッセージ・キュー・プール報告書を見れば、そのアクティビティーを調べることができます。

以下の例は、メッセージ・キュー・プール報告書を示しています。トレース間隔の開始時と終了時のアクティビティーのカウンタとともに、開始時の数と終了時の数の差を示しています。

I M S M O N I T O R BUFFER POOL STATISTICS TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0002

MESSAGE QUEUE POOL

	5:55:15 START TRACE	5:59:49 END TRACE	DIFFERENCE
NUMBER OF LOCATE CALLS FROM QMGR	54204	68436	14232
NUMBER OF RECORD RELEASE CALLS FROM QMGR	16431	20738	4307
NUMBER OF LOCATE AND ALTER CALLS FROM QMGR	131593	164744	33151
NUMBER OF REQUESTS TO PURGE THE Q POOL	2	2	0
NUMBER OF ADDRESS TO DRRN TRANSLATION REQUESTS	21351	27076	5725
NUMBER OF REQUESTS TO WAIT FROM QMGR	0	0	0
NUMBER OF READ REQUESTS	962	962	0
NUMBER OF WRITE REQUESTS(TOTAL)	499	499	0
NUMBER OF WRITES DONE BY PURGE	499	499	0
NUMBER OF WAITS FOR PURGE COMPLETION	1	1	0
NUMBER OF WAITS BECAUSE NO BUFFER AVAILABLE	0	0	0
NUMBER OF WAITS FOR OTHER DECB TO READ THIS BUFFER	823	823	0
NUMBER OF WAITS FOR OTHER DECB TO WRITE THIS BUFFER	0	0	0
NUMBER OF WAITS FOR CONFLICTING END DEQ BUFFER REQ	0	0	0
NUMBER OF PSBS UNCHAINED FROM BUFFERS	0	0	0
NUMBER OF CALLS TO QMGR.(TOTAL)	48164	62213	14049
NUMBER OF CALLS TO REPOSITION A LOST BUFFER	0	0	0
NUMBER OF CALLS TO ENQ A MESSAGE	10583	13441	2858
NUMBER OF CALLS TO DEQ ONE OR MORE MESSAGE	6321	7767	1446
NUMBER OF CALLS TO CANCEL INPUT OR OUTPUT	119	121	2

QUOTIENT : TOTAL NUMBER OF OSAM READS + OSAM WRITES + ALL IWAITS = 0.00

TOTAL NUMBER OF TRANSACTIONS

チェックポイントの影響の検出

チェックポイント・コマンドで SNAPQ を指定すると、すべてのメッセージ・キューの現在状況がシステム・ログに書き込まれます。これは、キュー管理のためのメッセージ処理を妨げてしまいます。

一般的な IWAIT 時間イベントには、SNAPQ によって生じる待ち時間が記録されます。以下の例では、要約行 QMGR SNAPQ CHECK にアクティビティーを示しています。オカレンスの数が、合計待ち時間、平均待ち時間、および最大待ち時間と一緒に示されています。

```

IMS MONITOR ** GENERAL REPORTS ** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0009
GENERAL IWAIT TIME EVENTS
EVENT          OCCURRENCES  TOTAL  MEAN  MAXIMUM  DISTRIBUTION
IWAITS
QMGR SNAPQ CHECK  0          0          0          0          0
REGION AND JOBNAME REPORT

```

REG. NO.	JOB NAME
1	MPR1A100
2	MPR1A209
3	MPR1A210
4	MPR1A211
5	MPR1A103
6	MPR1A101
7	MPR1A115
8	MPR1A116
9	MPR1A216
10	MPR1A200
11	MPR1A217
12	MPR1A119
13	MPR1A218
14	MPR1A219
15	MPR1A104
16	MPR1A220
17	MPR1A203
18	MPR1A123
19	MPR1A222
20	MPR1A105
21	MPR1A124
22	MPR1A223
23	MPR1A107
24	MPR1A224
25	MPR1A106
26	MPR1A206
27	MPR1A205
28	MPR1A108
29	MPR1A109
30	MPR1A208
31	MPR1A111
32	MPR1A112
33	MPR1A113
34	MPR1A204
35	MPR1A114
36	MPR1A102
48	MPR1A121
49	MPR1A122
50	MPR1A221

トランザクション・キューイング報告書

メッセージ処理の効率をモニターするだけでなく、トランザクション処理プログラムのスケジューリングのつど、トランザクション・キューのサイズを調べることに
より、各アプリケーションごとに、提供されたサービスをモニターすることができます。

以下の例に示されているトランザクション・キューイング報告書は、トランザク
ションごとの、スケジューリング時の最小カウント、平均カウント、および最大カウ
ントを記録したものです。モニター間隔中にデキューされたトランザクション (ま
たは完全に処理されたトランザクション) の合計数が、トランザクション・コード
ごとに示されています。スケジューリングごとの処理されたトランザクションの平
均数は、DEQUEUED MEAN 欄に示してあります。

IMS MONITOR ****TRANSACTION QUEUING**** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0181									
TRANSACTION	NUMBER DEQUED	NUMBER SCHEDS.	(B)			(A)		DISTRIBUTION NUMBER	
			..ON QUEUE MINIMUM	WHEN SCHEDULED MEAN MAXIMUM	DEQUED MEAN			
SC6X	13	1	0	0.00	0	13.00	883A,B		
IT8W	17	3	0	0.00	0	5.66	887A,B		
TS1Z	16	1	0	0.00	0	16.00	891A,B		
PS3X	23	1	0	0.00	0	23.00	895A,B		
PS3Y	17	7	0	0.00	0	2.42	899A,B		
IT1V	11	6	0	0.00	0	1.83	903A,B		
SC2Z	143	2	0	0.00	0	71.50	907A,B		
IT8U	12	7	0	0.00	0	1.71	911A,B		
PS2W	10	1	0	0.00	0	10.00	915A,B		

TS1U	12	4	0	0.00	0	3.00	919A,B
PS3W	14	1	0	0.00	0	14.00	923A,B
IT8Y	16	5	0	0.00	0	3.20	927A,B
IT1W	9	1	0	0.00	0	9.00	929A,B
SC2W	9	1	0	0.00	0	9.00	933A,B
IT2V	13	5	0	0.00	0	2.60	937A,B
IT2W	17	6	0	0.00	0	2.83	941A,B
TS1V	9	1	0	0.00	0	9.00	945A,B
SC2V	12	5	0	0.00	0	2.40	947A,B
TS1W	11	1	0	0.00	0	11.00	949A,B
PS3V	13	3	0	0.00	0	4.33	953A,B
IT1U	9	6	0	0.00	0	1.50	957A,B
SC4U	11	5	0	0.00	0	2.20	962A,B
SC6W	10	1	0	0.00	0	10.00	966A,B
PS2V	8	6	0	0.00	0	1.33	970A,B
IT8X	12	1	0	0.00	0	12.00	974A,B
SC4W	10	1	0	0.00	0	10.00	978A,B
SC6U	14	6	0	0.00	0	2.33	982A,B
IT2Y	9	3	0	0.00	0	3.00	986A,B
SC2X	15	1	0	0.00	0	15.00	990A,B
PS2Y	17	2	0	0.00	0	8.50	994A,B
SC4Y	152	4	0	0.50	1	38.00	998A,B
SC2Y	106	2	0	0.00	0	53.00	1000A,B
IT2X	20	1	0	0.00	0	20.00	1004A,B
SC2U	15	3	0	0.00	0	5.00	1008A,B
SC4Z	123	5	0	0.60	1	24.60	1010A,B
TS1X	15	1	0	0.00	0	15.00	1015A,B
SC4X	19	1	0	0.00	0	19.00	1019A,B
PS2X	13	1	0	0.00	0	13.00	1024A,B
PS2Z	20	5	0	0.00	0	4.00	1028A,B
SC6Z	130	1	0	0.00	0	130.00	1031A,B
SC6V	10	4	0	0.00	0	2.50	1035A,B
SC6Y	143	1	0	0.00	0	143.00	1037A,B
IT1X	10	1	0	0.00	0	10.00	1040A,B
PS3U	19	6	0	0.00	0	3.16	1131A,B
IT2U	13	4	0	0.00	0	3.25	1146A,B

データベース・バッファのモニター

オンライン・システムにおける重要なリソースの 1 つは、データベース・バッファ・プールです。DL/I 呼び出しサービスの効率は、必要なデータベース論理レコードがこのバッファ内にあるかどうかによって異なります。つまり、バッファ内であれば、セグメント検索では追加の入出力が不要になります。

これは特に、置き換え呼び出しの前にデータベース呼び出しが介在する HOLD (保持) 呼び出しの場合に当てはまります。

QUERY POOL コマンドは、高速機能バッファ・プールの使用統計を表示する際に使用します。

以下の例に示されているデータベース・バッファ・プール報告書を使用して、全機能プール管理の一般的な効率を査定することができます。この報告書に示されているイベント・カウントは、特定のデータベースまたはプログラムに固有のものでなく、データベース・プールの使用状況を示しています。

関連資料: データベース・バッファ・プール報告書の詳細については、852 ページの『データベース・バッファ・プール報告書』を参照してください。

いずれかのデータベースがアクセス方式として VSAM を使用する場合、IMS モニターは VSAM BUFFER POOL の表題の一連の報告書を (サブプールごとに 1 つずつ) 生成します。以下の例は、これらの報告書の 1 つです。

DATA BASE BUFFER POOL

FIX PREFIX/BUFFERS

Y/Y

SUBPOOL ID 004K
 SUBPOOL BUFFER SIZE 4096
 TOTAL BUFFERS IN SUBPOOL 1000

	17:08:15	17:10:16	
NUMBER OF LOCATE-TYPE CALLS	1117674	1676213	558539
NUMBER OF REQUESTS TO CREATE NEW BLOCKS	0	0	0
NUMBER OF BUFFER ALTER CALLS	215874	322936	107062
NUMBER OF PURGE CALLS	25077	37454	12377
NUMBER OF LOCATE-TYPE CALLS, DATA ALREADY IN OSAM POOL	870306	1301187	430881
NUMBER OF BUFFERS SEARCHED BY ALL LOCATE-TYPE CALLS	1258247	1886843	628596
NUMBER OF READ I/O REQUESTS	238165	360260	122095
NUMBER OF SINGLE BLOCK WRITES BY BUFFER STEAL ROUTINE	0	0	0
NUMBER OF BLOCKS WRITTEN BY PURGE	95057	142413	47356
NUMBER OF LOCATE CALLS WAITED DUE TO BUSY ID	780	1297	517
NUMBER OF LOCATE CALLS WAITED DUE TO BUFFER BUSY WRT	0	0	0
NUMBER OF LOCATE CALLS WAITED DUE TO BUFFER BUSY READ	0	0	0
NUMBER OF BUFFER STEAL/PURGE WAITED FOR OWNERSHIP RLSE	178	261	83
NUMBER OF BUFFER STEAL REQUESTS WAITED FOR BUFFERS	0	0	0
TOTAL NUMBER OF I/O ERRORS FOR THIS SUBPOOL	0	0	0
NUMBER OF BUFFERS LOCKED DUE TO WRITE ERRORS	0	0	0
QUOTIENT : TOTAL NUMBER OF OSAM READS + OSAM WRITES =	6.98		

TOTAL NUMBER OF TRANSACTIONS			

以下の例は、VSAM バッファ・プール報告書の例を示しています。

I M S M O N I T O R BUFFER POOL STATISTICS

V S A M B U F F E R P O O L

FIX INDEX/BLOCK/DATA N/Y/N
 SHARED RESOURCE POOL ID VPL1
 SHARED RESOURCE POOL TYPE D
 SUBPOOL ID 2
 SUBPOOL BUFFER SIZE 4096
 NUMBER HIPERSPACE BUFFERS 50
 TOTAL BUFFERS IN SUBPOOL 1000

	17:08:15	17:10:16	
NUMBER OF RETRIEVE BY RBA CALLS RECEIVED BY BUF HNDLR	152	330	178
NUMBER OF RETRIEVE BY KEY CALLS	117780	178424	60644
NUMBER OF LOGICAL RECORDS INSERTED INTO ESDS	132	310	178
NUMBER OF LOGICAL RECORDS INSERTED INTO KSDS	6460	9853	3393
NUMBER OF LOGICAL RECORDS ALTERED IN THIS SUBPOOL	0	0	0
NUMBER OF TIMES BACKGROUND WRITE FUNCTION INVOKED	0	0	0
NUMBER OF SYNCHRONIZATION CALLS RECEIVED	18566	27923	9357
NUMBER OF WRITE ERROR BUFFERS CURRENTLY IN THE SUBPOOL	0	0	0
LARGEST NUMBER OF WRITE ERRORS IN THE SUBPOOL	0	0	0
NUMBER OF VSAM GET CALLS ISSUED	124648	189220	64572
NUMBER OF VSAM SCHBFR CALLS ISSUED	0	0	0
NUMBER OF TIMES CTRL INTERVAL REQUESTED ALREADY IN POOL	33662	51088	17426
NUMBER OF CRTL INTERVALS READ FROM EXTERNAL STORAGE	91169	138505	47336
NUMBER OF VSAM WRITES INITIATED BY IMS	6022	9251	3229
NUMBER OF VSAM WRITES TO MAKE SPACE IN THE POOL	0	0	0
NUMBER OF VSAM READS FROM HIPERSPACE BUFFERS	0	0	0
NUMBER OF VSAM WRITES FROM HIPERSPACE BUFFERS	50	50	0
NUMBER OF FAILED VSAM READS FROM HIPERSPACE BUFFERS	0	0	0
NUMBER OF FAILED VSAM WRITES FROM HIPERSPACE BUFFERS	0	0	0
QUOTIENT : TOTAL NUMBER OF VSAM READS + VSAM WRITES =	2.08		

TOTAL NUMBER OF TRANSACTIONS			

回線アクティビティのモニター

モニター対象の間隔中にメッセージ・トラフィックを処理するノードごとに、アクティビティのすべてのオカレンスの要約を入手できます。経過時間と待ちなし時間については、通信要約報告書の中で通信回線ごとに、合計時間、平均時間、および最大時間のカテゴリーで示してあります。

以下の例は、この報告書を示しています。

どの物理装置が回線を使用中であるかを、システム定義のステージ 1 の出力と突き合わせる必要があります。回線番号は、ステージ 1 の入力デッキ内の物理的なオカレンスに従って順次に割り当てられます。

オンライン・システムで、制御領域の JCL に MFS ブロックの事前取り出しオプションを指定している場合は、報告書の最後の行に、すべての事前取り出しイベントの統計が表示されます。

```

IMS MONITOR ****COMMUNICATION SUMMARY**** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0089
              (A)                                     (B)
              .....ELAPSED TIME.....              NOT IWAIT TIME(ELAPSED-IWAIT)
              TOTAL      MEAN      MAXIMUM          TOTAL      MEAN      MAXIMUM          DISTRIBUTION
              OCCURRENCES                                     NUMBER
-----
PMT01A      3          2396      798      1547          2396      798      1547          1467A,B
  19        182         92155     506     1106          92155     506     1106          1493A,B
  2         59          2280      38       41           2280      38       41           1515A,B
  TOTAL
-----
              244         96831     396          96831     396
  
```

さらに、ノード用に伝送されたデータの量を、回線機能報告書で調べることができます。以下の例は、この報告書を示しています。この報告書では、入力データと出力データを区別しています。データ・ブロックの数およびブロックの平均サイズと最大サイズが、IMS が受信したデータと送信したデータのそれぞれについて記録されています。

この報告書は、回線がどのくらい非アクティブであるかを示す尺度も含んでいます。非アクティブ間隔は、最後の入力ブロックの受信の終了時刻と出力伝送の開始時刻の差と見なされます。これらの非アクティブ状態のオカレンスは、送受反転間隔と呼ばれており、この報告書では、これらの間隔に関連する平均時間と最大時間、ならびにオカレンスの回数もまとめて載せています。

回線を使用しているのが MFS サポート端末である場合は、複数ページ・メッセージで次のページを要求した回数のカウントも記録されます。

複数システムを結合したものについてのリンク・トラフィックが記録される場合は、3 つの報告書が回線機能報告書の後に続きます。これらについては、899 ページの『MSC の IMS モニター報告書』に説明があります。

```

IMS MONITOR ****LINE FUNCTIONS*** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0091
              (A)                                     (B)
              .....                               .....
              MEAN MAX. MEAN MAX. MEAN MAX. MEAN MAX. MEAN MAX. MEAN MAX. MEAN MAX.
              LINE NUMBER DEVICE RECEIVE RECEIVE RECEIVE TRANS. TRANS. TRANS. TRANS. TRANS. TRANS. TRANS. TRANS. TRANS. TRANS.
              BLOCKS BLKSIZE BLKSIZE BLOCKS BLKSIZE BLKSIZE NUMBER INTERVAL INTERVAL INTERVAL INTERVAL INTERVAL INTERVAL
-----
PMT01A      3270V      1      29      29      2      170     171     1468A,B      3      798     1547     1466
  19        XXXX      182     506     1106     182     506     1106     1492
  2         LOC SYS      59       38       41       59       38       41     1514
  ALL LINES
-----
              1      29      29      2      170     171     1468A,B      244     396     1466
  
```

メッセージ処理効率のモニター

IMS モニターは、IMS 制御領域で行われる非同期処理に関する要約と詳細情報の両方を作成します。非同期処理は、VTAM からの伝送データが到着したときに実行されます。アプリケーション・プログラムの応答によっても、非同期処理が行われません。

通信トラフィック全体について、4つの主要なバッファ・プールとフォーマット設定のためのアクセスのスペース (SPA) およびメッセージ・キュー・データ・セットが管理されます。プール・スペースまたは入出力に関する競合のために、回線アクティビティによって起動された通信タスクの処理が中断されたときに、待ち時間が記録されます。この情報は、以下の例の通信待機報告書に含まれます。

この報告書は、回線番号によって一連の通信処理タスクが識別されるため、通信要約報告書と似ています。

```

IMS MONITOR  ****COMMUNICATION IWAIT****      TRACE START 1993 130  5:55:15      TRACE STOP 1993 130  5:59:49  PAGE 0090
NODE OR      .....IWAIT TIME.....
LINE NUMBER  OCCURRENCES      TOTAL      MEAN      MAXIMUM      FUNCTION      BLKSIZE  MODULE      DIST.
NO.
ALL LINES...
PREFETCH I/O
_____
NONE

```

IMS 内部リソースの使用

リソースに関する内部競合のレベルを調べるときに使用できるいくつかの要約報告書があります。

以下に挙げるのは、それらの報告書についての簡単な説明です。

プール・スペース障害

プール・スペース障害要約報告書では、各領域において一定量のストレージを利用できなかった回数を示します。これは、ストレージが利用不能だったときの発生回数、バイト数、およびプール ID を示します。この要約を使用すれば、システム定義の変更または JCL 内の EXEC ステートメント内のバッファ数オーバーライドにより、バッファ・プール割り振りを増やす必要があるかどうかを判別できます。

この報告書の形式は、以下の例のようになります。

POOL SPACE FAILURE SUMMARY

POOL ID	BYTES REQ.	OCCURRENCES
DLMP	8888	1
DLDP	7777	1
TOTAL		2

デッドロックになっているプログラム

ある特定のデータベース・データ・セット内のセグメントの所有権に関して、1対のプログラムがデッドロック状態になるたびに、デッドロック・イベント要約 (DEADLOCK EVENT SUMMARY) 報告書にそのオカレンスが記録されます。この報告書内の各行には、関係している2つのPSBが表示され、どちらに処理の優先権が与えられ (REQ-ING PSB)、動的バックアウト後にどちらが再処理を行う必要があるか (LOSING PSB) が示されます。この報告書を、以下の例に示します。

DEADLOCK EVENT SUMMARY

REQ-ING PSB	LOSING PSB	DMBNAME	OCCURRENCES
PSBNAME1	TPPSBRE3	DBASEBAL	1
TOTAL			1

IMS ラッチ競合

IMS におけるタスク処理の基本的なシリアライゼーションは、IMS ラッチの所有権によって制御されます。各種のプログラムが実行中であるときには、それらは所有権を得るために競合します。それらがリソースを待つ場合

は、ラッチを所有しているプログラムは、リソースを待っている他方の ITASK に通知する必要があります。リソースに関する競合のレベルを判定するときは、ラッチ競合統計 (LATCH CONFLICT STATISTICS) 報告書を使用します。

ラッチ名は、シリアライズされている種々のタイプのリソースの省略語です。

省略語 ラッチ名

DISP SYS/DISPATCHER

DCSL DC/CHECKPOINT DC SYSTEM

LUML

DC/LU 6.2 LUM

CONV

DC/CONVERSATION CHECKPT

TERM

DC/TERMINAL

LUBT DC/LU62 LUB-TIB CHAIN

SCHD

TM/SCHEDULING

TCTB TM/TCT BLOCK

APSB TM/ALLOCATE PSB (BLK MVR)

PDRB

TM/PDIR BLOCK (BLK MVR)

PSBP TM/PSB POOL (BLK MVR)

DMBP

TM/DMB POOL (BLK MVR)

PSBB TM/PSB BLOCK (BLK MVR)

DMBB

TM/DMB BLOCK (BLK MVR)

PDRP TM/PDIR POOL (BLK MVR)

DBAU

TM/DBRC AUTH (BLK MVR)

DDRB

TM/DDIR BLOCK (BLK MVR)

DDRP

TM/DDIR POOL (BLK MVR)

DBBP DB/OSAM BUFFER POOL

DBLR

DB/DFSDBLR0 MODULE

SUBQ

TM/TM SUBQUEUEES

DBSL DB/DB CHECKPOINT
USER DC/USER
CCTL SYS/DBCTL RESOURCE
VTCB
 SYS/CBTS VTCB POOL
VLQB
 SYS/CBTS LQB POOL
CBTS SYS/CBTS POOLS (ALL)
BLKM
 TM/SMB QUEUE HASH TABLE
QMGR
 SYS/QUEUE MANAGER
QBSL SYS/QUEUE BUFFER
SMGT
 SYS/STORAGE MANAGEMENT
DBLK
 SYS/DEPENDENT REGION
XCNQ
 DB/EXCLUSIVE ENQ/DEQ
ACTL SYS/STATISTICS
LOGL
 SYS/LOGGER

モニターがアクティブの間にシステム・チェックポイントがとられた場合は、ラッチ競合統計はゼロにリセットされるため、この報告書に示された値は無効です。この状態のときは、以下のメッセージが報告書の先頭部分に挿入されます。

```

**** A CHECKPOINT OCCURRED DURING MONITOR RUN ****
**** LATCH CONFLICT STATISTICS ARE INVALID ****
**** SEE UTILITIES REFERENCE MANUAL ****
  
```

しかし、マスター端末オペレーターが、STATISTICS キーワード・パラメーターを指定して /CHECKPOINT コマンドを出す場合は、ラッチ競合統計はゼロにリセットされますが、このことは、IMS モニターには通知されません。したがって、統計が無効になっていることが DFSUTR20 には検出できないので、このメッセージが発行されることはありません。

推奨事項: モニターを実行中は、統計チェックポイントを出さないことをお勧めします。

ラッチ競合統計報告書には、競合のレベルを表す種々のタイプのラッチとカウンターが示されます。以下の例は、ラッチ競合統計報告書の例を示しています。項目は、ラッチ名にしたがって編成されています。

```

          IMS MONITOR  ** GENERAL REPORTS **          TRACE START 1993 209...

          LATCH CONFLICT STATISTICS
  
```

LATCH NAMES	COUNT FIELD	AT START	AT END	DIFF.
LOGL	CONTENTIONS	0	0	0
SMGT	CONTENTIONS	0	0	0
XCNQ	CONTENTIONS	0	0	0
ACTL	CONTENTIONS	0	0	0
CBTS	CONTENTIONS	0	0	0
DBLK	CONTENTIONS	0	0	0

IMS モニター出力の度数分布の使用方法

IMS モニター・データ・レコードから得られる報告書には、平均時間が示されている数多くの要約行が含まれています。単なる平均時間や最大時間ではなく、時間測定されるイベントの分布を知りたい場合には、ある範囲の間隔全体にわたる度数分布の形で、イベントを個別に記録するよう報告書印刷ユーティリティーに要求することができます。

トランザクション・キュー・ロードや伝送ブロック・サイズなどの一部の分布は、時間依存ではありません。

以下の各表には、主要な IMS モニター報告書と各報告書ごとにそれぞれ生成される度数分布のタイプが示してあります。各タイプは、報告書内の各セクションにどれだけ項目があるかに応じて、いくつかの分布に分かれます。度数分布のタイプごとに、適当な間隔または範囲でデータが累積されます。各タイプごとに使用される範囲の集合には、ID 列に示されているように ID が与えられます。

- 以下の表には、領域要約別にソートされた報告書分布が示してあります。

表 62. 領域要約別分布報告書

報告書名	ID	説明
スケジュールの終わりから最初の DL/I 呼び出しまで	D1	経過時間
実行経過時間	D2	待ちなし時間
DL/I 呼び出し	D3	N/A
	D4	N/A
外部サブシステム呼び出し	D5	経過時間
DL/I 呼び出し当たりの待ち	D6	待ちなし時間
意図のためのアイドル	D43	経過時間
チェックポイント	D7	N/A
	D8	N/A
	D20	経過時間
	D21	待ちなし時間

- 以下の表には、プログラム領域別にソートされた報告書分布が示してあります。

表 63. プログラム領域別報告書分布

報告書名	ID	説明
実行経過時間	D30	N/A
スケジュールの終わりから最初の DL/I 呼び出しまで	D31	N/A

- 以下の表には、プログラム要約別にソートされた報告書分布が示してあります。

表 64. プログラム要約別報告書分布

報告書名	ID	説明
スケジュール当たりのプロセッサ時間	D15	N/A
スケジュール当たりの、デキューされたトランザクション数	D14	N/A
スケジュール当たりの経過時間	D9	N/A
スケジュールの終わりから最初の DL/I 呼び出しまで	D10	N/A

- 以下の表には、通信要約別にソートされた報告書分布が示してあります。

表 65. 通信要約別報告書分布

報告書名	ID	説明
回線経過時間	D18	N/A
回線 NOT WAIT 時間	D19	N/A

- 以下の表には、回線機能別にソートされた報告書分布が示してあります。

表 66. 回線機能別報告書分布

報告書名	ID	説明
受信ブロックの長さ	D36	N/A
伝送ブロックの長さ	D37	N/A
非アクティブ間隔	D38	N/A

- 以下の表には、MSC キューイング要約別にソートされた報告書分布が示してあります。

表 67. MSC キューイング要約別報告書分布

報告書名	ID	説明
キューに入っていた時間	D39	N/A

- 以下の表には、トランザクション・キューイング別にソートされた報告書分布が示してあります。

表 68. トランザクション・キューイング別報告書分布

報告書名	ID	説明
スケジュール時にキューに入っているトランザクション数	D17	N/A
スケジュール当たりの、デキューされたトランザクション数	D16	N/A

表 68. トランザクション・キューイング別報告書分布 (続き)

報告書名	ID	説明
事前取り出し形式ブロック	D28	経過時間
	D29	待ちなし時間

- 以下の表には、呼び出し要約別にソートされた報告書分布が示してあります。

表 69. 呼び出し要約別報告書分布

報告書名	ID	説明
DL/I 呼び出し当たりの PSB 待ち数	D13	N/A
外部サブシステム呼び出し当たりの PSB 待ち数	D44	N/A
呼び出し当たりの PSB 経過時間	D11	N/A
呼び出し当たりの PSB 待ちなし時間	D12	N/A
PSB 外部サブシステム呼び出し	D45	経過時間

- 以下の表に、バッファ・プール統計から得られる、待ち時間に関するいくつかの分布を示します。

表 70. 待ち時間分布

機能	ID	モジュール・キー
ストレージ	D22	SMN
OSAM 入出力	D23	DBH
VSAM 入出力	D24	VBH
スケジューラー内部	D25	MSC
キュー・マネージャー入出力	D26	QMG
ブロック・ローダー入出力	D27	BLR
MFS ブロック入出力	D32	MFS
MFS ディレクトリー入出力	D33	MFS
HSAM 入出力	D34	DIE
形式バッファ・プール・スペース	D35	PMM
PI エンキュー	D40	ありません。
QMGR SNAPQ 検査	D42	ありません。

関連資料:

897 ページの『分布のデフォルト値の定義』

度数分布出力の入手方法

IMS モニター報告書印刷ユーティリティーに分布データを収集するよう要求するには、DIS 入力制御ステートメントを組み込む必要があります。こうすると、MEAN の表題の欄に項目を持つすべての報告書項目が、対応する度数分布を分布付録の一部として持つこととなります。

さらに、報告書の各行は、Distribution Number の表題の欄に識別参照番号を含んでいるので、その同じ番号が付けられている、付録の中の分布データを見つけることができます。

度数分布範囲の定義方法

要約行ごとに 10 個の間隔が定義され、各間隔の範囲に入るオカレンスが累積されます。間隔の範囲は、デフォルトの終点で事前設定されます。デフォルトの終点は、イベントに合うように選択されます。

例えば、DL/I 呼び出し経過時間についての終点は、0、1000、2000、4000、8000、16000、32000、64000、128000、256000、INF (時間はすべてミリ秒単位) です。最初の間隔の下限は常にデフォルトとしてゼロが使用され、10 番目の間隔の上限は無限大 (INF) です。

複数のタイプの分布で同じ終点セットを使用できますが、各タイプに 1 つの分布 ID が割り当てられます。この ID を使用して、終点を再定義できます。デフォルトの終点をオーバーライドするときは、入力制御ステートメントを報告書印刷ユーティリティに含めてください。そのステートメントで、分布 ID のタイプを指定し、希望する終点値を与えます。例えば、DL/I 呼び出し経過時間の終点は、次のように再指定できます。

```
D5 0,500,1000,1500,2000,4000,,,100000,500000
```

ここで指定されていない終点の値は、デフォルトの 32000 と 64000 のままであり、最後の値もそのまま (INF) です。

以下の例は、分布付録のサンプル・ページです。これを見れば、個々の分布に対してどのように番号が付けられているか、また分布のタイプによってどのように範囲が異なるかがわかります。行はペアになっており、2 番目の行に累積カウントが記録されます。

```
IMS MONITOR ****DISTRIBUTION APPENDIX**** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0200
# 1.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
# 2.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
# 3.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
# 4.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
# 5.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....6
# 6.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....8
# 7.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
# 8.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....9
# 9.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
# 10.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 1.....0.....0.....0.....0.....0.....0.....0.....0.....0.....7
# 11.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
# 12.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....8
# 13.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
# 14.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
# 15.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 1.....0.....0.....0.....0.....0.....0.....0.....0.....0.....8
# 16.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 0.....0.....0.....0.....0.....0.....0.....0.....0.....0.....1
```

分布のデフォルト値の定義

度数分布表および待ち時間分布表にある ID を使用すれば、次のリストで探すことによって、分布のデフォルト・エンドポイントを判別できます。

D1、D2、D5、D6、D9、D10、D11、D12、D15、D18、D19、D20、D21、D22、D25、D27、D28、D29、D30、D31、D43、および D45

0、1000、2000、4000、8000、16000、32000、64000、128000、256000、INF

D3 0、50000、100000、150000、200000、250000、300000、350000、400000、450000、INF

D4 0、200000、400000、600000、800000、1000000、1200000、1400000、1600000、1800000、INF

D7、D13、および D44

0、0、1、2、3、4、5、6、7、8、INF

D8 0、100000、200000、300000、400000、500000、600000、700000、800000、900000、INF

D14、D16、D17

0、1、2、3、4、5、10、15、30、90、INF

D23、D24、D26、D32、D40、D42

0、2000、8000、24000、50000、100000、150000、200000、250000、300000、INF

D33、D34、D35

0、2000、4000、8000、16000、32000、64000、96000、128000、160000、INF

D36、D37

0、10、20、40、80、100、200、400、800、1000、INF

D38 0、1000、10000、100000、200000、500000、800000、1000000、1500000、2000000、INF

D39 0、1000、5000、10000、50000、100000、500000、1000000、5000000、10000000、INF

関連概念:

893 ページの『IMS モニター出力の度数分布の使用方法』

分布付録の解釈

異常なイベントの組み合わせが報告書要約行で報告されたと思われるときには、分布付録の明細出力を使用できます。通常、リソース使用の問題を強調するには、平均時間と最大時間またはカウントのみで十分です。

しかし、平均値では異常な分布が覆い隠されていて分からないと思われる場合には、IMS モニター出力レコードに含まれている詳細を利用できます。

例えば、特定のトランザクションのスケジューリング・アルゴリズムでの変化を調べているときに、アプリケーション・プログラムのスケジューリングごとにどれだけのトランザクションが処理可能であったかを知る必要があるとします。以下の図

は、処理されたトランザクションをヒストグラムで示したものです。

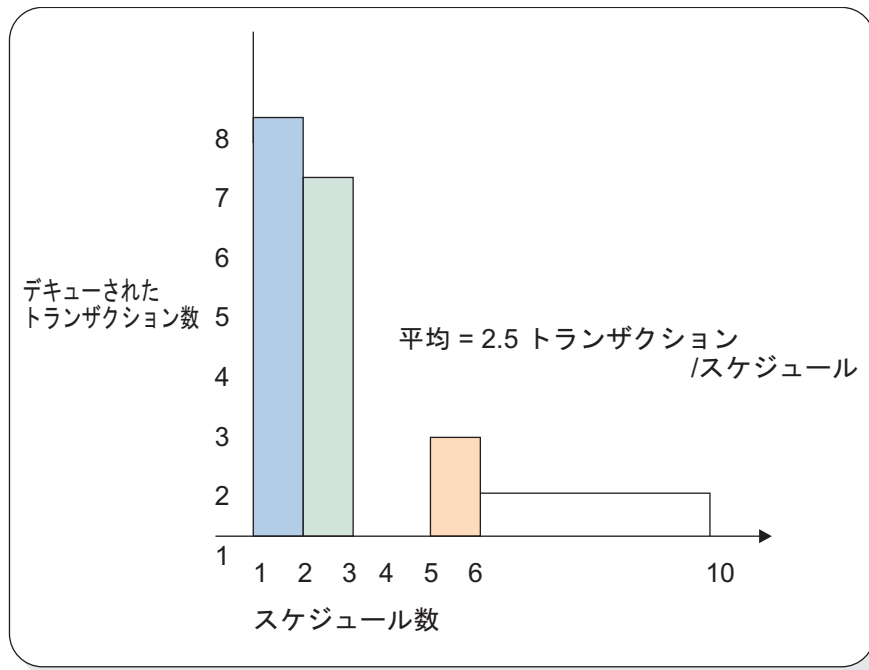


図 89. アプリケーション・プログラムのスケジューリングごとに処理されたトランザクションの数

平均 =2.5 トランザクション/スケジュールです。図の分布を見れば、多くのスケジュールで 1 つまたは 2 つのトランザクションしか処理できなかったこと、およびわずかなスケジュールでキューがほとんどいっぱいになったことが分かります。ヒストグラムの分布データは次のとおりです。

Number of schedules	1	2	3	4	5	6-10	>10
Transactions dequeued	8	7	0	0	2	1	0

分布付録では、このヒストグラムのデータが 2 行形式で表示されます。

- 最初の行には間隔が表示され、その接頭部には、それよりも前の出力の各行への相互参照が付いています。
- 2 番目の行には、これらの間隔内で発生したイベントの数が表示されます。

このデータは、次のように表示されます。

```
# 950B...0...1...2...3...4...5...10...15...30...90...INF
      8 7 0 0 2 1 0 0 0 0
```

相互参照の 950B は、固有の報告書行を指し示しています。例えば、トランザクション・キューイング報告書では、注目するトランザクションについての該当行で DISTRIBUTION NUMBER という見出し欄に 950A,B を表示しています。分布付録内でそのデータを見つけるためには、参照番号 950B を使用してください。参照番号 950A は、スケジュール時におけるキュー内のトランザクション数についてのデータを指し示します。

MSC の IMS モニター報告書

IMS モニター報告書印刷プログラムには、複数システム結合機能によって起こったメッセージ・イベントを特筆する 3 つの報告書が組み込まれます。

- MSC トラフィック報告書

この報告書には、モニター間隔中に様々なリンク・パスを使用するメッセージの、エンキューとデキューのカウン트가表示されます。

- MSC 要約報告書

この報告書には、以下の要約が示されます。

- 入力トランザクション名ごとのトラフィック・キュー
- 宛先名ごとのトラフィック・キュー
- リンク番号ごとのトラフィック・キュー
- 宛先システムごとのトラフィック・キュー

- MSC キューイング要約報告書

この報告書が生成されるのは、システム間メッセージが、宛先システムに送信される前にローカル・システム上でキューイングされる時のみです。ローカル・システムは、中間システムでなければなりません。この報告書は、次のものを表示します。

- メッセージがキューに入っていた最大時間
- メッセージがキューに入っていた平均時間
- キューの最大長
- キューの最大カウント
- このローカル・システムが関与しているすべてのリンクについて、キューに入れられたメッセージの合計数


3 つの報告書は、すべて分布付録内に項目を持つことができます。したがって、異常な伝送パターンの疑いがあるときには、トラフィックの度数分布を調べることができます。

関連概念:

926 ページの『IMS モニター報告書の概要』

942 ページの『回線アクティビティのモニター』

関連タスク:

 複数システムのモニターとチューニング (コミュニケーションおよび接続)

MSC トラフィック報告書

MSC トラフィック報告書は、モニター・システムがローカル・システムであるパートナー・システム間のすべてのトラフィックについての個々のキュー・ロードを明らかにします。この報告書は、そのローカル・システムの通信に定義されたすべての固有システム識別番号 (SID) をリストしています。

その次に、以下の変数の組み合わせごとにエンキューされたり、デキューされる全メッセージを要約します。

- 入力名 (メッセージ送信元である端末またはプログラム)
- 宛先名 (端末またはプログラム)
- 入力システム (SID)
- 宛先システム (SID)
- リンク番号
- リンク・タイプ (CTC、MTM、TCP/IP、または VTAM)

以下の例は、MSC トラフィック報告書を示しています。メッセージがローカル・システムから出される場合は、そのメッセージの存在はデキュー・カウントのみに含まれます。ローカル宛先を持つメッセージは、エンキュー・カウントのみに現れます。

```

IMS MONITOR ****MSC TRAFFIC REPORT****      TRACE START 1993 130  5:55:15  TRACE STOP 1993 130  5:59:49  PAGE 0151
LOCAL SID VALUES = 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115
INPUT  DESTIN.  INPUT  DEST.  LINK  LINK  ENQUEUE  DEQUEUE
NAME   NAME       SID    SID    NO.  TYPE  COUNT    COUNT
-----
DSWT3685 DSWT3685    2      2      2    C-C    0        1
          SC6Y      2    102    2    C-C    1        0
DSWT6161 DSWT6161    3      3      3    C-C    0        1
          SC6Z      3    103    3    C-C    1        0
DSWT3838 DSWT3838    2      2      2    C-C    0        1
          SC6Y      2    102    2    C-C    1        0
DSWT4618 DSWT4618    3      3      3    C-C    0        1
          SC6Z      3    103    3    C-C    1        0
DSWT3903 DSWT3903    3      3      3    C-C    0        1
          SC2Z      3    103    3    C-C    1        0
DSWT5418 DSWT5418    3      3      3    C-C    0        1
          SC4Z      3    103    3    C-C    1        0
DSWT4673 DSWT4673    3      3      3    C-C    0        1
          SC2Z      3    103    3    C-C    1        0
DSWT5141 DSWT5141    45     45     45    VTAM    0        1
          PS3X      45    145    45    VTAM    1        0
DSWT4391 DSWT4391    2      2      2    C-C    0        1
          SC4Y      2    102    2    C-C    1        0
DSWT3324 DSWT3324    17     17     17    VTAM    0        1
          IT1Y      17    117    17    VTAM    1        0
DSWT4781 DSWT4781    3      3      3    C-C    0        1
          SC4Z      3    103    3    C-C    1        0
DSWT3525 DSWT3525    3      3      3    C-C    0        1
          SC6Z      3    103    3    C-C    1        0
DSWT4542 DSWT4542    2      2      2    C-C    0        1
          SC6Y      2    102    2    C-C    1        0
DSWT5796 DSWT5796    3      3      3    C-C    0        1
          SC2Z      3    103    3    C-C    1        0
DSWT4782 DSWT4782    3      3      3    C-C    0        1
          SC6Z      3    103    3    C-C    1        0
DSWT4633 DSWT4633    2      2      2    C-C    0        1
          SC6Y      2    102    2    C-C    1        0
DSWT3655 DSWT3655    12     12     12    VTAM    0        1
          SC6U      12    112    12    VTAM    1        0
DSWT3892 DSWT3892    2      2      2    C-C    0        1
          SC6Y      2    102    2    C-C    1        0
DSWT3338 DSWT3338    4      4      4    VTAM    0        1
          SC2U      4    104    4    VTAM    1        0
DSWT4681 DSWT4681    3      3      3    C-C    0        1
          SC4Z      3    103    3    C-C    1        0
DSWT4482 DSWT4482    2      2      2    C-C    0        1
          SC6Y      2    102    2    C-C    1        0
DSWT4902 DSWT4902    3      3      3    C-C    0        1
          SC2Z      3    103    3    C-C    1        0
DSWT4558 DSWT4558    3      3      3    C-C    0        1
DSWT4558 SC6Z      3    103    3    C-C    1        0
DSWT3925 DSWT3925    2      2      2    C-C    0        1
TOTAL TRAFFIC
-----
1353      1359

```

関連タスク:

➡ 複数システムのモニターとチューニング (コミュニケーションおよびコネクション)

MSC 要約報告書

MSC 要約報告書は、ローカル・システムが処理するが、複数システム結合機能 (MSC) トラフィックの一部であるメッセージについてのエンキューおよびデキュー・アクティビティを示します。

以下の例に、この報告書の形式を示しています。

キューイング・カウントの最初のセットは、モニター間隔中に入力された各タイプのトランザクション数と、その後デキューされたものの数を表します。

カウントの 2 番目のセットは、宛先名ごとの合計トラフィックを要約します。1 次トランザクションと応答は、リソース名で区別できます。また、エンキュー・カウントとデキュー・カウントの差を使用してリンク伝送の相対サービスを調べることができます。

カウントの 3 番目のセットは、リンク番号別にアクティブ・リンクをリストしているので、エンキュー・カウントとデキュー・カウントの差によって、そのリンクで増大があったかどうかを判別することができます。

カウントの 4 番目のセットは、すべてのリンク・パスで他のシステムに向かうトラフィックを記録します。1 つ以上のシステムのリンク優先順位の全体的パターンが、システム間トラフィックの増大を引き起こしているかどうかは、エンキュー・カウントとデキュー・カウントの差で判定できます。

```


IMS MONITOR *****MSC SUMMARIES***** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0178
<--SUMMARY BY INPUT NAME--> <--SUMMARY BY DESTINATION NAME--> <--SUMMARY BY LOGICAL LINK--> <--SUMMARY BY DEST. SYS. ID-->

```

INPUT NAME	ENQUEUE COUNT	DEQUEUE COUNT	DESTIN. NAME	ENQUEUE COUNT	DEQUEUE COUNT	LINK NO.	ENQUEUE COUNT	DEQUEUE COUNT	DEST. SID	ENQUEUE COUNT	DEQUEUE COUNT
DSWT3577	1	1	DSWT4358	0	1						
DSWT4048	1	1	DSWT5988	0	1						
DSWT5216	1	1	DSWT5457	0	1						
DSWT4776	1	1	DSWT5187	0	1						
DSWT5496	1	1	DSWT3312	0	2						
DSWT5277	1	1	DSWT5604	0	1						
DSWT5711	1	1	DSWT3347	0	1						
DSWT5274	1	1	DSWT5338	0	1						
DSWT5807	1	1	DSWT3268	0	1						
DSWT3685	1	1	DSWT3676	0	1						
DSWT6161	1	1	DSWT5428	0	1						
DSWT3838	1	1	DSWT5395	0	1						
DSWT4618	1	1	DSWT4168	0	1						
DSWT3903	1	1	DSWT5061	0	1						
DSWT5418	1	1	DSWT3511	0	1						
DSWT4673	1	1	DSWT3363	0	1						
DSWT5141	1	1	DSWT3674	0	1						
DSWT4391	1	1	DSWT4467	0	1						
DSWT3324	1	1	DSWT4501	0	1						
DSWT4781	1	1	DSWT5037	0	1						
DSWT3525	1	1	DSWT4298	0	1						
DSWT4542	1	1	DSWT5778	0	1						
DSWT5796	1	1	DSWT4003	0	1						
DSWT4782	1	1	DSWT3988	0	1						
DSWT4633	1	1	DSWT4217	0	1						
DSWT3655	1	1	DSWT6135	0	1						
DSWT3892	1	1	DSWT5147	0	1						
DSWT3338	1	1	DSWT5381	0	1						

DSWT4681	1	1	DSWT5593	0	1
DSWT4482	1	1	DSWT3304	0	1
DSWT4902	1	1	DSWT5081	0	1
DSWT4558	1	1	DSWT4671	0	1
			DSWT3655	0	1
			DSWT3892	0	1
			DSWT3338	0	1
			DSWT4681	0	1
			DSWT4482	0	1
			DSWT4902	0	1
			DSWT4558	0	1
			DSWT3925	0	1

関連タスク:

 複数システムのモニターとチューニング (コミュニケーションおよびコネクション)

MSC キューイング要約報告書

MSC キューイング要約報告書では、システム間メッセージ・トラフィックだけの情報が得られます。IMS モニター間隔中に記録されたトラフィックのサンプルを使用すれば、メッセージがアクティブ・リンクで送信されるのを待ってキューに入れられていた最大時間および平均時間を調べることができます。


1 次メッセージが中間システムで過度に遅れる原因がリンク優先順位にあるかどうか、または応答の増大があるかどうかを検出できます。この報告書は、中間システムであるシステムの論理リンク・パスを示します。それぞれの着信リンク番号は、指定された発信リンク番号での伝送の前にキューイングされるメッセージの数を示します。中間システム・キュー上での最大時間および平均時間だけでなく、最大キュー・カウントが示されます。

以下の例は、MSC キューイング要約報告書を示しています。

IMS MONITOR *****MSC QUEUING SUMMARY***** TRACE START 2011 063, 10:33:13 TRACE STOP 2011 063, 10:34:54
PAGE 0022

ENQUE.....	DEQUE.....			MAX.Q	MAX.	MEAN	
LINK NO.	TYPE	LINK NO.	TYPE	MESSAGES	LENGTH	Q TIME	Q TIME
13	VTAM	6	M-M	2	1	2342	1234
TOTALS...				2			1234

関連タスク:

 複数システムのモニターとチューニング (コミュニケーションおよびコネクション)

第 48 章 DBCTL の IMS モニター報告書

DBCTL モニターでは、CCTL トランザクションが DBCTL のデータベースにアクセスするときに行われる処理に関する情報が提供されます。CCTL は、データベース・リソース・アダプター (DRA) 要求を使用してこのアクセスを行います。

このトピックでは、以下について説明します。

- IMS モニターが収集するイベント
- IMS モニター報告書印刷プログラムが作成する報告書の内容

DBCTL と DB/DC では、モニターの意味が異なります。DB/DC では、エンド・ユーザーは端末からトランザクションを入力します。トランザクションは IMS によって処理され、結果がユーザーに戻されます。モニターされるトランザクションの特性には、合計応答時間およびリソース競合 (例えば、PSB スケジュール待ち時間およびデータベース入出力など) のオカレンスがあります。

DBCTL には、トランザクションも端末エンド・ユーザーもありません。ただし、これは、CCTL 端末ユーザーが入力するトランザクションのための作業を行います。

これらの DRA 要求の一般的な順序は次のとおりです。

1. DBCTL でスケジュールされた PSB を獲得するための SCHED 要求
2. データベース呼び出しを行う DL/I 要求
3. 更新をコミットし、PSB を解放するための同期点要求 (COMMTERM)

これらの要求を包含する DBCTL 処理は、リカバリー単位 (UOR) と呼ばれています。

DBCTL では、UOR に関するモニター・データ、例えば、UOR が存在した合計時間、PSB スケジュールの待ち時間、およびデータベース呼び出し時の入出力数を提供します。これらの情報は、IMS トランザクション・モニター・データと非常に似ています。しかし、DBCTL/CCTL システムでは、UOR データが表すのは CCTL トランザクションの処理全体の一部にすぎません。したがって、CCTL トランザクション・パフォーマンスの全体像を知るためには、CCTL モニター・データが必要です。

DBCTL では、IMS モニター報告書の形式または使用法は変更されません。DBCTL にとっては不要な報告書や報告書内のフィールドがあります。一般に、そうしたものはトランザクション・マネージャーおよび通信域にあります。フィールドには、DBCTL 環境では異なった解釈になるものがあります。

DBCTL に不要な報告書については、データなしで見出しのみが表示されるか、あるいは報告書が作成されません。それらの報告書は次のとおりです。

- メッセージ・キュー・プール報告書
- メッセージ形式バッファ・プール報告書
- 通信要約報告書

- 通信 IWAIT 報告書
- 回線機能報告書
- MSC トラフィック報告書
- MSC 要約報告書
- MSC キューイング要約報告書

IMS モニター報告書の中の *region* (領域) という用語は、特定の IMS トランザクションを処理する、特定の従属領域に割り当てられた PST を指します。DBCTL モニター報告書でも、*region* という用語は PST のことです。PST は、一度に 1 つの CCTL スレッド (トランザクション) を処理することができます。しかし、CCTL スレッドが変更されると、1 つの PST が多数の異なる CCTL トランザクションを処理することになります。複数の CCTL を DBCTL に接続することができるので、実際には、PST は様々な CCTL からのトランザクションを処理できます。

1 つの CCTL 用に作成されたすべてのスレッドは、CCTL のジョブ名を持っています。これは、一般報告書の多数の領域で同じジョブ名として表されます。

トレース間隔中は、1 つのスレッドを複数の CCTL に割り当てることができますが、一時点では 1 つの CCTL にしか割り当てられません。したがって、DBCTL に接続される CCTL の個数に応じて、領域要約報告書には以下のものが表示されません。

- 1 つのジョブ名のみを持つ 1 領域
- いろいろなジョブ名を持つ 1 領域
- いろいろなジョブ名を持つ複数の領域。同じジョブ名を持つ領域もあれば、いろいろなジョブ名を持つ領域もあります。
- 1 つのジョブ名のみを持つ複数の領域

領域に関するモニター報告書は、トレース間隔中にスレッドで処理されたすべての CCTL の要約です (例、トレース間隔中にスレッドが割り当てられたすべての CCTL の経過時間)。

BMP と非メッセージ BMP について IMS モニターが生成する報告書は同じです。

UOR の経過時間は DRA 内ではなく、DBCTL 内で費やされます。DRA 内で費やされる時間は、CCTL の一部と見なされるため、DRA 時間は DBCTL 統計では報告されません。

IMS モニター・トレース・イベントの間隔

IMS モニター・トレースの間隔は、マスター端末のオペレーターが開始コマンドの入力と停止コマンドの入力の際に /TRACE コマンドを使用して定義します。オンライン IMS イベントは、IMSMON データ・セットに入れられる IMS モニター・レコードに記録されます。

イベント・タイミングは、従属領域アクティビティーに関連付けられます。以下の図に、時間測定が行われるイベントの間隔の境界を示します。

モニター・トレース間隔には、以下の間隔が含まれます。

- スケジューリングと終了
 - ブロック・ローダー使用中
 - 意図障害 (排他的意図およびデータ共用) およびスケジュール障害 (PSB 使用中およびスペース障害)
 - スケジューリングと終了の経過
 - 待ちなし
 - ACBLIB 待ち
 - DB フラッシュ待ち
 - DB CLOSE 待ち
- 領域占有 (「スケジューリングと終了の経過」全体とオーバーラップします)
 - スケジュールから最初の呼び出しまで
 - 実行の経過

領域の待ちなし時間は、待ち時間には含まれない経過時間のことです。ページング、またはプロセッサが優先順位の高いタスクにディスパッチされるために発生する遅延は、待ちなし時間の増加の原因となります。

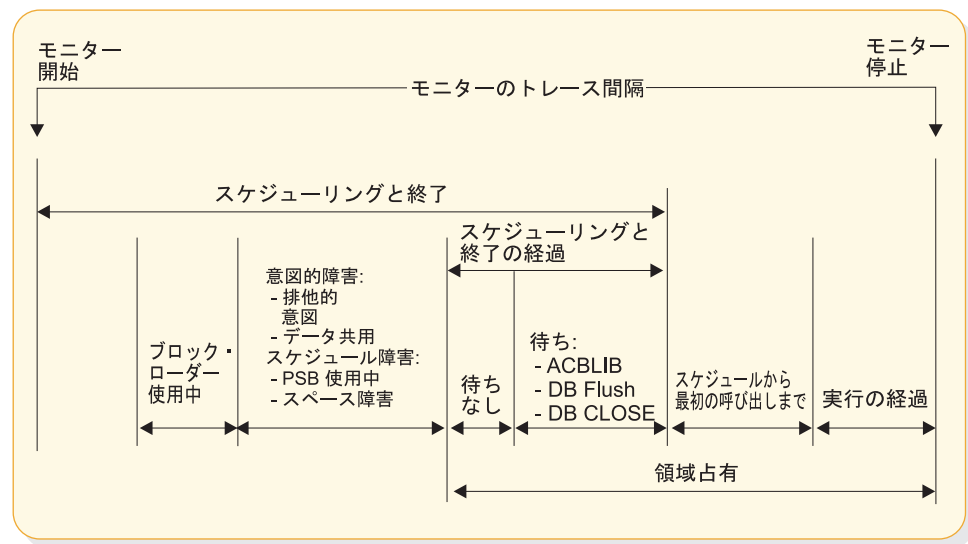


図 90. IMS モニター・トレース・イベントの間隔

IMS モニター報告書の概要

IMS モニターが収集するデータから得られる報告書のリストと、その報告書に含まれている主要なパフォーマンス・データを以下の表に示します。

関連資料: 「DB」のマークが付いている報告書については、839 ページの『第 46 章 DB モニター報告書』を参照してください。

表にリストした報告書の順序は、IMS モニター報告書印刷プログラムからの出力の順序と一致しています。モニター・スナップショットの期間内には、個々の報告書

に必要な特定のイベントが含まれていないことがあります。その場合には、報告書の見出しまたはデータの一部のみが作成されます。

表 71. IMS モニター報告書の出力順序および情報

報告書名	主な内容
システム構成	モニター実行の文書
データベース・バッファ・プール (DB)	トランザクションごとの DB 呼び出しと入出力のカウンタ
VSAM バッファ・プール (DB)	挿入と入出力のカウンタ
ラッチ競合統計	IMS 内部処理
領域およびジョブ名	モニター実行の文書
領域要約	経過時間と DL/I 呼び出しのカウンタ
領域待機	待ち時間
領域別プログラム	領域使用の経過時間
プログラム要約	プログラム全体の統計
プログラム入出力 (DB)	待ち時間/PCB
報告書	スペース障害とデッドロックのカウンタ
実行プロファイル	モニター実行の文書
呼び出し要約 (DB)	呼び出しカウンタと時間測定値/セグメント・タイプ
分布付録	イベントの度数分布

IMS モニター報告書のデータ項目のほとんどは、経過時間です。通常、それらはマイクロ秒単位で表されます。1876534 という項目は、1.876534 秒または 1876 ミリ秒のことです。この規則に従っていない時間については、報告書の中で計測単位が表示されています。

さらに、OCCURRENCES の見出しの下に、イベントの数と、バイト数を表す数字がいくつか示されます。

モニター実行の文書化

トレース間隔ごとに、行われた処理に関していくつかの一般報告書または要約報告書が作成されます。これらの報告書は、ご使用の IMS モニター実行の文書の一部として使用できます。

トレースが実行されたときの条件は、できるだけ正確に記録することが重要です。文書にはトレースの前後で何回か得られた (/DISPLAY コマンドによる) システム状況情報、予期されている CCTL トランザクション・アクティビティのプロファイル、および必要とされている任意の処理イベントを含めることができます。トレース間隔は、代表的な処理ロードを表すべきであり、偏った、あるいは不適当なヒストリー・レコードであってはなりません。

システム構成報告書のデータへの追加

最初の一般報告書 (SYSTEM CONFIGURATION という表題が付けられたもの) は、BUFFER POOL STATISTICS というページ見出しのもとにあります。これは、

IMS および z/OS システムの修正レベルを示しています。適用された IMS APAR のリストを追加し、アプリケーション・プログラムのサービス・レベルを含めることができます。このようにするのは、特にアプリケーション・プログラムが永続プログラムでなかったり、段階的に行われる実装の一部である場合です。システム構成の出力は、以下のセクションに例示してあります。

モニターのトレース間隔の記録

ほとんどの IMS モニター報告書の見出しには、トレースの開始時刻と停止時刻が示されています。その形式は、年 日 (ユリウス暦) HH:MM:SS です。トレース間隔の全長は、MONITOR OVERHEAD DATA という表題の下にミリ秒単位で示されています。以下の例には、IMSMON データ・セットに入れられたトレース・レコードの数が示されています。モニターのトレース間隔の記録例は、以下の図に示してあります。

```
***IMS MONITOR*** BUFFER POOL STATISTICS TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0001
```

SYSTEM CONFIGURATION

```
SYSTEM CONFIGURATION :
IMS VERSION           : 4
RELEASE LEVEL         :
MODIFICATION NUMBER  :
```

モニター実行プロファイルの完了

処理率が、実行プロファイル報告書の最後の部分にまとめて示されています。この統計は、モニター間隔に関して、UOR スループット、および DL/I と入出力アクティビティの度合いを要約したものです。以下の例は、この報告書の例です。

```
IMS MONITOR **RUN PROFILE** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0184
```

```
TRACE ELAPSED TIME IN SECONDS.....274.6
TOTAL NUMBER OF MESSAGES DEQUEUED.....1403
TOTAL NUMBER OF SCHEDULES.....173
NUMBER OF TRANSACTIONS PER SECOND.....5.1
TOTAL NUMBER OF DL/I CALLS ISSUED.....18632
NUMBER OF DL/I CALLS PER TRANSACTION.....13.2
NUMBER OF OSAM BUFFER POOL I/O'S.....11236, 8.0 PER TRANSACTION
NUMBER OF MESSAGE QUEUE POOL I/O'S.....0, 0.0 PER TRANSACTION
NUMBER OF FORMAT BUFFER POOL I/O'S.....0, 0.0 PER TRANSACTION
RATIO OF PROGRAM ELAPSED TO DL/I ELAPSED:
REGION 1: 1.09
REGION 2: 1.09
REGION 3: 1.00
REGION 4: 1.02
REGION 5: 1.01
REGION 6: 1.00
REGION 7: 1.00
REGION 8: 1.00
REGION 9: 1.17
REGION 10: 1.00
REGION 11: 1.00
REGION 49: 1.03
REGION 50: 1.19
RATIO OF DL/I ELAPSED TO DL/I IWAIT:
REGION 1: 325.65
REGION 2: 73.49
REGION 4: 100.35
REGION 5: 85.76
REGION 6: 82.99
REGION 47: 95.64
REGION 48: 45.93
REGION 49: 9.22
```

実行プロファイル報告書の下半分に、以下のいくつかの率が表示されます。

- 各領域ごとの、DL/I 経過時間に対するプログラム経過時間の率
- DL/I 処理時の、待ち時間に対する DL/I 経過時間の率
- 他のサブシステム呼び出し経過時間に対するプログラム経過時間の率
- 他のサブシステム呼び出し経過時間に対する DL/I 経過時間の率

各領域は、領域 1 から始まるシーケンス番号で識別されます。

一部の汎用プロセス率は、いくつかのバッファ・プール統計報告書の最後の部分に示されます。それらの率を、トレース間隔の文書化プロファイルに含めることができます。これらは、1 つの UOR またはシステム・リソースに固有のものではなく、一連のモニター実行全体のバリエーションの標識として使用できます。次の率が示されます。

- (OSAM 読み取り数 + OSAM 書き込み数 + すべての待機) の合計数を、トランザクションの合計数で除算したもの。

この率は、メッセージ・キュー・プール報告書からのもので、トランザクション 1 つ当たりの、スケジューリング機能の処理に必要な物理入出力アクティビティを示します。

- (OSAM 読み取り数 + OSAM 書き込み数 + BISAM 読み取り数) の合計数を、トランザクションの合計数で除算したもの。

この率は、データベース・バッファ・プール報告書からのもので、トランザクション 1 つ当たりの、データベース・バッファリング機能の処理に必要な物理入出力アクティビティを示します。

IMS モニター報告書オカレンスの検査

IMS モニター報告書印刷プログラムの出力を調べるとき、報告書の見出しがあっても、それは必ずしも該当するデータがリストされていることを意味しません。システム定義オプションおよびユーティリティー制御ステートメントにより、出力の内容が次のように影響を受けます。

- 制御ステートメントで DLI を指定しないと、呼び出し要約報告書は出力されません。
- 出力には、制御ステートメントで DIS または DISTRIBUTION を指定していない限り、1 組の分布報告書は含まれません。多数の報告書に現れる DISTRIBUTION NUMBER という表題の欄には、分布報告書に含まれている項目の相互参照が含まれています。
- 制御ステートメントで ONLY DLI を指定すると、呼び出し要約報告書だけしか出力されません。

多数の要約報告書では、開始値と終了値の差を計算するためにシステム状況が必要であり、しかもこの状況は /TRACE SET OFF 処理の間に得られるため、IMS モニターの実行を IMS 制御領域の終了前に終わらせる必要があります。トレースが正しく停止しなかった場合は、次のメッセージが出されます。

```
NO DATABASE BUFFER POOL TRACES AT END TIME ON MONITOR LOG TAPE
****DATABASE BUFFER POOL REPORT CANCELLED****
```

同様に、他の要約報告書も作成されません。

Buffer Pool Statistics という表題の付いた一連の報告書は、VSAM Buffer Pool セクションを含みません。ただし、IMS.ACBLIB 内のデータベースが VSAM アクセス方式を使用している場合は除きます。VSAM が使用されていなければ、次のメッセージが出されます。

```
NO VSAM BUFFER POOL TRACES ON MONITOR LOG TAPE
****VSAM BUFFER POOL REPORT CANCELLED****
```

IMS モニターが、特定の IMS モニター報告書またはその報告書の 1 セクションを作成するために使用するソース・データをトレース間隔中に記録していなかった場合は、報告書には見出しのみが表示されます。

従属領域内のアクティビティーのモニター

IMS モニターは、トレース間隔中にアクティブである、/TRACE コマンドで識別されたすべての従属領域 (CCTL スレッド) について、時間測定情報を収集します。各イベントごとの経過時間の合計、検出された最大の単独の時間、および平均時間をこのプログラムは記録します。

時間測定値を表示する主要な報告書は 3 つあります。それらの報告書とその内容のリストは次のとおりです。

- 領域要約報告書 (**REGION SUMMARY REPORT**)
 - スケジューリングと終了
 - スケジュールの終わりから最初の呼び出しまで
 - 以下について表示される、個別要約付きの実行経過
 - DL/I 呼び出し
 - 外部サブシステム・サービスとコマンド呼び出し
 - 外部サブシステムのデータベース・アクセス呼び出し
 - チェックポイント処理
 - 領域占有
- 領域待機 (**REGION WAIT**)
 - スケジューリングと終了時の待機
 - DL/I 呼び出し時の待機
 - 外部サブシステム呼び出し時の待機
 - チェックポイント時の待機
- 領域別プログラム (**PROGRAMS BY REGION**)
 - 実行の経過
 - スケジュールの終わりから最初の呼び出しまで

この報告書内に表示される「プログラム名」は、UOR の PSB 名です。

これら 3 つの報告書は、以下の例に示してあります。

従属領域でのアクティビティーは、次の 5 つの時間測定情報カテゴリーのいずれかに入ります。

- スケジューリングと終了の経過時間

スケジューリング処理には、アクティブ IMS.ACBLIBA/B データ・セットからのブロック・ロード、および PSB の所有権の獲得などの多数の準備イベントが含まれます。終了に必要な時間は、UOR の終了要求の受信後に DBCTL がこの処理を完了するためにかかる時間です。

- スケジュールの終わりから最初の呼び出しまでの経過時間

これは、DBCTL がスケジューリングを完了してから、DBCTL が最初の DL/I 呼び出しを調べるまでの時間です。この時間中に起こるイベントは、すべて DBCTL の範囲外で、データベース・リソース・アダプター (DRA) 内または CCTL 内のものです。

- プログラム経過時間 (すべての呼び出しを含む)

この時間には、主要な UOR 処理が含まれ、最初の DL/I 呼び出しから UOR を終了させる呼び出しまでが測定されます。

- DL/I 呼び出しの実行に要した経過時間

この時間は、すべての DL/I 呼び出しの時間を含みます。DBCTL での時間は記録され、合計されます。

以下の例は、領域要約報告書を示しています。

```

IMS MONITOR ****REGION SUMMARY****      TRACE START 1993 130 5:55:15      TRACE STOP 1993 130 5:59:49 PAGE 0011
              (A)                                     (B)
              .....ELAPSED TIME.....             NOT IWAIT TIME(ELAPSED-IWAIT)
              TOTAL      MEAN      MAXIMUM          TOTAL      MEAN      MAXIMUM
              -----      -      -              -----      -      -
SCHEDULING AND TERMINATION
**REGION 5 5 4146 829 948 4146 829 948 287A,B
**REGION 6 7 6028 861 1067 6028 861 1067 214A,B
**REGION 8 8 6847 855 1098 6847 855 1098 129A,B
**REGION 10 7 9664 1380 3668 9664 1380 3668 272A,B
**REGION 47 6 5482 913 1021 5482 913 1021 145A,B
**REGION 49 3 2612 870 917 2612 870 917 443A,B
**TOTALS 123 126042 1024 126042 1024
SCHEDULE TO FIRST CALL
**REGION 1 1 15479797 15479797 15479797 555
**REGION 2 1 22376350 22376350 22376350 564
**REGION 3 1 15169488 15169488 15169488 578
**REGION 4 1 48146258 48146258 48146258 584
**REGION 48 1 795351 795351 795351 592
**REGION 49 4 2960425 740106 2951746 442
**REGION 50 1 15713464 15713464 15713464 575
**TOTALS 168 514286738 3061230
ELAPSED EXECUTION
**REGION 1 1 290146255 290146255 290146255 1
**REGION 2 1 252290108 252290108 252290108 2
**REGION 3 1 259496970 259496970 259496970 3
**REGION 4 1 322812716 322812716 322812716 4
**REGION 48 1 273871107 273871107 273871107 48
**REGION 49 4 271703421 67925855 155176058 49
**REGION 50 1 290379922 290379922 290379922 50
**TOTALS 173 14238540145 82303700
DL/I CALLS
**REGION 1 60 264626241 4410437 88981490 263813671 4396894 88970053 0.76 247A,B,C
**REGION 2 223 230505269 1033655 61048758 227368742 1019590 61011153 0.73 237A,B,C
**REGION 3 29 257704383 8886358 69000514 257704383 8886358 69000514 0.00 98A,B,C
**REGION 4 792 313735347 396130 52439653 310609035 392183 52439653 0.22 180A,B,C
**REGION 49 592 262886317 444064 30202068 234394017 395935 30159782 2.46 177A,B,C
**REGION 50 36 242591451 6738651 48651260 242591451 6738651 48651260 0.00 289A,B,C
**TOTALS 18632 12386905286 664818 12024562411 645371 0.97
IDLE FOR INTENT
-----
CHECKPOINT
REGION OCCUPANCY
**REGION 1 100.0%
**REGION 2 100.0%
**REGION 3 100.0%

```



```

**REGION 4 100.0%
**REGION 48 100.0%
**REGION 49 100.0%
**REGION 50 100.0%

```

以下の例は、領域待機報告書を示しています。

```

IMS MONITOR ****REGION IWAIT**** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0023
.....IWAIT TIME..... DISTRIBUTION
**REGION 5 OCCURRENCES TOTAL MEAN MAXIMUM FUNCTION MODULE NUMBER
SCHEDULING + TERMINATION
SUB-TOTAL
TOTAL
DL/I CALLS

```

**REGION	OCCURRENCES	TOTAL	MEAN	MAXIMUM	FUNCTION	MODULE	DISTRIBUTION NUMBER
11		181816	16528	24375	DD=IMMSTR2A	DBH	117
1		1667	1667	1667	PSB=BMPFPE06	BLR-64BIT	11
1		1635	1635	1635	INT=BMPFPE06	BLR-64BIT	12
8		112831	14103	17846	DD=IMMSTR1A	DBH	118
5		85460	17092	33717	DD=IMMSTR3A	DBH	119
5		58420	11684	14643	DD=IMINDEXA	VBH	120
12		173866	14488	22152	DD=PRODCNTA	VBH	121
3		100576	33525	68373	DD=IMMSTR2B	DBH	428
1		17921	17921	17921	DD=IMMSTR3B	DBH	429
1		17195	17195	17195	DD=IMMSTR1B	DBH	430
1		13577	13577	13577	DD=IMINDEXB	VBH	431
3		49928	16642	20396	DD=PRODCNTB	VBH	432
4		10973	2743	2787	DD=ITEMACTB	DBH	453
2		37680	18840	27664	DD=IAINDEXB	VBH	454
49		1500067	30613	138284	DD=INVENTRA	DBH	472
23		345595	15025	27613	DD=VENDORDA	VBH	473
1		342952	342952	342952	PI=VENDORDA...1		498
1		14612	14612	14612	PI=VNSINDXA...1		499
6		69203	11533	19492	DD=VNSINDXA	VBH	500
TOTAL		3132672	23034				

以下の例は、領域別プログラム報告書を示しています。

```

IMS MONITOR ****PROGRAMS BY REGION**** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0069
(A) (B)
OCCURRENCES ELAPSED EXECUTION TIME SCHEDULING END TO FIRST CALL DISTRIBUTION
TOTAL MEAN MAXIMUM TOTAL MEAN MAXIMUM NUMBER

```

**REGION	OCCURRENCES	ELAPSED TOTAL	EXECUTION MEAN	TIME MAXIMUM	SCHEDULING TOTAL	END TO MEAN	FIRST CALL MAXIMUM	DISTRIBUTION NUMBER
1	1							
1	1	290146255	290146255	290146255	15479797	15479797	15479797	885A,B
PROGSC6D	1	290146255	290146255		15479797	15479797		
REGION TOTALS	1	290146255	290146255		15479797	15479797		
2	2							
2	1	252290108	252290108	252290108	22376350	22376350	22376350	889A,B
PROGIT8C	1	252290108	252290108		22376350	22376350		
REGION TOTALS	1	252290108	252290108		22376350	22376350		
3	3							
3	1	259496970	259496970	259496970	15169488	15169488	15169488	893A,B
PROGTS1C	1	259496970	259496970		15169488	15169488		
REGION TOTALS	1	259496970	259496970		15169488	15169488		
4	4							
4	1	322812716	322812716	322812716	48146258	48146258	48146258	897A,B
PROGSP3D	1	322812716	322812716		48146258	48146258		
REGION TOTALS	1	322812716	322812716		48146258	48146258		
5	5							
5	2	62893103	31446551	40693590	5435	2717	2862	901A,B
PROGSP3A	1	61794787	61794787	61794787	2790	2790	2790	1271A,B
PROGTS1B	1	18294458	18294458	18294458	3104	3104	3104	1350A,B
PROGSP3B	1	36095342	36095342	36095342	2731	2731	2731	1363A,B
PROGIT2B	1	93902771	93902771	93902771	1667791	1667791	1667791	1401A,B
PROGSC2A	1	272980461	45496743		1681851	280308		
REGION TOTALS	6	272980461	45496743		1681851	280308		
6	6							
6	2	39000315	19500157	23703429	5286	2643	2801	905A,B
PROGIT1B	1	34293636	34293636	34293636	3136	3136	3136	1207A,B
PROGTS1B	1	51887767	51887767	51887767	2534	2534	2534	1278A,B
PROGSP3A	1	67375031	33687515	40291430	17210570	8605285	17213287	1328A,B
PROGSP3B	2							

PROGIT8A	1	69132416	69132416	69132416	3291	3291	3291	1359A,B
PROGSC4A	1	30165017	30165017	30165017	2571	2571	2571	1433A,B
REGION TOTALS	8	291854182	36481772		17193752	2149219		
**REGION	7							
PROGSC2B	1	269618583	269618583	269618583	5047875	5047875	5047875	909A,B
REGION TOTALS	1	269618583	269618583		5047875	5047875		
**REGION	8							
PROGIT8A	1	5181039	5181039	5181039	2928	2928	2928	913A,B
PROGPS3A	1	27304257	27304257	27304257	3350	3350	3350	1132A,B
PROGSC4B	1	37286872	37286872	37286872	3009	3009	3009	1255A,B
PROGIT2A	1	36902995	36902995	36902995	2850	2850	2850	1298A,B
PROGIT1B	1	30407479	30407479	30407479	2565	2565	2565	1336A,B
PROGIT1A	3	109875360	36625120	45190114	4279008	1426336	4272096	1357A,B
PROGIT8B	1	23405220	23405220	23405220	2679	2679	2679	1395A,B
REGION TOTALS	9	270363222	30040358		4296389	477376		

データベース処理意図の競合の検出

IMS モニターは、既にスケジュールされている別のアプリケーション・プログラムによって排他所有されているデータベースが更新されるのを待機していて、領域がアイドル状態になっている時間間隔を記録します。

DL/I 呼び出しの後に、IDLE FOR INTENT に合計アイドル時間、最大アイドル時間、および平均アイドル時間が表示されます。その領域のプログラムのスケジュールリングが失敗したときの経過時間は、その領域の要約行の時間に含まれます。

そのデータベースの所有権が解放されている場合であっても、領域のスケジュールが失敗することがあります。意図障害によって処理が保留された回数は、

「INTENT FAILURE SUMMARY」という表題の下に別個に記録されます。この報告書を、876 ページの『データベース処理意図の競合の検出』に示します。この報告書は、セグメント・タイプおよび問題のデータベース名についての排他意図のためにどの PSB が競合状態にあるかを示します。

チェックポイントの影響の検査

領域要約報告書の領域 0 の要約の最後にあるチェックポイント行に、以下のものが表示されます。

- ・ モニター間隔中にシステム・チェックポイントがとられる回数
- ・ 経過時間
- ・ 待ちなし時間

チェックポイント処理は、システム・ログに入れられるレコード数によって決まる指定の頻度で、制御プログラムによって開始することができます。他のチェックポイントは、オペレーター・コマンドによって取ることができます。

チェックポイント処理時の待ち時間は、領域待機報告書の最初の領域要約の終わりに示されます。DD 名とモジュール・コードの組み合わせごとに遅延を検出することができます。この部分の代表的な項目は、メッセージ・キュー・データ・セットおよび再始動データ・セットに関するものです。ストレージ待ちが原因である場合には、FUNCTION 欄の下の項目は、STG.= とその後に続くプール ID になります。

領域占有の測定

領域占有には、モニター間隔の合計時間に対する、PST が UOR の処理に費やした経過時間の率が示されます。

アプリケーション・プログラムの経過時間のモニター

IMS モニターは、UOR ごとの経過時間の測定値を記録することができます。この記録は、モニター間隔中に行われ、他の UOR が並行して実行されています。経過時間は、最初の DL/I 呼び出し（または他の呼び出し）の開始からそのプログラムの終了までの時間から計算されます。

アプリケーション・コードで費やされる時間と DL/I 処理で費やされる時間を区別することができます。以下の図にイベント間隔を図示します。

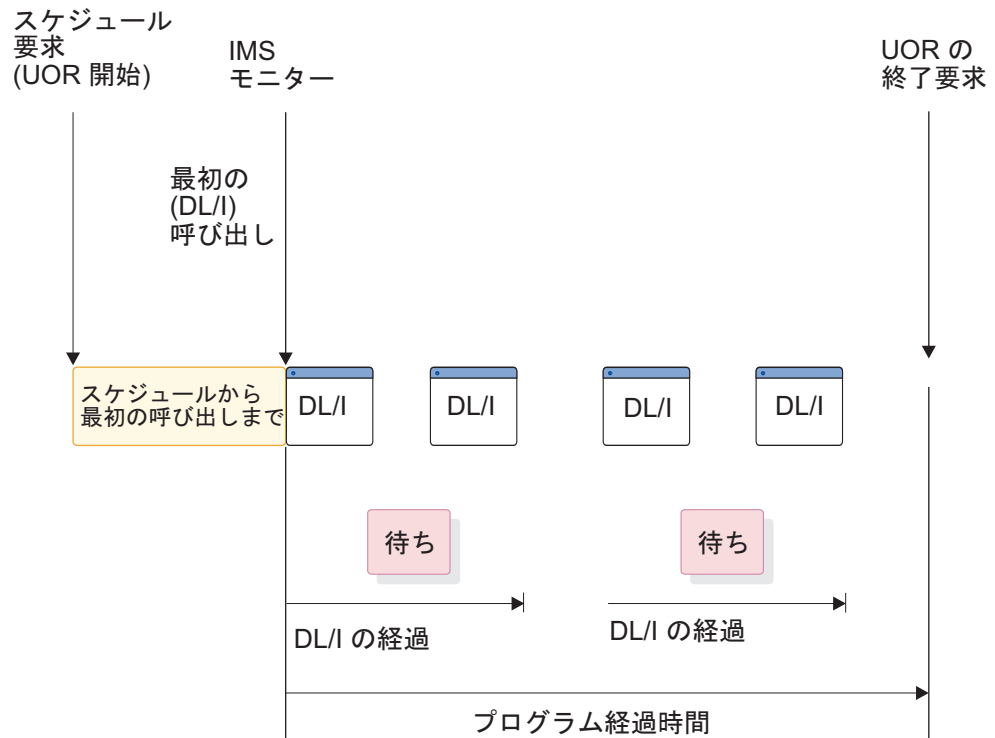


図 91. イベントの間隔

DL/I 呼び出しの経過時間内の、セグメント・データを手入手するための待ち時間は、別個に記録されます。同様に、スケジュールから最初の呼び出しまでの経過時間も別個に記録されます。この時間は、CCTL とデータベース・リソース・アダプター (DRA) 内での処理を含んでいます。

経過時間は、プログラム要約報告書に示されます。以下の例は、報告書を示しています。プログラムは、報告書内の各行にある PSB 名で識別されます。それぞれの行に、測定された間隔中の各 PSB のアクティビティーの要約が示されています。すなわち、スケジュール数、DL/I 呼び出し数、デキューされたトランザクション数、および入出力についての DL/I 呼び出しの待ち数の累計が示されます。報告書の行には、スケジュール当たりの経過時間、スケジュール当たりのプロセッサ時間、

スケジュール当たりのスケジュールから最初の DL/I 呼び出しまでの時間、および
 トランザクション当たりの経過時間についての平均計算値が示されます。トランザ
 クション当たりの呼び出しの頻度、DL/I 呼び出し当たりの入出力待ち数、および
 スケジュール当たりのデキューされたトランザクション数も示されます。TOTALS
 行は、モニター間隔中にアクティブだった PSB のすべてのアクティビティーを合計
 したものです。(PSB DUMMY 行は、スケジューリング中に領域が停止したために
 不完全になったスケジューリング、または疑似異常終了が起きたプログラムについ
 て調整する行です。)

この報告書のトランザクション とスケジュール は、UOR と解釈します。

IMS MONITOR		****PROGRAM SUMMARY****				TRACE START 1993 130 5:55:15				TRACE STOP 1993 130 5:59:49				PAGE 0075
		(A).....(B).....		(A).....(B).....		(A).....(B).....		(A).....(B).....		(A).....(B).....				
PSBNAME	NO. SCHEDS.	TRANS. DEQ.	CALLS	CALLS /TRAN	I/O IWAITS	I/O IWAITS /CALL	TRAN. DEQD. /SCH.	CPU TIME /SCHED.	DISTR. NO.	ELAPSED TIME /SCHED.	SCHED. TO 1ST CALL /SCHED.	DISTR. NO.	ELAPSED TIME /TRANS.	
PROGSC6D	1	13	60	4.6	46	0.7	13.0	10010	884A,B	290146255	15479797	886A,B	22318942	
PROGIT8C	3	17	225	13.2	166	0.7	5.6	90592	888A,B	256617508	73283259	890A,B	45285442	
PROGTS1C	2	25	47	1.8	0	0.0	12.5	10010	892A,B	239190808	7586234	894A,B	19135264	
PROGSP3D	1	23	792	34.4	182	0.2	23.0	10010	896A,B	322812716	48146258	898A,B	14035335	
PROGSP3A	13	36	1246	34.6	267	0.2	2.7	49782	900A,B	32801812	2228611	902A,B	11845098	
PROGIT1B	11	21	99	4.7	0	0.0	1.9	6341	904A,B	23212388	2036217	906A,B	12158870	
PROGSC2B	7	155	3068	19.7	1845	0.6	22.1	346112	908A,B	93655514	789390	910A,B	4229663	
PROGIT8A	12	28	434	15.5	293	0.6	2.3	34350	912A,B	30196795	1745815	914A,B	12941483	
PROGSP2C	1	10	179	17.9	205	1.1	10.0	10010	916A,B	221024429	53642029	918A,B	22102442	
PROGTS1B	8	20	54	2.7	0	0.0	2.5	5447	920A,B	39943245	2895	922A,B	15977298	
PROGSP3C	1	14	468	33.4	117	0.2	14.0	10010	924A,B	310644485	35978027	926A,B	22188891	
PROGIT1C	1	9	32	3.5	0	0.0	9.0	10010	930A,B	304892631	30226173	932A,B	33876959	
PROGSC2C	1	9	160	17.7	101	0.6	9.0	10010	934A,B	296909110	22242652	936A,B	32989901	
PROGIT2B	8	21	393	18.7	63	0.1	2.6	21703	938A,B	35126671	1798496	940A,B	13381589	
PROGIT2C	6	17	211	12.4	39	0.1	2.8	13312	942A,B	288883508	50698467	944A,B	101958885	
PROGTS1D	2	26	50	1.9	0	0.0	13.0	10010	950A,B	284944505	10613350	952A,B	21918808	
PROGSP3B	8	22	770	35.0	169	0.2	2.7	35737	954A,B	38016279	2149158	956A,B	13824101	
PROGIT1A	11	24	106	4.4	0	0.0	2.1	7925	958A,B	30883486	1935855	960A,B	14154931	
PROGSC4A	9	163	1775	10.8	5101	2.8	18.1	235921	963A,B	62172947	3011199	965A,B	3432862	
PROGSC6C	1	10	44	4.4	38	0.8	10.0	10010	967A,B	228098334	46568124	969A,B	22809833	
PROGSP2B	11	28	557	19.8	604	1.0	2.5	35069	971A,B	33309266	1181831	973A,B	13085783	
PROGIT8D	1	12	175	14.5	133	0.7	12.0	10010	975A,B	253392289	21274169	977A,B	21116024	
PROGSC4C	1	10	98	9.8	349	3.5	10.0	10010	979A,B	248736332	25930126	981A,B	24873633	
PROGSC6A	7	157	789	5.0	457	0.5	22.4	11703	983A,B	73936039	115979	985A,B	3296511	
PROGIT2A	7	22	430	19.5	71	0.1	3.1	28529	987A,B	37905001	2982	989A,B	12060682	
PROGSC2D	1	15	280	18.6	180	0.6	15.0	10010	991A,B	316194222	41527764	993A,B	21079614	
PROGSP2A	6	25	490	19.6	548	1.1	4.1	43177	995A,B	58277945	2467506	997A,B	13986707	
PROGSC2A	5	121	2363	19.5	1420	0.6	24.2	276187	1001A,B	88906184	6022954	1003A,B	3673809	
PROGIT2D	1	20	361	18.0	62	0.1	20.0	10010	1005A,B	386092737	111426279	1007A,B	19304636	
PROGSC4B	10	131	1421	10.8	4115	2.8	13.1	617016	1011A,B	53826667	2632409	1013A,B	4108905	
PROGSC4D	1	19	197	10.3	668	3.3	19.0	10010	1020A,B	227999124	46667334	1022A,B	11999953	
PROGSP2D	1	13	240	18.4	291	1.2	13.0	10010	1025A,B	327602445	52935987	1027A,B	25200188	
PROGSC6B	5	140	694	4.9	395	0.5	28.0	16884	1032A,B	78994223	3290769	1034A,B	2821222	
PROGIT1D	1	10	36	3.6	0	0.0	10.0	10010	1041A,B	290379922	15713464	1043A,B	29037992	
PROGIT8B	8	17	288	16.9	190	0.6	2.1	33436	1259A,B	35223857	2902	1261A,B	16575932	
**TOTALS	173	1403	18632	13.2	18115	0.9	8.1	90328		82303700	2972755		10148638	

プログラムごとの呼び出し処理の詳細 (呼び出しタイプ別に項目に分けられ、モニ
 ター間隔について要約されたもの) を調べるときには、呼び出し要約報告書を使用
 できます。複数ページの出力からの抜き出しを、以下の例に示してあります。入出
 力 PCB を使用した呼び出しが最初に示されて、小計が与えられています。次に、
 各データベース PCB および各外部サブシステムに対する、各タイプの呼び出しの
 合計がリストしてあります。PSB TOTAL 行は、各プログラムのデータの最後を表し
 ます。

IMS MONITOR		****CALL SUMMARY****				TRACE START 1993 130 5:55:15				TRACE STOP 1993 130 5:59:49				PAGE 0186
		(A)		(B)		(A)		(B)		(A)		(B)		
PSB NAME	PCB NAME	CALL FUNC	LEV NO. SEGMENT	STAT CODE	CALLS	IWAITS	IWAITS/ CALL	..ELAPSED MEAN	TIME... MAXIMUM	.NOT IWAIT MEAN	TIME.. MAXIMUM	DISTRIB. NUMBER		
PROGSC6B	I/O PCB	ISRT ()	()		138	0	0.00	372	1240	372	1240	598A,B,C		
		GU ()	()		134	133	0.99	2600917	20974615	2587532	20962866	602A,B,C		
		(GU) ()	()		3	0	0.00	15	16	15	16	716A,B,C		
		ASRT ()	()		3	0	0.00	330	333	330	333	869A,B,C		
		GU ()	()	QC	2	1	0.50	17639806	21219588	17634776	21209529	870A,B,C		
		I/O PCB SUBTOTAL			280	134	0.47	1370910		1364469				
	INVENTRB	DLET (03)IN060SUP			138	0	0.00	813	1289	813	1289	599A,B,C		

	GNP (03)IN060SUP	138	7	0.05	2112	112589	1047	112589	600A,B,C
	GU (01)IN010PAR	138	254	1.84	29511	75356	1195	19229	601A,B,C
	DL/I PCB SUBTOTAL								
		414	261	0.63	10812		1018		
	PSB TOTAL								
		694	395	0.56	559555		551114		
PROGSC2A	I/O PCB								
	TSRT ()	118	0	0.00	381	1496	381	1496	603A,B,C
	GU ()	114	284	2.49	3304809	21784513	3164423	21664181	632A,B,C
	(GU) ()	2	0	0.00	17	18	17	18	781A,B,C
	ASRT ()	3	0	0.00	367	444	367	444	871A,B,C
	GU ()	2	5	2.50	19931897	20045206	19799530	19925277	872A,B,C
	I/O PCB SUBTOTAL								
		239	289	1.20	1743339		1675270		
LOGVENDA	REPL (03)IN040SLQ	118	0	0.00	268	804	268	804	604A,B,C
	GNP (03)IN040SLQ	118	5	0.04	899	16995	218	305	605A,B,C
	REPL (02)VN030PAR	826	0	0.00	805	1578	805	1578	606A,B,C
	GNP (02)VN030PAR	826	873	1.05	19321	94521	456	1363	607A,B,C
	REPL (01)VN020REO	118	58	0.49	8879	48076	832	1682	623A,B,C
	GU (01)VN020REO	118	195	1.65	31688	360775	1300	1746	625A,B,C
	DL/I PCB SUBTOTAL								
		2124	1131	0.53	10145		636		
	PSB TOTAL								
		2363	1420	0.60	185445		170013		
PROGSC2D	I/O PCB								
	TSRT ()	14	0	0.00	377	621	377	621	608A,B,C
	GU ()	14	36	2.57	22360408	52048566	22221852	51901313	634A,B,C
	I/O PCB SUBTOTAL								
		28	36	1.28	11180393		11111115		
LOGVENDD	REPL (03)IN040SLQ	14	0	0.00	263	328	263	328	609A,B,C
	GNP (03)IN040SLQ	14	1	0.07	1407	16889	223	307	610A,B,C
	REPL (02)VN030PAR	98	0	0.00	820	1015	820	1015	611A,B,C

アプリケーション・プログラム DL/I 呼び出しのための入出力のモニター

IMS モニター報告書には、モニター間隔中における入出力オカレンスの合計数および UOR ごとに入出力オカレンスにかかった合計時間が表示されます。

プログラム入出力報告書は、モニター間隔中にアクティブだったすべての PSB についてこれら 2 つの合計を示し、プログラムが使用する各 PCB によって引き起こされた入出力待ち時間の詳細な内訳を含んでいます。

この報告書は、アプリケーション・プログラムの処理中に起きた競合を示します。競合の各タイプおよび起きた回数は、入出力 PCB またはデータベース PCB ごとに記録されます。この報告書は、合計待ち時間、最大待ち時間、および平均時間を示します。1 つの PSB のもとにある PCB ごとに、そして各 PSB のもとにあるすべての PCB について小計が示されます。

DDN/FUNC 欄には、データ・セットが DD 名でリストされます。MODULE 欄では、競合のソースを示すためにコードを使用しています。競合のタイプおよびコードを次に示します。示されている任意のコードは、IMS のみに適用します。

- スケジューリング

コード 競合

BLR ACBLIB からのロード/読み取り

SMN 仮想記憶管理

- データベースのアクセス

コード 競合

DBH OSAM 入出力

DLE DL/I 機能

VBH VSAM インターフェース

(物理セグメント・コード)

プログラム分離

入出力 PCB の呼び出しのための入出力待ちは、PSB の最初の項目としてグループ化されています。DL/I 呼び出しの場合は、入出力が行われたデータ・セットは、DDN/FUNC 見出しの下に示され、モジュール・コードにより、どのタイプの競合で待機が引き起こされたかが示されます。

データベース内の同じ物理セグメントの競合は、プログラム分離のための待機の原因になります。これは、PCB 行の DDN/FUNC 欄の PIdmb 項目 (ここで、dmb は物理データ・セットの DMB) によって示されます。MODULE 欄では、DBD 生成で割り当てられた物理セグメント・コードを使用してセグメント・タイプが識別されます。

アプリケーションが、アクセス方式として VSAM を使用してデータベースにアクセスしているときは、DL/I 呼び出しでは一般に入出力待ちになりません。MODULE 欄の項目 VBH は、VSAM のインターフェースが起きて入出力待ちがあったことを示します。

関係ないと思われる項目が、データベース PCB の DDN/FUNC 欄の下に表示されることがあります。その一例としては、検索されるデータ用の余地を作るためにバッファの除去が行われるような、データベース (DB-A) への検索呼び出しがあります。バッファの内容が別のデータベース (DB-B) に属するデータを含んでいると、報告書の入出力項目では、DB-B の DD 名が、DB-A への PCB アクセスに関して競合関係にあるものとして示されます。

トランザクション・キューイング報告書

以下の例のトランザクション・キューイング報告書には、DBCTL のトランザクションがリストされています。各トランザクション名は、スケジュール要求で CCTL によって指定された 8 バイトのトランザクション ID、または CCTL ID です。

CICS からのトランザクション ID は、トランザクション・マネージャーとして使用される場合、4 バイトの CICS トランザクション名と 4 バイトの CICS ID から構成されます。CCTL がトランザクション ID を指定しない場合には、DBCTL は接続時に得られた CCTL 領域 ID をデフォルトとして使用します。DBCTL に関するこの報告書では、トランザクション NUMBER DEQUEUED はスケジュール数であり、ON QUEUE WHEN SCHEDULED は常にゼロです。これは、IMS メッセージ・キューが関与していないためです。

IMS MONITOR	****TRANSACTION QUEUING****			TRACE START 1993 130 5:55:15	TRACE STOP 1993 130 5:59:49	PAGE 0181
TRANSACTION	NUMBER DEQUED	NUMBER SCHEDS.	..ON QUEUE WHEN SCHEDULED..... (B) MINIMUM MEAN MAXIMUM	DEQUED MEAN (A)	DISTRIBUTION NUMBER	
SC6X	13	1	0 0.00 0	13.00	883A,B	
IT8W	17	3	0 0.00 0	5.66	887A,B	
TS1Z	16	1	0 0.00 0	16.00	891A,B	
PS3X	23	1	0 0.00 0	23.00	895A,B	
PS3Y	17	7	0 0.00 0	2.42	899A,B	
IT1V	11	6	0 0.00 0	1.83	903A,B	
SC2Z	143	2	0 0.00 0	71.50	907A,B	
IT8U	12	7	0 0.00 0	1.71	911A,B	
PS2W	10	1	0 0.00 0	10.00	915A,B	
TS1U	12	4	0 0.00 0	3.00	919A,B	
PS3W	14	1	0 0.00 0	14.00	923A,B	
IT8Y	16	5	0 0.00 0	3.20	927A,B	

IT1W	9	1	0	0.00	0	9.00	929A,B
SC2W	9	1	0	0.00	0	9.00	933A,B
IT2V	13	5	0	0.00	0	2.60	937A,B
IT2W	17	6	0	0.00	0	2.83	941A,B
TS1V	9	1	0	0.00	0	9.00	945A,B
SC2V	12	5	0	0.00	0	2.40	947A,B
TS1W	11	1	0	0.00	0	11.00	949A,B
PS3V	13	3	0	0.00	0	4.33	953A,B
IT1U	9	6	0	0.00	0	1.50	957A,B
SC4U	11	5	0	0.00	0	2.20	962A,B
SC6W	10	1	0	0.00	0	10.00	966A,B
PS2V	8	6	0	0.00	0	1.33	970A,B
IT8X	12	1	0	0.00	0	12.00	974A,B
SC4W	10	1	0	0.00	0	10.00	978A,B
SC6U	14	6	0	0.00	0	2.33	982A,B
IT2Y	9	3	0	0.00	0	3.00	986A,B
SC2X	15	1	0	0.00	0	15.00	990A,B
PS2Y	17	2	0	0.00	0	8.50	994A,B
SC4Y	152	4	0	0.50	1	38.00	998A,B
SC2Y	106	2	0	0.00	0	53.00	1000A,B
IT2X	20	1	0	0.00	0	20.00	1004A,B
SC2U	15	3	0	0.00	0	5.00	1008A,B
SC4Z	123	5	0	0.60	1	24.60	1010A,B
TS1X	15	1	0	0.00	0	15.00	1015A,B
SC4X	19	1	0	0.00	0	19.00	1019A,B
PS2X	13	1	0	0.00	0	13.00	1024A,B
PS2Z	20	5	0	0.00	0	4.00	1028A,B
SC6Z	130	1	0	0.00	0	130.00	1031A,B
SC6V	10	4	0	0.00	0	2.50	1035A,B
SC6Y	143	1	0	0.00	0	143.00	1037A,B
IT1X	10	1	0	0.00	0	10.00	1040A,B
PS3U	19	6	0	0.00	0	3.16	1131A,B
IT2U	13	4	0	0.00	0	3.25	1146A,B

データベース・バッファのモニター

オンライン・システムにおける重要なリソースの 1 つは、データベース・バッファ・プールです。DL/I 呼び出しサービスの効率は、必要なデータベース論理レコードがこのバッファ内にあるかどうかによって異なります。つまり、バッファ内であれば、セグメント検索では追加の入出力が不要になります。

これは特に、置き換え呼び出しの前にデータベース呼び出しが介在する HOLD (保持) 呼び出しの場合に当てはまります。

QUERY POOL コマンドは、高速機能バッファ・プールの使用統計を表示する際に使用します。

以下の例に示されているデータベース・バッファ・プール報告書を使用して、全機能プール管理の一般的な効率を査定することができます。この報告書に示されているイベント・カウントは、特定のデータベースまたはプログラムに固有のものでなく、データベース・プールの使用状況を示しています。

関連資料: データベース・バッファ・プール報告書の詳細については、852 ページの『データベース・バッファ・プール報告書』を参照してください。

DATA BASE BUFFER POOL

FIX PREFIX/BUFFERS Y/Y
SUBPOOL ID 004K
SUBPOOL BUFFER SIZE 4096
TOTAL BUFFERS IN SUBPOOL 1000

16:09:59 16:25:10
START TRACE END TRACE DIFFERENCE
NUMBER OF LOCATE-TYPE CALLS 407636 4296793 3889157

NUMBER OF REQUESTS TO CREATE NEW BLOCKS	1	7	6
NUMBER OF BUFFER ALTER CALLS	75006	819359	744353
NUMBER OF PURGE CALLS	9137	93881	84744
NUMBER OF LOCATE-TYPE CALLS, DATA ALREADY IN OSAM POOL	313896	3317264	3003368
NUMBER OF BUFFERS SEARCHED BY ALL LOCATE-TYPE CALLS	453364	4779327	4325963
NUMBER OF READ I/O REQUESTS	86881	904487	817606
NUMBER OF SINGLE BLOCK WRITES BY BUFFER STEAL ROUTINE	0	0	0
NUMBER OF BLOCKS WRITTEN BY PURGE	32629	360434	327805
NUMBER OF LOCATE CALLS WAITED DUE TO BUSY ID	281	3173	2892
NUMBER OF LOCATE CALLS WAITED DUE TO BUFFER BUSY WRT	6	180	174
NUMBER OF LOCATE CALLS WAITED DUE TO BUFFER BUSY READ	0	0	0
NUMBER OF BUFFER STEAL/PURGE WAITED FOR OWNERSHIP RLSE	43	483	440
NUMBER OF BUFFER STEAL REQUESTS WAITED FOR BUFFERS	0	0	0
TOTAL NUMBER OF I/O ERRORS FOR THIS SUBPOOL	0	0	0
NUMBER OF BUFFERS LOCKED DUE TO WRITE ERRORS	0	0	0
QUOTIENT : TOTAL NUMBER OF OSAM READS + OSAM WRITES =	7.02		

TOTAL NUMBER OF TRANSACTIONS			

いずれかのデータベースがアクセス方式として VSAM を使用する場合、IMS モニターは VSAM BUFFER POOL の表題の一連の報告書を (サブプールごとに 1 つずつ) 生成します。以下の例は、これらの報告書の 1 つです。

I M S M O N I T O R BUFFER POOL STATISTICS

V S A M B U F F E R P O O L

FIX INDEX/BLOCK/DATA N/Y/N
 SHARED RESOURCE POOL ID VPL1
 SHARED RESOURCE POOL TYPE D
 SUBPOOL ID 2
 SUBPOOL BUFFER SIZE 4096
 NUMBER HIPERSPACE BUFFERS 0
 TOTAL BUFFERS IN SUBPOOL 4

	16:09:59 START TRACE	16:25:10 END TRACE	DIFFERENCE
NUMBER OF RETRIEVE BY RBA CALLS RECEIVED BY BUF HNDLR	432	6029	5597
NUMBER OF RETRIEVE BY KEY CALLS	40857	443840	402983
NUMBER OF LOGICAL RECORDS INSERTED INTO ESDS	414	6011	5597
NUMBER OF LOGICAL RECORDS INSERTED INTO KSDS	2132	25266	23134
NUMBER OF LOGICAL RECORDS ALTERED IN THIS SUBPOOL	0	0	0
NUMBER OF TIMES BACKGROUND WRITE FUNCTION INVOKED	0	0	0
NUMBER OF SYNCHRONIZATION CALLS RECEIVED	6494	70963	64469
NUMBER OF WRITE ERROR BUFFERS CURRENTLY IN THE SUBPOOL	0	0	0
LARGEST NUMBER OF WRITE ERRORS IN THE SUBPOOL	0	0	0
NUMBER OF VSAM GET CALLS ISSUED	44249	487181	442932
NUMBER OF VSAM SCHBFR CALLS ISSUED	0	0	0
NUMBER OF TIMES CTRL INTERVAL REQUESTED ALREADY IN POOL	11886	129668	117782
NUMBER OF CTRL INTERVALS READ FROM EXTERNAL STORAGE	32842	363635	330793
NUMBER OF VSAM WRITES INITIATED BY IMS	2370	29208	26838
NUMBER OF VSAM WRITES TO MAKE SPACE IN THE POOL	0	0	0
NUMBER OF VSAM READS FROM HIPERSPACE BUFFERS	0	0	0
NUMBER OF VSAM WRITES FROM HIPERSPACE BUFFERS	0	0	0
NUMBER OF FAILED VSAM READS FROM HIPERSPACE BUFFERS	0	0	0
NUMBER OF FAILED VSAM WRITES FROM HIPERSPACE BUFFERS	0	0	0
QUOTIENT : TOTAL NUMBER OF VSAM READS + VSAM WRITES =	2.19		

TOTAL NUMBER OF TRANSACTIONS			

IMS 内部リソースの使用

リソースに関する内部競合のレベルを調べるときに使用できるいくつかの要約報告書があります。

以下に挙げるのは、それらの報告書についての簡単な説明です。

プール・スペース障害報告書

プール・スペース障害要約報告書では、各領域内で一定量のストレージを利

用できなかった回数を示します。これは、ストレージが利用不能だったときの発生回数、バイト数、およびプール ID を示します。この要約を使用すれば、システム定義の変更または JCL 内の EXEC ステートメント内のバッファ数のオーバーライドにより、バッファ・プール割り振りを増やす必要があるかどうかを判別できます。

この報告書の形式は、以下の例のようになります。

POOL SPACE FAILURE SUMMARY

<i>POOL ID</i>	<i>BYTES REQ.</i>	<i>OCCURRENCES</i>
DLMP	8888	1
DLDP	7777	1
TOTAL		2

デッドロックになっているプログラム報告書

ある特定のデータベース・データ・セット内のセグメントの所有権に関して、1 対のプログラムがデッドロック状態になるたびに、デッドロック・イベント要約 (DEADLOCK EVENT SUMMARY) 報告書に記録されます。この報告書内の各行には、関係している 2 つの PSB が表示され、どちらに処理の優先権が与えられ (REQ-ING PSB)、動的バックアウト後にどちらが再処理を行う必要があるか (LOSING PSB) が示されます。デッドロック・イベント要約報告書は、以下の例に示してあります。

DEADLOCK EVENT SUMMARY

<i>REQ-ING PSB</i>	<i>LOSING PSB</i>	<i>DMBNAME</i>	<i>OCCURRENCES</i>
PSBNAME1	TPPSBRE3	DBASEBAL	1
TOTAL			1

IMS ラッチ競合報告書

IMS におけるタスク処理の基本的なシリアライゼーションは、IMS ラッチの所有権によって制御されます。各種のプログラムが実行中であるときには、それらは所有権を得るために競合します。それらがリソースを待つ場合は、ラッチを所有しているプログラムは、リソースを待っている他方の ITASK に通知する必要があります。リソースに関する競合のレベルを判定するときは、ラッチ競合統計 (LATCH CONFLICT STATISTICS) 報告書を使用します。

ラッチ競合統計報告書には、競合のレベルを表す種々のタイプのラッチとカウンターが示されます。以下の例は、ラッチ競合統計報告書の例を示しています。項目は、ラッチ名にしたがって編成されています。

モニターがアクティブの間にシステム・チェックポイントがとられた場合は、ラッチ競合統計はゼロにリセットされるため、この報告書に示された値は無効です。この状態のときは、以下のメッセージが報告書の先頭部分に挿入されます。

```
**** A CHECKPOINT OCCURRED DURING MONITOR RUN ****
**** LATCH CONFLICT STATISTICS ARE INVALID ****
**** SEE UTILITIES REFERENCE MANUAL ****
```

しかし、マスター端末オペレーターが、STATISTICS キーワード・パラメーターを指定して /CHECKPOINT コマンドを出す場合は、ラッチ競合統計はゼロにリセットされますが、このことは、IMS モニターには通知されません。したがって、統計が無効になっていることが DFSUTR20 には検出できないので、このメッセージが発行されることはありません。

推奨事項: モニターを実行中は、統計チェックポイントを出さないことをお勧めします。

IMS MONITOR ** GENERAL REPORTS ** TRACE START 1993 209...

LATCH CONFLICT STATISTICS

LATCH NAMES	COUNT FIELD	AT START	AT END	DIFF.
LOGL	CONTENTIONS	0	0	0
SMGT	CONTENTIONS	0	0	0
XCNQ	CONTENTIONS	0	0	0
ACTL	CONTENTIONS	0	0	0
CBTS	CONTENTIONS	0	0	0
DBLK	CONTENTIONS	0	0	0

IMS モニター出力の度数分布の使用方法

IMS モニター・データ・レコードから得られる報告書には、平均時間が示されている数多くの要約行が含まれています。

単なる平均時間や最大時間ではなく、時間測定されるイベントの分布を知りたい場合には、ある範囲の間隔全体にわたる度数分布の形で、イベントを個別に記録するよう報告書印刷ユーティリティに要求することができます。トランザクション・キュー・ロードや伝送ブロック・サイズなどの一部の分布は、時間依存ではありません。

- 以下の表には、領域要約別にソートされた報告書分布が示してあります。

表 72. 領域要約別報告書分布

報告書名	ID	説明
スケジューリングと終了	D1	経過時間
	D2	待ちなし時間
スケジュールの終わりから最初の DL/I 呼び出しまで	D3	N/A
実行経過時間	D4	N/A
DL/I 呼び出し	D5	経過時間
	D6	待ちなし時間
外部サブシステム呼び出し	D43	経過時間
DL/I 呼び出し当たりの待ち	D7	N/A
意図のためのアイドル	D8	N/A
チェックポイント	D20	経過時間
	D21	待ちなし時間

- 以下の表には、プログラム領域別にソートされた報告書分布が示してあります。

表 73. プログラム領域別報告書分布

報告書名	ID	説明
実行経過時間	D30	N/A
スケジュールの終わりから最初の DL/I 呼び出し まで	D31	N/A

- 以下の表には、プログラム要約別報告書分布が示してあります。

表 74. プログラム要約別報告書分布

報告書名	ID	説明
スケジュール当たりのプロセッサ時間	D15	N/A
スケジュール当たりの、デキューされたトランザク ション数	D14	N/A
スケジュール当たりの経過時間	D9	N/A
スケジュールの終わりから最初の DL/I 呼び出し まで	D10	N/A

- 以下の表には、呼び出し要約別にソートされた報告書分布が示してあります。

表 75. 呼び出し要約別報告書分布

報告書名	ID	説明
DL/I 呼び出し当たりの PSB 待ち数	D13	N/A
外部サブシステム呼び出し当たりの PSB 待ち数	D44	N/A
呼び出し当たりの PSB 経過時間	D11	N/A
呼び出し当たりの PSB 待ちなし時間	D12	N/A
PSB 外部サブシステム呼び出し	D45	N/A
	N/A	経過時間

- 以下の表に、バッファ・プール統計から得られる、待ち時間に関するいくつかの分布を示します。

表 76. 待ち時間分布

機能	ID	モジュール・キー
ストレージ	D22	SMN
OSAM 入出力	D23	DBH
VSAM 入出力	D24	VBH
ブロック・ローダー入出力	D27	BLR
HSAM 入出力	D34	DIE
PI エンキュー	D40	ありません。

度数分布出力の入手方法

IMS モニター報告書印刷ユーティリティに分布データを収集するよう要求するには、DIS 入力制御ステートメントを組み込む必要があります。こうすると、MEAN の表題の欄に項目を持つすべての報告書項目が、対応する度数分布を分布付録の一部として持つこととなります。さらに、報告書の各行は、DISTRIBUTION

NUMBER の表題の欄に識別参照番号を含んでいるので、その同じ番号が付けられている、付録の中の分布データを見つけることができます。

以下の各表には、主要な IMS モニター報告書と各報告書ごとにそれぞれ生成される度数分布のタイプが示してあります。各タイプは、報告書内の各セクションにどれだけ項目があるかに応じて、いくつかの分布に分かれます。度数分布のタイプごとに、適当な間隔または範囲でデータが累積されます。各タイプごとに使用される範囲の集合には、ID 列に示されているように ID が与えられます。

度数分布範囲の定義方法

要約行ごとに 10 個の間隔が定義され、各間隔の範囲に入るオカレンスが累積されます。間隔の範囲は、デフォルトの終点で事前設定されます。例えば、DL/I 呼び出し経過時間についての終点は、0、1000、2000、4000、8000、16000、32000、64000、128000、256000、INF (時間はすべてミリ秒単位) です。デフォルトの終点は、イベントに合うように選択されます。最初の間隔の下限は常にデフォルトとしてゼロが使用され、10 番目の間隔の上限は無量大 (INF) です。

複数のタイプの分布で同じ終点セットを使用できますが、各タイプに 1 つの分布 ID が割り当てられます。この ID を使用して、終点を再定義できます。デフォルトの終点をオーバーライドするときには、入力制御ステートメントを報告書印刷ユーティリティーに含めてください。そのステートメントで、分布 ID のタイプを指定し、希望する終点値を与えます。例えば、DL/I 呼び出し経過時間の終点は、次のように再指定できます。

```
D5 0,500,1000,1500,2000,4000,,,100000,500000
```

ここで指定されていない終点の値は、デフォルトの 32000 と 64000 のままであり、最後の値もそのまま (INF) です。

以下の例は、分布付録のサンプル・ページです。これを見れば、個々の分布に対してどのように番号が付けられているか、また分布のタイプによってどのように範囲が異なるかがわかります。行はペアになっており、2 番目の行に累積カウントが記録されます。

```
IMS MONITOR ****DISTRIBUTION APPENDIX**** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0200
# 1.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 2.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 3.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 4.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 5.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 6.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 7.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 8.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 9.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 10.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 11.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 12.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 13.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 14.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
```

# 15	0	0	0	0	0	0	0	0	0	1
	200000	400000	600000	800000	1000000	1200000	1400000	1600000	1800000	INF
# 16	1	0	0	0	0	0	0	0	0	8
	200000	400000	600000	800000	1000000	1200000	1400000	1600000	1800000	INF
	0	0	0	0	0	0	0	0	0	1

分布のデフォルト値の定義

度数分布表および待ち時間分布表にある ID を使用すれば、次のリストで探すことによって、分布のデフォルト・エンドポイントを判別できます。

D1、D2、D5、D6、D9、D10、D11、D12、D15、D18、D19、D20、D21、D22、D25、D27、D28、D29、D30、D31、D43、および D45

0、1000、2000、4000、8000、16000、32000、64000、128000、256000、INF

D3 0、50000、100000、150000、200000、250000、300000、350000、400000、450000、INF

D4 0、200000、400000、600000、800000、1000000、1200000、1400000、1600000、1800000、INF

D7、D13、および D44

0、0、1、2、3、4、5、6、7、8、INF

D8 0、100000、200000、300000、400000、500000、600000、700000、800000、900000、INF

D14、D16、D17

0、1、2、3、4、5、10、15、30、90、INF

D23、D24、D26、D32、D40、D42

0、2000、8000、24000、50000、100000、150000、200000、250000、300000、INF

D33、D34、D35

0、2000、4000、8000、16000、32000、64000、96000、128000、160000、INF

D36、D37

0、10、20、40、80、100、200、400、800、1000、INF

D38 0、1000、10000、100000、200000、500000、800000、1000000、1500000、2000000、INF

D39 0、1000、5000、10000、50000、100000、500000、1000000、5000000、10000000、INF

分布付録出力の解釈

異常なイベントの組み合わせが報告書要約行で報告されたと思われるときには、分布付録の明細出力を使用できます。

通常、リソース使用の問題を強調するには、平均時間と最大時間またはカウントのみで十分です。しかし、平均値では異常な分布が覆い隠されていて分からないと思われる場合には、IMS モニター出力レコードに含まれている詳細を利用できます。

第 49 章 DCCTL の IMS モニター報告書

DCCTL はトランザクション管理サブシステムであり、データベース・コンポーネントはありません。これは、外部サブシステム (ESS) 接続機能と併用することで、IMS 以外のデータベース・サブシステムにトランザクション管理機能を提供します。

このトピックでは、以下について説明します。

- IMS モニターが収集するイベント
- DCCTL 環境で IMS モニター報告書印刷プログラムが作成する報告書の内容

DCCTL は、IMS モニター報告書の形式や使用法を変更しません。データベースに固有の情報を含んでいる報告書および報告書内のフィールドがありますが、それらは DCCTL 環境には不要です。DCCTL に不要な報告書は、データなしで見出しのみが示されるか、あるいはまったく作成されません。DCCTL に不要な報告書には次のものがあります。

- データベース・バッファ・プール報告書
- VSAM バッファ・プール (DB) 報告書
- 呼び出し要約 (DB) 報告書
- プログラム入出力 (DB) 報告書

IMS モニターが時間測定情報を収集するイベント、システム・アクティビティ、およびストレージ域 (バッファ・プールまたはデータ・セット) の使用の詳細については、865 ページの『トランザクションのフローと IMS モニターのイベント』を参照してください。

IMS モニター・トレース・イベントの間隔

IMS モニター・トレースの間隔は、マスター端末のオペレーターが開始コマンドの入力と停止コマンドの入力のに /TRACE コマンドを使用して定義します。オンライン IMS イベントは、IMSMON データ・セットに入れられる IMS モニター・レコードに記録されます。

イベント・タイミングは、従属領域アクティビティに関連付けられます。以下の図に、時間測定が行われるイベントの間隔の境界を示します。

モニター・トレース間隔には、以下の間隔が含まれます。

- スケジューリングと終了
 - ブロック・ローダー使用中
 - スケジュール障害 (PSB 使用中およびスペース障害)
 - スケジューリングと終了の経過
 - 待ちなし
 - ACBLIB 待ち
- 領域占有 (「スケジューリングと終了の経過」全体とオーバーラップします)

- スケジュールから最初の呼び出しまで
- 実行の経過

領域の待ちなし時間は、待ち時間には含まれない経過時間のことです。ページング、またはプロセッサが優先順位の高いタスクにディスパッチされるために発生する遅延は、待ちなし時間の増加の原因となります。

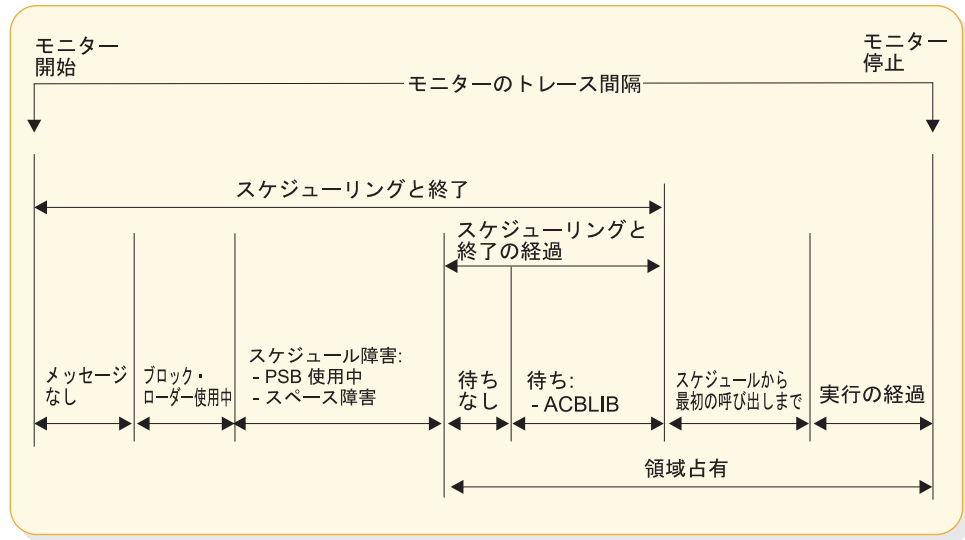


図 92. IMS モニター・トレース・イベントの間隔

IMS モニター報告書の概要

IMS モニターが収集するデータから得られる報告書のリストと、その報告書に含まれている主要なパフォーマンス・データを以下の表に示します。

リストで「MSC」が付いている報告書は、MSC がアクティブなど時のみ作成されます。

以下の表にリストした報告書の順序は、IMS モニター報告書印刷プログラムからの出力の順序と一致しています。モニター・スナップショットの期間内には、個々の報告書に必要な特定のイベントが含まれていないことがあります。その場合には、報告書の見出しまたはデータの一部のみが作成されます。

表 77. IMS モニター報告書の出力順序と情報

報告書名	主な内容
システム構成	モニター実行の文書
メッセージ・キュー・プール	トランザクションごとのバッファリングとメッセージ入出力
メッセージ形式バッファ・プール	入出力のカウント
ラッチ競合統計	IMS 内部処理
一般的な待ち時間イベント	SNAPQ の待ち時間
領域およびジョブ名	モニター実行の文書

表 77. IMS モニター報告書の出力順序と情報 (続き)

報告書名	主な内容
領域要約	経過時間と DL/I 呼び出しのカウンタ (DC)
領域待機	待ち時間
領域別プログラム	領域使用の経過時間
プログラム要約	プログラム全体の統計
通信要約	回線の経過時間
通信待機	回線別の待ち時間
回線機能	送信されたブロックのカウンタとサイズ
MSC トラフィック (MSC)	トランザクションのカウンタと経路指定
MSC 要約 (MSC)	宛先別のトランザクションのカウンタ
MSC キューイング要約 (MSC)	リンク別のカウンタとキューイング時間
トランザクション・キューイング	キューの負荷の統計
報告書	スペース障害とデッドロックのカウンタ
実行プロファイル	モニター実行の文書
分布付録	イベントの度数分布

IMS モニター報告書のデータ項目のほとんどは、経過時間です。通常、それらはマイクロ秒単位で表されます。1876534 という項目は、1.876534 秒または 1876 ミリ秒のことです。この規則に従っていない時間については、報告書の中で計測単位が表示されています。

さらに OCCURRENCES の見出しの下に、イベントのカウンタと、バイト数を表す数字がいくつか示されます。

関連概念:

899 ページの『MSC の IMS モニター報告書』

モニター実行の文書化

トレース間隔ごとに、実行された処理について、いくつかの一般報告書または要約報告書があります。これらの報告書は、ご使用の IMS モニター実行の文書の一部として使用できます。

トレースが実行されたときの条件は、できるだけ正確に記録することが重要です。文書には、トレースの前後に /DISPLAY コマンドで何回か得られたシステム状況情報、予期されているアプリケーション・プログラム・アクティビティのプロファイル、および必要な処理イベントを含めることができます。トレース間隔は、代表的な処理ロードを表すべきであり、偏った、あるいは不適当なヒストリー・レコードであってはなりません。

システム構成報告書のデータへの追加

SYSTEM CONFIGURATION という表題の付けられた最初の一般報告書は、BUFFER POOL STATISTICS というページ見出しのもとにあります。これは、IMS および z/OS システムの修正レベルを示しています。適用された IMS APAR のリストを追加し、アプリケーション・プログラムのサービス・レベルを含めることが

できます。このようにするのは特に、アプリケーション・プログラムが永続プログラムでなかったり、段階的に行われる実装の一部である場合です。システム構成の出力は、以下のセクションに例示してあります。

モニターのトレース間隔の記録

ほとんどの IMS モニター報告書の見出しには、トレースの開始時刻と停止時刻が示されています。その形式は、年 日 (ユリウス暦) HH:MM:SS です。トレース間隔の全長は、TRACE ELAPSED TIME IN SECONDS という見出しの下に秒単位で示されています。続く行には、IMS.MON データ・セットに入れられたトレース・レコードの数が示されています。モニターのトレース間隔の記録例は、以下の例に示してあります。

```
***IMS MONITOR*** BUFFER POOL STATISTICS TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0001
SYSTEM CONFIGURATION
```

```
SYSTEM CONFIGURATION :
IMS VERSION           : 4
RELEASE LEVEL        :
MODIFICATION NUMBER  :
```

モニター実行プロファイルの完了

処理率が、実行プロファイル報告書の最後の部分にまとめて示されています。この統計は、モニター間隔に関して、トランザクション・スループット、および DL/I と入出力アクティビティーの度合いを要約したものです。以下の例は、この報告書の例です。DCCTL 環境では、DL/I アクティビティーは、データ通信呼び出しおよび GSAM データベースの呼び出しに限定されます。他のタイプの DL/I データベースに対するデータベース呼び出しは、DCCTL ではサポートされていません。

実行プロファイル報告書の下の部分に、以下のいくつかの率が表示されます。

- 各領域ごとの、DL/I 経過時間に対するプログラム経過時間の率
- DL/I 処理時の、待ち時間に対する DL/I 経過時間の率
- 他のサブシステム呼び出し経過時間に対するプログラム経過時間の率
- 他のサブシステム呼び出し経過時間に対する DL/I 経過時間の率

各従属領域には、領域 1 から始まるシーケンス番号が付き、これで識別されます。

```
IMS MONITOR **RUN PROFILE** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0184
TRACE ELAPSED TIME IN SECONDS.....274.6
TOTAL NUMBER OF MESSAGES DEQUEUED....1403
TOTAL NUMBER OF SCHEDULES          .....173
NUMBER OF TRANSACTIONS PER SECOND   5.1
TOTAL NUMBER OF DL/I CALLS ISSUED...18632
NUMBER OF DL/I CALLS PER TRANSACTION 13.2
NUMBER OF OSAM BUFFER POOL I/O'S.    0,      0.0 PER TRANSACTION
NUMBER OF MESSAGE QUEUE POOL I/O'S.....0,    0.0 PER TRANSACTION
NUMBER OF FORMAT BUFFER POOL I/O'S.....0,    0.0 PER TRANSACTION
RATIO OF PROGRAM ELAPSED TO DL/I ELAPSED:
REGION 1: 1.09
REGION 2: 1.09
REGION 3: 1.00
REGION 4: 1.02
REGION 5: 1.01
REGION 6: 1.00
REGION 7: 1.00
REGION 8: 1.00
REGION 9: 1.17
REGION 10: 1.00
```

```

REGION 11: 1.00
REGION 49: 1.03
REGION 50: 1.19
RATIO OF DL/I ELAPSED TO DL/I IWAIT:
REGION 1: 325.65
REGION 2: 73.49
REGION 4: 100.35
REGION 5: 85.76
REGION 6: 82.99
REGION 47: 95.64
REGION 48: 45.93
REGION 49: 9.22

```

領域およびジョブ名報告書を使用して、領域と z/OS ジョブ名を突き合わせる事ができます。ジョブ名は、トレースが開始される前にオペレーターが開始したすべての従属領域の EXEC ステートメントのステップ名に相当します。領域ジョブ名は、GENERAL REPORTS という見出しの付いたモニター出力ページに表示されます。940 ページの『チェックポイントの影響の検出』に示されているとおりです。

一部の汎用プロセス率は、いくつかのバッファ・プール統計報告書の最後の部分に示されます。それらの率を、トレース間隔の文書化プロファイルに含める事ができます。それらの率は、1つのアプリケーションまたはシステム・リソースに固有のものではなく、一連のモニター実行全体のバリエーションの標識として使用することができます。DCCTL では、次の率が示されます。

- すべての待ち数をトランザクションの合計数で除算したもの

この値は、940 ページの『メッセージ・キュー処理のモニター』のメッセージ・キュー・プール報告書にあります。この率は、トランザクション 1 つ当たりの、メッセージ・キューイング機能の処理に必要な物理入出力アクティビティを示します。

- (事前取り出し入出力数 + 即時取り出し入出力数 + ディレクトリー入出力数) の合計数を、トランザクションの合計数で除算したもの

この値は 939 ページの『MFS アクティビティのモニター』に示されています。この率は、トランザクション 1 つ当たりの、トレース期間中に MFS 機能の処理に必要な物理入出力アクティビティを示します。

IMS モニター報告書オカレンスの検査

IMS モニター報告書印刷プログラムからの出力を調べる場合、報告書の見出しがあっても、該当するデータがリストされていると想定しないでください。システム定義オプションおよびユーティリティー制御ステートメントにより、出力の内容は次のような影響を受けます。

- 制御ステートメントで DLI を指定しないと、呼び出し要約報告書は出力されません。
- 出力には、制御ステートメントで DIS または DISTRIBUTION を指定していない限り、1組の分布報告書は含まれません。多数の報告書に現れる DISTRIBUTION NUMBER という表題の欄には、分布報告書に含まれている項目の相互参照が含まれています。
- 制御ステートメントで ONLY DLI を指定すると、呼び出し要約報告書だけしか出力されません。

多数の要約報告書では、開始値と終了値の差を計算するためにシステム状況が必要であり、しかもこの状況は /TRACE SET OFF 処理時に得られるため、IMS モニターの実行を IMS 制御領域の終了前に終わらせる必要があります。トレースが正しく停止しなかった場合は、次のメッセージが出されます。

```
NO QUEUE BUFFER POOL TRACES AT END TIME ON MONITOR LOG TAPE
****QUEUE BUFFER POOL REPORT CANCELLED****
```

同様に、他の要約報告書も作成されません。

MESSAGE FORMAT BUFFER POOL のセクションは、メッセージ形式サービス (MFS) を使用する装置がシステム定義に指定されている場合のみ、含まれます。

IMS モニターが、特定の IMS モニター報告書またはその報告書の 1 セクションを作成するために使用するソース・データをトレース間隔中に記録していなかった場合は、報告書には見出しのみが表示されます。

従属領域内のアクティビティのモニター

IMS モニターは、トレース間隔中にアクティブである、/trace コマンドで識別されたすべての従属領域について、時間測定情報を収集します。このプログラムでは、イベントごとの経過時間の合計、検出された最長イベントの時間、およびすべての記録イベントについての平均時間を記録します。

時間測定値を表示する主要な報告書は 3 つあります。それらの報告書とその内容のリストは次のとおりです。

領域要約報告書 (REGION SUMMARY REPORT)

- スケジューリングと終了
- スケジュールの終わりから最初の呼び出しまで
- 以下について表示される、個別要約付きの実行経過
 - DL/I 呼び出し
 - 外部サブシステム・サービスとコマンド呼び出し
 - 外部サブシステムのデータベース・アクセス呼び出し
 - チェックポイント処理
 - 領域占有

領域待機

- スケジューリングと終了時の待機
- DL/I 呼び出し時の待機
- 外部サブシステム呼び出し時の待機
- チェックポイント時の待機

領域別プログラム

- 実行の経過
- スケジュールの終わりから最初の呼び出しまで

これら 3 つの報告書は、以下の例に示してあります。

従属領域でのアクティビティーは、次の 5 つの時間測定情報カテゴリーのいずれかに入ります。

- スケジューリングと終了の経過時間

スケジューリング処理には、アクティブ IMS.ACBLIBA/B データ・セットからのブロック・ロード、および PSB の所有権の獲得などの多数の準備イベントが含まれます。アプリケーション・プログラムの終了後に領域アクティビティーを終わらせるのに必要な時間も含まれます。

- スケジュールの終わりから最初の呼び出しまでの経過時間

この時間は、制御プログラム・サービスの始まりにマークを付ける初期呼び出し(メッセージ・キューまたは外部サブシステムへの呼び出し)の前に行われる、アプリケーション・プログラムの初期設定とハウスキーピング用として確保されます。これは、複数トランザクションが単一のスケジューリングで処理されるときに、反復して行われない処理の時間です。

- プログラム経過時間 (すべての呼び出しを含む)

この時間には、アプリケーション・プログラムの主要な処理が含まれ、最初の呼び出しから、プログラムからの戻り(プログラムの終了)までの時間が測定されます。

- DL/I 呼び出しの実行に要した経過時間

この時間は、すべての DL/I 呼び出しの時間を含みます。各 DL/I 呼び出しイベントは、アプリケーション・プログラムへの戻りの呼び出し時刻から測定されます。

- 外部サブシステム呼び出しの実行に要した経過時間

この時間は、すべての外部サブシステム呼び出しの時間を含みます。各外部サブシステム・イベントは、呼び出し時刻から IMS への戻りまで測定されます。

以下の例は、領域要約報告書の例を示しています。

IMS MONITOR		****REGION SUMMARY****			TRACE START 1993 130 5:55:15			TRACE STOP 1993 130 5:59:49			PAGE 0011
		(A)			(B)						
OCCURRENCES	ELAPSED TIME.....			NOT IWAIT TIME(ELAPSED-IWAIT)			DISTRIBUTION			
		TOTAL	MEAN	MAXIMUM	TOTAL	MEAN	MAXIMUM	NUMBER			
SCHEDULING AND TERMINATION											
**REGION	5	5	4146	829	948	4146	829	948	287A,B		
**REGION	6	7	6028	861	1067	6028	861	1067	214A,B		
**REGION	8	8	6847	855	1098	6847	855	1098	129A,B		
**REGION	10	7	9664	1380	3668	9664	1380	3668	272A,B		
**REGION	47	6	5482	913	1021	5482	913	1021	145A,B		
**REGION	49	3	2612	870	917	2612	870	917	443A,B		
**TOTALS		123	126042	1024		126042	1024				
SCHEDULE TO FIRST CALL											
**REGION	1	1	15479797	15479797	15479797				555		
**REGION	2	1	22376350	22376350	22376350				564		
**REGION	3	1	15169488	15169488	15169488				578		
**REGION	4	1	48146258	48146258	48146258				584		
**REGION	48	1	795351	795351	795351				592		
**REGION	49	4	2960425	740106	2951746				442		
**REGION	50	1	15713464	15713464	15713464				575		
**TOTALS		168	514286738	3061230							
ELAPSED EXECUTION											
**REGION	1	1	290146255	290146255	290146255				1		
**REGION	2	1	252290108	252290108	252290108				2		
**REGION	3	1	259496970	259496970	259496970				3		
**REGION	4	1	322812716	322812716	322812716				4		
**REGION	48	1	273871107	273871107	273871107				48		
**REGION	49	4	271703421	67925855	155176058				49		

**REGION	50	1	290379922	290379922	290379922				50	
**TOTALS		173	14238540145	82303700						
DL/I CALLS							IWT/CALL(C)			
**REGION	1	60	264626241	4410437	88981490	263813671	4396894	88970053	0.76	247A,B,C
**REGION	2	223	230505269	1033655	61048758	227368742	1019590	61011153	0.73	237A,B,C
**REGION	3	29	257704383	8886358	69000514	257704383	8886358	69000514	0.00	98A,B,C
**REGION	4	792	313735347	396130	52439653	310609035	392183	52439653	0.22	180A,B,C
**REGION	49	592	262886317	444064	30202068	234394017	395935	30159782	2.46	177A,B,C
**REGION	50	36	242591451	6738651	48651260	242591451	6738651	48651260	0.00	289A,B,C
**TOTALS		18632	12386905286	664818	12024562411	645371			0.97	
IDLE FOR INTENT										
CHECKPOINT		NONE								
REGION OCCUPANCY		NONE								
**REGION	1	100.0%								
**REGION	2	100.0%								
**REGION	3	100.0%								
**REGION	4	100.0%								
**REGION	48	100.0%								
**REGION	49	100.0%								
**REGION	50	100.0%								

以下の例は、領域待機報告書の例を示しています。

IMS MONITOR	****REGION IWAIT****	TRACE START 1993 130 5:55:15			TRACE STOP 1993 130 5:59:49			PAGE 0023
**REGION	5 OCCURRENCES	TOTAL	MEAN	MAXIMUM	FUNCTION	MODULE	DISTRIBUTION NUMBER	
SCHEDULING + TERMINATION								
SUB-TOTAL								
TOTAL								
DL/I CALLS								
	11	181816	16528	24375	DD=IMMSTR2A	DBH	117	
	8	112831	14103	17846	DD=IMMSTR1A	DBH	118	
	5	85460	17092	33717	DD=IMMSTR3A	DBH	119	
	1	1633	1633	1633	INT=BMPFPE05	BLR-64BIT	14	
	1	4242	4242	4242	PSB=DDLTRN17	BLR-64BIT	20	
	5	58420	11684	14643	DD=IMINDEXA	VBH	120	
	12	173866	14488	22152	DD=PRODCNTA	VBH	121	
	3	100576	33525	68373	DD=IMMSTR2B	DBH	428	
	1	17921	17921	17921	DD=IMMSTR3B	DBH	429	
	1	17195	17195	17195	DD=IMMSTR1B	DBH	430	
	1	13577	13577	13577	DD=IMINDEXB	VBH	431	
	3	49928	16642	20396	DD=PRODCNTB	VBH	432	
	4	10973	2743	2787	DD=ITEMACTB	DBH	453	
	2	37680	18840	27664	DD=IAINDEXB	VBH	454	
	49	1500067	30613	138284	DD=INVENTRA	DBH	472	
	23	345595	15025	27613	DD=VENDORDA	VBH	473	
	1	342952	342952	342952	PI=VENDORDA...1		498	
	1	14612	14612	14612	PI=VNSINDXA...1		499	
	6	69203	11533	19492	DD=VNSINDXA	VBH	500	
TOTAL		136	3132672	23034				

以下の例は、領域別プログラム報告書の例を示しています。

IMS MONITOR	****PROGRAMS BY REGION****	TRACE START 1993 130 5:55:15			TRACE STOP 1993 130 5:59:49			PAGE 0069
**REGION	1	(A) ELAPSED EXECUTION TIME			(B) SCHEDULING END TO FIRST CALL			DISTRIBUTION
	OCCURRENCES	TOTAL	MEAN	MAXIMUM	TOTAL	MEAN	MAXIMUM	NUMBER
PROGSC6D	1	290146255	290146255	290146255	15479797	15479797	15479797	885A,B
REGION TOTALS	1	290146255	290146255		15479797	15479797		
PROGIT8C	1	252290108	252290108	252290108	22376350	22376350	22376350	889A,B
REGION TOTALS	1	252290108	252290108		22376350	22376350		
PROGTS1C	1	259496970	259496970	259496970	15169488	15169488	15169488	893A,B
REGION TOTALS	1	259496970	259496970		15169488	15169488		

**REGION	4							
PROGSP3D	1	322812716	322812716	322812716	48146258	48146258	48146258	897A,B
REGION TOTALS	1	322812716	322812716		48146258	48146258		
**REGION	5							
PROGSP3A	2	62893103	31446551	40693590	5435	2717	2862	901A,B
PROGTS1B	1	61794787	61794787	61794787	2790	2790	2790	1271A,B
PROGSP3B	1	18294458	18294458	18294458	3104	3104	3104	1350A,B
PROGIT2B	1	36095342	36095342	36095342	2731	2731	2731	1363A,B
PROGSC2A	1	93902771	93902771	93902771	1667791	1667791	1667791	1401A,B
REGION TOTALS	6	272980461	45496743		1681851	280308		
**REGION	6							
PROGIT1B	2	39000315	19500157	23703429	5286	2643	2801	905A,B
PROGTS1B	1	34293636	34293636	34293636	3136	3136	3136	1207A,B
PROGSP3A	1	51887767	51887767	51887767	2534	2534	2534	1278A,B
PROGSP3B	2	67375031	33687515	40291430	17210570	8605285	17213287	1328A,B
PROGIT8A	1	69132416	69132416	69132416	3291	3291	3291	1359A,B
PROGSC4A	1	30165017	30165017	30165017	2571	2571	2571	1433A,B
REGION TOTALS	8	291854182	36481772		17193752	2149219		
**REGION	7							
PROGSC2B	1	269618583	269618583	269618583	5047875	5047875	5047875	909A,B
REGION TOTALS	1	269618583	269618583		5047875	5047875		
**REGION	8							
PROGIT8A	1	5181039	5181039	5181039	2928	2928	2928	913A,B
PROGSP3A	1	27304257	27304257	27304257	3350	3350	3350	1132A,B
PROGSC4B	1	37286872	37286872	37286872	3009	3009	3009	1255A,B
PROGIT2A	1	36902995	36902995	36902995	2850	2850	2850	1298A,B
PROGIT1B	1	30407479	30407479	30407479	2565	2565	2565	1336A,B
PROGIT1A	3	109875360	36625120	45190114	4279008	1426336	4272096	1357A,B
PROGIT8B	1	23405220	23405220	23405220	2679	2679	2679	1395A,B
REGION TOTALS	9	270363222	30040358		4296389	477376		

チェックポイントの影響の検査

チェックポイント処理は、システム・ログに入れられるレコード数によって決まる指定の頻度で、制御プログラムによって開始することができます。他のチェックポイントは、オペレーター・コマンドによって取ることができます。

領域要約報告書の領域 0 の要約の最後にあるチェックポイント行に、以下のものが表示されます。

- モニター間隔中にシステム・チェックポイントがとられた回数
- 経過時間
- 待ちなし時間

チェックポイント処理時の待ち時間は、領域待機報告書の最初の領域要約の終わりに示されます。DD 名とモジュール・コードの組み合わせごとに遅延を検出することができます。この部分の代表的な項目は、メッセージ・キュー・データ・セットおよび再始動データ・セットに関するものです。ストレージ待ちが原因である場合には、FUNCTION 欄の下の項目は、STG.= とその後続くプール ID です。

領域占有の測定

領域アクティビティの尺度となるのは、領域占有の割合 (%) です。これは、一般的には、トレース間隔に対する、領域で処理を実行するときの経過時間の率です。

領域占有時間には、メッセージがないときの時間、ブロック・ロードが遅れたときの時間、あるいは PSB を使用できないときの時間は含まれません。領域要約報告書の最後のセクションには、時間測定が行われたイベントが収集されたすべてのアクティブ領域がリストされ、計算された領域占有百分率が示されます。

アプリケーション・プログラムの経過時間のモニター

IMS モニターは、各トランザクションの経過時間およびアプリケーション・プログラムのスケジューリングの経過時間の測定値を記録できます。このことは、他のプログラムが並行して実行されている間に、モニター間隔中に行われます。

経過時間は、最初の DL/I 呼び出し (または他の呼び出し) の開始からそのプログラムの終了までの時間から計算されます。アプリケーション・コードで費やされる時間と DL/I 処理で費やされる時間を区別することができます。以下の図にイベント間隔を図示します。

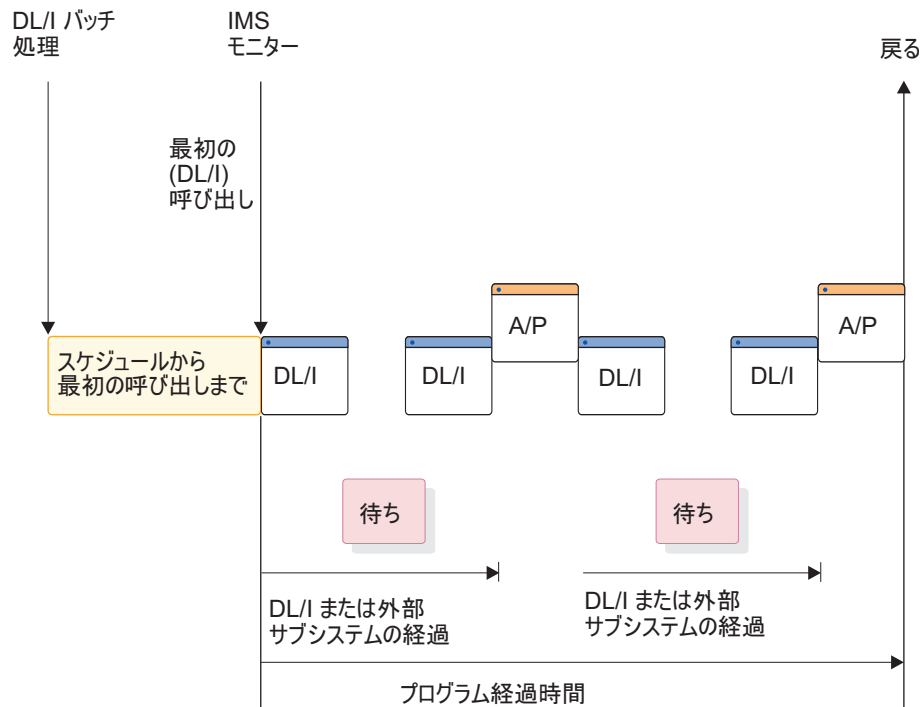


図 93. 経過時間イベントの間隔

DL/I 呼び出しの経過時間内の、セグメント・データを入手するための待ち時間は、別個に記録されます。同様に、外部サブシステム呼び出しの経過時間内の、外部サブシステムでの処理時間は、待ち時間として別個に記録されます。アプリケーション処理 (A/P) 時間には、プログラム・オブジェクト・コードによって費やされるマシン・サイクルのほかに、多数の種類の副次的サービス (サブルーチンのロード、z/OS データ・セットへの入出力、およびオーバーレイ処理など) の時間が含まれます。ディスパッチされるのをプログラムが待機している場合、または実記憶域を使用するためにプログラムでページングが必要な場合には、これらに起因する遅れも、アプリケーション・プログラム処理時間に含まれます。プログラムはスケジュールごとに多数のトランザクションを実行できるので、スケジュールから最初の呼び出しまでの経過時間は別個に記録されます。この時間には、アプリケーション・プログラムが実行する初期設定が含まれるとともに、プログラムのロードの時間も含まれます。

経過時間は、プログラム要約報告書に示されます。以下の例は、プログラム要約報告書の例です。プログラムは、報告書内の各行にある PSB 名で識別されます。それ

それぞれの行に、測定された間隔中の各 PSB のアクティビティーの要約が示されています。すなわち、スケジュール数、DL/I 呼び出し数、完了した (デキューされた) トランザクション数、および DL/I 呼び出し、入出力呼び出し、外部サブシステム処理の待ち数の累計が示されます。報告書の各行には、次のものについて計算された平均時間が示されています。

- スケジュール当たりの経過時間
- スケジュール当たりのプロセッサ時間
- スケジュール当たりの、スケジュールから最初の DL/I 呼び出しまでの経過時間
- トランザクション当たりの経過時間

トランザクション当たりの呼び出しの頻度、DL/I 呼び出し当たりの入出力待ち数、外部サブシステム呼び出し当たりの待ち数、およびスケジュール当たりのデキューされたトランザクション数も示されています。TOTALS 行は、モニター間隔中にアクティブだった PSB のすべてのアクティビティーを合計したものです。(PSB DUMMY 行は、スケジュール時に領域が停止したために不完全になったスケジュールリング、または疑似異常終了が起きたプログラムについて調整する行です。)

IMS MONITOR		****PROGRAM SUMMARY****										TRACE START 1993 130 5:55:15		TRACE STOP 1993 130 5:59:49		PAGE 0075
PSBNAME	NO. SCHEDS.	TRANS. DEQ.	CALLS	CALLS /TRAN	I/O IWAITS	I/O IWAITS /CALL	TRAN. DEQD. /SCH.	CPU TIME /SCHED.	DISTR. NO.	ELAPSED TIME /SCHED.	SCHED. TO 1ST CALL /SCHED.	DISTR. NO.	ELAPSED TIME /TRANS.	(A).....(B).....		
														(A).....(B).....	(A).....(B).....	
PROGSC6D	1	13	60	4.6	46	0.7	13.0	10010	884A,B	290146255	15479797	886A,B	22318942			
PROGIT8C	3	17	225	13.2	166	0.7	5.6	90592	888A,B	256617508	73283259	890A,B	45285442			
PROGTS1C	2	25	47	1.8	0	0.0	12.5	10010	892A,B	239190808	7586234	894A,B	19135264			
PROGSP3D	1	23	792	34.4	182	0.2	23.0	10010	896A,B	322812716	48146258	898A,B	14035335			
PROGSP3A	13	36	1246	34.6	267	0.2	2.7	49782	900A,B	32801812	2228611	902A,B	11845098			
PROGIT1B	11	21	99	4.7	0	0.0	1.9	6341	904A,B	23212388	2036217	906A,B	12158870			
PROGSC2B	7	155	3068	19.7	1845	0.6	22.1	346112	908A,B	93655514	789390	910A,B	4229663			
PROGIT8A	12	28	434	15.5	293	0.6	2.3	34350	912A,B	30196795	1745815	914A,B	12941483			
PROGSP2C	1	10	179	17.9	205	1.1	10.0	10010	916A,B	221024429	53642029	918A,B	22102442			
PROGTS1B	8	20	54	2.7	0	0.0	2.5	5447	920A,B	39943245	2895	922A,B	15977298			
PROGSP3C	1	14	468	33.4	117	0.2	14.0	10010	924A,B	310644485	35978027	926A,B	22188891			
PROGIT1C	1	9	32	3.5	0	0.0	9.0	10010	930A,B	304892631	30226173	932A,B	33876959			
PROGSC2C	1	9	160	17.7	101	0.6	9.0	10010	934A,B	296909110	22242652	936A,B	32989901			
PROGIT2B	8	21	393	18.7	63	0.1	2.6	21703	938A,B	35126671	1798496	940A,B	13381589			
PROGIT2C	6	17	211	12.4	39	0.1	2.8	13312	942A,B	288883508	50698467	944A,B	101958885			
PROGTS1D	2	26	50	1.9	0	0.0	13.0	10010	950A,B	284944505	10613350	952A,B	21918808			
PROGSP3B	8	22	770	35.0	169	0.2	2.7	35737	954A,B	38016279	2149158	956A,B	13824101			
PROGIT1A	11	24	106	4.4	0	0.0	2.1	7925	958A,B	30883486	1935855	960A,B	14154931			
PROGSC4A	9	163	1775	10.8	5101	2.8	18.1	235921	963A,B	62172947	3011199	965A,B	3432862			
PROGSC6C	1	10	44	4.4	38	0.8	10.0	10010	967A,B	228098334	46568124	969A,B	22809833			
PROGSP2B	11	28	557	19.8	604	1.0	2.5	35069	971A,B	33309266	1181831	973A,B	13085783			
PROGIT8D	1	12	175	14.5	133	0.7	12.0	10010	975A,B	253392289	21274169	977A,B	21116024			
PROGSC4C	1	10	98	9.8	349	3.5	10.0	10010	979A,B	248736332	25930126	981A,B	24873633			
PROGSC6A	7	157	789	5.0	457	0.5	22.4	11703	983A,B	73936039	115979	985A,B	3296511			
PROGIT2A	7	22	430	19.5	71	0.1	3.1	28529	987A,B	37905001	2982	989A,B	12060682			
PROGSC2D	1	15	280	18.6	180	0.6	15.0	10010	991A,B	316194222	41527764	993A,B	21079614			
PROGSP2A	6	25	490	19.6	548	1.1	4.1	43177	995A,B	58277945	2467506	997A,B	13986707			
PROGSC2A	5	121	2363	19.5	1420	0.6	24.2	276187	1001A,B	88906184	6022954	1003A,B	3673809			
PROGIT2D	1	20	361	18.0	62	0.1	20.0	10010	1005A,B	386092737	111426279	1007A,B	19304636			
PROGSC4B	10	131	1421	10.8	4115	2.8	13.1	617016	1011A,B	53826667	2632409	1013A,B	4108905			
PROGSC4D	1	19	197	10.3	668	3.3	19.0	10010	1020A,B	227999124	46667334	1022A,B	11999953			
PROGSP2D	1	13	240	18.4	291	1.2	13.0	10010	1025A,B	327602445	52935987	1027A,B	25200188			
PROGSC6B	5	140	694	4.9	395	0.5	28.0	16884	1032A,B	78994223	3290769	1034A,B	2821222			
PROGIT1D	1	10	36	3.6	0	0.0	10.0	10010	1041A,B	290379922	15713464	1043A,B	29037992			
PROGIT8B	8	17	288	16.9	190	0.6	2.1	33436	1259A,B	35223857	2902	1261A,B	16575932			
**TOTALS	173	1403	18632	13.2	18115	0.9	8.1	90328		82303700	2972755		10148638			

呼び出し要約報告書を使用すれば、プログラムごとの呼び出し処理詳細を調べることができます。その詳細は、タイプまたは呼び出し別に項目に分けられ、モニター間隔について要約されています。複数ページの出力からの抜き出しを、以下の呼び出し要約報告書の例に示してあります。入出力 PCB を使用した呼び出しが最初に示されて、小計が与えられています。次に、各データベース PCB および各外部サブシステムに対する、各タイプの呼び出しの合計がリストしてあります。PSB TOTAL 行は、各プログラムのデータの最後を表します。

IMS MONITOR		****CALL SUMMARY****			TRACE START 1993 130 5:55:15			TRACE STOP 1993 130 5:59:49			PAGE 0186	
PSB NAME	PCB NAME	CALL FUNC	LEV NO.SEGMENT	STAT CODE	CALLS	IWAITS	(C)	(A)	(B)	.NOT IWAIT	TIME.. MAXIMUM	DISTRIB. NUMBER
							IWAITS/CALL	..ELAPSED TIME.. MEAN	MEAN			
PROGSC6B	I/O PCB	ISRT ()			138	0	0.00	372	1240	372	1240	598A,B,C
		GU ()			134	133	0.99	2600917	20974615	2587532	20962866	602A,B,C
		(GU) ()			3	0	0.00	15	16	15	16	716A,B,C
		ASRT ()			3	0	0.00	330	333	330	333	869A,B,C
		GU ()		QC	2	1	0.50	17639806	21219588	17634776	21209529	870A,B,C
		I/O PCB SUBTOTAL			280	134	0.47	1370910		1364469		
		PSB TOTAL			280	134	0.47	1370910		1364469		
PROGSC2A	I/O PCB	ISRT ()			118	0	0.00	381	1496	381	1496	603A,B,C
		GU ()			114	284	2.49	3304809	21784513	3164423	21664181	632A,B,C
		(GU) ()			2	0	0.00	17	18	17	18	781A,B,C
		ASRT ()			3	0	0.00	367	444	367	444	871A,B,C
		GU ()		QC	2	5	2.50	19931897	20045206	19799530	19925277	872A,B,C
		I/O PCB SUBTOTAL			239	289	1.20	1743339		1675270		
		PSB TOTAL			239	289	1.20	1743339		1675270		
PROGSC2D	I/O PCB	ISRT ()			14	0	0.00	377	621	377	621	608A,B,C
		GU ()			14	36	2.57	22360408	52048566	22221852	51901313	634A,B,C
		I/O PCB SUBTOTAL			28	36	1.28	11180393		11111115		

アプリケーション・プログラム DL/I 呼び出しのための入出力のモニター

IMS モニター報告書は、モニター間隔中の入出力オカレンスの合計数と、実行されたアプリケーション・プログラムごとに入出力オカレンスにかかった合計時間を示します。プログラム入出力報告書は、モニター間隔中にアクティブだったすべての PSB についてこれら 2 つの合計を示し、プログラムが使用する各 PCB によって引き起こされた入出力待ち時間の詳細な内訳を含んでいます。

報告書の詳細は、アプリケーション・プログラム処理時に起きた競合について多くのことを明らかにします。競合の各タイプおよび起きた回数は、入出力 PCB ごとに記録されます。この報告書は、合計待ち時間、最大待ち時間、および平均時間を示します。1 つの PSB のもとにある PCB ごとに、そして各 PSB のもとにあるすべての PCB について小計が示されます。

DDN/FUNC 欄には、データ・セットが DD 名でリストされます。MODULE 欄では、競合のソースを示すためにコードを使用しています。競合のタイプおよびコードを次に示します。

- メッセージ処理

コード 競合

MFS MFS 形式ライブラリー・ディレクトリー

PMM メッセージ形式バッファー・プール・スペースまたは制御ブロック入出力

QMG メッセージ・キュー管理

- スケジューリング

コード 競合

BLR ACBLIB からのロード/読み取り

MSC MPP 領域の初期設定

SMN 仮想記憶管理

外部サブシステム呼び出しの場合には、処理の完了にかかる経過時間は待ち時間と見なされます。DDN/FUNC 欄では、以下に示す外部サブシステム呼び出し機能が示されます。

• 外部サブシステム

コード サブシステム呼び出し機能

- AB0** ABORT
- CT0** スレッドの作成
- D50** 識別またはスレッドの終了、サインオフ
- D80** INIT
- I30** 識別、コマンド、エコー、終了
- I30** 識別、サブシステムの終了
- I50** INIT
- I60** 疑わしいリソース解決
- PR0** サブシステム作動不能
- P10** コミット準備 (フェーズ 1)
- P20** コミット続行 (フェーズ 2)
- SO0** サインオン出口ルーチン
- SI0** 識別出口ルーチン

以下の例は、プログラム入出力報告書の例を示しています。

```

IMS MONITOR  ****PROGRAM I/O****      TRACE START 1993 022  14:00:18  TRACE STOP 1993 022  14:02:20 PAGE 0088
.....IWAIT TIME.....
PSBNAME PCB NAME      IWAIT   TOTAL      MEAN      MAXIMUM  DDN/FUNC  MODULE
-----
PROGHR1A I/O PCB      122    2341116    19189     70795    HOTELDBA  DBH
                34    24177936    711115    3950160    **W F I
                40    23652665    591316    2668917    **W F I
                5     67613     13522     21214     SHMSG      QMG
                4    110363    27590     60486     QBLKS      QMG

                PCB TOTAL
                131    2519092    19229

PSB TOTAL
PROGDE1A TRMNALDA    20     624677    31233     68252     TRMNALDA  VBH
                1     275811    275811    275811    PI TRMNALDA....

                PCB TOTAL
                21     900488    42880

                I/O PCB
                16     488812    30550     79980     TRMNALDA  VBH
                1     16118    16118     16118     SHMSG      QMG

                PCB TOTAL
                17     504930    29701

                TABLEDA
                16     290471    18154     33254     TABLEDA  DBH

                PCB TOTAL
                16     290471    18154

PSB TOTAL
PROGHR2B HOTELDBB    54    1695889    31405     184475    HOTELDBB  DBH
                8     698384    87298     184475    HOTELDBB  DBH
                4     5820650    1455162    1455278    PI HOSINDB....
                4     4481024    1120256    1209075    PI HOTELDBB....
                2     260817    130408    232750    HOSINDB   VBH
                7     106623    15231     16410    HOSINDB   VBH
                1     15366    15366     15366    HOTELDBD  DBH

```

```

PCB TOTAL
PSB TOTAL
PROGHR2A HOTELDBA
PCB TOTAL
I/O PCB
IMS MONITOR ****PROGRAM I/O**** TRACE START 1993 022 14:00:18 TRACE STOP 1993 022 14:02:20 PAGE 0089

```

PSBNAME	PCB NAME	IWAITS	TOTAL	MEAN	MAXIMUM	DDN/FUNC	MODULE
PROGHR2A	I/O PCB	5	4654544	930908	2020043	**W F I	
		8	32796604	4099575	9328891	**W F I	

```

PCB TOTAL
PSB TOTAL
PROGPS2A LOGIMA

```

```

PCB TOTAL
I/O PCB

```

```

PCB TOTAL
PSB TOTAL
PROGSC6C I/O PCB

```

```

PCB TOTAL
PSB TOTAL

```

```

PCB TOTAL
PSB TOTAL
PROGHR2D I/O PCB

```

```

PCB TOTAL
HOTELDBD
MONITOR ****PROGRAM I/O**** TRACE START 1993 022 14:00:18 TRACE STOP 1993 022 14:02:20 PAGE 0090

```

PSBNAME	PCB NAME	IWAITS	TOTAL	MEAN	MAXIMUM	DDN/FUNC	MODULE
PROGHR2D	HOTELDBD	31	2130585	68728	769130	HOSINDXD	VBH
		3	115999	38666	56394	HOTELDBA	DBH
		2	69833	34916	43470	HOTELDBC	DBH
		2	41430	20715	28020	HOSINDOD	VBH
		4	5515374	1378843	1458884	PI HOSINDXD....	
		4	3997017	999254	1026228	PI HOTELDBD....	

PCB TOTAL	142	18149345	127812
PSB TOTAL	192	21208696	110461

入出力 PCB の呼び出しのための入出力待ちは、PSB の最初の項目としてグループ化されています。DC DL/I 呼び出しの場合、入出力が行われたデータ・セットは DDN/FUNC 見出しの下に示され、モジュール・コードはどのようなタイプの競合が待機の原因になったかを示します。外部サブシステム呼び出しの場合は、機能が DDN/FUNC 見出しの下に示され、モジュール・コードにより、呼び出し項目の原因が示されます。

LGMSG および SHMSG 以外の名前が、入出力 PCB の DDN/FUNC 欄の中に現れることがあります。

プログラムが入力待ちとして指定されており、次のメッセージの入力を待たなければならない場合は、待機項目が DDN/FUNC の見出しの下に **WFI で示され、MODULE 欄には何の項目も示されません。次の入力メッセージを待つのにかかる時間は、待ち時間のもとに示されます。**WFI 項目は、参考情報として示されるのみで、それらの値は統計の計算には使用されません。

MFS アクティビティのモニター

メッセージ形式バッファ・プール報告書からは、メッセージ形式バッファ・プールの使用の管理に関して行われるすべてのアクティビティの要約を知ることができます。

この報告書を、以下の例に示します。このデータは、トレース間隔の開始時および終了時のカウントとそれらの差を示します。

メッセージのフォーマット設定が行われるときに、適切なメッセージ・ブロックがメッセージ形式バッファ・プール内に存在していなければなりません。そのブロックは、入力の場合には DIF/MID の対、出力の場合には DOF/MOD の対です。これらのブロックがバッファ内になければ、アクティブ IMSFORMATA/B ライブラリーへの入出力を行う必要があります。ブロック検索では、プール内にある索引を使用した先行ディレクトリー探索 (つまり、直接的な検索) も行われることがあります。

多数のカウント値により、内部イベント管理の詳細が明らかにされます。ブロックのディレクトリー項目がない回数は、余分なディレクトリー検索入出力を意味します。FRE 項目を利用できないために起こる遅延は、要求無視カウントとして記録されます。

IMS MONITOR BUFFER POOL STATISTICS TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0007

MESSAGE FORMAT BUFFER POOL

	5:55:15 START TRACE	5:59:49 END TRACE	DIFFERENCE
NUMBER OF P/F REQUESTS	0	0	0
NUMBER OF I/F REQUESTS	18	20	2
NUMBER OF I/F I/O'S	2	2	0
NUMBER OF TIMES POOL COMPRESS WOULD BE SUCCESSFUL	0	0	0
NUMBER OF DIRECTORY I/O OPERATIONS	2	2	0
NUMBER OF TIMES BLOCK WASHED FOR FRE	0	0	0
NUMBER OF TIMES P/F REQUEST IGNORED	0	0	0
NUMBER OF F/B REQUESTS	18	20	2

NUMBER OF TIMES F/B REQUEST IGNORED	0	0	0
NUMBER OF TIMES I/F ON F/B QUEUE	16	18	2
NUMBER OF TIMES I/F ON I/F QUEUE	0	0	0
NUMBER OF TIMES F/B ON I/F QUEUE	18	20	2
NUMBER OF TIMES P/F ON I/F QUEUE	0	0	0
NUMBER OF TIMES P/F ON F/B QUEUE	0	0	0
NUMBER OF TIMES THERE WAS NO DIR ENTR FOR A BLOCK	0	0	0
NUMBER OF TIMES I/O ERRORS POINT OR READ MACRO	0	0	0
NUMBER OF IMMEDIATE I/O REQUESTS WAITED DUE TO MAXIMUM I/O	0	0	0
NUMBER OF REQUESTS SATISFIED BY INDEX/DYNAMIC DIRECTORY	0	0	0

QUOTIENT : IMMEDIATE FETCH I/O'S + DIRECTORY I/O'S OPERATIONS = 0.00

TOTAL NUMBER OF TRANSACTIONS

メッセージ・キュー処理のモニター

トランザクション処理の効率に直接影響する重要なリソースは、メッセージ・キュー・プール、およびメッセージ・キューの入出力管理です。メッセージ・キュー・プール報告書を見れば、そのアクティビティーを調べることができます。

以下の例は、メッセージ・キュー・プール報告書の内容を示しています。トレース間隔の開始時と終了時のアクティビティーのカウンとともに、開始時の数と終了時の数の差を示しています。

IMS MONITOR BUFFER POOL STATISTICS TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0002

M E S S A G E Q U E U E P O O L

	5:55:15 START TRACE	5:59:49 END TRACE	DIFFERENCE
NUMBER OF LOCATE CALLS FROM QMGR	54204	68436	14232
NUMBER OF RECORD RELEASE CALLS FROM QMGR	16431	20738	4307
NUMBER OF LOCATE AND ALTER CALLS FROM QMGR	131593	164744	33151
NUMBER OF REQUESTS TO PURGE THE Q POOL	2	2	0
NUMBER OF ADDRESS TO DRRN TRANSLATION REQUESTS	21351	27076	5725
NUMBER OF REQUESTS TO WAIT FROM QMGR	0	0	0
NUMBER OF READ REQUESTS	962	962	0
NUMBER OF WRITE REQUESTS(TOTAL)	499	499	0
NUMBER OF WRITES DONE BY PURGE	499	499	0
NUMBER OF WAITS FOR PURGE COMPLETION	1	1	0
NUMBER OF WAITS BECAUSE NO BUFFER AVAILABLE	0	0	0
NUMBER OF WAITS FOR OTHER DECB TO READ THIS BUFFER	823	823	0
NUMBER OF WAITS FOR OTHER DECB TO WRITE THIS BUFFER	0	0	0
NUMBER OF WAITS FOR CONFLICTING END DEQ BUFFER REQ	0	0	0
NUMBER OF PSBS UNCHAINED FROM BUFFERS	0	0	0
NUMBER OF CALLS TO QMGR.(TOTAL)	48164	62213	14049
NUMBER OF CALLS TO REPOSITION A LOST BUFFER	0	0	0
NUMBER OF CALLS TO ENQ A MESSAGE	10583	13441	2858
NUMBER OF CALLS TO DEQ ONE OR MORE MESSAGE	6321	7767	1446
NUMBER OF CALLS TO CANCEL INPUT OR OUTPUT	119	121	2

QUOTIENT : TOTAL NUMBER OF OSAM READS + OSAM WRITES + ALL IWAITS = 0.00

TOTAL NUMBER OF TRANSACTIONS

チェックポイントの影響の検出

チェックポイント・コマンドで SNAPQ を指定すると、すべてのメッセージ・キューの現在状況がシステム・ログに書き込まれます。これは、キュー管理のためにメッセージ処理を妨げてしまいます。

General Iwait Time Events (一般的な IWAIT 時間イベント) 報告書では、SNAPQ によって起こる待ち時間を記録します。以下の例では、要約行 QMGR SNAPQ CHECK にアクティビティーを示しています。オカレンスの数が、合計待ち時間、平均待ち時間、および最大待ち時間と一緒に示されています。

EVENT IWAITS	OCCURRENCESIWAIT TIME.....			DISTRIBUTION NUMBER
		TOTAL	MEAN	MAXIMUM	
QMgr SNAPQ CHECK	0	0	0	0	0

REGION AND JOBNAME REPORT

REG. NO.	JOB NAME
1	MPRI1A100
2	MPRI1A209
3	MPRI1A210
4	MPRI1A211
5	MPRI1A103
6	MPRI1A101
7	MPRI1A115
8	MPRI1A116
9	MPRI1A216
10	MPRI1A200
11	MPRI1A217
12	MPRI1A119
13	MPRI1A218
14	MPRI1A219
15	MPRI1A104
16	MPRI1A220
17	MPRI1A203
18	MPRI1A123
19	MPRI1A222
20	MPRI1A105
21	MPRI1A124
22	MPRI1A223
23	MPRI1A107
24	MPRI1A224
25	MPRI1A106
26	MPRI1A206
27	MPRI1A205
28	MPRI1A108
29	MPRI1A109
30	MPRI1A208
31	MPRI1A111
32	MPRI1A112
33	MPRI1A113
34	MPRI1A204
35	MPRI1A114
36	MPRI1A102
48	MPRI1A121
49	MPRI1A122
50	MPRI1A221

トランザクション・キューイング報告書

メッセージ処理の効率をモニターするだけでなく、トランザクション処理プログラムのスケジューリングのつど、トランザクション・キューのサイズを調べることで、各アプリケーションごとに、提供されたサービスをモニターすることができます。

以下の例に示されているトランザクション・キューイング報告書は、トランザクションごとの、スケジューリング時の最小カウント、平均カウント、および最大カウントを記録したものです。モニター間隔中にデキューされたトランザクション（または完全に処理されたトランザクション）の合計数が、トランザクション・コードごとに示されています。スケジューリングごとの処理されたトランザクションの平均数は、DEQUEUED MEAN 欄に示してあります。

TRANSACTION	NUMBER DEQUED	NUMBER SCHEDS.	(B)			(A)		DISTRIBUTION NUMBER
			..ON QUEUE MINIMUM	WHEN MEAN	SCHEDULED..... MAXIMUM	DEQUED MEAN		
SC6X	13	1	0	0.00	0	13.00	883A,B	
IT8W	17	3	0	0.00	0	5.66	887A,B	

TS1Z	16	1	0	0.00	0	16.00	891A,B
PS3X	23	1	0	0.00	0	23.00	895A,B
PS3Y	17	7	0	0.00	0	2.42	899A,B
IT1V	11	6	0	0.00	0	1.83	903A,B
SC2Z	143	2	0	0.00	0	71.50	907A,B
IT8U	12	7	0	0.00	0	1.71	911A,B
PS2W	10	1	0	0.00	0	10.00	915A,B
TS1U	12	4	0	0.00	0	3.00	919A,B
PS3W	14	1	0	0.00	0	14.00	923A,B
IT8Y	16	5	0	0.00	0	3.20	927A,B
IT1W	9	1	0	0.00	0	9.00	929A,B
SC2W	9	1	0	0.00	0	9.00	933A,B
IT2V	13	5	0	0.00	0	2.60	937A,B
IT2W	17	6	0	0.00	0	2.83	941A,B
TS1V	9	1	0	0.00	0	9.00	945A,B
SC2V	12	5	0	0.00	0	2.40	947A,B
TS1W	11	1	0	0.00	0	11.00	949A,B
PS3V	13	3	0	0.00	0	4.33	953A,B
IT1U	9	6	0	0.00	0	1.50	957A,B
SC4U	11	5	0	0.00	0	2.20	962A,B
SC6W	10	1	0	0.00	0	10.00	966A,B
PS2V	8	6	0	0.00	0	1.33	970A,B
IT8X	12	1	0	0.00	0	12.00	974A,B
SC4W	10	1	0	0.00	0	10.00	978A,B
SC6U	14	6	0	0.00	0	2.33	982A,B
IT2Y	9	3	0	0.00	0	3.00	986A,B
SC2X	15	1	0	0.00	0	15.00	990A,B
PS2Y	17	2	0	0.00	0	8.50	994A,B
SC4Y	152	4	0	0.50	1	38.00	998A,B
SC2Y	106	2	0	0.00	0	53.00	1000A,B
IT2X	20	1	0	0.00	0	20.00	1004A,B
SC2U	15	3	0	0.00	0	5.00	1008A,B
SC4Z	123	5	0	0.60	1	24.60	1010A,B
TS1X	15	1	0	0.00	0	15.00	1015A,B
SC4X	19	1	0	0.00	0	19.00	1019A,B
PS2X	13	1	0	0.00	0	13.00	1024A,B
PS2Z	20	5	0	0.00	0	4.00	1028A,B
SC6Z	130	1	0	0.00	0	130.00	1031A,B
SC6V	10	4	0	0.00	0	2.50	1035A,B
SC6Y	143	1	0	0.00	0	143.00	1037A,B
IT1X	10	1	0	0.00	0	10.00	1040A,B
PS3U	19	6	0	0.00	0	3.16	1131A,B
IT2U	13	4	0	0.00	0	3.25	1146A,B

回線アクティビティのモニター

モニター対象の間隔中にメッセージ・トラフィックを処理するノードごとに、アクティビティのすべてのオカレンスの要約を入手できます。経過時間と待ちなし時間については、通信要約報告書の中で通信回線ごとに、合計時間、平均時間、および最大時間のカテゴリーで示してあります。

以下の例は、通信要約報告書の例を示しています。

要件: どの物理装置が回線を使用中であるかを、システム定義のステージ 1 の出力と突き合わせる必要があります。回線番号は、ステージ 1 の入力デッキ内の物理的なオカレンスに従って順次に割り当てられます。

オンライン・システムで、制御領域の JCL に MFS ブロックの事前取り出しオプションを指定している場合は、報告書の最後の行に、すべての事前取り出しイベントの統計が表示されます。

NODE OR LINE NUMBER	OCCURRENCESELAPSED TIME..... (A)			NOT IWAIT TIME(ELAPSED-IWAIT) (B)			DISTRIBUTION NUMBER
		TOTAL	MEAN	MAXIMUM	TOTAL	MEAN	MAXIMUM	
PMT01A	3	2396	798	1547	2396	798	1547	1467A,B
19	182	92155	506	1106	92155	506	1106	1493A,B

2	59	2280	38	41	2280	38	41	1515A,B
TOTAL	244	96831	396		96831	396		

さらに、ノード用に伝送されたデータの量を、回線機能報告書で調べることができます。以下の例は、回線機能報告書の例を示しています。この報告書では、入力データと出力データを区別しています。データ・ブロックの数およびブロックの平均サイズと最大サイズが、IMS が受信したデータと送信したデータのそれぞれについて記録されています。

この報告書は、回線がどのくらい非アクティブであるかを示す尺度も含んでいます。非アクティブ間隔は、最後の入力ブロックの受信の終了時刻と出力伝送の開始時刻の差と見なされます。これらの非アクティブ状態のオカレンスは、送受反転間隔と呼ばれており、この報告書では、これらの間隔に関連する平均時間と最大時間、ならびにオカレンスの回数もまとめて載せています。

回線を使用しているのが MFS サポート端末である場合は、複数ページ・メッセージの次のページに対する要求数のカウントも記録されます。

複数システムを結合したものについてのリンク・トラフィックが記録される場合は、3 つの報告書が回線機能報告書の後に続きます。

```

IMS MONITOR ****LINE FUNCTIONS*** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0091
(A).....(B).....
MEAN MAX. MEAN MAX.
RECEIVE RECEIVE RECEIVE TRANS. TRANS. TRANS. DIST. TURN
LINE NUMBER DEVICE BLOCKS BLKSIZE BLKSIZE BLOCKS BLKSIZE BLKSIZE NUMBER AROUND
PMT01A 3270V 1 29 29 2 170 171 1468A,B 3 798 1547 1466 0
19 XXXX 182 506 1106 1492 0
2 LOC SYS 59 38 41 1514 0
ALL LINES
1 29 2 170 244 396 0

```

関連概念:

899 ページの『MSC の IMS モニター報告書』

メッセージ処理効率のモニター

IMS モニターは、IMS 制御領域での非同期処理に関する要約と詳細情報の両方を作成します。端末からのデータを受信すると、処理が開始されます。

アプリケーション・プログラムの応答によっても処理が行われます。通信トラフィック全体について、4 つの主要なバッファ・プールとフォーマット設定のためのアクセスのスペース (SPA) およびメッセージ・キュー・データ・セットが管理されます。プール・スペースまたは入出力に関する競合のために、回線アクティビティによって起動された通信タスクの処理が中断されたときに、待ち時間が記録されます。この情報は、通信待機報告書に含まれます。以下の例は、この報告書を示しています。

この報告書は、一連の通信処理タスクの ID として回線番号を使用し、通信要約報告書を補足するものです。

```

IMS MONITOR ****COMMUNICATION IWAIT***** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0090
.....IWAIT TIME.....
LINE NUMBER OCCURRENCES TOTAL MEAN MAXIMUM FUNCTION BLKSIZE MODULE DIST.
ALL LINES...
PREFETCH I/O
NONE

```

IMS 内部リソースの使用

リソースに関する内部競合のレベルを調べるときに使用できるいくつかの要約報告書があります。

以下に挙げるのは、それらの報告書についての簡単な説明です。

プール・スペース障害要約報告書

プール・スペース障害要約報告書では、各領域において一定量のストレージを利用できなかった回数を示します。これは、ストレージの獲得に失敗したオカレンスの回数のほかに、バイト数とプール ID を示します。この要約を使用すれば、システム定義のオーバーライドまたは JCL 内の EXEC ステートメント内のバッファ数の変更により、バッファ・プール割り振りを増やす必要があるかどうかを判別できます。

この報告書の形式は、以下の例のようになります。

POOL SPACE FAILURE SUMMARY

	POOL ID	BYTES REQ.	OCCURRENCES
	DLMP	8888	1
TOTAL			1

ラッチ競合統計報告書

IMS におけるタスク処理の基本的なシリアライゼーションは、IMS ラッチの所有権によって制御されます。各種のプログラムが実行中であるときには、それらは所有権を得るために競合します。それらがリソースを待つ場合は、ラッチを所有しているプログラムは、リソースを待っている他方の ITASK に通知する必要があります。リソースに関する競合レベルを判定して、その競合状態を和らげるための一連の変更を検討することができます。

ラッチ競合統計報告書には、競合のレベルを表す種々のタイプのラッチとカウンターが示されます。以下の図は、この報告書の例です。項目は、ラッチ名にしたがって編成されています。

モニターがアクティブの間にシステム・チェックポイントがとられた場合は、ラッチ競合統計はゼロにリセットされるため、この報告書に示された値は無効です。この状態のときは、以下のメッセージが報告書の先頭部分に挿入されます。

```
**** A CHECKPOINT OCCURRED DURING MONITOR RUN ****
**** LATCH CONFLICT STATISTICS ARE INVALID ****
**** SEE UTILITIES REFERENCE MANUAL ****
```

しかし、マスター端末オペレーターが、STATISTICS キーワード・パラメーターを指定して /CHECKPOINT コマンドを出す場合は、ラッチ競合統計はゼロにリセットされますが、このことは、IMS モニターには通知されません。したがって、統計が無効になっていることが DFSUTR20 には検出できないので、このメッセージが発行されることはありません。

推奨事項: モニターを実行中は、統計チェックポイントを出さないことをお勧めします。

カウンターは、主にストレージ管理とサービスのロギングに関係したものです。記録される統計は、競合が起きた回数、つまりリソースがラッチを待った回数です。

LATCH CONFLICT STATISTICS

LATCH NAMES	COUNT FIELD	AT START	AT END	DIFF.
LOGL	CONTENTIONS	0	0	0
SMGT	CONTENTIONS	0	0	0
XCNQ	CONTENTIONS	0	0	0
ACTL	CONTENTIONS	0	0	0
CBTS	CONTENTIONS	0	0	0
DBLK	CONTENTIONS	0	0	0

IMS モニター出力の度数分布の使用方法

IMS モニター・データ・レコードから得られる報告書には、平均時間が示されている数多くの要約行が含まれています。単なる平均時間や最大時間ではなく、時間測定されるイベントの分布を知りたい場合には、ある範囲の間隔全体にわたる度数分布の形で、イベントを個別に記録するよう報告書印刷ユーティリティーに要求することができます。

トランザクション・キュー・ロードや伝送ブロック・サイズなどの一部の分布は、時間依存ではありません。

以下の各表には、主要な IMS モニター報告書と各報告書ごとにそれぞれ生成される度数分布のタイプが示してあります。各タイプは、報告書内の各セクションにどれだけ項目があるかに応じて、いくつかの分布に分かれます。度数分布のタイプごとに、適当な間隔または範囲でデータが累積されます。各タイプごとに使用される範囲の集合には、ID 列に示されているように ID が与えられます。

- 以下の表には、領域要約別にソートされた報告書分布が示してあります。

表 78. 領域要約別報告書分布

報告書名	ID	説明
スケジューリングと終了	D1	経過時間
	D2	待ちなし時間
スケジュールの終わりから最初の DL/I 呼び出しまで	D3	N/A
DL/I 呼び出しの経過実行時間	D4	N/A
	D5	経過時間
外部サブシステム呼び出し	D6	待ちなし時間
DL/I 呼び出し当たりの待ち	D43	経過時間
意図チェックポイントのためのアイドル	D7	N/A
	D8	N/A
	D20	経過時間
	D21	待ちなし時間

- 以下の表には、プログラム領域別にソートされた報告書分布が示してあります。

表 79. プログラム領域別報告書分布

報告書名	ID	説明
実行経過時間	D30	N/A
スケジュールの終わりから最初の DL/I 呼び出しまで	D31	N/A

- 以下の表には、プログラム要約別にソートされた報告書分布が示してあります。

表 80. プログラム要約別報告書分布

報告書名	ID	説明
スケジュール当たりのプロセッサ時間	D15	N/A
スケジュール当たりの、デキューされたトランザクション数	D14	N/A
スケジュール当たりの経過時間	D9	N/A
スケジュールの終わりから最初の DL/I 呼び出しまで	D10	N/A

- 以下の表には、通信要約別にソートされた報告書分布が示してあります。

表 81. 通信要約別報告書分布

報告書名	ID	説明
回線経過時間	D18	N/A
回線 NOT WAIT 時間	D19	N/A

- 表 82 には、回線機能別にソートされた報告書分布が示してあります。

表 82. 回線機能別報告書分布

報告書名	ID	説明
受信ブロックの長さ	D36	N/A
伝送ブロックの長さ	D37	N/A
非アクティブ間隔	D38	N/A

- 以下の表には、MSC キューイング要約別にソートされた報告書分布が示してあります。

表 83. MSC キューイング要約別報告書分布

報告書名	ID	説明
キューに入っていた時間	D39	N/A

- 以下の表には、トランザクション・キューイング別にソートされた報告書分布が示してあります。

表 84. トランザクション・キューイング別報告書分布

報告書名	ID	説明
スケジュール時にキューに入っているトランザクション数	D17	N/A

表 84. トランザクション・キューイング別報告書分布 (続き)

報告書名	ID	説明
スケジュール当たりの、デキューされたトランザクション数	D16	N/A
事前取り出し形式ブロック	D28	経過時間
	D29	待ちなし時間

- 以下の表には、呼び出し要約別にソートされた報告書分布が示してあります。

表 85. 呼び出し要約キューイング別報告書分布

報告書名	ID	説明
DL/I 呼び出し当たりの PSB 待ち数	D13	N/A
外部サブシステム呼び出し当たりの PSB 待ち数	D44	N/A
呼び出し当たりの PSB 経過時間	D11	N/A
呼び出し当たりの PSB 待ちなし時間	D12	N/A
PSB 外部サブシステム呼び出し	D45	経過時間

- 以下の表に、バッファ・プール統計から得られる、待ち時間に関するいくつかの分布を示します。

表 86. 待ち時間分布

機能	ID	モジュール・キー
ストレージ	D22	SMN
スケジューラー内部	D25	MSC
キュー・マネージャー入出力	D26	QMG
ブロック・ローダー入出力	D27	BLR
MFS ブロック入出力	D32	MFS
MFS ディレクトリー入出力	D33	MFS
形式バッファ・プール・スペース	D35	PMM
QMGR SNAPQ 検査	D42	ありません。

度数分布出力の入手方法

IMS モニター報告書印刷ユーティリティに分布データを収集するよう要求するときは、DIS 入力制御ステートメントを組み込んでください。こうすると、MEAN の表題を持つ欄に項目を持つすべての報告書項目が、対応する度数分布を分布付録の一部として持つこととなります。各報告書行には、Distribution Number という表題の欄の下に識別用の参照番号が含まれています。この参照番号を使用すれば、分布付録内でその番号の付けられた分布データを見つけることができます。

度数分布範囲の定義方法

要約行ごとに 10 個の間隔が定義され、各間隔の範囲に入るオカレンスが累積されます。間隔の範囲は、デフォルトの終点で事前設定されます。例えば、DL/I 呼び出し経過時間についての終点は、0、1000、2000、4000、8000、16000、32000、64000、128000、256000、INF (時間はす

べてミリ秒単位) です。デフォルトの終点は、イベントに合うように選択されます。最初の間隔の下限は常にデフォルトとしてゼロが使用され、10 番目の間隔の上限は無限大 (INF) です。

複数のタイプの分布で同じ終点セットを使用できますが、各タイプに 1 つの分布 ID が割り当てられます。この ID を使用して、終点を再定義できます。デフォルトの終点をオーバーライドするときには、入力制御ステートメントを報告書印刷ユーティリティに含めます。そのステートメントで、分布 ID のタイプを指定し、希望する終点値を与えます。

DL/I 呼び出し経過時間の終点は、次のように再指定できます。

```
D5 0,500,1000,1500,2000,4000,,100000,500000
```

これで指定されていない終点の値は、デフォルトの 32000 と 64000 のままであり、最後の値もそのまま (INF) です。

以下の例に、分布付録報告書のサンプル・ページを示します。これは、分布のタイプによって範囲が変化する例です。行はペアになっており、2 番目の行に累積カウントが記録されます。

```
IMS MONITOR ****DISTRIBUTION APPENDIX**** TRACE START 1993 130 5:55:15 TRACE STOP 1993 130 5:59:49 PAGE 0200
# 1.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 2.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 3.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 4.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 5.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 6.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 7.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 8.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 9.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 10.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 11.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 12.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 13.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 14.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 15.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
# 16.....0.....200000.....400000.....600000.....800000.....1000000.....1200000.....1400000.....1600000.....1800000.....INF
```

分布のデフォルト値の定義

度数分布表および待ち時間分布表にある ID を使用すれば、次のリストで探すことによって、分布のデフォルト・エンドポイントを判別できます。

D1、D2、D5、D6、D9、D10、D11、D12、D15、D18、D19、D20、D21、D22、D25、D27、D28、D29、D30、D31、D43、および D45
 0、1000、2000、4000、8000、16000、32000、64000、128000、256000、INF

- D3** 0、50000、100000、150000、200000、250000、300000、350000、400000、450000、INF
- D4** 0、200000、400000、600000、800000、1000000、1200000、1400000、1600000、1800000、INF
- D7、D13、D44**
0、0、1、2、3、4、5、6、7、8、INF
- D8** 0、100000、200000、300000、400000、500000、600000、700000、800000、900000、INF
- D14、D16、D17**
0、1、2、3、4、5、10、15、30、90、INF
- D23、D24、D26、D32、D40、D42**
0、2000、8000、24000、50000、100000、150000、200000、250000、300000、INF
- D33、D34、D35**
0、2000、4000、8000、16000、32000、64000、96000、128000、160000、INF
- D36、D37**
0、10、20、40、80、100、200、400、800、1000、INF
- D38** 0、1000、10000、100000、200000、500000、800000、1000000、1500000、2000000、INF
- D39** 0、1000、5000、10000、50000、100000、500000、1000000、5000000、10000000、INF

分布付録出力の解釈

異常なイベントの組み合わせが報告書要約行で報告されたと思われるときには、分布付録の明細出力を使用できます。通常、リソース使用の問題を強調するには、平均時間と最大時間またはカウントのみで十分です。

しかし、平均値では異常な分布が覆い隠されていて分からないと思われる場合には、IMS モニター出力レコードに含まれている詳細を利用できます。

例えば、特定のトランザクションのスケジューリング・アルゴリズムでの変化を調べているときに、アプリケーション・プログラムのスケジューリングごとにどれだけのトランザクションが処理可能であったかを知る必要があります。以下の図は、処理されたトランザクションを次のようにヒストグラムで示します。

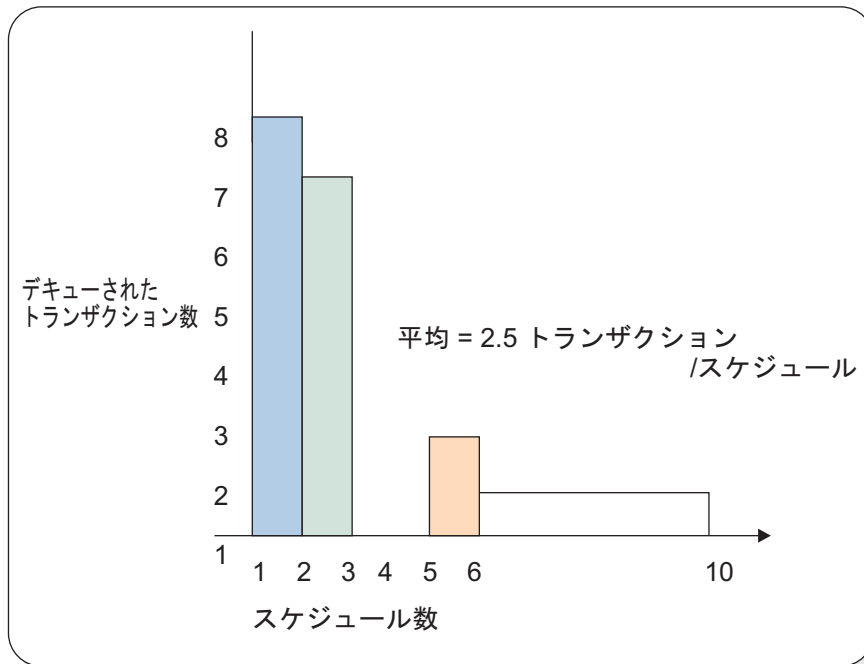


図 94. アプリケーション・プログラムのスケジューリングごとに処理されたトランザクションの数

平均は 1 スケジュール当たり 2.5 トランザクションです。図の分布を見れば、多くのスケジュールで 1 つまたは 2 つのトランザクションしか処理できなかったこと、およびわずかなスケジュールでキューがほとんどいっぱいになったことがわかります。ヒストグラムの分布データは次のとおりです。

Number of schedules	1	2	3	4	5	6-10	>10
Transactions dequeued	8	7	0	0	2	1	0

分布付録では、このヒストグラムのデータが 2 行形式で表示されます。

- 最初の行には間隔が表示され、その接頭部には、それよりも前の出力の各行への相互参照が付いています。
- 2 番目の行には、これらの間隔内で発生したイベントの数が表示されます。

このデータは、次のように表示されます。

```
# 950B...0...1...2...3...4...5...10...15...30...90...INF
      8  7  0  0  2  1  0  0  0  0
```

相互参照の 950B は、固有の報告書行を指し示しています。例えば、トランザクション・キューイング報告書では、注目するトランザクションについての該当行で DISTRIBUTION NUMBER という見出しの欄に 950A,B を表示しています。分布付録内でそのデータを見つけるためには、参照番号 950B を使用してください。参照番号 950A は、スケジュール時におけるキュー内のトランザクション数についてのデータを指し示します。

第 50 章 //DFSSTAT 報告書

//DFSSTAT 報告書には、アプリケーション・プログラムによって発行された DB 呼び出しと DC 呼び出しの数が示され、アプリケーション実行中のバッファリング・アクティビティーが表示されます。この報告書は、アプリケーション・プログラムが終了するときに作成されます。

報告書を手入するときは、バッチ領域またはオンライン従属領域の JCL に //DFSSTAT DD ステートメントを指定する必要があります。以下に、//DFSSTAT DD ステートメントの例を示します。

```
//DFSSTAT DD SYSOUT=A
```

推奨事項: サポートされてはいますが、MPP 領域用の JCL には //DFSSTAT DD ステートメントを入れないことをお勧めします。//DFSSTAT DD ステートメントを入れなければ、短時間実行 MPP が終了するたびに一連の報告書が作成されることによるオーバーヘッドと大量の出力を避けることができます。

以下のトピックでは、//DFSSTAT で作成される報告書について説明します。

関連概念:

480 ページの『詳細モニター用ツール』

PST 会計報告書

この報告書には、アプリケーション・プログラムが出した DB 呼び出しと DC 呼び出しの個数が表示されます。

PST 会計報告書のフィールド

以下の例は、PST 会計報告書の例で、各フィールドの名前を示しています。この報告書内の各フィールドは、1 つのタイプの DB 呼び出しまたは DC 呼び出しを表します。例えば、DB GU CALLS フィールドは、アプリケーションが出したデータベース単独読み取り呼び出しの個数を示します。

```
*** PST ACCOUNTING STATISTICS ***
DB GU CALLS                2
DB GN CALLS                2
DB GNP CALLS              1
DB GHU CALLS              1
DB GHN CALLS              1
DB GHNP CALLS            1
DB ISRT CALLS            2
DB DLET CALLS            1
DB REPL CALLS            1
DB CALLS (TOTAL)         12
DB DEQ CALLS              1
DB RLSE CALLS             0
MSG GU CALLS              2
MSG GN CALLS              2
MSG CHNG CALLS           4
MSG ISRT CALLS           8
MSG PURGE CALLS          4
MSG CMD CALLS             1
```

MSG GCMD CALLS	1
MSG AUTH CALLS	1
MSG SETO CALLS	4
SYS APSB CALLS	0
SYS DPSB CALLS	0
SYS GMSG CALLS	2
SYS ICMD CALLS	1
SYS RCMD CALLS	2
SYS CHKP CALLS	0
SYS XRST CALLS	0
SYS ROLB CALLS	1
SYS ROLS CALLS	2
SYS SETS CALLS	1
SYS SETU CALLS	1
SYS INIT CALLS	1
SYS INQY CALLS	3
SYS LOG CALLS	1

VSAM バッファース・プール報告書 (バッチ領域の場合のみ)

VSAM バッファース・プール報告書では、アプリケーション・プログラムの実行時の VSAM バッファース・プールのアクティビティーについて記述します。VSAM サブプールごとに別個の報告書が作成されます。最後の VSAM バッファース・プール報告書では、アプリケーションが使用したすべての VSAM サブプールでのバッファリングのアクティビティーが要約されます。

VSAM バッファース・プール報告書の使用方法

この報告書は、バッチ領域で実行したアプリケーションについてのみ作成されま
す。

VSAM バッファース・プール報告書 (以下に例を示してある) は、基本的には、各
VSAM サブプール内で実行された入出力操作の回数を見るために使用します。

```

*** VSAM BUFFER POOL STATISTICS ***
FIX INDEX/BLOCK/DATA                N/Y/N
SHARED RESOURCE POOL ID              VPL1
SHARED RESOURCE POOL TYPE            D
SUBPOOL BUFFER SIZE                  4,096
TOTAL BUFFERS IN SUBPOOL             1,000
TOTAL HIPERSPACE BUFFERS IN SUBPOOL  50

RETRIEVE BY RBA CALLS                370
RETRIEVE BY KEY CALLS                187583
LOGICAL RECORDS INSERTED INTO ESDS   310
LOGICAL RECORDS INSERTED INTO KSDS   9823
LOGICAL RECORDS ALTERED IN THIS SUBPOOL 0
TIMES BACKGROUND WRITE FUNCTION INVOKED 0
SYNCHRONIZATION CALLS RECEIVED      29923
PERM WRT ERROR BUFFS NOW IN THE SUBPOOL 0
LARGEST NBR OF PERM ERR BUFFS EVEN IN THE SUBPL 0
VSAM GET CALLS ISSUED                0
VSAM SCHBFR CALLS ISSUED             189290
CONTROL INTERVAL REQUESTED ALREADY IN POOL 0
*CONTROL INTERVAL READ FROM EXTERNAL STORAGE 51238
*VSAM WRITES INITIATED BY IMS        138637
*VSAM WRITES TO MAKE SPACE IN THE POOL 9288
VSAM READS FROM HIPERSPACE BUFFERS   0
VSAM WRITES FROM HIPERSPACE BUFFERS  0
FAILED VSAM READS FROM HIPERSPACE BUFFERS 0

```

VSAM バッファース・プール報告書のフィールド

VSAM バッファース・プール報告書は、DB モニターが作成する VSAM バッファース・プール報告書と同じですが、以下の点が異なります。

- DB モニターの VSAM バッファース・プール報告書の各フィールド名の前には「NUMBER OF」が付いていますが、両方の報告書のフィールドの意味は同じです。
- //DFSSTAT VSAM バッファース・プール報告書は、トレース開始時刻およびトレース終了時刻を記録しません。//DFSSTAT 報告書の情報は、常にアプリケーションの実行の開始から終了まで収集されるので、そのような記録は不要です。
- //DFSSTAT VSAM バッファース・プール報告書には、「TOTAL I/O OPERATIONS」フィールドが入っており、ここには以下のものの合計が表示されます。
 - データベースからバッファースに CI が読み取られた回数 (報告書の CONTROL INTERVAL READ FROM EXTERNAL STORAGE フィールド)
 - バッファースがデータベースに書き出された回数 (報告書の VSAM WRITES INITIATED BY IMS フィールド)
 - 新しい CI をバッファースに読み込めるように、バッファースをデータベースに書き出した回数 (報告書の VSAM WRITES TO MAKE SPACE IN THE POOL フィールド)
 - //DFSSTAT VSAM バッファース・プール報告書は、要約報告書を含んでいません。要約報告書の前には、SUBPOOL BUFFER SIZE=ALL がきます。要約報告書は、すべての VSAM バッファース・プール報告書についての読み取りおよび書き込み情報の要約です。

これらのフィールドは入出力操作を表すもので、左側にアスタリスク (*) を付けて強調してあります。

関連資料: 報告書内の各種フィールドについては、839 ページの『VSAM バッファース・プール報告書』を参照してください。

OSAM バッファース・プール報告書 (バッチ領域の場合のみ)

OSAM バッファース・プール報告書では、アプリケーションの実行時の OSAM バッファース・プールのアクティビティーを記述します。この報告書は、バッチ領域で実行されたプログラムについてのみ作成されます。

OSAM バッファース・プール報告書の使用方法

OSAM バッファース・プール報告書 (以下に例を示してある) は、基本的には、実行された OSAM 入出力操作の回数を見るために使用します。ただし、この報告書は、順次バッファリング (SB) 関連情報を示しません。

```

*** OSAM DATA BASE BUFFER POOL STATISTICS ***
FIX BLOCK DATA                                Y/Y
SUBPOOL ID                                    004K
SUBPOOL BUFFER SIZE                           4096

```

TOTAL BUFFERS IN SUBPOOL	1000
LOCATE-TYPE CALLS	1765296
REQUESTS TO CREATE NEW BLOCKS	0
BUFFER ALTER CALLS	340800
PURGE CALLS	39371
LOCATE-TYPE CALLS, DATA ALREADY IN SUBPOOL	1370897
BUFFERS SEARCHED BY ALL LOCATE-TYPE CALLS	1987604
*READ I/O REQUESTS	375355
*SINGLE BLOCK WRITES BY BUFFER STEAL RTN	0
*BLOCKS WRITTEN BY PURGE	150284
TOTAL NBR OF I/O ERRORS FOR THIS SUBPOOL	0
BUFFERS LOCKED DUE TO WRITE ERRORS	0
LOCATE CALLS WAITED DUE TO BUSY ID	1431
LOCATE CALLS WAITED DUE TO BUFR BUSY WRITE	0
LOCATE CALLS WAITED DUE TO BUFR BUSY READ	0
BUFR STEAL/PURGE WAITED FOR OWNERSHIP RLSE	296
BUFFER STEAL REQUEST WAITED FOR BUFFERS	0
*TOTAL I/O OPERATIONS	525639

OSAM バッファース・プール報告書のフィールド

図は、OSAM バッファース・プール報告書の例です。

OSAM バッファース・プール報告書は、次の点を例外として、DB モニターで作成されるデータベース・バッファース・プール報告書と同一です。

- 両方の報告書のフィールドの意味は同じですが、IMS モニターのデータベース・バッファース・プール報告書の各フィールド名の前には「NUMBER OF」が付いています。
- //DFSSTAT OSAM バッファース・プール報告書は、トレース開始時刻およびトレース終了時刻を記録しません。//DFSSTAT 報告書の情報は、常にアプリケーションの実行の開始から終了まで収集されるので、そのような記録は不要です。
- さらに、//DFSSTAT OSAM バッファース・プール報告書には、「TOTAL I/O OPERATIONS」フィールドが入っており、これは、以下のフィールドを合計したものと同じです。
 - READ REQUESTS ISSUED
 - OSAM WRITES ISSUED
 - QUEUED WRITES ISSUED
 - FORMAT LOGICAL CYLINDER REQUESTS
 - BISAM READS OR QISAM SETLS

これらのフィールドは入出力操作を表すもので、左側にアスタリスク (*) を付けて強調してあります。

順次バッファリング要約報告書

順次バッファリング要約報告書では、アプリケーションに関する SB 関連情報の概要が示されます。(VSAM 関連情報は含まれません。)

順次バッファリング要約報告書の使用方法

SB 要約報告書 (以下に例を示してある) によって、SB の使用による利点がアプリケーションにあったかどうかを判別できます。この報告書を使用する際には、特に次のフィールドに注意を払ってください。

- NBR BLOCKS READ SEQUENTIALLY と PCT OF TOTAL
- PERCENT READ PER SEARCH REQUEST

*** SEQUENTIAL BUFFERING SUMMARY FOR THE APPLICATION ***

DFSSBUX0 DISALLOWED USAGE OF SB:	NO	
DFSSBUX0 REQUESTED CONDITIONAL SB ACTIVATION:	NO	
AT LEAST ONE SB= KEYWORD IN PSB:	YES	
AT LEAST ONE SBPARM CONTROL STMT FOR APPLICATION:	NO	
SBPARM CONTROL CARD(S) READ FROM //DFSCTL:	YES	
AT LEAST ONE SBPARM PSB= SPECIFIED THAT MATCHED PSB:	YES	
AT LEAST ONE SBPARM DB= SPECIFIED THAT MATCHED DB:	YES	
AT LEAST ONE SBPARM PCB= SPECIFIED THAT MATCHED PCB:	NO	
AT LEAST ONE SBPARM DD= SPECIFIED THAT MATCHED DD:	NO	
NUMBER OF SEARCH REQUESTS ISSUED BY OSAM BH:		
SEARCH	2,213	
NUMBER OF READ I/O:		
RANDOM READ	686	
SEQUENTIAL READ	652	
NUMBER OF BLOCKS READ:		
TOTAL NUMBER BLOCKS READ	7,206	
NBR BLOCKS READ AT RANDOM	686	PCT OF TOTAL:
NBR BLOCKS READ SEQUENTIALLY	6,520	PCT OF TOTAL: 9
PERCENT READ PER SEARCH REQUEST	60.46	
NUMBER OF SEQUENTIAL I/O ERRORS	0	

順次バッファリング要約報告書のフィールド

以下の例は、順次バッファリング要約報告書を示しています。

この報告書の最初の部分には、順次バッファリングが使用された理由または使用されなかった理由が示されます。報告書のこの部分には、以下のことが記載されています。

- DFSVSMxx 内に SBONLINE 制御カードが入っていたかどうか (これは、IMS DC 環境にのみ適用されます)。
- アプリケーション・プログラムの開始時に、/STOP SB コマンドが有効であったかどうか (これは、IMS DC 環境にのみ適用されます)。
- SB 初期設定出口ルーチン (DFSSBUX0) が、SB の使用を許可しなかったかどうか。
- SB 初期設定出口ルーチン (DFSSBUX0) が、デフォルトで SB の条件付き活動化を要求したかどうか。
- PSBGEN 時に少なくとも 1 つは SB= キーワードがあったかどうか。
- アプリケーション・プログラムに適用された SBPARM 制御ステートメントが、//DFSCTL ファイル内に少なくとも 1 つ入っていたかどうか。
- SBPARM 制御カードが読み取られたかどうか。答えが Yes であれば、続く統計では、どんな SBPARM キーワードが使用されたかが示されます。これは、アプリケーション・プログラムに順次バッファリングが使用された理由または使用されなかった理由を判別するのに役立ちます。

- SBPARM 制御カードに PSB= キーワードが少なくとも 1 つ指定され、しかもそれが、アプリケーションで使用された PSB と一致しているかどうか。
- PSB が一致している SBPARM 制御カードに DB= キーワードが少なくとも 1 つ指定されているか、または指定されていないか、そしてデータベースがアプリケーションで使用されたものと一致しているかどうか。
- PSB および DB が一致している SBPARM 制御カードに PCB= キーワードが少なくとも 1 つ指定されているか、またはこれらが指定されていないか、そして PCB 名がアプリケーションで使用されたものと一致しているかどうか。
- PSB、DB、および PSB が一致している SBPARM 制御カードに DD= キーワードが少なくとも 1 つ指定されているか、またはこれらが指定されていないか、そして DD 名がアプリケーションで使用されたものと一致しているかどうか。
- SBPARM 制御カードは読み取られたかどうか。答えが「yes」であれば、続く統計では、どんな SBPARM キーワードが使用されたかが示されます。これは、アプリケーション・プログラムに順次バッファリングが使用された理由または使用されなかった理由を判別するのに役立ちます。
- SBPARM 制御カードに PSB= キーワードが少なくとも 1 つ指定され、しかも、アプリケーションが使用した PSB と一致しているかどうか。

この報告書に表示される他のフィールドには、以下のものがあります。

NUMBER OF SEARCH REQUESTS ISSUED BY THE OSAM BH

このフィールドには、OSAM バッファ・ハンドラーが、特定の OSAM ブロックを SB バッファ・プールから検索するよう SB バッファ・ハンドラーに要求した回数が示されます。

このフィールドの値は、SB なしで出せたはずの OSAM ランダム読み取り入出力命令の数と同じです。

NUMBER OF READ I/O

これらの各フィールドは、アプリケーション・プログラムが出した要求を満たすために出された OSAM ランダム読み取り入出力命令の数および OSAM 順次読み取り入出力命令の数を示します。これら 2 つの数を合計したものは、アプリケーションのために出された OSAM 読み取り入出力命令の合計数になります。この合計を NUMBER OF SEARCH REQUESTS ISSUED BY THE OSAM BH フィールドから引けば、SB を使用してどれだけの読み取り入出力命令を節約したかを計算することができます。

NUMBER OF BLOCKS READ

これらの各フィールドは、アプリケーション・プログラムからの要求を満たすために、OSAM データ・セット・ブロックが読み取られた数を示します。これらのフィールドに示されるのは、以下のものです。

- 読み取られたブロックの合計数
- ランダム読み取りで読み取られたブロック数とパーセンテージ
- 順次読み取りで読み取られたブロック数とパーセンテージ

順次読み取りで読み取られたブロックのパーセンテージが高い場合は、SB の使用によって、おそらくアプリケーション・プログラムの経過時間が短縮されたはずで

PERCENT READ PER SEARCH REQUEST

このフィールドには、SB バッファ・ハンドラーが出した読み取り入出力命令数が、OSAM バッファ・ハンドラーがブロックの検索を SB バッファ・ハンドラーに要求した回数のパーセンテージで表示されます。

パーセンテージが小さい場合、検索要求の多くが、入出力命令を出さなくても満たされたことを示します。したがって、このフィールド内の値が低いことは、SB の使用が、アプリケーション・プログラムの経過時間を少なくするのに役立ったはずであることを示します。

NUMBER OF SEQUENTIAL I/O ERRORS

このフィールドには、入出力エラーになった順次読み取りの数が示されます。順次読み取り時に入出力エラーが検出されると、IMS はこのフィールドの値を増やし、読み取りに関係している 10 個の SB バッファに無効のマークを付けます。その後、IMS は、OSAM バッファ・ハンドラーが要求したブロックのランダム読み取りを実行します。

順次バッファリング明細報告書

この報告書は、SB が特定の SB バッファ・プールにどのように使用されたかについて詳細な情報を示します。アプリケーション・プログラムが使用した SB バッファ・プールごとに、別々の報告書が作成されます。

各報告書は、A、B、および C の 3 ページから成っています。要約情報はページ A にあります。より詳細な情報はページ B および C にあります。

以下の各トピックには、それぞれ順次バッファリング明細報告書の関連ページ (A、B、または C) の例が記載してあります。

順次バッファリング明細報告書のページ A のフィールド

以下の例は、報告書のページ A を示しています。

```
//DFSSTAT STATISTICS FOR: JOB=OSBTC01 STEP=STEP1 . PGM=DFSDDL0 PSB=PBVDSALR DATE=93.058 TIME=09.39
-----
*** SB DETAIL STATISTICS (PAGE A) ***
PSB          PBVDSALR
DB           DBOVLFPC
PCB
DB-PCB NBR          1
DSG-CB NBR          1
DD           VLOSAM01
DB-ORG        HDAM
DD-TYPE       *PSDATA
NBR OF BUFSETS      4
COMPARE-OPTION IS ACTIVE
** NUMBER OF SEARCH REQUESTS ISSUED BY OSAM BH:
    SEARCH                      2,213
** NUMBER OF READ I/O:
    TOTAL                        1,338
    RANDOM READ                   686
    SYNCHRONOUS SEQUENTIAL READ   555
    OVERLAPPED SEQUENTIAL READ    97
** NUMBER OF BLOCKS READ:
    TOTAL                        7,206
    RANDOM READ                   686
    SYNCHRONOUS SEQUENTIAL READ   5,550
    OVERLAPPED SEQUENTIAL READ    970
    PCT OF TOTAL:                   9.51
    PCT OF TOTAL:                   77.01
    PCT OF TOTAL:                   13.46
** AVERAGE I/O WAIT TIMES (MILLIS):
    RANDOM READ                   15.70
    SYNCHRONOUS SEQUENTIAL READ   18.03
    OVERLAPPED SEQUENTIAL READ     .26
```

このページの最初のトピックを使用して、この報告書が適用される SB バッファークラスタ、データベース PCB、および DD 名を識別できます。ある特定のデータベース PCB および DD 名用に作成された SB バッファークラスタが複数 (したがって、複数の報告書が) あることがあります。これは、PCB が論理関係に参与している場合に起こることがあります。

このトピックには、以下の情報が含まれています。

- PSB 名
- DBD 名 (PSBGEN 時の PCB マクロ内にコーディングされる)
- PCB ラベル (PSBGEN 時の PCB マクロ内にコーディングされる)
- PSB 内のデータベース PCB の相対番号
- データベース PCB 内のデータ・セット・グループ制御ブロックの固有 ID (同じ PCB および DD 名に複数の SB バッファークラスタが作成された場合、この番号を使用して SB バッファークラスタを一意的に識別することができます。)
- DD 名
- データベース編成のタイプ (HDAM、HIDAM、または HISAM)
- データベース・データ・セットのタイプ。これは、次の 3 つ値のいずれかです。
 - *INDX (この報告書が、索引として使用されるデータ・セットに適用されることを表します。)
 - *PSDATA (この報告書が、基本順序に従ってアクセスされるデータが入っているデータ・セットに適用されることを表します。)
 - *SSDATA (この報告書が、副次索引順序に従って、または論理関係の交差によりアクセスされるデータが入っているデータ・セットに適用されることを表します。)
- SB バッファークラスタ内のバッファークラスタ数。バッファークラスタのデフォルト数は 4 です。ただし、このデフォルトは、SBPARM 制御ステートメントで、または SB 初期設定出口ルーチンの中で変更している場合があります。

NUMBER OF SEARCH REQUESTS ISSUED BY OSAM BH

このフィールドには、SB を使用せずに出せたはずの OSAM ランダム読み取り入出力命令の数が示されます。

NUMBER OF READ I/O

これらのフィールドは、この SB バッファークラスタで実際に出された読み取り入出力命令の数を示します。これらのフィールドは、次のことを示します。

- OSAM 読み取り入出力命令の合計数
- ランダム読み取りの数
- 同期順次読み取りの数
- オーバーラップした (非同期) 順次読み取りの数

これらのフィールドを使用すれば、このデータベース PCB と DD 名に使用された 3 つのタイプの読み取り入出力操作のそれぞれについてパーセンテージを計算することができます。さらに、NUMBER OF SEARCH REQUESTS ISSUED BY OSAM BH フィールドから OSAM 読み取り入出力操作の合計数を差し引けば、SB を使用することで出さずにすんだ読み取り入出力操作の数を知ることができます。

NUMBER OF BLOCKS READ

これらのフィールドは、各タイプの読み取り入出力操作によって読み取られたブロック数を示します。これらのフィールドは、次のことを示します。

- 読み取られたブロックの合計数
- ランダム読み取りで読み取られたブロック数とパーセンテージ
- 同期順次読み取りで読み取られたブロック数とパーセンテージ
- オーバーラップしている (非同期) 順次読み取りで読み取られたブロック数とパーセンテージ

順次読み取りで読み取られたブロックのパーセンテージが高いということは、少なくともこのデータベース PCB とデータ・セットの処理中は、おそらく、SB を使用したことがアプリケーション・プログラムの実行速度を上げるのに貢献したことを示しています。

一方、ランダム読み取りで読み取られたブロックのパーセンテージが高いということは、次のことを示している可能性があります。

- アプリケーション・プログラムによる大量のランダム処理
- このデータベース PCB に関連する OSAM データ・セットの再編成が必要であること

多数のブロックがランダム読み取りで読み取られる場合で、しかもプログラムがデータベースを順次処理している場合には、バッファ・セットの数を増やすことにより (例えば、//DFSCTL 内の SBPARM 制御ステートメントで BUFSETS パラメーターの値を増やして) バッファリング・パフォーマンスを向上させることができる場合があります。バッファ・セット数を増やした後、NUMBER OF READ I/O フィールドに示される数値が、次にアプリケーションが実行されたときにどのように変化するかを観察してください。

ランダムに読み取られたブロックのパーセンテージが低いということは、バッファ・セットの数を減らして、仮想記憶域スペースを節約できることを示している場合があります。

AVERAGE I/O WAIT TIMES (MILLIS)

これらのフィールドは、読み取り入出力命令の 3 つの各タイプごとの平均入出力待ち時間を示します。これらの時間は、アプリケーション・プログラムが、読み取られるデータを処理するために、その前に読み取り入出力操作が完了するのを待つ平均時間を示します。

これらのフィールドの時間の単位はミリ秒です。ミリ秒とは、1 秒の 1000 分の 1 のことです (つまり、50 ミリ秒は 0.050 秒です)。

これらのフィールドに示されるのは、以下のものです。

- ランダム読み取りの平均入出力待ち時間
- 同期順次読み取りの平均入出力待ち時間
- オーバーラップした (非同期) 順次読み取りの平均入出力待ち時間

これらの時間が測定されるのは、SB がアクティブなときか、IMS が入出力参照パターンをモニターしているときのみです。

順次バッファリング明細報告書のページ B のフィールド

以下の例は、報告書のページ B を示しています。

```

-----
*** SB DETAIL STATISTICS: REFERENCE STATISTICS (PAGE B) ***
** REFERENCES IN BUFFER-SETS:
    RATIO                                .23
** REFERENCES IN RANDOM SRAN CBS:
    RATIO                                .20
** RANDOM SRAN CBS WHICH HAVE BEEN CONVERTED:
    NUMBER                                0
    PCT OF STOLEN RANDOM SRAN           .00

```

```

***** DISTRIBUTION OF REFERENCES IN BUFFER-SETS *****
-----
REFERENCE COUNT  NBR OF OCCURRENCES  PCT OF OCCURRENCES  ACCUMUL. PCT
0                3                .46                 .46
1                551               84.50              84.96
2                0                .00                 84.96
3                0                .00                 84.96
4                0                .00                 84.96
5                0                .00                 84.96
6                0                .00                 84.96
7                0                .00                 84.96
8                0                .00                 84.96
9                4                .61                 85.58
=> 10            94                14.41              100.00

```

```

***** DISTRIBUTION OF REFERENCES IN RANDOM SRAN CBS *****
-----
REFERENCE COUNT  NBR OF OCCURRENCES  PCT OF OCCURRENCES  ACCUMUL. PCT
0                0                .00                 .00
1                0                .00                 .00
2                1                100.00             100.00

```

このページを使用すれば、この SB バッファ・プールの入出力参照パターンをモニターした SB アルゴリズムの効率を評価できます。このページを分析することにより、次の質問の答えが得られます。

- 順次読み取りを出す決定がどれほど効率的であるか。この質問の答えは、REFERENCES IN BUFFER SETS フィールドから得られます。
- ランダム読み取りを出す決定がどれほど効率的であるか。この質問の答えは、REFERENCES IN RANDOM SRAN CBS フィールドと RANDOM SRAN CBS WHICH HAVE BEEN CONVERTED フィールドから得られます。

ページ B 上のフィールドは、SB がアクティブである場合のバッファリング・アクティビティを反映します。IMS が入出力参照パターンのモニターのみを行っている場合のバッファリング・アクティビティは含まれません。

REFERENCES IN BUFFER SETS

このフィールドは、順次読み取りで読み取られたブロックを、OSAM バッファ・ハンドラーが参照した回数を示します。この率は、次のように算出されます。

バッファ・セット内での参照回数を、順次読み取りで読み取ったブロック数で除算

例えば、1.00 という率は、順次読み取りで読み取られた各ブロックが平均で 1 回参照されたことを意味します。0.50 という率は、順次読み取りで読み取られたブロックの半分のみが平均で 1 回参照されたことを意味します。

一般に、率が高いと (0.85 など)、順次読み取りを出す決定が効率的であって、おそらく、アプリケーションの実行時間を短縮するのに役立ったことを示します。率が低い (0.30 など) ということは、読み取ったブロックの多くが参照されなかったために順次読み取り命令を出した効果が小さくなったことを示します。

REFERENCES IN RANDOM SRAN CBS

このフィールドを見れば、ランダム読み取りを出した決定の有効性がわかります。この率が表している意味を理解するには、まず、SB バッファ・ハンドラーが、制御ブロックを用いて入出力参照パターンを追跡する方法を理解する必要があります。

SB バッファ・ハンドラーは、SB 範囲 (SRAN) と呼ばれる制御ブロックを用いて、10 個の連続ブロックの範囲内の入出力参照パターンを追跡します。各 SRAN には参照カウンターがあり、このカウンターは、SRAN が追跡する 10 個のブロックからなるセットに含まれる 1 つのブロックを OSAM バッファ・ハンドラーが要求した回数を示します。

SRAN 制御ブロックには 2 つのタイプがあります。1 つは、順次 SRAN と呼ばれるもので、順次読み取りで読み取られた 10 個の連続ブロックのカウンタを保持します。もう一方は、ランダム SRAN と呼ばれるもので、ランダム・パターンで参照された 10 個の連続ブロックのカウンタを保持します。SB バッファ・プール内のバッファ・セットごとに 1 つの順次 SRAN があります。ランダム SRAN の数は、順次 SRAN の数の 2 倍です。

SRAN は「使用チェーン」上に一緒につながられています。つまり、SRAN は、使用の順序に関して新しい順または古い順につながられています。SRAN の各タイプにはそれぞれの使用チェーンがあります。一番最近使用された SRAN は、使用チェーンの最上部にあり、最も以前に使用された SRAN は最下部にあります。現在追跡されていないブロック・セットの追跡に SRAN が必要なときには、SB バッファ・ハンドラーが必ず使用チェーンの下部の SRAN を選択します。これは、最近参照されていない連続ブロックの範囲は、近い将来に再び参照される可能性が少ないためです。

ランダム SRAN が再利用されるたびに、SRAN に保持されている参照カウンタは、後で利用するために記録されます。アプリケーション・プログラムの終了時には、これらの参照カウンタを使用してこのフィールド内の数値が計算されます。数値は次のように計算されます。

$$\text{率} = X / (Y * 10)$$

ここで、

- X** ランダム SRAN 内のランダム参照の合計数 (この数字は、通常、ランダム SRAN で追跡したブロック・セット内のランダム読み取りの数と等しいか、またはそれに近い数です。)
- Y** 再使用されたランダム SRAN の合計数

このフィールド内の率が低い (例えば、0.15) ということは、平均して、SB バッファ・ハンドラーがランダム入出力参照パターンを正しく認識したことを意味します。0.15 という率は、ランダム SRAN が追跡するブロックに、SB バッファ・ハンドラーが平均 1.5 回のランダム読み取りを出したことを意味します。通常は、10 個の連続ブロックのセットに 1.5 回のランダム読み取りを出すほうが、10 個の連続ブロックに 1 回の順次読み取りを出すよりも効率的です。

このフィールド内の率が高い (例えば、0.50) ということは、おそらく、SB バッファ・ハンドラーが頻繁に順次参照パターンをランダム参照パターンと間違えたことを示します。0.50 という率は、ランダム SRAN が追跡するブロックに、SB バッファ・ハンドラーが平均 5 回のランダム読み取りを出したこと

を意味します。通常は、10 個の連続ブロックのセットに 5 回のランダム読み取りを出すほうが、10 個の連続ブロックに 1 回の順次読み取りを出すよりも効率は悪くなります。

RANDOM SRAN CBS WHICH HAVE BEEN CONVERTED

これらのフィールドは、入出力参照パターンを分析するときに SB バッファーマ・ハンドラーの効率を測定する別の尺度になります。

SB バッファーマ・ハンドラーは、時には、連続ブロック・セットへの参照をランダム・パターンと解釈し、その後さらにいくつか参照した後で、ブロック・セットが実際には順次に参照されていることを検出することがあります。この場合には、SB バッファーマ・ハンドラーは、まず連続ブロックのセット内でいくつかのランダム読み取りを出し、後でその同じブロック・セットに 1 つの順次読み取りを出します。これが起きると、ブロック・セットを追跡するランダム SRAN は順次 SRAN に変換されます。SB バッファーマ・ハンドラーが最初から順次読み取りでブロック・セットを読み取っていれば、いくつかのランダム読み取りを出すというコストはかからなくてすんだはずですが。

このトピックのフィールドは、以下のとおりです。

NUMBER

これは、アプリケーション・プログラムの実行中にランダム SRAN が順次 SRAN に変換された回数を示します。

PCT OF STOLEN RANDOM SRAN

このフィールド内の値は、変換された SRAN の数を、SB バッファーマ・ハンドラーが使用チェーンからランダム SRAN を獲得しなかった回数で表します。パーセンテージが高い (例えば、40%) ということは、おそらく、SB バッファーマ・ハンドラーが頻繁に、順次参照パターンをランダム参照パターンと間違えたことを示します。

DISTRIBUTION OF REFERENCES IN BUFFER SETS

この表を見れば、REFERENCES IN BUFFER SETS フィールドに表示された率についてさらに詳しい情報がわかります。

例えば、REFERENCES IN BUFFER SETS フィールドは、SB バッファーマ・セット内のブロックの 80% (0.80) が参照されたことを示す場合があります。これはどのような意味を持つのでしょうか。その時間の 20% は、バッファーマ・セット内のどのブロックも参照されなかったことを意味するかもしれません。あるいは、その時間の 100% の間に、バッファーマ・セット内の 10 ブロックの内 8 ブロックだけしか参照されなかったことを意味することもあります。この質問には、この表を分析すれば答えることができます。

この表のフィールドには、次のものがあります。

REFERENCE COUNT

この欄には、バッファーマ・セット内のブロックを参照した回数が入力されます。この欄に示されるゼロは、バッファーマ・セット内のブロックへの参照は行われなかったことを意味し、1 が示される場合は、1 回参照されたことを意味します。以下同様です。この欄の最後の行の「=> 10」は、10 回以上の参照が行われたことを意味します。

NBR OF OCCURRENCES

この欄は、REFERENCE COUNT 欄に示された回数だけ参照されたバッファ

ァー・セットの個数を示します。例えば、図では、この欄の最初の数字は 3 ですが、これは、参照されなかったバッファァー・セットが 3 個だということです。2 番目の数字は、1 回のみ参照されたバッファァー・セットが 551 個だということです。

PCT OF OCCURRENCES

この欄は、NBR OF OCCURRENCES 欄の値を、バッファァー・セットが再利用された回数のパーセンテージで表しています。例えば、図では、この欄の最初の数字は 0.46 ですが、これは、再利用されたバッファァー・セットの 0.46% が参照されなかったことを示します。2 番目の数字は、再利用されたバッファァー・セットの 84.50% が 1 回のみ参照されたことを示します。

ACCUMUL. PCT

この欄は、ゼロから現行参照カウントまでの累積 PCT OF OCCURRENCES を示します。例えば、図では、この欄の 3 番目の数字は 84.96 ですが、これは、再利用されたバッファァー・セットの 84.96% が 2 回またはそれ以下の回数参照されたことを示します。

DISTRIBUTION OF REFERENCES IN RANDOM SRAN CBS

この表を見れば、REFERENCES IN RANDOM SRAN CBS フィールドに報告された率についてのさらに詳しい情報が分かります。この表は、DISTRIBUTION OF REFERENCES IN BUFFER SETS 表と似ていますが、ランダム SRAN 制御ブロックが追跡する参照に関する情報を示しているという点で異なっています。

REFERENCE COUNT

この欄には、ランダム SRAN が追跡するブロックへの参照の回数かリストされます。この欄に示されるゼロは、参照が行われなかったこと、したがって、ランダム SRAN が追跡するブロックにランダム読み取りは出されなかったことを意味します。「1」は、ランダム SRAN が追跡するブロックに 1 回参照が行われたこと (平均として、ランダム SRAN が追跡する 10 個目のブロックごとに 1 回の参照) を意味します。

NBR OF OCCURRENCES

この欄は、REFERENCE COUNT 欄に示された回数だけ参照されたランダム SRAN の個数を示します。

PCT OF OCCURRENCES

この欄は、NBR OF OCCURRENCES を、ランダム SRAN が再利用された回数のパーセンテージで表しています。

ACCUMUL. PCT

この欄は、ゼロから現行参照カウントまでの累積 PCT OF OCCURRENCES を示します。

順次バッファァリング明細報告書のページ C のフィールド

以下の例は、報告書のページ C を示しています。

```
//DFSSTAT STATISTICS FOR: JOB=OSBTC01 STEP=STEP1 . PGM=DFSDDLTO PSB=PBVDSALR DATE=93.058 TIME=09.39
-----
*** SB DETAIL STATISTICS: INTERNAL COUNTERS AND VALUES (PAGE C) ***
** DEACTIVATIONS:
   NBR OF SB-DEACTIVATION                1
   NBR OF MONITORING-DEACTIVATION         0
** RESULTS OF EVALUATION OF SEQUENTIALITY:
   NBR POSITIVE RESULTS                   3
```

NBR NEGATIVE RESULTS	1
** RESULTS OF EVALUATION OF ACTIVITY RATE:	
NBR POSITIVE RESULTS	4
NBR NEGATIVE RESULTS	0
** NBR RANDOM READ:	
DURING SEQUENTIAL BUFFERING PHASES	27
DURING "MONITORING-ONLY" PHASES	659
WHILE NOT MONITORING REFERENCE PATTERN	0
** NBR RANDOM READS WITH SEQUENTIAL REFERENCE PATTERN:	
ACCESS TO INVALID BUFFERS	25
ACCESS AT DATA SET END	2
** NBR OF BUFFERING POSITIONS:	2,213
** INTERNAL SB-ALGORITHM VALUES:	
SDSGBPTR: BLOCKS PER TRACK	31
SDSGNBRB: BLOCKS PER BUFSET	10
SDSGSCST: RELATIVE SEQ I/O COSTS	1.36
SDSGSINB: SIZE OF NEIGHBORHOOD	2
SDSGTHR1: THRESHOLD CURRENT+1	2
SDSGTHR2: THRESHOLD OVERLAP	3
SDSGTHR3: THRESHOLD NEIGHB	11

ページ C は、以下のトピックから構成されています。

DEACTIVATIONS

このトピックのフィールドは、以下のとおりです。

NBR OF SB DEACTIVATION

このフィールドには、SB が非アクティブ化された回数が表示されます。

NBR OF MONITORING DEACTIVATION

このフィールドには、入出力参照のモニターを非アクティブ化した回数が表示されます。

入出力参照のモニターの非活動化は、次の 2 つの理由のうちのいずれかで起こります。

- 連続して何回か定期的にバッファリング処理を評価した結果、SB を使用しても意味がないとわかった場合。
- SBONLINE 制御ステートメントで MAXSB キーワードを指定して、SB バッファ・スペースの上限を設定した場合。この上限に達すると、SB の使用は制限されます。

RESULTS OF EVALUATION OF SEQUENTIALITY

このトピックと次のトピック (RESULTS OF EVALUATION OF ACTIVITY RATE) は、アプリケーション・プログラムの実行中に SB が非アクティブにされた (またはアクティブにされなかった) 理由を説明するのに役立ちます。

SB の活動化または非活動化の決定は、特定の DB-PCB/DSG 制御ブロックの対のバッファリング・プロセスを定期的に評価することにより行われます。この評価は、次の基準で行われます。

- 連続性
- アクティビティ率

両方のテストの結果が肯定的であれば (つまり、両方のテストの結果により SB の使用が勧められる場合)、IMS は SB を活動化するか (まだアクティブでない場合)、SB の使用を続行します。少なくとも一方のテストの結果が否定的であれば、IMS は SB を非アクティブにするか (アクティブな場合)、あるいは引き続き SB を使用しません。SB を使用しない決定を何回か行った後に、IMS が入出力参照パターンのモニターを非アクティブ化する場合があります。

このトピックのフィールドは、以下のとおりです。

NBR POSITIVE RESULTS

このフィールドには、入出力参照パターンの定期的な評価により、SB の使用を効果的と認められるだけの順次参照パターンが検出された回数が表示されます。

NBR NEGATIVE RESULTS

このフィールドには、入出力参照パターンの定期的な評価により、SB の使用を効果的と認め得るだけの順次参照パターンが検出されなかった回数が表示されます。

RESULTS OF EVALUATION OF ACTIVITY RATE

このトピックのフィールドは、以下のとおりです。

NBR POSITIVE RESULTS

これは、定期的な評価により、SB の使用を効果的と認めるのに十分高い入出力アクティビティー率が検出された回数を示します。

NBR NEGATIVE RESULTS

これは、定期的な評価により、SB の使用を効果的と認めるほど高い入出力アクティビティー率ではないと判断された回数を示します。

NBR RANDOM READ

このトピックには、以下の各タイプのバッファリング・フェーズで、ランダム読み取りが出された回数が表示されます。

DURING SEQUENTIAL BUFFERING PHASES

このフィールドには、SB がアクティブなときに出力されたランダム読み取りの回数が表示されます。

DURING 『MONITORING ONLY』 PHASES

このフィールドには、SB がアクティブでなく、IMS がまだ入出力参照パターンをモニターしているときに出力されたランダム読み取りの回数が表示されます。

WHILE NOT MONITORING REFERENCE PATTERN

このフィールドには、SB がアクティブでなく、IMS がまだ入出力参照パターンをモニターしていないときに出力されたランダム読み取りの回数が表示されます。

NBR RANDOM READS WITH SEQUENTIAL REFERENCE PATTERN

このトピックには、入出力参照パターンが順次であってもランダム読み取りが出された回数を表示します。これらのカウンターは、SB がアクティブである場合にのみ更新されます。

このトピックのフィールドは、以下のとおりです。

ACCESS TO INVALID BUFFERS

このフィールドは、SB バッファーの内容が無効で使用できなかったために出力された、ランダム読み取りの回数を表示します。SB バッファーに無効のマークを付けられる理由には、次のようなものがあります。

- IMS システムがブロック・レベル共用環境で実行されている。ブロック・レベル共用環境では、複数の IMS システムが同じデータベースを読み書きできます。例えば、IMS システム「A」がブロックをバッファーに読み取り、そのブロックがまだシステム「A」のバッファー内にある間

に、IMS システム「B」がそのブロックを更新すると、システム「A」のバッファには無効のマークが付けられます。

- 定期的な評価時に SB の活動化および非活動化で SB バッファに無効のマークが付けられた。
- SB バッファ・ハンドラーがブロックを読み取っているのと同時に、同じ IMS サブシステム内の別のオンライン・アプリケーションが、そのブロックを書き出していた。
- 入出力エラーのために SB バッファに無効のマークが付けられた。

ACCESS AT DATA SET END

このフィールドは、参照されるブロックを含んでいる 10 個のブロックからなるセットが完全にフォーマット設定されなかったために SB バッファ・ハンドラーが出した、ランダム読み取りの回数を示します。SB は、連続ブロックの最後のセットのフォーマット設定が完全でなければ、データ・セットの最後で順次読み取りを出しません。

NBR OF BUFFERING POSITIONS

このフィールドは、データベース内の新しい位置を要求する DL/I 呼び出しをアプリケーション・プログラムが出したと、SB バッファ・ハンドラーが見なした回数を示します。例えば、現行ルート・セグメント以外のルート・セグメントのキー・フィールドで修飾されたほとんどの GU 呼び出しで、このカウンターの値が増分されます。このカウンターが保守されるのは、SB がアクティブで、IMS が入出力参照パターンをモニターしている間のみです。

このフィールドの値が大きいということは、アプリケーションが大量の論理ランダム処理を行ったことを表していると言えます。SB を使用してもこのアプリケーションにとって効果がなかったことを他のフィールドも示している場合は、このフィールドがその理由の説明になります。

INTERNAL SB ALGORITHM VALUES

このトピックは、内部カウンターの値を示します。これらは組み込まれて、IMS 開発で SB の問題の判別に役立ちます。

第 51 章 統計分析報告書、ログ・トランザクション報告書、およびログ・レコード分析

IMS には、IMS システム・ログからデータを抽出するいくつかのユーティリティがあります。

- 統計分析ユーティリティ。これは、メッセージ・アクティビティの要約報告書、回線と端末に関する報告書、およびメッセージ選択/リスト報告書を作成します。
- ログ・トランザクション分析ユーティリティ。これは、個々のトランザクションと処理アクティビティの詳細情報を与えます。

統計分析ユーティリティの報告書

統計分析ユーティリティの入力データは、1 組の IMS システム・ログ・データ・セット、またはログ保存ユーティリティが作成したユーザー・データ・セットです。

各入力データ・セットは、複数のボリュームで構成されることもあります。複数のデータ・セットを連結させることができます。共用キューまたは MSC で接続された複数の IMS システムからのデータ・セットを入れることもできます。

制約事項: バッチ・システムからのシステム・ログ出力を使用することはできません。

トランザクション分析ユーティリティを使用すれば、内容を少なくした新しいシステム・ログを入手でき、これによって処理時間を節約できます。

統計分析ユーティリティには、トランザクション・アクティビティのサブセットを選択するときを使用できる制御ステートメントが 6 つあります。

トランザクション・コード制御ステートメント

この制御ステートメントを使用すれば、特定のトランザクション・コードまたはトランザクション・コード・グループを選択できます。

シンボリック端末名制御ステートメント

この制御ステートメントを使用すれば、LTERM 名または総称名を指定できます。例えば、L3270M を指定すると、その LTERM から発信される、またはその LTERM に向けられるすべてのメッセージが選択されます。総称名 L3270* を指定すると、比較は * の前の文字に基づいて行われ、L3270M および L3270B のメッセージが選択されます。

指定された入力 LTERM から生じる特定のシンボリック名へのメッセージのみが選択されるように、出力 LTERM をさらに修飾することができます。

時間制御ステートメント

選択の基準として間隔を指定できます。開始時刻および停止時刻を、YYDDD および HHMM の形式 (ユリウス日付と分単位の時刻) で指定し

ます。この範囲基準は、トランザクション・コードおよび端末指定で選択されたすべてのメッセージに適用されます。

メッセージ選択出力順序ステートメント

この制御ステートメントを使用して、メッセージをリストする順序を決定します。この制御ステートメントが影響を与えるのは、メッセージ選択/リスト (コピー) 出力の内容のみです。

印刷不能文字制御ステートメント

メッセージ・テキスト中に印刷不能文字が含まれることが予期される場合には、それらを 16 進形式 (2 番目の文字の上に先頭文字) で印刷するよう指定できます。指定しない場合は、文字はブランクとして表されます。

統計分析ユーティリティ (DFSISTS0) について詳しくは、「IMS V15 システム・ユーティリティ」を参照してください。

トランザクション負荷の計算

統計分析ユーティリティが作成する報告書には、1 日 (24 時間) のトランザクション・アクティビティの分布を要約する報告書が 2 つあります。入力メッセージおよび出力メッセージの分布は、トランザクション・コードごと、および各装置ごとに別々に表にされます。

さらに他の報告書では、トランザクション・タイプごとに応答時間を百分位数で示します。報告されるデータは、このユーティリティの入力として選択したシステム・ログ・データにより異なります。報告書の範囲は、報告の間隔だけでなく、トランザクションのサブセットと回線トラフィックを選択することにより、さらに制限することができます。

以下の図に、回線および端末報告書の形式を示します。1 つの回線上の装置ごとに、LTERM 名が示され、送受信アクティビティの 2 行 1 組みの結果が示されています。「TOTAL MESSAGES」欄に続いて、メッセージの合計サイズと平均サイズ (バイト単位) が示されています。1 時間間隔で 1 日 (24 時間) が分割され、それらの間隔中のアクティブ・トランザクションのカウントが記録されています。

LINE	AND	TOTAL	TERMINAL	REPORT	DATE	06/05/07													PAGE
						00-07	07-08	08-09	09-10	10-11	11-12	12-13	13-14	14-15	15-16	16-17	17-18	18-19	
NODE	R/S	MESSAGES	TOTAL	AVG	HOURLY	DISTRIBUTION													
			CHARACTERS	SIZE															
CTRL	R	5	782	156	0	0	0	0	0	5	0	0	0	0	0	0	0	0	
	S	10	1,123	112	0	0	0	0	0	10	0	0	0	0	0	0	0	0	
CTRLA01	R	3	365	121	0	0	0	0	0	3	0	0	0	0	0	0	0	0	
	S	2	276	138	0	0	0	0	0	2	0	0	0	0	0	0	0	0	
CTRLA02	R	2	186	93	0	0	0	0	0	2	0	0	0	0	0	0	0	0	
	S	4	306	76	0	0	0	0	0	4	0	0	0	0	0	0	0	0	
L62MVS1	R	9	366	40	0	0	0	0	0	9	0	0	0	0	0	0	0	0	
	S	9	483	53	0	0	0	0	0	9	0	0	0	0	0	0	0	0	
SEGUNDO	S	28	1,842	65	0	0	0	0	0	28	0	0	0	0	0	0	0	0	
SYSTEM	R	19	1,699	89	0	0	0	0	0	19	0	0	0	0	0	0	0	0	
TOTALS	S	53	4,030	76	0	0	0	0	0	53	0	0	0	0	0	0	0	0	

キー: R/S - 受信/送信

入力のみ、トラフィックまたは出力のみ、トラフィックに限定された装置の項目には、報告行は 1 行しかありません。以下の例は、トランザクション報告書の形式を示しています。この報告書は、回線および端末報告書と似たデータ編成になっていますが、トランザクション・コード別に順序付けられているという点で異なります。

TRANSACTION REPORT				DATE 06/05/07												PAGE 00003		
TRANSACTION CODE	R/S	TOTAL MESSAGES	TOTAL CHARACTERS	AVG SIZE	HOURLY DISTRIBUTION					DISTRIBUTION								
					00-07	07-08	08-09	09-10	10-11	11-12	12-13	13-14	14-15	15-16	16-17	17-18	18-19	19-24
CONV21V0	R	5	246	49	0	0	0	0	0	5	0	0	0	0	0	0	0	0
	S	4	298	74	0	0	0	0	0	4	0	0	0	0	0	0	0	0
SMQR21C0	R	3	90	30	0	0	0	0	0	3	0	0	0	0	0	0	0	0
	S	2	50	25	0	0	0	0	0	2	0	0	0	0	0	0	0	0
TRAN21V0	R	1	30	30	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	S	1	25	25	0	0	0	0	0	1	0	0	0	0	0	0	0	0
SYSTEM	R	9	366	40	0	0	0	0	0	9	0	0	0	0	0	0	0	0
TOTALS	S	7	373	53	0	0	0	0	0	7	0	0	0	0	0	0	0	0

以下の例は、トランザクション報告書の形式を示しています。この報告書は、システム・ログ・セットから入力として選択されたデータに関して、トランザクション・コードごとに最長応答時間および最短応答時間を示します。4つの欄では、百分位数の応答時間を秒単位で記録しています。第25、第50、第75、および第95百分位数が示されています。例えば、第50百分位数内の応答時間は、そのトランザクションで処理される応答時間数の累計の50%より大きいまたは等しくなります。示されている応答時間の最初の行は、入力メッセージを完全に受信してから、応答メッセージが正常にデキューされるまでを示します。端末で、出力メッセージを完全に受信するのに非常に長い時間を要した場合には、2番目の行は、受信から応答メッセージが開始された時刻までを示します。

TRANSACTION RESPONSE REPORT				DATE 06/05/07				PAGE 00004
TRANSACTION CODE	TOTAL RESPONSES	LONGEST RESPONSE	95% RESPONSE	75% RESPONSE	50% RESPONSE	25% RESPONSE	SHORTEST RESPONSE	
CONV21V0	4	.36S	.31S	.31S	.24S	.19S	.19S	
	4	.36S	.31S	.31S	.24S	.19S	.19S	
SMQR21C0	2	.51S	.33S	.33S	.33S	.33S	.33S	
	2	.51S	.33S	.33S	.33S	.33S	.33S	
TRAN21V0	1	.36S	.36S	.36S	.36S	.36S	.36S	
	1	.36S	.36S	.36S	.36S	.36S	.36S	

プログラム間トラフィックの評価

メッセージ処理プログラムが出力メッセージを別のプログラムに向けると、その2次トランザクションはキューに入れられます。トランザクション・コードは、処理プログラムの論理の都合上、固有である場合があります。

固有でない場合は、2次トランザクションは、端末発信元からのメッセージと一緒にキューに入れられます。

別々にトランザクション・トラフィックをカウントするときには、2つのプログラム間メッセージ報告書を使用できます。以下の例は、2つの表を示しています。

「DESTINATION」という表題の列が、別のプログラムへのキューに入れられたトランザクション・コードのリストの上に付いています。元のプログラムがどれかは識別されていません。「TRANSACTION CODE」という表題の列が、2次トランザクションを出したプログラムを呼び出した初期トランザクション・コードのリストの上に付いています。

会話型トランザクションの処理時にプログラム間通信を行うと、これらはこの2つのリストに組み込まれます。

MESSAGES--PROGRAM TO PROGRAM		DATE 06/05/07
DESTINATION	TOTAL MESSAGES	
ELEANOR	1	
SW1050	1	
T2741N1	1	
T2742N3	1	

TRANSACTION CODE	TOTAL MESSAGES
TA10	107

未送信メッセージ数の入手

MESSAGES - QUEUED BUT NOT SENT という表題の 2 つの報告書では、入力テープがカバーする間隔中にまだメッセージ・キューに存在していた出力メッセージ数が合計されます。

この報告書を、以下の例に示します。端末に送られなかったコマンド応答は、(IMSSYS) で示されています。NOTAVA という項目は、「トランザクション使用不能」を示します。システム・ログ入力データに記録されなかった入力に関して出力メッセージが生成された場合、または記録されなかった同じ端末からのコマンド入力で出力メッセージが生成されたような場合が、これに該当します。

MESSAGES - QUEUED BUT NOT SENT		DATE	06/05/07	PAGE	00001
DESTINATION	TOTAL MESSAGES				
CTRLA01	1				
SEGUNDO	37				

重要なトランザクションの監査

統計分析ユーティリティで作成されるメッセージ報告書 (オプション) を DFSIST40 プログラムと一緒に使用すれば、特定のトランザクション・コードの入出力データを詳細に調べることができます。これにより、入力メッセージ内に何があったかを正確に監査でき、エラーがないかどうか出力内容を調べることができます。

この報告書を、以下の例に示します。

MESSAGES

```

INPUT SEG=001 LEN=020* CONV12V0 Y c *
INPUT SEG=002 LEN=023*MESSAGE TO CONV12V0 > *
PGM SW SEG=001 LEN=066* CONV21V0 Y THIS DATA INSERTED TO THE SPA OF CONV21V0 c *
PGM SW SEG=002 LEN=042*MESSAGE TO CONV21V0 > *
OUTPUT SEG=001 LEN=066* Y THIS TERMINATEED TO THE SPA OF CONV21V0 c *
OUTPUT SEG=002 LEN=035*RESPONSE TO INPUT TERMINAL > *

```

INPUT PREFIX	TRANSACTION CODE	LTERM	SEQ NO	DATE	TIME	OUTPUT PREFIX	LTERM	SEQ NO	DATE	TIME
	CONV12V0	TNDSE	00001	07.053	12.37.18		TNDSE	00001	07.053	12.37.19

ログ・トランザクション分析ユーティリティの報告書

ログ・トランザクション分析ユーティリティ (DFSILTA0) を使用すれば、個々のトランザクション・レベルでの詳細データを入手することができます。このユーティリティではデータは要約されませんが、明細報告書の各行には、多数の情報項目が集められ、トランザクション・タイプへのサービスおよびスケジューリング・アルゴリズムの効果を評価する際に役立ちます。入力データが IMS ログなので、報告書には実際の応答データが示されます。

IMS ログ全体を処理しない場合には、開始チェックポイントからカットオフ・ポイントまでのデータが提示されます。開始時刻と期間 (分単位) を指定して、分析するトランザクション処理サンプルを制限します。あるいは開始チェックポイントの後に含まれるチェックポイントの数を指定することができます。取り消されたメッセージは除外されます。

ログ分析報告書の形式および各報告書明細行のデータ項目の完全なリストは、『スケジューリング・アクティビティの検査』に示してあります。時間は経過時間で、10 分の 1 秒に最も近い値まで示されます。処理タイプ・フィールドは重要な説明項目であることが分かります。

報告書明細レコード内のフィールドの開始位置と長さを使用すれば、ユーティリティ実行の 2 番目のステップにソート順序を指定できます。

報告書をメッセージ・クラスとトランザクション優先順位別に順序付けを行うためのソート制御ステートメントは、次のようになります。

```
SORT FIELDS = (18,3,CH,A,16,1,CH,A)
```

さらに、明細報告書レコードの DASD データ・セットを作成するオプションを使用することができます。インストール先では、データの抽出と要約を行う分析プログラムを開発することができます。

スケジューリング・アクティビティの検査

ログ・トランザクション分析ユーティリティで抽出されたデータを使用すると、スケジューリング・アルゴリズムの効果を調べることができます。

トランザクションの各オカレンスでは、主に次の内容を示します。

- メッセージ優先順位とメッセージ・クラス
- 入力キューに入っていた時間
- 処理時間
- 出力キューに入っていた時間
- システム内にいた合計時間 (メッセージ・キューに完全に入力された時点から、出力のために検索されるまでの時間を測定)

送信時間と受信時間を見るときは、処理タイプ「S」の項目を調べることができます。トランザクション・コード別に明細報告書行をソートして、迅速な応答時間が求められる重要なトランザクションを調べることができます。各テーブル行項目の詳細については、以下の例を参照してください。

```
12.50.11 JOB00386 $HASP373 DFSILTA4 STARTED - INIT 4 - CLASS K - SYS STL1
12.50.12 JOB00386 SMF0001 DFSILTA4 ILTA DFSILTA0 0000
12.50.12 JOB00386 $HASP395 DFSILTA4 ENDED
----- JES2 JOB STATISTICS -----
05 JUN 2007 JOB EXECUTION DATE
30 CARDS READ
143 SYSOUT PRINT RECORDS
0 SYSOUT PUNCH RECORDS
10 SYSOUT SPOOL KBYTES
0.01 MINUTES EXECUTION TIME
1 //DFSILTA4 JOB 'TERRY',CLASS=K,MSGCLASS=A,MSGLEVEL=(1,1), JOB00386
// REGION=0M,TIME=1440,
// USER=USRT001,PASSWORD=
/*ROUTE PRINT THISCPU/IMSTST45
2 //JOBLIB DD DSN=IMSTESTL.TNUC0,DISP=SHR
3 // DD DSN=IMBLD.I10RTS17.CRESLIB,DISP=SHR
4 //ILTA EXEC PGM=DFSILTA0
```

```

5 //HEADING DD SYSOUT=A
6 //PRINTER DD SYSOUT=A
7 //SYSUDUMP DD SYSOUT=A
8 //REPORT DD DUMMY
9 //TITLE DD *
//*
//* LOG INPUT FOLLOWS...
//*
10 //LOGIN0A DD DSN=IMSTESTL.SLDSP.IMSA.TYTY,DISP=SHR,
// UNIT=SYSDA,VOL=SER=DSHR03
11 //LOGIN01 DD DSN=IMSTESTL.SLDSP.IMS1.TYTY,DISP=SHR,
// UNIT=SYSDA,VOL=SER=DSHR03
12 //LOGIN02 DD DSN=IMSTESTL.SLDSP.IMS2.TYTY,DISP=SHR,
// UNIT=SYSDA,VOL=SER=DSHR03
13 //LOGIN03 DD DSN=IMSTESTL.SLDSP.IMS3.TYTY,DISP=SHR,
// UNIT=SYSDA,VOL=SER=DSHR03
ICH70001 USRT001 LAST ACCESS AT 12:50:09 ON TUESDAY, JUNE 5, 2007
IEF236I ALLOC. FOR DFSILTA4 ILTA
IEF237I 04A1 ALLOCATED TO JOBLIB
IEF237I 04B4 ALLOCATED TO
IEF237I JES2 ALLOCATED TO HEADING
IEF237I JES2 ALLOCATED TO PRINTER
IEF237I JES2 ALLOCATED TO SYSUDUMP
IEF237I DMY ALLOCATED TO REPORT
IEF237I JES2 ALLOCATED TO TITLE
IEF237I 04A9 ALLOCATED TO LOGIN0A
IEF237I 04A9 ALLOCATED TO LOGIN01
IEF237I 04A9 ALLOCATED TO LOGIN02
IEF237I 04A9 ALLOCATED TO LOGIN03
IEF237I 04A9 ALLOCATED TO LOGIN04
IEF237I 04A9 ALLOCATED TO LOGIN05
IEF237I 04A9 ALLOCATED TO LOGIN036
IEF142I DFSILTA4 ILTA - STEP WAS EXECUTED - COND CODE 0000
IEF285I USRT001.DFSILTA4.JOB00386.D0000102.? SYSOUT
IEF285I USRT001.DFSILTA4.JOB00386.D0000103.? SYSOUT
IEF285I USRT001.DFSILTA4.JOB00386.D0000104.? SYSOUT
IEF285I USRT001.DFSILTA4.JOB00386.D0000101.? SYSIN
IEF285I IMSTESTL.SLDSP.IMSA.TYTY KEPT
IEF285I VOL SER NOS= DSHR03.
IEF285I IMSTESTL.SLDSP.IMS1.TYTY KEPT
IEF285I VOL SER NOS= DSHR03.
IEF285I IMSTESTL.SLDSP.IMS2.TYTY KEPT
IEF285I VOL SER NOS= DSHR03.
IEF285I IMSTESTL.SLDSP.IMS3.TYTY KEPT
IEF285I VOL SER NOS= DSHR03.
IEF285I IMSTESTL.SLDSP2.IMS1.TYTY KEPT
IEF285I VOL SER NOS= DSHR03.
IEF285I IMSTESTL.SLDSP2.IMS2.TYTY KEPT
IEF285I VOL SER NOS= DSHR03.
IEF285I IMSTESTL.SLDSP2.IMS3.TYTY KEPT
IEF285I VOL SER NOS= DSHR03.
IEF373I STEP/ILTA /START 2007156.1250
IEF374I STEP/ILTA /STOP 2007156.1250 CPU 0MIN 00.02SEC SRB 0MIN 00.01SEC VIRT 16K SYS 296K EXT 808K SYS 11924K
IEF285I IMSTESTL.TNUC0 KEPT
IEF285I VOL SER NOS= USER01.
IEF285I IMSBLD.I10RTS17.CRESLIB KEPT
IEF285I VOL SER NOS= MVSS16.
IEF375I JOB/DFSILTA4/START 2007156.1250
IEF376I JOB/DFSILTA4/STOP 2007156.1250 CPU 0MIN 00.02SEC SRB 0MIN 00.01SEC
SPECIFIED START TIME IS 16:53:47.3
TO END OF FILE PROCESSED 9 TRANSACTIONS THIS RUN
REPORT CONTAINS 14 COMPLETE RECORD SETS
IMS LOG DATA FOR IMS3 STARTS AT TIME = 12:47:41.8, DATE = 2007.156
JOB STEP DR CLASSES
NAME NAME ID *****
MPP1 MPP 1 1 2 3 4
MPP2 MPP 1 1 2 3 4
MPP3 MPP 1 1 2 3 4
IMS LOG DATA FOR IMS2 STARTS AT TIME = 12:46:39.5, DATE = 2007.156
IMS LOG DATA FOR IMS1 STARTS AT TIME = 12:34:26.8, DATE = 2007.156
IMS LOG DATA FOR IMSA STARTS AT TIME = 12:36:56.8, DATE = 2007.156
SEQ TRANS P C ***IN*** ****OUT** P PGM DR SMB*ENQ MSG*SCHD CNT*ENQ MSG*END CNT*GU SYS IN Q PROC OUT Q TOTAL
NBR CODE R L LTERM LTERM T NAME ID HHMMSST HHMMSST HHMMSST HHMMSST HHMMSST ID SSSST SSSST SSSST SSSST
-----
00001 CHKPT 0001***** 1234270 1234270 1234270 1234270 1234270 1
00002 CHKPT 0001***** 1236569 1236569 1236569 1236569 1236569 A
00003 CHKPT 0001***** 1239165 1239165 1239165 1239165 1239165 2
00004 CHKPT 0001***** 1240111 1240111 1240111 1240111 1240111 3
00005 CTRLA02 CTRLA02 M 1241134 1241135 2 2 1
00006 CTRL CTRL M 1241260 1241263 3 3 2
00007 CTRLA02 CTRLA02 M 1241398 1241401 2 2 2
00008 CTRL CTRL M 1241523 1241526 3 3 2
00009 CTRL CTRL M 1242176 1242179 3 3 3
00010 CTRL CTRL M 1242240 1242244 3 3 3
00011 TRAN21V0 1 1 L62MVS1 L62MVS1 S PGM2V0 1 1242411 1242411 1242413 1242414 131 0 2 1 3 3
00012 SMQR21C0 1 1 L62MVS1 L62MVS1 S PGM2C0 1 1242419 1242419 1242421 1242421 1242424 131 0 1 3 5
00013 CTRLA01 CTRLA01 M 1242582 1242585 1 1 2
00014 CTRLA01 CTRLA01 M 1243136 1243138 1 1 1
00015 SMQR21C0 1 1 L62MVS1 L62MVS1 S PGM2C0 1 1243147 1243147 1243149 1243150 131 0 1 1 3
00016 SMQR21C0 1 1 L62MVS1 L62MVS1 T PGM2C0 1 1243279 1243279 1243281 1243281 13 0 1 2

```

```

00017          CTRL      CTRL      M          1243450      1243453 3 3          2
00018 CONV21V0 1 1 L62MVS1 D CPGM2V0 1 1243498 1243499 1243506 13 0 7 8
00019 CONV21V0 1 1 L62MVS1 C CPGM2V0 1 1243545 1243545 1243546 1243546 1243546 133 0 0 0 *****
00020 CONV21V0 1 1 L62MVS1 L62MVS1 C CPGM2V0 1 1243577 1243577 1243577 1243577 1243577 133 0 0 0 *****
00021 CONV21V0 1 1 L62MVS1 L62MVS1 C CPGM2V0 1 1244049 1244049 1244049 1244049 1244049 133 0 0 0 *****
00022 CONV21V0 1 1 L62MVS1 L62MVS1 C CPGM2V0 1 1244080 1244080 1244081 1244081 1244081 133 0 0 0 *****
00026 CHKPT 0002***** 1248342 1248342 1248342 1248342 1248342 A
00027 CHKPT 0002***** 1249068 1249068 1249068 1249068 1249068 1
00028 CHKPT 0002***** 1249365 1249365 1249365 1249365 1249365 2
00029 CHKPT 0002***** 1250062 1250062 1250062 1250062 1250062 3

```

表 87. ログ分析報告書の行形式

識別	開始位置	長さ	注記
シーケンス番号	1	5	1
トランザクション・コード	7	8	
トランザクションの優先順位 (PR)	16	1	
トランザクション・クラス (CL)	18	3	
入力 LTERM 名	22	8	
出力 LTERM 名	33	8	
処理タイプ (PT)	44	1	2
プログラム名	46	8	
従属領域 ID	55	3	
SMB エンキューの時刻 (受信したトランザクシ ョン)	59	7	3
メッセージのスケジュール時刻または GU の時刻	68	7	3
CNT エンキューの時刻 (出力キューに入ったメッ セージ)	77	7	3
プログラム終了の時刻または次のメッセージ GU	86	7	3
CNT GU時刻 (端末への出力メッセージの開始)	95	7	3
システム ID (LOGINxxx DD ステートメントか ら)	103	3	
入力キューに入っていた時間	106	6	4、5
処理時間	113	6	5、6
出力キューに入っていた時間	120	6	5、7
合計時間	127	6	5、8

表の凡例:

- 開始位置の 1 は、ディスクに報告書を作成するときフィールドの開始位置を変更する紙送り制御文字です。
- 処理タイプ
 - A** 異常終了したトランザクション
 - C** 会話型送信/受信処理
 - D** 伝送のみの会話型処理
 - F** /FORMAT を入力 (トランザクション・コード・フィールドに入っているのは MODNAME)
 - M** メッセージ通信
 - O** 領域占有 (開始チェックポイントに出会う前に入力キューに入っていた

トランザクションを処理中のプログラム、またはリカバリー不能なメッセージでスケジュールされたプログラムによって領域は占有されます。)

- P プログラム間通信送信/受信処理
- Q 伝送のみのプログラム間通信処理
- S 送信/受信処理
- T 伝送のみの処理
- X 会話型プログラム間通信、送信/受信処理
- Y 伝送のみの会話型プログラム間通信処理

3. HHMMSST の時刻

4. 入力キューに入っていた時間は、SMB エンキューからメッセージのスケジュールまでです。
5. 時間 SSSST または OVRFLW (合計の秒数がフィールド・サイズを超えると、OVRFLW が印刷されます。)
6. 入力待ち (WFI) のシステム・オプションを使用した場合、処理時間フィールドにはトランザクション間の待ち時間も含まれています。
7. 出力キューに入っていた時間は、CNT エンキューから CNT GU までです。
8. 合計時間は、SMB エンキューから CNT デキューまでです。合計時間は、トランザクション全体に及びます。

IMS 会計情報

会計方式の種類は、データ処理インストール・システムによって非常に異なってきます。IMS Transaction Manager には特別の難しさがあります。制御領域と従属領域の共同作業で多数の各種トランザクションが処理されるからです。

さらに、操作上個別のアプリケーションが、並行して処理される場合もあります。

IMS 専用プロセッサを備えたインストール・システムでは、ハードウェアとサポート機能の全体的なコストは、関係グループによる実際の使用量と予測使用量に基づいて請求されることがよくあります。共用システムでは、プロセッサ使用量が、比例コストの基礎になります。

IMS には明示的な会計機能はありませんが、処理アクティビティーを構成する個々のイベントは、かなり詳細に IMS ログに記録されます。IMS ログ・レコードの分析は、チャージバック・アルゴリズムの基礎として使用できます。例えば、統計分析ユーティリティーから、トランザクション数と、各トランザクションの平均 DL/I 呼び出し数の報告書を入手できます。

リソース使用率を示す数字の元になるもう 1 つのものは、IMS モニターのデータ収集の結果として作成される各報告書です。処理アクティビティーのサンプルは、定期的にとることができます。会計アルゴリズムは、例えば、プログラムによるプロセッサ使用率を使用できます。

上記の方法のいずれも、リソース標識のさらなる操作および調整を必要とします。

DL/I アドレス・スペース・オプション (LSO=S) を使用すると、SMF データに基づく会計プロシージャは影響を受けます。IMS システム・データ・セットおよび高速機能データベースに関する SMF 統計は、制御領域プロシージャに属することになります。全機能データベースは、DL/I アドレス・スペース・プロシージャに属することになります。

アプリケーション会計報告書の使用

統計分析ユーティリティは、計算機料金を算定するのに使用できるアプリケーション会計報告書を作成します。次のような内訳が、トランザクションごと、およびプログラムごとに用意されています。

- メッセージ数と、それに関連するプロセッサ時間の合計と平均 (単位は秒)
- DL/I メッセージ呼び出しの回数とタイプ
- DL/I データベース呼び出しの回数とタイプ

以下の例は、出力形式を示しています。

```

APPLICATION ACCOUNTING REPORT DATE 06/05/07 PAGE 00005
PROGRAM TRANSACTION MESSAGE --- COUNTS DATA --- BASE --- COUNTS CC OR RC TOT PROG AVG
NAME CODE PRI QTY GU GN ISRT GU GN GNP GHU GHN GHNP ISRT DLET REPL NOT 0 CPU TIME TIME
CPGM2V0 CONV21V0 1 5 5 10 9 0 0 0 0 0 0 0 0 0 0 0 0.05 0.0005
PGM2C0 SMQR21C0 1 3 3 3 2 0 0 0 0 0 0 0 0 0 0 0 0.05 0.0005
PGM2V0 TRAN21V0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0.05 0.0005
ICE143I

```

IMS トランザクション・プロファイルの使用

使用率を評価するための基礎として、各トランザクションの複合ピクチャーを使用できます。IMS トランザクション・プロファイルには、呼び出しタイプ別の DL/I 呼び出し要件が含まれ、他の処理ブロックのパス長から得られる項目も含まれることがあります。トランザクションによる DL/I 使用量が多いことを考慮して、メッセージ・カウントを操作することができます。トランザクション統計は、/DISPLAY 出力から定期的に、例えば、1 日の終わるかシャットダウン前に入手することができます。

プロファイルには、増加傾向および予測負荷からの大きな逸脱が、その原因となったトランザクション・コードまでトレースできるように、IMS の作業負荷の特性を記述してください。

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

この情報では、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それに関連する情報と、汎用プログラミング・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ため、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」というマーキ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。

汎用プログラミング・インターフェースにより、お客様は IMS のサービスを取得するプログラムを作成することができます。汎用プログラミング・インターフェースと関連ガイダンス情報は、それらが記載されているセクションまたはトピックの冒頭で示されているか、「汎用プログラミング・インターフェース」というラベルで示されています。

商標

IBM、IBM ロゴおよび [ibm.com](http://www.ibm.com)[®] 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 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す。

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 15 ライブラリーのすべての資料が記載されています。

表題	頭字語	資料番号
IMS V15 アプリケーション・プログラミング	APG	SC27-6778
IMS V15 アプリケーション・プログラミング API	APR	SC27-6779
IMS V15 コマンド 第 1 巻: IMS コマンド A-M	CR1	SC27-6780
IMS V15 コマンド 第 2 巻: IMS コマンド N-V	CR2	SC27-6781
IMS V15 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	SC27-6782
IMS V15 コミュニケーションおよびコネクション	CCG	SC27-6783
IMS V15 データベース管理	DAG	SC27-6784
IMS V15 データベース・ユーティリティー	DUR	SC27-6785
IMS Version 15 Diagnosis	DGR	GC27-6786
IMS V15 出口ルーチン	ERR	SC27-6787SC43- 3856
IMS V15 インストール	INS	SC27-6788
IMS Version 15 Licensed Program Specifications	LPS	GC27-6799
IMS V15 メッセージおよびコード 第 1 巻: DFS メッセージ	MC1	GC27-6789
IMS V15 メッセージおよびコード 第 2 巻: DFS 以外メッセージ	MC2	GC27-6790
IMS V15 メッセージおよびコード 第 3 巻: IMS 異常終了コード	MC3	GC27-6791
IMS V15 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	GC27-6792
IMS V15 オペレーションおよびオートメーション	OAG	SC27-6793
IMS V15 リリース計画	RPG	GC27-6794
IMS V15 システム管理	SAG	SC27-6795
IMS V15 システム定義	SDG	GC27-6796
IMS V15 システム・プログラミング API	SPR	SC27-6797
IMS V15 システム・ユーティリティー	SUR	SC27-6798

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]

アーカイブ

オンライン・ログ・データ・セット

DBRC の使用 627

カスタマイズ 119

ログ・レコード

カスタマイズ 119

自動 116

手動プロセス 116

OLDS 110

アーカイブ, OLDS 86

アクセシビリティ

キーボード・ショートカット xiv

機能 xiv

アクセス管理方式

考慮事項 434

アクセス方式サービス・プログラム

(AMS) 721

REPRO コマンド 716

アクティブ IMS

監視信号の送信 758, 774

処理 762

代替 IMS システムによるトラッキング

758, 774

立ち上げる 771

テークオーバー時のプロシージャ

762, 776

トラッキング・フェーズ中のプロセス

747, 774

プロセスのサイクル 789

USERVAR テーブルの項目を初期設定

する 763

アクティブ IMS システム

処理 762

定義 745

圧縮

PRILOG 717

宛先コード 510

オンライン変更を伴う 510

宛先作成出口ルーチン (DFSINSX0)

共用キュー

動的制御ブロック 264

プログラム間通信 263

メッセージ通信 263

宛先作成出口ルーチン (DFSINSX0) (続き)

未定義のトランザクション

動的制御ブロック 264

アドレス・スペース

作業負荷管理 469

相対的重要度 469

ディスパッチング優先順位 469

IMS, CSL 内の 44

Repository Server 44

アプリケーション

逐次

共用キュー環境での 265

変更

アクティブ IMS システムでの 504

アプリケーション ID 名

汎用リソース環境内 359

アプリケーション会計報告書

会計に使用 974

例 974

アプリケーション制御ブロック (ACB)

(application control block (ACB))

オンラインの変更 591

オンライン変更 591

活動化, データ共用環境での ACB の

559

管理対象 ACB

活動化, データ共用環境での ACB

の 559

データベース・リソースの活動化、

概要 554

管理対象 ACB 環境での活動化、概要

554

パフォーマンスのための配置 520

ライブラリー

オンラインでのメンバーの変更 591

メンバー・オンライン変更 591

割り振り 505

ライブラリー・データ・セット

動的割り振り 35

動的割り振り (dynamic

allocation) 35

64 ビット・ストレージへの配置 35

ACB の IMS 管理

活動化, データ共用環境での ACB

の 559

データベース・リソースの活動化、

概要 554

アプリケーション設計

パフォーマンスの考慮事項 537

アプリケーションの処理意図 295

アプリケーションの文書化 93

アプリケーションの変更

アクティブ IMS システムへの導入

504

システムに与える効果についての計画

502

アプリケーション変更の管理、チェックリ

スト 502

アプリケーション・プリロード

指針 533

アプリケーション・プログラム

オンライン変更 509

コミット・ポイント (commit

point) 107

サポートするデータ・セット 35

使用不能なデータ

使用不能なデータに対するセンシテ

ィビティ 82

使用不能なデータに対するセンシテ

ィビティ 82

スケジューリング

使用不能なデータに対する 82

設計

パフォーマンスの考慮事項 537

設計の検討 94

データ共用への関与 295

同期点 107

メッセージ・ドリブン 84

CPI-C ドリブン

セキュリティに関する考慮事項

409

INIT 呼び出し 82

ODBA

セキュリティに関する考慮事項

409

XRF の使用 756

アプリケーション・プログラムのデッドロ

ック

ブロック・レベル共用を持つ 300

暗号 437

暗号化

セグメント編集/圧縮出口の使用 437

IMS でのデータ・セット暗号化のサポ

ート 375

VTAM 端末 437

異常終了 88

異常終了コード

Base Primitive Environment (BPE) 37

異常終了フォーマット設定モジュール

アンインストール 157

バインディング 157

- 異常終了フォーマット設定モジュール (続き)
 - DFSAFMD0 156
- 一般的な IWAIT 時間イベント報告書
 - IMS モニター (DB/DC) 885
 - IMS モニター (DCCTL) 940
- 意図障害要約報告書
 - IMS モニター (DBCTL) 909
 - IMS モニター (DB/DC) 876
- 違反の制御 444
- イベント
 - システム・レベルのトレース 482
- イベント通知機能 223
- イメージ・コピー
 - 数、指定 665
 - ファジー 659
- イメージ・コピー (image copy)
 - データ共用 (data sharing) 322, 326
 - 定義 131
 - 排他制御を使用して 325
- イメージ・コピーとデータ共用 296
- イメージ・コピーの除去時刻 122
- イメージ・コピー・グループ・レコード 712
- イメージ・コピー・データ・セット
 - 重複
 - 命名規則 657
 - 命名規則 656
 - リカバリー期間 665
- イメージ・コピー・データ・セットの暗号化 380
- 印刷出力 605
- 印刷ダンプ出口制御表 161
- インストール
 - サービス
 - 重要な注意 246, 251
 - 予防 245
 - 作業
 - データ共用 (data sharing) 306
 - 問題、予防 147
 - IMS のための考慮事項 147
 - z/OS インターフェース 147
- インストールに関する問題 253
- インターバル値
 - 設定 768
- インターフェースの考慮事項、重要な注意 147
- インターフェース・モジュール 152
- インタレスト
 - インタレストの登録
 - 概念 260
- インタレストの登録
 - 共用キュー
 - LTERM に関する 262
 - MSC (複数システム結合機能) リソースに関する 262
- インタレストの登録 (続き)
 - 共用キューに関する
 - 概念 260
 - トランザクションに関する 261
- 運用
 - 概要 101
 - 自動化 102
- エラー
 - アプリケーション・ロジック 139
 - オペレーション上の 139
 - 入力 139
 - INITIATE OLC コマンド 590
- エラー、RECON 入出力 721
- エリア・データ・セット
 - 開始、RECON データ・セットの 647
 - XRF 828
- エリア・データ・セットの登録
 - RECON データ・セットへの 647
- エリア・レベルのデータ共用 291
- エレメント名、ARM 165
- エンキュー問題の原因、RECON の 738
- エンキュー/デキュー・テーブル
 - 使用 88
- エンド・ユーザー・サービス
 - テークオーバー時 747
 - XRF テークオーバー時 750
- オーバーフロー
 - しきい値 218
 - 処理 218
 - モード 218
- オーバーフロー順次アクセス方式 (OSAM)
 - キャッシュ構造
 - サイズの変更 348
 - 構造
 - サイズの計算 347
- オーバーフロー処理
 - 構造 76
- 応答時間
 - 基準の定義 466
 - ユーザーを基準にした目標 466
 - XRF テークオーバー (XRF takeover) 798
- オフライン・ダンプ・フォーマット設定モジュール
 - DFSOFMD0 158
- オンライン更新プログラム
 - 復元 323
- オンライン再編成 (online reorganization)
 - 処理の説明 681
 - 推奨 681
 - 制約事項 681
 - 保守、リカバリー・レコードの 681
 - レコード、使用される 678
 - DBRC コマンド、サポートされる 678
 - REORG レコード 678
- オンラインでのリソース変更 551
- オンライン変更
 - オプション 551
 - オンライン変更機能 569
 - 管理対象 ACB 環境 553
 - グローバル
 - DBCTL 待機緊急時再始動後 584
 - データベース
 - 管理対象 ACB 環境 553
 - プログラム・ビュー
 - 管理対象 ACB 環境 553
 - ACB の IMS 管理 553
 - ACB、IMS 管理対象 553
 - DBDs
 - 管理対象 ACB 環境 553
 - IMPORT DEFN SOURCE (CATALOG) 553
 - PSBs
 - 管理対象 ACB 環境 553
- オンライン変更 (online change)
 - 開始 588
 - グローバル 55
 - エラー 594
 - コールド・スタート (cold start) 601
 - コマンド・シーケンス 578
 - シナリオ 595
 - 終了 593
 - ACB ライブラリー・メンバー 57, 591
 - IMSplex 577
 - XRF テークオーバー後 584
- グローバル状況 202
- コールド・スタート (cold start) 601
- コマンド・シーケンス 573
- システム定義 (system definition) 574, 579
- 実施できるシステム定義の変更 509
- 準備コマンドの失敗
 - シナリオ 599
 - タイムアウト・エラー 599
- 制御 321
- セキュリティー変更用 433
- トランザクション 575, 580
- パフォーマンスの考慮事項 577, 584
- プログラム 575, 580
- ローカル
 - コールド・スタート (cold start) 601
 - コマンド・シーケンス 573
 - シスプレックス (sysplex) 585
- ACB メンバー
 - 関連付け 582
 - DRD コマンドによる関連付け 582
- DBCTL のセキュリティー変更 443
- DEADB 575, 581

オンライン変更 (online change) (続き)
HALDB (高可用性ラージ・データベース) 576, 581
IMS 管理の ACB 環境 555, 557, 562, 563
XRF 577, 583
XRF (拡張回復機能)
RM のない IMSplex の制約事項 756
XRF 複合システムにおける 828
オンライン変更、ローカル
コマンド 573
オンライン変更機能
概要 569, 570
FDBR 576, 583
オンライン変更コマンド
INITIATE OLC 578
QUERY 578
TERMINATE OLC 578
オンライン変更データ・セット 570
オンライン変更のための出口ルーチン 510
オンライン変更ユーティリティ
(DFSUOCU0) 30
XRF での IMS データ・セットのコピーの保持 828
オンライン・システム
定義 5
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) 663
作成、将来の利用のためのデータ・セットの 664
実行、DBRC により記録される 658
説明 659
オンライン・データ・セット (OLDS)
テスト環境でのクローズ 737
オンライン・テスト
システム・テスト中 460
バッチ端末シミュレーターの使用 461
MFS 形式 461
MFSTEST モード 461
オンライン・ログ・データ・セット (OLDS) (online log data set (OLDS))
アーカイブ 110
アクセス方式 110
数 126
環境 110
クローズ、WADS からの 111
再使用 116
サイズ 126
装置 128
重複または単一 125
低下モード・ロギング 125
定義 108, 110, 125
停止 110
特性の変更 129

オンライン・ログ・データ・セット (OLDS) (online log data set (OLDS)) (続き)
バッファ 129
ブロック・サイズ 127
レコード 705
割り振り 110
RECON 110
オンライン・ログ・データ・セット (OLDS) の暗号化 380

[力行]

開始
コールド
実行 601
データ共用処理 310
IRLM サブシステム 310
解釈
ログ・トランザクション分析報告書 967
IMS モニター報告書
DBCTL 903
DCCTL 925
//DFSSTAT 報告書 951
解釈テーブル
初期設定 763
例 763
XRF による使用 763
回線および端末報告書
例 968
回線機能報告書
IMS モニター (DB/DC) 889
IMS モニター (DCCTL) 942
回線障害
XRF 制限 750
外部サブシステム
データ・キャプチャー出口ルーチン 90
外部サブシステム接続機能 (ESAF) 14
JMP または JBP アプリケーションの使用 615
外部システム
アクセス 615
概要 91
会話型トランザクション
共用キュー環境での 266
会話モード
IMSplex、内 373
拡張エラー・キュー・エレメント (EEQE)
XRF 複合システムでの 786
拡張回復機能 (XRF) 782
アクティブ IMS
障害 745
定義 745
アクティブ IMS からの信号
IMS システム・ログ 758

拡張回復機能 (XRF) (続き)
アクティブ IMS からの信号 (続き)
ISC リンク 758
RDS 758
アプリケーション・プログラム 756
オンライン変更
RM のない IMSplex の制約事項 756
概念 745
概要 743
監視
オブション 819
開始 758
確立 768
停止 758
変更 758
監視のための LNK パラメーター 819
監視のための LOG パラメーター 819
監視のための NO パラメーター 819
監視のための RDS パラメーター 819
強制テークオーバー、VTAM による 763
クラス 1 端末 800
所有権 810
クラス 2 端末 802
クラス 3 端末 803
グローバル・リソース・シリアライゼーション 812
計画 798, 812
コンポーネントの役割 758
システム定義 814
シスプレックス (sysplex)
グローバル・オンライン変更 583
制限 798
操作要件 756
ソフトウェア要件 743
代替 IMS システム 745
バックアップ・セッション 750
端末 800
端末切り替えに影響を与える VTAM
所有権 810
テークオーバー (takeover)
基準 768
原因 745
定義 745
ユーザーの視点 805
テークオーバー条件 819
データ・セットの配置 828
トラッキング・フェーズ
従属領域の状況 774
通信ネットワークの状況 774
データベースの状況 774
メッセージ・キューの状況 774
MFS プールの状況 774
ハードウェア要件 756
パスワードの IMS への定義 816

拡張回復機能 (XRF) (続き)

バックアップ・オプションの指定 804
バックアップ・セッション 750
複合システム
 初期設定フェーズ 771
 説明 747
 同期化フェーズ 773
プロセスのフェーズ 771, 789
編成 789
元のアクティブ・システムで 実行される作業
 コードの保守 785
 ハードウェア構成変更 785
 ハードウェア保守の実行 785
バックアップ IMS プロシージャのテスト 785
予防保守 785
ライブラリー保守の実行 785
要件 747
 ソフトウェア 756
 ハードウェア 756
用語 745
ライセンス・プログラム要件 756
リカバリー可能サービス・エレメント (RSE) 745, 819
利点 744
 IMS 758
ローカル・キュー・マネージャー・データ・セット 264
ログオン 758
1 つの CPC を持つ複合システム 790
1 つの XRF 複合システム、2 つの CPC、XRF のない IMS の例 792
1 つの複合システム、2 つの CPC の例 791
2 つの複合システム、3 つの CPC の例 793
2 つの複合システム、4 つの CPC の例 796
APPLID = キーワード 816
ARM に関する考慮事項 812
DBCTL 機能 750
DBRC に対する推奨 774
DFSHSBxx パラメーター 819
DSE 745
DSE (従属サービス・エレメント) 定義 758
HSBID パラメーター 814
HSBMBR パラメーター 814
IMS 2 次端末 817
IMS の VTAM への定義 827
IMS マスター端末 817
IMSpIex 744
 RM のないオンライン変更の制約事項 756
IMSpIex 内 25

拡張回復機能 (XRF) (続き)

IMS.PROCLIB メンバーの指定 819
ISC リンク 799
JES に関する考慮事項 812
MNPS と USERVAR の比較 749
MNPS パラメーター 814
MNPSPW パラメーター 814
MNPSPW= キーワード 819
MNPS= キーワード 819
MSC リンク 818
NCP
 バックアップ・セッション 750
NCP の計画 811
NCP の利点 768
RACF に関する考慮事項 812
RSENAME= キーワード 819
SSP の利点 768
USERVAR テーブル、VTAM
 IMS の VTAM への定義 827
USERVAR パラメーター 814
USERVAR=キーワード 819
VTAM
 強制テークオーバー 763
 バックアップ・セッション 750
VTAM アプリケーション名の IMS への定義 816
XRF に対するシステム定義マクロ 815
XRF の SWITCH= キーワード 819
XRF プロセスに対する DFSMS の利点 763
XRF プロセスに対する VTAM の利点 763
z/OS の利点 762
拡張回復機能 (XRF))
 ネットワーク変更、開始 781
XRF (拡張回復機能)
 ネットワーク変更、開始 781
拡張プログラム間通信機能/IMS 79
仮想記憶アクセス方式 (VSAM)
 キャッシュ構造
 サイズの変更 348
構造
 サイズの計算 347
仮想記憶通信アクセス方式 163
 XRF 要件 756
カタログ管理、RECON 内のデータ・セットの 736
カップリング・ファシリティ
 構造のモニター 448
 シスプレックス・データ共用 (sysplex data sharing) 338
カップリング・ファシリティ (coupling facility)
 構造
 サイズの変更 348
構造、サイズの計算 345

カップリング・ファシリティ (coupling facility) (続き)

構造障害 351
構造のモニター 315
再作成での障害 352
接続の障害 349
リソース構造 (resource structure) 60
CFRM ポリシーの定義
 例 270
IMS 再始動により障害 352
IMSpIex の一部 23
カップリング・ファシリティ・リソース管理 (CFRM)
 結合データ・セット・フォーマット・ユーティリティ 225
ポリシー、定義の 214
可用性マネージャー (AVM)
 オペレーターへのメッセージの送信 762
 オペレーター・メッセージ 762
定義 762
 入出力防止の実行 762
可用性マネージャー (AVM) (availability manager (AVM))
 立ち上げる 771
監査証跡 183, 184
監査ログ
 OM
 レコード・フォーマット 185
監視
 アクティブ IMS の 774
 XRF オプション 819
監視のための LNK パラメーター 819
監視のための NO パラメーター 819
監視プログラム呼び出し 153
監視メカニズム
 開始、XRF 758
 テークオーバーを発生させる信号がない 776
 停止、XRF 758
 変更、XRF 758
 XRF テークオーバーを発生させる信号がない 758
 XRF に対する確立 758
 XRF に対する定義 758
 XRF の例 758
管理
 概要図 3
システム
 概念 35
システム、概要 3
ODBA アプリケーション・プログラムセキュリティに関する考慮事項 442
キーボード・ショートカット xiv

- キーワード
 - MAXAPPL 163
- 記述子定義
 - IMSRSC リポジトリ
 - 表示 233
- 機能強化
 - IMS でのデータ・セット暗号化のサポート 375
- 基本プリミティブ環境 (BPE) (Base Primitive Environment (BPE))
 - 外部データ・セットへの書き込み 40
 - 外部トレースの開始 42
 - 外部トレースの停止 42
 - 外部トレース・データ・セットの定義 41
 - 構成 37
 - コマンド 37
 - 提供されたサービス 37
 - トレース
 - 使用可能化 39
 - 内部トレース・テーブルへの書き込み 40
 - メッセージ 37
 - ユーザー出口ルーチン 37
 - BPE と IMS の関係 37
 - BPE を使用するコンポーネント 37
 - CSL との関係 43
 - IMSplex の一部 23
 - z/OS PPT entry 150
 - z/OS PPT の項目 150
- 逆方向リカバリー
 - 定義 136
- キュー (queue)
 - 移動 74
 - コールド 74
 - 構造 74
 - 削除 74
 - ロック 74
 - control 74
 - type
 - 値 74
 - クライアント 74
 - 専用 74
 - CQS によって管理される専用 74
- キュー制御機能 (QCF)
 - コールド・キュー
 - 再キューイング 278
 - 構造
 - 再キューイング 278
 - モニター 278
- キュー・タイプ
 - 共用キュー環境での 260
- キュー・マネージャー
 - トレース報告書 543
 - の設計 264
 - ページ固定 520
- キュー・マネージャー (続き)
 - XRF 環境での 264
- 急送メッセージ処理 84
- 競合
 - RECON データ・セットでの回避 692
- 競合の除去
 - 入出力リソース
 - チューニング 541
- 強制テークオーバー
 - XRF 763
- 共通キュー・サーバー (CQS)
 - IMS コマンドによるアクセス 279
- 共通サービス層 (CSL) (Common Service Layer (CSL))
 - アドレス・スペース 47
 - 概要 43
 - 管理 165
 - 構成、Resource Manager のない 47
 - 構成の推奨 61
 - コマンド処理に関する考慮事項 181
 - 最小の構成 61
 - 自動 RECON 損失通知 58
 - 障害 835
 - マネージャー 47
 - 利点 43
 - IMS アドレス・スペース、組み込まれた 43
 - IMSplex の一部 23
 - ODBM (Open Database Manager)
 - 概要 47
 - Open Database Manager (ODBM)
 - 概要 47
 - Operations Manager
 - 概要 50
 - Resource Manager
 - 概要 50
 - 提供される機能 50
 - リソース構造 (resource structure) 50
 - Structured Call Interface
 - 概要 51
 - 提供される機能 51
- 業務の重要度
 - ワークロード・マネージャー 468
- 共用キュー
 - インタレストの登録
 - 概念 260
 - 概念 259
 - キュー・タイプ 260
 - 共通キュー・サーバー (CQS) 259
 - バックエンド IMS 259
 - フロントエンド IMS 259
 - IMS がトランザクションに関するインタレストを登録および登録解除する場合 261
- 共用キュー (続き)
 - 会話型トランザクション
 - 概念 266
 - 環境
 - パフォーマンスの計画 538
 - AOI トランザクション 287
 - APPC メッセージ 284
 - MSC (複数システム結合機能) の計画 287
 - OTMA メッセージ 284
 - キューでのメッセージのロックング
 - 概念 262
 - 区分構成 267
 - 構成 266
 - 高速機能メッセージ・キュー・リスト
 - 構造
 - 定義 270
 - 例 270
 - 自動ログオン端末 263
 - 使用可能化 270
 - 逐次アプリケーション処理 265
 - 逐次トランザクション処理 265
 - チューニング・パフォーマンス 288
 - 構造 288
 - パラメーター 288
 - 名前の一意性 367
 - の設計 259
 - 複製構成 266
 - プログラム間通信
 - 宛先作成出口ルーチン (DFSINSX0) 263
 - 未定義のトランザクション
 - 宛先作成出口ルーチン (DFSINSX0) 264
 - メッセージ通信
 - 宛先作成出口ルーチン (DFSINSX0) 263
 - メッセージ・キュー・リスト構造
 - 定義 270
 - 例 270
 - メッセージ・フロー
 - 概念 262
 - リソース構造 (resource structure)
 - 定義 270
 - リモート処理
 - 定義 259
 - CFRM ポリシーの定義 270
 - 例 270
 - CQS
 - グローバル構造定義 273
 - 実行パラメーター 273
 - 初期設定パラメーター 273
 - ローカル構造定義 273
 - CSL がいない場合 61
 - IMS 制御領域実行パラメーターの定義 273

共用キュー (続き)
 LTERM に関するインタレストの登録および登録解除
 概念 262
 MSC (複数システム結合機能) リソースに関するインタレストの登録
 概念 262
 TM リソース
 管理 367
 z/OS ログ・ストリーム
 定義 272
 共用データ 291
 共用副次索引データベース
 グローバル・リソースとしての状況 202
 共用レベル
 値の定義 642
 DBRC による割り当て 642
 DBRC、割り当て 642
 許可
 データベース、変更 734
 変更、データベースの 734
 切り替え装置
 XRF 828
 区画 DB レコード (DSPDBHRC)
 HALDB
 TYPE=PART 708
 区画 DBDS レコード (DSPDSHRC)
 HALDB
 DBDS のタイプ 708
 区画レコード (DSPPTNRC)
 HALDB
 PHDAM 708
 区分化
 共用キュー構成 267
 区分構成
 共用キュー 267
 クライアント
 キュー・タイプ (queue type) 74
 クラス 1 端末
 切り替えの優先順位の定義 817
 推奨される所有権 810
 テークオーバー時のバックアップ・セッションへの切り替え 750, 781
 テークオーバーによる影響 747, 750
 テークオーバーの後処理時のバックアップ・セッションへの切り替え 786
 定義 804
 定義、XRF に対する 747
 バックアップ・オプションの指定 804
 バックアップ・セッションのオープン 750
 ユーザーから見たテークオーバー 805
 XRF での所有権 810
 XRF のセッション・リカバリー 758
 XRF 複合システムでの 800
 クラス 2 端末
 セッション・リカバリーの優先順位の定義 817
 テークオーバー時の新しいセッションの確立 781
 テークオーバーによる影響 747, 781
 テークオーバーの後処理時の新しいセッションの確立 786
 定義、XRF に対する 747
 ユーザーから見たテークオーバー 805
 XRF のセッション・リカバリー 758
 XRF 複合システムでの 802
 クラス 3 端末
 テークオーバーによる影響 747, 781
 定義、XRF に対する 747
 XRF 複合システムでの 803
 グループ
 データベース 707
 グローバル状況
 システム障害時の可用性 207
 NULL にリセット 205
 グローバル情報
 シスプレックス (sysplex) 206
 グローバル呼び出し可能サービス 368
 グローバル・オンライン変更 (global online change)
 エラー 594
 開始 166
 概要 55, 569
 機能 55
 コールド・スタート (cold start) 601
 コマンド 578
 INITIATE OLC 588
 QUERY MEMBER 593
 TERMINATE OLC 578
 コマンド・シーケンス 578
 混合有効範囲 167
 サポートされた環境 551
 サポートされない環境 573
 シナリオ 595
 シャットダウン 167
 使用可能化
 IMSplex 166
 使用不可化
 IMSplex 166
 初期設定 29
 制約事項 166
 リソース構造なし 55
 RM なし 55
 保守 166
 ユーティリティ 55
 要件 55
 ACB ライブラリー・メンバー 57, 591
 DRD
 シナリオ 596
 IMSplex 577
 グローバル・オンライン変更 (global online change) (続き)
 使用可能化 166
 使用不可化 166
 MODBLKS=DYN
 シナリオ 596
 MODBLKS=OLC
 シナリオ 595
 TSO SPOC
 出力 595
 グローバル・オンライン変更ユーティリティ 601
 グローバル・オンライン変更ユーティリティ (DFSUOLC0) 30, 584, 586
 OLCSTAT の初期化 55
 グローバル・コマンド状況 202
 グローバル・リソース
 タイプ
 スケジュールされた逐次プログラム 201
 データベース名 201
 トランザクション 201
 ユーザー ID 201
 APPC 記述子 201
 DEADB エリア名 201
 LTERMS 201
 MSNAME 201
 グローバル・リソース情報 201
 グローバル・リソース・シリアライゼーションと XRF 812
 グローバル・リソース・シリアライゼーション・マクロ 695
 計画
 アプリケーション変更、例 502
 概要 101
 システム定義の変更 506
 計画テークオーバー (planned takeover)
 オペレーターの手順 785
 実際の用途 785
 使用 750
 定義 776
 例 786
 経過中トランザクション
 テークオーバー中の処理 780
 XRF に対する定義 758
 結合データ・セット・フォーマット・ユーティリティ 225
 検出、メッセージ処理のボトルネックの 971
 検討
 設計 94
 コールド・スタート (cold start)
 グローバル・オンライン変更 (global online change) 601
 実行 601
 実行するタイミング 602

コールド・スタート (cold start) (続き)
 テスト環境での複数の 737
 ローカル・オンライン変更 (local
 online change) 601
 高可用性ラージ・データベース (HALDB)
 オンライン変更 (online change) 576,
 581
 区画 DB レコード (DSPDBHRC) 708
 区画 DBDS レコード
 (DSPDSHRC) 708
 区画 (partition) 708
 区画選択出口ルーチン 91
 区画レコード (DSPPTNRC)
 PHDAM 708
 動的割り振りデータ・セット 36
 マスター (DSPDBHRC)
 DB ヘッダー・レコード 708
 命名規則 95
 DBDS のタイプ 708
 REORG レコード 712
 更新アクセス (update access) 296
 更新機能
 転送 324
 構成
 単純 47
 IMSplex
 一般的な 61
 共用キュー、CSL がある 61
 共用キュー、CSL がない 61
 混在 IMS バージョン 61
 最小の 61
 推奨 61
 単一システム 61
 DBCTL 61
 IMSRSC リポジトリを使用 61
 Resource Manager がない 61
 IMSplex、複数の SPOC ユーザーを含
 む 52
 構造
 オーバーフロー 221
 機能 218
 構造フル・モニター 221
 オーバーフロー処理 76
 機能 76
 コピー 225
 再移植 223
 再作成 76
 開始 223
 サイズ 214
 使用状況のモニター 219
 タイプ 74
 チェックポイント、開始 216
 二重化 76, 225
 使用可能化 225
 フル・モニター 220
 ペア 74

構造 (続き)
 への接続の許可 212
 変更 214
 リカバリー 224
 リソース 75
 構造化呼び出しインターフェース (SCI)
 (Structured Call Interface (SCI))
 概要 51
 管理 209
 構成要件 61
 提供される機能 51
 構造再作成 221
 構造のサイズ設定における z/OS の計算
 式 345
 構造の使用量
 管理 217, 219
 構造の使用量の管理 217, 219
 構造フル
 管理 217, 219
 構造フルしきい値の表示 220
 構造フルを防ぐ 217, 219
 構造を変更する 214
 高速機能
 アプリケーション・プログラム
 メッセージ・ドリブン 84
 急送メッセージ処理 84
 従属領域
 説明 84
 処理 84
 データベース
 DEDB 83
 トランザクション
 排他的 84
 メッセージ処理 84
 IFP 84
 高速機能 DEDB
 レコード、RECON データ・セットの
 708
 高速機能 (Fast Path)
 エリア・レベルのデータ共有 291
 オンライン変更の考慮事項 510
 概念と用語 83
 セキュリティに関する考慮事項
 DBCTL 441
 セキュリティの考慮事項
 DB/DC 408
 データベースおよび DEDB エリアの
 登録 642
 トランザクション
 推奨 289
 バッファ・プール 530
 メッセージ・キュー・リスト構造
 定義 270
 例 270
 モニター 487
 DBCTL に関する考慮事項 85

高速機能 (Fast Path) (続き)
 ISC および 84
 MSC および 84
 高速機能ログ分析ユーティリティー
 (DBFULTA0) 119, 445
 高速処理データベース 83
 高速処理データベース (DEDB)
 保管されたリソース情報 204
 高速処理データベース (DEDB) (data
 entry database (DEDB))
 レコード、RECON データ・セットの
 708
 構文検査機能
 開始、IMS Application Menu による
 47
 考慮事項
 システム管理 147
 コピー
 構造 225
 コマンド
 アクセス方式サービス・プログラム
 (AMS)
 REPRO 716
 監査証跡の作成 183, 184
 グローバル 202
 経路指定 181
 タイプ 2 43
 環境 78
 構成要件 78
 タイプ 2 コマンド環境
 RM のないオンライン変更の制約
 事項 55
 データ共有 (data sharing) 311
 複数リソース用 447
 要件 78
 リソースに与える影響 447
 ログ関連
 CHANGE.PRILOG 628
 CHANGE.RECON 628
 CHANGE.SECLOG 628
 DELETE.LOG 628
 GENJCL.ARCHIVE 628
 GENJCL.CLOSE 628
 LIST.LOG 628
 NOTIFY.PRILOG 628
 NOTIFY.SECLOG 628
 AO
 セキュリティ 401
 BACKUP.RECON 716
 CHANGE.PRILOG 627
 CHANGE.RECON、RECON のリカバ
 リー 721
 CHANGE.SECLOG 627
 CLEANUP.RECON
 削除、不要な RECON レコードの
 718

コマンド (続き)

- CQS 構造への接続の許可 213
- CQS 登録の許可 212
- CSL 内の処理に関する考慮事項 181
- DBCTL 環境 (DBCTL environment) 12
- DEFINE
 - GENERATIONDATAGROUP 41
- DELETE.LOG
 - 削除、不要な RECON レコードの 718
- DFSAPPL 141
- GENJCL 635
- GENJCL.ARCHIVE 627
- GENJCL.IC 678
- GENJCL.OIC 678
- INIT SELF 163
- INITIATE OLC 578
 - RM (リソース・マネージャー) を 使用しない 578
- INIT.ADS 647
- INIT.CAGRP 674
- INIT.DB 647
- INIT.DBDS 647, 665, 666
- INIT.RECON 692
- INIT.RECON、RECON データ・セッ トの確立 645
- INIT.RECON、RECON のリカバリー 721
- NOTIFY.IC
 - HALDB オンライン再編成の考慮 事項 678
- NOTIFY.UIC 658
 - HALDB オンライン再編成の考慮 事項 678
- QUERY 578
- QUERY OTMATI 486
- RECON データ・セットのための制限 683
- REPAIR.RECON 723
- REPRO 721
- TERMINATE 578, 593
- TRACE CT 483
- VTAM VARY 163
- /CHECKPOINT 87
- /DBDUMP データベース・バックア ヅプ・コピー 660
- /DISPLAY 573, 578
- /ERESTART
 - DBRC の障害後の再始動 653
 - IMS の障害後の再始動 653
- /MODIFY 573
- /NRESTART、IMS の障害後の再始動 653
- /OPNDST 163
- /RMGENJCL 635

コマンド (続き)

- /TRACE 480, 483
- コマンド許可
 - 説明 392
 - 入力コマンド・バッファ 393
 - DBRC、DSPDCAX0 の使用 685
 - DSPDCAX0 および RACF の使用 685
 - LU 6.2 393
- コマンド許可出口ルーチン
 - AO コマンド・セキュリティ 407
- コマンド・セキュリティ 188
 - コマンド許可出口ルーチン 407
 - システム定義 (system definition) 404
- コミットされていないデータベース変更を ロックする 774
- コミット・フェーズ 1
 - 障害
 - シナリオ 598
- コミット・ポイント (commit point) 106
 - アプリケーション・プログラム 107

[サ行]

サービス

- インストール 245, 252
 - 重要な注意 246
- 考慮事項 239
- サービス・レベルの保守 243
- 指針 242
- 実動システム
 - 推奨 242
 - 保守プロセスの例 244
- 修正サービス
 - インストール 245
 - プログラム診断依頼書 (APAR) 239
- 処理 239
- 推奨 242
- 推奨サービス・アップグレード (RSU) 241
- 説明 239
- 特別な考慮事項
 - 非 SYSDEF ターゲット・ライブラ リー 254
 - IVP 254
- 予防サービス
 - インストール 245
 - プログラム一時修正 (PTF) 239
- SYSGEN
 - SYSMOD 回帰、回避 253
- SYSMOD
 - 回帰、回避 253
 - SYSMOD のパッケージ化 240
- サービス定義 470, 472
- サービス・クラス 470, 472

サービス・レベル

- 実動システム
 - 推奨 242
 - 保守プロセスの例 244
- 保守 243
- 再始動
 - 概要 138
 - 緊急時
 - 定義 138
 - 試行失敗後のクリーンアップ 602
- 自動
 - 定義 138
- 正常
 - 定義 138
 - データ共有 (data sharing) 332, 649
 - ログの使用 86
- 再始動、データ共有における
 - 緊急時、DBRC の障害後 653
 - 緊急時、IMS の障害後 653
- 正常 653
 - DBRC の障害後 653
 - IMS 障害後の 653
- 再始動データ・セット 758
- 再始動データ・セット (RDS) (restart data set (RDS))
 - 定義 108
 - 内容 114
 - 割り振り 130
- サイズの計算、SSYS レコードの 733
- 最適化
 - アプリケーション・プログラムのロー ドの 533
 - スケジュール/終了 476
 - ディスクパッチング優先順位 519
 - 入出力の競合 548
 - ページ固定 520
 - メッセージ形式バッファ・プール 526
 - メッセージ形式ライブラリー 543
 - メッセージ領域の数 523
 - メッセージ・キュー 543
 - IMS システム・データ・セット 536, 543
 - IMS 領域のプロセス優先順位 541
 - PSB プールおよび DMB プール 526
- 再編成レコード 712
- 再利用、イメージ・コピー・データ・セッ トの 668
- サインオン検査
 - すべての静的端末の場合 390
 - セキュリティ 390
 - 出口ルーチン 421
 - DFSDCxxx PROCLIB メンバー 390
 - RACF 394
 - RACF パスチケット 421
 - VTAM 端末 390

- サインオン検査用の DFSCSGN0 モジュール 421
- サインオン出口ルーチン (DFSSGNX0) ETO を使用して 415
- 先書きデータ・セット (WADS) (write-ahead data set (WADS))
 - 環境 111
 - 定義 108
 - 割り振り 111
 - OLDS のクローズ 111
- 作業
 - システム管理 147
- 作業区分修飾子の定義 470, 472
- 作業単位 (UOW) (unit of work (UOW))
 - トラッキング
 - 共用キュー環境 263
- 作業負荷
 - 代替 XRF システムに与えるテークオーバーの影響 798
 - 代替 XRF システムの計画 798
 - モニター 465, 495
- 作業負荷管理
 - 操作モード
 - ゴール 471
 - 互換性 471
 - マイグレーション 471
- 索引/ILDS 再作成ユーティリティ (DFSPREC0)
 - DBRC に対する関係 636
- 削除、SSYS レコードの 733
- サブシステム
 - 接続 449
 - 切断 449
- サブシステム (SSYS) レコード
 - サイズの計算 733
 - 削除 733
 - 処理 733
 - IMS 再始動時の初期化 714
- サポートされるデータベース
 - DB/DC 環境 10
- サンプル 出口ルーチン
 - SMP/E アセンブル 256
- 時間制御操作 (TCO) 408
- 指針
 - アプリケーション・プログラム・ブロードの 533
 - IMS システム・データ・セットの配置の 536
- システム
 - 管理
 - 概要 3
 - システム設計の変更 501
 - システムの文書化 97
 - データ・セット
 - バックアップ 133
- システム (続き)
 - テスト
 - テスト・システムの確立 456
 - テスト・システムの設定 456
 - パフォーマンス
 - 概要 516
 - 報告書 482
 - モニター 465
- システム間拡張サービス (XES)
 - IMSplex 内での使用 23
- システム間キューイング、判別 900
- システム管理
 - 考慮事項 147
 - 作業 147
- システム管理作業 44
- システム管理の再作成 222
- システム間連絡 84
- システム構成報告書
 - IMS モニター
 - DBCTL 906
 - DB/DC 870
 - DCCTL 927
- システム始動
 - セキュリティー 430
 - JCL セキュリティー・オプション 430, 443
- システム障害
 - サービスの再開
 - IMSplex 743
 - XRF 743
 - IMS 再始動の同期化 353
- システム設計の変更、効果 502
 - オンライン変更の場合 502
 - スケジューリング変更 502
 - セキュリティー保守 502
 - 端末接続機構 502
 - チューニングの変更 502
 - 出口ルーチン 502
 - ネットワーク制御 502
 - 変更された出力 502
 - 変更されたデータベース 502
 - 変更されたトランザクション 502
 - 変更されたプログラム 502
 - 変更されたメッセージ形式 502
- システム定義 (system definition)
 - オンラインの変更 574, 579
 - オンライン変更のための変更 509
 - 簡易 IMSplex 28
 - サービス
 - SYSMOD 回帰、回避 253
 - タイプ ALL
 - 実行する時期 247
 - データ共用の要件 306
 - 文書 97
 - 保守上の変更点
 - SYSMOD 回帰、回避 253
- システム定義 (system definition) (続き)
 - SYSMOD 回帰、回避 253
 - XRF 複合システム (XRF complex) 814
- システム定義の管理
 - オンライン変更 (online change) 509
- システム定義の更新
 - オンライン変更の場合 509
 - 複数のマクロ変更 506
- システム定義の変更
 - オンライン変更の場合 509
 - 制御 506
 - 定義タイプの選択 507
- システム定義マクロ・ステートメント
 - 処理の要件 507
 - XRF に対するコーディング 815
- システム・コンソールの編成
 - XRF 756
- システム・サポート・プログラム (SSP)
 - バックアップ・セッション用の制御ブロックの生成 811
 - XRF に対する要件 756
 - XRF プロセスに与えられる利点 768
- システム・チェックポイント (system checkpoint) 106
 - 定義 87
 - システム・チェックポイントの開始 215
 - システム・チェックポイント・データ・セット、CQS 115
 - システム・データ・セットの保護 435
 - システム・プログラマー
 - 入出力防止について 762
 - システム・メッセージ
 - 初期設定時 773
 - 入出力防止中の 762
 - によって起動されたプロセス 780
 - ユーザーの代替 IMS へのログオン時 763
 - IRLM 操作の再開時 784
 - XRF テークオーバー時 750, 805
- システム・ライブラリーの保護 435
- システム・リカバリー
 - 障害後 353
- システム・リソース・マネージャー (SRM) 798
 - XRF に与える利点 762
- システム・ログ
 - 概要 86
 - セキュリティー違反レコード 433
- システム・ログ・データ・セット (SLDS) (system log data set (SLDS))
 - 圧縮 120
 - 環境 113
 - コピー 118
 - 作成 113
 - 単一または重複 129

システム・ログ・データ・セット (SLDS) (system log data set (SLDS)) (続き)
 定義 108, 129
 ブロック・サイズ 129
 レコード 705
 ログ 113
 割り振り 113
 RECON データ・セット 113

シスプレックス (sysplex)
 オンライン変更 (online change) 583
 直列化プログラム管理 206
 データ共用 (data sharing) 349
 データ共用環境 335
 プログラム、直列化 206
 ローカル・オンライン変更 (local online change) 585

シスプレックス環境
 処理に適している 283
 メッセージ 283
 APPC メッセージ 284
 同期 284
 非同期 284
 OTMA メッセージ 284
 同期 284
 非同期 284

シスプレックス・データ共用
 カップリング・ファシリティ (coupling facility) 338
 バッファ無効化 (buffer invalidation) 336

シスプレックス・データ共用 (sysplex data sharing) 335
 概念 335
 構成 339
 構造
 サイズの変更 348
 構造のサイズの計算 345
 使用する時期 344
 バッチ・ジョブの BMP ジョブへの交換 344
 用語 335
 XRF 339

実記憶域
 テークオーバー中に必要な 762
 ページ固定 774
 DFSFIXxx の定義 774
 XRF トラッキング中に使用される 774

実行プロシージャの調整
 データ共用用 308

実行プロファイル報告書
 概要 870
 汎用処理率の追加
 DBCTL 906
 DB/DC 870
 DCCTL 927

実行プロファイル報告書 (続き)
 IMS モニター
 DBCTL 906
 DB/DC 870
 DCCTL 927

実動
 サービス・レベル
 推奨 242, 244
 保守 243
 保守上の変更点 243
 推奨 242, 244
 APAR 242, 244
 PTF 242, 243, 244
 SYSMOD 242, 243, 244

実動構成の文書化 97

実動システム
 サービス・レベル
 推奨 242
 プロセスの例 244
 保守 243
 保守上の変更点 243
 推奨 242
 プロセスの例 244
 APAR 242
 保守プロセスの例 244
 PTF 242, 243
 保守プロセスの例 244
 SYSMOD 242, 243
 保守プロセスの例 244

自動 RECON 損失通知
 構成 58
 シャットダウン 168
 初期設定 28
 並列アクセス 168
 保守 168

自動アーカイブ 116

自動化オペレーション
 概要 102, 134
 自動化するもの 102
 利点 102

自動化操作プログラムのアプリケーション
 ユーザー ID の代替 405

自動化操作プログラム・インターフェース (AOI) (Automated Operator Interface (AOI))
 説明 134

自動再始動管理 (ARM)
 IMSplex 内での使用 23

自動手順
 サポートされるコンソール 52
 単一制御点 52

始動パラメーター 273

自動リスタート・マネージャー
 エレメント名 165
 使用 165
 使用可能化 165

自動リスタート・マネージャー (ARM) 91

自動ログオン端末
 共用キュー 263

シナリオ
 オンライン変更 (online change) 595

シミュレーション、オンライン実行の
 シャットダウン
 概要 137
 コマンド 137
 IMS 453
 IMSplex、内 373

修正サービス
 インストール 245
 IMSplex 252

従属サービス・エレメント (DSE) (dependent service element (DSE))
 定義 745

従属領域
 開始 8
 セキュリティー 416, 438
 リソース・アクセス・セキュリティー (RAS) 417
 セキュリティー・オプション 80
 代替 IMS システムでの始動 773
 タイプ 8
 定義 8
 ログに記録されている状況 774
 BLDL リスト 520

従属領域 (dependent region)
 アドレス・スペース
 実ストレージの割り振り 540

従属領域の BLDL リスト 520

修復 723

重要度
 アドレス・スペースへの割り当て 469
 業務
 ワークロード・マネージャー 468
 相対的 468, 469, 472
 MWP
 ワークロード・マネージャー 472

重要な注意
 インターフェースの考慮事項 147
 予防保守のインストール
 APPLY なしの ACCEPT 246
 APPLY の前の ACCEPT 251
 SYS1.NUCLEUS 155
 VTAM インターフェースの考慮事項 163

重要なトランザクションのスケジューリング
 検査 971

終了フェーズ (termination phase)
 説明 788

主記憶データベース 83

- 主記憶データベース (MSDB)
 - グローバル・リソースとしての状況 202
- 出力、IMS スプール API による印刷 605
- 出力、IMS モニター報告書からの DB/DC 869
- 出力順序と内容、IMS モニター報告書の DCCTL 926
- 種別の規則 470, 472
- 順次バッファリング、OSAM 530
- 順次バッファリング明細報告書 957
- 順方向データベース・リカバリー
 - データ共用 (data sharing) 649
 - 非データ共用 649
- 順方向リカバリー
 - データ共用 (data sharing) 332, 649
 - 定義 137
 - 非データ共用 649
- 障害
 - 後のログオン
 - 類似性 362
 - システム・リカバリー 353
- IRLM
 - 再始動 334
- 障害が発生したアクティブ IMS を新しいアクティブ IMS として復帰 786
- 使用可能化
 - 共用キュー 270
- 使用のシナリオ
 - オンライン変更 (online change) 595
- 商標 979, 981
- 使用不可化、制約の
 - リソース名の一意性 367
- 情報削除
 - RECON からのログ・レコードの 718
- 初期設定
 - DBRC
 - IMSCTRL マクロ 645
 - IMSplex、内 373
 - RECON データ・セット 645
- 初期設定出口ルーチン (DFSINTX0)
 - ETO を使用して 415
- 初期設定フェーズ
 - オペレーターのプロシージャ 771
 - 図 771
 - 説明 771
- 除去、SSYS の DB 許可からの 734
- 除去時刻
 - データベース変更累積 122
 - 定義 673
- ジョブ制御言語 (JCL)
 - 骨組み実行メンバー
 - 定義 620
 - DBRC による生成 620
- ジョブ入力サブシステム
 - 障害、XRF の制限 750
- ジョブ要約報告書 482
- 処理、高速機能の 84
- 処理インテント (processing intent) 295
- 処理対象がない旨のメッセージ
 - 変更累積 735
- シリアライゼーション
 - RECON データ・セットの 695
- 推奨
 - 共用キュー
 - リソース構造の使用 50
 - スケジューリング
 - アプリケーション・プログラム
 - 使用不能なデータに対する 82
 - AOI トランザクション 287
 - スケジューリング・アルゴリズム
 - 領域占有の要因 523
 - ステージング・ライブラリー 570
 - スプール API 605
 - スペース所要量、RECON データ・セット 691
- 制御プログラム 8
- 制御領域 (control region)
 - アドレス・スペース
 - 実ストレージの割り振り 540
 - 開始 451
 - 再始動 451
 - 実行パラメーター
 - 共用キュー環境での定義 273
- 制御プログラム 8
 - 定義 8
 - DCCTL 環境における 14
 - DL/I 呼び出し 8
 - IMSplex における 26
- 生産性向上エイド
 - データ・ディクショナリー 97
 - テストするためのバッチ端末シミュレーター 461
 - モニター用 476, 480
- 生産性向上ツール
 - データベース 446
- 正常再始動 (normal restart)
 - データ共用による 653
- 生成、(モバイル・ワークロード) レポートの 474
- 制約事項
 - オンライン再編成 (online reorganization) 681
 - 並行イメージ・コピー (CIC) 663
- 制約の使用不可化
 - リソース・タイプ整合性 368
- セキュリティ 375, 408, 684
 - 暗号化
 - 暗号サポートの使用 437
- セキュリティ (続き)
 - 暗号化 (続き)
 - セグメント編集/圧縮ルーチンの使用 437
 - ICSF/CCA インターフェースの使用 437
 - VTAM 端末 437
 - オンラインでのインプリメント 575, 580
 - オンラインの変更 575, 580
 - 高速機能に関する考慮事項
 - DBCTL 441
 - DB/DC 408
 - サインオン検査 390
 - システム始動オプション 443
 - システム定義時に行われる選択 387
 - システム・ライブラリーとデータ・セット
 - 概要 435
 - DLISAS プロシージャ 435
 - IMS プロシージャ 435
 - 始動オプション 430
 - 従属領域
 - サインオンなし 418
 - リソース・アクセス・セキュリティ (RAS) 417
 - 設計の考慮事項
 - コマンドの許可 392
 - 従属領域からのアクセスの制限 416
 - セキュリティ・タイプの選択 389
 - 端末からのアクセスの制限 390
 - トランザクションの許可 392
 - マスター端末 401
 - DBCTL 環境 440
 - RACF の使用 394
 - 端末、デフォルト 387
 - データベース
 - セグメント・レベルおよびフィールド・レベルのセンシティブティ 436
 - RACF セキュリティ 436
 - 定義
 - EXEC ステートメントのパラメーター 430
 - 出口ルーチンの準備 421
 - ネットワーク・セキュリティ資格情報の伝搬 395, 397, 399
 - パスワード検査 390
 - 表示バイパスとパスワード・マスキング 434
 - ファシリティ・クラス 209
 - 物理 434
 - 分散セキュリティ資格情報の伝搬 395, 397, 399
 - 保護できるリソース 386, 438

セキュリティ (続き)

リソース・アクセス管理機能 (RACF)
(Resource Access Control Facility
(RACF)) 575, 580
AO (自動化操作プログラム) のアプ
リケーション・プログラム 85, 401
CMD 呼び出し 401
CPI-C ドリブン・アプリケーション・
プログラムの考慮事項 409
DBCTL 439
DBCTL でのオンライン変更 443
DBCTL に関する考慮事項 438
DB/DC
概要 386
DCCTL
概要 386
ETO 415
ICMD 呼び出し 401
IMS Connect 420
IMS セキュリティの活動化 421
RACF の初期設定 422, 442
SECURITY マクロの定義 442
IMSplex 33
RACF OPERCMDS クラス 189
JMP アプリケーション 409
MSC
トランザクション 412
IMS が DFSCTRNO を呼び出す場
合 413
MSCSEC= 412
RACF および出口ルーチンが呼び
出された場合 413
MSC および共用キューでの有効範囲
410
ODBA アプリケーション・プログラム
409
ODBA アプリケーション・プログラ
ムの考慮事項 442
ODBM (Open Database
Manager) 175
Open Database Manager
(ODBM) 175
Open Transaction Manager Access
(OTMA) 419
OTMA (Open Transaction Manager
Access) 419
RACF 209
RACF (リソース・アクセス管理機能)
575, 580
RECON 用に許可されるコマンド 683
Repository Server 426
例 430
Resource Manager (RM) 428
Structured Call Interface 209
user
サインオンなし 418

セキュリティ違反 433
しきい値の設定 433
システム・ログに記録された 433
マスター端末への通知 433
セキュリティ機能
DB/DC リソース 388
DCCTL リソース 388
セキュリティの考慮事項
RECON データ・セット 702
セキュリティ・オプション
従属領域の 80
セグメント編集/圧縮出口ルーチン 437
セグメント・レベル・センシティブ
ィティ 436
世代別データ・グループ (GDG)
定義 41
モデル・データ・セット 42
設計の検討 94
参加 94
セッションの切り替え優先順位
クラス 1 端末の定義 817
デフォルト値 815
セッション・リカバリー
XRF テークオーバー後の速度 781
セッション・リカバリー (session
recovery)
定義、XRF 758
セッション・リカバリーの優先順位
クラス 1 端末の定義 817
クラス 2 端末の定義 817
IMS サポートに対するデフォルト
815, 817
接続の障害
カップリング・ファシリティ
(coupling facility) 349
全機能 VSAM データベース・デー
タ・セ
ット (非 HALDB または非 OLR 対応)
の暗号化 384
全機能 VSAM データベース・デー
タ・セ
ット (HALDB、OLR 対応) の暗号化
380, 384
先行書き込みデータ・セット (WADS) の
暗号化 385
占有、領域
測定 877
専用キュー・タイプ、CQS によって管理
される 74
専用トランザクション、高速機能 84
操作タスク
概要 101
操作手順のテスト 457
操作の監査 433
相対的重要度
アドレス・スペースへの割り当て 469
ワークロード・マネージャー 468

ソフトウェア要件

CQS 71
損傷した RECON データ・セット 721

[夕行]

代替 IMS システム

アクティブ IMS との同期化 773
アクティブ IMS になる 776
アクティブ IMS のトラッキング 758,
774
アクティブ IMS の問題の検出 758,
774
クラス 1 端末に対するバックアップ・
セッションのオープン 758
クラス 1 端末に対するバックアップ・
セッションのクローズ 758
計画作業負荷 747
システム・ログ、割り振り 773
始動 771
終了時 788
制御ブロックの更新 758, 774
テークオーバー時の新しいトランザク
ションの実行 758, 782
テークオーバー中の処理 776
テークオーバーと見なす 776
テークオーバーの後処理フェーズ中の
プロセス 786
データベース、割り振り 758
データベースのオープン 758
データベースの割り振り 758
定義 745
トラッキング・フェーズ中のプロセス
747, 774
入出力許容の実行 780
バックアップ・セッション、クラス 1
端末に対する 750
非 XRF 作業に対する可用性 747
非 XRF 作業負荷
使用可能な 747
テークオーバー時 798
プロセスのサイクル 789
別のシステム 786
ログオン 758, 763
IMS システム・ログの割り振り 773
MSDB のロード 773
代替 IMS システムのスワップ不能作業
798
代替 IMS システムの制御ブロックの更新
774
代替 IMS システムへのログオン 763
代替サブシステムへのログオン 758
タイプ 1 AO アプリケーション・プロ
グラム
セキュリティ 401

- タイプ 2 AO アプリケーション・プログラム
 - セキュリティ 401
- タイプ 2 SVC
 - バインディング 155
 - ロード 155
- タイプ 2 コマンド 43
 - 環境 78
 - 構成要件 78
 - 使用、Resource Manager なしでのタイプ 2 コマンド環境
 - RM のないオンライン変更の制約事項 55
 - 要件 78
- タイプ 2 コマンド (type-2 command)
 - RACF セキュリティ
 - リソース名と許可 190
- タイプ 2 コマンド環境
 - グローバル・オンライン変更の制約事項 55
 - 定義 28
- タイムアウト値
 - 設定 768
- タイムアウト値、定義 770
- タイム・シェアリング・オプション (TSO)
 - 代替 IMS システム上で動作する 798
- タイム・スタンプ
 - リカバリー 632
- ダイヤル呼び出し回線 810
- 単一制御点 (SPOC) (single point of control (SPOC))
 - 機能 52
 - ユーザー作成 52
 - REXX 54
 - TSO SPOC 52
- 単一ログイン 125
- ダンプ・フォーマット 662
- 端末
 - グローバル管理 58
 - システム定義マクロのキーワード 817
 - シスプレックス内の TM リソースとして 370
 - 自動ログオン (autologon)
 - 共用キュー 263
 - セッション平衡化
 - 持続している類似性 360
 - セッション・リカバリーの優先順位の定義 817
 - 名前の一意性 367
 - IMSplex、内 370
 - 管理 367
 - RM 定義 370
 - VTAM 端末ノード 370
 - XRF に対する要件 756
 - XRF 複合システムでのログオン 763
- 端末障害
 - XRF 制限 750
- 端末状況
 - リセット 361
- 端末プロファイルの文書化 97
- チェックポイント
 - オーバーヘッド 107
 - システム 107
- チェックポイント (checkpoint) 106
 - 概念 87
 - 構造、開始 216
 - システム、開始 215
 - 処理の影響のモニター
 - DBCTL 909
 - DB/DC 877
 - DCCTL 933
 - データ・セット 215
 - 頻度 520
 - 頻度の設定 520
 - プログラムの同期のための 88
 - IMSplex、内 373
- 逐次アプリケーション
 - 処理
 - 共用キュー環境での 265
- 逐次トランザクション
 - 処理
 - 共用キュー環境での 265
- 逐次プログラム
 - シスプレックス環境での直列化の管理 206
 - シスプレックスでの管理 206
 - GBL_SERIAL_PGM 206
- チューニング
 - アプリケーション 537
 - アプリケーション制御ブロックの配置 520
 - 概要 515
 - 実施計画 507
 - 入出力の競合の検討 548
 - 反復プロセスとしての 515
 - プロセッサ・リソースに関する問題の検出 541
 - ページング率の検討 520
 - 方法の定義 539
 - ユーティリティ 515
 - 64 ビット・ストレージ内の ACB 520
 - IMS システム 548
- 中央演算処理装置複合システム (CPC)
 - 定義 745
- 調整
 - データ共用のための実行 JCL の 308
 - XRF に対する IMS 819
 - z/OS 309
- 重複ログイン (dual logging) 125
 - 定義 110
- ツール
 - 生産性 133
- 通常操作
 - 操作中の代替トラッキング 758
 - 代替 IMS システム、トラッキング 774
- 通信 IWAIT 報告書 541, 543
- 通信コントローラー
 - 37x5 756
- 通信待機報告書
 - IMS モニター (DB/DC) 890
 - IMS モニター (DCCTL) 943
- 通信網シミュレーター (TPNS) 464
- 通信要約報告書 541
 - IMS モニター (DB/DC) 889
 - IMS モニター (DCCTL) 942
- 通知、DBRC にログ・データ・セットの移動を 736
- テークオーバー (takeover)
 - 後の問題判別 750
 - 基準の設定 768
 - さまざまな処理フェーズ 783
 - システム・メッセージ、XRF 758
 - 代替 IMS システムの非 XRF ジョブに与える影響 798
 - 端末に与える影響 781
 - テークオーバー処理の相違点 783
 - メッセージ 805
 - ユーザーにわかる場合 805
 - 予定
 - 使用 750
 - 処理 778, 785
 - XRF に対する IMS 処理 758
 - XRF に対する条件 750
 - XRF に対する条件の設定 750
 - XRF に対する定義 745
 - XRF についての VTAM 強制テークオーバー 763
 - XRF、使用 785
 - XRF、ユーザーにわかる場合 747
 - XRF、例 750
- テークオーバー時のセッション切り替え
 - バックアップ・セッション 781
- テークオーバー時のネットワーク変更の開始 781
- テークオーバー条件
 - 予定 776
 - ログ・レコードの送信失敗 768
 - ISC リンクの信号送信の失敗 768
 - RDS 信号の送信失敗 768
 - XRF 776
- テークオーバー処理の相違点 783
- テークオーバーの後処理フェーズ
 - 図 786
 - 説明 786

- テークオーバーの開始
 - テークオーバー中の処理 776
 - 手順 776
 - 目的 785
- テークオーバー・フェーズ
 - XRF、説明 778
- データ
 - アプリケーションのアクセス方法 295
- データ共用
 - ブロック・レベル 298
 - マージ、ログの 673
 - DBRC、役割 293
- データ共用 (data sharing)
 - イメージ・コピー 296, 322
 - イメージ・コピー (image copy) 325
 - 作成 326
 - インストール・タスク 306
 - エリア・レベル 291, 293
 - オンライン変更 (online change) 321
 - 開始 310
 - 構成の相違点 291
 - 構成例 305
 - コマンド 311
 - シズプレックス (sysplex) 335, 349
 - 概念 335
 - グループの定義 302
 - ユーザーの定義 302
 - 用語 335
 - ロック構造の定義 302
 - IRLM 302
 - 順方向リカバリー 332
 - 除外 308
 - 制御 448
 - 説明 291, 298
 - 通常操作 322
 - データベース
 - 再編成 321
 - データベースの可用性 291
 - データベースの共用レベル
 - 変更 327
 - データベースの処理
 - 開始 311
 - 停止 311
 - データベース健全性 (database integrity) 295, 642
 - データベース・レベル 291, 293, 641
 - 更新アクティビティ 303
 - 複数の読み取り 304
 - データ・リソースとしてのデータベース 291
 - 停止 310
 - 動的バックアウト (dynamic backout) 649
 - バックアウト (backout) 330, 331
 - プロシージャ 329
 - ブロック・レベル 641
- データ共用 (data sharing) (続き)
 - 命名規則の設定 295
 - リカバリー 328
 - リカバリーの計画 649
 - リソース、モニター 313, 448
 - レコード 714
 - レベル 641
 - ログ 328
 - DBCTL 環境における 13
 - DBRC 319, 332, 448
 - コマンド 320
 - DBRC 制御対象 641
 - DBRC による共用レベルの割り当て 642
 - DBRC の障害
 - 再始動 333
 - DB/DC 環境 11
 - DEDB の可用性 291
 - IMS の障害 354
 - 再始動 332
 - IRLM
 - 開始 310
 - 再始動 333
 - 停止 311
 - IRLM、役割 298
 - RECON データ・セット内の情報 641
 - RECON、モニター 315
- データの伝搬 90
- データベース
 - アクセス 449
 - アクセス意図の変更 319
 - アプリケーション設計の検討 94
 - イメージ・コピー、作成 326
 - イメージ・コピーの作成 659
 - オンライン変更
 - IMS 管理の ACB 環境 555, 557, 562, 563
 - オンライン変更 (online change)
 - 管理対象 ACB 環境 553
 - 活動化
 - データ共用および IMS 管理の ACB 環境 559
 - ACB (アプリケーション制御ブロック)、IMS 管理対象 554
 - IMS 管理対象 ACB 環境での 554
 - IMS 管理の ACB 環境、データ共用による 559
 - 可用性 321
 - 共用レベルの変更 327
 - グループ 707
 - 高速機能タイプ 83
 - 再編成 321, 325
 - 作成
 - アプリケーション制御ブロック (ACB)、IMS 管理対象、活動化 554
- データベース (続き)
 - 作成 (続き)
 - ACB (アプリケーション制御ブロック)、IMS 管理対象、活動化 554
 - 作成、HISAM コピーの 669
 - 順方向リカバリー
 - データ共用による 649
 - データ共用を使用しない 649
 - 動的、データ共用による 649
 - 動的、非データ共用 649
 - 非データ共用 649
 - 障害、XRF の制限 750
 - 生産性向上ツール 446
 - テークオーバー中の処理 778
 - データ共用における健全性 642
 - 動的反リカバリー
 - データ共用による 649
 - データ共用を使用しない 649
 - 非データ共用 649
 - 入出力の分析 548
 - バックアウト・プロセスによる損傷 649
 - バックアップ 131, 658
 - コマンド 659
 - バックアップのガイドライン 671
 - バッファのモニター 887, 917
 - ページ固定バッファ 526
 - 変更の許可 734
 - 保管されたリソース情報 204
 - 保護 436
 - 健全性の保証 762
 - リカバリー
 - 動的、データ共用による 649
 - 動的、非データ共用 649
 - バッチ 649
 - リカバリー・レコード 706
 - レコード 708
 - 割り振り解除
 - ロギング、RECON データ・セット内の 639
 - 割り振りレコード 713
 - DEDB 83
 - DL/I 呼び出しのモニター 874, 930
 - MSDB 83
 - XRF での動的割り振り 828
 - XRF に対する要件 747
 - XRF を備えた健全性の保証 750
 - データベース管理ブロック (DMB) テーブル・レコード 723
 - データベース管理ブロック (DMB) 番号 723
 - データベース記述子 (DBD)
 - アプリケーション制御ブロック (ACB)、IMS 管理対象、活動化 554

- データベース記述子 (DBD) (続き)
 - 活動化
 - データ共用および IMS 管理の ACB 環境 559
 - IMS 管理の ACB 環境、データ共用による 559
 - 活動化、IMS 管理の ACB 環境での 554
 - ACB (アプリケーション制御ブロック)、IMS 管理対象、活動化 554
 - データベースの再編成 321
 - データベースの登録
 - RECON データ・セットへの 647
 - データベース変更累積ユーティリティ (DFSUCUM0) 120
 - ガイダンスの使用法 673
 - 概要 672
 - 最大数、世代の 666
 - サブセット、ログ・ボリュームの 673
 - 実行、DBRC により記録される 674
 - 除去時刻 122
 - 説明 671
 - 入力 121
 - 有効なログ・サブセット、DBRC での 673
 - リカバリー期間 666
 - CA グループ
 - 再使用 675
 - 再使用しない 675
 - 定義 674
 - 定義、将来の利用のための 675
 - SLDS または RLDS の圧縮 120
 - データベース健全性 (database integrity)
 - データ共用サポート 295
 - データベース割り振りレコード 713
 - データベース・イメージ・コピー 2 ユーティリティ (DFSUDMT0)
 - 概要 661
 - データベース・イメージ・コピー・ユーティリティ (DFSUDMP0)
 - イメージ・コピー、作成 660
 - 最大数、世代の 665
 - 再利用、イメージ・コピー・データ・セットの 668
 - 作成、将来の利用のためのデータ・セットの 664
 - 実行、DBRC により記録される 658
 - 説明 659
 - 標準外イメージ・コピー・データ・セット 670
 - リカバリー期間 665
 - データベース・データ・セット
 - 開始、RECON データ・セットの 647
 - データベース・データ・セット (DBDS) (database data set (DBDS))
 - A から J、L および X 678
 - データベース・データ・セット (DBDS) (database data set (DBDS)) (続き)
 - M から V および Y 678
 - データベース・データ・セット (DBDS) グループ
 - コマンド、指定する
 - GENJCL.IC 676
 - GENJCL.OIC 676
 - GENJCL.RECEIVE 676
 - GENJCL.RECOV 676
 - GENJCL.USER 676
 - LIST.DBDS 676
 - LIST.HISTORY 676
 - コマンド、定義に影響を及ぼす
 - CHANGE.DBDSGRP 676
 - DELETE.DBDS 676
 - DELETE.DBDSGRP 676
 - INIT.DBDSGRP 676
 - 使用 676
 - ILDS (間接リスト・データ・セット) 676
- データベース・データ・セット・レコード 708
- データベース・バッファ・プール
 - 最適化 530
- データベース・バッファ・プール報告書
 - 説明 852
 - 報告書の使い方 852
 - 報告書のフィールド 852
 - IMS モニター (DBCTL) 917
 - IMS モニター (DB/DC) 887
- データベース・リカバリー (database recovery)
 - グループ 707
 - 処理の概要 631
 - データベース・イメージ・コピー 2 出力フォーマット 662
 - データベース・リカバリー
 - 概念 631
 - DBRC の役割 635
- データベース・リカバリー管理 457
- データベース・リカバリー管理 (DBRC)
 - アクティブ RECON データ・セット 702
 - いつ使用するかの指定 645
 - オープン・オンライン PRILOG レコードのクローズ 732
 - オンライン・コマンド
 - /RMCHANGE コマンド 622
 - /RMDELETE コマンド 622
 - /RMGENJCL コマンド 622
 - /RMINIT コマンド 622
 - /RMLIST コマンド 622
 - /RMNOTIFY 622
 - 概要 8, 619
 - 共用レベル、割り当て 642
- データベース・リカバリー管理 (DBRC) (続き)
 - 共用レベルの割り当て 642
 - 許可
 - 変更、データベースの 734
 - 許可のためのリソース名 686
 - 記録、イメージ・コピーの 658
 - 区分データ・セット・メンバー 620
 - グループ
 - 定義 677
 - 考慮事項、使用上の 655
 - コマンド
 - 発行 620
 - コマンド許可 686
 - セキュリティ・プロダクトの使用 684
 - DSPDCAX0 および RACF の両方の使用 685
 - DSPDCAX0 の使用 685
 - サブシステム (SSYS) レコード 733
 - システム考慮事項 655
 - 実行されるタスク 621
 - 使用、ヒントの 727
 - 使用する時期 619
 - 初期 RECON アクセス 702
 - 初期設定 645
 - スペアの RECON データ・セット 702
 - セキュリティ・リソース・プロファイル 684
 - 妥当性検査、ユーティリティ JCL の 636
 - データ共用管理 641
 - データ共用レコード 641
 - データベースの登録 647
 - データベースのバックアップ
 - イメージ・コピー・データ・セット および GENMAX のリカバリー期間 665
 - オンライン・データベース・イメージ・コピー (DFSUICP0) 663
 - 指針 671
 - 制御、管理されるイメージ・コピー数の 665
 - データベース・イメージ・コピー 2 (DFSUDMT0) 661
 - データベース・イメージ・コピー (DFSUDMP0) 660
 - 並行イメージ・コピー 663
 - 変更累積データ・セットのリカバリー期間、および GRPMAX 666
 - 方式 658
 - リカバリー、データベースの 635
 - HISAM コピー 669
 - データベース・リカバリー管理ユーティリティ (DSPURX00) 620

データベース・リカバリー管理 (DBRC)
(続き)
データ・セット
区分 620
RECON 620
RECON データ・セット 691
データ・セットのカタログ管理 736
データ・セット命名規則 655
との通信 622
バッチ・コマンド
BACKUP.RECON コマンド 622
CHANGE コマンド 622
CLEANUP.RECON コマンド 622
DELETE コマンド 622
GENJCL コマンド 622
INIT コマンド 622
LIST コマンド 622
NOTIFY コマンド 622
RESET.GSG コマンド 622
パラメーター
IMS.PROCLIB 実行パラメーター
645
変更累積の処理対象がない旨のメッセ
ージ 735
骨組み JCL 620
リカバリー・ユーティリティ 636
リカバリー・レコード・タイプ 706
ログ関連のコマンド
CHANGE.PRILOG 628
CHANGE.RECON 628
CHANGE.SECLOG 628
DELETE.LOG 628
GENJCL.ARCHIVE 628
GENJCL.CLOSE 628
LIST.LOG 628
NOTIFY.PRILOG 628
NOTIFY.SECLOG 628
ログ制御 625
ログ・データ・セット
移動の通知 736
API 要求
概要 622
API 要求許可
セキュリティ・プロダクトの使用
684
DBDS の登録 647
DBRC パラメーター
IMSCTRL マクロ 645
DBRC プロシージャ 645
GENMAX パラメーター
使用 665
GENMAX、リセット 729
GRPMAX、リセット 730
IMS からの呼び出し 621
IMS プロシージャと 645
IMSplex サポート 643

データベース・リカバリー管理 (DBRC)
(続き)
IRLM 状況 653
IRLM を伴うリカバリー 653
JCL の生成 620
OLR、考慮事項 678
PRILOG レコード
クローズ 732
RECON 初期設定トークン (RIT) 639
RECON データ・セット
置き換え、損傷した 721
置き換え、廃棄された
RECON 723
オンラインでの再編成 720
拡張 699
競合、回避 692
競合の回避 692
許可要求サポート 683
計画に関する考慮事項 692
コマンド許可サポート 683
再編成手順 720
再編成の概要 719
再編成前のバックアップ 720
作成 700
作成時の推奨 700
修復 723
初期設定 692
セキュリティの考慮事項 702
損失通知 725
データベースの登録 647
デッドロック状態の回避 695
入出力エラー処理 721
バックアップ 716
並列アクセス 693
並列アクセス、計画 693
ヘッダー・レコード 704
保守 716
リカバリー 721
リカバリー・レコード・タイプ 706
レコード・タイプ 704
レコード・マッピング 715
ログ・データ・セット・レコード
705
ALLOC レコード 713
BACKOUT レコード 706
CA レコード 707
CAGRP レコード 707
DBDS の登録 647
DBDSGRP レコード 707
DEFINE CLUSTER キーワード
700
IMAGE レコード 712
LOGALL レコード 713
LSR オプションを使用した処理
700
RECOV レコード 714

データベース・リカバリー管理 (DBRC)
(続き)
RECON データ・セット (続き)
REORG レコード 712
SSYS レコード 714
RECON データ・セットの初期設定
645
RECOVPD パラメーター
使用 665
RIT (RECON 初期設定トークン) 639
SLDS 停止時刻の検出、RECON デー
タ・セット内の最後の 727
データベース・レベルのデータ共用 291,
293
データ・キャプチャー出口ルーチン 130
説明 90
データ・スペース
RACF 410
データ・セット
A から J 678
ISPTABL 142
M から V 678
データ・セット、RECON 691
データ・セットの配置
XRF 828
データ・セットの別個のコピーの保持
XRF 828
データ・セットの割り振り
ACBLIB 505
RECON データ・セット
作成 700
複数プロセッサ間での共用 695
データ・ディクショナリー
IBM DB/DC データ・ディクショナリー
ー 97
低下モード・ロギング 125
ディクショナリー
データ・ディクショナリーの使用 97
IBM DB/DC データ・ディクショナリー
ー 97
停止
IRLM サブシステム 311
OLDS 110
ディスパッチング優先順位 541
出口
データベースのオープン 639
出口点
定義 89
出口ルーチン
の紹介 89
RECON 入出力
RECON に対する変更のトラッキ
ング 726
テスト
エイド 460
オンライン 460

- テスト (続き)
 - オンライン実行のシミュレーション 461
 - オンラインの MFS 形式 461
 - オンライン変更を伴う 462
 - オンライン・システム 455
 - システム
 - 設定 456
 - ストレス 464
 - 操作プロシージャー 457
 - 段階 455
 - データベース 457
 - テスト環境でのモニター 458
 - テスト・システム 456
 - テスト・システムの確立 456
 - テスト・システムの設定 456
 - テスト・データベース 457
 - ネットワークの作動可能性の確認 459
 - バッチ端末シミュレーターの使用 461
 - パフォーマンス 464
 - プログラム 464
 - プロシージャー、操作 457
 - MFSTEST モード 461
 - QCF、による 464
 - SYSIN/SYSOUT 464
- テスト環境
 - モニター 458
- デッドロック
 - 異常終了 88
 - 考えられる原因 88
 - CCTL スレッド 88
 - DBCTL 88
- デッドロック、RECON データ・セットの回避 695
- デッドロック・イベント要約報告書
 - IMS モニター (DBCTL) 918
 - IMS モニター (DB/DC) 890
- デフォルト・セキュリティ 375
 - セキュリティ・オプション 387
 - 定義 387
- 同期点 106
 - アプリケーション・プログラム 107
 - 概要 106
 - 定義 88, 106
 - トランザクション・フロー内の 487
- 同期フェーズ (synchronization phase)
 - 図 773
 - 説明 773
 - SNAPQ チェックポイントの使用 773
- 統計
 - トランザクション・レベル
 - 解釈 492
 - 分析 492
 - モニター 492
- 統計分析ユーティリティー 445
- 統計分析ユーティリティー (DFSISTS0)
 - アプリケーション会計報告書 974
 - 会計の使用率 974
 - 回線および端末報告書 968
 - 作成される報告書の説明と例 967
 - 実行の短縮 967
 - チューニングでの使用 515
 - トランザクション応答報告書 968
 - トランザクション負荷の計算 968
 - トランザクション・プロファイル 974
 - 入力 967
 - 複数システムで使用 465
 - メッセージの報告書 970
 - メッセージ・トラフィックの 526
 - IMS システム・ログの使用 476
- 動的データベース・バックアウト
 - データ共有 (data sharing) 649
 - 非データ共有 649
- 動的バックアウト (dynamic backout)
 - データ共有 (data sharing) 649
 - 非データ共有 649
- 動的バックアウト、データ共有による
 - 動的 649
- 動的リソース定義
 - IMS リソース
 - 更新 581
 - 削除 581
 - 作成 581
- 動的リソース定義 (DRD) (dynamic resource definition (DRD))
 - IMSplex のシナリオ 32
- 動的割り振り (dynamic allocation)
 - データベース・データ・セット 36
 - ACB ライブラリー 36
 - HALDB 36
 - OLDS 36
 - RECON データ・セット 699
 - RECON データ・セット (RECON data set) 36
 - SLDS 36
 - WADS 36
 - XRF での IMS データベース 828
 - XRF でのエリア・データ・セット 828
 - z/OS での可用性 36
- 登録解除、インタレストの
 - 共用キュー
 - LTERM に関する 262
 - MSC (複数システム結合機能) リソースに関する 262
 - トランザクションに関する 261
- 登録の許可 212
- 特記事項
 - 商標 979, 981
 - 特記事項 979
- トラッキング中に使用される仮想記憶域 774
- トラッキング・フェーズ (tracking phase)
 - 図 776
 - 説明 774
- トランザクション
 - 延期状態の再キューイング 287
 - オンラインの変更 575, 580
 - 会話型
 - 共用キュー環境での 266
 - 監査 970
 - シスプレックス内の TM リソースとして 371
 - 重要なスケジューリング 971
 - セキュリティ
 - MSC 412
 - 逐次
 - 共用キュー環境での 265
 - 定義
 - タイプ 1 AO コマンド 402
 - 入力 IMS に未定義
 - 宛先作成出口ルーチン (DFSINSX0) 264
 - 保管されたリソース情報 205
 - ユーザー IDの定義
 - タイプ 2 AO コマンド 403
 - IMS がインタレストを登録および登録解除する場合 261
- トランザクション (transaction)
 - イベント・フロー 487
 - 許可 392
 - 高速機能
 - 可能性のある 84
 - 定義 7
 - 出口ルーチンの準備 421
 - 出口ルーチンまたは RACF 392
 - フローと IMS モニターのイベント 865
 - プログラム間通信 392
 - ユーザーのサインオンなしのときのセキュリティ 418
 - RACF パスチケット 421
- トランザクション VSAM (TVS)
 - 使用可能化 693
 - RECON への並列アクセス
 - 使用可能化 693
- トランザクション応答報告書
 - 説明 968
 - 例 968
- トランザクション許可用の DFSCSTRN0
 - モジュール 421
- トランザクションの監査 970
- トランザクション報告書 969
- トランザクション・キューイング報告書
 - DBCTL
 - 例 916
 - IMS モニター (DCCTL) 941

- トランザクション・プロファイル
 - 基本プロファイルの取得 466
 - 重要なエレメント 466
 - 容量計画で 512
- トランザクション・マネージャー
 - リソース 369
 - トランザクション 371
 - ユーザー ID 372
 - ユーザー名 372
 - LTERM (論理端末) 369
 - MSNAME 370
 - VTAM 端末ノード 370
- 取り出し要求エレメント
 - 最適化 527
- トレース 485
 - 外部データ・セットの定義 41
 - 外部データ・セットへの書き込み 40
 - 使用を避けるとき 535
 - プログラム分離およびロック 483
 - ログ 119
 - BPE 外部トレースの開始 42
 - BPE 外部トレースの停止 42
 - BPE コンポーネント 39
 - BPE 内部トレース・テーブルへの書き込み 40
 - CTRACE レコード 302
 - CTRACE、使用 483
 - GTF 詳細トレース報告書 476
 - GTF トレース 482
 - GTF (汎用トレース機能) 476, 497
 - GTFPARS ジョブ要約報告書と詳細トレース報告書 543, 547, 548
 - IMS モニター 459, 476, 497
 - IMSPA DC キュー・トランザクション報告書 543
 - IRLM アクティビティ 302
- トレース機能 485
- トレース報告書 482

[ナ行]

- 内部リソース・ロック・マネージャー (IRLM)
 - データ共用 642
 - データ共用における 298
 - データベース保全性 642
- 内部リソース・ロック・マネージャー (IRLM) (internal resource lock manager (IRLM))
 - z/OS PPT entry 151
 - z/OS PPT の項目 151
- 名前タイプ (name type) 368
- 名前の一意性
 - リソース
 - 使用不可化、制約の 367

- 二重化
 - 構造 76, 225
 - 不要なオーバーヘッド 225
 - 明示的な停止 225
- 入出力エラー処理 721
- 入出力許容
 - オペレーターの表示 780
 - 定義 780
- 入出力サブシステム
 - 構成 537
 - 動的モニター 541
 - トレードオフ
 - IMS とページング 531
- 入出力防止
 - オペレーターによる 778
 - 可用性マネージャーによる 778
 - 完了の確認 762
 - 説明 762
 - XRF テークオーバー (XRF takeover) 762
- 入出力防止 (I/O prevention)
 - オペレーターによる 783
 - 可用性マネージャー (AVM) による 762
 - 完了の確認 783
 - 処理 762
 - 定義 750
- 入出力リソースの競合
 - 競合の除去
 - チューニング 541
- 入力待ち (WFI)
 - トランザクションとトランザクションの間の時間 971
 - プログラム入出力報告書
 - 入力を使用可能でない場合 865
 - 入力を使用可能な場合 936
 - プログラム要約報告書からの除外 877
- ネットワーク
 - 作動可能性の確認 459
 - 定義の変更 507
 - 文書 97
- ネットワーク・セキュリティ資格情報
 - 伝搬 395, 397, 399
- ノード名
 - 一意性 370

[ハ行]

- ハードウェア
 - XRF 要件 756
- ハードウェア要件
 - CQS 71
- 廃棄された RECON、置き換え 723
- 排他意図
 - スケジューリングへの影響 88
- 排他的アクセス 296
- ハイパースペース・バッファ、VSAM
 - バッファ・プール
 - IMS モニター (DBCTL) 917
 - IMS モニター (DB/DC) 887
 - //DFSSTAT 952
 - バインダー戻りコード
 - 解釈 256
 - パスチケット
 - セキュリティ 391
 - パスワード 375
 - マスキング 434
 - ACB の 816
 - パスワード検査
 - 使用可能 390
 - 使用不可 390
 - パスワード保護
 - /LOCK コマンド 394
 - /SET コマンド 394
 - /UNLOCK コマンド 394
 - バックアウト (backout)
 - 概要 136
 - 定義 136
 - 動的
 - 障害 330
 - データ共用 (data sharing) 330
 - バッチの考慮事項 331
 - バッチ・ユーティリティ 331
 - バックアウト、リカバリーのための
 - BACKOUT レコード 706
 - バックアップ
 - 概要 131
 - システム・データ・セット 133
 - データベース 131, 658
 - 用のコマンド 659
 - メッセージ・キュー (message queue) 132
 - RECON データ・セット 716
 - バックアップ・コピー
 - 定義 131
 - バックアップ・セッション、クラス 1 端末に対する
 - オープン 758
 - クローズ 758
 - ログオン 758
 - バックエンド IMS
 - 共用キュー環境での 259
 - バッチ環境
 - 概要 19
 - 例 19
 - TM 19
 - 有効な領域 19
 - Db2 for z/OS、接続 19
 - バッチ更新プログラム
 - スケジューリング 322
 - バッチ端末シミュレーター 461

- バッチ・ジョブ
 - 許可 683
 - DBRC を使用したトラッキング 120
- バッチ・ジョブ、BMP ジョブへの変換 344
- バッチ・バックアウト
 - DBRC サポート 649
- バッチ・バックアウト・ユーティリティ
 - SSID 命名規則 733
- バッチ・バックアウト・ユーティリティ (DFSBB00)
 - 緊急時再始動、関係 653
 - 再始動、関係 653
 - の紹介 136
 - DBRC 関連 649
- バッチ・メッセージ処理 (BMP) 領域
 - バッチ・ジョブ、BMP ジョブへの変換 344
- バッチ・ログ・データ・セットの暗号化 377
- バッファ
 - OLDS 129
- バッファ無効化 (buffer invalidation) 336
- バッファ・サブプールに対する DFSVSMxx
 - チューニングでの使用 531
 - IOBF の使用 531
 - OPTIONS ステートメントでの VSAMFIX の使用 531
- バッファ・プール
 - 最適化 526, 527
 - 要件の分析 526, 527
 - 割り振り 526
- バッファ・プール (buffer pool)
 - 考慮事項 531
- パフォーマンス
 - 概要 516
 - 管理
 - 変更 517
 - 基準 466
 - 計画
 - 共用キュー環境内 538
 - 設計変数 518
 - チューニング・エイド
 - リスト 515
 - ネットワークに関する考慮事項 535
 - 分析ツール
 - RMF II 515
 - 報告書作成エイド
 - モニターの一環として 476, 480
 - 目標 466, 516
 - ワークロード・マネージャー 468
 - 要因
 - システム初期設定の 518
 - 目標 466
- パフォーマンス (続き)
 - IMS オプションの選択 520
 - MSC の 486
 - NCP (ネットワーク制御プログラム) に関する考慮事項 535
 - VTAM 考慮事項 535
- パフォーマンス管理
 - 変更 517
- パフォーマンス調査
 - オンライン・システム設計 508
- パフォーマンス目標
 - ワークロード・マネージャー 468
- パラメータ
 - オンライン IMS 用の DBRC 645
 - ALTRESL 142
 - HLQ 141
 - IMS 制御領域実行パラメータ 定義 273
- 汎用トレース機能 (GTF)
 - 詳細モニターでの使用 476
- 汎用パフォーマンス分析報告プログラム (GPAR)
 - 詳細モニターでの使用 476
- 汎用リソース
 - 類似性
 - IMS 管理対象 356
 - APPLID 名 359
 - MNPS 名
 - セッションの開始 357
 - VTAM 362
 - 計画 355
 - 使用についての制約事項 355
 - XRF
 - MNPS 358
 - USERVAR 358
- 汎用リソース名
 - 指定
 - GRSNAME= 始動パラメータ 357
- 汎用リソース名 (generic resource name)
 - 指定 357
 - VGRS パラメータ 357
- 汎用リソース・グループ
 - 要件 355
- 汎用リソース・グループ (generic resource group)
 - IMS システムの除去 361
 - /START VGRS コマンド 361
 - /STOP VGRS コマンド 361
 - 非アクティブ ACB ライブラリー
 - オンラインでのサイズ変更 505
 - 非標準マクロ 149
 - 非メッセージ・ドリブン・プログラム (non-message-driven program)
 - BMP 8
 - IEP ユーティリティ 8
- 表示バイパス・セキュリティ 434
- 標準外イメージ・コピー・データ・セット 説明 670
- ヒント
 - DBRC の使用 727
 - PRILOG 圧縮が機能しない 717
 - PRILOG レコード・サイズ 717
 - RECON データ・セット内の最後の SLDS 停止時刻の検出 727
- プール・スペース障害要約報告書
 - IMS モニター (DBCTL) 918
 - IMS モニター (DB/DC) 890
 - IMS モニター (DCCTL) 944
- プール・スペースの障害
 - 解決 528
- ファイル選択およびフォーマット設定印刷
 - ユーティリティ (DFSERA10) 445
- ファイル選択ユーティリティ 211
- ファジー・イメージ・コピー
 - 定義 659
- ファシリティ・クラス 209
- フィールド・レベル・センシティブティ (field-level sensitivity) 436
- フォーマット設定印刷ユーティリティ 211
- 複数コールド・スタート
 - テスト環境 737
- 複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC))
 - 分布付録出力の解釈 949
 - リンク
 - XRF に対する定義 818
 - IMSplex との共存 24
- 複数のシステム障害
 - IMS 再始動の同期化 353
- 複製
 - 共用キュー構成 266
- 複製構成
 - 共用キュー 266
- 物理的セキュリティ 434
- プリンター
 - 関連した 282
- プログラム
 - 直列化
 - シスプレックスでの管理 206
 - GBL_SERIAL_PGM 206
 - GBL_SERIAL_PGM 206
 - プログラム一時修正 (PTF)
 - XRF 複合システムでの適用 785
- プログラム間
 - 共用キュー
 - 宛先作成出力ルーチン (DFSINSX0) 263
 - 未定義の宛先
 - 宛先作成出力ルーチン (DFSINSX0) 263

プログラム式暗号機能 437
 プログラム仕様ブロック (PSB)
 アプリケーション制御ブロック
 (ACB)、IMS 管理対象、活動化 554
 活動化
 データ共用および IMS 管理の
 ACB 環境 559
 IMS 管理の ACB 環境、データ共
 用による 559
 活動化、IMS 管理の ACB 環境での
 554
 ACB (アプリケーション制御ブロッ
 ク)、IMS 管理対象、活動化 554
 プログラム入出力報告書
 イベント、分布 862
 説明 855
 分布されたイベントのデフォルト範囲
 862
 報告書の使い方 855
 報告書のフィールド 855
 IMS モニター
 DB/DC 880
 DCCTL 936
 IMS モニター (DBCTL) 915
 プログラム分離 (PI) (program isolation
 (PI))
 DB/DC 環境での 88
 プログラム分離トレース報告書ユーティリ
 ティー (DFSRIIP0) 483
 プログラム要約報告書
 IMS モニター
 DBCTL 913
 DB/DC 877
 DCCTL 934
 プログラム・スケジューリング
 概要 81
 プログラム・テスト、SYSIN/SYSOUT を
 使用した 464
 プログラム・デッドロック
 解決方法 88
 プログラム・ビュー
 アプリケーション制御ブロック
 (ACB)、IMS 管理対象、活動化 554
 オンライン変更
 IMPORT コマンドの UPDATEPSB
 オプション 561
 IMS 管理の ACB 環境 555, 557,
 562, 563
 オンライン変更 (online change)
 管理対象 ACB 環境 553
 活動化
 データ共用および IMS 管理の
 ACB 環境 559
 IMS 管理の ACB 環境、データ共
 用による 559
 プログラム・ビュー (続き)
 活動化、IMS 管理の ACB 環境での
 554
 ACB (アプリケーション制御ブロッ
 ク)、IMS 管理対象、活動化 554
 プログラム・ライブラリーの最適化 547
 プログラム・ロード
 プリロード・オプション 533
 プロシージャ
 DBRC を含む 645
 プロセス継続性 86
 ブロック・サイズ
 OLDS 127
 SLDS 129
 ブロック・レベル共用 (block-level
 sharing) 291, 298
 フロントエンド IMS
 共用キュー環境での 259
 フロントエンド切り替え計画 283
 分散セキュリティ資格情報の伝搬
 伝搬 395, 397, 399
 分散データ管理 (DDM)
 Open Database Manager (ODBM)
 概要 47
 管理 171
 分散リレーショナル・データベース・アク
 セス (DRDA)
 Open Database Manager (ODBM)
 概要 47
 管理 171
 文書
 システム定義 (system definition) 97
 実動構成 97
 端末プロファイル 97
 ネットワーク 97
 文書化
 データ・ディクショナリー 97
 IBM DB/DC データ・ディクショナリ
 ー 97
 IMS システム 97
 分布付録報告書
 解釈 897
 出力 923
 出力の解釈 923
 説明 860
 デフォルトの範囲の再定義 863
 報告書の作成 862
 IMS モニター (DBCTL) 920
 IMS モニター (DB/DC) 896
 IMS モニター (DCCTL) 945
 分類規則 472
 ページ固定
 拡張ストレージ 520
 制御領域の 520
 メンバー DFSFIxx の使用 520
 DREF ストレージ 520
 並行イメージ・コピー (CIC) (concurrent
 image copy (CIC))
 制約事項 663
 データベースのバックアップ・コピー
 663
 登録されるデータベース、DBRC に
 663
 並列 RECON アクセス 168
 並列シスプレックス
 IMSplex との比較 20
 並列セッション・サポート 163
 ヘッダー・レコード
 RECON データ・セット 704
 変換
 バッチ・ジョブを BMP ジョブへ 344
 変更
 システム設計 501
 変更、アプリケーションの
 アクティブ IMS システムでの 504
 変更、割り振りおよび割り振り解除 639
 変更累積
 グループ 124
 再使用 675
 使用 676
 定義 674
 定義、将来の利用のための 675
 除去時刻 122
 処理対象がない旨のメッセージ 735
 データ・セット
 再使用 124
 定義 124
 命名規則 657
 定義 631
 ユーティリティ 121
 レコード
 グループ 707
 実行 707
 ログ 671
 ログ処理の停止、稼働中 734
 変更累積データ・セット
 リカバリ期間 666
 変更累積データ・セットの暗号化 378
 変更累積によるログ処理の誤停止 734
 報告書出力順序
 IMS モニター報告書
 DBCTL 905
 報告書
 システム 482
 ジョブ要約 482
 統計分析ユーティリティ 445
 トレース 482
 呼び出し要約 458
 IMS モニター 480
 IMSASAP II 459
 報告書、IMS
 使用 839

報告書の出力順序
 IMS モニター報告書
 DBCTL 905
 保護、RACF プロシージャおよび
 DLISAS プロシージャによるリソース
 の 435
 保守、グローバル・リソース情報の 201
 保守上の変更点
 考慮事項 239
 サービス・レベル 243
 指針 242
 実動システム 242, 243, 244
 推奨 242
 適用前の準備状況の査定 242
 テスト・システムによる検査 456
 SYSGEN
 SYSMOD 回帰、回避 253
 SYSMOD
 回帰、回避 253
 保守に関する問題 253
 ボトルネック
 メッセージ処理、検出 971

[マ行]

マージ、ログの
 データ共有 (data sharing) 673
 マイグレーション
 作業 256
 マクロ
 HOST 165
 マクロ・キーワード
 SECURITY マクロの SECLVL 401
 SECURITY マクロの TERMNL 401
 マスター IMS 制御領域 55
 マスター端末 (master terminal)
 セキュリティー
 サインオン検査の使用 401
 セキュリティー・オプションの実施
 401
 定義 10
 VTAM の 817
 IMSplex、内 182
 命名規則
 イメージ・コピー・データ・セット
 656
 確立 95
 重複イメージ・コピー・データ・セッ
 ト 657
 データ共有 295
 バッチ・バックアウトによって処理さ
 れる SSID の 733
 ヒント 95
 変更累積データ・セット 657
 例 95
 DBRC データ・セット 655

命名規則 (続き)
 HALDB 95
 RECON SSYS レコードの SSID の
 733
 メッセージ
 キューに入れられて送信されなかった
 報告書の例 970
 プログラム間報告書の例 969
 報告書の例 970
 メッセージ (message)
 キュー (queue)
 組み込み代替 773
 テークオーバー中の処理 778
 ログに記録されている 774
 急送メッセージ処理 84
 共用キューでロックされた
 概念 262
 形式
 オプション 520
 高速機能専用 84
 出力 7
 トランザクションと 7
 入力 7
 フロー
 共用キュー環境での 262
 CQS0033A 212
 CQS0034A 224
 CQS0205E 217
 IXC585E 220
 IXC586I 220
 MTO 265
 メッセージおよびコード
 Base Primitive Environment (BPE) 37
 メッセージ形式サービス (MFS)
 形式のオンライン・テスト
 オンライン実行要件 462
 システム定義要件 462
 MFSTEST モード 461
 テスト形式 462
 IMSGEN マクロの
 MFSTEST=YES 462
 MFSTEST モード 461
 メッセージ形式バッファ・プール
 最適化 527
 メッセージ形式バッファ・プール報告書
 IMS モニター (DB/DC) 884
 IMS モニター (DCCTL) 939
 メッセージ経路指定
 パフォーマンス
 分析 486
 メッセージ処理プログラム (MPP)
 (message processing program (MPP))
 スケジューリング 81

メッセージ通信
 共用キュー
 宛先作成出口ルーチン
 (DFSINSX0) 263
 未定義の宛先
 宛先作成出口ルーチン
 (DFSINSX0) 263
 メッセージ・キュー (message queue)
 削除 275
 バックアップ 132
 メッセージ・キューおよびデータベース・
 データのリカバリ
 XRF 780
 メッセージ・キュー・プール
 最適化 526
 メッセージ・キュー・プール報告書
 IMS モニター
 DB/DC 885
 IMS モニター (DCCTL) 940
 メッセージ・キュー・リスト構造
 共用キュー環境での定義 270
 定義
 高速機能 270
 メッセージ・ドリブン・プログラム 84
 メッセージ・ドリブン・プログラム
 (message-driven program)
 IFP アプリケーション・プログラム 8
 MPP 8
 メモリー・ベースのデータ・セット ENQ
 管理
 使用可能化 156
 メンバー DFSHSBxx の LNK パラメー
 ター 768
 メンバー DFSHSBxx の監視パラメーター
 768
 戻りコード
 オンライン変更失敗 597
 バインダー
 解釈 256
 INITIATE OLC コマンド 590
 モニター
 アプリケーション・プログラムの
 DL/I 呼び出しのための入出力
 DCCTL 936
 アプリケーション・プログラムの経過
 時間
 DCCTL 934
 回線アクティビティ
 DB/DC 889
 DCCTL 942
 構造 315, 448
 高速機能システム 487
 ご使用のシステム 465
 システム 448
 従属領域
 DBCTL 909

モニター (続き)
 従属領域 (続き)
 DB/DC 873
 DCCTL 930
 ツール 480
 データ共用システム 313
 データベース・バッファ
 DBCTL 917
 DB/DC 887
 テスト環境での 458
 動的 541, 543, 545
 度数分布の使用
 DBCTL 920
 DB/DC 893
 DCCTL 945
 内部リソースの使用
 DBCTL 918
 DB/DC 890
 DCCTL 944
 パフォーマンス目標
 ワークロード・マネージャー 468
 頻度 476
 複数システム 465
 プロシージャ 465, 493
 方法 465, 493
 メッセージ処理
 DB/DC 890
 DCCTL 943
 メッセージ・キューの処理
 DBCTL 915
 DB/DC 880, 885
 DCCTL 940
 目標の確立 495
 理由 465
 ワークロード・マネージャー 472
 DBCTL 497
 DBCTL に関する考慮事項 493
 DISPLAY (/DISPLAY) コマンド 476, 497
 IMS 445
 IMS モニター 480
 IMS モニター・ユーザー出口 465
 IRLM アクティビティ 313, 448
 MFS アクティビティ
 DB/DC 884
 DCCTL 939
 /DISPLAY コマンド 476, 497
 モニター、IMS 458
 モニターのトレース間隔
 IMS モニター報告書
 DBCTL 906
 DB/DC 870
 DCCTL 927
 モニター・オーバーヘッド報告書
 説明 864
 報告書のフィールド 864

モバイル・ワークロード 472
 モバイル・ワークロードの価格設定 474

[ヤ行]

ユーザー ID
 シスプレックス内の TM リソースとして 372
 IMSplex、内 372
 名前の一意性 372
 ユーザー、サインオンなしのときのセキュリティ 418
 ユーザー管理再作成 222
 ユーザー名
 シスプレックス内の TM リソースとして 372
 IMSplex、内 372
 名前の一意性 372
 ユーティリティ
 解釈
 DCCTL の IMS モニター報告書 925
 IMS モニター報告書、DBCTL の 903
 //DFSSTAT 報告書 951
 グローバル・オンライン変更 601
 高速機能ログ分析ユーティリティ (DBFULTA0) 445
 チューニング 515
 データベース変更累積 121
 データベース・リカバリー管理ユーティリティ (DFSURDB0) 649
 統計分析報告書およびログ・トランザクション報告書の解釈 967
 ファイル選択 211
 フォーマット設定の印刷 211
 DB モニター報告書の読み方 839
 VSAM AMS 721
 ユーティリティ制御機能 (UCF)
 DBRC、との関係 652
 ユーティリティ制御ステートメント
 INIT.CA 675
 INIT.CAGRP 675
 有効なログ・サブセット、データ共用においてサイズを圧縮するための 673
 要件
 分析 93
 CQS
 ソフトウェア 71
 ハードウェア 71
 IMS システム 93
 予測
 暗黙的 518
 明示的 518
 予定テークオーバー
 制限 785

予定テークオーバー (続き)
 IMS システム定義の変更 785
 予定テークオーバーの実際の用途 785
 呼び出し可能サービス
 グローバル 368
 呼び出し要約報告書 458
 IMS モニター
 DB/DC 877
 DCCTL 934
 IMS モニター (DBCTL) 913
 予防サービス
 プログラム一時修正 (PTF) 239
 プログラム診断依頼書 (APAR) 239
 予防保守
 インストール 245
 IMSplex 252
 読み取りアクセス 296
 読み取り専用アクセス (read-only access) 296

[ラ行]

ライブラリー
 アクティブおよび非アクティブ 570
 ライブラリー・ルックアサイド機能
 プログラム・ライブラリー 533
 ラッチ競合統計報告書
 IMS モニター (DBCTL) 918
 IMS モニター (DB/DC) 890
 IMS モニター (DCCTL) 944
 ランタイム・リソース定義
 変更 581
 リカバリー
 オーバーヘッド 140
 関連する機能の実行 446
 期間、イメージ・コピー・データ・セットの 665
 期間、変更累積データ・セットの 666
 機能 330
 逆方向
 定義 136
 グループ 707
 計画 329
 構造 224
 サービス
 概要 133
 時刻指定 632
 システム要件 105
 シスプレックス (sysplex)
 データ共用 (data sharing) 349
 順方向
 データ共用 (data sharing) 332
 定義 137
 情報 71
 処理 105
 ステップ 105

- リカバリー (続き)
 - 説明 103
 - タイム・スタンプ 632
 - データ共用 (data sharing) 332
 - シスプレックス (sysplex) 349
 - 手順の計画 329
 - DBRC を使用しない 652
 - IRLM 333
 - データ共用における
 - 手順の計画 649
 - トランザクション・リカバリー 333
 - データベース
 - 順方向リカバリー、データ共用による 649
 - 動的バックアウト、データ共用による 649
 - リカバリー・メカニズムの設定 649
 - DBRC の役割 635
 - DBRC を使用しない 652
 - 手順、検討 94
 - 動的バックアウト (dynamic backout) 330
 - の紹介 103
 - バッチ・バックアウト 331
 - 複雑性 139
 - 保守、オンライン再編成用のレコードの 681
 - メカニズム 106
 - メカニズム、セットアップ 649
 - ユーティリティ
 - 概要 133
 - 入力 114
 - ユーティリティ、への入力 113
 - リカバリー、CQS の 221
 - リカバリー・メカニズムのないシステムの例 104
 - レコード (RECOV) 714
 - ログの使用 108
 - CQS 機能 77
 - DBCTL 環境 (DBCTL environment) 134
 - DBRC
 - 概要 134
 - IMS リカバリーが不可能な場合 139
 - IMSplex 内 25
 - RECON データ・セット 721
 - XRF 環境 135
- リカバリー、データベース
 - 処理の概要 631
 - リカバリー、データベース
 - 概念 631
- リカバリー、データベースの
 - バッチ・サポート 649
 - ログ・レコードのアーカイブ 627
- リカバリー、RECON データ・セットの
 - RECON 損失の通知 725
- リカバリー可能サービス・エレメント (RSE) (recoverable service element (RSE))
 - 定義 745
- リカバリー可能データベース
 - イメージ・コピー・ユーティリティ 659
 - CHANGE.DB コマンド 659
 - INIT.DB コマンド 659
- リカバリー管理データ・セット 691
 - 入出力エラー処理 721
- リカバリー期間
 - 定義 665, 666
- リカバリー・ユーティリティ
 - データベース 636
- リカバリー・レコード、データベース 706
- リカバリー・ログ・データ・セット (RLDS) (recovery log data set (RLDS))
 - 圧縮 120
 - コピー 118
 - 作成 118
 - 定義 108, 130
 - レコード 705
 - ログ 114
- リセット、GENMAX パラメーターの 729
- リセット、GRPMAX パラメーターの 730
- リソース
 - オンラインの変更 551
 - 構造 75, 76
 - リカバリー 224
 - ログに記録される変更 211
 - タイプ整合性 368
 - データ共用環境でのモニター 313, 448
 - 名前の一意性 367
 - ユーザー名 372
 - LTERM 369
 - VTAM 端末ノード 370
- 変更 447
- IMSplex、内
 - 管理 367
 - 呼び出し可能サービス (callable services) 368
- TM リソース
 - トランザクション 371
 - ユーザー ID 372
 - ユーザー名 372
 - MSNAME 369, 370
 - VTAM 端末ノード 370
- リソース構造
 - 直列化プログラム 201
 - 保管されたリソース情報
 - データベース用 204
 - トランザクション用 205
- リソース構造 (resource structure) 58, 201
- リソース構造 (resource structure) (続き)
 - オンライン変更 (online change)
 - リソース構造の場合の制約事項 55
 - 構成 60
 - 再作成 208
 - 障害 207
 - 定義 270
 - 保管された情報 201
 - 保管されたリソース情報
 - DEADB エリア用 204
 - リカバリー 207
 - CQS 208
 - IMSplex
 - トランザクション・マネージャー・リソース 205
 - メッセージ宛先リソース 205
 - リソース使用状況
 - 変更 508
 - リソース測定機能 II
 - モニター・ツールとして 476
 - リソース測定機能 II (RMF II)
 - ページング率の使用 476
 - リソース定義
 - IMSRSC リポジトリ
 - 表示 233
 - リソース名の一意性
 - 制約の使用不可化 367
 - リソース・アクセス・セキュリティ
 - (RAS) (resource access security (RAS))
 - 従属領域 417
 - RACF リソース・クラス 417
 - リソース・クラス
 - IMS 定義への対応 422
 - RACF
 - リソース・アクセス・セキュリティ 417
 - リソース・タイプ整合性
 - 制約の使用不可化 368
 - リポジトリ・サーバー (RS) (Repository Server (RS))
 - IMSplex の一部 23
 - リモート処理
 - 共用キュー 259
 - MSC (複数システム結合機能) 259
 - 理由コード
 - オンライン変更失敗 597
 - INITIATE OLC コマンド 590
 - 領域 IWAIT 報告書 526
 - 領域およびジョブ名報告書
 - IMS モニター
 - DB/DC 870
 - DCCTL 927
 - 領域占有
 - 測定 877, 933

領域待機報告書	レコード (続き)	ログ (続き)
IMS モニター	RECON データ・セットのヘッダー	WADS 111
DBCTL 909	704	z/OS 115
DB/DC 873	レコード再始動 71	ログイン、MSC への 486
DCCTL 930	レポート (モバイル・ワークロード) 474	ログオン出口ルーチン (DFSLGNX0)
領域別プログラム報告書	ローカル共用リソース (LSR) オプション	ETO を使用して 415
IMS モニター	700	ログオン・プロシージャ
DBCTL 909	ローカル処理	汎用リソース名 (generic resource
DB/DC 873	定義 259	name) 359
DCCTL 930	ローカル・オンライン変更 (local online	IMS APPLID 名 359
領域要約報告書	change)	ログオン・メッセージ
領域占有率のパーセンテージ 523	概要 569	処理 763
NOT-IWAIT 時間の意味 476	コールド・スタート (cold start) 601	図 763
領域要約報告書 (REGION SUMMARY	コマンド・シーケンス 573	例 763
REPORT)	サポートされた環境 551, 573	ログ関連のコマンド
IMS モニター	ローカル・オンライン変更機能	API 要求
DB/DC 873	概要 570	ログ照会 628
DCCTL 930	ロギング 77	OLDS 照会 628
IMS モニター (DBCTL) 909	圧縮、DFSUCUM0 を使用したログの	CHANGE.PRILOG 628
利用可能トランザクション、高速機能 84	671	CHANGE.RECON 628
リンク、IMS から z/OS への 152	概要 108, 625	CHANGE.SECLOG 628
リンク・キューイング時間の評価 902	削減、高速機能に対する 119	DELETE.LOG 628
類似性	状態 125	GENJCL.ARCHIVE 628
管理、バイパス 364	単一 125	GENJCL.CLOSE 628
終了 359	重複 125	LIST.LOG 628
例	定義 110	NOTIFY.PRILOG 628
構造フルしきい値の表示 220	累積、DFSUCUM0 を使用したログの	NOTIFY.SECLOG 628
データ共用 (data sharing) 305	671	ログ情報 625
二重化の明示的な停止 225	IMS イベント 625	ログの使用
CQS 構造への接続を許可するための	ログ	概要 108
RACF コマンド 213	アーカイブ 116	ログ保存ユーティリティ
CQS 登録を許可するための RACF コ	圧縮 130	(DFSUARC0) 111, 113
マンド 212	概念 86	アーカイブ
レコード	再始動での使用 86	手動 116
オンライン再編成 (online	削減 130	説明 627
reorganization) 678	システム・ユーティリティ 445	出口ルーチンによるカスタマイズ 119
再編成 712	障害、XRF の制限 750	OLDS の保存 116
サブシステム 714	データ共用 (data sharing) 328	ログ割り振りレコード 713
データ共用 (data sharing) 714	データ・キャプチャー出口ルーチン	ログ・ストリーム
データベース 708	130	z/OS
データベース・データ・セット 708	データ・セット 110	定義 272
データベース・データ・セット・グル	選択の指定 125	ログ・データ・セット
ープ 707	特性の制御 449	移動、DBRC への通知 736
データベース・リカバリー (database	トレース 119	ログ・データ・セット、レコード 705
recovery) 706	内容 130	ログ・トランザクション分析ユーティリテ
変更累積グループ 707	の紹介 108	ィー (DFSILTA0) 445
変更累積実行 707	変更累積 121	キュー時間の節約 526
ログ割り振り 713	レコード	作成される報告書
ログ・データ・セット 705	印刷 445	説明 971
BACKOUT 706	報告書 445	ログ分析報告書 971
DBDS グループ 707	OLDS (オンライン・ログ・データ・	チューニング・エイドとしての 515
HALDB オンライン再編成 678	セット) 110	複数システムで使用 465
RECON	RECON での制御レコード	モニターに使用する 476
不要レコードの削除 716	変更 627	ログ・ボリューム
保守 716	RLDS 114	変更累積用の指定 673
RECON データ・セット内の 704	SLDS 113	

ログ・レコード
 分析 967
 OM 監査ログ
 レコード・フォーマット 185
 OM、印刷 186
 RECON からの削除 718
 ログ・レコードの削除、方法 716
 ロック構造
 サイズの計算 345
 ロックされたメッセージ
 共用キューで 262
 論理端末 (LTERM)
 共用キュー
 IMS がインタレストを登録する場合 262
 リソース・タイプ整合性 368
 論理端末 (LTERM) (logical terminal (LTERM))
 シスプレックス内の TM リソースとして 369
 名前の一意性 367, 369
 RM に対する定義 369
 論理の検討 94

[ワ行]

ワークロード・マネージャー 465
 業務の重要度 468
 パフォーマンス目標 466, 468
 分散 465
 モバイル・ワークロード 472
 割り振り
 データベース・レコード (database record) 713
 ロギング、RECON データ・セット内の 639
 ログ・レコード 713
 OLDS
 OLDS の定義 110
 RECON データ・セット 691
 SLDS 113
 WADS 111
 割り振り解除
 ロギング、RECON データ・セット内の 639

[数字]

2 次端末
 XRF 817
 37x5 通信コントローラー
 アクティブ IMS システムと代替 IMS 間の ISC リンク 758
 ゲートウェイの考慮事項 811
 XRF に対する要件 756

A

ACB
 IMS 管理対象
 オンラインでの PSB 変更の活動化 561
 オンライン変更 (online change) 553
 ACB (アプリケーション制御ブロック)
 オンラインの変更 591
 オンライン変更 591
 活動化、データ共用環境での ACB の 559
 管理対象 ACB
 活動化、データ共用環境での ACB の 559
 データベース・リソースの活動化、概要 554
 管理対象 ACB 環境での活動化、概要 554
 パフォーマンスのための配置 520
 ライブラリー
 オンラインでのメンバーの変更 591
 メンバー・オンライン変更 591
 割り振り 505
 ライブラリー・データ・セット
 動的割り振り 35
 動的割り振り (dynamic allocation) 35
 64 ビット・ストレージへの配置 35
 ACB の IMS 管理
 活動化、データ共用環境での ACB の 559
 データベース・リソースの活動化、概要 554
 XRF のパスワード 816
 ACB メンバー
 オンライン変更 (online change)
 関連付け 582
 DRD コマンドによる関連付け 582
 ACB ライブラリー
 グローバル・オンライン変更 (global online change) 591
 非アクティブ・ライブラリー (inactive library)
 オンラインでのサイズ変更 505
 メンバー・オンライン変更 57, 591
 ACB ライブラリー・メンバー・オンライン変更
 概要 591
 ACBLIB
 グローバル・オンライン変更 (global online change) 57
 メンバー・オンライン変更 57

ACBGMGT
 ACB の IMS 管理
 新規データベース・リソースの活動化 554
 ACBGMGT (ACB の IMS 管理)
 データ共用環境
 リソースの活動化 559
 データ共用環境でのリソースの活動化 559
 ALTRESL パラメーター 142
 AMS 716, 721
 AO アプリケーション
 ユーザー ID の代替 405
 AO アプリケーション・プログラム・セキュリティ
 AOI1= 実行パラメーター 402
 AO コマンド・セキュリティ
 コマンド許可出口ルーチン 407
 初期設定 EXEC パラメーター 404
 RACF を使用した実施 406
 AO (自動化操作プログラム) のアプリケーション・プログラム
 セキュリティ 401
 説明 85
 動的モニター 476
 ICMD 呼び出し 401
 AOI (自動化操作プログラム・インターフェース)
 説明 134
 AOI トランザクション
 スケジューリング 287
 AOI1= 実行パラメーター
 AO アプリケーション・プログラム・セキュリティ 402, 403
 APAR
 指針 242
 実動システム
 推奨 242
 保守プロセスの例 244
 推奨 242
 APAR (プログラム診断依頼書)
 SMP/E を使用 239
 APF 許可
 必要とする、z/OS が 159
 IRLM に必要 161
 z/OS インターフェースの場合 148
 API (アプリケーション・プログラミング・インターフェース)
 セキュリティ 684
 API 要求
 セキュリティ 684
 APPC (拡張プログラム間通信機能)
 記述子 (descriptor)
 TM リソースとして 369
 汎用リソース名、指定 357

APPC (拡張プログラム間通信機能)/IMS
共用キュー環境 284
同期メッセージ 284
非同期メッセージ 284
セキュリティ 415
説明 79
トランザクション・プロファイル名 97
APPC/MVS 管理ダイアログの更新 160
APPLCTN マクロ・ステートメント
オンライン変更に使用 510
APPLID= キーワード
XRF のための VTAM アプリケーション
名の定義 816
APSB SAF セキュリティー
使用可能化 426
使用不可化 426
ARM 165
AVM (可用性マネージャー)
オペレーターへのメッセージの送信
762
オペレーター・メッセージ 762
立ち上げる 771
定義 762
入出力防止の実行 762
メッセージの接頭部 762

B

BACKUP
定義 763
BACKUP= キーワード
クラス 1 端末の定義 804
BMP (バッチ・メッセージ処理) 領域
アプリケーション・プログラム 8, 438
高速機能 85
特性 8
バッチ・ジョブ、BMP ジョブへの変
換 344
DBCTL 環境 85
BPE (基本プリミティブ環境)
外部データ・セットへの書き込み 40
外部トレースの開始 42
外部トレースの停止 42
外部トレース・データ・セットの定義
41
構成 37
コマンド 37
提供されたサービス 37
トレース
使用可能化 39
内部トレース・テーブルへの書き込み
40
メッセージ 37
ユーザー出口ルーチン 37
リスト PROCLIB メンバー 199
BPE と IMS の関係 37

BPE (基本プリミティブ環境) (続き)
BPE を使用するコンポーネント 37
CSL との関係 43
z/OS PPT entry 150
z/OS PPT の項目 150
BPE トレース・データ・セットの暗号化
378
BTS (バッチ端末シミュレーター)
オンライン実行のシミュレーション
461
特長 461

C

CANCEL コマンド
非 XRF 作業負荷 798
CCTL スレッド
異常終了 88
高速機能 (Fast Path) 85
デッドロック 88
PSB のスケジューリング 85
CCTL 領域 12
アプリケーションのパフォーマンス
523
パフォーマンス目標 495
モニター 493
DRA リソースに対する競合 523
PSB 要求 523
CFRM (カップリング・ファシリティ・
リソース管理)
結合データ・セット・フォーマット・
ユーティリティ 225
ポリシー、定義の 214
CFRM ポリシー
定義 270
例 270
CHANGE (/CHANGE)
SURVEILLANCE コマンド
XRF の使用法 758
CHANGE.PRILOG コマンド
RECON ログ制御レコードの変更 627
CHANGE.RECON コマンド
割り振り、RECON データ・セットへ
のスペースの 699
RECON の再編成 720
RECON のリカバリー 721
CHANGE.SECLOG コマンド
RECON ログ制御レコードの変更 627
CHECKPOINT (/CHECKPOINT)
コマンド 87
CIC パラメーター
オンライン・イメージ・コピーの 663
CMD 呼び出し
セキュリティの概要 401
CMDSEC= パラメーター 199
COBOL サブルーチンのプリロード 533

COMCYCL パラメーター 165
COMM マクロ・ステートメント
APPLID キーワード 816, 817
PASSWORD キーワード 816, 817
COMMIT コマンド 575, 580
Common Queue Server (CQS)
アドレス・スペース 43
ウォーム・スタート (warm start) 275
ログ・トークン 275
開始 275
カップリング・ファシリティ
構造サイズの変更 275
計画
ソフトウェア 71
ハードウェア 71
コールド・スタート (cold start) 275
構造
コピー 275
再作成 275
削除 275
使用状況のモニター 275
チェックポイント、開始 275
リカバリー 275
構造リカバリー 76
再始動 275
システム・チェックポイント後 275
システム・チェックポイント (system
checkpoint)
開始 275
再始動 275
データ・セット 275
使用 275
定義
グローバル構造パラメーター 273
実行パラメーター 273
初期設定パラメーター 273
ローカル構造パラメーター 273
ユーザー提供の出口ルーチン 275
リスト構造
計画 280
ロギング 77
OC/390 ロガー 275
IMSplex の一部 23
z/OS PPT 項目 150
z/OS PPT の項目 150
CONTROLINTERVALSIZE キーワード
DEFINE CLUSTER キーワード 700
CPC (中央演算処理装置複合システム)
障害
XRF テークオーバーの原因として
の 750, 776
定義 745
XRF に対する要件 747, 756
CPI-C ドリブ・アプリケーション・プ
ログラム
セキュリティに関する考慮事項 409

CQS SRDS データ・セットの暗号化 379
CQS 管理の再作成 222
CQS (共通キュー・サーバー)
 アドレス・スペース 43
 ウォーム・スタート (warm start) 275
 ログ・トークン 275
オペレーティング・システムの要件 71
開始 275
概要 71
カップリング・ファシリティ
 構造サイズの変更 275
管理 211
機能
 オーバーフロー処理 71
 構造再作成 71
 構造チェックポイント 71
 システム・チェックポイント 71
 要求 71
 レコード再始動 71
許可 212
クライアント障害 212
クライアント接続
 確立 212
クライアント・システムおよびカップ
 リング・ファシリティのダイアグ
 ラム 71
計画
 ソフトウェア 71
 ハードウェア 71
コールド・スタート (cold start) 275
構造
 コピー 275
 再作成 275
 削除 275
 使用状況のモニター 275
 チェックポイント、開始 275
 リカバリー 275
構造オーバーフロー機能 218
構造再作成 221
構造リカバリー 76
構造リカバリー・データ・セット 116
コンポーネント 71
 オーバーフロー構造 71
 基本構造 71
 構造リカバリー・データ・セット
 71
 チェックポイント・データ・セット
 71
 リソース構造 71
 z/OS ログ・ストリーム 71
再始動 275
 システム・チェックポイント後 275
再始動情報 211
再始動のための情報 211
作業の通知 71

CQS (共通キュー・サーバー) (続き)
 システム・チェックポイント (system
 checkpoint) 275
 開始 275
 再始動 275
 データ・セット 275
システム・チェックポイントの再始動
 215
システム・チェックポイント・デー
 タ・セット 115
使用 275
定義 26
 グローバル構造パラメーター 273
 実行パラメーター 273
 初期設定パラメーター 273
 ローカル構造パラメーター 273
ユーザー提供の出口ルーチン 275
リカバリー 221
リスト構造
 計画 280
リソース構造
 属性 60
利点 71
 自動的な作業負荷平準化 71
 信頼性 71
 増分拡張 71
ロギング 77
 OC/390 ロガー 275
IMS コマンドによるアクセス 279
z/OS PPT 項目 150
z/OS PPT の項目 150
z/OS ログ・データ・セット 115
CQS クライアント
 要求 77
CQS の機能 71
CQS のコンポーネント 71
CQS の利点 71
CQS を許可する
 接続 213
 登録 212
CSBLK DSECT 368
CSL (共通サービス層) 20
 アドレス・スペース 47
 概要 43
 管理 165
 基本プリミティブ環境との関係 43
 構成、Resource Manager のない 47
 構成規則 59
 構成の推奨 61
 最小の構成 61
 障害 835
 セキュリティ 33
 マネージャー 47
 利点 20, 43
CSLZQRY 169

CSL (共通サービス層) (続き)
 IMS アドレス・スペース、組み込まれ
 た 43
 Resource Manager
 概要 50
 提供される機能 50
 リソース構造 (resource
 structure) 50
 Structured Call Interface
 概要 51
 提供される機能 51
 XRF 744
CSLDCxxx
 ODBM の構成 27
CSLDIxxx
 ODBM の初期設定 27
CSLOIxxx 27
CSLSIxxx 27
CSLSREG 209
CTC (チャネル間)
 障害、XRF の制限 750
CTRACE サービス 483
CTRACE レコード 302
CYLINDERS キーワード
 DEFINE CLUSTER キーワード 700

D

DATABASE マクロ・ステートメント
 オンライン変更で使用 510
 データ共用 297
DB 区画レコード
 HALDB 708
DB グループ
 コマンド、影響を及ぼす
 CHANGE.DBDSGRP 677
 DELETE.DBDSGRP 677
 INIT.DBDSGRP 677
 LIST.DBDSGRP 677
DB バッチ 19
DB ヘッダー・レコード
 HALDB 708
DB モニター 480
DB モニター報告書
 データベース・バッファ・プール報
 告書 852
 プログラム入出力報告書 855
 分布付録報告書 860
 モニター・オーバーヘッド報告書 864
DL/I 呼び出し要約報告書 857
VSAM 統計報告書 846
VSAM バッファ・プール報告書 839

- Db2 for z/OS
 - データ・キャプチャー出口ルーチン 90
- DB2 Recoverable Resource Manager (DB2 リカバリー可能リソース・マネージャー) サービス接続機能 (DB2RRMS) JMP または JBP アプリケーションの使用 615
- DB2 リカバリー可能リソース・サービス接続機能 (RRSAF) 14
- DB2RRMS (DB2 Recoverable Resource Manager (DB2 リカバリー可能リソース・マネージャー) サービス接続機能) JMP または JBP アプリケーションの使用 615
- DBBBATCH
 - IMSpIex コンポーネント (IMSpIex component) 44
- DBCTL
 - セキュリティ機能 439
- DBCTL 環境 (DBCTL environment)
 - 概要 12
 - コマンド 12
 - サポートされるデータベース 13
 - 端末 12
 - データ共有 (data sharing) 13
 - テスト 460
 - CCTL 12
 - control 12
 - DL/I 12
 - DRA 12
 - XRF 代替 14
- DBCTL 待機緊急時再始動 584
- DBCTL (データベース制御)
 - 概要 134
 - 環境
 - 高速機能 85
 - リカバリー 134
 - XRF 機能 750
- DBCTL (データベース制御) 環境
 - セキュリティ 438
 - モニター 493
- DBD (データベース記述子)
 - アプリケーション制御ブロック (ACB)、IMS 管理対象、活動化 554
 - 活動化
 - データ共有および IMS 管理の ACB 環境 559
 - IMS 管理の ACB 環境、データ共有による 559
 - 活動化、IMS 管理の ACB 環境での 554
 - ACB (アプリケーション制御ブロック)、IMS 管理対象、活動化 554
- DBDS
 - 開始、RECON データ・セットの 647
- DBDs
 - オンライン変更
 - IMS 管理の ACB 環境 555, 557, 562, 563
 - オンライン変更 (online change)
 - 管理対象 ACB 環境 553
- DBDS (データベース・データ・セット)
 - A から J、L および X 678
 - M から V および Y 678
- DBDS (データベース・データ・セット)
 - グループ
 - コマンド、指定する
 - GENJCL.IC 676
 - GENJCL.OIC 676
 - GENJCL.RECEIVE 676
 - GENJCL.RECOV 676
 - GENJCL.USER 676
 - LIST.DBDS 676
 - LIST.HISTORY 676
 - コマンド、定義に影響を及ぼす
 - CHANGE.DBDSGRP 676
 - DELETE.DBDS 676
 - DELETE.DBDSGRP 676
 - INIT.DBDSGRP 676
 - 使用 676
 - レコード 707
 - ILDS (間接リスト・データ・セット) 676
- DBDUMP (/DBDUMP) コマンド
 - データベースのバックアップ・コピー 660
- DBFULTA0 (高速機能ログ分析ユーティリティ) 445
- DBRC
 - 管理 619
 - セキュリティ 683
- DBRC SCI 登録出口ルーチン (DSPSCIX0)
 - IMSpIex 名の定義 28
- DBRC コマンド 684
- DBRC コマンド許可サポート
 - DSPDCAX0 および RACF の両方の使用 685
- DBRC コマンド許可出口ルーチン (DSPDCAX0)
 - 使用法 685
- DBRC セキュリティ
 - セキュリティのオーバーライド
 - RECON データ・セット 690
- DBRC (データベース・リカバリー管理)
 - アクティブ RECON データ・セット 702
 - いつ使用するかの指定 645
 - オープン・オンライン PRILOG レコードのクローズ 732
- DBRC (データベース・リカバリー管理) (続き)
 - オンライン・コマンド
 - /RMCHANGE コマンド 622
 - /RMDELETE コマンド 622
 - /RMGENJCL コマンド 622
 - /RMINIT コマンド 622
 - /RMLIST コマンド 622
 - /RMNOTIFY 622
 - 概要 8, 134, 619
 - 共用レベル、割り当て 642
 - 共用レベルの割り当て 642
 - 許可
 - 変更、データベースの 734
 - 許可のためのリソース名 686
 - 記録、イメージ・コピーの 658
 - 記録、変更累積の 674
 - 区分データ・セット・メンバー 620
 - グループ
 - 定義 677
 - 考慮事項、使用上の 655
 - コマンド
 - 発行 620
 - INIT.CA 675
 - INIT.CAGRP 675
 - コマンド許可 686
 - セキュリティ・プロダクトの使用 684
 - DSPDCAX0 および RACF の両方の使用 685
 - DSPDCAX0 の使用 685
 - コンポーネント 620
 - 再始動 333
 - サブシステム (SSYS) レコード 733
 - システム考慮事項 655
 - 実行されるタスク 621
 - 自動 RECON 損失通知 58
 - 使用、ヒントの 727
 - 使用する時期 619
 - 初期 RECON アクセス 702
 - 初期設定 645
 - スペアの RECON データ・セット 702
 - セキュリティ 683, 684
 - セキュリティ・リソース・プロファイル 684
 - 妥当性検査、ユーティリティ JCL の 636
 - データ共有
 - コマンド 320
 - データ共有 (data sharing) 319, 332, 448
 - データ共有管理 641
 - データ共有管理、および 291, 293
 - データ共有レコード 641
 - データベースの登録 647

DBRC (データベース・リカバリー管理)
(続き)

データベースのバックアップ
イメージ・コピー・データ・セット
および GENMAX のリカバリー
期間 665
オンライン・データベース・イメ
ージ・コピー (DFSUICP0) 663
再利用、イメージ・コピー・デー
タ・セットの 668
指針 671
将来の利用のためのイメージ・コピ
ー・データ・セットの作成 664
制御、管理されるイメージ・コピー
数の 665
データベース・イメージ・コピー 2
(DFSUDMT0) 661
データベース・イメージ・コピー
(DFSUDMP0) 660
並行イメージ・コピー 663
変更累積データ・セットのリカバリ
ー期間、および GRPMAX 666
方式 658
リカバリー、データベースの 635
HISAM コピー 669
データベース・リカバリー管理 631
データベース・リカバリー管理ユーテ
ィリティー (DSPURX00) 620
データ・セット
区分 620
リカバリー要件の定義 631
RECON 620, 716
RECON データ・セット 691
SSYS レコード 714
データ・セットのカタログ管理 736
データ・セット命名規則 655
テスト 457
との通信 622
バッチ・コマンド
BACKUP.RECON コマンド 622
CHANGE コマンド 622
DELETE コマンド 622
GENJCL コマンド 622
INIT コマンド 622
LIST コマンド 622
NOTIFY コマンド 622
RESET.GSG コマンド 622
パラメーター
IMS.PROCLIB 実行パラメーター
645
必要な JCL 更新 308
変更累積によるログ処理の誤停止 734
変更累積の処理対象がない旨のメッセ
ージ 735
骨組み JCL 620
リカバリー・ユーティリティー 636

DBRC (データベース・リカバリー管理)
(続き)

リカバリー・レコード・タイプ 706
ログ関連のコマンド
CHANGE.PRILOG 628
CHANGE.RECON 628
CHANGE.SECLOG 628
DELETE.LOG 628
GENJCL.ARCHIVE 628
GENJCL.CLOSE 628
LIST.LOG 628
NOTIFY.PRILOG 628
NOTIFY.SECLOG 628
ログ制御 625
ログ・データ・セット
移動の通知 736
ログ・レコードの削除 716
API 要求
概要 622
API 要求許可
セキュリティ・プロダクトの使用
684
DBDS の登録 647
DBRC アドレス・スペース、開始 8
DBRC コンポーネント 620
DBRC パラメーター
IMSCTRL マクロ 645
DBRC プロシージャ 645
DSPSCIX0 28
GENMAX パラメーター
使用 665
GENMAX、リセット 729
GRPMAX、リセット 730
IMS からの呼び出し 621
IMS プロシージャと 645
IMSpIex サポート 643
IMSpIex の一部 23
IRLM 状況 653
IRLM を伴うリカバリー 653
JCL の生成 620
OLR、考慮事項 678
PRILOG の圧縮 717
PRILOG レコード
クローズ 732
RECON 初期設定トークン (RIT) 639
RECON データ・セット 115
置き換え、損傷した 721
置き換え、廃棄された
RECON 723
オンラインでの再編成 720
拡張 699
競合、回避 692
競合の回避 692
許可要求サポート 683
計画に関する考慮事項 692
コマンド許可サポート 683

DBRC (データベース・リカバリー管理)
(続き)

RECON データ・セット (続き)
再編成手順 720
再編成の概要 719
再編成前のバックアップ 720
作成 700
作成時の推奨 700
修復 723
初期設定 692
セキュリティの考慮事項 702
損失通知 725
データベースの登録 647
デッドロック状態の回避 695
入出力エラー処理 721
バックアップ 716
並列アクセス 693
並列アクセス、計画 693
ヘッダー・レコード 704
保守 716
リカバリー 721
リカバリー・レコード・タイプ 706
レコード・タイプ 704
レコード・マッピング 715
ログ・データ・セット・レコード
705
ALLOC レコード 713
BACKOUT レコード 706
CA レコード 707
CAGRP レコード 707
DBDS の登録 647
DBDSGRP レコード 707
DEFINE CLUSTER キーワード
700
IMAGE レコード 712
LOGALL レコード 713
LSR オプションを使用した処理
700
RECOV レコード 714
REORG レコード 712
SSYS レコード 714
RECON データ・セット、リカバリー
要件の定義 631
RECON データ・セットの初期設定
645
RECON レコードの保守 716
RECOVPD パラメーター
使用 665
RIT (RECON 初期設定トークン) 639
SLDS 停止時刻の検出、RECON デー
タ・セット内の最後の 727
DBRC によって実行されるタスク 621
DBRC による共用レベルの割り当て 642
DBRC の使用上の考慮事項 655

DB/DC
 セキュリティー
 概要 386

DB/DC 環境
 サポートされるデータベース 10
 制御領域 (control region) 8
 通信設計 509
 データ共用 (data sharing) 11
 定義 7
 例 7
 XRF 11

DB/DC データ・ディクショナリー
 システムの文書化 97
 ネットワークの文書化 97

DCCTL
 セキュリティー
 概要 386

DCCTL 環境
 アプリケーション・プログラム 14
 外部サブシステム 14
 概要 14
 再始動 18
 サブシステム接続 14
 サポートされるアプリケーション呼び出し 17
 サポートされるデータベース 16
 システム・サービス呼び出し 17
 終了 18
 状況コード AD 17
 初期設定 18
 診断 18
 ステージ 1 入力 18
 制御領域 (control region)
 データ通信マネージャー 14
 メッセージ・マネージャー 14
 Transaction Manager 14
 通信設計 509
 データ通信呼び出し 17
 定義 18
 出口ルーチン 18
 テスト 460
 領域タイプ 14
 例 14
 AO トランザクション 18
 DBRC 14
 ESAF 14, 16
 IMS.PROCLIB 16
 RRSAP 16

DDM (分散データ管理)
 Open Database Manager (ODBM)
 概要 47
 管理 171

DEDB (高速処理データベース)
 アクセス意図の変更 319
 オンライン変更 (online change) 575, 581

DEDB (高速処理データベース) (続き)
 高速機能に関する考慮事項 83
 レコード、RECON データ・セットの 708

DBCTL 環境 (DBCTL environment) 85

DEFINE CLUSTER
 キーワード
 推奨値 700

DEFINE CLUSTER キーワード
 CONTROLINTERVALSIZE 700
 CYLINDERS 700
 FREESPACE 700
 INDEXED 700
 KEYS 700
 LOG 700
 NAME 700
 NOWRITECHECK 700
 RECORDSIZE 700
 SHAREOPTIONS 700
 SPEED 700

DEFINE GENERATIONDATAGROUP
 コマンド 41

DELAY パラメーター 165

DEQ マクロ 695

DFP レコード管理サービス 695

DFSAPMD0
 アンインストール 157
 異常終了フォーマット設定モジュール 156
 バインディング 157

DFSAPPL コマンド 141

DFSBB00 (バッチ・バックアウト・ユーティリティ)
 緊急時再始動、関係 653
 再始動、関係 653
 制限 649
 の紹介 136

DFSCCBLK 368

DFSCGxxx
 グローバル・オンライン変更 (global online change) 29
 OLCSTAT データ・セット 29

DFSCMC10 モジュール
 IMS のインストール 152
 MSC CTC リンクのインストール 152

DFSDCxxx PROCLIB メンバー
 サインオン検査 390

DFSERA10 (ファイル選択・フォーマット設定印刷ユーティリティ) 445

DFSHSBxx の SWITCH パラメーター 768

DFSILTA0 (ログ・トランザクション分析ユーティリティ) 445
 モニターに使用する 476

DFSINSX0 (宛先作成出口ルーチン) 共用キュー
 動的制御ブロック 264
 プログラム間通信 263
 メッセージ通信 263

DFSINTRS、意図リストのページ固定に使用する 526

DFSINTX0 (初期設定出口ルーチン) ETO を使用して 415

DFSIRP0 (プログラム分離トレース報告書ユーティリティ) 483

DFSISTS0 (統計分析ユーティリティ) 445

DFSKBLA3 (Knowledge-Based Basic Formatting Print ルーチン) 445

DFSKBLAK (Knowledge-Based Formatting Print ルーチン) 445

DFSKBLAS (Knowledge-Based Summary Formatting Print ルーチン) 445

DFSLGNX0 (ログオン出口ルーチン) ETO を使用して 415

DFSMPLEXx、プログラム・プリロードの 533

DFSMS
 XRF に対する要件 747, 756
 XRF プロセスに与えられる利点 763

DFSOFMD0
 オフライン・ダンプ・フォーマット設定モジュール 158

DFSPREC0 (索引/ILDS 再作成ユーティリティ)
 DBRC に対する関係 636

DFSSGNX0 (サインオン出口ルーチン) ETO を使用して 415

DFSUARC0 (ログ保存ユーティリティ) 111, 113
 説明 627
 OLDS の保存 116

DFSUDMP0 (データベース変更累積ユーティリティ)
 最大数、世代の 666
 リカバリー期間 666

DFSUDMP0 (データベース・イメージ・コピー・ユーティリティ)
 最大数、世代の 665
 作成、将来の利用のためのデータ・セットの 664
 説明 659
 標準外イメージ・コピー・データ・セット 670
 リカバリー期間 665

DFSUDMT0 (データベース・イメージ・コピー 2 ユティリティ)
 概要 661

DFSUIC00 322

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) 作成、将来の利用のためのデータ・セットの 664
実行、DBRC により記録される 658
説明 659

DFSUOLC0 601

DFSUOLC0 (グローバル・オンライン変更ユーティリティー) 584, 586

DFSURUL0
の紹介 131
HISAM 再編成アンロード・ユーティリティー 131

DFSURUL0 (HISAM 再編成アンロード・ユーティリティー)
バックアップ用 669

DISPLAY (/DISPLAY) コマンド
会計での使用 974
モニター用 476
MODIFY 573, 578

DLIBATCH
IMSplex コンポーネント (IMSplex component) 44

DLISAS プロシージャー、セキュリティ 435

DL/I データベース
考慮事項 538
設計の考慮事項 538

DL/I 呼び出し要約報告書
説明 857
報告書の使い方 857
報告書のフィールド 857

DMB テーブル・レコード 723

DMB バッファー・プール
要件の分析 528

DMB 番号 723

DRA (データベース・リソース・アダプター)
概要 12
リソース
CCTL トランザクション要求 493
PSB スケジュール要求によって超過した 523

DRD (動的リソース定義)
IMSplex のシナリオ 32

DRDA (分散リレーショナル・データベース・アクセス)
Open Database Manager (ODBM)
概要 47
管理 171

DSE (従属サービス・エレメント)
定義 745

DSPDBHRC 708
区画 DB レコード
HALDB 708

DSPDCAX0 (DBRC コマンド許可出口ルーチン)
使用法 685

DSPDSHRC
区画 DBDS レコード
HALDB 708

DSPPTNRC
HALDB 区画レコード
HALDB 708

E

EMH (急送メッセージ・ハンドラー)
共用キュー環境、内 286

ENF 223

ERESTART (/ERESTART) BACKUP コマンド 771, 786

ESAF (外部サブシステム接続機能) 14
DB2 for z/OS へのアクセス 615
JMP または JBP アプリケーションの使用 615

ESTAE プロセス
類似性管理のバイパス 364

ETO (拡張端末オプション)
セキュリティ 415
説明 79

DFSLGNX0 (ログオン出口ルーチン) 415

DFSSGNX0 (サインオン出口ルーチン) 415

EXEC ステートメントのパラメーター SSM 14

External Subsystem Attach Facility (ESAF)
DB2 for z/OS へのアクセス 615

EXVR オンライン・パラメーター
チューニングでの使用 520

E-MCS 52

F

FACILITY クラス 213

FDBR
IMSplex コンポーネント 25

FDBR (高速データベース・リカバリー)
オンライン変更機能 576, 583
定義 11
DBCTL 環境における 13
DB/DC 環境 11
FDRMBR パラメーター 11

FDRMBR パラメーター 11

FREESPACE キーワード
DEFINE CLUSTER キーワード 700

G

GBL_SERIAL_PGM 206

GDG (世代別データ・グループ)
定義 41
モデル・データ・セット 42

GENJCL コマンド
JCL の生成 635

GENJCL.ARCHIVE コマンド
RECON データ・セット内のログ制御要件 627

GENJCL.IC
HALDB オンライン再編成の考慮事項 678

GENJCL.OIC
HALDB オンライン再編成の考慮事項 678

GENMAX パラメーター
指定、イメージ・コピー要件の 665
将来の利用のためのイメージ・コピー・データ・セット 665
リセット 729

GPAR (汎用パフォーマンス分析報告プログラム)
詳細モニターでの使用 476

GPSB (生成されたプログラム仕様ブロック) 19

GRESTAE コマンド 364

GRPMAX パラメーター
変更累積要件の指定 666
リセット 730

GRS マクロ 695

GRSNAME=
汎用リソース名の指定 357

GSAM (汎用順次アクセス方式) 291

GTF (汎用トレース機能)
詳細モニターでの使用 476, 497

GTF (汎用トレース機能) トレース 482

H

HALDB (高可用性ラージ・データベース)
オンライン再編成
DBRC の考慮事項 678
区画 DB レコード (DSPDBHRC) 708
区画 DBDS レコード (DSPDSHRC) 708
区画 (partition) 708
区画レコード (DSPPTNRC)
PHDAM 708
動的割り振りデータ・セット 36
マスター (DSPDBHRC)
DB ヘッダー・レコード 708
命名規則 95
DBDS のタイプ 708
DBRC コマンド、サポートされる 678

HALDB (高可用性ラージ・データベース)
(続き)
REORG レコード 712
HISAM 再編成アンロード・ユーティリテ
ィー (DFSURUL0)
の紹介 131
バックアップ用 669
HLQ パラメーター 141
HOST マクロ 165
HSBID パラメーター
XRF で使用する 814
HSBMBR パラメーター
XRF で使用する 814
HSSP データ・セット 669
データベース、DBRC に登録される
669
HWSSMPL0
ネットワーク・セキュリティ資格情
報 397
分散セキュリティ資格情報 397
HWSSMPL1
ネットワーク・セキュリティ資格情
報 397
分散セキュリティ資格情報 397

I

ICDSN パラメーター・コマンド
再使用 668
最大数、世代の 665
作成、将来の利用のための 664
重複、命名規則 657
命名規則 656
リカバリー期間 665
レコード 712
ICMD 呼び出し
セキュリティの概要 401
IDCAMS
コマンド
DEFINE 247
DELETE 247
REPRO 247
IFP (IMS 高速機能) 84
アプリケーション・プログラム 8
非メッセージ・ドリブン・プログラム
8
メッセージ・ドリブン・プログラム 8
ユーティリティー 8
領域
特性 8
ILDS (間接リスト・データ・セット)
DBDS グループとしての 676
DBRC に対する関係 636
IMPORT DEFN SOURCE(CATALOG)
オンライン変更の概要 553

IMPORT DEFN SOURCE(CATALOG)
(続き)
活動化、オンライン・システムでの
ACB の 555, 557, 562, 563
IMS
送達不能キュー 285
IMS 2 次端末
XRF に対する定義 817
IMS Application Menu 47
開始 141
概要 141
IMS Connect
z/OS PPT の項目 151
IMS Connect レコーダー・データ・セッ
ト (非 BPE トレース) の暗号化 379
IMS DataPropagator 90
IMS SOAP Gateway
IMS TM での使用 615
IMS TM
コールアウト機能 615
IMS SOAP Gateway での使用 615
IMS TM Resource Adapter
IMS TM での使用 615
IMS TM リソース・アダプター
ネットワーク・セキュリティ資格情
報 399
分散セキュリティ資格情報 399
IMS 外部トレース・データ・セットの暗
号化 379, 385
IMS から DBRC への呼び出し 621
IMS コマンド
/CHANGE SURVEILLANCE
XRF 758
/DISPLAY
モニター用 476
/ERESTART BACKUP 771, 786
/START IMS 771
/START SURVEILLANCE
XRF 758
/STOP SURVEILLANCE
XRF 758
/SWITCH SYSTEM
計画テークオーバー (planned
takeover) 785
テークオーバー処理 776, 783
予定テークオーバー 776
/SWITCH SYSTEM FORCE 776
/UNLOCK SYSTEM
テークオーバー中の処理 780
データベース保全性の保証 762
IMS コンポーネント
BPE 37
IMS システム
概要 3
管理
概要 3

IMS システム、文書化 93
IMS システム管理
概要 3
IMS システム定義
クラス 1 端末の定義 817
クラス 2 端末の定義 817
端末用の XRF 関連キーワードを使用
したマクロ 817
XRF に対するシステム定義マクロのコ
ーディング 815
XRF に対するマスター端末および 2
次端末の定義 817
IMS システムの文書化 93
IMS システム・アクセス
宣言
オンライン・システム用、オンライ
ン変更 297
オンライン・システム用、概説 297
データを共用するオンライン・デー
タベース 296
バッチ・システム用 297
IMS システム・ログ
監視メカニズムとしての 774
監視メカニズムとしての確立 768
記録
従属領域のアクティビティーの状況
774
メッセージ・キュー 774
MFS プール・ロード 774
XRF のセッション情報 758
障害、XRF の制限 750
使用されるトラッキング 774
テークオーバー中 778
トラッキング、XRF 758
USERVAR バックアップ・セッション
での 750
XRF テークオーバー指示のオーバーラ
イド 758
XRF テークオーバー中 758
XRF に対する配置 828
IMS 初期化
IMSRSC リポジトリ 234
IMS スプール API 605
アプリケーション要件 605
JES 印刷データ・セット 605
印刷データ・セットの特性 608
固有の端末サポート 605
出力データ・セット 608
設計の考慮事項 605
操作上の考慮事項 605
データ・マネージャーとしての
JES 608
動的出力 608
AFP 605
CHNG 呼び出し
説明 609

- IMS スプール API (続き)
 - CHNG 呼び出し (続き)
 - OUTN オプションによる 609
 - PRTO オプションによる 609
 - TXTU オプションによる 609
 - OEM プリント・サーバー 605
 - OUTN オプション 609
 - PRTO オプション 609
 - SETO 呼び出し 610
 - TXTU オプション 609
- IMS データ・セットの配置
 - XRF に対する要件 756
 - XRF プロセスに与えられる利点 758
- IMS でのデータ・セット暗号化のサポート 375
- IMS トランザクション・マネージャー
 - コールアウト機能 615
- IMS の開始 451
- IMS の再始動 451
- IMS のシャットダウン 453
- IMS の操作
 - 作業、リスト 445
 - システム・ログの処理 445
 - データ共用の制御 448
 - トレース
 - プログラム分離およびロック 483
 - IMS トレース機能 485
 - z/OS コンポーネント・トレース・サービス 483
 - z/OS 汎用トレース機能 482
 - モニター・ツールの選択 480
 - リカバリー
 - 作業 446
 - サブシステム接続 449
 - サブシステムの切断 449
 - ログ・データ・セットの調整 449
 - ログ・データ・セットの変更 449
 - IMS のモニター 445
 - IMS モニター 480
- IMS プロシーチャー
 - セキュリティー 435
- IMS 報告書
 - 使用 839
- IMS マスター端末
 - XRF に対する定義 817
- IMS モニター 458, 480
 - 時間測定されるイベントの要約
 - DBCTL 904
 - 詳細データの 476
 - 説明 492
 - テスト 458
 - パフォーマンス分析 515
 - 複数システムで使用 465
 - 分布の定義の値 897
 - 報告書
 - 呼び出し要約 458
- IMS モニター (続き)
 - DBCTL
 - 時間測定されるイベントの要約 904
 - DCCTL の報告書
 - 概要 926
 - 計測単位 926
 - 出力順序 926
- IMS モニター報告書
 - 一般的な IWAIT 時間イベント 885
 - 意図障害 876
 - 印刷プログラムと MSC 899
 - 回線機能 889
 - 概要 865, 869
 - 計測単位 869
 - システム構成 870
 - システム構成報告書
 - 追加 906
 - 実行プロファイル 870
 - 出力
 - DB/DC 869
 - 出力選択オプション 872, 906
 - 通信待機 890
 - 通信要約 889
 - データベース・バッファ・プール 887
 - デッドロック・イベント要約 890
 - バッファ・プール統計 870
 - プール・スペース障害要約 890
 - プログラム入出力 880
 - プログラム要約 877
 - 分布付録報告書 896, 920
 - 報告書出力の順序 869
 - 報告書オカレンスの検査 872, 906
 - メッセージ形式バッファ・プール 884
 - メッセージ・キュー・プール 885
 - 呼び出し要約 877
 - ラッチ競合統計 890
 - 領域およびジョブ名 870
 - 領域待機 873, 909
 - 領域別プログラム 873
 - 領域要約 873
- DBCTL
 - 意図障害要約報告書 909
 - 概要 905
 - 計測単位 905
 - システム構成 906
 - 実行プロファイル 906
 - 出力選択オプション 906
 - デッドロック・イベント要約 918
 - プール・スペース障害要約 918
 - プログラム入出力 915
 - プログラム要約 913
 - 報告書出力の順序 905
 - 報告書オカレンスの検査 906
 - 呼び出し要約 913
- IMS モニター報告書 (続き)
 - DBCTL (続き)
 - ラッチ競合統計 918
 - 領域別プログラム 909
 - 領域要約 909
 - VSAM バッファ・プール 917
 - DB/DC
 - トランザクション・キューイング 886
 - DCCTL
 - 一般的な IWAIT 時間イベント 940
 - 回線機能 942
 - 概要 925
 - システム構成 927
 - 実行プロファイル 927
 - 出力選択オプション 927
 - 通信待機 943
 - 通信要約 942
 - 適用外の報告書 925
 - トランザクション・キューイング 941
 - 内部リソースの使用、モニター 944
 - プール・スペース障害要約 944
 - プログラム入出力 936
 - プログラム要約 934
 - 分布付録報告書 945
 - 報告書オカレンスの検査 927
 - メッセージ形式バッファ・プール 939
 - メッセージ・キュー・プール 940
 - 呼び出し要約 934
 - ラッチ競合統計 944
 - 領域およびジョブ名 927
 - 領域待機 930
 - 領域別プログラム 930
 - 領域要約 930
 - MSC キューイング要約 902
 - MSC トラフィック 900
 - MSC 要約 901
 - VSAM バッファ・プール 887
- IMS モニター報告書印刷ユーティリティー
 - 制御 895
- IMS モニター・プログラムで時間を測定されるイベント
 - 意図のためのアイドル 865
 - 最初の DL/I 呼び出しのスケジュール 865
 - 実行の経過 865
 - スケジューリングと終了 865
 - 説明 865
 - チェックポイント操作 865
 - トレース間隔
 - DBCTL 906
 - DB/DC 870

IMS モニター・プログラムで時間を測定
されるイベント (続き)
トレース間隔 (続き)
DCCTL 927
入力待ち (WFI)
DB/DC 865, 880
DCCTL 936
待ち時間 865
待ちなし時間
DCCTL 925
メッセージ入力時 865
要約
DB/DC 868
DL/I 呼び出しの NOT-WAIT 時間
DB/DC 868
DCCTL 925

IMS 領域タイプ
高速機能 (Fast Path) 8
制御領域 (control region) 8
バッチ・メッセージ処理 8
メッセージ処理 8

IMSASAP II 報告書
チューニング 515
テスト 459

IMSCTF マクロ
DFSCMC10 モジュール 152

IMSGEN マクロ・ステートメント、セキュリティー・オプション 439

IMSplex
一般的な構成 61
オンライン変更の終了 593
簡易 28
関与するアドレス・スペース 44
共用キュー 59
リソース構造の使用 50
グローバル・オンライン変更 (global online change) 29, 577
機能 55
使用可能化 166
使用不可化 166
利点 55
ACB ライブラリー・メンバー 57, 591
構成 59
構成に関する推奨 61
コマンド 44, 182
形式 182
利点 182
コマンド環境、タイプ 2 78
コンポーネント 20
オプション 59
コンポーネント、必須 59
自動 RECON 損失通知 58
修正、リソースの 551
使用される IMS システム・サービス 23

IMSplex (続き)
使用される z/OS システム・サービス 23
図 44
整合性の保証 363
セキュリティー 33
タイプ 2 コマンド 189
例 209
タイプ 2 コマンド環境 78
単一システム構成 61
逐次アプリケーション処理 265
調整 25
データ共用 291
定義 20, 25, 44
バッチ環境 59
パラメーター
グローバル状況 206
汎用リソース・グループ (generic resource group)
整合性 363
表示情報 593
並行データベース・アクセス 291
並列シスプレックスとの比較 20
マスター IMS 制御領域 55
メンバー
定義 20
OM 44
RM 44
SCI 44
問題の診断 169
ユーザーの可用性の維持 743
リカバリー 835
リソース構造 (resource structure) 60
推奨 50, 60
メッセージ宛先リソース 205
ARM 835
CQS 59
CSL 43
構成 59, 61
リカバリー 835
CSL (共通サービス層)
モニター 169
CSLZQRY 169
DSECT 機能 169
STATS 機能 169
DBRC サポート 643
DLISAS 59
FDBR 25
IMS 制御領域
定義 26
IMSRSC リポジトリを使用する構成 59, 61
MSC との共存 24
ODBM
構成 27
初期設定 27

IMSplex (続き)
OM
セキュリティー・ユーザー出口ルーチン 199
定義 27
OM API 182
RACF 189
PLEXPARM= パラメーター 202
RM
推奨 60
定義 27
RM (リソース・マネージャー) を使用しない 577, 586
SCI
初期設定 27
SPOC 182
コマンド 182
SPOC (単一制御点) 52
TM リソース
管理 367
XRF 25, 744
XRF (拡張回復機能)
RM のないオンライン変更の制約事項 756

IMSplex コンポーネント (IMSplex component) 20
IMSplex メンバー (IMSplex member) 20
IMSRSC リポジトリ
管理 229
更新
指定 231
構成、IMSplex での 59
構成要件 61
除去、RS カタログ・リポジトリからの 232
表示
記述子定義 233
リソース定義 233
IMS 初期化 234
Resource Manager (RM) 235
終了 238
初期設定 235
IMS.ACBLIB ライブラリー
データ・セットの動的な割り振り 505
IMS.PGMLIB
検索の最適化 520
メンバーとしての PSB 名 81
IMS.PROCLIB
外部サブシステム 16
DFSFIXxx メンバー 774
DFSHSBxx メンバー
パラメーター 776
OLDSDEF ステートメント (PROCLIB) 110
XRF パラメーター 819

IMS.PROCLIB の DFSFIXxx メンバー
チューニングでの使用 531
ページ固定の 531
DFSINTRS、意図リストのページ固定
に使用する 531
IMS.PROCLIB の DFSHSBxx メンバー
テークオーバー基準の確立 776
XRF 監視方式の選択 758
INDEXED キーワード
 DEFINE CLUSTER キーワード 700
INIT SELF コマンド 163
INIT 呼び出し
 データ・センシティブティの指定 82
INITIATE OLC 55
INITIATE OLC コマンド 578, 588
 エラー処理 590
 戻りコードおよび理由コード 590
INIT.ADS コマンド 647
INIT.DB コマンド
 データベースの定義 647
INIT.DBDS コマンド
 指定、イメージ・コピー要件の 665
 データベースの定義 647
 変更累積要件の指定 666
 REUSE キーワード 668
INIT.RECON コマンド
 RECON データ・セット (RECON
 data set) 645
 RECON データ・セットの初期設定
 692
 RECON のリカバリー 721
IRLM
 活動化
 実行 JCL 297
 IRLMNM パラメーター 297
 システムの初期設定 301
 障害 353
 再始動 334
 データベース・レベル共有
 (database-level sharing) 297
 z/OS サブシステム 301
IRLM 構造
 サイズの計算 345
IRLM (内部リソース・ロック・マネージ
ヤー)
 インストール 299
 開始 310
 活動化
 バッチ・システム用 307
 共有モード 300
 実行パラメーター 299, 300
 障害 776
 テークオーバーの原因としての 776
 データ共有 642
 データ共有、役割 298
 データベース保全性 642

IRLM (内部リソース・ロック・マネージ
ヤー) (続き)
 停止 311
 トレース 483
 バッチ・バックアウトによる 649
 モニター 313, 448
 リカバリー 333
IMSplex の一部 23
IRLM IVP サブセット
 サブシステム名 161
 ダンプ・フォーマット設定モジュー
 ル 161
 APF 許可 161
 PPT 項目要件 161
 VTAM インターフェースの命名の
 提案 161
XRF テークオーバー中の処理 783
XRF テークオーバー中の図 783
XRF テークオーバーの原因としての障
害 750
XRF でのローカル・ロック・マネージ
ヤーとしての使用 783
z/OS PPT entry 151
z/OS PPT の項目 151
ISC (システム間連絡)
 アクティブ IMS システムと代替 IMS
 システム間のリンク
 監視メカニズムとしての 774
 監視メカニズムとしての確立 768
 初期設定 771
 図 799
 テークオーバー条件 (takeover
 condition) 758
 定義 758
 同期化で使用する 773
 XRF に推奨される 799
 高速機能および 84

IVP
 サービス 254

J

JCL (ジョブ制御言語)
 骨組み実行メンバー
 定義 620
 ユーティリティのための調整 627
 DBRC による生成 620
JCL 割り振り
 RECON データ・セット 699
JES 構成 148
JES (ジョブ入力サブシステム)
 計画に関する考慮事項 812
 障害、XRF の制限 750
JMP (Java メッセージ処理)
 アプリケーションのセキュリティ
 409

JMP (Java メッセージ処理) (続き)
 アプリケーション・セキュリティ
 409
JOBJCL
 基幹 JCL 実行メンバー 635

K

KEYS キーワード
 DEFINE CLUSTER キーワード 700
Knowledge-Based Basic Formatting Print
ルーチン (DFSKBLA3) 445
Knowledge-Based Formatting Print ルー
チン (DFSKBLAK) 445
Knowledge-Based Log Analysis (KBLA)
ユーティリティ 445
Knowledge-Based Summary Formatting
Print ルーチン (DFSKBLAS) 445

L

LOG キーワード
 DEFINE CLUSTER キーワード 700
LOG パラメーター
 監視のための 819
 メンバー DFSHSBxx の 768
LOGCHAR マクロ 763
LOGMODE パラメーター 163
LOGON APPLID コマンド 763
LSR (ローカル共用リソース) オプション
700
LTERM (論理端末)
 共用キュー
 IMS がインタレストを登録する場
 合 262
 シブプレックス内の TM リソースと
 して 369
 名前の一意性 367, 369
 リソース・タイプ整合性 368
 RM に対する定義 369
LU 6.2
 コマンド許可 393

M

MAXAPPL キーワード 163
MFS 形式
 変更 575, 580
MFS サービス・ユーティリティ
 索引の MFSRVC プロシージャ 543
 チューニングでの使用 526
MFS (メッセージ形式サービス)
 形式のオンライン・テスト
 オンライン実行要件 462
 システム定義要件 462

MFS (メッセージ形式サービス) (続き)
形式のオンライン・テスト (続き)
MFSTEST モード 461
テスト形式 462
ログに記録されているプール・ロード
774
論理ページングと XRF テークオーバ
ー 805
IMSGEN マクロの
MFSTEST=YES 462
MFSTEST モード 461
MFSTEST のオンライン実行要件 462
MFSTEST モード 461
MNPS パラメーター
XRF で使用する 814
MNPS 名
汎用リソース環境内 357
MNPSPW パラメーター
XRF で使用する 814
MODBLKS
システム定義 (system definition)
オンライン変更機能に必要な 570
MODE パラメーター 163
MODETBL パラメーター 163
MODIFY コマンド
VTAM 768
MODIFY (/MODIFY) コマンド
オンライン変更で使用 573
PREPARE 585
MODSTAT
DD ステートメント 55
MODSTAT データ・セット
XRF での配置 828
MODSTAT2 55
MPP (メッセージ処理プログラム)
領域
開始 8
特性 8
MSC
セキュリティ
トランザクション 412
RACF および出口ルーチンが呼び
出された場合 413
MSC キューイング要約報告書
リンク・キューイング時間の評価 902
例 902
MSC のセキュリティ
サインオン/オフ・セキュリティ出
口ルーチン (DFSCSGN0) 414
セキュリティ再検証出口ルーチン
(DFSCTSE0) 414
トランザクション許可出口ルーチン
(DFSCTRNO) 414
ユーザー ID の指定 413

MSC のセキュリティ (続き)
TM および MSC メッセージ経路指定
および制御出口ルーチン
(DFSMSCE0) 414
MSC (複数システム結合機能) 84
共用キュー
IMS がインタレストを登録する場
合 262
共用キュー環境内 287
リモート処理
定義 259
MSC (マルチシステム結合機能)
解釈
IMS モニター MSC 報告書 899
高速機能および 84
システム間キューイングの判別 900
パフォーマンス情報 486
分布付録出力の解釈 949
モニター 465
リンク
XRF に対する定義 818
IMS モニター報告書印刷プログラム
899
MSC キューイング報告書
解釈 899
MSC キューイング要約報告書 902
MSC トラフィック報告書
システム間キューイング、判別 900
内容 900
例 900
MSC 要約報告書
解釈 899
キュー・サイズの評価 901
内容 901
例 901
MSC 要約報告書
IMS モニター (DB/DC) 899
MSCSEC=
MSC トランザクション・セキュリテ
ィー 412
MSDB (主記憶データベース)
グローバル・リソースとしての状況
202
高速機能に関する考慮事項 83
代替 IMS システムによるロード 773
データ共用では未サポート 291
定義 773
XRF での配置 828
MSGQ 基本リスト構造 270
MSNAME
シスプレックス内の TM リソースと
して 370
RM 定義 370
MSPLINK マクロ
BACKUP キーワード 818

MTO (マスター端末オペレーター)
責任 10
説明 451
バックアップ 10
メッセージ 265
2 次 10

N

NAME キーワード
DEFINE CLUSTER キーワード 700
NCP 遅延 165
NCP (ネットワーク制御プログラム)
障害、XRF の制限 750
ストレージ上の考慮事項 811
通信ネットワークのチューニング 535
テークオーバー時のセッションの切り
替え 750, 781
XRF に対する要件 747, 756
XRF プロセスに与えられる利点 768
NCP の考慮事項 165
NOREUSE パラメーター
将来の利用のためのイメージ・コピ
ー・データ・セット 664
CA データ・セット 675
NOTIFY.IC コマンド
HALDB の考慮事項 678
NOTIFY.UIC コマンド
HALDB の考慮事項 678
NOWRITECHECK キーワード
DEFINE CLUSTER キーワード 700

O

OBTAIN マクロ 695
ODBA
ODBM 向けの ODBA アプリケーシ
ョン・サーバーの構成 173
ODBA アプリケーション・プログラム
セキュリティに関する考慮事項 409,
442
ODBM
アカウントिंग 175, 176
メッセージ経路指定 174
ODBM (Open Database Manager)
概要 47
構成 27
構成要件 61
サポートされるインターフェース 47
初期設定 27
セキュリティ
概要 175
ODBA
ODBM 向けの ODBA アプリケー
ション・サーバーの構成 173

ODBM (Open Database Manager) (続き)
RRS 172
OLCSTAT
コールド・スタート (cold start) 601
OLCSTAT データ・セット 29, 205
形式 30
推奨属性 30
属性 30
ヘッダー・サンプル 30
XRF (拡張回復機能)
RM のない IMSplex の制約事項
756
OLC= パラメーター 29
OLDS
システム・ログ入出力の最小化 533
バッファ 533
ブロック・サイズ 533
OLDS (オンライン・データ・セット)
テスト環境でのクローズ 737
OLDS (オンライン・ログ・データ・セッ
ト)
アーカイブ 110
アクセス方式 110
数 126
環境 110
クローズ、WADS からの 111
再使用 116
サイズ 126
装置 128
重複または単一 125
低下モード・ロギング 125
定義 108, 110, 125
停止 110
特性の変更 129
バッファ 129
ブロック・サイズ 127
保存 116
レコード 705
割り振り 110
DBRC のアーカイブ 627
RECON 110
XRF に対する要件 747
OLDSDEF ステートメント 110
OM 44
監査ログ
レコード・フォーマット 185
コマンド・セキュリティ 188
ログ・レコード、印刷 186
OM (Operations Manager)
監査証跡 183, 184
構成要件 61
定義 27
OM API 182
XRF 744
Open Database Manager (ODBM)
概要 47

Open Database Manager (ODBM) (続き)
管理 171
クライアントの登録 172
構成 27
構成要件 61
サポートされるインターフェース 47
初期設定 27
セキュリティ
概要 175
ODBA 173
ODBM 向けの ODBA アプリケー
ション・サーバーの構成 173
RRS 172
Operations Manager (OM)
概要 50
管理作業 181
構成要件 61
コマンドの経路指定 181
コマンド・セキュリティ 188
提供される機能 50
OPNDST (/OPNDST) コマンド 163
OSAM (オーバーフロー順次アクセス方
式)
キャッシュ構造
サイズの変更 348
構造
サイズの計算 347
バッファ・プール 530
OSAM 順次バッファリング 530
OSAM バッファ・プール報告書 953
OTMA (Open Transaction Manager
Access)
共用キュー環境 284
同期メッセージ 284
非同期メッセージ 284

P

PI キーワード 483
PI (プログラム分離)
DB/DC 環境での 88
PLEXPARM パラメーター
GSTSAREA 206
GSTSDB 206
GSTSTRAN 206
PL/I サブルーチンのプリロード 533
PPT 項目要件 161
PREPARE コマンド 575, 580
PRILOG
圧縮、自動 717
圧縮、手動 717
圧縮、問題の診断 717
レコード・サイズ 717
PRILOG レコード
オープンしているオンラインのクロー
ズ 732

PSB 作業バッファ・プール
ストレージ要件 528
PSB バッファ・プール
要件の分析 528
PSB (プログラム仕様ブロック)
アプリケーション制御ブロック
(ACB)、IMS 管理対象、活動化 554
活動化
データ共用および IMS 管理の
ACB 環境 559
IMS 管理の ACB 環境、データ共
用による 559
活動化、IMS 管理の ACB 環境での
554
プログラム名の規則 81
要求
CCTL 領域での 523
リソース・セキュリティ 438
ACB (アプリケーション制御ブロ
ック)、IMS 管理対象、活動化 554
PSBs
オンライン変更
IMPORT コマンドの UPDATEPSB
オプション 561
IMS 管理の ACB 環境 555, 557,
562, 563
オンライン変更 (online change)
管理対象 ACB 環境 553
活動化、オンライン・システムでの
IMPORT コマンドの UPDATEPSB
オプション 561
管理対象 ACB 環境
システムのサブセットでの活動化
561
PST 会計報告書 951
PTF
サービス・レベルの保守 243
実動システム
推奨 242
保守プロセスの例 244
推奨サービス・アップグレード
(RSU) 241
PTF (プログラム一時修正)
SMP/E を使用 239
Q
QASTSPE 変数 160
QCF (キュー制御機能)
コールド・キュー
再キューイング 278
構造
再キューイング 278
モニター 278
IMS のテスト 464
QUERY MEMBER コマンド 593

QUERY OLC コマンド 593

QUERY コマンド 578

R

RACF 209

オンライン変更 (online change) 433

コマンド・セキュリティ 188

リソース・クラス

リソース・アクセス・セキュリティ
ー 417

AO アプリケーション・セキュリティ
ー 401

AO コマンド・セキュリティ 406

MSC

IMS が RACF を呼び出す場合
413

RACF セキュリティ

IMSplex 検査 190

RACF (リソース・アクセス管理機能)

許可 209

計画に関する考慮事項 812

サインオン・パスワード再検証 394

システム・ライブラリーおよびデー

タ・セットの保護 435

データベースの保護 436

データ・セットの配置 812

パスチケット 421

パスチケットの作成 392

VTAM VGR 環境 391

パスワード保護 394

リソース・クラス記述子テーブル 422

APPL クラス

例 189

IMS 用のリソース・クラス 422, 439

IMS を使用したインプリメント 422

IMSplex

許可 189

セキュリティ 209

セキュリティの例 209

OPERCMDS

例 189

OM 189

RDS (再始動データ・セット)

監視メカニズムとしての 774

監視メカニズムとしての確立 768

定義 108, 758

内容 114

パラメーター

監視のための 819

メンバー DFSHSBxx の 768

割り振り 130

XRF での配置 828

RECON

自動損失通知 58

データ共用環境における 293

RECON (続き)

並列アクセス 168

IMSplex の一部 23

RECON 初期設定トークン (RIT) 639

RECON データ・セット

アクティブ 702

エンキュー問題の原因 738

置き換え、損傷した 721

置き換え、廃棄された RECON 723

オンラインでの再編成 720

概要 620, 691

拡張 699

可用性 692

競合問題 692

計画に関する考慮事項 692

検索、ログ関連情報 628

コマンド

BACKUP.RECON 716

再編成 719

再編成、事前のバックアップ 720

再編成手順 720

再編成前のバックアップ 720

作成時、推奨 700

サブシステム (SSYS) レコード 714

出力

タイム・スタンプ 727

使用 115

使用不可 721

初期アクセス 702

初期設定 645, 692

処理

LSR (ローカル共用リソース) オプ
ション 700

シリアライゼーション 695

スペア 702

セキュリティの考慮事項 702

説明 691

損失通知 725

タイム・スタンプ

出力 727

データ共用環境でのモニター 315

データ共用レコード 641

データ共用レコード・タイプ 714

データベースの登録 647

データベース・イメージ・コピー 2 デ
ータ・セット 661

定義 692

デッドロック状態の回避 695

動的割り振り (dynamic

allocation) 699

トラッキング、へ行われた変更の 726

入出力エラー処理 721

バックアップ 716

並行イメージ・コピー・デー
タ・セット 664

並列アクセス 693

RECON データ・セット (続き)

並列アクセス、計画 693

ヘッダー・レコード 704

保守 716

保全性の維持 683

要件の定義 631

リカバリー 721

リカバリー・レコード・タイプ 706

ALLOC 713

BACKOUT 706

CA 707

CAGRP 707

DBDSGRP 707

IMAGE 712

LOGALL 713

RECOV 714

REORG 712

SSYS 714

両方のセットが使用不能 721

レコード 704

イメージ・コピー 712

データベース 708

データベース・データ・セット 708

レコードの保守 716

レコードへのマップを行う

DSECT 715

レコード・タイプ 704

レコード・マッピング 715

ログ情報 625

ログ制御レコードの変更 627

割り振り 691

ALLOC レコード 713

BACKOUT レコード 706

CA レコード 707

CAGRP レコード 707

DBDSGRP レコード 707

DBRC セキュリティのオーバーライ
ド 690

HSSP イメージ・コピー・デー
タ・セット 669

IMAGE レコード 712

LOGALL レコード 713

OLDS 情報 110

RECOV レコード 714

REORG レコード 712

SLDS 情報 113

SSYS レコード 714

VSAM CREATE モード 699

RECON 入出力出口ルーチン

RECON に対する変更のトラッキング
726

RECON の再編成

手順 720

CHANGE.RECON の使用 719

RECON への並列アクセス

の設計 693

RECORDSIZE キーワード
 DEFINE CLUSTER キーワード 700
 RECOVER コマンド 637
 RECOVPD パラメーター
 使用 665
 使用法 666
 useage 665
 REORG レコード
 フィールド、オンライン再編成で使用
 される 678
 RECON データ・セット 712
 REPAIR.RECON コマンド
 RECON の修復 723
 Repository Server 44
 終了 238
 セキュリティー 426
 例 430
 Resource Manager (RM) 428
 リカバリー 835
 REPRO コマンド
 バックアップに使用する 716
 復元、RECON データ・セットの 721
 RESERVE コマンド
 バックアップ時の 716
 RESERVE マクロ 695
 Resource Manager
 オンライン変更 (online change)
 RM 不使用時の制約事項 55
 Resource Manager (RM)
 概要 50
 管理作業 201
 グローバル状況
 処理エラー 202
 グローバル・コマンド状況の保守 202
 構成、CSL がない 61
 構成要件 61
 終了
 IMSRSC リポジトリを使用 238
 初期設定
 IMSRSC リポジトリ 235
 提供される機能 50
 保管されたリソース情報
 データベース用 204
 トランザクション用 205
 DEDB エリア用 204
 保守、グローバル・リソース情報の
 201
 リソース構造 (resource structure) 50
 グローバル・リソースの例 201
 再移植 207
 リソース情報の保管 201
 IMSRSC リポジトリ 235
 PLEXPARM= パラメーター 202
 Repository Server の終了 238

REUSE パラメーター
 将来の利用のためのイメージ・コピー
 ・データ・セット 664
 REUSE パラメーター・コマンド
 INIT.DBDS 668
 REXX SPOC API 54
 RIT (RECON 初期設定トークン) 639
 RLDS (リカバリー・ログ・データ・セッ
 ト)
 圧縮 120
 コピー 118
 作成 118
 定義 108, 130
 累積、DFSUCUM0 を使用した変更の
 671
 レコード 705
 ログ 114
 RM 44
 RM (Resource Manager)
 概要 50
 構成、CSL がない 61
 構成要件 61
 端末の管理 58
 定義 27
 提供される機能 50
 リソース構造 (resource structure) 50
 XRF 744
 RMF II (リソース測定機能 II)
 チューニングでの使用 515
 入出力分析の 548
 ページング率の使用 476
 モニター・ツールとして 476
 RMGENJCL (/RMGENJCL) コマンド
 JCL の生成 635
 RMxxxxxx (/RMxxxxxx) コマンド
 概要 622
 ROLB 呼び出し
 動的バックアウト (dynamic
 backup) 649
 RRS
 アーカイブ・ログ・ストリーム
 考慮事項 160
 ODBM サポート 172
 RRSF (DB2 リカバリー可能リソース・
 サービス接続機能) 14
 RS カタログ・リポジトリ
 IMSRSC リポジトリ の除去 232
 RSE (リカバリー可能サービス・エレメン
 ト)
 可用性マネージャー (AVM) への通知
 819
 定義 745, 758
 RSENAME キーワード 819
 RTCODE マクロ・ステートメント
 オンライン変更で使用 510

S

SB (OSAM 順次バッファリング) 530
 SB 要約報告書 955
 SCI 44
 SCI (Structured Call Interface)
 概要 51
 構成要件 61
 自動 RECON 損失通知 58
 初期設定 27
 セキュリティー 209
 提供される機能 51
 登録出口ルーチン 58
 XRF 744
 SECURITY マクロ
 DBCTL 用の定義 442
 SECURITY マクロ・ステートメント
 SECLVL キーワード 401
 TERMNL キーワード 401
 SETO (オプションの設定) 呼び出し 610
 印刷データ・セットの制御 613
 高速代替 PCB 613
 出力 DD ステートメント 610
 スプール API 記述子 612
 データの書き込み 611
 パージ呼び出し (PURG) 611
 割り振りエラー 614
 ISRT 呼び出し、データの書き込み、
 および 611
 ROLB 呼び出し 612
 ROLL 呼び出し 612
 ROLS 呼び出し 612
 SETS 呼び出し 612
 SETU 呼び出し 612
 XRF、および 614
 SHARELVL パラメーター
 指定の記述 642
 SHAREOPTIONS キーワード
 DEFINE CLUSTER キーワード 700
 SIGN (/SIGN) ON/OFF セキュリティー
 出口ルーチン 421
 SLDS (システム・ログ・データ・セット)
 圧縮 120
 環境 113
 コピー 118
 作成 113
 単一または重複 129
 定義 108, 129
 ブロック・サイズ 129
 累積、DFSUCUM0 を使用した変更の
 671
 レコード 705
 ログ 113
 割り振り 113
 RECON データ・セット 113

SLDS 停止時刻の検出、RECON データ・セット内の最後の 727
SLDS/RLDS (IMS アーカイブ・ログ・データ・セット) の暗号化 383
SMF
DDCONS パラメーターの考慮事項 160
SMP/E
インストール方法 245
コマンド
重要な注意 251
ACCEPT 247, 251
ACCEPT CHECK
GROUPEXTEND
BYPASS(APPLYCHECK) 251
ACCEPT GROUPEXTEND 246
ACCEPT GROUPEXTEND
BYPASS(APPLYCHECK) 247, 251
APPLY 247
APPLY CHECK
GROUPEXTEND 246, 251
APPLY GROUPEXTEND 246, 251
CLEANUP 247
GENERATE 247, 254
JCLIN 247
LIST 247
RECEIVE 246, 247, 251
RESTORE 247, 251
UNLOAD 247
ZONEDELETE 247
ZONEMERGE 247
サービス 239
SNA 端末
XRF 756
SNAPQ チェックポイント 773
SPEED キーワード
DEFINE CLUSTER キーワード 700
SPOC (単一制御点)
概要 52
機能 52
ユーザー作成 52
要件 52
MTO 52
OM API 182
REXX SPOC API 52
TSO SPOC 52
WTOR 52
SSM EXEC パラメーター
外部サブシステムに接続 14
START (/START) AVM コマンド 771
START (/START) IMS コマンド 771
START (/START) SUBSYSTEM SSM コマンド 14

START (/START) SURVEILLANCE コマンド
XRF の使用法 758
START (/START) コマンド
共用レベルの定義 642
STARTNEW パラメーター
使用法 721
STOP (/STOP) SURVEILLANCE コマンド
XRF の使用法 758
Structured Call Interface
セキュリティ 209
SVC
タイプ 2、バインディング 155
タイプ 2、ロード 155
SVC (監視プログラム呼び出し) モジュール 153
SWITCH (/SWITCH) SYSTEM コマンド
計画テークオーバー (planned takeover) 785
テークオーバー処理 776, 783
予定テークオーバー 776
SYS1.NUCLEUS
重要な注意 155
の説明 155
IMS タイプ 2 SVC のインストール 155
SYSGEN
サービス
SYSMOD 回帰、回避 253
保守上の変更点
SYSMOD 回帰、回避 253
SYSMOD 回帰、回避 253
SYSIN/SYSOUT、プログラム・テストでの 464
SYSLIB
適切な連結 255
SYSMOD
回帰、回避 253
システム定義
回帰、回避 253
SYSGEN
回帰、回避 253
T
TCO (時間制御操作) 408
TERMINAL マクロ・ステートメント
BACKUP キーワード 817
NAME キーワード 817
TERMINATE OLC コマンド 578, 593
Tivoli NetView for z/OS 768
VTAM アプリケーション名
USERVAR テーブルの変更 781

Tivoli NetView for z/OS (続き)
VTAM アプリケーション名 (VTAM application name)
新しい名前の通信 768
XRF プロセスに与えられる利点 768
TM バッチ
概要 19
DBBBATCH 19
DLIBATCH 19
GPSB、指定 19
TM リソース 369
トランザクション 371
ユーザー ID 372
ユーザー名 372
APPC 記述子 369
LTERM (論理端末) 369
MSNAME 370
VTAM 端末ノード 370
TPEND 出口
VTAM の障害 758
TPNS (通信網シミュレーター) 464
TRACE CT コマンド 483
TRACE (/TRACE) SET OFF コマンド 36
TRACE (/TRACE) SET ON コマンド 36
TRACE (/TRACE) コマンド
プログラム分離用 483
IMS モニター 480
IMS モニターをオンにするための 492
TRANSACT マクロ
AO コマンド・セキュリティ 404
TRANSACT マクロ・ステートメント
オンライン変更で使用 510
Transaction Manager
リソース
APPC 記述子 369
TSO SPOC 50
開始、IMS Application Menu による 47
概要 52
TSO、代替 IMS システム上で動作する 747
TVS (トランザクション VSAM)
使用可能化 693
RECON への並列アクセス
使用可能化 693
TYPE マクロ・ステートメント、
BACKUP キーワード 817
U
UCF (ユーティリティ制御機能)、データ共用での 652
UIC、NOTIFY.UIC
更新、RECON データ・セットの 658

UNLOCK SYSTEM コマンド
 データベース保全性の保証 762
 UNLOCK (/UNLOCK) SYSTEM コマ
 ンド
 テークオーバー中の処理 780
 UOW (作業単位)
 トラッキング
 共用キュー環境 263
 UPDATEPSB オプション
 PSB の活動化 561
 user
 IMSplex、内
 RM 定義 372
 usermod、サービス 239
 USERVAR
 テーブル
 初期設定プロシージャ 763
 定義 763
 例 763
 IMS の項目の変更 781
 変数定義 763
 USERVAR パラメーター
 XRF で使用する 814

V

VGRS パラメーター
 汎用リソース名の指定 357
 VS Pascal サブルーチンのプリロード
 533
 VSAM
 データ・セット定義 309
 VSAM AMS (アクセス方式サービス・プ
 ログラム)
 復元、RECON データ・セットの 721
 RECON データ・セット、復元 721
 VSAM (仮想記憶アクセス方式)
 キャッシュ構造
 サイズの変更 348
 構造
 サイズの計算 347
 バッファ・プール 530
 VSAM (仮想記憶アクセス方式) 作成モー
 ド
 RECON データ・セット 699
 VSAM 統計報告書
 アプリケーション・プログラム 846
 説明 846
 入出力操作 846
 パフォーマンスの向上
 アプリケーション・プログラムのチ
 ユーニング 846
 データベースのチューニング 846
 報告書のフィールド 847
 VSAM バッファ・プール報告書
 説明 839, 952
 VSAM バッファ・プール報告書 (続き)
 フィールド 953
 報告書の使い方 839
 報告書のフィールド 840
 IMS モニター (DBCTL) 917
 IMS モニター (DB/DC) 887
 //DFSSTAT 952
 VTAM
 シスプレックス内の TM リソースと
 して 370
 端末ノード 370
 汎用リソース 362
 計画 355
 VTAM アプリケーション名
 新しい名前の通信 768
 処理 781
 VTAM アプリケーション名 (VTAM
 application name)
 処理 763
 IMS への定義 816
 XRF (拡張回復機能) 816
 VTAM (仮想記憶通信アクセス方式)
 インターフェースの考慮事項、重要な
 注意 163
 テークオーバーの原因としての障害
 776
 パフォーマンスの考慮事項 535
 パラメーター
 COMCYCL 165
 DELAY 165
 LOGMODE 163
 MODE 163
 MODETBL 163
 並列セッション・サポート 163
 マスター端末の定義 817
 モード・テーブル項目 163
 ログオン・メッセージ処理 763
 論理装置定義 163
 IRLM 命名の提案 161
 MNPS を使用する XRF 複合システム
 についての強制テークオーバー 763
 NCP 遅延 165
 USERVAR 変数 827
 VARY コマンド 163
 XRF テークオーバーの原因としての障
 害 750
 XRF での役割 763
 XRF の計画の考慮事項 810
 XRF 複合システムの端末切り替えに影
 響を与える所有権 810
 XRF プロセスに与えられる利点 763
 XRF 要件 756
 VTAM コマンド
 LOGON APPLID 763
 MODIFY 768
 MODIFY USERVAR 781

VTAM 端末、暗号化 437
 VTAM の論理装置定義 163
 VTAM 汎用リソース
 使用についての制約事項 355

W

WADS (先行書き込みデータ・セット)
 環境 111
 定義 108
 割り振り 111
 OLDS のクローズ 111
 XRF に対する要件 747
 WLM 変更状態パフォーマンス・ブロッ
 ク
 トランザクション (transaction)
 現在の状態 471

X

XRF
 システム障害
 サービスの再開 743
 汎用リソース
 MNPS 358
 USERVAR 358
 XRF IMS へのログオン
 関 763
 ログオン・メッセージの処理 763
 XRF (拡張回復機能)
 アクティブ IMS
 障害 745
 定義 745
 アクティブ IMS からの信号
 IMS システム・ログ 758
 ISC リンク 758
 RDS 758
 アプリケーション・プログラム 756
 オンライン変更 577, 583
 RM のない IMSplex の制約事項
 756
 概念 745
 概要 743
 監視
 オプション 819
 開始 758
 確立 768
 停止 758
 変更 758
 監視のための LNK パラメーター 819
 監視のための LOG パラメーター 819
 監視のための NO パラメーター 819
 監視のための RDS パラメーター 819
 強制テークオーバー、VTAM による
 763

XRF (拡張回復機能) (続き)

クラス 1 端末
 所有権 810
 説明 800
 クラス 2 端末 802
 クラス 3 端末 803
 グローバル・リソース・シリアライゼーション 812
 計画 798, 812
 個々の端末のカスタマイズ 817
 コンポーネントの役割 758
 システム定義 814
 シスプレックス (sysplex)
 グローバル・オンライン変更 583
 終了フェーズ 788
 障害が発生したアクティブ IMS を新しいアクティブ IMS として復帰 786
 制限 750, 798
 セッション・リカバリー 817
 操作要件 756
 ソフトウェア要件 743
 代替 IMS システム 745
 バックアップ・セッション 750
 代替システムとしての別のシステム 786
 端末 800
 端末切り替えに影響を与える VTAM
 所有権 810
 テークオーバー (takeover)
 新しいトランザクションの実行 782
 後の問題判別 750, 786
 基準 768
 原因 745
 速度に影響する要因 781
 定義 745
 ユーザーの視点 805
 テークオーバー後のダンプ・アクティビティ 750, 786
 テークオーバー後の問題判別 750, 786
 テークオーバー条件 819
 テークオーバーの後処理フェーズ 786
 テークオーバーの後処理フェーズにおける EEQE 786
 データ・セットの配置 828
 トラッキング・フェーズ
 アプリケーション・プログラムとトランザクションの状況 774
 従属領域の状況 774
 通信ネットワークの状況 774
 データベースの状況 774
 メッセージ・キューの状況 774
 MFS プールの状況 774
 ネットワーク変更、開始 781
 ハードウェア要件 756
 パスワードの IMS への定義 816

XRF (拡張回復機能) (続き)

バックアップ・オプションの指定 804
 バックアップ・セッション 750
 複合システム
 終了時 788
 初期設定フェーズ 771
 説明 747
 テークオーバー後 750, 786
 テークオーバー前 750
 テークオーバー中の 750, 776
 同期化フェーズ 773
 プロセスのフェーズ 771, 789
 編成 789
 元のアクティブ・システムで実行される作業
 コードの保守 785
 ハードウェア構成変更 785
 ハードウェア保守の実行 785
 バックアップ IMS プロシージャのテスト 785
 予防保守 785
 ライブラリー保守の実行 785
 要件 747
 ソフトウェア 756
 ハードウェア 756
 用語 745
 ライセンス・プログラム要件 756
 リカバリー 135
 リカバリー可能サービス・エレメント (RSE) 745, 819
 利点 744
 IMS 758
 ローカル・キュー・マネージャー・データ・セット 264
 ログオン 758
 ログのクローズと切り替え 778
 ログ保護 778
 1 つの CPC を持つ複合システム 790
 1 つの XRF 複合システム、2 つの CPC、XRF のない IMS の例 792
 1 つの複合システム、2 つの CPC の例 791
 2 つの複合システム、3 つの CPC の例 793
 2 つの複合システム、4 つの CPC の例 796
 APPLID = キーワード 816
 ARM に関する考慮事項 812
 BACKUP= キーワード 804
 DBCTL 機能 750
 DBRC に対する推奨 774
 DB/DC 環境 11
 DFSHSBxx パラメーター 819
 DSE 745
 DSE (従属サービス・エレメント) 定義 758

XRF (拡張回復機能) (続き)

HSBID パラメーター 814
 HSBMBR パラメーター 814
 IMS 2 次端末 817
 IMS の VTAM への定義 827
 IMS マスター端末 817
 IMSplex 744
 RM のないオンライン変更の制約事項 756
 IMSplex 内 25
 IMS.PROCLIB メンバーの指定 819
 ISC リンク 799
 JES に関する考慮事項 812
 MNPS と USERVAR の比較 749
 MNPS パラメーター 814
 MNPSPW パラメーター 814
 MNPSPW= キーワード 819
 MNPS= キーワード 819
 MSC リンク 818
 NCP
 バックアップ・セッション 750
 NCP の計画 811
 NCP の利点 768
 RACF に関する考慮事項 812
 RSENAME= キーワード 819
 SNAPQ 87
 SSP の利点 768
 USERVAR テーブル、VTAM
 テークオーバー時 763
 IMS の VTAM への定義 827
 Tivoli NetView for z/OS による自動化 781
 USERVAR パラメーター 814
 USERVAR 変数
 項目の更新 768
 USERVAR=キーワード 819
 VTAM
 強制テークオーバー 763
 バックアップ・セッション 750
 VTAM アプリケーション名の IMS への定義 816
 XRF に対するシステム定義マクロ 815
 XRF の SWITCH= キーワード 819
 XRF プロセスに対する DFSMS の利点 763
 XRF プロセスに対する VTAM の利点 763
 z/OS の利点 762
 XRF 監視のインターバル値 768
 XRF 監視の確立 768
 XRF クラス 1 端末のバックアップ・セッション
 オープン 750
 クローズ 750
 XRF サポートに対する X.25 端末の要件 756

XRF システム定義のためのマクロ 815
XRF テークオーバー時のセッション切り
替え
 クラス 1 端末 758
XRF テークオーバー時のネットワーク変
更 781
XRF テークオーバーの原因としての IMS
の異常終了 776
XRF での SNA 端末 747
XRF に対する監視メカニズム
 インターバル値の設定 768
 確立 768
 タイムアウト値の設定 768
 パラメーターの概要 768
XRF に対するセッション・リカバリーの
優先順位
 MSC リンクの定義 818
XRF の MNPSPW= キーワード 819
XRF の MNPS= キーワード 819
XRF の RSENAME= キーワード 819
XRF の USERVAR= キーワード 819
XRF の監視の確立 768
XRF 複合システムの EEQE (拡張エラ
ー・キュー・エレメント) 786
XRF 複合システムのオペレーター
 クラス 3 端末のサービスの再確立 747
 相互の通信 778, 783
 代替 IMS システムの従属領域の始動
 773
 立ち上げる
 新しい代替 IMS システム 786
 代替 IMS システムとしての別のシ
 ステム 786
 テークオーバーでの手動制御 776
 テークオーバーの開始
 実際の用途 785
 テークオーバー処理 776
 手順 776
 テークオーバーの作業 805
 入出力防止の完了の確認 783
 入出力防止の実行
 手動での 762, 783
 メッセージ AVM005A に GO と応答
 する
 経過中トランザクション 780
 入出力防止完了時の代替 IMS シス
 テム 778
 メッセージ AVM006E へ GO と応答
 する 762
CPC のリセット 783
IRLM の初期設定 784
RACF 再確認機能の回復 812
SNAPQ チェックポイントの要求 773
XRF から生じる責任 756

XRF 複合システムのネットワーク・オペ
レーター
 MODIFY USERVAR コマンド 763,
 781
 USERVAR テーブルの項目の更新 768
 USERVAR テーブルの項目の追加 763
 XRF から生じる責任 756
XRF プロセスのフェーズ 771

Z

z/OS
 アップグレード 254
 異常終了フォーマット設定 156
 インストール、z/OS PPT 項目の 149
 インストールの問題の予防 147
 インターフェースの考慮事項、重要な
 注意 147
 インターフェース・モジュール 152
 オフライン・ダンプ・フォーマット設
 定 158
 可用性マネージャーの開始 771
 計画に関する考慮事項 812
 コマンド
 CANCEL 798
 START 771
 コンポーネント・トレース (CTRACE)
 サービス 483
 自動リスタート・マネージャー
 (ARM) 91
 障害
 XRF テークオーバーの原因として
 の 750
 調整 309
 のもとで稼働するために必要なステッ
 プ 152
 バインディング 152
 非標準マクロ 149
 への必要な IMS リンク 152
 ログ・ストリーム 115
 定義 272
 ログ・データ・セット 115
APF 許可
 規則 159
 IRLM 考慮事項 161
 JCL の考慮事項 148, 159
 JES 構成 148
APPC/MVS 管理ダイアログの更新
160
DBRC タイプ 4 SVC 159
IMS SVC の定義 153
IMS SVC モジュール 153, 155
IMSpIex 内のコンポーネント 23
IMSpIex 内のシステム・サービス 23
IRLM PPT 161

z/OS (続き)
 IRLM サブシステム名
 作成 161
 SMF DDCONS パラメーター 160
 VTAM 汎用リソース類似性 356
 XRF に対する要件 756
 XRF の計画の考慮事項 812
 XRF プロセスに与えられる利点 762
z/OS システム間カップリング・ファシリ
ティ
 IMSpIex 内での使用 23
z/OS の MEMDSENQMGMT 機能
 使用可能化 156
z/OS の PPT (プログラム特性テーブル)
 更新 149
 BPE の項目 150
 CQS の項目 150
 IMS Connect の項目 151
 IMS の項目 149
 IRLM の項目 151
z/OS のプログラム特性テーブル (PPT)
 更新 149
 BPE 項目 150
 CQS 項目 150
 IMS Connect 項目 151
 IMS 項目 149
 IRLM 項目 151
z/OS マクロ
 DEQ 695
 GRS 695
 OBTAIN 695
 RESERVE 695

[特殊文字]

(モバイル・ワークロード) レポート 474
/CHANGE SURVEILLANCE コマンド
 XRF の使用法 758
/CHECKPOINT コマンド 87
/DBDUMP コマンド
 データベースのバックアップ・コピー
 660
/DISPLAY コマンド
 会計での使用 974
 モニター用 476
 MODIFY 573, 578
/ERESTART BACKUP コマンド 771,
786
/MODIFY コマンド
 オンライン変更で使用 573
 PREPARE 585
/NRESTART
 IMS の障害後の再始動 653
/OPNDST コマンド 163
/RMGENJCL コマンド
 JCL の生成 635

/RMxxxxxx コマンド
概要 622

/SIGN ON/OFF セキュリティー出口ル
ーチン 421

/START AVM コマンド 771

/START IMS コマンド 771

/START SUBSYSTEM SSM コマンド 14

/START SURVEILLANCE コマンド
XRF の使用法 758

/START コマンド
共用レベルの定義 642

/STOP SURVEILLANCE コマンド
XRF の使用法 758

/TRACE SET OFF コマンド 36

/TRACE SET ON コマンド 36

/TRACE コマンド
プログラム分離用 483
IMS モニター 480
IMS モニターをオンにするための 492

//DFSSTAT 報告書
順次バッファリング明細報告書 957
順次バッファリング要約報告書 955
報告書のタイプ 951
DD ステートメント 951
JCL での指定 951
OSAM バッファァー・プール報告書
953
PST 会計報告書 951
VSAM バッファァー・プール報告書 952



プログラム番号: 5635-A06
5655-DS5
5655-TM4

Printed in Japan

SC43-4271-00



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 15.1.0

システム管理

