

IMS  
15.1.0

コミュニケーションおよびコネクション  
(2021-06-25 版)



## お願い

本書および本書で紹介する製品をご使用になる前に、[987 ページの『特記事項』](#)に記載されている情報をお読みください。

本書は、IMS 15 (プログラム番号 5635-A06)、IMS Database Value Unit Edition V15.01.00 (プログラム番号 5655-DS5)、IMS Transaction Manager Value Unit Edition V15.01.00 (プログラム番号 5655-TM4)、および新しい版で明記されていない限り、以降のすべてのリソースおよびモディフィケーションに適用されます。

© Copyright International Business Machines Corporation 1974, 2020.

# 目次

<b>この情報について</b> .....	<b>xxiii</b>
前提知識.....	xxiii
新規および変更された情報の識別方法.....	xxiii
構文図の読み方.....	xxiii
IMS 15 のアクセシビリティ機能.....	xxv
<b>第 1 部 外部 Java 環境接続の構成</b> .....	<b>1</b>
第 1 章 IMS Universal ドライバー : IMS への接続の構成.....	3
IMS Universal ドライバー を使用した EJB 開発用の WebSphere Application Server の構成.....	3
IMS Universal ドライバー : WebSphere Application Server タイプ 4 接続.....	4
WebSphere Application Server でのタイプ 4 IMS Universal Database リソース・アダプター のインストール.....	5
WebSphere Application Server でのタイプ 4 IMS Universal Database リソース・アダプター に対する接続ファクトリーの定義.....	6
WebSphere Application Server でのタイプ 4 IMS Universal Database リソース・アダプター を使用する EAR ファイルのインストール.....	7
IMS Universal ドライバー : WebSphere Application Server for z/OS タイプ 2 接続.....	7
WebSphere Application Server for z/OS でのタイプ 2 IMS Universal Database リソース・ア ダプターのインストール.....	8
オプション: タイプ 2 IMS Universal Database リソース・アダプター を使用するアプリケー ション用の WebSphere Application Server for z/OS クラスパスの設定.....	10
WebSphere Application Server for z/OS でのタイプ 4 IMS Universal Database リソース・ア ダプターに対する接続ファクトリーの定義.....	10
WebSphere Application Server for z/OS でのタイプ 2 IMS Universal Database リソース・ア ダプターを使用する EAR ファイルのインストール.....	12
IMS Universal ドライバー : WebSphere Application Server Liberty タイプ 4 接続.....	13
WebSphere Application Server Liberty タイプ 4 接続の server.xml 構成ファイル例.....	15
IMS Universal ドライバー : WebSphere Application Server Liberty タイプ 2 接続.....	17
WebSphere Application Server Liberty タイプ 2 接続の server.xml 構成ファイル例.....	19
IMS Universal ドライバー : CICS 接続.....	20
タイプ 2 IMS Universal ドライバー 用の CICS の構成.....	20
CICS でのタイプ 2 IMS Universal ドライバー を使用するアプリケーションの実行.....	22
<b>第 2 部 CPI 通信および APPC/IMS</b> .....	<b>23</b>
第 2 章 CPI 通信.....	25
CPI-C ドリブン・アプリケーション・プログラム.....	25
SAA リソース・リカバリー・コミット処理.....	25
正常終了.....	25
バックアウト処理.....	25
異常終了.....	26
セッションの失敗.....	26
戻りコード.....	26
システム再始動/未確定解決処理.....	27
CPI-C アプリケーション・プログラムのリカバリー.....	27
プログラミング要件.....	27
仮名ファイル.....	27
RRS および分散同期点/保護会話.....	28
2 フェーズ・コミット・プロトコル.....	29

ローカル・リソース・リカバリーと分散リソース・リカバリー.....	29
リソース・マネージャーとしての IMS.....	30
保護会話の活動化.....	31
<b>第 3 章 APPC/IMS および LU 6.2 装置の管理.....</b>	<b>33</b>
APPC/IMS の概要.....	33
APPC/IMS フラッディング制御.....	34
APPC/IMS アプリケーション・プログラム・インターフェース.....	35
APPC/IMS アプリケーション・プログラム.....	35
標準 IMS アプリケーション・プログラム.....	36
MSC および標準 IMS アプリケーション・プログラム.....	36
修正済みの IMS アプリケーション・プログラム.....	37
MSC および変更 IMS アプリケーション・プログラム.....	38
CPI 通信ドリブン・アプリケーション・プログラム.....	38
MOD 名および LTERM インターフェースの使用.....	39
APPC/IMS の設定.....	40
TP_Profile.....	40
APPC/MVS 管理ユーティリティー (ATBSDFMU) の例.....	41
アウトバウンド LU 指定.....	43
アウトバウンド・サイド情報.....	43
PARMLIB メンバー.....	44
APPC/MVS タイムアウト・サービス.....	45
APPC/MVS エラー抽出サービス.....	45
LU 6.2 記述子の初期設定および変更.....	45
APPC/IMS 環境における MSC の使用.....	46
MSC 環境における APPC トランザクションの回復.....	47
リカバリー可能トランザクションとリカバリー不能トランザクション.....	47
ローカル APPC トランザクションの廃棄可能性と廃棄不能性.....	48
トランザクション処理の障害点.....	48
LU 6.2 トランザクションのリカバリー可能性フロー.....	51
XRF および APPC の計画.....	52
トランザクション再試行特性.....	53
ネットワーク LU 名の修飾.....	53
単一 IMS システムでの複数の LU の管理.....	54
別の IMS システムへの LU の再割り当て.....	54
DFSAPPC システム・サービス.....	54
メッセージ通信.....	55
非同期出力の送達.....	57
APPC トランザクション・セキュリティー.....	58
<b>第 3 部拡張端末オプション (ETO) (Extended Terminal Option (ETO)).....</b>	<b>61</b>
<b>第 4 章拡張端末オプションの概説.....</b>	<b>63</b>
ETO 用語.....	63
ETO 記述子.....	65
ETO の概念.....	67
記述子および出口ルーチン.....	67
記述子を作成し使用する方法.....	68
ETO 使用の要約.....	69
<b>第 5 章拡張端末オプションの管理.....</b>	<b>71</b>
ETO の計画.....	71
ユーザーの要件の識別.....	72
ETO の制約事項.....	72
物理端末の定義.....	73
VTAM 装置タイプ、画面サイズ、およびモデルの識別.....	76
ETO を用いた高セキュリティー環境の計画.....	81



MFS の計画.....	82
ユーザー ID の計画.....	82
ユーザー・キュー名の計画.....	83
運用の計画.....	83
ETO を用いた MSC サポートの計画.....	83
ETO 記述子のコーディング.....	84
システム定義プロセスを用いた記述子の作成方法.....	84
記述子の格納.....	85
ログオン記述子の作成.....	85
ユーザー記述子の作成.....	88
MFS 装置記述子の作成.....	90
MSC 記述子の作成.....	92
出口ルーチン.....	92
ETO 開始.....	93
ETO 端末へのログオン.....	94
特定の端末タイプへの動的ログオンの制限.....	94
LTERM 制御ブロックの作成および再利用.....	94
デフォルト CINIT または BIND ユーザー・データ形式の使用.....	95
サインオンとキュー LTERM の割り振り.....	95
サインオン・データの提供.....	96
ISC、SLU-P、金融機関、および出力専用装置に対するサインオン・データの提供.....	96
複数のサインオン.....	96
DFS3649A (「サインオンが必要」メッセージ) の受信.....	97
DFS3650I (セッション状況メッセージ) の受信.....	98
ETO 端末と LTERM の関連.....	98
IMS が割り振るキューを判別する方法.....	98
特殊処理モードの設定.....	99
ETO を使用したプリンター.....	100
直接印刷.....	100
印刷の関連付け.....	100
プリンターの定義.....	101
ETO を使用したプリンターの共用.....	102
オペレーター・コマンド.....	102
ETO に関するシステム定義パラメーター.....	103
DLQT パラメーターを使用した DEADQ 状況時間の設定.....	103
自動サインオフ (ASOT).....	104
自動ログオフ (ALOT).....	105
自動サインオフおよび自動ログオフ・タイマー.....	106
自動ログオン.....	106
出力の割り当て.....	107
非同期出力.....	107
非発信元端末への出力メッセージの送達.....	108
不注意による出力データ・ストリーム.....	109
サインオフ.....	109
ログオフ.....	109
ETO 制御ブロック削除によるパフォーマンスの向上.....	109
IDC0 トレース機能.....	110
ETO と LU 6.1 (ISC) 端末.....	110
ETO および STSN 端末.....	111
SNA STSN 端末考慮事項.....	111
ETO、3600/ 金融機関、および SLU P.....	111
ETO STSN 装置 (ISC、金融機関、および SLU P) に対する /SIGN サポート.....	112
ETO を使用した会話モードおよび応答モード.....	112

## 第 4 部外部サブシステム接続機能..... 115

第 6 章 DB2 接続機能.....	117
---------------------	-----

DB2 接続機能を使用するためのシステムの準備.....	117
Java 従属領域の Db2 for z/OS アクセス方法の管理.....	117
<b>第 7 章 External Subsystem Attach Facility (ESAF).....</b>	<b>119</b>
外部サブシステムで用意するもの.....	120
IMS への外部サブシステムの指定方法.....	120
接続処理の基本原則.....	122
サブシステム接続.....	122
アプリケーション呼び出し処理.....	124
リソースの調整.....	124
外部サブシステム・コマンド・サポート.....	125
ESAP で使用できる IMS サービス.....	126
<b>第 8 章外部サブシステム・モジュール・テーブルの作成.....</b>	<b>127</b>
DFSEMODL マクロ.....	127
DFSEWAL マクロ.....	130
<b>第 9 章 IMS 外部サブシステム接続機能の処理.....</b>	<b>135</b>
外部サブシステム接続パッケージのロード.....	135
EEVT 制御ブロックの作成.....	135
外部サブシステム・モジュールのロード.....	136
ESAP の作業域の作成.....	137
外部サブシステム接続の開始.....	137
制御領域識別の延期.....	138
IMS サブシステム始動サービスの使用.....	138
従属領域接続の確立.....	138
通知メッセージ.....	139
アプリケーション・プログラム要求のサポート.....	140
言語インターフェースの定義.....	140
外部サブシステム固有の言語インターフェース・エントリー・ポイント.....	140
複数の外部サブシステムへのアクセス.....	141
リソース・リカバリー・トークン.....	141
外部サブシステム接続の終了.....	142
外部サブシステムからの終了要求.....	142
従属領域接続.....	143
停止状態とは.....	143
<b>第 5 部 IMS Connect および TCP/IP 通信.....</b>	<b>145</b>
<b>第 10 章 IMS Connect の概要.....</b>	<b>147</b>
IMS Connect クライアント・サポート.....	148
IMS Connect による IMS DB へのアクセスのサポート.....	151
IMS Connect の IMS TM Resource Adapter サポート.....	152
IMS Connect による ISC TCP/IP 通信のサポート.....	153
IMS Connect による IMS 間 TCP/IP 通信のサポート.....	154
MSC と IMS 間 TCP/IP 通信.....	156
OTMA と IMS 間 TCP/IP 通信.....	158
IMS Connect の XML 変換サポートの概要.....	159
IMS Connect の z/OS Sysplex Distributor サポート.....	160
IMS Connect セキュリティーの概要.....	161
IMS Connect の定義および実行の概要.....	162
<b>第 11 章 IMS Connect 出口ルーチンの概要.....</b>	<b>163</b>
ユーザー・メッセージ出口ルーチンの概要.....	164
セキュリティーおよび IMS Connect ユーザー・メッセージ出口ルーチン.....	165
ユーザー定義メッセージ.....	166
機能に固有の出口ルーチンの概要.....	166

IMS Connect 出口ルーチンをサポートするマクロ.....	167
出口インターフェース・ブロック.....	168
IMS TM への接続用の XIB 出口インターフェース・ブロック.....	168
IMS TM データ・ストア情報用の XIBDS 出口インターフェース・ブロック.....	169
IMS DB への接続用の XIB1 出口インターフェース・ブロック.....	172
ODBM および IMS DB データ・ストア情報用の XIBOD 出口インターフェース・ブロック.....	173
第 12 章 IMS Connect による IMSplex および共用キューのサポート.....	177
IMS Connect による IMSplex のサポート.....	177
IMSplex サポート環境.....	177
IMS Connect による IMSplex のサポートのインストール.....	178
共用キュー環境での ALTPCB 出力のリトリブ.....	179
第 13 章 IMS Connect セキュリティー・サポート.....	181
IMS Connect の RACF サポート.....	181
RACF 検証用の汎用戻りコードまたはメッセージの有効化.....	182
IMS Connect 内の RACF セキュリティー検査の使用可能化.....	183
IMS Connect に対する RACF セキュリティー統計の使用可能化.....	183
IMS Connect のデフォルトの RACF ユーザー ID.....	184
IMS Connect RACF ユーザー ID キャッシュ.....	185
IMS DB のクライアントに対する IMS Connect のセキュリティー.....	185
IMS Connect によるネットワーク・セキュリティー資格情報の受け渡し.....	186
IMS 間 TCP/IP 接続の保護.....	188
IMS Connect セキュリティー出口ルーチン.....	190
IMS Connect セキュリティーおよび OTMARTUX ユーザー出口.....	190
HWSSMPL0 および HWSSMPL1 セキュリティー処置.....	191
サンプル出口からの RACROUTE 呼び出しでのエラーに対する IMS Connect 応答.....	193
IMS Connect パスワード管理.....	198
クライアント・メッセージの使用による RACF パスワードの変更.....	198
クライアント・メッセージの使用による RACF パスワード・フレーズの変更.....	199
大/小文字混合パスワード・サポートの使用可能化.....	200
IMS Connect の RACF パスケットのサポート.....	201
SSL および IMS Connect.....	206
IMS Connect メッセージのトラステッド・ユーザー・サポート.....	206
IMS Connect 構成メンバーでの OTMA ACEE エージング値の指定.....	207
第 14 章 IMS Connect のコールアウト要求サポート.....	209
同期コールアウト要求のためのユーザー作成 IMS Connect クライアントの構成.....	209
同期コールアウト・メッセージのフォーマット.....	211
RESUME TPIPE による同期コールアウト要求のリトリブ.....	212
同期コールアウト・メッセージの受信の応答.....	214
IMS に対するコールアウト応答のリターン.....	216
IMS に対するエラー応答のリターン.....	217
第 15 章 IMS Connect XML メッセージ変換.....	219
IMS Connect の XML コンバーター.....	219
XML メッセージの構造.....	220
メッセージ変換の例.....	222
第 16 章 IMS Connect メッセージ構造.....	223
IMS Connect クライアント・メッセージの IRM 構造.....	223
IMS Connect へ送信されるメッセージ内の IRM の固定部分のフォーマット.....	224
HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口ルーチンの IRM のユーザー部分のフォーマット.....	229
IRM 拡張のフォーマット.....	237
クライアント出口からの出力構造.....	239
その他の IMS Connect 構造.....	239
メッセージ構造と IMS Connect ユーザー・メッセージ出口ルーチン.....	241

クライアントからの入力メッセージ.....	241
クライアントへの出力メッセージ.....	242
メッセージ構造.....	243
単純な対話でのメッセージ構造の例.....	259
第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド.....	261
IMS Connect で使用される OTMA メッセージ制御フィールド.....	261
IMS Connect で使用される OTMA 状態データ・フィールド.....	267
IMS Connect で使用される OTMA セキュリティー・データ・フィールド.....	281
IMS Connect で使用される OTMA ユーザー・データ・フィールド.....	285
OTMA ヘッダーの表に関する注記.....	292
第 18 章 IMS Connect プロトコル.....	293
トランザクションの制約事項および制限.....	293
コミット・モードおよび同期レベルの定義.....	293
IMS Connect プロトコル・レベル.....	294
IMS Connect 会話型サポート.....	295
OTMA 会話型プロトコル.....	295
IMS Connect 会話型プロトコル.....	297
配信不能なコミット後送信出力のパーズ.....	301
配信不能なコミット後送信出力に対するパーズ機能の指定.....	302
IMS が配信不能なコミット後送信出力をパーズする状況.....	302
コミット後送信出力の転送.....	303
コミット後送信出力に対する転送機能の指定.....	303
転送出力の宛先の指定.....	304
IMS がコミット後送信出力を転送する状況.....	304
リカバリー可能 IMS トランザクション.....	305
送信専用プロトコル.....	306
確認応答付き送信専用プロトコル.....	307
順次配信付き送信専用プロトコル・オプション.....	308
同期コールアウト応答のための送信専用プロトコル.....	308
ソケット接続.....	309
持続ソケット.....	309
トランザクション・ソケット.....	310
非持続ソケット.....	310
IMS TM クライアントのソケット・タイプの設定.....	311
IMS 間 TCP/IP 通信のソケット接続.....	312
トランザクションのソケット処理.....	312
ソケット数の管理.....	313
重複クライアント ID の解決.....	316
IMS Connect による z/OS TCP/IP KeepAlive インターバルのオーバーライド.....	318
TCP/IP 障害.....	318
IMS Connect のタイムアウト指定.....	319
IMS DB クライアントのタイムアウト指定.....	319
IMS TM クライアントのタイムアウト指定.....	319
IMS 間接続のタイムアウト指定.....	329
IMS Connect のトランザクション有効期限サポート.....	329
IMS Connect でのトランザクション有効期限の設定.....	330
OTMA T パイプ保留キューでの出力のリトリート.....	331
RESUME TPIPE/受信プロトコル.....	332
非同期出力サポートの実施.....	334
並列 <b>RESUME TPIPE</b> 要求による出力のリトリート.....	335
出力メッセージのリトリートの管理.....	339
代替 OTMA T パイプ保留キューからの出力のリトリート.....	345
共用非同期出力のグループ定義.....	346
非同期出力メッセージ・フロー.....	347
IMS Connect クライアント呼び出しのフロー.....	348

IMS Connect デッド・レター・キュー (HWS\$DLQ).....	352
IMS Connect の ping サポート.....	352
<b>第 19 章 IMS Connect 2 フェーズ・コミット・サポート.....</b>	<b>355</b>
2 フェーズ・コミット・プロトコルの概要.....	355
分散 2 フェーズ・コミット・サポート.....	356
IMS Universal ドライバー のサポート.....	356
IMS Universal ドライバー を使用するグローバル (XA) トランザクション.....	356
IMS Universal ドライバー を使用する 1 フェーズ・コミット・グローバル・トランザクション.....	359
IMS TM Resource Adapter のサポート.....	363
IMS TM Resource Adapter を使用するグローバル (XA) トランザクション.....	363
IMS TM Resource Adapter を使用する 1 フェーズ・コミット・グローバル・トランザクション.....	367
<b>第 20 章 IMS Connect のユニコードに関する考慮事項.....</b>	<b>369</b>
メッセージ変換.....	369
<b>第 21 章 IMS Connect の TCP/IP 設定.....</b>	<b>371</b>
<b>第 6 部 IMS VTAM ネットワーク管理.....</b>	<b>373</b>
<b>第 22 章 IMS Transaction Manager ネットワークの紹介.....</b>	<b>375</b>
IMS TM ネットワーク概説.....	375
IMS Transaction Manager サービス.....	379
データ通信制御機能 (DCCTL) 環境.....	381
IMS ネットワークの操作.....	382
共用キュー環境.....	383
共用キュー使用の利点.....	384
共用キュー環境に必要なコンポーネント.....	384
Common Queue Server の概説.....	386
汎用リソースによるセッションの平衡化.....	387
IMSplex 端末管理.....	388
リソース構造によるリソースの管理の利点.....	389
共用 TM リソース.....	389
リソース名の一意性.....	389
リソース・タイプ整合性.....	390
高速機能急送メッセージ・ハンドラー.....	390
<b>第 23 章 ネットワークの設計.....</b>	<b>393</b>
ネットワーク管理の設計.....	393
ネットワークと端末要件の文書化.....	394
IMS 端末ネットワーク.....	394
IMS への端末接続.....	395
XRF 端末サポート.....	395
論理端末 (LTERM).....	395
APPC/IMS と LU 6.2 の端末サポート.....	395
IMS メッセージとスケジューリング.....	396
IMS オンライン・システム内のメッセージ・フロー.....	398
会話型トランザクション.....	399
メッセージ通信.....	401
論理端末ネットワークの設計.....	401
論理端末チェーン.....	401
論理端末キュー.....	402
入出力装置の分離.....	402
論理端末と物理端末の関係.....	403
マスター端末.....	404
XRF 複合システムにおけるマスター端末.....	404
NTO 端末.....	404

リソース・モードおよび状態.....	405
端末とユーザーの操作モード.....	405
端末とユーザーの状態.....	406
リソース状況リカバリー.....	407
セキュリティ設計.....	412
TM ネットワーク内でのトランザクション許可.....	413
TM ネットワーク内でのコマンド許可.....	413
トランザクション・コマンド・セキュリティ.....	414
パスワード・セキュリティ.....	414
APPC/IMS のセキュリティ.....	414
ETO のセキュリティ.....	415
高速機能端末の計画.....	415
拡張回復機能 (XRF) の計画.....	416
XRF 複合システムにおける MNPS または USERVAR の使用.....	417
XRF 複合システムの端末.....	418
端末サポートの指定.....	420
高速ネットワーク再接続 (RNR) の計画.....	421
サポート・レベルの指定.....	421
持続セッション・トラッキング.....	422
IMS シャットダウンと RNR.....	423
XRF または VGR との RNR の併用.....	423
端末再接続プロトコル.....	423
サインオン・セキュリティ.....	424
第 24 章 ネットワークの定義.....	425
操作可能ネットワークの準備.....	425
IMS 定義とネットワーク定義の調整.....	426
ホスト・サブシステムとしての IMS の使用.....	426
VTAM ノードの定義.....	426
VTAM ストレージ要件の見積もり.....	427
VTAM バッファ・プール値の決定.....	427
NCP バッファ・プール値の決定.....	427
静的および動的な端末サインオン要件の決定.....	427
LOGON MODE テーブルに関する要件の検査.....	428
初期 VTAM 構成の指定.....	429
IMS における SON/COS サポートの使用.....	430
IMS ネットワークの開始.....	431
セッション開始.....	431
IMS トランザクション・タイプおよびトランザクション状態.....	432
トランザクション状態の判別.....	432
高速ネットワーク再接続 (RNR) についての VTAM の定義.....	433
持続サポート・レベルの定義.....	434
RNR サポート・レベルの定義.....	434
第 25 章 IMS メッセージの編集とフォーマット設定.....	435
メッセージ形式サービス.....	435
MFS コンポーネント.....	436
MFS の管理.....	440
MFS 使用の利点.....	441
MFS 制御ブロック.....	442
SDF II での MFS 形式の作成.....	443
基本編集.....	444
システム間連絡機能 (ISC) の IMS 編集.....	445
透過オプション.....	446
無保護画面オプション.....	446
MFS 編集のバイパス.....	446
端末キーボードのロックとアンロック.....	447
IMS 非グラフィック・メッセージ・データに対するセンシティビティ.....	447

出力メッセージ・セグメントの編集.....	447
MFS による入力メッセージ・セグメントの編集.....	447
基本編集による入力メッセージ・セグメントの編集.....	448
出力装置の制御.....	448
小バッファ装置.....	449
出力制御.....	449
プリンター・コンポーネントの使用.....	450
スプール出力制御.....	450
IBM 3270 情報表示システムのプリンター・コンポーネントの使用 .....	450
候補プリンターの指定.....	451
操作上の考慮事項.....	451
システム間でのプリンターの共用.....	451

## 第 7 部システム間連絡 (ISC)..... 453

第 26 章システム間連絡機能の概要.....	455
ISC と MSC の比較.....	456
ISC に使用可能な IMS 機能.....	457
サンプルのシステム構成.....	460
TCP/IP の ISC サポート.....	462
IMS と CICS との間での ISC.....	463
端末の装置依存データ.....	463
IMS への CICS データの受け渡し.....	464
第 27 章 ISC 接続に使用される VTAM 機能.....	465
VTAM コマンドと標識.....	466
VTAM アプリケーション・プログラミング・インターフェースの使用.....	467
接続確立時のログオン・モードの指定.....	467
2 次論理装置の設計考慮事項.....	468
第 28 章 ISC によって影響を受ける IMS 機能.....	469
メッセージの編集.....	469
ISC セッションからの IMS コマンドの発行.....	470
並列セッションへの影響.....	470
ISC VTAM セッション用 IMS テスト・モードの使用.....	470
ISC 並列セッションでの IMS 制御ブロック・ストレージ.....	471
ISC と IMS の実行モードの関係.....	471
実行モードの外部指定.....	471
実行モードの内部定義.....	472
ISC VTAM 通信中の結果としての処理モード.....	472
LTERM ユーザー (サブプール) とコンポーネント.....	473
第 29 章 ISC プロトコルを使用した通信の設計.....	477
出力プロトコルの判別.....	477
ISC での既存のアプリケーション・プログラムへのアクセス.....	478
MFS を使用するプログラムへのアクセス.....	478
MFS を使用しないプログラムへのアクセス .....	479
メッセージの経路指定.....	479
IMS 間 ISC セッションの考慮事項.....	489
IMS への ISC ノードの静的定義.....	491
パラメーター選択: システム設計の考慮事項.....	494
COMM マクロ・ステートメント.....	494
NAME マクロ.....	494
SUBPOOL マクロ.....	494
TERMINAL マクロ.....	495
システム定義の要約.....	496

第 30 章 VTAM 接続用の ISC プロトコル.....	499
ネットワーク操作.....	499
IMS を作動可能にする.....	499
ISC の IMS ネットワークの開始.....	499
ISC の IMS ネットワークのシャットダウン.....	500
セッションの制御 (セッション制御プロトコル).....	500
ISC VTAM セッションの開始.....	500
XRF 複合システムとの接続の確立.....	501
セッションのバインド.....	502
セッションの再同期.....	504
再始動/再同期プロシージャの設計.....	505
STSN を使用するセッション同期の判別.....	509
再同期の実行.....	511
セッション開始の完了.....	512
セッションの実行.....	512
ISC VTAM セッションの終了.....	513
セッションの再同期のための STSN の使用.....	514
1 次から 2 次へのフロー・マトリックス.....	514
2 次から 1 次へのフロー・マトリックス.....	515
STSN コマンドの形式.....	516
IMS 応答モードまたは会話型出力エラーの処理.....	518
応答モード・エラー.....	519
会話モード・エラー.....	519
ISC による拡張会話の正常終了.....	520
ハーフセッションの同期の維持.....	520
IMS への入力で要求される同期点.....	521
IMS による出力で要求された同期点.....	521
同期点および応答要件.....	522
メッセージ上の同期点標識.....	524
データ・フロー制御プロトコルの関連情報.....	528
BID プロトコル.....	528
ブラケットおよび半二重プロトコル.....	528
CANCEL プロトコル.....	536
プロトコルのチェーン.....	537
CHASE プロトコル.....	538
ERP パージ.....	538
LUSTATUS プロトコル.....	542
ページ化メッセージ ERP.....	544
受信準備済みプロトコル.....	545
RSHUT プロトコル.....	545
選択受信側 ERP.....	545
送信側 ERP.....	550
IMS が受信するセンス・コード.....	552
IMS が送信するセンス・コード.....	553
SIGNAL プロトコル.....	553
LU 6.1 (SBI および BIS) の対称セッション・シャットダウン.....	554
機能管理ヘッダー.....	555
ISC 編集を呼び出すための FM ヘッダーの使用.....	556
プロセスの開始: ATTACH FM ヘッダー.....	557
エラー・リカバリー・プロシージャ FM ヘッダー.....	557
アクティブ・プロセスのリセット: RAP FM ヘッダー.....	558
非同期トランザクション処理の要求: SCHEDULER FM ヘッダー.....	558
システム・メッセージ・プロセス (SYSMSG) および関連 FM ヘッダー.....	559
第 31 章 ISC での MFS の使用.....	561
MFS 入力フォーマット設定の活動化.....	561
MFS 出力フォーマット設定の活動化.....	561



MFS 分散表示管理 (DPM) メッセージ.....	562
MFS ページ削除機能.....	562
MFS オンライン・エラー検出.....	563
MFS での ATTACH および SCHEDULER FM ヘッダー.....	564
データ記述子 FM ヘッダー.....	565
入力データ記述子 FM ヘッダー.....	565
出力データ記述子 FM ヘッダー.....	566
要求時ページ化メッセージの制御: QMODEL FM ヘッダー.....	566
要求 (入力) QMODEL FM ヘッダー.....	567
QGETN FM ヘッダー.....	567
QGET FM ヘッダー.....	568
QPURGE FM ヘッダー.....	568
応答 (出力) QMODEL FM ヘッダー.....	568
QXFR FM ヘッダー.....	568
QSTATUS FM ヘッダー.....	569
MFS での RAP FM ヘッダー.....	569
第 32 章 FM ヘッダー形式の参照.....	571
ATTACH FM ヘッダーの形式.....	571
ATTIU .....	573
ATTDSP .....	574
ATTDBA .....	574
ATTDPN .....	575
ATTPRN .....	577
ATTRDPN および ATTRPRN.....	577
ATTDQN および ATTDQ .....	578
ATTACC .....	579
データ記述子 FM ヘッダーの形式.....	579
エラー・リカバリー・プロシージャ (ERP) FM ヘッダー.....	580
QMODEL FM ヘッダー.....	581
QGET FM ヘッダーの形式.....	581
QGETN FM ヘッダーの形式.....	582
QPURGE FM ヘッダーの形式.....	583
QSTATUS FM ヘッダーの形式.....	584
QXFR FM ヘッダーの形式.....	585
リセット付加プロセス (RAP) FM ヘッダーの形式.....	588
SCHEDULER FM ヘッダーの形式.....	588
SYMSMSG プロセス・ヘッダー.....	589
第 33 章 ISC 編集 ATTACH パラメーターの使用の例.....	591
ISC 編集での ATTACH および SCHEDULER パラメーター.....	591
IMS SYMSMSG プロセスにおける ATTACH パラメーター.....	593
IMS MFS における ATTACH および SCHEDULER パラメーター.....	595
第 34 章 IMS および CICS が ISC インターフェースを使用する方法.....	601
ISC セッションで使用可能な機能.....	601
ISC の CICS 同期処理および非同期処理の概要.....	601
ISC TCP/IP セッションで使用可能な機能.....	603
ISC VTAM セッションで使用可能な機能.....	604
TCP/IP を介する CICS との ISC 通信.....	607
ISC TCP/IP サポートの概要.....	608
ISC TCP/IP サポートの要件.....	609
ISC TCP/IP サポートの制約事項.....	610
ISC TCP/IP 接続のセキュリティー.....	610
CICS との ISC TCP/IP 接続のセットアップ.....	611
ISC TCP/IP リンクでの CICS とのセッションの開始.....	617
ISC TCP/IP セッションの終了.....	619
ISC TCP/IP セッションの再始動.....	620

TCP/IP を介する場合に ISC によってサポートされる CICS フロントエンド・トランザクシ ン・タイプ.....	621
CICS アプリケーション内での CICS EXEC コマンドの一般的なフロー.....	621
SEND/RECEIVE EXEC コマンドを使用して CICS から IMS へ.....	621
SEND LAST EXEC コマンドを使用して CICS から IMS へ.....	623
RECEIVE EXEC コマンドを使用して IMS から CICS へ.....	624
非同期メッセージのコーディング.....	626
START/RETRIEVE EXEC コマンドを使用して CICS から IMS へ.....	626
RETRIEVE EXEC コマンドを使用して IMS から CICS へ.....	628
ISC セッション上で使用してはならないコマンド.....	628
ISC のための該当する CICS インストール・オプションの選択.....	629
CICS システム定義オプションのコーディング.....	629
CICS リソース定義の準備.....	629
IMS-CICS LU 6.1 リンクの定義.....	629
互換性のある IMS ノードと CICS ノードの定義.....	630
システム名.....	630
セッションの数.....	631
セッション名.....	631
他のセッション・パラメーター.....	631
IMS システムとの複数のリンクの定義.....	634
IMS-CICS ISC の CICS トランザクションの定義.....	636
CICS バックアウト未確定処理の定義.....	636
IMS に対する非同期通信用 CICS トランザクションの定義.....	636
CICS からのセッションの開始および割り振り.....	636
セッション開始の他の方法.....	637
CICS からのセッションの終了.....	638
ISC の CICS アプリケーションの設計.....	638
アプリケーション関連の概念.....	639
サブシステム設計: 直接制御とキュー.....	639
ISC VTAM リンクでの同期処理と非同期処理.....	639
基本および代替機能.....	642
CICS 対 IMS 会話モード.....	642
CICS からの IMS コマンドの送信.....	643
同期点.....	644
CICS のための機能管理ヘッダーのコーディング.....	645
ATTACH 機能管理ヘッダー.....	646
SCHEDULER 機能管理ヘッダー.....	649
キュー・モデル機能管理ヘッダー.....	651
データ記述子機能管理ヘッダー.....	651
システム・メッセージ・プロセス (SYSMSG) 機能管理ヘッダー.....	652
エラー・リカバリー・プロシージャ機能管理ヘッダー.....	652
リカバリーおよび再始動の概念.....	652
論理作業単位.....	652
障害後の未解決メッセージ・トラフィックのリカバリー.....	653
トランザクションの異常終了の処理.....	656
再始動のための CICS アプリケーションのコーディング.....	657
第 35 章 ISC データ・フロー制御の例.....	659
MFS 以外のブラケットおよび半二重プロトコルの例.....	659
MFS ブラケットと半二重プロトコルの例.....	660
MFS 出力の例.....	660
MFS 入力 of 例.....	664
SBI/BIS の例.....	665
シグナル・プロトコルの例.....	667
第 36 章 ISC エラー・リカバリー・プロシージャの例.....	669
送信側検出エラーの例.....	669
受信側検出エラーの例.....	670

第 37 章 IMS-CICS ISC のサンプル・プログラム.....	673
インストール手順.....	673
IMS サンプル・プログラム (DFSIS00).....	674
サンプル・プログラムのジョブ制御ステートメント.....	676
IMS システム定義ステートメント.....	677
MFS 形式.....	678
サンプル・プログラムのプログラム仕様ブロック (PSB) 生成.....	679
アプリケーション制御ブロック (ACB) 生成.....	680

## 第 8 部複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC))..... 681

第 38 章複数システム結合機能の概要.....	683
複数システム結合機能の概念.....	683
MSC 物理リンク.....	683
MSC 論理リンク.....	685
MSC 論理リンク・パス.....	686
MSC ネットワークおよび経路指定.....	688
リモート・システムとローカル・システム.....	688
複数システム内のデータ・フロー.....	688
メッセージ経路指定.....	689
経路指定パス.....	689
論理宛先.....	690
入力システム、宛先システム、および中間システム.....	692
システム ID (SYSID).....	693
宛先名および SYSID によるメッセージの経路指定.....	696
リモート LTERM.....	697
複数システム結合機能 (MSC) 直接経路指定.....	699
リモート宛先検査.....	700
第 39 章複数システム結合機能の管理.....	701
複数システムの設計考慮事項.....	701
リソース消費の最小化.....	701
MSC リンクの帯域幅の制御.....	702
リソース要求の平衡化.....	702
会話型処理の設計.....	703
会話を伴う経路指定出口ルーチン.....	704
会話のリモート宛先検査.....	704
切り捨てられたデータの SPA への保管.....	704
会話の終了.....	705
会話の異常終了.....	705
複数システム結合リソースの定義.....	705
IMS システムでの MSC の使用可能化.....	706
MSC= 実行パラメーターによる MSC の使用不可化.....	710
ローカル・システムの定義.....	710
パートナー・システムの定義.....	711
物理リンクの定義.....	711
論理リンクの定義.....	715
論理パスの定義.....	716
リモート・トランザクション用リンク優先順位の設定.....	716
MSC ネットワーク内での逐次トランザクション処理.....	718
出口ルーチンの指定.....	718
複数システムによるネットワーク定義への影響.....	719
システム間のトランザクション定義の検査.....	720
複数システム検査ユーティリティーの使用.....	720
システム定義状況のオンライン検査.....	721
MSC に対するセキュリティーの考慮事項.....	722

複数システム結合機能の操作.....	722
MSC リンク統計.....	723
ベンチマーク・リンク・アクティビティー .....	723
最適な MSC リンク・タイプの判別.....	724
システム・チェックポイントでの定期的な統計のリセット.....	724
リンク・バッファ・サイズをメッセージのサイズに調整する.....	724
MSC 帯域幅モードの論理リンク容量の調整.....	725
最適な MSC リンク・バッファ・サイズの決定.....	725
高位値のリンク統計を使用した MSC リンクの問題の診断.....	728
複数システムのモニターとチューニング.....	729
パフォーマンス情報の調整.....	729
MSC 用の IMS モニターによって生成される報告書 .....	730
複数システム・トランザクション統計の抽出.....	730
ログ・マージの制御.....	730
トランザクション分析報告書の解釈.....	731
MSC および共用キューを使用する IMSplex.....	731
MSC 環境と IMSplex 環境にわたるメッセージ経路指定.....	732
MSC ネットワークから IMSplex ネットワークへのマイグレーション.....	734
MSC と IMSplex が共存する場合の APPC および OTMA のリモート・トランザクション管理 .....	738
疑似異常終了 U0830 の回避.....	744
MSC TCP/IP 汎用リソース.....	745
TCP/IP 汎用リソース・グループ内の MSC リンクの管理.....	746
TCP/IP 汎用リソース・グループ内の MSC リンク類似性の持続性.....	747
TCP/IP 汎用リソース・グループ内の MSC リンク類似性のクリア.....	747
IMS Connect 内の MSC リンク類似性のクリア.....	748
XRF、MSC、および TCP/IP 汎用リソース.....	749
VTAM 汎用リソース (VGR) および MSC.....	750
TM/MSC メッセージ経路指定および制御ユーザー出口ルーチンの概要.....	750
DFSMSCE0 出口ルーチンの IMSplex 類似性経路指定オプション.....	752
MSC での IMSRSC リポジトリの使用.....	753
IMSRSC リポジトリ定義および MSC.....	753
リモート・トランザクションおよび記述子の SIDR 値と SIDL 値の格納方法.....	754
IMSRSC リポジトリ内の MSC リソースの保守.....	754
リポジトリ内の MSC リソースの作成または更新.....	757
リポジトリを使用したリモートからローカルへのトランザクションの更新.....	758
リポジトリを使用したローカルからリモートへのトランザクションの更新.....	758

## **第 9 部 ODBA および DRA 接続..... 761**

第 40 章 CICS を使用した IMS データベースへのアクセス .....	763
PSB に関するコーディング上の考慮事項.....	763
順次バッファリングの使用.....	764
DL/I に接続された CICS.....	764
IMS DBCTL システムへの CICS CCTL 接続の構成.....	765
CICS のタスク.....	766
第 41 章 ODBA インターフェースを介した IMS データベースへのアクセス.....	767
ODBA DRA 始動テーブルの作成.....	767
z/OS アプリケーション領域での ODBA モジュールと DRA モジュールのロードおよび実行.....	768
アプリケーション・プログラムのバインディング .....	768
セキュリティの確立と定義.....	768
RAS セキュリティー.....	769
APSB セキュリティーの定義.....	770

## **第 10 部 Open Transaction Manager Access (OTMA)..... 771**

第 42 章 OTMA の概要.....	773
----------------------	-----

OTMA とは？.....	773
OTMA の機能.....	774
OTMA を使用する利点.....	775
OTMA プロトコルの利点.....	777
OTMA 環境での IMS メッセージ・フロー.....	777
基本的な OTMA メッセージ・フロー.....	778
OTMA の IMS 間 TCP/IP 通信フロー.....	780
コミット後送信トランザクション処理フローのサンプル.....	781
OTMA でのトランザクション・パイプの使用.....	783
トランザクション・パイプの違い.....	784
トランザクション・パイプを使用したメッセージ・フロー.....	784
第 43 章 OTMA の使用可能化と使用.....	789
OTMA の使用可能化.....	789
OTMA 構成パラメーターの要約.....	789
XCF グループ名の定義.....	790
OTMA XCF メンバー名の定義.....	790
OTMA 開始時の定義.....	791
OTMA セキュリティー検査のレベルの定義.....	791
複数のアクティブ RESUME TPIPE 要求の並列処理に対応する OTMA T パイプのサポート.....	792
IBM MQ 用の同期化 T パイプの指定.....	794
OTMA クライアントの OTMAYPRX メンバー名のオーバーライドの使用可能化.....	795
プログラム間通信出力メッセージの非同期送達の指定.....	795
OTMA 記述子.....	795
OTMA クライアント記述子.....	796
OTMA 宛先記述子.....	797
DFSOTMA 記述子.....	800
OTMA 記述子の制限の変更.....	800
IMS 間通信の OTMA サポート.....	801
IMS 間通信のスーパーメンバー・サポート.....	802
リモート・トランザクション・コードの指定.....	803
リモート IMS システムに送信されるメッセージのフォーマット.....	803
OTMA サポートの出口ルーチン.....	804
宛先決定のための OTMAYPRX ユーザー出口および DFSYDRU0 出口ルーチンの使用.....	805
OTMA 用に IMS を管理する.....	807
IMS の会話および OTMA.....	807
MSC および OTMA トランザクション.....	807
高速機能および OTMA トランザクション.....	808
IMS 再始動処理および OTMA.....	808
XRF 処理および OTMA.....	808
IMS Queue Control Facility および OTMA.....	808
OTMA での共用キューの使用.....	810
IMS の終了および OTMA.....	813
IMS 終了の OTMA クライアント通知.....	813
IMS の終了と IMS 間 TCP/IP メッセージ.....	813
OTMA の制約事項および要件.....	814
システム・リソースおよび OTMA の管理.....	814
OTMA リソース・モニター.....	815
現行のトランザクション・ワークロードの表示.....	816
ストレージ上の OTMA メッセージ TIBs の影響.....	816
OTMA ACEE フラッディング制御.....	818
メッセージあふれ検出.....	818
OTMA T メンバーの管理.....	819
T パイプ・リソースのインパクト.....	820
アイドル状態の T パイプの自動除去.....	822
OTMA での会話型トランザクションの終了.....	822
IMS が OTMA 入力メッセージに対して割り振る SAP 数の指定.....	823
IMS メッセージ・キュー・データ・セットのサイズおよび OTMA.....	824

OTMA のバッファ・プールの使用.....	824
従属領域の占有率および OTMA.....	825
OTMA メッセージへの確認応答のタイムアウト間隔.....	825
OTMA のトランザクション有効期限サポート.....	827
OTMA セキュリティー.....	829
OTMA の RACF セキュリティー・レベル.....	830
OTMA セキュリティーの指定.....	831
非同期保留キュー内のメッセージの保護.....	838
OTMA IMS 間 TCP/IP 接続のセキュリティー.....	839
一般的な OTMA のセキュリティー考慮事項.....	840
OTMA 環境での DL/I 呼び出しの使用.....	841
OTMA プログラム間通信処理.....	842
OTMA の単一ストリームのプログラム間通信.....	842
I/O PCB への ISRT なしの OTMA プログラム間通信.....	843
高速 PCB を使用する OTMA プログラム間通信.....	843
複数プログラムへの OTMA プログラム間通信.....	843
保護トランザクションの OTMA プログラム間通信.....	845
その他の OTMA プログラム間通信の考慮事項.....	845
第 44 章 OTMA クライアント.....	847
OTMA クライアントとは?.....	847
OTMA の命名規則.....	848
OTMA クライアントから送信されるメッセージ.....	848
OTMA クライアントからのタイプ 1 コマンドの送信.....	850
OTMA コミット処理.....	850
OTMA コミット処理の要約.....	851
OTMA コミット処理フローのサンプル.....	852
OTMA メッセージ・フローのサンプル.....	857
OTMA でのトランザクション保護.....	862
OTMA クライアントからの保護トランザクションの開始.....	862
IMS における保護トランザクションの処理.....	863
OTMA でのクライアント/サーバー再同期.....	863
OTMA 再同期の前提事項.....	864
リカバリー可能 OTMA トランザクション.....	864
リカバリー不能 OTMA トランザクション.....	864
IMS トランザクションとコマンドの結果の要約.....	865
OTMA 再同期プロトコル.....	866
OTMA 再同期メッセージ・フローのサンプル.....	869
OTMA 再同期メッセージのサンプル.....	871
コミット後送信出力の管理.....	872
コミット後送信非同期出力のページ.....	873
コミット後送信非同期出力の転送.....	873
コミット後送信出力の確認応答のタイムアウト.....	874
非同期コミット後送信出力の共用: OTMA スーパーメンバー機能.....	875
非同期保留キューへの出力の表示.....	876
第 45 章 コールアウト要求の OTMA サポート.....	877
IMS アプリケーション・プログラムからのコールアウト要求.....	877
同期コールアウト要求.....	878
同期プログラム間通信要求.....	879
非同期コールアウト要求.....	881
非同期コールアウト機能の実装.....	881
IMS TM Resource Adapter と非同期コールアウト要求.....	883
SOAP Gateway と非同期コールアウト要求.....	883
IBM MQ と非同期コールアウト要求.....	884
IMS アプリケーション・プログラムおよび非同期コールアウト機能.....	884
RESUME TPIPE 要求のコールアウトおよび OTMA 並列処理.....	887

第 46 章 OTMA メッセージ接頭語.....	889
メッセージ制御情報セクション.....	889
OTMA メッセージ制御情報フィールドの説明.....	892
状態データ・セクション.....	899
サーバー使用可能コマンドおよびクライアント・ビッド・コマンド.....	899
SRVresynch コマンド.....	902
REQresynch コマンド.....	902
REPresynch コマンド.....	903
TBresynch コマンド.....	904
トランザクションおよびコールアウト・メッセージ.....	904
サーバー状態プロトコル・コマンド.....	909
T パイプに対する出力の再開.....	911
「すべての T パイプからの出力再開」プロトコル・コマンドのフォーマット.....	912
T パイプ用の保留キューからの出力再開.....	912
T パイプの保留キューからの出力再開要求の取り消し.....	913
T パイプの保留キュー上にメッセージがない.....	914
セキュリティ・データ・セクション.....	914
OTMA セキュリティ・データ・フィールドの説明.....	916
ユーザー・データ・セクション.....	917
OTMA ユーザー・データ・フィールドの説明.....	918
アプリケーション・データ・セクション.....	918
サンプル OTMA メッセージ.....	919
第 47 章 OTMA 呼び出し可能インターフェース.....	921
OTMA C/I の初期設定.....	923
OTMA C/I のセキュリティ.....	924
OTMA C/I の制約事項.....	924
CM1 トランザクションの otma_send_receive API 呼び出しの後に OTMA C/I セッションがタ イムアウトになる.....	924
第 48 章 OTMA 設計済みトランザクション属性.....	927
<b>第 11 部 SLU P および金融機関通信.....</b>	<b>929</b>
第 49 章 SLU P と金融機関通信の概要.....	931
IMS-SLU P ネットワーク.....	931
システム構成.....	932
SLU P および金融機関ワークステーション.....	932
システム・コントローラー・アプリケーション・プログラム.....	932
MFS および XRF を使用するコントローラー・アプリケーション・プログラムの作成.....	932
XRF システム用のコントローラー・アプリケーション・プログラムに関する考慮事項.....	933
金融機関から SLU P へのコントローラー・アプリケーション・プログラムの変換.....	934
使用する VTAM 機能.....	934
SLU P で使用される VTAM コマンドと標識.....	935
リカバリー要求コマンド.....	936
方向変換標識.....	937
接続の確立とログオン・モードの指定.....	937
XRF 複合システムとの接続の確立.....	938
ブラケットと送信 / 受信の管理.....	938
第 50 章 SLU P および金融機関システムに使用される IMS 機能.....	941
コンポーネントの定義.....	941
LTERM 命名.....	941
出力コンポーネント選択.....	941
入力コンポーネントの判別.....	942
端末応答モード.....	942

端末応答モードのためのワークステーション定義.....	943
ブラケット間状態中に送信された出力メッセージ.....	944
ブラケット間状態中に送信される出力メッセージのための設計.....	945
IMS メッセージ形式サービス.....	946
ワークステーション環境のための MFS の設計.....	946
MID/MOD チェーン.....	946
SLU P システムのための MFS 出力フォーマット設定.....	947
MFS メッセージ・リカバリー.....	947
MFS 制御機能 (金融機関).....	947
MFS 制御機能 (SLU P).....	947
MFS ページングおよび BID オプション.....	948
金融機関ステーションの表示画面保護.....	948
拡張出力コンポーネントの保護 (SLU P).....	949
入出力編集オプション (SLU P).....	950
応答またはブラケットによるリカバリー可能入力の確認.....	951
メッセージ・リカバリー.....	952
メッセージ再同期.....	952
XRF 複合システムの金融機関および SLU P.....	953
金融機関および SLU P での高速機能メッセージ.....	953
高速機能出力メッセージ (金融機関).....	954
高速機能出力メッセージ (SLU P).....	954
高速機能メッセージ再同期.....	955
第 51 章 SLU P および金融機関でのネットワーク操作.....	957
IMS ネットワークの開始.....	957
IMS を作動可能にする.....	957
セッション開始 (ワークステーション開始).....	957
セッション開始伝送順序.....	958
メッセージ再同期でのコントローラー・アプリケーション・プログラム関与.....	959
設計の考慮事項.....	959
シーケンス番号管理.....	959
シーケンス番号の設定およびテスト (STSN).....	960
IMS からの出力の延期.....	963
セッション終了.....	963
順序正しい終了.....	964
即時終了.....	965
IMS ネットワークのシャットダウン (SLU P).....	965
SLU P メッセージ.....	966
送信 / 受信とブラケット・プロトコル.....	966
第 52 章 SLU P メッセージ・プロトコル.....	967
入力機能管理ヘッダーの一般形式 (金融機関).....	967
入力メッセージ記述子バイト (金融機関).....	967
入力機能管理ヘッダーの一般形式 (SLU P).....	968
入力メッセージ記述子バイト (SLU P).....	969
入力コンポーネント識別 (SLU P).....	970
入力ブラケット付けプロトコル.....	970
金融機関ワークステーションに対する MFS 入力フォーマット設定の活動化.....	971
出力メッセージ.....	971
MFS 分散表示管理出力 (SLU P).....	973
出力機能管理ヘッダーの一般形式 (金融機関).....	973
出力メッセージ記述子バイト (金融機関).....	974
出力コンポーネント ID バイト (金融機関).....	974
MFS データ・バイト (金融機関).....	974
出力機能管理ヘッダーの一般形式 (SLU P).....	975
出力メッセージ記述子バイト (SLU P).....	975
MFS データ・バイト (SLU P).....	976
出力ブラケット付けプロトコル.....	977



SLU P に対する MFS 出力フォーマット設定の活動化.....	977
応答要求 (金融機関).....	978
応答要求 (SLU P).....	978
入力応答要件.....	978
出力応答要件.....	979
IMS トランザクション・タイプ.....	980
リカバリー可能な照会トランザクション.....	980
リカバリー不能照会トランザクション.....	981
リカバリー不能メッセージの IMS 受信の検証.....	981
IMS メッセージ通信.....	981
IMS コマンド.....	982
VTAM コマンドと標識.....	982
MFS 制御要求.....	982
エラー処理.....	982
IMS 検出エラー.....	982
コントローラーまたはステーション検出エラー.....	983
VTAM 論理装置状況 (LUSTATUS) コマンド.....	985
VTAM 受信可 (RTR) コマンド.....	985
VTAM CANCEL コマンド.....	985
VTAM リカバリー要求コマンド.....	986
<b>特記事項.....</b>	<b>987</b>
プログラミング・インターフェース情報.....	988
商標.....	989
製品資料に関するご使用条件.....	989
IBM オンライン・プライバシー・ステートメント.....	990
<b>参考文献.....</b>	<b>991</b>
<b>索引.....</b>	<b>993</b>



## この情報について

これらのトピックでは、IMS コミュニケーションおよびコネクション (CPI コミュニケーションと APPC/IMS、外部サブシステムへの接続機能、IMS 拡張端末オプション (ETO)、IMS Connect、IMS Universal ドライバー接続、システム間連絡 (ISC)、複数システム結合機能 (MSC)、IMS Open Database Access (ODBA) およびデータベース・リソース・アダプター (DRA) インターフェース、IMS Open Transaction Manager Access (OTMA)、SLU P および金融機関通信システム、TCP/IP 通信、および VTAM® ネットワーキング) の管理方法について説明しています。

この情報は、[IBM® 資料](#) で参照できます。

## 前提知識

最初に *IMS V15* システム管理を読み取る必要があります。計画アクティビティー、イムス環境、および管理の概念をカバーする導入部の章は、本書の背景として特に役立ちます。

APPC/IMS を使用する場合は、APPC/MVS 構成を正しく定義するために、APPC/MVS について十分に理解しておく必要があります。APPC/MVS の詳細については、*CPI Communications Reference* を参照してください。

イムスと CICS® の間に ISC セッションをインプリメントする場合は、*CICS Transaction Server for z/OS CICS Intercommunication Guide* 内の情報を理解しておく必要があります。

TCP/IP 接続を使用してイムスに接続する場合は、*z/OS® Communications Server: IP 構成ガイド* で説明されているように、*z/OS* システム TCP/IP を理解する必要があります。

*z/OS* の詳細については、[IBM 資料](#) の「*z/OS basic skills*」トピックを参照してください。

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

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

## 新規および変更された情報の識別方法

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

改訂マーカーは、以下の一般的な規則に従っています。

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

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

## 構文図の読み方

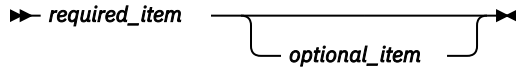
本書で使用されている構文図には、以下の規則が適用されています。

- 構文図は、経路を示す線に沿って、左から右、上から下に読み取ります。以下の規則が使用されます。
  - >>--- 記号は、構文図の始まりを示します。
  - ---> 記号は、構文図が次の行に続くことを示します。

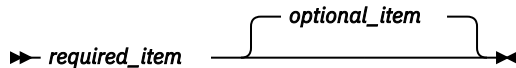
- >--- 記号は、この構文図が直前の行から続いていることを示します。
- --->< 記号は、構文図の終わりを示します。
- 必須項目は、水平線 (メインパス) 上に表示されます。

▶▶ *required\_item* ▶▶

- オプション項目は、メインパスより下に示されます。

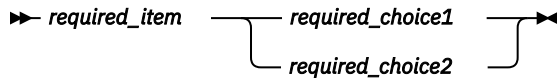


メインパスより上にオプション項目が示されている場合は、その項目が構文エレメントの実行に影響することはなく、読みやすくするためのみの表記です。

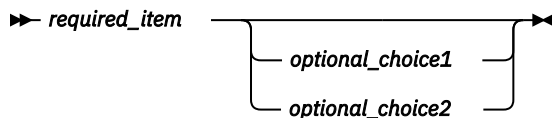


- 複数の項目から選択できる場合は、縦方向に並べて (スタック) 示されます。

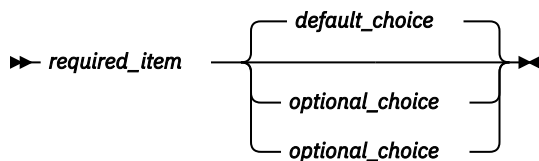
それらの項目の中から 1 つ を選択する必要がある 場合は、スタックの中の 1 つの項目がメインパス上に表示されます。



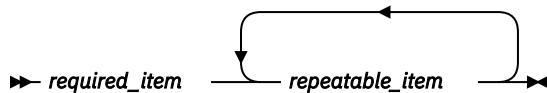
それらの項目から 1 つを選択することがオプションである場合は、スタック全体がメインパスの下に表示されます。



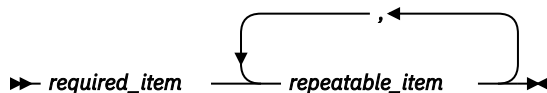
デフォルト項目が含まれている場合、その項目はメインパスより上に示され、他の選択項目はメインパスより下に示されます。



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

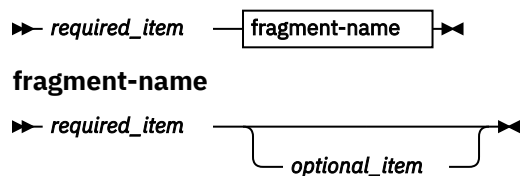


反復矢印線にコンマが含まれている場合は、反復項目をコンマで区切る必要があります。



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

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



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

## IMS 15 のアクセシビリティ機能

アクセシビリティ機能は、運動障害または視覚障害など身体に障害を持つユーザーが情報技術製品を快適に使用できるようにサポートします。

### アクセシビリティ機能

以下のリストは、IMS 15 を含む z/OS 製品の主なアクセシビリティ機能を示しています。これらの機能は、以下をサポートしています。

- キーボードのみの操作。
- スクリーン・リーダー (読み上げソフトウェア) およびスクリーン拡大鏡によって通常使用されるインターフェース。
- 色、コントラスト、フォント・サイズなど表示属性のカスタマイズ。

### キーボード・ナビゲーション

IMS 15 ISPF パネル機能には、キーボードまたはキーボード・ショートカット・キーを使用してアクセスできます。

TSO/E または ISPF を使用して IMS 15 ISPF パネルをナビゲートする詳細については、「z/OS TSO/E 入門」、*「z/OS TSO/E ユーザーズ・ガイド」*、および「z/OS 対話式システム生産性向上機能 (ISPF) ユーザーズ・ガイド 第 1 巻」を参照してください。上記の資料には、キーボード・ショートカットまたはファンクション・キー (PF キー) の使用方法を含む、各インターフェースのナビゲート方法が記載されています。それぞれの資料では、PF キーのデフォルトの設定値とそれらの機能の変更方法についても説明しています。

### 関連のアクセシビリティ情報

IMS 15 のオンライン資料は、IBM 資料で参照できます。

### IBM におけるアクセシビリティ

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



---

# 第 1 部 外部 Java 環境接続の構成

IMS Universal ドライバー は、IMS にアクセスする Java™ アプリケーションをサポートします。IMS Universal ドライバー は業界標準およびオープン仕様に基づいて構築されており、接続、データ・アクセス方式、およびトランザクション処理のオプションに対する柔軟なサポートを提供します。

## 関連概念

[IMS Universal Database リソース・アダプターによりサポートされるトランザクション・タイプおよびプログラミング・インターフェース \(アプリケーション・プログラミング\)](#)

[Java 開発用の IMS ソリューションの概要 \(アプリケーション・プログラミング\)](#)





# 第 1 章 IMS Universal ドライバー : IMS への接続の構成

IMS Universal ドライバー は、z/OS 環境および分散環境で実行できます。これには、WebSphere® Application Server for z/OS、分散プラットフォーム用 WebSphere Application Server、および WebSphere Application Server Liberty が含まれます。

## このタスクについて

IMS Universal ドライバー には、以下のアダプターとドライバーが含まれます。

- Java プラットフォーム、Enterprise Edition (Java EE) サービスを利用する IMS Universal Database リソース・アダプター
- IMS データに直接アクセスする SQL 呼び出しをサポートする IMS Universal JDBC ドライバー
- Java プログラミング・インターフェースで DL/I に類似した呼び出しを提供する IMS Universal DL/I ドライバー

サーバー上の分散環境 (分散プラットフォーム用 WebSphere Application Server など)、サーバー上のリモート z/OS 環境 (WebSphere Application Server for z/OS など)、または WebSphere Application Server Liberty で実行されている IMS Universal ドライバー は、TCP/IP 通信およびソケット管理をサポートするタイプ 4 接続アーキテクチャーを使用して IMS に接続します。

IMS と同じ論理区画 (LPAR) 上でローカルに実行される IMS Universal ドライバー は、IMS Open Database Access (ODBA) インターフェースおよび IMS データベース・リソース・アダプター (DRA) インターフェースを介して IMS との直接通信をサポートするタイプ 2 接続アーキテクチャーを使用して、IMS に接続します。

WebSphere Application Server は、分散環境と z/OS 環境の両方で、すべての IMS Universal ドライバー をサポートします。

## 関連概念

[IMS Universal ドライバーを使用したプログラミング \(アプリケーション・プログラミング\)](#)

[IMS Universal ドライバーを使用した分散およびローカル・コネクティビティー \(アプリケーション・プログラミング\)](#)

## 関連資料

[IMS にアクセスする IMS Universal ドライバー・プログラミング方式の比較 \(アプリケーション・プログラミング\)](#)

## IMS Universal ドライバー を使用した EJB 開発用の WebSphere Application Server の構成

z/OS プラットフォームまたは分散プラットフォームで、WebSphere Application Server 上で稼働する Enterprise JavaBeans (EJB) アプリケーションを開発するには、Java 仮想マシン (JVM) 用および XML/Java バインディング用のカスタム・プロパティを正しく構成する必要があります。

## 手順

IMS Universal ドライバーを使用する EJB アプリケーション用の WebSphere Application Server の構成

1. 管理コンソールで、以下のように適切なサーバー・インスタンスおよびサーバントを選択します。
  - a) 「**Servers**」 > 「**Application servers**」 > 「**server\_name**」をクリックします。
  - b) 「Server Infrastructure」セクションで、「**Java and process management**」 > 「**Process Definition**」を選択します。
  - c) 「**Servant**」を選択します。
2. Java 仮想マシンのカスタム・プロパティ  
**com.ibm.ws.runtime.component.ResourceMgr.postBindNotify** を true に設定します。

- a) 「Additional Properties」の下で、「**Java virtual machine**」をクリックします。
  - b) 「Additional Properties」で、「**Custom Properties**」をクリックします。
  - c) **com.ibm.ws.runtime.component.ResourceMgr.postBindNotify** という名前のカスタム・プロパティを **true** に設定します。  
このカスタム・プロパティが定義済みカスタム・プロパティのリストに存在しない場合は、このカスタム・プロパティを作成して、**true** に設定します。
3. IMS で IMS カタログが使用可能に設定されており、まだ **javax.xml.bind.JAXBContext** カスタム・プロパティが存在しない場合は、以下の手順を実行します。
- a) 「Additional Properties」の下で、「**Java virtual machine**」をクリックします。
  - b) 「Additional Properties」で、「**Custom Properties**」をクリックします。
  - c) **javax.xml.bind.JAXBContext** という名前のカスタム・プロパティを作成し、それを **com.sun.xml.internal.bind.v2.ContextFactory** に設定します。
4. 変更を保管します。
5. サーバーを再始動します。

### タスクの結果

接続ファクトリー MBean によって公開された `getProperty` メソッドを使用して、対応する接続ファクトリーのカスタム・プロパティを照会することができます。

## IMS Universal ドライバー :WebSphere Application Server タイプ 4 接続

---

WebSphere Application Server 上で実行される Java アプリケーションは、IMS Universal ドライバー が提供するタイプ 4 接続を使用して IMS データベースにアクセスできます。IMS Universal ドライバー のタイプ 4 接続を使用することにより、Java アプリケーション・プログラムはさまざまな分散環境やメインフレーム環境から、スタンドアロン・モードまたはアプリケーション・サーバーの下で、グローバル・トランザクションに対する XA サポートの有無にかかわらず、IMS データベースにアクセスできます。

### このタスクについて

以下の図は、IMS Universal Database リソース・アダプターのタイプ 4 接続を使用して分散プラットフォーム用 WebSphere Application Server から IMS に接続する構成の概要を示しています。

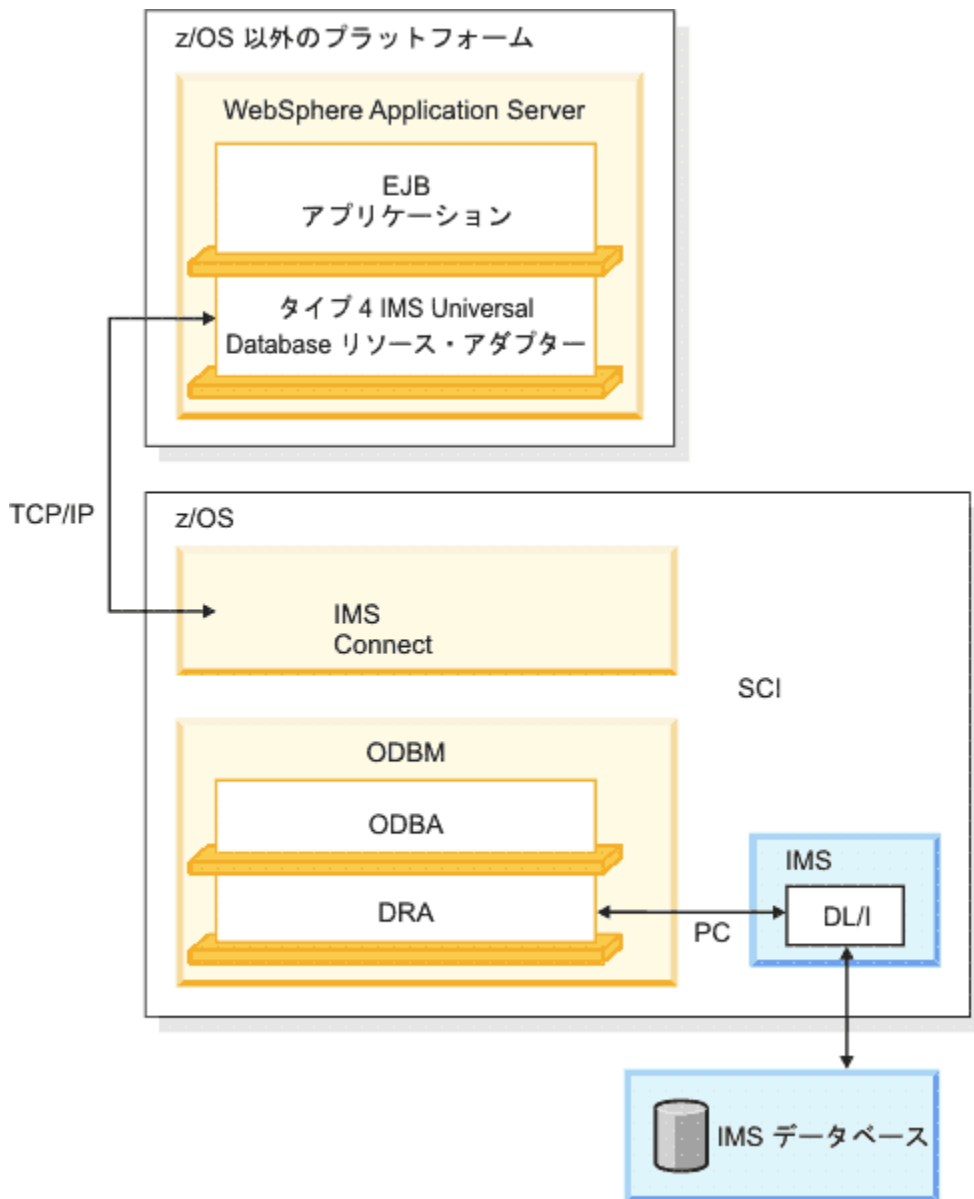


図 1. タイプ 4 IMS Universal Database リソース・アダプターを使用する WebSphere Application Server EJB アプリケーション

以下の手順は、分散プラットフォーム用 WebSphere Application Server と WebSphere Application Server for z/OS の両方に適用されます。

#### 関連概念

IMS Universal Database リソース・アダプターによりサポートされるトランザクション・タイプおよびプログラミング・インターフェース (アプリケーション・プログラミング)

## WebSphere Application Server でのタイプ 4 IMS Universal Database リソース・アダプターのインストール

WebSphere Application Server から IMS Universal Database リソース・アダプターを使用して IMS データベースにアクセスするには、事前にそのリソース・アダプターをインストールする必要があります。

### 始める前に

前提条件：

- 3 ページの『[IMS Universal ドライバー を使用した EJB 開発用の WebSphere Application Server の構成](#)』の手順に従って、WebSphere Application Server を構成する必要があります。
- WebSphere Application Server からアクセス可能なストレージに、IMS Universal ドライバー の RAR ファイルをコピーします。

## このタスクについて

IMS Universal Database リソース・アダプターをインストールするには、以下のようになります。

### 手順

1. WebSphere Application Server 管理コンソールで、「**Resources**」 > 「**Resource Adapters**」 > 「**Resource Adapters**」をクリックします。
2. 「**Install RAR**」をクリックします。  
リソース・アダプターをインストールするためのダイアログが開きます。
3. RAR ファイルのパスを入力します。パスには、ローカルの場所またはサーバー上の場所を指定できません。
  - RAR ファイルがローカル・ワークステーション上にある場合は、「**Local path**」を選択し、ファイルを検索します (C:\¥install\_directory¥imsudbLocal.rar など)。
  - RAR ファイルがサーバー上にある場合は、「**Server path**」を選択し、ファイルの絶対パスを指定します
4. 「**Next**」をクリックします。  
構成パネルが開きます。
5. 「**OK**」をクリックします。  
IMS Universal Database リソース・アダプター がリストされます。
6. メッセージ・ボックスで、「**Save**」をクリックします。  
保管ページが表示されます。
7. 「**Save**」をクリックして、マスター・リポジトリの変更を更新します。

### 関連資料

[IMS にアクセスする IMS Universal ドライバー・プログラミング方式の比較 \(アプリケーション・プログラミング\)](#)

## WebSphere Application Server でのタイプ 4 IMS Universal Database リソース・アダプターに対する接続ファクトリーの定義

IMS Universal Database リソース・アダプターをインストールした後に、WebSphere Application Server 管理コンソールを使用して接続ファクトリーを定義します。

### このタスクについて

IMS Universal Database リソース・アダプターに対して接続ファクトリーを定義するには、以下のようになります。

### 手順

1. WebSphere Application Server 管理コンソールの左方フレームで、「**Resources**」 > 「**Resource Adapters**」 > 「**Resource Adapters**」をクリックします。
2. ユーザーがアダプターのインストール時に選択した IMS Universal Database リソース・アダプター の名前をクリックします。
3. 「Additional Properties」で、「**J2C connection factories**」をクリックします。
4. 「**New**」をクリックします。
5. 以下の情報を入力します。

「Name」: 接続ファクトリーの名前 (例えば PhonebookCF)

「JNDI Name」: 当該サーバー内で固有の、接続ファクトリーの JNDI 名 (例えば PhonebookCF)

6. 「Apply」をクリックします。

「J2C Connection Factories」にデータ・ソースがリストされます。

7. 「Additional Properties」の下で、「Custom Properties」をクリックする。

8. ご使用のシステムで以下の各プロパティに定義されている値を指定します。そのためには、プロパティ名をクリックし、開いた構成ペインの「Value」フィールドでプロパティの値を指定します。プロパティの値を指定したら、「OK」をクリックしてプロパティのリストに戻ります。

接続プロパティのリストについては、[管理対象環境での IMS Universal Database リソース・アダプターを使用した接続 \(アプリケーション・プログラミング\)](#)を参照してください。

9. メッセージ・ボックスで、「Save」をクリックします。

10. WebSphere Application Server を再始動します。

## 次のタスク

### 関連概念

[IMS Connect の定義と調整 \(システム定義\)](#)

## WebSphere Application Server でのタイプ 4 IMS Universal Database リソース・アダプターを使用する EAR ファイルのインストール

タイプ 4 IMS Universal Database リソース・アダプターをインストールして接続ファクトリーを構成した後、クライアント・アプリケーション・プログラム (EAR ファイル) を WebSphere Application Server にインストールする必要があります。

### このタスクについて

EAR ファイルを WebSphere Application Server にインストールするには、以下のようにします。

### 手順

1. WebSphere Application Server 管理コンソールの左側で、「Applications」 > 「Install New Application」をクリックします。
2. EAR ファイルのパスを入力するか、「Browse」ボタンをクリックして EAR ファイルの場所を探します。「Next」をクリックします。
3. 「Selection installation options」パネルで、デフォルトを受け入れて「Next」をクリックします。
4. 「Map modules to servers」パネルで、デフォルトを受け入れて「Next」をクリックします。
5. 「Summary」パネルで、オプションが適切であることを確認して「Next」をクリックします。アプリケーションがインストール中であることを示す一連のメッセージが表示されます。
6. アプリケーションが正常にインストールされたことを示すメッセージが表示されたら、「Save」をクリックします。

### タスクの結果

インストールが終了すると、アプリケーションは開始可能となります。

### 関連概念

[IMS Universal ドライバーを使用したプログラミング \(アプリケーション・プログラミング\)](#)

## IMS Universal ドライバー : WebSphere Application Server for z/OS タイプ 2 接続

WebSphere Application Server for z/OS と IMS が同じ論理区画 (LPAR) 上にある場合、WebSphere Application Server for z/OS で実行されている Java アプリケーションは、IMS Universal Database リソース・アダプターが提供するタイプ 2 接続を使用して IMS データベースにアクセスできます。

## このタスクについて

IMS Universal Database リソース・アダプターのタイプ 2 接続では、IMS データベースに TCP/IP を使用しないでローカルにアクセスします。

タイプ 2 接続を使用するアプリケーションをデプロイするには、WebSphere Application Server for z/OS に IMS Universal Database リソース・アダプターのいずれかをインストールし、IMS Open Database Access (ODBA) インターフェースを構成する必要があります。

IMS Universal Database リソース・アダプターのタイプ 2 接続は、Bean 管理の Bean メソッドおよびコンテナ管理の Bean メソッドの両方をサポートします。任意選択で、トランザクション・アプリケーションを z/OS リソース・リカバリー・サービスのローカル・オプションで管理することができます。

以下の図に、IMS データにアクセスする EJB アプリケーションを示します。データベース要求は、タイプ 2 IMS Universal Database リソース・アダプターに渡され、そこで DL/I 呼び出しに変換されます。IMS Universal Database リソース・アダプターは、これらの呼び出しを ODBA に渡し、ODBA は IMS データベース・リソース・アダプター (DRA) インターフェースを使用して、IMS 内の DL/I 領域にアクセスします。

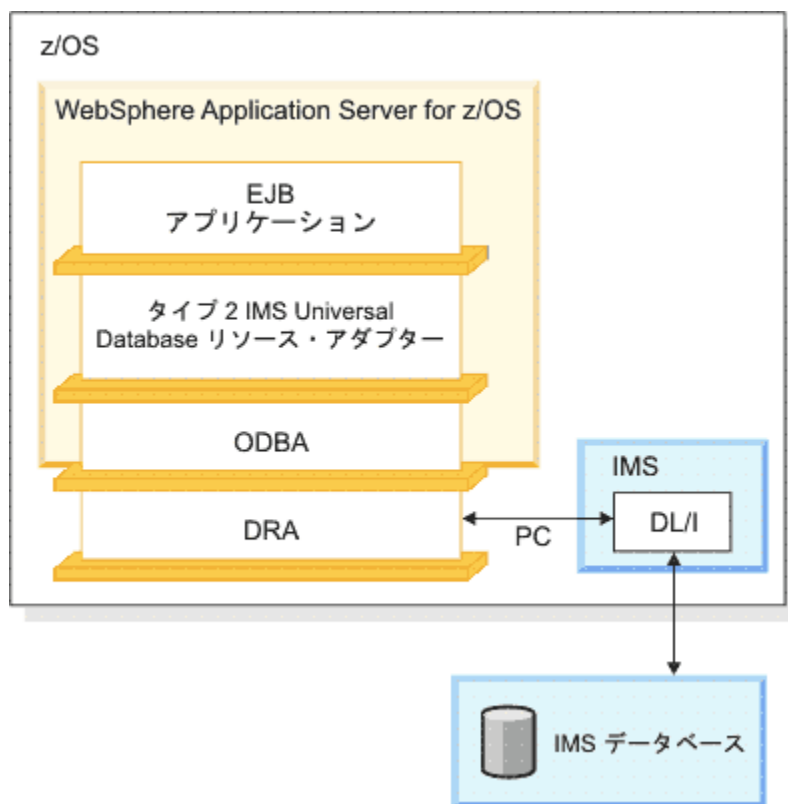


図 2. タイプ 2 IMS Universal Database リソース・アダプターを使用する WebSphere Application Server for z/OS EJB アプリケーション

## WebSphere Application Server for z/OS でのタイプ 2 IMS Universal Database リソース・アダプターのインストール

IMS データベースにアクセスできるように WebSphere Application Server for z/OS を構成した後で、WebSphere Application Server for z/OS にタイプ 2 IMS Universal Database リソース・アダプターをインストールする必要があります。

### 始める前に

#### 前提条件:

1. 3 ページの『IMS Universal ドライバーを使用した EJB 開発用の WebSphere Application Server の構成』で説明されているステップを実行します。



2. WebSphere Application Server for z/OS からアクセス可能なストレージに、IMS Universal ドライバー の RAR ファイルをコピーします。
3. まだ作成していない場合は、ODBA 始動テーブルを作成します。ODBA 始動テーブル・モジュール名は、5 バイトから 8 バイトの長さに指定でき、以下の命名規則に準拠していなければなりません。
  - バイト 1 から 3 は、「DFS」でなければならない
  - バイト 4 から 7 は、1 バイトから 4 バイトの ID とする
  - 最終バイトは、数字の「0」でなければならない例えば、DFS10 と DFSIMSA0 はどちらも ODBA 始動テーブルの有効なモジュール名です。  
**推奨事項:** IMS ID は 1 バイトから 4 バイトまでの ID として使用します。
4. まだリンクしていない場合は、ODBA 始動テーブルをロード・ライブラリーにリンクします。
5. STEPLIB に以下のデータ・セットを追加して、WebSphere Application Server for z/OS の JCL を更新します。
  - ODBA 始動テーブルおよび ODBA ランタイム・コードを含むロード・ライブラリー。
  - SDFSJLIB データ・セット。このデータ・セットには DFSCLIB メンバーが含まれています。
6. MBR パラメーターによって定義される ODBA 名に注意してください。データ・ソースのインストール時に、この名前のバイト 4 から 7 を知っている必要があります。その部分は通常 IMS システム ID です。

## このタスクについて

IMS Universal Database リソース・アダプターをインストールするには、以下のようになります。

### 手順

1. WebSphere Application Server for z/OS 管理コンソールで、「**Resources**」 > 「**Resource Adapters**」 > 「**Resource Adapters**」をクリックします。  
リソース・アダプターのリストが表示されます。
2. 「**Install RAR**」をクリックします。  
リソース・アダプターをインストールするためのパネルが表示されます。
3. 「Install RAR」パネルで、以下を行います。
  - 「**Scope**」の下で、ノードを選択する
  - 「**Path**」の下で、RAR ファイルへのパスを入力するか、「**Browse**」ボタンをクリックして RAR ファイルを見つける
4. 「**Next**」をクリックします。  
構成パネルが開きます。
5. 「Configuration」パネルの「**General properties**」 > 「**Native library path**」の下で、libT2DLI.so ファイルを含むディレクトリーへのパスを入力して、「**OK**」をクリックします。  
libT2DLI.so ファイルは、UNIX システム・サービスで適切な読み取り許可および実行許可が与えられている必要があります。また、WebSphere サーバント領域の STEPLIB に SDFSJLIB が含まれていることも必要です。  
「**OK**」をクリックすると、インストールした IMS Universal Database リソース・アダプターがリストされます。
6. メッセージ・ボックスで、「**Save**」をクリックします。  
変更内容をノードと同期化する必要があるかどうかを尋ねるページが表示されます。
7. 「**OK**」をクリックして、マスター・リポジトリーの変更を更新します。

### 関連概念

データベース・リソース・アダプター (DRA) (システム・プログラミング API)

### 関連タスク

[767 ページの『ODBA インターフェースを介した IMS データベースへのアクセス』](#)

Open Database Access (ODBA) は、z/OS の リカバリー可能な、リソースにより管理された z/OS アドレス・スペースがある場合、それが DL/I データベース呼び出しを IMS DB サブシステムに対して行うための呼び出し可能インターフェースを提供します。

## オプション: タイプ 2 IMS Universal Database リソース・アダプターを使用するアプリケーション用の WebSphere Application Server for z/OS クラスパスの設定

ご使用のアプリケーションに IMS データベース・メタデータ・クラス (IMS Enterprise Suite Explorer for Development によって生成された DLIDatabaseView サブクラス) を組み込むことも、メタデータ・クラスを別の場所に保管することもできます。

### このタスクについて

アプリケーションにメタデータ・クラスが組み込まれていない場合は、WebSphere Application Server for z/OS クラスパスを、アプリケーションが使用する IMS データベース・メタデータ・クラスをポイントするように設定する必要があります。

クラスパスを設定する 1 つの方法として、これらのファイルを IMS Universal Database リソース・アダプターのクラスパスに追加する方法があります。

必要なファイルを IMS Universal Database リソース・アダプターのクラスパスに追加するには、以下のようになります。

### 手順

1. WebSphere Application Server for z/OS 管理コンソールから、「**Resources**」 > 「**Resource Adapters**」をクリックします。  
リソース・アダプターのリストが表示されます。
2. IMS Universal Database リソース・アダプターの名前をクリックします。  
構成ダイアログが表示されます。
3. クラスパスのテキスト・ボックスで、メタデータ・クラスの場所を追加します。
4. 「**OK**」をクリックします。
5. メッセージ・ボックスで、「**Save**」をクリックします。  
「Save」ページが表示されます。
6. 「**Save**」をクリックして、マスター・リポジトリの変更を更新します。

### 関連資料

[実行時 Java メタデータ・クラスの生成 \(アプリケーション・プログラミング\)](#)

## WebSphere Application Server for z/OS でのタイプ 4 IMS Universal Database リソース・アダプターに対する接続ファクトリーの定義

DataSource 機能は、物理データ・ソース (データベース) への接続のファクトリーです。データ・ソースは、Java Naming and Directory (JNDI) API に基づくネーミング・サービスに登録されます。DataSource オブジェクトには、アプリケーションがアクセスする必要のある実際のデータ・ソースに関連するプロパティがあります。

### このタスクについて

要件 : DriverManager 機能は Java EE Connection Architecture Specification でサポートされていないため、DriverManager 機能に代わる DataSource 機能を使用する必要があります。

アプリケーションのデータ・ソースをインストールするには、以下のようになります。



## 手順

1. WebSphere Application Server for z/OS 管理コンソールの左方フレームで、「**Resources**」 > 「**Resource Adapters**」 > 「**Resource Adapters**」をクリックします。  
リソース・アダプターのリストが表示されます。
2. ユーザーがアダプターのインストール時に選択した IMS Universal Database リソース・アダプター の名前をクリックします。  
構成ダイアログが表示されます。
3. 「Additional Properties」で、「**J2C connection factories**」をクリックします。
4. 「**New**」をクリックします。  
構成ダイアログが表示されます。
5. 「General Properties」の下で、以下の情報を入力します。  
「**Name**」: 接続ファクトリーの名前 (例えば PhonebookCF)。  
「**JNDI Name**」: 当該サーバー内で固有の、接続ファクトリーの JNDI 名 (例えば PhonebookCF)。
6. 「**OK**」をクリックします。  
新規の接続ファクトリーが、管理可能なリソースのテーブルにリストされます。
7. インストールしたデータ・ソースの名前をクリックします。
8. 「Additional Properties」で、「**Custom Properties**」をクリックします。  
プロパティがテーブルにリストされます。
9. ご使用のシステムで以下の各プロパティに定義されている値を指定します。そのためには、プロパティ名をクリックし、開いた構成ペインの「**Value**」フィールドでプロパティの値を指定します。プロパティの値を指定したら、「**OK**」をクリックしてプロパティのリストに戻ります。

### **DatastoreName**

DRA 始動テーブル・モジュール名 (通常は IMS システム ID) の 4 バイト目から 7 バイト目を入力します。DRA 始動テーブルについての詳細は、[8 ページの『WebSphere Application Server for z/OS でのタイプ 2 IMS Universal Database リソース・アダプターのインストール』](#)を参照してください。

### **DriverType**

driverType を、以下のいずれかの値に設定します。

#### **2**

1 つの作業単位の有効範囲が特定の接続に限定される、ローカル・トランザクション・モデルを選択します。複数の接続が、それぞれの接続に関連付けられた独立した作業単位を持つことができます。

アプリケーション・プログラムは、JDBC 接続インターフェースまたは CCI LocalTransaction インターフェースを使用して、ローカルのコミット呼び出しおよびロールバック呼び出しを発行できます。

DriverType=2 は、UserTransaction インターフェースをサポートしません。

コンテナ管理の Bean メソッドは、EJB デプロイメント記述子内に以下の特性が必要です。

- 「**Bean**」タブの「**LocalTransaction**」見出しの下で、以下の特性を指定します。

- **Boundary:** BeanMethod
- **Resolver:** ContainerAtBoundary
- **Unresolved action:** Rollback

- 「**Assembly**」タブで、トランザクションの有効範囲を NotSupported に設定します。

#### **2\_CTX**

1 つの作業単位が複数の Bean メソッドにまたがることのできる、グローバル・スコープ・トランザクション・モデルを指定します。

アプリケーション・プログラムは、UserTransaction インターフェースを使用して、明示的なコミット呼び出しおよびロールバック呼び出しを行うことができます。

アプリケーション・プログラムは、JDBC 接続インターフェースまたは CCI LocalTransaction インターフェースを使用して、ローカルのコミット呼び出しおよびロールバック呼び出しを発行することはできません。

2\_CTX が指定されている場合は、EJB デプロイメント記述子のデフォルト・プロパティを使用してください。

#### DatabaseName

ターゲットの IMS データベースを表すデータベース・メタデータの場所を入力します。このプロパティは、以下のいずれかの方法で指定できます。

- ターゲット・データベースにアクセスするために使用される PSB の名前。このオプションは、IMS システムが IMS カタログを使用する場合にのみ選択可能です。
- IMS Enterprise Suite Explorer for Development によって生成された Java メタデータ・クラスの完全修飾名。URL には、class:// の接頭部を付ける必要があります (例えば、class://com.foo.BMP255DatabaseView)。

10. メッセージ・ボックスで、「**Save**」をクリックします。

「Save」ページが表示されます。

11. 「**Save**」をクリックして、マスター・リポジトリの変更を更新します。

12. サーバーを再始動します。

## WebSphere Application Server for z/OS でのタイプ 2 IMS Universal Database リソース・アダプターを使用する EAR ファイルのインストール

このトピックでは、WebSphere Application Server for z/OS にアプリケーションをデプロイする方法について説明します。

### 始める前に

前提条件: 10 ページの『[WebSphere Application Server for z/OS でのタイプ 4 IMS Universal Database リソース・アダプターに対する接続ファクトリーの定義](#)』で説明されているステップを実行します。

### このタスクについて

アプリケーションをインストールするには、以下のようにします。

### 手順

1. WebSphere Application Server for z/OS 管理コンソールの左側のペインで、「**Applications**」 > 「**New Application**」 > 「**New Enterprise Application**」をクリックします。
2. EAR ファイルのパスを入力するか、「**Browse**」 ボタンをクリックして EAR ファイルの場所を探します。「**Next**」をクリックします。
3. 「Preparing for the application installation」パネルで、デフォルトを受け入れて「**Next**」をクリックします。
4. 「Selection installation options」パネルで、デフォルトを受け入れて「**Next**」をクリックします。
5. 「Map modules to servers」パネルで、デフォルトを受け入れて「**Next**」をクリックします。
6. 「Summary」パネルで、オプションが適切であることを確認して「**Next**」をクリックします。アプリケーションがインストール中であることを示す一連のメッセージが表示されます。
7. アプリケーションが正常にインストールされたことを示すメッセージが表示されたら、「**Save**」をクリックします。

# IMS Universal ドライバー : WebSphere Application Server Liberty タイプ 4 接続

---

WebSphere Application Server Liberty 上で実行される Java アプリケーションは、IMS Universal Database リソース・アダプターが提供するタイプ 4 接続を使用して IMS データベースにアクセスできます。

## このタスクについて

IMS Universal ドライバーのタイプ 4 接続を使用することにより、Java アプリケーション・プログラムはさまざまな分散環境やメインフレーム環境から、スタンドアロン・モードまたはアプリケーション・サーバーの下で、グローバル・トランザクションに対する XA サポートの有無にかかわらず、IMS データベースにアクセスできます。

以下の図は、IMS Universal Database リソース・アダプターのタイプ 4 接続を使用して分散プラットフォーム用 WebSphere Application Server から IMS データベースに接続する EJB アプリケーションの概要を示しています。データベース要求は、タイプ 4 IMS Universal Database リソース・アダプターに渡され、TCP/IP を介して IMS Connect に送信され、ODBM によって内部管理されて IMS データベースにアクセスします。

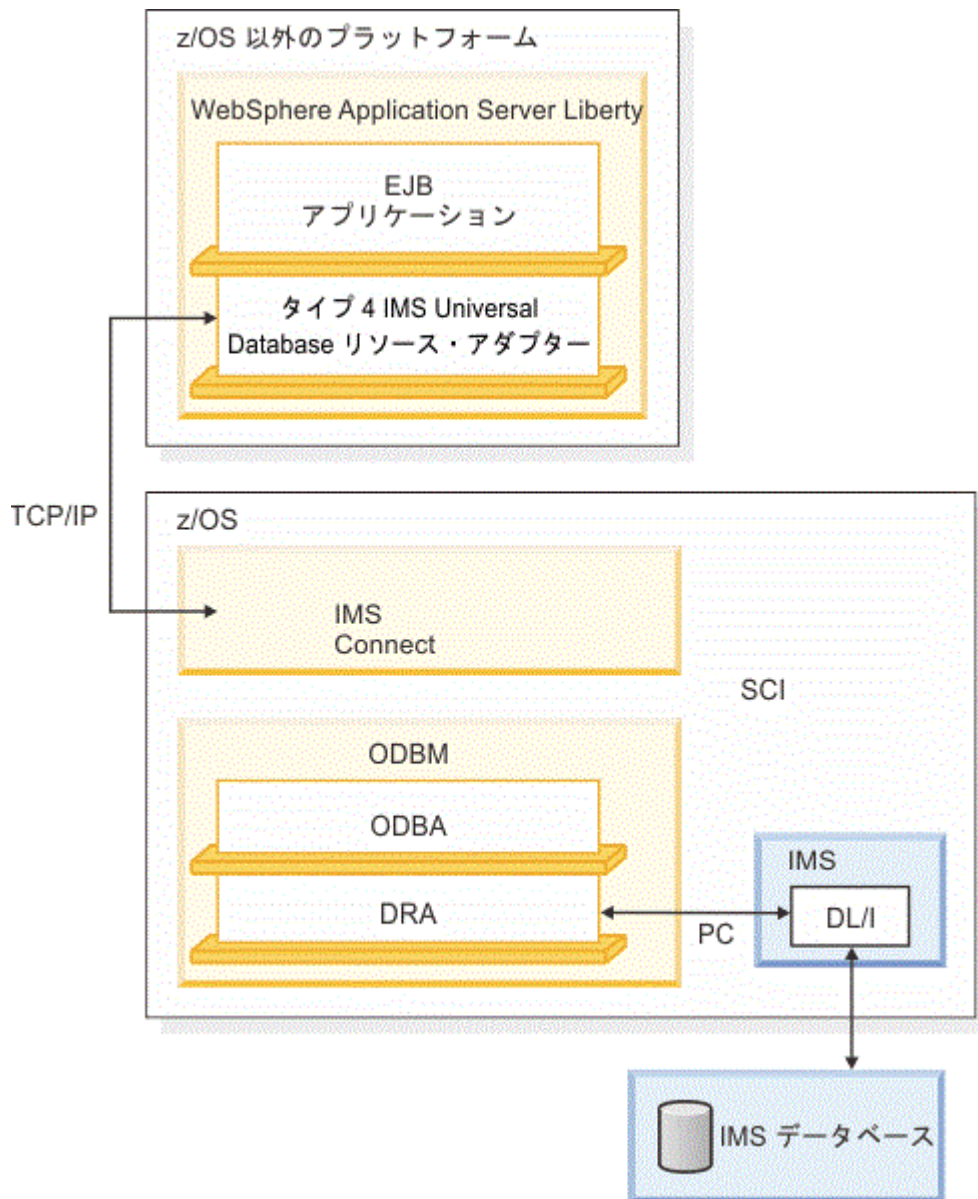


図 3. タイプ 4 IMS Universal Database リソース・アダプターを使用する WebSphere Application Server Liberty EJB アプリケーション

WebSphere Application Server Liberty 上で実行する Enterprise JavaBeans (EJB) アプリケーションを使用可能にするには、分散プラットフォームで、まず `server.xml` 構成ファイルを構成する必要があります。この構成ファイルを使用して、IMS Universal Database リソース・アダプターをインストールし、IMS™ データベースに接続する接続ファクトリーを定義して、WebSphere Application Server Liberty で RESTful サービス・アプリケーションをデプロイします。

#### フィーチャーの使用可能化

Liberty で JCA および JDBC サポートを有効にするには、`<featureManager>` エlement 内に、以下のサンプルにリストされている `<features>` タグを追加します。

```
<featureManager>
  <feature>jca-1.7</feature>
  <feature>jndi-1.0</feature>
  <feature>jdbc-4.1</feature>
  <feature>localConnector-1.0</feature>
</featureManager>
```

**IMS Universal Drivers Library** の位置を指定します。

<library> エlement内、IMS Universal ドライバー ライブラリーを指す <fileset> エlementを追加します。

```
<library id="global">
  <!-- Include imsudb.jar -->
  <fileset dir="usr/lpp/...imsjava/" includes="imsudb.jar"/>
</library>
```

### IMS Universal Database リソース・アダプターのインストール

リソース・アダプターをインストールするには、<resourceAdapter> エlementを追加し、その検索用の id プロパティとリソースの location プロパティを定義します。

```
<resourceAdapter id="imsudbJLocal"
  location="usr/lpp/...imsjava/rar/imsudbJLocal.rar"/>
<resourceAdapter id="imsudbLocal"
  location="usr/lpp/...imsjava/rar/imsudbLocal.rar"/>
<resourceAdapter id="imsudbJXA"
  location="usr/lpp/...imsjava/rar/imsudbJXA.rar"/>
<resourceAdapter id="imsudbXA"
  location="usr/lpp/...imsjava/rar/imsudbXA.rar"/>
```

### 接続ファクトリーの定義

接続ファクトリーをリソース・アダプターに関連付けるには、<connectionFactory> エlementを追加し、適切なリソース・アダプター id を指定して properties サブElementを定義し、該当のリソース・アダプターを選択します。

```
<!-- Associate a connection factory to a resource adapter
through the resource adapter's id. -->
<connectionFactory jndiName="HOSP_JDBC_T4">
  <properties.imsudbJLocal databaseName="MYPSB"
    datastoreName="IMS1" datastoreServer="myHostName"
    portNumber="1234" driverType="4"
    user="myUser" password="myPassword"/>
</connectionFactory>
<connectionFactory jndiName="HOSP_CCI_T4">
  <properties.imsudbLocal databaseName="MYPSB"
    datastoreName="IMS1" datastoreServer="myHostName"
    portNumber="1234" driverType="4"
    user="myUser" password="myPassword"/>
</connectionFactory>
<connectionFactory jndiName="HOSP_JDBC_T4XA">
  <properties.imsudbJXA databaseName="MYPSB"
    datastoreName="IMS1" datastoreServer="myHostName"
    portNumber="1234" driverType="4"
    user="myUser" password="myPassword"/>
</connectionFactory>
<connectionFactory jndiName="HOSP_CCI_T4XA">
  <properties.imsudbXA databaseName="MYPSB"
    datastoreName="IMS1" datastoreServer="myHostName"
    portNumber="1234" driverType="4"
    user="myUser" password="myPassword"/>
</connectionFactory>
```

### トレースとロギングの使用可能化

トレースとロギングは、デフォルトで使用不可になっています。IMS Universal ドライバーのトレースとロギングを使用可能にするには、以下のように、traceSpecification 属性にストリング引数「com.ibm.ims.db.opendb」を指定して、<logging> エlementを追加します。

```
<!-- The log files can be found at: [usr/lpp/.../servers/server_name/logs] -->
<logging traceSpecification="*=info:com.ibm.ims.db.opendb.*=finest"/>
```

## WebSphere Application Server Liberty タイプ 4 接続の server.xml 構成ファイル例

server.xml ファイルの例は、IMS Universal Database リソース・アダプター のインストール、IMS™ データベースに接続する接続ファクトリーの定義、および WebSphere Application Server Liberty での RESTful サービス・アプリケーションのデプロイのために使用されます。

## このタスクについて

```
<server description="new server">

  <featureManager>
    <feature>jca-1.7</feature>
    <feature>jndi-1.0</feature>
    <feature>jdbc-4.1</feature>
    <feature>localConnector-1.0</feature>
  </featureManager>

  <!-- To access this server from a remote client add a host
  attribute to the following element, e.g. host="*" -->
  <httpEndpoint host="*" httpPort="1692" httpsPort="9443"
    id="defaultHttpEndpoint"/>

  <!-- Automatically expand WAR files and EAR files -->
  <applicationManager autoExpand="true"/>

  <library id="global">
    <!-- Include imsudb.jar -->
    <fileset dir="usr/lpp/...imsjava/" includes="imsudb.jar"/>

    <!-- (OPTIONAL) Include jars that contain local database
    metadata (dbviews) -->
    <fileset dir="usr/lpp/.../" includes="dbviews.jar"/>
  </library>

  <!-- Defining resource adapters -->
  <resourceAdapter id="imsudbJLocal"
    location="usr/lpp/...imsjava/rar/imsudbJLocal.rar"/>
  <resourceAdapter id="imsudbLocal"
    location="usr/lpp/...imsjava/rar/imsudbLocal.rar"/>
  <resourceAdapter id="imsudbJXA"
    location="usr/lpp/...imsjava/rar/imsudbJXA.rar"/>
  <resourceAdapter id="imsudbXA"
    location="usr/lpp/...imsjava/rar/imsudbXA.rar"/>

  <!-- Defining connection factories -->
  <!-- Associate a connection factory to a resource adapter through
  the resource adapter's id. -->
  <connectionFactory jndiName="HOSP_JDBC_T4">
    <properties.imsudbJLocal databaseName="MYPSB"
      datastoreName="IMS1" datastoreServer="myHostName"
      portNumber="1234" driverType="4"
      user="myUser" password="myPassword"/>
  </connectionFactory>
  <connectionFactory jndiName="HOSP_CCI_T4">
    <properties.imsudbLocal databaseName="MYPSB"
      datastoreName="IMS1" datastoreServer="myHostName"
      portNumber="1234" driverType="4"
      user="myUser" password="myPassword"/>
  </connectionFactory>
  <connectionFactory jndiName="HOSP_JDBC_T4XA">
    <properties.imsudbJXA databaseName="MYPSB"
      datastoreName="IMS1" datastoreServer="myHostName"
      portNumber="1234" driverType="4"
      user="myUser" password="myPassword"/>
  </connectionFactory>
  <connectionFactory jndiName="HOSP_CCI_T4XA">
    <properties.imsudbXA databaseName="MYPSB"
      datastoreName="IMS1" datastoreServer="myHostName"
      portNumber="1234" driverType="4"
      user="myUser" password="myPassword"/>
  </connectionFactory>

  <!-- Enable or Disable (Default) JDBC trace -->
  <!-- The log files can be found at:
  [usr/lpp/.../servers/server_name/logs] -->
  <logging traceSpecification=
    "*=info:com.ibm.ims.db.opendb.*=finest"/>

  <webApplication id="myApp" location="myApp.war" name="myApp"/>
</server>
```

# IMS Universal ドライバー : WebSphere Application Server Liberty タイプ 2 接続

WebSphere Application Server Liberty と IMS が同じ論理区画 (LPAR) 上にある場合、WebSphere Application Server Liberty で実行されている Java アプリケーションは、IMS Universal ドライバー が提供するタイプ 2 接続を使用して IMS データベースにアクセスできます。

## このタスクについて

IMS Universal ドライバー のタイプ 2 接続では、Java アプリケーション・プログラムはローカルな、非 TCP/IP IMS データベースにアクセスできます。

以下の図は、IMS Universal Database リソース・アダプター のタイプ 2 接続を使用して WebSphere Application Server Liberty から IMS データベースにアクセスする EJB アプリケーションの概要を示しています。データベース要求は、タイプ 2 IMS Universal Database リソース・アダプター に渡され、DL/I 呼び出しに変換されて、IMS データベース・リソース・アダプター (DRA) インターフェースを使用して DL/I 領域にアクセスする ODBA に渡されます。

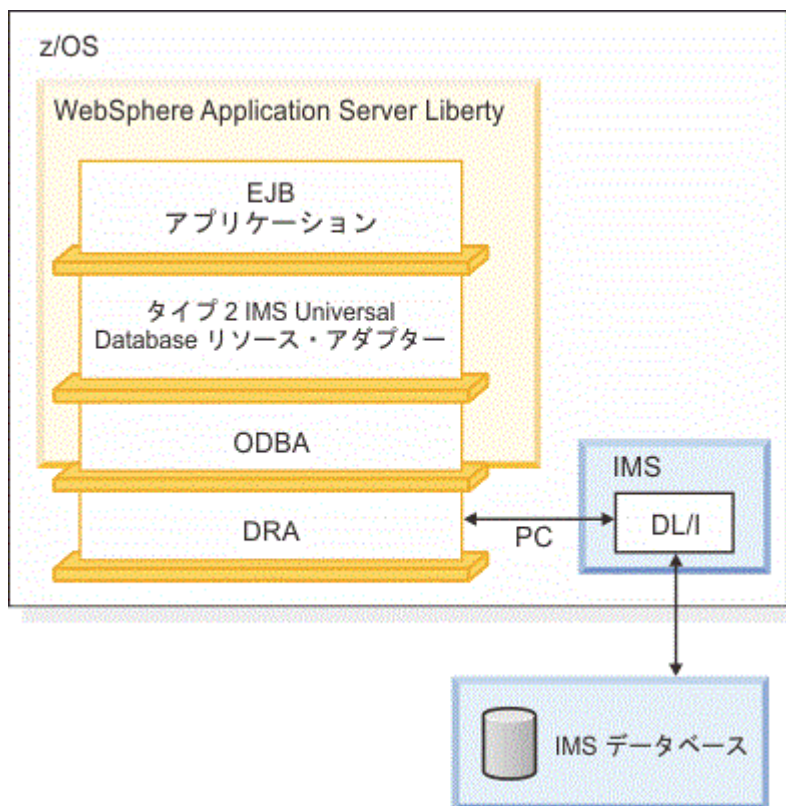


図 4. タイプ 2 IMS Universal Database リソース・アダプターを使用する WebSphere Application Server Liberty EJB アプリケーション

z/OS プラットフォーム上の WebSphere Application Server Liberty で実行する Enterprise JavaBeans (EJB) アプリケーションを使用可能にするためには、最初に server.xml 構成ファイルを構成する必要があります。これを使用して、IMS Universal Database リソース・アダプター のインストール、IMS™ データベースに接続する接続ファクトリーの定義、および WebSphere Application Server Liberty での RESTful サービス・アプリケーションのデプロイが行われます。

## フィーチャーの使用可能化



<featureManager> エlement内で、以下の例に示されている <features> を追加して、Liberty に対する JCA および JDBC のサポートを有効にします。RRS タイプ 2 接続には、zosTransaction-1.0 フィーチャーが必要です。

```
<featureManager>
  <feature>jca-1.7</feature>
  <feature>jndi-1.0</feature>
  <feature>jdbc-4.1</feature>
  <feature>localConnector-1.0</feature>

  <!-- Required for RRS Type-2 connectivity -->
  <feature>zosTransaction-1.0</feature>
</featureManager>
```

### IMS Universal ドライバー・ライブラリーの位置の指定

<library> エlement内で、IMS Universal ドライバー・ライブラリーを指す <fileset> サブElementを追加します。

```
<library id="global">
  <!-- Include imsudb.jar -->
  <fileset dir="usr/lpp/...imsjava/" includes="imsudb.jar"/>

  <!-- libT2DLI.so or libT2DLI_64.so native code required for Type-2 Connectivity -->
  <fileset dir="usr/lpp/..." includes="libT2DLI.so"/>
</library>
```

### ネイティブ・トランザクション・マネージャーの有効化

RRS タイプ 2 接続サポート用に以下の <nativeTransactionManager> Elementを追加します。

```
<resourceAdapter id="imsudbJLocal"
location="usr/lpp/...imsjava/rar/imsudbJLocal.rar"/>
<resourceAdapter id="imsudbLocal"
location="usr/lpp/...imsjava/rar/imsudbLocal.rar"/>
```

### 接続ファクトリーの定義

<connectionFactory> Elementを追加し、適切なリソース・アダプター id で properties サブElementを定義することで適切なリソース・アダプターを選択して、接続ファクトリーをリソース・アダプターに関連付けます。

```
<!-- Associate a connection factory to a resource adapter
through the resource adapter's id. -->
<connectionFactory jndiName="HOSP_JDBC_T2">
  <properties.imsudbJLocal databaseName="MYPSB"
  dataStoreName="IMS1" dataStoreServer="myHostName"
  driverType="2" user="myUser" password="myPassword"/>
</connectionFactory>
<connectionFactory jndiName="HOSP_CCI_T2">
  <properties.imsudbLocal databaseName="MYPSB"
  dataStoreName="IMS1" dataStoreServer="myHostName"
  driverType="2" user="myUser" password="myPassword"/>
</connectionFactory>
```

### トレース/ロギングの有効化

トレースおよびロギングは、デフォルトでは無効になっています。IMS Universal ドライバーのトレースおよびロギングを有効にするために、以下のようにストリング引数「com.ibm.ims.db.opendb」を持つ traceSpecification 属性が含まれている以下の <logging> Elementを追加します。

```
<!-- The log files can be found at: [usr/lpp/.../servers/server_name/logs] -->
<logging traceSpecification="*=info:com.ibm.ims.db.opendb.*=finest"/>
```



## WebSphere Application Server Liberty タイプ 2 接続の server.xml 構成ファイル例

server.xml ファイルの例は、IMS Universal Database リソース・アダプター のインストール、IMS™ データベースに接続する接続ファクトリーの定義、および WebSphere Application Server Liberty での RESTful サービス・アプリケーションのデプロイのために使用されます。

### このタスクについて

```
<server description="new server">
  <featureManager>
    <feature>jca-1.7</feature>
    <feature>jndi-1.0</feature>
    <feature>jdbc-4.1</feature>
    <feature>localConnector-1.0</feature>

    <!-- Required for RRS Type-2 connectivity -->
    <feature>zosTransaction-1.0</feature>
  </featureManager>

  <!-- To access this server from a remote client add a host
  attribute to the following element, e.g. host="*" -->
  <httpEndpoint host="*" httpPort="1692" httpsPort="9443"
  id="defaultHttpEndpoint"/>

  <!-- Automatically expand WAR files and EAR files -->
  <applicationManager autoExpand="true"/>

  <library id="global">
    <!-- Include imsudb.jar -->
    <fileset dir="usr/lpp/...imsjava/" includes="imsudb.jar"/>

    <!-- libT2DLI.so or libT2DLI_64.so native code required for
    Type-2 Connectivity -->
    <fileset dir="usr/lpp/.../" includes="libT2DLI.so"/>

    <!-- (OPTIONAL) Include jars that contain local database
    metadata (dbviews) -->
    <fileset dir="usr/lpp/.../" includes="dbviews.jar"/>
  </library>

  <nativeTransactionManager shutdownTimeout="5s"/>

  <!-- Defining resource adapters -->
  <resourceAdapter id="imsudbJLocal"
  location="usr/lpp/...imsjava/rar/imsudbJLocal.rar"/>
  <resourceAdapter id="imsudbLocal"
  location="usr/lpp/...imsjava/rar/imsudbLocal.rar"/>

  <!-- Defining connection factories -->
  <!-- Associate a connection factory to a resource adapter through
  the resource adapter's id. -->
  <connectionFactory jndiName="HOSP_JDBC_T2">
    <properties.imsudbJLocal databaseName="MYPSB"
    datastoreName="IMS1" datastoreServer="myHostName"
    portNumber="1234" driverType="2"
    user="myUser" password="myPassword"/>
  </connectionFactory>
  <connectionFactory jndiName="HOSP_CCI_T2">
    <properties.imsudbLocal databaseName="MYPSB"
    datastoreName="IMS1" datastoreServer="myHostName"
    portNumber="1234" driverType="2"
    user="myUser" password="myPassword"/>
  </connectionFactory>

  <!-- Enable or Disable (Default) JDBC trace -->
  <!-- The log files can be found at:
  [usr/lpp/.../servers/server_name/logs] -->
  <logging traceSpecification=
  "*=info:com.ibm.ims.db.opendb.*=finest"/>

  <webApplication id="myApp" location="myApp.war" name="myApp"/>
</server>
```

## IMS Universal ドライバー : CICS 接続

IBM CICS Transaction Server for z/OS 上で実行されている Java アプリケーションは、IMS Universal ドライバー が提供するタイプ 2 接続を使用して IMS データベースにアクセスできます。Java レイヤー以外の点では、Java アプリケーションから IMS へのアクセスは、Java 以外のアプリケーションの場合と同じです。

注：タイプ 2 接続は、IMS サービス・プロセスを介してサポートされます。

CICS は以下の IMS Universal ドライバー をサポートします。

- IMS Universal JDBC ドライバー
- IMS Universal DL/I ドライバー

以下の図は、IMS データベース・リソース・アダプター (DRA) インターフェースおよび IMS Universal ドライバー のタイプ 2 接続を使用して IMS データベースにアクセスする JCICS アプリケーションを示しています。

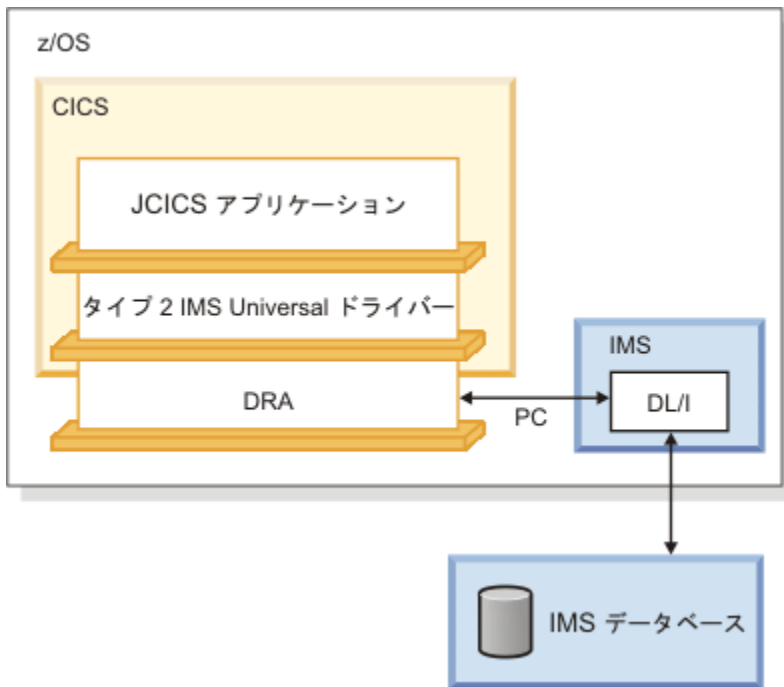


図 5. タイプ 2 IMS Universal ドライバー を使用する CICS アプリケーション

### タイプ 2 IMS Universal ドライバー 用の CICS の構成

CICS 環境でタイプ 2 IMS Universal ドライバー を介して IMS データベースにアクセスする Java アプリケーションを実行するには、IBM CICS Transaction Server for z/OS サブシステムにタイプ 2 IMS Universal ドライバー をインストールしておく必要があります。

#### 始める前に

**前提条件：**CICS サブシステムがアクセスできるパスに、タイプ 2 IMS Universal ドライバー のインストール・ファイルをロードします。

#### このタスクについて

タイプ 2 IMS Universal ドライバー 用に CICS を構成するには、以下のようにします。

## 手順

1. IMS Universal ドライバー OSGi バンドルを作成します。バンドルを作成するには、以下のようにします。

- a) テキスト・エディターでバンドル Manifest.mf ファイルを作成します。以下のサンプル・ファイルは、Manifest.mf の例です。

```
Manifest-Version: 1.0
Bundle-ManifestVersion: 2
Bundle-Name: IMS Universal driver OSGi
Bundle-SymbolicName: com.ibm.ims.osgi.Udb
Bundle-Version: 1.0.0
Bundle-ClassPath: imsudb.jar
Export-Package: com.ibm.ims.application,
com.ibm.ims.base,
com.ibm.ims.db,
com.ibm.ims.db.cci,
com.ibm.ims.db.hybrid,
com.ibm.ims.db.spi,
com.ibm.ims.dbd,
com.ibm.ims.dli,
com.ibm.ims.dli.conversion.util,
com.ibm.ims.dli.conversion.util.bidi,
com.ibm.ims.dli.converters,
com.ibm.ims.dli.dm,
com.ibm.ims.dli.logging,
com.ibm.ims.dli.t2,
com.ibm.ims.dli.tm,
com.ibm.ims.dli.types,
com.ibm.ims.dli.xa,
com.ibm.ims.drda.base,
com.ibm.ims.drda.converters,
com.ibm.ims.drda.db,
com.ibm.ims.drda.t4,
com.ibm.ims.drda.t4.util,
com.ibm.ims.drda.t4nativesql,
com.ibm.ims.jdbc,
com.ibm.ims.jdbc.batch,
com.ibm.ims.jdbc.xa,
com.ibm.ims.jms,
com.ibm.ims.opendb,
com.ibm.ims.psb,
com.ibm.ims.smf,
com.ibm.ims.xmldb,
com.ibm.ims.xmldb.dm,
com.ibm.ims.xmldb.shredder,
com.ibm.ims.xmldb.xms
Import-Package: com.ibm.cics.server;version="[1.300.0,2.0.0]";resolution:=optional
Bundle-RequiredExecutionEnvironment: JavaSE-1.7
```

- b) Manifest.mf ファイルと IMS Universal ドライバー imsudb.jar ファイルの両方を zip アーカイブに追加します。
- c) zip アーカイブの名前を com.ibm.ims.osgi.Udb\_1.0.0.jar に変更します。
- d) CICS エクスプローラーを使用して、作成した OSGi バンドルをデプロイします。
2. JVM プロファイルを含む CICS 環境の UNIX システム・サービス・ファイル (DFHJVMPR) を変更します。

- a) OSGI\_BUNDLES 変数を更新して、以下のように、作成した OSGi バンドルへのパスが含まれるようにします。

```
OSGI_BUNDLES=pathprefix/com.ibm.ims.osgi.Udb_1.0.0.jar
```

- b) LIBPATH 変数を更新して、libT2DLI\_64.so ファイルへのパスが含まれるようにします。これは、次のようにします。

```
LIBPATH_SUFFIX=pathprefix/usr/lpp/ims/ims15
/imsjava/lib
```

3. CICS AIBTDLI が IMS AIBTDLI インターフェースを介してロードされたことを確認します。

- a) CICS SDFHLOAD メンバーを CICS STEPLIB 内の IMS SDFSRESL メンバーの上位に設定します。

## タスクの結果

関連資料: CICS システム 定義について詳しくは、「*CICS Transaction Server for z/OS CICS システム 定義ガイド*」を参照してください。

## CICS でのタイプ 2 IMS Universal ドライバー を使用するアプリケーションの実行

IMS Universal ドライバー を介して IMS DB にアクセスする CICS アプリケーション・プログラムを実行するには、いくつかのステップを実行する必要があります。

### このタスクについて

CICS でタイプ 2 IMS Universal ドライバー を介して IMS DB にアクセスする Java アプリケーションを実行するには、以下のステップを実行します。

### 手順

1. IMS DB と CICS を始動します。
2. CE0T NOUCTRAN と入力して、CICS の大文字変換機能をオフにします。
3. Java アプリケーション (JVM クラス) を実行できるプログラムを定義します。
4. プログラムを実行できるトランザクションを定義します。
5. Java アプリケーション (JVM クラス) を実行するように定義したプログラムをインストールします。
6. Java アプリケーション (JVM クラス) を実行するように定義したトランザクションをインストールします。

---

## 第 2 部 CPI 通信および APPC/IMS

これらのトピックでは、CPI 通信と APPC/IMS を紹介します。このトピックでは、CPI 通信ドリブンのアプリケーション・プログラムの機能する方法、APPC/IMS を管理し、CPI アプリケーション・プログラムを構築するために CPI 通信インターフェース付きの APPC/IMS を使用する方法を説明します。



## 第 2 章 CPI 通信

このトピックでは、CPI 通信ドリブン・アプリケーション・プログラムおよび分散同期点保護会話を説明します。

### CPI-C ドリブン・アプリケーション・プログラム

CPI 通信ドリブン・アプリケーション・プログラムは、SQL 呼び出しを使用して IMS 外部サブシステム (ESS) 接続機能を介して Db2 for z/OS にアクセスする方法と、APSB 呼び出しを使用して IMS リソースを割り振る方法という 2 つの方法で、IMS が管理しているリソースを使用することができます。

SQL 呼び出しを使用して、IMS 外部サブシステム (ESS) 接続機能を介して Db2 for z/OS にアクセスする場合に、Db2 for z/OS リソース変換テーブル (RTT) が使用されないときは、初期 Db2 for z/OS 計画名はアプリケーション・プログラム名になります。APSB 呼び出し後では、Db2 for z/OS 計画名は APSB 呼び出しで指定された PSB 名です。

IMS リソース または Db2 for z/OS リソースへの変更を、アプリケーション・プログラムにコミットまたはバックアウトさせたい場合には、次の SAA リソース・リカバリー呼び出しを使用することができます。

- 変更をコミットするには、コミット呼び出し (SRRCMIT) を使用する。
- 変更をバックアウトするには、バックアウト呼び出し (SRRBACK) を使用する。

### SAA リソース・リカバリー・コミット処理

アプリケーション・プログラムは、SAA リソース・リカバリー呼び出しである SRRCMIT を発行することで、データベース・リソースへの変更をコミットすることを IMS に通知します。

アプリケーション・プログラムは SRRCMIT 呼び出しを出すことにより、データベース・リソースへの変更をコミットすることを IMS に知らせます。

- アプリケーション・プログラムが IMS リソースを更新するか、あるいは Db2 for z/OS リソースにアクセスする場合は、SRRCMIT 呼び出しを出してください。
- IMS リソースまたは Db2 for z/OS リソースへの以降の変更の後には、SRRCMIT 呼び出しを再度出してください。
- アプリケーション・プログラムを終了するには、事前に SRRCMIT 呼び出しを出してください。

リソースに変更をコミットしないで、アプリケーション・プログラムを終了する場合 IMS リソースに変更をコミットしないで、アプリケーション・プログラムを終了する場合、IMS はこれらの変更をコミットしようとしています。不完全な会話は正常に割り振り解除されません。

SRRCMIT 呼び出しを出すると IMS に制御が渡り、内部同期点呼び出しを生成します (会話が SYNCLVL=SYNCPT で割り振られていない場合)。すべてのデータベース変更がコミットされます。代替 PCB (プログラム制御ブロック) に挿入したメッセージは最終の宛先へすべて送信されます。

### 正常終了

IMS では正常終了は、アプリケーション・プログラムが異常なしで終了した場合に起こります。CPI 通信ドリブン・アプリケーション・プログラムの場合、暗黙コミットが行われます。

**定義:** 暗黙コミットは、アプリケーション・プログラムが現行トランザクションをコミットするための明示的な呼び出しを出さないが、以下のいずれかが発生したときに行われます。

- アプリケーション・プログラムが正常に終了した。

### バックアウト処理

トランザクションの更新は、さまざまな理由でバックアウトされる場合があります。

トランザクションの更新は以下のいずれかが起きた場合、バックアウトされます。

- バックアウトがアプリケーション・プログラムによって出された。
- アプリケーション・プログラムが異常終了した。
- アプリケーション・プログラムが正常に終了したが、暗黙コミットに失敗した。
- IMS システムの再始動中に、IMS リソースがフライト中 (未完了) である。

バックアウトには以下のようなアクションがあります。

- データベースの更新がすべてバックアウトされる。
- 非高速代替 PCB へ挿入されたメッセージがすべて廃棄される。
- キューになかった高速 PCB に挿入されたメッセージがすべて廃棄される。
- APPC/MVS™ verb ATBCMTP TYPE=ABEND が出される。<sup>1</sup>

SRRBACK を出す、または異常終了することにより、バックアウトが必要なことをアプリケーション・プログラムが IMS に伝えます。

## 異常終了

アプリケーション・プログラムが異常終了すると、IMS は最後の IMS 同期点までバックアウトします。

以下のどちらかが起こると、IMS はアプリケーション・プログラムが異常終了したと判断します。

- 暗黙コミットが失敗する。
- アプリケーション・プログラムが異常終了する。

## セッションの失敗

会話の最中に LU 6.2 セッションのどれかが失敗した場合、アプリケーション・プログラムを終了するか、プロセスを継続することを選択できます。

IMS TM は関与しないので、更新をコミットするか、バックアウトするかの選択を制限することはありません。アプリケーション・プログラマーがこの判断をします。

IMS TM はセッションの失敗が分からないため、処置を取りません。コミットとバックアウトの通常の処理規則を適用します。

## 戻りコード

SAA リソース・リカバリー SRRCMIT および SRRBACK 呼び出しでは、アプリケーション・プログラムは IMS から戻りコードを受け取ります。

アプリケーション・プログラムは以下の戻りコードを受信できます。

### RR\_OK

バックアウトまたはコミット操作が正常に完了しました。バックアウトされていれば、すべての保護されたリソースが直前の同期点へ戻されました。コミットされていれば、すべての保護されたリソースは新しい同期点へと進み、論理作業単位中に行われたすべての変更は永続的なものとされました。

### RR\_PROGRAM\_STATE\_CHECK

非 CPI 通信ドリブン IMS アプリケーション・プログラムが、SAA リソース・リカバリー・コミット呼び出しを発行しました。コミット、またはバックアウトが実行されませんでした。

### RR\_BACKED\_OUT

リソース・マネージャーは、同期点処理を行っている間に"いいえ"を選びました。同期点が SAA リソース・リカバリー・コミット呼び出しによって開始されました。リソースの状態はすべてのリソースでバックアウトされます。

<sup>1</sup> verb ATBCMTP を出すことによって、この TPI に関連したすべての LU 6.2 会話が CM\_DEALLOCATE\_ABEND で終了します。



## システム再始動/未確定解決処理

システム障害の後における、再始動処理の重要な部分は、未確定を解決する処理です。システムに障害が起きた場合、IMS は IMS 保護リソースに対して未確定解決処理を行うかどうかを判別します。

IMS 保護リソースの例には次のものがあります。

- IMS DB データベース
- Db2 for z/OS データベース
- IMS TM メッセージ・キュー・メッセージ

トランザクションが 2 フェーズ・コミット・プロセス (同期点) のフェーズ 1 を完了する前に、IMS システムが障害を生じると、IMS は、IMS の再始動の間にバックアウトします。障害発生時点で処理中であったトランザクションもバックアウトされます。

トランザクションがコミット・プロセスのフェーズ 1 を完了した場合、未確定解決処理は IMS 再始動の間に行われます。IMS リソースだけが該当した場合、コミット処理が起こります。Db2 for z/OS リソースが該当する場合、未確定解決処理は IMS と Db2 for z/OS の間で行われます。

明示的 CPI 通信ドリブン・アプリケーション・プログラムを使用するトランザクションは IMS の再始動に当たって保管されません。

## CPI-C アプリケーション・プログラムのリカバリー

明示の CPI 通信ドリブン・インターフェースを使用するアプリケーション・プログラムには、リカバリー・プロセスは存在しません。

IMS は、障害を起こした時の状態とは無関係に、再始動時に CPI 通信ドリブン・トランザクションをすべて廃棄します。アプリケーション・プログラム設計者は、SAA リソース・リカバリー再同期機能に注意を払って、アプリケーション・プログラム設計への影響を考慮する必要があります。

アプリケーション・プログラムは、セッション障害の場合は、SAA リソース・リカバリー・コミットあるいはバックアウト呼び出しを出すことによって完全な安全性を提供しなければなりません。リカバリー援助を必要とするアプリケーション・プログラムは標準 DL/I アプリケーション・プログラムでなければなりません。

### 関連資料

[CALL ステートメント \(アプリケーション・プログラミング API\)](#)

## プログラミング要件

暗黙の同期点を開始する呼び出し (メッセージ・キューへの DL/I GU、CHKP、および SYNC) は、CPI 通信ドリブン・アプリケーション・プログラムでは無効であり、状況 AD を受け取ります (無効な機能パラメーター)。CPI 通信ドリブン・アプリケーション・プログラムは、SRRCMIT 呼び出しおよび SRRBACK 呼び出しを出すことによって、IMS 同期点処理を活動化します。

SYNCLVL=NONE または SYNCLVL=CONFIRM を使用して会話を割り振る場合は、バインド・ステップでアプリケーション・プログラムにモジュール DFSCPIRO を含めてください。このモジュールを含めることにより、アプリケーション・プログラムは SRRCMIT と SRRBACK の外部参照を解決できるようになります。

IMS には SAA リソース・リカバリー・インターフェースのための言語固有のプログラミング考慮事項はありません。

## 仮名ファイル

APPC/IMS は APPC/MVS サービスを使用して、SAA リソース・リカバリー・サポートを提供します。APPC/MVS は、SAA リソース・リカバリーの仮名ファイルを提供しません。しかし、ユーザーは独自の仮名ファイルを作成することはできます。

**関連資料:** サンプル仮名ファイルについて詳しくは、「SAA CPI 資源回復解説書」を参照してください。このサンプル仮名ファイルには、異なる言語で作業用ストレージを定義する方法に関する例が含まれています。

要約すると、異なる言語のプログラマーは次のものを定義する必要があります。

- IMS TM C プログラマーは、オペレーティング・システムとして z/OS を定義する必要があります。
- IMS TM COBOL プログラマーは、バッファーを作業用ストレージに定義しなければなりません。
- IMS TM FORTRAN プログラマーは、SRRRCMIT および SRRBACK 用の EXTERNAL ステートメントを定義する必要があります。

## RRS および分散同期点/保護会話

SYNCLVL の設定が NONE、CONFIRM、または SYNCPOINT のいずれであるかに関係なく、RRS=Y の場合は、z/OS リソース・リカバリー・サービスが同期点マネージャーであり、複数の保護リソースの更新およびリカバリーを調整します。RRS は、RRS に登録されたリソース・マネージャー (IMS など) と調整しながら、保護リソースをいつ、どのようにコミットするかを制御します。

RRS は、リソース・リカバリー (CPI-RR) の共通プログラミング・インターフェースをサポートします。これは、SAA CPI のエレメントであり、リソース・リカバリーを指定し、ローカル・リソースと分散リソースのリカバリーを調整します。

### 定義:

- 保護リソースとは、同期化され、制御された方法で更新される、ローカル・データまたは分散データの集合です。APPC 環境では、保護リソースとは、SYNCLVL=SYNCPT を指定して割り振られた会話内で更新されたリソースのことです。
- リソース・マネージャーとは IMS のような製品であり、SYNCLVL=SYNCPT を指定した APPC 会話型環境内で更新される保護データ・リソースを所有します。IMS は、DL/I データ、高速機能データ、およびメッセージ・キューのためのリソース・マネージャーとして働きます。

リソース・リカバリーにおける 3 つの参加プログラムとして次のものがあります。

RRS (同期点マネージャー)

リソース・マネージャー (IMS または Db2 for z/OS など)

アプリケーション・プログラム

以下の図は、リソース・リカバリー・プロセスにおける 3 つの参加プログラムと、これらの参加プログラムの相互作用を示しています。

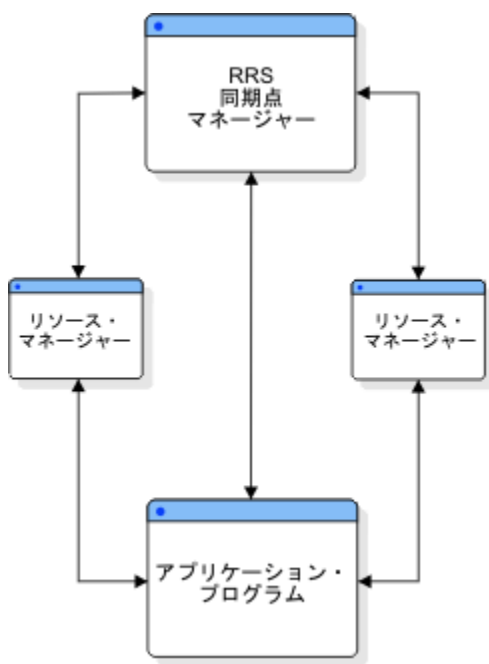


図 6. リソース・リカバリーへの参加プログラム

## 2 フェーズ・コミット・プロトコル

2 フェーズ・コミット・プロトコルとは、z/OS リソース・リカバリー・サービス (RRS) とリソース・マネージャーが関係して、リソースの集合に対してアプリケーション・プログラムが加えた更新をすべて行うか、あるいはまったく行わないようにするプロセスです。

アプリケーション・プログラムは自分がリソースに対して行った変更をコミットするかどうかを決定します。このコミットは RRS に対して行われ、RRS/MVS はこのコミット呼び出しの実現可能性を、すべてのリソース・マネージャーに問い合わせます。各リソース・マネージャーは、更新をコミットするかどうかの投票を行います。これは、2 フェーズ・コミット・プロトコルのフェーズ 1 と呼ばれます。

RRS が投票結果を収集した後、フェーズ 2 が始まります。すべての投票が更新をコミットしていれば、フェーズ 2 の処置はコミットすることであり、そうでない場合は、フェーズ 2 は更新をバックアウトすることになります。システム障害、通信障害、リソース・マネージャーの障害、またはアプリケーション・プログラム障害は、2 フェーズ・コミット・プロトコルの完了を妨害しません。

### 定義:

- リカバリー単位とは、1つのコミット(同期)点から次のコミット・ポイントにまたがる作業の単位です。
- 作成された時点(または直前の同期点)からリソース・マネージャーが更新のコミットに投票するまでの間、リカバリー単位は未完了であるとされます。リカバリー単位が未完了であるときにリソース・マネージャーに障害が起これば、リソース・マネージャーは後で開始された時に、すべてのデータベース更新をバックアウトします。
- リソース・マネージャーが更新をコミットすることに投票した時点から、同期点マネージャーがリソース・マネージャーを呼び出してコミットを行うまでの間、リカバリー単位は未確定であるとされます。リカバリー単位が未確定である時に IMS に障害が起これば、IMS はこれが解決されるまでデータベース更新を保留にします。

## ローカル・リソース・リカバリーと分散リソース・リカバリー

ローカル・リソース・リカバリー環境では、リカバリーへの参加プログラムは同じ z/OS システム上にあります。分散リソース・リカバリー環境では、リカバリーの参加プログラムと更新されたリソースは複数のシステムに散らばっています。

分散リソース・リカバリー環境では、リモート・システムへの同期点呼び出しのための通信は、APPC/PC (APPC/ 保護会話) リソース・マネージャーを使用して提供されます。

以下の図は、分散リカバリー環境がどのように作動するかを示しています。

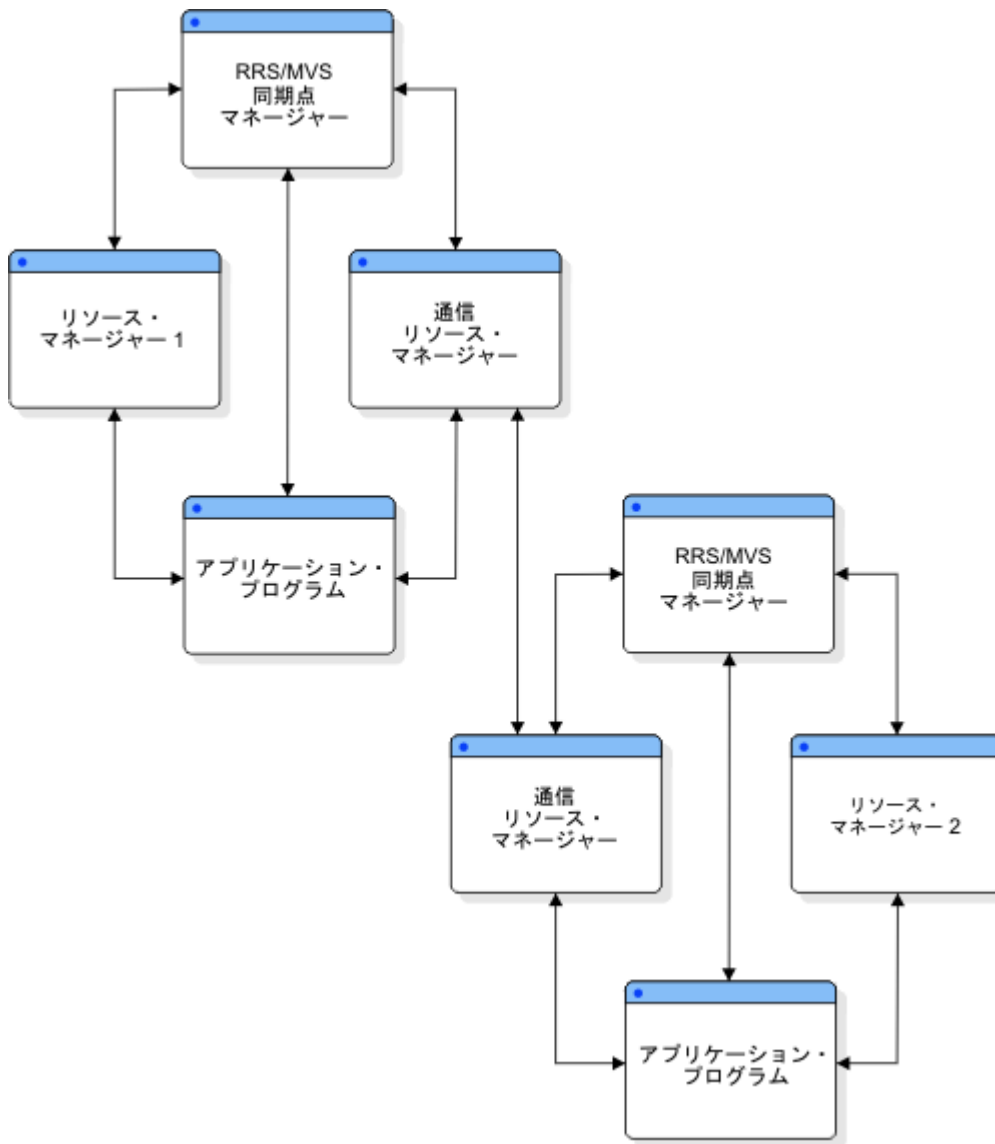


図 7. 分散リソース・リカバリ

## リソース・マネージャーとしての IMS

リソース・マネージャーは、保護リソースを制御します。

通常、リソース・マネージャーは以下のことを行います。

- アプリケーション・プログラムがそのリソースにアクセスできるようにするため、アプリケーション・プログラミング・インターフェース (API) を提供する。
- 変更を永続的なものにする前に、データへの変更をログに記録する。
- 作業単位の状況をログに記録する。
- 更新されたリソースのコミット処置またはバックアウト処置に参加する。
- データを直前の状態に復元するためのリカバリ・メカニズムを含む。

IMS 自身のリソースにとっては、IMS は同期点マネージャーであり、リソース・マネージャーでもありません。

2 フェーズ・コミット・プロトコルにおいて、IMS は次のとおりを行う必要があります。

- z/OS リソース・リカバリ・サービス (RRS) にリソース・マネージャーとして登録する
- 同期点プロセスに参加する

- 作業単位にインタレストを持つことを表明する
- 障害の後で、その作業単位状況を回復する

## 登録

RRS の 1 つのコンポーネントが登録サービスを提供し、IMS が自分をリソース・マネージャーとして登録できるようにしています。登録により、保護会話に関連付けられたリソースの整合性の維持を援助するための、一連のサービスが IMS に提供されます。

z/OS システムで DB/DC アクティブ・システム用に制御領域が開始されるたびに、IMS はリカバリー・プラットフォーム・サポートに対して登録を行います。XRF 環境では、アクティブ・システムは再始動中に登録します。代替システムはテークオーバーの時に登録を行います。

## インタレストの表明

登録および、2 フェーズ・コミット・プロトコルの特定のステージ用にリソース・マネージャー出口ルーチンを提供することに加えて、IMS は特別のリカバリー単位の場合、2 フェーズ・コミット・プロセスに参加する意志を表明する必要があります。

## 未完了のインタレストの解決

IMS 障害または z/OS 障害の場合は、IMS の再始動中に、未完了のインタレストの UR 表明を解決する必要があります。

RRS はリカバリー単位の情報 (例えば、ID、状態、 およびリソース・マネージャーの専用データ) を保守し、RRS はこれを、前にインタレストを表明した、再始動中のリソース・マネージャーに提示します。

## 同期点の参加

IMS が正常に登録し再始動すると、IMS はその出口ルーチンのアドレスを RRS に提供します。いくつかの出口ルーチン (例えば、準備、コミット、バックアウト) は、2 フェーズ・コミット・プロトコル内の特定の点を表し、これらの点は、IMS がこれら呼び出すことにより、プロセスの中に参加できる点です。

## 保護会話の活動化

z/OS は、コンテキストと呼ばれる構成物を z/OS リソース・リカバリー・サービス (RRS) で使用します。

**定義:** コンテキストとは、リソース・マネージャーがサービスを実行する対象のエンティティであり、そこにリソース・マネージャーがリソースとロック所有権を割り振り、また、そこにおいて、リソースが規則正しい方法で更新されるようにするためにプロトコルへの参加のインタレストをリソース・マネージャーが表明できるエンティティのことです。

リソース・マネージャーが作成、所有、および操作するコンテキストのタイプは、専用コンテキストと呼ばれます。リソース・マネージャーは他のリソース・マネージャーに代わってコンテキストを作成することができます。RRS は専用コンテキストを使用して、リソース・マネージャー用の情報を保守するアプリケーション・プログラムの作業単位を識別し、この情報はリソース・マネージャーのどのリソースが作業単位と関連付けられているかを示すものです。

## コミュニケーション・マネージャーとしての APPC

APPC がコミュニケーション・マネージャーである場合、RRS サポートは会話が SYNCLVL=SYNCPT で割り振られた時に活動化されます。この会話のタイプは、保護会話です。

SYNCLVL=SYNCPT が指定されていると、APPC は IMS に代わって専用コンテキストを取得します。IMS は、識別呼び出し内で、APPC にそのリソース・マネージャー名を提供します。APPC は専用コンテキストをメッセージ・ヘッダーとして IMS に提供します。すると IMS はこのコンテキストを使用して、同期点マネージャー RRS との 2 フェーズ・コミット・プロセスにおける、参加プログラムの役割を想定します。

保護会話用にどの LUS を使用可能にする必要がある場合にも対応できるように、SYNCLVL=SYNCPT に加えて、VTAM 定義ファイルでキーワード ATNLOSS=ALL も指定する必要があります。

## 保護会話での OTMA の使用

OTMA 環境では、OTMA は RRS に登録されたリソース・マネージャーではありません。プロセスは、サーバー IMS) と多くのクライアント (アプリケーション・プログラム) との間のプロセス間プロトコルのままです。したがって、OTMA は APPC が行うような、IMS に渡すための専用コンテキスト・トークンを取得することができません。OTMA を使用するクライアント・アダプター・コードに、専用コンテキストの取得、所有、およびコンテキスト ID の提供の責任があります。パートナー間で渡されるメッセージの中で、コンテキスト ID フィールドにはコンテキスト・トークンが含まれています (保護会話の場合)。

IMS がメッセージ内にコンテキスト ID を見つけると、IMS は APPC 環境で行うように、2 フェーズ・コミット処理の参加プログラムの役割を想定します。

## XRF と保護会話

IMS-XRF 環境で (RRS を APPC/PC または OTMA と一緒に使用して) 保護会話を実行することは、アクティブ・システムが開始した、終了していないすべての作業を、代替システムが再開し、解決することを保証するものではありません。失敗したリソース・マネージャーは、リソース・マネージャーが再始動する時に RRS システムがまだ使用できる場合は、その元の RRS システムに再登録する必要があります。アクティブ・システムの RRS が使用できない場合のみ、XRF 代替システムはシスプレックス内の別の RRS に登録でき、失敗したアクティブ・システムの未完了の回復単位データを入手できます。

**推奨事項:** IMS は、未確定のリカバリー単位が解決されるまでそれらを保存するため、できるだけ早くアクティブ・システムにスイッチバックしてリカバリー単位情報を入手し、リソース・マネージャーのすべての作業を解決して完了させてください。

## 第 3 章 APPC/IMS および LU 6.2 装置の管理

このトピックでは、APPC/IMS について紹介し、APPC/IMS および LU 6.2 装置の管理方法について説明します。

### APPC/IMS の概要

APPC/IMS は IMS TM の一部であり、CPI 通信インターフェースを使用して CPI アプリケーション・プログラムを作成することができるようにします。

以下の図に示すように、APPC/IMS を使用すると、APPC をインプリメントしたシステムと IMS の間で分散処理および連携処理を行えます。APPC/IMS は、APPC/MVS が提供する機能により、APPC をサポートします。(APPC/IMS インターフェースは APPC/MVS によって提供され、CPI 通信インターフェースをサポートします。IMS TM は、CPI リソース・リカバリー・インターフェースをサポートします。) APPC/IMS は、IMS が管理するローカル・リソースについて、CPI リソース・リカバリー・コミット (SRRCMIT) およびバックアウト (SRRBACK) の呼び出しをサポートします。これらの保護されるリソースには次のものがあります。

- IMS TM メッセージ・キュー・メッセージ
- IMS DB データベース
- Db2 for z/OS データベース

また、APPC/IMS は、CPI 通信インターフェースの機能なしでもアプリケーション・プログラムが LU 6.2 通信を使用できるようにする、既存の IMS DL/I アプリケーション・プログラミング・インターフェース (API) もサポートします。これにより、ほとんどの既存アプリケーションは、LU 6.2 の APPC/IMS サポートのもとで機能し続けることができます。

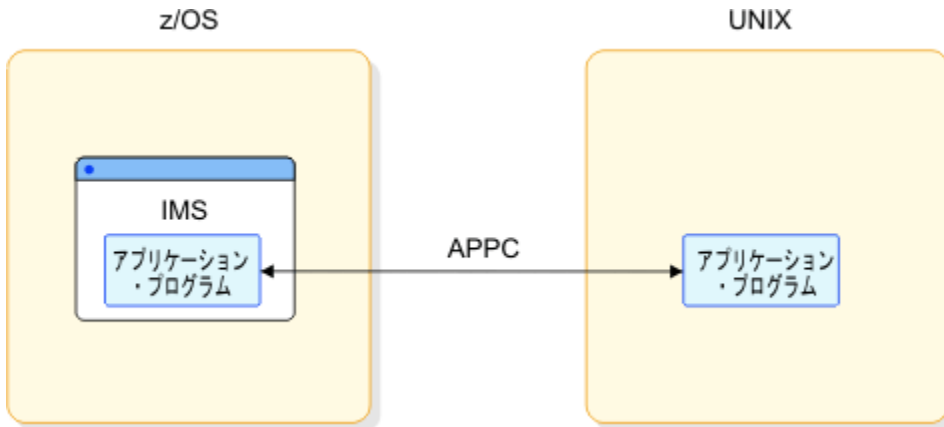


図 8. IMS に対する APPC サポート

#### 定義:

- IMS TM 管理のコンテキスト内では、「トランザクション・プログラム」、「アプリケーション」、「アプリケーション・プログラム」、および「プログラム」は同義語です。
- APPC/IMS および LU 6.2 装置の管理のコンテキスト内では、「APPC アプリケーション・プログラム」は、「LU 6.2 アプリケーション・プログラム」と同義語です。
- "LU 6.2 トランザクション"は、LU 6.2 アプリケーション・プログラムから発生するトランザクションです。

**推奨事項:** APPC/IMS の場合は、次のようにしてください。

- MSC を用いて、LU 6.2 リモート・システムから入力された IMS の標準または変更 アプリケーション・プログラムをスケジュールします。CPI-C ドリブン・アプリケーション・プログラムはリモート・システムで実行されるトランザクションをもつことができないことに注意してください。
- APPC アプリケーション・プログラムだけでなく APPC/MVS でも使用するよう APPC/IMS LU を定義します。



- メッセージの最初のセグメントで LTERM と MOD 名を使用します。LTERM は、非 LU 6.2 装置への出力の宛先を変更するために使用します。MOD 名は、エラー・メッセージをフォーマット設定するために使用します。
- ネットワーク修飾された LU 名を使用します。システムが異なっても、LU に固有の名前は不要です。

IMS 従属領域は、IMS スケジューラーの従属アドレス・スペースとして自動的に APPC に定義されます。IMS BMP は ASCH 制御アプリケーションとして定義することはできません。これは IMS の基本 LU を使用した明示会話サービスを使用できます。

IMS は APPC/IMS メッセージ・バッファを自動的に管理するので、定義は不要です。EMH には特別の考慮事項は必要ありません。

## APPC/IMS フラッディング制御

APPC/IMS フラッディング制御機能は、APPC/IMS トランザクション要求の数が突然増えたために IMS 31 ビット専用ストレージが使い尽くされないようにする上で役立ちます。

デフォルトでは、APPC/IMS フラッディング制御はアクティブであり、アクティブな APPC 会話の数が 5,000 に達した時点で 64 ビット・ストレージに対する着信 APPC トランザクション要求をキューに入れ始めます。フラッディング状態がさらに悪化する場合、APPC/IMS フラッディング制御は、64 ビット・ストレージ内のキューに入れられた APPC 要求の数が 10,000 というデフォルトのしきい値に達した時点ですべての APPC 入力を停止します。

すべての APPC 入力が定義された場合を除き、APPC/IMS フラッディング制御は、IMS コマンドのサブミットに使用される APPC 要求や、共用キュー環境のバックエンド IMS システム上で受け取る APPC 要求には適用されません。

アクティブでキューに入れられた APPC 要求の数がしきい値に近くなるにつれて、IMS は警告メッセージを出します。

### APPC/IMS フラッディング制御の変更または使用不可化

64 ビット・ストレージに対する要求をキューに入れるための初期しきい値と、IMS PROCLIB データ・セットの DFSDCxxx メンバーに APPC\_MAXC=(31\_bit\_max, 64\_bit\_max) パラメーターを指定することによってすべての APPC/IMS 入力を停止する 2 次しきい値の両方を変更または使用不可にすることができます。

31\_bit\_max 値は、IMS が 64 ビット・ストレージに対する新しい APPC トランザクション要求のキューイングを始める前に同時に処理できるアクティブな APPC 会話の最大数を定義します。0 を指定すると、APPC/IMS フラッディングは完全に使用不可になります。

IMS タイプ 1 コマンド **/DISPLAY A DC** を発行すると、現在の 31\_bit\_max 値を表示できます。表示される値が 0 の場合、APPC/IMS フラッディング制御は使用不可です。

64\_bit\_max は、IMS が z/OS からのすべての APPC 入力を停止する前に 64 ビット・ストレージでキューに入れることができる APPC トランザクション要求の最大数を定義します。0 を指定すると、64 ビット・ストレージでの APPC 要求のキューイングは使用不可になります。

IMS タイプ 1 コマンド **/PURGE APPC** を発行すると、64 ビット・ストレージのキューに入れられた APPC 要求をすべてクリアすることができます。APPC 会話は、TP\_Not\_Available\_No\_Retry センス・コードでリジェクトされます。

64 ビット・キューイングが使用不可の場合、フラッディング状態が発生すると、31 ビット最大数は、IMS がすべての APPC 入力を停止するしきい値を定義します。

### APPC 入力が停止された場合

IMS が z/OS からの APPC 入力を停止すると、IMS 自身は、着信 APPC 要求をリジェクトしませんが、APPC/MVS に対する呼び出しを発行して、それ以上の APPC 要求の送信を停止することを要求します。IMS が要求を発行してから APPC/MVS が入力の送信を停止するまでの間、IMS は APPC 要求を受け取ることができるため、APPC/IMS が受け取るか、またはキューに入れる APPC 要求の総数は、定義済みの最大数を超えることがあります。



APPC 入力 が 停止 した 後、IMS 31 ビット・ストレージ での アクティブ APPC 会 話 の 数 が 31\_bit\_max 値 の 50% を 下 回 る と、IMS は 自 動 的 に、APPC/MVS から の APPC 入 力 の 再 開 を 要 求 し ます。

APPC 入 力 が 停 止 さ れ る と、IMS は DFS4157E を 発 行 し ます。

## APPC/IMS フラッディング制御に代わるものとしての VTAM

APPC/IMS フラッディング制御の手段に加えて、あるいはそれに代わるものとして、VTAM ACB 内の個々の論理装置 (LU) のセッション限度を指定することができます。VTAM は、セッション限度に達した後、APPC/MVS へのメッセージの送信を停止します。IMS システムに対して LU が 1 つだけ定義されている場合、アクティブ APPC 要求の最大数は、VTAM で定義されているセッションの数です。

### 関連資料

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

## APPC/IMS アプリケーション・プログラム・インターフェース

APPC/IMS には、2 つの明確に異なるアプリケーション・プログラム・インターフェース (API)、すなわち暗黙インターフェースと明示インターフェースがあります。同じアプリケーション・プログラムが両方の API を使用することができます。

### 暗黙 API

暗黙 API は IMS 標準 DL/I API (xxxTDLI の呼び出し) の拡張です。これにより、IMS アプリケーション・プログラムは、LU 6.2 プロトコルに依存せずに、またプログラマーが LU 6.2 の知識を持っていなくても、LU 6.2 アプリケーション・プログラムと通信することができます。APPC/IMS は、LU 6.2 アプリケーション・プログラムが通常では使用できない機能、すなわちメッセージ・キューイングと非同期メッセージの自動送達およびリカバリーを提供します。既存の IMS トランザクションは、暗黙 API を使用して、APPC と通信します。

暗黙 API メッセージは、IMS メッセージ・キューか、あるいは高速機能トランザクションの高速機能急送メッセージ処理 (EMH) バッファに入れられます。入力メッセージに廃棄可能または廃棄不能のいずれのマークを付けるかは、発信元 IMS が決定します。

暗黙 API が使用されると、IMS は必要となるすべての CPI 通信呼び出しを出します。アプリケーション・プログラムは IMS メッセージ・キューまたは高速機能 EMH バッファと正確に対話をします。

### 明示 API

明示 API は CPI 通信 API であり、任意の IMS アプリケーション・プログラムが使用できます。アプリケーション・プログラムは、IMS を使用せずに、CPI 通信インターフェースを介して APPC を呼び出します。これらの CPI 呼び出しは、APPC/MVS によって直接処理されます。CPI 通信インターフェースで送信あるいは受信したメッセージは、IMS メッセージ・キューにも EMH バッファにも保管されないため、トランザクション再始動には使用できません。これらのメッセージには、IMS 提供機能は使用されません。

代わりに、z/OS で APPC/MVS TP サービスの ATBxxxx 呼び出しを使用することもできます。これらの呼び出しについては、「z/OS MVS プログラミング: APPC/MVS トランザクション・プログラムの書き方」を参照してください。

## APPC/IMS アプリケーション・プログラム

APPC/IMS には、標準、変更、および CPI 通信ドリブンという 3 つの異なるタイプのアプリケーション・プログラムがあります。

アプリケーション・プログラムは、以下のように定義されます。

#### 標準

CPI 通信機能を明示的に使用しない。

#### 変更

入出力 PCB を使用して元の入力端末と通信する。CPI 通信呼び出しを使用して、新規会話の割り振りおよびデータの送受信を行う。

## CPI 通信ドリブン

CPI 通信呼び出しを使用して、入ってくるメッセージを受け取り、同じ会話で応答を送る。DL/I APSPB 呼び出しを使用して、PSB を割り振り、IMS データベースおよび代替 PCB にアクセスする。

MSC を用いて、ローカルおよびリモートで標準および変更アプリケーション・プログラムをスケジュールすることができます。ローカル・アプリケーション・プログラム用の論理とリモート・アプリケーション・プログラム用の論理は異なります。以下のトピックで、これらの相違点について説明します。

## 標準 IMS アプリケーション・プログラム

標準 IMS アプリケーション・プログラムは、既存の IMS 呼び出しインターフェースを使用します。

IMS 標準 API を使用するアプリケーション・プログラムは、LU 6.2 プロトコルを利用することができます。標準 IMS アプリケーション・プログラムは、DL/I GU 呼び出しを使用して、同期点を起動し、着信トランザクションを入手します。これらの標準 IMS アプリケーション・プログラムは、また DL/I ISRT 呼び出しを使用して、同じ端末または異なる端末 (LU 6.2 端末も可能) に出力メッセージを生成します。(非メッセージ・ドリブン BMP は、明示 API を使用しない場合、標準 IMS アプリケーション・プログラムと見なされます。) LU 6.2 と非 LU 6.2 の両方の端末タイプに対して、同一のプログラムが正常に機能することができます。IMS は、APPC/MVS サービスに対する適切な呼び出しを生成します。

IMS は、標準 IMS アプリケーション・プログラムに以下のようなサービスを提供します。

- LU 6.2 アプリケーション・プログラムからの着信トランザクションを受信する。
- 入力メッセージ経路指定出口ルーチン呼び出し。
- ローカルおよびリモート IMS 従属領域へのトランザクションをスケジュールする。
- 必要なトランザクションをリカバリー可能にする。
- 必要なトランザクションをロールバックして再試行する。
- すべての APPC Sync\_Level オプション (NONE、CONFIRM、SYNCPT) の同期点で、IMS が制御している、データベース更新を伴う会話のフローを統合する。
- APPC/MVS サービスに対する必要なすべての LU 6.2 呼び出しを提供する。
- LU 6.2 アプリケーション・プログラムに同期か非同期かどちらかの出力を送信する。
- 伝送が成功するまで、IMS メッセージ・キューに非同期出力を保持する。
- DL/I ISRT 呼び出しを使用して代替 PCB に挿入されたメッセージに新規 LU 6.2 会話を割り振る。

既存のアプリケーション・プログラムで、カーソル位置や MFS 形式名のような端末のハードウェア特性に依存するものは、LU 6.2 アプリケーション・プログラムと通信するためには変更する必要があります。

### 制約事項:

1. LU 6.2 同期会話の暗黙トランザクションが他のトランザクションを初期化する場合 (プログラム間通信)、高速 PCB は ISRT には使用できません。高速 PCB は競争条件を引き起こし、TPNAME DFSASYNCR を使用する新しい非同期会話上で、出力は入力端末にランダムに返されることとなります。元の会話は割り振り解除されません。
2. トランザクションが複数の子トランザクションを初期化し、その子トランザクションが今度は別のトランザクションを初期化し、子トランザクションの 1 つが応答を提供すると、結果は予測不能です。

これらのトランザクションの実行順序により、LU はデフォルトの TP 名 DFSASYNCR に送られた応答と一緒に DFS2082 メッセージを受け取るか、または LU は応答を受け取り DFS2082 メッセージは出されません。

## MSC および標準 IMS アプリケーション・プログラム

APPC アプリケーション・プログラムがリモート IMS で実行される IMS トランザクションを入力すると、その APPC アプリケーション・プログラムとローカル IMS との間に LU 6.2 会話が確立されます。

ローカル IMS は、LU 6.2 会話のパートナー LU と見なされます。そうすると、トランザクションは、ローカル IMS のリモート・トランザクション・キューに入れられます。この時点から、トランザクションは通常の MSC 処理に入ります。リモート IMS がトランザクションを実行した後、出力がローカル IMS に戻され、その後、発信元 LU 6.2 アプリケーション・プログラムに送られます。

発信元 (ローカル) IMS は次のサービスを提供します。

- LU 6.2 アプリケーション・プログラムからの着信トランザクションを受信する。
- 入力メッセージ経路指定出口ルーチン呼び出す。
- トランザクションをそのリモート・トランザクション・キューに入れる。
- MSC リンクを介してトランザクションを送信する。
- トランザクション応答を受信する。
- LU 6.2 アプリケーション・プログラムに同期か非同期かどちらかの出力を送信する。

リモート IMS は、リモートの標準アプリケーション・プログラムのために次のサービスを提供します。

- MSC リンクを介してパートナー IMS (発信元または中間 IMS) から着信トランザクションを受信する。
- 従属領域へのトランザクションをスケジュールする。
- 同期点でデータベース変更をコミットする。
- 必要なトランザクションをリカバリー可能にする。
- 必要なトランザクションをロールバックして再試行する。
- 伝送が成功するまで、トランザクション出力を IMS メッセージ・キューに保持する。
- MSC リンクを介してトランザクション出力をローカル IMS に返す。

**制約事項:** MSC は、発信元の LU 6.2 会話が SYNCLVL=SYNCPT を使用して割り振られている場合はサポートされません。

## 修正済みの IMS アプリケーション・プログラム

変更 IMS アプリケーション・プログラムは、DL/I GU 呼び出しを使用して、入ってくるトランザクションを入手し、同期点を起動します。

これらの変更 IMS アプリケーション・プログラムはまた、DL/I ISRT 呼び出しを使用して、同じ端末または異なる端末 (LU 6.2 端末またはそれ以外) への出力メッセージを生成します。<sup>2</sup> 標準 IMS アプリケーション・プログラムと違って、変更された IMS アプリケーション・プログラムは CPI 通信呼び出しを使用して、新しい会話の割り振りとデータの送受信を行います。IMS はこれらの CPI 通信会話を直接制御しません。

修正済みの IMS トランザクションは、プログラムの実行まで、標準 IMS トランザクションと区別できません。実際に、同じアプリケーション・プログラムが、ある実行では"標準 IMS"アプリケーションであり、別の実行では"変更 IMS"アプリケーションであることができます。その区別は簡単で、アプリケーション・プログラムが CPI 通信リソースを使用しているかどうかです。

IMS は変更 IMS アプリケーション・プログラムに以下のようなサービスを提供します。

- LU 6.2 アプリケーション・プログラムからの着信トランザクションを受信する。
- ローカルおよびリモート従属 IMS 領域へのトランザクションをスケジュールする。
- トランザクションがスケジュールされる前にリカバリー可能にする。
- APPC Sync\_Level オプション (NONE、CONFIRM、SYNCPT) の同期点で、IMS が制御している、データベース更新を伴う会話のフローを統合する。
- IMS 制御 LU 6.2 会話のための APPC/MVS サービスに対する必要なすべての LU 6.2 呼び出しを提供する。
- LU 6.2 アプリケーション・プログラムに同期か非同期かどちらかの出力を送信する。
- 送信が成功するまで、非同期出力を IMS メッセージ・キューに保持する。
- DL/I ISRT 呼び出しを使用して代替 PCB に挿入されたすべてのメッセージに新規 LU 6.2 会話を割り振る。

<sup>2</sup> 非メッセージ駆動型の BMP は、明示 API を使用している場合は変更された標準 IMS アプリケーション・プログラムと見なされます。

IMS は、アプリケーション・プログラムで明示的に割り振られた会話にはサービスを何も提供しません。明示的に割り振られた会話は、プログラム異常終了が起こった場合には割り振りを解除する必要があります。

## MSC および変更 IMS アプリケーション・プログラム

APPC プログラムがリモート IMS で実行される IMS トランザクションを入力すると、APPC プログラムとローカル IMS 間に LU 6.2 会話が確立されます。

ローカル IMS は、LU 6.2 会話のパートナー LU と見なされます。そうすると、トランザクションは、ローカル IMS のリモート・トランザクション・キューに入れられます。この時点から、トランザクションは通常の MSC 処理に入ります。リモート IMS がトランザクションを実行した後、出力がローカル IMS に戻され、その後、発信元 LU 6.2 プログラムに送られます。

発信元 (ローカル) IMS は次のサービスを提供します。

- LU 6.2 アプリケーション・プログラムからの着信トランザクションを受信する。
- 入力メッセージ経路指定出口ルーチン呼び出す。
- トランザクションをそのリモート・トランザクション・キューに入れる。
- MSC リンクを介してトランザクションを送信する。
- トランザクション応答を受信する。
- LU 6.2 アプリケーション・プログラムに同期か非同期かどちらかの出力を送信する。

リモート IMS は、リモートの変更アプリケーション・プログラムのために次のサービスを提供します。

- MSC リンクを介してパートナー IMS (発信元または中間システム) から入ってくるトランザクションを受信する。
- 従属領域へのトランザクションをスケジュールする。
- トランザクションがスケジューリングされる前にリカバリー可能にする。
- 同期点でデータベース変更をコミットする。
- 必要なトランザクションをリカバリー可能にする。
- 必要なトランザクションをロールバックして再試行する。
- 伝送が成功するまで、トランザクション出力を IMS メッセージ・キューに保持する。
- MSC リンクを介してトランザクション出力をローカル IMS に返す。

**制約事項:** MSC は、発信元の LU 6.2 会話が SYNCLVL=SYNCPT を使用して割り振られている場合はサポートされません。

## CPI 通信ドリブン・アプリケーション・プログラム

CPI 通信ドリブン・アプリケーション・プログラムは、APPC/MVS の TP\_Profile データ・セットだけに定義され、IMS に対しては定義されません。

CPI 通信ドリブン・アプリケーション・プログラム定義は、IMS 再始動後に APPC/MVS TP\_Profile 定義に基づいて、トランザクションが APPC/MVS によるスケジューリング用に提示されたとき、IMS によって動的に作成されます。この定義のキーとして使用されるのは TP 名です。APPC/MVS は TP\_Profile 情報を管理します。

CPI 通信ドリブン・トランザクション・プログラムが PSB を要求する場合は、システム定義中に APPLCTN マクロを使用し、また APPLCTN PSB= 指定時には、プログラム仕様ブロック (PSB) 生成ユーティリティーおよびアプリケーション制御ブロック保守ユーティリティーを使用して適切な PSB および ACB を生成することで、PSB を IMS に事前に定義しておく必要があります。APPLCTN GPSB= 指定時には、PSB および ACB を生成する必要はありません。

CPI 通信ドリブン・アプリケーション・プログラムは、CPI 通信呼び出しを使用して、着信する会話を受け入れ、同じ会話で応答を送信しなければなりません。DL/I GU 呼び出しは、LU 6.2 アプリケーション・プログラムから開始トランザクションを取得するためには使用されません。アプリケーション・プログラ

ムがスケジュールされても、IMSのリソースは割り振られません。その代わりに、アプリケーション・プログラムは、DL/I APSB 呼び出しを使用して、IMS データベースと代替 PCB (複数の場合もある) へのアクセスを提供する PSB を割り振ることができます。CPI 通信ドリブン・アプリケーション・プログラムは、該当の PSB を割り振りした後、代替 PCB に挿入することにより、他の端末 (LU 6.2 または非 LU 6.2) または他の IMS トランザクション (ローカルまたはリモート) にメッセージを送信することができます。明示と暗黙の両 API は同じアプリケーション・プログラム上で使用できます。

IMS は、CPI 通信ドリブン・アプリケーション・プログラムに以下のサービスを提供します。

- トランザクションのスケジュール

IMS はスケジューリング前には入力を受信しません。インバウンド割り振り要求を拒否する可能性がある時以外、IMS は会話にかかわりません。IMS がインバウンド割り振り要求を拒否すると、トランザクションはスケジュールされません。

- ローカル・リソースの同期点を提供する。
- アプリケーション・プログラムによって呼び出された場合に PSB をスケジュールする。
- アプリケーション・プログラムによって行われた、代替またはデータベース PCB への呼び出しを処理する。

#### 関連概念

##### 28 ページの『RRS および分散同期点/保護会話』

SYNCLVL の設定が NONE、CONFIRM、または SYNCPOINT のいずれであるかに関係なく、RRS=Y の場合は、z/OS リソース・リカバリー・サービスが同期点マネージャーであり、複数の保護リソースの更新およびリカバリーを調整します。RRS は、RRS に登録されたリソース・マネージャー (IMS など) と調整しながら、保護リソースをいつ、どのようにコミットするかを制御します。

##### 25 ページの『CPI 通信』

このトピックでは、CPI 通信ドリブン・アプリケーション・プログラムおよび分散同期点保護会話を説明します。

##### APPC のアプリケーションの設計 (アプリケーション・プログラミング)

## MOD 名および LTERM インターフェースの使用

LU 6.2 アプリケーション・プログラムは、インターフェースを使用して MFS をエミュレートすることができます。

### このタスクについて

例えば、アプリケーション・プログラムは MOD 名を使用して、エラー・メッセージのフォーマット設定方法を IMS に伝えることができます。LU 6.2 以外のアプリケーション・プログラムの場合には、IMS の MFS フォーマット設定モジュールに MOD 名が与えられ、LU 6.2 アプリケーション・プログラムの場合には、MFS モジュールが呼び出されずに、MOD 名がパラメーターとして LU 6.2 編集出力ルーチン (DFSLUEEO) に渡されます。LU 6.2 編集出力ルーチンは、プログラマーが MOD 名を使用して指定すること (エラー・メッセージのフォーマット設定など) は何でも行うことができます。

LU 6.2 アプリケーション・プログラムは LU 名を使用して、LU 6.2 アプリケーション・プログラムにデータを送信します。ただし、プリンターなどの非 LU 6.2 装置にデータを送信したい場合には、LU 名でなく、LTERM を使用することができます。

初期設定出力ルーチン (DFSINTX0) を使用して、メッセージのフォーマット設定に使用する可能性のある MOD 名、およびプリンターとして使用する可能性のある LTERM のユーザー・テーブルを作成することができます。このユーザー・テーブルは DFSLUEEO によって使用され、ユーザーのアプリケーション・プログラムに該当する MOD 名または LTERM を見つけることができます。

LU 6.2 アプリケーション・プログラムは、LTERM と MOD 名の両方をメッセージの最初のセグメントで送信することができます。LU 6.2 編集出力ルーチン (DFSLUEEO) が、最初のメッセージ・セグメントの内容を検査します。この出力ルーチンは、ユーザー・テーブルで見つけた情報に基づいて、LTERM と MOD 名を IMS に戻すかどうかを決定します。IMS は、LTERM および MOD 名を入出力 PCB に保管します。出力をフォーマット設定する場合には、IMS が、メッセージの最初のセグメントにある MOD 名のアドレスを LU 6.2 編集出力ルーチン (DFSLUEEO) に渡します。宛先を非 LU 6.2 装置へ変更する場合には、IMS が、メッ

ページの最初のセグメントにある LTERM を LU 6.2 編集出口ルーチン (DFSLUEEO) に渡します。初期設定出口ルーチン (DFSINTX0) を使用して、ユーザー・テーブルを作成することができます。この出口ルーチンは、ユーザー・テーブルのアドレスを IMS に渡す必要があります。IMS はこのアドレスを DFSLUEEO に渡します。

## APPC/IMS の設定

APPC/IMS を活動化する前に、IMS システム定義で、IMSCTRL マクロの SYSTEM キーワードの 3 番目のパラメーターとして 390 を指定する必要があります。

### このタスクについて

CPI 通信ドリブン・アプリケーション・プログラムと LU 6.2 アプリケーション・プログラムは、システム定義では定義できません。LU 6.2 アプリケーション・プログラムは VTAM に対してのみ定義されます。

APPC/IMS は、IMS 始動パラメーターに APPC=Y を指定することによって開始します。デフォルトは APPC=N です。「N」を指定すると、APPC/MVS サービスへの接続は IMS の初期設定時には設定されません。「Y」を指定すると、IMS の初期設定時に、IMS は APPC/MVS との接続を確立します。/START APPC コマンドは APPC=N を上書きします。

## TP\_Profile

TP\_Profile は APPC/MVS が所有している VSAM データ・セットであり、APPC/MVS 管理ユーティリティー (ATBSDFMU) により保守されるか、または TSO/ISPF ダイアログを使用して管理者により保守されます。TP\_Profile 項目の目的は、TP 名に属性情報を提供することです。

### このタスクについて

CPI 通信ドリブン・アプリケーション・プログラムは、APPC/MVS TP\_Profile に定義しなければなりません。IMS システムで定義されたトランザクション・コードは、オプションとして TP\_Profile に定義することができます。IMS 定義が (IMS GEN に、あるいはオンライン変更によって) 存在すると、トランザクションは標準 DL/I か、あるいは変更された標準アプリケーションであると見なされます。

TP\_Profile (APPC/MVS リソース) には、トランザクション・プログラム名 (TPN) とそれらの特性の定義があります。TP\_Profile を定義することによって、TPN とは異なるトランザクション・コードを使用する IMS トランザクション・プログラムをスケジュールすることができます。

IMS は、CPI 通信ドリブン・アプリケーション・プログラムにトランザクション・スケジューリング特性を設定するために TP\_Profile を使用します。APPC/MVS によって再始動した後に、トランザクションがスケジューリングのために示されると、IMS はこれらの特性を APPC/MVS TP\_Profile 定義の IMS 従属セクションに基づいて動的に定義します。

CPI コミュニケーション駆動トランザクション・プログラムは、TP\_Profile 内でのみ定義されます。定義は TPN によって行われます。同じ TPN は、異なる TP\_Profile データ・セットを使用して異なる LU 名を別に定義できます。LU 名は、IMS と関連しています。

デフォルト TP\_Profile データ・セット名は SYS1.APPCTP です。SYS1.PARMLIB (APPCPMxx) メンバーにある LUADD TPDATA オプションはこの LU に使用する TP\_Profile データ・セット名を指定します。

TP\_Profile を定義するには、TP\_Profile ダイアログ機能または、APPC/MVS 管理ユーティリティー (ATBSDFMU) を使用してください。

**例:** 以下の図は、TP\_Profile 定義 (パネル 1) の IMS 特定域の例です。



```

----- IMS TP_Profile Panel -----
TP Name . . . : INQUIRY_Part

Transaction Code . . . . . PART
Security Type . . . . . _____ (NONE,CHECK,FULL, default=CHECK)

CPI Communications Driven Options
  Transaction Class . . . ___ (Range 1 - 999, default=1)
  Maximum Regions . . . ___ (Range 0 - 999, default=1)

Comments . . . (Optional 1 to 10 lines)
> TP_PROFILE Created 10/8/91 _____ <
> Access IMS Sample Parts DATABASE via. program DFSSAM02_____ <
> _____ <
> _____ <
> _____ <
> _____ <
> _____ <
> _____ <
> _____ <
> _____ <
> _____ <
> _____ <
> _____ <
> _____ <
> _____ <

PF01 = Help      PF03 = Exit      PF12 = Cancel      Enter = Accept

```

図 9. IMS 特定の TP\_Profile パネル 1

ISPF を使用して IMS TP\_Profile を保守するには、以下のことを実行してください。

## 手順

1. TSO ユーザー ID から ISPF TP\_Profile システム・データ機能保守ユーティリティを開始するには、TSO/E の TSO コマンド行から TSO ICQASRM0 を入力してください。このユーティリティが利用不能な場合、z/OS システム・プログラマーに相談してください。
2. TP\_Profile 選択に **S** と IMS LU に対する LUADD ステートメントの TPDATA キーワードに指定した TP\_Profile データ・セット名を入力してください。(LUADD ステートメントは APPCTPxx PARMLIB メンバーにあります。XX は APPC の接尾部です。)
3. TP\_Profile のリストが表示されます。**A** を選択して新規 TP\_Profile か、**E** を選択して既存の TP\_Profile を編集してください。TP\_Profile を追加している場合、スケジューラー名を提供する必要があります。この名前は、IMS インストール時に設定されたものです。推奨する名前は IMS です。
4. 一般的な TP\_Profile 特性が提供された後、ISPF エディター・パネルが表示されます。IMS TP\_Profile 保守パネルを表示するには、コマンド行で DFSTPROF を入力してください。
5. IMS スケジューラーに関する特性を提供します。変更を保管するには実行キーを、変更を取り消すには PF3 か PF12 を押してください。PF1 を押して、このパネルで表示されたフィールドのオンライン・ヘルプを見ることもできます。

## タスクの結果

TP\_Profile 名は TSO/E のリリースすべてで利用できるわけではないので、"利用不能" の値が表示されることがあります。これは、問題ではありません。

**関連資料:** このユーティリティについて詳しくは、「z/OS MVS プログラミング: APPC/MVS トランザクション・プログラムの書き方」を参照してください。

## APPC/MVS 管理ユーティリティ (ATBSDFMU) の例

以下の例は、APPC/MVS 管理ユーティリティ (ATBSDFMU) 項目です。

```

TPADD TPSCHED_EXIT(DFSTPPE0)
      TPNAME(INQUIRY_PART)
      SYSTEM
      ACTIVE(YES)
      TPSCHED_DELIMITER(##)
      TRANCODE=PART
      CLASS=1
      MAXRGN=1

```

この例で、IMS セクションは TRancode=PART から始まります。他の制御ステートメントは完全にするために示されています。

IMS TP\_Profile 構文解析モジュール DFSTPPE0 は、IMS 内の入力データの妥当性検査と構文解析を行います。このモジュールは、TP\_Profile を追加するステップの STEPLIB データ・セットにロードする必要があります。APPC/MVS 管理ユーティリティ (ATBSDFMU) を使用するには、STEPLIB を APF 許可する必要があります。

以下の 5 つのキーワードは、IMS セクションを TP\_Profile 項目に追加するために使用されます。キーワード・パラメーターの集合は、1 つ以上の空白で分離する必要があります。キーワード・パラメーターの集合は、桁 1-72 の間に指定する必要があります。桁 1 のアスタリスクは、コメントを意味しています。

#### TRancode= 1 文字から 8 文字

英数字、'#'、'\$'、'@' を含むこの TP 名に関連する IMS トランザクション・コード名。IMS は TP 名を TRancode に変換します。IMS は有効文字 (00640 文字セット) かどうかをスキャンします。無効な文字があると、IMS は、非 00640 文字を持つ トランザクション・コードの代わりに、デフォルト・トランザクション・コード IMSTRAN を使用します。

#### CLASS= 1 - 999

スケジューリングに使用されるクラスを指定します。デフォルトは 1 です。

**推奨:** 非 CPI トランザクションに使用されるメッセージ・クラスとは別のメッセージ・クラスを使用して、CPI トランザクションを定義してください。IMS は、トランザクション・クラス内ではすべての CPI トランザクションを優先順位ゼロとして扱います。

#### MAXRGN= 0 - 999

この CPI 通信ドリブン・トランザクション・プログラムが使用できる従属領域の数を制限します。デフォルト値は 1 です。

#### RACF®=NONE、CHECK、あるいは FULL

RACF=NONE とすると、IMS は トランザクション許可出口ルーチン (DFSCTRNO) を呼び出します。

RACF=CHECK と設定すると、IMS は、トランザクションを受信した際に、セキュリティ検査用に RACF を呼び出します (TIMS または CIMS の RCLASS を使用)。ただし、トランザクションの実行時に、セキュリティ環境を従属領域に複製することはいけません。

RACF=FULL は実行時に従属領域へセキュリティ環境を複製します。このパラメーターを指定し、IMS コマンド /SECURE APPC PROFILE を出すと、この CPI-C アプリケーション・プログラムに APSP SAF セキュリティが使用可能になります。

#### CPUTIME= 0 - 1440

CPI-C プログラムの使用が許可される CPU 秒数を指定します。制限を超えた場合、ABENDU0240 によって終了されます。この時間制限は、その他のアプリケーションからのリソースをロックする プログラム・ループに対するプロテクトです。デフォルトは 0 で、これは制限なしを意味します。

2 つの方法で TP\_Profile 項目を使用することができます。

- IMS で定義された IMS トランザクション・コードを指定するために使用。TP\_Profile 内の CLASS と MAXRGN パラメーターは無視され、IMS の トランザクション値は変更されません。TP\_Profile 項目は、64 文字 TPN と 8 文字の トランザクション・コードとの対応付けを提供します。
- IMS で定義されていない IMS トランザクション・コードを指定するために使用。IMS トランザクション・コードは CPI 通信ドリブン・トランザクションです。この トランザクション・コードは、スケジューリングされたアプリケーション・プログラムのロード・モジュール名および動的に作成された トランザクション名として使用されます。

TP\_Profile が定義されないと、IMS は IMS 文字セットに変換された TPN の最初の 8 バイトを トランザクション・コードとして使用します。

トランザクション・コードが無効な場合、割り振り要求は拒否されます。

**関連資料:** APPC/MVS 管理ユーティリティ (ATBSDFMU) の使用方法については、「z/OS MVS 計画: APPC/MVS 管理」を参照してください。



## アウトバウンド LU 指定

アウトバウンド LU と同様 APPC LU を指定することができます。

### このタスクについて

定義済み APPC LU のデフォルト設定はベース LU です。アウトバウンド LU が使用不可の状態の場合、IMS は APPC 会話を割り振らないので、アウトバウンド LU を変更することは有効です。

アウトバウンド LU は、SYS1.PARMLIB ライブラリーの APPCPMxx メンバー内で定義する必要があります。LU をアウトバウンド LU として指定するには、DFSDCxxx PROCLIB メンバー内の OUTBND= パラメーターを使用します。/CHANGE APPC OUTBND コマンドを使用して、アウトバウンド LU を設定することができます。ただし、指定した場合、再始動がアウトバウンド LU を DFSDCxxx メンバー内の値に設定します。指定しなかった場合、そのアウトバウンド LU はベース LU に設定されます。

## アウトバウンド・サイド情報

APPC/MVS サイド情報は、パートナー・プログラムの名前、パートナーのノードでの LU の名前、ログオン・モードの名前といった宛先情報を提供します。

CPI 通信は、これらの必須フィールドにシステム定義値を使用する方法を提供します。これらのシステム定義値はサイド情報と呼ばれます。この情報は、CPI 通信、IMS LU 6.2 記述子、DL/I 変更呼び出し (CHNG)、または DFSAPPC メッセージ通信を使用して APPC 会話を割り振る (確立する) IMS アプリケーション・プログラムによって使用することができます。

システム・プログラマーが、CPI 通信プログラムのサイド情報の提供と保守を行います。

サイド情報はシンボリック宛先名によってアクセスされます。IMS TM 管理のコンテキスト内で `sym_dest_name` と呼ばれるシンボリック宛先名は、以下の情報を含むサイド情報ファイルの項目に相当します。

### パートナー LU 名

パートナー・プログラムがある LU の名前を表示します。この LU 名は、会話を割り振るローカル LU によって認識されるリモート LU のどれかの名前です。例えば USERVAR 名です。

この LU 名は、17 バイトのネットワーク修飾 LU 名とすることができます。

### ログオン・モード名

会話を割り振るセッションの特性を指定するため LU 6.2 によって使用されます。特性とは、例えば、会話で使用されるサービスのクラスなどです。ネットワーク管理者は、ローカル LU で使用されるモード名の集合を定義して、パートナーとセッションを設定してください。システム・プログラマーは、サイド・テーブル項目でこれらの値の 1 つを使用します。モード名が無効な場合、会話の割り振りは行われません。

### TP 名

トランザクション・プログラム (TP) 名は、リモート・アプリケーション・プログラムの名前を指定します。

IMS および z/OS は、Initialize\_Conversation 呼び出しでブランクの `sym_dest_name` 値を受け入れません。

サイド情報ファイルのデフォルト名は SYS1.APPCSI です。このファイルは、以下の例に示すように SYS1.PARMLIB (APPCPMxx) に定義してください。

```
SIDEINFO
  DATASET (SYS1.APPCSI)
```

以下の例に示すように、宛先名、パートナー LU 名、モード名、および TP 名は、APPC/MVS 管理ユーティリティ (ATBSDFMU) を使用して定義できます。

```
SIADD
  DESTNAME (DESTX)
  TPNAME (LU62USER_TPX)
  MODENAME (APPCMODE)
  PARTNER_LU (APPCCLUX)
```

**関連資料:** APPC 呼び出しについては、「*CPI Communications Specification*」を参照してください。

## PARMLIB メンバー

APPC アドレス・スペースは SYS1.PARMLIB の APPCPMxx メンバーを使用します。IMS を、APPC アドレス・スペースによって制御されるローカル APPC コンポーネント LU として定義してください。

スケジューラー名は、IMSCTRL マクロに使用されているのと同じ IMSID です。IMS が APPC を識別すると、APPC メンバー APPCPMxxx のスケジューラー名 SCHED (IMS1) としてその IMSID を渡します。以下は APPCPMxx メンバーの例です。

```
LUADD
  ACBNAME (IMSLU62)
  SCHED (IMS1)
  BASE
  TPDATA (SYS1.APPCTP)
  TPLEVEL (SYSTEM)
```

XRF の追加の場合

```
USERVAR=uservar_name ALTLU=luname
```

LUADD オプション・キーワードは以下のように定義されます。

**ACBNAME=IMS** のローカル **LUNAME**

**SCHED=IMS id**

**BASE** 必須パラメーター

**TPDATA(TP\_Profile** データ・セット名)

**TPLEVEL(システム)** 推奨値

**USERVAR=(uservar\_name)**

**ALTLU=(LUNAME)**

**関連資料:** これらのキーワードについては、「z/OS MVS 計画: APPC/MVS 管理」を参照してください。

VTAMM とそのアプリケーション・プログラムとの間の通信には、ACB (アプリケーション制御ブロック) が必要であり、この ACB 名は、SYS1.VTAMLST APPL ステートメントおよび、APPCPMxx LUADD ステートメントの ACBNAME パラメーターと同じ名前に定義する必要があります。

APPC は IMS ACB を管理します。IMS が APPC を識別したら、APPC は IMS に APPC が管理している ACB 名 (LUNAME) の名前を割り振ります。IMS は ACB を管理しないので、APPC LUNAME は IMS には定義されません。SYS1.PARMLIB メンバー APPCPMxx の項目には、IMS スケジューラー名 (IMSID) と、IMS を LUNAME に関連づける LUNAME ACBNAME (xxxxxxx) の両方が含まれます。

この ACBNAME は、非 LU 6.2 端末用として IMS で使用される ACBNAME と異なっていなければなりません。APPC/MVS は LU が SNA ネットワークをアクセスできるように、LU が VTAM リソースに定義されていることを想定しています。VTAM アプリケーション・プログラム (APPL) 定義マクロは、各 APPC/MVS LU ごとに指定する必要があります。LU 6.2 アプリケーション・プログラムは VTAM に対してのみ定義され、IMS に対しては定義されません。SYS1.VTAMLST メンバーの例は以下のとおりです。

```
IMSLU62 APPL ACBNAME=IMSLU62
          APPC=Y
          ...
```

## APPC/MVS タイムアウト・サービス

APPC/MVS タイムアウト・サービスを使用すれば、アプリケーションが会話を終了して、APPC/MVS 呼び出し可能サービスから制御を取り戻す前に、アプリケーションが待つことになる最大時間間隔を指定することができます。

APPC/MVS が (例えば、ネットワーク遅延が原因で) APPC 呼び出しに応答しないと、従属領域がハングし、呼び出し元は制御を取り戻せません。

タイムアウト機能は、IMS.PROCLIB データ・セットの DFSDCxxx メンバーで APPCIOT=(*mmmm:ss,mmmm*) パラメーターを指定することにより、始動時にアクティブになります。APPC タイムアウト値は、分 (*mmmm*) と秒 (*ss*) で指定します。*mmmm* に有効な値は 00 から 1440 までです。*ss* に有効な値は 00 から 59 までです。APPCIOT=00 の場合、タイムアウトの検出は行われません。トランザクションがタイムアウトのために終了すると、メッセージ DFS1965E および DFS1959E が MTO 端末と z/OS コンソールに送信されます。タイムアウト値は、**/CHANGE** コマンドを使用して変更することができます。

同期 APPC 会話の場合は、APPC タイムアウトがアクティブであると、IMS は ATBSTO6 サービス (SET\_TIMEOUT\_VALUE) を使用して、会話ごとにタイムアウト値を設定します。

非同期 APPC 会話の場合は、APPC タイムアウトがアクティブであると、IMS は、会話の割り振り時にタイムアウト値を設定します。いずれの場合も、会話が割り振り解除されるまで、タイムアウト値はアクティブのままです。会話が割り振り解除されるのは、IMS 会話型トランザクションの場合は、IMS 会話が終了したときです。

共通プログラミング・インターフェース・コミュニケーション (CPI-C) トランザクションは、APPC/MVS タイムアウト・サービスによって自動的にサポートされるわけではありませんが、適切にコーディングすれば、ATBSTO5 サービスを用いて APPC/MVS タイムアウト・サービスを利用することができます。

### 関連資料:

- MVS サービスのプログラミングについて詳しくは、「z/OS MVS プログラミング: APPC/MVS トランザクション・プログラムの書き方」を参照してください。

### 関連資料

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

## APPC/MVS エラー抽出サービス

APPC/MVS サービス呼び出しが予期しない戻りコードを戻すたびに、IMS は DFS1995E という接頭部を持つ APPC/MVS エラー抽出サービス呼び出し ATBEES3 を出します。

**関連資料:** ATB 戻りコードについて詳しくは、以下を参照してください。

- z/OS MVS システム・メッセージ 第 3 巻 (ASB-BPX)
- z/OS MVS ダンプ出力メッセージ

## LU 6.2 記述子の初期設定および変更

LU 6.2 記述子により、システム・プログラマーは、出力の宛先と LU 6.2 アプリケーション・プログラムとを関連づける LTERM を指定することができます。また、システム・プログラマーは、アプリケーション・プログラムのコーディング変更を行わなくても、代替 PCB を使用して、アプリケーション・プログラムの宛先を LU 6.2 アプリケーション・プログラムに変更することができます。

### このタスクについて

LU 6.2 記述子はオプションですが、キュー制御ブロックを動的に作成したり、処理オプションを定義する場合には必要です。

アプリケーション・プログラムはシンボリック宛先として LTERM 名を使用します。システム・プログラマーはこの名前に関連した実際の用語を認識しておく必要があります。

LU 6.2 記述子項目には次のものが入っています。

- APPC/MVS サイド情報項目名 (このパラメーターは省略可能です)
- APPC 会話タイプ (BASIC または MAPPED)
- APPC Sync\_Level オプション (NONE、CONFIRM)
- LTERM 名
- LU 6.2 アプリケーション・プログラムの宛先の LU 名 (サイド情報を上書きします)。これは 17 バイトの長さまでのネットワーク修飾 LU 名にできます。
- OUTBND パラメーターで指定された、アウトバウンド会話を割り振るために LU IMS が使用するローカル LU の名前。
- スケジュールされる TP 名 (サイド情報を上書きします)
- 使用される VTAM モード・テーブル項目 (サイド情報を上書きします)

以下に、LU 6.2 記述子の例を示します。

```
U L62TERM1 LUNAME=L62IMS1 TPNAME=CPICTRN1 MODE=L62MDE02
U L62TERM1 SYNCLEVEL=N OUTBND=MYLU02
```

パラメーターを指定せず、**SIDE=b** のようにブランクのままにしておいてください。そうしないと、エラー・メッセージが出されます。あるいは、パラメーターを完全に省略します。

これらの LU 6.2 記述子 LTERM は出力専用であり、入力メッセージと関連した LTERM 名としては IMS では決して使用されません。DFSAPPC はメッセージ通信機能のために IMS で予約されている名前です。

LU 6.2 記述子は、IMS PROCLIB メンバー DFS62DTx 内で指定されたとおりに IMS 初期設定時に構築されます。記述子は、IMS を再始動することなく、追加、削除、または変更することができます。記述子はいくつでも指定できます。エラーが起きた場合、z/OS システム・コンソールと IMS JOBLIB はエラー・メッセージを記録します。IMS 初期設定は、記述子の初期設定を行っている間に起こるどのようなエラーも無視して継続されます。

IMS の実行中に記述子を追加するには、最初に、LU 6.2 記述子を PROCLIB メンバー DFS62DTx 内で定義する必要があります。/START DESC コマンドを使用して、LU 6.2 記述子を IMS PROCLIB からロードしてください。記述子を削除するには、/DELETE DESC コマンドを使用してください。記述子を変更するには、/CHANGE DESC コマンドを使用してください。

**関連資料:** これらのパラメーターのコーディングについては、「IMS V15 システム定義」を参照してください。

## APPC/IMS 環境における MSC の使用

APPC/IMS は、APPC/MVS および MSC のサービスを使用して、MSC 構成のための通信インターフェースを提供します。

MSC と APPC/IMS は一緒に、以下のことを可能にします。

- LU 6.2 プログラムが、IMS リモート標準アプリケーション・プログラムまたは IMS リモート変更アプリケーション・プログラムの TP 名を使用する。(トランザクションはリモート IMS に送信されて実行されます。トランザクションの応答は、MSC リンクを介してローカル IMS に送られ、次いで LU 6.2 アプリケーション・プログラムに送られます。)
- DFSAPPC システム・サービスを通じてリモート論理端末 (LTERM) とメッセージ通信を行う。
- DFSAPPC を使用して、IMS リモート・トランザクションおよびデータを送信する。
- MSC 経路指定のリモート・アプリケーション・プログラムへの、即時の、または遅らせたプログラム間通信を行う。

CPI 通信ドリブン・アプリケーション・プログラムには、リモート IMS システムで実行されるトランザクションを含めることはできません。

高速機能を除くすべての IMS トランザクション・タイプ、すなわち会話型、非会話型、応答モード、および非応答モードがサポートされます。

IMS は、LU 6.2 メッセージが MSC リンクを介して送信されるとき、そのメッセージに接頭部を追加します。この接頭部の最小サイズは 480 バイトです。MSC リンクに定義されるバッファ・サイズは、完全な 1 つのメッセージを保持するのに十分な容量が必要です。有効な MSC バッファ・サイズは、1024 バイトから 65536 バイトです。

メッセージを受信してから処理するまでの間に、入力メッセージ宛先を IMS ローカルまたはリモート宛先に変更するには、TM および MSC メッセージ経路指定と制御ユーザー出口 (DFSMSCEO) を使用します。

**定義:** MSC を APPC/IMS と一緒に使用するには、さまざまな MSC システムで使用される用語を理解する必要があります。

- 発信元システム (ローカル) は、LU 6.2 プログラムが IMS トランザクションを入力するシステムです。
- リモート・システム は、リモート・トランザクションが実行されるシステムです。
- 中間システム は、ローカル・システムとリモート・システムとの間でメッセージを経路指定する IMS です。

これらの 3 つのシステムのどれもが LU 6.2 トランザクションをいつでも受信することができます。

### 関連概念

683 ページの『複数システム結合機能の概要』

複数システム結合機能 (MSC) は、ある IMS に入力されたトランザクションを別の IMS で処理することを可能にします。

## MSC 環境における APPC トランザクションの回復

IMS LU 6.2 トランザクションのリカバリー可能性は、メッセージがリカバリー可能か、リカバリー不能か、廃棄可能か、廃棄不能か、またいつエラーが起こったかにより異なります。

### このタスクについて

MSC 環境における APPC メッセージのリカバリー可能性を判断することができます。リソース障害はリカバリーに影響を与えます。

MSC 環境で APPC トランザクションを回復するために、起こりうる障害のタイプを分析します。エラーをどう処理するかは、以下の事項に依存します。

- 障害のあるリソースは何か: LU 6.2 セッション障害か、IMS 障害か、アプリケーション・プログラム障害か、または MSC リンク障害か。
- トランザクション・モードは何か: リカバリー可能か、リカバリー不能か。
- トランザクション・タイプは何か: ローカルか、リモートか。
- LU 6.2 会話モードは何か: 非同期か、同期か。

トランザクションを定義する方法により、リカバリーを制御することができます。以下のトピックでは、関連する内容を明らかにし、IMS ライブラリー内のどこで、それらが詳細に説明されているかを示してあります。

## リカバリー可能トランザクションとリカバリー不能トランザクション

TRANSACT マクロに INQUIRY= キーワードをコーディングすることによって、トランザクションのリカバリー状況を IMS に知らせます。非照会モードのトランザクションはリカバリー可能であり、照会モードのトランザクションは、TRANSACT マクロに RECOVER パラメーターを指定しない限り、リカバリー不能です。

リカバリー可能トランザクションは、COLDSTART、COLDSYS、または COLDCOMM 再始動が実行されない限り、どんな IMS 障害、シャットダウン、あるいは再始動からでも回復されます。

LU 6.2 セッションが開始されるローカル・システムと、トランザクションがアプリケーション・プログラムによって処理されるリモート・システムとで、同一のリカバリー可能性属性をもつリモート・トランザクションを定義する必要があります。中間 IMS でトランザクションを定義する必要はありません。

メッセージ通信 (1 つの LTERM から別の LTERM へのメッセージ) は、常にリカバリー可能です。

## 関連概念

複数システムのリカバリーの考慮事項 (オペレーションおよびオートメーション)

## ローカル APPC トランザクションの廃棄可能性と廃棄不能性

ローカル APPC トランザクションが廃棄可能であるか廃棄不能であるかは、トランザクションを IMS に送信するために選択した LU 6.2 プロトコルと、選択したトランザクション・モード (リカバリー可能またはリカバリー不能) によって決まります。

IMS がローカル APPC トランザクションを廃棄するのは、次のいずれかの場合です。

- CPI 通信ドリブン・アプリケーション・プログラム (SYNCLVL=SYNCPT が指定されていない) である場合
- 照会専用でリカバリー不能と定義されている場合
- LU 6.2 アプリケーション・プログラムからの同期入力の結果である場合
- APPC Sync\_Level オプション NONE を使用している場合

上記以外の場合には、トランザクションは廃棄不能です。IMS は可能な場合にはいつでも廃棄不能トランザクションを回復しますが、廃棄可能トランザクションを回復することはありません。

## トランザクション処理の障害点

トランザクション処理における障害点もトランザクションのリカバリー可能性に影響します。

例えば、ローカルまたはリモートのトランザクションが処理されてコミット・ポイント (同期点) に達すると、そのトランザクションがリカバリー不能と定義されていても、IMS は出力応答をログから回復します。ローカル APPC 廃棄可能トランザクションは、IMS が出力応答メッセージを入力元の APPC アプリケーション・プログラムに送信した後で、コミット・ポイントに達します。この場合には、コミット・ポイントに達した後で障害が起こると、IMS には回復または廃棄する出力応答メッセージがありません。IMS がトランザクションを MSC リンクでキューに入れた場合には、IMS はリンク障害からトランザクションを回復します。

メッセージは、起こりうる障害のタイプにより、リカバリー可能、リカバリー不能のいずれか、および廃棄可能、廃棄不能のいずれかにすることができます。このトピックの説明では、LU 6.2 セッション、MSC リンク、ローカル IMS、中間 IMS、リモート IMS、またはアプリケーション・プログラムの障害が、それぞれユーザーのトランザクションにどのように影響するかを示します。この説明では、読者は障害の発生箇所と回復に必要な処置を理解できるものと想定しています。

## LU 6.2 セッション障害後のトランザクションの回復

IMS が入力メッセージを受信中に LU 6.2 セッションに障害が起こると IMS はそのメッセージを廃棄します。

IMS が完全にメッセージを受信すると、会話が次のどれであるかにより処理は異なります。

CPI-C または非 CPI-C

同期または非同期

ローカルまたはリモート

## CPI-C トランザクション

CPI-C トランザクションのプロセス中に LU 6.2 セッションに障害が起こった場合には、アプリケーション・プログラムは会話を終了するか、または処理を継続することを選択することができます。IMS TM は関係しないので、更新をコミットするか、バックアウトするかを選択を制限することはありません。アプリケーション・プログラムが判断をします。IMS TM はセッションの障害を知らないため、何の処置も取りません。コミットとバックアウトの通常の処理規則を適用します。IMS は LU 6.2 会話をリカバリーしません。

**関連資料:** CPI-C LU 6.2 アプリケーション・プログラムの設計について詳しくは、「IMS V15 アプリケーション・プログラミング」を参照してください。



## 非 CPI-C トランザクション

ローカル IMS が非 CPI-C のトランザクション出力を LU 6.2 プログラム へ送信中に LU 6.2 セッション障害が起これ、会話が同期である場合には、IMS はメッセージ制御/エラー出口ルーチン呼び出し、処理を打ち切ってバックアウトするかまたは処理を継続するかを判断します。デフォルトのアクションは、トランザクションを停止し、出力メッセージを廃棄することです(これは、すべての保護会話 (SYNCLVL=SYNCPT を使用して割り振られた会話) の操作のモードです)。会話が非同期である場合には、IMS はメッセージ制御/エラー出口ルーチン呼び出さずに、DFSASYNC という TP 名のメッセージ・キューに出力を入れます。

**関連資料:** メッセージ制御/エラー出口ルーチンのコーディングについては、「IMS V15 出口ルーチン」を参照してください。

## リモート APPC トランザクション

リモート APPC トランザクションの処理中に LU 6.2 セッション障害が起こった場合には、出力メッセージがローカル・システムの MSC リンクでエンキューされていれば、IMS はそのメッセージを回復します。トランザクションが MSC リンクでエンキューされる点に達していなければ、IMS はそのトランザクションを廃棄します。リカバリー可能性モードにかかわらず、また LU 6.2 会話が同期か非同期にかかわらず、IMS はそのトランザクションを廃棄します。この場合には、IMS はメッセージ制御/エラー出口ルーチン呼び出しません。

- トランザクションが非同期である場合には、LU 6.2 セッション障害のリモート・トランザクションからの出力が、リモート・システムから発信元システムに戻ると、IMS は DFSASYNC という TP 名を使用して、LU 6.2 アプリケーション・プログラムに非同期で応答を送信します。
- トランザクションが同期である場合には、LU 6.2 セッション障害のリモート・トランザクションからの出力がリモート・システムから発信元システムに戻ると、IMS はメッセージ制御/エラー出口ルーチン呼び出して、トランザクション出力を廃棄または再度経路指定します。デフォルトの処置では、出力は廃棄されます。

**関連資料:** アプリケーション・プログラムでの DFSASYNC の使用については、「IMS V15 アプリケーション・プログラミング」を参照してください。

### 関連概念

#### 25 ページの『CPI 通信』

このトピックでは、CPI 通信ドリブン・アプリケーション・プログラムおよび分散同期点保護会話を説明します。

## MSC リンク障害後のトランザクションの回復

MSC リンク障害が起こると、IMS は常にすべてのメッセージを回復し、このメッセージには、キューにすでに入っているか、キューに入れようとしている、あるいは MSC リンクを使用して送信中の IMS トランザクションおよび応答が含まれます。このリカバリー可能性は、メッセージがローカル、中間、リモートのどの MSC システム への途中であっても、保証されます。

このリカバリー可能性は、トランザクション・モード(リカバリー可能またはリカバリー不能)やローカル IMS へトランザクションを送信するために使用される LU 6.2 プロトコルの廃棄可能または廃棄不能特性の影響を受けません。

リンク障害は、他の IMS システムからのメッセージを遅らせたり、同期 LU 6.2 会話が予想以上に長く応答を待機する原因になったりすることがあります。

### 関連タスク

[論理リンクの再始動\(オペレーションおよびオートメーション\)](#)

## ローカル IMS 障害後のトランザクションの回復

ローカル APPC トランザクションが廃棄可能性基準を満たしている場合には、ローカル IMS 障害が生じると、IMS はそれらのローカル APPC トランザクションを廃棄します。上記の廃棄可能性テストに当てはまらない場合には、トランザクションは廃棄不能であるため IMS はローカル APPC トランザクションを廃棄しません。

リモート APPC トランザクションを照会タイプ・トランザクションとして定義し、ローカル IMS の TRANSACT マクロ定義で RECOVER を指定しないと、IMS はローカル IMS 障害の後でそれらのトランザクションを回復しません。その他の場合には、IMS はすべてのリカバリー可能、非会話型トランザクションを回復します。

ローカル IMS がトランザクション・メッセージを中間またはリモート IMS へ送信した後は、ローカル IMS に障害が起ころうとも、トランザクションのリカバリー可能性には何の影響もありません。トランザクションはその宛先での存続を続け、処理されます。リモート IMS が障害後にトランザクションの応答を発信元 IMS に送信すると、IMS はデフォルトのトランザクション・プログラム名 (TPN) DFSASYNC を使用して非同期で応答をその宛先に送信します。LU 6.2 アプリケーション・プログラマーはこの状況への対応を予定する必要があります。

## 関連概念

48 ページの『ローカル APPC トランザクションの廃棄可能性と廃棄不能性』

ローカル APPC トランザクションが廃棄可能であるか廃棄不能であるかは、トランザクションを IMS に送信するために選択した LU 6.2 プロトコルと、選択したトランザクション・モード (リカバリー可能またはリカバリー不能) によって決まります。

APPC のアプリケーションの設計 (アプリケーション・プログラミング)

## リモート IMS 障害後のトランザクションの回復

リモート IMS 障害が生じると、IMS は、リモート IMS で処理のためにキューに入っているか、リモート IMS で処理されているトランザクションを、リモート IMS における APPC トランザクションのリカバリー可能性属性に基づいて回復します。

リカバリー可能トランザクションは回復され、リカバリー不能トランザクションは回復されません。前に述べたように、一度トランザクションがコミット・ポイントに達すると、IMS はトランザクションのリカバリー属性にかかわらず、出力応答メッセージを回復します。発信元 IMS における APPC 会話の廃棄可能および廃棄不能特性は、リモート IMS におけるリモート IMS 障害前後でのトランザクションのリカバリー可能性とは関係ありません。

IMS は、リモート IMS 障害が起こったときにリモート IMS への経路上にあるトランザクション (つまり、トランザクション・メッセージがまだローカルまたは中間 IMS 内の経路上にあるトランザクション) を、トランザクションのリカバリー可能性特性にかかわらず回復します。障害後、リモート IMS は、障害の発生時に経路内にあったトランザクションを受信して処理します。

## 中間 IMS 障害後のトランザクションの回復

IMS は、中間 IMS のキューに入れられている、中間 IMS への経路上、または中間 IMS からの経路上にあるすべてのメッセージを、トランザクションまたはメッセージのリカバリー可能性特性には関係なく、常に回復します。

中間 IMS が COLDSTART、COLDCOMM、または COLDSYS で再始動した場合には、メッセージは失われます。

## アプリケーション・プログラム障害後のトランザクションの回復

LU 6.2 アプリケーション・プログラムから IMS へ送信されたトランザクションは、アプリケーション・プログラム障害が起こると、非 LU 6.2 開始トランザクションと同様に処理されます。

LU 6.2 装置からのローカルまたはリモート・トランザクションの処理中に、コミット・ポイントに達する前にアプリケーション・プログラムに障害が起こった場合には IMS は、代替高速 PCB に挿入され、PURG 呼び出しでコミットされたメッセージを除く、すべてのメッセージをバックアウトします。トランザクションがコミット・ポイントに達した後に、障害が発生すると、IMS は、すべてのメッセージを回復します。障害のあるアプリケーション・プログラムの入力メッセージが、別のアプリケーション・プログラムから受信したものである場合には (プログラム間通信)、この前のアプリケーション・プログラムの処理は依然として (非 LU 6.2 アプリケーション・プログラムと同様に) コミットされています。



## LU 6.2 トランザクションのリカバリー可能性フロー

このトピックでは、同期および非同期トランザクションのフロー、およびトランザクションがリカバリー可能、リカバリー不能、廃棄可能、または廃棄不能である場合を示す4つのリストが示されています。

以下のリストは、LU 6.2 同期会話からローカル IMS に送信されたトランザクションのフローを示しています。

1. LU 6.2 プログラム: ALLOC LU=IMS LU 名
2. LU 6.2 プログラム: ローカル IMS へ SEND
3. LU 6.2 プログラム: RECEIVE\_AND\_WAIT
4. ローカル IMS がトランザクションを受信
5. トランザクションがエンキューされる
6. トランザクションを実行(コミット・ポイントに到達する前にアプリケーションに障害が発生すると、メッセージは廃棄される)
7. メッセージが入出力 PCB に挿入される
8. ローカル IMS が出力を送信する(ここで LU 6.2 セッションに障害が発生した場合、メッセージ制御/エラー出口ルーチンが制御を受け取る)
9. コミット・ポイント
10. LU 6.2 プログラム: DEALLOCATE

以下のリストは、LU 6.2 非同期会話からローカル IMS に送信されたトランザクションのフローを示しています。

1. LU 6.2 プログラム: ALLOC LU=IMS LU 名
2. LU 6.2 プログラム: ローカル IMS へ SEND
3. LU 6.2 プログラム: DEALLOCATE
4. ローカル IMS がトランザクションを受信
5. トランザクションがエンキューされる
6. トランザクションを実行(コミット・ポイントに到達する前にアプリケーションに障害が発生すると、メッセージは廃棄される)
7. メッセージが入出力 PCB に挿入される
8. コミット・ポイント
9. ローカル IMS: TPN=DFSASYNC を指定して ALLOCATE
10. ローカル IMS が出力を送信
11. ローカル IMS: DEALLOCATE

以下のリストは、LU 6.2 同期会話からリモート IMS に送信されたトランザクションのフローを示しています。

1. LU 6.2 プログラム: ALLOC LU=IMS LU 名
2. LU 6.2 プログラム: ローカル IMS へ SEND
3. LU 6.2 プログラム: RECEIVE\_AND\_WAIT
4. ローカル IMS がトランザクションを受信
5. リモート・キュー上でエンキューされたトランザクション および MSC リンク (MSC リンク上でエンキューされた後の MSC リンク障害に渡ってメッセージはリカバリー可能です)
6. ローカル IMS が MSC リンクを越えてリモート IMS にメッセージを送信
7. リモート IMS がトランザクションを受信
8. トランザクションが実行
9. 出力メッセージが入出力 PCB に挿入される

10. リモート IMS が出力メッセージを MSC リンクへエンキュー (MSC リンク上でエンキューされた後の MSC リンク障害に渡ってメッセージはリカバリー可能です)
11. コミット・ポイント
12. リモート IMS が MSC リンクを越えてローカル IMS に出力メッセージを送信
13. ローカル IMS が出力メッセージを受信
14. ローカル IMS が出力メッセージを LU 6.2 プログラム用にエンキュー
15. ローカル IMS が出力メッセージを LU 6.2 プログラムに送信 (ここで LU 6.2 セッションに障害が発生したとき、メッセージ制御/エラー出口ルーチンが制御を受け取る)
16. LU 6.2 プログラム: DEALLOCATE

以下のリストは、LU 6.2 非同期トランザクションからリモート IMS に送信された トランザクションのフローを示しています。

1. LU 6.2 プログラム: ALLOC LU=IMS LU 名
2. LU 6.2 プログラム: ローカル IMS へ SEND
3. LU 6.2 プログラム: DEALLOCATE
4. ローカル IMS がトランザクションを受信
5. リモート・キュー上でエンキューされたトランザクション および MSC リンク (MSC リンク上でエンキューされた後の MSC リンク障害に渡ってメッセージはリカバリー可能です)
6. ローカル IMS が MSC リンクを越えてリモート IMS にメッセージを送信
7. リモート IMS がトランザクションを受信
8. リモート IMS がトランザクションをエンキュー
9. トランザクションが実行
10. 出力メッセージが入出力 PCB に挿入される
11. リモート IMS が出力メッセージを MSC リンクへエンキュー (MSC リンク上でエンキューされた後の MSC リンク障害に渡ってメッセージはリカバリー可能です)
12. コミット・ポイント
13. リモート IMS が MSC リンクを越えてローカル IMS に出力メッセージを送信
14. ローカル IMS が出力メッセージを受信
15. ローカル IMS が出力メッセージを LU 6.2 プログラム用にエンキュー
16. ローカル IMS: TPN=DFSASYNC を指定して ALLOCATE
17. ローカル IMS が出力メッセージを LU 6.2 プログラムに送信 (ここで LU 6.2 セッションに障害が発生したとき、メッセージ制御/エラー出口ルーチンが制御を受け取る)
18. ローカル IMS: DEALLOCATE

## XRF および APPC の計画

障害中の IMS へ送信された LU 6.2 会話メッセージはすべて、XRF テークオーバーの時点で破壊されます。テークオーバー処理が完了した後で、新しいアクティブ IMS に、新しい会話を割り振る必要があります。LU 6.2 アプリケーション・プログラムに対する XRF サポートはクラス 3 サービスです。

アプリケーション・プログラムは、ALLOCATE verb の LU 名パラメーターとして USERVAR パラメーターを使用します。これにより、同じ LU 名を使用して新規アクティブ IMS の会話を割り振ることができます。これは、非 LU 6.2 IMS 端末における LU の使用法と同じ概念ですが、異なる USERVAR 名が使用されます。テークオーバー時に、代替 IMS は VTAM MODIFY USERVAR コマンドを出し、失敗したアクティブ IMS から代替 IMS に USERVAR をリダイレクトします。さらに、障害のあったアクティブ IMS に対して VTAM VARY TERM コマンドを出し、テークオーバー中に保留されたすべての会話を終了させます。次に、VTAM は、必要に応じて他の VTAM インスタンスに USERVAR の変更を伝えます。

リモート IMS にエンキューされた APPC トランザクションは MSC プロトコルによって処理されます。ローカル IMS でのテークオーバー後、キューにある回復不能メッセージは失われます。同じメッセージがリ

モート IMS にキューイングされた場合には、ローカル IMS が引き継ぐときに回復されます。処理は継続し、応答はアクティブ IMS へ送信されます。

VTAM、z/OS、Tivoli® NetView® for z/OS、およびその他の製品の要件は、注意して検討する必要があります。APPC/IMS はこれらの他の製品の正しい操作を必要とします。

## トランザクション再試行特性

IMS は、ある種の異常終了条件を再試行します。

再試行される条件の例をいくつか次に示します。

- デッドロック
- ロック拒否
- 高速機能再試行条件

これらの再試行条件は、標準 DL/I アプリケーション・プログラム (これらが LU 6.2 アプリケーション・プログラムからメッセージを受け取る場合でも) に適用されます。再試行できる異常終了が起こると、IMS は APPC/MVS に APPC ATBEXAI 呼び出しを出し、確立された会話が存在するかどうかを判別します。会話が割り振られていると、異常終了は再試行されません。そして、アプリケーション・プログラムは終了します。

アプリケーション・プログラムがいずれかの CPI 通信リソースを使用していると、異常終了条件は再試行できません。したがって、異常終了が起こる前に LU 6.2 会話を割り振られていた CPI 通信ドリブン・アプリケーション・プログラムと変更された IMS アプリケーション・プログラムは、再試行できません。IMS は CPI 通信リソースの状態を制御できないので、このように再試行を禁止する必要があります。IMS は、CPI 通信ドリブン・アプリケーション・プログラムについては、DL/I INIT STATUS GROUPB 呼び出しをサポートしますが、デッドロックが検出される前に LU 6.2 会話を割り振られた変更済み IMS アプリケーション・プログラムについては、これらの呼び出しをサポートしません。

## ネットワーク LU 名の修飾

ネットワーク ID を追加することによって、異なるシステムの LU に同じ名前を使用することができます。このため、複合システムにおいて、すべてのシステムのすべての LU に固有名を付ける必要はありません。ネットワーク修飾 LU 名をパートナー LU の名前として使用して、リモート LU 6.2 会話およびセッションを割り振ることができます。

### このタスクについて

ネットワーク修飾 LU 名は、発信元システムのネットワーク ID (1 文字から 8 文字) の後にピリオドと LU 名を続けて構成します。ネットワーク修飾 LU 名は、コマンドに使用する場合は以下に示すように、単一引用符で囲まなければなりません。

**例:** /DISPLAY LUNAME 'NETID1.LUAPPC2'

データをリモート宛先へ転送するときには、ネットワーク修飾 LU 名を使用してください。ネットワーク ID が存在しない場合、IMS は z/OS が宛先を決定できるようにします。

コマンドの中のネットワーク修飾 LU 名で、ネットワーク ID または LU 名に対してパラメーター ALL を指定しても置き換えは行われません。

**例:** /DISPLAY LUNAME 'NETID1.ALL'

LU 6.2 記述子の LU 名はネットワークで修飾することができます。LUNAME キーワードをサポートするコマンドでは、ネットワーク修飾 LU 名はオプションです。

APPC/MVS は、LU 6.2 ネットワーク修飾 LU の名前を使用して、会話を割り振ります。APPC/MVS は、ネットワーク ID を除去して、8 バイトの LU 名を VTAM に渡します。APPC/MVS のネットワーク修飾名サポートが導入されていない場合には、異なるネットワークで LU 名を固有にしなければなりません。

ネットワーク ID または LU 名に対して ALL パラメーターを指定しても、コマンド内のネットワーク修飾 LU 名では置き換えは行われません。

LU 6.2 記述子の LU 名はネットワークで修飾することができます。

**関連資料:** コマンドでのネットワーク修飾名の使用については、「IMS V15 コマンド 第 1 巻: IMS コマンド A-M」を参照してください。

## 単一 IMS システムでの複数の LU の管理

IMS システムは、ネットワーク上では複数の LU 名により表現できます。単一の IMS システムに対して複数の LU 名が存在する場合は、非同期のアウトバウンド・メッセージを処理するのがどの LU かを指定する必要があります可能性があります。

### このタスクについて

IMS システムに対して複数の LU が定義されている場合、その LU の 1 つがデフォルト、つまり基本 LU の役割を果たします。通常、アウトバウンド会話に対するすべての割り振り要求を処理するのは基本 LU です。ただし、場合によっては、リモート・パートナー LU は基本 LU 以外の LU からのアウトバウンド会話を予想していることがあり、その場合はどの LU がアウトバウンド会話の割り振り要求を処理するかを制御することができます。以下の方法を使用することにより、どの LU がアウトバウンド会話の割り振り要求を処理するかを制御することができます。

- IMS.PROCLIB データ・セットの DFS62DTx メンバーで、LU 6.2 記述子の OUTBND パラメーターに LU 名を指定する。IMS は OUTBND パラメーターで指定されたローカル LU 名を使用して、LU 6.2 記述子に送信されたアウトバウンド・メッセージを経路指定します。

/CHANGE DESC コマンドを使用して、LU 6.2 記述子に対するローカル LU を変更します。

/DISPLAY DESC コマンドを使用して、ローカル LU とその他の記述子の指定を表示します。

- DFSDCxx 始動プロシージャの OUTBND パラメーターで、LU 名を指定する。LU 名を DFSDCxx 始動プロシージャの OUTBND パラメーターで指定すると、指定された LU は元のインバウンド会話を受信した LU がどれだったかに関係なく、すべての非同期アウトバウンド・メッセージのデフォルト LU の役割を果たします。
- DFSDCxx 始動プロシージャのローカル LU オプション APPCLU=Y を指定する。ローカル LU オプションを指定すると、IMS は、元のインバウンド会話を受信した LU を介して、非同期のアウトバウンド会話を経路指定します。
- LU 6.2 編集出口ルーチン (DFSLUEE0) の LU 名指定をオーバーライドする。

## 別の IMS システムへの LU の再割り当て

MVS コマンドを使用すると、ある IMS システムから別の IMS システムに LU を再割り当てできます。

### このタスクについて

LU を別の IMS システムに再割り当てするには、以下のようにします。

### 手順

1. MVS コマンド SETAPPC LUDEL を発行して、現行の IMS システムから LU を削除します。
2. MVS コマンド SETAPPC LUADD,ACBNAME=*luname*,SCHED=*new\_IMS*,NQN を発行して、新規の IMS システムで LU を再定義します。

## DFSAPPC システム・サービス

DFSAPPC は IMS システム・サービスであり、LU 6.2 アプリケーション・プログラム間 (LU 6.2 から LU 6.2)、および LU 6.2 アプリケーション・プログラムと IMS が管理する LTERM 間のメッセージ通信を行います。メッセージ送達は非同期であり、メッセージは送達されるまで IMS メッセージ・キューに保留されます。

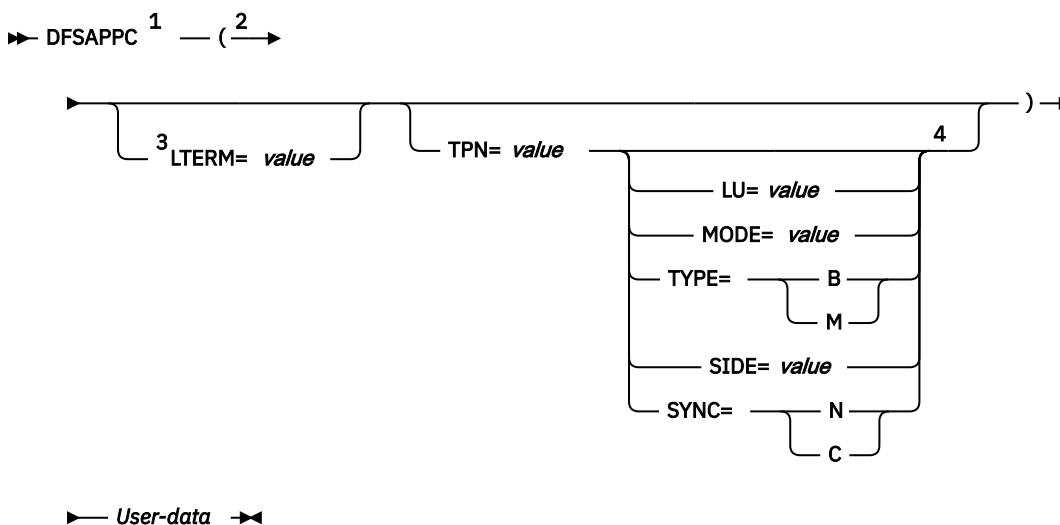
LU 6.2 アプリケーション・プログラムは、DFSAPPC を使用して、メッセージを IMS 管理の LTERM へ送信することができます。この機能を選択するには、DFSAPPC サービスの LTERM オプションを使用してください。

## メッセージ通信

メッセージ通信は暗黙 APPC インターフェースの一部であり、IMS 端末と LU 6.2 アプリケーション・プログラムがメッセージを交換できるようにします。LU 6.2 アプリケーション・プログラムに渡されるメッセージは、LU 6.2 アプリケーション・プログラムを開始します。

DFSAPPC を使用すると、リモート装置は LTERM か LU 6.2 TPN オプションのどちらかを使用してメッセージを送る経路の選択をすることができます。LTERM オプションで送信されるメッセージは、IMS が管理するローカルまたはリモートの LTERM に送信されます。<sup>3</sup>LTERM オプションなしでメッセージを送信すると、指定された LU 6.2 アプリケーション・プログラムに送信されます。

DFSAPPC のメッセージ形式は、以下の図に表示されています。



注:

- <sup>1</sup> DFSAPPC とオプションの間には空白が必須です。
- <sup>2</sup> キーワードあるいは値を除いて、DFSAPPC オプションではどこでも空白を使用することができます。キーワード・パラメーターの集合、あるいは空白個所の間では区切り文字としてコンマを使用してください。ただし、TP 名文字セットではコンマを使用できるので、TPN 値を終了するために少なくとも1つの空白を使用しなければなりません。
- <sup>3</sup> LTERM= あるいは TPN= オプションのどちらかを指定することができますが、両方を指定することはできません。TPN= オプションを指定した場合だけ、他のキーワード・オプションを使用することができます。
- <sup>4</sup> 値が他の発信元から提供されない場合は、パートナー・プログラムとの LU 6.2 会話を確立するには DFSAPPC オプションに IMS デフォルトを使用してください。DFSAPPC サービスが、なにもオプションを指定しないでコーディングされている場合は、IMS デフォルト LU 6.2 会話特性を使用してください。

図 10. DFSAPPC メッセージ・フォーマット

DFSAPPC オプション・キーワードを、以下のようにオカレンス順に定義します(ただし、TPN= の後のキーワードはアルファベット順にリストされています)。

### LTERM=

LTERM オプションは 1-8 文字で、IMS LTERM の名前です。LTERM オプションを使用して送信されたメッセージは、IMS が管理しているローカルまたはリモート LTERM へ送られます。LTERM が LU 6.2 記述子に関連している場合、LU 6.2 アプリケーション・プログラムが明示的に選択されたものとして扱わ

<sup>3</sup> LTERM が LU 6.2 宛先と関連付けられている場合には、あたかも LU 6.2 アプリケーション・プログラムが明示的に選択されたかのように、メッセージが送信されます。

れます。LTERM 名には、英大文字、数字、および国別文字 ('@'、'\$'、'#') を使用することができます。LTERM が指定されると、他のキーワードは指定できません。

#### **TPN=**

TPN オプションは 1-64 文字で、パートナー・プログラムとの LU 6.2 会話を確立するために論理装置名と共に使用されるパートナー・トランザクション・プログラム名です。TP 名文字セットではコンマを使用できるので、TPN 値を終了するために少なくとも 1 つの空白を使用する必要があります。

TP 名は、空白を除いて 00640 文字セットにある任意の文字を含むことができます。00640 文字セットについては、「*CPI Communications Specification*」に文書化されており、大文字と小文字の A から Z、数字の 0 から 9、および 20 個の特殊文字が含まれます。

TPN と SIDE オプションが指定されると、TPN 名はサイド情報項目にある TP 名を上書きします。

DFSAPPC では 00640 文字セットを使用できますが、IMS コマンドはこの文字セットを使用しません。IMS コマンドは TPN 上でのみ操作することができ、TPN は英大文字、数字、および国別文字 ('@'、'\$'、'#') を使用します。IMS コマンドは拡張 TPN では操作できません。

#### **LU=**

LU オプションは 1-17 文字で、パートナー・プログラムとの LU 6.2 会話を確立するためにトランザクション・プログラム名 (TPN) と共に使用されるパートナー論理装置名です。

LU 名は、APPC/MVS タイプ A 文字セットにある任意の文字を含むことができます。LU 名には、大文字の英字、数字、および国別文字 ('@'、'\$'、'#') を含めることができ、英字または国別文字で始める必要があります。また、LU フィールドに 17 バイトのネットワーク修飾 LU 名を使用することもできます。

LU と SIDE オプションが指定されると、LU 名はサイド情報項目にある LU 名を上書きします。

#### **MODE=**

MODE オプションは 1-8 文字で、パートナー・プログラムとの LU 6.2 会話を確立するために論理装置名およびトランザクション・プログラム名と共に使用されるパートナー・モード名です。

MODE 名は、APPC/MVS タイプ A 文字セットにある任意の文字を含むことができます。MODE 名は、大文字の英字、数字、および国別文字 ('@'、'\$'、'#') を含めることができ、英字または国別文字で始める必要があります。

MODE= と SIDE オプションが指定されると、MODE 名はサイド情報項目にあるモード名を上書きします。

#### **SIDE=**

SIDE オプションは 1-8 文字で、パートナー・プログラムとの LU 6.2 会話を確立するために使用されるサイド情報項目名です。

SIDE 名は、01134 文字セットにある任意の文字を含むことができます。01134 文字セットについては、「*CPI Communications Specification*」に文書化されており、英大文字の A から Z、および数字の 0 から 9 が含まれます。

SIDE オプションが指定されると、LU、TPN、さらに MODE オプションはサイド情報項目登録にある値を上書きするために指定することもできます。

#### **SYNC=**

SYNC オプションにより、アプリケーションは IMS によって提供される LU 6.2 会話同期レベルのデフォルトをオーバーライドできます。

##### **SYNC=N**

Sync\_Level は NONE です。

##### **SYNC=C**

Sync\_Level は CONFIRM です。

#### **TYPE=**

TYPE オプションにより、アプリケーションは IMS によって提供される LU 6.2 会話同期レベルのデフォルトをオーバーライドできます。

##### **TYPE=B**

会話タイプは BASIC です。

## TYPE=M

会話タイプは MAPPED です。

オプション・リストを処理している時にエラーが見つかったら、エラー・メッセージ DFS1957 DFSAPPC ERROR が端末に送信されます。

**関連資料:** タイプ A 文字セットについては、「z/OS MVS プログラミング: APPC/MVS トランザクション・プログラムの書き方」を参照してください。

## 非同期出力の送達

LU 6.2 アプリケーション・プログラム宛てのメッセージを作成する場合は、IMS は会話特性を設定します。これらの特性は以下から持って来ることができます。

LU 6.2 記述子

DL/I 変更呼び出しオプション・リスト

DFSAPPC メッセージ通信オプション

IMS が上記リストから特定の会話特性を取り出せない場合、IMS は、以下の表に示されているデフォルトを使用します。サイド・テーブル名が持ってこられた場合、デフォルト・モード名は使用されません。IMS は、サイド・テーブル項目がモード・テーブル項目名を持っていると想定します。特定の LUNAME - TPNAME 宛先に対する /ALLOCATE コマンドでモード・テーブル項目名を指定すると、その項目名はメッセージに指定されたモード・テーブル名を上書きします。

IMS は、以下の表の情報を使用して、代替 PCB に関連付けられている LU 6.2 アプリケーション・プログラムとの会話を開始します。LU 名などある種のフィールドはアプリケーションに固有のものです。デフォルトは提供されますが、メッセージに関連するパラメーターによって上書きされる可能性があります。IMS がデフォルト値を使用するのは、他のどのソースからも値が提供されない場合だけです。アプリケーション・プログラムは、DL/I CHNG 呼び出しとの拡張インターフェースを使用して、デフォルト会話特性を変更することができます。

表 1. APPC/IMS デフォルト会話特性

特性	デフォルト値
Conversation_Type	Mapped
Deallocate_Type	Deallocate_Sync_Level
Error_Direction	Receive_Error
Fill	Fill_LL
Log_Data	NULL
Log_Data_Length	0
Mode_Name	'DFSMODE'
	IMS は、アウトバウンド会話を割り振るため LU 6.2 パートナーを入力することによって提供されるのと同じモード名を使用します。すなわち、会話を入力しているモード名が何であろうと、IMS はアウトバウンド割り振りにそのモード名を使用します。
Mode_Name_Length	7
Partner_LU_Name	'DFSLU'
Partner_LU_Name_Length	5
Prepare_to_receive_type	Prep_To_Receive_Sync_Level
Receive_type	Receive_and_Wait
Return_Control	When_Session_Allocated



表 1. APPC/IMS デフォルト会話特性 (続き)

特性	デフォルト値
Send_Type	Buffer_Data
Sync_Level	Confirm
TP_Name	'DFSASYNC'
TP_Name_Length	8

## APPC トランザクション・セキュリティ

APPC/IMS と LU 6.2 アプリケーション・プログラムのセキュリティ・オプションは、強力で広範囲な可能性を持っています。パートナー・システムは、単一ユーザー端末またはワークステーションから複数ユーザー・システムまで広範囲に渡ります。すべてのシステムが、それぞれの複合システム・セキュリティ環境を持つことができます。IMS のセキュリティは単純にもできるし、複雑にもできます。

すべてのトランザクション・プログラム名 (TPN) は、実行される前にセキュリティ検査をパスしなければなりません。トランザクションを開始するユーザー ID は、LU 6.2 形式ヘッダー (FMH5) で識別されます。SECURITY=NONE を指定したためにユーザー ID がない場合、UACC (NONE) を使用して定義されていないリソースだけをアクセスできます。すべての状況においてアクセスできる TPN は、UACC (NONE) を使用して定義してはなりません。TPN セキュリティの定義は必須です。

z/OS セキュリティは 2 つの部分から成ります。最初に、z/OS はトランザクション・ユーザーを認証します。LU 6.2 トランザクションにはセキュリティ情報があります。FMH5 には、ユーザー ID、グループ名として使用される「プロファイル」名、およびセキュリティ・オプションが含まれます。ユーザーはユーザー ID とパスワードを入力します。ユーザー ID は RACF に定義されていて、パスワードはそのユーザー ID に対して有効でなければなりません。

Already\_Verified が FMH5 に指定されると、送信システムはユーザー ID を検査します。このユーザー ID は、受信側 z/OS システムで RACF に定義されなければなりません。この場合、パスワードは必要ありません。

SECURITY=NONE が指定されると、z/OS はなにも検査しません。その代わりに z/OS は、SECURITY=NONE に対応する特別のセキュリティ・プロファイルを作成します。これにより、NONE 以外の任意のレベルで指定された UACC を持つ z/OS リソースと APPC/IMS リソースにアクセスできます。UACC (NONE) があるリソース、または指定した UACC が無いリソースはアクセスできません。

ユーザー ID が設定されると、z/OS はユーザー ID が特定のトランザクションを実行できるかどうかを検査します。z/OS は、ユーザー ID のアクセス・プロファイルに CLASS (APPCTP) にあるエンティティ *dbtoken.x.tpname* の ACCESS (EXECUTE) があることを検査します。*dbtoken* の値は、TP\_Profile データ・セットに指定された *dbtoken* 値です。この LU について指定された APPCTPxx パラメーターをもとにして、x の値はユーザー ID、グループ、あるいは SYS1 のいずれかです。

これらのセキュリティ検査のどれかが失敗すると、z/OS はトランザクションをリジェクトしますが、IMS にはそのことは通知されません。z/OS は以下についても検査することができます。

- セッション・レベル・セキュリティ (RACF リソース・クラス APPCLU)
- 入力のポート (RACF リソース・クラス APPCPORT)
- ローカル・アプリケーション (RACF リソース・クラス APPL)

IMS 管理担当者はこれらのセキュリティ検査が成功していることを検証する必要があります。

**関連資料:** RACF リソース・クラスのコーディングについては、「z/OS MVS 計画: APPC/MVS 管理」を参照してください。

IMS のセキュリティ検査は IMS トランザクション・コードか実行コマンド名を基に行われています。TPN が DFSAPPC の場合、セキュリティの追加検査は行われません。RACF がシステム上で使用されている場合、z/OS は、RACF がアクティブでないと、トランザクションを拒否します。TPN に関連している IMS コマンド名、あるいはトランザクション・コードは、この IMS に関連している RACF リソース・クラ



スで使用されます(「C」または「T」)。RACF は、同じ方法で他の IMS 端末タイプすべてに対して IMS コマンドと トランザクションを検査します。

RACF 検査が成功すると、トランザクションにはトランザクション許可出口ルーチン (DFSCTRNO) が呼び出され、コマンド許可には DFSCCMD0 が呼び出されます。ただし、RACF には次の規則が適用されます。

- コマンドについては、RACF が使用されていない場合にだけ デフォルトのセキュリティが適用されません。
- リモート・トランザクションの場合には、RACF はオプションです。

他の場合、出口ルーチンがセキュリティの判断を行います。

意図されている環境は、RACF セキュリティーをアクティブにして APPC/IMS を実行するものです。APPC/IMS システムで RACF をアクティブにせずに実行することは可能ですが、z/OS システムで RACF をアクティブにせずに実行することはできません。この意味で、RACF は LU 6.2 には必須です。

セキュリティ環境の複雑さは、関係するリソースの数が多いこと (VTAM、z/OS、および IMS) および保護対象が細分化されていることに一部起因しています。セキュリティの定義は、アプリケーション・システムの運用が正常に行えるように綿密に調整しておく必要があります。

## **関連概念**

IMS セキュリティー (システム管理)



---

## 第 3 部 拡張端末オプション (ETO) (Extended Terminal Option (ETO))

これらのトピックでは、拡張端末オプション (ETO) を紹介し、ETO の概要 および IMS TM ネットワーク内の ETO 端末の管理に必要な情報を説明します。



## 第4章 拡張端末オプションの概説

IMS の拡張端末オプション (ETO) を使用すると、システム定義中に VTAM 端末と ISC TCP/IP 端末およびユーザーを事前定義せずに、それらを IMS に追加することができます。

ETO は IMS Transaction Manager (TM) の一部であり、出力セキュリティー、自動ログオフ、および自動サインオフなどの追加機能をユーザーに提供します。

このトピックでは、システム・プログラマーに、ETO のインプリメントと管理に必要な概念情報を提供します。ETO に関してよく知らない場合に、このトピックの情報をお読みください。

**注:** ETO は、ISC の動的 TCPIP 端末を定義するために必要です。ただし、ETO は主に VTAM 端末に使用されます。以下のトピックの一部の情報は、動的に定義された ISC TCP/IP 端末に適用されますが、ETO に関するトピックのほとんどは、VTAM のみに関連しており、ETO の概念および管理について説明しています。

### ETO を使用する利点

ETO は、トランザクション・マネージャーの環境に 2 つの主な拡張機能を追加することができます。ETO では、以下のことができます。

- システム定義時に IMS に対して定義していない VTAM または ISC TCP/IP 端末との IMS セッションを設定することができます。
- 宛先が特定ユーザーである出力メッセージの安全が保護され、出力メッセージは特定ユーザーにのみ送信されます。

ETO をインストールすることによって、以下のこともできます。

- VTAM および ISC TCP/IP 端末の追加または削除のために予定されるダウン時間の減少によりシステムの使用可能性を改善します。
- ユーザーがネットワーク内の任意の VTAM および ISC TCP/IP 端末から IMS セッションを設定できるので、システムの迅速な使用が可能になります。
- 端末向けよりむしろユーザー向けとしての出力メッセージにより IMS のセキュリティーを改善します。
- 端末ネットワークを定義するために必要なマクロの数を削減します。これによって、システム定義時間とストレージを削減できます。
- チェックポイント時間および再始動時間を削減します。ETO 端末およびユーザー構造のリソースは、実際に必要になるまで割り振られません。また、必要でなくなると、削除されます。
- 静的端末定義の保守に必要な熟練したシステム・プログラマーの数が削減されます。

### 関連タスク

71 ページの『拡張端末オプションの管理』

IMS 拡張端末オプション (ETO) を使用すると、VTAM 端末とユーザーをシステム定義時に事前定義しなくても、IMS に動的に追加できるようになります。

## ETO 用語

特定の用語には、ETO に固有の、したがって ETO の理解と管理にとって重要な意味があります。

### 端末

端末、静的端末、および動的端末に対する定義がこのトピックで説明されています。

### 定義:

- 端末は物理的な VTAM 論理装置 (LU) であり、IMS とのセッションを設定します。物理端末は、制御ブロックを使用して表されます。

- 端末が ETO によって構築されていないが、システム定義で定義されている場合、静的端末と呼ばれます。静的端末に送信されたメッセージは、論理端末 (LTERM) メッセージ・キューに入れられ、受信側によってリトリブされるのを待ちます。
- 端末がシステム定義で定義されておらず、ETO によって構築された場合、動的端末または ETO 端末と呼ばれます。動的端末の場合、論理端末 (LTERM) は動的ユーザー・メッセージ・キューと呼ばれ、LTERM はメッセージを物理端末ではなく、ユーザーと関連付けます。メッセージをユーザーと関連付けることによって、ユーザーは固有のユーザー ID を使用してサインオンした時にだけメッセージにアクセスできるので、このようなユーザーのセキュリティが改善されます。さらに、ネットワーク内のすべてのユーザーが、任意の物理端末から自分宛てのメッセージにアクセスできます。特定の物理端末を使用する必要はありません。

## 動的ユーザー

**定義:** ETO 動的ユーザーは、動的端末にサインオンし、IMS がメッセージの送達に使用する固有の ID (ユーザー ID) を持っているユーザーです。このユーザーは、通常、個人に関連付けられますが、プリンターなどの別のものにも関連付けられます。

## 端末構造

端末構造とは、IMS に認識されている特定の端末を表す IMS 制御ブロックのことです。端末構造は、各端末が IMS にログオンすると作成されます。端末構造は、端末がログオフし、関連アクティビティー (IMS への次の接続のために保持する必要がある状況など) が残っていない場合には、削除されます。

## ユーザー構造

ユーザー構造とは、1つのユーザー・ブロックと1つ以上の LTERM ブロックを含む、IMS 制御ブロックの集まりのことです。メッセージ・キューは、物理端末ではなく動的ユーザーに関連付けられ、ユーザー ID のキューに入れられます。

動的ユーザー構造は、ユーザーがサインオンした場合にだけ物理端末に接続されます。これにより、同じ端末を使用する別々のユーザーが互いのメッセージを受信できなくなるので、セキュリティ機能のある環境が提供されます。

IMS は、以下のいずれかのイベントが発生した場合に、ユーザー構造を作成します。

- 動的ユーザーが IMS にサインオンした場合
- 動的ユーザー宛ての出力メッセージがそのユーザーに送信されたが、そのユーザーが IMS にサインオンしていない場合

通常、ユーザー構造は IMS を使用する個人を表します。ユーザー構造名は、通常はユーザー ID と同じです。ユーザー構造は、プリンターなどの論理宛先を表すこともできます。この場合、ユーザー構造名は、インストール・システムがそのアプリケーション・プログラムおよびその出口ルーチンで使用する LTERM 名と同じであっても同じでなくても構いません。例えば、プリンターのユーザー構造名を、その LTERM の宛先ノード名と同じにすることができます。ただし、この場合、出力はユーザーではなく端末のキューに入れられます。

以下の図は、静的リソースと ETO 動的リソースとの間の相違を示しています。

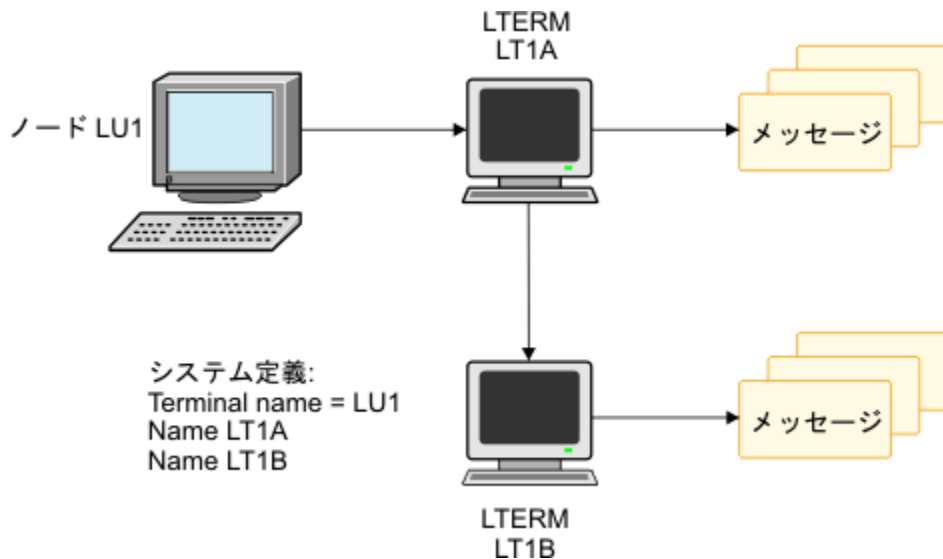


図 11. 静的リソース

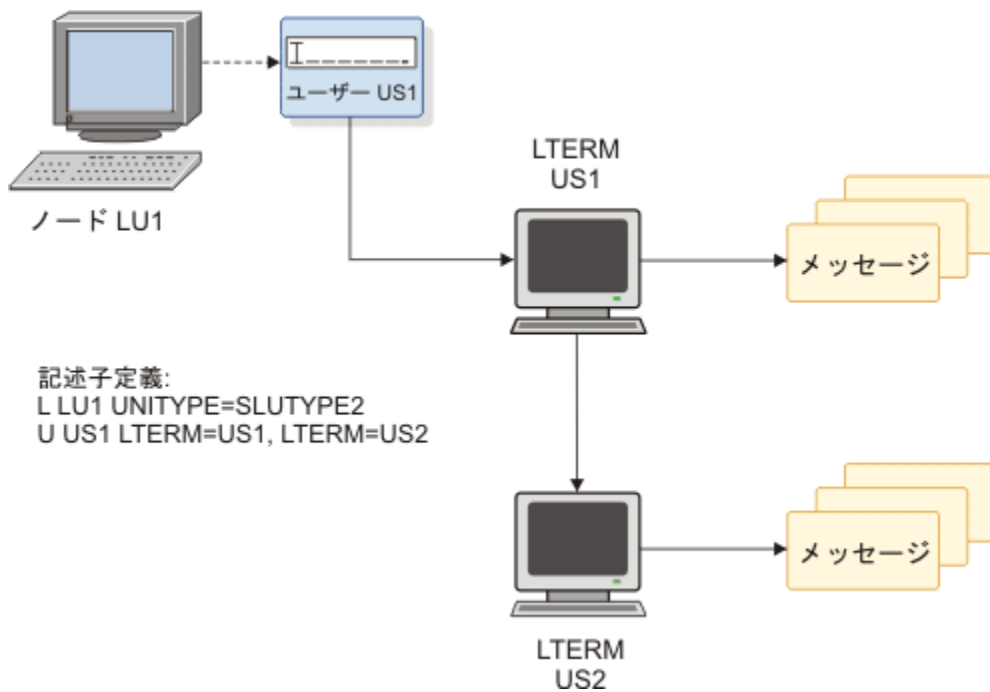


図 12. ETO 動的リソース

## ETO 記述子

記述子は、IMS がログオンまたはサインオンに必要な動的リソースを構築する際に、IMS に情報を提供します。ETO 記述子には、ログオン記述子、ユーザー記述子、MSC 記述子、および MFS 装置記述子の 4 つのタイプがあります。

IMS は以下の IMS.PROCLIB メンバーに記述子を保管します。

### DFSDSCMx

IMS システム 定義時に自動的に生成される記述子が入ります。DFSDSCMx の接尾部は、インストール・システムで IMSGEN システム 定義マクロの SUFFIX= パラメーターに指定した接尾部と一致します。

### DFSDSCTy

インストール・システムで作成したカスタマイズ済みの装置記述子が入ります。DFSDSCTy 内の記述子は DFSDSCMx 内の重複記述子をオーバーライドします。最後に定義した記述子を使用されます。

## ログオン記述子

ログオン記述子は、IMS が ETO 動的端末を構築するために使用するスケルトンです。これは、端末の物理的特性に関する情報を提供します。IMS は、ログオン記述子と一緒に出口ルーチンを使用して、端末構造を作成します。

ログオン記述子には、汎用、グループ、および特定という 3 つのタイプがあります。

### 汎用ログオン記述子

特定のタイプのすべての端末の特性を提供します。例えば、すべての SCS プリンターが、1 つの汎用記述子を共有します。同様に、すべての 3270 端末が 1 つの汎用記述子を共有します。

### グループ・ログオン記述子

互換性のあるハードウェア特性を持ち、すべてが IMS に同じ方法で定義される、端末の集合の特性を提供します。このような端末の実際の特性は通常同じですが、異なる場合もあります。IMS は、グループ記述子を使用してこのような端末の特性を入手します。

**例:** 自動ログオフ (ALOT) 時間の設定値だけが違う別のグループの端末に、別のログオン記述子を作成することができます。

### 特定ログオン記述子

は、単一の端末の特性を提供します。これらの特性はその端末にだけ適用されます。この場合、記述子名は記述子が記述する端末の名前と一致します。

**注:** ETO への実際のマイグレーション時には特定ログオン記述子を使用する必要がある場合もありますが、ETO にマイグレーションした後では、汎用またはグループ・ログオン記述子を使用するようにしてください。汎用またはグループ記述子の方が、ネットワークを管理しやすいからです。

## ユーザー記述子

ユーザー記述子は、ユーザー構造を構築するためのスケルトンです。ユーザー記述子はユーザー・オプションとキュー名を提供することができます。

## MSC 記述子

MSC 記述子は、リモート LTERM (ローカル IMS に存在していない LTERM) を作成するために使用されます。リモート LTERM の物理端末定義 (静的か動的) はリモート IMS にあります。

リモート LTERM の各 MSC 記述子は、IMS の初期設定の間にロードされて、出力を該当リモート LTERM に送るために使用する MSC リンクを IMS に伝えます。

## MFS 装置記述子

MFS 装置記述子を使用すると、IMS のシステム定義がなくても MFS フォーマット設定用の新しい装置特性を追加できます。MFSUCT ユーティリティ (DFSUTB00) は、MFS 装置記述子を使用して、MFS ライブラリー内のデフォルトの形式を更新します。

IMS は、MFS 装置特性テーブルを更新するためにも、MFS 装置記述子を使用します。IMS はこのテーブルを初期設定時のみロードします。したがって、更新は次の IMS 初期設定まで有効になりません。

### 関連概念

683 ページの『[複数システム結合機能の概要](#)』



複数システム 結合機能 (MSC) は、ある IMS に入力されたトランザクション を別の IMS で処理することを可能にします。

## ETO の概念

ETO の主な目的は IMS に端末を動的に定義することです。このトピックでは、ETO 端末とユーザー構造、記述子、および動的端末の出口ルーチンなどについて説明します。

### 構造が作成される時期と削除される時期

以下の状況のときに構造が作成されます。

- ログオン時
- サインオン時
- 出力がユーザーの LTERM のキューに入れられた時
- /ASSIGN コマンドを使用して、存在しないユーザーに LTERM が割り当てられるとき
- /ASSIGN コマンドを使用して、存在しない LTERM がユーザーに割り当てられるとき
- /CHANGE USER *username* AUTOLOGON コマンドが、存在しないユーザーに送信される時

すべての場合に、IMS は新しい構造を作成する前に既存の構造 (端末構造 またはユーザー構造) があるか検索します。

IMS は以下の順序でユーザー構造を作成し削除します (この順序は、端末のログオンとログオフ、およびユーザーのサインオンと サインオフにのみ適用されます。非同期出力がユーザーのキューに入れられた場合、IMS は必要に応じてユーザー構造を作成します)。

1. IMS と未定義端末の間にセッションを設定する場合、IMS は ログオン記述子を選択します。
2. ログオン記述子、カスタマイズ・デフォルト、および VTAM 情報を使用して、IMS は新しい端末について記述する VTAM 端末制御ブロックを構築します。
3. サインオンした時にユーザー構造が存在しない場合、IMS は、選択した ユーザー記述子からの情報を使用してユーザー構造を構築し、それを 端末構造に接続します。
4. IMS は、セッションを保持するのに端末またはユーザー構造が不要になった場合、それらの構造を削除します。ユーザー構造は一般的に、サインオフした場合、特別な状況を保持する必要がない場合、およびキューにメッセージがない場合に削除されます。IMS は、端末状況が存在しない場合 (例えば、トレース・モード)、ユーザーがサインオンしていない場合、および端末がアイドル状態の場合には、端末構造を削除します。

リソース・マネージャーおよびリソース構造を使用している場合、IMS は通常、ローカル制御ブロックの代わりにリソース構造の状況を保持します。したがって IMS はそれらの構造を削除します。

**制約事項:** 以下の 端末構造とユーザー構造は 削除されません。

- SRM=LOCAL の場合、SLU P 端末構造、金融機関端末構造、およびユーザー構造は、通常は IMS コールド・スタート時にしか削除されません。ただし、これらの構造は /CHANGE NODE COLDSESS コマンドを使用すれば、このコマンドの後の最初のチェックポイントで削除されます。
- SRM=LOCAL の場合、ISC 端末構造およびユーザー構造は、コールド・セッション終了後にしか削除されません。

## 記述子および出口ルーチン

記述子と出口ルーチンを使用すれば、ETO 動的端末に特性を割り当て、これらの端末に関連付けるユーザー構造を割り当てることができます。

記述子は動的端末に関する基本情報を提供します。出口ルーチンはこの情報を補足または変更します。記述子と出口ルーチンを使用する方法には、以下の 2 つのタイプがあります。

- 多くの記述子は、出口ルーチンの処理ロジックをほとんどあるいはまったく コーディングせずに使用することができます。

- 一部の記述子を使用するには、多くの処理を実行する 出口ルーチンをコーディングする必要があります。

## 記述子を作成し使用する方法

すべての記述子は IMS を始動する前の、IMS の初期設定時に作成されます。ユーザーは、ETO 機能サポートが必要であることを指定し、ETO 初期設定出口ルーチン (DFSINTX0) が ETO を使用不能にしないことを確認する必要があります。

IMS 初期設定時に、IMS はすべての ETO 記述子を読み取り、妥当性を検査します。IMS 初期設定は継続し、記述子は IMS が実行している期間ストレージに残ったままです。記述子に対して行うすべての変更は、IMS の次の初期設定後に有効になります。

IMS は記述子を使用して、端末構造とユーザー構造の両方を作成します。IMS は、適切であれば、IMS の再始動時に構造を再構築します。例えば、メッセージが構造のキューに入っている時に IMS がダウンした場合、この構造は IMS の再始動時に再構築されます。IMS は、このような構造を、IMS の再始動前と同じ構造になるように再構築します。IMS はこのような構造を再構築するために、記述子または 出口ルーチンを使用しません。したがって、記述子に行った変更は、IMS の再始動後に構築される新しい構造にだけ反映されます。IMS の再始動時に再構築される 構造には反映されません。

**例:** USERA が、ASOT=20 を指定する記述子 DESCA を使用してサインオンしたとします。USERA が IMS 会話を開始し、その後、IMS が異常終了したとします。システム・プログラマーは DESCA を ASOT=10 に変更します。IMS の再始動後、USERB が DESCA を使用してサインオンします。IMS の再始動時に、USERA は再構築されました。このような場合、USERA はまだ ASOT=20 ですが、USERB は ASOT=10 を持っています。

## ETO 使用の要約

次の図は、ETO の概念と ETO のインプリメンテーションの概要を示しています。

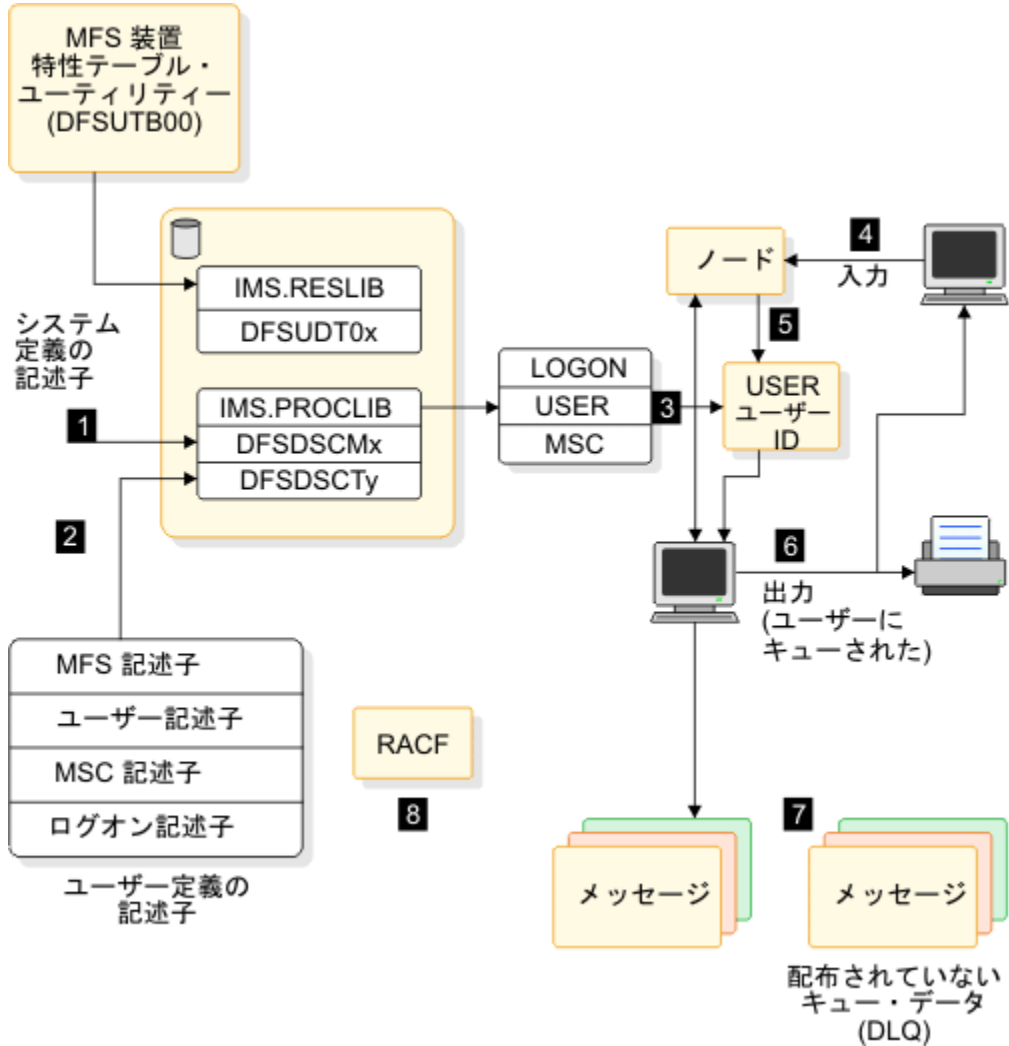


図 13. ETO 使用の要約

- **1** システム定義時に構築されるシステム定義記述子は、IMS.PROCLIB にメンバー DFSDSCMx として保管されます。
- **2** システム定義デフォルトをオーバーライドするために作成されたユーザー定義記述子は、IMS.PROCLIB にメンバー DFSDSCTy として保管されます。MFS 装置特性テーブル・ユーティリティ (DFSUTB00) によって処理された MFS 記述子は、装置特性テーブルに保管されます。
- **3** ログオン、ユーザー、および MSC の各記述子は、IMS の初期設定時に IMS.PROCLIB からの入力を使用してロードされます。
- **4** ログオン出口ルーチンとログオフ出口ルーチンは、ログオン時とログオフ時に呼び出されます。
- **5** サインオン出口ルーチンとサインオフ出口ルーチンは、サインオン時とサインオフ時に呼び出されます。
- **6** 出力は、サインオン時にユーザーが作成されていない限り、宛先作成出口ルーチンに指定されている宛先に送達されます。
- **7** IMS が出力の送達先を判別できない場合、メッセージは送達不能キューに追加されます。メッセージを送達できない理由には以下のものがあります。
  - ユーザー名が有効なユーザー ID ではない。
  - ユーザーのサインオンが拒否された。

- 装置と LTERM の間に物理 / 論理関係が存在していない。
- **8** ETO 環境では、RACF (または同等の製品) がセキュリティーを管理します。

## 第 5 章 拡張端末オプションの管理

IMS 拡張端末オプション (ETO) を使用すると、VTAM 端末とユーザーをシステム定義時に事前定義しなくても、IMS に動的に追加できるようになります。

### このタスクについて

以下のいずれかの時点で、ETO は端末セッション用のユーザー構造を動的に作成します。

- ユーザーが IMS にサインオンする時点
- 出力メッセージがユーザーに送信され、ユーザーによるリトリブを待機する時点
- **/ASSIGN** コマンドを使用して、存在しないユーザーに LTERM が割り当てられる時点
- **/ASSIGN** コマンドを使用して、存在しない LTERM がユーザーに割り当てられる時点
- **/CHANGE USER username AUTOLOGON** コマンドが、存在しないユーザーに送信される時点。

ETO を使用すると、以下に示す管理上の利点が得られます。

- システム定義のステージ 1 入力ストリーム用に以下のマクロをコーディングする必要がありません。

MSC NAME マクロ

VTAM マクロの TYPE、TERMINAL、NAME、VTAMPOOL、SUBPOOL

これらのマクロを取り除くことで、ネットワーク管理の複雑さを低減できます。

- 実行する必要があるシステム定義が少なくなります。
- 新規システム定義に対して予定されるシステムの停止が少なくなります。

ETO を使用すると、システム定義時に IMS に定義されていない端末やユーザーであっても、IMS を用いてセッションを設定できるようにすることができます。

また、実行時パラメーターと出口ルーチンを使用して、ETO が提供する機能のすべてまたは一部にユーザーがアクセスすることを許可することができます。

## ETO の計画

静的端末環境を ETO 環境にマイグレーションするには、計画が必要です。

### このタスクについて

引き続き、システム定義時に IMS に対して VTAM 端末と LTERM を定義することはできますが、その場合は次のようになります。

- 定義した端末用に存在する ETO 機能を利用できません。
- 端末を完全に定義する必要があります。すべての TERMINAL マクロ、NAME マクロ、およびパラメーターを提供するか、タイプ 2 CREATE コマンドを使用して情報を提供する必要があります。

ETO 端末は、VTAM 端末でなければなりません。

**制約事項:** 以下の VTAM 端末を ETO 端末にすることはできません。

- IMS マスター端末 (MTO)
- IMS 2 次マスター端末
- MSC 物理リンクと論理リンク
- 監視のため XRF によって使用される ISC セッション
- LU 6.2 端末 (APPC/IMS によって動的に作成され管理される)

## ユーザーの要件の識別

IMS システムにわたって ETO をインプリメントする程度は、ご使用のシステムの要件によって異なります。

### このタスクについて

は、システム内に静的 VTAM 端末が存在しない場合、および端末とユーザーの大部分がデフォルト・ログオン記述子とユーザー記述子を使用して定義されている場合に、完全にインプリメントされるものと考えられます。ただし、インストールによってアプリケーション・プログラムの従属関係が変わるため、ETO を完全インプリメントするコストも変わります。

インストール・システムで、以下の要件に基づいて、完全な ETO 機能サポートがどの程度まで必要かを判断する必要があります。

#### 完全なユーザー・メッセージのセキュリティー

完全なユーザー・メッセージのセキュリティーには、ETO の完全インプリメントが必要です。この環境においては、ノード名ユーザー記述子は存在しません。デフォルト・ユーザー構造が提供しないユーザー構造についての要件があれば、ユーザー記述子またはサインオン出口ルーチン (DFSSGNX0) によってそれらを定義する必要があります。

#### 動的端末サポートのみ

動的端末サポートに必要なのは、一部のインプリメンテーションだけです。ネットワーク定義ステートメントを、システム定義から ETO 記述子の PROCLIB メンバーに移すことができます。このインプリメンテーションには、次の利点があります。

- IMS ウォーム・スタート間で記述子を変更できるため、ネットワーク定義の維持に必要なシステム定義が削減できます。
- VTAM 端末ネットワークを定義する必要がないため、システム定義に必要な実行時間が短縮できます。
- 動的端末およびユーザー・リソースが、使用時にのみ割り振られるため、IMS チェックポイントおよび再始動のパフォーマンスが向上します。

ただし、部分的インプリメンテーションでは、各 LTERM は物理端末と固定関係を持っているため、より高いユーザー・メッセージのセキュリティーは達成されません。

## ETO の制約事項

ETO をインプリメントする前に、ETO には注意すべき少数の制約事項があります。

ETO の制約事項は、以下のとおりです。

- 端末関連の MSDB、または LTERM キーを持つ非端末関連の MSDB については、動的端末はサポートされません。
- 特定の LTERM 名を使用するアプリケーション・プログラムは、それぞれ特別な ETO カスタマイズを必要とする場合があります。

DFSUSER ユーザー記述子は、I/O PCB に格納されている LTERM 名に依存するアプリケーション・プログラム用に、ETO をカスタマイズするのに役立ちます。

#### 関連概念

90 ページの『DFSUSER ユーザー記述子の使用』

IMS、ユーザー ID と同じ名前のユーザー記述子、またはサインオンしている端末を検出できず、さらに出口ルーチンもそれを提供していない場合、IMS はデフォルト記述子として DFSUSER を使用します。

## 物理端末の定義

ETO をインプリメントする場合、希望する VTAM 端末ネットワークを実現するためには、ネットワーク内の物理端末の計画と定義を行うときに、特定の要件、および ETO と VTAM の連携のさまざまな側面に留意する必要があります。

### このタスクについて

以下のアクションを実行すると、希望する VTAM 端末ネットワークを確実に実現できます。

- IMS アプリケーション・プログラムが特定の端末特性に依存する頻度を査定します。
- 各 VTAM 端末定義の正確度を検査します。ETO は、各動的端末に対して、その端末の特性 (LU タイプ、画面サイズ、および型など) についての VTAM 定義を基にした端末構造を作成します。  
ユーザーの IMS システム定義で指定された端末特性は、VTAM 定義で指定された端末特性とは異なる (さらに変更される) 場合があります。IMS システム定義における端末特性が、実際の端末の特性と互換性がある場合、一見矛盾はないように見えます。<sup>4</sup>
- デフォルト・ログオン記述子を使用して適切に定義されていない端末に対して、特定のノード名ログオン記述子を提供するか、あるいは ログオン出口ルーチン (DFSLGNX0) を使用します。
- 次の 2 つの出口ルーチンのいずれかをコーディングして、装置タイプを判別します。
  - ログオン出口ルーチン (DFSLGNX0) は、デフォルトのログオン記述子を検査することによって装置タイプを判別することができます。
  - サインオン出口ルーチン (DFSSGNX0) は、端末制御ブロックを検査することによって、プロセスの後半で装置タイプを判別することができます。

端末セッションを設定する要求を受け取ったとき、IMS は、VTAM セッション・パラメーターからの以下の情報を基にします。

#### • UNITYPE

ログオンを試みているノードの装置タイプです。IMS は、以下のフィールドを使用して UNITYPE を判別します。

- CINIT 要求の LUTYPE フィールド<sup>5</sup>は、UNITYPE 情報の一部を提供するものです。LUTYPE 値は、通常、MODEENT マクロの PSERVIC オペランドの先頭バイトに入っています。このマクロは、ログオンするために使用される VTAM モード・テーブル項目の生成に用いられます。以下の図は、VTAM MODEENT マクロにおける PSERVIC オペランドのフィールドを示しています。

<sup>4</sup> 実際の端末特性が IMS 定義または VTAM 定義における端末特性と一致していない場合でも、その端末が IMS で作動することは可能です。

<sup>5</sup> CINIT は、システム・サービス制御点 (SSCP) から論理装置 (LU) に送信されたネットワーク・サービス要求で、他の LU を用いてセッションを確立し、そのセッションの 1 次側の役割を果たすよう LU に依頼するものです。

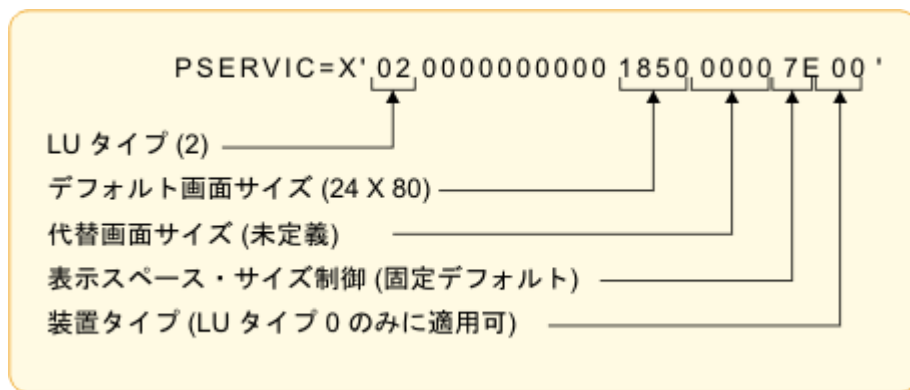


図 14. IMS が使用する VTAM MODEENT マクロ PSERVIC オペランドのフィールド

以下の表は、LUTYPE 値の IMS UNITYPE への マッピングを示しています。

表 2. VTAM LUTYPE 値の IMS UNITYPE への マッピング

VTAM LUTYPE	IMS UNITYPE
LUTYPE X'06'	LUTYPE6
LUTYPE X'02'	SLUTYPE2
LUTYPE X'01'	SLUTYPE1 (デフォルト) または NTO
LUTYPE X'00'	3270、SLUTYPEP、もしくは 3600/ 金融機関

- LUTYPE フィールドが X'00' である (その端末が 3270 非 SNA、金融機関、もしくは SLU P であることを示す) 場合、IMS は、CINIT 要求の TS フィールドに入っている伝送サービス・プロファイル指定を検査する必要があります。TS 値は、通常、VTAM モード・テーブルの生成に用いられる MODEENT マクロの TSPROF オペランドに入っています。この LUTYPE 値は、IMS がログオンのために選択したログオン記述子の値と一致していることが必要です。

以下の表は、TSPROF 指定の IMS UNITYPE へのマッピングを示しています。

表 3. TSPROF 指定の IMS UNITYPE へのマッピング

TSPROF 指定	IMS UNITYPE
X'02' または X'03'	3270
X'04'	SLUTYPEP (デフォルト) または 3600/ 金融機関

#### • 入力 RU サイズ

BIND 内の入力 RU (要求単位) サイズは、IMS の RECANY バッファ・サイズより小さいか等しくなければなりません (静的端末の場合にも必要です)。

#### • 出力 RU サイズ

BIND 内の出力 RU サイズは、選択されたログオン記述子から IMS が端末用に判別した OUTBUF サイズより大きい等しくなければなりません。静的端末には、このパラメーターも必須です。

#### • 画面サイズと型式番号

非 SNA 3270 装置および SLUTYPE2 装置の場合、IMS は、BIND から画面サイズ (行と列) と型式番号をリトリーブします。

- 静的端末の場合、画面サイズは、システム定義で指定されている 値になります。
- 動的端末の場合、IMS は、VTAM 定義またはログオン出口ルーチン (DFSLGNX0) から画面サイズを判別します。



**推奨事項:** ETO 機能が使用可能になるまで、IMS は BIND 内の画面サイズと型式番号値を無視していました。というのは、この情報は IMS システム定義に保持されているからです。そのため、この定義が正確であるか必ず検査してください。

VTAM 定義が不正確であると判別した場合、ログオン出口ルーチン (DFSLGNX0) を使用して、VTAM 提供の画面サイズと型式番号をオーバーライドすることができます。例えば、端末命名規則や MODETAB 定義規則を使用します。また、ログオン出口ルーチンは、適切ならば USER=NODE を名前として割り当てることもできます。

IMS は、MODEENT マクロの PSERVIC オペランドに入っている値を使用します。このマクロは、ログオンに使用される VTAM モード・テーブル項目を生成するのに用いられます。MFS の形式は、IMS が動的に作成するすべての画面サイズに使用できるものでなければなりません。

**制約事項:** IMS は、装置コントローラーから画面サイズを判別する際に、3270 区分読み取り照会 (RPQ) コマンドを使用しません。

ログオン出口ルーチン (DFSLGNX0) の詳細については、「IMS V15 出口ルーチン」を参照してください。

## 関連タスク

76 ページの『VTAM 装置タイプ、画面サイズ、およびモデルの識別』

VTAM ログオン CINIT ユーザー・データは、セッション制御ブロックを作成するための情報を IMS に提供します。この情報には、ログオン記述子、画面サイズ、モデル番号、および RU サイズが含まれます。

## 静的および動的の両端末の計画

静的端末と動的端末は、同じ IMS に共存できます。ただし、静的端末と動的端末の間をユーザーが移動する場合には、計画が必要となることがあります。

### このタスクについて

静的端末と動的端末の間をユーザーが移動する場合には、以下の状態に対する計画を立ててください。

- IMS が、静的端末と動的端末に対して別個のキューを保守している。静的端末が、IMS システム定義 (または LTERM を別の端末に移すための /ASSIGN コマンド) によって制御されているが、1 つ以上の LTERM がこの端末に関連付けられています。ユーザーの中には、出力キューを ETO 端末から ETO 端末へ次々と続けていくことに慣れてしまうユーザーもいます。静的端末は、この機能を備えていません。ユーザーは、静的端末と動的端末とを区別できるようにしておく必要があります。そうしないと、混乱する結果になるおそれがあります。
- 1 台の静的端末と 1 台の動的端末を指定すると、別々の IMS 会話が同時に存在可能となります。動的端末の会話はユーザー構造に関係があり、端末から端末へとユーザーの後に続きます。<sup>6</sup> 静的会話は、静的端末に属しており、静的端末にだけ解放されます。この状態は、ユーザーに都合のよいものであり、端末タイプ (静的か動的か) がユーザーに明確に分かっている場合にのみ予測できるものです。

通常、端末オペレーターは、DFS3650 (SESSION STATUS) メッセージの終わりに提供されるセキュリティ情報をチェックすることで、端末が静的であるか動的であるかを判別することができます。

### OUTPUT SECURITY AVAILABLE

その端末が動的であり、出力はサインオン ID に関連付けられていることを示します。

### NO OUTPUT SECURITY AVAILABLE

その端末は、静的に定義されているか、あるいは以下の 2 つの方式のいずれかを使用して ETO が作成したものであるかのいずれかであることを示します。

- ノード・ユーザー記述子を使用
- サインオン出口ルーチンを使用して、ノード名をユーザー構造に割り当てる

いずれの場合も、出力はユーザーにではなく、端末に関連付けられます。

**例外:** NOTERM オプションを使用すると、メッセージ DFS3650 が抑止されることがあります。

<sup>6</sup> これは、通常の場合と同様に、サインオン出口ルーチン (DFSSGNX0) が毎回同じユーザー名を設定していることを前提としています。

注: IMS PROCLIB データ・セットの DFSDCxxx メンバーに **STATICOUTSEC** パラメーターを指定することによって、静的に定義された VTAM 端末のトランザクション出力を保護することができます。この場合、出力はサインオン ID に関連付けられます。現在のユーザー ID がサインオン・ユーザー ID と一致しない場合、出力は廃棄されます。

推奨事項: マイグレーションを容易にし、起こり得る混乱を制限するために、編成内の論理的なグループ分け (部門や階など) を使用して、動的 ETO 端末に変換してください。

## 成長に備えた端末の定義

ETO のインプリメンテーションを設計する際、必ず、ネットワークの成長に備えた計画を行ってください。

### このタスクについて

#### 推奨事項:

- 将来の成長の可能性とシステム使用可能性を増すためにも、記述子の使用を最小限に抑えてください。ログオン、サインオン、および宛先作成出口ルーチンは、十分なカスタマイズを行うようにしてください。そうすることによって、特定の端末タイプに必要な以上の固有な記述子が不要になります。
- ユーザー・データが正しく処理されるように、出口ルーチンの設計は注意深く行ってください。特に、ログオン・プロセス中に指定されるユーザー・データの計画には気を付けてください。出口ルーチンは、ユーザー・データが指定されているかどうかに関係なく、正しく動作することが必要です。

金融機関端末および SLU P 端末など、端末タイプによっては、サインオン情報を指定するユーザー・データを必要とするものがあります。このデータがない場合には、同等の情報をログオン出口ルーチン (DFSLGNX0) で提供する必要があります。

#### 関連タスク

76 ページの『VTAM 装置タイプ、画面サイズ、およびモデルの識別』

VTAM ログオン CINIT ユーザー・データは、セッション制御ブロックを作成するための情報を IMS に提供します。この情報には、ログオン記述子、画面サイズ、モデル番号、および RU サイズが含まれます。

#### 関連資料

[トランザクション・マネージャー出口ルーチン \(出口ルーチン\)](#)

## VTAM 装置タイプ、画面サイズ、およびモデルの識別

VTAM ログオン CINIT ユーザー・データは、セッション制御ブロックを作成するための情報を IMS に提供します。この情報には、ログオン記述子、画面サイズ、モデル番号、および RU サイズが含まれます。

### このタスクについて

ログオン記述子内の情報から正しい端末関連制御ブロックが作成されるようにするために、この情報は正確でなければなりません。注意深く、VTAM PSERVIC パラメーターを定義して、IMS が適切なログオン記述子を選択して特定の端末特性の使用により画面サイズを設定するようにしてください。

BIND 内の入出力 RU サイズが IMS の RECANY サイズおよび記述子の OUTBUF サイズと互換性がない場合、BIND は拒否されます。

### 装置タイプの定義

定義が誤ってコーディングされると、IMS は間違った MFS 形式と装置特性を選択することになり、画面形式エラーを引き起こすおそれがあります。

IMS 動的端末制御ブロックは、以下のものを組み合わせた定義から作成されます。

- ログオン記述子
- ログオン時に VTAM から IMS に渡される情報
- MFS 装置特性テーブル

VTAM は、物理端末特性を IMS に渡します。IMS が装置タイプを決めるのに使用する情報は、以下のフィールドにあります。

- VTAM MODEENT マクロの VTAM PSERVIC パラメーターからの LUTYPE フィールド
- VTAM MODEENT マクロの TSPROF パラメーターからの TS プロファイル

## 非 SNA 3270 プリンターおよびディスプレイ

非 SNA 3270 プリンターとディスプレイのデフォルト 記述子名は DFS327P と DFS3270 です。

非 SNA 3270 装置を定義するのに必要な規則は、次のとおりです。

- VTAM MODEENT マクロの TSPROF パラメーターは 2 または 3 でなければなりません。
- VTAM MODEENT マクロの PSERVIC パラメーターでは、LU タイプ=0 を指定しなければなりません。
- プリンターをビデオと区別するように、VTAM PSERVIC パラメーターのバイト 12 を変更する必要があります。以下の表は、バイト 12 のビット・ロケーションを基にして各装置タイプを指定する内容を示します。

表 4. VTAM PSERVIC パラメーターのバイト 12 のビット

バイト 12 のビット	内容
0-1	装置タイプ
	<b>00</b> 指定解除の装置タイプ
	<b>01</b> プリンター装置
	<b>10</b> ディスプレイ装置
	<b>11</b> ディスプレイ/プリンター装置 (3275)
2 から 7	予約済み。

何も指定しなかった場合 (B'00') のデフォルトは B'10' で、これはディスプレイ装置です。<sup>7</sup>

**関連資料:** 非 SNA 3270 装置の定義の詳細については、「z/OS Communications Server: SNA Programming」を参照してください。

## LU タイプ 2 装置

以下のリストは、SLU-2 装置を定義するために必要な規則です。

- VTAM MODEENT マクロの TSPROF パラメーターは 3 でなければなりません。
- VTAM MODEENT マクロの PSERVIC パラメーターでは、LU タイプ=2 を指定しなければなりません。

**関連資料:** LU タイプ 2 装置の定義の詳細については、「z/OS Communications Server: SNA Programming」を参照してください。

## 3275 装置

VTAM 定義単独では、3275 装置のログオンを識別できません。追加情報 (UNIT=3275) をログオン記述子で指定する必要があります。CINIT ユーザー・データか、ログオン出口ルーチンにこの記述子を定義してください。記述子自身は、3275 装置タイプとして識別する必要があります。

静的 3275 装置は、IMS のシステム定義では以下のように定義されます。

```
TYPE          UNITYPE=3270 TERMINAL UNIT=3275,TYPE=3270-An,SIZE=(24x80),COMPT=PRT1
```

<sup>7</sup> B'00' は、ビット 0 と 1 が 00 に等しいことを意味します。B'10' は、ビット 0 とビット 1 が 10 に等しいことを意味します。

3275 装置は 1 つだけしかバッファを持っていません。このため、ディスプレイとプリンター・コンポーネントが同一モデルになります。静的に定義された、動的 3275 装置用の VTAM 定義は次のとおりです。

```
PSERVIC=X'000000000000185000007EC0'
```

## NTO 装置

以下の方法のいずれかで NTO 装置として端末を指定してください。

- BIND イメージの LU 表示サービス・プロファイルでは、LU1 を指定しなければなりません。
- LUTYPE が NTO のとき、IMS はデータ・ストリーム互換バイトを使用して、特定の NTO 装置タイプを正確に定義します。指定した LUTYPE が SLUTYPE1 であると、データ・ストリーム互換バイトは無視されます。端末はログオン記述子に定義されたように実際の SLUTYPE1 であると想定されます。
- ログオン出口ルーチンが利用可能な場合は、DFSNT0 (デフォルト) ログオン記述子を受け入れるか、そうでない場合は、ノード名か LOGOND ユーザー・データ・パラメーターを使用してインストール・システムで定義した NTO ログオン記述子を指定してください。
- CINIT のデータ・ストリーム互換バイトは以下の装置タイプを指定します。WTTY は、TTY NTO 装置を示します。何も指定されない場合、装置タイプは LUNS NTO となります。

## LU2 と非 SNA 3270 の画面サイズとモデル情報

ログオン出口ルーチンがログオンを許可すると、ログオン記述子からの BIND イメージ・データ (画面サイズ) と機能情報を使用して、適切な MFS 装置情報を得るために MFS 装置特性テーブルが検索されます。

モデルが画面サイズ制御バイトで指定された場合、MFS 装置特性テーブルは検索されません。画面サイズと機能に関して一致するものがないと、DFS3646 エラー・メッセージが出されます。その場合は、MFS 装置特性テーブルの適切な項目の情報がその装置に使用されます。MFS 装置特性テーブルの情報は、IMS システム定義もしくは MFSDCT ユーティリティ (DFSUTB00) により準備されます。

画面サイズ制御バイトが X'7F' で、デフォルトと代替の両方の画面サイズが指定された場合は、代替画面サイズを使用して MFS 装置特性テーブルの検索が開始されます。一致するものが見つからないと、デフォルト画面サイズを使用して再度検索されます。画面サイズが見つからないと、メッセージ DFS3646I がオペレーターに対して出されます。

画面サイズ (行と列の積) は 80 - 16384 の範囲です。行と列は、それぞれ 1 - 255 の範囲です。

ログオンしている 3270 装置の画面サイズと機能が複数の MFS 装置特性テーブル項目にマッピングされると、テーブルにある画面サイズと機能で最初に一致した項目が選択されます。

画面サイズ制御バイト (VTAM MODEENT の PSERVIC の 11 番目のバイト) が X'00' で、デフォルトと代替の両方の画面サイズが指定された場合は、デフォルトの画面サイズを使用して MFS 装置特性テーブルの検索が行われます。一致するものが見つからないと、代替画面サイズを使用して再度検索されます。

ログオンの間に装置の画面サイズまたはモデルをオーバーライドするには、ログオン出口ルーチン DFSLGNX0 を使用してください。

## LU2 画面サイズとモデル情報

LU タイプ 2 装置には、画面サイズとモデル情報が適用されます。

## このタスクについて

以下は、ETO 端末のモデルまたは画面サイズを決めるために IMS が使用するアルゴリズムです。

## 手順

1. 画面サイズは、VTAM PSERVIC にあるモデル仕様に基づいて設定されます。IMS は VTAM PSERVIC フィールドにある画面サイズ制御バイトを検査して、モデルが X'01'、X'02'、あるいは X'03' であるかを判別します。モデルは、その結果にしたがって設定されます。画面サイズ仕様は、モデルが指定されると無視されます。X'01' は、画面サイズが 12x40 のモデル 1 を表し、X'02' と X'03' は画面サイズが 24x80

のモデル 2 を表します。モデルが指定されると、MFS 装置特性テーブルでは MFS 装置情報の検索はされません。この機能はログオン記述子から持ってきます。

- 画面サイズは、VTAM PSERVIC にある画面サイズ仕様に基づいて設定されます。画面サイズ制御バイトに X'7E' の値が入っていると、VTAM PSERVIC にあるデフォルト画面サイズが用いられます。X'7F' の値が入っていると、MFS 装置特性テーブルを検索するのに、最初に代替画面サイズが用いられます。一致したデータが見つからないと、デフォルト画面サイズを用いて検索が行われます。最初に一致したデータが画面サイズになります。一致したデータが見つからない場合、メッセージ DFS3646I がオペレーターに送信されます。
- 画面サイズ制御バイトが X'00' の場合、VTAM PSERVIC にあるデフォルトと代替画面サイズ仕様を使用して、MFS 装置特性テーブルの MFS 情報を検索します。デフォルト・サイズが一致すると、デフォルトが使用されます。代替サイズが一致すると、代替が使用されます。一致したデータがないと、ログオンは拒否されます。
- 画面サイズ制御バイトが X'00' で画面サイズの指定がないと、装置はデフォルトのモデル 2 装置に、画面サイズは (24x80) に設定されます。
- 画面サイズは、デフォルト画面サイズ・フィールドの BIND パラメーターで設定されます。消去 / 書き込み (EW) コマンドが常に使用されます。

## タスクの結果

モデルの値を変更したい場合は、ログオン出口ルーチン (DFSLGNX0) を使用します。有効な値は X'01' と X'02' で、ログオン記述子でモデル 1 とモデル 2 に対応しています。VTAM PSERVIC の画面サイズとモデル仕様は無視されます。モデル=X'01' は 12x40 画面サイズを表します。モデル=X'02' は 24x80 画面サイズを表します。MFS 装置特性テーブルでは、MFS 装置情報の検索はされません。この機能はログオン記述子から持ってきます。

画面サイズの値を変更したい場合は、ログオン出口ルーチン DFSLGNX0 を使用してください。モデルを変更する場合は、画面サイズの変更は無視されますので気を付けてください。ログオン記述子の機能と画面サイズを使用して、MFS 装置特性テーブルを検索して MFS 装置情報を入手します。

## 非 SNA 3270 画面サイズとモデル情報

VTAM 3270 レコード・モード装置 (非 SNA 3270) には、モデル情報が適用されます

## このタスクについて

IMS は次のアルゴリズムを使用して、ETO 端末のモデルまたは画面サイズを判別します。

## 手順

- 画面サイズは、VTAM PSERVIC フィールドにあるモデル仕様に基づいて設定されます。IMS は VTAM PSERVIC フィールドにある画面サイズ制御バイトを検査して、モデルが X'01'、X'02'、あるいは X'03' であるかを判別します。モデルは、その結果にしたがって設定されます。画面サイズ仕様は、モデルが指定されると無視されます。X'01' は、画面サイズが 12x40 のモデル 1 を表し、X'02' と X'03' は画面サイズが 24x80 のモデル 2 を表します。モデルが指定されると、MFS 装置特性テーブルでは MFS 装置情報の検索はされません。この機能はログオン記述子から持ってきます。装置タイプと画面サイズは、モデルの値から判別されます。
- 画面サイズは、VTAM PSERVIC フィールドにある画面サイズ仕様に基づいて設定されます。画面サイズ制御バイトに X'7E' の値が入っていると、VTAM PSERVIC フィールドにあるデフォルト画面サイズが用いられます。X'7F' の値が入っていると、MFS 装置特性テーブルを検索するのに、最初に代替画面サイズが用いられます。一致したデータが見つからないと、デフォルト画面サイズを用いて検索が行われます。最初に一致したデータが画面サイズになります。一致したデータが見つからない場合、メッセージ DFS3646I がオペレーターに送信されます。装置タイプと画面サイズはモデルの値で暗黙指定されます。
- 画面サイズ制御バイトが X'00' の場合、VTAM PSERVIC フィールドにあるデフォルトと代替画面サイズ仕様を使用して、MFS 装置特性テーブルの MFS 情報を検索します。デフォルト・サイズが一致すると、デフォルトが使用されます。代替サイズが一致すると、代替が使用されます。一致するものが見つからないと、ログオンは拒否されます。装置タイプと画面サイズはモデルの値で暗黙指定されます。

4. モデル情報と画面サイズが VTAM PSERVIC フィールドに指定されていない場合、IMS は CINIT のモデル・バイトを使用します。モデル・タイプは、VTAM 定義ステートメント FEATUR2 によって設定されます。モデル情報は、画面サイズ制御バイトが X'00' で、画面サイズも X'00' である場合にのみ適用されます。このことは、プリンターおよびディスプレイに適用されます。X'00' はデフォルトです。これは、モデル 1 に対応します。X'01' はモデル 2 に対応します。装置タイプと画面サイズはモデルの値で暗黙指定されます。
5. 画面サイズによって、使用される書き込みコマンドのタイプが決まります。画面サイズが 960 に等しいか、1920 より大きいと、IMS は代替消去書き込み (EWA) を使用します。画面サイズが 1920 以下で、かつ 960 ではない場合、IMS は消去書き込み (EW) を使用します。装置タイプと画面サイズは、モデルの値から判別されます。

## タスクの結果

モデルの値を変更したい場合は、ログオン出口ルーチン (DFSLGNX0) を使用します。有効な値は X'01' と X'02' で、ログオン記述子でモデル 1 とモデル 2 に対応しています。VTAM PSERVIC の画面サイズとモデル仕様は無視されます。モデル=X'01' は 12x40 画面サイズを表します。モデル=X'02' は 24x80 画面サイズを表します。モデルが指定されると、MFS 装置特性テーブルでは MFS 装置情報の検索はされません。この機能はログオン記述子から持ってきます。装置タイプと画面サイズは、モデルの値から判別されます。

画面サイズの値を変更したい場合は、ログオン出口ルーチン (DFSLGNX0) を使用します。モデルを変更する場合は、画面サイズの変更は無視されますので気を付けてください。ログオン記述子の機能と画面サイズを使用して、MFS 装置特性テーブルを検索して MFS 装置情報を入手します。装置タイプと画面サイズはモデルの値で暗黙指定されます。

## 画面定義の例

次の例では、VTAM モード・テーブルの PESRVIC パラメーター、静的端末に対して同等の TERMINAL マクロは何か、静的および ETO 端末の両方で使用される、対応する MFS DEV ステートメントなどを示します。

- LU0 (非 SNA 3270 ビデオ)
  - VTAM モード・テーブル: PSERVIC=X'000000000000000000000000200'
  - TERMINAL マクロ: UNITYPE=3270 UNIT=3284/86 MODEL=2
  - MFS DEV ステートメント: TYPE=(3270P,2)
- モデル 2 非 SNA 3270 プリンター
  - VTAM モード・テーブル: PSERVIC=X'000000000000000000000000240'
  - TERMINAL マクロ: UNITYPE=3270 UNIT=3284/86 MODEL=2
  - MFS DEV ステートメント: TYPE=(3270P,2)
- 非 SNA 3270 ディスプレイ (モデル指定)
  - VTAM モード・テーブル: PSERVIC=X'00000000000000000000000080'
  - TERMINAL マクロ: UNITYPE=3270 MODEL=1
  - MFS DEV ステートメント: TYPE=(3270,1)

非 SNA 3270 のモデル情報は FEATUR2 パラメーターから提供されます。指定されない場合はモデル 1 (画面サイズ 12x40) です。FEATUR2=1 (指定またはデフォルト) が想定されます。

- 非 SNA 3270 ディスプレイ (モデル指定)
  - VTAM モード・テーブル: PSERVIC=X'00000000000000000000000080'
  - TERMINAL マクロ: UNITYPE=3270 MODEL=2
  - MFS DEV ステートメント: TYPE=(3270,2)

非 SNA 3270 のモデル情報は FEATUR2 パラメーターから提供されます。指定されない場合はモデル 2 (画面サイズ 24x80) です。FEATUR2=2 が想定されます。

- 非 SNA 3270 ディスプレイ (画面サイズ指定)

- VTAM モード・テーブル: PSERVIC=X'000000000000185000007E80'
- TERMINAL マクロ: UNITYPE=3270 TYPE=3270-A2,SIZE=(24,80)
- MFS DEV ステートメント: TYPE=3270-A2、ここで A2=24x80

SLU 2 装置では、UNITYPE=SLUTYPE2 が指定されている可能性があります (PSERVIC フィールドへの変更にも注意してください)。

画面サイズは、デフォルト画面サイズ・フィールドから提供されます (24x80)。(SLU 2 装置では、PSERVIC フィールドの先頭のバイトは X'02' です。最後のバイトは、X'00' になります。)

- 非 SNA 3270 ディスプレイ (画面サイズ指定)

- VTAM モード・テーブル: PSERVIC=X'000000000000205000000080'
- TERMINAL マクロ: NITYPE=3270 TYPE=3270-A3,SIZE=(32,80)
- MFS DEV ステートメント: TYPE=3270-A3、ここで A2=32x80

画面サイズは、デフォルト画面サイズ・フィールドから提供されます (32x80)。(SLU 2 装置では、PSERVIC フィールドの先頭のバイトは X'02' です。最後のバイトは、X'00' になります。)

- 非 SNA 3270 ディスプレイ (モデル指定)

- VTAM モード・テーブル: PSERVIC=X'00000000000000000000280'
- TERMINAL マクロ: UNITYPE=3270 MODEL=2
- MFS DEV ステートメント: TYPE=(3270,2)

これはモデル 2 非 SNA 3270 ディスプレイ (24x80) です。

- SNA 3270 ディスプレイ (モデル指定)

- VTAM モード・テーブル: PSERVIC=X'02000000000000000000280'
- TERMINAL マクロ: UNITYPE=3270 MODEL=2
- MFS DEV ステートメント: TYPE=(3270,2)

これはモデル 2 SNA 3270 ディスプレイ (24x80) です。

## ETO を用いた高セキュリティ環境の計画

ETO は、IMS システムのセキュリティを強化します。ETO のセキュリティ機能は、ご使用のシステムのニーズに応じてカスタマイズできます。

例えば、静的端末と ETO 動的端末の両方に適用される出口ルーチンをカスタマイズすることができます。

ETO セキュリティ機能を使用すると、以下のものを制御することができます。

- 端末から IMS への物理接続
- IMS へのユーザーのサインオン
- ユーザーまたはノードへの出力
- ユーザーに対するメッセージ・キューイング。以下の 2 つの方法でメッセージ・キューイングをカスタマイズすることができます。
  - サインオン時に指定するユーザーに自動的に LTERM を割り当てる。
  - 宛先作成出口ルーチン (DFSINSX0) を使用する。
- RACF (もしくは同等の SAF 準拠のセキュリティ製品) 使用による、コマンドおよびトランザクション・セキュリティ

### 静的端末と動的端末

システム定義時に静的端末を定義して、LTERM を特定の物理端末に関連付けます。端末のどのユーザーも、その端末にサインオンすることが必要かどうかに関係なく、その端末についてキューイングされている出力メッセージを受け取ることができます。

ETO の主な利点は、動的 LTERM (出力メッセージ・キュー) が端末とは切り離されて管理され、ユーザー名に割り当てられることです。ユーザーが出力を得られるのは、サインオン後だけです。この LTERM とユーザー間の動的な関連は、ユーザーがサインオフするまで維持され、IMS がユーザー構造を削除しなければサインオフ後も維持されます。

## MFS の計画

ETO 動的端末サポートをインプリメントすると、端末タイプの数と多様性はおそらく増加します。端末が IMS セッションを動的に設定すると、そのセッションを要求している装置タイプに MFS 形式を使用できない場合があります。

### このタスクについて

この問題は以下のように解決できます。

- アプリケーション・プログラムの MFS 定義に使用されている装置タイプだけに限定します。

**推奨事項:** ログオン出口ルーチン (DFSLGNX0) を使用して、希望するログオン記述子を選択してください。

- IMS に接続する新規端末タイプを含めるように MFS 装置出力形式を拡張します。

**推奨事項:**

- STACK ステートメントなどの MFS 生成ユーティリティー (MFSGU) 機能を使用して、追加形式の作成をより簡単にしてください。
- MFSDCT ユーティリティー (DFSUTB00) を使用して、生成の実行を防止してください。

多くの場合、複数の装置タイプはその機能が同じか、もしくは似ているので、その装置タイプの相違は、ユーザーの環境ではそれほど重要ではありません。IMS アプリケーション・プログラムでは、通常、装置タイプはそれほど厳密ではありませんが、それに依存することもあります。例えば、装置入力形式仕様の違いのため、MFS は同じ入力データ・ストリームから異なる入力メッセージを作成することが可能です。

## ユーザー ID の計画

ETO に使用するユーザー ID を計画するときは、いくつかのステップを実行する必要があります。

### このタスクについて

それらのステップは、以下のとおりです。

### 手順

- 動的ユーザー構造が、IMS に固有の名前を持っていることを確認します。固有のユーザー ID のサポートは、ユーザー・ベースのセキュリティーが重要であるかどうかによって異なります。ユーザー・ベースのセキュリティーの場合、ユーザー ID はユーザーに対して固有でなければなりません。固有でないと、ユーザー ID が端末 LU 名と同じになる可能性があります。
- ユーザーが、同時に同じユーザー ID を使用して複数の端末にサインオンする必要がある場合は、サインオン出口ルーチン (DFSSGNX0) をコーディングして、ユーザー ID 接尾部を付けます。SGN および RCF EXEC キーワード・パラメーターを検討し、選択します。
- アプリケーション・プログラムが入出力 PCB 内のユーザー・フィールドをどのように使用するかを分析します。



## ユーザー・キュー名の計画

ETO に使用するユーザー・キュー名を計画するときは、いくつかのステップを実行する必要があります。

### このタスクについて

ユーザー・キュー名を計画する場合は、以下を実行してください。

### 手順

- 動的ユーザー・キュー名が IMS 内で固有であることを確認します。固有のキュー名のサポートは、ユーザー・ベースのセキュリティが重要であるかどうかによって異なります。ユーザー・ベースのセキュリティの場合、ユーザー・キュー名はユーザーに対して固有でなければなりません。固有でない、ユーザー・キュー名が端末 LU 名と同じになる可能性があります。
- ユーザー・キュー名をユーザー名と同じにすることができない場合は、サインオン出口ルーチン (DFSSGNX0) でユーザー記述子を指定するか、もしくはロジックを提供します。
- アプリケーション・プログラムが入出力 PCB 内の LTERM フィールドをどのように使用するかを分析します。

## 運用の計画

運用を計画するときは、いくつかのステップを実行する必要があります。

### このタスクについて

運用を計画するときに実行する必要があるステップは、以下のとおりです。

- 動的リソースの概念を反映するように、MTO プロシージャーを更新します。MTO では、コマンド入力フィールドと応答形式に精通することが必要です。
- IMS 端末とユーザー・リソース構造、ならびにユーザーの問題を診断する際に必要なコマンドを反映するように、ヘルプ・デスク・プロシージャーを更新します。
- 必要ならば、自動化操作プログラム・プロシージャーを検討し、更新します。
- 送達不能キューを処理するためのプロシージャーを開発します。
- 自動サインオフおよび自動ログオフの要件を確認します。
- システム全般に及ぶ値を使用する条件、およびそれらの値を変更する条件についての標準を開発します。

## ETO を用いた MSC サポートの計画

静的システム定義マクロを使用して MSC 物理リンクと論理リンクを定義でき、また、タイプ 2 CREATE コマンドを使用してこれらのリンクを動的に作成することもできます。

### このタスクについて

MSC 記述子を使用して、リモート MSC NAME を IMS に指定することができます。記述子は、生成された MSNAME マクロのリンク・パス名に各リモート・リソースを結び付けます。

**推奨事項:** リモート・システムとローカル・システム間の整合性を維持するために、MSC 記述子を使用してリモート LTERM を定義してください。

MSC 記述子を適切に使用しなかった場合、MSC 検査ユーティリティー (DFSUMSV0) はリモート LTERM を認識しますが、ターゲット・システムにおける対応するローカル LTERM は認識しません。この場合、IMS は警告メッセージ DFS2331W を出します。

IMS は、IMS 内に定義されている MSC リンクに関連する MSC 記述子は処理しますが、他の MSC 記述子は無視します。リモート LTERM の単一ネットワーク定義は、MSC を使用する複数の相互接続された IMS システムの IMS.PROCLIB で保守できます。この場合は、各論理リンク・パス名はネットワーク全体を通して固有な名前 でなければなりません。

## ETO 記述子のコーディング

IMS は ETO 記述子を使用して、端末構造とユーザー構造を動的に構築します。

次の 4 タイプの ETO 記述子があります。

ログオン  
User  
MFS 装置  
MSC

すべての ETO 記述子の基本形式は次のとおりです。

### コラム

#### 説明

#### 1

以下の記述子タイプの 1 つ。

#### L

ログオン記述子

#### U

ユーザー記述子

#### D

MFS デバイス記述子 (MFS device descriptor)

#### M

MSC 記述子 (MSC descriptor)

#### \*

コメント行 (IMS は無視します)

#### 3-10

以下の規則を使用した、記述子の名前。

- 1 文字から 8 文字の長さの英数字でなければならない。
- ログオン記述子およびユーザー記述子の場合、使用できる文字は A-Z、#、\$、および @ である。
- MSC 記述子の場合にはリンク名を使用する。

#### 12-72

ブランクで区切られた 1 つ以上のキーワードとそのパラメーター。1 つのキーワードに対する複数のパラメーターを区切るには、コンマを使用します。

#### 73-80

オプションのシーケンス番号。IMS はこれを無視します。

### 関連概念

[ETO 記述子 \(システム定義\)](#)

## システム定義プロセスを用いた記述子の作成方法

マイグレーションを容易にするために、システム定義時に IMSCTRL マクロの ETOFEAT キーワードを用いて、ETO 記述子のスターター・セットを作成することができます (ただし、MFS 記述子は除きます)。追加の IMS.PROCLIB メンバーを作成することで、起動セットの追加あるいは変更を行うことができます。

### このタスクについて

MFS 記述子の場合、システム定義は、MFS DCT ユーティリティ (DFSUTB00) への入力の 1 つとして使用できるステージ 1 入力を基にして装置特性テーブルを作成します。装置特性テーブルには、システム定義の装置特性テーブルの項目に加え、装置記述子によって追加される項目も含まれます。

システム定義プロセスを用いて記述子を作成すると、作業時間は短くなり、記述子は正しいものになります。IMS は、ETO 記述子とそれらが表す IMS システム定義リソースとの関係についての情報を提供する ETO 記述子報告書を生成します。

## 関連概念

[IMS システムへの IMS ETO の組み込み \(システム定義\)](#)

## 関連資料

[IMSCTRL マクロ \(システム定義\)](#)

## 記述子の格納

システム定義時に作成された記述子は、IMS.PROCLIB メンバーの DFSDSCMx に格納されます。同じステージ 1 入力デッキの後続のシステム定義は、DFSDSCMx メンバーを上書きします。

### このタスクについて

**推奨事項:** メンバー DFSDSCMx を置き換えるときに記述子が失われるのを避けるために、TSO または z/OS ユーティリティを使用して、作成した記述子を IMS.PROCLIB メンバーの DFSDSCTy に保管してください。システム定義時に作成した DFSDSCMx 内の記述子を更新する必要がある場合は、TSO または z/OS ユーティリティ (IEBUPDTE など) を使用して更新してください。

## ログオン記述子の作成

ログオン記述子は、ログオン・セッションを設定した端末の物理的特性についての情報を IMS に提供します。これらの特性は、VTAM ログオン BIND 特性と一致している必要があります。

### このタスクについて

ここでは、ログオン記述子の作成方法と使用方法について説明します。

### システム定義時のログオン記述子の作成

IMS システム定義オプションを指定して ETO ログオン記述子を作成すると、IMS は、指定された固有の VTAM TERMINAL マクロごとに、1 つのログオン記述子を動的に作成します (各 TERMINAL マクロ定義ごとに、ETO へのマイグレーション・ステップとして使用するためのノード・ユーザー記述子も作成されます)。

### このタスクについて

IMS システム定義は、各装置タイプごとに最大 37 までの共通ログオン記述子を作成できます。そのタイプの最大数の端末を定義する記述子が、デフォルト・ログオン記述子となります。このログオン記述子は、そのタイプの IMS 定義デフォルト名の役割を果たします。そのタイプの他の端末は、デフォルト名の接尾部を使用して、独自の固有のログオン記述子を作成します。

37 個の共通記述子のいずれとも一致しない端末定義について、IMS は個別のログオン記述子を作成しません。

これらの記述子は、コメント (1 桁目にアスタリスクが入っている) として生成されます。例えば、\*L3270A のようになります。必要な記述子を選択する場合は、そのアスタリスクを削除してください。

システム定義プロセス時に作成される 37 個の共通ログオン記述子に対する命名規則は、次のとおりです。

- 名前の最後の文字は固有でなければなりません。
- ブランクが一般的で次に 0-9 と A-Z となります。

### 制約事項:

- システム定義時、1 次または 2 次のマスター端末、あるいは XRF ISC リンクとして定義される LU 6.1 端末に対しては、IMS は ETO ログオン記述子を作成しません。
- ログオン記述子では、PU、SIZE、MODEL、TYPE、MSGDEL の各キーワードはサポートされません。

## ログオン記述子選択の基準

ログオン記述子には、ノードと VTAM CINIT に関係するデータが含まれます。このデータは、IMS がセッションをサポートする制御ブロック構造を作成できるようにします。

IMS は下記の基準を (順序どおりに) 使用して、ログオン記述子を選択します。

1. IMS は、端末がある既存の制御ブロックを使用します。既存の制御ブロックを検出した場合には IMS は記述子を検索しません。
2. IMS は、ログオン記述子名を定義する際にログオン出口ルーチン が使用されているかどうか判別します。

VTAM 端末の場合、出口ルーチンは VTAM CINIT ユーザー・データ、または別の適切なアルゴリズムから名前を抽出します。VTAM CINIT データ内の LUTYPE および TS= (伝送サービス・レベル) フィールドは、選択された記述子と一致している必要があります。一致していない場合、ログオンは拒否されます。IMS は、無効なログオン記述子名を受け入れません。

3. セッション設定 (ログオン) 要求に答えて、IMS は LOGOND パラメーターを検査します。LOGOND パラメーターで、IMS が使用するログオン記述子を指示することができます。

LOGOND は、VTAM CINIT ユーザー・データ内のパラメーターです。LOGOND はまた、IMS /OPNDST コマンドのキーワードでもあります。

**制約事項:** LOGOND は、VTAM を使用する ISC 並列セッションには無効です。

4. IMS は、ログオン記述子名を検出できなかった場合、VTAM CINIT LUNAME と 同じ名前のログオン記述子を検索します。

**推奨事項:** 以下のいずれかの状態を希望する場合は、ログオン記述子でノード名を使用してください。

- 同じタイプの端末をこれ以上追加したくない。
- 端末タイプがそれほど多くない。
- ログオン出口ルーチンの論理を単純化したい。
- ISC に対して、デフォルト ISC ログオン記述子以外にパラメーター (OUTBUF= など) の指定を希望し、出口ルーチンをコーディングしていない。

この基準を使用しても有効なログオン記述子名が得られない場合、IMS はデフォルト基準を使用して記述子を選択します。

## デフォルトのログオン記述子の選択基準

IMS は各 VTAM 端末タイプにデフォルト・ログオン記述子を準備します。ログオン記述子名は、LU タイプと TS プロファイルを基にしています。

必要に応じて、ログオン記述子を追加することができます。ユーザーがログオン記述子を指定せず、IMS も IMS.PROCLIB またはログオン 出口ルーチン (DFSLGNX0) にログオン記述子を配置していない場合、IMS は以下の表に示すアルゴリズムを使用して、デフォルト・ログオン記述子名を割り当てます。各端末タイプごとに VTAM LUTYPE、IMS UNITYPE、および IMS デフォルト・ログオン 記述子名を示してあります。

デフォルト・ログオン記述子は以下の 3 つのうちの 1 つの方法で決定されます。

- IMS は、IMS システム 定義時に作成された最も一般的なログオン記述子を 検索し、それをデフォルトとして使用します。
- その端末タイプに対して端末定義が存在しない場合は、IMS がデフォルト・ログオン記述子を作成します。
- インストール先で、デフォルト・ログオン記述子を作成することができます。

表 5. VTAM LUTYPE、IMS UNITYPE、およびデフォルト・ログオン記述子名のマッピング

VTAM LUTYPE	IMS UNITYPE	デフォルト 記述子名
X'06'	LUTYPE6	DFSLU61

表 5. VTAM LUTYPE、IMS UNITYPE、およびデフォルト・ログオン記述子名のマッピング (続き)

VTAM LUTYPE	IMS UNITYPE	デフォルト記述子名
X'02'	SLUTYPE2	DFSSLU2
X'01'	SLUTYPE1 (デフォルト) <sup>87 ページの『1』</sup> NTO	DFSSLU1 DFSNT0
X'00' (TS = X'04')	SLUTYPEP (デフォルト) <sup>87 ページの『2』</sup> 金融機関 3601	DFSSLUP DFSFIN DFSFIN
X'00' (TS = X'02' または X'03')	3270 3270, UNIT=3284	DFS3270 DFS327P

#### 表に関する注:

- IMS は、SLUTYPE1 端末と NTO 端末とを区別しないため、SLUTYPE1 がデフォルトと想定されます。SLUTYPE1 と NTO の両方の端末をサポートするためには、次のようにしてください。
  - ログオン出口ルーチン (DFSLGNX0) を使用するか、もしくは LOGOND パラメーターを指定して、DFSSLU1 デフォルト・ログオン記述子を変更します。
  - ユーザーのインストール先に SLUTYPE1 端末がない場合は、DFSNT0 ログオン記述子を DFSSLU1 に名前変更して、これを LU タイプ X'01' 端末に対するデフォルトにします。
- IMS は、SLU タイプ P、金融機関、および 3601 の端末を区別しないため、SLUTYPEP がデフォルトと想定されます。ETO は、金融機関および 3600 シリーズの端末を同じであると見なします。SLUTYPEP と 3600/ 金融機関の両方の端末をサポートするためには、次のようにしてください。
  - ログオン出口ルーチン (DFSLGNX0) を使用するか、もしくは LOGOND パラメーターを指定して、DFSSLUP デフォルト・ログオン記述子を変更します。
  - ユーザーのインストール先に SLUTYPEP 端末がない場合は、DFSFIN ログオン記述子を DFSSLUP に名前変更して、これを LU タイプ X'00' (TS = X'04') 端末に対するデフォルトにします。

### NTO 端末、3600/ 金融機関端末の使用

CINIT 情報が矛盾していると、IMS はデフォルト・ログオン記述子名として DFSNT0 または DFSFIN を生成することができません。SLU 1 に対する CINIT パラメーターは NTO と同一で、SLU P は 3600/ 金融機関と同一です。

#### このタスクについて

**推奨事項:** 3600/ 金融機関および NTO 端末タイプの場合には、以下の処置を行ってください。

- ログオン出口ルーチン (DFSLGNX0) で、常に適切なログオン記述子名を提供します。この名前は、LU 名、LU タイプ、CINIT ユーザー・データを基にした定数として生成することができます。
- ログオン記述子名が、必ず BIND ユーザー・データから LOGOND パラメーターとして提供されるようにします。
- ログオンしている端末のノード名と一致するログオン記述子が存在するようにします。IMS がデフォルト・ログオン記述子を検出できないと、ログオンの試行は失敗します

## XRF を使用した ETO 端末のリカバリー

XRF 環境で ETO 端末を回復するには、ログオン記述子の指定時に BACKUP= パラメーターを使用します。

### このタスクについて

BACKUP= パラメーターは、セッション切り替えの順序を制御する優先順位を設定します。

XRF 環境において、3600/ 金融機関および SLU-P 端末がクラス 2 として定義されている場合には、テークオーバー時に再ログオンおよび再サインオンが自動的に行われます。

**関連資料:** XRF の一般情報については、「IMS V15 システム管理」を参照してください。

## ユーザー記述子の作成

ユーザー記述子はユーザー・オプションとユーザー構造に関連している情報を提供します。IMS は、ユーザーが ETO 端末を使用できるようにする制御ブロック作成のためにユーザー記述子を必要とします。

### このタスクについて

ユーザー記述子には、次の 3 つのタイプがあります。

- インストール時作成
- ノード・ユーザー
- DFSUSER

IMS は、上記のシーケンスで、最初の有効な記述子を選択します。IMS は、システム定義オプションを使用して、DFSUSER およびノード・ユーザー記述子を作成します。

ここでは、ユーザー記述子の作成方法について説明します。

## システム定義中のユーザー記述子の作成

ユーザー記述子は、各 VTAM TERMINAL マクロまたは VTAMPOOL SUBPOOL マクロから生成されます。

システム定義時に VTAM TERMINAL マクロを使用して作成されたユーザー記述子は、その端末と同じ名前になります。VTAMPOOL SUBPOOL マクロを使用して作成されたユーザー記述子は、そのサブプールと同じ名前になります。

SUBPOOL マクロを使用して作成されたユーザー記述子に対しては、応答オプション (TRANSRESP、NORESP、FORCRESP) を設定できません。それは、このオプションが静的定義用に TERMINAL マクロで定義されているためです。SUBPOOL マクロを使用して作成されたユーザー記述子には、インストール・システムに適切な応答オプションを追加することが必要です。

## ユーザー記述子選択の基準

ユーザー構造が存在しない場合には、ユーザーがサインオンすると、IMS はユーザー構造を構築するためのユーザー記述子を選択します。

IMS は、以下の基準に従って、ユーザー記述子を選択します。

1. インストール先作成の出口ルーチンは、ユーザー記述子 (ユーザー ID、ノード名、または DFSUSER) の名前を選択できます。
2. IMS は、ログオン時に提供されている USERD を検索します。IMS は、/SIGN コマンドまたは /OPNDST コマンドの USER 記述子フィールドに指定されている記述子名 (ユーザー ID、ノード名、または DFSUSER) も検索します。
3. IMS はユーザー ID と同じ名前の記述子を検索します (インストール先作成ユーザー記述子のみ)。IMS がその記述子を検出し、その記述子に LTERM キーワードが指定されていない場合、IMS はユーザー構造と 1 つの LTERM (どちらもユーザー ID と同じ名前を持ちます) を作成します。
4. IMS は、VTAM ノードと同じ名前の記述子 (ノード・ユーザー記述子) を検索します。IMS がその記述子を検出し、その記述子に LTERM キーワードが指定されていない場合、IMS はユーザー構造と 1 つの LTERM (どちらも VTAM ノードと同じ名前を持ちます) を作成します。ただし、このユーザー構造には出

カセキュリティは関連付けられていません。端末にサインオンするユーザーはすべて、その端末についてキューイングされているメッセージを受信することができます。

5. IMS は、デフォルト・ユーザー記述子の DFSUSER を選択します。IMS は、ユーザー構造と 1 つの LTERM (どちらもユーザー ID と同じ名前を持ちます) を作成します。

## インストール先作成ユーザー記述子の使用

ユーザー・サイトの重要な基準に適合する、独自のインストール先特有のユーザー記述子を作成することができます。

インストール先作成ユーザー記述子の名前はユーザー ID と同じです。これらのユーザー記述子には、ノード・ユーザー記述子や DFSUSER 記述子では提供されない値を追加することができます。

## ノード・ユーザー記述子の使用

ノード・ユーザー記述子は、静的端末定義から ETO 動的端末定義へのマイグレーションを援助します。

IMS システム定義時、ノード・ユーザー記述子はオプションとして作成されます。ノード・ユーザー記述子は、記述子選択を実行する出口ルーチンがまだ完了していない場合に役立ちます。

ノード・ユーザー記述子は、関連する端末と同じ名前であればなりません。そのため、IMS は、システム定義内にあるユーザー・オプションとユーザー構造名を保存する固有なノード・ユーザー記述子を作成します。IMS システム定義が ETO 記述子を作成すると、VTAM TERMINAL マクロか VTAMPOOL SUBPOOL マクロを持つ端末ごとに、ノード・ユーザー記述子が作成されます。ノード・ユーザー記述子は、DFSUSER オプションと一致した場合でも作成されます (並列セッション・サポート用に定義された ISC 端末の記述子は作成されません)。

ノード・ユーザー記述子の LTERM パラメーターが、静的に定義された LTERM と同じ名前である場合、ノード・ユーザー記述子は無視されます。IMS はメッセージ DFS3673W を出します。そのため、システム定義時に作成されるノード・ユーザー記述子はすべて、コメント・ステートメントとして生成されます。ノード・ユーザー記述子を使用する場合は、アスタリスクを削除してください。

ほとんどの場合、ノード・ユーザー記述子は必要ありません。ノード・ユーザー記述子が必要になるのは、デフォルト・ユーザー記述子 (DFSUSER) またはサインオン出口ルーチン (DFSSGNX0) が必要なユーザー・オプションを提供できない場合だけです。TSO または z/OS ユーティリティを使用すると、システム定義時に作成されるノード・ユーザー記述子のほとんどを廃棄することができます。

ノード・ユーザー記述子を使用した場合は、IMS に接続する端末を事前定義する必要がなくなるという ETO 機能は利用できないことを理解しておいてください。さらに、ノード・ユーザー記述子を使用すると、各ユーザーの出力セキュリティも失われます。

### 推奨事項:

- IMS の連続可用性が重要な要素である場合には、可能な限りノード・ユーザー記述子を取り除いてください。ノード・ユーザー記述子を使用して新規端末を追加するには IMS の再始動が必要となるため、ノード・ユーザー記述子の代わりに出口ルーチンを使用してください。
- ノード・ユーザー記述子の使用を避けるために、サインオン出口ルーチン (DFSSGNX0) を使用して、ユーザー名としてノード名を使用するユーザー構造を IMS に作成させてください。この方法は、ノード・ユーザー記述子を使用するのと似ていますが、IMS 初期設定時に、多数のこれらの記述子が事前定義されるのを回避します。
- /OPNDST、LOGON、または自動ログオンを使用すると、出力が利用可能な場合に、ノード・ユーザー記述子を使用してセッションを開始することができます。自動ログオンを使用するには、TERMINAL マクロで SHARE オプションを指定してください。
- サインオン出口ルーチン (DFSSGNX0) を使用してデフォルト・ユーザー記述子を選択すると、IMS にユーザー構造名のノード名を使用してユーザー構造を作成するよう指示するビットを設定できます。これはノード・ユーザー記述子を選択する方法と同じです。各端末ごとにノード・ユーザー記述子を定義するのではなく、サインオン出口ルーチンを使用してノード名構造を作成することにより、大規模ネットワーク・インストールには複雑さが軽減される利点があります。



## DFSUSER ユーザー記述子の使用

IMS、ユーザー ID と同じ名前のユーザー記述子、またはサインオンしている端末を検出できず、さらに出口ルーチンもそれを提供していない場合、IMS はデフォルト記述子として DFSUSER を使用します。

IMS は、サインオンとアプリケーション・プログラム出力処理の両方に DFSUSER を使用します。また DFSUSER を使用すると、IMS は、これ以外のユーザー記述子が利用不可能な場合に、サインオン要求のためのユーザー構造を動的に作成することもできます。

ETO 記述子を作成するシステム定義オプションを指定すると、IMS は DFSUSER を作成します。DFSUSER には、IMS システム定義で指定される頻度をもっとも高いオプションが入っているため、ほとんどのユーザーの要求は満たせるはずですが、ただし、DFSUSER 記述子の使用は、これらのオプションのすべてを使用する必要があるということではありません。DFSUSER を使用して作成された構造を変更するためのサインオン出口ルーチン (DFSSGNX0) をコーディングすることができます。

DFSUSER は、/SIGN コマンドで指定されたユーザー ID と同じ名前のユーザー構造を構築します。IMS は、ユーザー ID と同じ名前のこのユーザー構造に 1 つの LTERM を割り振ります。必要に応じて、サインオン出口ルーチン (DFSSGNX0) を使用して、複数のキューを提供することができます。

**推奨事項:** ETO の完全インプリメントには、DFSUSER を使用してほとんどのユーザー・リソースを作成することが含まれます。ETO へマイグレーション後は、可能な限り多くのユーザーに DFSUSER 記述子を使用してください。記述子の数を最小限にすることで、IMS ネットワークの管理に要するワークロードが削減されます。

ユーザー構造名はユーザー ID になります。DFSUSER 記述子を使用すると、サインオン出口ルーチン (DFSSGNX0) を使用して、ユーザー構造名およびその他のオプションを変更することができます。

アプリケーション・プログラムに、入出力 PCB 内にある LTERM 名への依存要素がある場合には、以下の 2 つの方法のいずれかを使用して ETO をカスタマイズすることができます。

- 特定のアプリケーション・プログラムを使用する予定のユーザーごとに、ユーザー記述子を用意する。
- サインオン出口ルーチン (DFSSGNX0) を使用して、ユーザー集団のサブセットに対するデフォルト・アクションを調整する。

サインオン出口ルーチン (DFSSGNX0) および宛先作成出口ルーチン (DFSINSX0) を使用すると、指定されたユーザー記述子に LTERM を作成するためのデータが入っていない場合でも、ユーザー構造を動的に作成することができます。

### 関連資料

[サインオン出口ルーチン \(DFSSGNX0\) \(出口ルーチン\)](#)

[宛先作成出口ルーチン \(DFSINSX0\) \(出口ルーチン\)](#)

## MFS 装置記述子の作成

MFS 装置記述子を使用すると IMS システム定義時に生成されない画面サイズと機能の組み合わせを定義することができます。

### このタスクについて

MFS DCT ユーティリティ (DFSUTB00) は、MFS 装置記述子を使用して、MFS 装置特性テーブル内の装置タイプ、画面サイズ、および機能を更新します。また、MFS DCT は MFS 装置記述子を使用して、システム定義を変更することなしに新しい MFS デフォルト形式を生成します。MFS 装置記述子はシステム定義の一部として作成されるものではなく、IMS システム定義後のオプション・ステップです。

MFS 装置記述子は、静的端末とは異なる動的端末の端末特性を定義します。出口ルーチンを使用して画面サイズを変更していない場合には、静的端末用に定義されている画面サイズとは異なる ETO 端末の画面サイズがすべて、MFS 装置記述子を使用して定義されていることを確認する必要があります。

**推奨事項:** 装置特性が異なる端末にログオンするユーザーがいる場合には、MFS 形式の拡張セットを作成する必要があります。非発行元端末に送達される入力メッセージおよび出力メッセージは、その装置用に定義されている MFS フォーマット設定に従って端末にフォーマット設定されます。



## 装置特性テーブルの構築

IMS に接続されている ETO 3270 または SLU-2 端末にログオンすると、通常、画面サイズは VTAM CINIT PSERVIC データで IMS に提供されます。時には、型式番号も一緒に指定されます。

### このタスクについて

IMS は、ログオン記述子で指定されたこの画面サイズと機能を使用して、1 つ以上の項目が入っている MFS 装置特性テーブルを検索します。各項目は、以下のものを指定します。

MFS 装置タイプ  
画面サイズ  
機能

テーブルを検索すると、MFS 装置タイプを検出することができます。検索が失敗すると、ログオンの試行は受け入れられず、メッセージ DFS3646I と DFS3672I が出力されます。

MFS 装置特性テーブルは、次の 2 つの方法のいずれかで構築されます。

- IMS システム定義プロセスが、TERMINAL マクロの TYPE=、SIZE=、および FEATURE= 仕様を基にして MFS 装置特性テーブルを構築します。それぞれ固有な組み合わせが MFS 装置特性テーブルの項目にあります。
- MFSDCT ユーティリティ (DFSUTB00) が、MFS 装置記述子の入力を基にして MFS 装置特性テーブルを構築、あるいは変更します。これらの記述子は TYPE=、SIZE=、および FEATURE= を持っており、その仕様は TERMINAL マクロが持っています。VTAM CINIT PSERVIC が画面サイズではなく型式番号を指示している場合には、MFS 装置特性テーブルの検索は行われません。型式番号が、画面サイズと MFS 装置タイプを指示します。

VTAM CINIT PSERVIC フィールドに入っているものが画面サイズではなく型式番号である場合には、MFS 装置特性テーブルの検索は行われません。

**関連資料:** MFSDCT ユーティリティ (DFSUTB00) の使用方法については、「IMS V15 システム・ユーティリティ」を参照してください。

## MFSDCT ユーティリティ (DFSUTB00) の使用

IMS は特定の状態にある場合に限り、セッション開始時に装置特性テーブルを検索して、MFS 装置情報を見つけます。

### このタスクについて

IMS がセッション開始時に装置特性テーブルを検索するのは、以下の環境です。

- 画面サイズが変更されている。
- PSERVIC のモデル値が X'00'、X'7E'、X'7F' のいずれかである。

MFSDCT ユーティリティ (DFSUTB00) を使用すると、IMS システム定義時に定義されていない画面サイズを定義することができます。MFSDCT ユーティリティは、IMS システム定義を実行せずに、PROCLIB メンバー DFSDSCMx および DFSDSCTy 内の MFS 装置記述子を使用します。新しい画面サイズ定義は、すでに定義されている画面サイズ定義に追加されます。

MFS ユーティリティは、以下を実行します。

- IMS.PROCLIB メンバーの DFSDSCMx および DFSDSCTy から、装置記述子を読み取ります。
- 各新規装置記述子ごとに 1 つの装置記述子テーブル項目ステートメントを構築します。
- 記述子が指定されていない場合は、終了します。
- オプションで、IMS.SDFSRESL から既存の装置特性テーブルをロードし、既存の各テーブル項目ごとに 1 つの項目ステートメントを構築します。
- 入力として、テーブル項目ステートメントと DCTBLD マクロおよび MFSINIT マクロをアセンブラーに渡します。

- デフォルト MFS 形式定義の新規セットのほかに、新規または更新済み 装置特性テーブルとしてアセンブラ出力を準備します。その後の処理のために、出力は別個のファイルに入れます。

MFSDCT ユーティリティーのプロシージャは、IMS.PROCLIB. 内にあります。ユーザーは、画面サイズ要件を評価して、その要件に適合するように MFS 装置記述子をコーディングする必要があります。MFSDCT ユーティリティーは、インストール先で必要となる可能性のあるすべての画面サイズと機能の組み合わせを生成しなければなりません。

MFSDCT ユーティリティーを使用する場合は、以下を実行してください。

## 手順

1. DFSUTB00 (MFSDCT ユーティリティー) を実行します。
2. 新規装置特性テーブルをアセンブルします。
3. 新規装置特性テーブルを IMS.SDFSRESL にリンク・エディットします。
4. MFS 言語ユーティリティーのフェーズ 1 を実行して、新規の デフォルト MFS 形式制御ブロックを生成します。
5. MFS 言語ユーティリティーのフェーズ 2 を実行すると、新規のデフォルト MFS 形式が IMS.FORMAT にロードされます。

## タスクの結果

装置特性テーブルの問題は、多くの場合、ログオン処理時にエラー・メッセージ DFS3646I および DFS3672I によって示されます。

### 関連資料

[MFS 装置特性テーブル・ユーティリティー \(DFSUTB00\) \(システム・ユーティリティー\)](#)

## MSC 記述子の作成

MSC 記述子は、リモート LTERM を、静的に定義された MSC リンクと関係付けます。

### このタスクについて

ETO 端末記述子を作成するために IMS システム定義オプションを指定すると、IMS は MSC 記述子を作成します。MSC 記述子は、定義済み MSC リンクにリモート NAME マクロを関係付けます。

**推奨事項:** MSC 記述子を使用して、リモート IMS システム内のすべての LTERM を定義してください。定義後、IMS ターゲット・システムの MSC 検査ユーティリティーは、リモート LTERM を対応するローカル LTERM に関連付けます。対応するローカル LTERM が検出されないと、IMS はメッセージ DFS2331W を出します。

IMS は、初期設定されているシステム内で定義されている MSC リンクに関連付けられている MSC 記述子を処理するだけです。IMS は、その他の MSC 記述子はすべて無視します。各論理リンク・パス名がネットワーク内で固有であると、ネットワークの MSC 定義ソースを 1 つの PROCLIB メンバー IMS.PROCLIB 内に保持することができます。

## 出口ルーチン

IMS に添付されているさまざまな出口ルーチンを使用して、ETO をカスタマイズできます。

以下の出口ルーチンを使用すると、ETO をカスタマイズすることができます。

- 初期設定出口ルーチン (DFSINTX0)
- ログオン出口ルーチン (DFSLGNX0)
- ログオフ出口ルーチン (DFSLGFX0)
- サインオン出口ルーチン (DFSSGNX0)
- サインオフ出口ルーチン (DFSSGFX0)

- 宛先作成出口ルーチン (DFSINSX0)
- グリーティング・メッセージ出口ルーチン (DFSGMSG0)

ETO が使用可能ならば、これらの出口ルーチンは IMS の 初期設定時にロードされます。すべての非 MSC と非 LU 6.2 VTAM 端末 (静的または動的) はこれらの 出口ルーチンを使用できます。

以下のいずれかの目的のために、広範な処理論理をもってこれらの ETO 出口ルーチンをコーディングすることができます。

- 命名規則を実施する。
- ユーザー・キューを選択する。
- 画面サイズなどの端末特性を変更する。
- IMS へのアクセスを制限することでセキュリティーを増す。
- RACF などのセキュリティー製品を用いずにサインオンできるようにすることにより、セキュリティーを変更する。
- タイムアウト値などの稼働パラメーターを選択する。

#### 関連資料

[トランザクション・マネージャー出口ルーチン \(出口ルーチン\)](#)

## ETO 開始

IMS システムに ETO を組み込むには、IMS または DCC 始動プロシージャーで ETO=Y を指定してください。

### このタスクについて

始動プロシージャーのデフォルト ETO= キーワードは、ETO=N です。ETO=N を指定したままにすると、IMS は、未定義端末に対するセッションの確立要求をリジェクトします。存在していないキューへのメッセージは拒否され、IMS はメッセージ DFS064 もしくは A1 状況コードを出します。

IMS は初期設定時に、初期設定出口ルーチン (DFSINTX0) を呼び出します。この出口ルーチンをコーディングして、ETO を使用禁止にしたり、ETO に使用させたいデータの作成およびロードを行うことができます。IMS は、このデータへのポインターを保守して、このポインターを入力パラメーターとして ETO 出口ルーチンに渡します。IMS は、システム目録ディレクトリー (SCD) の SCDINTXP フィールドを通じて、このポインターを非 ETO 出口ルーチンに渡します。

ETO を使用可能にすると、IMS は各 ETO 記述子の妥当性検査を行います。

- 記述子が正しくコーディングされていないと、IMS はその記述子を見捨て、メッセージ DFS3640W を MTO に出します。
- IMS は、有効な記述子の中に無効なパラメーターを検出すると、デフォルト・パラメーターに置換し、メッセージ DFS3641W を出して処理を続けます。
- すべての記述子を (例えば、読み取りエラーのため) 読み取れない場合、IMS は ABEND U0015 で異常終了します。1つの有効ログオン記述子と1つの有効なユーザー記述子がある場合には、IMS は異常終了しません。
- IMS は、利用できる記述子情報が不完全であるシステムを始動しません。

IMS の初期設定時前であれば、記述子、出口ルーチン、および MFS 装置特性 テーブルは、追加、削除、あるいは更新することができます。しかし、セッションの制御ブロックが再始動した時に存在している場合は、制御ブロックが削除されるまで、それを構築する記述子の変更は効力がありません。例えば、制御ブロックの作成後に元の記述子を変更または削除されても、3600/ 金融機関、SLU-P、および ISC セッションは、もとの記述子から作成された制御ブロックを用いてウォーム・スタートできます。

## ETO 端末へのログオン

端末には、次の3つの方法でログオンできます。

### このタスクについて

以下を使用することができます。

- IMS /OPNDST コマンド、および任意でサインオン・データを組み込みます。
- SNA コマンドの **INITSELF**、**INITOTHER**、あるいは **USS LOGON**。さらに任意でサインオン用のユーザー・データを組み込むこともできます。
- ユーザー記述子や出口ルーチンを基にしたユーザー・データを含む ETO 自動ログオン。そしてこのユーザー・データはサインオンに渡されます。

ETO=Y を指定すると、下記の各事項が真であれば、ETO 端末とのセッションを設定することができます。

- セッション要求の受け入れに必要な制御ブロックを作成できるだけの十分な情報を提供するログオン記述子が利用可能である。
- 必須のログオン記述子に加えて、3270 または SLU-2 端末には、その画面サイズと機能が一致する MFS 装置特性テーブルの項目が必要である。ただし、3270 または SLU-2 端末がモデル 1 または 2 として識別される場合は、IMS は MFS 装置特性テーブルを検索しません。
- 画面サイズ制御バイトが、VTAM MODEENT マクロの PSERVIC フィールドで X'7F' である場合、IMS は代替画面サイズを使用して MFS 装置特性テーブルを検索する。一致するものが見つからないと、IMS はデフォルト画面サイズを使用して MFS 装置特性テーブルを検索します。両方のサイズを指定した場合に、デフォルトの画面サイズを使用したい場合は、ログオン出口ルーチン (DFSLGNX0) で、オーバーライドとしてデフォルトの画面サイズを指定する必要があります。
- ユーザーがユーザー・データでログオン記述子を指定できない場合には、ログオン出口ルーチン (DFSLGNX0) を使用するか、もしくは IMS に端末タイプを基にしたデフォルト・ログオン記述子を選択させる必要がある。

## 特定の端末タイプへの動的ログオンの制限

特定の端末タイプへの動的ログオンを制限したい場合は、動的ログオンを行いたくない端末タイプに対するデフォルト・ログオン記述子を削除してください。

### このタスクについて

また、ログオン出口ルーチン (DFSLGNX0) を使用して、特定の端末タイプに対するデフォルト・ログオンを拒否することもできます。こうすると、これらの端末タイプの動的ログオンは失敗します。

### 関連概念

86 ページの『[デフォルトのログオン記述子の選択基準](#)』

IMS は各 VTAM 端末タイプにデフォルト・ログオン記述子を準備します。ログオン記述子名は、LU タイプと TS プロファイルを基にしています。

## LTERM 制御ブロックの作成および再利用

LTERM を含むユーザー構造は、さまざまな多くの方法で作成できます。

以下のいずれかの状態が発生すると、LTERM を含むユーザー構造が作成されます。

- ユーザーによるサインオン時、ユーザー構造が存在していない。
- LTERM に対して非同期メッセージが作成された時点で、その LTERM が存在していない。
- /ASSIGN コマンドを使用して、存在していないユーザーに LTERM を割り当てる場合。
- /ASSIGN コマンドを使用して、存在しない LTERM がユーザーに割り当てられる時点
- /CHANGE USER user AUTOLOGON コマンドが、存在しないユーザーに送信される時点。

サインオンが正常に行われた後、ユーザー構造が端末に割り振られます。

## デフォルト CINIT または BIND ユーザー・データ形式の使用

それぞれのセッション開始要求に、VTAM CINIT または BIND ユーザー・データを組み込んで、ログオン記述子またはサインオン・データを提供することができます。インストール先でこのデータを処理するためのログオン出口ルーチンを用意することができます。

### このタスクについて

以下の方式のいずれかを使用してセッションを設定すると、IMS はオプションのユーザー・データを受け取ることができます。<sup>8</sup>

- IMS /OPNDST コマンドを使用する。
- 自動ログオンを使用する。
- RTO がユーザー・ログオンを提供する。

ユーザーの要件に合うようにユーザー・データ形式を拡張することもできます。ユーザーは、ユーザー・データを使用してログオン出口ルーチンでログオン記述子名を提供するか、あるいは IMS アルゴリズムを使用してログオン記述子名を作成することができます。

ユーザー・データは、CINIT ユーザー・データ・フィールドに表示され、VTAM ログオン出口ルーチンがスケジュールされると、IMS に使用できます。オプション・パラメーターの 1 つであるログオン記述子名が IMS ログオン・プロセスに適用されます。残りのパラメーターは、IMS サインオン・プロセスに適用され、オプションで RACF にも適用されます。いずれのプロセス時も、IMS は、オプションのインストール・システム・ログオン出口ルーチン (DFSLGNX0) を呼び出す前と、サインオン出口ルーチン (DFSSGNX0) を呼び出した後に、CINIT ユーザー・データ・パラメーターに最小限の処理を行います。

ログオン出口ルーチンおよびサインオン出口ルーチンは、インストール先定義のいかなるユーザー・データ形式も変換できますが、IMS では、デフォルトのユーザー・データ形式を次のように定義しています。

- インストール先で拡張できる。
- 出口ルーチンがない場合は、IMS で処理できる。この形式は、IMS /SIGN コマンドと同じ形式でサインオン・データが後に付いたログオン記述子名です。

**制約事項:** STSN 装置のウォーム・セッションを開始するには、ユーザー・データが元のログオンのデータと同じでなければなりません。

### 関連資料

[IMS 制御領域出口ルーチン \(出口ルーチン\)](#)

[/SIGN コマンド \(コマンド\)](#)

[CINIT ユーザー・データ・パラメーターの形式 \(システム・プログラミング API\)](#)

## サインオンとキュー LTERM の割り振り

ETO 端末へサインオンすることは、IMS に対してユーザーを識別し、ユーザー構造を作成し、このユーザー構造を端末構造に接続することです。

### このタスクについて

VTAM 端末のユーザーは、以下のいずれかを実行することでサインオン することができます。

- /SIGN コマンドを出す。
- DFS3649 メッセージ画面でサインオンする。
- セッション開始要求でユーザー・データを提供する。

IMS を用いて動的 VTAM セッションを設定するユーザーは、そのセッションに LTERM が割り振られる前に有効なサインオン・データを入力する必要があります。RACF などのセキュリティ製品でサインオン・データの妥当性検査をするよう求めることもできます。

<sup>8</sup> ETO ISC 端末には、使用可能なオプションのデータはありません。

## サインオン・データの提供

ユーザーはいくつかある方式の1つを使用することによって、サインオン・データを 入力できます。

### このタスクについて

サインオン・データを 入力するためにユーザーが使用できる方式は、以下のとおりです。

- /SIGN コマンドを使用する。
- サインオン・データをログオン・ユーザー・データに組み込む。
- ログオン出口ルーチン (DFSLGNX0) をコーディングする。

## ISC、SLU-P、金融機関、および出力専用装置に対するサインオン・データの提供

動的な ISC、SLU-P、金融機関、および出力専用装置の場合には、セッション開始時にユーザー・サインオン・データを 提供する 必要があります。

### このタスクについて

IMS は、以下の各事項に基づいてそのサインオン・データの妥当性検査を行い、LTERM を端末に割り振ります。

IMS.PROCLIB 内のユーザー記述子  
適用可能な LTERM およびユーザー・オプション・デフォルト  
サインオン出口ルーチン (DFSSGNX0) の出力

**制約事項:** ログオン時、または /SIGN コマンドの使用時には、ユーザーは出力専用装置からサインオン・データを 入力することができません。動的出力専用端末の場合には、サインオン・データをセッション開始時に 入力することが必要です。

出力専用装置の場合に、ユーザーがセッション開始時にサインオン・データを 組み込み忘れると、IMS は MTO にメッセージ DFS2085I (戻りコード 264 とともに) を出します。動的 ISC、SLU-P、もしくは金融機関端末の場合にサインオン・データが省略されると、IMS はメッセージ DFS3645I と DFS3672I を出します。

## 複数のサインオン

ユーザーは、1つ以上の端末に並行してサインオンすることができます。これらの端末は、静的端末と動的端末を組み合わせたものでもかまいません。

### このタスクについて

静的および動的の両 VTAM 端末に対して、EXEC パラメーター SGN=M (複数サインオンを可能にする) を指定する必要があります。

動的端末の場合、IMS に対してユーザーを表すユーザー構造の名前は、固有でなければなりません。ユーザー構造を固有なものにするためには、次の4つの方法があります。

- サインオン出口ルーチン (DFSSGNX0) を使用して、ユーザー ID に 1 から 3 バイトの接尾部を戻す。ユーザー ID と接尾部を合わせた全長が 8 文字を超えることはできません。

**推奨事項:** ユーザー ID を固有なものにする命名規則を使用してください。例えば、サインオン出口ルーチン (DFSSGNX0) を使用して接尾部が付いたユーザー ID を作成し、IMS 呼び出し可能サービスを使用してそのユーザー ID が固有であることを検査します。

**推奨事項:** シスプレックス環境で運用している場合は、下記に示すように接尾部を作成して、IMSplex 全体でユーザー ID が固有になるような命名規則を作成してください。

- 接尾部の最初の部分として、サインオン出口ルーチンは、IMSID パラメーターで指定された IMS システム ID の固有な部分をユーザー ID に付加することができます。



- 接尾部の 2 番目の部分として、IMS 呼び出し可能サービスを使用して、その IMS システムのユーザーが使用できる値を選択します。

**例:** USER というユーザー ID を持つユーザーが IMSA にサインオンします。この 2 ステップ・プロセスによって、固有な接尾部 A1 が作成されます。

1. サインオン出口ルーチンが IMSA からの A をユーザー ID に追加します。
2. サインオン出口ルーチンは IMS 呼び出し可能サービスを呼び出します。その場合、出口ルーチンは IMSA にサインオンする USER という名前の最初のユーザーを表す 1 を追加します。

この例では、ユーザー ID の長さは 6 文字より短くなければなりません。

- サインオン出口ルーチンを使用して、ユーザー構造名がノード名と同じになるように指定する。
- サインオン出口ルーチンを使用して、ユーザー ID またはノード名と関連しない名前を指定する。
- ノード名と同じユーザー記述子名を指定する。これによって、ユーザー構造の名前がノード名と同じになります。これは、以下のいずれかの方法を使用することで行うことができます。
  - ユーザー記述子名を含むサインオン・データを入力する。
  - サインオン出口ルーチンを使用して、ノード・ユーザー記述子を指定する。
  - ユーザー記述子を指定せず、デフォルト解釈によって IMS にノード・ユーザー記述子を選択させる。

SGN=M を指定して複数のサインオンを使用可能にしていない場合は、動的ユーザーのユーザー ID は固有でなければなりません。ユーザー ID がユーザー構造名として使用される場合は、SGN=M が指定されているかどうかに関係なく、固有でなければなりません。ユーザー ID が (例えば、サインオン出口ルーチンを使用して) ユーザー構造名として使用されない場合には、複数のサインオンを表す SGN=M を指定する必要があります。

**推奨事項:** IMS ユーザーに単一の名前を割り当てると、出力の状況を判別するのに役立ちます。個々のユーザーに複数の名前を使用する場合 (例えば、サインオン出口ルーチンが名前を割り当てる場合) には、サインオンによって作成されるユーザー名を判別する手段を用意する必要があります。

**制約事項:** システム定義時に定義された動的 LTERM と静的 LTERM に対して同一の名前を使用することはできません。

#### 関連資料

[サインオン出口ルーチン \(DFSSGNX0\) \(出口ルーチン\)](#)

[IMS 呼び出し可能サービス \(出口ルーチン\)](#)

## DFS3649A (「サインオンが必要」メッセージ) の受信

セッションがトランザクションまたはコマンドを入力できるようになるには、その前に ETO ユーザーがサインオンする必要があります。

サインオン前は、ユーザーは **/RCLSDST** コマンドしか入力できません。以下のいずれかの状態では IMS はメッセージ DFS3649A (**/SIGN** コマンドが必要であることを示す) を出します。

- サインオンが失敗した後
- サインオフの後
- ユーザー・データが入力されないログオンの後

どの端末ユーザーも **/SIGN ON** コマンドを出して、次のセッションを始めることができます。

以下の状態の場合、DFS3649A メッセージを受信しません。

- サインオン・データが VTAM CINIT または BIND に含まれている。
- セッションが、ISC、SLU-P、金融機関、または 3270 プリンターのセッションである。
- 端末が不在モードで実行中の SLU-1 端末である。
- 端末が自動ログオン試行の対象である。

#### 関連資料

[/SIGN コマンド \(コマンド\)](#)

## 関連情報

[DFS3649A \(メッセージおよびコード\)](#)

## DFS3650I (セッション状況メッセージ) の受信

ユーザーが正常にサインオンすると (もしくは、サインオンが不要な場合)、IMS はメッセージ DFS3650I を出して、IMS とのセッションの状況を示します。

**制約事項:** 以下の状態では、IMS はメッセージ DFS3650I を送信しません。

- SLU-1 端末が不在モードで実行している。
- ユーザー記述子に NOTERM オプションを指定した場合。

メッセージ DFS3650I は、以下に示すような、セッションに関する情報を提供します。

- ユーザーが会話モードであるかどうか。
- その端末に対してユーザー出力セキュリティーが存在するかどうか。

## 関連情報

[DFS3650I \(メッセージおよびコード\)](#)

## ETO 端末と LTERM の関連

システム・プログラマーは、IMS とのセッションを持っている端末と特定ユーザーの LTERM との関係に対して責任があります。

単一構成の端末 (SLU 2 など) の場合、IMS は、サインオン時に提供されたユーザー ID と同じ単一のデフォルト LTERM 名を作成します。出口ルーチンまたはユーザー記述子を使用すると、LTERM に別の名前を付けることができます。

複数構成の端末 (SLU P など) の場合は、その端末を使用するためには十分な LTERM が作成されていることを確認する必要があります。出口ルーチン、インストール先作成の記述子、あるいはノード・ユーザー記述子を使用して、十分な数の LTERM が作成されるようにすることができます。それ以外の場合は、IMS が端末の最初のコンポーネントに割り振られた単一の LTERM を作成します。ノード・ユーザー記述子については、ユーザーはノード・ユーザー記述子を使用して、LTERM とノードとの正しい関係を明示的に、またはサインオン出口ルーチン (DFSSGNX0) の中で設定する必要があります。

アプリケーション・プログラムが特定の LTERM 名を特定の端末に関連付けることは可能です。システム・プログラマーは、代替 PCB の名前が、以下のもので定義されている LTERM 名と整合することを確認する必要があります。

ユーザー記述子  
サインオン出口ルーチン  
挿入出口ルーチン  
ISC、SLU-P、および金融機関のセッションのリカバリー要件

## IMS が割り振るキューを判別する方法

IMS は、次の方法を使用して、どの LTERM を割り振るかを決定します。

1. ユーザー制御ブロックが IMS 内にすでに存在している場合には、それらの制御ブロックは再割り当てされ、サインオン出口ルーチン (適用可能ならば) が呼び出されます。
2. 制御ブロックが存在せず、ユーザー記述子がサインオン時に指定されている場合、IMS は指定された記述子を探索します。
  - 指定された記述子がユーザー ID、ノード名、または DFSUSER 記述子のいずれでもない場合、IMS はエラー・メッセージ DFS3649A (戻りコード 148 とともに) を送信します。
  - 指定された記述子が、サインオンしているユーザーの有効なユーザー記述子である場合、IMS は、利用可能な記述子のテーブルを構築し、サインオン出口ルーチン (適用可能ならば) を呼び出します。
  - サインオン出口ルーチンが存在していて RC=0 の場合、IMS は通常のサインオン処理を行います。戻りコードが 0 以外の場合、サインオンは拒否されます。



3. ユーザー記述子がサインオン時にも指定されず、サインオン出口ルーチンでも指定されない場合、IMS は下記のものから (順序どおりに) 有効な記述子を選択します。
  - a. ユーザー ID
  - b. ノード名
  - c. DFSUSER

#### 関連資料

[サインオン出口ルーチン \(DFSSGNX0\) \(出口ルーチン\)](#)

## 特殊処理モードの設定

ユーザーはサインオンした後、IMS コマンドを使用して以下の処理モードを設定することができます。

### このタスクについて

#### 排他モード

**/EXCLUSIVE** コマンド

#### 事前設定宛先モード

**/SET** コマンド

#### MFS テスト・モード

**/TEST MFS** コマンド

#### テスト・モード

**/TEST** コマンド

これらの特殊処理モードは、テスト・モードおよび事前設定宛先モードを除いて、サインオフ後もユーザーのために保存され、ユーザーが次にサインオンする端末で再設定されます。

以下のコマンドを出すと、IMS は、端末またはユーザーに代わって必須の制御ブロックを作成するか、もしくはユーザーの状況を保持します。

#### **/EXC USER**

ユーザー構造を排他モードに置きます。

#### **/STOP NODE** または **/STOP USER**

ノードまたはユーザー構造を停止します。

#### **/TEST MFS USER**

ユーザー構造を MFSTEST モードに置きます。ユーザーがサインオンすると、端末が MFS をサポートする場合には IMS は、その端末を MFSTEST モードに置きます。

#### **/TRACE NODE**

ログオン・ノードをトレースします。

制御ブロックは、特定の状況が取り除かれたとき即座に削除されるわけではありません。制御ブロックは、すべての削除要件を満たした場合に、次のチェックポイントで削除されます。未解決の状況があると、制御ブロックは削除には不適合です。

端末状況をリセットし、制御ブロックを次の単純チェックポイントでの削除に合格とするためには、以下のコマンドを使用します。

#### **/END**

排他モードとテスト・モードをクリアします。

#### **/RESET**

事前設定した宛先を除去します。

#### **/RSTART**

停止したリソースの利用を開始します。

#### **/START**

停止したリソースの利用を開始します。**/START** コマンドによってリセットされる他の状況については、「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」を参照してください。

## /TRACE

トレースをオフに設定します。

状況を保持するため制御ブロックが単独で存在する場合は、上記のコマンド・セットを使用してください。サインオフ出口ルーチン DFSSGFX0 を使用して、これらの状態をリセットすることができます。

### 関連概念

109 ページの『ETO 制御ブロック削除によるパフォーマンスの向上』

ETO を使用して、IMS は制御ブロックを動的に削除することができます。制御ブロックを動的に削除すると、ストレージの使用量が削減され、パフォーマンスを高めることができます。

## ETO を使用したプリンター

ETO 環境では、プリンター・サポートを実現する方法は 2 つあります。直接印刷と印刷関連付けです。

これらの 2 つの方式を実現する方法は、アプリケーション・プログラムがプリンター LTERM 名を識別する方法によって異なります。

### 直接印刷

直接印刷とは、アプリケーション・プログラムが VTAM LU 名にメッセージを挿入する際に使用する印刷技法です。

動的 LTERM とユーザーおよび端末のリソースはすべて、VTAM、VTAM LU 名の後に命名されます。多数のアプリケーション・プログラムが同じプリンター LTERM にデータをキューイングすることはできませんが、これは、問題がはさみ込まれたデータを作成するおそれがあります。

### 印刷の関連付け

印刷の関連付けは、アプリケーション・プログラムのプリンター出力を特定のプリンター・ノード名に送る技法です。

印刷の関連付けでは、アプリケーション・プログラムが、画面 - ユーザー・キュー名に関連するキューにメッセージを挿入します。ユーザーが印刷要求を行うと、アプリケーション・プログラムは、データを別のプリンター LTERM のキューに入れます。これで、問題がはさみ込まれたデータを回避します。

ユーザー・キューを変更するには、プリンターのサインオフとサインオン が必要です。プリンターのユーザーが多数の場合には、オーバーヘッドが多くなる可能性があります。

関連するプリンターが自動的にログオンされるのは、それらの LTERM のキューにメッセージが入れられたときで、これは通常は、IMS 再始動時または LTERM の作成時です。

**推奨事項:** アプリケーション・プログラム間でプリンターを共用する方法については、注意して計画してください。

出口ルーチンおよびアプリケーション・プログラムをインプリメントすることによって、端末オペレーターは、ログオン時またはサインオン時に宛先を提供することができます。

### プリンター・ノード名の識別

次の 2 つの方法のいずれかで、プリンター・ノード名を識別することができます。

#### このタスクについて

プリンター・ノード名を識別できる方法は、以下のとおりです。

#### 手順

- ユーザーがセッションを設定するとき、ログオン・ユーザー・データとして 1 つ以上のプリンター・ノード名を組み込むことができます。

- ユーザーがサインオン時にプリンター・ノード名を提供できるようにするため、DFS3649A グリーティング・メッセージの MFS 形式を変更することができます。ただし、サインオン時に、サインオン出口ルーチンもユーザー・データとしてプリンター・ノード名を検出できなければなりません。

## 印刷の関連付けを実現するサインオン出口ルーチンのコーディング

印刷の関連付けを使用するには、サインオン出口ルーチン (DFSSGNX0) をコーディングします。

### このタスクについて

サインオン出口ルーチンをコーディングして、以下を行ってください。

- 各画面ユーザーごとに、プリンター LU 名とユーザー名を識別します。
- 入力ユーザー・データから、プリンター・ノード名を判別します。
- ユーザー関連の LTERM 構造に名前を付け、選択されたプリンターにサービスを行います。
- アプリケーション・プログラムが、正しいメッセージ・キュー (代替 PCB) に挿入できるようにするため、特定のユーザーに関連するキューを判別できる (ユーザー ID を使用して) ようにします。
- 関連する印刷パラメーターとして、プリンター・ノード名を IMS に渡します。
- ユーザーがプリンター・ノード名を提供するときに、各プリンターに割り振られるユーザー名を決定します。アルゴリズムまたはテーブルのいずれかを使用して、この決定を行うことができます。出口ルーチンでこれらのユーザー名を IMS に渡さなければなりません。こうすると、必要なユーザー制御ブロックを IMS が作成します。

サインオンしたユーザー ID 用、および必須の各プリンター関連ユーザー用に固有な名前が必要です。

**例:** ユーザーのプリンター ID の名前を示すには、ユーザー ID の末尾に「P」を付加します。ユーザー ID が AAA であれば、このユーザーのプリンター識別の名前は AAAP になります。

- 最大 4 台 (ユーザー構造削除前の単純チェックポイントの数) のプリンターのそれぞれに 1 つの値を指定します。

アプリケーション・プログラムは、サインオン出口ルーチンと同じ方式を使用して、入力端末ユーザー ID からプリンターの LTERM 名を判別し、代替 PCB を通じて必要な出力データをキューに入れます。出力データがキューに入れられると、IMS はユーザー構造を正しいプリンターに割り振って出力を送達します。

関連プリンター LTERM が割り振られてから空にされると、キューは端末から割り振り解除されます。

### 関連資料

[サインオン出口ルーチン \(DFSSGNX0\) \(出口ルーチン\)](#)

## プリンターの定義

プリンターは通常は出力専用装置ですが、入力送信用にインプリメントすることもできます。

単一ユーザーなのか、複数ユーザーが同じプリンターを共用するのかを決定する必要があります。アプリケーション・プログラムがキュー名を検知する場合は、一方または他方に限定されることがあります。

### 単一ユーザー構造

1 台のプリンターを表す単一ユーザー構造を使用することは、プリンターを定義するための最も簡単な方式です。この場合、プリンターに送信されたメッセージは、発信された順序で印刷されます。出力のインターリーピングが発生する可能性があります。複数セグメント・メッセージでは、常に、すべてのセグメントが連続した順序で印刷されます。複数のメッセージは、それぞれの発信方式によって異なる動作をする可能性があります。

### 複数ユーザー構造

プリンターに対して複数のユーザーがサポートされます。単一ユーザーに対するすべてのメッセージが印刷され、次のユーザーが印刷のために選択されるのは、現行ユーザーの出力がなくなった時点だけです。同じユーザーに対するメッセージのインターリーピングは、単一ユーザー構造の場合と同じように発生し

ますが、ユーザーを切り替えるということも、アプリケーションの観点からすると、メッセージ・インターリーピングと同じこととなります。プリンターがどのように共用されるかをアプリケーションと操作上の観点から理解しておくことが重要です。

プリンターの共用は動的端末だけに固有の課題ではなく、実際は、静的端末にも同様の課題です。

次のメッセージが別のユーザーから印刷される場合、IMS は分離ページを印刷します。この分離ページの内容は、その内容を変更する出口ルーチンを使用して制御することができますが、このページは空白であっても常に印刷されます。同じユーザーからのメッセージは、分離ページでは分離されません。NOTERM オプションを使用すると、DFS3650 分離メッセージの使用を回避できます。

## ETO を使用したプリンターの共用

同じ出力端末を使用することにより、複数のユーザーがプリンターを共用できます。

同じ出力端末を使用するには、2つの方式があります。

- 複数のユーザーが、出力装置 (ノード) にデータを送達するために使用されるすべての LTERM 名を含む単一ユーザー記述子を定義することができます。
- 複数のユーザーが、1つのノードに対する自動ログオン・パラメーターを定義することができます。

どちらの方式を使用するかを決定する前に、下記の事項を検討してください。

- LU - LTERM の関係は、通常、変更がありませんか。
- メッセージ・レベルのインターリーピングは受け入れられますか。
- 他のユーザーの出力が印刷される間、あるユーザーの出力の遅延を容認できますか。
- 各 LU ユーザーの最小メッセージ送達率のために、自動サインオフと自動ログオンを連続して処理すると、特定の端末に負担がかかり過ぎますか。

上記事項の回答が、インプリメンテーション方法を判別する上で役に立ちます。

- 上記の質問事項の回答がすべて「はい」の場合は、プリンターに対して単一の多重 LTERM ユーザーを作成するようにしてください。
- 回答のいずれかが「いいえ」の場合には、自動ログオンを使用して、複数ユーザーの LTERM を動的に割り振るようにしてください。
- これらのオプションを組み合わせてインプリメントすることもできます。

## オペレーター・コマンド

ここでは、ETO で使用する場合の **/OPNDST** コマンドと **/ASSIGN** コマンドの使用法について説明します。

### **/OPNDST**

VTAM モード・テーブルは、端末が最初にログオンする際に使用されるもので、装置特性を判別します。ただし、端末への最初の参照が **/OPNDST** コマンドを通じて行われる場合は、そのコマンドの **MODETABLE** オペランドが装置タイプを判別します。

このコマンドで **MODETABLE** オペランドを省略すると、デフォルトが採用されて、希望しないものになることがあります。端末が IMS とのセッションを持ってからは、ブロック削除に対する規則が適用されません。装置タイプは、ブロックが削除されるまで設定された状態にあります。端末が一度クローズされてから再オープンされた場合、IMS は、利用可能であれば既存のブロックを使用します。ブロックが削除された場合、そのブロックは、次の端末ログオン時に再構築されます。そのため、**/OPNDST** コマンドは、以下の2つの事項によって異なる結果になる可能性があります。

- ブロックがまだ利用可能な状態にある (かつ、削除されていない) かどうか。
- 前の初期設定以降、出口ルーチンまたは記述子に変更されたかどうか。

## /ASSIGN

**ASSIGN** コマンドを使用すると、ある端末から別の端末にキューを移すことができます。静的端末を動的(ETO) 端末に、あるいは動的端末を静的端末に、再割り当てするためにこのコマンドを使用することはできません。

### 関連資料

[/ASSIGN コマンド \(コマンド\)](#)

[/OPNDST コマンド \(コマンド\)](#)

## ETO に関するシステム定義パラメーター

ここでは、ETO で使用できるシステム定義パラメーターについて説明します。

### DLQT パラメーターを使用した DEADQ 状況時間の設定

ユーザー制御ブロック構造は、通常、いくつもの状態で作成されます。

#### このタスクについて

ユーザー制御ブロック構造が作成される状態には、以下の場合があります。

- 端末がログオンされ、ユーザーがサインオンしたとき。
- AO 出口ルーチン (DFS AEOU0) が LTERM またはトランザクションにメッセージを挿入したとき。
- 非同期トランザクション出力メッセージが送信されたとき、もしくは、端末メッセージ通信または **/BROADCAST LTERM** コマンドが出されたとき。

未知の宛先、無効になった宛先、あるいは存在しない宛先にメッセージが送信されることがあります。IMS は、ETO ユーザー制御ブロック構造またはそれに関連したメッセージ・キューが、DLQT 実行パラメーターで設定された制限時間内にアクセスされなかった場合に、送達不能キュー (DEADQ) の状況を設定します。(DLQT の場合、有効な値は 1 日から 365 日までですが、値 1 は通常では推奨されません。これにより、最初のチェックポイント時に、不完全、かつまぎらわしい DEADQ 状況設定が数多く発生することがあるためです。) ユーザーのキューにメッセージがあるかどうかに関係なく、またユーザーが端末にまだ割り振られているかどうかに関係なく、DEADQ 状況は設定できます。DEADQ 状況は IMS チェックポイント時に設定され、メッセージ DFS3643 で MTO に通知されます。

ユーザーにキューイングされているメッセージがある場合は、ユーザーにサインオンするか、**/DEQUEUE** コマンドまたは **/ASSIGN** コマンドを出すことによって、DEADQ 状況を除去してください。

キューにメッセージがないユーザー制御ブロック構成は、DEADQ 状況という結果になる可能性があります。特定の状況が保留されていると、ユーザー制御ブロックは削除されません。前のサインオン時 (応答または会話モードなど) に、あるいはコマンド (**/STOP** または **/EXCLUSIVE**) の結果として、状況が設定されていることもあります。制御ブロックが、DLQT 実行パラメーターに指定された時間よりも長く未使用のままになっていると、IMS は制御ブロックに DEADQ 状況を割り当て、メッセージ DFS3643 で MTO に通知されます。

ユーザー制御ブロックが DEADQ 状況にある場合、この状況が除去されるのは、ユーザーがサインオンしたとき、あるいは次のチェックポイント時に、すべてのメッセージがデキューされて、リカバリー可能な状況条件が適切なコマンドを使用して除去された後です。後者では、ユーザーの制御ブロックと、関連したメッセージ・キューは削除されます。

DEADQ 状況にあるユーザー制御ブロック構造は、端末に割り振られる場合と、割り振られない場合があります。LU タイプ 6、SLU-P、および金融機関端末の場合には、ユーザー構造を、アクティブ・セッションのない端末に割り振ることができます。ログオフあるいは他のセッション終了は、これらの端末をメッセージ・リカバリー (SNA STSN) 保留の状態にすることがあります。ユーザーは、端末に割り振られたままになり、メッセージはキューイングされる場合とされない場合があります。(この組み合わせは、他の VTAM 端末タイプでは存在しません。端末とユーザー ID の割り振り解除がログオフとサインオフ時に強制されるためです。)



ユーザー制御ブロック構造が LU タイプ 6、SLU-P、あるいは金融機関端末に割り振られている場合、ユーザーにログオンしてサインオンするか、もしくは **/DEQUEUE** コマンドを使用することで、DEADQ 状況を取り除くことができます。LU タイプ 6 (ISC) 端末の場合は、**/DEQUEUE** コマンドを使用するか、あるいは LU タイプ 6 セッションを強制的にコールド・スタートしてください。強制セッション・コールド・スタートとは、**/STO NODE**、**/ASSIGN (USER TO VTAMPOOL)**、**/STA NODE** をこの順序で出すことであり、LU 6 の場合のみ有効です。SLU-P または金融機関端末については、強制コールド・スタートは不可能です。SLU-P または金融機関端末に対する ETO ユーザーの割り振りをクリアするには、IMS のコールド・スタートが必要です。

共用キュー環境では、**/DISPLAY QCNT MSGAGE** コマンドを使用して、キューの経過日数を調べることができます。

## 自動サインオフ (ASOT)

自動サインオフは、アイドル・セッションからユーザーを割り振り解除します。割り当てられた時間内にセッションでアクティビティーが発生しなかった場合、ユーザーは自動的にサインオフされ、そのセッションを使用するためには、再度サインオンする必要があります。

### このタスクについて

システム・プログラマーは、DFSPByyy メンバー内の ASOT パラメーターを使用して、自動サインオフ時間を設定します。ASOT EXEC パラメーターの時間値は、3600、SLU P、または ISC 装置には適用されません。

割り当てられた時間値が複数存在する場合、IMS は以下の基準を使用して、そのうちのどの値を使用するかを判別します。

- 有効な ASOT 値がユーザー記述子で指定されている場合は、この割り当てられた時間値が使用されます。
- ユーザー記述子が、割り当てられた時間値を指定していない場合、ログオン記述子から得た、割り当てられた時間値が使用されます。
- ログオン記述子が、割り当てられた時間値を指定していない場合、DFSPByyy メンバーから得られた時間値が使用されます。
- DFSPByyy メンバーが、割り当てられた時間値を指定していない場合は、デフォルト値の 1440 が使用されます。
- DFSPByyy メンバーの値が無効な場合は、デフォルト値の 10 が使用されます。

ログオン出口ルーチン (DFSLGNX0) は、ログオン時に ASOT と ALOT の値を変更することができます。サインオン出口ルーチン (DFSSGNX0) は、制御ブロック構造が存在する場合でも、サインオン時に ASOT の値を変更することができます。

DFSPByyy メンバーの ASOT パラメーターで指定される、割り当てられる時間の値は、次のとおりです。

- ASOT = 0
  - 送信に使用できる出力がない場合には、ユーザーは即時にサインオフされます。この仕様は、通常、以下の場合に即時にサインオフするために自動ログオン端末で使用されます。
    - IMS 入力または出力メッセージが利用不可能である。
    - 使用可能な最後の出力メッセージが完了した後。
  - 3270 端末や SLU2 端末などの対話式端末には、この仕様はお勧めできません。これらの端末セッションは、通常、PA キーを戻して、後続のサインオンを続けます。アイドル時間は、端末入力を待機せずに、即時にサインオフとなります。

DFSPByyy メンバーの値はこれらの装置タイプには使用されません。

- ASOT = (10 - 1439)

割り当てられた分数内に端末アクティビティーがないとユーザーはサインオフされます。

- ASOT = 1440

ユーザーは自動的にサインオフされません。これは自動サインオフ機能がないのに等しいです。SLU-P、3600/金融機関、およびISC 端末のシステム・デフォルトは、1440 です。

自動サインオフの完了後、IMS は、自動ログオンを待っている同じノード名のユーザーを探します。IMS が、出力を待っている他のユーザーを検出した場合、そのユーザーが端末に割り振られ、キューが処置されます。

## 自動ログオフ (ALOT)

自動ログオフは、割り当てられた期間の間にサインオフされた端末のIMS とのセッションを終了します。割り当てられた時間が経過した後、端末は自動的にログオフされます。

### このタスクについて

システム・プログラマーは、次のいずれかでこの時間を設定します。

- DFSPByyy メンバーの ALOT (自動ログオフ時間) パラメーターで
- セッション制御ブロックを作成するために使用されるログオン記述子で
- 初期設定時に EXEC パラメーターで

割り当てられた時間が複数ある場合、割り当てられた時間値を判別するために以下の基準が使用されます。

- ALOT 値がログオン記述子で指定されている場合は、この割り当てられた時間値が使用されます。
- ログオン記述子が、有効な割り当てられた時間値を指定していない場合は、DFSPByyy メンバーに指定されている時間が使用されます。
- DFSPByyy メンバーが、割り当てられた時間値を指定していない場合は、デフォルト値の 1440 が使用されます。
- DFSPByyy メンバーの値が無効な場合は、デフォルト値の 10 が使用されます。

3600/金融機関、SLU P、あるいはISC 端末のログオン記述子に ALOT=1440 を指定すると、自動ログオフを希望しないことを指定することができます。これらの装置のシステム・デフォルト値は 1440 です。ISC 端末用に作成されるログオン記述子には、自動ログオフが起こってはならないことを示す ALOT=1440 を指定してください。ログオン出口ルーチン (DFSLGNX0) は、制御ブロック構造が存在する場合でも、ログオン時に ALOT の値を変更することができます。

DFSPByyy メンバーの ALOT パラメーターで指定される、割り当てられる時間の値は、次のとおりです。

#### ALOT = 0

サインオンが無効な場合、端末は即時にログオフされます。この仕様は、通常、ユーザーがログオン・プロセス中に自動的にサインオンされた場合に、端末セッションで使用されます。自動ログオン時には、次のいずれかの方法でサインオン・データを与えることができます。

- **IMS /OPNDST** コマンドによって提供されるサインオン・データ
- ログオン・ユーザー・データ (BIND) によって提供されるサインオン・データ
- ログオン出口ルーチン (DFSLGNX0) によって提供されるサインオン・データ

ALOT=0 を使用するには 2 つの操作モードがあります。いずれも、DFSINTX0 ユーザー初期設定出口パラメーター・リストを使用して設定できます。

デフォルト・モードでは、サインオン・エラーが検出されると、セッションは自動的にサインオフされ、そのあとログオフされます。メッセージは送信されません。DFSINTX0 出口を提供しない場合、あるいは、この出口を提供し、ALOT=0 に対してデフォルト・モードを示した場合は、ログオン・プロセス時にサインオン・データを与えなければなりません。以下のエラー条件が起こると、どの場合も、その結果として自動ログオフが起こります。

1. 非サインオン、あるいはサインオンまたは入力処理中に検出されたエラーは、即時ログオフとなります。
2. **/SIGNOFF** は、即時ログオフとなります。

3. **/SIGNON** は、現行ユーザーをサインオフさせ、新規ユーザーをサインオンさせます。しかし、サインオン・プロセス時に検出されたエラー (正しくないパスワードやパスワードの期限切れなど) は、即時ログオフになります。

**制約事項:** デフォルト・モードは、メッセージ DFS3649 への応答を必要とする対話式端末セッションには使用しないでください。このようなセッションは入力サインオンを待たずに、即時にログオフします。

代替モードでは、サインオン・エラーが検出されると、セッションは自動的にサインオフされ、メッセージが送信されてから、セッションはログオフされます。サインオン・データを与えることはできませんが、必要ありません。以下のエラー条件が起こると、どの場合も、その結果として自動ログオフが起こります。

1. 入力処理中に検出された非サインオン・エラーは、即時ログオフとなります。
2. ログオン・ユーザー・データ (BIND) またはログオン出口 (DFSLGNX0) によってサインオン・データが提供されませんでした。
3. **/SIGNOFF**、または **/SIGNON** から生じるエラーは、メッセージ DFS3649(A) (サインオンが必要) を送信し、固定の 10 分間タイマーが新規セッションを待つように設定します。その時間間隔内にサインオンが行われないと、セッションはログオフされます。

#### **ALOT = (10 - 1439)**

サインオンするユーザーがないまま、割り当てられた分数が経過すると、セッションは終了します。

#### **ALOT = 1440**

セッションは自動的に終了しません。これは自動ログオフがないのと同じです。

## 自動サインオフおよび自動ログオフ・タイマー

VTAM I/O タイムアウト機能 (アクティブならば) は自動的に終了するユーザーあるいはセッションを検出します。

VTAM I/O タイムアウト機能がアクティブである場合、タイマーは 1 分間隔で、また、このタイムアウト機能がアクティブでない場合には 5 分間隔で、時を刻みます。タイマーは、ルーチンを開始してどのリソースを自動サインオフあるいは自動ログオフにするかを決めます。指定するタイムアウト値は、ユーザーあるいはセッションが自動的に終了する最小値です。終了は、実際には、指定されたタイムアウト値の次のタイマーの刻みのときに起こります。

## 自動ログオン

出力が使用可能な場合、TERMINAL マクロの SHARE オプションを使用して、IMS による端末セッションの自動開始を要求する代わりに、ETO は ETO 端末とユーザーの自動ログオンをサポートします。ユーザーを IMS に定義するときに、自動ログオン・パラメーターを指定してください。

### このタスクについて

**定義:** 自動ログオンは、IMS が自動的に端末をログオンとサインオンできるようにします。ユーザーの自動ログオン・オプションを指定した場合、いずれかのユーザー・キューにデータが置かれると、IMS はセッションを確立します。以下のいずれかを使用して、自動ログオンを指定することができます。

- ユーザー記述子の AUTLGN= パラメーター
- 宛先作成出口ルーチン (DFSINSX0)
- 関連プリンターに対するサインオン出口ルーチン (DFSSGNX0)
- AUTOLOGON キーワードを指定した **/CHANGE** コマンド

自動ログオンには、自動ログオンと自動サインオンの両方が含まれます。再始動時、IMS は、自動ログオンで定義されていて、待機しているキュー・データがある端末とのセッションを開始しようとします。セッションが設定されると、IMS は自動的に端末にサインオンします。待機ユーザーのキューが存在していて、割り振られたキューが処置されると、既存の ASOT 値に関係なく、自動ログオン・ユーザーはサインオフされます。



複数のユーザーが同じ自動ログオン端末を使用する場合、IMS は待っているユーザーの順番を管理します。自動ログオン・ユーザーがサインオフされると、同じ端末の次の自動ログオン・ユーザーが自動的にサインオンされます。すべての自動ログオン・ユーザーがサインオフすると、端末は解放され、セッションを終了するための ALOT サイクルが始まります。

自動ログオンは、静的端末の TERMINAL マクロの OPTIONS=SHARE を置換します。自動ログオン・プリンターは、ログオン記述子の OPTIONS=NOASR (自動セッション再始動なし) を無視します。IMS は自動ログオン・プリンターには常に OPTIONS=ASR を想定しています。

自動ログオンは、一般的に、出力専用端末に対して指定されます。一般には、自動ログオン・ユーザーと非定型ユーザーが端末セッションを共用することはありませんが、対話式端末では端末セッションの共用が行われる可能性があります。自動ログオンする可能性のある端末の対話式ユーザーは、自動ログオン出力で競合しないように、セッション開始要求 (ログオン) でユーザー・サインオン・データを提供する必要があります。非定型端末ユーザーは、出力がサインオフ後に可能になった場合には、自動ログオンを使用して出力をデフォルト端末に送達することができます。端末が停止した場合は、/START NODE コマンドではセッションを開始できません。セッションを再始動するには、/OPNDST NODE USER コマンドを使用してください。

## 出力の割り当て

以下のトピックでは、非同期出力と出力宛先の管理について説明します。

### 非同期出力

非同期出力は、単純なタイプミス・エラーで簡単に無効な宛先へ送信される場合があります。

これは、サインオンされたすべての許可ユーザーがユーザー構造を作成できるという柔軟性を、ETO が IMS に提供するためです。LTERM が見つからないとき、IMS はメッセージ通信用とアプリケーション挿入呼び出し (ISRT) プロセスに必要なユーザー構造を自動的に作成します。

ETO では、出口ルーチンによって拒否されない限り、すべての宛先が有効です。したがって、以下のよう  
な情報を入力するときにタイプミスをする、誤ってキューを作成してしまう可能性があります。

- 代替 PCB
- メッセージ通信
- /BROADCAST コマンド
- MSC 出力

前の状態で使用される LTERM は、送達不能キューと呼ばれます。IMS は、それらのキューを MTO がモニターして処置するためにコマンドを提供します。

非同期出力の宛先には、有効と無効の 2 つのタイプがあります。

有効宛先とは、出力を受信するよう意図された宛先です。無効宛先とは、出力を受信できない宛先 (例えば、誤ったつづりで指定された宛先) のことです。

### 有効な宛先への非同期出力

非同期出力 (挿入、ブロードキャスト、およびメッセージ通信など) を送信するには、前もって ETO を使用可能にし、有効な ETO 記述子と有効な宛先 LTERM 名を定義しておく必要があります。

### このタスクについて

ETO を使用可能にするには、IMS または DCC 始動プロシージャーで ETO=Y を指定します。

ETO 記述子は、ユーザー構造の作成に使用されます。

直前にユーザー構造と LTERM を作成した制御ブロックがあれば、その制御ブロックが再利用されます。

## 無効な宛先への非同期出力

IMS では、送達できないデータを「送達不能」と言います。

### このタスクについて

以下の状態にある場合、データは送達できません。

- キューされた出力の自動ログオンの宛先が利用可能になっていない。
- データに関連したユーザー ID が有効なユーザー ID ではない。
- ユーザー・サインオンが、常にインストール・サインオン出口ルーチンで拒否される。
- 入出力メッセージで無効な宛先が指定された (例えば、タイプ入力エラーなど)。

初期設定時に EXEC パラメーターで DLQT 値を指定すると、LTERM キューがこの値を超えて、データをデキューしていなかったり、状況が取り除かれなかった場合に MTO に自動的に通知することができます。送達不能キューに対しては、以下のコマンドを使用することができます。

- 送達不能キュー時間 (DLQT) よりも古いユーザーの LTERM キューで、データをデキューしていないユーザーや、状況を除去していないユーザーを識別するには、**/DISPLAY USER DEADQ** コマンドまたは **/DISPLAY STATUS USER** コマンドを使用してください。
- 他の動的ユーザーがキューに入ったデータを検討できるよう送達不能キューを再割り当てするには、**/ASSIGN** コマンドを使用します。
- 送達不能キューのデータを除去するには、**/DEQUEUE** コマンドを使用します。

共用キュー環境では、**/DISPLAY QCNT MSGAGE** コマンドを使用して、送達不能キューと見なされるメッセージを判別することができます。

### 関連資料

[IMS コマンド \(コマンド\)](#)

## 非発信元端末への出力メッセージの送達

IMS は、ユーザーの出力をそのユーザーがサインオンされている端末へ送信します。

ETO を使用すると、入力データを入力した端末以外の端末で出力メッセージを受信することができます。ただし、入出力メッセージは、以下に示すように、端末とメッセージの両方を定義した MFS 仕様に合うようにフォーマット設定されます。

- MFS フォーマット設定は、MFS バイパスを使用した場合を除き、3270R (非 SNA) および SLU-2 端末には必須です。MFS 形式制御は他のすべての VTAM 端末ではオプションです。MID と MOD 制御ブロックの可用性を基に、メッセージごとに MFS を使用します。MFS ページング・サポートは、SLU-1 または NTO 端末では利用できません。
- 適切な DIF と DOF 制御ブロックを生成する装置ステートメントを使用して、MID と MOD 制御ブロックを定義する必要があります。これにより、メッセージが端末へ送信または端末から受信したときに、特定の端末タイプへのメッセージまたは特定の端末からのメッセージのマップを可能にします。適切な MFS ブロックが無いと、デフォルト・マッピングが起きます。フォーマット設定ブロックが検出できないと、IMS はエラー・メッセージ DFS057 を出します。

非発信元端末へ出力メッセージを送達することを計画する場合、MFS の形式、プロシーチャー、さらに制限を新規に作成するか、あるいは拡張する必要があります。これによってユーザーは自由に端末間を移動できます。

## 不注意による出力データ・ストリーム

ETO を使用すると、動的端末ユーザーは、インストール・システムの制約の範囲内であれば、自由に端末間を移動できます。IMS /ASSIGN コマンドの入力ミスのため、端末固有のデータを誤って間違った端末タイプに送信してしまうことがあります。

データを誤った端末へ送信した場合、そのデータは、配信された時点でエラーになるか認識不能になります。ユーザーがログオンする可能性のあるすべての端末タイプに対して、必ず MFS 定義を作成してください。

## サインオフ

ETO 端末からのサインオフは、IMS に対するユーザーの識別を終了し、さらに (ほとんどの場合) ユーザー構造を端末構造から切断して、ユーザー構造を削除します。

ユーザーが ETO VTAM 端末からサインオフすると、IMS はサインオフ 出口ルーチン (DFSSGFX0) を呼び出します。

**推奨事項:** システム情報を保守するサインオン出口ルーチン (DFSSGNX0) を提供している場合には、その処理を補うサインオフ出口ルーチン (DFSSGFX0) も提供するようにしてください。

## ログオフ

ユーザーが ETO 使用時に VTAM 端末からログオフすると、IMS はログオフ 出口ルーチン (DFSLGFX0) を呼び出します。

**推奨事項:** システム情報を保守するログオン出口ルーチン (DFSLGNX0) を提供している場合には、その処理を補うログオフ出口ルーチン (DFSLGFX0) も提供するようにしてください。必ず、そのログオフ出口ルーチンが、IMS と通信するすべての非 MSC 端末 および非 LU 6.2 端末を取り扱うようにしてください。

ログオフ出口ルーチンを使用すると、ログオンされている端末の数を保守することができます。

## ETO 制御ブロック削除によるパフォーマンスの向上

ETO を使用して、IMS は制御ブロックを動的に削除することができます。制御ブロックを動的に削除すると、ストレージの使用量が削減され、パフォーマンスを高めることができます。

特定の端末処理オプション (TRACE および STOPPED) がリセットされている場合、以下のいずれかが起こると IMS はセッション制御ブロックを削除します。

- サインオンしているユーザーがなく、かつチェックポイントが発生した。
- MTO コマンドまたは自動ログオフ・タイムアウトのいずれかにより、セッションが正常終了または異常終了している。

ノードが 3600/ 金融機関、または SLU P 端末の場合には、メッセージ再同期が必要です。制御ブロックは、ウォーム・セッション終了の後には削除されませんが、/CHANGE NODE COLDSESS コマンドの後には削除されます。ISC 端末の場合、制御ブロックはコールド・セッション終了後に削除されます。ただし、ISC のウォーム・セッション終了後は制御ブロックは削除されずに残ります。

ユーザーが、/SET、/TEST MFS、または /EXCLUSIVE コマンドを発行した後に存在するような特定の処理オプションをリセットし、/SIGN OFF コマンドを発行した (あるいは自動的にサインオフされた) 場合、IMS は下記の条件の下でユーザー制御ブロックを削除します。

- このユーザーに関連するすべての LTERM へのキューにメッセージがない。
- ユーザーが会話型、高速機能モード、全機能応答モードのいずれでもない。

上記の条件に適合する場合、動的に作成されたユーザー制御ブロックは削除されます。上記の条件に適合しない場合、条件に適合するようになるまで、あるいは IMS のコールド・スタートが発生するまで、ユーザー制御ブロックは存在し続けることができます。特定の処理オプションのリセット後、他のすべての削除基準を満たしている場合、制御ブロックは次のチェックポイントで削除されます。

**重要:** ユーザー制御ブロックは、SAVE キーワードが指定された /CHANGE または /ASSIGN コマンドを使用すれば、セッション間および IMS 再始動間で保管することができます。これらのユーザー制御ブロックは、NOSAVE キーワードが指定されたコマンドが再び入力されるまで保持されます。

SRMDEF=LOCAL および RCVYRESP=YES の両方が指定されている場合、全機能応答モードの端末またはユーザーについては、IMS は端末のログオフまたはユーザーのサインオフの後もユーザー制御ブロックを削除しません。この場合は全機能応答モードがリカバリー可能であるためです。

**注:** IMSplex 内で、状況リカバリー・モードが GLOBAL または NONE の場合、ローカル制御ブロックはログオフまたはサインオフの直後に削除されます。

## IDCO トレース機能

IDCO トレース機能を使用して、ログオンおよびログオフのエラーを診断することができます。

### このタスクについて

この機能は、IMS メッセージ DFS3672I では提供できない情報を提供します。この機能を使用するには、/TRACE SET ON TABLE IDCO と入力します。この機能は、以下のイベントをトレースします。

- IMS VTAM 出口ルーチン (モジュール DFSCNXAO 内) で発生したエラー。このエラーは、IDCO トレース機能が有効であるかどうかに関係なく、DFS3672I メッセージでも示されます。
- 存在しない VTAM ノードにログオンしようとしたとき (存在しない端末に対して /OPNDST コマンドを入力したときなど) に発生するエラー。IMS は、関連メッセージ DFS2061I または DFS2062I を出します。
- IMS VTAM 出口ルーチンが処理を完了した時刻と、通常の IMS 処理によって要求が受け入れられる時刻との間に発生する同期のずれ。結果は、VTPO スtring で示される X'6701' ログ・レコードです。

### 関連資料

[/TRACE TABLE コマンド \(コマンド\)](#)

[VTPO ID を使用した 6701 ログ・レコードのフォーマット \(診断\)](#)

[IDCO トレース・テーブル項目 \(診断\)](#)

## ETO と LU 6.1 (ISC) 端末

LU 6.1 (ISC) 端末の場合、IMS は同じノード名への並列セッションをサポートします。この場合、別の構造がセッションごとに構築されます。ただし、各セッションとその関連構造は、別個の端末のように独立して機能します。

ISC は、BIND 内の SNA 定義のユーザー・データ域をサポートします。ISC のセッションを設定した場合、各ハーフセッション・パートナーは、ログオン時にユーザー・データとして組み込まれた適切なセッション修飾子で識別されます。DFSLGNXO 出口ルーチンを使用して、これら 2 つの修飾子を指定することはできません。各ハーフセッションに 1 つずつ含まれます。IMS は、現行ハーフセッションのセッション修飾子をユーザー構造として使用します。このユーザー構造は、関連した LTERM キューの集合を割り振り、必要に応じて自動的に RACF サインオンを行うために使用されます。もう一方のハーフセッション修飾子は IMS ユーザー構造と一緒に保管されます。両修飾子とも、セッション再始動要求とセッション障害後の SNA STSN メッセージ再同期に使用されます。

**制約事項:** ユーザー・データの SNA 事前定義済み形式は、非 ISC エンド・ユーザー形式のパラメーターとオプションをすべてサポートするわけではありません。

- RACF パスワードとグループ名はサポートされません。IMS では、セッション開始の一部としてユーザー構造を割り振るとき、PASSCHK=NO を指定して ISC の RACF サインオンができます。
- LOGOND と USERD はサポートされません。ログオン出口ルーチンおよびサインオン出口ルーチンで指定されていない限り、IMS はデフォルトを使用します。
- ISC 端末に対して自動ログオンが生成されると、ユーザー記述子内の AUTLDESC キーワードは無視され、ユーザー・データ内の LOGOND キーワードは省略されます。

### 関連タスク

[95 ページの『デフォルト CINIT または BIND ユーザー・データ形式の使用』](#)



それぞれのセッション開始要求に、VTAM CINIT または BIND ユーザー・データを組み込んで、ログオン記述子またはサインオン・データを提供することができます。インストール先でこのデータを処理するためのログオン出口ルーチンを用意することができます。

#### 関連資料

[ログオン出口ルーチン \(DFSLGNX0\) \(出口ルーチン\)](#)

## ETO および STSN 端末

このトピックでは、STSN 端末に対する ETO の管理について説明します。

### SNA STSN 端末考慮事項

SNA STSN 機能 (金融機関、SLU-P、および ISC) を使用する ETO 端末では、ログオン時にユーザー・キュー名を提供する必要があります。それは、このタイプの端末に対する IMS 接続プロセスの一部であるシーケンス番号交換を解決するためにこのキューが使用されるからです。

#### このタスクについて

ユーザー名の提供方法としては、以下のものがあります。

- 端末によって送信される CINIT データ。
- VTAM 非送信請求システム・サービス (USS) などの、別個の端末ホスト製品を使用して提供される CINIT データ。
- ISC を除くユーザー・ログオン出口ルーチン (DFSLGNX0)。
- IMS コマンド (/OPNDST ではユーザー名を指定することができます)。
- ユーザー記述子または宛先作成出口ルーチンによって提供される自動ログオン・パラメーター。

ログオン出口ルーチンがユーザー名を提供する最後の機能です。ETO STSN 端末にユーザー名が提供されないと、ログオンは拒否されて、エラー・メッセージが戻されます。これは、ユーザーがログオンをした後サインオンできる他の端末タイプとは少し違います。サインオンは、IMS アクティビティー (トランザクションやコマンド) のために端末を使用できるようにするために必要であるため、相違点は、STSN 端末の場合よりも早い時期にサインオン・データが必要になるということだけです。/SIGN コマンドは STSN 端末用にサポートされています。

### ETO、3600/ 金融機関、および SLU P

静的にシステム定義された 3600/ 金融機関と SLU P 端末には、/SIGN コマンドを使用するかログオン・ユーザー・データを使用することでサインオンできます。

#### このタスクについて

他の IMS /SIGN コマンドを使用するといつでも、サインオン識別を変更できます。LTERM は、システム定義時に端末に割り当てられますが、サインオン・プロセスや SNA STSN メッセージ・リカバリー・プロセスによって影響を受けることはありません。サインオンは単にユーザー・アクセスと入力許可をするだけです。

IMS は、3600/ 金融機関および SLU-P 端末を動的端末としてサポートします。これらの端末では、ログオン要求で自動ログオン・データかサインオン・データを使用して、ユーザー構造を動的に割り振ることができます。ETO 3600/ 金融機関および SLU-P 端末の場合は、セッション開始時にサインオン・データを提供しなければなりません。サインオン・データは、セッションのコールド・スタート時にログオン出口ルーチン (DFSLGNX0) で提供することができます。コールド・スタート・セッションで割り振られた LTERM とユーザー ID は、VTAM STSN メッセージ再同期要求により、複数セッションにわたって、さらに IMS 障害後も保存されます。このため、後続のウォーム・スタート・セッションで同じユーザー・サインオン・データを使用してユーザーを再検証し、メッセージ再同期を許可することが必要です。

動的 XRF 金融機関および SLU-P 端末が XRF クラス 2 として定義されている場合には、テークオーバー時に自動の再サインオンとログオンが行われます。

## ETO STSN 装置 (ISC、金融機関、および SLU P) に対する /SIGN サポート

ETO STSN 装置では、ユーザー構造を作成するためにセッションが割り振られるときには、ユーザー・データが必要になります。

### このタスクについて

ユーザー構造が作成され、端末に割り振られると、ETO STSN 端末からの **/SIGN** コマンドは受け入れられます。

ETO STSN 端末から **/SIGN** コマンドを出すと、IMS は完全なサインオン・プロセスを開始して、新規ユーザーのセッションに関連するセキュリティー・プロファイルを作成します。

ユーザーがサインオフすると、ユーザー・セキュリティー・プロファイルは削除され、セッションはセキュリティーなしのまま残ります。RACF で保護されている資源へのアクセスは RACF がすべて拒否し失敗します。失敗したリソースがコマンドの場合は、DFS3662 メッセージが表示されます。また失敗したリソースがトランザクションの場合は、DFS2469 メッセージが表示されます。あるユーザーが STSN 装置にサインオンすると、セッション割り振り時に割り振られた同じユーザー構造が、新規ユーザーにも使用されます。IMS はセッションのセキュリティー・プロファイルを更新して、サインオン情報を保管します。

**/SIGN** コマンドを用いて、ETO 金融機関、SLU-P、または ISC 装置にサインオンしているユーザーのユーザー ID は、ユーザー構造名とは異なる名前です。これらの装置のユーザー構造はすでにセッション開始中に作成されているため、これらのユーザーは、複数のサインオンをサポートするためのサインオン出口ルーチン (DFSSGNX0) による接尾部の付加は必要ありません。

**制約事項:** 金融機関、SLU-P、または ISC 装置に割り振られたユーザー構造を変更することはできないため、サインオン出口ルーチンの使用可能オプションのほとんどはサポートされず、作業域がサインオン出口ルーチンに渡されません。

DFS3650 メッセージはサインオン後に表示されて、DFS058 メッセージはサインオフ後に表示されます。DFS3650 メッセージのユーザー・フィールドは RACF ユーザー ID よりむしろユーザー構造名を反映しています。これは他のどの ETO 端末も同じです。

**制約事項:** **/SIGN** コマンドを STSN 出力専用装置から出すことはできません。

### 関連概念

[IMS 実行パラメーターの指定 \(システム定義\)](#)

[出口ルーチン \(出口ルーチン\)](#)

## ETO を使用した会話モードおよび応答モード

端末状況が、端末ごとではなくユーザーごとに保守されるため、ETO ユーザー構造は静的端末構造とは異なります。

静的端末での IMS の会話は、たとえユーザー名が同じでも動的端末での IMS 会話とは異なります。静的端末での会話は、その静的端末に保留されることがありますが、また、保留解除 (再開) されなければなりません。動的端末に移動することはできません。

ETO を使用すると、IMS 会話は同じ端末または他の動的端末で再開できます。会話はユーザー構造の 1 属性であるため、通常は、ユーザーが IMS にサインオンすると、別の端末までそのユーザーを追跡します。ユーザーが、どの端末が静的でどの端末が動的か識別できない場合、付属の会話属性が混乱を起こす原因になることもあります。さらに、出口ルーチンが、異なる物理端末へのサインオンに異なるユーザー構造を選択すると、混乱が起きることもあります。

全機能および高速機能の応答モードは同じ静的端末で再開されなければなりません。ETO の場合は、全機能および高速機能の応答モードは、同じ動的端末または別の動的端末からでも再開することができます。すべての動的端末からの高速機能応答モードを再開する機能は、前段落で説明した会話と状態が似ており、同じ考え方が適用されます。

## 会話モード

ETO の場合、会話は、会話を開始する端末ではなく、ユーザーと関連付けられます。また、会話は端末とも関連付けられますが、それはユーザーがサインオンしている間だけです。サインオフすることにより、ユーザーは別の端末で会話を続けることができます。この柔軟性を持たせるためには、インストール・システムで出力フォーマット設定問題に取り組む必要があります。

応答モードではない会話中のユーザーは、サインオフすることができます。応答モードであるか否かに関係なく、会話中のユーザーは、自動サインオフあるいは MTO コマンドを使用することによって自動的にサインオフすることができます。いずれのサインオフ形式でも、後のユーザーのために端末は利用可能状態にしておきます。会話モード状況は、そのユーザーを次の端末まで追跡します。あるいは、リソース・マネージャーを使用して、IMSplex 内の別の IMS 上でグローバル状況リカバリー・モード (SRM=GLOBAL) になります。

## 応答モード

応答モードは、静的端末には TERMINAL マクロで、動的 (ETO) ユーザーには ETO ユーザー記述子で、さらにトランザクションには TRANSACT マクロで定義されます。また、応答モードは全機能か高速機能のいずれかです。また、高速機能で実行していない場合には、会話モードと一緒に応答モードを使用することもできます。応答モードは、動的端末ではなく、主にユーザーおよびトランザクションと関連付けられます。

ユーザーが応答モードであるとき、キーボードまたは入力応答は、出力応答が利用可能になるまでロックされます。この間、その端末からは入力できません。通常のサインオフとログオフ・コマンドは使用できませんが、これらの機能は、異常セッション終了時でも自動的に行われる可能性があります。それは、以下のような場合に起こります。

- VTAM がエラーを検出して異常終了した場合
- MTO が IMS /CLSDST あるいは /STOP コマンドを出した場合
- IMS が指定した自動サインオフ間隔後に自動サインオフした場合

サインオフがどのように行われたかに関係なく、RCVYRESP=YES または RCVYFP=YES の場合は、自動サインオフしたユーザーの応答モードが保存されます。そのユーザーの応答モード操作は、そのユーザーが次にサインオンした端末で再確立され、応答モードの出力応答が使用可能になるまで残されます。

マスター端末オペレーターは、マスター端末から /STOP USER コマンドと /START USER コマンドを順番に発行することによって、応答が返される前に ETO 動的ユーザーの高速機能応答モードをリセットできます。マスター端末オペレーターは、マスター端末から /STOP NODE コマンドと /START NODE コマンドを順番に発行することによって、静的ノードの高速機能入力応答モードをリセットすることもできます。

## 関連概念

[108 ページの『非発信元端末への出力メッセージの送達』](#)

IMS は、ユーザーの出力をそのユーザーがサインオンされている端末へ送信します。





---

## 第 4 部 外部サブシステム 接続機能

IMS では、IMS システムから外部サブシステムにアクセスするためのいくつかのオプションを使用できます。

### このタスクについて



## 第 6 章 DB2 接続機能

IMS 内の Java メッセージ処理プログラム (JMP) および Java バッチ・プログラム (JBP) は、DB2® リソース・リカバリー・サービス (RRS) 接続機能 (以下、DB2 接続機能と呼びます) を使用して Db2 for z/OS データにアクセスできます。

### このタスクについて

このサポート用にセットアップされた各従属領域は、Db2 for z/OS データにアクセスするための独自の RRSF スレッドを構築します。このスレッドにより、アプリケーション・プログラムが IMS と Db2 for z/OS の両方のリソース・マネージャーのリソースに対して行う更新を調整することができます。IMS JMP および JBP が Db2 for z/OS データのアクセスに DB2 接続機能を使用する場合、IMS は、ESAF を使用する場合のような更新とコミットの同期点コーディネーターではありません。DB2 接続機能を使用する場合、IMS は参加者であり、z/OS リソース・リカバリー・サービスが同期点コーディネーターになります。

## DB2 接続機能を使用するためのシステムの準備

DB2 接続機能を使用できるようにシステムを準備するには、2 つのタスクを実行する必要があります。

### このタスクについて

DB2 接続機能を使用するには、以下のようになります。

### 手順

1. IMS PROCLIB データ・セットに接続機能の定義を追加する。

DB2 接続機能を使用するには、サブシステム・メンバー (SSM) が IMS.PROCLIB に定義されている必要があります。IMS.PROCLIB に SSM メンバーがまだ定義されていない場合は、作成しなければなりません。SSM には、IMS とアプリケーション・プログラムが連絡する Db2 for z/OS システムのエントリが含まれます。IMS 内のすべての Java 従属領域は、単一の Db2 for z/OS システムにアクセスします。

2. IMS JMP および IMS JBP 領域から Db2 for z/OS RESLIB を使用可能にする。

接続機能の定義後は、IMS 内の Java 領域から Db2 for z/OS RESLIB にアクセスできるようにしなければなりません。JBP および JMP 領域タイプの JCL に、RRSAF で使用されるモジュールの入った Db2 for z/OS ライブラリーをポイントする DFSDB2AF DD ステートメントを使用して、Db2 for z/OS ライブラリーを一定義を追加します。Db2 for z/OS ライブラリーは APF 許可されている必要があります。

### 関連概念

[外部サブシステム・データへのアクセス \(システム定義\)](#)

## Java 従属領域の Db2 for z/OS アクセス方法の管理

IMS 制御領域の開始時に DB2 接続機能定義が組み込まれていれば、IMS 従属領域で実行されている Java アプリケーション・プログラムは、z/OS リソース・リカバリー・サービスの同期点制御の下で Db2 for z/OS にアクセスできます。

IMS 制御領域の初期設定処理によって、Db2 for z/OS へのアクセスが準備されます。その後 Java 従属領域が開始されると、その領域にあるアプリケーション・プログラムは Db2 for z/OS と IMS の両方を直接呼び出せるようになります。

DB2 接続機能の初期設定によって、その他のタイプの従属領域の実行が影響を受けることはありません。DB2 接続機能の定義は、IMS Java 従属領域が使用されない場合でも、IMS.PROCLIB メンバーに保存できます。IMS.PROCLIB メンバーに DB2 接続機能の定義が存在し、Db2 for z/OS ライブラリーが Java 従属領域 JCL に定義されている場合、開始されるすべての Java 従属領域によって Db2 for z/OS へのアクセス・スレッドが構築されます。DB2 接続機能の定義は存在しているが、Java 従属領域が Db2 for z/OS へのアクセスを必要としない場合は、Db2 for z/OS システムへのアクセスを停止することにより、アクセス・ス

レッドが構築されないようにすることができます。**/STO SUBSYS** コマンドを使用してください。このコマンドは、その後 **/STA SUBSYS** コマンドが実行されるまで有効です。

#### 関連資料

[/START SUBSYS コマンド \(コマンド\)](#)

[/STOP SUBSYS コマンド \(コマンド\)](#)

## 第 7 章 External Subsystem Attach Facility (ESAF)

External Subsystem Attach Facility (外部サブシステム接続機能) は、BMP、IFP、JBP、JMP、および MPP の各アプリケーション・プログラムが、DL/I データベースに加え、他のサブシステムによって管理されているデータベースにアクセスできるようにします。

外部サブシステム (ESS) プロダクトのデータ・リソースに IMS アプリケーションからアクセスできるようにするには、ESS は、IMS サブシステムに接続し、IMS と共にデータ・アクセスを調整するために必要な機能を提供する必要があります。IMS 接続機能は、外部サブシステム・プロダクトへのプログラミング・インターフェースを提供します。ESAF をインストールするには特定の手順が必要です。この手順については以下で説明します。他の IMS 資料にも ESAF 情報が含まれており、該当する場合は参照が組み込まれています。

複数の外部サブシステムを、1 つのオンライン IMS システムだけで接続できます。これらのサブシステムは、同じプロダクト・タイプのもでも異なるタイプのもでもかまいません。このインストールで外部サブシステムを IMS に対して定義します。IMS の特定の従属領域は、インストール時の指定に従って、この IMS システムに定義された外部サブシステムすべてに対してアクセス可能、その中の一部に対してアクセス可能、またはどの外部サブシステムにもアクセス不能です。従属領域で実行中のアプリケーション・プログラムは、タイプの異なる複数のサブシステムにアクセスすることができます。インストールでは各サブシステムごとに固有のトークンを定義し、IMS はこのトークンを用いて、外部リソースへのアプリケーションの呼び出しを経路指定します。IMS では、同じタイプの複数のサブシステムをアプリケーション・プログラムからアクセスすることができますが、外部サブシステムではこれを行えないことがあります。

この機能は、外部サブシステムのデータ・リソースを IMS のデータ・リソースと同期させます。同期化処理では、IMS はリカバリー・コーディネーターであり、アプリケーション・プログラムに代わってコミット・アクションまたは打ち切りアクションを指示する責任があります。外部サブシステムは、この処理に関与しており、IMS の指示に従って、IMS アプリケーションによるデータ更新のコミットと打ち切りを行います。リソースをコミットするとき、IMS は関与サブシステムでコミットできるかどうかをポーリングしてから、最終的なコミット (または打ち切り) の指示を出します。

外部サブシステム上の作業をリカバリーするために、高速データベース・リカバリー (FDBR) 領域を構成することもできます。FDBR 領域は、モニター中の IMS システムに障害が起きた場合、ESAF 未確定通知出口ルーチン (DFSFDNO) から外部サブシステム上の未確定作業に関する情報を受け取ります。

IMS バッチでは、IMS は、1 つの外部サブシステムだけに接続することが許されています。IMS は、この外部サブシステムをリカバリー・コーディネーターと見なします。この外部サブシステムは IMS が接続する他の外部サブシステムと調整する手段を持っていないため、バッチでは、IMS は 1 つの外部サブシステム接続だけに制限されることになります。

外部サブシステム接続機能は、DBCTL に関連するコーディネーター・コントローラー (CCTL) とは異なります。

JMP および JBP 領域は、IMS.PROCLIB メンバーで COORD=RRS パラメーターを使用する、制御領域で定義された Db2 for z/OS サブシステムにアクセスできます。JMP 領域または JBP 領域に対してこの接続方法が選択された場合は、Db2 for z/OS ライブラリーを指す DD ステートメント (DFSDB2AF) を DFSJMP プロシージャまたは DFSJBP プロシージャに追加する必要があります。

このトピックには汎用プログラミング・インターフェース情報が含まれています。

### 関連タスク

#### 117 ページの『DB2 接続機能』

IMS 内の Java メッセージ処理プログラム (JMP) および Java バッチ・プログラム (JBP) は、DB2<sup>®</sup> リソース・リカバリー・サービス (RRS) 接続機能 (以下、DB2 接続機能と呼びます) を使用して Db2 for z/OS データにアクセスできます。

JMP アプリケーションまたは JBP アプリケーションからの Db2 for z/OS データベースへのアクセス (アプリケーション・プログラミング)

## 外部サブシステムで用意するもの

外部サブシステムは、外部サブシステム接続パッケージ (ESAP)、外部サブシステム・モジュール・テーブル (ESMT)、およびリソース変換テーブル (RTT) の3つを用意する必要があります。

### 外部サブシステム接続パッケージ (ESAP)

IMS 接続機能は出口ルーチン・インターフェースを使用します。つまり、従属領域から外部サブシステムにアクセスするために、IMS は処理の特定の時点で出口ルーチンを活動化します。これらの出口ルーチンは、外部サブシステムが用意する必要があります。この出口ルーチンの機能は IMS により規定されます。外部サブシステムは、それ自体の固有のインプリメンテーションを提供します。実際に、この出口ルーチンが外部サブシステムへのリンケージそのものを提供する必要があります。IMS 自体は、使用されるこのリンケージ・メカニズムの影響を受けません。

IMS は、外部サブシステムにアクセスできる制御領域と個々の従属領域に、外部サブシステム提供の出口ルーチン・モジュールをロードします。外部サブシステムは、接続出口ルーチンの処理に必要な追加モジュールを提供することができます。IMS はこれらのモジュールもロードします。IMS 領域での接続処理のために用意されている外部サブシステムのモジュールが、IMS で通常呼ばれている、外部サブシステム接続パッケージ (ESAP) になります。

### 外部サブシステム・モジュール・テーブル (ESMT)

外部サブシステムは、IMS がロードするモジュールを外部サブシステム・モジュール・テーブル (ESMT) に指定しなければなりません。外部サブシステムは、IMS 提供のマクロを用いてモジュール・テーブルを作成し、お客様システム環境で使用できるようにします。このシステム環境では、IMS に対してこの ESMT 名を指定します。これを行うには、IMS に対して外部サブシステムを定義時にその名前を指定します。

### リソース変換テーブル (RTT)

IMS は、PSB (プログラム仕様ブロック) を用いて、アプリケーション・プログラムに必要な DL/I リソースを定義します。MPP の場合は、PSB 名はアプリケーション・プログラム名と同じです。BMP または IFP の場合は、PSB 名はこれと異なる場合があります。外部サブシステムは、アプリケーション・プログラムに必要な外部サブシステム・リソースを定義するために使用するエンティティに、PSB 名や IMS アプリケーション・プログラム名以外の名前を使用することができます。外部サブシステムが異なる名前を使用する場合は、その外部サブシステムは、PSB 名またはアプリケーション・プログラム名をサブシステムのエンティティ名にマッピングするための、リソース変換テーブル (RTT) を提供することができます。

外部サブシステムが RTT を作成し、インストール・システムで使用できるようにします。インストール・システムは、IMS に外部サブシステムを定義するときに RTT の名前を指定します。IMS は、ESAP をロードするときに RTT をロードします。

実際にマップを行うのは、外部サブシステムです。IMS は RTT にはアクセスしません。このテーブルをロードし、ESAP がそのテーブルのアドレスを使用できるようにするだけです。IMS は RTT のフォーマットを規定しません。

#### 関連概念

127 ページの『外部サブシステム・モジュール・テーブルの作成』

外部サブシステムは、IMS がロードする外部サブシステム・モジュールの定義を入れておく外部サブシステム・モジュール・テーブル (ESMT) を作成します。

## IMS への外部サブシステムの指定方法

IMS.PROCLIB メンバーでは、IMS アプリケーションでアクセスする予定の外部サブシステムをすべて定義します。制御領域の EXEC ステートメントは、SSM パラメーターでこのメンバーを指します。

### このタスクについて

IMS に対して定義する外部サブシステムごとに、PROCLIB メンバーで指定します。

- 外部サブシステムのタイプ

- 外部サブシステムの z/OS 名
- 外部サブシステム 接続パッケージ (ESAP) 内のモジュールを指定する、外部サブシステム・モジュール・テーブル (ESMT) の名前。
- IMS がアプリケーション呼び出しを外部サブシステムに経路指定するために使用する言語インターフェース・トークン (LIT)。
- IMS アプリケーション・プログラムが必要とする外部リソースを識別するために、必要な場合に外部サブシステムによって提供されるリソース変換テーブル (RTT) の名前。
- IMS がオペレーター・コマンドを外部サブシステムに経路指定するために使用するコマンド認識文字 (CRC)。
- 外部サブシステムへのアプリケーション呼び出しが処理できない場合に取りられる処置を示す領域エラー・オプション (REO) コード。リソース変換テーブル (RTT) を使用すると、RTT で指定された OPTION 値によって SSM メンバーの REO オプションが上書きされます。

また、外部サブシステム提供のテーブル (ESMT と RTT) を適切なロード・モジュール・ライブラリー内に用意する必要があります。

従属領域用の外部サブシステム定義を用意することもできます。これらのタイプの従属領域に対して SSM EXEC パラメーターが指定されていない場合、領域では、制御領域に定義されたすべてのサブシステムにアクセスすることができます。EXEC パラメーターの SSM を指定すると、従属領域ではその PROCLIB メンバーで定義されているサブシステムだけにアクセスすることができます。(これらのサブシステムも、制御領域に対して定義されていなければなりません。) 従属領域でどの外部サブシステムもアクセスしない場合は、ダミーの PROCLIB メンバー (つまり、定義が入っていないメンバー) を使用します。

**注:** JMP 領域および JBP 領域は、IMS.PROCLIB メンバーで COORD=RRS パラメーターを使用する、制御領域で定義された Db2 for z/OS サブシステムにもアクセスできます。DFSJMP および DFSJBP プロシージャで、Db2 for z/OS ライブラリーを指す DD ステートメント (DFSDB2AF) を追加します。

外部サブシステムを IMS に接続するには、以下のタスクを実行する必要があります。

外部サブシステムを IMS に接続するには、次のようにします。

## 手順

1. IMS に外部サブシステムを定義します。
  - a) IMS プロシージャ・ライブラリー (IMS.PROCLIB) にメンバーを追加します。このライブラリーには、IMS と連絡する外部サブシステムそれぞれに関する情報が入っています。
  - b) IMS 制御領域または従属領域の EXEC ステートメントでは、前のステップで作成した PROCLIB メンバーを SSM パラメーターに指定します。  
Db2 for z/OS グループ名を使用して Db2 for z/OS データベースにアクセスする場合は、従属領域の EXEC ステートメントにそのグループ名を指定する必要があります。Db2 for z/OS グループ名は、IMS 制御領域の EXEC ステートメントには指定できません。
2. IMS が提供する言語インターフェース・モジュール以外のものを使用する場合は、その言語インターフェース・モジュールを定義します。
3. IMS OPTIONS ステートメントで、外部サブシステム・リンクのトレースを行うかどうかを指定します。
4. 外部サブシステムの場合は、ESMT とオプションの RRT を指定します。
5. IMS が使用する外部サブシステムのモジュールとデータベースを、必ず適切な APF 許可ライブラリーに入れておいてください。

## 関連概念

外部サブシステム・データへのアクセス (システム定義)



## 接続処理の基本原則

IMS 制御領域から外部サブシステムへ接続を確立することによって、外部サブシステムは IMS サブシステムに接続されます。

その外部サブシステムにアクセスする個々の従属領域からも、接続が確立されます。これらの接続を開始するのは、IMS です。

### サブシステム接続

IMS アプリケーション・プログラムと外部サブシステムとの間の接続を、スレッドと呼びます。アプリケーション・スレッドとは、IMS アプリケーション・プログラムと外部サブシステム・リソースとを結ぶ両方向の通信パスのことです。

1 回の実行で複数の外部サブシステムにアクセスするアプリケーション・プログラムは、複数のスレッドをもつことができます。ただし、同じタイプの複数のサブシステム (同じサブシステム・タイプの複数のインスタンス) へのアクセスは IMS ではサポートされても外部サブシステム・プロダクトではサポートされないことがあります。

### 接続の確立

IMS は、「識別」プロセスを使用して、外部サブシステムへの接続を確立します。

IMS は、外部サブシステムに対する制御領域または従属領域 TCB を識別するために、ESAP に含まれている識別出口ルーチンを活動化します。すると、外部サブシステムは、IMS の異常終了に備えて IMS の TCB をモニターできます。この識別プロセスが正しく完了した時点、つまり、領域を外部サブシステムに正しく知らせた時点で接続が確立されます。

IMS には通知メッセージ・メカニズムが備わっているので、IMS が制御領域に接続しようとしたときに外部サブシステムがまだ開始されていなかった場合は、その外部サブシステムが開始されていると、外部サブシステムは、IMS に接続を確立させるよう通知することができます。外部サブシステムがこの通知機能を使用する場合は、IMS と外部サブシステムの開始順序は重要ではありません。

制御領域からの接続がまず確立されてから、従属領域接続が確立されます。従属領域の開始時点で制御領域接続がまだ確立されていない場合、外部サブシステムはその従属領域を識別しません。IMS は、制御領域接続と従属領域接続間の階層関係を使用して、制御領域が従属領域のリカバリー・コーディネーターとして機能できるようにしています。従属領域で障害が起こると、従属領域に代わって制御領域がリカバリー処置を取ります。

外部サブシステムは初期設定出口ルーチンを用意することもできます。初期設定出口ルーチンがある場合、IMS は、制御領域と従属領域の初期設定の時点で、領域が自身を外部サブシステムに識別させる前に、初期設定出口ルーチンを活動化します。外部サブシステムは、接続が確立される前に、外部サブシステムに必要な初期設定処理をこの出口ルーチンで行うことができます。

外部サブシステムがすでに開始されていれば、IMS は制御領域の初期設定中に制御領域接続を自動的に確立します。ただし、接続を延期することができます。初期設定出口ルーチンが用意されていない場合、または出口ルーチンから相応の戻りコードが戻された場合、制御領域の識別は自動的に行われません。この場合、外部サブシステムが接続を確立する必要があるときに、IMS 提供のサブシステム始動サービスを活動化することができます。または、従属領域を識別できるときに、IMS が接続を確立しようとします。

**/START SUBSYS** オペレーター・コマンドが出されても、IMS は制御領域接続を確立します。

### ユーザー許可処理

従属領域接続が確立された後、その領域で処理される IMS トランザクションに関連したユーザー ID を外部サブシステムに通知するために、サインオン処理が実行されます。IMS は、この目的のために外部サブシステムが提供するサインオン出口ルーチンを活動化します。アプリケーションのスレッドを作成するためには、この領域に関する最初のサインオンが成功しなければなりません。

アプリケーション・プログラムの実行中 (つまり、スレッドの作成後) にサインオン処理が再び行われます。サインオン出口ルーチンは、アプリケーション・プログラムが処理する各メッセージを処理するごとに活動化されます。従属領域の識別後に実行される最初のサインオンは、アプリケーションが処理する最初のメッセージ (メッセージ・キューへの最初の固有の読み取り呼び出し) と関連しています。後続の各メッセ



ージが処理されるたびに、新しいユーザー ID を渡すためにサインオン出口ルーチンが再度活動化されます。つまり、複数モードのトランザクションの場合は、間にコミット処理をはさまずに、複数回サインオンが行われることになります。

外部サブシステムはサインオフ 出口ルーチンを用意し、IMS は、従属領域の接続を終了する前にこの出口ルーチンを活動化します。1つのアプリケーションで複数回のサインオンがある場合、新メッセージのサインオンが新たに行われる前にサインオフ処理が行われることはありません。新規サインオンが直前のサインオンに取って代わります。

## アプリケーション・スレッド

外部サブシステム 所有のデータ・リソースを初めてアプリケーション・プログラムが呼び出したときに、そのアプリケーションを外部サブシステムに接続するスレッドが作成されます。

IMS は、ESAP の中に用意されているスレッド作成出口ルーチンを活動化して、外部サブシステムに対してアプリケーション・プログラムを識別させます。外部サブシステムは、必要に応じて特定のアプリケーション・プログラムからのデータ要求を受け取れるようになっていなければなりません(つまり、リソースを確保し、処理構造を作成する、など)。アプリケーションが終了すると、IMS はスレッド終了出口ルーチンを活動化してスレッドを終了します。

## 接続の終了

識別終了出口ルーチン を ESAP 内に用意しておかなければなりません。IMS は、接続の終了が必要な場合にこの出口ルーチンを活動化します。

制御領域接続の終了処理は、IMS、外部サブシステム、またはオペレーター・コマンド (**/STOP SUBSYS**) によって開始することができます。IMS をシャットダウンすると、IMS はこの接続を終了します。**/STOP SUBSYS** コマンドは、この接続を終了させ、停止状態にします。IMS は、**/START SUBSYS** コマンドが処理されるまで、この接続の再確立を許可しません。

外部サブシステムは、2つのうちのいずれかの方法で制御領域接続の終了を要求することができます。1つの方法は、終了 ECB に通知することにより終了要求します。IMS は、外部サブシステムが終了時に使用できるように、識別出口ルーチンの呼び出しのときに ECB アドレスを渡します。外部サブシステムが終了 ECB に通知すると、IMS は、従属領域接続の休止を許可してから、その接続を終了して停止状態にします (**/STOP SUBSYS** コマンドの場合と同じ)。制御領域接続を終了するもう1つの方法は、外部サブシステム 出口ルーチンから IMS 提供のサブシステム終了サービスを活動化する方法です。

制御領域内で識別終了出口ルーチンを活動化した後で、IMS は、ESAP 内に用意されているサブシステム終了出口ルーチンを活動化します。この出口ルーチンは初期設定出口ルーチンの逆であると考えことができ、外部サブシステムはこの出口ルーチンを用いて作業域のリセット、ストレージの解放などを行えます。

外部サブシステムまたは IMS **/STOP SUBSYS** コマンドによって (制御領域) 接続の終了を IMS に要求しない限り、従属領域がアクティブである間は、従属領域接続が維持されます。一般に、外部サブシステムの識別終了出口ルーチンが従属領域用に起動されるのは、IMS が接続の終了を要求した場合だけです。したがって、従属領域が終了しても、必ずしもこの出口ルーチンが活動化されるわけではありません。このことは、サインオフ出口ルーチンについても言えます。識別終了出口ルーチンの呼び出しは、常にサインオフ出口ルーチンの呼び出しの後です。

外部サブシステムは、識別された IMS TCB を z/OS タスク終了出口ルーチンを通じてモニターし、その特定の TCB が終了するときには、必要なサインオフと識別終了プロセスを実行する必要があります。

従属領域と外部サブシステムとの間にスレッドが存在する場合は、従属領域接続が正常終了する前に必ずスレッド終了出口ルーチンが呼び出されます。したがって、サインオフ出口ルーチンと識別終了出口ルーチン (これらがある場合) が呼び出される前に、または従属領域の終了前に、スレッド終了出口ルーチンが呼び出されます。

IMS では、制御領域が接続されていない限り従属領域接続はありえないので、すべての従属領域が終了してしまうか、従属領域の識別終了出口ルーチンが活動化されるまで、制御領域の識別終了出口ルーチンは活動化されません。

従属領域については、サブシステム終了出口ルーチンは活動化されません。

## 照会パラメーターの処理

INQ パラメーターは、IMS トランザクションがスレッド作成出口ルーチンを発行した場合にのみ検査されます。

後続のトランザクションでは、INQ パラメーターは検査されません。このため、スレッド作成出口ルーチンとスレッド終了出口ルーチンの間に高速機能領域でなんらかの更新が行われる場合は、最初のトランザクション内の照会フラグは INQ=NO でなければなりません。

例えば、特定の高速機能領域から Db2 for z/OS を呼び出す最初のトランザクションが、Db2 for z/OS データを読み取ろうとするだけで更新を行わない場合は、スレッド作成パラメーター・リスト内に INQ=YES フラグを設定して、この最初のトランザクションとその高速機能領域内の後続のすべてのトランザクションが、照会のみトランザクションであることを示します。同じ高速機能領域で実行する後続のトランザクションが、更新のために Db2 for z/OS を呼び出した場合、そのトランザクションが正しく INQ=NO と定義されていたとしても、その高速機能領域のスレッドは INQ=YES に設定されることとなります。これは、SQLCODE817 エラーとなります。

## アプリケーション呼び出し処理

アプリケーション・プログラムから外部サブシステムへスレッドが作成されると、外部データ・リソースを要求するアプリケーション呼び出しは、ESAP 内の通常呼び出し出口ルーチンに渡されます。

アプリケーションにバインドされている言語インターフェースは、外部サブシステムへの呼び出しを処理するために IMS を活動化するとき、外部サブシステムの言語インターフェース・トークン (LIT) を渡します。ご使用のシステム環境で、IMS に定義するそれぞれの外部サブシステムごとに固有の LIT を指定します。IMS は、言語インターフェース・スタブで提供される LIT と、呼び出しを外部サブシステム宛てに経路指定するために定義されている LIT とを突き合わせます。

## リソースの調整

リカバリー・コーディネーターである IMS は、IMS アプリケーション・プログラムが開始した外部サブシステム・リソース更新のコミット処理を指示します。IMS は、2 フェーズ・コミット処理を用いて、外部サブシステム間のリソースを同期させます。外部サブシステムはこのプロセスに関与します。

アプリケーションのコミット処理の第 1 フェーズでは、IMS は、更新をコミットする用意ができていかどうか知るために関与サブシステムをポーリングします。第 2 フェーズで、IMS は関与サブシステムにコミットまたは打ち切りを指示します。第 1 フェーズですべての関与サブシステムが 'yes' と応答すると、IMS は、第 2 フェーズでそれらにコミットを指示します。そうでない場合は、IMS は打ち切りを指示します。

外部サブシステムのリソースが (コミット処理が不要な) 非更新トランザクションと関連していると外部サブシステムが判断した場合、外部サブシステムは、第 1 フェーズですべてのコミット処理を実行することができ、第 2 フェーズは必要ありません。この場合、外部サブシステムは、第 1 フェーズが正しく完了したので第 2 フェーズは不要であることを示す戻りコード 'XC' をセットして、コミット準備出口ルーチンから IMS に制御を戻します。IMS は、この外部サブシステムについてはコミット処理の第 2 フェーズを開始しません。

IMS は、1 つ以上のサブシステムにまたがる作業単位を識別するために 16 バイトのリカバリー・トークンを使用します。初めに、作業単位のリカバリー・トークンは、サインオン出口ルーチン呼び出しで渡されます。

アプリケーションの更新をコミットするとき、IMS は、外部サブシステムが用意したコミット準備出口ルーチンを活動化します。該当するリカバリー・トークンがその呼び出しで渡されます。外部サブシステムは、そのリカバリー・トークンのコミット処理を実行するための準備ができていかどうかを、出口ルーチンからの戻りコードによって伝えます。アプリケーションが分散シンコペート環境 (保護会話環境ともいいます) で実行中で、サブシステム SIGNON を必要とする場合は、IMS は XID トークンを取得し、そのトークンのアドレスを出口パラメーター・リストに入れてから、サブシステムの SIGNON 出口を呼び出します。

コミット処理が必要な場合、その第 2 フェーズでは、IMS は、3 つの外部サブシステム出口ルーチンのいずれかを活動化することができます。それは、コミット続行出口ルーチン、打ち切り続行出口ルーチン、そしてスレッド終了出口ルーチンです。アプリケーションが終了中ではなく、すべての関与サブシステム

がコミット可能な状態である場合、IMS はコミット続行出口ルーチンを開始します。コミット処理の完了時点で、アプリケーションは既存スレッドの現行 PSB の処理を継続します。更新は打ち切られるが、アプリケーションは終了しないかまたは終了中の場合、打ち切り 続行出口ルーチンが活動化されます。この場合、アプリケーションは同じリカバリー・トークンで処理を継続します。

外部サブシステムのスレッド終了出口ルーチンは、コミットの第 2 フェーズを処理できなければなりません。アプリケーションの終了時に、IMS はスレッド終了出口ルーチンの呼び出しでリカバリー・トークンとコミット・オプションを渡します。コミット・オプションで、未解決の更新をコミットするか打ち切るかを指示します。

IMS、外部サブシステム、またはアプリケーション・プログラムが異常終了すると、コミットも打ち切りもされていない作業単位は未解決のままになります。未解決の作業単位を解決するために、IMS は外部サブシステムの未確定解決出口ルーチンを呼び出します。IMS は、制御領域接続を確立した後で、未確定解決出口ルーチンを少なくとも 1 回は活動化します。IMS は、それぞれの未解決のリカバリー・トークンごとに 1 回ずつこの出口ルーチンを活動化して、作業単位をコミットするか打ち切るかを指示します。解決しなければならない作業単位がないか、または IMS が未解決のリカバリー・トークン・リストを処理してしまつた場合、IMS はこの出口ルーチンを活動化して、外部サブシステムにそのことを伝えます。IMS は、z/OS リソース・リカバリー・サービスに関連した未解決のリカバリー・トークンを検出すると、(ABORT または COMMIT の) どちらのアクションを取るべきかを RRS または IMS ユーザーが指示するまで、サブシステム未確定解決出口呼び出しを遅らせます。RRS 未確定解決のために呼び出される場合、リカバリー・トークンが正しい順序で解決されるようにするのは、サブシステムが行う必要があります。

IMS は、通常の再始動 (ウォーム・スタート) と IMS の緊急時再始動、およびサブシステムの再接続に備えて、未解決のリカバリー・トークンを維持しています。IMS は、すべてのリカバリー・トークンが解決されなくても (つまり、未確定解決出口ルーチンの戻りコードが、リカバリー・アクションが実行されなかったことを示している) 接続を許可します。IMS は、コールド・スタートのときは未解決のリカバリー・トークンを破棄します。

未確定解決出口ルーチンは、アプリケーション・プログラムの異常終了時にリソースを調整するためにも使用されます。アプリケーション・プログラムが異常終了すると、外部サブシステムへのスレッド接続があれば、制御領域からこの出口ルーチンが活動化されます。

外部サブシステム上の作業をリカバリーするために、高速データベース・リカバリー (FDBR) 領域を構成することもできます。FDBR 領域は、モニター中の IMS システムに障害が起きた場合、ESAF 未確定通知出口ルーチン (DFSFDNO) から外部サブシステム上の未確定作業に関する情報を受け取ります。z/OS リソース・リカバリー・サービスに関連した作業単位は、FDBR によってリカバリーされません。

## 関連概念

[高速データベース・リカバリー \(FDBR\) 領域 \(オペレーションおよびオートメーション\)](#)

## 外部サブシステム・コマンド・サポート

IMS 提供のコマンド (/SSR) を使用すれば、IMS オペレーターはコマンドを外部サブシステムに送信することができます。

IMS からコマンドを受信するためには、外部サブシステムがコマンド出口ルーチンを用意しなければなりません。IMS は、/SSR 入力に入っているコマンドを、この出口ルーチンに渡します。AOI (自動化操作プログラム・インターフェース) プログラムも、/SSR を用いて外部サブシステムにコマンドを送信することができます。/SSR コマンドの入力内に、このコマンドの宛先である外部サブシステムの識別 (コマンド認識文字 (CRC)) が入っています。サブシステムの CRC は、サブシステムを IMS に定義するとき定義の一部として指定します。

## 関連タスク

[120 ページの『IMS への外部サブシステムの指定方法』](#)

IMS.PROCLIB メンバーでは、IMS アプリケーションでアクセスする予定の外部サブシステムをすべて定義します。制御領域の EXEC ステートメントは、SSM パラメーターでこのメンバーを指します。

## 関連資料

[/SSR コマンド \(コマンド\)](#)

## ESAP で使用できる IMS サービス

IMS は、外部サブシステムが特定の IMS システム・サービスにアクセスするときに活動化できる出口ルーチンを備えています。

外部サブシステムは次のことを行えます。

- 接続の開始を要求する (サブシステム始動サービス)。
- 接続の静止を要求する (サブシステム終了サービス)。
- ログ・レコードを IMS ログに書き込む (ログ・サービス)。
- IMS の宛先にメッセージを送信する (メッセージ・サービス)。

### 関連概念

[出口ルーチン \(出口ルーチン\)](#)

## 第 8 章 外部サブシステム・モジュール・テーブルの作成

外部サブシステムは、IMS がロードする外部サブシステム・モジュールの定義を入れておく外部サブシステム・モジュール・テーブル (ESMT) を作成します。

外部サブシステムの出口ルーチン・モジュール、ならびに ESAP で必要な他のモジュールも、ESMT で定義しておきます。ご使用のシステム環境で、外部サブシステム定義の一部として ESMT の名前を IMS に知らせます。制御領域と従属領域での接続処理の初期設定中に、IMS は、ESMT をロードしてから、その中で定義されているモジュールをロードします。

外部サブシステムは、自身の ESAP で必要な作業域も ESMT で定義しておくことができます。作業域が定義されていると、IMS は、定義されているモジュールをロード後に、指定された作業域を各領域内に獲得します。

IMS は、外部サブシステムが ESMT を作成できるように、マクロを 2 つ用意しています。DFSEMODL マクロは、IMS がロードする ESAP モジュールを定義するためのものです。DFSEWAL マクロは、IMS が作成する作業域を定義するためのものです。モジュールを定義する一連の DFSEMODL ステートメントの後ろに、作業域を定義する DFSEWAL ステートメントを続け (任意)、END=LAST を指定した DFSEMODL ステートメントを最後に付ければ、表が作成されます。

### DFSEMODL マクロ

外部サブシステム・モジュール・テーブル (ESMT) は、一連の DFSEMODL ステートメント (モジュール定義ごとに 1 つ) から生成されます。

モジュール定義情報だけではなく、モジュールのロード時にモジュールのアドレスを入れる制御ブロックに関する情報も、DFSEMODL ステートメントで用意します。IMS は、モジュールをロードする前にこの制御ブロックを作成します。

IMS は、最大 3 組までモジュールをロードして、それらのモジュールを別の制御ブロック上でアンカーすることができます。したがって、ESMT は 1 個から 3 個までのサブテーブルで構成され、サブテーブルそれぞれに一組のモジュールとそのモジュール・アドレス制御ブロックの仕様が入ります。外部サブシステム出口ルーチンのモジュール・アドレス制御ブロックは EEVT (外部エントリー・ベクトル・テーブル) です。

モジュール・アドレス制御ブロックが作成されると、IMS はそのアドレスをソース制御ブロックに保管します。これが EEVTP です (EEVT の接頭部)。

DFSEMODL マクロのフォーマットは次のとおりです。

```
(label)  DSFEMODL  DSNAME=,SOURCE=,MODNAME=,DSLABEL=,  
          SUBPOOL=,OPTION=,END=
```

ここで、

#### (label)

これはオプションです。これを指定すると、マクロで ESMT サブテーブル・ラベルが生成されます。サブテーブルを生成する一連のマクロ・ステートメントのうちの最後のラベルが、サブテーブル・ラベルとして使用されます。

以下のパラメーターは制御ブロック情報を用意するためのもので、サブテーブルにつき 1 回だけ指定します (例えば、一連のステートメントのうちの最初の DFSEMODL ステートメントで指定)。複数のステートメントを指定すると、検出した最初の指定を使用してテーブルを生成します。

#### DSNAME=

(p1,p2,p3)

#### p1

モジュール・アドレス制御ブロックの名前。ESMT サブテーブルごとに (少なくとも 1 つの DFSEMODL ステートメントに) この名前を指定しなければなりません。

**p2**

モジュール・アドレス制御ブロックのサイズ。サイズは必ず指定しなければなりません。IMS は、ここで指定されているサイズのストレージを獲得して、モジュール・アドレス制御ブロックを作成します。

**p3**

モジュール・アドレス制御ブロック・ストレージ要求のサブプール番号。このパラメーターはオプションです。251 が指定されていないければ、IMS はサブプール 230 からストレージを獲得します。

#### **SOURCE=**

(p1,p2)

**p1**

ソース制御ブロックの名前。このパラメーターは必須です。DFSEEVTP を指定しなければなりません。(以下の説明をお読みください。)

**p2**

モジュール・アドレス制御ブロックのアドレスを保管する、ソース制御ブロック内の保管場所のラベル。このパラメーターは必須です。

以下のパラメーターでモジュール定義情報を与えます。

#### **MODNAME=**

IMS がロードするモジュールの名前。END パラメーターを指定しないすべての DFSEMODL ステートメントに、MODNAME を指定しなければなりません。(MODNAME を指定してもしなくても、END を指定できます。)

#### **DSLABEL=**

モジュールをロード後にモジュール・アドレスを入れる、モジュール・アドレス制御ブロック内の保管場所のラベル。(MODNAME を指定するときは) DSLABEL を必ず指定しなければなりません。

#### **SUBPOOL=**

##### **PDS データ・セットにあるリソースの場合:**

IMS がモジュールをロードするサブプール。MODNAME を指定するときは、SUBPOOL を必ず指定しなければなりません。制御領域の場合、IMS は指定されたサブプールにモジュールをロードします。従属領域の場合は、IMS は、SUBPOOL=251 が指定されていればサブプール 251 にモジュールをロードします。そうでない場合、モジュールはサブプール 230 にロードされます。指定できる値は、0、229、230、231、241、251、252 です。

##### **PDSE データ・セットにあるモジュールの場合:**

SUBPOOL パラメーターは使用されません。PDSE にあるモジュールは、次のいずれかのメソッドでロードされます。

- ・再入可能 (RENT) としてリンクされているモジュールは、サブプール 252、キー 0 にロードされます。これらのモジュールはフェッチ保護されません。
- ・再入不可 (NORENT) としてリンクされているモジュールは、サブプール 251、TCBKEY にロードされ、フェッチ保護されます。これらのモジュールにアクセスする前に、正しい保護キーが使用されることを確認する必要があります。

#### **OPTION=**

(p,p)

このパラメーターはオプションです。NOCTL と NODEP の 2 つのオプションがあります。(サブパラメーター・リスト内のオプションの位置に意味はありません。)

##### **NOCTL**

制御領域でロードしてはならないモジュール。

##### **NODEP**

従属領域でロードしてはならないモジュール。

END パラメーターは、ESMT の生成を制御します。

**END=**

**YES**

生成される ESMT 内のサブテーブルの終わりを示すときに必ず指定します。ESMT に複数のサブテーブルを入れるときにのみ END=YES を指定します。これで、最後 (または唯一) のサブテーブル以外の、それぞれのサブテーブルを終わらせます。END=YES の指定の後ろに、ESMT の次のサブテーブルの DFSEMODL ステートメント を続けます。

**LAST**

ESMT を生成する最後の DFSEMODL 定義ステートメントで指定しなければなりません。次の DFSEMODL または DFSEWAL ステートメントで (存在する場合)、新規の ESMT の生成が開始されません。

ユーザーは、バインダーの ENTRY ステートメントでエントリー・ポイントとして MAINEP を指定して、ESMT モジュールをプログラム・ライブラリー (SDFSRESL) にバインドしなければなりません。ESMT モジュールの終わりに、テーブル定義ヘッダーが生成されます。ENTRY ステートメントにより、IMS は後続の処理でヘッダーを正しく参照することができます。

DFSEMODL は、内部使用に限り実行形式 (MF=E) をサポートします。ESMT 生成ではこの形式を使えません。説明しているリスト形式 (MF=L) はデフォルトです。

ESMT の生成でソースに、モジュール・アドレスとソース制御ブロックをマップする DSECT を組み込んでおかないと、アセンブルに失敗します。

以下の制約事項が外部サブシステムに適用されます。

- ソース制御ブロックの指定

IMS で定義するすべてのサブテーブルのソース制御ブロック名 (SOURCE(p1)) として、DFSEEVTP を指定しなければなりません。DFSEEVTP を指定しない場合、DFSEMODL は DFSEEVTP 以外の指定でも受け入れますが、そのモジュールのロードは失敗して、失敗に終わった領域に対する接続は禁止されます。ユーザーは、EEVTP マッピング・レイアウトを使用しなければなりません。IMS ではこのレイアウトを使用するからです。

- サブシステム 出口ルーチン・モジュールの定義

サブシステム 出口ルーチン・モジュールの定義を入れるサブテーブルのモジュール・アドレス制御ブロック名 (DSNAME(p1)) には、DFSEEVTP を指定しなければなりません。

モジュール・アドレス制御ブロックのサイズ (DSNAME(p2)) は、「IMS V15 出口ルーチン」の「制御ブロックのマッピング」で示す EEVT マッピング (EEVTLGTH) で指示されたサイズに従って指定しなければなりません。

EEVPEEA を、ソース制御ブロック内のラベル (SOURCE(p2)) として指定し、モジュール・アドレス制御ブロック (EEVT) へのアンカーにする必要があります。IMS はこの値 (マクロも) を検査しませんが、指定されているラベルから生成されるオフセットを用いて、アドレスを保管します。このオフセットが間違っていると、IMS は出口ルーチンを活動化できません。

特定のサブシステム 出口ルーチン・モジュール用に指定したラベル (DSLABEL) は、IMS がその出口ルーチン・アドレスをモジュール・アドレス制御ブロック (例えば、EEVT 内) に保管するときに使用するオフセットを生成するためのものです。したがって、EEVT マッピングに基づいてこれらのラベルを指定する必要があります。

- 追加のサブテーブルの生成

ESMT には、出口ルーチン・モジュールの定義が入っているサブテーブルを、必ず 1 つは入れなければなりません。外部サブシステムは、自身の ESAP に入れておく必要がある他のモジュールを、別の制御ブロックでアンカーにすることができます。つまり、別のサブテーブルが生成されることになります。

DFSEMODL マクロで生成可能なサブテーブルの個数は制限されていませんが、サブテーブルを 4 個以上生成すると、処理で問題が起きます。IMS は、サブテーブルごとに、モジュール・アドレス制御ブロックを作成し、そのアドレスを EEVTP に保管します。これらの制御ブロックへのアンカーとして使用できるフィールドが EEVTP には 3 つしかなく、そのうちの 1 つは、EEVT へのアンカーです。



IMS は、EEVTLDIR と EEVPEWA のフィールドを使用しないので、これらのフィールドをこのために使用することができます。130 ページの『DFSEWAL マクロ』の外部サブシステム 作業域の定義に関する説明で、EEVPEWA を作業域アドレス制御ブロックへのアンカーにする方法を述べています。

- 外部サブシステム 固有のモジュールの定義

ESAP に IMS 以外の出口ルーチン・モジュール (例えば、IMS がその存在を何も認識していない、または IMS がサポートせずに外部サブシステムが活動化するモジュール) が必要な場合、外部サブシステムは、上述のように、追加のサブテーブルでこれらのモジュールを定義することができます。外部サブシステムは、これらのモジュールのモジュール・アドレス制御ブロックをマップする DSECT を用意しなければなりません。

**推奨事項:** 出口ルーチン・モジュールを含むサブテーブルで他のモジュールを定義することや、それらのモジュールのアドレスを含めるために EEVT のサイズを拡張することは行わないでください。この EEVT は、IMS 側がいつでもそれを拡張する権利を保有している制御ブロックです。このため、IMS 側が EEVT を拡張すると、外部サブシステム側は ESMT を再生成し、モジュールを再コンパイルする必要が生じる可能性があります。

## DFSEWAL マクロ

外部サブシステムのために IMS が作成する作業域は、ESMT 生成用に提供されている DFSEMODL ステートメントと一緒に DFSEWAL マクロ・ステートメントを組み込んで、定義しなければなりません。

DFSEWAL ステートメント (定義する作業域ごとに 1 つ) は、DFSEMODL ステートメントの後ろに続けますが、一連のステートメントの最後のステートメントは、END=LAST を指定した DFSEMODL ステートメントでなければなりません。DFSEWAL ステートメントは、生成される ESMT 内に、作業域定義テーブルを作成します。

ESMT 内に生成される各サブテーブルで、作業域を定義することができます。各サブテーブル内で、モジュールを少なくとも 1 つは定義する必要があります。作業域定義だけしか入っていないサブテーブルを生成すると、IMS が ESMT を処理するときにエラーになります。

IMS は、サブテーブルで定義されているモジュールをロード後に、サブテーブルで定義されている作業域を作成します。IMS は、作成した作業域のアドレスを、作業域リスト制御ブロックに保管します。この制御ブロックも、DFSEWAL マクロによって定義されるもので、サブテーブルのモジュール・アドレス制御ブロック内に入っているか、別の制御ブロックとして作成されます。このために、外部サブシステム 作業域リスト制御ブロックを指すために EWAL を使用します。

**推奨事項:** EWAL は、別の制御ブロックとして作成せずに、モジュール・アドレス制御ブロック内に入れてください。IMS が EWAL の作成時に、そのアドレスは (明示的に) 外部サブシステムに提供されません。その代わりに、その EWAL がモジュール・アドレス制御ブロック (IMS が EEVTP 内でアンカーとして使用する) の中に入っていると、外部サブシステムはその位置を (DFSEWAL を使って) 指定して、そのアクセス方法を認識します。(IMS が EWAL を作成するとき、IMS は内部使用のためにそのアドレスをストレージ内の ESMT に保管します。この ESMT のフォーマットは、接続インターフェースの説明の中に含まれていません。) 以下の図は、EWAL、EEVTP、および EEVT の間の関係を示しています。



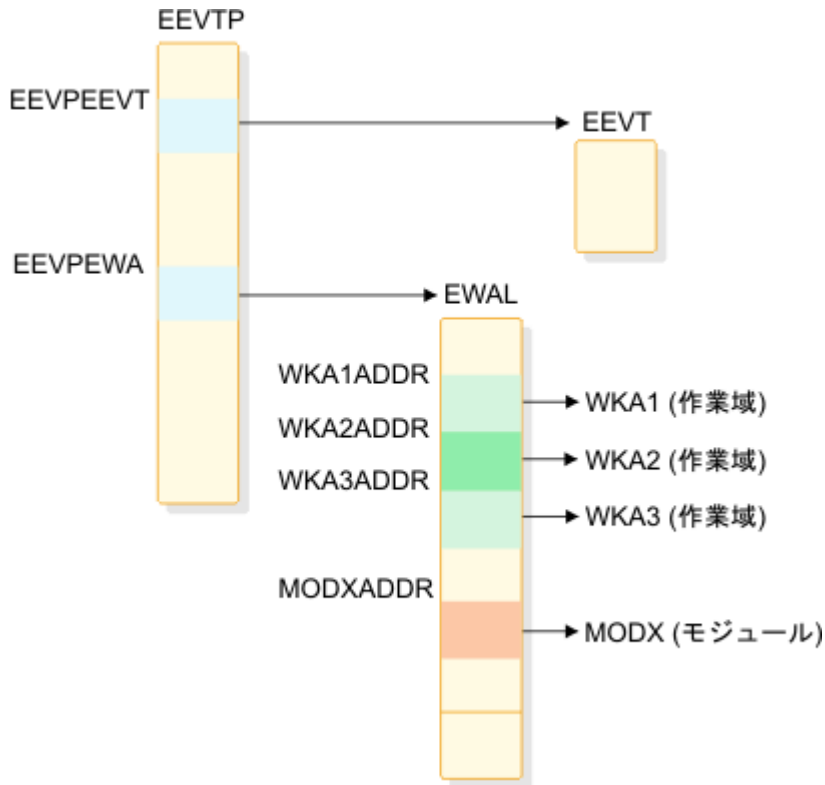


図 15. EWAL、EEVTP、および EEVT の関係

外部サブシステムが自身の ESAP 用の作業域を IMS に作成してもらうには、ESMT 内に 2 つ (できれば 3 つ) のサブテーブルを定義しなければなりません。モジュール定義が 1 つのサブテーブルに入ります。このサブテーブルのモジュール・アドレス制御ブロックは EEVT です。2 番目のサブテーブルに作業域の定義が入ります。このサブテーブルのモジュール・アドレス制御ブロックには、EWAL が入っているか、EWAL として使用され、EEVT と共に EEVTP 内にアンカーがあります。モジュールは 2 つのサブテーブル内で定義できます。1 つは出口ルーチン用のテーブルであり、もう 1 つは、出口ルーチンにより活動化される他の外部サブシステム・モジュール用のテーブルです。

以下のサンプル・コードの 2 番目に大きいブロックは、外部サブシステムで作業域定義を指定する方法を例示しています。

DFSEWAL マクロのフォーマットは次のとおりです。

```
DFSEWAL DSNAME, SOURCE=, WALSP=, NAME=, DSLABEL=,
        SUBPOOL=, LV=, OPTION=
```

各項目の説明は、以下のとおりです。

以下のパラメーターは制御ブロック情報を用意するためのもので、サブテーブルにつき 1 回だけ指定します (例えば、一連のステートメントのうちの最初の DFSEWAL ステートメントで指定)。複数のステートメントを指定すると、検出した最初の指定を使用してテーブルを生成します。

**DSNAME=**

(p1,p2)

**p1**

作業域リスト制御ブロック・マッピング DSECT の名前。DSECT 名は必ず指定しなければなりません。IMS が作業域リストを作成する場合は、獲得したストレージのジョブ・パック・エントリーにこの名前が付きます。

**p2**

作業域リスト・サイズ。サイズを指定すると、IMS はこのサイズのストレージを獲得して作業域リストを作成します。サイズを指定しなければ、作業域リストのソース制御ブロック (DFSEWAL SOURCE(p1) の指定) が、サブテーブルで定義されているモジュールのモジュール・アドレス制御ブ

ロック (DFSEMODL DSNAME(p1)) でない限り、IMS は作業域を作成しません。(以下の説明をお読みください。)

**SOURCE=**

(p1,p2)

**p1**

作業域リストがアンカーされる制御ブロックの DSECT 名。このパラメーターは必ず指定します。(以下の説明をお読みください。)

**p2**

ソース制御ブロック DSECT 内の、作業域リストが入っている位置のラベル。このパラメーターは必須です。IMS は、作業域リストのアドレスをこの制御ブロックに保管しません。(以下の説明をお読みください。)

**WALSP=**

作業域リスト・ストレージ要求のサブプール番号。このパラメーターはオプションです。WALSP=251 が指定されていない場合は、IMS はサブプール 230 からストレージを獲得します。

以下のパラメーターは作業域定義情報を用意するためのもので、各 DFSEWAL ステートメントで指定しなければなりません。

**NAME=**

獲得した作業域ストレージのジョブ・パック・ディレクトリー・エントリーに与えられた名前。このパラメーターは必須です。

**DSLABEL=**

IMS が作業域アドレスを保管する、作業域リスト制御ブロック DSECT 内の保管場所のラベル。DSLABEL は必ず指定しなければなりません。

**SUBPOOL=**

IMS が作業域用のストレージを獲得するサブプール。サブプールは必ず指定しなければなりません。IMS は、SUBPOOL=251 が指定されていればサブプール 251 を獲得し、指定されていない場合は、サブプール 230 の中に作業域が作成されます。このマクロで指定できる値は、0、229、230、231、241、251、または 252 です。

**LV=**

作業域のサイズ。サイズは必ず指定しなければなりません。

**OPTION=**

(p,p)

このパラメーターはオプションです。NOCTL と NODEP の 2 つのオプションがあります。(サブパラメーター・リスト内のオプションの位置に意味はありません。)

**NOCTL**

制御領域に作成してはならない作業域。

**NODEP**

従属領域に作成してはならない作業域。

参照するすべての制御ブロックをマップする DSECT を、ESMT 生成ソース内に組み込んでおく必要があります。そうしないと、アセンブルできません。

ソース制御ブロック DSECT 名とラベルは、必ず指定してください。ただし、IMS は EWAL アドレスをこの制御ブロックには保管しません。

EWAL をモジュール・アドレス制御ブロック内に組み入れるには、

- EWAL のサイズ (DFSEWAL DSNAME(p2)) を指定してはなりません。
- モジュール・アドレス制御ブロック名 (DFSEMODL DSNAME(p1)) は、EWAL ソース制御ブロック DSECT 名 (DFSEWAL SOURCE(p1)) として指定しなければなりません。
- ソース制御ブロック・ラベル (SOURCE(p2)) は、モジュール・アドレス制御ブロック内の作業域リストの位置を指定しなければなりません。

EWAL のサイズが指定されていて、モジュール・アドレス制御ブロックが EWAL ソースとして指定されていれば、IMS は検査をせずに EWAL のストレージを獲得します。EWAL のサイズが指定されておらず、モ

ジュール・アドレスと EWAL ソース制御ブロック DSECT 名が一致していない場合、IMS は作業域を作成しません。(IMS は、ソース制御ブロックのアドレスを知るすべがありません。IMS は、作業域が作成されていないことを示しません。)

IMS は、EEVTP 制御ブロック内の EEVPEWA フィールドを、EWAL のアドレスを入れるために確保しています。以下のコードは、このフィールドに作業域リストをアンカーするための定義を外部サブシステムで指定する方法を例示しています。実際に行われることは、モジュール・アドレス制御ブロックが作成され、EEVPEWA でアンカーされて、EWAL として使用されます。

```
DFSEMODL DSNAME=(DFSEEVTP,68,230),SOURCE=(DFSEEVTP,EEVPEEA),
          MODNAME=INITEXIT,DSLABEL=EEVTINIT,SUBPOOL=230
DFSEMODL MODNAME=IDEXIT,DSLABEL=EEVTID,SUBPOOL=230
DFSEMODL MODNAME=RIDEXIT,DSLABEL=EEVTRID,SUBPOOL=230,
          OPTION=NODEP
          .
          .
          .
          .
          .
DFSEMODL MODNAME=CMDEXIT,DSLABEL=EEVTCMD,SUBPOOL=230,
          OPTION=NODEP
DFSEMODL END=YES
DFSEMODL DSNAME=(ESSEWAL,40,230),SOURCE=(DFSEEVTP,EEVPEWA),
          MODNAME=MODX,LABEL=MODXADDR,SUBPOOL=230
DFSEWAL DSNAME=(ESSEWAL),SOURCE=(ESSEWAL,ESSEWAL),WALSP=230,
          NAME=WKA1,DSLABEL=WKA1ADDR,SUBPOOL=230,LV=200
DFSEWAL NAME=WKA2,DSLABEL=WKA2ADDR,SUBPOOL=230,LV=100
DFSEWAL NAME=WKA3,DSLABEL=WKA3ADDR,SUBPOOL=230,LV=100
DFSEMODL END=LAST
```

#### 例に関する注:

- EWAL をマップするために ESSEWAL という名前の DSECT が、外部サブシステムによって作成されているものとします。
- 2つのサブテーブルが完全に定義されています。
  - 最初のサブテーブルには、出口ルーチン・モジュールの定義が入っています。
  - 2番目のサブテーブルには、作業域の定義が入っています。
    - このサブテーブルのモジュール定義には、モジュール・アドレス制御ブロックのアンカー・フィールドとして EEVPEWA が指定されています。(外部サブシステムが実際にこのサブテーブルのモジュールをロードしたくない場合は、NOCTL と NODEP の両方のオプションを指定することができます。)
    - モジュール・アドレス制御ブロック DSECT の ESSEWAL は、EWAL ソース制御ブロック DSECT として指定されており、EWAL サイズは指定されていません。これは、EWAL をモジュール・アドレス制御ブロックに入れる必要があることを表します。
    - ESSEWAL は、EWAL のソース・ブロック内のラベルとしても指定されており、これは、モジュール・アドレス制御ブロックのオフセット 0 から EWAL が開始されることを表します。したがって、モジュール・アドレス制御ブロックそのものが EWAL であり、EEVTP 内の EEVPEWA でアンカーされます。



## 第 9 章 IMS 外部サブシステム 接続機能の処理

IMS 外部サブシステム 接続機能では、制御領域の初期設定の際に処理を実行します。

### 外部サブシステム接続パッケージのロード

IMS は、ご使用のシステム環境で指定したテーブル名を用いて外部サブシステム提供のテーブル (IMS.PROCLIB 内の外部サブシステム定義メンバーに入っている) を、制御領域の初期設定中にロードします。

IMS は、外部サブシステム・モジュール・テーブル (ESMT) をロード後に、そのテーブルで定義されている外部サブシステム・モジュールをロードします。リソース変換テーブル (RTT) もあればロードされます。この処理中にエラーが起これば、IMS はそのサブシステムを「停止」状態に置き、接続を確立しません。ただし、**/START SUBSYS** コマンドを受け取ると、IMS は定義 (PROCLIB) にアクセスし直して、この処理を再度実行します。

起こる可能性のあるエラーは、次のとおりです。

- PROCLIB メンバーを処理できない
- 外部サブシステム・ロード・ライブラリーをオープンできない
- ESMT をロードできない (指定した名前が間違っていて、ライブラリーに入っていない)

IMS は、ESMT と RTT のアドレスを EEVTP 制御ブロックのフィールド EEVPESMT と EEVPRTTA にそれぞれ保管します。

### EEVT 制御ブロックの作成

IMS は、DFSEMODL マクロ・ステートメントから生成される外部サブシステム・モジュール・テーブル (ESMT) に入っている情報に基づいて、EEVT 制御ブロックの作成と初期設定を行います。

EEVT のストレージ要求に必要なサイズとサブプールは、ESMT から得られます。EEVT は、ESMT でサブプール 251 が指定されていない限り、サブプール 230 に作成されます。EEVT のサイズとして、IMS EEVT マッピングで指示されているサイズと同じサイズが最小限指定されていることを、外部サブシステムは確認する必要があります。

IMS は、ESMT で指定されているオフセットに基づいて、EEVT のアドレスを EEVTP 制御ブロックに保管します。したがって、(DFSEMODL マクロを用いて) ESMT で生成されるオフセットが IMS EEVTP マッピングの EEVPEEA フィールドを指していることを、外部サブシステムが確認する必要があります。

IMS は、EEVTP 内の EEVT ポインター・フィールド (EEVPEEA) がこの処理で初期設定されているかどうかを検査しません。実際、ESMT のオフセットが原因で、IMS が、他の用途に指定された EEVTP 内のフィールドにアドレスを保管することがあり、これが問題が起これば原因になります。したがって、**外部サブシステムは、ESMT 内に正しいオフセットが確実に生成されるようにする必要があります。**

この処理では、外部サブシステムは、IMS がロードする別のモジュール・セットを指定することができます (IMS は、これらのモジュールを活動化しません)。両方のモジュール・アドレス・リスト (1 つは EEVT) は、EEVTP 内にアンカーがあります。

IMS は EEVTLDIR フィールドを使用しません。実際には、アドレス・リストのアンカーにするだけのフィールドが EEVTP 内にあれば、3 組以上のモジュール・セットを ESMT (サブテーブル) で定義し、IMS でロードすることができます。

#### 関連概念

127 ページの『外部サブシステム・モジュール・テーブルの作成』

外部サブシステムは、IMS がロードする外部サブシステム・モジュールの定義を入れておく外部サブシステム・モジュール・テーブル (ESMT) を作成します。

## 外部サブシステム・モジュールのロード

ESMT で定義されている外部モジュールを IMS がロードすると、そのモジュールのアドレスが EEVT に保管されます。

このモジュール定義では、これらの各モジュール・アドレスに対して EEVT 内の位置までのオフセットを提供します。IMS は、モジュール・ロード処理が必要な出口ルーチン・アドレスをセットしたかどうかを検査しません。外部サブシステムが選択すれば、IMS が ESAP に制御をいったん渡してしまうと ESAP は各出口ルーチン・アドレスを EEVT にセットできます。例えば、外部サブシステムは複数の出口ルーチンを 1 つのロード・モジュールの中に提供し、ESAP に個々の出口ルーチンのモジュール・アドレスをセットさせることができます。

ESMT 内のモジュール定義を用いる場合でも、ESAP 処理による場合でも、外部サブシステムは、IMS が出口ルーチンを活動化する必要があるときに、その出口ルーチン・アドレスが EEVT に存在するようにしておく必要があります。出口ルーチンのアドレスは、IMS EEVT マッピングに従って EEVT に入れなければなりません。

IMS により規定されている出口ルーチンには、制御領域内では活動化されないものもあり、また従属領域内では活動化されないものもあります。外部サブシステムは、モジュールを制御領域、あるいは従属領域にロードしないことを、モジュール定義で指示することができます。出口ルーチンのモジュール定義では、下記に従ってロード方法を指定しなければなりません。

- 制御領域と従属領域のどちらの中でも活動化される出口ルーチン
  - 識別出口ルーチン
  - 初期設定
  - 識別終了出口ルーチン
- 制御領域内でのみ活動化される出口ルーチン
  - コマンド
  - エコー
  - 未確定状態の解決
  - サブシステム終了
- 従属領域内でのみ活動化される出口ルーチン
  - 打ち切り続行出口ルーチン
  - スレッドの関連付け
  - コミット続行出口ルーチン
  - コミット準備出口ルーチン
  - コミット確認出口ルーチン
  - スレッド作成出口ルーチン
  - 通常呼び出し出口ルーチン
  - サインオフ出口ルーチン
  - サインオン出口ルーチン
  - サブシステム作動不能出口ルーチン
  - スレッド終了出口ルーチン

制御領域の場合、IMS は、モジュール定義で指定されているサブプールに外部サブシステム・モジュールをロードします。従属領域内のモジュールについてサブプール 251 が指定されている場合は、IMS はそのモジュールをサブプール 251 にロードします。その他の場合は、サブプール 230 にモジュールをロードします。

外部サブシステムは IMS と連絡するのに必要な出口ルーチンだけを使用します。ただし、一部の出口ルーチンは必須であり、残りはオプションです。必須出口ルーチンが存在していない場合は、IMS はその出口

ルーチン呼び出そうとした時点でエラー・メッセージを生成し、外部サブシステムとの接続を終了します。

外部サブシステムが実行するように設計されている機能が外部サブシステムに必要でない場合は、IMS の呼び出しに備えて存在するだけで何も実行しない出口ルーチンを作成します。(出口ルーチンには、SR 15,15 および BR 14 のロジックなどの共通コードを入れます。ESS は、出口ルーチンが呼び出された時点でこの共通コードにブランチし、この共通コードは固有の処理は何も行いません。) 初期設定出口ルーチンの処理中に、外部サブシステムは (必要ならば、制御領域と従属領域の両方の) DFSEV DSECT 内のアドレスを更新して、これらの出口ルーチンを指すようにすることができます。これで、出口ルーチンが存在していても、エラー・メッセージを発行せずに IMS が正常に機能し、しかも外部接続を終了することができます。

## ESAP の作業域の作成

IMS は、ESAP モジュールをロード後に、ESMT 内に作業域の定義があれば ESAP 用の作業域を獲得します。

IMS DFSEWAL マクロを使用して、ESMT 内に作業域定義を生成します。

IMS による作業域の作成手順は、IMS が次のことを行える以外は、ESAP モジュールをロードするときの手順に似ています。

- 作業域アドレス用の制御ブロックを作成する (ESAP モジュールのアドレス用に EEVT を作成したのと同じ)、または、
- モジュール・アドレスが入っている同じ制御ブロックに作業域アドレスを保管する。

EEVTP の EEVPEWA フィールドは、外部サブシステム作業域リスト (EWAL) として参照される制御ブロックのアドレスを入れるためのもので、このリストには ESAP 用の作業域のアドレスが入っています。ただし、IMS が EWAL のアドレスをこのフィールドに入れるためには、外部サブシステムが ESMT 内に適切な指定を提供しておく必要があります。

IMS は、定義に従って各作業域を制御領域か従属領域に、またはこの両方に作成します。ストレージは、サブプール 251 が指定されている場合はその中で獲得され、そうでない場合はサブプール 230 の中で獲得されます。

### 関連概念

127 ページの『外部サブシステム・モジュール・テーブルの作成』

外部サブシステムは、IMS がロードする外部サブシステム・モジュールの定義を入れておく外部サブシステム・モジュール・テーブル (ESMT) を作成します。

## 外部サブシステム接続の開始

外部サブシステムが制御領域を確認することを後に延ばすように選択していない限り、IMS は、制御領域の初期設定処理中 (例えば、IMS の開始時) に外部サブシステムに自動接続します。

外部サブシステムは、制御領域の初期設定出口ルーチンから戻りコード 4 (識別しない) を戻すか、制御領域用の初期設定出口ルーチンを用意しないことにより、接続を延期します。

外部サブシステムが IMS 提供の通知メッセージ・メカニズムを使用する場合は (しかも IMS が識別出口ルーチンを活動化するとき外部サブシステムはまだ立ち上がっていない場合)、外部サブシステムが開始される時に接続が自動的に確立されます。識別出口ルーチンから戻りコード 4 が戻されると、IMS は、出口ルーチンに渡した通知メッセージを外部サブシステムが送信するまで待ち、そのメッセージを受け取ったときに識別出口ルーチンを再度呼び出します。

この通知メカニズムを使用しないときは、識別出口ルーチンが戻りコード 12 を戻さなければなりません。この場合は、接続は停止状態に置かれます。/START SUBSYS コマンドを用いてこの停止状態を解除しないと、IMS は接続を確立しません。

いったん接続が確立されると、外部サブシステムに関する未解決のリカバリー・トークンを解決するために、IMS は未確定解決処理を実行します。未解決のリカバリー・トークンがあり、未確定解決出口ルーチンが用意されていない場合は、IMS は終了し、接続を中止します。その他の場合は、従属領域は外部サブシステムに接続することができます。



## 制御領域識別の延期

IMS 制御領域との接続をあとで確立した方がよい場合、外部サブシステムは IMS 制御領域を確認することを延期することができます。

IMS は、アプリケーション・プログラムと外部サブシステム間の接続の有無とは関係なく、アプリケーション・プログラムをスケジュールし、制御を渡します。したがって、外部サブシステムは、アプリケーション・プログラム呼び出し (外部サブシステム・リソースを要求する最初の呼び出し) に応じなければならない時点まで待ってから、IMS に接続することができます。もちろん、呼び出しを処理するには、制御領域接続と従属領域接続およびアプリケーション・スレッドが存在していなければなりません。

制御領域識別が延期されると、次の時点で識別が行われます。

- 外部サブシステムが、出口ルーチンで、IMS サブシステム始動サービスを活動化するとき。
- 外部サブシステムにアクセスできる MPP または IFP 従属領域が開始されたとき。
- 外部サブシステム名を指定した `/START SUBSYS` コマンドが処理されたとき。

## IMS サブシステム始動サービスの使用

制御領域の識別を延期したときは、外部サブシステムが制御領域の識別出口ルーチンを実行する必要がある時点で IMS サブシステム始動サービスを活動化することができます。

もっと具体的に言うと、外部サブシステムのサービスを要求するアプリケーションの最初の呼び出しを従属領域で処理する時点で接続がまだ存在していない場合、IMS は外部システムを自動識別しようとしません。外部サブシステムは、接続を確立するために始動サービスを活動化する必要があります (アプリケーション呼び出しを処理したい場合)。

従属領域接続を確立するときにも始動サービスを使用します。制御領域または従属領域が外部サブシステムに対して識別される前にアプリケーションからの外部サブシステム呼び出しが処理されると、従属領域はサブシステム作動不能出口ルーチンを活動化します。外部サブシステムは、この出口ルーチンからサブシステム始動サービスを呼び出して接続を確立することになっています。

活動化された始動サービスは、制御領域の識別がまだ行われていない場合は、制御領域および従属領域の接続を確立します。制御領域の識別がすでに行われていれば、従属領域接続だけを確立します。IMS が、外部サブシステムが通知メッセージ (前の識別出口ルーチン呼び出しで受け入れたもの) を送るのを待っている場合は、始動サービスはエラー戻りコードを戻し、接続の確立は行いません。サブシステム始動サービス出口ルーチンの詳細については、「IMS V15 出口ルーチン」を参照してください。

### 関連概念

138 ページの『[従属領域接続の確立](#)』

MPP 領域、IFP 領域、または BMP 領域に対して接続を確立できます。

## 従属領域接続の確立

MPP 領域、IFP 領域、または BMP 領域に対して接続を確立できます。

### MPP 領域と IFP 領域

MPP または IFP 従属領域に対する識別処理は、制御領域に対する識別処理に似ており、そこでは、従属領域は自動的に識別出口ルーチンを活動化して、従属領域初期設定時に接続を確立します。ただし、外部サブシステムが識別を延期している場合は例外です。

従属領域が自動的に識別処理を行う場合に制御領域接続がまだ確立されていない場合は、IMS が制御領域を識別しようとします。これが正しく完了すると、従属領域の識別が行われます。したがって、制御領域の識別が延期されても、従属領域が自動的に接続できるのであれば、従属領域の始動時に、制御領域の識別出口ルーチンが自動的に活動化されることがあります。

従属領域の識別が延期されていると、その領域でアプリケーション・プログラムから外部サブシステムへの最初の呼び出しが出されたときに、外部サブシステムへの接続が確立されます。この場合の接続処理は、BMP 従属領域の場合の接続処理と同じです。

## BMP 領域

BMP 従属領域からの接続は、アプリケーションから外部サブシステムへの最初の呼び出しがその領域で処理されるまで、確立されません。この接続は、自動的に確立されます。

BMP 領域の初期設定出口ルーチンから戻りコード 4 (識別の延期) が戻されても効果はありません。

## 通知メッセージ

IMS は、制御領域の識別出口ルーチン呼び出し時に通知メッセージのアドレスを渡します。

外部サブシステムがアクティブでない (まだ開始されていない) 場合、識別出口ルーチンは、通知メッセージを受け入れたことと、外部サブシステムがアクティブになったときに IMS に通知メッセージを送信することを IMS に (戻りコード 4 で) 通知します。外部サブシステムが開始されると、内部の z/OS **MODIFY** コマンド (SVC 34) を用いてそのメッセージを IMS に送信し、接続可能であることを IMS に警告します。IMS は、通知メッセージを受け取ると、接続を確立するために識別出口ルーチンを再度活動化します。

通知メッセージを外部サブシステムに渡す手段としては、例えば、z/OS システム内に常に存在する外部サブシステム・コード (既存コード) が使用されます。識別出口ルーチンは、外部サブシステムが開始されたときに外部サブシステムで使えるように、そのメッセージを既存コードのキューに入れます。

IMS は、通知メッセージを以下の図に示すフォーマットで識別出口ルーチンに渡します。



図 16. 通知メッセージ・フォーマット

ここで、

### LL

メッセージ長 (LL + ZZ + MESSAGE\_TEXT) が入っている 2 バイト・フィールド。

### ZZ

2 進ゼロが入っている 2 バイト・フィールド。

### MESSAGE\_TEXT

**MODIFY** コマンドで IMS が受け取る予定の通知メッセージ・テキスト。このメッセージ・テキストを変更してはなりません。

次の **MODIFY (F)** コマンドを発行します。

```
MODIFY ims_jobname,message_text
```

外部サブシステムは、メッセージを送信する前に、IMS が渡す通知メッセージ・テキストの先頭に、**MODIFY** *ims\_jobname* (または *ims\_jobname*) を付加する必要があります。以下の図に、SVC 34 コマンド入力フォーマットを示します。

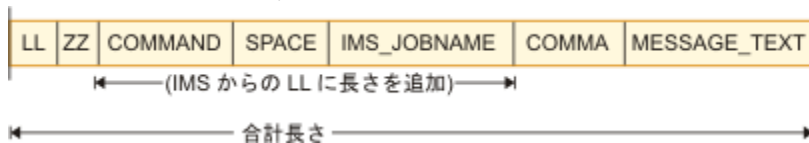


図 17. SVC 34 コマンド入力フォーマット

ここで、

### LL

コマンド入力域の全長 (識別出口ルーチンに渡された LL フィールド内の長さ、COMMAND + SPACE + IMS\_JOBNAME + COMMA の各フィールド長を追加したもの) が入っている 2 バイト・フィールド。

### ZZ

2 進ゼロが入っている 2 バイト・フィールド。

### COMMAND

**MODIFY** コマンド verb (C'MODIFY' または C'F') が入っています。

## SPACE

空白 (C') が 1 個入っている 1 バイト・フィールド。

## IMS\_JOBNAME

IMS 制御領域のジョブ名が入っている 8 バイト・フィールド。名前は左寄せされ、右側は空白で埋め込まれます。識別出口ルーチンは、現行 TCB が指している TIOT からこのジョブ名を入手します。

## COMMA

コンマ (C',) が入っている 1 バイト・フィールド。

## MESSAGE\_TEXT

識別出口ルーチンに渡された通知メッセージ・テキスト。

## アプリケーション・プログラム要求のサポート

アプリケーションからの呼び出しは、アプリケーション・プログラムにバインドされた言語インターフェース・モジュールから IMS に渡されます。言語インターフェースは、該当する IMS プログラム要求ハンドラーにブランチして、アプリケーション・プログラム呼び出しパラメーター・リストを渡します。

外部サブシステム宛での呼び出しの場合、言語インターフェースは外部サブシステム・パラメーター・リストも作成して渡す必要があります。このパラメーター・リストの目的は、呼び出しの宛先になる外部サブシステムの LIT (言語インターフェース・トークン) を渡すためです。IMS は、アプリケーションからの呼び出しを、その呼び出しで渡された LIT 値と一致する LIT 値を持つ外部サブシステム宛てに経路指定します。

IMS は、呼び出しパラメーター・リストと外部サブシステム・パラメーター・リストの両方を、目的の外部サブシステムの通常呼び出し出口ルーチンに渡します。外部サブシステム・パラメーター・リストの第 1 ワードには、LIT 値が文字形式で入っている 4 バイト・フィールド (左寄せされ、右側は空白で埋め込まれる) のアドレスが入っています。IMS が規定しているのは、外部サブシステム・パラメーター・リストの第 1 ワード (LIT のアドレス) だけです。このパラメーター・リストを拡張して、外部サブシステム固有の情報を通常呼び出し出口ルーチンに渡すことができます。

## 言語インターフェースの定義

IMS が提供する言語インターフェース・モジュール DFSLI000 は、SYS1 の値をサポートします。インストール・システムはこのモジュールを使用できます。あるいは、SYS1 以外の LIT 値を使用したい場合は、独自の言語インターフェースを定義することができます。

IMS システムが複数の外部サブシステムにアクセスする場合は、各サブシステムの LIT 値は固有でなければなりません。インストール・システム独自の言語インターフェース・モジュールを定義しなければなりません。

IMS には、インストール・システムで言語インターフェース・モジュールを生成するときに役立つ DFSLI マクロがあります。DFSLI マクロを展開すると、言語インターフェース機能を実行するのに必要なコードが生成されます。マクロ・ステートメントをコンパイルしてモジュールを生成するときは、IMS マクロ・ライブラリーを用意しておかなければなりません。

### 関連概念

[言語インターフェース・モジュールの定義 \(システム定義\)](#)

## 外部サブシステム固有の言語インターフェース・エン트리・ポイント

IMS 言語インターフェース・モジュールにはエン트리・ポイントが 3 つあり、外部サブシステム宛でのアプリケーション呼び出しではこのいずれか 1 つを使用できます。

これらのエン트리・ポイントのうちの 2 つは、DFSLI マクロで指定される暗黙の LIT 値と関連付けられています。(DFSLI マクロで生成される言語インターフェース・モジュールには、指定の LIT 値がハードコーディングの定数として入ります。) 第 3 のエン트리・ポイントは、暗黙の LIT 値と関連付けられるものではありません。このエン트리・ポイントを使用すると、アプリケーション・プログラムは、アプリケーション呼び出しを発行するときに LIT 値を指定することができます。どのエン트리・ポイントの場合も、IMS が外部サブシステムに渡すパラメーター・リストのアドレスがレジスター 1 に入っています。外部サブシステムへの言語インターフェース・エン트리・ポイントは、次のとおりです。

## DSNHLI

暗黙の LIT 値と関連付けられているエントリー・ポイント。

アプリケーション・プログラムは、自身が使用する外部リソースをどのサブシステムがアクセスするのかわかる必要はありません。(外部サブシステムが Db2 for z/OS の場合、このエントリー・ポイントは SQL 呼び出しに使用されます。)

## DSNWLI

暗黙の LIT 値と関連付けられているエントリー・ポイント。

アプリケーション・プログラムは、自身が使用する外部リソースをどのサブシステムがアクセスするのかわかる必要はありません。(外部サブシステムが Db2 for z/OS の場合、このエントリー・ポイントは計測機能呼び出しに使用されます。)

## DFSESS

アプリケーション・プログラムが LIT 値を指定できるエントリー・ポイント。

アプリケーション・プログラムは、自身が使用する外部リソースをどのサブシステムがアクセスするのかわかっていなければなりません。アプリケーション・プログラムは、LIT 値のアドレスをアプリケーション呼び出しリストの第 1 パラメーターで指定しなければなりません。アプリケーション・プログラムから外部サブシステムに制御を渡す前に、IMS はアプリケーション呼び出しリストのアドレスを 4 増やし、LIT 値パラメーターをスキップさせます。

**制約事項:** DFSESS エントリー・ポイントを Db2 for z/OS サブシステムとの通信には使用しないでください。

## 複数の外部サブシステムへのアクセス

アプリケーション・プログラムは、同一の実行で DL/I と外部サブシステムにアクセスすることができます。アプリケーション・プログラムが同一の実行で複数の外部サブシステムにアクセスできるかどうかは、言語インターフェースによって制約されます。

ある外部サブシステム (プロダクト) によってアプリケーション・プログラムに提供されたデータ (呼び出し) インターフェースが、(例えば DL/I 呼び出しが SQL 呼び出しと異なるように) 別の外部サブシステム提供のインターフェースと異なっている場合、言語インターフェースのパスが異なるため、アプリケーションは両方のサブシステムにアクセスすることができます。2つの外部サブシステムのタイプが同じ場合 (同一の外部サブシステムを 2つ使用) や、2つの外部サブシステム・プロダクトが同じ呼び出しインターフェース (SQL など) を使用する場合には、データ・インターフェースが同じである場合は、アプリケーションがデータ位置と関連付けられていない限り、アプリケーションの同一の実行中に両方のサブシステムにアクセスすることはできません。(従属関係は、それぞれの呼び出しインターフェースのケースに組み込みです。)

## リソース・リカバリー・トークン

アプリケーションとスレッド接続しているすべてのサブシステムにまたがる作業単位を個々に識別するために、16 バイトのリカバリー・トークンが使用されます。IMS は、スレッドの作成前にサインオン出口ルーチンにこのトークンを渡します。

コミット処理と未確定解決処理の場合、要求した処置が実施される作業単位を識別するために、IMS はリカバリー・トークンを渡します。

リカバリー・トークンの構成は、次の図に示すとおりです。

IMS_id	OASN	commit_number
8	4	4

図 18. リカバリー・トークンのフォーマット

ここで、

### IMS-id

IMS システム ID (1 文字から 4 文字) で、左寄せされ、右側は空白で埋め込まれた、8 バイトです。



## OASN

アプリケーションがスケジュールされる時に割り当てられる 4 バイトの 2 進数の起点アプリケーション・シーケンス番号。OASN は、最後のコールド・スタート以後に IMS システム内でスケジュール順に割り当てられます。これは、アプリケーション・スケジュール番号とも言われます。

## commit\_number

4 バイトの 2 進数のコミット番号。アプリケーションがスケジュールされる時にコミット番号は 2 進ゼロで初期設定され、そのアプリケーションでコミット処理が行われるたびに増やされます。

外部サブシステムは、サインオン時に渡されるリカバリー・トークンを検査して、確かにそのものであることを確認しなければなりません。IMS のコールド・スタートでは、外部サブシステムで未確定であるリカバリー・トークンと同じリカバリー・トークンが生成されることがあります。サインオン出口ルーチンが、渡されたリカバリー・トークンが重複していることを IMS に通知した場合、IMS はアプリケーション・プログラムを異常終了させます。コミット準備出口ルーチンも、トークンの重複を通知しますが、この場合は、コミットが処理されるまでそのリカバリー・トークンを作業単位に関連付けないことを外部サブシステムが選択することができます。

インストール・システムは、OASN キーワードを指定した **/DISPLAY SUBSYS** コマンドを用いて、IMS で未確定状況にある作業単位を判別します。インストール・システムは、(必要に応じて) **/CHANGE** コマンドを用いて、IMS で未確定状況にある作業単位を手作業で削除することができます。**/CHANGE** コマンドは、IMS 内の作業単位に影響を与えるだけです。外部サブシステムとは通信しません。これらのコマンドは OASN だけを使用し、完全リカバリー・トークンは使用しません。**/DISPLAY** はリカバリー・トークンの OASN 部分だけを (10 進形式で) リストし、**/CHANGE** は OASN だけを (これも 10 進形式で) 受け入れます。(IMS 内では、OASN は、既知のすべての作業単位で固有です。)

## 外部サブシステム接続の終了

IMS は、多数のさまざまな事情のときに、外部サブシステム接続 (制御領域接続) を規則正しい方法で終了させます。

IMS は、下記の 1 つが起きたときに、外部サブシステム接続 (制御領域接続) を規則正しく終了させます。

- IMS が **/STOP SUBSYS** コマンドを処理したとき。
- 外部サブシステム (ESAP) が IMS サブシステム終了サービス出口ルーチンを活動化したとき。
- 外部サブシステムが、識別出口ルーチンの呼び出しで提供された終了 ECB を通知したとき。
- 特定の接続処理エラーが検出されたとき。
- IMS がシャットダウンしているとき。

IMS は、外部サブシステムへの既存のスレッドで処理が完了できるようにします。すべてのスレッドが終了してしまおうと、IMS は、制御領域から識別終了出口ルーチンを活動化して、接続を終了します。

**/STOP SUBSYS** コマンド、終了 ECB の記入、または処理エラーが原因で接続を終了する場合、IMS は外部サブシステム接続を停止状態に置きます。いったん停止状態になると、IMS は接続の再確立を許可しません。この停止状態を解除するには、**/START SUBSYS** コマンドが必要です。

## 外部サブシステムからの終了要求

外部サブシステムは、終了 ECB に通知するか、または ESAP からサブシステム終了サービス出口ルーチンを活動化することにより、接続を終了させることができます。このサービスを使用するときは、接続が停止状態にされません。

外部サブシステム・コマンド出口機能と一緒に、終了サービスを使用することができます。例えば、IMS **/SSR** コマンドで提供された外部サブシステム終了コマンドによりコマンド出口ルーチンが活動化された場合、その出口ルーチンは、サブシステム終了サービス出口ルーチンを活動化して、外部サブシステムへの接続を終了させることができます。

制御領域で最初に識別処理が実行される時に、IMS は終了通知 ECB のアドレスを外部サブシステムに渡します。サブシステムは、接続を終了する必要がある場合、ECB を通知します。ECB は CSA 内にあります。通知コードに従って、IMS は接続を次のように終了します。

すべてのアクティブなスレッドを非活動化し、新規スレッドの開始を禁止してから、接続を終了します。終了機能が完了した時点で、この接続は停止状態に設定されます。

サポートされる通知コードは、以下のとおりです。

**X'40000000'**

IMS用に予約済み。

**X'40000008'**

外部サブシステムは正常に終了しようとしている。

**X'4000xxxx'**

他のすべての通知コードは、外部サブシステムの即時シャットダウンまたは破局的なシャットダウンと解釈される。

**関連資料**

[サブシステム終了サービス出口ルーチン \(出口ルーチン\)](#)

## 従属領域接続

従属領域の終了時には、制御領域接続が終了する場合以外は、サインオフ出口ルーチンおよび識別終了出口ルーチンは活動化されません。

すなわち、従属領域のサインオフ出口ルーチンおよび識別終了出口ルーチンは、IMSシステムが外部サブシステムに接続せずに実行を継続する場合にのみ活動化されます。例えば、識別処理で用意された終了 ECB を外部サブシステムがポストしたか、ESAP が IMS サブシステム終了サービスを活動化したか、または IMS が **/STOP SUBSYS** コマンドを処理したときです。それ以外の場合、IMS は、従属領域終了を外部サブシステムに連絡しません。

IMS は、外部サブシステムが z/OS タスク終了 (EOT) 出口ルーチンを用いて従属 TCB をモニターするものと想定しています。EOT 出口ルーチンに領域終了が通知されたときには、サブシステムにとって必要なサインオフ処理と識別終了処理をサブシステムが行わなければなりません。

外部サブシステムも、スレッドが作成されている従属領域の EOT 出口ルーチンをモニターしなければなりません。アプリケーション・プログラムが異常終了した場合は、スレッド終了出口ルーチンは呼び出されません。

制御領域接続が切れるときは、領域終了時に、またはシステム接続の切断要求を受け取ったときにスレッドがアクティブであった場合はスレッド終了後に、従属領域のサインオフ処理に続いて識別終了処理が実行されます。領域でサインオンが複数回実行されていたとしても、サインオフ出口ルーチンは 1 回しか呼び出されません。IMS は、自身のサインオフ処理と識別終了処理を続行し、エラーが検出されてもこれらの出口ルーチンを再度活動化することはありません。

## 停止状態とは

インストール・システムは、どのような条件の場合に IMS が外部サブシステム接続を停止するのかを知っていなければなりません。

外部サブシステム接続が停止されると、接続を再確立するためには **/START SUBSYS** コマンドを使用しなければなりません。停止状態は再始動してもその状態が持ち越されます。停止状態を引き起こす条件を以下に挙げます。

- 識別処理で用意されるサブシステム終了 ECB に、外部サブシステムがポストしたとき。通知コードのタイプ (例えば、正常シャットダウンであるか突然のシャットダウンであるか) と関係なく、終了処理の完了時に接続は停止されます。
- 制御領域で、ESAP を活動化した IMS タスク (TCB) が異常終了すると、外部サブシステム接続には停止状態のマークが付きます。

制御領域が異常終了した場合は、停止状態に設定されることはありません。IMS が正常に再始動されると、接続は異常終了前の状態になっています。

- 明白なのは、**/STOP** コマンドの処理後です。**/STOP** コマンドの処理中に IMS が異常終了したときも、停止状態にされます。

- IMS の再始動時に、もはや IMS に対して定義されていない外部サブシステムに未解決のリカバリー・トークンが存在していると (例えば、インストール・システムが IMS.PROCLIB の外部サブシステム定義メンバーからその外部サブシステムの定義を削除した場合など)、その外部サブシステムは停止状態に設定されます。

---

## 第 5 部 IMS Connect および TCP/IP 通信

IMS の IMS Connect 機能は、TCP/IP 対応の環境から IMS DB および IMS TM の両方へのアクセスを提供します。

### このタスクについて





## 第 10 章 IMS Connect の概要

IMS Connect は、1 つ以上の IMS Connect クライアントと 1 つ以上の IMS システムの間で、高性能の TCP/IP 通信を可能にします。IMS Connect は、IMS DB システムと IMS TM システムの両方をサポートします。

IMS Connect では、以下のことができます。

- ISC ユーザーが、TCP/IP 接続を使用して IBM CICS Transaction Server for z/OS にリンクする。
- MSC および OTMA のユーザーが、IMS 間 TCP/IP 接続を使用して、1 つの IMS システムから別の IMS システムにメッセージを送信。
- 分散クライアントが、TCP/IP 接続と IMS 共通サービス層 (CSL) の Open Database Manager (ODBM) コンポーネントを使用して、IMS DB とメッセージを交換。
- 分散クライアントが、TCP/IP 接続と OTMA を使用して IMS TM とメッセージを交換。
- IBM Management Console for IMS and Db2 for z/OS を使用する IMS オペレーターが、TCP/IP および IMS Operations Manager (OM) を使用することにより、IMSplex にコマンドを発行し、コマンド応答を受け取る。

IMS Connect は、以下の機能を提供します。

- 通信環境を管理するためのコマンド。
- ワークロード・バランシングの支援。
- クライアント・アプリケーションの設計およびコーディングの労力の削減。
- IMSplex に対するサポート。これにより、次のような他の IMSplex メンバーとの通信が可能になります。
  - IMS (システム間連絡機能 (ISC) および複数システム結合機能 (MSC) のサポート用)
  - IMS (複数システム結合機能 (MSC) サポート用)
  - ODBM
  - OM
- ISC TCP/IP リンクでの IMS と CICS の間の接続。
- IMS Connect インスタンス間の接続。これは、MSC および OTMA などの IMS 通信コンポーネントをサポートします。
- セキュリティおよびトランザクション保全性の拡張機能で、IMS アプリケーション・プログラムおよび操作へのオンデマンドの TCP/IP アクセス。
- IMS Universal ドライバー および ODBM などのクライアントを介した、分散アプリケーションの IMS データベースへのオンデマンドの TCP/IP アクセス。
- IMS Enterprise Suite SOAP Gateway、IMS Web 2.0 Solution for IBM Mashup Center などの特定の IMS Connect クライアントに対する XML 変換サポート。IMS Connect XML 変換サポートでは、選択されたプログラミング言語で記述された IMS アプリケーション・プログラムが想定するデータ構造に入力メッセージを変換します。したがって、XML を処理するように IMS アプリケーション・プログラムを作成または変更する必要がなくなります。

IMS Connect は ODBM を介して IMS DB に接続し、IMS DB によって管理されるデータベースに直接アクセスします。IMS Connect は、IMS TM に接続し、Open Transaction Manager Access (OTMA) を介してトランザクション処理をサポートします。

IMS Connect と他の IMSplex メンバー (IMS (MSC サポート用)、ODBM、および OM など) の間の通信には、IMS 構造化呼び出しインターフェース (SCI) を使用する必要があります。

IMS Connect は、そのクライアントと IMS および IMSplex リソースとの間でルーター機能を実行します。TCP/IP 接続を通して分散クライアントから受け取った要求メッセージは、z/OS システム間カップリング・ファシリティー (XCF) セッションを介して、データ・ストアと呼ばれる IMS システムに渡されます。IMS Connect は、データ・ストアから応答メッセージを受け取り、それを、発信元の TCP/IP クライアントに返します。

データ・ストアが停止すると、XCF を介して OTMA から IMS Connect にデータ・ストアの状況が送られます。データ・ストアが終了しているときに IMS Connect がデータ・ストアに接続された場合、データ・ストアが再始動されると、IMS Connect に通知され、データ・ストアに自動的に再接続されます。データ・ストアに手動で再接続する必要はありません。

一般的に、データ・ストアという用語は IMS システムを示します。ただし、より正確に言うと、データ・ストアという用語は、IMS システムへの OTMA ターゲット・メンバー (T メンバー) 接続を表します。例えば、IMS Connect のインスタンスは、同じ IMS システムに対して複数のデータ・ストア定義を持つことが可能です。この場合、各データ・ストアは同一 IMS システムへの異なる OTMA T メンバー接続を表します。

IMS Connect は、IMS 従属領域で実行される IMS アプリケーション・プログラムからのコールアウト要求もサポートします。IMS アプリケーション・プログラムは、コールアウト要求を発行して、IMS インストールの外部にあるプロバイダーにデータまたはサービスを要求します。コールアウト要求時、IMS はクライアントとして機能し、外部プロバイダーはサーバーとして機能します。

## 関連概念

[IMS Connect の定義と調整 \(システム定義\)](#)

## 関連タスク

[IMS 間の TCP/IP 接続 \(システム定義\)](#)

[209 ページの『IMS Connect のコールアウト要求サポート』](#)

IMS Connect は、IMS アプリケーション・プログラムが OTMA を介して、IMS インストール済み環境の外部にあるデータおよびサービス・プロバイダーにコールアウト要求を発行する場合に必要なコンポーネントです。両方のタイプのコールアウト要求で、IMS Connect は TCP/IP を使用する IMS Connect クライアントと IMS の OTMA コンポーネントとの間の TCP/IP ゲートウェイとして機能します。

## 関連資料

[IMS PROCLIB データ・セットの HWSCFGxx メンバー \(システム定義\)](#)

[IMS Connect コマンド \(コマンド\)](#)

[IMS Connect 出口ルーチン \(出口ルーチン\)](#)

[HWS メッセージ \(IMS Connect\) \(メッセージおよびコード\)](#)

[IMS Connect 戻りコードおよび理由コード \(メッセージおよびコード\)](#)

# IMS Connect クライアント・サポート

IMS 用の TCP/IP サーバーおよびメッセージ・ルーターとして、IMS Connect は、IMS TM、IMS DB、および CSL Operations Manager (OM) へのアクセスを提供します。IMS Connect によって提供されるクライアント・サポートは、IMS Connect クライアントに必要なアクセス・タイプによって異なります。

IMS Connect は、ソケット呼び出しで通信する TCP/IP クライアント、および別の入力データ・ストリーム・フォーマットで通信する TCP/IP クライアントもサポートします。

## IMS DB クライアント・サポート

IMS DB へのアクセスに関して、IMS Connect は CSL Open Database Manager (ODBM) を使用して、以下のタイプのクライアントをサポートします。

- Java EE プラットフォーム用の IMS Universal Database リソース・アダプターを使用するアプリケーション・プログラム
- IMS Universal JDBC ドライバーを使用するアプリケーション・プログラム
- IMS Universal DL/I ドライバーを使用するアプリケーション・プログラム
- オープン・スタンダード DRDA 通信アーキテクチャーを使用するユーザー作成クライアント・アプリケーション・プログラム

以下の表は、IMS DB クライアント通信をサポートする IMS Connect システム構成を示しています。

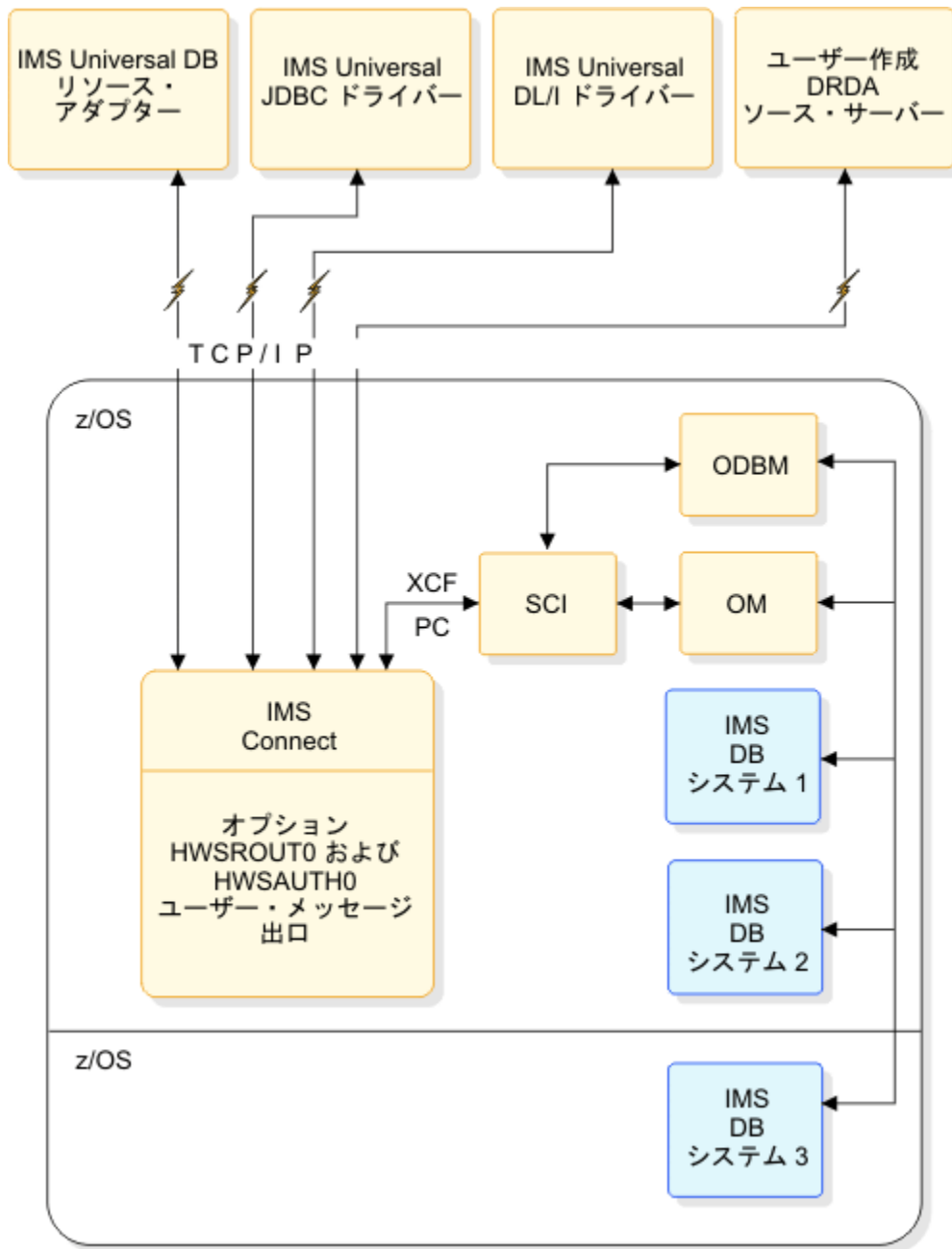


図 19. IMS DB システムのための IMS Connect サポートの概要

## IMS TM クライアント・サポート

IMS TM へのアクセスに関して、IMS Connect は以下のタイプのクライアントをサポートします。

- IMS 要求メッセージ (IRM) ヘッダーを使用して、通信プロトコル・オプションの IMS Connect への通信を行うユーザー作成 TCP/IP クライアント・アプリケーション・プログラム
- IMS TM Resource Adapter (IMS Connector for Java)
- IMS Enterprise Suite SOAP Gateway。XML メッセージ変換サポートが含まれます
- IMS Web 2.0 Solution for IBM Mashup Center。XML メッセージ変換サポートが含まれます

以下の図は、IMS TM クライアント通信をサポートする IMS Connect システム構成を示しています。

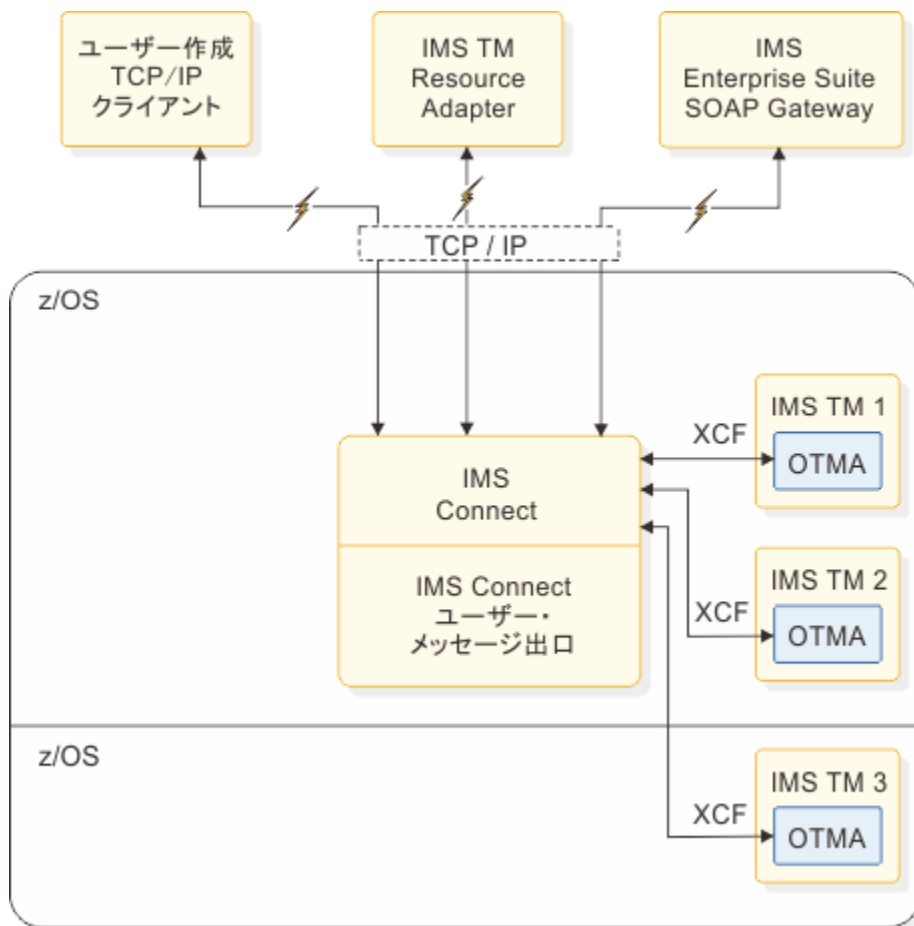


図 20. IMS TM システムのための IMS Connect サポートの概要

IMS TM IMS Connect クライアントの場合、IMS Connect が IMS にメッセージを送信する前に、ユーザーは、クライアント入力メッセージのフォーマットを OTMA メッセージ・フォーマットに変換するために、IMS Connect アドレス・スペースで実行するユーザー・メッセージ出口を作成できます。このユーザー作成メッセージ出口は、また、メッセージを IMS Connect に戻す前に、OTMA メッセージ・フォーマットからカスタマー・メッセージ・フォーマットに変換します。その後、IMS Connect は出力をクライアントに送ります。

IMS Connect は、分散プラットフォーム用 WebSphere Application Server のいずれかで実行される IMS TM Resource Adapter を介して、IMS TM と分散 Java アプリケーション間の TCP/IP 通信をサポートします。

IMS TM の場合、IMS Connect は SOAP Gateway もサポートします。これは Web サービス・ソリューションであり、これにより、IMS アプリケーションから SOAP を介して IMS 環境の外部と相互運用を行い、プラットフォーム、環境、アプリケーション言語、またはプログラム・モデルに関係なく、サービスを提供および要求できるようになります。

IMS Connect は、IMS Web 2.0 Solution for IBM Mashup Center をサポートします。これにより、既存の IMS 資産を Web 2.0 Mashup およびアプリケーション・ソリューションに統合できるようになり、ユーザーは、RSS、ATOM、または XML フィードを介して、IMS トランザクションにアクセスできるようになります。

SOAP Gateway および IMS Web 2.0 Solution クライアントでは、IMS Connect は XML 変換サポートも提供します。これにより、着信 XML メッセージを IMS アプリケーション・プログラムがサポートする一部の共通プログラミング言語のデータ構造に変換できます。

## IMSplex 操作サポート

CSL OM に対して IMS タイプ 2 コマンドを発行し、TCP/IP 接続を介してコマンド応答を受け取るために、IMS Connect は、クライアント (IBM Management Console for IMS and Db2 for z/OS など) をサポートしま

す。TCP/IP クライアントと、IMSplex 内の任意の IMS との間の通信は、単一の IMS Connect でサポートできます。

以下の図は、IMS TM クライアント通信をサポートする IMS Connect システム構成を示しています。

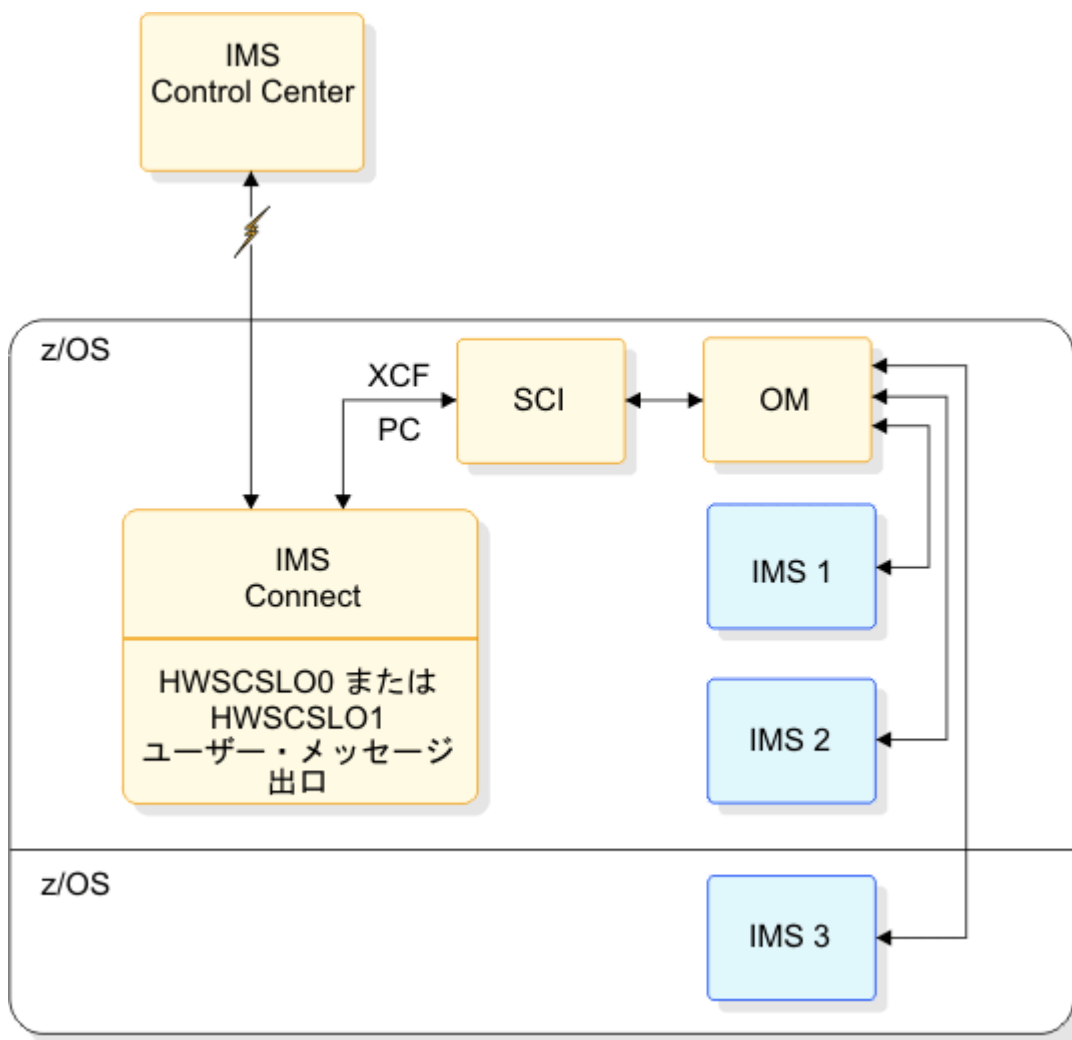


図 21. TCP/IP クライアントのための IMS Connect サポートの概要

## IMS Connect による IMS DB へのアクセスのサポート

IMS Universal ドライバー など、DBCTL および DB/TM 環境で IMS DB によって管理されるデータベースにアクセスする IMS Connect クライアントに関して、IMS Connect は、IMSplex 内の CSL Open Database Manager (ODBM) のインスタンスおよび IMS DB システム間で TCP/IP 接続を管理し、着信アクセス要求を経路指定します。

IMS Connect は、IMS Universal ドライバー の TCP/IP サーバーおよびフロントエンド IMSplex メッセージ・ルーターであり、以下が含まれます。

- Java EE プラットフォーム用 IMS Universal Database リソース・アダプター
- IMS Universal JDBC ドライバー
- IMS Universal DL/I ドライバー

IMS Universal ドライバー は、分散アプリケーション・プログラムまたはローカル z/OS アプリケーション・プログラムに対して、TCP/IP 接続を通して IMS データベースに対する直接的な非トランザクション型のアクセスを提供します。

IMS Universal ドライバー によって、TCP/IP 接続を通して IMS DB にアクセスする IMS Connect クライアント・アプリケーション・プログラムの開発が簡潔になります。IMS Universal ドライバー は、IMS Connect

IMS 要求メッセージ (IRM) 通信プロトコルを使用しません。また、IMS Universal ドライバー は、IMS Universal ドライバー と IMS Connect の間の通信に内部的に使用される、基盤となる分散リレーショナル・データベース体系 (DRDA) から、アプリケーション開発者を守ります。

IMS Connect は DRDA プロトコルをサポートし、また ODBM を使用すると完全な DRDA ターゲット・サーバーとなるため、DRDA プロトコルに対して直接アプリケーション・プログラムを作成することができます。ただし、分散環境から TCP/IP を通して IMS データベースにアクセスするための推奨 API は、Java EE プラットフォーム用の IMS Universal Database リソース・アダプターです。

IMS Connect による IMS Universal ドライバー のサポートには、グローバル 2 フェーズ・コミット・トランザクションが含まれます。

IMS Connect は、専用 DRDA ポート上でのみ、また共用可能な持続ソケットを通してのみ IMS Universal ドライバー との通信をサポートします。

IMS Connect セキュリティー・サポートには、IMS Connect DB セキュリティー・ユーザー出口ルーチン (HWSAUTHO) が含まれます。これは、IMS DB にアクセスする接続でのユーザー ID の認証を、より完全に制御するために使用できます。RACF もサポートされます。

IMS Connect による IMS Universal ドライバー のサポートは、IMS Connect 構成 PROCLIB メンバー内の ODACCESS 構成ステートメントによって定義されており、また IMS Connect と同じ IMSplex 上で少なくとも ODBM のインスタンスが 1 つ稼働している必要があります。

## IMS DB に接続する IMS Connect クライアントの接続経路指定

IMS Connect は、別名、およびその別名が属する ODBM のインスタンスを基に、IMS DB に接続するクライアント・アプリケーションから着信接続を経路指定します。クライアントのアプリケーション・プログラムは、その着信要求メッセージに別名を指定できます。

IMS Connect は、ODBM の各インスタンスを使用した登録中にデータが追加された内部トラッキング・テーブルで、ODBM のどのインスタンスにどの別名が属するののかについての追跡を行います。ODBM インスタンスと別名に関する情報も、HWSXIBOD マクロによってマップされた出口インターフェース・ブロック内の ODBM リストに保管されます。ODBM リストのアドレスは、HWSXIB マクロによってマップされたテーブル内に保管されます。

IMS Connect では、IMS Connect DB 経路指定ユーザー出口ルーチン (HWSROUTO) も使用できます。これを使用して、IMS DB にアクセスするクライアントからの着信メッセージの経路指定を詳細に制御することができます。

IMS Connect が、クライアント・アプリケーションまたは HWSROUTO 出口ルーチンからブランクの別名を受け取ると、IMS Connect は、IMS Connect が認識する ODBM のすべてのインスタンスの中で、着信接続をラウンドロビン方式で経路指定します。IMS Connect が、クライアント・アプリケーションまたは HWSROUTO 出口ルーチンから、ODBM の複数インスタンスによって共用される別名を受け取ると、IMS Connect は、別名を共用する ODBM のすべてのインスタンスの中で、その別名を指定する着信接続をラウンドロビン方式で経路指定します。

## IMS Connect の IMS TM Resource Adapter サポート

IMS TM に接続する Java アプリケーション・プログラムをサポートするため、IMS Connect では IMS TM Resource Adapter がサポートされています。

IMS TM Resource Adapter は、分散環境で、WebSphere Application Server のもとで稼働します。また、ホスト IMS システムで実行される IMS トランザクションにアクセスするために、Java アプリケーション、Java プラットフォーム、Enterprise Edition (Java EE) アプリケーション、または Web サービスによって使用されます。さらに、外部 Java EE アプリケーションに非同期コールアウト要求を出すために、IMS 従属領域で実行される IMS アプリケーションによっても使用されます。

IMS TM Resource Adapter は TCP/IP 通信プロトコルを介して IMS Connect に接続します。

Java アプリケーションが IMS TM Resource Adapter を通じて IMS にトランザクション要求をサブミットすると、IMS Connect は z/OS システム間カップリング・ファシリティー を使用して、そのトランザクション要求を IMS Open Transaction Manager Access (OTMA) に送信し、トランザクションが IMS で実行されます。応答は同じ経路を使用して Java アプリケーションに返されます。



IMS アプリケーションがコールアウト 要求を使用して、外部 Enterprise JavaBeans (EJB) コンポーネントまたは Web サービスを起動すると、IMS Connect は保留キューからそのコールアウト 要求をリトリブして、IMS TM Resource Adapter に渡します。次に IMS TM Resource Adapter は、WebSphere Application Server 内の、コールアウト 要求を受け取るようセットアップされている EJB アプリケーションにその要求を渡します。EJB は IMS TM Resource Adapter を介して、IMS Connect への接続を開始します。IMS TM Resource Adapter は IMS Connect に対してポーリングを行い、保留キューからコールアウト 要求をリトリブさせます。EJB は要求を処理し、通常の IMS トランザクション 要求を発行して IMS に何らかの応答データを返します。

IMS Connect は、IMS TM Resource Adapter によってサブミットされるクライアント ID を使って、TCP/IP 接続を識別します。

持続 TCP/IP ソケットでの接続の場合、IMS Connect は以下のいずれの場合にも IMS TM Resource Adapter 用にクライアント ID を生成できます。

- IMS TM Resource Adapter によってサブミットされたクライアント ID が既存の IMS TM Resource Adapter クライアント ID と重複しているときに、IMS TM Resource Adapter が IMS Connect に固有のクライアント ID を生成するよう要求した場合。
- IMS TM Resource Adapter がブランクのクライアント ID をサブミットした場合。

### 関連概念

[IMS TM Resource Adapter の概要](#)

## IMS Connect による ISC TCP/IP 通信のサポート

IMS Connect は、システム間連絡 (ISC) 並列セッションで IBM CICS Transaction Server for z/OS へのリンクに TCP/IP が使用される場合に、IMS のための TCP/IP 接続とプロトコルを管理します。

以下の図は、TCP/IP を使用する単一 ISC 並列セッションの基本的なフローを示しています。この図では、ISC TCP/IP 端末は、ETO ログオン記述子を使用して IMS に動的に定義されています。

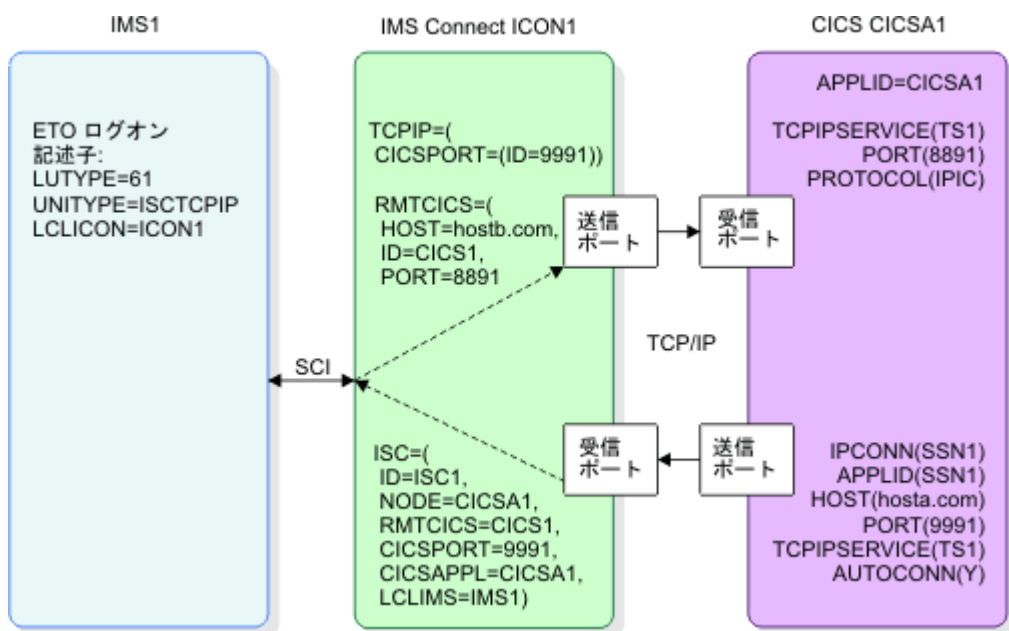


図 22. TCP/IP を使用する ISC 並列セッションに対する IMS Connect サポートの概要

それぞれの ISC 並列セッションには、IMS Connect に 2 つのソケット (送信ソケットと受信ソケット) が必要です。CICS でも、各 ISC 並列セッションに 2 つのソケットが必要です。ISC リンクは、複数の ISC 並列セッションをサポートすることができます。

ISC ステートメントと RMTCICS ステートメントは、TCPIP ステートメントの CICSPORT キーワードと共に、IMS Connect から IMS、および IMS Connect から CICS への TCP/IP 接続を定義します。これらのステートメントは、IMS.PROCLIB データ・セットの IMS Connect 構成メンバー内で定義されます。

各 TCP/IP ソケットは、IMS Connect と CICS の間のトランザクションおよび応答データ・メッセージの片方向パスを提供します。確認応答メッセージ (ACK または NAK) は、トランザクションまたは応答データ・メッセージが受信されたソケット接続で送信されます。

IMS Connect と IMS の間の接続は、IMS Common Service Layer (CSL) の Structured Call Interface (構造化呼び出しインターフェース) (SCI) によって使用可能にされます。

ISC TCP/IP 通信では、RACF PassTicket はサポートされていません。

CICS と IMS の間の ISC メッセージは、どの IMS Connect 出口ルーチンによっても変更または経路指定できません。IMS Connect ユーザー・メッセージ出口ルーチン (HWSJAVA0、HWSSMPLO、HWSSMPL1 など) は、ISC TCP/IP 通信には使用されません。ポート・メッセージ編集出口ルーチンは、サポートされません。

ISC TCP/IP 通信は、IMS タイプ 2 コマンドによってサポートされています。限られた数の IMS Connect WTOR コマンドおよび z/OS MODIFY コマンド (VIEWHWS および QUERY MEMBER など) のみが、ISC TCP/IP 通信をサポートしています。

IMS Connect イベント・レコーダー出口ルーチン (HWSTECLO) は、ISC TCP/IP 通信に固有のイベントを記録します。

### 関連タスク

607 ページの『[TCP/IP を介する CICS との ISC 通信](#)』

IMS サブシステムと IBM CICS Transaction Server for z/OS サブシステムの間の ISC 接続をサポートするために、TCP/IP を使用できます。

### 関連資料

[ISC ステートメント \(システム定義\)](#)

[RMTICICS ステートメント \(システム定義\)](#)

## IMS Connect による IMS 間 TCP/IP 通信のサポート

---

IMS Connect は、TCP/IP ネットワークを介して相互に通信する IMS システムの TCP/IP 接続およびプロトコルを管理します。

IMS 間 TCP/IP 接続は通常、異なる場所にインストールされた 2 つの IMS システムをリンクします。各 IMS システムは、同じ場所にある IMS Connect インスタンスと通信します。2 つの IMS システムが相互に通信するために使用する TCP/IP 接続は、それぞれの場所にある IMS Connect インスタンスによって確立されます。

次の図は、IMS 間 TCP/IP 接続の可能な構成例を示しています。

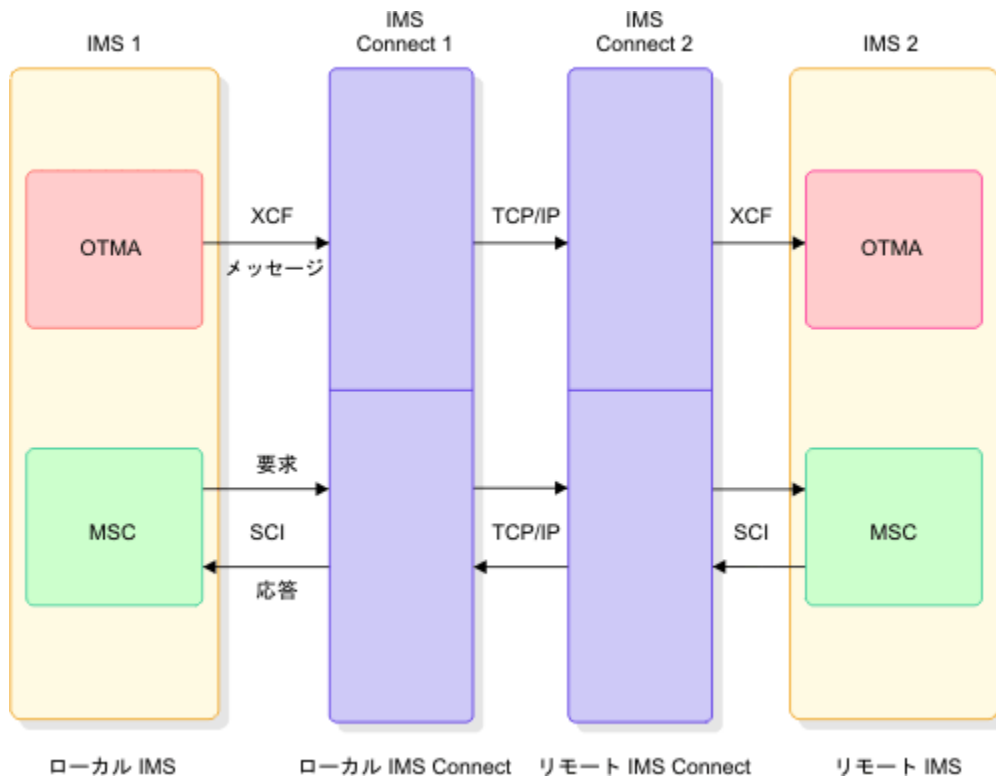


図 23. IMS 間 TCP/IP 通信の構成例

IMS.PROCLIB データ・セット内の IMS Connect 構成メンバーの RMTIMSCON ステートメントが、一方の IMS Connect インスタンスから他方への TCP/IP 接続を定義します。どの IMS 通信コンポーネントが TCP/IP 接続を使用するかに応じて、2つの IMS システム間の通信パスを形成するために、追加の構成ステートメント、記述子、またはシステム定義マクロが必要になります。

定義された各 TCP/IP 接続は、メッセージが一方の IMS システムから他方にフローするための片方向パスを提供します。メッセージを反対方向にフロー・バックするには、逆方向に 2 つ目の TCP/IP 接続を定義する必要があります。確認応答メッセージ (ACK または NAK) は、この規則の例外です。

オプションで、IMS Connect は RACF パスチケットを使用して、IMS 間 TCP/IP 接続をセキュアにすることができます。パスチケット・セキュリティーが使用可能な場合、送信側 IMS Connect インスタンスはパスチケットを生成し、受信側 IMS Connect インスタンスは RACF を使用してパスチケットを検証します。パスチケットが検証されると、受信側 IMS Connect はその接続をトラステッド・ユーザーから来たものとして分類します。接続が存続する限り、それ以上のセキュリティー検査は行われません。

IMS Connect ユーザー・メッセージ出力ルーチン (HWSJAVA0、HWSSMPL0、HWSSMPL1 など) は、IMS 間 TCP/IP 通信には使用されません。

IMS 間 TCP/IP 通信は、以下の IMS Connect コマンド・フォーマットによってサポートされています。

- WTOR
- z/OS MODIFY
- IMS タイプ 2

IMS Connect イベント・レコーダー出力ルーチン (HWSTECL0) は、IMS 間 TCP/IP 通信に固有のイベントを記録します。

IMS 間 TCP/IP 通信を使用する IMS 通信コンポーネントには、複数システム結合機能 (MSC) および Open Transaction Manager Access (OTMA) が含まれます。

### 関連タスク

IMS 間の TCP/IP 接続 (システム定義)

188 ページの『IMS 間 TCP/IP 接続の保護』

IMS 間 TCP/IP 接続を保護するために、IMS Connect は RACF パスチケットを使用して、IMS Connect の 1 つのインスタンスを IMS Connect のもう 1 つ別のインスタンスのトラステッド・ユーザーとして設定します。

#### 関連資料

[IMS PROCLIB データ・セットの HWSCFGxx メンバー \(システム定義\)](#)

[IMS Connect コマンド \(コマンド\)](#)

[QUERY IMSCON コマンド \(コマンド\)](#)

[UPDATE IMSCON コマンド \(コマンド\)](#)

[IMS Connect イベント・レコーダー出口ルーチン \(HWSTECLO\) \(出口ルーチン\)](#)

## MSC と IMS 間 TCP/IP 通信

MSC の場合、2 つの IMS Connect インスタンス間の TCP/IP 接続は、MSC 物理リンク用のパスを形成します。MSC 物理リンクには、1 つ以上の MSC 論理リンクを割り当てることができます。

MSC リンク上のメッセージは両方向に伝送されます。MSC 論理リンクごとに、送信ソケットと受信ソケットの 2 つのソケット接続が開かれます。MSC リンクをサポートするには、ペアになった対応する TCP/IP 接続を、IMS 間 TCP/IP 接続の両側の各 IMS Connect インスタンスで 1 つずつ定義する必要があります。

MSC 用の完全な IMS 間 TCP/IP 通信パスを定義するには、接続の両側にある IMS Connect インスタンスと IMS インスタンスで、以下の項目をコーディングする必要があります。

- IMS Connect では、IMS Connect 構成ステートメント。これには、IMS 間 TCP/IP 通信に必要な RMTIMSCON ステートメントおよび、IMS Connect と MSC の間の通信に必要な MSC ステートメントが含まれます。
- IMS では、すべての MSC リンク・タイプを定義するために必要な MSC システム定義マクロまたはタイプ 2 CREATE コマンド。MSPLINK マクロおよび CREATE MSPLINK コマンドには、TCP/IP MSC 物理リンク・タイプに固有のパラメーターがあります。

MSC 用の IMS 間 TCP/IP 通信をサポートするには、単純化された IMSplex 構成が必要です。IMS Connect と MSC の間の通信パスは、IMSplex の構造化呼び出しインターフェース (SCI) コンポーネントによって管理されます。IMS Connect と MSC に対する IMS タイプ 2 コマンド・サポートには、IMSplex の Operations Manager (OM) コンポーネントも必要です。

TCP/IP 汎用リソースを使用する MSC リンクの場合、IMS Connect は、経路指定および類似性管理のサポートを提供します。

TCP/IP 接続用のセキュリティは、オプションの RACF パスチケット・サポートを使用して実装できます。IMS システムには、トランザクション許可も実装できます。

各 IMS システムでは、TCP/IP MSC リンクは VTAM MSC リンクと同様に動作します。MSC 物理リンクを開始するには、リンクされた IMS システムのいずれか一方で、IMS タイプ 1 コマンド **/RSTART LINK** または IMS タイプ 2 コマンド **UPDATE MSLINK START(COMM)** を発行します。

IMS Connect では、リンクが開始された後に、IMS Connect WTOR コマンド、z/OS MODIFY コマンド、または IMS タイプ 2 コマンドを使用して、MSC リンクおよび関連のソケット接続のモニター、停止、および再始動が行えます。IMS では、定義、操作、および管理に関するタスクおよびプロセスの大部分が、MSC TCP/IP リンク・タイプの場合と MSC VTAM リンク・タイプの場合とで同じです。

#### 関連概念

683 ページの『[複数システム結合機能の概要](#)』

複数システム結合機能 (MSC) は、ある IMS に入力されたトランザクションを別の IMS で処理することを可能にします。

#### 関連タスク

[IMS 間の TCP/IP 接続 \(システム定義\)](#)

[MSC の TCP/IP 汎用リソース・グループの定義 \(システム定義\)](#)

#### 関連資料

[CREATE MSPLINK コマンド \(コマンド\)](#)

## IMS Connect、MSC、および TCP/IP 汎用リソース

IMS Connect は、TCP/IP 汎用リソース・グループに接続する MSC リンクに対して、経路指定および接続管理のサポートを提供します。

TCP/IP 汎用リソースを使用すると、IMSplex 外部の MSC パートナー・システムへのシステム定義の変更を行うことなく、IMSplex に参加している IMS システム間で MSC リンクを切り替えることができます。

TCP/IP 汎用リソース・グループに参加する IMS システムはすべて、それぞれの IMS DFSDCxxx PROCLIB メンバー内で、共用の汎用 IMS ID を指定します。IMS Connect では、この汎用 IMS ID は、各 IMS システムの MSC 構成ステートメントの GENIMSID パラメーターで指定します。

IMS Connect は、TCP/IP 汎用リソース・グループの物理リンク上で最初の MSC 論理リンク要求を受け取ると、グループに参加しているすべての IMS システムに要求をブロードキャストします。IMS Connect は、最初に IMS Connect に応答した IMS システムにリンク要求を経路指定します。

IMS システムがリンク要求を受け入れると、その IMS システムとその IMS システムへの物理リンクを使用するすべての論理リンクの間に、類似性が確立されます。類似性は、特定の MSC 物理リンク上のすべての論理リンクが正常に終了するか、物理リンク・パスが IMS Connect 内で停止されるまで、その物理リンク上のすべての論理リンクに対して持続します。最後の論理リンクが終了すると、類似性は解放され、その物理リンク上の次のリンク要求では、別の IMS システムとの類似性を確立することができます。

論理リンクの類似性情報の表示は、IMS 内および IMS Connect 内の両方で行えます。IMS Connect では、以下のいずれかの IMS Connect コマンドを指定することにより、類似性情報を表示できます。

- IMS タイプ 2 フォーマット・コマンド **QUERY IMSCON TYPE(MSC) NAME(lclPlkid)**
- IMS タイプ 2 フォーマット・コマンド **QUERY IMSCON TYPE(LINK) NAME(lclPlkid)**
- WTOR フォーマット・コマンド **VIEWMSC lclPlkid**
- z/OS MODIFY コマンド **QUERY MSC NAME(lclPlkid)**

TCP/IP 汎用リソース・グループ内のどの IMS システムが物理リンク上の最初の論理リンク要求を受け入れるかを制御することができます。これを行うための最も簡単な方法は、汎用リソース・グループ内のその IMS システムからリンクを開始することです。

リモート IMS システムからリンクを開始する必要がある場合は、どの IMS システムがリンクを受け入れるかを、TCP/IP 汎用リソース・グループからでも IMS Connect からでも制御できます。TCP/IP 汎用リソース・グループでは、類似性を必要とする IMS システムを除く、汎用リソース・グループ内のすべての IMS システム内の汎用 IMS ID へのログオンを停止します。あるいは、IMS Connect では、類似性を必要とする IMS システムへのパスを除く、すべての MSC 物理リンク・パスを停止します。

IMS Connect で物理リンク・パスを停止するには、以下のいずれかの IMS Connect コマンドを発行します。

- IMS タイプ 2 フォーマット・コマンド **UPDATE IMSCON TYPE(MSC) NAME(lclPlkid) STOP(COMM)**
- WTOR フォーマット・コマンド **STOPMSC lclPlkid**
- z/OS MODIFY コマンド **UPDATE MSC NAME(lclPlkid) STOP(COMM)**

### 関連タスク

746 ページの『[TCP/IP 汎用リソース・グループ内の MSC リンクの管理](#)』

MSC で TCP/IP 汎用リソースが使用される場合、TCP/IP 汎用リソース・グループに接続する各リンクは、グループ内の特定の IMS システムへの類似性を持ちます。

[MSC の TCP/IP 汎用リソース・グループの定義 \(システム定義\)](#)

### 関連資料

[MSC ステートメント \(システム定義\)](#)

[/DISPLAY AFFIN コマンド \(コマンド\)](#)

[QUERY MSLINK コマンド \(コマンド\)](#)

[QUERY IMSCON TYPE\(LINK\) コマンド \(コマンド\)](#)

[QUERY IMSCON TYPE\(MSC\) コマンド \(コマンド\)](#)

[VIEWMSC コマンド \(コマンド\)](#)

[IMS Connect QUERY MSC コマンド \(コマンド\)](#)



## OTMA と IMS 間 TCP/IP 通信

OTMA の場合、2 つの IMS Connect インスタンス間の TCP/IP 接続は、ローカルの送信側 IMS システムからリモートの受信側 IMS システムに OTMA トランザクション・メッセージを非同期で送信するための片方向パスを形成します。

リモート IMS システムによって生成された応答は、非同期でリトリブするために、リモート IMS システム内の T パイプ保留キューに入れられます。

OTMA 用の IMS 間 TCP/IP 通信パスを定義するには、接続の両側にある IMS Connect インスタンスおよび IMS インスタンスで、以下の項目をコーディングする必要があります。

- 送信側 IMS Connect では、IMS Connect 構成ステートメント。これには、IMS 間 TCP/IP 通信に必要な RMTIMSCON ステートメントおよび、IMS Connect と OTMA の間の通信に必要な DATASTORE ステートメントが含まれます。
- IMS では、IMS.PROCLIB データ・セットの DFSYDTx メンバー内の OTMA 宛先記述子、または OTMA ユーザー・データ・フォーマット設定出口ルーチン (DFSYDRU0)。

TCP/IP 接続用のセキュリティーは、オプションの RACF パスチケット・サポートを使用して実装できます。IMS システムには、トランザクション許可も実装できます。

OTMA では、TCP/IP 接続上でメッセージを送信する IMS Connect インスタンスにのみ、RMTIMSCON ステートメントが必要です。

OTMA は、コミット・モード 0 および確認応答付き送信専用プロトコルを使用して、IMS TCP/IP 接続を介してトランザクション・メッセージを送信します。リモート IMS システムで生成された、OTMA トランザクション・メッセージに対する応答は、非同期でリトリブするために T パイプの保留キューに入れられます。

応答を発信元 IMS システムに返すために別個の TCP/IP 接続を定義することができます。ただし、この場合、応答は発信元 IMS システムに対する新規トランザクションとして発信元システムに返ししなければならず、応答と元のトランザクションとの相関は、ご使用のシステムで管理する必要があります。

IMS Connect 内で、IMS Connect WTOR コマンド、z/OS MODIFY コマンド、または IMS タイプ 2 コマンドを使用して、TCP/IP 接続のモニター、停止、および再始動を行うことができます。

### 関連概念

801 ページの『IMS 間通信の OTMA サポート』

IMS 間 TCP/IP 通信を使用して、ローカル IMS システムからリモート IMS システムに OTMA メッセージを送信することができます。

### 関連タスク

[IMS 間の TCP/IP 接続 \(システム定義\)](#)

## IMS Connect、OTMA スーパーメンバー、および IMS 間 TCP/IP 接続

IMS Connect は、IMS 間 TCP/IP 通信での OTMA スーパーメンバー 機能をサポートします。OTMA スーパーメンバー機能は、IMS Connect インスタンス間のワークロードの可用性および分散に役立ちます。

IMS 間 TCP/IP 通信では、IMS Connect から OTMA スーパーメンバー・グループへの接続を最大 8 つ定義できます。接続は、IMS Connect の DATASTORE 構成ステートメントで定義されます。OTMA は各接続を 1 つのターゲット・メンバー (T メンバー) として数えます。接続は、1 つの IMS Connect インスタンスからも、複数の IMS Connect インスタンスからも可能です。

スーパーメンバー・グループを使用するために 8 つを超える接続が定義されている場合、OTMA は最初の 8 つの接続のみを使用します。アクティブに稼働している IMS Connect から 8 つを超える接続が定義されているときに IMS が再始動された場合、接続はランダムに結合されるため、同じ 8 つの接続がスーパーメンバー・グループに受け入れられるとは限りません。

接続が停止されたか、IMS Connect がシャットダウンしたために、接続がシャットダウンされると、OTMA はその接続をスーパーメンバー・グループから削除します。接続が開始された場合、グループが既にフルになっていない場合にのみ、OTMA はその接続をスーパーメンバー・グループに結合します。

フルのスーパーメンバー・グループから接続が削除された場合、OTMA は、以前に結合を試みた他の接続を自動的に追加しません。既に稼働中の IMS Connect インスタンスの接続を追加するには、その接続または IMS Connect インスタンスを再始動します。

#### 関連概念

802 ページの『IMS 間通信のスーパーメンバー・サポート』

OTMA スーパーメンバー機能を使用すると、リモート IMS システムに送信されるメッセージを、IMS Connect の複数のローカル・インスタンス間に分散させることができます。

#### 関連タスク

IMS 間の TCP/IP 接続 (システム定義)

#### 関連資料

DATASTORE ステートメント (システム定義)

## OTMA 接続の自動再接続の試行

IMS Connect がリモート IMS Connect インスタンスとの接続を確立できない場合、IMS Connect は自動的に 2 分ごとにリモート IMS Connect インスタンスへの再接続を試行します。

RMTIMSCON ステートメントに AUTOCONN=Y が指定されている場合、IMS Connect は始動時にリモート IMS Connect インスタンスへの接続を作成します。IMS Connect が始動時に作成する接続の数は、RMTIMSCON 構成ステートメントの RESVSOC キーワードで決定されます。

AUTOCONN=N の場合、IMS Connect は、リモート IMS システムへの配信のために OTMA からメッセージを受け取ったときにのみ、接続を作成します。

いずれの場合も、以下のいずれかの理由で IMS Connect が接続を作成できない場合、IMS Connect は 2 分ごとにリモート IMS Connect インスタンスへの再接続を試行します。

- ローカル TCP/IP が使用できない
- リモート TCP/IP が使用できない
- リモート IMS Connect が使用できない

再接続の試行の前、最中、または試行と試行の間に OTMA から何らかのメッセージを受け取った場合、IMS Connect は NAK を OTMA に送信して、T パイプ・キューの先頭にメッセージを残すように OTMA に指示します。リモート IMS Connect への接続が確立されると、IMS Connect は OTMA に対して、その接続でのメッセージの送信を再開するように通知します。

IMS Connect が再接続を試行している間、接続の状況は RETRY CONN です。また、開いているソケットの数を示す NUMSOC 値が変動します。再接続をアクティブに試行している間、IMS Connect はソケットをオープンします。IMS Connect は再接続の試行の 2 分間の間隔には、ソケットをクローズします。

#### 関連タスク

IMS 間の TCP/IP 接続 (システム定義)

## IMS Connect の XML 変換サポートの概要

一部の IMS Connect クライアントでは、IMS Connect は入力メッセージに収められた XML データを、COBOL または PL/I で作成された IMS アプリケーション・プログラムが使用するデータ構造に変換できます。これに対応する出力メッセージのデータも IMS アプリケーション・プログラムのプログラミング言語から IMS Connect クライアントが想定する XML データに変換されます。

IMS Connect の XML 変換サポートを使用すると、XML をサポートするように IMS アプリケーション・プログラムを作成または変更することなく、IMS が XML 形式のメッセージを受け入れることができます。

IMS Connect の XML 変換サポートは、HWSSOAP1 ユーザー・メッセージ出口を使用して、適切な XML アダプターと XML コンバーターを識別します。HWSSOAP1 ユーザー・メッセージ出口は、IMS Enterprise Suite SOAP Gateway および IMS Web 2.0 Solution for IBM Mashup Center で使用されます。

IMS Connect は、XML コンバーターへのインターフェースとして機能する XML アダプターを呼び出します。XML コンバーターは、IMS アプリケーション・プログラムの作成言語に応じて、IMS アプリケーション・プログラムの COBOL コピーブックまたは PL/I ソースに基づいて生成されます。XML データが IMS ア



アプリケーション・プログラムのプログラミング言語のデータ構造に変換されると、その入力メッセージは OTMA に渡されます。

#### 関連概念

219 ページの『IMS Connect XML メッセージ変換』

一部の IMS Connect クライアント・アプリケーション・プログラムでは、IMS Connect が XML メッセージを COBOL または PL/I に変換できるため、IMS に送信される XML メッセージを処理するために既存の IMS アプリケーション・プログラムを変更する必要はありません。

#### 関連タスク

IMS Connect クライアントの XML 変換サポートの構成 (システム定義)

## IMS Connect の z/OS Sysplex Distributor サポート

IMS Connect には、z/OS Sysplex Distributor 環境での実行を容易にするさまざまな機能が組み込まれています。

z/OS Sysplex Distributor 環境では、着信メッセージは通常、IMS Connect の複数インスタンス間に配布されて、ワークロードのバランスを取り、可用性を高めます。このような環境では、クライアント・アプリケーションは入力メッセージを受信する IMS Connect のインスタンスおよび非同期出力に対する後続の要求を受け取る IMS Connect を制御できません。OTMA を使用すると、IMS Connect はこのような環境での運用をサポートするための複数の機能を提供します。例えば、非同期出力の代替 T パイプへの転送、OTMA スーパーメンバー T パイプを使用した非同期出力の共用、別のクライアントに関連付けられた代替 T パイプ・キューからの出力のリトリブ、配信不能出力のページなどの機能があります。

IMS Connect は、サーバーの始動時に、サーバーのヘルス状況レポートを z/OS ワークロード・マネージャー (WLM) に自動的に送信します。ヘルス状況は、IMS Connect のヘルスを示す 0 から 100 の範囲の数値で、初期時点では 100 に設定されています。ヘルス値は、使用可能なソケットのパーセンテージ (つまり、TCPIP 構成ステートメントの MAXSOC パラメーターで設定される最大許容ソケット数に対する使用可能なソケット数のパーセンテージ) に基づいて定義されます。使用可能なソケットのパーセンテージおよびそれに対応するヘルス状況の数値は、以下の方法でマップされます。

- 最大許容ソケット数が 1000 以上の場合 (MAXSOC >= 1000):
  - 使用可能なソケットのパーセンテージが 11% から 100% の場合、ヘルス状況の数値は 100 です。
  - 使用可能なソケットのパーセンテージが 0% から 10% の場合、ヘルス状況の数値は使用可能なソケットのパーセンテージの数値と同等 (0 から 10) です。
- 最大許容ソケット数が 1000 より少ない場合 (MAXSOC < 1000):
  - 使用可能なソケットのパーセンテージが 21% から 100% の場合、ヘルス状況の数値は 100 です。
  - 使用可能なソケットのパーセンテージが 0% から 20% の場合、ヘルス状況の数値は使用可能なソケットのパーセンテージの数値と同等 (0 から 20) です。

サーバーのヘルスが変更されると、更新されたヘルス状況レポートが送信されます。

z/OS TCP/IP プロファイルの **VIPADISTRIBUTE** ステートメントで配布方式 **SERVERWLM** が指定されている場合、z/OS Sysplex Distributor は、この情報を使用して着信接続を経路指定します。また、**PORT** 定義に **SHAREPORTWLM** パラメーターを設定することで、Sysplex Distributor が WLM ヘルス状況を使用して、共用ポートを持つ同じホスト上で稼働している IMS Connect の複数のインスタンス間で着信接続のバランスを取るように構成することもできます。

#### 関連概念

303 ページの『コミット後送信出力の転送』

コミット後送信 (コミット・モード 0) IOPCB 出力をリトリブのために代替 OTMA T パイプ保留キューに転送するように IMS を構成することができます。

#### 関連タスク

875 ページの『非同期コミット後送信出力の共用: OTMA スーパーメンバー 機能』

保留キュー対応 OTMA クライアント (IMS Connect など) は、OTMA のスーパーメンバー 機能を有効にすることにより、非同期コミット後送信 (CM0) 出力メッセージを共用できます。OTMA スーパーメンバー 機能

は、z/OS Sysplex Distributor 環境で IMS Connect の複数のインスタンスをサポートするために、特に設計されたものです。

#### 301 ページの『配信不能なコミット後送信出力のページ』

トランザクションを開始した OTMA クライアント・アプリケーションにコミット後送信 (コミット・モード 0) IOPCB 出力を返せない場合に、その出力をページするように OTMA を構成することができます。

#### 345 ページの『代替 OTMA T パイプ保留キューからの出力のリトリブ』

クライアント・アプリケーションは、代替の T パイプの名前を、代替のクライアント ID である RESUME TPIPE 呼び出しとして指定することで、代替の T パイプ保留キューから非同期出力またはコールアウト・メッセージをリトリブできます。

#### **関連資料**

TCPIP ステートメント (システム定義)

## **IMS Connect セキュリティーの概要**

IMS Connect は、クライアントが IMS DB にアクセスするか、IMS TM にアクセスするかに応じて、異なるセキュリティ・オプションを提供します。

IMS DB クライアントは、IMS Connect DB セキュリティー・ユーザー出口ルーチン (HWSAUTH0)、RACF のようなセキュリティ製品、あるいはその両方を使用して、セキュリティを実装できます。IMS DB クライアントに対しては、IMS Connect は RACF パスチケットのサポートも提供します。Secure Sockets Layer (SSL) サポート用に、IMS DB クライアントは IBM z/OS Communications Server Application Transparent Transport Layer Security feature (AT-TLS) を使用できます。IMS Connect は、IMS DB クライアントに対して SSL サポートを提供しません。

IMS TM クライアントは、IMS Connect ユーザー・メッセージ出口ルーチン、ユーザー・セキュリティ出口ルーチン、および RACF のようなセキュリティ製品を任意に組み合わせて使用して、セキュリティを実装できます。IMS TM クライアントに対しては、IMS Connect は SSL の直接的なサポートおよび RACF パスチケットのサポートを提供します。

IMS 間 TCP/IP 接続の場合、IMS Connect は、RACF パスチケットを使用して、オプションの接続セキュリティを提供します。

IMS Connect 構成では、以下のコンポーネントのさまざまな組み合わせを使用して、セキュリティを実装できます。

- クライアント・サイド
  - クライアント・アプリケーション
  - クライアント・アプリケーションのサーバー
- IMS Connect
- RACF などのセキュリティ製品
- IMS TM 接続の場合
  - IMS Connect ユーザー・メッセージ出口ルーチン
  - OTMA (OTMA RESUME TPIPE セキュリティー ユーザー出口 (OTMARTUX) を含む)
- IMS DB 接続では、IMS Connect DB セキュリティー・ユーザー出口ルーチン
- IMS
- IMS 出口ルーチン
- IMS アプリケーション・プログラム

IMS Connect を構成する際に、以下の場所にセキュリティ指定を入力できます。

- HWSCFGxx 構成メンバー
- RACF FACILITY クラス

#### **関連概念**

181 ページの『IMS Connect セキュリティー・サポート』

IMS Connect には、さまざまなオプションが組み込まれており、それらのオプションによって、メッセージが IMS Connect に届いたとき、および、IMS TM 接続の場合はデータ・ストアに届いたときに、そのメッセージに対して実行するセキュリティー検査を実装および変更できます。

### 関連タスク

202 ページの『IMS DB との IMS Connect クライアント 接続用の RACF パスチケット』

RACF パスチケットを使用して、IMS DB との IMS Connect クライアント 接続を認証することができます。パスチケットは、RACF パスワードおよびパスワード・フレーズの代わりとして使用でき、ネットワーク経由でパスワードおよびパスワード・フレーズを暗号化されていないテキストとして送信しなくてもすむようになるため、セキュリティーを強化します。

## IMS Connect の定義および実行の概要

---

以降のステップでは、IMS Connect の定義および実行に関する概要について説明します。

### このタスクについて

IMS Connect の構成には、以下の概略ステップが含まれます。

### 手順

1. オンライン・コマンドまたは JCL ジョブを実行することにより、許可プログラム機能 (APF) に対して SDFSRESL を許可する。
2. z/OS プログラム特性テーブル (PPT) を更新して、IMS Connect を許可監視プログラム状態、およびキー 7 で確実に実行できるようにする。z/OS PPT 内の仕様が、IMS Connect 始動 JCL の EXEC ステートメントでのプログラム仕様と一致する必要があります。BPEINIOO または HWSHWS00 を指定できます。
3. IMS Connect 構成メンバーを作成し、IMS Connect が初期設定時に使用する IMS Connect 構成ステートメントを定義する。
4. IMS Connect の BPE 構成メンバーを作成する。
5. HWSUINIT 初期設定出口ルーチン (ロード・モジュールとしては出荷されない) を作成する。
6. IMS TM に接続している場合は、HWSSMPL0、HWSSMPL1、または HWSJAVA0 などの、必要なユーザー・メッセージ出口ルーチンを作成する。これらのユーザー・メッセージ出口ルーチンも、ロード・モジュールとしては出荷されません。
7. IMS Connect のセキュリティー、およびオプションで OTMA クライアントのセキュリティーを定義する。
8. オプションで、IMS Connect XML メッセージ変換サポートを使用可能にする。

### タスクの結果

z/OS プロシージャまたは z/OS ジョブを使用する IMS Connect を実行します。同一構成で IMS Connect の複数インスタンスを開始すると、接続障害が発生することがあります。

### 関連概念

[IMS Connect の定義と調整 \(システム定義\)](#)

## 第 11 章 IMS Connect 出口ルーチンの概要

IMS は、IMS Connect をサポートするために、各種の出口ルーチンを提供しています。

IMS Connect 出口ルーチンは、以下の 2 つの一般カテゴリーに分類されます。

- さまざまなタイプの IMS Connect TCP/IP クライアントが送受信するメッセージを管理するユーザー・メッセージ出口ルーチン
- セキュリティーやルーティングなどの一般機能を提供する出口ルーチン

IMS TM に接続するクライアントによってのみ使用される IMS Connect ユーザー・メッセージ出口ルーチンには、以下のものが含まれます。

- HWSSMPL0 および HWSSMPL1 ユーザー・メッセージ出口 (ユーザー作成の IMS Connect クライアント・アプリケーション用)
- HWSJAVA0 ユーザー・メッセージ出口 (IMS TM Resource Adapter 用)
- HWSSOAP1 ユーザー・メッセージ出口 (IMS Enterprise Suite SOAP Gateway 用)
- HWSCSLO0 および HWSCSLO1 ユーザー・メッセージ出口 (OM コマンド・クライアント 用)

すべての IMS Connect クライアント・ユーザー・メッセージ出口を使用することで、z/OS TCP/IP IMS リスナー・セキュリティー出口ルーチン (IMSLSECX) を呼び出したり、これらのユーザー・メッセージ出口ルーチンで RACF 機能を実行したり、あるいは IMS Connect ユーザー RACF 機能を使用したりできます。

**注意:** ユーザー・メッセージ出口内では、MVS WAIT を引き起こす z/OS 呼び出しを発行しないでください。ユーザー・メッセージ出口を変更して、MVS WAIT を引き起こすコードを追加すると、WAIT に対する通知が行われるまで TCP/IP PORT 上のすべての作業が停止します。ユーザー・メッセージ出口に渡されたストレージを解放するように出口を変更することはできません。また、ユーザー・メッセージ出口から IMS Connect に戻ったときに、IMS Connect は、出口によって獲得されたストレージを解放しません。IMS Connect によって獲得されたストレージはすべて、IMS Connect が解放しなければならず、ユーザー・メッセージ出口が解放しようとするると障害が発生します。

IMS Connect に一般機能を提供する出口ルーチンには、以下のものが含まれます。

- サンプル IMS Connect 宛先解決出口ルーチン (HWSYDRU0)。これは、OTMA ユーザー・データ・フォーマット設定出口ルーチン (DFSYDRU0) の変更バージョンです。
- IMS Connect ユーザー初期設定出口ルーチン (HWSUINIT)
- z/OS TCP/IP IMS リスナー・セキュリティー出口ルーチン (IMSLSECX)
- IMS Connect パスワード変更出口ルーチン (HWSPWCHO)
- IMS Connect イベント・レコーダー出口ルーチン (HWSTECL0)
- TCP/IP クライアント用 IMS Connect ポート・メッセージ編集出口ルーチン
- IMS Connect DB 経路指定ユーザー出口ルーチン (HWSROUT0)
- IMS Connect DB セキュリティー・ユーザー出口ルーチン (HWSAUTH0)

IMS Connect は、HWSUINIT と HWSJAVA0 を常にロードしますが、HWSSMPL0 と HWSSMPL1 はオプションであり、IMS.PROCLIB データ・セット (HWSCFGxx) の IMS Connect メンバー内の TCPIP ステートメントでこれらを組み込んだ場合にのみロードされます。この 4 つの出口ルーチンは、使いやすいようにロード・モジュールとして提供されています。ソース・コードも提供されているので、ご使用のシステムに合わせて出口ルーチンを変更することができます。

このトピックにはプロダクト・センシティブ・プログラミング・インターフェース情報が含まれています。

### 関連資料

[IMS Connect 出口ルーチン \(出口ルーチン\)](#)

## ユーザー・メッセージ出口ルーチンの概要

ほとんどのタイプの IMS Connect クライアントの場合、IMS Connect は、クライアントとの間で送受信されるメッセージを管理するために、ユーザー・メッセージ出口ルーチンを使用する必要があります。

ユーザー・メッセージ出口ルーチンは、以下のように、メッセージの管理に関連したさまざまなタスクを実行することができます。

- 入力メッセージを IMS および IMS Open Transaction Manager Access (OTMA) コンポーネントに必要なプロトコルまたはフォーマットに変換する
- メッセージを転送する
- 入力メッセージのセキュリティをチェックする
- 特定のユーザー定義基準に対応したユーザー定義メッセージを返す

以下に IMS Connect クライアントを、必要な IMS Connect ユーザー・メッセージ出口とともにリストします。

### IMS TM にアクセスするユーザー提供のクライアント

HWSSMPL0 または HWSSMPL1 ユーザー・メッセージ出口ルーチン、またはユーザー作成のユーザー・メッセージ出口ルーチン。

オプションで、IMS Connect ポート・メッセージ編集出口ルーチンも使用できます。これは、IMS Connect と z/OS TCP/IP スタック間で制御を受け取って、入出力メッセージのフォーマットを変更します。

HWSSMPL0 出口ルーチンおよび HWSSMPL1 出口ルーチンとそれらの関連のマクロは、ソース・コードおよびロード・モジュールの両方として IMS に付属しています。

### IMS TM Resource Adapter

HWSJAVA0 ユーザー・メッセージ出口ルーチン。

オプションで、IMS Connect ポート・メッセージ編集出口ルーチンも使用できます。これは、IMS Connect と z/OS TCP/IP スタック間で制御を受け取って、入出力メッセージのフォーマットを変更します。

HWSJAVA0 出口ルーチンと関連のマクロは、ロード・モジュールおよびソース・コードの両方として提供されます。

### IMS Enterprise Suite SOAP Gateway

HWSSOAP1 ユーザー・メッセージ出口ルーチン。これは、オブジェクト・コードとしてのみ提供されます。

オプションで、IMS Connect ポート・メッセージ編集出口ルーチンも使用できます。これは、IMS Connect と z/OS TCP/IP スタック間で制御を受け取って、入出力メッセージのフォーマットを変更します。

### IBM WebSphere DataPower®

HWSDPWR1 メッセージ出口ルーチン。これは、オブジェクト・コードとしてのみ提供されます。

オプションで、IMS Connect ポート・メッセージ編集出口ルーチンも使用できます。これは、IMS Connect と z/OS TCP/IP スタック間で制御を受け取って、入出力メッセージのフォーマットを変更します。

### IMS DB にアクセスするクライアント (IMS Universal ドライバー など)。

IMS Connect は、IMS DB にアクセスするクライアント (IMS Universal ドライバー や、分散リレーショナル・データベース体系 (DRDA) インターフェースを使用するユーザー提供のクライアントなど) に対しては、ユーザー・メッセージ出口ルーチンをサポートしません。代わりに、メッセージ経路指定、セキュリティ、およびメッセージ編集用の以下の IMS Connect 出口ルーチンを使用できます。

- IMS Connect DB 経路指定ユーザー出口ルーチン (HWSROUT0)
- IMS Connect DB セキュリティ・ユーザー出口ルーチン (HWSAUTH0)
- IMS Connect ポート・メッセージ編集出口ルーチン



## Operations Manager (OM) に対してコマンドをサブミットするクライアント

HWSCSLO0 または HWSCSLO1 ユーザー・メッセージ出口ルーチン。これらは、オブジェクト・コードとしてのみ提供されます。

IMS 間 TCP/IP 接続では、IMS Connect ユーザー・メッセージ出口ルーチンは使用されません。

### 関連資料

[IMS Connect ユーザー・メッセージ出口ルーチン \(出口ルーチン\)](#)

## セキュリティおよび IMS Connect ユーザー・メッセージ出口ルーチン

IMS Connect ユーザー・メッセージ出口ルーチンは、セキュリティ検査を実行できます。セキュリティを検査するように出口ルーチンを構成する場合、セキュリティ出口を提供するか、あるいは z/OS TCP/IP IMS リスナー・セキュリティ出口 (IMSLSECX) を使用する必要があります。

多くのオプションをセキュリティに使用でき、さらに、大多数のインストール済み環境が独自の特定セキュリティ方式を使用しているため、IMS は、サンプル・セキュリティ出口を提供していません。メッセージ出口がメッセージを戻すとき、OTMA ヘッダーに RACF パラメーターの指定があれば、IMS Connect は RACF の呼び出しを実行します。

IMSLSECX は、以下の IMS Connect ユーザー・メッセージ出口ルーチンによって呼び出されるセキュリティ出口の名前です。

- HWSSMPL0
- HWSSMPL1
- HWSOAP1
- HWSCSLO0

HWSSMPL0 または HWSSMPL1 を使用する場合、EXTRN IMSLSECX を選択した名前に変更することで、これらが呼び出すセキュリティ出口の名前を変更できます。セキュリティ出口の名前を変更する場合には、HWSSMPL0 または HWSSMPL1 ユーザー・メッセージ出口にそのセキュリティ出口を定義する必要があります。

また、HWSJAVA0 によって呼び出されるセキュリティ出口の名前を準備し、それを HWSJAVA0 メッセージ出口で定義することもできます。

IMSLSECX 出口ルーチンについての詳細は、を参照してください。

- *IMS V15* 出口ルーチン
- *z/OS Communications Server: IP IMS* ソケット・ガイド

### 関連概念

[190 ページの『IMS Connect セキュリティ出口ルーチン』](#)

IMS Connect ユーザー・メッセージ出口でセキュリティ検査を実行する場合、セキュリティ出口ルーチンを提供するか、あるいは z/OS TCP/IP IMS リスナー・セキュリティ出口ルーチン (IMSLSECX) を使用する必要があります。

[190 ページの『IMS Connect セキュリティおよび OTMARTUX ユーザー出口』](#)

OTMA RESUME TPIPE セキュリティ・ユーザー出口 (OTMARTUX) は IMS Connect 出口ルーチンではありませんが、OTMA 非同期保留キューに入れられたメッセージを保護するために使用できる 2 つの可能な方法の 1 つです。もう 1 つは、RACF などの外部セキュリティ製品を使用する方法です。OTMARTUX ユーザー出口と外部セキュリティ製品をそれぞれ単独で、または組み合わせて使用することができます。

### 関連資料

[191 ページの『HWSSMPL0 および HWSSMPL1 セキュリティ処置』](#)

サンプル・ユーザー・メッセージ出口 HWSSMPL0 および HWSSMPL1 は、特定のセキュリティー処置を常に実行し、IMSLSECX セキュリティー出口が呼び出された場合または呼び出されない場合にのみ他のセキュリティー処置を実行します。

## ユーザー定義メッセージ

IMS Connect の HWSJAVA0、HWSSMPL0、および HWSSMPL1 ユーザー・メッセージ出口では、ユーザーが定義した基準を満たしたとき、IMS Connect クライアントにユーザー定義のメッセージを返すことができます。

ユーザー定義メッセージを受け取る可能性のあるクライアント・アプリケーション・プログラムは、ユーザー定義メッセージ、およびすべての関連する戻りコードと理由コードを認識して処理することができなければなりません。ユーザー定義メッセージがクライアント・アプリケーション・プログラムに返される場合、元の入力メッセージは IMS に送信されません。

ユーザー・メッセージ出口がユーザー定義メッセージを返すとき、元の入力メッセージは渡されません。

このようなユーザー定義メッセージの 1 つを返す際に、必要に応じて、IMS Connect がソケット接続を開いた状態または閉じた状態にしておくように、ユーザー・メッセージ出口を設定することもできます。

ユーザー定義メッセージの長さは、1 文字から 128 文字の間で設定できます。128 文字より長いメッセージは切り捨てられます。

定義するメッセージが、IMS Connect からユーザー・メッセージ出口ルーチンが受信するクライアント入力メッセージよりも長い場合は、その出口が返す EXPINI\_BUFINC フィールドで、必要な追加のバイトを指定して、出口ルーチンが使用するバッファー・サイズを増やします。

ユーザー定義メッセージを返したあとに IMS Connect がソケット接続を開いたままにするよう要求するには、出口ルーチンが EXPREA\_RETCODE を 20 (X'14') に設定する必要があります。ソケット接続を終了させるには、出口ルーチンは EXPREA\_RETCODE を 4 (X'04') に設定する必要があります。IMS Connect は他の要因によっても接続を終了する可能性があるため、EXPREA\_RETCODE に 20 を指定しても、ソケット接続が確実に開いたままでは限りません。

HWSJAVA0 の場合、ユーザー定義メッセージ・テキストを返すことも、戻りコードと理由コードのみを返すこともできます。ユーザー定義メッセージ・テキストを返すには、HWSJAVA0 ユーザー・メッセージ出口は OMUSER\_RETCODE を 48 (X'30') に、OMUSR\_RSNCODE を ICONSUCC に設定する必要があります。OMUSER\_RETCODE と OMUSR\_RSNCODE にこれ以外の値の組み合わせを設定すると、ユーザー定義メッセージ・テキストを伴わない戻りコードと理由コードが返されます。

HWSSMPL0 および HWSSMPL1 の場合は、メッセージ・テキストを出力メッセージ・バッファーに入れることにより、ユーザー定義メッセージ・テキストを返します。

### 関連資料

[ユーザー・メッセージ出口ルーチン HWSSMPL0 および HWSSMPL1 \(出口ルーチン\)](#)

[IMS TM Resource Adapter ユーザー・メッセージ出口ルーチン \(HWSJAVA0\) \(出口ルーチン\)](#)

## 機能に固有の出口ルーチンの概要

IMS では、柔軟性を高めるために、IMS Connect 用の特定の機能を実行するいくつかの IMS Connect 出口ルーチンが提供されています。

IMS Connect では、次のような機能固有の出口ルーチンを使用できます。

- IMS Connect ユーザー初期設定出口ルーチン (HWSUINIT)
- IMS Connect DB 経路指定ユーザー出口ルーチン (HWSROUT0)
- IMS Connect DB セキュリティー・ユーザー出口ルーチン (HWSAUTH0)
- IMS Connect サンプル OTMA 宛先解決出口ルーチン (HWSYDRU0)
- z/OS TCP/IP IMS リスナー・セキュリティー出口 (IMSLSECX)
- IMS Connect イベント・レコーダー出口ルーチン (HWSTECL0)
- IMS Connect パスワード変更出口ルーチン (HWSPWCHO)



## 関連資料

IMS Connect 機能固有の出口ルーチン (出口ルーチン)

# IMS Connect 出口ルーチンをサポートするマクロ

IMS は、IMS Connect 出口ルーチンをサポートするマクロを提供します。

## IMS Connect 出口ルーチンに使用されるマクロ

マクロには、以下のものがあります。

### HWSAUTPM

IMS Connect DB セキュリティー・ユーザー出口ルーチン (HWSAUTH0) のパラメーター・リストをマップします。このマクロのコピーは SDFSMAC に入っています。

### HWSEXPIO

IMS Connect ポート・メッセージ編集出口ルーチン (HWSPIOX0) のパラメーター・リストをマップします。このマクロのコピーは SDFSMAC に入っています。

### HWSEXPRM

各サブルーチンの呼び出し時に、ユーザー出口ルーチンに渡されるパラメーター・リストをマップします。このマクロのコピーは SDFSMAC に入っています。その構造を調べるには、このマクロをアセンブルしてください。

### HWSOMPFX

各 READ サブルーチン呼び出しでユーザー出口ルーチンが戻す出力バッファーに、また、各 XMIT サブルーチン呼び出しでユーザー出口に渡される入力バッファーに、OTMA メッセージ接頭語フォーマットをマップします。このマクロのコピーは SDFSMAC に入っています。その構造を調べるには、このマクロをアセンブルしてください。

### HWSIMSCB

HWSSMPLO および HWSSMPL1 ユーザー・メッセージ出口ルーチンによって使用される IMS 要求メッセージ (IRM) ヘッダーおよび BPE ヘッダー・フォーマットをマップします。このマクロのコピーは SDFSMAC に入っています。その構造を調べるには、このマクロをアセンブルしてください。

### HWSIMSEA

HWSSMPLO および HWSSMPL1 ユーザー・メッセージ出口ルーチンによって使用されるストレージ域をマップします。このマクロのコピーは SDFSMAC に入っています。その構造を調べるには、このマクロをアセンブルしてください。

### HWSROUPM

各サブルーチンの呼び出し時に、IMS Connect DB 経路指定ユーザー出口ルーチン (HWSROUT0) に渡されるパラメーター・リストをマップします。このマクロのコピーは SDFSMAC に入っています。その構造を調べるには、このマクロをアセンブルしてください。

### HWSXIB

IMS Connect ユーザー・メッセージ出口ルーチンおよび HWSUINIT 出口ルーチンによって使用される出口インターフェース・ブロックをマップします。IMS Connect DB 経路指定ユーザー出口ルーチンによって使用されるデータ・ストア・リスト (HWSXIBDS) と HWSXIB1 制御ブロックのアドレスが含まれます。このマクロのコピーは SDFSMAC に入っています。その構造を調べるには、このマクロをアセンブルしてください。

### HWSXIB1

HWSUINIT ユーザー出口ルーチンが使用する出口インターフェース・ブロックをマップします。HWSXIB1 には、ODBM リストとオプションのユーザー・データのアドレスが含まれます。HWSXIB1 出口インターフェース・ブロックは、HWSXIB が指しています。このマクロのコピーは SDFSMAC に入っています。その構造を調べるには、このマクロをアセンブルしてください。

### HWSXIBDS

IMS Connect ユーザー・メッセージ出口ルーチンおよび HWSUINIT 出口ルーチンによって使用される出口インターフェース・ブロック・データ・ストア・リスト内の項目をマップします。このリストには、データ・ストア名、データ・ストアの可用性および状況情報、およびユーザー・フィールドが含まれます。このマクロのコピーは SDFSMAC に入っています。その構造を調べるには、このマクロをアセンブルしてください。

## HWSXIBOD

IMS Connect にとって既知の各 ODBM インスタンスの名前と状況、さらにユーザー・フィールドおよび各 ODBM インスタンスに関連付けられた IMS 別名の名前と状況が含まれる ODBM リストをマップします。HWSXIBOD のアドレスは、HWSXIB1 出口インターフェース・ブロックに保管されています。このマクロのコピーは SDFSMAC に入っています。その構造を調べるには、このマクロをアSEMBルするか、またはマクロのプロログを参照してください。

## 出口インターフェース・ブロック

IMS Connect は、出口インターフェース・ブロックを提供して、IMS DB システムまたは IMS TM システムのいずれかへの接続をサポートする IMS Connect 出口ルーチンの処理をサポートします。

## IMS TM への接続用の XIB 出口インターフェース・ブロック

IMS Connect は、XIB 出口インターフェース・ブロックを提供して、IMS TM への接続時に使用される IMS Connect ユーザー・メッセージ出口ルーチンの処理をサポートします。XIB 出口インターフェース・ブロックとそれが含むユーザー域を使用して、ユーザーの出口ルーチンによって使用される情報を保管することができます。

IMS Connect は、出口インターフェース・ブロックに以下のタイプの情報を保管します。

- XIBDS に関する情報。
- 使用される IRM アーキテクチャー・レベルに関する情報。
- ユーザー・データ。

また、いかなる目的にも出口インターフェース・ブロックの XIB\_USERAREA フィールドを使用できます。IMS Connect ユーザー初期設定出口ルーチン (HWSUINIT) または各自の IMS Connect ユーザー・メッセージ出口ルーチンをコーディングすると、XIB\_USERAREA フィールドを使用できます。

この出口インターフェース・ブロックは、HWSXIB マクロによってマップされます。

## XIB 出口インターフェース・ブロックのフォーマット

XIB 出口インターフェース・ブロックは、HWSXIB マクロによってマップされます。

XIB 出口インターフェース・ブロックは、IMS Connect 出口ルーチンの処理をサポートするために用意されています。例えば、XIB 出口インターフェース・ブロックは、IMS Connect ユーザー・メッセージ出口ルーチンが処理中に使用するテーブルのアドレスを保管するために使用できます。

以下の表に、出口インターフェース・ブロックのフィールドおよびフィールド・オフセットについて記述します。

フィールド	10 進 オフ セッ ト	16 進 オフ セッ ト	長さ	値
XIB_HEADER	0	X'00'	0	
XIB_EYE	0	X'00'	4	目印。
XIB_DATASTORES	4	X'04'	4	XIBDS データ・ストア・リストのアドレス。
XIB_UFLD_CNT	8	X'08'	4	XIB_USERAREA 内のフルワード・ユーザー・フィールドの数 (16 進形式)。
XIB_XIBDS_LEN	12	X'0C'	2	XIBDS エントリーの長さ。

フィールド	10進 オフ セット	16進 オフ セット	長さ	値
XIB_ARCHLVL	14	X'0E'	1	XIB 出口インターフェース・ブロックのアーキテクチャー・レベル。 <b>X'01'</b> XIB_ARCH1: アーキテクチャー・レベル 1 <b>X'02'</b> XIB_ARCH2: アーキテクチャー・レベル 2 <b>X'03'</b> XIB_ARCH3: アーキテクチャー・レベル 3 <b>XIB_ARCH3</b> 最高位のアーキテクチャー・レベル
	15	X'0F'	1	予約済み。
XIB_VERSION	16	X'10'	4	IMS Connect インスタンスの IMS バージョン。
XIB_XIB1	20	X'14'	4	IMS DB への接続に対する XIB1 出口インターフェース・ブロックのアドレス。
	24	X'18'	12	予約済み。
XIB_USERAREA	36	X'24'	0F	XIB のユーザー域の開始。

## IMS TM データ・ストア情報用の XIBDS 出口インターフェース・ブロック

IMS TM への接続では、IMS Connect は、XIBDS 出口インターフェース・ブロック内のデータ・ストア情報に関する項目で、IMS データ・ストアの状況を追跡します。IMS Connect ユーザー・メッセージ出口ルーチンは、XIBDS 出口インターフェース・ブロックを参照して、データ・ストアの状況に基づいて着信メッセージの経路指定を判断することができます。

IMS Connect は、IMS データ・ストアに関して出口インターフェース・ブロック・データ・ストア項目に以下のタイプの情報を保管します。

- IMS データ・ストアの可用性。
- IMS データ・ストアが、IMS Connect と異なる z/OS イメージ上で実行されるかどうか。
- 異なる z/OS イメージ上で稼働する IMS データ・ストアにグローバル RRS トランザクションをカスケードするためのサポートを使用可能にするかどうか。
- IMS データ・ストアの状態。つまり、データ・ストアによるメッセージの処理状況、および処理機能が低下している場合、または完全に使用不可の場合には、IMS データ・ストアのどのような条件によって機能低下状態や使用不可状態が発生しているか。
- 最新の状況変更または OTMA からのハートビート・メッセージの時刻を記録したタイム・スタンプ。

データ・ストア接続が最初に確立されたとき、または OTMA が IMS Connect にデータ・ストアの状態の変更を通知したときに、IMS Connect は、IMS データ・ストアの状態情報を更新します。

OTMA は、60 秒ごとハートビート・メッセージを発行して、データ・ストアが引き続き通信していることを示します。データ・ストア項目のタイム・スタンプが 60 秒より前の場合、OTMA に問題が発生している可能性があります。

また、いかなる目的にも出口インターフェース・ブロック・データ・ストア項目の XIBDS\_USER フィールドを使用できます。IMS Connect ユーザー初期設定出口ルーチン (HWSUINIT) をコーディングして、IMS Connect の始動時に XIBDS\_USER フィールドを設定できます。

以下の表に示すように、XIBDS 出口インターフェース・ブロック・データ・ストア項目は、HWSXIBDS マクロによってマップされます。

## XIBDS 出口インターフェース・ブロックのフォーマット

XIBDS 出口インターフェース・ブロック・データ・ストア・エンタリーは、HWSXIBDS マクロによってマップされます。

以下の表に、出口インターフェース・ブロックのエンタリーのフィールドおよびフィールド・オフセットについて記述します。

フィールド	10 進オ フセ ット	16 進オ フセ ット	長さ	値
XIBDS_NAME	0	X'00'	8	データ・ストアの名前。
XIBDS_STATUS	8	X'08'	1	データ・ストアの可用性。とりうる値は次のとおりです。 <b>X'00'</b> データ・ストアはアクティブではありません。 <b>X'01'</b> データ・ストアがアクティブです。 <b>X'02'</b> データ・ストアは切断されています。
XIBDS_FLAG	9	X'09'	1	データ・ストア・エンタリー・フラグ。 <b>X'80'</b> 出口インターフェース・ブロックの最終エンタリーを識別します。 <b>X'40'</b> IMS データ・ストアは、IMS Connect とは異なる z/OS イメージ (LPAR) 上で実行します。 このフラグを設定できるのは、データ・ストア接続が IMS Connect でアクティブであり、IMS データ・ストアが XCF グループでアクティブであるときのみです。 <b>X'20'</b> データ・ストアに対して、グローバル RRS トランザクション synchlevel=2 (syncpoint) のカスケード・トランザクション・サポートが使用可能になりました。 このフラグを設定できるのは、XIBDS_FLAG フィールドにも X'40' が含まれている場合のみです。これは、IMS Connect と IMS データ・ストアが異なる z/OS イメージ (LPAR) 上にあることを示します。 <b>X'10'</b> IMS データ・ストアのバージョンがオフセット 10 に格納されていることを示します。 <b>X'08'</b> IMS データ・ストア用のエンタリー。 <b>X'04'</b> IMSPlex 用のエンタリー。

フィールド	10 進オ フセ ット	16 進オ フセ ット	長さ	値
IMS バージョン	10	X'0A'	2	IMS データ・ストアのバージョンがここに含まれるのは、X'10' フラグがオフセット 9 にある XIBDS_FLAG フィールドでオンになっている場合のみです。バージョン番号の形式は、次のとおりです。  <ul style="list-style-type: none"> <li>• IMS リリース (1 バイト)</li> <li>• IMS レベル (1 バイト)</li> </ul>
XIBDS_USER	12	X'0C'	4	ユーザー・フィールド
XIBDS_ST_STATUS	16	X'10'	2	IMS データ・ストアの全体の状態。このフィールドの値は、データ・ストアによる入力メッセージの処理状況を示します。とりうる値は次のとおりです。  <b>3</b> 正常状態: IMS は使用可能であり、入力メッセージを正常に処理しています。  <b>2</b> 機能低下状態: IMS は低速で OTMA メッセージを処理しています。IMS が通常のように高速で入力メッセージを処理していないことが、1 つ以上の状態により示されます。  <b>1</b> 使用不可状態: IMS は入力メッセージを受け入れて処理することができません。1 つ以上の重大な条件が、IMS による OTMA メッセージの処理を妨げています。
XIBDS_ST_SVRSTT			0	データ・ストア処理の使用不可状態の原因となっている条件を識別するために使用されるフィールドの始まり。
XIBDS_ST_SVRFLG1	18	X'12'	1	予約済み
XIBDS_ST_SVRFLG2	19	X'13'	1	予約済み
XIBDS_ST_SVRFLG3	20	X'14'	1	予約済み
XIBDS_ST_SVRFLG4	21	X'15'	1	<b>X'01</b> この IMS Connect インスタンスからのメッセージでデータ・ストアがあふれているため、このインスタンスからの入力をこれ以上受け入れません。
XIBDS_ST_WRNSTT			0	データ・ストア処理の機能低下状態の原因となっている条件を識別するために使用されるフィールドの始まり。
XIBDS_ST_WRNFLG1	22	X'16'	1	<b>X'80'</b> データ・ストアによる処理を待機しているメッセージの全体数が、この z/OS システム間カップリング・ファシリティー (XCF) グループの全 OTMA クライアントについて定義されている待機メッセージの最大許容数の 80 パーセントを超えました。待機メッセージ数が最大許容数の 100 パーセントに到達すると、OTMA はあふれ状態を設定し、XCF グループ内の全 OTMA クライアントからのすべての着信メッセージを拒否します。
XIBDS_ST_WRNFLG2	23	X'17'	1	予約済み
XIBDS_ST_WRNFLG3	24	X'18'	1	予約済み

フィールド	10 進オ フセ ット	16 進オ フセ ット	長さ	値
XIBDS_ST_WRNFLG4	25	X'19'	1	<b>X'01'</b> データ・ストアによる処理を待機しているメッセージの数が、この IMS Connect のインスタンスについて OTMA で定義されている待機メッセージの最大許容数の 80 パーセントを超えました。待機メッセージ数が最大許容数の 100 パーセントに到達すると、OTMA はこの IMS Connect のインスタンスからのすべての着信メッセージを拒否します。
XIBDS_ST_UTC	26	X'1A'	12	この状況が OTMA によって発生した UTC 時刻。

## IMS DB への接続用の XIB1 出口インターフェース・ブロック

XIB1 出口インターフェース・ブロックは、Open Database Manager (ODBM) を介した IMS DB への接続に対して、IMS Connect 出口ルーチンをサポートします。

XIB1 出口インターフェース・ブロックは、ODBM および IMS データ・ストアに関する情報を保管する XIBOD 出口インターフェース・ブロックのアドレスを含みます。また、XIB1 出口インターフェースを使用して、IMS DB 経路指定ユーザー出口ルーチン (HWSROUT0) など、出口ルーチンによって使用されるユーザー・データを保管することもできます。

いかなる目的にも出口インターフェース・ブロックの XIB1\_USERAREA フィールドを使用できます。IMS Connect ユーザー初期設定出口ルーチン (HWSUINIT) と IMS Connect DB 経路指定ユーザー出口ルーチンの両方とも、XIB1\_USERAREA フィールドを使用するためにコーディングできます。

この出口インターフェース・ブロックは、HWSXIB1 マクロによってマップされます。

### XIB1 出口インターフェース・ブロックのフォーマット

XIB1 出口インターフェース・ブロックは、HWSXIB1 マクロによってマップされます。

以下の表に、XIB1 出口インターフェース・ブロックのフィールドおよびフィールド・オフセットについて記述します。

フィールド	10 進オ フセ ット	16 進オ フセ ット	長さ	値
XIB1_HEADER	0	X'00'	0	ダブルワードで調整。
XIB1_EYE	0	X'00'	4	目印。
XIB1_ODBMS	4	X'04'	4	ODBM とデータ・ストアの情報を含む XIBOD 出口インターフェース・ブロックのアドレス。
XIB1_UFLD_CNT	8	X'08'	4	ユーザー・フィールド・カウント。
XIB1_ARCHLVL	12	X'0C'	1	XIB1 出口インターフェース・ブロックのアーキテクチャー・レベル。 <b>X'01'</b> XIB1_ARCH1 - アーキテクチャー・レベル 1 <b>XIB1_ARCH1</b> 最高位のアーキテクチャー・レベル
予約済み	13	X'0D'	3	IMS Connect 用に予約済み。
XIB1_XIB	16	X'10'	4	XIB のアドレス。

フィールド	10 進オ フセ ット	16 進オ フセ ット	長さ	値
予約済み	20	X'24'	16	予約済み。
XIB1_USERAREA	36	X'24'	0F	XIB1 のユーザー域の開始。

## ODBM および IMS DB データ・ストア情報用の XIBOD 出口インターフェース・ブロック

IMS DB への接続では、IMS Connect は、ODBM およびデータ・ストア情報に関する XIBOD 出口インターフェース・ブロック内の項目で、Open Database Manager (ODBM) インスタンスおよび IMS データ・ストアの状況を追跡します。IMS Connect DB 経路指定ユーザー出口ルーチンは、XIBOD を参照し、ODBM インスタンスとデータ・ストアがアクティブかどうかに基づいて、着信メッセージの経路指定を判断することができます。

IMS Connect にとって既知の ODBM インターフェースに関して、XIBOD は、以下の ODBM 状態を追跡します。

- ODBM が実行中であり、IMS Connect に接続されている。
- ODBM は実行中であるが、IMS Connect には接続されていない。IMS Connect WTOR コマンド **STARTOD** または IMS Connect タイプ 2 コマンド **UPDATE IMSCON TYPE(ODBM) START(COMM)** を発行して、ODBM への接続を確立することができます。
- ODBM が実行されていない。IMS Connect と ODBM の間に接続を確立する前に、ODBM を再始動する必要があります。

XIBOD ブロックも、ODBM バージョン情報を保管します。

データ・ストアは、ODBM システム定義中に ODBM 構成メンバー CSLDCxxx 内でデータ・ストアに割り当てられた別名によって IMS Connect に認識されています。IMS Connect が着信要求を特定のデータ・ストアに経路指定するためには、データ・ストアの別名が、それが定義されている ODBM 内、および IMS Connect 内の両方でアクティブでなければなりません。

IMS Connect にとって既知の ODBM インスタンスに対して定義された各別名に関して、XIBOD は、別名で表されたデータ・ストアに対する接続の以下の状態を追跡します。

- データ・ストアに対する接続が完了している。別名は、ODBM インスタンスと IMS Connect (XIBOD\_ICACTIVE EQU X'20') の両方でアクティブです。
- データ・ストアに対する接続が完了していない。別名は、ODBM でアクティブですが、IMS Connect (XIBOD\_IACTIVE EQU X'80') ではアクティブ化されていません。
- データ・ストアに対する接続が完了していない。別名は、ODBM と IMS Connect の両方でアクティブでしたが、IMS Connect ではもうアクティブではありません。別名は、引き続き ODBM (XIBOD\_ICINACTIVE EQU X'10') でアクティブです。
- データ・ストアに対する接続が完了していない。別名は、ODBM と IMS Connect の両方でアクティブでしたが、ODBM ではもうアクティブではありません。その結果として、別名は、IMS Connect (XIBOD\_IINACTIVE EQU X'40') ではもうアクティブではありません。

また、いかなる目的にも、出口インターフェース・ブロック・データ・ストア項目の XIBOD\_USER フィールドを使用することができます。IMS Connect ユーザー初期設定出口ルーチン (HWSUINIT) をコーディングして、IMS Connect の始動時に XIBOD\_USER フィールドを設定できます。

この出口インターフェース・ブロック・データ・ストア項目は、HWSXIBOD マクロによってマップされます。



## XIBOD 出口インターフェース・ブロックのフォーマット

ODBM およびデータ・ストアの情報用の XIBOD 出口インターフェース・ブロックは、HWSXIBOD マクロによってマップされます。

### XIBOD 出口インターフェース・ブロックでの ODBM エントリーのマップ

XIBOD 出口インターフェース・ブロックの ODBM エントリーは、以下の表に示すように、HWSXIBOD マクロの HWSXIBOD DSECT によってマップされます。

フィールド	10 進オ フセ ット	16 進オ フセ ット	長さ	値
XIBOD_HDR	0	0	0	
XIBOD_EYE	0	0	4	文字データ。 <b>XIBOD</b> XIBOD_EYEID: 目印。
XIBOD_NAME	4	X'04'	8	Open Database Manager (ODBM) インスタンスの名前。
XIBOD_OSTATUS	12	X'0C'	1	ODBM の状況。とりうる値は次のとおりです。 <b>X'80'</b> XIBOD_OACTIVE: ODBM は稼働中であり、IMS Connect に接続されています。 <b>X'40'</b> XIBOD_OINACTIVE: ODBM は稼働中ですが、IMS Connect に接続されていません。IMS Connect WTOR コマンド <b>STARTOD</b> または IMS Connect タイプ 2 コマンド <b>UPDATE IMSCON TYPE (ODBM) START (COMM)</b> を発行して、ODBM への接続を確立することができます。 <b>X'20'</b> XIBOD_ODISC: ODBM が稼働していないか、IMSplex のメンバーではなくなっています。IMS Connect と ODBM との間で接続が作成できるようにするには、まず IMSplex で ODBM を再始動する必要があります。
XIBOD_ODBMRRS	13	X'0D'	1	文字 Y または N。 Y は、ODBM が z/OS リソース・リカバリー・サービス (RRS) を使用していることを示します。 N は、ODBM が RRS を使用していないことを示します。
予約済み	14	X'0D'	2	予約済み
XIBOD_ODBMVER	16	X'10'	4	ODBM バージョン番号
XIBOD_USER	20	X'14'	4	ユーザー・フィールド
XIBOD_NEXTODBM	24	X'18'	4	次の ODBM のアドレス
XIBOD_NEXTATBL	28	X'1C'	4	アドレス XIBOD_NEXTODBM で ODBM に定義された別名のテーブルのアドレス。
予約済み	32	X'20'	16	予約済み
XIBOD_IMSATABLE	48	X'30'	512	文字データ。別名テーブルには、32 個の別名エントリーがあります。それぞれの別名エントリーの長さは 16 バイトです。別名エントリーのマップについては、以下の表を参照してください。

## XIBOD 出口インターフェース・ブロックでの別名エントリーのマップ

別名エントリーは、以下の表に示すように、HWSXIBOD マクロの XIBOD\_IMSAENT DSECT によってマップされます。XIBOD 出口インターフェース・ブロックの別名テーブルには、32 個の別名エントリーが含まれます。

フィールド	10 進オ フセ ット	16 進オ フセ ット	長さ	値
XIBOD_IMSA	0	0	4	文字データ。別名。
XIBOD_ISTATUS	4	X'04'	1	別名で指定されたデータ・ストアへの接続状況。 <b>X'80'</b> XIBOD_IACTIVE: データ・ストアへの接続が完全ではありません。別名は ODBM 内ではアクティブですが、IMS Connect 内ではアクティブになっていません。 <b>X'40'</b> XIBOD_IINACTIVE: データ・ストアへの接続が完全ではありません。別名は ODBM および IMS Connect の両方でアクティブでしたが、ODBM ではもうアクティブではありません。このため、別名が IMS Connect 内でアクティブではなくなりました。 <b>X'20'</b> XIBOD_ICACTIVE: データ・ストアへの接続が完全です。別名は ODBM 内および IMS Connect 内の両方でアクティブです。 <b>X'10'</b> XIBOD_ICINACTIVE: データ・ストアへの接続が完全ではありません。別名は ODBM および IMS Connect の両方でアクティブでしたが、IMS Connect ではもうアクティブではありません。ODBM では、別名はまだアクティブです。 <b>X'08'</b> XIBOD_DELETED: IMS 別名は ODBM CSLDCxxx 構成メンバーから削除されており、使用できなくなりました。
予約済み	5	X'05'	3	IMS Connect 用に予約済み。
予約済み	8	X'08'	8	IMS Connect 用に予約済み。

### 関連資料

[IMS Connect UPDATE ODBM コマンド \(コマンド\)](#)

[STARTOD コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(ODBM\) コマンド \(コマンド\)](#)



---

# 第 12 章 IMS Connect による IMSplex および共用キューのサポート

IMS Connect は、IMS Structured Call Interface (SCI) を使用することによって、IMS Operations Manager (OM) コマンドおよび応答ストリング・メッセージを TCP/IP クライアントと IMSplex 内の OM との間で送受信できます。

## このタスクについて

IMS Connect は、IMS Structured Call Interface (SCI) を使用することによって、IMS Operations Manager (OM) コマンドおよび応答ストリング・メッセージを TCP/IP クライアントと IMSplex 内の OM との間で送受信できます。IBM Management Console for IMS and Db2 for z/OS は、このインターフェースを使用する IBM 提供のクライアントです。

以下のサブセクションでは、環境要件に関する詳細情報、および OM をサポートするように IMS Connect をセットアップする方法について説明します。

---

## IMS Connect による IMSplex のサポート

IMSplex の IMS Connect サポートにより、TCP/IP を使用して IBM Management Console for IMS and Db2 for z/OS (Management Console) または他の OM コマンド・クライアントから IMSplex 内の OM へのアクセスが可能になります。

IMSplex サポートは、IMS SCI を介して OM をアクセスします。IMS Connect 構成ファイル (HWSCFGxx) の IMSplex ステートメントは、IMS Connect for IMSplex サポートを定義します。IMSplex ステートメントを省略すると、IMSplex サポートは使用できません。

IMSplex サポートは、IMS コマンド・ストリング・メッセージをクライアント (例えば、Management Console) のために IMSplex 内の特定の OM に直接に送信します。IMS Connect が Management Console コマンド・メッセージを受信するように、1 つ以上の IMSplex を定義できます。SCI は、IMS Connect と IMSplex との間で通信するのに使用されます。選択した OM にアクセスするために、IMS.PROCLIB データ・セット内の IMS Connect 構成メンバーに IMSPLEX 構成ステートメントを組み込みます。

データ・ストアへのアクセスに使用されるのと同じセキュリティー方式 (ユーザー ID、グループ ID、およびパスワードの認証) が、IMSplex サポートにも適用されます。IMS Connect コマンド・メッセージ出口は、IMS Connect ユーザー・メッセージ出口と同じ機能を実行します。

OM コマンド・クライアント 出口ルーチン (HWSCSLO0 および HWSCSLO1) は、IMSplex サポート専用設計されています。HWSCSLO0 および HWSCSLO1 出口ルーチンは、データ・ストアへのクライアント・アクセス用に IMS Connect によって提供されるユーザー・メッセージ出口と似ています。HWSCSLO0 および HWSCSLO1 メッセージ出口は、OM コマンド・クライアントでのみ使用するように設計され、データ・ストアにメッセージを送信するクライアントでは使用できません。HWSCSLO0 出口ルーチンおよび HWSCSLO1 出口ルーチンは、OM コマンド・クライアントからのコマンド・ストリング・メッセージのみを処理し、オブジェクト・コードとしてのみ配信されます。

---

## IMSplex サポート環境

IMS Connect が Operations Manager (OM) と通信するためには、IMS、CSL OM、および CSL Structured Call Interface (SCI) が稼働している必要があります。

IMS は、IMS Connect と同じ z/OS イメージ内で稼働しているか、同じ IMSplex 上の別の z/OS イメージで稼働している必要があります。OM は、同じ z/OS イメージに存在するか、同じ IMSplex 内の別の z/OS イメージに存在する必要があります。SCI は、IMS Connect と同じ z/OS イメージ内に存在する必要があります。

IMS Connect の立ち上げは、IMS、SCI、OM、および RM の立ち上げの前と後のどちらでも構いません。IMS Connect の初期設定時に、SCI への接続が行われます。IMS Connect は、SCI への接続を 30 分間試み

ます。SCI 接続が行われなかった場合、SCI が初期設定された後で、**OPENIP** コマンドを発行して SCI に接続する必要があります。SCI が正常終了しても、異常終了しても、IMS Connect は、SCI が再始動したときに自動的に SCI に再接続します。

## 関連概念

IMS システム管理の考慮事項とタスク (システム管理)

# IMS Connect による IMSplex のサポートのインストール

IMS Connect による IMSplex のサポートには、CSL Operations Manager (OM)、CSL Structured Call Interface (SCI)、および TCP/IP が必要です。

## このタスクについて

IMS Connect IMSplex サポートを使用するためには、IMS がインストールされていなければなりません。IMS.SDFSRESL データ・セットに STEPLIB が追加され、IMS Connect IMSplex サポートのために SCI にアクセスできることを確認してください。

OM では、適切な IMSplex コンポーネント (IMS、IMS Connect、RM、または ODBM) が OM コマンド機能を提供するために稼働していることが必要です。IMS Connect IMSplex サポートでは、これらのコンポーネントが稼働している必要はありません。ただし、これらのコンポーネントが稼働していない場合、コマンドはいずれも処理できません。

IMS Connect による IMSplex サポート用の以下のコンポーネントを、以下の順序でインストールまたは変更します。

## 手順

1. IMS Connect 構成ファイル:
  - a. IMSplex ステートメントを追加する。
  - b. HWSCSLOO 出口を TCPIP ステートメントの EXIT= パラメーターに追加する。
2. BPE 構成ファイル:
  - a. IMS Connect トレース項目が別々にリストされている場合、OMDR ステートメントおよび HWSO ステートメントを追加する。
3. OM コマンド・クライアントは次のことを行います。
  - a. IMS Connect HWS ID= 値を確認する。
  - b. IMS Connect IMSPLEX tmember= 値を確認する。

## タスクの結果

インストールが完了したら、順番に以下の機能を開始します。

1. SCI を開始します。
2. OM を開始します。
3. RM を使用する場合は、RM を開始します。
4. IMS を開始します。

上記のステップの前、処理中、後のいつでも、IMS Connect を始動できます。IMS Connect は、SCI への接続を 30 分間試みます。SCI が立ち上がっていないかまたは接続の試みが失敗した場合は、**OPENIP implex\_name** コマンドを発行する必要があります。

IMSplex サポートによって、以下の IMS Connect コマンドが使用可能になります。

- STOPIP
- OPENIP
- VIEWIP

## 共用キュー環境での ALTPCB 出力のリトリブ

---

共用キュー環境では、IMS Connect が直接接続されていない IMS システムによって ALTPCB 出力キューに挿入されたメッセージをリトリブするには、OTMA スーパー・メンバー関数を使用します。

### このタスクについて

複数の IMS システムを含む構成内で OTMA スーパーメンバーを使用するには、それらすべての IMS システムで共用キューを使用可能にしておく必要があります。バックエンド IMS システムがスーパーメンバー対応環境で ALT-PCB 出力を作成した場合は、スーパーメンバー・セット内の OTMA クライアントを使用する任意のフロントエンド IMS からその出力をリトリブすることができます。

スーパーメンバー関数をアクティブにするには、以下の手順を実行します。

### 手順

IMS Connect 構成メンバー (HWSCFGxx) 内の HWS ステートメントまたは DATASTORE ステートメントのどちらかの SMEMBER パラメーターに、1 文字から 4 文字までのスーパーメンバー名を指定します。

スーパーメンバー名は固有でなければならず、既存の OTMA メンバー名と同じ名前にはできません。





## 第 13 章 IMS Connect セキュリティー・サポート

IMS Connect には、さまざまなオプションが組み込まれており、それらのオプションによって、メッセージが IMS Connect に届いたとき、および、IMS TM 接続の場合はデータ・ストアに届いたときに、そのメッセージに対して実行するセキュリティー検査を実装および変更できます。

IMS TM への接続のために、IMS Connect は、IMS Connect 内のセキュリティー検査用の 2 つのオプションを提供します。これらのオプションによって、IMS Connect から RACF を直接呼び出すように構成するか、あるいは IMS Connect のユーザー・メッセージ出口ルーチンからセキュリティー・ユーザー出口ルーチンと呼び出すことができます。

IMS DB への接続では、IMS Connect DB セキュリティー・ユーザー出口ルーチン (HWSAUTH0) がユーザーの認証を行い、さらに RACF も使用することができます。IMS DB への接続では、IMS Connect はいかなるアクションを実行するユーザーの権限も検査しませんが、許可の目的で CSL Open Database Manager に RACO トークンを渡すことがあります。

IMS 間 TCP/IP 通信をサポートする、IMS Connect のリモート・インスタンスへの接続の場合、IMS Connect は RACF パスチケットおよびトラステッド・ユーザー接続状況の確立をサポートします。

IMS Connect が提供する追加セキュリティー機能には、以下のものがあります。

- パスワード管理サポート
- データ・ストアに到着するメッセージ用のトラステッド・ユーザー種別
- OTMA アクセサー環境エレメント (ACEE) タイムアウト指定サポート
- IMS DB に接続するクライアントからの接続の場合:
  - Secure Sockets Layer (SSL) サポート。IMS DB に接続するクライアントからの接続を、SSL を使用して保護するには、IBM z/OS Communications Server Application Transparent Transport Layer Security 機能 (AT-TLS) を使用できます。
  - RACF パスチケットのサポート
- IMS TM クライアントに対しては、IMS Connect は RACF パスチケットのサポートを提供します。
- 分散環境において、ユーザーが入力したセキュリティー資格情報 (ネットワーク・ユーザー ID およびネットワーク・セッション ID を含む) の IMS との間での受け渡しをサポートします。

### IMS Connect の RACF サポート

IMS Connect は、RACF を直接呼び出すように、および、RACF パスチケットをサポートするように構成することができます。

デフォルトでは、IMS Connect は RACF を呼び出しません。IMS Connect が RACF を呼び出すように構成されている場合、IMS Connect は RACF を使用して、着信メッセージのユーザー ID とパスワードを直接検証することができます。

直接 RACF をサポートするように構成すると、IMS Connect は RACF パスチケットもサポートします。

RACF が大/小文字混合パスワードをサポートするように構成されている場合、IMS Connect も大/小文字混合パスワードをサポートするように構成できます。

IMS Connect は、RACF コマンド **RACROUTE REQUEST=VERIFY** を発行して RACF を呼び出し、着信メッセージの IRM でクライアントから受け取ったユーザー ID およびパスワードを検査します。IMS 用のデフォルトの RACF ID も定義できます。

RACF セキュリティー障害が発生した場合、IMS Connect は、診断の目的で、**RACROUTE REQUEST=VERIFY** コマンドからの戻りコードを要求状況メッセージ (RSM) に含めます。

RACF が IMS Connect クライアントからのサインオンの検証に使用される場合、指定されたユーザー ID またはパスワードが無効であれば、実際の RACF 戻りコードまたは IMS 戻りコードではなく、汎用の戻りコードまたはメッセージが IMS Connect によって返されるようにすることができます。汎用の戻りコードま

たはメッセージが返されるようにすることで、有効なユーザー ID およびパスワードが指定されるまで、RACF によって検査されたサインオンに関する情報へのアクセスを禁止できます。

IMS Connect から RACF を呼び出すように構成する場合、RACF 呼び出しが IMS Connect のパフォーマンスに与える影響を評価してください。パフォーマンスを向上させるために、RACF ユーザー ID キャッシュを使用可能にすることを検討してください。

### 関連タスク

[198 ページの『IMS Connect パスワード管理』](#)

IMS Connect は、RACF パスワードの管理に役立ついくつかの機能を提供しています。これらの機能の中には、IMS Connect が RACF を直接呼び出すように構成されている場合にのみ適用されるものもあります。

### 関連資料

[191 ページの『HWSSMPL0 および HWSSMPL1 セキュリティー処置』](#)

サンプル・ユーザー・メッセージ出口 HWSSMPL0 および HWSSMPL1 は、特定のセキュリティー処置を常に実行し、IMSLSECX セキュリティー出口が呼び出された場合または呼び出されない場合にのみ他のセキュリティー処置を実行します。

## RACF 検証用の汎用戻りコードまたはメッセージの有効化

RACF が IMS Connect クライアントからのサインオンの検証に使用される場合、指定されたユーザー ID またはパスワードが無効であれば、実際の RACF 戻りコードまたは IMS 戻りコードではなく、汎用の戻りコードまたはメッセージが IMS Connect によって返されるようにすることができます。汎用の戻りコードまたはメッセージが返されるようにすることで、有効なユーザー ID およびパスワードが指定されるまで、RACF によって検査されたサインオンに関する情報へのアクセスを禁止できます。

### 始める前に

実際の RACF 戻りコードまたは IMS 戻りコードではなく、汎用の戻りコードまたはメッセージが IMS Connect によって返されるようにするには、その前に必ず、RACF から提供される IRRSPW00 モジュールを以下のいずれかの場所に組み込んでください。

- LPA
- LINKLIST 内のライブラリー
- LINKLIST 内のライブラリー

### 手順

実際の RACF 戻りコードまたは IMS 戻りコードではなく、汎用の戻りコードまたはメッセージが IMS Connect によって返されるようにするには、以下のいずれかの方法を使用します。

- IMS PROCLIB データ・セットの HWSCFGxx メンバーの HWS ステートメントで、**RACFGENRC=Y** を指定します。
- **SET(RACFGENRC(ON))** を指定した **UPDATE IMSCON TYPE(CONFIG)** コマンドを発行します。

### 関連資料

[HWS ステートメント \(システム定義\)](#)

[QUERY IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

## IMS Connect 内の RACF セキュリティー検査の使用可能化

HWS 構成ステートメントで RACF=Y を指定するか、またはオンライン IMS Connect コマンドを発行することによって、IMS Connect 内の RACF セキュリティー検査を使用可能にできます。

### このタスクについて

HWS 構成ステートメントで RACF を使用可能にするには、RACF=Y パラメーターを追加します。以下に例を示します。

```
HWS ID=HWS01 RACF=Y
```

オンラインで RACF を使用可能にするには、以下のようにします。

### 手順

- 以下のいずれかの IMS Connect コマンドを発行します。
  - IMS タイプ 2 フォーマット・コマンド、**UPDATE IMSCON TYPE(CONFIG) SET(RACF(ON))**
  - WTOR フォーマット・コマンド、**SETRACF ON**
  - z/OS MODIFY コマンド・フォーマット、**UPDATE MEMBER TYPE(IMSCON) SET(RACF(ON))**
- 以下のいずれかの IMS Connect コマンドを発行して、RACF セキュリティー検査が使用可能であることを確認します。
  - IMS タイプ 2 フォーマット・コマンド、**QUERY IMSCON TYPE(CONFIG) SHOW(RACF)**
  - WTOR フォーマット・コマンド、**VIEWHWS**
  - z/OS MODIFY コマンド・フォーマット、**QUERY MEMBER TYPE(IMSCON) SHOW(ALL)**

### タスクの結果

RACF が使用可能な場合、IMS タイプ 2 フォーマット・コマンド **QUERY IMSCON TYPE(CONFIG) SHOW(RACF)** からの出力には、コマンド出力の「Racf」列に「Y」が表示されます。

WTOR コマンドおよび z/OS MODIFY コマンドの両方からの出力に、HWSC0001 HWSID=HW01 RACF=Y と同様の行が含まれます。

### 関連資料

[IMS Connect UPDATE MEMBER コマンド \(コマンド\)](#)

[SETRACF コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[QUERY IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[HWS ステートメント \(システム定義\)](#)

[IMS Connect QUERY MEMBER コマンド \(コマンド\)](#)

[VIEWHWS コマンド \(コマンド\)](#)

## IMS Connect に対する RACF セキュリティー統計の使用可能化

IMS Connect が RACF を呼び出すように構成されている場合、IMS Connect が RACF 呼び出し **RACROUTE REQUEST=VERIFY** を発行するときに、RACF セキュリティー統計が記録され、更新されるようにすることができます。IMS DB への ODBM クライアント接続および IMS TM への OTMA クライアント接続の RACF 統計が記録され、更新されるようにすることができます。RACF 統計を使用可能にした後は、統計は 1 日 1 回まで更新されます。

### 始める前に

IMS Connect が RACF 呼び出し **RACROUTE REQUEST=VERIFY** を発行するときに RACF セキュリティー統計が記録されるようにするには、その前に必ず **HWSCFGxxIMS PROCLIB** メンバーの HWS ステートメント

で RACF=Y を指定してください。HWS ステートメントで RACF=N が使用されている場合、IMS Connect は RACF 呼び出し **RACROUTE REQUEST=VERIFY** を発行しないため、RACF 統計は記録されません。

## 手順

IMS DB への ODBM クライアント 接続 の RACF 統計を使用可能にするには、以下のいずれかの方法を使用します。

- IMS PROCLIB データ・セットの HWSCFGxx メンバーの ODACCESS ステートメントで RACF 統計を使用可能にするには、ODRACFST=Y を指定します。
- RACF 統計オプションをオンラインで更新して RACF 統計を使用可能にするには、**UPDATE IMSCON TYPE(CONFIG)** コマンドで ODRACFST(ON) キーワードを使用します。

IMS TM への OTMA クライアント 接続 の RACF 統計を使用可能にするには、以下のいずれかの方法を使用します。

- IMS PROCLIB データ・セットの HWSCFGxx メンバーの HWS ステートメントで RACF 統計を使用可能にするには、TMRACFST=Y を指定します。
- RACF 統計オプションをオンラインで更新して RACF 統計を使用可能にするには、**UPDATE IMSCON TYPE(CONFIG)** コマンドで TMRACFST(ON) キーワードを使用します。

## タスクの結果

RACF 統計を使用可能にすると、IMS Connect は、**RACROUTE REQUEST=VERIFY** 呼び出しで STAT=ASIS パラメーターを使用します。STAT=ASIS を使用した場合、RACF メッセージおよび統計は、インストール済み環境における RACF **SETROPTS** コマンド上の現行のオプションによって制御されます。

ラクトール 統計を使用可能にした後、この統計は、ラクトールによって、1日に1回以下のシステム管理機能 (SMF) データ・セットまたはログ・ストリームに記録されます。ラクトール 統計を記録するために使用される SMF データ・セットまたはログ・ストリームは、ラクトール 構成に指定されます。

### 関連資料

[ODACCESS ステートメント \(システム定義\)](#)

[QUERY IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

## IMS Connect のデフォルトの RACF ユーザー ID

IMS Connect 用のデフォルトの RACF ユーザー ID を設定して、入力メッセージの IRM に IRM\_RACF\_USERID フィールドが含まれていない場合、または IRM\_RACF\_USERID フィールドがブランクの場合に、このユーザー ID を使用できます。

### このタスクについて

デフォルトの RACF ユーザー ID が使用される場合、IMS Connect は入力メッセージの OMSECUID フィールドにあるそのユーザー ID を OTMA に渡します。OTMA セキュリティー 検査が使用可能な場合は、OTMA はその RACF ユーザー ID を使用して、RACF によるコマンド、トランザクション、および RESUME TPIPE 呼び出しを許可します。

デフォルトの RACF ユーザー ID が定義されており、さらに着信メッセージの IRM\_RACF\_USERID フィールドもブランクでない場合は、IMS Connect は IRM\_RACF\_USERID フィールドに定義されているユーザー ID を使用します。

デフォルトの RACF ユーザー ID は、IMS Connect TCPIP 構成ステートメントの RACFID パラメーターで定義します。

## IMS Connect RACF ユーザー ID キャッシュ

トランザクションごとに RACF 要求を発行する代わりに、RACF ユーザー ID 用のメモリー・キャッシュを使用するように IMS Connect を構成することができます。

IMS Connect が RACF セキュリティーを使用するように構成されている場合、メッセージを処理する前に、各メッセージで提供される RACF ユーザー ID を検証する必要があります。この RACF セキュリティー認証方式は、IMS Connect のパフォーマンスに悪影響を与えることがあります。RACF サポートを使用不可にせずパフォーマンスを改善するには、IMS Connect RACF ユーザー ID キャッシュを使用可能にすることができます。キャッシュには、すべてのセッションからの以前に検証済みの RACF ユーザー ID が保管されます。可能な場合、キャッシュされた情報が OTMA に渡され、新たに RACF 検証要求を出す必要はありません。

システム定義時に、PROCLIB データ・セット (HWSCFGxx) の IMS Connect メンバーの TCP/IP ステートメントを使用して、RACF ユーザー ID キャッシュの設定を構成します。具体的には、キャッシュの設定は、RACF パラメーター、UIDCACHE パラメーター、および UIDAGE パラメーターを使用して定義します。RACF パラメーターは、IMS Connect が RACF 認証を使用するかどうかを定義します。UIDCACHE パラメーターは、RACF ユーザー ID キャッシュが使用可能かどうかを定義します。UIDAGE パラメーターは、キャッシュされた ID のデフォルトのリフレッシュ間隔を指定します。

また、IMS Connect の稼働中に、以下のいずれかのコマンドを使用して、キャッシュを使用可能または使用不可にすることもできます。

- WTOR コマンド **SETUIDC**
- z/OS Modify コマンド **UPDATE MEMBER**
- タイプ 2 コマンド **UPDATE IMSCON TYPE(CONFIG) SET(UIDCACHE(ON | OFF))**

IMS Connect は、実行中、キャッシュされたユーザー ID に関連付けられた RACF イベント通知機能を (ENF) イベントを自動的にモニターします。RACF ENF が、RACF の CONNECT コマンドまたは REMOVE コマンドに対して、あるいは ALTUSER REVOKE コマンドに対してタイプ 71 イベントを発行した場合、IMS Connect は自動的にユーザー ID を最新表示します。IMS Connect は、キャッシュ内の指定された ID を自動的に最新表示した後、イベント番号 258 を発行します。

また、次のコマンドを使用して、特定のユーザー ID を手動で最新表示することもできます。

- WTOR コマンド **REFRESH RACFUID**
- z/OS Modify コマンド **UPDATE RACFUID**
- タイプ 2 コマンド **UPDATE IMSCON TYPE(RACFUID) NAME(userid) OPTION(REFRESH)**

### 関連資料

[IMS Connect UPDATE RACFUID コマンド \(コマンド\)](#)

[IMS Connect UPDATE MEMBER コマンド \(コマンド\)](#)

[REFRESH RACFUID コマンド \(コマンド\)](#)

[SETUIDC コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(RACFUID\) コマンド \(コマンド\)](#)

[イベント・タイプ \(出口ルーチン\)](#)

## IMS DB のクライアントに対する IMS Connect のセキュリティ

ODBM を介して IMS DB に接続するクライアント (Distributed Relational Database Architecture™ (DRDA) を使用する IMS Universal ドライバー およびクライアントなど) の場合、IMS Connect はユーザーの認証を行います。いかなるアクションを実行するユーザーの権限も検査しません。

IMS DB クライアントのユーザー ID を認証する場合、IMS Connect は IMS Connect DB セキュリティー・ユーザー出口ルーチン (HWSAUTH0) か、RACF などのセキュリティ製品のいずれか一方、またはその両方を使用することができます。IMS DB クライアントに対しては、IMS Connect は RACF パスチケットのサポートも提供します。



IMS Connect は、RACF または別のセキュリティー製品が使用可能かどうかにかかわらず、常に HWSAUTH0 ユーザー出口を呼び出します。ご使用の IMS Connect 構成に RACF サポートが組み込まれている場合、IMS Connect は HWSAUTH0 ユーザー出口を呼び出してから RACF を呼び出します。

IMS Connect が RACF を呼び出すように構成されている場合、IMS DB に対して ODBM クライアント接続を認証するために IMS Connect が RACF 呼び出し **RACF RACROUTE REQUEST=VERIFY** を発行するときに、RACF セキュリティー統計が記録されるようにすることができます。ODACCESS ステートメントで ODRACFST=Y を指定するか、オンライン IMS Connect コマンド **UPDATE IMSCON TYPE(CONFIG)** を発行することで、RACF 統計を使用可能にすることができます。RACF 統計を使用可能にした後、統計は、RACF によってシステム管理機能 (SMF) データ・セットまたはログ・ストリームに 1 日 1 回まで記録されます。RACF 統計の記録に使用される SMF データ・セットまたはログ・ストリームは、RACF 構成で指定されま

す。

HWSAUTH0 ユーザー出口ルーチンは入力されたユーザー ID を異なるユーザー ID でオーバーライドすることができ、また IMS Connect によって追加で認証される RACF グループ ID を提供できます。

HWSAUTH0 ユーザー出口ルーチンは BPE タイプ 1 のユーザー出口ルーチンであり、リフレッシュ可能です。

IMS Connect は、IMS DB に接続するクライアントに対し、Secure Sockets Layer (SSL) を直接にはサポートしません。SSL を使用して IMS DB への接続を保護するには、IBM z/OS Communications Server Application Transparent Transport Layer Security 機能 (AT-TLS) を使用します。AT-TLS の使用は、IMS Connect に対して透過的です。

## 関連概念

[IMS Connect の定義と調整 \(システム定義\)](#)

## 関連タスク

[183 ページの『IMS Connect 内の RACF セキュリティー検査の使用可能化』](#)

HWS 構成ステートメントで RACF=Y を指定するか、またはオンライン IMS Connect コマンドを発行することによって、IMS Connect 内の RACF セキュリティー検査を使用可能にできます。

[183 ページの『IMS Connect に対する RACF セキュリティー統計の使用可能化』](#)

IMS Connect が RACF を呼び出すように構成されている場合、IMS Connect が RACF 呼び出し **RACROUTE REQUEST=VERIFY** を発行するときに、RACF セキュリティー統計が記録され、更新されるようにすることができます。IMS DB への ODBM クライアント接続および IMS TM への OTMA クライアント接続の RACF 統計が記録され、更新されるようにすることができます。RACF 統計を使用可能にした後は、統計は 1 日 1 回まで更新されます。

[202 ページの『IMS DB との IMS Connect クライアント接続用の RACF パスチケット』](#)

RACF パスチケットを使用して、IMS DB との IMS Connect クライアント接続を認証することができます。パスチケットは、RACF パスワードおよびパスワード・フレーズの代わりとして使用でき、ネットワーク経由でパスワードおよびパスワード・フレーズを暗号化されていないテキストとして送信しなくてもすむようになるため、セキュリティーを強化します。

## 関連資料

[UPDATE IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[ODACCESS ステートメント \(システム定義\)](#)

# IMS Connect によるネットワーク・セキュリティー資格情報の受け渡し

分散ネットワーク環境でアプリケーションからセキュリティー資格情報が入力され、アプリケーションがユーザー・メッセージ出口ルーチン HWSSMPL0、HWSSMPL1、または HWSJAVA0 を使用する場合、IMS Connect を介して IMS に資格情報を渡すようにすることができます。また、IMS から IMS Connect を介して IMS コールアウト要求で分散ネットワーク・セキュリティー資格情報を渡すこともできます。

ネットワーク・セキュリティー資格情報は、OTMA メッセージ接頭部のセキュリティー・データ・セクションに格納され、IMS Connect から IMS に送信されます。その後、ネットワーク・ユーザー ID とネットワーク・セッション ID が含まれるネットワーク・セキュリティー資格情報を、OTMA メッセージ接頭部がに含まれる IMS ログ・レコード (X'01' および X'03' など) に含めることができます。IMS Connect が、IMS

DL/I インターフェースの ICAL 呼び出しによって開始された同期コールアウト・メッセージ内の分散ネットワーク・セキュリティ資格情報を、RESUME TPIPE 呼び出しを発行するアプリケーションに渡せるようにする場合、セキュリティ資格情報も OTMA 接頭部のセキュリティ・データ・セクションで渡されません。

**制約事項:** DataPower、IMS Connect API、および SOAP Gateway クライアントからの分散ネットワーク・セキュリティ資格情報は、IMS Connect によってサポートされません。

## ユーザー・メッセージ出口 HWSSMPL0 または HWSSMPL1 を使用するユーザー作成の IMS Connect クライアント・アプリケーションからの分散ネットワーク・セキュリティ資格情報の引き渡し

ユーザー・メッセージ出口 HWSSMPL0 または HWSSMPL1 を使用するユーザー作成の IMS Connect クライアント・アプリケーションから分散ネットワーク・セキュリティ資格情報を渡すには、IMS 要求メッセージ (IRM) ヘッダーで IRM 拡張を使用します。ネットワーク・セッション ID が含まれる IRM 拡張には ID \*NETSID\* を指定し、ネットワーク・ユーザー ID が含まれる IRM 拡張には ID \*NETUID\* を指定します。

\*NETSID\* 拡張または \*NETUID\* 拡張 (あるいはその両方) が含まれるメッセージがユーザー・メッセージ出口 HWSSMPL0 または HWSSMPL1 に渡された後、ユーザー・メッセージ出口は、ネットワーク・セキュリティ資格情報が含まれるように OTMA メッセージ接頭部を作成します。

**推奨事項:** ネットワーク・セキュリティ資格情報が IMS Connect クライアントの入力メッセージに含まれている場合は、IMS Connect レコーダー・トレース機能に対して BPE 外部トレース機能を使用可能にしてください。ネットワーク・セキュリティ資格情報が IMS Connect に渡される場合、IMS Connect との間でやり取りされる入出力メッセージのサイズが 670 バイトを超えることがあり、メッセージ全体のデータを取り込むには BPE 外部トレース機能が必要になります。

## RESUME TPIPE を呼び出しを発行するアプリケーションへの分散ネットワーク・セキュリティ資格情報の引き渡し

IMS Connect が、IMS DL/I インターフェースの ICAL 呼び出しによって開始された同期コールアウト・メッセージ内の分散ネットワーク・セキュリティ資格情報を、RESUME TPIPE 呼び出しを発行するアプリケーションに渡せるようにするには、IRM 接頭部の以下のフィールドを指定して RESUME TPIPE 呼び出しを定義します。以下のフィールド指定が定義されていない場合、IMS は、コールアウト要求における OTMA メッセージ接頭部のセキュリティ・データ・セクションから分散ネットワーク・セキュリティ資格情報を除去します。

### IRM\_ARCH

X'05' (IRM\_ARCH5)

### IRM\_F6

X'80' (IRM\_F6\_NWSE)

## IMS TM リソース・アダプターのクライアント・アプリケーションからの分散ネットワーク・セキュリティ資格情報の引き渡し

IMS TM リソース・アダプター が、HWSJAVA0 ユーザー・メッセージ出口ルーチンを使用する Java EE アプリケーションから IMS にネットワーク・セキュリティ資格情報を渡せるようにするには、アプリケーションを構成し、IMS TM リソース・アダプター に付属の Java 認証・承認サービス (JAAS) ログイン・モジュールにリンクする必要があります。アプリケーションを JAAS ログイン・モジュールにリンクした後、ユーザーは、外部ユーザー・アカウント・レジストリーによる認証のために IMS トランザクションを呼び出す際に、セキュリティ資格情報を入力する必要があります。外部ユーザー・アカウント・レジストリーは、WebSphere Application Server または WebSphere Liberty によってサポートされるいずれかのユーザー・アカウント・レジストリー (LDAP サーバーなど) です。資格情報が正常に認証されると、IMS TM リソース・アダプター は、OTMA メッセージ接頭部のセキュリティ・データ・セクションを使用して分散資格情報を IMS Connect に送信します。

また、IMS 従属領域で実行される IMS アプリケーションが外部 Java EE アプリケーションに対して同期または非同期のコールアウト要求を行う場合に、IMS TM リソース・アダプター がネットワーク・セキュリティ資格情報をサポートできるようにすることもできます。



## 同期コールアウト・メッセージでのネットワーク・セキュリティ資格情報

IMS TM リソース・アダプター が同期コールアウト・メッセージでのネットワーク・セキュリティ資格情報をサポートできるようにするには、IMSActivationSpec オブジェクトの resumeTpipeNsc プロパティを true に設定します。

## 非同期コールアウト・メッセージでのネットワーク・セキュリティ資格情報

IMS TM リソース・アダプター が非同期コールアウト・メッセージでのネットワーク・セキュリティ資格情報をサポートできるようにするには、IMSInteractionSpec オブジェクトに対して setResumeTpipeNSC(int resumeTpipeNSC) メソッドを呼び出し、setResumeTpipeNSC プロパティの値を 1 に設定する必要があります。setResumeTpipeNSC プロパティに 1 を設定した場合、IMS TM リソース・アダプター は、IMS に送信される OTMA メッセージ接頭部のフラグ・バイトを設定し、ネットワーク・セキュリティ資格情報がコールアウト・メッセージに含まれるように指示します。

## 関連概念

### [332 ページの『RESUME TPIPE/受信プロトコル』](#)

IMS Connect クライアントは RESUME TPIPE プロトコルを使用して、IMS の T パイプ保留キューからコミット後送信 (CMO) 出力または同期コールアウト要求をリトリブします。

## 関連タスク

### [212 ページの『RESUME TPIPE による同期コールアウト要求のリトリブ』](#)

RESUME TPIPE 呼び出しを発行して、同期コールアウト要求をリトリブする場合、同期コールアウト・メッセージのみ、または同期コールアウト・メッセージと非同期出力の両方をリトリブするように、RESUME TPIPE 呼び出しをコーディングすることができます。

## 関連資料

### [258 ページの『返されるネットワーク・セキュリティ・セグメントのフォーマット』](#)

ネットワーク・セッション ID (NETSID) とネットワーク・ユーザー ID (NETUID) のセグメントには、ネットワーク・セキュリティ情報が含まれており、クライアントが RESUME TPIPE 呼び出しを発行するとクライアントに返されます。

### [237 ページの『IRM 拡張のフォーマット』](#)

IMS 要求メッセージ (IRM) 拡張を使用すると、IRM をマップする DSECT を拡張することなく、IMS Connect クライアント・アプリケーションから情報を送信することができます。

### [916 ページの『OTMA セキュリティ・データ・フィールドの説明』](#)

以下に、メッセージ接頭語のセキュリティ・データ・セクションの内容について、さらに詳しい情報を提供します。

## 関連情報

[ネットワーク・セキュリティ資格情報および IMS TM リソース・アダプター](#)

[分散ネットワーク・セキュリティ資格情報に対応した Websphere Application Server の構成](#)

[分散ネットワーク・セキュリティ資格情報に対応した Websphere Liberty の構成](#)

[IMS TM リソース・アダプターによるコールアウト・メッセージでのネットワーク・セキュリティ資格情報のサポートの有効化](#)

## IMS 間 TCP/IP 接続の保護

IMS 間 TCP/IP 接続を保護するために、IMS Connect は RACF パスチケットを使用して、IMS Connect の 1 つのインスタンスを IMS Connect のもう 1 つ別のインスタンスのトラステッド・ユーザーとして設定します。

## このタスクについて

接続が最初に確立されると、メッセージを送信する IMS Connect のインスタンスは RACF パスチケットを生成して、メッセージを受信する IMS Connect のインスタンスに渡します。受信側 IMS Connect インスタンスが RACF を使用してパスチケットを正常に検証した後は、その接続上で受信されるメッセージはすべてトラステッド・ユーザーからのものと見なされ、追加のセキュリティ検査を受けません。

送信側 IMS Connect インスタンスは、RMTIMSCON ステートメントの APPL パラメーターと USERID パラメーターで提供される値から RACF パスチケットを生成します。

受信側 IMS Connect インスタンスは、RACF を呼び出してユーザー ID を認証し、送信側 IMS Connect インスタンスが送信したパスチケット、アプリケーション名、およびユーザー ID を使用して、アプリケーションへのアクセス権限を確認します。

#### 推奨事項:

- 受信側 IMS Connect インスタンスで RACF が使用可能になっていない場合は、送信側 IMS Connect インスタンスでパスチケットを生成するように構成しないでください。RACF=N の場合、受信側 IMS Connect インスタンスはセキュリティー検査を実行せず、送信されたパスチケット・データを無視します。送信側でパスチケットを作成するのは処理リソースの無駄になります。
- 非持続接続では RACF パスチケットを使用しないでください。パスチケットを使用すると、大きな処理上のオーバーヘッドが生じるからです。新しい接続が確立されるたびに、新しいパスチケットが生成されて送信されます。

IMS Connect は、MSC 通信および OTMA 通信の両方に対して、RACF パスチケットのセキュリティーをサポートします。

MSC 通信の場合、IMS Connect の各インスタンスがトランザクション・メッセージと応答を送受信することができます。MSC の IMS 間 TCP/IP 接続をセキュアにするには、RACF サポートを有効にし、両方の IMS Connect インスタンスでアプリケーション名およびユーザー ID を定義する必要があります。一方のインスタンスに定義されたアプリケーション名およびユーザー ID は、他方のインスタンスに定義されたアプリケーション名およびユーザー ID と異なっていても構いません。さらに、両方の z/OS インストール済み環境にある RACF 内に、パスチケット・クラス、アプリケーション名、およびユーザー ID も作成する必要があります。

IMS Connect の 2 つのインスタンス間の IMS 間 TCP/IP 接続をセキュアにするには、以下の手順を実行します。

#### 手順

1. 送信側 IMS Connect インスタンスの場合、IMS.PROCLIB データ・セット内の IMS Connect 構成メンバーで、以下のようにセキュリティー設定を指定します。
  - RMTIMSCON ステートメントで、以下のパラメーターを使用してアプリケーション名とユーザー ID を指定します。
    - APPL
    - USERID
  - 接続が MSC 用に使用される場合は、HWS ステートメントで RACF=Y を指定して、必要な戻りの接続に対するセキュリティー検査を有効にします。
2. 送信側 IMS システムのセキュリティー管理者は、IMS Connect によるパスチケットの生成をサポートするために、RACF パスチケット・クラス項目を作成する必要があります。

送信側システムで定義されたパスチケット・クラスは、リモート・システムで定義されたパスチケット・クラス項目と同じキーを持っていないければなりません。

次の例は、RACF に対してパスチケット・クラスを定義します。

```
SETROPTS CLASSACT (PTKTDATA)
SETROPTS RACLIST (PTKTDATA)
RDEF PTKTDATA APPLI2I SSIGNON (KEYMASKED (E001193519561977)) UACC(N)
SETROPTS REFRESH RACLIST (PTKTDATA)
```

3. 受信側 IMS Connect インスタンスの場合、以下のように IMS.PROCLIB データ・セット内の構成メンバーでセキュリティー設定を指定します。
  - HWS ステートメントで、RACF=Y を指定します。
  - 接続が MSC 用に使用される場合、戻りの接続に必要なセキュリティー・データを提供するために、RMTIMSCON の以下のパラメーターで、アプリケーション名とユーザー ID を指定します。
    - APPL
    - USERID

4. 受信側システムのセキュリティー管理者は、RACF に対してユーザー ID、アプリケーション名、およびパスチケット・クラスを定義する必要があります。

パスチケットのキーは、送信側システムで定義されたパスチケット・クラスで使用されるキーと一致していなければなりません。

以下の例では、2つのユーザー ID (USER0001 および USER0002)、パスチケット・クラス (APPLI2I)、およびアプリケーション名 (APPLI2I) を RACF に対して定義します。USER0002 のみに RACF 内で APPLI2I へのアクセス権限が与えられているため、USER0002 のみがトラステッド・ユーザー接続を確立できます。

```
DELUSER USER001
ADDUSER USER001 PASSWORD(USER0001) TSO(ACCTNUM(D1001) PROC(TPROC02))
DELUSER USER002
ADDUSER USER002 PASSWORD(USER0002) TSO(ACCTNUM(D1001) PROC(TPROC02))

SETROPTS CLASSACT(PTKTDATA)
SETROPTS RACLIST(PTKTDATA)
RDEF PTKTDATA APPLI2I SSIGNON(KEYMASKED(...)) UACC(N)
RDEF PTKTDATA APPLI2I.USER002 SSIGNON(KEYMASKED(...)) +
    UACC(N) APPLDATA('NO REPLAY PROTECTION')
SETROPTS REFRESH RACLIST(PTKTDATA)

SETROPTS CLASSACT(APPL)
SETROPTS RACLIST(APPL)
RDEFINE APPL APPLI2I UACC(N)
PE APPLI2I ACCESS(READ) CLASS(APPL) ID(USER0002)
SETROPTS RACLIST(APPL) REFRESH
RLIST APPL APPLI2I AU
```

## IMS Connect セキュリティー出口ルーチン

IMS Connect ユーザー・メッセージ出口でセキュリティー検査を実行する場合、セキュリティー出口ルーチンを提供するか、あるいは z/OS TCP/IP IMS リスナー・セキュリティー出口ルーチン (IMSLSECX) を使用する必要があります。

多くのオプションをセキュリティーに使用でき、さらに、大多数のインストール済み環境が独自の特定セキュリティー方式を使用しているため、IMS は、サンプル・セキュリティー出口ルーチンを提供していません。メッセージ出口ルーチンがメッセージを戻すとき、OTMA ヘッダーに RACF パラメーターの指定があれば、IMS Connect は RACF の呼び出しを実行します。

HWSSMPLO、HWSSMPL1、HWSSOAP1、または HWSCSLOO が呼び出すセキュリティー出口ルーチンの名前は、IMSLSECX です。

HWSSMPLO または HWSSMPL1 を使用する場合、EXTRN IMSLSECX を選択した名前に変更することで、これらが呼び出すセキュリティー出口ルーチンの名前を変更できます。セキュリティー出口ルーチンの名前を変更する場合には、HWSSMPLO または HWSSMPL1 ユーザー・メッセージ出口にそのセキュリティー出口ルーチンを定義する必要があります。

また、HWSJAVA0 によって呼び出されるセキュリティー出口ルーチンの名前を準備し、それを HWSJAVA0 メッセージ出口ルーチンに定義することもできます。

## IMS Connect セキュリティーおよび OTMARTUX ユーザー出口

OTMA RESUME TPIPE セキュリティー・ユーザー出口 (OTMARTUX) は IMS Connect 出口ルーチンではありませんが、OTMA 非同期保留キューに入れられたメッセージを保護するために使用できる 2つの可能な方法の 1つです。もう 1つは、RACF などの外部セキュリティー製品を使用する方法です。OTMARTUX ユーザー出口と外部セキュリティー製品をそれぞれ単独で、または組み合わせて使用することができます。

OTMARTUX ユーザー出口は、IMS Connect のアドレス・スペースではなく、IMS 制御領域内で実行されません。

### 関連タスク

838 ページの『非同期保留キュー内のメッセージの保護』

非同期保留キューに入っているメッセージを RESUME TPIPE 呼び出しの無許可使用から保護するためには、RACF、OTMA RESUME TPIPE セキュリティー ユーザー出口 (OTMARTUX)、またはその両方を使用する必要があります。

#### 関連資料

[OTMARTUX: OTMA RESUME TPIPE セキュリティー・ユーザー出口 \(DFSYRTUX およびその他の OTMARTUX タイプの出口\) \(出口ルーチン\)](#)

## HWSSMPLO および HWSSMPL1 セキュリティー処置

サンプル・ユーザー・メッセージ出口 HWSSMPLO および HWSSMPL1 は、特定のセキュリティ処置を常に実行し、IMSLSECX セキュリティー出口が呼び出された場合または呼び出されない場合にのみ他のセキュリティ処置を実行します。

### HWSSMPLO および HWSSMPL1 が IMSLSECX を呼び出さない場合のセキュリティ処置

以下の表では、セキュリティ出口 (IMSLSECX) を呼び出さなかったときに HWSSMPLO および HWSSMPL1 が取る処置を定義します。

表 6. セキュリティー出口が呼び出されない場合の USERID の結果

	IRM に USERID フィールドがある	IRM USERID フィールドが 空白 / ヌル	OTMA セキュリティー・ヘッダー で渡された RACF パラメーター の結果
USERID	あり	あり	デフォルトの RACFID
USERID	あり	なし	IRM USERID
USERID	なし	N/A	デフォルトの RACFID

表 7. セキュリティー出口が呼び出されない場合の GROUPID の結果

	IRM に GROUPID フィールドがある	IRM USERID フィールドが 空白 / ヌル	OTMA セキュリティー・ヘッダー で渡された RACF パラメーター の結果
GROUPID	あり	あり	空白 / ヌル
GROUPID	あり	なし	IRM GROUPID
GROUPID	なし	N/A	空白 / ヌル

表 8. セキュリティー出口が呼び出されない場合の Password の結果

	IRM にパスワード・フィールドがある	IRM PASSWORD フィールドが 空白 / ヌル	OTMA セキュリティー・ヘッダー で渡された RACF パラメーター の結果
PASSWORD	あり	あり	空白 / ヌル
PASSWORD	あり	なし	IRM PASSWORD
PASSWORD	なし	N/A	空白 / ヌル

### HWSSMPLO および HWSSMPL1 が IMSLSECX を呼び出す場合のセキュリティ処置

以下の表では、セキュリティ出口 (IMSLSECX) を呼び出すときに HWSSMPLO および HWSSMPL1 が取る処置を定義します。

表 9. セキュリティー出口が呼び出されたときの *USERID* の結果; ブランク または非ブランクの *USERID* を戻す

	<b>IRM に <i>USERID</i> フィールドがある</b>	<b>IRM <i>USERID</i> フィールドがブランク / ヌル</b>	<b>セキュリティー出口が <i>USERID</i> を戻す</b>	<b>OTMA セキュリティー・ヘッダーで渡された <i>RACF</i> パラメーターの結果</b>
<i>USERID</i>	あり	あり	なし	デフォルトの <i>RACF USERID</i>
<i>USERID</i>	あり	あり	あり	セキュリティー出口が <i>USERID</i> を戻した
<i>USERID</i>	あり	なし	なし	IRM で渡された <i>USERID</i>
<i>USERID</i>	あり	なし	あり	セキュリティー出口が <i>USERID</i> を戻した
<i>USERID</i>	なし	N/A	なし	デフォルトの <i>RACF USERID</i>
<i>USERID</i>	なし	N/A	あり	セキュリティー出口が <i>USERID</i> を戻した

表 10. セキュリティー出口が呼び出されたときの *GROUPID* の結果; 非ブランクの *USERID* を戻す

	<b>IRM に <i>GROUPID</i> フィールドがある</b>	<b>IRM <i>GROUPID</i> フィールドがブランク / ヌル</b>	<b>セキュリティー出口が <i>GROUPID</i> を戻す</b>	<b>OTMA セキュリティー・ヘッダーで渡された <i>RACF</i> パラメーターの結果</b>
<i>GROUPID</i>	あり	あり	なし	ブランクの <i>GROUPID</i>
<i>GROUPID</i>	あり	あり	あり	セキュリティー出口が <i>GROUPID</i> を戻した
<i>GROUPID</i>	あり	なし	なし	ブランクの <i>GROUPID</i>
<i>GROUPID</i>	あり	なし	あり	セキュリティー出口が <i>GROUPID</i> を戻した
<i>GROUPID</i>	なし	N/A	なし	ブランクの <i>GROUPID</i>
<i>GROUPID</i>	なし	N/A	あり	セキュリティー出口が <i>GROUPID</i> を戻した

**重要:** セキュリティー出口が、以下の表に示すようにブランクの *USERID* を戻した場合は、出口から戻された *GROUPID* は使用されません。

表 11. セキュリティー出口が呼び出されたときの *GROUPID* の結果; ブランクの *USERID* を戻す

	<b>IRM に <i>GROUPID</i> フィールドがある</b>	<b>IRM <i>GROUPID</i> フィールドがブランク / ヌル</b>	<b>セキュリティー出口が <i>GROUPID</i> を戻す</b>	<b>OTMA セキュリティー・ヘッダーで渡された <i>RACF</i> パラメーターの結果</b>
<i>GROUPID</i>	あり	あり	なし	ブランクの <i>GROUPID</i>
<i>GROUPID</i>	あり	あり	あり	ブランクの <i>GROUPID</i>
<i>GROUPID</i>	あり	なし	なし	IRM <i>GROUPID</i>
<i>GROUPID</i>	あり	なし	あり	IRM <i>GROUPID</i>
<i>GROUPID</i>	なし	N/A	なし	ブランク
<i>GROUPID</i>	なし	N/A	あり	ブランク

## HWSSMPL0 および HWSSMPL1 が常に行うセキュリティ処置

IRM に ID \*NETUID\* または \*NETSID\* を持つ IRM 拡張が含まれている場合、HWSSMPL0 および HWSSMPL1 は、IRM 拡張内のネットワーク・セキュリティ資格情報が含まれるように OTMA メッセージ接頭部のセキュリティ・データ・セクションを作成します。

以下の表は、セキュリティ出口 (IMSLSECX) が呼び出されたかどうかに関係なく、HWSSMPL0 および HWSSMPL1 も取る処置が定義されています。パスワードは、セキュリティ出口ではなく、IRM に基づいて使用されます。

表 12. セキュリティ出口の呼び出しの有無に関係しない Password の結果

	IRM に PASSWORD フ ィールドがある	PASSWORD フ ィールドがブラ ンク / ヌル	セキュリティ 出口が PASSWORD を 戻す	OTMA セキュリティ・ヘッダーで 渡された RACF パラメーターの結 果
PASSWORD	あり	あり	N/A	ブランク / ヌル
PASSWORD	あり	なし	N/A	IRM PASSWORD
PASSWORD	なし	N/A	N/A	ブランク / ヌル

## サンプル出口からの RACROUTE 呼び出しでのエラーに対する IMS Connect 応答

IMS Connect は、IMS Connect 構成ファイルに設定されている RACF パラメーター や、エラーの原因となった特定の環境などに応じて、どのようなエラー処置を取るかを決定します。

IMS Connect が取る別の処置については、以下の一連の表で説明されています。

### RACF=Y および RACROUTE 呼び出しパラメーター・エラーの場合の IMS Connect 応答

以下の表では、必須の RACROUTE 呼び出しパラメーターに基づいて、RACF=Y のときに IMS Connect が取るエラー処置が説明されています。

表 13. RACROUTE 呼び出しパラメーターに基づいて取られる IMS Connect エラー処置 (RACF=Y)

USERID	PASSWORD	GROUPID	取られる処置
非ブランク	非ブランク	非ブランク	RACROUTE 呼び出しが発行された
非ブランク	ブランク	ブランク	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1503 が発行された</li> <li>入力のリジェクトされた</li> <li>RACROUTE 呼び出しが発行されなかった</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='SECFNOPW' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>* Security failed, no password * (セキュリティは失敗した、パスワードなし)</li> </ul>

表 13. RACROUTE 呼び出しパラメーターに基づいて取られる IMS Connect エラー処置 (RACF=Y) (続き)

USERID	PASSWORD	GROUPID	取られる処置
非blank	blank	非blank	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1503 が発行された</li> <li>入力がりジェクトされた</li> <li>RACROUTE 呼び出しが発行されなかった</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='SECFNOPW' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>* Security failed, no password * (セキュリティは失敗した、パスワードなし)</li> </ul>
blank	非blank	blank	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1503 が発行された</li> <li>入力がりジェクトされた</li> <li>RACROUTE 呼び出しが発行されなかった</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='SECFNUID' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>* Security failed, no password * (セキュリティは失敗した、パスワードなし)</li> </ul>
blank	非blank	非blank	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1503 が発行された</li> <li>入力がりジェクトされた</li> <li>RACROUTE 呼び出しが発行されなかった</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='SECFNUID' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>* Security failed, no password * (セキュリティは失敗した、パスワードなし)</li> </ul>
blank	blank	非blank	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1503 が発行された</li> <li>入力がりジェクトされた</li> <li>RACROUTE 呼び出しが発行されなかった</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='SECFNPUI' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>* Security failed, no password * (セキュリティは失敗した、パスワードなし)</li> </ul>



表 13. RACROUTE 呼び出しパラメーターに基づいて取られる IMS Connect エラー処置 (RACF=Y) (続き)

USERID	PASSWORD	GROUPID	取られる処置
ブランク	ブランク	ブランク	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1503 が発行された</li> <li>入力がりジェクトされた</li> <li>RACROUTE 呼び出しが発行されなかった</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='SECFNPUI' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>* Security failed, no password * (セキュリティは失敗した、パスワードなし)</li> </ul>
非ブランク	ブランク	非ブランク	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1503 が発行された</li> <li>入力がりジェクトされた</li> <li>RACROUTE 呼び出しが発行されなかった</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='SECFNOPW' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>* Security failed, no password * (セキュリティは失敗した、パスワードなし)</li> </ul>

### RACF=Y、および、RACROUTE 呼び出し失敗または OTMA ヘッダー・データ・エラーの場合の IMS Connect 応答

以下の表では、OTMA ヘッダー・データに基づいて、あるいは RACROUTE 呼び出しが失敗した場合のいずれかで、RACF=Y のときに IMS Connect が取るエラー処置が説明されています。

表 14. RACROUTE 呼び出し障害または OTMA ヘッダー・データ・エラーで RACF=Y の場合に IMS Connect が取る処置

RACROUTE 呼び出し障害または 取られる処置 OTMA ヘッダー・データ	
セキュリティ・ヘッダーなし	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1503 が発行された</li> <li>入力がりジェクトされた</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='NOSECHDR' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>RACF 呼び出しが行われなかった</li> </ul>
セキュリティ・ヘッダー < X'6A'	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1503 が発行された</li> <li>入力がりジェクトされた</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='INVSECHL' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>RACF 呼び出しが行われなかった</li> </ul>

表 14. RACROUTE 呼び出し障害または OTMA ヘッダー・データ・エラーで RACF=Y の場合に IMS Connect が取る処置 (続き)

**RACROUTE 呼び出し障害または 取られる処置  
OTMA ヘッダー・データ**

会話が続行する	<ul style="list-style-type: none"> <li>エラー・メッセージが発行されなかった</li> <li>入力を受け入れられた</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>RACF 呼び出しが行われなかった</li> </ul>
応答メッセージ	<ul style="list-style-type: none"> <li>エラー・メッセージが発行されなかった</li> <li>入力を受け入れられた</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>RACF 呼び出しが行われなかった</li> </ul>
UTOKEN が存在	<ul style="list-style-type: none"> <li>エラー・メッセージが発行されなかった</li> <li>入力を受け入れられた</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>RACF 呼び出しが行われなかった</li> </ul>
RACROUTE 呼び出しが失敗した	<ul style="list-style-type: none"> <li>エラー・メッセージ HWSP1500 が発行された</li> <li>入力のリジェクトされた</li> <li>OMUSR_RETCODE=X'04' を設定</li> <li>OMUSR_RESCODE='SECFAIL' を設定</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>RACF 戻りコード/理由コードが HWSP1500 メッセージにある</li> </ul>
その他のすべて	193 ページの表 13 を参照してください。

**RACF=N および RACROUTE 呼び出しパラメーター・エラーの場合の IMS Connect 応答**

以下の表では、必須の RACROUTE 呼び出しパラメーターに基づいて、RACF=N のときに IMS Connect が取るエラー処置が説明されています。

表 15. RACROUTE 呼び出しパラメーターに基づいて取られる IMS Connect エラー処置 (RACF=N)

USERID	PASSWORD	GROUPID	取られる処置
非ブランク	非ブランク	非ブランク	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>これらのパラメーターを OTMA に渡す</li> </ul>
非ブランク	ブランク	ブランク	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>これらのパラメーターを OTMA に渡す</li> </ul>
非ブランク	ブランク	非ブランク	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>これらのパラメーターを OTMA に渡す</li> </ul>

表 15. RACROUTE 呼び出しパラメーターに基づいて取られる IMS Connect エラー処置 (RACF=N) (続き)

USERID	PASSWORD	GROUPID	取られる処置
ブランク	非ブランク	ブランク	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>これらのパラメーターを OTMA に渡す</li> </ul>
ブランク	非ブランク	非ブランク	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>これらのパラメーターを OTMA に渡す</li> </ul>
ブランク	ブランク	非ブランク	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>これらのパラメーターを OTMA に渡す</li> </ul>
ブランク	ブランク	ブランク	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>これらのパラメーターを OTMA に渡す</li> </ul>
非ブランク	ブランク	非ブランク	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>これらのパラメーターを OTMA に渡す</li> </ul>

### RACF=N、および、RACROUTE 呼び出し失敗または OTMA ヘッダー・データ・エラーの場合の IMS Connect 応答

以下の表では、OTMA ヘッダー・データに基づいて、あるいは RACROUTE 呼び出しが失敗した場合のいずれかで、RACF=N のときに IMS Connect が取るエラー処置が説明されています。

表 16. RACROUTE 呼び出しの障害または OTMA ヘッダー・データに対して取られる IMS Connect エラー処置 (RACF=N)

RACROUTE 呼び出し障害または OTMA ヘッダー・データ	取られる処置
セキュリティ・ヘッダーなし	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>OTMA ヘッダーとデータを IMS OTMA に渡す</li> </ul>
セキュリティ・ヘッダー < X'6A'	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>OTMA ヘッダーとデータを IMS OTMA に渡す</li> </ul>
会話が続行する	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>OTMA ヘッダーとデータを IMS OTMA に渡す</li> </ul>

表 16. RACROUTE 呼び出しの障害または OTMA ヘッダー・データに対して取られる IMS Connect エラー処  
置 (RACF=N) (続き)

**RACROUTE 呼び出し障害または 取られる処置  
OTMA ヘッダー・データ**

応答メッセージ	<ul style="list-style-type: none"> <li>エラー・メッセージが発行されなかった</li> <li>入力を受け入れられた</li> <li>OTMA ヘッダーでパスワードが消去された</li> <li>RACF 呼び出しが行われなかった</li> </ul>
UTOKEN が存在	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>OTMA ヘッダーとデータを IMS OTMA に渡す</li> </ul>
RACROUTE 呼び出しが失敗した	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>OTMA ヘッダーとデータを IMS OTMA に渡す</li> </ul>
その他のすべて	<ul style="list-style-type: none"> <li>パスワードが消去された</li> <li>RACROUTE 呼び出しをバイパス</li> <li>OTMA ヘッダーとデータを IMS OTMA に渡す</li> </ul>

## IMS Connect パスワード管理

IMS Connect は、RACF パスワードの管理に役立ついくつかの機能を提供しています。これらの機能の中には、IMS Connect が RACF を直接呼び出すように構成されている場合にのみ適用されるものもあります。

### このタスクについて

## クライアント・メッセージの使用による RACF パスワードの変更

IMS Connect が RACF を直接呼び出すように構成されている場合、ユーザー・メッセージ出口ルーチン HWSSMPL0、HWSSMPL1、および HWSJAVA0 のユーザーは、パスワード変更要求のキーワードを含むクライアント・メッセージをサブミットすることで、RACF パスワードを変更できます。

### このタスクについて

この機能を使用可能にするには、使用しているユーザー・メッセージ出口ルーチンに HWSPWCHO オブジェクト・コードをバインドする必要があります。HWSPWCHO オブジェクト・コードは、配布ライブラリーの IMS.ADFSLOAD メンバーに保管されています。

パスワード変更要求キーワードは、メッセージのアプリケーション・データ・セクションの先頭に表示し、その後には空白、古いパスワード、新しいパスワード、およびもう一度新しいパスワードを続ける必要があります。

例えば、HWSSMPL0 および HWSSMPL1 ユーザー・メッセージ出口ルーチンのパスワード変更要求メッセージのフォーマットを以下に示します。

```
1111IRM11zzHWSPWCH old-password/new_password/new_password|11zz
```

HWSJAVA0 ユーザー・メッセージ出口ルーチンのパスワード変更要求メッセージのフォーマットを以下に示します。

```
1111IRM0TMALLzzHWSPWCH old_password/new_password/new_password|11zz
```

サンプル・ユーザー・メッセージ出力ルーチンに定義されているパスワード変更要求キーワードは、「HWSPWCH」です。ユーザー・メッセージ出力ルーチンを変更することによって、このキーワードを変更できます。どのようなパスワード変更要求キーワードを定義する場合でも、後に区切り文字として空白を続ける必要があります。

IMS Connect は、以下のいずれかのフォーマットで、クライアント・アプリケーションに応答メッセージを返します。

- ユーザー・メッセージ出力ルーチン HWSSMPL0 の場合のフォーマットは以下のとおりです。

```
1lzzmessage_textCSM1lzz
```

- ユーザー・メッセージ出力ルーチン HWSSMPL1 の場合のフォーマットは以下のとおりです。

```
1l11message_textCSM1lzz
```

- ユーザー・メッセージ出力ルーチン HWSJAVA0 の場合のフォーマットは以下のとおりです。

```
11110TMAheadermessage_text1lzz
```

パスワード変更要求の通信シーケンスは以下のとおりです。

## 手順

1. 接続します。
2. パスワード変更要求を送信します。
3. メッセージ HWSC00 $xy$ を受信します。ここで、 $xx$ は、メッセージ番号の最終の2桁であり、 $y$ はメッセージ・タイプ ID です。

HWSSMPL0 または HWSSMPL1 を使用している場合、ユーザー作成のクライアント・アプリケーションが受信するメッセージ HWSC00 $xy$ には、\*CSMOKY\* が付加されます。

HWSJAVA0 を使用している場合、IMS TM Resource Adapter のクライアント・アプリケーションは、OTMA ヘッダーのアプリケーション・データ部分に HWSC00 $xy$  メッセージを受信し、OTMA ヘッダーのユーザー・データ部分に対応する戻りコードと理由コードを受信します。

## 関連資料

[IMS TM Resource Adapter ユーザー・メッセージ出力ルーチン \(HWSJAVA0\) \(出力ルーチン\)](#)  
[ユーザー・メッセージ出力ルーチン HWSSMPL0 および HWSSMPL1 \(出力ルーチン\)](#)

## クライアント・メッセージの使用による RACF パスワード・フレーズの変更

IMS Connect が RACF を直接呼び出すように構成されている場合、ユーザー・メッセージ出力ルーチン HWSSMPL0、HWSSMPL1、および HWSJAVA0 のユーザーは、パスワード・フレーズ変更要求キーワードを含むクライアント・メッセージをサブミットすることで、RACF パスワード・フレーズを変更できます。

### このタスクについて

この機能を使用可能にするには、使用しているユーザー・メッセージ出力ルーチンに HWSPWCHO オブジェクト・コードをバインドする必要があります。HWSPWCHO オブジェクト・コードは、配布ライブラリーの IMS.ADFSLOAD メンバーに保管されています。

パスワード・フレーズの変更要求は、パスワードの変更要求に似ており、同じキーワードを使用します。よって、パスワード変更要求キーワードをメッセージのアプリケーション・データ・セクションの先頭に表示し、そのあとに空白、古いパスワード・フレーズ、空白、新規パスワード・フレーズ、空白、さらにもう一度新規パスワード・フレーズを続ける必要があります。パスワード・フレーズは単一引用符で囲まなければなりません。単一引用符がパスワード・フレーズの一部である場合は、そのあとに2つ目の単一引用符を付ける必要があります。パスワード・フレーズの間には少なくとも1個の空白が必要です。

例えば、HWSSMPL0 および HWSSMPL1 ユーザー・メッセージ出力ルーチンのパスワード変更要求メッセージのフォーマットを以下に示します。

```
1111IRM11zzHWSPWCH 'old_phrase' 'new_phrase' 'new_phrase'|11zz
```

HWSJAVA0 ユーザー・メッセージ出口ルーチンの場合、パスワード・フレーズ変更要求メッセージのフォーマットは、次のようになります。

```
1111IRMOTMA11zzHWSPWCH 'old_phrase' 'new_phrase' 'new_phrase'|11zz
```

サンプル・ユーザー・メッセージ出口ルーチンに定義されているパスワード変更要求キーワードは、「HWSPWCH」です。ユーザー・メッセージ出口ルーチンを変更することによって、このキーワードを変更できます。どのようなパスワード変更要求キーワードを定義する場合でも、後に区切り文字としてブランクを続ける必要があります。

IMS Connect は、以下のいずれかのフォーマットで、クライアント・アプリケーションに応答メッセージを返します。

ユーザー・メッセージ出口ルーチン HWSSMPL0 の場合のフォーマットは以下のとおりです。

```
11zzmessage_textCSM11zz
```

ユーザー・メッセージ出口ルーチン HWSSMPL1 の場合のフォーマットは以下のとおりです。

```
1111message_textCSM11zz
```

ユーザー・メッセージ出口ルーチン HWSJAVA0 の場合のフォーマットは以下のとおりです。

```
11110TMAheadermessage_text11zz
```

パスワード変更要求の通信シーケンスは以下のとおりです。

## 手順

1. 接続します。
2. パスワード変更要求を送信します。
3. メッセージ HWSC00xy を受信します。ここで、xx は、メッセージ番号の最終の 2 桁であり、y はメッセージ・タイプ ID です。HWSJAVA0 を使用している場合、IMS TM Resource Adapter のクライアント・アプリケーションは、OTMA ヘッダーのアプリケーション・データ部分に HWSC00xy メッセージを受信し、OTMA ヘッダーのユーザー・データ部分に対応する戻りコードと理由コードを受信します。

## 大/小文字混合パスワード・サポートの使用可能化

IMS Connect は大/小文字混合パスワードをサポートします。大/小文字混合パスワードを使用するには、RACF (または、類似のセキュリティー製品) も大/小文字混合パスワードをサポートしていなければなりません。

### このタスクについて

デフォルトでは、IMS Connect による大/小文字混合パスワードのサポートは、使用中のセキュリティー製品の構成によって決まります。例えば、RACF が大/小文字混合パスワードをサポートする場合には、IMS Connect も自動的に大/小文字混合パスワードをサポートします。

IMS Connect は、セキュリティー製品が大/小文字混合パスワードをサポートするか判断できない場合に、システム定義中または IMS Connect の実行中にオンライン・コマンドで他の処理を行うよう指示されていなければ、すべてのパスワードを大文字に変換します。

IMS Connect システムの定義時に、HWS 構成ステートメントで PSWDMC パラメーターを使用することにより、大/小文字混合パスワードのサポートに関する RACF 仕様を使用可能にする、使用不可にする、または受け入れることができます。

オンライン・システムでは、以下のいずれかのコマンドを使用して、大/小文字混合パスワードのサポートに関する RACF 仕様を使用可能にする、使用不可にする、または受け入れることができます。

- IMS Connect WTOR コマンド **SETPWMC ON | OFF | RCF**

- IMS Connect タイプ 2 コマンド **UPDATE IMSCON TYPE(CONFIG) SET(PSWDMC(ON | OFF | RCF))**
- IMS Connect z/OS のコマンド **UPDATE MEMBER TYPE(IMSCON) SET(PSWDMC(ON | OFF | RCF))**  
大/小文字混合パスワード・サポートの現行設定を表示するには、以下のいずれかのコマンドを発行します。
- IMS Connect WTOR コマンド **VIEWHWS**
- タイプ 2 コマンド **QUERY IMSCON TYPE(CONFIG)**
- IMS Connect z/OS のコマンド **QUERY MEMBER**

#### 関連資料

[IMS Connect UPDATE MEMBER コマンド \(コマンド\)](#)

[IMS Connect QUERY MEMBER コマンド \(コマンド\)](#)

[SETPWMC コマンド \(コマンド\)](#)

[VIEWHWS コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[QUERY IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[HWS ステートメント \(システム定義\)](#)

## IMS Connect の RACF パスチケットのサポート

RACF パスワードに代わるものとして、パスチケットがあります。パスチケットを使用すると、RACF パスワードを使用せずにホストと通信できます。IMS Connect が RACF を直接呼び出すように構成されている場合、パスチケットを使用して、ユーザー ID を認証し、RACF を収めるコンピューター・システムにログオンすることができます。

### IMS TM への IMS Connect クライアント接続用の RACF パスチケット

RACF パスワードに代わるものとしてパスチケットがあり、IMSTM への IMS Connect クライアント・アクセスに使用できます。パスチケットを使用すると、RACF パスワードを使用せずにホストと通信できます。IMS Connect が RACF を直接呼び出すように構成されている場合、パスチケットを使用して、ユーザー ID を認証し、RACF を収めるコンピューター・システムにログオンすることができます。

IMS TM にアクセスする IMS Connect クライアントでは、クライアントを介してパスチケット・サポートを選択し、RACF パスワードの代わりにパスチケットを IRM で送信することができます。IMS Connect は、パスチケットを使用して RACF 呼び出しを発行し、IMS にメッセージを送信する前に OTMA ユーザー・データ・ヘッダーのパスチケット・フィールドを隠します。パスチケットは RACF パスワードと同じフィールドを占め、大文字に変換できないため、RACF パスワードも大文字に変換されません。大文字に変換するために、ユーザー・メッセージ出口を使用できます。

IMS TM クライアントの IMS Connect パスチケット・サポートは、IMS パスチケット・サポートに類似しています。

- 新たに IMS Connect クライアントと接続するために、既存の APPL 名定義を使用できます。
- 各 DATASTORE ステートメントに、パラメーター `APPL=APPLname` が含まれます。ここで、
  - 各 APPL= は、データ・ストアごとに固有の RACF APPL 名に設定することができます。
  - 各 APPL= は、VGR サポートの場合に要求されるようにデータ・ストアごとに同じ名前にするこも、データ・ストアごとに固有にすることもできます。
- デフォルトの `APPL=APPLname` 値はブランクです。
- IMS Connect クライアントは、OTMA ユーザー・データ・ヘッダーに `APPLname` を設定するユーザー・メッセージ出口に、IRM で `APPLname` を渡すことができます。あるいは、ユーザー・メッセージ出口が適切な `APPLname` を OTMA ユーザー・データ・ヘッダーに渡して設定することもできます。

パスチケット・サポートの場合、RACF へのすべての定義はお客様が責任を負います。RACF エンコードおよびデコード・ルーチンを確立し、分散プラットフォームにエンコード・ルーチンを提供する必要があります。



IMS TM クライアントでは、RACF パスチケットは、お客様作成の IMS Connect クライアント・アプリケーションの場合のみサポートされます。IMS TM Resource Adapter は、現在、RACF パスチケットをサポートしていません。

このサポートでは、お客様作成のユーザー・メッセージ出口およびお客様作成のクライアント・アプリケーション・コードの変更が必要になることもあります。次のリストは、IMS TM にアクセスする IMS Connect クライアントのパスチケット・サポート用に選択できるオプションを示しています。

#### • IRM の APPL 名を IMS Connect に渡すためのサポート

このサポートが IRM 定義に追加されました。新規の 8 バイト・フィールド IRM\_APPL\_NM が、IRM 構造の末尾に追加されました。IMS TM への IMS Connect クライアント・アクセスのためにパスチケット機能をインプリメントする場合、クライアント・コードは、このフィールドを使用して APPL 名を IMS Connect に渡す必要があります。

**注:** これにより、IRM の長さが 8 バイトおよびメッセージの全長が 8 バイト変更されます。

提供されたユーザー・メッセージ出口 (HWSSMPL1 および HWSSMPL0) は、クライアントが IRM\_APPL\_NM フィールドを使用して APPL 名を IMS Connect に送信できるように変更されています。

このオプションを選択した場合、必要なアクションは IRM で APPL 名を渡すのみです。HWSIMSCB および IMS Connect は、この機能をサポートするように変更されています。

#### • IRM の APPL 名を IMS Connect に渡すためのサポートなし

このサポートが IRM 定義に追加されました。新規の 8 バイト・フィールド IRM\_APPL\_NM が、IRM 構造の末尾に追加されました。IMS TM への IMS Connect クライアント・アクセスのためにパスチケット機能をインプリメントする必要がない場合、以下の 2 つのオプションがあります。

##### – オプション 1: ブランク APPL 名

IMS Connect への IRM\_APPL\_NM フィールドでブランクの APPL 名を IMS Connect に渡すことを選択できます。

**注:** これにより、IRM の長さが 8 バイトおよびメッセージの全長が 8 バイト変更されます。

提供されたユーザー・メッセージ出口 (HWSSMPL1 および HWSSMPL0) は、クライアントが IRM\_APPL\_NM フィールドを使用してブランク APPL 名を IMS Connect に送信できるように変更されています。

このオプションを選択した場合、必要なアクションは IRM でブランク APPL 名を渡すのみです。HWSIMSCB および IMS Connect は、このブランク APPL 名機能をサポートするように変更されています。

##### – オプション 2: APPL 名なし

IMS Connect への IRM\_APPL\_NM フィールドで APPL 名を IMS Connect に渡さないことを選択できます。

**注:** これにより、IRM の長さも、メッセージの全長も変更されません。

提供されたユーザー・メッセージ出口 (HWSSMPL1 および HWSSMPL0) は、クライアントが IRM\_APPL\_NM フィールドを使用して APPL 名を IMS Connect に送信する必要がないように変更されています。

このオプションを選択した場合、どのようなアクションも取る必要がありません。HWSIMSCB および IMS Connect は、APPL 名を渡さないこの機能をサポートするように変更されています。

APPLname は常に RACF に渡されます。これは、パスチケットが使用されない場合にも当てはまります。その結果、パスチケットが使用されない場合でも、DATASTORE ステートメントの APPL= キーワードを使用して、IMS Connect データ・ストアにアクセスするためのユーザーの権限を確認できます。

## IMS DB との IMS Connect クライアント接続用の RACF パスチケット

RACF パスチケットを使用して、IMS DB との IMS Connect クライアント接続を認証することができます。パスチケットは、RACF パスワードおよびパスワード・フレーズの代わりとして使用でき、ネットワーク経由でパスワードおよびパスワード・フレーズを暗号化されていないテキストとして送信しなくてもすむようになるため、セキュリティを強化します。

## このタスクについて

DRDA クライアントから IMS DB へのユーザー・アクセスを認証するために RACF パスチケットが使用される場合、SQL バッチ・ユーティリティによってパスチケットを生成できます。SQL バッチ・ユーティリティの代わりに別の DRDA クライアントを使用して IMS DB にアクセスする場合は、RACF パスチケット生成アルゴリズムを使用する別の方法で、DRDA クライアントのパスチケットを生成して評価することができます。

次の手順の概要では、クライアントが DRDA を使用する場合に IMS DB との IMS Connect クライアント接続が RACF パスチケットによって認証される方法を示しています。

1. クライアント接続が最初に確立されると、IMS DB との接続の認証に使用される RACF パスチケットは、SQL バッチ・ユーティリティによって生成されるか、他の DRDA クライアントの場合は RACF パスチケット生成アルゴリズムを使用するサービスによって生成されます。
2. クライアント・アプリケーションは、生成されたパスチケットと、アクセスを必要とするユーザーの ID を、**SECCHK** コマンド (X'106E') で IMS Connect に送信します。パスチケットは、**SECCHK** コマンドの **PASSWORD** パラメーターでコード・ポイント X'11A1' に指定されます。ユーザー ID は、**SECCHK** コマンドの **USRID** パラメーターでコード・ポイント X'11A0' に指定されます。
3. IMS Connect は、クライアント接続を認証するために、RACROUTE REQUEST=VERIFY 呼び出しを RACF に対して発行します。RACF RACROUTE REQUEST=VERIFY 呼び出しでは、IMS Connect は以下の情報を組み込みます。
  - クライアント・アプリケーションから **SECCHK** コマンド (X'106E') で送信された RACF パスチケットとユーザー ID。
  - ODACCESS ステートメントの **APPL=** パラメーターに指定されるアプリケーション名。このステートメントは、IMS PROCLIB データ・セットの HWSCFGxx メンバーにあります。ODACCESS ステートメントの **APPL=** パラメーターにアプリケーション名が指定されていない場合、IMS Connect は、HWS ステートメントの **ID=** パラメーターに指定されている値を代わりに使用します。このステートメントも HWSCFGxx メンバーにあります。

### ヒント:

- RACF が IMS Connect インスタンスで使用可能になっていない場合は、パスチケットを生成するように DRDA クライアントを構成しないでください。RACF=N の場合、IMS Connect インスタンスはセキュリティー検査を実行せず、送信されたパスチケット・データを無視します。DRDA クライアント・サイドでパスチケットを作成するのは処理リソースの無駄になります。
- 非持続接続では RACF パスチケットを使用しないでください。パスチケットを使用すると、大きな処理上のオーバーヘッドが生じるからです。新しい接続が確立されるたびに、新しいパスチケットが生成されて送信されます。

## 手順

RACF パスチケットを使用して DRDA クライアントから IMS DB への接続を保護するには、以下のステップを実行します。

1. RACF に対して、パスチケット・クラス、アプリケーション・プロファイル、アプリケーション名、およびユーザー ID を定義します。
  - a) PTKTDATA クラスを活動化します。PTKTDATA クラスは、パスチケット情報を含むすべてのプロファイルが定義されているクラスです。このクラスと関数を活動化するには、次のコマンドを入力します。

```
SETROPTS CLASSACT(PTKTDATA) RACLIST(PTKTDATA)
```

- b) 次のコマンドを入力して、ユーザーがパスチケットを使用してアクセスする必要があるアプリケーションの名前を定義します。

```
RDEFINE APPL <applname> UACC(NONE)  
SETROPTS CLASSACT(APPL)  
SETROPTS GENERIC(PTKTDATA)
```

ただし、次のとおりです。

#### **applname**

アプリケーションの 1 文字から 8 文字の名前。

- c) **RDEFINE** コマンドを入力して、ユーザーがパスチケットを使用してアクセス権を取得できるアプリケーションのプロファイルを定義します。このプロファイルは、シークレットで保護されたサインオン・アプリケーション・キーをアプリケーションに関連付けます。

```
RDEFINE PTKTDATA <applname> SSIGNON(<key_description>(<key>))
```

ただし、次のとおりです。

#### **applname**

ステップ 203 ページの『1.b』で定義した 1 文字から 8 文字のアプリケーション名。

#### **key\_description**

ホストの RACF データベースで保護されたサインオン・アプリケーション・キーを保護するために RACF が使用する方法を指定します。以下のいずれかの値を指定できます。

#### **KEYMASKED**

保護されたサインオン・アプリケーション・キーをマスクします。

#### **KEYENCRYPTED**

保護されたサインオン・アプリケーション・キーを暗号化します。

#### **key**

保護されたサインオン・アプリケーション・キー。ユーザーによって指定される 16 文字の 16 進値 (0 から 9 および A から F) です。

- d) **PERMIT** コマンドを入力して、アプリケーションに対してユーザー ID を許可します。

```
PERMIT APPLNAME CLASS(APPL) ID(<userid>) ACCESS(UPDATE)
```

ただし、次のとおりです。

#### **userid**

アプリケーションへのアクセスを許可されるユーザー ID。SQL バッチ・ユーティリティーを使用してパスチケットを生成する場合は、ユーザー ID は、バッチ・ジョブに関連付けられている z/OS ユーザー ID でなければなりません。

- e) SQL バッチ・ユーティリティーを使用してパスチケットを生成する場合は、次のコマンドを入力し、アプリケーションに対して RACF パスチケット生成サービスの使用を許可します。

```
RDEFINE PTKTDATA IRRPTAUTH.<applname>.* UACC(NONE)  
PERMIT IRRPTAUTH.<applname>.* CLASS(PTKTDATA) ID(<userid>) ACCESS(UPDATE)
```

ただし、次のとおりです。

#### **applname**

ステップ 203 ページの『1.b』で定義されたアプリケーションの 1 文字から 8 文字の名前。

#### **userid**

アプリケーションへのアクセスを許可されるユーザー ID。SQL バッチ・ユーティリティーを使用してパスチケットを生成する場合は、ユーザー ID は、バッチ・ジョブに関連付けられている z/OS ユーザー ID でなければなりません。

- f) 次のコマンドを入力して、PTKTDATA クラスをリフレッシュし、変更を活動化します。

```
SETROPTS RACLIST(APPL) REFRESH  
SETROPTS RACLIST(PTKTDATA) REFRESH
```

2. HWSCFGxx メンバーの HWS ステートメントに **RACF=Y** が指定されていることを確認します。
3. 以下のいずれかの方法を使用して、ODACCESS ステートメントの **APPL=** パラメーターに、PTKTDATA クラスで RACF に定義されているアプリケーション名を指定します。つまり、**APPL=** パラメーターに、ステップ 203 ページの『1.b』で定義したアプリケーション名を指定します。IMS DB との DRDA クライアント接続を認証するための RACF 呼び出し RACROUTE REQUEST=VERIFY で、IMS Connect によ

り、ユーザー ID と RACF パスチケットに加えて、**APPL=** パラメーターに指定されたアプリケーション名が使用されます。

- HWSCFGxx メンバーの ODACCESS ステートメントに **APPL=** パラメーターを直接追加します。
- 次の IMS タイプ 2 コマンドを発行します。

```
UPDATE IMSCON TYPE(CONFIG) SET(ODBMAPPL(applname))
```

ここで、*applname* は、PTKTDATA クラスで RACF に定義されているアプリケーション名です。

**APPL=** パラメーターが指定されていない場合は、RACF 呼び出し RACROUTE REQUEST=VERIFY で、IMS Connect により、HWSCFGxx メンバーにある HWS ステートメントの **ID=** パラメーターの値が代わりに使用されます。そのため、**APPL=** パラメーターが指定されていない場合は、HWS ステートメントの **ID=** パラメーターに、PTKTDATA クラスで定義されているアプリケーション名が代わりに指定されていることを確認してください。

4. DRDA クライアント用にパスチケットを生成します。

SQL バッチ・ユーティリティを使用してパスチケットを生成する場合、`DriverManager.getConnection` メソッドの **appName** URL プロパティを指定します。

パスチケットを生成するために SQL バッチ・ユーティリティを使用しない場合は、RACF パスチケット生成アルゴリズムを使用してパスチケットを生成および評価するための他の方法について、[パスチケットの生成および評価](#)を参照して。

5. 生成されたパスチケットを使用して、IMS Connect クライアントのユーザーを RACF で認証するには、**SECCHK** コマンド (X'106E') を使用して、ユーザー ID とパスチケットを IMS Connect に送信します。**SECCHK** コマンド (X'106E') では、コマンドの **USRID** パラメーターのコード・ポイント X'11A0' にユーザー ID を組み込み、コマンドの **PASSWORD** パラメーターのコード・ポイント X'11A1' にパスチケットを組み込みます。

## 関連概念

[RACF パスチケットの使用 \(システム管理\)](#)

[パスチケットの生成および評価](#)

## 関連タスク

[JDBC DriverManager インターフェースを使用した IMS データベースへの接続 \(アプリケーション・プログラミング\)](#)

183 ページの『IMS Connect 内の RACF セキュリティ検査の使用可能化』

HWS 構成ステートメントで RACF=Y を指定するか、またはオンライン IMS Connect コマンドを発行することによって、IMS Connect 内の RACF セキュリティ検査を使用可能にできます。

## 関連資料

[SECCHK コマンド \(X'106E'\) \(アプリケーション・プログラミング API\)](#)

[QUERY IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[SQL バッチ・ユーティリティ \(データベース・ユーティリティ\)](#)

[ODACCESS ステートメント \(システム定義\)](#)

## パスチケット・リプレイ保護の考慮事項

パスチケット・リプレイ保護をバイパスすることができます。これは、同じユーザー ID を共有する複数のエンド・ユーザーが存在する場合に行うことがあります。

同じユーザー ID を使用する複数のユーザーが存在する場合、それらのユーザーは同じ時間間隔内でアプリケーションへのアクセスを要求することができます。この状態では、別々のユーザーに対して同じパスチケットが生成されます。結果として、パスチケット・リプレイ保護がバイパスされない場合、複数ユーザーが同じパスチケットを使用しているため、アプリケーションへのアクセスが拒否されます。パスチケット・リプレイ保護をバイパスすると、同じパスチケットを複数のユーザーで使用することができます。

同様に、IMS Connect への、考慮時間なしの駆動が要求されるシステムのストレス・テストを行う場合、および同じ時間間隔内で同じアプリケーションに対する多数の要求がある場合、パスチケット・リプレイ

保護のバイパスを考慮に入れることができます。このオプションにより、同じパスチケットを 10 分間以内に使用することができます。

選択したアプリケーションの 1 つ以上が 10 分間以内に同じパスチケットを生成するのを許可するために、PTKTDATA プロファイルの APPLDATA フィールドで「NO REPLAY PROTECTION」を指定できます。

リプレイなしオプションについての詳細は、「z/OS Security Server RACF セキュリティー管理者のガイド」の『一般リソースの保護』を参照してください。

## SSL および IMS Connect

IMS Connect への TCP/IP 接続上で Secure Socket Layer (SSL) をセットアップするには、IBM z/OS Communications Server Application Transparent Transport Layer Security (AT-TLS) を使用します。

AT-TLS を使用すると、AT-TLS は、着信メッセージを TCP/IP ポート上の IMS Connect に送信する前に z/OS で暗号化解除し、出力メッセージを IMS Connect クライアントに送信する前に暗号化します。IMS Connect は、AT-TLS または SSL のどちらが使用されているかを認識しません。詳しくは、[IMS Connect 用の AT-TLS SSL のセットアップ \(システム定義\)](#)を参照してください。

## IMS Connect メッセージのトラステッド・ユーザー・サポート

IMS Connect が RACF を直接呼び出すように構成されている場合、ユーザー・メッセージ出口を変更して、特定のメッセージをトラステッド・ユーザーとして扱うことができます。メッセージがトラステッド・ユーザーに分類されると、IMS Connect はそのメッセージのセキュリティーを検査するために RACF を呼び出さずに、指定されたユーザー ID を認証なしで OTMA に渡します。

### このタスクについて

IMS Connect のセキュリティーをバイパスした後もなお、IMS Connect に対してトラステッド・ユーザーに分類されているメッセージは、IMS OTMA が行う可能性のあるすべてのセキュリティー検査の対象になります。OTMA および IMS は、IMS Connect のトラステッド・ユーザー種別を認識しません。

以下の IMS Connect ユーザー・メッセージ出口が、トラステッド・ユーザー機能をサポートします。

- HWSSMPL0
- HWSSMPL1
- HWSJAVA0
- ユーザー作成のユーザー・メッセージ出口

トラステッド・ユーザー・サポートを使用可能にするには、以下のようになります。

- 入力メッセージをトラステッド・ユーザーと特定するためのフラグを収める 1 つ以上の IRM フィールドまたは HWSJAVA0 OTMA ヘッダー・フィールドを選択します。
- IRM フィールドまたはお客様が作成した接頭部フィールドの場合、ご使用のシステムで固有な定義になるように、トラステッド・ユーザー・フラグを表すバイトおよびバイト設定値を定義します。IMS Connect は、どのフラグ・バイトを設定するか、あるいはどの設定値を使用するかを定義しません。
- ユーザー・メッセージ出口がフィールドでこのフラグを読み取り、トラステッド・ユーザーが検出されたときに、この出口が IMS Connect に渡すメッセージの OTMA ユーザー・データ・セクションの OMUSR\_FLAG2 フィールド内の OMUSR\_TRSTUSR EQU フラグ用に X'80' ビットを設定するように、ユーザー・メッセージ出口をコーディングします。

トラステッド・ユーザー・メッセージを特定するために使用できる IRM ヘッダー・フィールドには、PORTID、CLIENTID、USERID、TRANSACTION CODE の各フィールドの中の 1 つ以上のフィールドとユーザー・データを含めることができます。

例えば、IRM 内に 3 つの 1 バイト・フィールドを追加して、それぞれのフィールドにさまざまな値を設定するように決めることもできます。クライアント・アプリケーションがこのフラグを設定します。メッセージがユーザー・メッセージ出口に渡されると、出口はそれらの 3 つのフィールドを調べます。これらのフィールドによってメッセージがトラステッド・ユーザーと特定されると、ユーザー・メッセージ出口は IMS Connect への要求を OTMA ヘッダーに渡し、RACF への呼び出しを迂回します。

サンプル・ロジック (コメント化されています) が HWSSMPL0 と HWSSMPL1 の両方で提供されています。これは、以下のコメント行を探すことで検出できます。

```
*****  
*****TRUSTED USER SUPPORT*****  
*****
```

HWSJAVA0 ユーザー・メッセージ出口を使用している場合、出口は OMUSR\_DESTID (DataStore)、OMUSR\_ORIGIN (ClientID)、OMUSR\_PORTID (PortId)、OMUSR\_PASSTICKET (Password)、その他のメッセージ値などの OTMA ヘッダー・フィールド内の既存データによって、トラステッド・ユーザー・メッセージを識別します。

HWSJAVA0 ユーザー・メッセージ出口をコーディングして、OMUSR\_FLAG2 フラグを OMUSR\_TRSTUSR に設定します。OMUSR\_FLAG2 フラグを OMUSR\_TRSTUSR に設定すると、IMS Connect は RACF の呼び出しを迂回します。

## IMS Connect 構成メンバーでの OTMA ACEE エージング値の指定

アクセサー環境エレメント (ACEE) のエージング値を OTMA に渡すように IMS Connect を構成することができます。ACEE エージング値を指定すると、OTMA はキャッシュされた RACF ACEE を、ご使用のシステムに適した頻度でリフレッシュすることができます。

### このタスクについて

ACEE には、ユーザー ID およびその他のセキュリティ情報が含まれます。OTMA は実行するセキュリティ検査のパフォーマンスを高めるために、ACEE をキャッシュに入れます。ACEE は、IMS Connect が要求されたトランザクションおよびコマンドに渡す入力メッセージに含まれるユーザー ID の権限を検証するために使用されます。

このセキュリティ情報を常に正確に保つために、OTMA は、セキュリティ情報が変更されたことを z/OS が IMS に通知したときに RACF に保管されているセキュリティ情報を使用して ACEE を自動的にリフレッシュします。ただし、OTMA は、z/OS が変更について IMS に通知するかどうかに関係なく、定義された間隔でも ACEE をリフレッシュします。ACEE のリフレッシュを実行する間隔は、OTMA に設定されている ACEE エージング値によって決まります。

IMS Connect が OTMA に渡す ACEE エージング値を定義するには、IMS Connect PROCLIB メンバー HWSCFGxx の DATASTORE 構成ステートメント内の OAAV= キーワードを使用します。300 秒から 999999 秒までの値を指定できます。デフォルトは 999999 秒です。ACEE のリフレッシュを有効にするために、OTMA では少なくとも 300 の値を指定する必要があります。OAAV= キーワードで指定されたエージング値は、以下の状態ではオーバーライドされます。

- **TMEMBER** パラメーターを使用せずに **/SECURE OTMA ACEEAGE** コマンドを発行して、OTMA クライアントを指定した場合。
- **/SECURE OTMA ACEEAGE** コマンドを発行し、**TMEMBER** パラメーターを使用して IMS Connect を指定した場合。

### 関連資料

[/SECURE コマンド \(コマンド\)](#)

[IMS PROCLIB データ・セットの HWSCFGxx メンバー \(システム定義\)](#)





# 第 14 章 IMS Connect のコールアウト要求サポート

IMS Connect は、IMS アプリケーション・プログラムが OTMA を介して、IMS インストール済み環境の外部にあるデータおよびサービス・プロバイダーにコールアウト要求を発行する場合に必要なコンポーネントです。両方のタイプのコールアウト要求で、IMS Connect は TCP/IP を使用する IMS Connect クライアントと IMS の OTMA コンポーネントとの間の TCP/IP ゲートウェイとして機能します。

## このタスクについて

IMS Connect は、IMS Connect の構成ステートメントを変更しなくても、コールアウト要求をサポートします。ただし、コールアウト要求をサポートするように IMS Connect クライアントを構成する必要があります。

以下のタイプの IMS Connect クライアントを使用して、同期および非同期のコールアウト要求をサポートできます。

- IMS TM Resource Adapter
- IMS Enterprise Suite SOAP Gateway
- ユーザー作成の IMS Connect クライアント

以下の IMS Connect 出口ルーチンは、コールアウト要求をサポートします。

- HWSJAVA0
- HWSSOAP1
- HWSSMPL0
- HWSSMPL1

IMS には、コールアウト・サポートをテストするためのさまざまなサンプル・アプリケーション・プログラムが用意されています。詳しくは、[コールアウト機能のサンプル \(インストール\)](#)を参照してください。

IMS TM Resource Adapter および SOAP Gateway の構成方法について詳しくは、以下を参照してください。

- [コールアウト・プログラミング・モデル \(TM Resource Adapter\)](#)
- [Web サービス・コンシューマーとしての IMS アプリケーションの使用可能化](#)

## 同期コールアウト要求のためのユーザー作成 IMS Connect クライアントの構成

同期コールアウト要求をサポートするには、IMS から新しいコールアウト要求をリトリートし、コールアウト要求の受信を確認して (ACK または NAK)、IMS Connect を介して同期コールアウト応答を IMS に返すように、ユーザー作成 IMS Connect クライアントを構成する必要があります。

### このタスクについて

同期コールアウト要求メッセージは、非同期出力とほぼ同じ方法で IMS Connect および OTMA によって処理されます。すなわち、同期コールアウト要求メッセージは RESUME TPIPE 呼び出しの発行によってリトリートされます。非同期出力の場合と同じ規則および指針の多くが、同期コールアウト要求メッセージのリトリートに適用されます。

IMS には、コールアウト・サポートをテストするためのさまざまなサンプル・アプリケーション・プログラムが用意されています。詳しくは、[コールアウト機能のサンプル \(インストール\)](#)を参照してください。

同期コールアウト・メッセージをサポートするようにユーザー提供の IMS Connect クライアントを構成するための概略的なステップには、以下のようにクライアントをコーディングすることが含まれます。

### 手順

1. RESUME TPIPE 呼び出しを使用して、コールアウト要求をリトリートします。

新しい同期コールアウト要求をリトリブするとき、ユーザー作成の IMS Connect クライアントは通常、非常に長いタイムアウト値を使用するか、またはタイムアウトを使用しないで、RESUME TPIPE 呼び出しを発行します。RESUME TPIPE 要求の IRM で、クライアントはオプションとして、クライアント・アプリケーションが ICAL コールアウト・メッセージからの制御データをサポートすることを示すことができます。

IMS が RESUME TPIPE 呼び出しを受信するときに、同期コールアウト要求メッセージがキューに入れていると、OTMA はこれを IMS Connect を介してクライアントに送信します。キューにコールアウト・メッセージが存在しない場合、IMS Connect クライアントは受信状態で待機し、OTMA は出力キューにメッセージが到着すると即座にクライアントに次のコールアウト要求を送信します。

## 2. 同期コールアウト要求の受信の成功 (ACK) または失敗 (NAK) を確認します。

同期コールアウト要求の受信または拒否を確認するには、ユーザー作成の IMS Connect クライアントが ACK または NAK 応答を IMS に送信して、新しいコールアウト・メッセージのために出力キューを解放する必要があります。同期コールアウト要求または肯定応答メッセージ自体のいずれかに指定されたタイムアウト値が満了しても ACK または NAK が受信されない場合、OTMA はそのコールアウト要求を破棄し、出力キューを解放します。

## 3. 送信専用プロトコルの同期コールアウト・オプションを使用して、IMS Connect に応答メッセージを返します。この応答には、IMS Connect が元のコールアウト要求に組み込んだ相関関係子トークンを含める必要があります。

応答は、IMS アプリケーションが要求したデータまたはエラー・メッセージのいずれかです。

## 次のタスク

IMS Connect クライアントを構成するための上記のステップに加えて、ご使用のシステムを同期コールアウト・サポート用に構成するには、以下の追加ステップが必要です。

- DL/I ICAL 呼び出しを発行して同期コールアウト要求を開始するように、IMS アプリケーション・プログラムをコーディングします。オプションとして、ICAL 呼び出しに制御データを追加して、ポートの URL、ユーザー・トークン、セキュリティー情報、その他の任意の情報を指定できます。
- 同期コールアウト要求を IMS Connect クライアントに送付するように、OTMA 宛先記述子をコーディングします。

以下の図は、IMS とユーザー作成 IMS Connect クライアントとの間の同期コールアウト要求のフローを示しています。

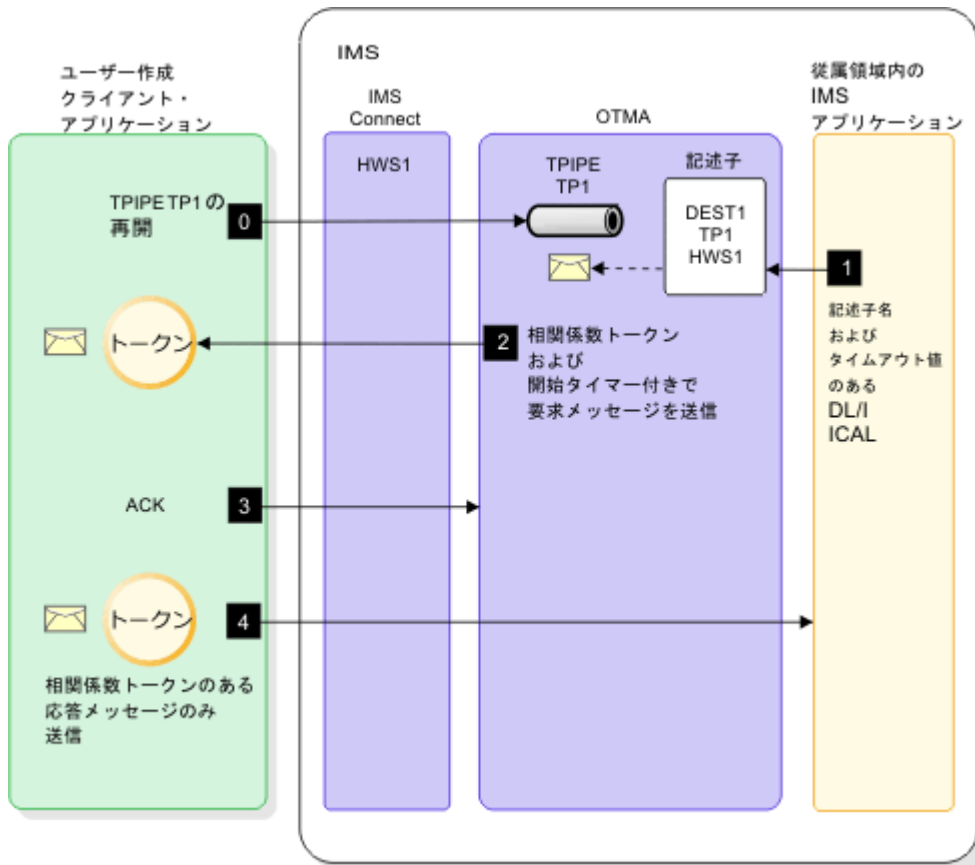


図 24. IMS アプリケーションとユーザー作成 IMS Connect クライアントとの間の同期コールアウト要求のフロー

### 関連タスク

212 ページの『RESUME TPIPE による同期コールアウト要求のリトリブ』

RESUME TPIPE 呼び出しを発行して、同期コールアウト要求をリトリブする場合、同期コールアウト・メッセージのみ、または同期コールアウト・メッセージと非同期出力の両方をリトリブするように、RESUME TPIPE 呼び出しをコーディングすることができます。

## 同期コールアウト・メッセージのフォーマット

同期コールアウト・メッセージには、相関係数トークンのセグメントが収められます。相関係数トークンは、コールアウト要求を発行する IMS アプリケーション・プログラムに応答を相関させるために使用されます。ユーザー提供のクライアント・アプリケーション・プログラムは、コールアウト要求メッセージの読み取り時に、このトークンを取得してコールアウト応答メッセージと一緒に返す必要があります。

相関トークンの存在によって、IMS Connect クライアントへのメッセージが同期コールアウト・メッセージであると識別されます。IMS は相関トークンを生成し、これを出力コールアウト要求に自動的に組み込みます。IMS Connect クライアントは、IMS に送信する応答で相関トークンを返す必要があります。

同期コールアウト・メッセージのアプリケーション・データ・セグメントの長さは、32 キロバイトよりも大きくなる場合があります。場合によっては何メガバイトにもなることがあります。IMS Connect では、4 バイトの長さフィールドである LLLL のセグメントの先頭に、アプリケーション・データ・セグメントの長さが含まれます。LLLL フィールドに指定される長さには、LLLL フィールド自体の長さも含まれます。IMS Connect クライアントは、長さフィールドの 4 バイトすべてを読み取るように構成する必要があります。

IMS Connect クライアントは、コールアウト要求に対する応答を返すときに、同じ LLLL フォーマットを使用して、セグメントの先頭の 4 バイトの長さフィールドにアプリケーション・データ・セグメントの長さを組み込む必要があります。



**重要:** 複数の ICAL 呼び出しが同時に発行される場合、メッセージ・サイズが大きくなると、それだけ拡張専用ストレージの消費量も大きくなる場合があります。

以下の例は、同期コールアウト要求および同期コールアウト応答のフォーマットを示しています。これらの例で、中括弧{および}で囲まれているセグメントはオプションです。

関連トークン自体の構造は変更できません。ただし、その構造は HWSIMSCB および HWSOMPFX マクロで定義されています。

IMS Connect がクライアント・アプリケーションに送信する同期コールアウト要求メッセージの構造は、以下のフォーマットに準拠します。

```
LLLL | LLZZ COR | {LLZZ RMM} | {Control data} | LLLL data | LLZZ CSM
```

クライアント・アプリケーションが返す同期コールアウト応答の構造は、以下のフォーマットに準拠する必要があります。

```
LLLL | LLZZ IRM | LLLL data | 00040000
```

比較用として、同期コールアウト要求以外の、4 バイト・フォーマットの長さフィールドをサポートしないメッセージの場合は、アプリケーション・データ・セグメントは、以下のようになります。

```
LLLL | {RMM} | LLZZ data | {LLZZ data} | CSM
```

## 関連資料

[249 ページの『メッセージ出口からクライアントへの出力メッセージ』](#)

メッセージ構造のフォーマットは、IMS Connect クライアントのタイプとクライアントのサポートに使用されるユーザー・メッセージ出口によって異なります。

[229 ページの『HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口ルーチンの IRM のユーザー部分のフォーマット』](#)

HWSSMPL0、HWSSMPL1、またはユーザー作成メッセージ出口によってサポートされるユーザー作成クライアント・アプリケーションでは、IMS Connect クライアント入力メッセージ内の IMS 要求メッセージ (IRM) ヘッダーの 4 バイトの長さフィールドと 28 バイトの固定部分に続けて、IRM にユーザー定義セクションを組み込むことができます。

[285 ページの『IMS Connect で使用される OTMA ユーザー・データ・フィールド』](#)

OTMA ヘッダー内のユーザー・データ・フィールドのフォーマットは、HWSOMPFX マクロ内の HWSOMUSR DSECT によって定義されており、すべての IMS Connect メッセージに共通です。

[257 ページの『メッセージ内のコールアウト制御データのフォーマット』](#)

クライアント・アプリケーションに送信される出力メッセージ内のコールアウト制御データのフォーマットは、resume tpipe 要求を発行するクライアント・アプリケーションのタイプによって異なります。

## RESUME TPIPE による同期コールアウト要求のリトリブ

RESUME TPIPE 呼び出しを発行して、同期コールアウト要求をリトリブする場合、同期コールアウト・メッセージのみ、または同期コールアウト・メッセージと非同期出力の両方をリトリブするように、RESUME TPIPE 呼び出しをコーディングすることができます。

### このタスクについて

同期コールアウト・メッセージの RESUME TPIPE 呼び出しでは、IRM\_F0 フィールドに IRM アーキテクチャ 3 (IRM\_ARCH3) および X'80' (IRM\_F0\_SYNONLY) または X'40' (IRM\_F0\_SYNASYN) の値が必要です。IRM\_F0 フィールドにオプションが指定されない場合、RESUME TPIPE 呼び出しは非同期出力のみをリトリブします。

IRM\_F0\_SYNONLY は、IMS Connect クライアントが同期コールアウト要求のみをリトリブできることを指定します。IRM\_F0\_SYNONLY が指定された RESUME TPIPE 呼び出しを受信したときに、T パイプ保留キューに非同期出力メッセージが存在する場合、その非同期メッセージはクライアントに返されずに、T パイプ保留キューに残ります。

IRM\_F0\_SYNASYN は、IMS Connect クライアントが同期コールアウト要求とキュー上の他のあらゆる非同期出力メッセージの両方をリトリブできることを指定します。IRM\_F0\_SYNASYN を指定すると、新しい

RESUME TPIPE 呼び出しが受信されたときに、OTMA は最初にすべての同期コールアウト要求を送信してから、非同期出力メッセージを送信します。

クライアントからの RESUME TPIPE 呼び出しが、同期コールアウト要求を開始して処理したものと異なる IMS 共用キュー・メンバーに接続された場合、クライアントは、たとえスーパーメンバー機能が活動化されているときでも、メッセージを受信しません。メッセージはタイムアウトが発生するまで OTMA 保留キューに保持され、後で削除されます。同期コールアウト要求は、T パイプ保留キューに入れられるため、その T パイプを所有する IMS によってのみ認識されます。スーパーメンバー機能は、複数の IMS Connect クライアントが同じ IMS 共用キュー・メンバーに接続されているときにのみ、同期コールアウト要求に適用されます。

## 手順

- IRM 接頭部の以下のフィールドの指定で、同期コールアウト・メッセージの RESUME TPIPE 呼び出しを定義します。

### IRM\_ARCH

X'03' (IRM\_ARCH3).

### IRM\_F0

以下のいずれかを指定します。

- 同期コールアウト・メッセージのみをリトリブする場合は、X'80' (IRM\_F0\_SYNONLY)
- 同期コールアウト・メッセージと非同期出力メッセージの両方をリトリブする場合は、X'40' (IRM\_F0\_SYNASYN)

### IRM\_F4

R 文字値 (IRM\_F4\_RESUMET)

### IRM\_F5

RESUME TPIPE 呼び出しのリトリブ・オプション (X'02' (IRM\_F5\_AUTO) など)

クライアント・アプリケーションが制御データをサポートする場合は、ビット X'20' (IRM\_F5\_CTLDATA) もオンにします。

### IRM\_F6

クライアントがネットワーク・セキュリティー資格情報を受け取るようにする必要がある場合は、X'80' (IRM\_F6\_NWSE) を指定して、クライアントが \*NETSID\* および \*NETUID\* の出力メッセージ・セグメントをサポートすることを示します。

### IRM\_TIMER

RESUME TPIPE 呼び出しのタイムアウト値。IRM\_TIME\_FF などと指定します。このように指定すると RESUME TPIPE 呼び出しはタイムアウトになりません。

## 関連概念

[339 ページの『出力メッセージのリトリブの管理』](#)

RESUME TPIPE 呼び出しで非同期出力メッセージまたは同期コールアウト要求メッセージをリトリブする場合、メッセージを返す方法について、いくつかのオプションがあります。

[320 ページの『入力メッセージのタイムアウトの指定』](#)

IMS Connect クライアントからの個々の入力メッセージは、いずれも IMS 要求メッセージ (IRM) ヘッダーの固定部分の IRM\_TIMER フィールドに異なるタイムアウト値を設定できます。

[309 ページの『ソケット接続』](#)

IMS Connect は、3 種類のクライアント TCP/IP 接続プロトコルを提供しています。これらをソケットと呼びます。TCP/IP ソケットは、IMS Connect が切断メッセージを送信するときに IMS Connect がクライアント TCP/IP 接続を管理する方法を定義します。

## 関連資料

[223 ページの『IMS Connect メッセージ構造』](#)

z/OS プログラム呼び出しインターフェースを使用する TCP/IP クライアントは、各入力メッセージの IMS 要求メッセージ (IRM) ヘッダーを使用して、IMS Connect と通信します。プロトコル・オプションを IMS Connect に伝えるため、IMS Connect クライアント・アプリケーション・プログラムからの入力メッセー



ジで IRM ヘッダーが使用されます。IRM ヘッダーは、IMS Connect HWSIMSCB マクロによってマップされます。

#### 249 ページの『メッセージ出口からクライアントへの出力メッセージ』

メッセージ構造のフォーマットは、IMS Connect クライアントのタイプとクライアントのサポートに使用されるユーザー・メッセージ出口によって異なります。

## RESUME TPIPE のエラー・シナリオ

IRM\_FO フィールドを IRM\_FO\_SYNONLY または IRM\_FO\_SYNASYN のいずれかに設定しているが、IRM\_ARCH が IRM\_ARCH3 以上の状態で、IMS Connect クライアントが同期コールアウト・メッセージの RESUME TPIPE 呼び出しを発行した場合、IMS Connect クライアントは以下の戻りコードと理由コードを RSM 構造で IMS Connect クライアントに返します。

- RSM\_RETCOD = RSMRTC\_EXIT X'04'
- RSM\_RSNCOD = RSMRSN\_INVBUF X'09'

IMS Connect が同期コールアウト・メッセージのために使用されている T パイプ上のメッセージをリトリートしたが、メッセージを IMS Connect クライアントに送信できない場合、IMS Connect は OTMA に対して NAK メッセージで応答します。OTMA は NAK メッセージの受信後、IMS Connect クライアントの状況、および配信不能メッセージが同期コールアウト・メッセージであるか、または非同期出力メッセージであるかに応じて、メッセージを処理します。

非同期出力メッセージの場合、OTMA はその非同期出力メッセージを T パイプ保留キューに返します。

同期コールアウト・メッセージの場合は、以下の処理を実行します。

- IMS Connect が、クライアントのタイムアウトまたは切断が原因で同期コールアウト・メッセージを送信できなかった場合、OTMA は、別の RESUME TPIPE 呼び出しによってリトリートできるように、あるいは、ICAL 呼び出しに関して指定されているタイムアウト間隔が満了するまで、その同期コールアウト・メッセージをキューに保持します。
- IMS Connect が、TCP/IP エラーなどのタイムアウトおよび切断以外の理由で同期コールアウト・メッセージを送信できなかった場合、OTMA はその同期コールアウト・メッセージを廃棄します。

## 同期コールアウト・メッセージの受信の応答

ユーザー作成 IMS Connect クライアントは、OTMA が同期コールアウト要求メッセージを送信した後に、肯定応答 (ACK) または否定応答 (NAK) メッセージを IMS に送信する必要があります。ACK または NAK メッセージが受信されるまで、または確認応答または元の要求のいずれかに対してタイムアウト間隔が満了するまで、T パイプ・キューは待ち状態 (WAIT\_S) のままとなり、他のメッセージを送受信することはできません。

### このタスクについて

ACK メッセージが受信されると、T パイプ・キューが解放され、他のコールアウト・メッセージを送受信できるようになります。IMS アプリケーション・プログラムは、同期コールアウト・メッセージに対する応答が受信されるか、または同期コールアウト要求がタイムアウトになるまで、従属領域で待機し続けます。

NAK メッセージが受信されると、T パイプ・キューが解放されるだけでなく、OTMA に、同期コールアウト要求メッセージの処理内容と、現在の RESUME TPIPE 呼び出しを継続するかまたは終了するかが通知されます。

IMS Connect クライアントは、NAK メッセージに関して以下のオプションを指定できます。

- 拒否された同期コールアウト要求メッセージを廃棄し、RESUME TPIPE 接続を終了します。
- 拒否された同期コールアウト要求メッセージを廃棄しますが、RESUME TPIPE 接続は維持し、他の同期コールアウト要求メッセージのリトリートを続行します。
- 拒否された同期コールアウト要求メッセージを T パイプ・キューに保持しますが、RESUME TPIPE 接続は終了します。

## 関連概念

328 ページの『OTMA への IMS Connect の確認応答のタイムアウト間隔』

OTMA が、IMS Connect からの確認応答を待機する時間を決定するタイムアウト間隔を指定することができます。また、タイムアウト T パイプ・キューを指定して、タイムアウト間隔が経過した後に、コミット後送信 (CM0) 出力を保持することができます。

## コールアウト・メッセージを廃棄して RESUME TPIPE 呼び出しを終了する場合の NAK メッセージのコーディング

NAK メッセージをコーディングして、拒否された同期コールアウト・メッセージを廃棄し、RESUME TPIPE 呼び出しを終了するように OTMA に指示するには、以下のようになります。

### 手順

1. IRM\_F4 フィールドに N を指定します
2. オプション: 以下を指定して、拡張エラー・コードを含めます。

- IRM\_F0 = X'10'
- IRM\_NAK\_RSNCDE = 2 バイト 16 進数拡張エラー・コード

拡張エラー・コードは、ICAL 呼び出しの AIB の AIBERRXT フィールドを使用して、IMS アプリケーションに返されます。

### タスクの結果

この NAK メッセージを受信すると、OTMA は IMS アプリケーション・プログラムに戻りコード X'100' および理由コード X'108' を発行します。

非同期出力メッセージに対する NAK の場合、その非同期出力は T パイプ保留キューに返されます。

## メッセージは破棄するが接続は保持する場合の NAK メッセージのコーディング

NAK メッセージをコーディングして、拒否された同期コールアウト・メッセージを廃棄するが、RESUME TPIPE 呼び出しは保持して他の同期コールアウト要求メッセージをリトリブするように OTMA に指示するには、以下のようになります。

### 手順

1. NAK メッセージの IRM に、以下のフィールド値を指定します。

- IRM\_F4 = N
  - IRM\_F3 = X'08' (IRM\_F3\_REROUT)
2. オプション: 以下を指定して、拡張エラー・コードを含めます。
- IRM\_F0 = X'10'
  - IRM\_NAK\_RSNCDE = 2 バイト 16 進数拡張エラー・コード

拡張エラー・コードは、ICAL 呼び出しの AIB の AIBERRXT フィールドを使用して、IMS アプリケーションに返されます。

### タスクの結果

この NAK メッセージを受信するとき、IMS アプリケーション・プログラムは、戻りコード X'100' および理由コード X'108' を受信します。

非同期出力メッセージに対する NAK の場合、その非同期出力は返されて T パイプ保留キューに転送されません。



## メッセージは保持するが RESUME TPIPE 呼び出しは終了する場合の NAK メッセージのコーディング

NAK メッセージをコーディングして、拒否された同期コールアウト・メッセージを T パイプ・キューに保持するが、RESUME TPIPE 呼び出しは終了するように OTMA に指示するには、以下のようになります。

### このタスクについて

#### 手順

- NAK メッセージの IRM に、以下のフィールド値を指定します。
  - IRM\_F4 = N
  - IRM\_F0 = X'20' (IRM\_F0\_SYNCNAK)

#### タスクの結果

NAK メッセージを受信すると、RESUME TPIPE 呼び出しは終了しますが、OTMA は同期コールアウト要求メッセージを T パイプ・キューに保持します。これで、拒否された同期コールアウト要求がタイムアウトになるまで、OTMA が別の RESUME TPIPE 呼び出しを受信すると、その呼び出しによって、このコールアウト要求メッセージをリトリブすることができます。

非同期出力メッセージに対する NAK の場合、OTMA はそのメッセージを保留キューに返し、RESUME TPIPE 呼び出しを終了します。

## IMS に対するコールアウト応答のリターン

ユーザー作成の IMS Connect クライアントは、同期コールアウト応答用の送信専用プロトコルを使用して、コールアウト応答メッセージを IMS に返します。

### このタスクについて

同期コールアウト応答用の送信専用プロトコルを使用する場合、コールアウト応答の受信を確認するように IMS を構成することも、実際の送信専用の動作に対して確認応答を抑止することもできます。

この送信専用メッセージにはトランザクション・コードは含まれておらず、IMS アプリケーション・プログラムの応答データかエラー情報のいずれかが含まれる可能性があります。

同期コールアウト要求に対する応答は、元の DL/I ICAL 呼び出しが発行された同じ IMS システムに返す必要があります。ただし、応答は、元のコールアウト要求を処理したのと同じ IMS Connect クライアントが返す必要はありません。応答メッセージは、アウトバウンド・コールアウト要求を処理したインスタンス以外の IMS Connect インスタンスを介して、IMS に返すこともできます。応答メッセージに含まれる相関トークンによって、応答は、応答を待機している正しい IMS アプリケーション・プログラムに確実に返されます。

IMS Connect クライアントによって返される応答メッセージの最大長は、HWS 構成ステートメントの MAXSIZE パラメーターで指定される長さ、または IMS Connect のデフォルト値 10 MB です。

IMS Connect クライアントは、返されるメッセージが同期コールアウト応答であることを、メッセージ接頭語の IRM\_F4 フィールドで示す必要があります。

クライアントは、IMS がコールアウト応答を受信した後に確認応答を返すよう IMS に対して要求する場合、IRM メッセージ接頭語の IRM\_F4 フィールドに L を指定し、応答の送信後に受信状態に切り替える必要があります。

クライアントが IMS による確認応答の返信を要求しない場合、クライアントは IRM\_F4 フィールドに M を指定する必要があります。

クライアントが同期コールアウト応答を IMS Connect に返したが、IRM\_ARCH が IRM\_ARCH3 よりも小さいか、相関トークンが IRM\_CORTKN フィールドに含まれていない場合、IMS Connect はその応答を拒否し、同期コールアウト要求メッセージはタイムアウトになります。

IMS がコールアウト応答を確認する場合、IMS Connect は、送達が成功した場合は CSM (ACK) を返し、失敗した場合は RSM (NAK) を返します。確認応答が無効にされている場合、IMS Connect はクライアントに RSM メッセージを返しません。

## 手順

コールアウト応答をコーディングするには、以下のようにします。

- 応答メッセージの IRM に、以下のフィールド値を指定します。
    - IRM\_ARCH = X'03' (IRM\_ARCH3)
    - IRM\_F4 = L (IRM\_F4\_SYNRESPA) または M (IRM\_F4\_SYNRESP)
    - IRM\_CORTKN = 元のコールアウト要求に付属していた 40 バイトの相関トークン (CORTKN)
- 同期コールアウトのエラー応答のコーディングについては、[217 ページの『IMS に対するエラー応答のリターン』](#)を参照してください。

## 関連概念

308 ページの『同期コールアウト応答のための送信専用プロトコル』

IMS Connect クライアントは、送信専用プロトコルを使用して、IMS アプリケーション・プログラムからの同期コールアウト要求に対する応答を返します。

## 関連資料

223 ページの『IMS Connect メッセージ構造』

z/OS プログラム呼び出しインターフェースを使用する TCP/IP クライアントは、各入力メッセージの IMS 要求メッセージ (IRM) ヘッダーを使用して、IMS Connect と通信します。プロトコル・オプションを IMS Connect に伝えるため、IMS Connect クライアント・アプリケーション・プログラムからの入力メッセージで IRM ヘッダーが使用されます。IRM ヘッダーは、IMS Connect HWSIMSCB マクロによってマップされます。

## IMS に対するエラー応答のリターン

ユーザー作成 IMS Connect クライアントが既に ACK メッセージを IMS Connect に返した後にエラーが発生し、IMS Connect クライアントあるいはデータまたはサービス・プロバイダーがコールアウト要求を完了できない場合、IMS Connect クライアントは、要求されたデータの代わりに、エラー応答を返すことができます。

## このタスクについて

DL/I ICAL 呼び出しに対するエラー応答をコーディングするには、以下のようにします。

## 手順

エラー応答の IRM 接頭部に、以下のフィールド値を指定します。

- IRM\_ARCH = X'03' (IRM\_ARCH3)
- IRM\_F0 = X'10' (IRM\_F0\_NAKRSN): NAK 理由コードがエラー応答と一緒に送信される場合
- IRM\_F0 = X'20' (IRM\_F0\_SYNCNAK): NAK 理由コードがエラー応答と一緒に送信されない場合
- IRM\_NAK\_RSNCDE = 2 バイト 16 進数拡張エラー・コード
- IRM\_F4 = L (IRM\_F4\_SYNRESPA): IMS が確認応答を返すことをクライアントが要求する場合
- IRM\_F4 = M (IRM\_F4\_SYNRESP): IMS が確認応答を返すことをクライアントが要求しない場合
- IRM\_CORTKN = 元の DL/I ICAL 呼び出しによって生成された 40 バイトの相関トークン

## 関連概念

306 ページの『送信専用プロトコル』

クライアント・アプリケーション・プログラムは、応答を待機することなく、コミット後送信 (CM0) 入力メッセージを IMS に対し連続して高速にサブミットするために送信専用プロトコルを使用します。送信専用プロトコルは、高速で大ボリュームの入力用に設計されています。

#### **関連資料**

223 ページの『[IMS Connect メッセージ構造](#)』

z/OS プログラム呼び出しインターフェースを使用する TCP/IP クライアントは、各入力メッセージの IMS 要求メッセージ (IRM) ヘッダーを使用して、IMS Connect と通信します。プロトコル・オプションを IMS Connect に伝えるため、IMS Connect クライアント・アプリケーション・プログラムからの入力メッセージで IRM ヘッダーが使用されます。IRM ヘッダーは、IMS Connect HWSIMSCB マクロによってマップされます。

## 第 15 章 IMS Connect XML メッセージ変換

一部の IMS Connect クライアント・アプリケーション・プログラムでは、IMS Connect が XML メッセージを COBOL または PL/I に変換できるため、IMS に送信される XML メッセージを処理するために既存の IMS アプリケーション・プログラムを変更する必要はありません。

IMS Connect 変換サポートは、XML ユーザー・データを IMS アプリケーション・プログラムが想定する言語構造に変換し、IMS アプリケーション・プログラムは、それを通常の IMS トランザクション・メッセージとして処理できます。

IMS Connect XML 変換サポートは、COBOL XML コンバーターと共に、XML アダプター機能および HWSOAP1 メッセージ出口によって提供されます。

XML メッセージの実際の変換は、IMS Connect が呼び出す XML コンバーターによって実行されます。

IMS Connect がサポートする XML 変換が可能な言語とクライアントを以下に示します。

### COBOL

- IMS Enterprise Suite SOAP Gateway
- IMS Web 2.0 Solution for IBM Mashup Center

### PL/I

- IMS Web 2.0 Solution for IBM Mashup Center

IMS Connect は、Web サービス・プロバイダーとして機能し、かつ複数の言語構造を使用する IMS アプリケーション・プログラムと SOAP Gateway の間での複数セグメント・メッセージの変換もサポートします。複数セグメント・メッセージ・サポートには、いくつかの制約事項が適用されます。これらの制約事項についての詳細は、SOAP Gateway および IBM Developer for System z® の資料を参照してください。

IMS Connect HWSOAP1 出口ルーチンも XML 変換をサポートしており、OTMA 宛先記述子に XML アダプターおよび XML コンバーターの名前を指定できます。

入力メッセージの IMS 要求メッセージ (IRM) ヘッダーで、IMS Connect クライアントは XML アダプター名と XML コンバーター名を指定することで、XML 変換サポートを要求します。IMS Connect は、入力メッセージを受信したときに IRM を読み取り、XML 変換が必要な場合は XML アダプターを呼び出します。すると、XML アダプターが XML コンバーターを呼び出して、実際の変換を実行します。メッセージが XML フォーマットから変換されると、IMS Connect は得られたメッセージを IMS に送信します。

IMS Connect は、応答メッセージを IMS から受信すると XML アダプターを呼び出し、IMS Connect の XML アダプターが XML コンバーターを呼び出して、応答メッセージ内のユーザー・データを XML に変換します。その後、IMS Connect は出力メッセージを IMS Connect クライアントに送信します。

**同期コールアウト・メッセージ内での制御データの使用による XML コンバーター名のオーバーライド：**  
SOAP Gateway によって送信されるアウトバウンド同期コールアウト要求では、IMS Connect によって使用される XML コンバーター名をオーバーライドできます。このためには、IMS アプリケーション・プログラムがコールアウト要求を発行する際に、そのアプリケーション・プログラムで DL/I ICAL 呼び出しの制御データ域に別のコンバーター名を指定します。制御データ域でコンバーター名を指定する際に、その名前は大文字の EBCDIC 文字で指定し、<DFSCVTNR> タグと </DFSCVTR> タグで囲む必要があります。

### 関連タスク

[IMS Connect クライアントの XML 変換サポートの構成 \(システム定義\)](#)

### 関連資料

[ICAL 呼び出し \(アプリケーション・プログラミング API\)](#)

## IMS Connect の XML コンバーター

IMS Connect の XML 変換サポートを使用可能にする場合、データ構造を XML から IMS アプリケーション・プログラムで使用されるプログラミング言語に変換するために、およびその逆の変換を行うために、IMS Connect によって使用される XML コンバーターを作成する必要があります。

XML コンバーターは、アプリケーション・プログラムです。別途ライセンス交付を受けるツールである IBM Developer for System z を使用すると、COBOL コピーブックまたは PL/I ソース・コードから自動的に XML コンバーターを生成することができます。

#### 推奨事項:

COBOL アプリケーション・プログラムでは、XML コンバーターはメッセージを処理する IMS COBOL アプリケーション・プログラムの COBOL コピーブックに基づいています。PL/I アプリケーション・プログラムでは、XML コンバーターは PL/I アプリケーション・プログラムのソースに基づいています。XML から変換されたメッセージを処理するそれぞれの IMS アプリケーションには、独自の固有な XML コンバーターが必要です。

XML コンバーターは、IMS Connect 領域の IBM Language Environment® for z/OS エンクレーブ内で実行されます。必要なスペース量は、ご使用環境に必要な XML コンバーターのサイズと数によって異なります。このストレージ要件に対応できるように、IMS Connect の領域サイズを増やす必要があります。

IMS PROCLIB データ・セットの構成メンバー HWSCFGxx に指定された MAXCVRT 構成パラメーターを使用して、IMS Connect のこのインスタンスが同時にロードできる XML コンバーターの最大数を指定できます。ご使用のアプリケーションが MAXCVRT に指定された XML コンバーターの最大数を超えると、IMS Connect は、最初に、最長未使用時間 XML コンバーターをアンロードします。

XML コンバーターは、BPE 出口リスト・メンバーとして定義されます。XML コンバーターを最初に作成した後、IMS Connect を再始動せずに、以下のいずれかのコマンドを使用して、そのコンバーターを更新およびリフレッシュすることができます。

- WTOR コマンド **REFRESH CONVERTER**
- z/OS Modify コマンド **UPDATE CONVERTER**
- タイプ 2 コマンド **UPDATE IMSCON TYPE(CONVERTER) NAME(converter\_name) OPTION(REFRESH)**

#### 関連タスク

[IMS Connect クライアントの XML 変換サポートの構成 \(システム定義\)](#)

#### 関連資料

[ADAPTER ステートメント \(システム定義\)](#)

## XML メッセージの構造

XML スキーマでは、COBOL IMS アプリケーション・プログラムで使用される COBOL データ構造に対応する XML タグが定義されます。XML 変換に使用される XML スキーマは、COBOL IMS アプリケーション・プログラムの COBOL コピーブックに基づいています。

XML スキーマは IMS Connect で必須ではありませんが、XML 入力メッセージを生成する Web サービスを開発するアプリケーション・プログラマーにとって必要です。

IBM Developer for System z を使用して、IMS アプリケーション・プログラムの COBOL コピーブックから XML コンバーターを自動生成する場合は、IBM Developer for System z によって XML スキーマも生成されます。

XML スキーマで定義される XML タグは、COBOL データ構造のフィールドに直接対応しています。

サンプルの電話帳アプリケーション・プログラムの COBOL コピーブックで定義される、入力および出力メッセージ・データ構造の例を次に示します。このアプリケーション・プログラムは、IMS Enterprise Suite SOAP Gateway ダウンロード・サイト ([www.ibm.com/software/data/ims/soap/](http://www.ibm.com/software/data/ims/soap/)) から入手できます。

```
01 INPUT-MSG.
   02 IN-LL          PICTURE S9(3) COMP.
   02 IN-ZZ          PICTURE S9(3) COMP.
   02 IN-TRCD        PICTURE X(10).
   02 IN-CMD         PICTURE X(8).
   02 IN-NAME1       PICTURE X(10).
   02 IN-NAME2       PICTURE X(10).
   02 IN-EXTN        PICTURE X(10).
   02 IN-ZIP         PICTURE X(7).

01 OUTPUT-MSG.
```

02	OUT-LL	PICTURE S9(3) COMP.
02	OUT-ZZ	PICTURE S9(3) COMP.
02	OUT-MSG	PICTURE X(40).
02	OUT-CMD	PICTURE X(8).
02	OUT-NAME1	PICTURE X(10).
02	OUT-NAME2	PICTURE X(10).
02	OUT-EXTN	PICTURE X(10).
02	OUT-ZIP	PICTURE X(7).
02	OUT-SEGNO	PICTURE X(4).

コピーブックの各フィールドには、それに相当する XML タグがあり、XML メッセージ内のフィールドを表しています。XML タグでは大文字と小文字が区別されます。コピーブック内のフィールド名にあるダッシュ記号「-」は、対応する XML タグ内で下線「\_」で表されます。SOAP Gateway クライアントは、その XML スキーマを使用して XML メッセージを構築する必要があります。

例えば、上記のコピーブックにある IN-TRCD フィールドは、XML ではオープン・タグ <in\_trcd> とクローズ・タグ </in\_trcd> で表されます。データ構造入力メッセージでは、IN-TRCD の値がバイト位置 5 から 14 に配置されなければなりません。XML 入力メッセージでは、IN-TRCD の同じ値が、タグ <in\_trcd> と </in\_trcd> の間に配置されなければなりません。

上記の電話帳アプリケーションの COBOL コピーブックの例を使用する IMS Connect クライアントからの入力メッセージには、相当する COBOL データ構造フィールドに対する以下の XML タグがあります。

```
<INPUTMSG>
<in_ll> </in_ll>
<in_zz> </in_zz>
<in_trcd> </in_trcd>
<in_cmd> </in_cmd>
<in_name1> </in_name1>
<in_name2> </in_name2>
<in_extn> </in_extn>
<in_zip> </in_zip>
</INPUTMSG>
```

上記の電話帳アプリケーションの COBOL コピーブックには、相当するアウトバウンド COBOL データ構造フィールドに対する以下の XML タグがあります。

```
<cb1:OUTPUTMSG>
<out_ll> </out_ll>
<out_zz> </out_zz>
<out_msg> </out_msg>
<out_cmd> </out_cmd>
<out_name1> </out_name1>
<out_name2> </out_name2>
<out_extn> </out_extn>
<out_zip> </out_zip>
<out_segno> </out_segno>
</cb1:OUTPUTMSG>
```

入力メッセージの XML タグは、COBOL コピーブック内の 01 INPUT-MSG. 定義に基づいた XML のオープン・タグおよびクローズ・タグで囲まれている必要があります。上記の例で、オープン・タグとクローズ・タグは <INPUTMSG> と </INPUTMSG> です。

アウトバウンド・コンバーターで生成された出力メッセージの XML タグは、01 OUTPUT-MSG. 定義に対応する XML のオープン・タグおよびクローズ・タグで囲まれます。上記の例で、オープン・タグとクローズ・タグは <cb1:OUTPUTMSG> と </cb1:OUTPUTMSG> です。

各 XML コンバーターでこれらのタグがどのようになるべきかは、XML スキーマを見て確認してください。

データ構造メッセージで必ずしもすべてのフィールドが必要ではないのと同様に、各 XML メッセージにすべてのタグを指定する必要はありません。必要なフィールドは、COBOL アプリケーションによって決まります。

## メッセージ変換の例

この例のコードでは、XML 入力メッセージから COBOL への変換、およびその COBOL 応答メッセージから XML への変換について説明します。

入力の XML メッセージの例を、次に示します。

```
<INPUTMSG><in_ll>32</in_ll><in_zz>0</in_zz>  
<in_trcd>IVTNO</in_trcd><in_cmd>DISPLAY</in_cmd>  
<in_name1>LAST1</in_name1></INPUTMSG>
```

XML データ構造を、COBOL 電話帳アプリケーション・プログラムで必要な COBOL データ構造に変換した後の、同じメッセージの例を次に示します。

```
IVTNO      DISPLAY LAST1
```

**注:** 上記の例で、IVTNO は 5 バイト目から始まっています。最初の 4 バイトは IMS Connect で使用され、LAST1 の後ろには 5 つのスペースがあります。

データ構造メッセージの最初の 4 バイトは LLZZ と呼ばれ、XML アダプターによって埋められます。最初の 2 バイトの LL 部分には、データ構造メッセージの長さが入ります。その次の 2 バイトの ZZ 部分には、ゼロが入ります。XML コンバーターは、各 XML タグ内の値を取得し、それを対応するフィールド位置に入れることで、XML メッセージをアプリケーション固有のフォーマットに変換します。変換後のメッセージは COBOL アプリケーションで処理することができ、COBOL アプリケーションは固有のデータ構造の出力メッセージを戻します。この出力メッセージは、クライアントに戻す前に、XML に変換する必要があります。出力メッセージと、XML 変換後のメッセージの例を、次に示します。

IMS データ・ストアからの出力にある応答メッセージの COBOL データ構造の例を、次に示します。

```
ENTRY WAS DISPLAYED      DISPLAY LAST1      FIRST1      8-111-1111D01/R010001
```

上記の例で、出力メッセージの最初の行のテキストは、4 バイトの LLZZ フィールドに続く 5 バイト目から始まっています。

COBOL データ構造を IMS Connect クライアントで必要な XML に変換した後の、同じメッセージの例を次に示します。

```
<cb1>  
  
<out_ll> 093</out_ll><out_zz> 000</out_zz>  
<out_msg>ENTRY WAS DISPLAYED</out_msg><out_cmd>DISPLAY</out_cmd>  
<out_name1>LAST1</out_name1><out_name2>FIRST1</out_name2>  
<out_extn>8-111-1111</out_extn><out_zip>D01/R01</out_zip>  
<out_segno>0001</out_segno></cb1>
```

上記の例で、XML のオープン・タグ <out\_ll> と <out\_zz> タグに続く値は、出力 COBOL データ構造メッセージの最初の 4 バイトの値です。

IMS アプリケーションからのアウトバウンド・メッセージでは、変換が逆に行われます。アプリケーション・データ構造内の各フィールドは、対応する XML タグで囲まれます。この XML から COBOL アプリケーション・データ構造メッセージ・フォーマットへの変換およびその逆変換は、XML アダプターから呼び出された COBOL XML コンバーターで行われます。



## 第 16 章 IMS Connect メッセージ構造

z/OS プログラム呼び出しインターフェースを使用する TCP/IP クライアントは、各入力メッセージの IMS 要求メッセージ (IRM) ヘッダーを使用して、IMS Connect と通信します。プロトコル・オプションを IMS Connect に伝えるため、IMS Connect クライアント・アプリケーション・プログラムからの入力メッセージで IRM ヘッダーが使用されます。IRM ヘッダーは、IMS Connect HWSIMSCB マクロによってマップされます。

また、IMS Connect は、OTMA メッセージ・ヘッダーを使用して z/OS システム間カップリング・ファシリティ セッションを介して OTMA と通信します。クライアント固有の IMS Connect ユーザー・メッセージ 出口ルーチンの基本タスクの 1 つは、入力時に IRM ヘッダーを OTMA ヘッダーに変換することです。

通信手段として TCP/IP ソケット呼び出しを使うクライアントは、以下のようにフォーマット間のメッセージを変換するために、IMS Connect で実行するユーザー・メッセージ 出口ルーチンを設計できます。

- クライアント・メッセージ・フォーマットを OTMA メッセージ・フォーマットに変換する。
- OTMA メッセージ・フォーマットの IMS 応答を、クライアント・メッセージ・フォーマットに変換する。
- メッセージのユーザー・データ部分の XML を COBOL に変換し、再び元の XML に変換する。

これらの変換によって、クライアントは、TCP/IP 接続を介して IMS データをリトリブできます。IMS Connect は、メッセージが正しくフォーマットされると、自動的にメッセージを送受信します。

IMS Connect クライアントのタイプおよびクライアントをサポートするために使用されるユーザー・メッセージ 出口に応じて、IRM メッセージ・ヘッダーのフォーマットは異なります。

以下のユーザー・メッセージ 出口ルーチンには、特定のメッセージ・フォーマットが必要です。

- HWSSMPL0 および HWSSMPL1
- HWSJAVA0
- HWSCSLO0 および HWSCSLO1
- HWSSOAP1

### 関連概念

163 ページの『IMS Connect 出口ルーチンの概要』

IMS は、IMS Connect をサポートするために、各種の出口ルーチンを提供しています。

### 関連資料

[IMS Connect 出口ルーチン \(出口ルーチン\)](#)

## IMS Connect クライアント・メッセージの IRM 構造

IMS Connect は、受信したすべてのクライアント・メッセージが合計 4 バイトの長さフィールドで始まり、その後に IMS 要求メッセージ (IRM) ヘッダー、メッセージ・データ・セグメントの順に続くものと想定します。

4 バイトの長さフィールドの後には、すべての IRM が 28 バイトの固定形式セクションで始まります。これはすべての IMS Connect クライアント・アプリケーションからのすべてのメッセージで共通しています。IRM の固定セクションのフォーマットは変更できません。

ユーザー作成のクライアント・アプリケーションでは、固定形式セクションの後の IRM にユーザー定義セクションを組み込むことができます。IRM のユーザー定義セクションを使用すると、追加オプションを指定することができます。IRM のユーザー定義セクションにフィールドを追加または削除する場合、使用するユーザー・メッセージ 出口を調整するか、またはユーザー独自のユーザー・メッセージ 出口を提供する必要があります。IRM のユーザー定義セクションは、HWSSMPL0 および HWSSMPL1 ユーザー・メッセージ 出口ルーチンでのみサポートされます。

IRM のユーザー定義セクションの後に、IRM 拡張を定義することができます。IRM 拡張を使用することで、IRM をマップする DSECT を拡張しなくても、IMS Connect クライアント・アプリケーションから IMS に情報を送信することができます。

ソースとして提供されていない HWSOAP1 ユーザー・メッセージ出口または HWSJAVA0 ユーザー・メッセージ出口を使用している場合、IRM のユーザー部分は変更できません。

注: ユーザー・メッセージ出口 HWSCSLO0 および HWSCSLO1 は、変更または置換することができないため、メッセージ定義は存在しません。

## IMS Connect へ送信されるメッセージ内の IRM の固定部分のフォーマット

IMS 要求メッセージ (IRM) ヘッダーには、IMS TM と通信するすべての IMS Connect クライアント・アプリケーションからのすべてのメッセージに共通の 28 バイトの固定フォーマット・セクションがあります。

以下の表は、クライアントから IMS Connect に送信される入力メッセージの先頭にある固定フォーマットを示しています。これには、メッセージ・フィールド、フィールド長、およびメッセージについての簡潔な説明が含まれています。

llll フィールドは、IMS Connect にメッセージの長さを知らせ、IRM は、データが渡される宛先となる特定のユーザー出口などの追加情報を提供します。

表 17. IRM 接頭部の固定部分

フィールド	長さ	オフセット		意味
		10 進	16 進	
llll	4 バイト	0	0	2 進数として読み取られる全メッセージの長さ。  合計メッセージ長には、llll フィールドの長さ (4 バイト)、IRM (可変長、ユーザーの要件に応じて異なる)、メッセージ長、およびメッセージ終結標識 X'00040000' の長さ (4 バイト) が含まれます。  すべてのメッセージの最小の長さは、X'58' です。IMS TM Resource Adapter の場合、最大長は X'00989680' (10,000,000 バイト) です。ユーザー作成のクライアント・アプリケーションの場合、最大長は X'7FFFFFFF' です。
以下のフィールドは、28 (10 進数) バイトの IRM 接頭部である。				
IRM_LEN	2 バイト	4	4	IRM 構造の長さ。ユーザー作成出口用の IRM 最小サイズは、X'24' すなわち 2 進数 '00100100'。HWSMPL0 では、最小 IRM 長として X'50' すなわち 2 進数 '01010000' が必要。

表 17. IRM 接頭部の固定部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_ARCH	1 バイト	6	6	<p>IMS Connect がクライアントから受信するメッセージの IRM 接頭部のアーキテクチャー・レベルを指定する。</p> <ul style="list-style-type: none"> <li>• X'00' IRM 接頭部のユーザー部分の基本アーキテクチャー構造、IRM_ARCH0 を指定する。</li> <li>• X'01' 以下のスペースを含む IRM 接頭部のユーザー部分のアーキテクチャー構造、IRM_ARCH1 を指定する。 <ul style="list-style-type: none"> <li>- IRM_REROUT_NM フィールド</li> <li>- IRM_RT_ALTCID フィールド</li> </ul> </li> <li>• X'02' 以下のスペースを含む IRM 接頭部のユーザー部分のアーキテクチャー構造、IRM_ARCH2 を指定する。 <ul style="list-style-type: none"> <li>- IRM_REROUT_NM フィールド</li> <li>- IRM_RT_ALTCID フィールド</li> <li>- IRM_TAG_ADAPT フィールド</li> <li>- IRM_TAG_MAP フィールド</li> </ul> </li> <li>• X'03' IRM_ARCH2 および以下の追加フィールドに含まれるすべてのフィールドのスペースを収めた IRM 接頭部のユーザー部分のアーキテクチャー構造、IRM_ARCH3 を指定する。 <ul style="list-style-type: none"> <li>- 同期コールアウト関連トークン・フィールド</li> <li>- MFS MOD の名前のための IRM_MODNAME フィールド</li> </ul> </li> <li>• X'04' IRM_ARCH3 に含まれるすべてのフィールド用、および IMS 間接続のためのセッション・トークンの IRM_SESTKN フィールド用のスペースを含む IRM 接頭部のユーザー部分のアーキテクチャー構造である IRM_ARCH4 を指定する。</li> <li>• X'05' IRM_ARCH4 および以下の追加フィールドに含まれるすべてのフィールドのスペースを収めた IRM 接頭部のユーザー部分のアーキテクチャー構造である IRM_ARCH5 を指定する。 <ul style="list-style-type: none"> <li>- IRM_EXTN_OFF フィールド</li> <li>- 2 バイトの予約フィールド</li> </ul> </li> </ul>

表 17. IRM 接頭部の固定部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_F0	1 バイト	7	7	<ul style="list-style-type: none"> <li>• X'80' IRM_F0_SYNONLY を指定する。これは、IMS 従属領域で実行されている IMS アプリケーション・プログラムから同期コールアウト・メッセージのみをリトリートする RESUME TPIPE 呼び出しである。</li> <li>• X'40' IRM_F0_SYNASYN を指定する。これは、IMS アプリケーション・プログラムからの同期コールアウト・メッセージと非同期メッセージの両方をリトリートする RESUME TPIPE 呼び出しである。</li> <li>• X'20' IRM_F0_SYNCNAK を指定する。これは NAK をトリガーしたメッセージを T パイプ・キューに保持するように OTMA に指示する IMS Connect クライアントからの NAK メッセージである。</li> <li>• X'10' IRM_F0_NAKRSN を指定する。これは、NAK 応答の理由コードを含む NAK メッセージである。</li> <li>• X'04' IRM_F0_EXTENS を指定する。これは、メッセージに 1 つ以上の IRM 拡張子が含まれていることを示す。</li> <li>• X'01' IRM_F0_XMLTD を指定する。これは、トランザクション・コードとデータの両方を含む XML タグ付きメッセージを、IMS アプリケーション・プログラムが想定するフォーマットに変換するように求める IMS Enterprise Suite SOAP Gateway クライアントからの要求を示す。</li> <li>• X'02' IRM_F0_XML_D を指定する。これは、データのみを収める XML タグ付きメッセージを、IMS アプリケーション・プログラムが想定するフォーマットに変換するように求める SOAP Gateway クライアントからの要求を示す。</li> </ul>
IRM_ID	8 バイト	8	8	<p>文字ストリング。完全なメッセージが受信された後で実行されるユーザー出口の ID を指定する。また、IMS Connect はこのフィールドを読み取って、着信メッセージが ASCII と EBCDIC のどちらであるかを判別する。</p> <p>IMS Connect 提供のユーザー・メッセージ出口は、以下の ID を予約しており、それを使用する。</p> <ul style="list-style-type: none"> <li>• *HWSCSL*-- HWSCSLOO 用</li> <li>• *HWSJAV*-- HWSJAVA0 用</li> <li>• *HWSOA1*-- HWSSOAP1 用</li> <li>• *SAMPL1*-- HWSSMPL1 用</li> <li>• *SAMPLE*-- HWSSMPL0 用</li> </ul>
IRM_NAK_RSNCDE	2 バイト	16	10	NAK 応答のオプションの理由コード
IRM_RES1	2 バイト	18	12	将来の利用のために予約済み。2 進ゼロに初期設定する。

表 17. IRM 接頭部の固定部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_F5	1 バイト	20	X'14'	<p>入力メッセージのタイプを示す。</p> <p><b>X'80'</b> クライアントが組み立てる OTMA ヘッダー。</p> <p><b>X'40'</b> クライアントが行った変換。</p> <p><b>X'20'</b> RESUME TPIPE を発行するクライアント・アプリケーションは、ICAL コールアウト・メッセージ内の制御データをサポートする。</p> <p><b>X'10'</b> 待機オプションを指定した単一メッセージ。RESUME TPIPE 呼び出しに続いて戻される唯一のメッセージ。メッセージが存在しない場合、OTMA はメッセージの到着を待ってから、その単一メッセージを IMS Connect に送信する。IMS Connect にメッセージが戻される前に、RESUME TPIPE 呼び出しに設定されたタイマーが期限切れになることがある。その場合、IMS Connect はメッセージを受信したときに NAK 応答を発行する。</p> <p><b>X'04'</b> メッセージの非自動フロー。現行のすべてのメッセージが 1 つずつ戻される。  非自動フロー・オプションを使用するのは、クライアントが専用出力クライアントの場合のみ。これは、Auto (自動) に似ているが、IRM_TIMER が最後の受信を終了させることだけが異なる。  IRM_TIMER は、小さな値に設定しておくこと。クライアントから ACK が送信されるたびに、IRM_TIMER 値がリセットされる。RESUME_TPIPE で設定される IRM_TIMER 値は、最初の受信状態に対してのみ適用される。</p> <p><b>X'02'</b> メッセージの自動フロー。現行のすべてのメッセージが 1 つずつ戻される。自動フロー・オプションを使用するのは、クライアントが専用出力クライアントの場合のみ。  IRM_TIMER は、大きな値に設定しておくこと。クライアントから ACK が送信されるたびに、IRM_TIMER 値がリセットされる。RESUME_TPIPE で設定される IRM_TIMER 値は、最初の受信状態に対してのみ適用される。</p> <p><b>X'01'</b> 単一メッセージ。RESUME TPIPE 呼び出しに続いて戻される唯一のメッセージ。メッセージが存在しない場合、OTMA はメッセージを待たず、指定されたタイムアウト値に基づいて IMS Connect のタイマーのタイムアウトが発生する。</p> <p><b>X'08'</b> IRM_F5_XID。このメッセージには、X/Open ID (XID) が含まれる。</p> <p><b>X'00'</b> メッセージの非オプション・フロー (意味は X'04' を参照)。値を指定しなければ、これがデフォルトとなる。</p>

表 17. IRM 接頭部の固定部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_TIMER	1 バイト	21	X'15'	IMS が IMS Connect にデータ (結果としてクライアントに送られる) を戻すのを IMS Connect が待機する時間遅延。以下の機能が IRM_TIMER 設定をサポートする。 <ul style="list-style-type: none"> <li>• RESUME TPIPE 呼び出しの TCP/IP SEND</li> <li>• ACK または NAK メッセージの TCP/IP SEND</li> <li>• データの TCP/IP SEND</li> <li>• RESUME TPIPE 呼び出しの PC SEND</li> <li>• ACK または NAK メッセージの PC SEND</li> <li>• データの PC SEND</li> </ul>
IRM_SOCT	1 バイト	22	X'16'	ソケット接続タイプ。クライアントは、この値を次のように設定できる。 <p><b>X'00'</b> トランザクション・ソケット。ソケット接続は、単一トランザクションの間だけ持続する。</p> <p><b>X'10'</b> 持続ソケット。このソケット接続は、複数のトランザクションにわたって持続する。</p> <p><b>X'40'</b> 非持続ソケット。ソケット接続は、1つの入力と1つの出力から成る単一交換の間、持続する。<b>推奨:</b> 会話型トランザクションをインプリメントすることを計画している場合は、このソケット・タイプは使用しないでください。複数の接続と切断が起こるからです。</p>
IRM_ES	1 バイト	23	X'17'	ユニコード・エンコード・スキーマ。2進ゼロに初期設定する。 <p><b>X'01'</b> UTF8 エンコード・スキーマ。</p> <p><b>X'02'</b> UCS2 エンコード・スキーマ。</p> <p><b>X'02'</b> UTF16 エンコード・スキーマ。</p>
IRM_CLIENTID	8 バイト	24	X'18'	左揃え、ブランク埋め込みの、1文字から8文字までの英数字大文字 (A から Z、0 から 9) または特殊文字 (@、#、\$) のストリング。IRM_CLIENTID は、IMS Connect が使用するクライアント ID の名前を指定する。このストリングがクライアントから提供されない場合は、ユーザー出口がストリングを生成する必要がある。 <p>クライアント ID は、EXIT PARMLIST フィールドの EXPREA_CLID に入れて、出口から IMS Connect に戻される。</p>

### 関連概念

320 ページの『入力メッセージのタイムアウトの指定』

IMS Connect クライアントからの個々の入力メッセージは、いずれも IMS 要求メッセージ (IRM) ヘッダーの固定部分の IRM\_TIMER フィールドに異なるタイムアウト値を設定できます。

### 関連資料

229 ページの『HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口ルーチンの IRM のユーザー部分のフォーマット』

HWSSMPL0、HWSSMPL1、またはユーザー作成メッセージ出口によってサポートされるユーザー作成クライアント・アプリケーションでは、IMS Connect クライアント入力メッセージ内の IMS 要求メッセージ

(IRM) ヘッダーの 4 バイトの長さフィールドと 28 バイトの固定部分に続けて、IRM にユーザー定義セクションを組み込むことができます。

## HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口ルーチンの IRM のユーザー部分のフォーマット

HWSSMPL0、HWSSMPL1、またはユーザー作成メッセージ出口によってサポートされるユーザー作成クライアント・アプリケーションでは、IMS Connect クライアント入力メッセージ内の IMS 要求メッセージ (IRM) ヘッダーの 4 バイトの長さフィールドと 28 バイトの固定部分に続けて、IRM にユーザー定義セクションを組み込むことができます。

IMS Connect クライアントによってサブミットされるメッセージには、メッセージのタイプに応じて、IRM の後にアプリケーション・データまたは IMS データが含まれている場合も含まれていない場合もあります。例えば、IRM\_F4 フィールドに指定される以下のメッセージ・タイプの場合、IRM の後にデータ・エレメントを含みません。

- ACK (A)
- CANCEL TIMER (C)
- DEALLOCATE (D)
- NAK (N)
- RESUME TPIPE (R)

IRM の後にアプリケーション・データを含むメッセージ・タイプには以下のタイプがあり、これらも IRM\_F4 フィールドで指定されます。

- SENDONLY (S)
- SENDONLYA (K)
- 送受信 (X'40')
- ACK 付き同期コールアウト応答 (L)
- 同期コールアウト応答 (M)

IRM のユーザー部分の直後にあるデータは、LLZZDATA のフォーマットであることが必要です。ここで、LL はメッセージ・セグメントの合計の長さ (LL フィールドを含む) で、ZZ は 2 進ゼロ、DATA は IMS トランザクション・コードの後にトランザクション・データを続けたものです。

**制約事項:** IMS TM Resource Adapter、IMS Enterprise Suite SOAP Gateway、または IBM Management Console for IMS and Db2 for z/OS などの IMS Connect クライアント・アプリケーションを使用している場合は、IRM にユーザー定義部分を含めることはできません。

以下の表に、HWSSMPL0、HWSSMPL1、およびその他のユーザー作成ユーザー・メッセージ出口によって使用される IRM のユーザー部分のフォーマットを示します。

このトピックにはプロダクト・センシティブ・プログラミング・インターフェース 情報が含まれています。

表 18. HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口の IRM のユーザー部分

フィールド	長さ	オフセット		意味
		10 進	16 進	
llll	4 バイト	0	X'00'	IRM、アプリケーション・データ、メッセージ終結標識、およびこの長さフィールドを含むメッセージ全体の長さ。
固定フォーマット IRM	28 バイト	4	X'04'	IRM 接頭部の共通固定部分。
IRM_MAP	0 バイト	32	X'20'	XID マップが存在する場合、そのマップの位置。



表 18. HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口の IRM のユーザー部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_F1	1 バイト	32	X'20'	<p>このフィールドの目的は、その値によって異なります。</p> <p><b>X'80' - IRM_F1_MFSREQ</b> ユーザーが、IMS が MFS MOD 名を返すことを要求。</p> <p><b>X'40' - IRM_F1_CIDREQ</b> ユーザーは、IMS Connect が client_id を戻すよう要求しています。</p> <p><b>X'20' - IRM_F1_UC</b> ユニコード・メッセージ。</p> <p><b>X'10' - IRM_F1_UCTC</b> ユニコード・トランザクション・コード。</p> <p><b>X'04' - IRM_F1_SOARSP</b> ACK 付き送信専用要求に対して、ACK でメッセージ・テキストが返されない</p> <p><b>X'02' - IRM_F1_NOWAIT</b> この CMO 送受信メッセージは、予期される ACK または NAK 応答に対して NOWAIT オプションを使用。</p> <p><b>X'01' - IRM_F1_TRNEXP</b> ユーザーが、IMS Connect が入力トランザクションの有効期限を設定することを要求。</p> <p><b>X'00'</b> ユーザーが MFS MOD 名を戻さないことを要求。</p> <p>クライアントがこの値を指定しない場合は、ユーザー出口はデフォルト値を使用する必要がある。</p> <p>MFS MOD 名フラグは、EXIT PARMLIST フィールドの EXPREA_UFLAG1 に入れて、出口から IMS Connect に戻される。</p> <p>Unicode エンコード・スキーマ (UTF8、UCS2、UTF16) は、IRM の固定部分の IRM_ES フィールドで識別される。</p>
IRM_F2	1 バイト	33	X'21'	<p>コミット・モードを指定します。</p> <ul style="list-style-type: none"> <li>• X'40' - コミット・モード「0」(CM0)。コミット後送信とも呼ばれる。</li> <li>• X'20' - コミット・モード「1」(CM1)。送信後コミットとも呼ばれる。</li> <li>• X'01' - 重複 client_ID エラー条件を防止するために、IMS Connect に対して、固有の client_ID を生成するよう指定する。</li> </ul> <p>クライアントがこの値を指定しない場合は、ユーザー出口はデフォルト値を使用する必要がある。</p> <p>コミット・モード・フラグは、OTMA ヘッダー・フィールドの OMHDRSYN に入れて、出口から IMS Connect に戻される。</p>

表 18. HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口の IRM のユーザー部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_F3	1 バイト	34	X'22'	<p>このフィールドは、値に応じて、同期レベル、CM0 出力のページ・オプションか転送オプション、または送信専用入力の順次配信オプションを指定する。</p> <ul style="list-style-type: none"> <li>• X'00' - 同期レベルは「NONE」</li> <li>• X'01' - 同期レベルは「CONFIRM」</li> <li>• X'02' - 同期レベルは「SYNCPT」</li> <li>• X'04' - 配信不能 CM0 出力のページ (IRM_F3_PURGE)</li> <li>• X'08' - 配信不能 CM0 出力の転送 (IRM_F3_REROUT)</li> <li>• X'10' - 順次配信付き送信専用 (IRM_F3_ORDER)。IRM_F3_ORDER は、送信専用トランザクションの場合、順次配信オプションを呼び出す。</li> <li>• X'20' - 複数セグメント CM0 出力メッセージの場合、出力メッセージのすべてのセグメントが TP PCB に挿入されるまで、TP PCB への DL/I PURG 呼び出し (IRM_F3_IPURG) を無視する。このオプションは、IMS アプリケーションが TP PCB に挿入する複数セグメント・メッセージの各セグメントごとに、PURG 呼び出しを発行する場合に役立つ。</li> <li>• X'40' - OTMA CM0 入力メッセージの場合、IMS アプリケーションが IOPCB に応答しないか、別のトランザクションへのメッセージ通信を完了しない場合、トランザクションの応答モードに関係なく、DFS2082 メッセージ (IRM_F3_DFS2082) を受け取る。CM0 入力メッセージに対するこの DFS2082 メッセージは、元の入力トランザクションにのみ適用され、プログラム間通信はサポートしない。</li> <li>• X'80' - 重複するクライアント ID を取り消す (IRM_F3_CANCID)。新しいセッションの最初の要求時に、既存セッションが新しいセッションと同じクライアント ID を使用している場合、IRM_F3_CANCID は既存セッションを終了し、新しいセッションを続行できるようにする。</li> </ul> <p>CM0 の場合、同期レベルを CONFIRM に設定する必要がある。クライアントが同期レベルを指定しない場合は、ユーザー出口はデフォルト値を使用する必要がある。</p> <p>同期レベル・フラグは、OTMA ヘッダー・フィールドの OMHDRSLV に入れて、出口から IMS Connect に戻される。</p> <p>配信不能ページ・フラグは、OTMA ヘッダー・フィールドの OMHDRCLFL を OMHDRPND X'10' に設定して、ユーザー・メッセージ出口から IMS Connect に戻される。</p> <p>転送フラグが設定されている場合、IRM_REROUT_NM フィールドはオプションになる。</p> <p>配信不能ページ機能と転送機能は同時に指定できない。両方の機能を指定すると、出力メッセージはページも転送もされず、OTMA がメッセージ DFS2407W を発行する。</p>

表 18. HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口の IRM のユーザー部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_F4	1 バイト	35	X'23'	<p>IRM_F4 フラグは、クライアントが送信するメッセージのタイプを示す。メッセージ・タイプは、ASCII または EBCDIC 文字値で指定される。この値が IMS Connect に送信されて、ユーザー出口に渡されると、出口は適切な OTMA 構造を作成し、それを IMS Connect に返し、IMS に転送する。有効な値およびメッセージ・タイプを以下に示す。</p> <p><b>A</b></p> <p>ACK (IRM_F4_ACK) - IMS Connect から受信した出力に対する ACK 応答。ACK は、以下のいずれかのメッセージの受け入れを示すためにクライアントによって使用される。</p> <ul style="list-style-type: none"> <li>IMS アプリケーション・プログラムから出される同期コールアウト要求メッセージ</li> <li>クライアントからの元の入力メッセージが同期レベルに CONFIRM を指定している場合の出力メッセージ</li> </ul> <p><b>C</b></p> <p>IRM タイマーの取り消し (IRM_F4_CANTIMER) - 別の接続の IRM タイマーを取り消す要求であり、クライアントは、その接続上で同じクライアント ID を使用して出力データを待機している。</p> <p><b>D</b></p> <p>割り振り解除 (IRM_F4_DEALLOC) - 会話の割り振りを解除する要求。</p> <p><b>K</b></p> <p>ACK が必須の送信専用 (IRM_F4_SNDONLYA) - IMS Connect からの ACK 応答を要求する送信専用トランザクション・メッセージ。</p> <p><b>L</b></p> <p>ACK を要求する同期コールアウト応答メッセージ (IRM_F4_SYNRESPA) - 送信専用プロトコルを使用し、IMS Connect からの確認応答を要求する、同期コールアウト応答メッセージ。</p> <p><b>M</b></p> <p>同期コールアウト応答メッセージ (IRM_F4_SYNRESP) - 送信専用プロトコルを使用する同期コールアウト応答メッセージ。</p> <p><b>N</b></p> <p>NAK (IRM_F4_NACK) - IMS Connect からの以下のいずれかのタイプの出力が拒否されたことを示す、クライアントからの NAK 応答。</p> <ul style="list-style-type: none"> <li>IMS アプリケーション・プログラムから出される同期コールアウト要求メッセージ</li> <li>クライアントからの元の入力メッセージが同期レベルに CONFIRM を指定している場合の出力メッセージ</li> </ul> <p><b>R</b></p> <p>RESUME TPIPE (IRM_F4_RESUMET) - IMS からの非同期出力データを求める RESUME TPIPE 呼び出し。RESUME TPIPE 呼び出しは、CMO を使用してトランザクション・ソケットまたは永続ソケットで実行しなければならない。</p>
IRM_F4 (続き)	1 バイト	35	X'23'	<p><b>S</b></p> <p>送信専用 (IRM_F4_SENDOONLY) - 非応答モードの非会話型トランザクションの送信専用対話を開始する送信専用トランザクション・メッセージ。ホスト・アプリケーションが IO PCB に対して ISRT を発行せずに終了した場合、DFS2082 メッセージはクライアントに返されません。SENDONLY 対話では、CMO を使用する必要があります。</p> <p><b>ブランク (X'40')</b></p> <p>会話型または非会話型応答モード・トランザクション用の送受信対話。</p>
IRM_TRNCOD	8 バイト	36	X'24'	文字ストリング。IMS トランザクション・コードを指定する。
IRM_IMSDESTID	8 バイト	44	X'2C'	文字ストリング。データ・ストア名 (IMS 宛先 ID) を指定する。このフィールドは、クライアントが指定する必要がある。このデータ・ストア名は、OTMA ヘッダー・フィールドの OMUSR_DESTID に入れて、出口から IMS Connect に戻される。

表 18. HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口の IRM のユーザー部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_LTERM	8 バイト	52	X'34'	<p>文字ストリング。IMS 論理端末 (LTERM) オーバーライド名を指定する。このフィールドは、有効な名前またはブランクに設定することができる。</p> <p>LTERM オーバーライド名は、OTMA ヘッダー・フィールドの OMHDLRM に入れて出口から IMS Connect に戻される。</p> <p>IMS ホスト・アプリケーションの場合、このフィールドの値はユーザー・メッセージ出口によって設定される。この出口は、この値を OTMA フィールド OMHDLRM に移動するか、またはあらかじめ決められた値を OMHDLRM に設定する。LTERM オーバーライド値を指定した場合、OTMA はその値を IOPCB LTERM フィールドに設定する。LTERM オーバーライド値を指定しなかった場合、OTMA は IMS Connect 定義の T パイプ名を代わりに IOPCB LTERM フィールドに設定する。T パイプ名は、コミット・モードがゼロの場合は CLIENT ID、コミット・モードが 1 の場合は PORT ID に設定される。</p> <p>IOPCB 内の LTERM 値を使用して論理判定を行う場合、IOPCB LTERM 名の命名規則を理解しておく必要がある。</p> <p>入力トランザクションに対して IBM ワークロード・マネージャーを呼び出すために LTERM オーバーライド名を使用する必要がある場合は、DFSOTMA 記述子で <b>WLMLTRM=YES</b> を指定する必要がある。</p>
IRM_RACF_USERID	8 バイト	60	X'3C'	<p>文字ストリング。RACF ユーザー ID を指定する。RACF を使用するのであれば、クライアントはこれを指定する必要がある。</p> <p>RACF ユーザー ID 名は、OTMA ヘッダー・フィールドの OMSECUID に入れて出口から IMS Connect に戻される。</p>
IRM_RACF_GRPNAME	8 バイト	68	X'44'	<p>文字ストリング。RACF グループ名を指定する。</p> <p>RACF グループ名は、OTMA ヘッダー・フィールドの OMSECGRP に入れて出口から IMS Connect に戻される。</p> <p>このフィールドに要求されるオフセットを保存するために、IRM_RACF_GRPNAME を指定するときは、以下のフィールドも有効な値またはブランクとともに指定する必要がある。</p> <ul style="list-style-type: none"> <li>IRM_RACF_USERID</li> </ul>

表 18. HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口の IRM のユーザー部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_RACF_PW	8 バイト	76	X'4C'	<p>文字ストリング。RACF パスチケットまたはパスワードを指定する。RACF を使用するのであれば、クライアントはこれを指定する必要がある。</p> <p>パスチケット値またはパスワード値は、OTMA ヘッダー・フィールドの OMUSR_PASSTICK に入れて、ユーザー・メッセージ出口から IMS Connect に戻される。</p> <p>このフィールドに要求されるオフセットを保存するために、IRM_RACF_PW を指定するときは、以下のフィールドも有効な値またはブランクとともに指定する必要がある。</p> <ul style="list-style-type: none"> <li>• IRM_RACF_USERID</li> <li>• IRM_RACF_GRNAME</li> </ul> <p>RACF パスワードには、以下のどの特殊文字でも含めることができる。IMS Connect は EBCDIC ページ 037 を使用して、パスワード文字の妥当性を検査する。示されている記号は、EBCDIC コード・ページ 1047 および 037 である。</p> <ul style="list-style-type: none"> <li>• . (16 進数値は 4B)</li> <li>• &lt; (16 進数値は 4C)</li> <li>• + (16 進数値は 4E)</li> <li>•   (16 進数値は 4F)</li> <li>• &amp; (16 進数値は 50)</li> <li>• ! (16 進数値は 5A)</li> <li>• * (16 進数値は 5C)</li> <li>• - (16 進数値は 60)</li> <li>• % (16 進数値は 6C)</li> <li>• _ (16 進数値は 6D)</li> <li>• &gt; (16 進数値は 6E)</li> <li>• ? (16 進数値は 6F)</li> <li>• : (16 進数値は 7A)</li> <li>• = (16 進数値は 7E)</li> </ul> <p>RACF パスワードには、以下の国別文字を含めることもできる。</p> <ul style="list-style-type: none"> <li>• \$ (16 進数値は 5B)</li> <li>• # (16 進数値は 7B)</li> <li>• @ (16 進数値は 7C)</li> </ul>
IRM_APPL_NM	8 バイト	84	X'54'	<p>文字ストリング。PTKTDATA 定義で RACF に定義された RACF APPL 名を指定する。このフィールドが含まれていないか、またはブランクである場合、RACF APPL 名は DATASTORE 構成ステートメントの APPL パラメーターから取得される。</p> <p>このフィールドに要求されるオフセットを保存するために、IRM_APPL_NM を指定するときは、以下のフィールドも有効な値またはブランクとともに指定する必要がある。</p> <ul style="list-style-type: none"> <li>• IRM_RACF_USERID</li> <li>• IRM_RACF_GRNAME</li> <li>• IRM_RACF_PW</li> </ul>

表 18. HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口の IRM のユーザー部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_REROUT_NM	8 バイト	92	X'5C'	<p>A から Z、0 から 9 または特殊文字 (@、#、\$ など) の文字ストリング。クライアント・リルート要求の転送 T パイプ名。このフィールドはオプションです。</p> <p><b>推奨事項:</b> デフォルト値としてブランクを使用。</p> <p>IRM_REROUT_NM フィールドを指定する場合は、以下も指定する必要がある。</p> <ul style="list-style-type: none"> <li>IRM のユーザー部分で IRM_F3 フィールドに IRM_F3_REROUT</li> <li>IRM の共通固定部分で IRM_ARCH フィールドに IRM_ARCH1 または IRM_ARCH2</li> </ul> <p>このフィールドに要求されるオフセットを保存するために、IRM_REROUT_NM を指定するときは、以下のフィールドも有効な値またはブランクとともに指定する必要がある。</p> <ul style="list-style-type: none"> <li>IRM_RACF_USERID</li> <li>IRM_RACF_GRNAME</li> <li>IRM_RACF_PW</li> <li>IRM_APPL_NM</li> </ul> <p>IRM_REROUT_NM および IRM_RT_ALTCID は同じオフセットを使用する。同じメッセージに両方を指定することはできない。</p>
IRM_RT_ALTCID	8 バイト	92	X'5C'	<p>RESUME TPIPE 呼び出しの代替クライアント ID。これはオプションのフィールドである。代替クライアント ID を指定した場合は、IRM_ARCH フィールドを IRM_ARCH1 に設定する必要がある。</p> <p>このフィールドに要求されるオフセットを保存するために、IRM_RT_ALTCID を指定するときは、以下のフィールドも有効な値またはブランクとともに指定する必要がある。</p> <ul style="list-style-type: none"> <li>IRM_RACF_USERID</li> <li>IRM_RACF_GRNAME</li> <li>IRM_RACF_PW</li> <li>IRM_APPL_NM</li> </ul>
IRM_TAG_ADAPT	8 バイト	100	X'64'	<p>XML メッセージを、選択したプログラミング言語に変換するため、およびその言語から変換するために IMS Connect が呼び出すアダプターの名前。</p> <p>このフィールドに要求されるオフセットを保存するために、IRM_TAG_ADAPT を組み込むときは、以下のフィールドも有効な値またはブランクとともに指定する必要がある。</p> <ul style="list-style-type: none"> <li>IRM_RACF_USERID</li> <li>IRM_RACF_GRNAME</li> <li>IRM_RACF_PW</li> <li>IRM_APPL_NM</li> <li>IRM_REROUT_NM または IRM_RT_ALTCID</li> </ul>

表 18. HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口の IRM のユーザー部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
IRM_TAG_MAP	8 バイト	108	X'6C'	XML メッセージを、選択したプログラミング言語に、およびその言語から実際に変換するために XML アダプターが呼び出すコンバーターの名前。  このフィールドに要求されるオフセットを保存するために、IRM_TAG_MAP を組み込むときは、以下のフィールドも有効な値またはブランクとともに指定する必要がある。  <ul style="list-style-type: none"> <li>IRM_RACF_USERID</li> <li>IRM_RACF_GRNAME</li> <li>IRM_RACF_PW</li> <li>IRM_APPL_NM</li> <li>IRM_REROUT_NM または IRM_RT_ALTICID</li> <li>IRM_TAG_ADAPT</li> </ul>
IRM_MODNAME	8	116	74	入力メッセージの MFS MOD 名。
IRM_CORTKN	0XL40			同期コールアウト・メッセージの関連トークン。関連トークンには、要求を発行した IMS システムおよび IMS アプリケーション・プログラムに、同期コールアウト要求への応答を関連させるための情報が含まれる。
IRM_CT_LEN	2	124	7C	関連トークンの長さ。
IRM_CT_RESV1	2	126	7E	予約済み。
IRM_CT_IMSID	4	128	80	同期コールアウト要求を発行した IMS アプリケーションをスケジュールに入れる IMS システムの IMSID。
IRM_CT_MEMTK	8	132	84	OTMA TMEMBER トークン。
IRM_CT_AWETK	8	140	8C	OTMA メッセージ・トークン。
IRM_CT_TPIPE	8	148	94	OTMA TPIPE 名。
IRM_CT_USERID	8	156	9C	ICAL 呼び出しで指定されるユーザー ID。
IRM_SESTKN	8	164	A4	IMS Connect と IMS Connect 間の接続に使用されるセッション・トークン値。
IRM_EXTN_OFF	2	172	AC	IRM の開始から最初の IRM 拡張子へのオフセット値。
IRM_F6	1	174	AE	このフラグの目的は、その値によって異なる。  <b>X'80' - IRM_F6_NWSE</b> RESUME TPIPE 要求に有効。ユーザー作成のクライアント・アプリケーションは、分散ネットワーク・セキュリティ資格情報が含まれるネットワーク・セキュリティ・セグメントを含む IMS からの出力メッセージをサポートします。ネットワーク・セキュリティ・セグメントには、*NETSID* セグメント (ネットワーク・セッション ID が含まれる) および *NETUID* セグメント (ネットワーク・ユーザー ID が含まれる) が含まれます。
IRM_RESERVED1	1	175	AF	予約済み。
ORG IRM_MAP				
IRM_XID	XL140	32	20	IRM_XID フィールドは、2 フェーズ・コミット・プロトコル・メッセージに使用されます。IRM_XID フィールドは、IRM_ARCH が 5 未満の IMS TMRA メッセージに対してのみ有効です。IRM_ARCH が 5 以上の場合は、IRM_LEN フィールドの値を使用して、入力バッファ内の XID の位置を判別する必要があります。これは、IRMMASK が可変長 IRM 拡張子をサポートするように拡張されたために必要な作業です。IRM 拡張子は IRMMASK の一部として組み込まれており、IRM_LEN に指定された合計の長さに含まれます。したがって、XID は、より大きくなった IRMMASK (これには、IRM 拡張子も含まれている可能性があります) の末尾の後続のバイトで開始します。IRM_ARCH が 5 以上の場合、メッセージは次のようになります。LLLL   IRM + IRM EXTENSION(S)   XID   OTMA HEADERS...



表 18. HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口の IRM のユーザー部分 (続き)

フィールド	長さ	オフセット		意味
		10 進	16 進	
ORG IRM_MAP: オフセット 32-35 の再マップ。				
IRM_TMRA_EXTN_OFF	2	32	20	IRM の開始から IRM アーキテクチャー・レベル (IRM_ARCH) が 5 以上の IMS TMRA メッセージの最初の IRM 拡張子までのオフセット値。
IRM_TMRA_SRVD1	2	34	22	IMS Connect 用に予約済み。
IRM_TMRA_EXTN_START	X	36	24	IMS TMRA メッセージの最初の IRM 拡張子の開始を示します。
*IRM_XID_ARCH5	XL140	X	X	このフィールドは、5 以上の IRM_ARCH を使用する IMS TMRA メッセージの XID のサイズを表します。メモリー内のこのフィールドの位置は、IRM 拡張子の数やサイズによって異なります。そのため、IRM_LEN の値を使用して XID の実際の位置を判別する必要があります。詳しくは、IRM_XID の「意味」の列を参照してください。

## IRM 拡張のフォーマット

IMS 要求メッセージ (IRM) 拡張を使用すると、IRM をマップする DSECT を拡張することなく、IMS Connect クライアント・アプリケーションから情報を送信することができます。

各 IRM における IRM 拡張の数はさまざまで、IRM 内の IRM は任意の順序で定義することができます。すべての IRM 拡張に 12 バイトのヘッダーが含まれます。ユーザー・メッセージ出口は、各 IRM 拡張の固有 ID を使用して拡張のコンテンツを判別します。デフォルトでは、ユーザー・メッセージ出口は、IRM 拡張 ID を EBCDIC として処理し、IRM 拡張データをバイナリー・データとして処理します。IRM 拡張 ID と拡張データがそれぞれ EBCDIC およびバイナリー・データで定義されていない場合は、ユーザー・メッセージ出口または TCP/IP アプリケーションを変更します。

さまざまなユーザー・メッセージ出口によってさまざまな IRM 拡張がサポートされます。IRM 拡張が含まれる IRM がその拡張をサポートしないユーザー・メッセージ出口によって処理される場合、その拡張は無視されます。

表 19. IRM 拡張のフォーマット

フィールド	長さ	16 進オフセット	説明および設定
IRMEXTN_LL	2	0	この IRM 拡張の長さ。
IRMEXTN_ZZ	2	2	予約フィールド。

表 19. IRM 拡張のフォーマット (続き)

フィールド	長さ	16 進オフセット	説明および設定
IRMEXTN_ID	8	4	<p>この IRM 拡張の固有 ID。各 ID の先頭と末尾はアスタリスク (*) です。</p> <p>このフィールドでは、以下の ID を定義することができます。</p> <p><b>*CONXT*</b> IRM 拡張にコンテキスト情報が含まれることを示します。この ID は EBCDIC または ASCII でエンコードされ、HWSSOAP1 ユーザー・メッセージ出口によってサポートされます。</p> <p><b>*TRCKID*</b> IRM 拡張にトラッキング ID が含まれることを示します。この ID は EBCDIC または ASCII でエンコードされ、HWSSOAP1 ユーザー・メッセージ出口によってサポートされます。</p> <p><b>*PHRASE*</b> IRM 拡張に RACF パスワード・フレーズが含まれることを示します。この ID は EBCDIC でエンコードされ、HWSJAVA0 ユーザー・メッセージ出口によってサポートされます。</p> <p><b>*NETUID*</b> IRM 拡張にネットワーク・セキュリティ・ユーザー ID が含まれることを示します。この ID は EBCDIC または ASCII でエンコードされ、HWSSMPLO および HWSSMPL1 ユーザー・メッセージ出口によってサポートされます。</p> <p><b>*NETSID*</b> IRM 拡張にネットワーク・セキュリティ・セッション ID が含まれることを示します。この ID は EBCDIC または ASCII でエンコードされ、HWSSMPLO および HWSSMPL1 ユーザー・メッセージ出口によってサポートされます。</p>
IRMEXTN_DATA	*	12	<p>このフィールドを使用して、IBM 拡張 ID の直後に続く可変長データを見つけることができます。このデータの構造は、IRM 拡張 ID によって異なります。デフォルトでは、IRM 拡張データはバイナリー・データとして扱われます。</p>

## クライアント出口からの出力構造

以下の表は、ユーザー作成のクライアント・アプリケーション出口ルーチンが返すメッセージの構造(メッセージごとに1回の出現)を示しています。この表には、フィールド名、フィールドの長さ、およびフィールドについての簡単な説明が記載されています。

表 20. ユーザー作成の IMS Connect クライアント・アプリケーション出口ルーチンが返すメッセージ構造

フィールド	長さ	意味
BPE ヘッダー	64 バイト	240 ページの『BPE ヘッダー・フォーマット』に定義。
OTMA 構造	OTMA ヘッダーの全長	HWSOMPFX マクロ (全 OTMA 構造) の詳細については、167 ページの『IMS Connect 出口ルーチンをサポートするマクロ』を参照してください。
LLZZTRANCODEDATA	n バイト	<ul style="list-style-type: none"><li>• LL - セグメントの長さ</li><li>• ZZ - 2 進ゼロに設定する</li><li>• TRANCODE - IMS の 1 バイトから 8 バイトのトランザクション・コード</li><li>• DATA - ユーザー・データ</li></ul>
LLZZDATA	n バイト	<ul style="list-style-type: none"><li>• LL - セグメントの長さ</li><li>• ZZ - 2 進ゼロに設定する</li><li>• DATA - ユーザー・データ</li></ul>
LLZZDATA は反復され、最大全長は 32 KB。さらにデータがある場合は、239 ページの『その他の IMS Connect 構造』のように構造は継続する。最初のセグメントだけが IMS トランザクション・コードを含んでおり、以降のセグメントは、そのトランザクションが処理されるのに必要なセグメント・データを含む。		
LL	2 バイト	LL - 2 進ゼロに設定されて、この構造の終わりを示す。この LL フィールドはセグメント長の部分には入らない。

### 関連資料

261 ページの『IMS Connect で使用される OTMA ヘッダー・フィールド』

IMS Connect は、OTMA に送信されるメッセージのヘッダー内のフィールドを使用して、処理オプションおよびその他の情報を IMS に伝えます。

## その他の IMS Connect 構造

IRM に加えて、IMS Connect が処理するメッセージには、その他の反復構造が含まれます。

以下の表は、IMS Connect に返されるデータがマップされるまで反復されるその他の構造を示しています。この表には、フィールド名、フィールドの長さ、およびフィールドについての簡単な説明が記載されています。

表 21. その他の IMS Connect 反復メッセージ構造

フィールド	長さ	意味
BPE ヘッダー	64 バイト	<a href="#">240 ページの『BPE ヘッダー・フォーマット』</a> に定義。
OTMA 構造	32 バイト	IMS Connect マクロ HWSOMPFX によってマップされる OTMA メッセージ制御構造。
LLZZDATA LLZZDATA フィールドは反復され、最大全長は 32 KB。さらにデータがある場合は、この構造が継続する。	n バイト	<ul style="list-style-type: none"> <li>• LL - セグメントの長さ</li> <li>• ZZ - 2 進ゼロに設定する</li> <li>• DATA - ユーザー・データ</li> </ul>
LL	2 バイト	LL - 2 進ゼロに設定されて、この構造の終わりを示す。

#### 関連資料

[261 ページの『IMS Connect で使用される OTMA メッセージ制御フィールド』](#)

このトピックの表では、OTMA メッセージ制御ヘッダーのフィールドおよびそれらのフィールドの順序を定義します。

### BPE ヘッダー・フォーマット

BPE ヘッダーは、IMS Connect によって処理されるメッセージ内に繰り返し生じる構造です。

以下の表は、BPE ヘッダーのレイアウトの各フィールドを示しています。このトピックには**プロダクト・センシティブ・プログラミング・インターフェース**情報が含まれています。

表 22. BPE ヘッダーのレイアウト

フィールド	長さ	意味
llll	4 バイト	構造の全長。最初の BPE ヘッダーにのみ設定される。このフィールドは、IMS Connect が管理するもので、出口で変更してはならない。
CHAIN PTR	4 バイト	<p>このメッセージ内の次の BPE ヘッダーを指すチェーン・ポインター。メッセージ内の最後の BPE ヘッダーは、メッセージ内の BPE ヘッダーの終わりを示すために、チェーン・ポインター値として 2 進ゼロを持つ必要がある。</p> <p>これらのチェーン・ポインターは、ユーザー作成のクライアント・アプリケーション出口ルーチンによって設定される。</p>
STORAGE TYPE	8 バイト	このフィールドは、IMS Connect が管理するもので、ユーザー出口で変更してはならない。
TYPE ACCESS	4 バイト	このフィールドは、IMS Connect が管理するもので、ユーザー出口で変更してはならない。

表 22. BPE ヘッダーのレイアウト (続き)

フィールド	長さ	意味
SUBPOOL	1 バイト	このフィールドは、IMS Connect が管理するもので、ユーザー出口で変更してはならない。
予約済み	43 バイト	このフィールドは、IMS Connect が管理するもので、ユーザー出口で変更してはならない。

## メッセージ構造と IMS Connect ユーザー・メッセージ出口ルーチン

IMS Connect は、254 までのユーザー出口を構成ファイルに定義することができます。IMS Connect がサポートする入力メッセージ構造は 2 つあります。また、ユーザー出口から戻るときにも 2 つのメッセージ構造がサポートされます。

### クライアントからの入力メッセージ

ユーザー・メッセージ出口ルーチンに渡される、およびユーザー・メッセージ出口ルーチンから返される入力メッセージの構造は、ユーザー・メッセージ出口ルーチンが IMS TM Resource Adapter またはユーザー作成の IMS Connect クライアントをサポートするかどうかを含めて、いくつかの要因に基づいて異なります。

以下の表は、IMS Connect がクライアントから受け取る入力メッセージの構造を示しています。この表では、入力メッセージ構造タイプ、OTMA ヘッダーが存在するかどうか、出口データがクライアント・コードで変換されるかどうか、出口タイプ・フラグ、およびサポートするメッセージ・タイプが提供されます。

表 23. 入力メッセージ構造

入力メッセージ構造タイプ	OTMA ヘッダーがある	出口データはクライアント・コードで変換される	出口タイプ・フラグ (IRMHDR_FLG5)	サポートするメッセージ・タイプ
1	Y	Y	11000000	HWSJAVA0
1	Y	N	10000000	クライアント/サーバーが OTMA ヘッダーを組み立てるときは HWSSMPL0 および HWSSMPL1 が OTMA ヘッダーを構築しないように変更された
2	N	Y	01000000	HWSSMPL0 および HWSSMPL1 はデータを変換しないように変更された
2	N	N	00000000	HWSSMPL0 HWSSMPL1 HWSCSLO0

以下の表は、出口が受け取った入力構造に基づいて、出口が戻す入力メッセージ構造を示しています。この表では、入力メッセージ構造タイプ、出口出力メッセージ構造タイプ、出口タイプ・フラグ、およびサポートするメッセージ・タイプが提供されます。

表 24. 出口から返される入力メッセージ構造

入力メッセージ構造タイプ	出口出力メッセージ構造タイプ	出口タイプ・フラグ (IRMHDR_FLG5)	サポートするメッセージ・タイプ
1	1	11000000	HWSJAVA0
1	1	10000000	クライアント/サーバーが OTMA ヘッダーを組み立てるときは HWSSMPL0 および HWSSMPL1 が OTMA ヘッダーを構築しないように変更された
2	3	01000000	HWSSMPL0 および HWSSMPL1 はデータを変換しないように変更された
2	3	00000000	HWSSMPL0 および HWSSMPL1

## クライアントへの出力メッセージ

IMS からの出力メッセージは、クライアントから呼び出されたユーザー・メッセージ出口ルーチンに渡されます。

このユーザー出口は、出口が入力の OTMA ヘッダーを追加した場合は、通常、出力の OTMA ヘッダーを除去します。また、通常、ユーザー出口は、入力で変換をしている場合、データを EBCDIC から ASCII に変換します。入力に変換を行っていない場合は、出力のための変換も行われません。

OTMA ヘッダーは、最高 4 つのセクションとアプリケーション・データから構成されます。出口が(入力時に存在しなかった) OTMA ヘッダーを除去する場合は、セクションごとのチェックが必要です。4 つのセクションは、次のとおりです。

- 制御 (OTMA 構造に常に存在する)
- ヘッダー (OTMA 構造に存在する場合も、しない場合もある)
- セキュリティー (OTMA 構造に存在する場合も、しない場合もある)
- ユーザー (OTMA 構造に存在する場合も、しない場合もある)

## IMS から IMS Connect への出力メッセージ

IMS から IMS Connect が受け取る出力メッセージは、すべて同じ構造になっています。すなわち OTMA ヘッダーの後ろに LLZZ DATA が続きます。メッセージが複数のセグメントを含む場合は、OTMA ヘッダーと LLZZ DATA がメッセージ内のセグメント数に応じて反復されます。

CM0 複数セグメント出力メッセージの場合、IMS アプリケーション・プログラムが TP PCB に各セグメントを挿入してから、DL/I PURG 呼び出しを発行すると、出力メッセージの各セグメントは、IMS Connect クライアントに個別に送信されます。

IMS アプリケーションが複数セグメント CM0 出力メッセージの各セグメントごとに PURG 呼び出しを発行する場合、OTMA メッセージ・ヘッダーの状態データ・セクションの OMHDRFCFL フィールドに X'02' (OMHDRIPG) を指定することで、メッセージの最後のセグメントが TP PCB に挿入されるまで、PURG 呼び出しを無視するように OTMA に指示することができます。OMHDRIPG を指定すると、OTMA は出力メッセージを単一の複数セグメント・メッセージとして IMS Connect に返します。

ユーザー作成の IMS Connect クライアントは、入力トランザクション・メッセージの IRM 接頭部のユーザー部分にある IRM\_F3 フィールドに X'20' (IRM\_F3\_IPURG) を指定することで、OMHDRIPG フラグを設定できます。

IMS への入力メッセージで、OTMA メッセージ接頭部のセキュリティー・データ・セクションに分散ネットワーク・セキュリティー資格情報が含まれている場合、RESUME TPIPE 呼び出しを発行した IMS Connect クライアントへの出力メッセージにも、ネットワーク・セキュリティー資格情報を含めることができます。

出力メッセージ内にネットワーク・セキュリティー資格情報を含めることができるようにするには、入力トランザクション・メッセージの IRM 接頭部のユーザー部分にある IRM\_F6 フィールドに X'80' を指定します。IRM\_F6 フィールドに X'80' が指定されていない場合、IMS は、たとえ入力メッセージにセキュリティー資格情報が組み込まれていても、出力メッセージ内の OTMA ヘッダーのセキュリティー・セクションからネットワーク・セキュリティー資格情報を削除します。

## 出口が IMS Connect に戻す IMS からの出力メッセージ

出口から IMS Connect に戻されるメッセージは、以下の 2 つの構造のうちのいずれかです。

- OTMA 構造がメッセージの中に組み込まれているメッセージ
- OTMA 構造がメッセージの中に組み込まれていないメッセージ

## メッセージ構造

IMS Connect 内を流れるメッセージの必須構造は、メッセージが入力メッセージであるか出力メッセージであるか、およびメッセージがユーザー・メッセージ出口によって既に変更されているかどうかによって異なります。以下のトピックでは、IMS TM Resource Adapter およびユーザー作成の IMS Connect クライアント・アプリケーションのメッセージ構造について説明します。

## クライアントからの入力メッセージおよび出口に渡される入力メッセージ

クライアントからの入力メッセージは、IMS TM Resource Adapter およびユーザー作成 IMS Connect クライアント・アプリケーションのメッセージ構造で構成されます。

このトピックにはプロダクト・センシティブ・プログラミング・インターフェース 情報が含まれています。

### IMS TM Resource Adapter メッセージ構造 - タイプ 1

以下の表は、IMS Connect がサポートする、IMS TM Resource Adapter クライアントからの入力メッセージのフォーマットを示しています。

表 25. IMS TM Resource Adapter のクライアント・アプリケーションの IMS Connect メッセージ・フォーマット

フィールド	長さ	意味
llll	4 バイト	llll フィールドを含む全メッセージの長さ
IRM	28 バイト	IRM の共通固定部分。
IRM_MAP	0 バイト	XID マップが存在する場合、そのマップの位置。
OTMA HDR	466 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、 <a href="#">261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』</a> を参照してください。
LL	2 バイト	データ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	最初にトランザクション・コードを持つユーザー・データ
OTMA CTL HDR	32 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、 <a href="#">261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』</a> を参照してください。
LL	2 バイト	2 番目のデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)



表 25. IMS TM Resource Adapter のクライアント・アプリケーションの IMS Connect メッセージ・フォーマット (続き)

フィールド	長さ	意味
DATA	n バイト	2 番目のデータ・セグメントのユーザー・データ (トランザクション・コードなし)
...		
OTMA CTL HDR	32 バイト	IMS.SDFS MAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』を参照してください。
LL	2 バイト	この (最後の) データ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	このデータ・セグメントのユーザー・データ (トランザクション・コードなし)
	ORG IRM_MAP	
IRM_XID	140 バイト	2 フェーズ・コミット処理のためのグローバル・トランザクション ID。

## ユーザー作成のメッセージ構造 - タイプ 2

以下の表は、IMS Connect のユーザー作成クライアント・アプリケーションがサポートする入力メッセージ・フォーマットを示しています。

このメッセージ・フォーマットは、HWSSMPL0 および HWSSMPL1 ユーザー・メッセージ出口ルーチン、ならびにユーザー作成のユーザー・メッセージ出口ルーチンでサポートされます。このメッセージ・フォーマットは、HWSJAVA0、HWSSOAP1、HWSCSLO0、および HWSCSLO1 ではサポートされません。

ユーザー作成のメッセージ出口ルーチンを使用している場合、ユーザー出口の要件に応じてメッセージ・フィールドの構造を作成できます。ユーザー・メッセージ出口のみが使用するフィールドや、MID 名などの OTMA ヘッダーで渡すことができるその他の項目を組み込むことができます。

以下に、すべてのユーザー作成クライアント・アプリケーションのデータ構造を示します。

ユーザー作成の IMS Connect クライアント・アプリケーションからの入力メッセージ内のトランザクション・データは、ユーザー定義の IRM あるいは IRM 拡張 (拡張が定義されている場合) の後にあり、1 つ以上のデータ・セグメントに含まれています。このデータ・セグメントは、LLZZDATA のフォーマットに準拠する必要があります。ここで、LL にはデータ・セグメントの合計の長さ、ZZ には 2 進ゼロ、DATA にはトランザクション・コードとトランザクション・データ、または IMS コマンドを取めます。

ユーザー作成 IMS Connect クライアント・アプリケーションからの入力メッセージ内のトランザクション・データの後は、ユーザー定義の IRM が続き、1 つ以上のデータ・セグメントに取められます。このデータ・セグメントは、LLZZDATA のフォーマットに準拠する必要があります。ここで、LL にはデータ・セグメントの合計の長さ、ZZ には 2 進ゼロ、DATA にはトランザクション・コードとトランザクション・データ、または IMS コマンドを取めます。

IMS コマンド入力は、EOM の後に続く単一のデータ・セグメントでサブミットする必要があります。

表 26. ユーザー作成クライアント・アプリケーションの IMS Connect メッセージ・フォーマット

フィールド	長さ	意味
llll	4 バイト	llll フィールドを含む全メッセージの長さ

表 26. ユーザー作成クライアント・アプリケーションの IMS Connect メッセージ・フォーマット (続き)

フィールド	長さ	意味
IRM の固定接頭部	28 バイト	IMS.SDFSMAC データ・セット内の HWSIMSCB マクロによってマップされる、IRM の共通固定部分。フィールドについての説明は、224 ページの『IMS Connect へ送信されるメッセージ内の IRM の固定部分のフォーマット』を参照してください。
ユーザー定義 IRM	可変長	IMS.SDFSMAC データ・セット内の HWSIMSCB マクロによってマップされる、IRM のユーザー定義部分。フィールドについての説明は、229 ページの『HWSSMPLO、HWSSMPL1、およびユーザー作成メッセージ出口ルーチンの IRM のユーザー部分のフォーマット』を参照してください。  HWSJAVA0、HWSSOAP1、HWSCSLO0、および HWSCSLO1 は、IRM のユーザー定義部分をサポートしない。
IRM 拡張子	可変長	IRM DSECT の拡張を必要とせずに IMS Connect に情報を送信できるようにする IRM の部分。フィールドについての説明は、237 ページの『IRM 拡張のフォーマット』を参照してください。
LL	2 バイト	1 番目のデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	最初にトランザクション・コードを持つユーザー・データ
LL	2 バイト	2 番目のデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データの 2 番目のデータ・セグメント (トランザクション・コードなし)
...		
LL	2 バイト	このデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ・セグメント (トランザクション・コードなし)
LL	2 バイト	メッセージ終結 (2 進数 0000 0000 0000 0100 にセットされる)
zz	2 バイト	予約済み (2 進ゼロにセットされている)

#### 関連資料

224 ページの『IMS Connect へ送信されるメッセージ内の IRM の固定部分のフォーマット』

IMS 要求メッセージ (IRM) ヘッダーには、IMS TM と通信するすべての IMS Connect クライアント・アプリケーションからのすべてのメッセージに共通の 28 バイトの固定フォーマット・セクションがあります。

## メッセージ出口から戻される入力メッセージ

メッセージ出口からの入力メッセージは、IMS TM Resource Adapter およびユーザー作成 IMS Connect クライアント・アプリケーションのメッセージ構造で構成されます。

### IMS TM Resource Adapter メッセージ構造 - タイプ 1

IMS Connect がサポートする IMS TM Resource Adapter 出口出力メッセージ・フォーマットは、入力メッセージのメッセージ・フォーマットと同じです。

メッセージの全長は 10,000,000 バイトとすることができます。メッセージ内の各セグメントの長さ (BPE ヘッダーから次の BPE ヘッダーまで) は、BPE ヘッダーと OTMA ヘッダーを除いて、最大 32 KB です。

## ユーザー作成の IMS Connect クライアント・アプリケーションのメッセージ構造 - タイプ 3

以下の表は、提供されている HWSSMPL0 出口 (ユーザー作成の IMS Connect クライアント・アプリケーション出口ルーチン) からの、IMS Connect がサポートする出力メッセージのフォーマットを示しています。この表では、フィールド、長さ、および意味について記載しています。可変長の OTMA ヘッダーがサポートされるので、OTMA ヘッダーの長さは 466 バイト でなくてもかまいません。以下の例では、提供出口が使用する 466 バイト になっています。

表 27. HWSSMPL0 出口でサポートされる出力メッセージ・フォーマット

フィールド	長さ	意味
BPE HEADER	64 バイト	BPE ヘッダーのフォーマットについては、 <a href="#">248 ページの『BPE ヘッダー・フォーマット』</a> を参照してください。
OTMA HDR	466 バイトから 970 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、 <a href="#">261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』</a> を参照してください。OTMA ヘッダーのセキュリティー・データ・セクションにネットワーク・セキュリティー情報が含まれている場合、ヘッダーのサイズは、ネットワーク・セキュリティー情報が含まれていない場合より最大で 504 バイト大きくなる可能性があります。
LL	2 バイト	1 番目のデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	最初にトランザクション・コードを持つユーザー・データ
ll、zz、DATA の反復		
LL	2 バイト	このデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ (トランザクション・コードなし)
yy	2 バイト	2 進数値ゼロ
BPE HEADER	64 バイト	BPE ヘッダーのフォーマットについては、 <a href="#">248 ページの『BPE ヘッダー・フォーマット』</a> を参照してください。
OTMA CTL HDR	32 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、 <a href="#">261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』</a> を参照してください。

表 27. HWSMPL0 出口でサポートされる出力メッセージ・フォーマット (続き)

フィールド	長さ	意味
LL	2 バイト	このデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ・セグメント (トランザクション・コードなし)
LL、zz、DATA の反復		
LL	2 バイト	このデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ (トランザクション・コードなし)
...		
LL	2 バイト	このデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ (トランザクション・コードなし)
yy	2 バイト	2 進数値ゼロ
BPE HEADER	64 バイト	BPE ヘッダーのフォーマットについては、248 ページの『 <a href="#">BPE ヘッダー・フォーマット</a> 』を参照してください。
OTMA CTL HDR	32 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、261 ページの『 <a href="#">第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド</a> 』を参照してください。
LL	2 バイト	このデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ (トランザクション・コードなし)
...		
LL	2 バイト	このデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ (トランザクション・コードなし)
yy	2 バイト	2 進数値ゼロ

**制約事項:** 1つの BPE ヘッダーから次の BPE ヘッダー までのデータの長さは、32K を超えてはなりません。ただし、その BPE ヘッダーと OTMA ヘッダーは除きます。

## BPE ヘッダー・フォーマット

以下の表は、BPE ヘッダー・フォーマット内のフィールドの長さおよび意味を示しています。

**制約事項:** ユーザー・メッセージ出口ルーチンでは、チェーン・ポインター・フィールドを除いて、BPE ヘッダー内のどのフィールドも変更してはなりません。ユーザー・メッセージ出口ルーチンは、チェーン・ポインター・フィールドを変更して、BPE ヘッダーと一緒にチェーンングし、最後の BPE チェーン・ポインターを 2 進ゼロに設定します。

このトピックには**プロダクト・センシティブ・プログラミング・インターフェース**情報が含まれています。

表 28. BPE ヘッダー・フォーマット

フィールド	長さ	意味
llll	4 バイト	全バッファのフィールドの長さ
CHAIN PTR	4 バイト	次の BPE ヘッダーを指すチェーン・ポインター
STORAGE TYPE	8 バイト	ストレージ・タイプ
TYPE ACCESS	4 バイト	タイプ・アクセス
SUBPOOL	1 バイト	サブプール
RESV	43 バイト	予約済み

### 関連資料

243 ページの『クライアントからの入力メッセージおよび出口に渡される入力メッセージ』  
クライアントからの入力メッセージは、IMS TM Resource Adapter およびユーザー作成 IMS Connect クライアント・アプリケーションのメッセージ構造で構成されます。

## メッセージ出口に渡される出力メッセージ

IMS Connect からの出力メッセージは、出口が IMS TM Resource Adapter クライアント用であるか、ユーザー作成の IMS Connect クライアント・アプリケーション用であるかに関係なく、特定のメッセージ構造でユーザー・メッセージ出口に渡されます。

以下の表は、IMS Connect からクライアントの出口へのメッセージ・フォーマットを示しています。この表では、出力メッセージ内のフィールドの長さおよび意味について記載しています。

表 29. IMS Connect からユーザー・メッセージ出口への出力メッセージのフォーマット

フィールド	長さ	意味
OTMA HDR	OTMA ヘッダーの全長	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』を参照してください。
制御データ (同期コールアウト要求に適用)	n バイト	IMS アプリケーション・プログラムによって送信される制御データ
LL	2 バイト	データ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)

表 29. IMS Connect からユーザー・メッセージ出口への出力メッセージのフォーマット (続き)

フィールド	長さ	意味
DATA	n バイト	ユーザー・データ
OTMA CTL HDR	32 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』を参照してください。
LL	2 バイト	このセグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ
...		
OTMA CTL HDR	32 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』を参照してください。
LL	2 バイト	このセグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ

#### 関連資料

[IMS TM Resource Adapter ユーザー・メッセージ出口ルーチン \(HWSJAVA0\) \(出口ルーチン\)](#)

### メッセージ出口からクライアントへの出力メッセージ

メッセージ構造のフォーマットは、IMS Connect クライアントのタイプとクライアントのサポートに使用されるユーザー・メッセージ出口によって異なります。

このトピックには**プロダクト・センシティブ・プログラミング・インターフェース**情報が含まれています。

IMS Connect クライアント・アプリケーションに送信されるメッセージ出口からの出力メッセージのフォーマットは、以下のいずれか 1 つです。

- [249 ページの『IMS TM Resource Adapter メッセージ構造』](#)
- [251 ページの『ユーザー作成のアプリケーションのメッセージ構造』](#)

### IMS TM Resource Adapter メッセージ構造

以下の表は、ユーザー・メッセージ出口 HWSJAVA0 からの出力メッセージのフォーマットを示します。この表では、出力メッセージ内のフィールドの長さおよび意味について記載しています。

表 30. ユーザー・メッセージ出口 HWSJAVA0 からの出力メッセージのフォーマット

フィールド	長さ	意味
LLLL	4 バイト	メッセージの全長

表 30. ユーザー・メッセージ出口 HWSJAVA0 からの出力メッセージのフォーマット (続き)

フィールド	長さ	意味
Id	8 バイト	*HWSJAV*
OTMA HDR	466 バイトから 970 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』を参照してください。OTMA ヘッダーのセキュリティー・セクションにネットワーク・セキュリティー情報が含まれている場合、ヘッダーのサイズは、その情報が含まれていない場合より最大で 504 バイト大きくなる可能性があります。
制御データ (同期コールアウト要求に適用)	n バイト	IMS アプリケーション・プログラムによって送信される制御データ
LL	2 バイト	データ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ
OTMA CTL HDR	32 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』を参照してください。
LL	2 バイト	このデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ
...		
OTMA CTL HDR	32 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』を参照してください。
LL	2 バイト	このセグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ



表 30. ユーザー・メッセージ出口 HWSJAVA0 からの出力メッセージのフォーマット (続き)

フィールド	長さ	意味
OTMA CTL HDR	32 バイト	IMS.SDFSMAC データ・セット内の HWSOMPFX マクロによってマップされる。フィールドについての説明は、261 ページの『第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド』を参照してください。
LL	2 バイト	データ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ
...		
LL	2 バイト	このデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ

### ユーザー作成のアプリケーションのメッセージ構造

ユーザー作成の IMS Connect クライアント・アプリケーションのメッセージ構造は、1 つ以上の TCP/IP メッセージ構造から構成できます。

サンプル・ユーザー・メッセージ出口ルーチン HWSSMPL1 (\*SAMPL1\*) がクライアントに返すメッセージ構造は、LLLL から開始されます。これは 4 バイトのフィールドで、メッセージの合計の長さを収めます。サンプル・ユーザー・メッセージ出口ルーチン HWSSMPL0 (\*SAMPLE\*) では、これが返すメッセージ構造の先頭に LLLL フィールドは追加されません。

ユーザー作成の IMS Connect クライアント・アプリケーションに送信されるメッセージ出口からの出力メッセージのフォーマットは、以下のいずれか 1 つです。

#### MFS MOD 名が要求される場合のメッセージ・フォーマット

以下の表は、MFS MOD 名が要求される場合にクライアント・アプリケーションに送信される出力メッセージのフォーマットを示しています。この表では、出力メッセージ内のフィールドの長さおよび意味について記載しています。

表 31. RMM、DATA、および CSM を含む出力メッセージ・フォーマット

フィールド	長さ	意味
LLLL (HWSSMPL1 のみ)	4 バイト	HWSSMPL1 が返す出力メッセージの合計の長さ。HWSSMPL0 が返す出力メッセージには、このフィールドは含まれない。
RMM ヘッダー (オプション)	20 バイト	RMM の先頭 4 バイトは LLzz で、RMM の 2 バイトの長さとして 2 進ゼロの 2 バイトとなる。
CORTKN ヘッダー (同期コールアウト要求に適用)	52 バイト	この同期コールアウト要求の相関トークン
制御データ (同期コールアウト要求に適用)	n バイト	IMS アプリケーション・プログラムによって送信される制御データ

表 31. RMM、DATA、および CSM を含む出力メッセージ・フォーマット (続き)

フィールド	長さ	意味
LL	2 バイト	データ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ
LL	2 バイト	2 番目のデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データの 2 番目のデータ・セグメント
...		
LL	2 バイト	n 番目のセグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データの n 番目のデータ・セグメント
CSM	12 バイト	完了状況メッセージ

**MFS MOD 名が要求されない場合のメッセージ・フォーマット**

以下の表は、MFS MOD 名が要求されず、データと CSM のみが送信される場合に、クライアント・アプリケーションに送信される出力メッセージのフォーマットを示しています。この表では、出力メッセージ内のフィールドの長さおよび意味について記載しています。

表 32. 出力データおよび CSM のみを含む出力メッセージ・フォーマット

フィールド	長さ	意味
LLLL (HWSSMPL1 のみ)	4 バイト	HWSSMPL1 が返す出力メッセージの合計の長さ。HWSSMPL0 が返す出力メッセージには、このフィールドは含まれない。
CORTKN ヘッダー (同期コールアウト要求に適用)	52 バイト	この同期コールアウト要求の相関トークン
制御データ (同期コールアウト要求に適用)	n バイト	IMS アプリケーション・プログラムによって送信される制御データ
LL	2 バイト	データ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データ
LL	2 バイト	2 番目のデータ・セグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データの 2 番目のデータ・セグメント

表 32. 出力データおよび CSM のみを含む出力メッセージ・フォーマット (続き)

フィールド	長さ	意味
...		
LL	2 バイト	n 番目のセグメントの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
DATA	n バイト	ユーザー・データの n 番目のデータ・セグメント
CSM	12 バイト	完了状況メッセージ

### IMS コマンド応答メッセージのフォーマット

IMS Common Service Layer (CSL) の Operations Manager (OM) コンポーネントに対してコマンドをサブミットするクライアントに送信される出力メッセージを、以下の表に示します。この表では、出力メッセージ内のフィールドの長さおよび意味について記載しています。

表 33. OM コマンド・クライアントに送信される出力メッセージ・フォーマット

フィールド	長さ	意味
LLLL	4 バイト	出力の長さ
DATA	n バイト	IMS コマンド出力

### 要求 MOD メッセージのフォーマット

MFS MOD 名が要求され、データ出力が存在する場合、IMS Connect の MOD 要求メッセージ (RMM) は、メッセージ出口からの出力メッセージの最初の構造として返されます。

以下の表は、ユーザー・メッセージ出口 HWSSMPL0、および HWSSMPL1 によって構築される MOD 要求メッセージの出力メッセージ・フォーマットを示しています。この表には、フィールド名、フィールド長、およびフィールドの意味が含まれています。

表 34. MOD 要求メッセージの出力メッセージ・フォーマット

フィールド	長さ	意味
LL	2 バイト	RMM メッセージの長さ
zz	2 バイト	予約済み (2 進ゼロにセットされている)
ID	8 バイト	*REQMOD* の文字値
MOD	8 バイト	要求された MFS MOD 名の文字値

### 返されるクライアント ID メッセージのフォーマット

IMS Connect が返すクライアント ID メッセージ (GENCID) は、クライアント ID が要求されて出力が存在する場合、メッセージ出口からの出力メッセージの最初の構造 (RMM セグメントがない場合) または 2 番目の構造 (RMM セグメントの直後) として返されます。

次の表は、ユーザー・メッセージ出口 HWSSMPL0、HWSSMPL1、および HWSDPWR1 によって作成される、要求されたクライアント ID の出力メッセージ・フォーマットを示しています。この表には、フィールド名、フィールド長、およびフィールドの意味が含まれています。

表 35. 要求クライアント ID メッセージのフォーマット

フィールド	長さ	意味
LL	2 バイト	GENCID メッセージの長さ

表 35. 要求クライアント ID メッセージのフォーマット (続き)

フィールド	長さ	意味
ZZ	2 バイト	予約済み (2 進ゼロにセットされている)
ID	8 バイト	(*GENCID*) の文字値
CLIENT ID	8 バイト	クライアント ID 名の文字値

### 完了状況メッセージのフォーマット

IMS Connect の完了状況メッセージ (CSM) は、入力メッセージが正常に処理された場合に、メッセージ出口からの出力メッセージの最後の構造として返されます。

以下の表は、CSM の出力メッセージ・フォーマットを示しています。この表には、フィールド名、フィールド長、およびフィールドの意味が含まれています。

表 36. 完了状況メッセージの出力メッセージ・フォーマット

フィールド	長さ	意味
CSM_LEN	2 バイト	CSM メッセージの長さ
CSM_FLG1	1 バイト	フラグ・バイト: <b>X'80'</b> IMS でキューに入れられた非同期メッセージ <b>X'40'</b> 会話型出力メッセージ <b>X'20'</b> ACK/NAK が必要 <b>X'10'</b> 使用可能なプロトコル・レベル
CSM_PRLVLFLG	1 バイト	IMS Connect プロトコル・レベル・フラグ: <b>X'00'</b> 基本プロトコル・レベル。 <b>X'01'</b> 予約済み。 <b>X'02'</b> CM0 ACK NOWAIT トランザクションのサポートが有効。
CSM_ID	8 バイト	*CSMOKY* の文字値

### 要求状況メッセージのフォーマット

IMS Connect 要求状況メッセージ (RSM) は、IMS Connect またはメッセージ出口がエラーが発生したと判別した場合に、メッセージ出口からの唯一の出力メッセージ構造として返されます。

以下の表は、エラー状態の場合の状況要求メッセージの出力メッセージ・フォーマットを示しています。この表には、フィールド名、フィールド長、およびフィールドの意味が含まれています。

表 37. 要求状況メッセージの出力メッセージ・フォーマット

フィールド	長さ	意味
RSM_LEN	2 バイト	RSM メッセージの長さ

表 37. 要求状況メッセージの出力メッセージ・フォーマット (続き)

フィールド	長さ	意味
RSM_FLG1	1 バイト	フラグ・バイト: <b>X'80'</b> IMS でキューに入れられた非同 期メッセージ <b>X'40'</b> 会話型出力メッセージ <b>X'20'</b> ACK/NAK が必要
RSM_RACFRC または RSM_OTMRSN	1 バイト	この RSM が RACF セキュリティー・ エラーによって生成された場合、こ れは RSM_RACFRC フィールドであ り、 <b>RACROUTE REQUEST=VERIFY</b> コマンドからの戻りコードが入りま ず。 それ以外の場合、これは RSM_OTMRSN フィールドであり、 OTMA からの理由コードが入りま ず。
RSM_ID	8 バイト	*REQSTS* の文字値
RSM_RETCOD	4 バイト	戻りコード
RSM_RSNCOD	4 バイト	理由コード

### PING 応答のフォーマット

PING IMS\_CONNECT (ping 要求) を IMS Connect に送信すると、PING 応答が IMS Connect クライアントに送信されます。

次の表は、ユーザー・メッセージ出口 HWSJAVA0 によって作成される PING 応答データのフォーマットを示しています。この表には、フィールド名、フィールド長、およびフィールドの意味が含まれています。

表 38. PING 応答出力データ・フォーマット

フィールド	長さ	意味
PNG_LEN	2 バイト	PING 応答データの長さ
PNG_FLG1	1 バイト	フラグ・バイト: <b>X'02'</b> IMS Connect は、IMS Transaction Message リソース・ アダプター (IMS TM リソース・ アダプター) が送信する OTMA ヘッダーのセキュリティ・セ クションにあるネットワーク・ セキュリティー資格情報をサポ ートする。 <b>X'01'</b> IMS 拡張子の存在に対するサポ ート (IRM_ARCH=5 が必要)
PNG_FLG2	1 バイト	予約済み

表 38. PING 応答出力データ・フォーマット (続き)

フィールド	長さ	意味
PNG_RESP	15 バイト	「*PING RESPONSE*」を含む場合がある
ORG	PNG_RESP	PNG_RESP の再マップ
PNG_RESP1	25 バイト	「HWSC0030I *PING RESPONSE*」を含む場合がある

### IMS アプリケーションからの同期コールアウト・メッセージの相関トークンのフォーマット

同期コールアウト要求に対する IMS Connect メッセージ出口からの出力メッセージには、コールアウト要求に対する応答を、要求を発行した IMS アプリケーションと相関させるための相関トークンが組み込まれています。

相関トークンは、同期コールアウト要求内の最初の構造であり、その後にオプションの MOD 要求メッセージ (RMM) 構造およびアプリケーション・データが続くことがあります。

以下の表は、IMS Connect からの出力の相関トークンの構造を示しています。これは、HWSIMSCB マクロの CORMask DSECT によってマップされます。

表 39. 出力メッセージ内の同期コールアウト要求のための相関トークンのフォーマット

フィールド	長さ	意味
COR_Len	2 バイト	構造ヘッダーおよび相関トークンを含む、相関構造の長さ
COR_Rsvd	2 バイト	予約済み (2 進ゼロにセットされている)
COR_Id	8 バイト	*CORTKN* の文字値
COR_LL	2 バイト	相関トークンの長さ
COR_PSTNR	2 バイト	同期コールアウト・メッセージの領域 ID
COR_IMSID	4 バイト	コールアウト要求を発行した IMS アプリケーション・プログラムを実行している IMS システムの ID
COR_MEMTK	8 バイト	OTMA TMEMBER トークン
COR_AWETK	8 バイト	OTMA メッセージ・トークン
COR_TPIPE	8 バイト	OTMA TPIPE 名
COR_USERID	8 バイト	コールアウト要求を発行した IMS アプリケーションによる ICAL 呼び出しに含まれるユーザー ID。

### 関連タスク

209 ページの『同期コールアウト要求のためのユーザー作成 IMS Connect クライアントの構成』

同期コールアウト要求をサポートするには、IMS から新しいコールアウト要求をリトリートし、コールアウト要求の受信を確認して (ACK または NAK)、IMS Connect を介して同期コールアウト応答を IMS に返すように、ユーザー作成 IMS Connect クライアントを構成する必要があります。

## メッセージ内のコールアウト制御データのフォーマット

クライアント・アプリケーションに送信される出力メッセージ内のコールアウト制御データのフォーマットは、resume tpipe 要求を発行するクライアント・アプリケーションのタイプによって異なります。

### IMS TM リソース・アダプター以外のクライアントの場合

IMS Connect がクライアント・アプリケーションへ送信する出力メッセージには、制御データを含む構造が組み込まれています。この構造は、マクロ HWSIMSCB に組み込まれた CTLDATA DSECT および CTLDATASEG DSECT を使用してマップすることができます。制御データ構造は、コールアウト要求データの前に置かれます。

制御データ構造のフォーマットは、以下のとおりです。

表 40. IMS TM リソース・アダプター以外のクライアントのコールアウト制御データ・メッセージのフォーマット

フィールド	長さ	意味
CTL_LLLL	4 バイト	制御データ構造の長さ
CTL_ID	8 バイト	制御データ構造 ID。これは *CTLDAT* の文字値です。
zzzz	4 バイト	IMS Connect 用に予約済み
CTL_HDRLEN1	4 バイト	最初の制御データ・セクションのヘッダー (タグ) の長さ。このフィールドの直後に、最初のオープン・タグとクローズ・タグの対が続きます。
CTL_DATA1	d1	最初の制御データ・セクション。これは、<tag1>data1</tag1> というフォーマットです。
CTL_HDRLEN2	4 バイト	2 番目の制御データ・セクションのヘッダー (タグ) の長さ。このフィールドの直後に 2 番目のオープン・タグとクローズ・タグの対が続きます。
CTL_DATA2	d2	2 番目の制御データ・セクション。これは、<tag2>data2</tag2> というフォーマットです。
:	:	:
CTL_HDRLENn	4 バイト	最後の制御データ・セクションのヘッダー (タグ) の長さ。このフィールドの直後に最後のオープン・タグとクローズ・タグの対が続きます。
CTL_DATA <sub>n</sub>	dn	最後の制御データ・セクション。これは、<tag <sub>n</sub> >data <sub>n</sub> </tag <sub>n</sub> > というフォーマットです。

### IMS TM リソース・アダプター・クライアントの場合

制御データ構造のフォーマットは、以下のとおりです。

表 41. IMS TM リソース・アダプター・クライアントのコールアウト制御データ・メッセージのフォーマット

フィールド	長さ	意味
LLLL	4 バイト	制御データ構造の長さ



表 41. IMS TM リソース・アダプター・クライアントのコールアウト制御データ・メッセージのフォーマット (続き)

フィールド	長さ	意味
LLLL	4 バイト	最初の制御データ・セクションのヘッダー (タグ) の長さ。このフィールドの直後に、最初のオープン・タグとクローズ・タグの対が続きます。
DATA	d1	最初の制御データ・セクション。これは、<tag1>data1</tag1> というフォーマットです。
LLLL	4 バイト	2 番目の制御データ・セクションのヘッダー (タグ) の長さ。このフィールドの直後に 2 番目のオープン・タグとクローズ・タグの対が続きます。
DATA	d2	2 番目の制御データ・セクション。これは、<tag2>data2</tag2> というフォーマットです。
:	:	:
LLLL	4 バイト	最後の制御データ・セクションのヘッダー (タグ) の長さ。このフィールドの直後に最後のオープン・タグとクローズ・タグの対が続きます。
DATA	dn	最後の制御データ・セクション。これは、<tagn>datan</tagn> というフォーマットです。

### 返されるネットワーク・セキュリティ・セグメントのフォーマット

ネットワーク・セッション ID (NETSID) とネットワーク・ユーザー ID (NETUID) のセグメントには、ネットワーク・セキュリティ情報が含まれており、クライアントが RESUME TPIPE 呼び出しを発行するとクライアントに返されます。

NETSID セグメントおよび NETUID セグメントは、以下の両方の状態が発生した場合にのみクライアントに返されます。

- IRM 接頭部の IRM\_F6 フィールドに X'80' (IRM\_F6\_NWSE) を指定する。
- ID \*NETUID\* または \*NETSID\* (またはその両方) を持つ IRM 拡張が、IMS Connect クライアント入力メッセージに含まれている。

以下の表は、ネットワーク・セキュリティ情報が含まれる出力セグメントの形式を示しています。この情報は、HWSSMPL0 および HWSSMPL1 ユーザーメッセージ出口によって作成されます。この表には、フィールド名、フィールド長、およびフィールドの意味が含まれています。

フィールド	バイト長	意味
LL	2	NETSID セグメントの長さ
RESERVED	2	予約済み (2 進ゼロにセットされている)
ID	8	(*NETSID*) の文字値
RESERVED	4	予約済み (2 進ゼロにセットされている)

フィールド	バイト長	意味
DATA	n	このセグメントのデータ。このデータの長さは可変です。

フィールド	バイト長	意味
LL	2	NETUID セグメントの長さ
RESERVED	2	予約済み (2 進ゼロにセットされている)
ID	8	(*NETUID*) の文字値
RESERVED	4	予約済み (2 進ゼロにセットされている)
DATA	n	このセグメントのデータ。このデータの長さは可変です。

## 単純な対話でのメッセージ構造の例

以下の例は、IMS Connect とユーザー提供のクライアント・アプリケーション・プログラムとの間の単純な対話でのメッセージ構造を示しています。

それぞれの例で表されるフローは、コミット・モード 1 および 同期レベル = CONFIRM を使用する単純な送受信トランザクションであり、以下の IRM 値で定義されています。

- IRM\_F2 = X'20' (CM1)
- IRM\_F3 = X'01' (同期レベル = confirm)
- IRM\_F4 = X'40' (送受信対話)

### 単純な対話での HWSSMPL0 メッセージ構造

以下の表で示されるフローの例は、サンプル・ユーザー・メッセージ出力ルーチン HWSSMPL0 が使用される場合 (IRM\_ID = \*SAMPLE\*) のユーザー提供 IMS アプリケーション・プログラムと IMS Connect との間の単純な送受信トランザクションです。

表では、次のようになっています。

- フローは表の上部から開始され、下部へと進行します。
- 垂直バーは、各メッセージ内の各種構造を分離します。
- 入力トランザクションおよびトランザクション応答には、1 つ以上の LLZZ データ構造を含めることができます。
- トランザクション応答で示される LL RMM 構造はオプションです。
- 00040000 は、4 バイトのメッセージ終結構造です。

表 44. 単純な対話での HWSSMPL0 のメッセージ構造の例

クライアントの状態	メッセージ構造の例
クライアント送信 (入力トランザクション)	LLLL   LL IRM   LLZZ data   LLZZ data   0004 0000
クライアント受信 (トランザクション応答)	LL RMM   LLZZ data   LLZZ data   LL CSM
クライアント送信 (ACK)	LLLL   LL IRM   0004 0000
クライアント受信 (割り振り解除の確認)	LL RSM

## 単純な対話での HWSSMPL1 メッセージ構造

以下の表で示されるフローの例は、サンプル・ユーザー・メッセージ出口ルーチン HWSSMPL1 が使用される場合 (IRM\_ID = \*SAMPL1\*) のユーザー提供クライアント・アプリケーション・プログラムと IMS Connect との間の単純な送受信トランザクションです。

表では、次のようになっています。

- フローは表の上部から開始され、下部へと進行します。
- 垂直バーは、各メッセージ内の各種構造を分離します。
- 入力トランザクションおよびトランザクション応答には、1つ以上の LLZZ データ構造を含めることができます。
- トランザクション応答で示される LL RMM 構造はオプションです。
- 00040000 は、4 バイトのメッセージ終結構造です。

表 45. 単純な対話での HWSSMPL1 のメッセージ構造の例

クライアントの状態	メッセージ構造の例
クライアント送信 (入力トランザクション)	LLLL   LL IRM   LLZZ data   LLZZ data   0004 0000
クライアント受信 (トランザクション応答)	LLLL   LL RMM   LLZZ data   LLZZ data   LL CSM
クライアント送信 (ACK)	LLLL   LL IRM   0004 0000
クライアント受信 (割り振り解除の確認)	LLLL   LL RSM

# 第 17 章 IMS Connect で使用される OTMA ヘッダー・フィールド

IMS Connect は、OTMA に送信されるメッセージのヘッダー内のフィールドを使用して、処理オプションおよびその他の情報を IMS に伝えます。

OTMA はメッセージ・ヘッダーのフォーマットと有効値を定義します。OTMA メッセージ・ヘッダーの各フィールドは、フィールドの目的に基づいて複数のセクションにグループ化されています。

送信されるメッセージのタイプまたは選択するオプションに応じて、OTMA ヘッダーの所定のセクションのフォーマットを変更できます。例えば、サブミットされるメッセージのタイプに応じて、OTMA ヘッダーのメッセージ制御セクションのフォーマットが異なります。

OTMA ヘッダーのセキュリティー・データ・セクションにネットワーク・セキュリティー資格情報の NETSID セクションまたは NETUID セクション、あるいはその両方が含まれる場合、HWSOMPFIX マクロを使用するコードを変更してヘッダーをマップすることが必要になる可能性があります。ネットワーク・セキュリティー資格情報がセキュリティー・データ・セクションに含まれている場合、セキュリティー・セクションのサイズは異なることがあり、セキュリティー・セクションより下にあるフィールドの位置も変わり、アクセスできなくなる可能性があります。OTMA ヘッダー内のデータにアクセスできるようにヘッダーのセクションを検査したり、変更したりするには、以下のステップを実行します。

1. OMCTLPFL フィールドの値を検査して、メッセージの中に状態データ、セキュリティー・データ、ユーザー・データ、およびアプリケーション・データのセクションが存在しているかどうかを判別します。
2. OTMA ヘッダーの特定のセクションにアクセスするには、OMCTLLEN、OMHDRLEN、OMSECLN、および OMUSRLN の各フィールドの値を使用して (存在する場合)、メッセージの制御データ、状態データ、セキュリティー・データ、およびユーザー・データの各セクションをそれぞれスキップオーバーします。

以下のサブセクションの表には、OTMA ヘッダー・フィールドを設定するための、またはそのフィールドを IMS Connect ユーザー・メッセージ出口と統合するための IMS Connect の要件が記載されています。以下の表は、OTMA ヘッダーの各セクションに対応しています。各表でリストされるフィールドの追加の注記は、最終列の番号で示されています。対応する注記のテキストは、292 ページの『OTMA ヘッダーの表に関する注記』で番号ごとにリストされています。

## 関連資料

889 ページの『OTMA メッセージ接頭語』

OTMA メッセージには、IMS.ADFSMAC データ・セットの DFSYMSG マクロによってマップされるフォーマットに準拠した接頭部があります。

## IMS Connect で使用される OTMA メッセージ制御フィールド

このトピックの表では、OTMA メッセージ制御ヘッダーのフィールドおよびそれらのフィールドの順序を定義します。

表内の番号付きの注の説明は、292 ページの『OTMA ヘッダーの表に関する注記』にあります。

表 46. HWSOMCTL DSECT - OTMA メッセージ制御ヘッダー。全メッセージの制御データ共通セクション

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMCTLALV	1	0		ARCHITECTURE LEVEL X'01' アーキテクチャー・レベル 1 に設定。すべてのメッセージ に対して設定される	1

表 46. HWSOMCTL DSECT - OTMA メッセージ制御ヘッダー。全メッセージの制御データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMCTLMGT	1	1	MESSAGE TYPE		
			OMCTLDTA X'80'	MESSAGE TYPE=データ 会話型トランザクションに対して設定 (最初の入力を除く)。 EXPREA FLAG1 が EXPREA_CONVERS に設定されているときは、OMCTLMGT を OMCTLDTA に設定。 会話のための最初の入力では、EXPREA FLAG1 は EXPREA_CONVERS に設定されない。	1
			OMCTLTXN X'40'	MESSAGE TYPE=トランザクション 最初のトランザクション入力に対して設定。つまり、会話型または非会話型の最初の入力、EXPREA_FLAG1 は EXPREA_CONVERS には設定されない。	1
			OMCTLRSP X'20'	MESSAGE TYPE=応答 クライアントに送信されたメッセージへの ACK または NAK 応答に対して設定。 次の場合に必須: • コミット・モード 0 (同期レベル=CONFIRM) • コミット・モード 1 (同期レベル=CONFIRM)	1
			OMCTLCMD X'10'	MESSAGE TYPE-コマンド 設定する - RESUME TPIPE 呼び出し	1
			OMCTLCMT X'08'	MESSAGE TYPE = コミットの確認 SEND ONLY または DEALLOCATE に対して設定。 SEND ONLY または DEALLOCATE が、クライアントからの IRM で指示される。	1

表 46. HWSOMCTL DSECT - OTMA メッセージ制御ヘッダー。全メッセージの制御データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMCTLRSI	1	2	RESPONSE INDICATOR		
			OMCTLACK X'80'	RESPONSE = ACK	
				ACK に対して設定。ACK が IRM で指示される。	
			OMCTLNAK X'40'	RESPONSE = NAK	1
				NAK に対して設定。NAK が IRM で指示される。	
			OMCTLRREQ X'20'	RESPONSE = 要求される応答	1
				設定された場合は、会話型トランザクションと IMSEA_RSNCODE を 96 (X'60') に設定して、会話を続行することをクライアント・アプリケーションに通知する必要がある。	
OMCTLERQ X'10'	RESPONSE=要求される拡張応答。出口によるテストも設定も行われぬ。	4			
OMCTLSYR X'08'	同期コールアウト・メッセージへの応答				
OMCTLDAN X'02'	遅延 ACK または NAK 応答のサポート				
OMCTLCCI	1	3	COMMIT CONFIRMATION INDICATOR		
			OMCTLCTD X'80'	CONFIRM=コミット済み	
				設定された場合、IMS アプリケーションは、既に会話を終了しているため、IMSEA_RSNCODE を 97 (X'61') に設定して、IMS アプリケーションが正常に終了したことをクライアント・アプリケーションに通知する必要がある。	
OMCTLABT X'40'	Confirm=打ち切り。出口によるテストも設定も行われぬ。	4			

表 46. HWSOMCTL DSECT - OTMA メッセージ制御ヘッダー。全メッセージの制御データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMCTLTYP	1	4	コマンド・タイプ		
			OMCTLBID X'04'	COMMAND=クライアント送信権要求。出口によるテストも設定も行われぬ。	4
			OMCTLAVL X'08'	COMMAND=サーバー使用可能。出口によるテストも設定も行われぬ。	4
			OMCLTRSN X'0C'	Command=再同期。出口によるテストも設定も行われぬ。	4
			OMCTLSPA X'14'	COMMAND=すべての T パイプに対する I/P の延期。出口によるテストも設定も行われぬ。	4
			OMCTLRSA X'18'	COMMAND=すべての T パイプに対する I/P の再開。出口によるテストも設定も行われぬ。	4
			OMCTLSPN X'1C'	COMMAND=名前付き T パイプに対する I/P の延期。出口によるテストも設定も行われぬ。	4
			OMCTLRSM X'20'	COMMAND=名前付き T パイプに対する I/P の再開。出口によるテストも設定も行われぬ。	4
			OMCTLRTP X'24'	COMMAND=オプションなしで名前付き T パイプの O/P を再開。オプションなしの RESUME TPIPE 呼び出しに設定。	1
			OMCTLRID X'28'	COMMAND=オプション付きの単一の T パイプを再開。オプション付きの RESUME TPIPE 呼び出しに設定。	1
			OMCTLMTR X'3C'	COMMAND=リソース状態プロトコル・コマンド。  OTMA は、IMS Connect およびその他の OTMA クライアントに、OTMA クライアントから受け取った入力 IMS システムによってどのように良好に処理されているかを通知するために、このコマンドを送信します。  IMS Connect は、このコマンド内で送信される情報で、出口インターフェース・ブロック・データ・ストア項目 (HWSXIBDS) を更新します。	



表 46. HWSOMCTL DSECT - OTMA メッセージ制御ヘッダー。全メッセージの制御データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMCTLPFG	1	5	処理フラグ		
			OMCTLLPG X'80'	プログラムをロード。出口によるテストも設定も行われない。	4
			OMCTSHDN X'80'	すべての T パイプ (OMCTLSPA) における中断状態の処理は IMS シャットダウンが原因です。	7
			OMCTLSYP X'40'	同期化された T パイプ。出口によるテストも設定も行われない。	4
			OMCTLASY X'20'	非同期 / 非送信請求のキューされたメッセージ。出口によるテストも設定も行われない。	4
			OMCTLERR X'10'	NAK 付きのエラー・メッセージがある。出口によるテストも設定も行われない。	4
			OMCTLQUE X'08'	非同期メッセージが IMS 保留キューにある。  設定するときは、CSM の送信中であれば CSM_FLG1 を CSM_AMSG に設定し、RSM の送信中であれば RSMFLG1 を RSM_AMSG に設定。	4
OMCTLOME X'01'	SCI はエラー・メッセージを出さない。				
OMCTLTNM	8	6		T パイプ名。出口によるテストも設定も行われない。	4
OMCTLCHN	1	E	CHAIN STATE FLAG		
			OMCTLFIC X'80'	チェーン内の最初のもの。チェーン内の最初のメッセージ・セグメントに対して設定。	1
			OMCTLMIC X'40'	チェーン内の真中のもの。チェーン内の最初でも最後でもないメッセージ・セグメントに対して設定。	1
			OMCTLLIC X'20'	チェーン内の最後のもの。チェーン内の最後のメッセージ・セグメントに対して設定。	1
			OMCTLCAN X'10'	このメッセージを取り消す。出口によるテストも設定も行われない。	4

表 46. HWSOMCTL DSECT - OTMA メッセージ制御ヘッダー。全メッセージの制御データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMCTLPFL	1	F	PREFIX FLAG (接頭部フラグ)		
			OMCTLSTD X'80'	状態データが存在する。作成中の OTMA ヘッダーに状態データ・ヘッダーが存在するときに設定。	1
			OMCTLSEC X'40'	セキュリティー・データが存在する。作成中の OTMA ヘッダーにセキュリティー・データ・ヘッダーが存在するときに設定。	1
			OMCTLUSR X'20'	ユーザー・データが存在する。作成中の OTMA ヘッダーにユーザー・データ・ヘッダーが存在するときに設定。	1
			OMCTLAPP X'10'	アプリケーション・データが存在する。作成中の OTMA ヘッダーにアプリケーション・データ・ヘッダーが存在するときに設定。	1
OMCTLSSN	4	10		SEND SEQUENCE NUMBER. NEITHER TESTED NOR SET BY EXIT.	4
OMCTLSNS	4	14		SENSE CODE. この後の、OMCTLSNC と OMCTLRSC を参照。	
		ORG OMCTLSNS			
OMCTLSNC	2	14		SENSE CODE 非ゼロ値の場合は、クライアント・アプリケーションに送信する NAK RSM を構築して、理由コードとしてセンス・コードを RSM に入れて渡し、戻りコードを X'0C' に設定。	1
OMCTLRSC	2	16		REASON CODE NEITHER TESTED NOR SET BY EXIT.	4
OMCTLRSQ	4	18		RECOVERABLE MESSAGE SEQUENCE NUMBER NEITHER TESTED NOR SET BY EXIT.	4

表 46. HWSOMCTL DSECT - OTMA メッセージ制御ヘッダー。全メッセージの制御データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMCTLSEQ	2	1C		SEGMENT SEQUENCE NUMBER	1
	1	1E		RESERVED.	3
OMCTLCDN	1	1F		状態データに OMHDRCTD が設定されている場合、制御データ・セグメントの数 (1 から 255) を指定します。このフィールドは、IMS がメッセージのアプリケーション・データ・セクションで制御データを含む ICAL コールアウト・メッセージを送信する場合にのみ使用されます。	
		ORG OMCTLR SQ			
OMCTLUID	8	18		RESUME TPIPE 呼び出しをサブミットするクライアントのユーザー ID	

#### 関連資料

889 ページの『メッセージ制御情報セクション』

各 OTMA メッセージごとに、OTMA メッセージ接頭語の最初のセクションにメッセージ制御情報を指定する必要があります。

## IMS Connect で使用される OTMA 状態データ・フィールド

このトピックの表では、OTMA 状態データ・ヘッダーのフィールドおよびそれらのフィールドの順序を説明します。

各表内の番号付きの注の説明は、292 ページの『OTMA ヘッダーの表に関する注記』にあります。

### サーバー使用可能コマンドおよびクライアント・ビッド・コマンドのフォーマット

表 47. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。サーバー使用可能コマンドおよびクライアント・ビッド・コマンドのフォーマットの状態データ共通セクション

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMHDRLEN	2	0		STATE DATA LENGTH。状態データ長に設定。	1
OMHDRORG		2		「サーバー使用可能」と「クライアント・ビッド」コマンドに対する状態データ。	

表 47. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。サーバー使用可能コマンドおよびクライアント・ピッド・コマンドのフォーマットの状態データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDRONM	16	2		MEMBER NAME OF ORIGINATING SERVER. NEITHER TESTED NOR SET BY EXIT.	4
OMHDROMT	8	12		MEMBER TOKEN OF COMMAND ORIGINATOR. NEITHER TESTED NOR SET BY EXIT.	4
OMHDRDMT	8	1A		MEMBER TOKEN OF COMMAND DESTINATION. NEITHER TESTED NOR SET BY EXIT.	4
OMHDRUEN	8	22		UNRESOLVED DESTINATION EXIT NAME. NEITHER TESTED NOR SET BY EXIT.	4
OMHDRMBS	2	2A		XCF TRANSMISSION MAX BLOCKSIZE. NEITHER TESTED NOR SET BY EXIT.	4
OMHDRQUE	1	2C	OMHDRCMQ X'80'	CREATE HOLD MESSAGE QUEUE. NEITHER TESTED OR SET BY EXIT.	4
			X'40'	OTMA 呼び出し可能インターフェース用に予約済み。NEITHER TESTED OR SET BY EXIT.	4
			OMHDRICN X'20'	IMS Connect クライアント・タイプ用に予約済み。NEITHER TESTED OR SET BY EXIT.	4
			X'10'	クライアント・タイプ用に予約済み。	3
			X'08'	クライアント・タイプ用に予約済み。	3

表 47. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。サーバー使用可能コマンドおよびクライアント・ピッド・コマンドのフォーマットの状態データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDRFL2	1	2D		CLIENT FLAGS.	3
			X'80'	OMHDRMXI - IMS が同時に処理できるアクティブ・メッセージの数を OTMA が制限することを指定。最大数は OMHDRTIB フィールドに指定される。	
			X'40'	予約済み	
			X'20'	OMHDRTMO - OTMA が OMHDRTO フィールドに指定された間隔で ACK メッセージを受信しない場合に CM1 接続がタイムアウトになる。	
			X'10'	OMHDCSC - synchlevel=2 (同期点) トランザクションで RRS カスケード・トランザクション・サポートを使用することを示すために、IMS Connect によって設定される。	
			X'08'	OMHDRSMI - このクライアントは OMHDRSMN フィールドに指定されたスーパーメンバー・グループのメンバーである。	
			X'04'	OMHDR2SY - このクライアントは IMS アプリケーション・プログラムからの同期コールアウト・メッセージを処理する。	
			X'02'	OMHDRTOQ - CM0 ACK タイムアウト・キュー名。	
			X'01'	OMHDRITI - IMS Connect が OTMA メッセージ用の IMS 間 TCP/IP 通信をサポートすることを示すために、IMS Connect によって設定される。	
OMHDRUAV	4	2E		SAF USER ID TABLE AGING VALUE. NEITHER TESTED OR SET BY EXIT.	4

表 47. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。サーバー使用可能コマンドおよびクライアント・ビッド・コマンドのフォーマットの状態データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDRHTS	4	32		MESSAGE RE-ASSEMBLY HASH TABLE SIZE. NEITHER TESTED OR SET BY EXIT.	4
OMHDRSMN	4	36			
OMHDROSY	2	3A			
OMHDRODE	2	3C		メッセージ接頭語のユーザー・データ・セクション内の OTMA 宛先記述子へのオフセット。	
OMHDRTIB	2	3E			
OMHDRFL3	1	40		クライアント・ビッド・フラグ	
			X'80'	OMHDRRTY - この接続用に MULTIRTP=Y が設定される。T パイプは、複数のアクティブな RESUME TPIPE 要求をサポートします。	
			X'40'	OMHDRRTN - この接続用に MULTIRTP=N が設定される。T パイプは、アクティブな RESUME TPIPE 要求を一度に 1 つだけサポートします。	
OMHDRTO	1	41			
OMHDRTQN	8	42			
		2	ORG OMHDRORG		

### 非同期オプション NO OPTION を選択した場合の、単一の名前付き TPIPE の再開出力

表 48. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。非同期オプション NO OPTION を選択した場合の、単一の名前付き TPIPE の再開出力の状態データ共通セクション

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDCRSM_COUNT	2	2		NUMBER OF TPIPES IN THE ARRAY. RESUME TPIPE 呼び出しからの出力をリトリブするため、T パイプの数に設定。有効な値は 1 個 (1) のみ。	1

表 48. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。非同期オプション NO OPTION を選択した場合の、単一の名前付き TPIPE の再開出力の状態データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDCRSM_TPIPEN	n	4		TPIPE ARRAY. RESUME TPIPE 呼び出しからの 出力をリトリブする ために、T パイプの名前 に設定。名前は 1 個だ け有効。	1
		2	ORG OMHDRORG		

## NOAUTO、SINGLE、WAIT 付き SINGLE、および AUTO の各オプションを選択した場合の、単一の名前付き TPIPE の再開出力

表 49. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。NOAUTO、SINGLE、WAIT 付き SINGLE、および AUTO の各オプションを選択した場合の、単一の名前付き TPIPE の再開出力の状態データ共通セクション

フィールド	長さ	16進 オフセット	フィールド値	説明および設定	注 記
OMHRRHQ	1	2	RESUME TPIPE 呼び出しのメッセージの戻りオプション		
			OMHRRHQ_NOAUTO - X'00'	呼び出しを受け取ったときに IMS キュー上にあるすべてのメッセージを返す。RESUME TPIPE 呼び出しを受け取った後に受信したメッセージは送信されない。キューにメッセージがない場合は、待機しない。HWSIMS00、HWSIMS01、HWSSMPL0、HWSSMPL1 のデフォルト・オプション。	1
			OMHRRHQ_AUTO1 - X'04'	キューから 1 つのメッセージのみを返す。キューにメッセージがない場合、待機して、次に到着したメッセージを送信する。このオプションでは、IMS Connect からの次の出力を待つために、クライアントから IMS Connect への ACK で IRM_TIMER を X'E9' に設定する必要がある。	
			OMHRRHQ_AUTO - X'02'	現在 IMS キュー上にあるすべてのメッセージを返す。現在のすべてのメッセージを送信した後、待機し、新しいメッセージが到着したときに各メッセージを送信する。このオプションでは、IMS Connect からの次の出力を待つために、クライアントから IMS Connect への ACK で IRM_TIMER を X'E9' に設定する必要がある。	1
			OMHRRQ_ONE - X'01'	キューから 1 つのメッセージのみを返す。キューにメッセージがない場合、RESUME TPIPE 要求を取り消す。後続のメッセージを取得するには、新たな RESUME TPIPE 受信シーケンスが必要となる。	1
OMHRRHQ_M ODE	1	3	同期コールアウト・メッセージの処理モード		
			OMHRRHQ_SYNC - X'80'	このクライアントは同期コールアウト・メッセージのみを処理する。	
			OMHRRHQ_SYAS - X'40'	このクライアントは同期コールアウト・メッセージと非同期メッセージの両方を処理する。	
			OMHRRHQ_CTLDATA - X'20'	このクライアントは、制御データを含むコールアウト・メッセージをサポートする。	
			OMHRRHQ_NWSE - X'10'	このクライアントは、ネットワーク・セキュリティ情報が入っている出力メッセージをサポートする。	



表 49. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。NOAUTO、SINGLE、WAIT 付き SINGLE、および AUTO の各オプションを選択した場合の、単一の名前付き TPIPE の再開出力の状態データ共通セクション (続き)

フィールド	長さ	16 進 オフセット	フィールド値	説明および設定	注記
OMHDCRHQ_TP IPEN	8	4		T パイプ名。RESUME TPIPE 呼び出しからの出力をリトリブするために、T パイプの名前に設定。名前は 1 個だけ有効。	1
		2	ORG OMHDRORG		

### トランザクション・メッセージ

表 50. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。トランザクション・メッセージの状態データ共通セクション

フィールド	長さ	16 進 オフセット	フィールド値	説明および設定	注記
OMHDRIST	1	2	OMHDCNV X'80'	IMS STATE FLAG 会話型状態。	2
			OMHRRSP X'40'	応答モード。出口によるテストも設定も行われない。	4
			OMHDMHQ X'20'	保留キューからのメッセージ。	
			OMHRRROT X'08'	転送されたメッセージ	
			OMHNRWS X'02'	RESUME TPIPE 呼び出しを発行したクライアントの出力メッセージには、セキュリティー・ヘッダーにネットワーク・セキュリティー情報が含まれます。	
			OMHDXP1 X'01'	入力トランザクション・メッセージの場合、X'01' は、トランザクション有効期限が、ユーザー・データ・セクション内の OMHRSXP で示されたオフセット位置に指定されていることを示す。	
			OMHRTOE X'01'	RESUME TPIPE 要求および応答の場合、X'01' は、RESUME TPIPE トークンが OMHRTKN フィールドに存在することを示す。	

表 50. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。トランザクション・メッセージの状態データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDRSYN	1	3	OMHDRCTD X'80'	制御データは、メッセージのアプリケーション・データ・セクションに存在する。この値は、OTMA によってアウトバウンド同期コールアウト要求で設定される。	
			OMHDRCM0 X'40'	コミット・モード 0 出口 1 または IRM のデフォルトで、コミット・モード 0 を要求する場合に設定し、フィールド IRM_F2 が "IRM_CMODE0" に設定される。	
			OMHDRCM1 X'20'	コミット・モード 1 出口 1 または IRM のデフォルトで、コミット・モード 1 を要求する場合に設定し、フィールド IRM_F2 が "IRM_CMODE1" に設定される。	
			OMHDRNTX X'10'	転送の通知。NEITHER TESTED NOR SET BY EXIT.	4
			OMHDRSYC X'08'	これは同期コールアウト・メッセージである。	
			OMHDRI2I X'04'	このメッセージは IMS Connect 間の接続を経由してリモート IMS システムの宛先に送られることを示すために、OTMA によって設定される。	

表 50. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。トランザクション・メッセージの状態データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDRSLV	1	4	OMHDRSL0 X'00'	SYNCH LEVEL Synchlevel=0 (None)。出口または IRM のデフォルトが同期レベル NONE を要求する場合に設定し、フィールド IRM_F3 は "IRM_CONFIRM" に設定されない。同期レベル = 0 は、コミット・モード 1 の場合のみ有効。	1
			OMHDRSL1 X'01'	Synchlevel=1 (Confirm)。出口または IRM のデフォルトが同期レベル CONFIRM を要求する場合に設定し、フィールド IRM_F3 は "IRM_CONFIRM" に設定される。Synchlevel=1 は、コミット・モード 0 と 1 の場合に有効。	1
			OMHDRSL2 X'02'	Synchlevel=2 (Syncpt) NEITHER TESTED NOR SET BY EXIT.	

表 50. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。トランザクション・メッセージの状態データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDCFL	1	5	OMHDCFL X'80'	SENDONLY に対して設定。SENDONLY は、クライアントからの IRM で示される。	1
			OMHDCGN X'40'	以下のいずれかを指定する。  <ul style="list-style-type: none"> <li>エージング値は入力メッセージに対して有効である。</li> <li>これは同期コールアウト・メッセージに対する NAK 応答メッセージで、T パイプ保留キューの未配布メッセージが別の RESUME TPIPE によってリトリートされるか、タイムアウトになるまで、このメッセージを保持するように OTMA に指示する。</li> </ul>	
			OMHDCRRQ X'20'	配信不能 CM0 出力の REROUTE REQUEST に対して設定。	
			OMHDCRPND X'10'	PURGE NOT DELIVERABLE に対して設定。SENDONLY と PURGE NOT DELIVERABLE は相互に排他的である。	
			OMHDCREWC X'04'	廃止。EWLM は、IMS によってサポートされなくなりました。このフラグが指定されている場合、OTMA によって無視されます。	
OMHDCMAP	8	6		MAP NAME。クライアント・アプリケーションが MODname を戻すように要求した場合は、出口が、クライアントに戻されているデータの前に RRM を構築する必要があります。MODname (OMHDCMAP) は、出口によって、RRM からフィールド RRM_MODNAME に移動される。	1

表 50. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。トランザクション・メッセージの状態データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDRTOK	16	E		SERVER TOKEN. NEITHER TESTED NOR SET BY EXIT.	4
ORG OMHDRTOK + 4					
OMHDRSXP	2	12		OFFSET IN USER DATA PREFIX TO THE TRAN EXPIRATION UTC	
OMHDRSXQ	2	14		OTMA が同期コールアウト・メッセージの相関トークンを配置したユーザー・データ・セクションのオフセット。このオフセットは IMS Connect で指定され、クライアント・ビッドの OMHDROSY フィールドの OTMA に渡される。	
OMHDRUID	8	16		USERID RETURNED FOR RT REQ	
OMHDCOR	16	1E		CORRELATOR (相関係数)。NEITHER TESTED NOR SET BY EXIT.	4
ORG OMHDCOR + 8					
OMHDRTIM	8	26		タイム・スタンプ相関関係子 ID。送受信メッセージで、このフィールドには、応答を元の入力メッセージに相関させるために使用されるタイム・スタンプが入る。	
OMHDRTKN	8	2E		RESUME TPIPE トークン。RESUME TPIPE 要求および関連の応答で、このフィールドには RESUME TPIPE トークンが入る。	
ORG OMHDRTKN					
OMHDRPDR	8	36		DL/I ICAL を使用した同期コールアウト・メッセージのプログラム名。	
OMHDCID	16	2E		CONTEXT ID. NEITHER TESTED NOR SET BY EXIT.	4

表 50. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。トランザクション・メッセージの状態データ共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMHDRLTM	8	3E		OVERRIDE LTERM NAME. IRM LTERM フィールドから "IRM_LTERM" を設定。	1
OMHDRLIU	2	46		LENGTH OF IMS HEADER USER DATA. その後に続くユーザー・ヘッダー・データの長さを設定。このフィールドの長さは、全体の長さには含まれない。	5
OMHDRIUD	n	48		IMS HEADER USER DATA. 可変長で、ユーザーが設定する。	5
ORG OMHDR TOK+4					
OMHDRSXP	2	12	OTMA ヘッダーのユーザー・データ・セクション内でトランザクション有効期限を配置するオフセット	このフィールドは IMS Connect によって設定される	2

## リソース状態プロトコル・コマンド

OTMA は、IMS サーバー・リソースの現行状態をクライアントに通知するため、またはハートビート・メッセージをクライアントに送信するための 2 つの目的でリソース状態プロトコル・コマンドを使用します。IMS Connect は、現在、ハートビート・メッセージを無視します。

リソース状態プロトコル・コマンドは、OTMA ヘッダーのメッセージ制御情報セクションのコマンド・タイプ・フィールド (OMCTLTYP) に X'3C' が入ることで識別されます。

IMS Connect は、リソース状態プロトコル・コマンドを受け取ると、リソース状態プロトコル・コマンド・メッセージに含まれる情報で、出口インターフェース・ブロック・データ・ストア項目 (HWSXIBDS) を更新します。

以下の表は、リソース状態プロトコル・コマンドの状態データのフォーマットを要約したものです。この要約には、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 51. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。リソース状態プロトコル・コマンド・フォーマットの状態データ共通セクション

内容	長さ	10 進オフセット	16 進オフセット	値
OMHDRLLEN	2	0	X'00	長さフィールド自体を含む、状態データ・セクションの長さ。

表 51. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。リソース状態プロトコル・コマンド・フォーマットの状態データ共通セクション (続き)

内容	長さ	10 進オ フセ ット	16 進オ フセ ット	値
OMHDRSIM_STATUS	2	2	X'02	IMS システム・サーバー内の OTMA リソースの状態。 <b>X'03' - 正常状態</b> IMS は使用可能であり、メッセージを正常に処理している。 OTMA は、以下のタイミングで正常状態プロトコル・コマンドを発行する。 <ul style="list-style-type: none"> <li>OTMA クライアントが新しい T パイプ接続を確立したとき</li> <li>OTMA クライアントが T パイプ接続を再確立したとき</li> <li>60 秒間隔のハートビート・メッセージとして</li> </ul> <b>X'02' - 機能低下状態</b> IMS は、メッセージを低速で処理している。1 つ以上の状態によって IMS が必要な速さでメッセージを処理していないことが示された場合、OTMA は、機能低下状態プロトコル・コマンドを発行します。 <b>X'01' - 使用不可状態</b> IMS は、メッセージを受け入れて処理することができなくなっている。OTMA は、使用不可状態プロトコル・コマンドを発行して、1 つ以上の重大な状態により IMS が OTMA メッセージを処理できないというアラートを OTMA クライアントに出します。
OMHDRSIM_SVRFLG1	1	4	X'04	予約済み
OMHDRSIM_SVRFLG2	1	5	X'05	予約済み
OMHDRSIM_SVRFLG3	1	6	X'06	予約済み
OMHDRSIM_SVRFLG4	1	7	X'07	使用不可リソースの 4 番目のグループのリソース用フラグ。 <b>X'01' - OMHDRSIM_S4FLOOD</b> サーバーは、処理を待っている OTMA メッセージであふれ、使用できなくなっている。
OMHDRSIM_WRNFLG1	1	8	X'08	機能低下リソースの 1 番目のグループのリソース用フラグ。 <b>X'80' - OMHDRSIM_W1FLOOD</b> すべての OTMA クライアントに関する全体的なあふれ警告 <b>X'40' - OMHDRSIM_W1MTP</b> グローバル T パイプ警告。サーバー上の OTMA でモニターされている T パイプの総数が、いずれか 1 つの OTMA クライアントで指定された最高の制限以上である。
OMHDRSIM_WRNFLG2	1	9	X'09	予約済み
OMHDRSIM_WRNFLG3	1	10	X'0A	予約済み

表 51. HWSOMHDR DSECT - OTMA 状態データ・ヘッダー。リソース状態プロトコル・コマンド・フォーマットの状態データ共通セクション (続き)

内容	長さ	10 進オ フセ ット	16 進オ フセ ット	値
OMHDRSIM_WRNFLG4	1	11	X'0 B'	機能低下リソースの 4 番目のグループのリソース用フラグ。 <b>X'08' - OMHDRSIM_W5MTP</b> この OTMA クライアント用の T パイプの数が、OTMA クライアント記述子の MAXTP パラメーターでこのクライアントに設定された T パイプの最大許容数に達しました。T パイプの数が減少するまでは、この OTMA クライアントに対して新規の T パイプは作成できません。 <b>X'04' - OMHDRSIM_W4MTP</b> この OTMA クライアント用の T パイプの数が、OTMA クライアント記述子の MAXTP パラメーターでこのクライアントに対して設定された T パイプの最大許容数の 80% に達しました。 <b>X'02' - OMHDRSIM_AWE</b> メッセージ AWE が 80% のあふれに達している。 <b>X'01' - OMHDRSIM_W4FLOOD</b> このクライアントのみに対するあふれ警告。サーバーで処理を待機中の OTMA メッセージの数が、サーバーに定義された最大許容数の 80% に達しています。
OMHDRSIM_NRSFLGS	1	12	X'0C '	非リソース関連標識に関するその他のフラグ。 <b>X'80' - OMHDRSIM_HB60S</b> このメッセージをハートビート・メッセージとして識別します。サーバーは使用可能であり、リソースの使用量は通常の制限内です。ハートビート・メッセージは、60 秒ごとに送信される。
予約済み	3	13	X'0 D'	
OMHDRSIM_SRVNAME	16	16	X'10 '	OTMA サーバーの 16 文字の z/OS システム間カップリング・ファシリティ (XCF) メンバー名。
OMHDRSIM_CLTNAME	16	32	X'20 '	OTMA クライアントの 16 文字の XCF メンバー名。
予約済み	20	48	X'30 '	
OMHDRSIM_UTC	12	68	X'44 '	このメッセージの UTC 時間

#### 関連資料

899 ページの『状態データ・セクション』



状態データは、すべての OTMA メッセージで必須です。これは、メッセージ接頭語のメッセージ制御情報セクションのすぐ後にあります。トランザクション関連情報が入っています。

## IMS Connect で使用される OTMA セキュリティー・データ・フィールド

このトピックの表では、OTMA セキュリティー・データ・ヘッダーのフィールドおよびそれらのフィールドの順序を定義します。

各表内の番号付きの注の説明は、292 ページの『OTMA ヘッダーの表に関する注記』にあります。

ネットワーク・セキュリティ情報用の DSECT (HWSECDNDS および HWSECARDS) は、HWSOMPFX マクロ内で以下の両方のオプションを指定した場合にのみ、生成されます。

### DSECT=

OTMA ヘッダーの各セクション用に DSECT を個別に生成します。ただし、HWSECDNDS DSECT および HWSECARDS DSECT は生成されません。

### NETSEC\_OPT=YES

DSECT= オプションも指定した場合、HWSECDNDS DSECT および HWSECARDS DSECT が生成されません。

### 全メッセージの共通セキュリティ・データ・セクション

表 52. HWSOMSEC DSECT - OTMA セキュリティー・データ・ヘッダー。全メッセージのセキュリティ・データ共通セクション

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMSECLN	2	0		SECURITY DATA LENGTH	
OMSECFLG	1	2	OMSECNON C'N'	SECURITY FLAG RACF 検査なし。 OTMA RACF 呼び出しが行われない場合は、'N' に設定される。	1
			OMSECCHK C'C'	トランザクションおよびコマンドの検査。 NEITHER TESTED NOR SET BY EXIT.	4
			OMSECFUL C'F'	トランザクション、コマンド、および MPR の検査。 OTMA が RACF 呼び出しを発行している場合は、'F' に設定。	1

表 52. HWSOMSEC DSECT - OTMA セキュリティー・データ・ヘッダー。全メッセージのセキュリティ・データ 共通セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMSECFLN	1	3		LENGTH OF FOLLOWING FIELDS USERID および GROUPID セクションの長さに設定。 <ul style="list-style-type: none"> <li>• USERID だけの場合は X'0A' に設定。</li> <li>• USERID と GROUPID の場合は X'14' に設定。</li> <li>• USERID も GROUPID もない場合は、X'00' に設定。</li> </ul>	1

### 全メッセージの USERID セキュリティー・データ・セクション

表 53. HWSECUDS DSECT - OTMA USERID の定義。全メッセージのセキュリティ・データ USERID セクション

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMSECULN	1	0		LENGTH OF USERID FIELDS USERID フィールドの長さに設定。この長さは、このフィールドに含まれる。USERID があるときは、X'09' に設定。	1
OMSECUTY	1	1	OMSECUXX X'02'	FIELD TYPE USERID タイプ。存在する USERID を識別するために、X'02' に設定。	1
OMSECUID	8	2		USERID IRM フィールド IRM_RACF_USERID から USERID に設定。	1

## 全メッセージの GROUPID セキュリティー・データ・セクション

表 54. HWSECGDS DSECT - OTMA GROUPID の定義。全メッセージのセキュリティ・データ GROUPID セクション

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMSECGLN	1	0		LENGTH OF GROUPID FIELDS  GROUPID フィールドの長さに設定。この長さは、このフィールドに含まれる。GROUPID があるときは、X'09' に設定。	1
OMSECGTY	1	1	OMSECGXX X'02'	FIELD TYPE GROUPID タイプ。存在する GROUPID を識別するために、X'03' に設定。	1
OMSECGRP	8	2		RACF GROUPID。IRM フィールド IRM_RACF_GROUPID からの GROUPID、または IMS Connect 構成ファイルのデフォルトの GROUPID からの GROUPID に設定。	1

## 全メッセージの UTOKEN セキュリティー・データ・セクション

表 55. HWSECFDS DSECT - OTMA RACF UTOKEN の定義。全メッセージのセキュリティ・データ UTOKEN セクション

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMSECRLN	1	0		LENGTH OF UTOKEN FIELDS  UTOKEN フィールドの長さに設定。この長さは、このフィールドに含まれる。ユーザー・セキュリティ出口が RACF 呼び出しを発行した場合は、X'51' に設定。	1
OMSECRTY	1	1	OMSECRTXX X'02'	FIELD TYPE UTOKEN タイプ。  存在する UTOKEN を識別するために、X'00' に設定。	1
OMSECPRF	80	2		UTOKEN  ユーザー・セキュリティ出口から UTOKEN に設定。	1

## 全メッセージの NETUID セキュリティー・データ・セクション

表 56. HWSECDNDS DSECT - すべてのメッセージ用のネットワーク・ユーザー ID (識別名) セキュリティー・データ・セクション

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMSECDNLN	1	0		LENGTH OF NETUID FIELDS. このセクションの長さは可変で、最大値は 247 です。長さには、このフィールドを含みません。	1
OMSECDNTY	1	1	OMSECDNXX X'04'	FIELD TYPE NETUID タイプ。 ネットワーク・ユーザー ID (NETUID) が存在することを示すには、X'04' に設定します。	1
OMSECDN	1 - 246	2		このフィールドのサイズは、1 から 246 バイトの範囲にすることができます。このフィールドには、ID *NETUID* を持つ IRM 拡張のコンテンツが含まれます。	1

## 全メッセージの NETSID セキュリティー・データ・セクション

表 57. HWSECARDS DSECT - すべてのメッセージ用のネットワーク・セッション ID (レルムまたは認証レジストリー) セキュリティー・データ・セクション

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMSECARLN	1	0		LENGTH OF NETSID FIELDS. このセクションの長さは可変で、最大値は 255 です。長さには、このフィールドを含みません。	1
OMSECARTY	1	1	OMSECARXX X'05'	FIELD TYPE NETSID タイプ。 ネットワーク・セッション ID (NETSID) が存在することを示すには、X'05' に設定します。	1

表 57. HWSECARDS DSECT - すべてのメッセージ用のネットワーク・セッション ID (レルムまたは認証レジストリー) セキュリティー・データ・セクション (続き)

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMSECAR	最大 254 バイト	2		このフィールドのサイズは、1 から 254 バイトの範囲にすることができます。このフィールドには、ID *NETSID* を持つ IRM 拡張のコンテンツが含まれます。	1

#### 関連資料

914 ページの『セキュリティ・データ・セクション』

セキュリティ・データ・セクションは、各トランザクションあるいはコマンドで必須で、OTMA プロトコル・コマンドではオプションです。

## IMS Connect で使用される OTMA ユーザー・データ・フィールド

OTMA ヘッダー内のユーザー・データ・フィールドのフォーマットは、HWSOMPFIX マクロ内の HWSOMUSR DSECT によって定義されており、すべての IMS Connect メッセージに共通です。

表内の番号付きの注の説明は、292 ページの『OTMA ヘッダーの表に関する注記』にあります。

表 58. HWSOMUSR DSECT - ユーザー・データ・ヘッダー。全メッセージのユーザー・データ共通セクション

フィールド	長さ	16 進オフセット	フィールド値	説明および設定	注記
OMUSRLN	2	0		ユーザー・データ長。 ユーザー・データ・ヘッダーの長さに設定。	1
	2	2		フルワードの調整。	
OMUSR_DESTID	8	4		宛先 ID。 IRM フィールド IRM_IMSDESTID から宛先 ID (データ・ストア) に設定。	1
OMUSR_ORIGID	8	C		ORIGIN ID. 出口によるテストも設定も行われません。	4
OMUSR_PORTID	8	14		PORTID. 出口によるテストも設定も行われません。	4
OMUSR_LTOKEN	8	1C		LOGON TOKEN. 出口によるテストも設定も行われません。	4
OMUSR_RETCODE	4	24		通信戻りコード。出口によるテストも設定も行われません。	4
OMUSR_RESCODE	8	28		通信理由コード。出口によるテストも設定も行われません。	4

表 58. HWSOMUSR DSECT - ユーザー・データ・ヘッダー。全メッセージのユーザー・データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMUSR_RTOKEN	4	30		RESPONSE TOKEN - A(SVT) COMMUNICATION RETURN CODE。出口によるテストも設定も行われな い。	4
OMUSR_PASSTICK	8	34		RACF PASSWORD/ PASSTICKET。 IRM フィールド IRM_RACF_PW から パスワードに設定。  メッセージを IMS Connect に戻す前 に、このフィールドを消去する必要が ある。	1

表 58. HWSOMUSR DSECT - ユーザー・データ・ヘッダー。全メッセージのユーザー・データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMUSR_FLAG1	1	3C	OMUSR_CONV_OP T EQU X'80'	IMS TM Resource Adapter の場合、サービス指向アーキテクチャーの複合ビジネス・アプリケーション (プロセス・コレオグラフィー) の IMS 会話型トランザクション・サポートを指定する。	4
OMUSR_NPSOCKE T X'40'				非持続ソケット。 出口によるテストも設定も行われ ない。	4
OMUSR_CANCID X'20'				取り消しクライアント ID  新しいセッションを確立したときに、取り消しクライアント ID は、同じポートの既存のセッションが新しいセッションと同じクライアント ID を使用している場合、既存のセッションを終了し、新しいセッションの接続を許可するように要求する。	
OMUSR_PSOCKET X'10'				持続ソケット。  IRM フィールド IRM_SOCT で指定された持続ソケットが IRM_SOCT_PER に設定されている場合に設定。	4
OMUSR_RTALTCID X'08'				代替クライアント ID を使用した RESUME TPIPE 呼び出し。	1
OMUSR_RRDFLT X'04'				使用されるデフォルトの転送名。	4
OMUSR_CANTMR X'02'				タイマー要求をキャンセルします。  これを使用すると、データ・ストアからのデータを待機中に (IRM_TIMER またはデフォルトの構成タイマー値からの) タイマー待機を取り消すことができます。	1
OMUSR_REROUT X'01'				配信不能出力を転送します。  Commit Mode 0 (CM0) 出力が配信不可の場合、このオプションにより、出力は代替 IMS Connect 宛先に転送されます。	1
OMUSR_TRAN X'00'				トランザクション・ソケット。  IRM フィールド IRM_SOCT で指定されたトランザクション・ソケットが IRM_SOCT_TRAN に設定されている場合に設定。	

表 58. HWSOMUSR DSECT - ユーザー・データ・ヘッダー。全メッセージのユーザー・データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMUSR_FLAG2	1	3D	OMUSR_PWDTEXT X'01'	PASSTCKT フィールドがテキスト・パスワードである。  出口によるテストも設定も行われ ない。	4
			OMUSR_PWDBIN X'02'	PASSTCKT フィールドが 2 進数パ ワードである。  出口によるテストも設定も行われ ない。	
			OMUSR_HWSPLSE T X'10'	IMS TM Resource Adapter では、IMS Connect プロトコル・レベルが OMUSR_PROLEV フィールドに指定 されている場合に設定。	
			OMUSR_TRSTUSR X'80'	トラステッド・ユーザーを出口で設定 できる。	
			OMUSR_F2_CIDRE Q X'40'	IMS TM Resource Adapter クライ アントに関して、現行接続の ID が既存 の接続の ID と重複する場合、この接 続に対して IMS Connect が固有のク ライアント ID を生成するよう指定し ます。	
			OMUSR_F2_CIDGE N X'20'	IMS Connect が IMS TM Resource Adapter に送信した応答メッセ ージに、生成されたクライアント ID が 含まれる場合に設定する。	
OMUSR_FLAG3	1	3E	OMUSR_HDRCM0 X'40'	ORIGINAL SYNCHRONIZATION.  コミット・モード 0。  出口によるテストも設定も行われ ない。	4
			OMUSR_HDRCM1 X'20'	コミット・モード 1 (コミット送信)。  出口によるテストも設定も行われ ない。	
			OMUSR_SOA X'08'	確認応答付き送信専用プロトコル・オ プション。	
			OMUSR_SOO X'04'	順次配信付き送信専用プロトコル・オ プション。	
			OMUSR_ALT_ANA K X'02'	ACK または NAK が出力を代替ク ライアント ID に転送することを要求 する。	
			OMUSR_OM_MSG X'01'	存在する OM メッセージ	



表 58. HWSOMUSR DSECT - ユーザー・データ・ヘッダー。全メッセージのユーザー・データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMUSR_TIMER	1	3F	OMUSR_ZERO X'E9' - X'nn' は待機なしを指定する。 OMUSR_FF は永久に待機を指定する。 タイマー値の範囲については <a href="#">320 ページの『入力メッセージのタイムアウトの指定』</a> を参照。	OMUSR_TIMER は IMS Connect が OTMA からの応答を待機する時間を指定する。値は IRM_TIMER バイト、IMS Connect TCPIP 構成ステートメントの TIMEOUT= キーワード、または IMS Connect の内部ロジックによって提供される。	1
OMUSR_USTAT	4	40		USTAT ADDRESS. 出口によるテストも設定も行われな ない。	4
OMUSR_APPL_NM	8	44		ブランクまたは IRM 値に設定された パスチケット APPL 名。	1
OMUSR_RRS_RCD	4	4C		z/OS リソース・リカバリー・サービ ス戻りコード	
OMUSR_ARCLEV	1	50	IMS TM Resource Adapter メッセージのメッセージ構造 のアーキテクチャー・レベル。  OMUSR_AL00 - X'00': 基本アーキ テクチャー・レベル IMS TM Resource Adapter を指示。  OMUSR_AL02 - X'02': アーキテク チャー・レベル 2 を 指示。以下のサポ ートでは必須。  • 配信不能 CMO 出 力の転送  • 代替クライアント ID を使用した RESUME TPIPE 呼び出し  OMUSR_AL03 - X'03': アーキテク チャー・レベル 3 を 指示。MSC および OTMA の IMS 間 TCP/IP 通信をサポ ートするために必 要。		

表 58. HWSOMUSR DSECT - ユーザー・データ・ヘッダー。全メッセージのユーザー・データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMUSR_PROLEV	1	51		OMUSR_HWSPLSET が設定されている場合、このバイトはこの IMS Connect インスタンスのプロトコル・レベルを指定する。	
			OMUSR_PRO0 - X'00'	基本プロトコル・レベル	
			OMUSR_PRO1 - X'01'	現在使用されていない	
			OMUSR_PRO2 - X'02'	CM0 NOWAIT ACK サポート	
			OMUSR_PRO3 - X'03'	RACROUTE ユーザー検査呼び出しからの要求状況メッセージ (RSM) での戻りコードのサポート。	
OMUSR_RES4	1	52		IMS Connect で使用するために予約済み。 出口によるテストも設定も行われ ない。	3
OMUSR_RES5	4	54		IMS Connect で使用するために予約済み。 出口によるテストも設定も行われ ない。	3
OMUSR_FLAG5	1	53	OMUSR_SOCORTIM X'80'	OMHDRTIM 内のメッセージ相関係数値が SendOnly トランザクションに対して有効である。	
OMUSR_RES6	4	58		IMS Connect で使用するために予約済み。 出口によるテストも設定も行われ ない。	3
OMUSR_REROUT_NM	8	5C		クライアント転送名。 IRM_REROUT_NM で指定された転送名に設定。このフィールドは OMUSR_RT_ALTCID フィールドと同じオフセットを使用する。	1
			ORG OMUSR_REROUT_NM		
OMUSR_RT_ALTCID	8	5C		代替クライアント ID。出口によってテストされる。OMUSR_ARCLEV を OMUSR_AL02 に設定する必要もある。このフィールドは OMUSR_REROUT_NM フィールドと同じオフセットを使用する。	6
OMUSR_ADPTNM	8	64		アダプター名。	

表 58. HWSOMUSR DSECT - ユーザー・データ・ヘッダー。全メッセージのユーザー・データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
OMUSR_DRVNM	8	6C		入力ドライバー名。	
OMUSR_LCLIMSID	8	74		メッセージを送信する IMS システムの IMS ID。	
OMUSR_RMTICON	8	7C		ローカル IMS Connect に定義されているリモート IMS Connect 接続 (RMTIMSCON) の名前。	
OMUSR_RMTIMSID	8	84		メッセージを受信する IMS システムの IMS ID。	
OMUSR_RMTTRAN	8	8C		受信側 IMS システムがメッセージの処理をスケジュールするトランザクション・コード。	
OMUSR_RMTUID	8	94		受信側 IMS システムがメッセージのトランザクション処理を許可するために使用するユーザー ID。	
OMUSR_SESTKN	8	9C		IMS Connect が IMS 間メッセージの処理中にイベントを追跡するために使用するセッション・トークン。	
	2	A4		予約済み。フルワードに調整。	
OMUSR_UTC_TO	16	A6		UTC 形式のトランザクション有効期限	
OMUSR_CORTKN	0XL40			同期コールアウト・メッセージの関連トークン	
OMUSR_CT_LEN	2	B6		関連トークンの長さ	
OMUSR_CT_PSTNR	2	B8		同期コールアウト・メッセージの領域	
OMUSR_CT_IMSID	4	BA		IMS ID	
OMUSR_CT_MEMTK	8	BE		OTMA TMEMBER トークン	
OMUSR_CT_AWETK	8	C6		OTMA メッセージ・トークン	
OMUSR_CT_TPIPE	8	CE		OTMA TPIPE 名。	
OMUSR_CT_USERID	8	D6		ICAL 呼び出しからのユーザー ID	
OMUSR_TRCKID_OFF	2	DE		トラッキング ID に対するオフセット。IRM 拡張子の 1 つからユーザー・データ・セクションにコピーされたトラッキング ID を示すためにユーザー・メッセージ出口によって設定されます。	
OMUSR_CONTXT_OF F	2	E0		コンテキスト情報に対するオフセット。このフィールドは、IRM 拡張子の 1 つからユーザー・データ・セクションにコピーされたコンテキスト情報を示すためにユーザー・メッセージ出口によって設定されます。	

表 58. HWSOMUSR DSECT - ユーザー・データ・ヘッダー。全メッセージのユーザー・データ共通セクション (続き)

フィールド	長さ	16進オフセット	フィールド値	説明および設定	注記
	2	E2		IMS Connect で使用するために予約済み。	
	2	E4		IMS Connect で使用するために予約済み。	
OMUSR_RES6	24	E6		IMS Connect で使用するために予約済み。	
予約済み	2	FE		フルワードに調整。	

#### 関連資料

917 ページの『ユーザー・データ・セクション』

OTMA メッセージ接頭語のユーザー・データ・セクションは可変長で、セキュリティー・データ・セクションの後に続きます。どんなデータでも含むことができます。

## OTMA ヘッダーの表に関する注記

以下の注記は、OTMA ヘッダーの各フィールドを記述する表内に示された番号に対応しています。

- 注 1: ユーザー作成出口の READ ルーチンによって設定。
- 注 2: IMS Connect によって設定。
- 注 3: 予約フィールド。
- 注 4: IMS Connect により設定されるもので、ユーザー出口では解析されない。
- 注 5: ユーザー定義の領域。
- 注 6: ユーザー出口によって解析される。
- 注 7: 単なる出力フラグ。

## 第 18 章 IMS Connect プロトコル

IMS Connect は複数のトランザクション・プロトコルを提供します。

### このタスクについて

以下のプロトコルがあります。

- 会話型サポート
- 送信専用
- 非同期出力のための RESUME TPIPE/RECEIVE
- ソケット接続
- 非同期出力サポート

### トランザクションの制約事項および制限

トランザクションの制約事項および制限は、特定のトランザクション・タイプによって異なる可能性があります。

以下は、特定のトランザクションについての制約事項および制限を示します。

- IMS 高速機能、会話型、リカバリー不能トランザクションは、コミット・モード 1 を使用して発行する必要があります。これは、IMS OTMA の制約事項です。
- 非応答トランザクションは、SENDONLY オプションを使用して IMS Connect に送ることができ、トランザクション・ソケットまたは持続ソケットではコミット・モード 0 を使用して発行しなければなりません。

### コミット・モードおよび同期レベルの定義

IMS Connect は、以下のコミット・モードおよび同期レベルをサポートします。

#### コミット・モード 0

コミット・モード 0 (CM0) は、コミット後送信とも呼ばれます。CM0 は、持続ソケットとトランザクション・ソケットの両方でサポートされ、また、同期レベル CONFIRM のみをサポートします。

ご使用のバージョンの IMS Connect がプロトコル・レベル X'02' をサポートしている場合、インバウンド CM0 メッセージで IRM\_F1\_NOWAIT フラグを設定することができます。このフラグは、クライアントがメッセージの受信を確認した後、クライアントは IMS Connect からの最終タイムアウト・メッセージを待たないことを示します。

#### コミット・モード 1

コミット・モード 1 (CM1) は、送信後コミットとも呼ばれます。CM1 は、持続ソケットとトランザクション・ソケットの両方でサポートされ、また、同期レベル NONE、CONFIRM、および SYNCH をサポートします。

CM1 入力メッセージでは、常に IMS が IMS Connect からの ACK および NAK 応答を待機する場合のタイムアウト値を指定します。ACK または NAK 応答を IMS が受け取るまで、IMS 従属領域は占有されたままであり、必要なデータベース・ロックの保持を続けます。有効なタイムアウト間隔は 0 秒から 255 秒までで、これは DATASTORE 構成ステートメントの ACKTO パラメーターで指定します。

現行タイムアウト値を表示するには、IMS Connect コマンド VIEWHWS または VIEWDS を発行するか、あるいは MVS コマンド QUERY MEMBER を発行します。

#### 同期レベル = NONE

同期レベルは、トランザクションごとの確認応答のレベルを示します。トランザクションが同期レベル = NONE と指定された場合、確認応答はクライアントから要求されません。データベース変更は、出力メッセージが IMS Connect に送信されたがクライアントには送信されない場合であっても、依然とし

てコミットされます。しかし、OTMA が出力メッセージを IMS Connect に送信できない場合、入出力メッセージは廃棄され、データベース変更はバックアウトされ、IMS アプリケーションは終了して 119 異常終了で戻ります。

#### 同期レベル = CONFIRM

トランザクションが同期レベル = CONFIRM で指定された場合、クライアントは、出力メッセージがクライアントによって正常に処理されたか (ACK)、不成功だったか (NAK) を IMS Connect に通知するために、確認応答を送信するように要求されます。

CONFIRM の処理は、指定したコミット・モードのタイプによって以下のように異なります。

- 同期レベル = CONFIRM が CMO で要求され、クライアントが ACK で応答した場合、IMS は出力メッセージをデキューします。クライアントが NAK 応答を返した場合、その出力メッセージは、クライアントが後からリトリブできるように、IMS で再キューイングされます。
- 同期レベル = CONFIRM がコミット・モード 1 で要求され、クライアントが ACK で応答した場合、データベース変更はコミット済みです。クライアントが NAK で応答した場合、IMS によって、データベース変更はバックアウトされ、出力メッセージは廃棄されます。

#### 同期レベル = SYNCH

トランザクションが同期レベル = SYNCH と指定された場合、2 フェーズ・コミット処理が必要になります。複数の参加プログラムが同期点処理に関係している場合は、同期レベル = SYNCH を使用してください。同期レベル = SYNCH は、z/OS リソース・リカバリー・サービスによって管理されます。

#### 関連概念

309 ページの『ソケット接続』

IMS Connect は、3 種類のクライアント TCP/IP 接続プロトコルを提供しています。これらをソケットと呼びます。TCP/IP ソケットは、IMS Connect が切断メッセージを送信するときに IMS Connect がクライアント TCP/IP 接続を管理する方法を定義します。

294 ページの『IMS Connect プロトコル・レベル』

IMS Connect プロトコル・レベルは、どのトランザクション機能およびモードを IMS Connect がサポートするように構成されているかを識別します。

## IMS Connect プロトコル・レベル

IMS Connect プロトコル・レベルは、どのトランザクション機能およびモードを IMS Connect がサポートするように構成されているかを識別します。

クライアントに送信される IMS Connect メッセージ出口からの正常に処理された出力メッセージには、完了状況メッセージ (CSM) と呼ばれるセグメントが含まれています。CSM の 3 番目のバイトは、状況フラグです。X'10' のフラグ値は、CSM の 4 番目のバイトが、現行の IMS Connect プロトコル・レベルを含む第 2 のフラグ・バイトであることを示します。

#### X'00'

送受信トランザクションに対する CMO ACK NOWAIT トランザクション・サポートを除く、すべての機能およびモードが使用可能です。

#### X'01'

このプロトコル・レベルは予約済みです。

#### X'02'

他のすべてのサポートに加えて、CMO ACK NOWAIT トランザクション・サポートが使用可能です。

#### X'03'

RACROUTE ユーザー検査呼び出しからの戻りコードは、要求状況メッセージ (RSM) でサポートされません。

CMO ACK NOWAIT サポートが使用可能であることを CSM が示している場合、クライアントは、IMS Connect に送信される入力メッセージの IMS 要求メッセージ (IRM) セグメントに IRM\_F1\_NOWAIT フラグを設定し、X'E9' の IRM\_TIMER 値を指定して、CMO ACK NOWAIT トランザクションをサブミットすることができます。NOWAIT フラグは、クライアントがトランザクションの正常な受信を確認した後、IMS Connect からの最終タイムアウト・メッセージを待たないことを示します。

## 関連資料

229 ページの『HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口ルーチンの IRM のユーザー部分のフォーマット』

HWSSMPL0、HWSSMPL1、またはユーザー作成メッセージ出口によってサポートされるユーザー作成クライアント・アプリケーションでは、IMS Connect クライアント入力メッセージ内の IMS 要求メッセージ (IRM) ヘッダーの 4 バイトの長さフィールドと 28 バイトの固定部分に続けて、IRM にユーザー定義セクションを組み込むことができます。

## IMS Connect 会話型サポート

会話型プログラムは、数ステップから構成されるトランザクションを処理するメッセージ処理プログラム (MPP) です。MPP は、そのトランザクション全体を即時に処理することはありません。

IMS Connect のための会話型サポートによって、特定のクライアントから入るメッセージに対しては、割り込みされない接続 (連続性) を維持できる会話型トランザクションを持つことができます。通常、会話は、メッセージが送信されて、デキューされ、アプリケーション・プログラムが SPA にブランクを入れていれば終了します。あるいは、メッセージの COMMIT CONFIRMED をクライアントから発信したときにも、会話は終了します。IMS Connect の会話型サポートでは、会話には送信後コミット (send-then-commit) モードが必要で、会話はリカバリー不能です。

**要件 :** IMS の SOA 複合ビジネス・アプリケーション・サポートでは、所定の IMS 会話のすべての反復を同じ IMS Connect および同じ IMS が処理する必要があります。より具体的に言うと、所定の IMS 会話のすべての反復は同じ会話型 ID、ポート番号、IMS Connect、およびデータ・ストアを使用する必要があります。これを実現するには、会話のすべての反復に対して同じ接続ファクトリーを使用します。各反復ごとに同じ接続ファクトリーを使用すると、常に同じホスト名、ポート番号、およびデータ・ストアが使用されます。また、IMS TM Resource Adapter のクライアントも同じ IMSInteractionSpec convID プロパティを保管および再利用して、会話の各反復ごとに常に同じ会話型 ID を使用します。

さらに、IMSTCPIPManagedConnection インスタンスで表される物理ソケット接続は、上記の基準を満たしている限り、IMS 会話の各反復ごとに異なるものを使用できます。これらの接続は同じ接続ファクトリーから発信された場合、同じ管理接続プールのメンバーになります。

IMS の SOA 複合ビジネス・アプリケーション・サポートを使用する場合、これらの要件には、z/OS TCP/IP プロファイルの PORT ステートメントの Sysplex Distributor または SHAREPORT キーワードの使用は含まれません。

## OTMA 会話型プロトコル

以下の例は、OTMA 会話型プロトコルを示しています。

### 送信後コミット、同期レベル = NONE

送信後コミット・フローでは、IMS が同期点処理を完了する前に IMS 出力が送信されます。

送信後コミット・フローを使用するには、メッセージ接頭語の状態データ・セクションでコミット・モードを 1 に指定します。

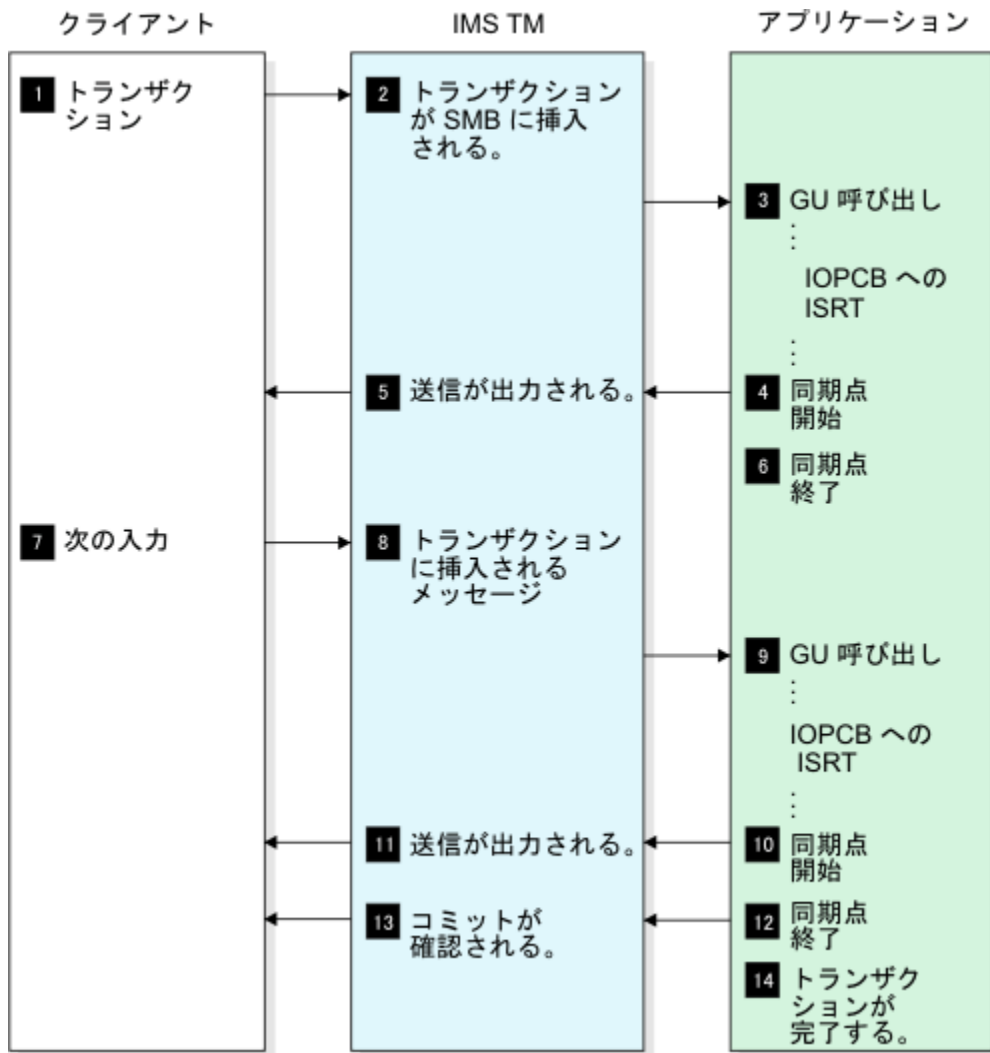


図 25. 送信後コミット、同期レベル = NONE の場合の OTMA 会話型プロトコルのフロー

上記の図に示されているサンプル・フローは、以下を想定しています。

- トランザクション・パイプは同期化されない。
- 同期レベルは、メッセージ接頭語の状態データ・セクションで NONE と指定されている。したがって、IMS は、出力の送信時に応答 (ACK) を要求しません。

### 送信後コミット、同期レベル = CONFIRM

送信後コミット・フローは、IMS で処理されるように、トランザクションに対する同期がないことを想定しています。

以下の図では、トランザクションの受信時に、トランザクションをすべて確認する (各メッセージが応答を要求する) フローを示します。



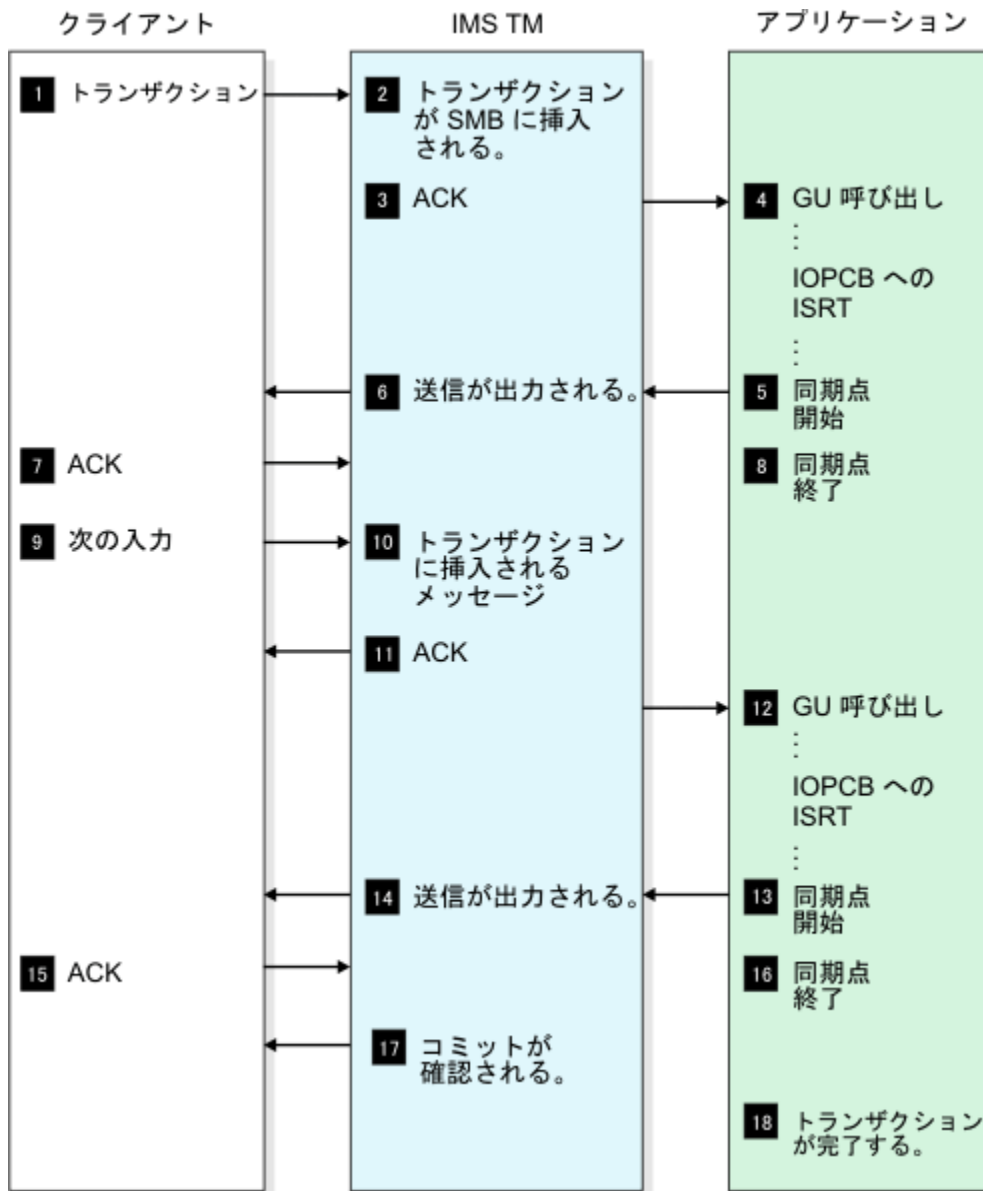


図 26. 送信後コミット、同期レベル = CONFIRM の場合の OTMA 会話型プロトコルのフロー

上記の図に示されているサンプル・フローは、以下を想定しています。

- メッセージ接頭語の状態データ・セクションに、コミット・モード 1 が指定されている。
- トランザクション・パイプは同期化されない。
- 同期レベルは、状態データ・セクションで CONFIRM (確認) と指定されている。

## IMS Connect 会話型プロトコル

IMS Connect 会話型プロトコルは、送信後コミット・プロトコルを使用します。以下の例は、none および confirm の同期レベルを持つ会話型プロトコルのバリエーションを示しています。

### 送信後コミット、同期レベル = NONE、トランザクションはプログラムから終了

送信後コミット・フローは、IMS が同期点処理を完了する前に、IMS 出力を送信します。

送信後コミット・フローを使用するには、メッセージ接頭語の状態データ・セクションでコミット・モードを 1 に指定します。

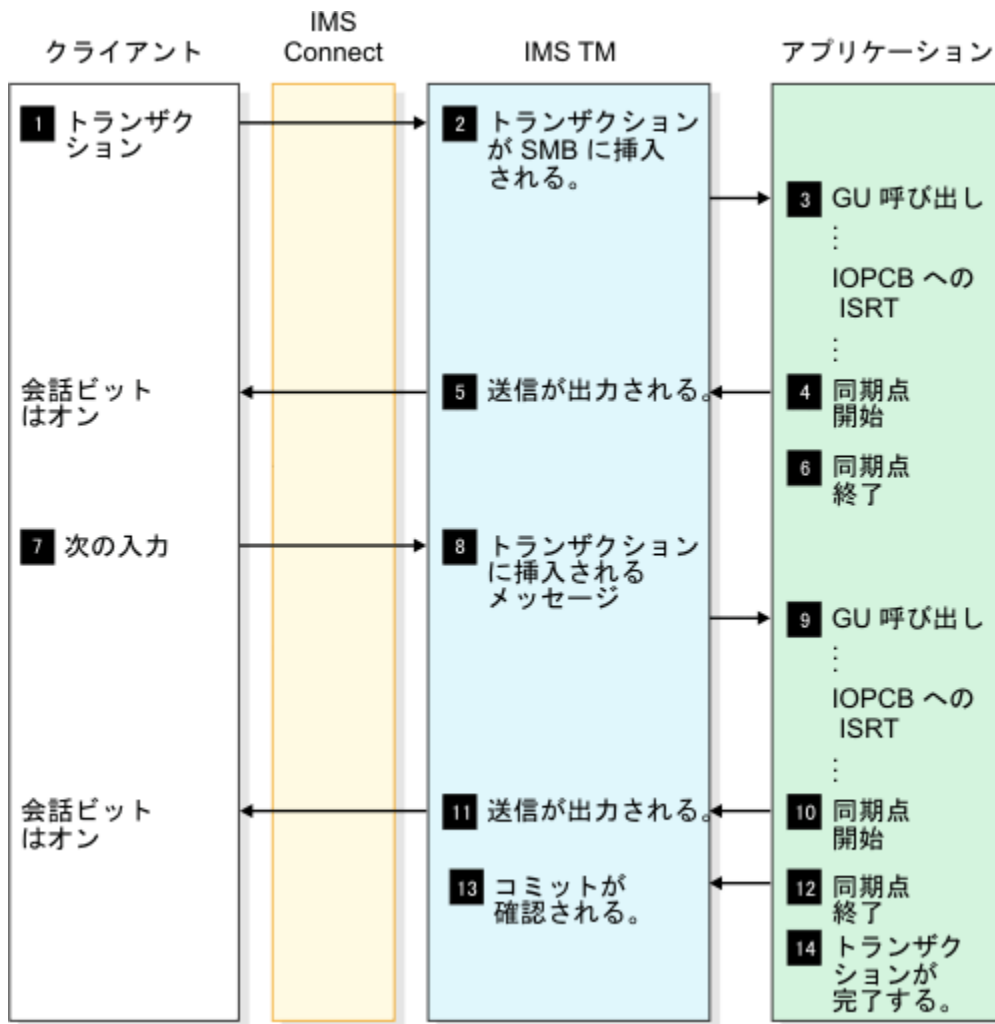


図 27. 送信後コミット、同期レベル = NONE (トランザクションはプログラムから終了) の場合のフロー

上記の図に示されているサンプル・フローは、以下を想定しています。

- トランザクション・パイプは同期化されない。
- 同期レベルは、メッセージ接頭語の状態データ・セクションで NONE と指定されている。したがって、IMS は、出力の送信時に応答 (ACK) を要求しません。
- トランザクションはプログラムから終了される。
- IMS Connect は、「コミット確認」が IMS によって送信されると、ただちにソケットをクローズする。

### 送信後コミット、同期レベル = NONE、トランザクションはクライアントから終了

送信後コミット・フローは、IMS が同期点処理を完了する前に、IMS 出力を送信します。

送信後コミット・フローを使用するには、メッセージ接頭語の状態データ・セクションでコミット・モードを 1 に指定します。

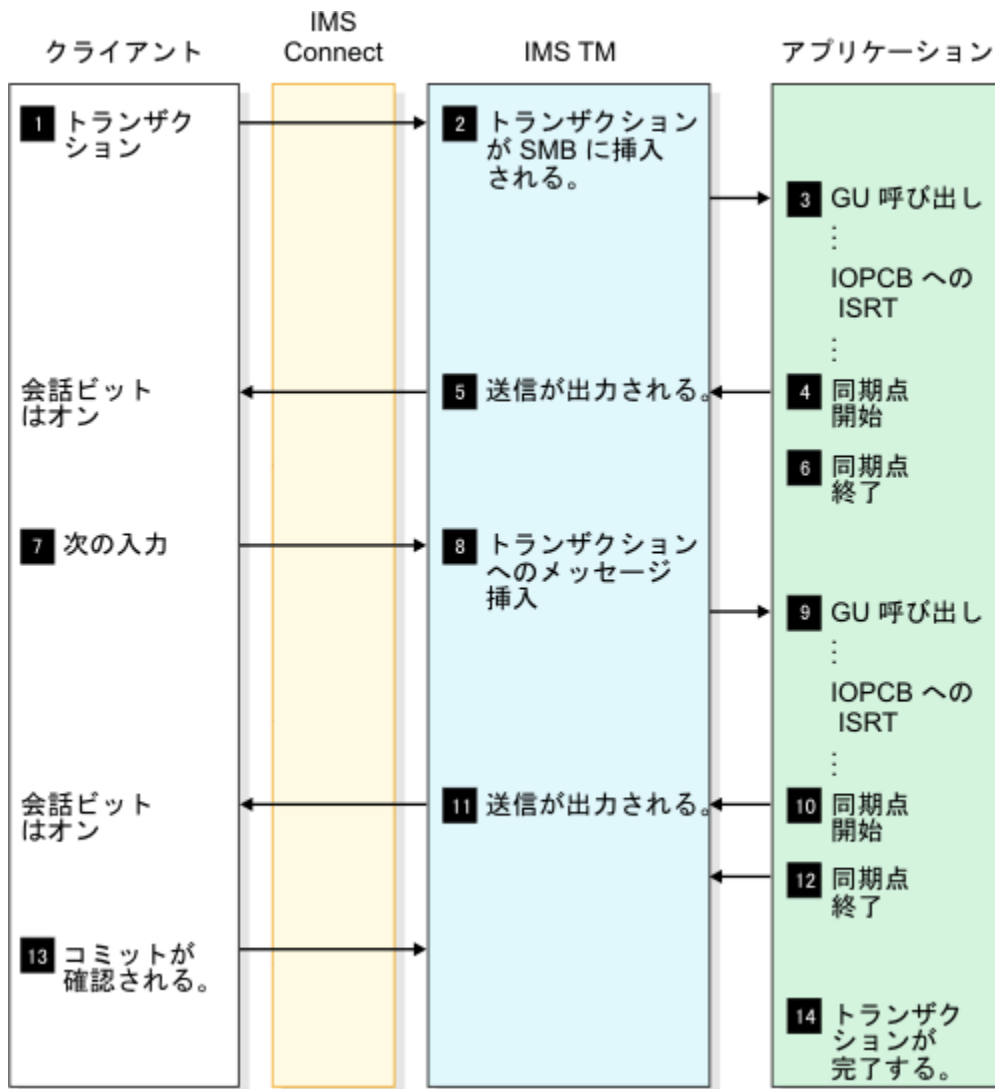


図 28. 送信後コミット、同期レベル = NONE (トランザクションはクライアントから終了) の場合のフロー

上記の図に示されているサンプル・フローは、以下を想定しています。

- トランザクション・パイプは同期化されない。
- 同期レベルは、メッセージ接頭語の状態データ・セクションで NONE と指定されている。したがって、IMS は、出力の送信時に応答 (ACK) を要求しません。
- トランザクションはクライアントから終了される。

### 送信後コミット、同期レベル = CONFIRM (ACK 応答)

この送信後コミット・フローでは、トランザクションが IMS で処理されるため、トランザクションに対する同期が行われないことを想定しています。

以下の図では、トランザクションの受信時に、トランザクションをすべて確認する (各メッセージが応答を要求する) フローを示します。

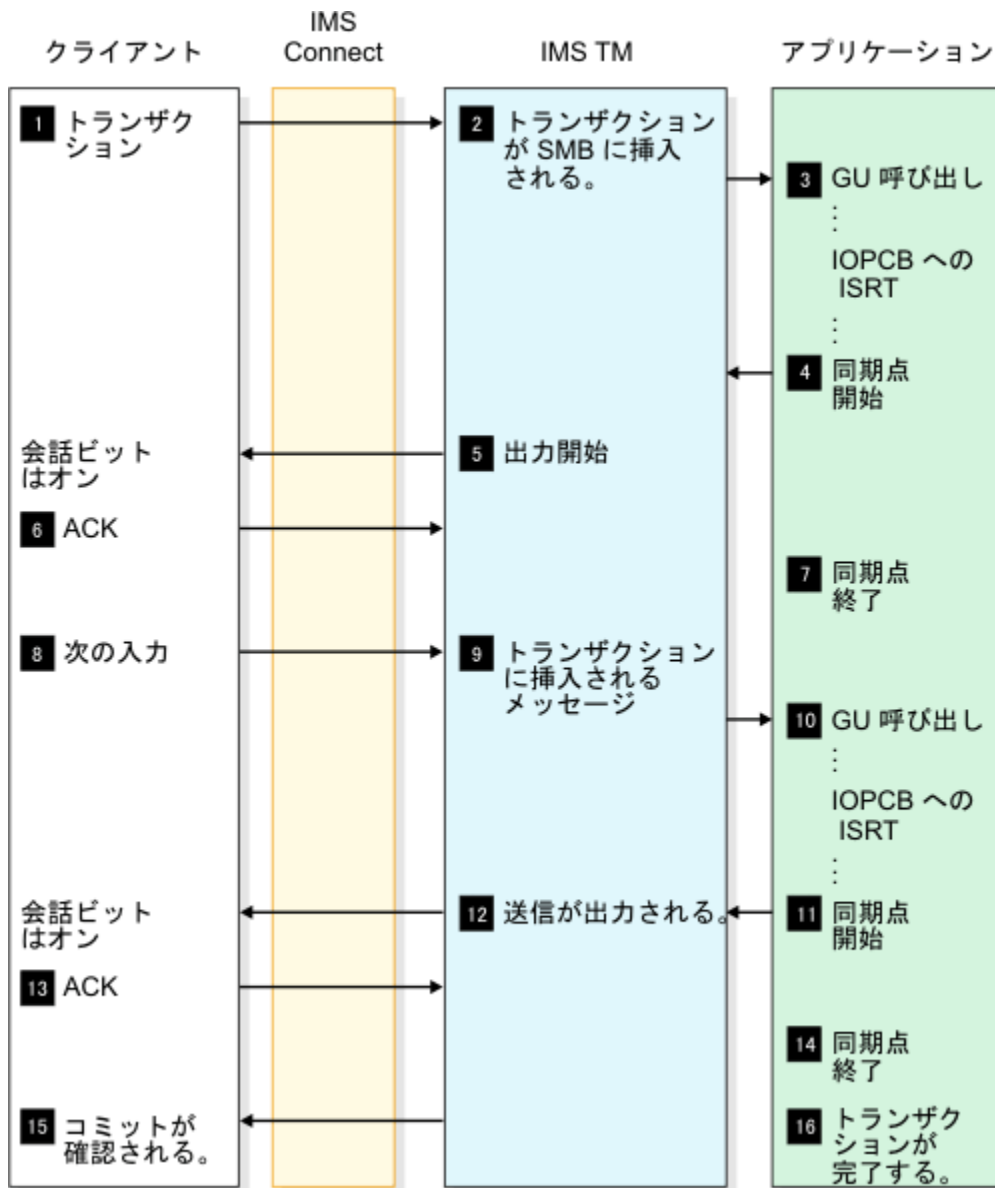


図 29. 送信後コミット、同期レベル = CONFIRM (ACK 応答) の場合のフロー

上記の図に示されているサンプル・フローは、以下を想定しています。

- メッセージ接頭語の状態データ・セクションに、コミット・モード 1 が指定されている。
- トランザクション・パイプは同期化されない。
- 同期レベルは、状態データ・セクションで CONFIRM (確認) と指定されている。
- ACK は、応答検査要求ビットの前に、リモート・ワークステーションから応答される。

### 送信後コミット、同期レベル = CONFIRM (NAK 応答)

この送信後コミット・フローでは、トランザクションが IMS で処理されるため、トランザクションに対する同期が行われないことを想定しています。

以下の図では、トランザクションの受信時に、トランザクションをすべて確認する (各メッセージが応答を要求する) フローを示します。

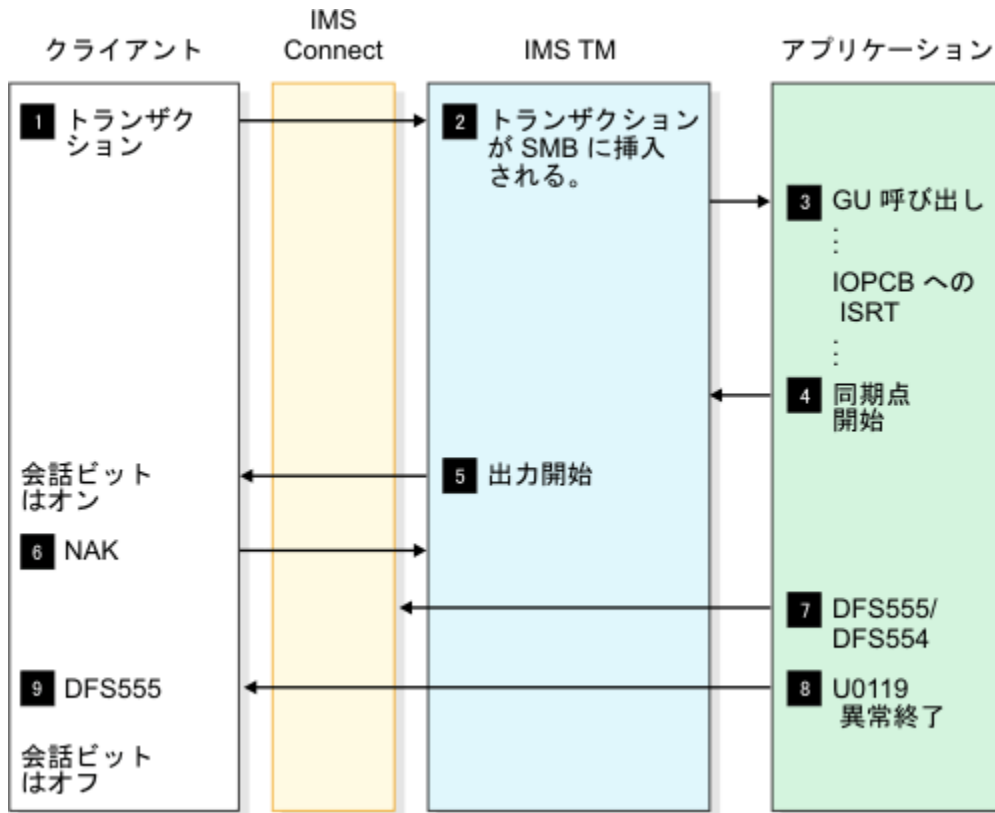


図 30. 送信後コミット、同期レベル = CONFIRM (NAK 応答) の場合のフロー

上記の図に示されているサンプル・フローは、以下を想定しています。

- メッセージ接頭語の状態データ・セクションに、コミット・モード 1 が指定されている。
- トランザクション・パイプは同期化されない。
- 同期レベルは、状態データ・セクションで CONFIRM (確認) と指定されている。
- NAK は、検査要求ビットの前に、IMS Connect またはリモート・ワークステーションから応答される。
- クライアントが、ソケットをクローズする前に NAK/ACK を送信するのを忘れると、IMS Connect は NAK を IMS に送り、U0119 異常終了を起こす。

## 配信不能なコミット後送信出力のページ

トランザクションを開始した OTMA クライアント・アプリケーションにコミット後送信 (コミット・モード 0) IOPCB 出力を返せない場合に、その出力をページするように OTMA を構成することができます。

### このタスクについて

この構成を行うと、OTMA クライアントがページ機能を要求した場合に、OTMA は配信不能なコミット・モード 0 (CM0) IOPCB 出力を IMS 出力キューからデキューし、廃棄します。ページ機能はメッセージ単位で要求されます。

ページ機能が指定されていない場合、IMS は配信不能なコミット後送信 IOPCB 出力を、元の入力メッセージをサブミットしたクライアント・アプリケーションに関連付けられた OTMA T パイプの非同期保留キューに格納します。出力メッセージは保留キューに残り、RESUME TPIPE 呼び出しを使用して後でリトリブされます。

ページ機能は、CM0 とコミット・モード 1 (CM1) のどちらの入力メッセージでも指定できます。ただし、CM1 入力メッセージで指定した場合は、IMS がページするのはプログラム間通信などで生成された CM0 IOPCB 出力のみです。例えば、第 2 トランザクションへのプログラム間通信を行う CM1 トランザクション入力でページ機能を指定した場合、第 1 トランザクションが IOPCB への挿入を行うと、ページ機能は IOPCB に挿入する後続のトランザクションのみに適用されます。

持続ソケットまたはトランザクション・ソケットで、ユーザー作成アプリケーションと IMS TM Resource Adapter アプリケーションの両方から、パージ機能を要求できます。

## 配信不能なコミット後送信出力に対するパージ機能の指定

HWSSMPL0 または HWSSMPL1 IMS Connect ユーザー・メッセージ出口を使用している場合は、特定タイプの入力メッセージの IRM\_F3 フィールドに IRM\_F3\_PURGE フラグ (X'04') を指定することで、配信不能なコミット後送信出力に対するパージ機能を有効にできます。

### このタスクについて

パージ機能は、クライアント・アプリケーションからの以下の入力メッセージに対して指定できます。

- コミット後送信 (CM0) トランザクションの SEND
- 送信後コミット (CM1) トランザクションの SEND CM1 入力のパージ要求は、CM1 入力で生成される CM0 出力のみに適用されます。
- CM0 出力に対する NAK 応答の SEND

**制約事項:** パージ機能と転送機能を同時に指定することはできません。両方の機能を指定すると、出力メッセージは元の出力キューからパージも転送もされず、OTMA がメッセージ DFS2407W を発行します。

## IMS が配信不能なコミット後送信出力をパージする状況

パージ機能が指定されていると、IMS は特定の環境でのみコミット後送信 (コミット・モード 0) 出力をパージします。

パージ機能が指定されていると、IMS は次の場合にコミット後送信 (コミット・モード 0) 出力をパージします。

- IMS Connect がクライアント・アプリケーションから NAK 応答を受信した
- IMS Connect が出力をクライアント・アプリケーションに配信できない
- OTMA が出力を IMS Connect に配信できない

IMS Connect は、クライアントから NAK 応答を受信するか、出力をクライアントに配信できない場合、出力メッセージを IOPCB キューから廃棄するよう OTMA に通知します。OTMA は出力を IMS Connect に配信できない場合、IMS Connect からの通知を待たずに出力を廃棄します。

IMS Connect STOPCLNT コマンドが、パージ機能を指定しているクライアント ID 宛てに発行された場合は、応答メッセージがパージされます。

IMS は、次のタイプの出力に対するパージ機能をサポートしていません。

- ALTPCB 宛ての IMS アプリケーション出力。IMS アプリケーションの CM0 出力に対してパージ機能が指定されていても、パージ機能は ALTPCB への挿入に対して適用されません。
- CM1 出力。配信不能な CM1 トランザクションからのすべての出力は廃棄済みであり、トランザクションはバックアウトされています。
- 送信専用トランザクションに関連する出力。送信専用トランザクションの出力は、非同期保留キューに直接経路指定されます。
- RESUME TPIPE 呼び出し入力メッセージに関連する出力。定義上、RESUME TPIPE 呼び出しでは出力の配信が必要であり、保証されています。

## パージ機能、複数メッセージ出力、および NAK

パージ機能の指定された複数の関連メッセージの 1 つについて、1 つのクライアント・アプリケーションから NAK 応答を受信すると、残りの出力メッセージについては、IMS Connect はクライアント・アプリケーションへの配信を試行せずにパージ要求を発行し、OTMA はそのメッセージを IOPCB から廃棄します。

その複数のメッセージが、単一のアプリケーション・プログラムで生成されたのか、プログラム間通信によって生成されたのかによって、IMS Connect がパージ要求を発行する方法が少し異なります。

IOPCB の 1 つのアプリケーション・プログラムに対し複数のコミット後送信出力メッセージがあり、そのメッセージの 1 つに対してクライアントが NAK 応答を発行した場合、IMS Connect はパージ要求を OTMA に送信します。IOPCB にある残りの出力メッセージについても、IMS Connect はクライアント・アプリケーションへの配信を試行せずに、OTMA にパージ要求を送信します。

IOPCB でのプログラム間通信で生成された複数のコミット後送信出力メッセージがあり、そのメッセージの 1 つに対してクライアントが NAK 応答を発行した場合、IMS Connect はパージ要求を OTMA に渡してから、現在 IOPCB キューにあるその他すべての関連出力メッセージについての追加パージ要求を生成します。最初の NAK 応答の受信後に、プログラム間通信で関連する出力メッセージが生成された場合、IMS Connect はその出力をクライアント・アプリケーションに渡すことなく、それらのメッセージについてもパージ要求を発行します。

## コミット後送信出力の転送

コミット後送信(コミット・モード 0) IOPCB 出力をリトリブのために代替 OTMA T パイプ保留キューに転送するように IMS を構成することができます。

通常、IMS がコミット・モード 0 (CM0) 出力をアプリケーション・クライアントに返すことができない場合は、元のメッセージをサブミットしたクライアント・アプリケーションに関連付けられた T パイプ保留キューに出力が送付されます。ただし、ユーザーが転送機能を要求した場合は、IMS が出力をユーザー指定の T パイプ保留キュー、またはデフォルトの T パイプ保留キューである HWS\$DEF に転送します。ユーザー指定の T パイプまたはデフォルトの T パイプのいずれを転送する場合にも、転送する T パイプは常に元の T パイプの T メンバーに関連付けられます。

転送機能は、送信専用トランザクションで生成された出力の管理、および接続のタイムアウトまたは障害によって元のクライアントに配信できなかった出力の管理に使用できます。

また、転送機能は、IMS TM Resource Adapter (旧称 IMS Connector for Java) を共有可能な持続ソケットと一緒に使用する場合にも便利です。IMS TM Resource Adapter は、IMS Connect に接続するときに、自動的にクライアント ID を生成します。したがって、CM0 出力をリトリブするためにクライアント ID を必要とするクライアント・アプリケーションでは、クライアント ID が不明になります。

転送機能は、CM0 と CM1 のどちらの入力メッセージでも指定できます。ただし、CM1 の場合は、IMS が転送できるのは、プログラム間通信などで生成されることがある CM0 出力のみです。

持続ソケットまたはトランザクション・ソケットで、ユーザー作成アプリケーションと IMS TM Resource Adapter アプリケーションの両方から、転送機能を要求できます。

**制約事項:** 転送機能は、次のものに対してサポートされていません。

- CM1 出力メッセージ
- RESUME TPIPE 呼び出しで生成された出力
- ALTPCB への挿入で生成された出力

## コミット後送信出力に対する転送機能の指定

コミット後送信出力に対して転送機能を有効にするには、入力メッセージの IRM ヘッダーにフラグを設定するか、ユーザー作成ユーザー・メッセージ出口が OTMA 状態データに適切なフラグを設定するようコーディングします。

### このタスクについて

HWSSMPL0 または HWSSMPL1 IMS Connect ユーザー・メッセージ出口を使用している場合は、クライアント・アプリケーションの以下の入力メッセージの IRM\_F3 フィールドに IRM\_F3\_REROUT フラグ (X'08') を指定することで、コミット後送信出力に対する転送機能を有効にできます。

- コミット後送信 (CM0) トランザクションの SEND
- ユーザー作成クライアント・アプリケーションからの CM0 送信専用トランザクションの SEND
- 送信後コミット (CM1) トランザクションの SEND (CM1 入力の転送要求は、CM1 入力で生成される CM0 出力のみに適用されます)



- CMO 出力に対する NAK 応答の SEND

**制約事項:** パージ機能と転送機能を同時に指定することはできません。両方の機能を指定すると、出力メッセージは元の出力キューからパージも転送もされず、OTMA がメッセージ DFS2407W を発行します。

## 転送出力の宛先の指定

転送要求名を指定することにより、転送先を定義できます。出力を転送できるのは、同じ T メンバーに関連付けられている T パイプのみです。転送出力の宛先の指定はオプションです。

### このタスクについて

クライアント・アプリケーションが出力の転送を要求しているが、T パイプ名を指定することで転送先を識別していない場合、デフォルトの転送先は T パイプ HWS\$DEF になります。

次の 1 つ以上の場所で転送要求名を指定すると、転送先を定義できます。

- IMS Connect DATASTORE 構成ファイルの RRNAME= キーワード
- IMS Connect ユーザー・メッセージ出口
- クライアント・アプリケーションからの SEND/RECEIVE 要求に関連付けられた入力メッセージの固定 IRM フォーマット内の IRM\_REROUT\_NM
- クライアント・アプリケーションからの NAK メッセージの固定 IRM フォーマット内の IRM\_REROUT\_NM

初期入力メッセージで指定したものと異なる転送先を NAK 応答メッセージで指定できますが、そのような指定を行うと、複数の出力メッセージを生成するトランザクションで問題が発生することがあります。NAK 応答メッセージで別の転送先を指定し、初期入力メッセージによって複数の出力メッセージが生成された場合、OTMA は NAK 応答をトリガーしたメッセージのみを NAK 応答メッセージで指定された宛先に転送します。NAK 応答の受信後、OTMA は同じトランザクションの後続のすべての出力メッセージを、初期入力メッセージで指定された宛先に自動的に転送します。

以下を参照してください。

- IMS Connect DATASTORE 構成ファイルの RRNAME= キーワードについては、「IMS V15 システム定義」を参照してください。
- IMS Connect ユーザー・メッセージ出口のコーディングについては、「IMS V15 出口ルーチン」を参照してください。

### 関連資料

223 ページの『IMS Connect メッセージ構造』

z/OS プログラム呼び出しインターフェースを使用する TCP/IP クライアントは、各入力メッセージの IMS 要求メッセージ (IRM) ヘッダーを使用して、IMS Connect と通信します。プロトコル・オプションを IMS Connect に伝えるため、IMS Connect クライアント・アプリケーション・プログラムからの入力メッセージで IRM ヘッダーが使用されます。IRM ヘッダーは、IMS Connect HWSIMSCB マクロによってマップされます。

[DATASTORE ステートメント \(システム定義\)](#)

## IMS がコミット後送信出力を転送する状況

転送機能が指定されていると、IMS は特定の環境でのみコミット後送信 (CMO) 出力を転送します。

転送機能が指定されていると、IMS は次の場合にコミット後送信 (CMO) 出力を転送します。

- IMS Connect が出力をクライアント・アプリケーションに配信できない
- IMS Connect がクライアント・アプリケーションから NAK 応答を受信した
- OTMA が出力を IMS Connect に配信できない
- IMS が送信専用トランザクションの出力を IOPCB に挿入する

IMS Connect は、出力を配信できないか、または NAK 応答を受信した場合、出力メッセージを代替の宛先に転送するよう OTMA に通知します。



クライアント・アプリケーションからの NAK 応答の場合、その NAK が IOPCB への IMS アプリケーション挿入に対する応答であれば、OTMA は出力を転送します。NAK が RESUME TPIPE 呼び出しに関連する出力への応答であれば、IMS は出力を転送しません。

OTMA と IMS Connect の間で通信障害が発生した場合、元の入力メッセージが転送機能を要求している場合のみ、OTMA はコミット後送信出力を転送します。

送信専用トランザクションでは、転送機能が指定されていると、OTMA は必ず出力を転送します。

IMS Connect が、出力メッセージに対する ACK または NAK 応答について TCP/IP READ で切断通知を受信した場合、IMS Connect が OTMA に CM0 出力の転送を要求するのは、入力メッセージで転送機能が指定されていたときのみです。IMS Connect が OTMA から出力メッセージを受信する前にクライアントが切断するかタイムアウトになった場合、CM0 出力メッセージが転送されるのは、入力メッセージで転送機能が指定されていたときのみです。

## 転送機能、複数メッセージ出力、および NAK 応答

トランザクションによって複数の出力メッセージが生成される場合、OTMA は最初に NAK 応答をトリガーする出力メッセージ、およびその NAK 応答メッセージを受信したときに IOPCB にある 同じトランザクションの後続のすべての出力メッセージを転送します。

別の NAK 応答のトリガー後に限り、OTMA は最初の転送後に IOPCB に届いた同じトランザクションの後続の出力メッセージを転送します。

トランザクションによってプログラム間通信が開始され、IMS Connect が最初の出力メッセージに対する NAK 応答をアプリケーション・プログラムから受信した場合、OTMA がプログラム間通信後に 2 次側アプリケーション・プログラムによって送信された出力メッセージを転送するのは、最初の NAK 応答メッセージを受信したときにその出力メッセージが既に出力キューに入っていたとき、または元の入力メッセージで転送先が指定されていたときのみです。元の入力メッセージで転送先が指定されていなければ、OTMA はプログラム間通信後に生成された配信不能な出力を転送しません。

送信後コミット (CM1) トランザクション・メッセージが第 2 の CM0 トランザクション・メッセージへのプログラム間通信を行い、第 1 トランザクションが IOPCB への挿入を行う場合、IMS は IOPCB に挿入する第 2 または後続の CM0 メッセージのみを転送します。IMS が CM1 入力メッセージの CM0 出力を転送するのは、元の CM1 トランザクション・メッセージで、クライアント・アプリケーションが転送を要求している場合のみです。

## リカバリー可能 IMS トランザクション

このトピックには、IMS Connect 環境でリカバリー可能トランザクションを実行するときのシナリオがいくつか含まれています。

以下のシナリオでは、次のようになっています。

- 入力メッセージは OTMA によって削除される。
- 入力メッセージの再キューイングは起きない
- CM1 (送信後コミット) では、IMS キューに入れられる (ENQUEUED) 出力は存在しない。

コミット・モード 0 (CM0、またはコミット後送信とも呼ばれます) のみがリカバリー可能として処理されます。CM1 はリカバリー可能ではありません。CM0 を使用すると、IMS Connect は、CM0 を使用するクライアントごとに個別の TPIPE を作成します。この TPIPE は IMS に残存するため、CM0 を使用するクライアントには、固定のクライアント名を使用することを強くお勧めします。

コミット・モードと同期レベルの組み合わせは、重要です。以下のシナリオでは、異なった組み合わせの使い方とその結果を説明します。

- CM1 で同期レベル = NONE の場合

入力メッセージは IMS で処理され、出力メッセージが IMS Connect に送り戻されます。IMS Connect はそのメッセージをクライアントに送ります。この出力メッセージに対する応答でクライアントからの ACK/NAK は、エラーとなります。これは、ACK/NAK は予期しないためであり、IMS Connect はクライアントからアプリケーション・データを持たないメッセージを受け取ります。

- CM1 で同期レベル = CONFIRM の場合

入力メッセージは IMS で処理され、出力メッセージが IMS Connect に送り返されます。IMS Connect はそのメッセージをクライアントに送ります。クライアントから ACK が届くと、アプリケーションは正常に完了したことになります。このシナリオでは、予期した結果になりました。

入力メッセージが IMS で処理され、出力メッセージが IMS Connect に送り返され、IMS Connect がそのメッセージをクライアントに送ります。この後、クライアントから NAK が届くと、IMS MPP 119 異常終了となり、IMS メッセージ DFS555 が出力されます。119 異常終了はデータベース変更をバックアウトします。入力メッセージも出力メッセージも共に廃棄されます。この結果、システムはトランザクションがまったく入ってこなかったのと同じになり、トランザクションの再入力が必要になります。

- CM0 で同期レベル = CONFIRM の場合

入力メッセージは IMS で処理され、アプリケーション・プログラムはこの変更をコミットし、出力メッセージは IMS Connect に送り返されます。IMS Connect はそのメッセージをクライアントに送信し、クライアントから ACK 応答が届くと、IMS は出力をデキューします。これでトランザクション処理は正常終了したことになります。CM0 を選択すると、同期レベルは強制的に CONFIRM になります。このシナリオでは、予期した結果になりました。

入力メッセージが IMS で処理され、出力メッセージが IMS Connect に送り返され、IMS Connect がそのメッセージをクライアントに送ります。この後、クライアントから NAK が届いても、データベース変更はバックアウトされません。入力メッセージは廃棄されますが、出力メッセージは、再表示のために IMS キューに再キューイングされます。このような出力メッセージは、OTMA によって保留非同期キューに移され、RESUME TPIPE、RECEIVE、および ACK 処理によってのみ再びリトリブすることができます。

**推奨:** IMS Connect 環境でリカバリー可能トランザクションを実行するには、CM0 で SYNC LEVEL = CONFIRM を使用してください。また、CM0 で SYNC LEVEL = CONFIRM を使用する各クライアントには、固有の CLIENT\_ID を 1 つ使用してください。

## 送信専用プロトコル

クライアント・アプリケーション・プログラムは、応答を待機することなく、コミット後送信 (CM0) 入力メッセージを IMS に対し連続して高速にサブミットするために送信専用プロトコルを使用します。送信専用プロトコルは、高速で大ボリュームの入力用に設計されています。

同期コールアウト要求をサポートするように構成されているクライアントも、送信専用プロトコルを使用して、外部のデータ・プロバイダーまたはサービス・プロバイダーから、同期コールアウト要求を発行して IMS 従属領域で応答を待機している IMS アプリケーション・プログラムに対して応答を返します。

送信専用入力への応答として IMS で生成された出力は、クライアント・アプリケーションで使用される T パイプに関連付けられた非同期保留キューに保管され、後から RESUME TPIPE 呼び出しを発行してリトリブすることができます。

クライアントが送信する順序で送信専用入力メッセージを IMS で処理する必要がある場合、送信専用プロトコルは以下の 2 つのオプションを提供します。これらのオプションにより、IMS は IMS Connect がクライアント・アプリケーションから受信した順序でメッセージを受信することができます。

- 確認応答付き送信専用プロトコル・オプション
- 順次配信付き送信専用プロトコル・オプション

送信専用プロトコルを使用して、IMS インストール済み環境の外部にあるデータまたはサービス・プロバイダーから応答を返す場合、送信専用メッセージにはトランザクション・コードが含まれませんが、IMS アプリケーションが要求したデータ、またはコールアウト要求の処理で問題が発生した場合には、エラー・コードが含まれます。

以下の図は、コミット確認フラグがオンの場合の CM0 送信専用プロトコルのフローの例を表しています。IMS 標準フローでもある CM0 フローは、IMS 出力をクライアントに送信する前にエンキューします。しかし、この場合の非応答トランザクションでは、クライアントは、IMS からの出力を予期しません。

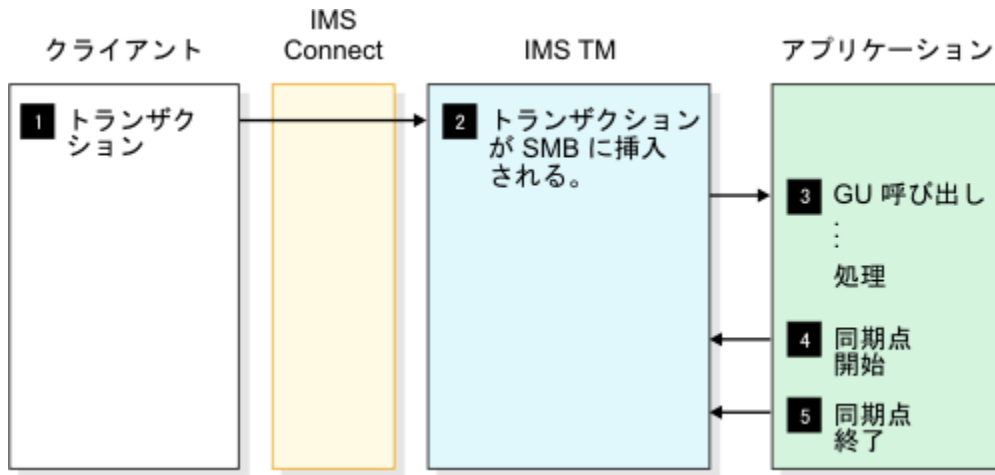


図 31. 送信専用プロトコルのフロー

上記のサンプル・フローは、以下を想定しています。

- メッセージ接頭語の状態データ・セクションに CM0 が指定されている。
- トランザクション・ビットとコミット確認ビットがメッセージ接頭語の制御データ・セクションに指定されている。

## 確認応答付き送信専用プロトコル

確認応答付き送信専用プロトコル (SNDONLYA) は、クライアント・アプリケーション・プログラムがサブミットした順序と同じ順序で、送信専用トランザクション入力を IMS でエンキューするために使用します。

SNDONLYA メッセージは、非応答モードの非会話型トランザクションの送信専用対話を実行します。ホスト・アプリケーションが IO PCB に対して ISRT を発行せずに終了した場合、DFS2082 メッセージはクライアントに返されません。SNDONLYA は、IMS Connect が、出力の ACK または NAK がクライアントから要求されることをその応答内に示す必要があることも指定します。SNDONLYA 対話では、CM0 を使用する必要があります。

SNDONLYA を指定した場合、クライアント・アプリケーションは IMS が正常に入力メッセージをエンキューするごとに、OTMA から ACK 応答メッセージを受信します。それ以外の、送信専用トランザクションによって生成された出力は、すべて非同期保留キューへ送られます。

クライアント・アプリケーションは、次の送信専用入力メッセージを送信する前に、ACK 応答を待ち、それを処理する必要があります。クライアント・アプリケーションから同じ T パイプへ送信された後続の入力メッセージは、前の入力メッセージがエンキューされるまで送信されないため、トランザクション・メッセージは送信された順に IMS にエンキューされます。

確認応答付き送信専用プロトコル・オプションを選択するには、HWSSMPLO および HWSSMPL1 ユーザー・メッセージ出口 IRM フォーマットの IRM\_F4 フィールドで、L を指定します。

確認応答付き送信専用プロトコル・オプションと順次配信付き送信専用プロトコル・オプションは、同時に使用することはできません。両方を指定した場合は、確認応答付き送信専用プロトコルが有効になります。

確認応答付き送信専用プロトコル・オプションは、HWSJAVA0 ではサポートされません。

### 関連概念

#### 順次配信付き送信専用プロトコル・オプション

順次配信付き送信専用プロトコル・オプションを指定した場合、IMS Connect は、送信専用トランザクションを OTMA にサブミットする順序が、実際に IMS がトランザクションを受信する順序になるように処理します。

#### 同期コールアウト応答のための送信専用プロトコル

IMS Connect クライアントは、送信専用プロトコルを使用して、IMS アプリケーション・プログラムからの同期コールアウト要求に対する応答を返します。

## 順次配信付き送信専用プロトコル・オプション

順次配信付き送信専用プロトコル・オプションを指定した場合、IMS Connect は、送信専用トランザクションを OTMA にサブミットする順序が、実際に IMS がトランザクションを受信する順序になるように処理します。

HWSSMPL0 および HWSSMPL1 ユーザー・メッセージ出口 IRM フォーマットの IRM\_F3 フィールドで、IRM\_F3\_ORDER に 'X'10' を指定すると、順次配信付き送信専用プロトコル・オプションを選択できます。

順次配信付き送信専用プロトコル・オプションと確認応答付き送信専用プロトコル・オプションは、同時に使用することはできません。両方を指定した場合は、確認応答付き送信専用プロトコルが有効になります。

順次配信付き送信専用プロトコル・オプションは、HWSJAVA0 ではサポートされません。

以下の要件は、順次配信付き送信専用プロトコル・オプションを使用する場合に送信専用プロトコルに適用されます。

- 順次配信付き送信専用プロトコル・オプションを使用するすべてのトランザクションを SCHDTYP=SERIAL として定義する必要があります。トランザクションを SCHDTYP=PARALLEL と定義した場合、OTMA は IMS によるトランザクションの順次処理を保証できません。
- タイプが同じである他のすべてのトランザクションに対して順次配信する必要がある同じトランザクション・タイプのトランザクションをはじめとする、相互に順次配信する必要のある送信専用トランザクションは、同じ T メンバーおよび T パイプ接続を使用する必要があります。トランザクションを複数の T メンバーおよび T パイプ接続で送信した場合、OTMA は IMS によるトランザクションの順次処理を保証できません。

### 関連概念

#### 確認応答付き送信専用プロトコル

確認応答付き送信専用プロトコル (SNDONLYA) は、クライアント・アプリケーション・プログラムがサブミットした順序と同じ順序で、送信専用トランザクション入力を IMS でエンキューするために使用します。

#### 同期コールアウト応答のための送信専用プロトコル

IMS Connect クライアントは、送信専用プロトコルを使用して、IMS アプリケーション・プログラムからの同期コールアウト要求に対する応答を返します。

## 同期コールアウト応答のための送信専用プロトコル

IMS Connect クライアントは、送信専用プロトコルを使用して、IMS アプリケーション・プログラムからの同期コールアウト要求に対する応答を返します。

IMS Connect クライアントは、応答メッセージの IRM 接頭部の IRM\_F4 フィールドに、同期コールアウト応答のための送信専用プロトコルを指定します。

同期コールアウト応答のための送信専用プロトコルでは、コールアウト応答への確認応答を含めることも、確認応答を無効にして、クライアントが IMS Connect への応答の送信後に受信状態に切り替える必要がないようにすることもできます。

クライアントは、IMS がコールアウト応答を受信した後に確認応答を返すよう IMS に対して要求する場合、IRM メッセージ接頭語の IRM\_F4 フィールドに L を指定することができます。L を指定すると、応答の送達に成功した場合、IMS Connect は CSM (ACK) を返します。応答の送達に失敗した場合、IMS Connect は RSM (NAK) を返します。クライアントは、CSM または RSM をリトリブするために追加の受信を実行する必要があります。これはクライアントのパフォーマンスに影響を与えますが、待機中の IMS アプリケーションへの応答の戻りには影響を与えません。

クライアントが IMS による確認応答の返信を要求しない場合、クライアントは IRM\_F4 フィールドに M を指定して、確認応答を無効にすることができます。確認応答が無効にされている場合、IMS Connect はクライアントに RSM メッセージを返しません。



この送信専用メッセージにはトランザクション・コードは含まれておらず、IMS アプリケーション・プログラムの応答データかエラー情報のいずれかが含まれる可能性があります。コールアウト応答に使用される送信専用メッセージは、出力のタイプを生成しません。

ユーザー作成の IMS Connect クライアントでは、応答メッセージの IRM に以下のフィールド値を指定して、コールアウト応答をコーディングします。

- IRM\_ARCH = X'03' (IRM\_ARCH3)
- IRM\_F0 = X'10' (IRM\_F0\_NAKRSN): NAK 理由コードがエラー応答と一緒に送信される場合
- IRM\_F0 = X'20' (IRM\_F0\_SYNCNAK): NAK 理由コードがエラー応答と一緒に送信されない場合
- IRM\_NAK\_RSNCDE = 2 バイト 16 進数拡張エラー・コード
- IRM\_F4 = L (IRM\_F4\_SYNRESPA) または M (IRM\_F4\_SYNRESP)
- IRM\_CORTKN = 元のコールアウト要求からの 40 バイト関連トークン (CORTKN)

## 関連概念

### 確認応答付き送信専用プロトコル

確認応答付き送信専用プロトコル (SNDONLYA) は、クライアント・アプリケーション・プログラムがサブミットした順序と同じ順序で、送信専用トランザクション入力を IMS でエンキューするために使用します。

### 順次配信付き送信専用プロトコル・オプション

順次配信付き送信専用プロトコル・オプションを指定した場合、IMS Connect は、送信専用トランザクションを OTMA にサブミットする順序が、実際に IMS がトランザクションを受信する順序になるように処理します。

## ソケット接続

IMS Connect は、3 種類のクライアント TCP/IP 接続プロトコルを提供しています。これらをソケットと呼びます。TCP/IP ソケットは、IMS Connect が切断メッセージを送信するときに IMS Connect がクライアント TCP/IP 接続を管理する方法を定義します。

IMS Connect が提供する 3 つのソケット・タイプは、以下のとおりです。

- 持続
- トランザクション
- 非持続

## 持続ソケット

持続ソケットは、クライアントと IMS Connect との間の接続であり、クライアントまたは IMS Connect のいずれかが特に切断要求を出すまで接続状態を保ちます。持続ソケットは、複数トランザクションの間で存在することができます。

クライアントが終了を強制できる方法は 2 通りあります。

- IMS Connect に切断要求を送信する方法
- 入力される最後のトランザクション、例えば、ログオフ・トランザクションのソケット・タイプを「トランザクション」に変更する方法

IMS Connect は、エラーが発生したときも接続を終了できます。

IMS Connect ユーザー・メッセージ出口 HWSSMPL0、HWSSMPL1、HWSSOAP1、および HWSJAVA0 は、持続ソケットの使用をサポートします。

持続ソケットは、コミット・モード 1 (CM1、または送信後コミット) 処理とコミット・モード 0 (CM0、またはコミット後送信) 処理の両方をサポートします。

以下のケースでは、IMS Connect は、持続ソケット上の IMS TM Resource Adapter からの接続に対して固有のクライアント ID を生成します。

- IMS TM Resource Adapter は、IMS Connect に対して、重複するクライアント ID を検査するよう要求し、また新しい接続要求に対して提供された ID が既存の接続のクライアント ID と重複するか検査するよう要求します。
- IMS TM Resource Adapter は、ブランクのクライアント ID を IMS Connect に渡します。

#### 関連概念

[312 ページの『トランザクションのソケット処理』](#)

トランザクション・ソケットまたは持続ソケット上のトランザクションの場合、クライアント・アプリケーションは、すべての TCP/IP SEND に続けて TCP/IP READ を必ず発行する必要があります。

#### 関連タスク

[311 ページの『IMS TM クライアントのソケット・タイプの設定』](#)

IMS TM にアクセスする IMS Connect クライアントの場合、クライアント・コードがソケット設定を制御し、IMS Connect ユーザー・メッセージ出口およびユーザー初期設定出口が、ソケット設定を適用します。

#### 関連資料

[ユーザー・メッセージ出口ルーチン HWSSMPL0 および HWSSMPL1 \(出口ルーチン\)](#)

[IMS TM Resource Adapter ユーザー・メッセージ出口ルーチン \(HWSJAVA0\) \(出口ルーチン\)](#)

[SOAP Gateway 出口ルーチン \(HWSSOAP1\) \(出口ルーチン\)](#)

## トランザクション・ソケット

トランザクション・ソケットは、クライアントと IMS Connect との間の接続で、単一トランザクションまたは IMS 会話の間は接続が保持されているものです。この接続は、IMS 自体が終了するか、またはエラーが発生するかのいずれかの場合に、IMS Connect のみが終了させることができます。

トランザクション・ソケットは、コミット・モード 1 (CM1、または送信後コミット) 処理とコミット・モード 0 (CM0、またはコミット後送信) 処理の両方をサポートします。

#### 関連概念

[312 ページの『トランザクションのソケット処理』](#)

トランザクション・ソケットまたは持続ソケット上のトランザクションの場合、クライアント・アプリケーションは、すべての TCP/IP SEND に続けて TCP/IP READ を必ず発行する必要があります。

#### 関連タスク

[311 ページの『IMS TM クライアントのソケット・タイプの設定』](#)

IMS TM にアクセスする IMS Connect クライアントの場合、クライアント・コードがソケット設定を制御し、IMS Connect ユーザー・メッセージ出口およびユーザー初期設定出口が、ソケット設定を適用します。

## 非持続ソケット

非持続ソケットは、IMS Connect への単一の入出力ペアの間だけ、接続が維持されます。

IMS Connect は、非会話型および会話型のトランザクションのクライアントに出力を送信した後に接続を終了します。入力と出力の交換が 3 回起きたとすると、切断は、IMS Connect からの各出力ごとに一度ずつ、計 3 回出されます。

**制約事項:** HWSSMPL0、HWSSMPL1、HWSSOAP1、および HWSJAVA0 ユーザー・メッセージ出口ルーチンは、非持続ソケットをサポートしません。また、IMS TM Resource Adapter もこれをサポートしません。

#### 関連タスク

[311 ページの『IMS TM クライアントのソケット・タイプの設定』](#)

IMS TM にアクセスする IMS Connect クライアントの場合、クライアント・コードがソケット設定を制御し、IMS Connect ユーザー・メッセージ出口およびユーザー初期設定出口が、ソケット設定を適用します。

## IMS TM クライアントのソケット・タイプの設定

IMS TM にアクセスする IMS Connect クライアントの場合、クライアント・コードがソケット設定を制御し、IMS Connect ユーザー・メッセージ出口およびユーザー初期設定出口が、ソケット設定を適用します。

### このタスクについて

クライアントは、IMS 要求メッセージ (IRM) ヘッダーのフィールド IRM\_SOCT にフラグを設定することによってソケット接続タイプを選択します。IRM\_SOCT フラグの値を、以下の表に示します。

表 59. IRM\_SOCT フラグ

フラグ	定義	ソケット・タイプ
IRM_SOCT_PER	X'10'	持続
IRM_TRAN	X'00'	トランザクション
IRM_SOCT_NONPER	X'40'	非持続

IMS TM にアクセスする IMS Connect クライアントから IMS Connect に送信される各メッセージに、IRM\_SOCT フラグを設定する必要があります。

**推奨:** 単一トランザクションに関連するメッセージは、すべて同じソケット・タイプに設定してください。異なるソケット・タイプに設定すると、以下の例で説明するような、予期しない結果になることがあります。

- 会話型トランザクションの最初のメッセージが持続に設定され、最後のメッセージがトランザクションに設定されていると、このソケット接続は、最後のメッセージの後、終了する。
- 会話型トランザクションの中間のメッセージの 1 つがソケット・タイプをトランザクションに設定されていて、IMS トランザクションが何らかの理由で終了すると、IMS Connect はそのソケットを切断する。これは、「トランザクション」が最後にわかっているソケット・タイプであったからです。

ユーザー・メッセージ出口はソケット・タイプを判別し、次に、IMS Connect に返す OTMA メッセージ・ヘッダーのユーザー・データ・セクションにそのソケット・タイプ情報を移動します。ソケット・タイプ情報を OTMA メッセージ・ヘッダーに転送するために、ユーザー出口は、HWSOMPFX マクロの HWSOMUSR DSECT でマップされる OMUSR\_FLAG1 フィールドを、以下の表に示されるフラグのいずれかに設定します。

表 60. OMUSR\_FLAG1 フラグ

フラグ	定義	ソケット・タイプ
OMUSR_PSOCKET	X'10'	持続
OMUSR_TRAN	X'00'	トランザクション
OMUSR_NPSOCKET	X'40'	非持続

### 関連資料

223 ページの『IMS Connect クライアント・メッセージの IRM 構造』

IMS Connect は、受信したすべてのクライアント・メッセージが合計 4 バイトの長さフィールドで始まり、その後に IMS 要求メッセージ (IRM) ヘッダー、メッセージ・データ・セグメントの順に続くものと想定します。

285 ページの『IMS Connect で使用される OTMA ユーザー・データ・フィールド』

OTMA ヘッダー内のユーザー・データ・フィールドのフォーマットは、HWSOMPFIX マクロ内の HWSOMUSR DSECT によって定義されており、すべての IMS Connect メッセージに共通です。

## IMS 間 TCP/IP 通信のソケット接続

2 つの IMS Connect インスタンスを接続する IMS 間 TCP/IP 通信のソケット接続は、持続または非持続にすることができます。

RMTIMSCON 構成ステートメントの PERSISTENT キーワードを使用して、ソケット接続の持続性を指定します。

接続が MSC 用に使用される場合、接続は持続でなければなりません。PERSISTENT=NO を指定した場合、IMS Connect は警告メッセージを出して、値を PERSISTENT=YES に設定します。

接続が OTMA 用に使用される場合、接続は持続でも非持続でも構いません。しかし、大容量の環境下でリモート IMS システム上の T パイプ数が急増するリスクを避けるために、PERSISTENT=YES を指定してください。

### ソケット終了のシナリオ

WTOR **CLOSEHWS** コマンド、あるいは類似の z/OS MODIFY コマンドまたは IMS タイプ 2 コマンドが発行された場合、IMS Connect は、CONN 状態にあるすべての送信ソケット接続を閉じ、RECV 状態にあるすべての送信ソケット接続を静止します。RECV 状態にあるソケット接続の場合、IMS Connect はリモート IMS Connect からの ACK または NAK の確認応答を無期限に待ちます。RECV 状態にあるすべての接続が確認応答を受信した後でのみ、IMS Connect は終了します。

WTOR **CLOSEHWS FORCE** コマンド、あるいは類似の z/OS MODIFY コマンドまたは IMS タイプ 2 コマンドが発行されると、IMS Connect はすべての送信ソケット接続を閉じます。いずれかの送信ソケット接続が RECV 状態にある場合、IMS Connect はローカル OTMA に NAK を送信し、OTMA は確認応答を受信しなかった送信メッセージをタイムアウト・キューに転送します。

IMS Connect が異常終了した場合、OTMA は確認応答を受信しなかったすべての送信メッセージをタイムアウト・キューに転送します。

MSC ソケット接続では、IMS または SCI が正常終了または異常終了したために MSC との通信が停止した場合、IMS Connect は、影響を受けた MSC 物理リンク上のすべての MSC 論理リンクを終了します。IMS Connect は、終了した MSC 論理リンクで使用されていたすべての送信ソケットおよび受信ソケットを閉じます。影響を受けた MSC 物理リンクの IMS Connect 内の状況が、DISCONNECT に設定されます。

### 関連概念

315 ページの『IMS 間通信用の送信ソケットの予約』

RMTIMSCON 構成ステートメントで RESVSOC パラメーターを指定することにより、IMS 間 TCP/IP 接続で使用する IMS Connect 送信ソケットを予約できます。

[IMS Connect の定義と調整 \(システム定義\)](#)

### 関連資料

[CLOSEHWS コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[RMTIMSCON ステートメント \(システム定義\)](#)

## トランザクションのソケット処理

トランザクション・ソケットまたは持続ソケット上のトランザクションの場合、クライアント・アプリケーションは、すべての TCP/IP SEND に続けて TCP/IP READ を必ず発行する必要があります。

例外は、SENDONLY の TCP/IP SEND、または IRM\_TIMER を NO\_WAIT (X'E9' 文字 Z) に設定した ACK の TCP/IP SEND の場合です。後者は、RESUME\_TPIPE 単一要求の READ に応答して発行されます。

以下のシナリオでは、トランザクション・ソケット上のトランザクションを説明します。持続ソケット上のトランザクションの場合、処理はトランザクション・ソケットのトランザクションと同じです。ただし、クライアント・アプリケーションと IMS Connect は切断しません。また、タイムアウトになった場合、ク



クライアント・アプリケーションは戻りコード X'28' を受け取ります。この戻りコードは、切断が不要であることを示します。

コミット・モード 0、同期レベル CONFIRM、トランザクション・ソケット上の非会話型トランザクションの場合、次のシナリオが考えられます。

1. クライアント・アプリケーションは、IMS Connect にトランザクション・データを送るために SEND を発行する。
2. IMS Connect は、クライアント・アプリケーションに出力を戻す。
3. クライアント・アプリケーションは、その出力を受信し、ACK を送信してから、次の出力またはタイムアウト通知を受信するために READ を発行する。
4. IMS Connect は、トランザクション・ソケットの場合は戻りコード X'20' または X'24'、持続ソケットの場合は X'28' を指定してタイムアウト通知を発行する。IMS Connect は、戻りコード X'20' および X'24' の場合はソケットを切断し、戻りコード X'28' の場合は接続を維持する。
5. クライアント・アプリケーションは、戻りコード X'20' および X'24' の場合、切断要求を発行する。クライアントは、戻りコード X'28' の場合は切断要求を発行するか、または次の入力を送信することができます。

CM1、同期レベル CONFIRM、トランザクション・ソケット上の非会話型トランザクションの場合、次のシナリオが考えられます。

1. クライアント・アプリケーションは、IMS Connect に対してトランザクション・データの SEND を発行する。
2. IMS Connect は、クライアント・アプリケーションに出力を戻す。
3. クライアント・アプリケーションは、出力を受信し、ACK または NAK を送信し、READ を発行する。
4. ACK が送信された後、クライアントは以下のいずれかの応答を受信する。
  - IMS トランザクションが正常に完了した場合、割り振り解除コミット。
  - IMS トランザクションが失敗した場合、DFS メッセージ。
  - トランザクション・ソケットの場合は戻りコード X'20' または X'24'、持続ソケットの場合は戻りコード X'28' が設定されたタイムアウト通知。クライアント・アプリケーションは、戻りコード X'20' および X'24' の場合、切断要求を発行する必要があります。
5. クライアント・アプリケーションは切断要求を発行する。

CM1、同期レベル CONFIRM、トランザクション・ソケット上の会話型トランザクションの場合、次のシナリオが考えられます。

1. クライアント・アプリケーションは、IMS Connect にトランザクション・データを送るために SEND を発行する。
2. IMS Connect は、クライアント・アプリケーションに出力を戻す。
3. クライアント・アプリケーションは、出力を受信し、ACK を送信し、次の入力データを送信する。クライアントは、トランザクションが完了するまで、SEND、READ、ACK を続行します。
4. IMS Connect は、RSM 割り振り解除コミット、割り振り解除の打ち切り、またはタイムアウト通知を発行する。タイムアウト通知は、X'20' または X'24' を戻します。これは IMS Connect が切断することを示します。
5. クライアント・アプリケーションは切断要求を発行する。

## ソケット数の管理

IMS Connect では、一度に 50 から 65,535 個のソケットをサポートします。その範囲内で、ソケットの最大許容数と、ソケット数が最大に近づいたときに IMS Connect が警告を出す時点の両方を定義することができます。

IMS Connect が全 TCP/IP ポートで同時にサポートするソケットの合計数は、IMS Connect 構成メンバーの TCP/IP 構成ステートメントの MAXSOC パラメーターで設定されます。ソケット数が最大に達すると、IMS Connect は新しい接続をすべて拒否し、メッセージ HWSS0771W を発行します。接続数が MAXSOC 値未満になると、IMS Connect は接続の受け入れを再開します。

IMS Connect は、listen 用に TCP/IP ポートごとに 1 つのソケットを使用するため、IMS Connect がサポートする物理接続の最大数は、MAXSOC 値から TCP/IP ポート数を引いた数になります。例えば、MAXSOC=80 を指定し、かつ、5 個の TCP/IP ポートを備えている場合、75 の物理接続を作成することが可能です。

デフォルトでは、MAXSOC パラメーターはソケットの最大数を 50 に設定します。

IMS Connect は、ソケット数が最大許容数に近づくと、WARNSOC パラメーターで設定された警告しきい値で警告を発行し、それ以降ソケット数が増え続ける場合は、WARNINC パラメーターで設定された増分ごとに警告を発行します。WARNSOC パラメーターおよび WARNINC パラメーターは、どちらも IMS Connect HWSCFGxx PROCLIB メンバーの TCPIP 構成ステートメントで指定します。

警告しきい値および警告増分の現行値、ならびに IMS Connect の特定のインスタンスの現在開いているソケットの数を表示するには、以下のいずれかのコマンドを発行します。

- IMS Connect WTOR コマンド **VIEWHWS**
- IMS Connect z/OS のコマンド **QUERY MEMBER**
- IMS Connect タイプ 2 コマンド **QUERY IMSCON TYPE (CONFIG)**

以下のいずれかのコマンドを発行することにより、特定のポート上の現在のソケット数も表示できます。

- IMS Connect WTOR コマンド **VIEWPORT**
- IMS Connect z/OS のコマンド **QUERY PORT**
- IMS Connect タイプ 2 コマンド **QUERY IMSCON TYPE (PORT)**

現行のソケット数を表示する際に、アクティブなクライアント 接続がポートに存在しない場合でも、IMS Connect ポートは常に 1 つの開いているソケット、すなわちポートの listen 用ソケットを表示します。

IMS Connect SSL ポートでは、少なくとも 2 つのソケット、すなわちポートの listen 用ソケットと、SSL 関連ファイル記述子を表すソケットが表示されます。IMS Connect SSL ソケットが最初の接続を受信すると、追加の SSL 関連ファイル記述子が 1 つ作成されるため、IMS Connect SSL ソケットはその最初のクライアント接続について 4 つのソケット (ポートの listen 用ソケット、2 つのファイル記述子、およびクライアント接続ソケット) を表示します。

## 関連概念

[IMS Connect の定義と調整 \(システム定義\)](#)

## 関連資料

[IMS PROCLIB データ・セットの HWSCFGxx メンバー \(システム定義\)](#)

## z/OS UNIX システム・サービスで設定されるソケット数の制限

IMS Connect でサポートされるソケットの数は、z/OS UNIX システム・サービスのパラメーター MAXFILEPROC による影響を受けます。このパラメーターは、IMS Connect が任意のユーザー ID に対して各ポートで開くことができるソケットの数に制限を課します。

例えば、UNIX システム・サービスの MAXFILEPROC パラメーターの値が 100 の場合、IMS Connect は、1 つのポートで開くことができるソケットは 100 個以下です。これに対し、IMS Connect MAXSOC パラメーターは、使用されるポートの数にかかわらず、IMS Connect がサポートするソケットの合計数に制限を課します。

**重要:** UNIX システム・サービスのパラメーター MAXFILEPROC の値は、IMS Connect パラメーター MAXSOC の値以上でなければなりません。そうでないと、IMS Connect はどのポートも開くことができません。

UNIX システム・サービスのスーパーユーザー特権を IMS Connect に付与することで、MAXFILEPROC パラメーターの値が適切に設定されるようにすることができます。この特権を付与すると、IMS Connect は MAXFILEPROC パラメーターの値を自動的に変更できます。UNIX システム・サービスのスーパーユーザー特権を IMS Connect に付与するには、RACF コマンド **ALTERUSER** を使用して、UID が 0 の OMVS セグメントを IMS Connect 開始済みタスクのユーザー ID に割り当てます。代わりに、UNIX システム・サービスの管理者が、z/OS SYS1.PARMLIB データ・セットの BPXPRMxx メンバー内の MAXFILEPROC パラメーターの値を調整することもできます。

IMS Connect にスーパーユーザー特権がなく、MAXSOC パラメーター値が MAXFILEPROC パラメーター値より大きい場合、IMS Connect はメッセージ HWSP1415E TCP/IP SOCKET FUNCTION CALL FAILED; F=SETRLIMI, R=-1, E=139, M=SDOT を発行し、どのポートも開きません。

IMS Connect の MAXFILEPROC パラメーターの値は、UNIX コマンド **D OMVS,L,PID=** を発行することによって確認できます。ここで、PID は、IMS Connect のプロセス ID です。IMS Connect の PID は、UNIX コマンド **D OMVS,V** を発行すると判別できます。

IMS Connect ポート上のソケットの数が MAXFILEPROC パラメーター値に近づくと、UNIX システム・サービスはメッセージ BPXI040I を発行します。例えば、BPXI040I PROCESS LIMIT MAXFILEPROC HAS REACHED 85% OF ITS CURRENT 404 が出力されます。BPXI040I メッセージが UNIX システム・サービスによって表示されるのは、SYS1.PARMLIB(BPXPRMxx) データ・セット内で LIMMSG が SYSTEM または ALL に設定されているか、SETOMVS コマンドを使用した場合のみです。

MAXFILEPROC パラメーター値に達すると、IMS Connect は以下のメッセージを発行します。

- HWSP1415E TCP/IP SOCKET FUNCTION CALL FAILED; F=ACCEPT4, R=-1, E=124, M=SDCO
- HWSS0771W LISTENING ON PORT=*portid* FAILED; R=*rc*, S=*sc*, M=*mc*

MAXSOC パラメーター制限に達した場合、IMS Connect は HWSS0771W メッセージのみを発行します。MAXFILEPROC パラメーター制限に達した場合、IMS Connect は、HWSP1415E メッセージと HWSS0771W メッセージの両方を実行します。

UNIX システム・サービスのスーパーユーザーおよび UNIX システム・サービスのパラメーター MAXFILEPROC について詳しくは、「z/OS UNIX システム・サービスの計画」を参照してください。

## IMS 間通信の送信ソケットの予約

RMTIMSCON 構成ステートメントで RESVSOC パラメーターを指定することにより、IMS 間 TCP/IP 接続で使用する IMS Connect 送信ソケットを予約できます。

RESVSOC パラメーターに指定されているソケット数は、TCPIP 構成ステートメントの MAXSOC パラメーターに指定されている、IMS Connect の 1 つのインスタンスが 1 度にオープンできるソケットの総数から減算されます。したがって、すべての構成ステートメントのすべての RESVSOC パラメーターの合計値は、MAXSOC パラメーターに指定された値を超えることはできません。

RESVSOC パラメーターは、送信ソケットのみを予約します。受信ソケットは予約できません。

接続が、MSC に使用される IMS 間接続の場合、TCP/IP 接続を使用する各 MSC 論理リンクには、データ送信用とデータ受信用の 2 つのソケットが必要です。RESVSOC パラメーターは、送信ソケットのみを予約します。したがって、MSC に使用される、IMS Connect の別のインスタンスへの接続では、RESVSOC パラメーターに指定されているソケット数の 2 倍のソケットが使用されます。

接続が、OTMA に使用される IMS 間の TCP/IP 接続の場合、各接続は単一の送信ソケットのみを使用します。したがって、RESVSOC パラメーターによって予約されているソケット数は、各接続に使用されるソケット数を表しています。

### 関連概念

[IMS Connect の定義と調整 \(システム定義\)](#)

### 関連タスク

[IMS 間の TCP/IP 接続 \(システム定義\)](#)

### 関連資料

[RMTIMSCON ステートメント \(システム定義\)](#)

[RMTICICS ステートメント \(システム定義\)](#)

## ソケット数の警告

IMS Connect は、ソケット数がデフォルトの警告しきい値、すなわち、MAXSOC パラメーターで設定される最大許容ソケット数の 80 パーセントに到達すると、警告メッセージ HWSS0772W を発行します。

## ソケット数の警告

ソケット数が 80 パーセントを過ぎて増え続ける場合、IMS Connect はデフォルトの増分で (ソケット数が 5 パーセント増えるごとに) 新しく HWSS0772W メッセージを発行します。ソケット数が最大許容数に達すると、IMS Connect は新しい接続をすべて拒否し、メッセージ HWSS0771W を発行します。

警告しきい値は、IMS Connect HWSCFGxx PROCLIB メンバーの TCPIP ステートメントで WARNSOC パラメーターを使用して設定できます。警告増分は、TCPIP ステートメントで WARNINC パラメーターを使用して設定できます。

ソケットの数がわずかに変動するだけでコンソールがメッセージであふれるのを防ぐために、IMS Connect は、警告しきい値で 1 回のみ、そしてそれ以降は警告増分ごとに 1 回のみ、警告メカニズムがリセットされるまで、HWSS0772W メッセージを発行します。

IMS は、WARNINC パラメーター値の 2 倍または 5 パーセント (結果のリセットしきい値が低くなる方) を MAXSOC パラメーターの値から引いて、リセットしきい値を決定します。リセットしきい値に達すると、IMS Connect は警告メカニズムをリセットして、メッセージ HWSS0773I を発行します。

デフォルトでは、IMS Connect は、ソケット数が MAXSOC パラメーター値の 70 パーセントまで減少すると、警告メカニズムをリセットします (70 パーセントは MAXSOC パラメーターのデフォルト値より 10 パーセント少なく、10 パーセントは WARNINC パラメーターのデフォルト値の 2 倍に相当します)。

### 関連資料

[IMS PROCLIB データ・セットの HWSCFGxx メンバー \(システム定義\)](#)

### 関連情報

[HWSS0771W \(メッセージおよびコード\)](#)

[HWSS0772W \(メッセージおよびコード\)](#)

[HWSS0773I \(メッセージおよびコード\)](#)

## 重複クライアント ID の解決

接続の失敗またはクライアントの予期しない終了が原因で、IMS Connect と IMS Connect クライアントとの間で障害が発生し、そのクライアントが同じクライアント ID を使用して再接続を試行する前に、IMS Connect がその障害を検出できなかった場合、IMS Connect は、そのクライアント ID が失敗した元の接続に関連付けられた ID と重複するという理由で、クライアントの再接続を禁止します。

障害発生後にクライアント ID が重複する状態を解決するために、着信接続によって元のクライアント ID を取り消すか、クライアントの元の接続先であるソケット接続の IMS Connect メッセージ・タイマーを取り消すことができます。これらの 2 つのオプションでは、クライアント ID を取り消すことの方がより簡単です。

まず第一に、クライアント ID が重複した状態になる可能性を低くするために、所定のポートでのすべてのソケット接続に小さい KeepAlive インターバルを指定できます。小さい KeepAlive インターバルを指定すると、IMS Connect がクライアントによる再接続の試行前の初期段階でクライアント接続の失敗を検出してクリーンアップできる可能性が高くなります。ただし、小さい KeepAlive 値を指定すると、ネットワーク・トラフィックが増大する場合があります。

また、IMS TM Resource Adapter クライアントは、IMS Connect に、着信接続に対する固有のクライアント ID を生成させることができます。これにより、IMS TM Resource Adapter からの着信接続が、既存の接続で使用されている ID を重複して指定するのを回避します。

## クライアント ID が重複する接続の取り消し

IMS Connect への着信接続で、別の既存接続で使用されているものと重複するクライアント ID を指定している場合、IMS Connect が新しい接続を受け入れることができるように、その既存接続を取り消すことが

できます。クライアント ID が取り消されると、IMS Connect は接続の状態に関係なく、その既存接続を廃棄し、HWSS0743I DUPLICATE CLIENT ID TERMINATED というメッセージを出します。

クライアント ID が重複する接続の取り消し処理は、コミット後送信 (CM0) または送信後コミット (CM1) トランザクションで持続ソケットまたはトランザクション・ソケットを使用するクライアントに対してサポートされます。

重複クライアント ID が取り消された場合、以前のセッション (存在する場合) がクリーンアップされ、新しい要求が完了されます。次に、ソケット・タイプが持続ソケットの場合にはセッションが維持され、ソケットがトランザクション・ソケットの場合にはセッションが終了します。

ユーザー作成の IMS Connect クライアント・アプリケーションの場合、以下の方法で重複するクライアント ID 接続を取り消すことができます。

- IMS Connect クライアント・アプリケーションをコーディングして、クライアントが IMS Connect で新しい接続を確立するときに、HWSIMSCB マクロによってマップされる IRM メッセージ・ヘッダーのユーザー・セクションの IRM\_F3 フィールドに X'80' (IRM\_F3\_CANCID) を指定します。
- 後続の出口ルーチンをコーディングして、HWSOMPFX マクロによってマップされる OTMA メッセージ・ヘッダーのユーザー・データ・セクションの OMUSR\_FLAG1 フィールドに X'20' (OMUSR\_CANCID) を指定します。

IMS Connect が既にエラーを検出し、元の接続を終了していた場合は、IMS Connect は重複するクライアント ID の取り消し指定を無視します。

## メッセージ・タイマーの取り消しによる重複 クライアント ID の解決

失敗した元のクライアント接続の IRM タイマーを取り消すことで、クライアント ID が重複した状態を解決できます。ただし、タイマーの取り消しに必要なステップには、ACK、SEND、および READS の発行、接続と切断などが含まれるため、クライアント ID が重複した状態を解決する方法として、重複したクライアント ID を取り消すことをお勧めします。

## IMS TM Resource Adapter 用に自動生成された ID

IMS TM Resource Adapter のユーザーに関して、新しい接続要求で IMS TM Resource Adapter が提供したユーザー ID が、既存の接続で既に使用されているユーザー ID と重複する場合には、IMS Connect は、固有のユーザー ID を生成することができます。

IMS Connect がユーザー ID を生成した後は、IMS TM Resource Adapter に返されるすべての応答で、生成されたユーザー ID が使用されます。

IMS TM Resource Adapter クライアントが、IMS Connect に対してユーザー ID を生成するよう要求するときは、OTMA メッセージ接頭語内で OMUSR\_FLAG2 フィールドに X'40' (OMUSR\_F2\_CIDREQ) を設定します。IMS Connect が、生成されたユーザー ID が含まれる応答を返すときは、OMUSR\_FLAG2 に X'20' (OMUSR\_F2\_CIDGEN) が設定されます。

### 関連タスク

[327 ページの『メッセージ・タイマーの取り消し』](#)

ユーザー作成の IMS Connect クライアント・アプリケーションは、データ・ストアからの出力を待っているとき、アクティブなメッセージ・タイマーを取り消すことができます。

### 関連資料

[229 ページの『HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口ルーチンの IRM のユーザー部分のフォーマット』](#)

HWSSMPL0、HWSSMPL1、またはユーザー作成メッセージ出口によってサポートされるユーザー作成クライアント・アプリケーションでは、IMS Connect クライアント入力メッセージ内の IMS 要求メッセージ (IRM) ヘッダーの 4 バイトの長さフィールドと 28 バイトの固定部分に続けて、IRM にユーザー定義セクションを組み込むことができます。

[285 ページの『IMS Connect で使用される OTMA ユーザー・データ・フィールド』](#)



OTMA ヘッダー内のユーザー・データ・フィールドのフォーマットは、HWSOMPFX マクロ内の HWSOMUSR DSECT によって定義されており、すべての IMS Connect メッセージに共通です。

## IMS Connect による z/OS TCP/IP KeepAlive インターバルのオーバーライド

IMSS Connect は、TCP/IP ソケット接続のために z/OS で定義されているデフォルトの KeepAlive インターバルをオーバーライドできます。

KeepAlive 機能は、TCP/IP プロトコルが提供する機能です。これにより、指定したインターバルで非アクティブであるソケットの KeepAlive パケットを送信することで、所定のソケット・エラー状態を検出できます。例えば、KeepAlive 機能では、IMS Connect への通知なしに、クライアントが突然切断されたために有効ではなくなったソケットを検出できます。

デフォルトでは、IMS Connect は TCP/IP ソケット用に z/OS 層に設定されている指定を受け入れます。ただし、z/OS で設定されている KeepAlive インターバルの値が大きい場合、ネットワーク・トラフィックを軽減しようとするインストールの場合と同様に、IMS Connect による無効なソケットの検出が遅れる場合があります。

IMS Connect ソケットに関するエラー状態を初期段階で検出するには、IMS Connect ポートの KeepAlive インターバルに小さい値を指定します。IMS Connect に対して定義されるポートごとに、異なる KeepAlive インターバルを指定できます。各ポートごとに指定される KeepAlive インターバルは、そのポートを使用するすべてのソケットに適用されます。

一般的には、IMS Connect ソケットで TCP/IP スタック用に z/OS で定義されている値とは異なる KeepAlive インターバルが必要であるかどうかは、ネットワーク・マネージャーが決定します。IMS Connect 用の KeepAlive インターバルが決定されたら、IMS システム・プログラマーがその KeepAlive インターバルを使用して IMS Connect HWSCFGxx PROCLIB メンバーの PORT または DRDAPORT パラメーターを更新します。

IMS TM にアクセスする IMS Connect クライアントの場合は、HWSCFG PROCLIB メンバーの TCPIP 構成ステートメントで PORT キーワードに KEEPAV パラメーターを使用して、KeepAlive インターバルを指定できます。

IMS DB にアクセスする IMS Connect クライアントの場合は、HWSCFG PROCLIB メンバーの ODACCESS 構成ステートメントで DRDAPORT キーワードに KEEPAV パラメーターを使用して、KeepAlive インターバルを指定できます。

KEEPAV パラメーターの有効な値の範囲は、TCP/IP プロトコルによって定義されており、1 から 2147460 秒までです。KEEPAV にゼロを指定すると、TCP/IP スタック用に z/OS 層で設定されている KeepAlive インターバルが受け入れられます。デフォルトは、KEEPAV=0 です。

現行の KeepAlive インターバルを表示するには、以下のコマンドを発行します。

- VIEWHWS
- VIEWPORT
- QUERY MEMBER
- QUERY PORT

IMS Connect で定義される KeepAlive インターバルは、PORTID パラメーターを使用して定義されるポート、「LOCAL」ポート、および SSL ポートではサポートされません。これらのポートに KEEPAV パラメーターを指定すると、IMS Connect は初期設定時に異常終了します。

### 関連資料

[IMS PROCLIB データ・セットの HWSCFGxx メンバー \(システム定義\)](#)

## TCP/IP 障害

アクティブな IMS Connect ポートが、予期せず TCP/IP ネットワークまたは z/OS TCP/IP スタックで提供される TCP/IP サービスへの接続を失った場合、IMS Connect は、TCP/IP サービスが再び使用可能になったときにそれらのポートを自動的に再接続します。

TCP/IP サービスが使用可能になった場合にそれを検出するために、IMS Connect はすべてのアクティブ・ポートを listen し続けます。TCP/IP サービスでの通信が再開されると、IMS Connect はポートを再接続してメッセージ HWSS0780I を発行します。

ポートと TCP/IP サービスとの通信がコマンドを使用して終了していた場合、IMS Connect は、TCP/IP サービスが使用可能になっても、サービスとの通信を自動的に再開しません。

## IMS Connect のタイムアウト指定

通信プロセスのさまざまな段階で、さまざまなタイプの相互作用について、IMS Connect に対してタイムアウト間隔を指定できます。

IMS DB にアクセスする IMS Connect クライアントを使用しているのか、IMS TM にアクセスする IMS Connect クライアントを使用しているのか、あるいは OTMA 通信および IMS 間 TCP/IP 通信経由でリモート IMS システムに出力を送信する IMS アプリケーション・プログラムを使用しているのかによって、各タイムアウト間隔を指定する方法が異なります。

### 関連概念

[148 ページの『IMS Connect クライアント・サポート』](#)

IMS 用の TCP/IP サーバーおよびメッセージ・ルーターとして、IMS Connect は、IMS TM、IMS DB、および CSL Operations Manager (OM) へのアクセスを提供します。IMS Connect によって提供されるクライアント・サポートは、IMS Connect クライアントに必要なアクセス・タイプによって異なります。

[154 ページの『IMS Connect による IMS 間 TCP/IP 通信のサポート』](#)

IMS Connect は、TCP/IP ネットワークを介して相互に通信する IMS システムの TCP/IP 接続およびプロトコルを管理します。

## IMS DB クライアントのタイムアウト指定

IMS DB にアクセスする IMS Connect クライアントの場合は、ODACCESS 構成ステートメントの 2 つのパラメーターでタイムアウト値を指定できます。

ODBMOT パラメーターは、以下の両方について IMS Connect の待機時間を制御します。

- ODBM との接続時の応答メッセージ
- クライアント・アプリケーションとの接続時にソケット接続が確立された後の初期入力メッセージ

PORTMOT キーワードは、クライアントが入力の送信を停止した後、IMS Connect が既存の接続をオープン状態に保つ時間の長さを制御します。

### 関連資料

[ODACCESS ステートメント \(システム定義\)](#)

## IMS TM クライアントのタイムアウト指定

IMS TM にアクセスする IMS Connect クライアントの場合、TCP/IP 構成ステートメントおよび DATASTORE 構成ステートメント、ならびに入力メッセージの IMS 要求メッセージ (IRM) ヘッダーで、タイムアウト値を指定することができます。

TCPIP 構成ステートメントでは、以下の通信ステージで IMS Connect が待機する時間に対して、以下の制限を設定することができます。

- 接続が最初に確立された後にクライアントが入力を何も送信しない場合、IMS Connect が接続を開いたままの状態に保つ時間の長さ。TIMEOUT パラメーターは、この制限を設定します。
- 前のクライアント対話が完了した後に、IMS Connect が RECV 状態の接続を開いたままの状態に保つ時間の長さ。この制限は、TCPIP 構成ステートメントの IDLETO パラメーター、あるいは CREATE IMSCON コマンドによって設定することができます。
- IMS Connect がクライアントにタイムアウトを通知してソケット接続を RECV 状態に戻す前に、IMS Connect が IMS からの応答を待つ時間の長さ。この制限は、TIMEOUT パラメーターで設定することもでき、IRM でのタイムアウト間隔を指定する入力メッセージによってオーバーライドされます。

アイドル接続タイムアウト値 (IDLETO) は、IMS TM との通信に使用されるポートにのみ適用されます。IMS TM ポートは、TCPIP ステートメントの PORT または PORTID パラメーターによって定義されるか、CREATE IMSCON コマンドによって PORTTYPE(REG) として定義されます。IDLETO 値は、TCPIP 構成ステートメントで、あるいは IMS タイプ 2 CREATE IMSCON TYPE(PORT) コマンド、UPDATE IMSCON TYPE(PORT) コマンド、または UPDATE IMSCON TYPE(CONFIG) コマンドを使用して設定することができます。

DATASTORE 構成ステートメントでは、ACKTO キーワードは、OTMA が出力を T パイプ保留キューに転送する前に IMS Connect からの確認応答を待つ時間の長さを制御します。

TCPIP 構成ステートメントおよび DATASTORE 構成ステートメントで指定されるタイムアウト値は、IMS TM クライアントから受信するメッセージの IRM メッセージ・ヘッダーでも指定できます。

IRM は、クライアントが RESUME TPIPE 要求を出した後、出力を待つ時間の長さを指定するためにも使用できます。

## 関連資料

[DATASTORE ステートメント \(システム定義\)](#)

[TCPIP ステートメント \(システム定義\)](#)

[CREATE IMSCON TYPE\(PORT\) コマンド \(コマンド\)](#)

[UPDATE IMSCON コマンド \(コマンド\)](#)

## 入力メッセージのタイムアウトの指定

IMS Connect クライアントからの個々の入力メッセージは、いずれも IMS 要求メッセージ (IRM) ヘッダーの固定部分の IRM\_TIMER フィールドに異なるタイムアウト値を設定できます。

IMS が IMS Connect にデータを戻すまでの適切な待ち時間を IRM\_TIMER 値に設定します。

IRM\_TIMER の設定は、以下のリストで示すように実施されます。

1. IRM\_TIMER が X'00' に設定されている場合、以下のデフォルト値が使用されます。
  - すべての RESUME\_TPIPE に対するデフォルトは 2 秒です。
  - すべての RESUME\_TPIPE 非単一 ACK に対するデフォルトは .25 秒です。
  - その他のすべての場合では、IMS Connect TCPIP 構成ステートメント内の TIMEOUT パラメーターの値です。
2. 要求された場合のみ、X'FF' および X'01' から X'9E' までが使用されます。
3. X'E9' (文字 Z) NO\_WAIT は、IMS 出力を待機しないことを意味します。NO\_WAIT は、一部のクライアント SEND に対しては無効です。IMS Connect は、IMS からの出力を待機しないため、トランザクション・ソケット接続の場合、IMS Connect はソケットを切断します。また、持続ソケット接続の場合、IMS Connect は、ソケットを切断せずにクライアントからの次の入力を要求します。NO\_WAIT を使用すると、以下のように実施されます。
  - 以下の場合、2 秒間の遅延が生じます。
    - RESUME\_TPIPE 要求
    - 会話型トランザクション・コード
    - 会話型データ
    - 会話型トランザクションに関連付けられた ACK または NAK
    - 非会話型トランザクション・コード
  - 以下のそれぞれで .25 秒遅延が使用されます。
    - 非会話型トランザクション、コミット・モード 1、CONFIRM に関連付けられた ACK または NAK
    - 非同期出力オプション AUTO または NOAUTO の RESUME\_TPIPE に関連付けられた ACK または NAK
    - 非会話型トランザクション、コミット・モード 0、CONFIRM に関連付けられた ACK または NAK
  - NO\_WAIT は、以下の場合に使用できます。



- SENDONLY
- 非同期出力オプション SINGLE の RESUME\_TPIPE に関連付けられた ACK または NAK
- 非会話型トランザクション、コミット・モード 0、CONFIRM (IRM NOWAIT フラグ付き) に関連付けられた ACK または NA

X'E9' の誤用により、以下のいずれかの問題が発生することがあります。

1. ソケットが切断する。
2. トランザクション・ソケットのクライアントへの出力メッセージが失われる。
3. クライアントと IMS Connect との間、または IMS Connect と OTMA との間でハング状態が発生する。  
例えば、クライアントが IMS Connect からの出力を待機しながら READ 状態にあり、一方で、IMS Connect がクライアントからの入力待機しながら READ 状態にあり、さらに OTMA が確認応答を待機しながら READ 状態にあるという場合があります。
4. CM1、同期レベル =CONFIRM の場合の割り振り解除コミットまたは割り振り解除の打ち切り通知が失われる。
5. その他の予測不能な状態が発生する。

IMS が IMS Connect にデータを戻すまでの適切な待ち時間を決定するには、以下のガイドラインを検討してください。

- トランザクション・コードとデータ、またはデータのみクライアント SEND の場合、IRM\_TIMER 値を設定して、IMS Connect が IMS からの出力を待機する時間量を反映させてください。

**推奨事項:** IRM で NOWAIT フラグが設定されているメッセージを除いて、X'E9' のタイマー値を使用しないでください。

- 最後に受け取ったメッセージがそのトランザクションのクライアントへの最後の出力メッセージであることがクライアント・アプリケーションに分かる場合、ACK または NAK のクライアント送信で IRM\_TIMER を X'01' (.01 秒) に設定してください。IRM\_TIMER の X'01' は、非 RESUME TPIPE ACK メッセージに対して設定可能な最小値です。ただし、ACK メッセージが RESUME TPIPE 呼び出しからの出力に関連付けられている場合は、IRM\_TIMER 値を X'E9' (文字 Z) に設定しないでください。
- RESUME TPIPE 呼び出しのクライアント SEND の場合、タイマー値を以下のように設定できます。

#### **AUTO オプション**

専用出力装置の場合は X'FF'、非専用出力装置の場合は X'00' から X'9E' までの任意の値

#### **NOAUTO オプション**

X'FF'、X'E9' 以外の任意の値

#### **SINGLE オプションまたは WAIT 付き SINGLE オプション**

X'FF'、X'E9' 以外の任意の値

### **関連概念**

#### OTMA への IMS Connect の確認応答のタイムアウト間隔

OTMA が、IMS Connect からの確認応答を待機する時間を決定するタイムアウト間隔を指定することができます。また、タイムアウト T パイプ・キューを指定して、タイムアウト間隔が経過した後に、コミット後送信 (CM0) 出力を保持することができます。

### **タイマー間隔の指定**

タイマーの値は、いくつかの増分範囲で指定できます。

各範囲の値を選択するには、クライアントからの入力メッセージの IRM で、IRM\_TIMER フィールドに 16 進値を入力します。指定できる 16 進値、およびそれらが表す時間間隔は、以下のとおりです。

- 100 分の 1 秒の増分は、[322 ページの表 61](#) に示すように、X'01' から X'19' までの値で表します。
- 100 分の 5 秒の増分は、[322 ページの表 62](#) に示すように、X'1A' から X'27' までの値で表します。
- 1 秒の増分は、[323 ページの表 63](#) に示すように、X'28' から X'63' までの値で表します。
- 1 分の増分は、[325 ページの表 64](#) に示すように、X'63' から X'9E' までの値で表します。
- デフォルトのタイマー値、タイマーなし、および無期限の待機のオプションは、[327 ページの表 65](#) に示すように、それぞれ、X'00'、X'E9'、および X'FF' で指定します。

次の表は、100分の1秒の増分を単位とする IRM\_TIMER の値とそれに対応する時間を示したリストです。

表 61. 100 分の 1 秒単位の IRM\_TIMER 値

時間	16 進値
.01 秒	X'01'
.02 秒	X'02'
.03 秒	X'03'
.04 秒	X'04'
.05 秒	X'05'
.06 秒	X'06'
.07 秒	X'07'
.08 秒	X'08'
.09 秒	X'09'
.10 秒	X'0A'
.11 秒	X'0B'
.12 秒	X'0C'
.13 秒	X'0D'
.14 秒	X'0E'
.15 秒	X'0F'
.16 秒	X'10'
.17 秒	X'11'
.18 秒	X'12'
.19 秒	X'13'
.20 秒	X'14'
.21 秒	X'15'
.22 秒	X'16'
.23 秒	X'17'
.24 秒	X'18'
.25 秒	X'19'

次の表は、100分の5秒の増分を単位とする IRM\_TIMER の値とそれに対応する時間を示したリストです。

表 62. 100 分の 5 秒単位の IRM\_TIMER 値

時間	値
.30 秒	X'1A'
.35 秒	X'1B'
.40 秒	X'1C'
.45 秒	X'1D'
.50 秒	X'1E'

表 62. 100 分の 5 秒単位の *IRM\_TIMER* 値 (続き)

時間	値
.55 秒	X'1F'
.60 秒	X'20'
.65 秒	X'21'
.70 秒	X'22'
.75 秒	X'23'
.80 秒	X'24'
.85 秒	X'25'
.90 秒	X'26'
.95 秒	X'27'

次の表は、1 秒の増分を単位とする *IRM\_TIMER* の値とそれに対応する時間を示したリストです。

表 63. 秒単位の *IRM\_TIMER* 時間値

時間	値
1 秒	X'28'
2 秒	X'29'
3 秒	X'2A'
4 秒	X'2B'
5 秒	X'2C'
6 秒	X'2D'
7 秒	X'2E'
8 秒	X'2F'
9 秒	X'30'
10 秒	X'31'
11 秒	X'32'
12 秒	X'33'
13 秒	X'34'
14 秒	X'35'
15 秒	X'36'
16 秒	X'37'
17 秒	X'38'
18 秒	X'39'
19 秒	X'3A'
20 秒	X'3B'
21 秒	X'3C'
22 秒	X'3D'

表 63. 秒単位の *IRM\_TIMER* 時間値 (続き)

時間	値
23 秒	X'3E'
24 秒	X'3F'
25 秒	X'40'
26 秒	X'41'
27 秒	X'42'
28 秒	X'43'
29 秒	X'44'
30 秒	X'45'
31 秒	X'46'
32 秒	X'47'
33 秒	X'48'
34 秒	X'49'
35 秒	X'4A'
36 秒	X'4B'
37 秒	X'4C'
38 秒	X'4D'
39 秒	X'4E'
40 秒	X'4F'
41 秒	X'50'
42 秒	X'51'
43 秒	X'52'
44 秒	X'53'
45 秒	X'54'
46 秒	X'55'
47 秒	X'56'
48 秒	X'57'
49 秒	X'58'
50 秒	X'59'
51 秒	X'5A'
52 秒	X'5B'
53 秒	X'5C'
54 秒	X'5D'
55 秒	X'5E'
56 秒	X'5F'

表 63. 秒単位の *IRM\_TIMER* 時間値 (続き)

時間	値
57 秒	X'60'
58 秒	X'61'
59 秒	X'62'
60 秒	X'63'

次の表は、1 分の増分を単位とする *IRM\_TIMER* の値とそれに対応する時間を示したリストです。

表 64. 分単位の *IRM\_TIMER* 時間値

時間	値
1 分	X'63'
2 分	X'64'
3 分	X'65'
4 分	X'66'
5 分	X'67'
6 分	X'68'
7 分	X'69'
8 分	X'6A'
9 分	X'6B'
10 分	X'6C'
11 分	X'6D'
12 分	X'6E'
13 分	X'6F'
14 分	X'70'
15 分	X'71'
16 分	X'72'
17 分	X'73'
18 分	X'74'
19 分	X'75'
20 分	X'76'
21 分	X'77'
22 分	X'78'
23 分	X'79'
24 分	X'7A'
25 分	X'7B'
26 分	X'7C'
27 分	X'7D'

表 64. 分単位の IRM\_TIMER 時間値 (続き)

時間	値
28 分	X'7E'
29 分	X'7F'
30 分	X'80'
31 分	X'81'
32 分	X'82'
33 分	X'83'
34 分	X'84'
35 分	X'85'
36 分	X'86'
37 分	X'87'
38 分	X'88'
39 分	X'89'
40 分	X'8A'
41 分	X'8B'
42 分	X'8C'
43 分	X'8D'
44 分	X'8E'
45 分	X'8F'
46 分	X'90'
47 分	X'91'
48 分	X'92'
49 分	X'93'
50 分	X'94'
51 分	X'95'
52 分	X'96'
53 分	X'97'
54 分	X'98'
55 分	X'99'
56 分	X'9A'
57 分	X'9B'
58 分	X'9C'
59 分	X'9D'
60 分	X'9E'

次の表は、IRM\_TIMER フィールドで指定できる追加オプションと、それらを指定するために使用できる値のリストです。

表 65. 追加の IRM\_TIMER オプション

タイマー・オプション	値
デフォルト値を使用。RESUME TPIPE 呼び出しおよび関連する ACK メッセージの場合、デフォルトは 0.25 秒。その他すべての SEND の場合、デフォルトは構成ファイルの TIMEOUT 値。	X'00'
待機しない。	X'E9' C'Z'
無期限に待機する。この設定値は、非同期出力機能の AUTO オプションのサポートを目的としている。	X'FF'

## メッセージ・タイマーの取り消し

ユーザー作成の IMS Connect クライアント・アプリケーションは、データ・ストアからの出力を待っているとき、アクティブなメッセージ・タイマーを取り消すことができます。

## このタスクについて

タイマー取り消し機能は、長いタイムアウト間隔を指定している IMS Connect クライアントが、データ・ストアからの応答が送信されないときに失われるのを防止します。もしタイマー取り消し機能がなければ、ソケット接続を切断するために IMS Connect の STOPCLNT コマンドが発行される必要があります。タイマーの取り消し要求がサブミットされた場合、IMS Connect はクライアントに通知を出します。

タイマー取り消し機能は、ユーザー作成のメッセージ出口ルーチンと、サンプルのユーザー・メッセージ出口ルーチン HWSSMPL0 および HWSSMPL1 によってサポートされます。

クライアントが RESUME TPIPE 呼び出しを発行した後に CONN 状態で待機している場合、データ・ストアが IMS または STOPDS コマンドによってクローズされると、クライアントは RC=X'2C' (つまり、10 進数 44) の RSM メッセージを受け取ることに注意してください。

タイマーの取り消し要求は、IRM\_F4 フィールドの C で指定し、クライアントの単一インスタント、または同じクライアント ID を持つクライアントの 2 つのインスタンスのいずれかからサブミットすることができます。

単一クライアント・インスタンスからタイマーの取り消し要求をサブミットするには、以下のようになります。

## 手順

1. ACK の SEND を発行します。
2. ローカル・タイマーを設定します。
3. RESPONSE の READ を発行します。タイマーは、データの受信でなくポップを行います。
4. DISCONNECT を発行します。
5. CONNECT を発行します。
6. IRM に CANCEL TIMER を設定して SEND を発行します。
7. CANCEL TIMER の READ を発行します。ユーザー・メッセージ出口は、RSM で戻りコード 8 と理由コード X'3B' (つまり、10 進数 59) を発行します。
8. DISCONNECT を発行します。

## タスクの結果

同じクライアント ID を持つ同じクライアントの 2 つのインスタンスから、タイマーの取り消し要求をサブミットするには、以下のようにします。

1. 最初のクライアントから
  - a. ACK の SEND を発行します (例えば、ICON01 のクライアント ID を使用)。
  - b. RESPONSE の READ を発行します。
2. 2 番目のクライアント・インスタンスから
  - a. CONNECT を発行します。
  - b. IRM に CANCEL TIMER を設定して SEND を発行します (ICON01 のクライアント ID を使用)。
  - c. CANCEL TIMER の READ を発行します。ユーザー・メッセージ出口は、RSM で戻りコード 8 と理由コード X'3B' (つまり、10 進数 59) を発行します。
  - d. DISCONNECT を発行します。
3. 最初のクライアントは、戻りコード X'2C' (つまり、10 進数 44) と理由コードがタイマー値に設定された RSM を受け取ります。このインスタンスは、接続されたままで残り、RECV 状態です。
  - a. DISCONNECT を発行するか、処理を続行します。

## OTMA への IMS Connect の確認応答のタイムアウト間隔

OTMA が、IMS Connect からの確認応答を待機する時間を決定するタイムアウト間隔を指定することができます。また、タイムアウト T パイプ・キューを指定して、タイムアウト間隔が経過した後に、コミット後送信 (CM0) 出力を保持することができます。

DATASTORE 構成ステートメント内の ACKTO パラメーターを使用して、タイムアウト間隔を指定することができます。ACKTO パラメーターで指定されるタイムアウト値は、以下のメッセージ・タイプで OTMA に送信される確認応答に適用されます。

- リモート IMS システムに送信されるトランザクション・メッセージ
- CM0 出力
- 送信後コミット (CM1) 出力

リモート IMS システムに送信されるトランザクション・メッセージの場合、ACK タイムアウト間隔が有効期限切れになると、OTMA はトランザクション・メッセージをタイムアウト・キューに転送します。トランザクション・メッセージがタイムアウトになった後に OTMA がローカル IMS Connect から ACK メッセージを受け取った場合、OTMA はローカル IMS Connect に対して X'2B' 戻りコードとともに NAK を発行します。

CM0 出力については、タイムアウト間隔が経過したときに、OTMA によって T パイプ・キューからの出力が除去され、以下のいずれかに出力が転送されます。

- 指定された転送 T パイプ・キュー
- 指定されたタイムアウト T パイプ・キュー
- デフォルトの OTMA タイムアウト T パイプ・キュー DFSS\$TOQ

HWS 構成ステートメントおよび DATASTORE 構成ステートメントの CM0ATOQ パラメーターで、リモート IMS システムに送信される CM0 出力用およびトランザクション・メッセージ用のタイムアウト T パイプ・キューの名前を指定することができます。

HWS 構成ステートメントで、CM0ATOQ パラメーターは、IMS Connect のインスタンスに定義されたすべてのデータ・ストア接続に対して、デフォルトのタイムアウト T パイプ・キューを定義します。DATASTORE ステートメントで、CM0ATOQ パラメーターは、定義されたデータ・ストア接続によってのみ使用されるようにタイムアウト T パイプ・キューを定義します。DATASTORE ステートメントの CM0ATOQ に対する指定は、HWS 構成ステートメントで行われた CM0ATOQ に対する指定をオーバーライドします。

タイムアウト間隔が経過したら、OTMA は CM1 出力を破棄するため、CM1 出力はタイムアウトはタイムアウト・キューに転送されません。



タイムアウトが発生して、IMS アプリケーションが IOPCB に応答しないか、別のトランザクションへのメッセージ通信を完了しない場合、トランザクションの応答モードに関係なく、OTMA は CMO と CM1 の両方の入力メッセージに対して DFS2082 メッセージを発行します。

### 関連概念

#### 入力メッセージのタイムアウトの指定

IMS Connect クライアントからの個々の入力メッセージは、いずれも IMS 要求メッセージ (IRM) ヘッダーの固定部分の IRM\_TIMER フィールドに異なるタイムアウト値を設定できます。

## IMS 間接続のタイムアウト指定

OTMA 経由でトランザクションをリモート IMS システムに送信する IMS アプリケーション・プログラムの場合、RMTIMSCON 構成ステートメントおよび DATASTORE 構成ステートメントでタイムアウト値を指定することができます。

RMTIMSCON 構成ステートメントの IDLETO キーワードは、IMS Connect が、持続ソケット接続について、IMS から追加メッセージを受け取らない場合にその接続をオープン状態に保つ時間の長さを制御します。

DATASTORE 構成ステートメントの ACKTO キーワードは、OTMA が出力をタイムアウト T パイプ・キューに転送してエラー・メッセージを発行する前に、IMS Connect からの確認応答を待つ時間の長さを制御します。

確認応答のタイムアウト間隔は、OTMA で設定することもできます。これは、**/START TMBER** コマンドの TIMEOUT キーワード、または DFSYDTx PROCLIB メンバー内の OTMA クライアント記述子の T/O パラメーターで指定します。

### 関連資料

RMTIMSCON ステートメント (システム定義)

OTMA クライアント記述子の構文およびパラメーター (システム定義)

/START TMEM コマンド (コマンド)

## IMS Connect のトランザクション有効期限サポート

IMS Connect は、IMS トランザクションがサブミットされるソケット接続のタイムアウト値に一致するように、そのトランザクションの有効期限を調整することができます。

IMS で有効期限を指定すると、その有効期限を超える前に IMS がトランザクションを処理しない場合、トランザクションは有効期限が切れて廃棄される可能性があります。

トランザクションの有効期限は、IMS のトランザクション定義で設定されるものであり、OTMA に固有のものではありません。IMS では、以下の方法でトランザクションの有効期限を設定または変更できます。

- トランザクションが IMS システムの定義時に定義される場合は、TRANSACT ステージ 1 システム定義マクロに **EXPRTIME** パラメーターを指定します。
- トランザクションが動的リソース定義を使用して実行時に作成される場合は、**CREATE TRAN** タイプ 2 コマンドで **EXPRTIME** キーワードを指定します。
- 宛先作成出口ルーチン (DFSINSX0) によって作成されたトランザクションの場合は、出口ルーチンが有効期限を設定できます。
- 既存のトランザクションの場合は、**UPDATE TRAN SET(EXPRTIME)** タイプ 2 コマンドで **EXPRTIME** キーワードを指定します。動的リソース定義を使用可能にする必要はありません。

IMS Connect で設定したトランザクション有効期限は、IMS のトランザクション定義で指定したいかなるトランザクション有効期限もオーバーライドします。

以下の IMS Connect ユーザー・メッセージ出口では、トランザクション有効期限の設定をサポートします。

- HWSSMPL0
- HWSSMPL1
- HWSSOAP1

HWSSMPL0、HWSSMPL1、またはユーザー作成出口ルーチンを使用する IMS Connect クライアントは、IRM 接頭部のユーザー・セクションで X'01' (IRM\_F1\_TRANEXP) を指定してトランザクションの有効期限を設定するように IMS Connect に指示することができます。IMS Enterprise Suite SOAP Gateway クライアントは、HWSSOAP1 ユーザー・メッセージ出口ルーチンを変更して IRM\_F1\_TRANEXP フィールドを設定できます。

IMS Connect は、有効期限をソケット接続のタイムアウト値に一致させるために、TCPIP 構成ステートメントの TIMEOUT パラメーターの値、または入力メッセージの IRM 接頭部の固定セクションの IRM\_TIMER フィールドの値を読み取って、クライアントのソケット接続がタイムアウトになる時間を計算します。次に、IMS Connect はその時間をストア・クロック (STCK) 形式で OTMA メッセージ接頭語のユーザー・データ・セクションに入れて、OTMA メッセージ接頭語の状態データ・セクションのフィールド OMHDRXP1 および OMHDRSXP を設定することにより、新しい有効期限を OTMA に通知します。

IMS Connect でサブミットされるトランザクションについてトランザクション有効期限が使用可能である場合、OTMA はトランザクションの処理をモニターします。そして、トランザクション有効期限が過ぎると、OTMA はそのトランザクションを廃棄して IMS Connect にメッセージを返します。

トランザクションがいつタイムアウトになるかに応じて、OTMA は IMS Connect への NAK 応答またはメッセージ DFS3688I のいずれかを返す可能性があります。

OTMA がいつ有効期限切れトランザクションの確認を行うかについて詳しくは、[827 ページの『OTMA のトランザクション有効期限サポート』](#)を参照してください。

IMS Connect がトランザクション有効期限を設定しなくても、トランザクションが最初に IMS に定義されたときに EXPRTIME パラメーターが指定されていた場合、トランザクションの有効期限が切れる可能性があります。

#### 関連概念

[827 ページの『OTMA のトランザクション有効期限サポート』](#)

トランザクションに有効期限を指定して、クライアントが使用できなくなったトランザクションを IMS に処理させないようにすることで、処理コストを削減できます。

#### 関連資料

[IMS PROCLIB データ・セットの HWSCFGxx メンバー \(システム定義\)](#)

[TRANSACT マクロ \(システム定義\)](#)

## IMS Connect でのトランザクション有効期限の設定

IMS Connect に指示して、IMS にサブミットされるトランザクションの有効期限を、そのトランザクションがサブミットされるソケット接続のタイムアウト値に一致するように設定させることができます。

### 始める前に

IMS Connect がトランザクション有効期限を設定するためには、まず、TCPIP 構成ステートメントの TIMEOUT パラメーターまたは IRM 接頭部の固定セクションの IRM\_TIMER フィールドで、ソケット接続のタイムアウト値を指定する必要があります。

### このタスクについて

以下のいずれかの方式を使用してトランザクション有効期限を設定するように、IMS Connect に指示することができます。

### 手順

- 入力メッセージの IRM 接頭部にあるユーザー・セクションの IRM\_F1 フィールドで、X'01' (IRM\_F1\_TRANEXP) を指定します。  
IRM\_F1 が X'01' に設定されると、ユーザー・メッセージ出口は OTMA ヘッダーの状態データ・セクションで OMHDRIST フィールドを X'01' (OMHDRXP1) に設定します。
- 以下のいずれかのユーザー・メッセージ出口ルーチンを変更して、IRM\_F1 フィールドを X'01' (IRM\_F1\_TRANEXP) に設定します。

- HWSSMPL0
- HWSSMPL1
- HWSSOAP1

IRM\_F1 が X'01' に設定されると、ユーザー・メッセージ出口は OTMA ヘッダーの状態データ・セクションで OMHDRIST フィールドを X'01' (OMHDRXP1) に設定します。

#### 関連資料

223 ページの『IMS Connect クライアント・メッセージの IRM 構造』

IMS Connect は、受信したすべてのクライアント・メッセージが合計 4 バイトの長さフィールドで始まり、その後に IMS 要求メッセージ (IRM) ヘッダー、メッセージ・データ・セグメントの順に続くものと想定します。

267 ページの『IMS Connect で使用される OTMA 状態データ・フィールド』

このトピックの表では、OTMA 状態データ・ヘッダーのフィールドおよびそれらのフィールドの順序を説明します。

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

## OTMA T パイプ保留キューでの出力のリトリブ

OTMA は、送達のためにキューに入れる必要のある出力を、T パイプ保留キューを使用して IMS Connect に送信します。IMS Connect クライアントは、T パイプ保留キューから出力をリトリブするために、出力メッセージがキューに入れられている T パイプの名前を指定する RESUME TPIPE 要求を発行します。

OTMA が T パイプ保留キューを介して送信する出力メッセージのタイプには、次のものがあります。

- IMS アプリケーション・プログラムが DL/I ICAL 呼び出しを発行して送信する同期コールアウト・メッセージ。
- IMS アプリケーション・プログラムが代替 PCB (ALTPCB) への DL/I ISRT 呼び出しを発行して送信する出力 (非同期コールアウト・メッセージなど)。
- 受信側 OTMA クライアントが NAK を返した CM0 IOPCB 出力メッセージ。
- CM0 SendOnly 入力に対する応答メッセージ。

送信と受信の対話を使用する単一のクライアントによって送信される CM0 トランザクションの場合、T パイプの名前は通常、クライアント ID です。CM0 トランザクションからの出力は、次の方法でリトリブすることができます。

- T パイプ名に一致するクライアント ID を持つ元のクライアントによって。
- 元のクライアントが終了していた場合、T パイプ名に一致する同じクライアント ID を使用する別のクライアントによって。
- RESUME TPIPE 要求で T パイプ名を代替クライアント ID として指定する別のクライアントによって。

同期コールアウト・メッセージの場合、T パイプ名は通常 IMS 内の OTMA 宛先記述子で定義されます。非同期コールアウト・メッセージの場合、T パイプ名は、OTMA 宛先記述子または OTMA 経路指定出口ルーチンのいずれかで定義できます。どちらのタイプのコールアウト・メッセージの場合も、T パイプ名は通常、コールアウト・メッセージをリトリブする RESUME TPIPE 要求で代替クライアント ID として指定されます。代替クライアント ID は、OTMA 宛先記述子または OTMA 経路指定出口ルーチンで指定された T パイプ名と一致している必要があります。

また、T パイプが並列のアクティブ RESUME TPIPE 要求をサポートする場合も、T パイプ名は代替クライアント ID として指定されます。

**注:** 同期コールアウト要求メッセージは、非同期出力とほぼ同じ方法で OTMA および IMS Connect によって処理されます。すなわち、同期コールアウト要求メッセージは RESUME TPIPE 呼び出しの発行によってリトリブされます。非同期出力の場合と同じ規則および指針の多くが、同期コールアウト要求メッセージのリトリブに適用されます。ただし、同期コールアウト・メッセージの場合、クライアントからの RESUME TPIPE 呼び出しが、同期コールアウト要求を開始して処理したものと異なる IMS 共有キュー・メンバーに接続されると、クライアントはメッセージを受信しません。同期コールアウト要求は、T パイプ保留キューに入れられるため、その T パイプを所有する IMS によってのみ認識されます。スーパーメンバ

一機能は、複数の IMS Connect クライアントが同じ IMS 共用キュー・メンバーに接続されているときのみ、同期コールアウト要求に適用されます。

IMS Connect は、以下の方法のいずれかによって、非同期出力があることを CM0 出力応答メッセージでクライアントに知らせます。

- フラグ CSM\_AMSG を、CSM (完了状況メッセージ) の CSM\_FLG1 フィールドに入れて戻す。
- フラグ RSM\_AMSG を、RSM (要求状況メッセージ) の RSM\_FLG1 フィールドに入れて戻す。

OTMA T パイプ保留キューから出力をリトリートするために IMS Connect を使用しない場合は、クライアント・アプリケーションは CSM や RSM を分析する必要はありません。IMS Connect は、クライアント・アプリケーションが非同期出力を要求するかどうかにかかわらず、非同期出力があることを知らせます。

RESUME TPIPE 呼び出しを使用して、クライアントからの非同期出力をリトリートしてください。持続ソケットとトランザクション・ソケットの両方で非同期出力を取得できます。

**制約事項:** IMS TM Resource Adapter は、非同期オプション SINGLE のみをサポートします。

## RESUME TPIPE/受信プロトコル

IMS Connect クライアントは RESUME TPIPE プロトコルを使用して、IMS の T パイプ保留キューからコミット後送信 (CM0) 出力または同期コールアウト要求をリトリートします。

同期コールアウト要求は、IMS 従属領域で実行されている IMS アプリケーション・プログラムから発行されます。コールアウト要求は、IMS インストールの外部のプロバイダーからのデータまたはサービスに対するものです。コールアウト・シナリオでは、IMS アプリケーションがクライアントで、外部プロバイダーがサーバーです。

IMS からの非同期出力には、IMS トランザクションからの応答メッセージと非同期コールアウト・メッセージが含まれている可能性があります。

ユーザー作成の IMS Connect クライアント・アプリケーション・プログラムでは、RESUME TPIPE 呼び出しは、IRM の接頭部に以下のフィールド値を指定してコーディングします。

### IRM\_F4

R 文字値 (IRM\_F4\_RESUMET)

### IRM\_F5

RESUME TPIPE 呼び出しのリトリート・オプションを指定する 16 進値。このリトリート・オプションでは、単一の RESUME TPIPE 呼び出しでリトリートできるメッセージ数、およびメッセージが T パイプ保留キューに残っていないときに IMS Connect クライアントが追加メッセージを待機する時間を指定します。

### IRM\_F6

値 X'80' (IRM\_F6\_NWSE) は、RESUME TPIPE 呼び出しが、\*NETSID\* および \*NETUID\* の分散ネットワーク・セキュリティ情報セグメントを含む同期または非同期のコールアウト要求をリトリートできることを指定します。

### IRM\_TIMER

RESUME TPIPE 呼び出しのタイムアウト値。IRM\_TIMER の値は、T パイプ保留キューに出力が存在しないときに、クライアントが新しいメッセージを待機する時間を指定します。

同期コールアウト・メッセージをリトリートする場合、上記のフィールドに加えて、以下のフィールドを指定して、IRM アーキテクチャー・レベル (IRM\_ARCH3) を設定し、IMS Connect クライアントが同期コールアウト・メッセージのみをリトリートするか、非同期出力も一緒にリトリートするかを指定する必要があります。

### IRM\_ARCH

X'03' (IRM\_ARCH3)。IRM\_ARCH3 は、同期コールアウト・メッセージをリトリートするときに必要です。

### IRM\_F0

同期コールアウト・メッセージをリトリートする場合、以下のいずれかの値を指定する必要があります。

- X'80' (IRM\_F0\_SYNONLY)。IRM\_F0\_SYNONLY を指定すると、同期コールアウト・メッセージのみが IMS Connect クライアントに返されます。
- X'40' (IRM\_F0\_SYNASYN)。IRM\_F0\_SYNASYN を指定すると、同期コールアウト・メッセージと非同期出力の両方が IMS Connect クライアントに返されます。

## 関連資料

224 ページの『IMS Connect へ送信されるメッセージ内の IRM の固定部分のフォーマット』

IMS 要求メッセージ (IRM) ヘッダーには、IMS TM と通信するすべての IMS Connect クライアント・アプリケーションからのすべてのメッセージに共通の 28 バイトの固定フォーマット・セクションがあります。

229 ページの『HWSSMPL0、HWSSMPL1、およびユーザー作成メッセージ出口ルーチンの IRM のユーザー部分のフォーマット』

HWSSMPL0、HWSSMPL1、またはユーザー作成メッセージ出口によってサポートされるユーザー作成クライアント・アプリケーションでは、IMS Connect クライアント入力メッセージ内の IMS 要求メッセージ (IRM) ヘッダーの 4 バイトの長さフィールドと 28 バイトの固定部分に続けて、IRM にユーザー定義セクションを組み込むことができます。

321 ページの『タイマー間隔の指定』

タイマーの値は、いくつかの増分範囲で指定できます。

## RESUME TPIPE プロトコルのフローの例

以下の図は、非同期コミット後送信 (CM0) 出力を受信する RESUME TPIPE プロトコルの例を示しています。

以下の図では、CM0 のフローは、IMS 出力をエンキューしてから、これをクライアントに送信して、クライアント・アプリケーションが両方の出力に対して肯定応答 (ACK) を送信します。ACK によって、IMS キューから出力が除去されます。

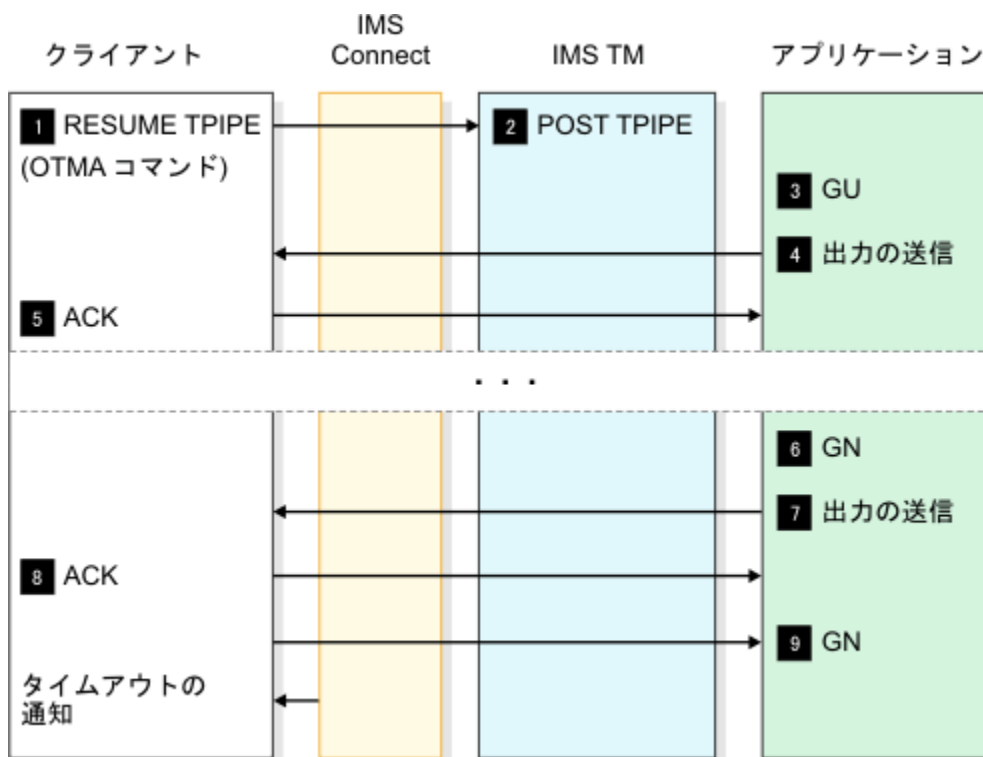


図 32. コミット後送信、非同期出力受信 (クライアントは出力を待機) のフロー

前の図のサンプル・フローは、以下を想定しています。

- クライアントは OTMA 呼び出し RESUME TPIPE を送信して、IMS OTMA に、指定された TPIPE (クライアント名) を通知するように指示する。
- クライアントは、RECEIVE 要求を出して、IMS から出力を受け取る。



- クライアントは ACK を IMS に送る (コミット後送信には必要)。
- クライアントは IMS から次の出力を受け取る。
- クライアントは ACK を IMS に送信する。
- クライアントは、IMS からの次の出力、またはタイムアウト通知を待機する。

以下の図では、コミット後送信フローは、IMS 出力をエンキューしてから、これをクライアントに送信して、クライアント・アプリケーションが、最初の出力に対して肯定応答 (ACK) を送信し (IMS キューから出力を除去)、2 番目の出力に対して NAK を送信します (結果として、出力がキューに残存)。

**要件：**このプロトコルはタイムアウト機能と一緒に使用してください。この機能がないと、送信するメッセージがなくなった場合に、クライアントはハングします。

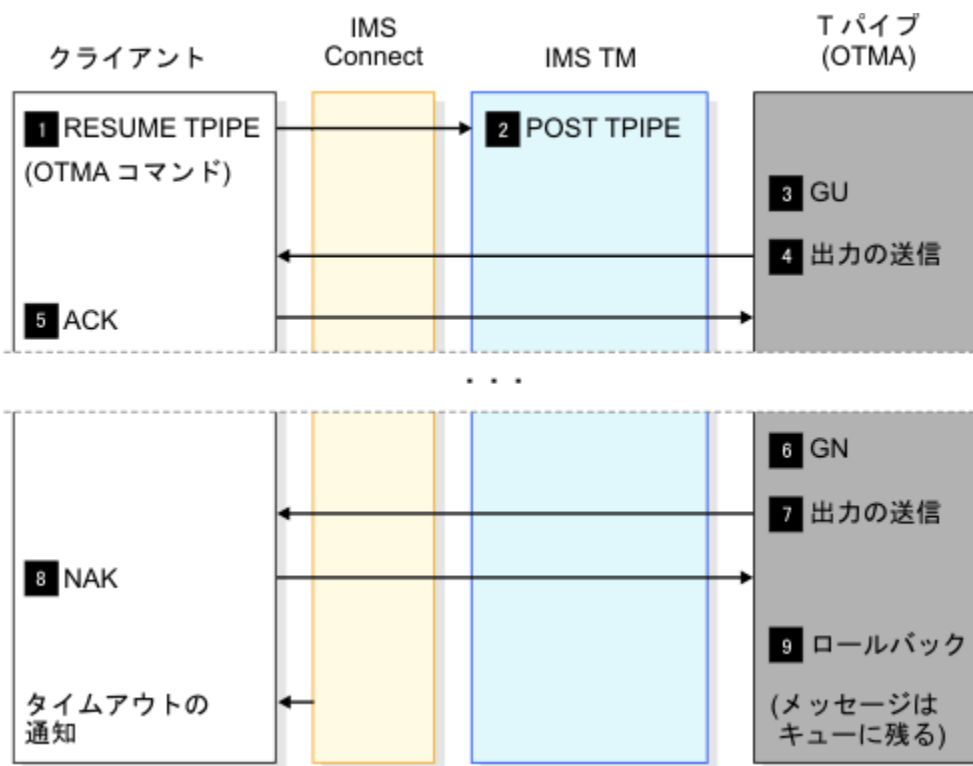


図 33. コミット後送信、非同期出力受信 (出力はキューに残存) のフロー

前の図のサンプル・フローは、以下を想定しています。

- クライアントは OTMA 呼び出し RESUME TPIPE を送信して、IMS OTMA に、指定された TPIPE (クライアント名) を通知するように指示する。
- クライアントは IMS から出力を受け取る。
- クライアントは ACK を IMS に送る (コミット後送信には必要)。
- クライアントは IMS から次の出力を受け取る。
- クライアントは NAK を IMS に送信する。
- メッセージはキュー内に残る。

## 非同期出力サポートの実施

非同期出力サポートは、非同期出力の受信を可能にすることによって、実施します。

### このタスクについて

クライアント・アプリケーションのエンド・ユーザーは、非同期出力を要求する時点を決めることができます。あるいは、クライアント・アプリケーションそのものが、非同期出力を要求する時点を決めることもできます。

**推奨:** 非同期出力サポートは、クライアント・アプリケーションではなく、エンド・ユーザーが非同期出力の要求時点を決められるように、インプリメントしてください。このようにしておくと、次のような利点があります。

- トランザクション入出力を非同期出力と切り離すことができる
- エンド・ユーザーが自分の選択した時間間隔で、非同期出力をリトリートする時点を決定できる

エンド・ユーザーまたはクライアント・アプリケーションが非同期出力を要求するかどうかに関係なく、以下のアクションを以下の順序で行う必要があります。

## 手順

1. CONNECT コマンドを出す。
2. OTMA RESUME TPIPE 呼び出しの TCP/IP SEND が 1 次クライアント・アプリケーションからの TCP/IP READ 機能の直後に続く。
3. 出力メッセージの受信で ACK または NAK 応答の TCP/IP SEND を出す。タイマー値 NOWAIT で ACK が送信された場合 (SINGLE オプションまたは WAIT 付き SINGLE オプションを指定した RESUME TPIPE 呼び出しの場合、NOWAIT のみが有効)、ステップ 5 に進みます。NAK が送信された場合、ステップ 5 に進みます。
4. 1 次クライアント・アプリケーションからの TCP/IP READ 機能。すべてのメッセージが受信されるまで、エンド・ユーザーが必要なすべてのメッセージを受信するまで、エラーが発生するまで、あるいはタイムアウト通知が発生するまで、ステップ 2 と 3 を繰り返します。
5. トランザクション・ソケットを使用している場合、DISCONNECT コマンドを発行する。持続ソケットを使用している場合は、接続は行われたままです。

## タスクの結果

### エンド・ユーザーによる非同期出力要求の使用可能化

クライアント・アプリケーション画面で、グラフィカル・ユーザー・インターフェースのボタンを使用して CONNECT、RESUME TPIPE、READ、ACK/NAK、および DISCONNECT 機能を簡単にインプリメントすることができます。

- CONNECT ボタンを作成する。
- RESUME TPIPE 呼び出しを IMS Connect に送信するための RESUME TPIPE ボタンを作成する。これにより、IMS Connect は RESUME TPIPE 要求を OTMA に送ることができます。
- TCP/IP READ 要求を出すための READ ボタンを作成する。OTMA は、RESUME TPIPE または ACK 応答に続けて、メッセージを IMS Connect に送信します。
- ACK/NAK ボタンを作成する。
- READ 要求を出し、その後、メッセージの受信時に ACK を送信するために、READ 要求と ACK 要求を 1 つのボタンに結合することもできる。
- DISCONNECT ボタンを作成する。

### 並列 RESUME TPIPE 要求による出力のリトリート

OTMA 出力メッセージ (特に、コールアウト要求メッセージ) のスループットを向上させるには、HWS 構成ステートメントまたは DATASTORE IMS Connect 構成ステートメントで MULTIRTP=Y を指定して、OTMA T パイプが複数のアクティブ RESUME TPIPE 要求を並列でサポートすることを使用可能に設定できます。

出力メッセージの逐次処理を保持するために、MULTIRTP=N を指定して、RESUME TPIPE 要求の並列処理を使用不可にすることもできます。

デフォルトでは、OTMA は単一のアクティブ RESUME TPIPE 要求のみをサポートする TPIPE を作成し、アクティブ RESUME TPIPE 要求が終了するまで、その他の RESUME TPIPE 要求をキューに入れます。単一のアクティブ RESUME TPIPE 要求のみをサポートすることによって、OTMA からの出力メッセージの処理順序をより高度に制御できます。

複数のアクティブな **RESUME TPIPE** 要求の並列処理のサポートを有効にすると、出力メッセージに対する OTMA T パイプのスループットを大幅に高めることができ (特に同期または非同期のコールアウト要求の場合)、OTMA T パイプのフェイルオーバー保護を大幅に改善できます。OTMA は、出力メッセージを作成される順に IMS アプリケーション・プログラムから送信しますが、ネットワーク接続と IMS Connect クライアント・アプリケーション・プログラムの両方のパフォーマンスの差があるため、出力の確認と処理が行われる順序が予測不能になります。

MULTIRTP パラメーターが HWS ステートメントと DATASTORE ステートメントの両方に指定されている場合は、DATASTORE ステートメントでの指定が HWS ステートメントでの指定をオーバーライドします。MULTIRTP パラメーターが HWS ステートメントにも DATASTORE ステートメントにも指定されていない場合、並列処理オプションは、OTMA で有効になっている MULTIRTP 値によって決定されます。

## 関連概念

792 ページの『複数のアクティブ RESUME TPIPE 要求の並列処理に対応する OTMA T パイプのサポート』  
OTMA クライアント記述子で MULTIRTP=Y を指定すると、OTMA クライアントに関連付けられた OTMA T パイプは複数のアクティブな RESUME TPIPE 要求を並列にサポートできます (MULTIRTP の指定がクライアントによってオーバーライドされない限り)。

## 並列 RESUME TPIPE 要求のサポートに対する IMS Connect デフォルトの指定

HWS 構成ステートメント内で MULTIRTP パラメーターを指定することにより、この IMS Connect インスタンスからのすべてのデータ・ストア接続を対象に、RESUME TPIPE 要求の並列処理に関するデフォルト・オプションが定義されます。

## このタスクについて

HWS ステートメント内で指定された MULTIRTP 値を個々のデータ・ストア接続についてオーバーライドするには、そのデータ・ストア接続を定義している DATASTORE ステートメント内で MULTIRTP パラメーターを指定します。

## 手順

- このインスタンスからのすべてのデータ・ストア接続に対して、RESUME TPIPE 要求のデフォルトの処理オプションを設定するには、以下のいずれかの MULTIRTP 値を指定します。
  - ブランクのままにした場合、または MULTIRTP パラメーターが省略されている場合、デフォルトで、この IMS Connect インスタンスからのデータ・ストア接続要求に MULTIRTP サポートの指定が含まれていないことを指定します。MULTIRTP support is determined by either the IMS Connect data store definition or the OTMA client descriptor in the DFSYDTx member of the IMS PROCLIB data set.
  - N** IMS Connect データ・ストア接続要求が、IMS Connect が単一のアクティブな再開 TPIPE 要求のみをサポートする OTMA TPIPE を必要としていることを示していることを指定します。TPIPE 上の出力メッセージは、順次に送信されます。
  - Y** IMS Connect データ・ストア接続で、複数のアクティブな再開 TPIPE 要求を並行してサポートできる OTMA TPIPE を必要とすることを指定します。複数のデータ・ストア接続からの複数のアクティブな再開 TPIPE 要求が TPIPE にある場合、OTMA は、受信状態にある最初の使用可能なデータ・ストア接続に、コールアウトまたは CMO 出力メッセージを送信します。



## データ・ストア接続に対する並列 RESUME TPIPE 要求のサポートの指定

データ・ストア接続の定義内で MULTIRTP パラメーターを指定することにより、個々の IMS Connect データ・ストア接続に対して、並列 RESUME TPIPE 要求のサポートを使用可能または使用不可にすることができます。

### このタスクについて

データ・ストア接続の定義における MULTIRTP の指定は、IMS Connect インスタンスのデフォルトの MULTIRTP 値をオーバーライドします。

### 手順

- 個々のデータ・ストア接続について RESUME TPIPE 要求の処理オプションを設定するには、以下のいずれかの MULTIRTP 値を指定します。

**N**

このデータ・ストア接続が、単一のアクティブな RESUME TPIPE 要求のみをサポートする OTMA T パイプを必要とすることを指定します。TPIPE 上の出力メッセージは、順次に送信されます。

**Y**

このデータ・ストア接続で、複数のアクティブ RESUME TPIPE 要求を並行してサポートする OTMA T パイプを必要とすることを指定します。複数の DATASTORE 接続からの複数のアクティブな RESUME TPIPE 要求が T パイプにある場合、OTMA は、受信状態にある最初の使用可能接続にコールアウトまたは CMO 出力メッセージを送信します。

**#**

このデータ・ストア接続に対する MULTIRTP サポートが、OTMA に有効な MULTIRTP 値によって決定されることを指定します。

## 並列 RESUME TPIPE 要求の実装

複数のアクティブ RESUME TPIPE 要求をサポートする OTMA T パイプから出力をリトリブするために、RESUME TPIPE 要求を発行するそれぞれの IMS Connect クライアントは、代替クライアント ID として T パイプの名前を指定します。

### このタスクについて

代替 クライアント ID の指定方法は、使用している IMS Connect クライアントのタイプによって異なります。

並列 RESUME TPIPE 要求をサポートしている OTMA T パイプからの出力をリトリブするには、以下のようになります。

### 手順

- ユーザー提供の IMS Connect クライアントの場合は、以下の値を IRM メッセージ・ヘッダーに指定して、RESUME TPIPE 要求を発行します。

固定 IRM セクションの **IRM\_ARCH** フィールドに X'01' を指定

IRM のユーザー定義セクションの **IRM\_RT\_ALTCID** フィールドに、ターゲット T パイプの名前を指定。ターゲット T パイプの名前は、IMS 内で OTMA 宛先記述子内の TPIPE パラメーターによって定義されるのが一般的です。

- IMS TM リソース・アダプター クライアント・アプリケーションの場合、クライアント・アプリケーションは以下のいずれかのプログラミング・モデルで、IMSInteractionSpec オブジェクトの altClientID プロパティを使用します。

非同期出力プログラミング・モデル

非同期コールアウト・プログラミング・モデル

同期コールアウト・プログラミング・モデル

## タスクの結果

Tパイプに接続する各 **RESUME TPIPE** 要求は、Tパイプに対するアクティブな RESUME TPIPE 要求の最大数に到達するまでアクティブであり、出力を受信することができます。Tパイプに対するアクティブな RESUME TPIPE 要求の最大数は、クライアントの OTMA クライアント記述子内の LIMITRTP パラメーターによって定義されます。

## 並列 RESUME TPIPE 要求に関する問題の解決

IMS Connect と、並列 RESUME TPIPE 要求をサポートする OTMA Tパイプの間の通信に関する問題の診断は、複数の RESUME TPIPE 要求が Tパイプ上で同時にアクティブになる可能性があるため、複雑な作業になります。

### このタスクについて

また、OTMA は、OTMA クライアントが使用する Tパイプの名前によって OTMA クライアントを識別するので、クライアントが代替クライアント ID フィールドに指定した Tパイプ名だけを認識し、実際のクライアント ID を認識しません。

並列 RESUME TPIPE がサポートされているとき、実際のクライアント ID を識別するために、RESUME TPIPE トークン、代替クライアント ID、およびクライアント ID を使用して、IMS Connect クライアントと、そのクライアントが OTMA Tパイプでアクティブにしている特定の RESUME TPIPE 要求を関連付けることができます。

並列 RESUME TPIPE 要求をサポートする OTMA Tパイプ上でアクティブな各 **RESUME TPIPE** 要求をどのクライアントが発行したかを識別するには、以下のようにします。

### 手順

- IMS コマンド **/DISPLAY TMEMBER** *ims\_connect\_name* TPIPE *tpipe\_nm* OUTPUT を発行します。
- /DISPLAY TMEMBER** コマンドの出力に表示される、IMS Connect インスタンスと、それに関連する RESUME TPIPE トークンに注意してください。
- IMS Connect コマンド **QUERY IMSCON TYPE(CLIENT) RTTOKEN**(*rt\_token\_number*) SHOW(ALTCID RTTOKEN) を発行します。
- QUERY IMSCON コマンドによって表示される情報の中で、RESUME TPIPE トークンを見つけます。RESUME TPIPE 呼び出しを発行したクライアント ID は、関連するトークンと同じ行にある見出しの ClientID の下に表示されます。

### 例

次の例では、IMS Connect インスタンス HWS1 の CLIENT03 が OTMA Tパイプ CLIENT99 への RESUME TPIPE 呼び出しを発行しました。ただし、OTMA は、OTMA クライアントが使用している Tパイプの名前によってクライアントを識別するので、CLIENT99 だけが表示され、CLIENT03 は表示されません。

CLIENT03 によって発行された特定の RESUME TPIPE 呼び出しを判別するには、IMS コマンド **/DISPLAY TMEMBER** を HWS1 および TPIPE ALL の指定を付けて発行し、現在 Tパイプ上でアクティブな RESUME TPIPE 呼び出しの RESUME TPIPE トークンを表示します。特定の Tパイプへの出力を制限するには、ALL キーワードの代わりに Tパイプ名 (この例では CLIENT99) を指定します。

```
/DIS TMEM HWS1 TPIPE ALL OUTPUT
DFS000I MEMBER/TPIPE ENQCT DEQCT QCT INPCT STATUS
DFS000I HWS1
DFS000I -CLIENT99 0 0 0 0 PMRY
DFS000I -CLIENT99 0 0 0 0 HLDQ
DFS000I -CLIENT99 RT CBB45E89AAF02E8A OPT A MODE S
```

RESUME TPIPE トークンを判別した後、そのトークンを RTTOKEN キーワードに指定し、SHOW(ALTCID) を指定して、以下のように QUERY IMSCON TYPE(CLIENT) コマンドを発行します。

```
QRY IMSCON TYPE(CLIENT) RTTOKEN(CBB45E89AAF02E8A) SHOW(ALTCID RTTOKEN)
```

RTTOKEN および ALTCID に対応するクライアント ID が、RTTOKEN 値および ALTCID 値とともに表示されます。

ClientID	MbrName	CC	ALTCID	RTToken
CLIENT03	IMSPLEX1	0	CLIENT99	CBB45E89AAF02E8A

## 出力メッセージのリトリブの管理

RESUME TPIPE 呼び出しで非同期出力メッセージまたは同期コールアウト要求メッセージをリトリブする場合、メッセージを返す方法について、いくつかのオプションがあります。

RESUME TPIPE 要求メッセージの IRM にリトリブ・オプションを指定し、ユーザー・メッセージ出口で OTMA ヘッダーにオプションを設定します。

指定できるリトリブ・オプションを以下に示します。

- Single
- Single with wait
- Noauto
- Nooption
- Auto

IMS Connect ユーザー・メッセージ出口 HWSSMPL1 および HWSSMPL0 は、これらのオプションをすべてサポートします。メッセージ制御のタイプを選択するには、クライアント・コードで、IRM フィールドの IRM\_FLG5 を以下の値のいずれかに設定します。

### IRM\_F5\_ONE

単一メッセージを取得します (単一 (single))。

### IRM\_F5\_SWAIT

IMS メッセージ・キュー内に現在何も存在しない場合、単一メッセージを待機します (待機付き単一 (single with wait))。

### IRM\_F5\_NOAUTO

キューに入れられているすべてのメッセージを取得します (非自動 (noauto))。

### IRM\_F5\_AUTO

キューに入れられていたすべてのメッセージを取得し、その後、後でキューに入れられる追加メッセージも取得します (自動 (auto))。

### IRM\_F5

X'00' に設定すると、RESUME TPIPE 呼び出し機能が NOAUTO (オプション) に類似するものになります。

HWSSMPL0 および HWSSMPL1 ユーザー・メッセージ出口は、デフォルトでは非自動 (noauto) タイプの非同期出力メッセージ管理になります。

以降のサブセクションでは、非同期出力メッセージの制御オプションについて詳しく説明します。

## 単一メッセージ制御

単一メッセージ制御オプションを使用すると (IRM\_F5 フィールド を IRM\_F5\_ONE に設定)、クライアントはメッセージを **1** つだけ受信できます。

要求が行われたときにクライアント ID に対応する IMS OTMA 非同期キューにメッセージが存在しない場合、メッセージは戻されずに、タイムアウトが発生します。単一メッセージ制御オプションを使用すると、以下の一連のイベントの発生を強制されます。

1. クライアントが CONNECT 機能を出す。
  - a. ソケット・タイプが持続ソケットの場合、CONNECT 機能に続けて 1 つ以上のトランザクションを送信することができ、その応答は RESUME TPIPE 呼び出し処理前に受信される。
  - b. ソケット・タイプがトランザクション・ソケットの場合、CONNECT 機能の後で RESUME TPIPE 呼び出しを発行する必要がある。

2. クライアントが、正しい IRM 設定で RESUME TPIPE 呼び出しを発行する。
3. クライアントが、非同期出力を受信するために RECEIVE 機能を出す。
4. クライアントが ACK または NAK を IMS Connect に送信する。
  - a. ユーザー指定の数字のタイムアウト値を設定して ACK または NAK を送信できる。  
あるいは
  - b. タイムアウト値に NOWAIT を指定する。
5. タイムアウト値の数字が指定された場合、クライアントは、タイムアウト通知を受信するために RECEIVE 機能を出さなければならない。NOWAIT オプションが指定された場合、タイムアウト通知が送信されません。したがって、NOWAIT が指定された場合は、クライアントは RECEIVE 機能を出してはなりません。
6. ソケット接続がトランザクション・ソケットの場合、IMS Connect はホスト・エンドからソケットを切断する。ソケット接続が持続ソケットの場合は、IMS Connect はソケットを切断しません。
7. ソケット接続がトランザクション・ソケットの場合、クライアントは DISCONNECT 機能を出さなければならない。ソケットが持続ソケットの場合は、クライアントはソケットを切断するか、トランザクション・コードおよびデータの SENDONLY、SEND などの新規要求を送信するか、あるいは別の RESUME TPIPE 呼び出しを発行することを選択できます。

クライアントが ACK ではなく NAK で応答すると、NAK で応答されたメッセージは、OTMA 非同期保留キューに戻されます。これは、後で再リトリブすることができます。NAK がクライアントによって IMS Connect に送信された場合、IMS Connect は、イベント 5 から 7 までに説明されているとおりに処理を継続します。

## 待機付き単一メッセージ制御

待機付き単一メッセージ制御オプションを使用する場合 (IRM\_F5 を IRM\_F5\_SWAIT に設定)、クライアントは 1 つのメッセージのみを取得できます。ただし、単一メッセージ制御とは異なり、待機付き単一メッセージ制御は、クライアント ID に対応する IMS OTMA 非同期キューに入れられているメッセージを受信できます。

待機付き単一メッセージ制御オプションを使用するには、以下の一連のイベントを行う必要があります。

1. クライアントが CONNECT 機能を出す。
  - a. ソケット・タイプが持続ソケットの場合、CONNECT 機能に続けて 1 つ以上のトランザクションを送信することができ、その応答は RESUME TPIPE 呼び出し処理前に受信される。
  - b. ソケット・タイプがトランザクション・ソケットの場合、CONNECT 機能の後で RESUME TPIPE 呼び出し処理を実行する必要がある。
2. クライアントが、正しい IRM 設定で RESUME TPIPE 呼び出しを発行する。
3. クライアントが、非同期出力を受信するために RECEIVE 機能を出す。
4. クライアントが ACK または NAK を IMS Connect に送信する。
  - a. ユーザー指定のタイムアウト通知を使用して ACK または NAK を送信できる。  
あるいは
  - b. タイムアウト値に NOWAIT を指定する。
5. タイムアウト値の数字が指定された場合、クライアントは、タイムアウト通知を受信するために RECEIVE 機能を出さなければならない。NOWAIT オプションが指定された場合、タイムアウト通知が送信されません。したがって、NOWAIT が指定された場合は、クライアントは RECEIVE 機能を出してはなりません。
6. ソケット接続がトランザクション・ソケットの場合、IMS Connect はホスト・エンドからソケットを切断する。ソケット接続が持続ソケットの場合は、IMS Connect はソケットを切断しません。
7. ソケット接続がトランザクション・ソケットの場合、クライアントは DISCONNECT 機能を出さなければならない。ソケットが持続ソケットの場合は、クライアントはソケットを切断するか、トランザクション・コードおよびデータの SENDONLY、SEND などの新規要求を送信するか、あるいは別の RESUME TPIPE 呼び出しを発行することを選択できます。

クライアントが ACK ではなく NAK で応答すると、NAK で応答されたメッセージは、OTMA 非同期保留キューに戻されます。これは、後で再リトリブすることができます。NAK がクライアントによって IMS Connect に送信された場合、IMS Connect は、イベント 5 から 7 までに説明されているとおりに処理を継続します。

## 非自動メッセージ制御

非自動メッセージ制御オプションを使用すると (IRM\_F5 フィールドを IRM\_F5\_NOAUTO に設定)、クライアントは、OTMA 非同期キュー上のすべてのメッセージを受信できます。

非自動メッセージ制御オプションを使用すると、以下の一連のイベントの発生を強制されます。

1. クライアントが CONNECT 機能を出す。
  - a. ソケット・タイプが持続ソケットの場合、CONNECT 機能に続けて 1 つ以上のトランザクションを送信することができ、その応答は RESUME TPIPE 呼び出し処理前に受信される。
  - b. ソケット・タイプがトランザクション・ソケットの場合、CONNECT 機能の後で RESUME TPIPE 呼び出し処理を実行する必要がある。
2. クライアントが RESUME TPIPE 呼び出しを発行する。
3. クライアントが、非同期出力を受信するために RECEIVE 機能を実行する。
4. クライアントが ACK メッセージを IMS Connect に送信する。
5. クライアントがイベント 3 と 4 を、イベント 6 が起きるまで繰り返す。
6. IMS Connect は、ソケットが持続ソケットでない場合は、ホスト・エンドからソケットを切断する。ソケットが持続ソケットである場合は、IMS Connect はタイムアウト・メッセージを送信し、ソケットは接続されたままになります。これにより、クライアントは接続を切断するか、別の要求を送信することができます。
7. クライアントが DISCONNECT 機能を出す。

非自動メッセージ制御オプションを使用する場合、クライアントは、ACK メッセージを IMS Connect に送信した後、DISCONNECT 機能を実行することによっていつでも終了することができます。

クライアントが ACK ではなく NAK メッセージで応答すると、NAK 応答を受信したメッセージは、OTMA 非同期保留キューに書き戻されるため、後からリトリブすることができます。IMS Connect は、イベント 6 で説明したように、ソケットが持続ソケットでない場合はソケットを終了します。

## 非オプション・メッセージ制御

非オプション・メッセージ制御オプションを使用すると (フィールド IRM\_F5 を X'00' に設定)、クライアントは、OTMA 非同期キュー上のすべてのメッセージを受信できます。

非オプション・メッセージ制御オプションを使用するには、以下の一連のイベントを行う必要があります。

1. クライアントが CONNECT 機能を出す。
  - a. ソケット・タイプが持続ソケットの場合、CONNECT 機能に続けて 1 つ以上のトランザクションを送信することができ、その応答は RESUME TPIPE 呼び出し処理前に受信される。
  - b. ソケット・タイプがトランザクション・ソケットの場合、CONNECT 機能の後で RESUME TPIPE 呼び出し処理を実行する必要がある。
2. クライアントが RESUME TPIPE 呼び出しを発行する。
3. クライアントが、非同期出力を受信するために RECEIVE 機能を実行する。
4. クライアントが ACK メッセージを IMS Connect に送信する。
5. クライアントがイベント 3 と 4 を、イベント 6 が起きるまで繰り返す。
6. IMS Connect は、ソケットが持続ソケットでない場合は、ホスト・エンドからソケットを切断する。ソケットが持続ソケットである場合は、IMS Connect はタイムアウト・メッセージを送信し、ソケットは接続されたままになります。これにより、クライアントは接続を切断するか、別の要求を送信することができます。
7. クライアントが DISCONNECT 機能を出す。

非オプション・メッセージ制御オプションを使用する場合は、クライアントは、ACK を IMS Connect に送った後、DISCONNECT 機能を出すことによっていつでも終了することができます。

クライアントが ACK ではなく NAK メッセージで応答すると、NAK 応答を受信したメッセージは、OTMA 非同期保留キューに書き戻されるため、後からリトリブすることができます。IMS Connect は、上記のイベント 6 で説明したように、ソケットが持続ソケットでない場合はソケットを終了します。

## 自動メッセージ制御

自動メッセージ制御オプションを使用すると (IRM\_F5 フィールドを IRM\_F5\_AUTO に設定)、クライアントは、OTMA 非同期キュー上のすべてのメッセージを受信し、かつ、現在のメッセージがすべて除かれた後、OTMA 非同期キューに置かれるメッセージがあればそれも受信できます。

自動メッセージ制御オプションを使用すると、以下の一連のイベントの発生を強制されます。

1. クライアントが CONNECT 機能を出す。
2. クライアントが RESUME TPIPE 呼び出しを発行する。
3. クライアントが、非同期出力を受信するために RECEIVE 機能を実行する。
4. クライアントが ACK メッセージを IMS Connect に送信する。
5. クライアントがイベント 3 と 4 を繰り返す。

すべてのメッセージがキューから除かれていれば、ユーザーが IRM に指定したタイマーの有効期限切れるまで、イベント 3 がアクティブ (すなわち、受信状態) のままになります。次に、IMS Connect は、ソケットが持続ソケットでない場合はソケットを終了します。

**推奨事項:** イベント 3 またはイベント 5 でソケットの切断を受け取った場合は、クライアントはいったん切断し、時間間隔が経過するのを待ってから、イベント 1 から 5 を反復してください。

自動メッセージ制御オプションを使用すると、クライアントは、次のいずれかの方法で非持続ソケット接続をいつでも終了できます。

- NAK 応答を使用して出力メッセージに応答する。
- ACK 応答ではなく、DEALLOCATE 要求を送信する。

処理中のメッセージは、IMS 出力キューに書き戻され、ソケットが持続ソケットでない場合は、IMS Connect はソケットを終了します。

クライアントが ACK メッセージで応答してから、DISCONNECT を実行した場合、接続はクライアントと TCP/IP の間でのみ終了します。IMS Connect に関してはクライアントは、CONN 状態のままです。IMS Connect が次の非同期出力メッセージを送信しようとする時、IMS Connect は、接続が失われたことを通知されます。IMS Connect は OTMA に応答せず (NAK)、メッセージは IMS 出力キューに書き戻されます。その後、IMS Connect は、ソケットを終了します。クライアントが ACK メッセージを出してから、DISCONNECT を実行し、続いて CONNECT およびデータの伝送を行った場合、IMS Connect はクライアント ID の重複エラーで応答し、ソケット接続を切断します。

クライアントがイベント 3 または 5 で、ACK ではなく NAK メッセージで応答すると、NAK メッセージを受信したメッセージは、OTMA 非同期保留キューに書き戻され、IMS Connect はソケットを終了します。このメッセージは、後からリトリブすることができます。

**注:** IMS Connect の AUTO サポートは、ソケット接続が出力専用装置として占有されているという前提に基づいています。RESUME TPIPE 呼び出し (自動非同期オプションを指定) を同じソケット接続上のトランザクションと結合するか、または持続ソケット上の SENDONLY のトランザクションと結合すると、予測不能の結果が発生する可能性があります。RESUME TPIPE 自動オプション・モードからトランザクション処理が可能なモードに変更する場合、以下のいずれかのオプションを実行して自動非同期オプションを変更する必要があります。

1. RESUME TPIPE 出力の 1 つに NAK で応答します。これにより、非同期モードが自動から非自動に変更されます。自動モードに戻すには、AUTO を設定した RESUME TPIPE 呼び出しを指定する必要があります。
2. RESUME TPIPE AUTO と関連付けられたタイムアウト通知の場合、クライアント・アプリケーションは、切断、再接続、および RESUME TPIPE 呼び出し (非常に短い IRM\_TIMER 値を設定した、単一、待機付き単一、非自動、または非オプションを指定) の発行を行うことができます。IRM\_TIMER 値は、タイム

アウト通知が即時に戻されるように、小さな値でなければなりません。これらの4つの非同期モード・オプションの1つを指定した RESUME TPIPE 呼び出しを発行すると、モードが「自動」から指定したオプションに変更されます。RESUME TPIPE 呼び出しが発行され、タイムアウト通知が返されたら、クライアント・アプリケーションはトランザクションを送信できます。

3. RESUME TPIPE AUTO と関連付けられたタイムアウト通知の場合、クライアント・アプリケーションは、切断、再接続、および RESUME TPIPE 呼び出し (有効な IRM\_TIMER 値を設定した、単一、待機付き単一、非自動、または非オプションを指定) の発行を行うことができます。出力メッセージを受信したら、IRM\_TIMER を NOWAIT または有効な値に設定して ACK を送信します。IRM\_TIMER 値が NOWAIT に設定された場合、クライアントはトランザクションを送信できます。IRM\_TIMER が有効な値に設定された場合、タイムアウト通知を受信すると、クライアント・アプリケーションはトランザクションを送信できます。

## 関連概念

320 ページの『入力メッセージのタイムアウトの指定』

IMS Connect クライアントからの個々の入力メッセージは、いずれも IMS 要求メッセージ (IRM) ヘッダーの固定部分の IRM\_TIMER フィールドに異なるタイムアウト値を設定できます。

## 自動メッセージ制御オプションを指定した RESUME TPIPE 呼び出し処理時の実行タイムアウト

自動メッセージ制御オプションを指定した RESUME TPIPE 呼び出しを使用しているときに、IRM\_TIMER 値がタイムアウトになった場合、予測不能な結果が発生する可能性があります。

### このタスクについて

RESUME TPIPE 呼び出しで自動オプションを選択しているときに、タイムアウトが発生した場合、タイムアウト通知を取得してトランザクションを再送信するには、自動オプション処理モードを非自動に変更する必要があります。自動オプション処理モードを解消するために、以下のオプションのいずれか1つを選択できます。

- 自動オプションを指定した RESUME TPIPE 呼び出しを発行し、大きな IRM\_TIMER 値を設定して、クライアント・アプリケーションが出力に NAK で応答するようにします。出力が NAK の場合、OTMA は、非同期モードを自動から非自動に変更して、非同期出力の送信を停止します。続いて、クライアント・アプリケーションは、READ を発行してタイムアウト通知を取得します。タイムアウト通知を受信すると、クライアントは IMS Connect へのトランザクションの送信を開始できます。
- 非自動オプションを指定した RESUME TPIPE 呼び出しを発行し、IRM\_TIMER フィールドに任意の値を設定します。ACK 出力の受信後、タイムアウト通知が受信されるまで、非同期出力の READ および ACK の SEND を繰り返します。(非自動を指定して RESUME TPIPE 呼び出しを発行すると、処理モードが自動から非自動に変更されます。)また、これにより、OTMA の非同期モードが非自動に再設定されます。これで、OTMA は、IMS メッセージ・キューが空の場合、非同期出力の自動送信をサポートしなくなります。)続いて、クライアント・アプリケーションは、READ を発行してタイムアウト通知を取得します。タイムアウト通知を受信すると、クライアントは IMS Connect へのトランザクションの送信を開始できます。
- 非自動オプションを指定した RESUME TPIPE 呼び出しを発行し、IRM\_TIMER フィールドに任意の値を設定します。NAK 出力を受信した場合、処理モードおよび OTMA 非同期モードは非自動に再設定されます。OTMA 非同期モードを非自動に再設定すると、非同期出力の送信が停止され、NAK 出力は処理を終了します。続いて、クライアント・アプリケーションは、READ を発行してタイムアウト通知を取得します。タイムアウト通知を受信すると、クライアントは IMS Connect へのトランザクションの送信を開始できます。
- 単一オプションを指定した RESUME TPIPE 呼び出しを発行し、IRM\_TIMER フィールドに任意の値を設定します。OTMA 非同期モードは、自動から単一に再設定され、非同期メッセージはこれ以上送信されません。IRM\_TIMER に NO\_WAIT 以外の任意の値が設定された ACK または NAK 出力を受信すると、単一オプションは完了し、クライアント・アプリケーションは READ を発行してタイムアウト通知を取得することができます。タイムアウト通知を受信すると、クライアントは IMS Connect へのトランザクションの送信を開始できます。
- 単一オプションを指定した RESUME TPIPE 呼び出しを発行し、IRM\_TIMER フィールドに任意の値を設定します。OTMA 非同期モードは、自動から単一に再設定され、非同期メッセージはこれ以上送信されません。IRM\_TIMER に NO\_WAIT が設定された ACK または NAK 出力を受信すると、単一オプションは完了

し、クライアント・アプリケーションはタイムアウト通知を取得するために READ を発行する必要があります。クライアント・アプリケーションは、IMS Connect へのトランザクションの送信を開始できません。

## 非同期出力処理のための値

OTMA T パイプ保留キューから非同期出力をリトリブするには、IMS Connect クライアント・アプリケーションによって RESUME TPIPE 要求を出す必要があります。RESUME TPIPE 要求を出すには、ソケット・タイプ、コミット・モード、同期レベル、タイマー設定、および RESUME TPIPE 呼び出しオプションを指定する必要があります。

## このタスクについて

RESUME TPIPE 呼び出しを出すには、以下の値を指定する必要があります。

### 手順

- ソケット・タイプ  
「トランザクション」または「持続」
- コミット・モード  
ゼロ
- 同期レベル  
確認
- タイマー設定  
貴社で要求されるタイムアウト範囲
- RESUME TPIPE オプション  
単一 (single)、待機付き単一 (single with wait)、自動 (auto)、非自動 (noauto)、または非オプション (nooption)

## タスクの結果

例えば、非送信請求出力のみを受信する専用出力クライアントを作成する場合、以下の一連の処理を完了するようにクライアント・アプリケーションを開始します。

1. クライアント・アプリケーションは接続シーケンスを実行する。
2. クライアント・アプリケーションは、IRM に正しい設定値を指定して RESUME TPIPE 呼び出しを送信する。

**推奨事項:** IRM\_TIMER 値を X'FF' に設定してください。これにより、IMS Connect は、構成ファイルの TIMEOUT 値をオーバーライドして、常に待機します。

3. クライアント・アプリケーションは、出力メッセージを受信するために TCP/IP READ を送信する。
4. クライアント・アプリケーションは確認応答 (ACK-IRM\_TIMER 値を RESUME TPIPE 呼び出しで設定したのと同じ値に設定してください) を送信し、TCP/IP READ に戻る。

IRM\_TIMER に設定される時間間隔は、IMS Connect 構成ファイルに設定されている時間間隔 (TIMEOUT= の値) とは別のものです。

IRM\_TIMER 値は、RESUME TPIPE 呼び出しに続いてクライアントから発行される RECEIVE を待機する値、または RESUME TPIPE 呼び出しに続く RECEIVE に対する ACK を待機する値です。



## 代替 OTMA T パイプ保留キューからの出力のリトリブ

クライアント・アプリケーションは、代替の T パイプの名前を、代替のクライアント ID である RESUME TPIPE 呼び出しとして指定することで、代替の T パイプ保留キューから非同期出力またはコールアウト・メッセージをリトリブできます。

### このタスクについて

IMS Connect が、代替クライアント ID を指定した RESUME TPIPE 呼び出しを OTMA に渡すと、OTMA は、代替クライアント ID に一致する T パイプのキューに入っているメッセージがあれば、それを呼び出し側へ返します。

代替クライアント ID の指定は、以下のシナリオで出力をリトリブするために使用されます。

- クライアント ID が実行時に自動的に生成されたために不明である場合に、IMS TM リソース・アダプターによって。
- Sysplex Distributor 環境で。この環境では、クライアント・アプリケーションは通常、出力がどの T パイプ保留キューに入れられているか知りません。
- T パイプが複数のアクティブ RESUME TPIPE 要求をサポートしている場合、T パイプから出力をリトリブするすべてのクライアントは、T パイプ名を代替クライアント ID として指定します。
- コールアウト環境で。この環境では、T パイプ名は通常 IMS 内で OTMA 宛先記述子、または OTMA 経路指定出口 (非同期コールアウトの場合のみ) によって定義されます。この場合、コールアウト・メッセージは、T パイプ名を代替クライアント ID として指定してリトリブされます。

代替クライアント ID の使用により、クライアント・アプリケーション・プログラムは、IRM ヘッダーまたは OTMA ヘッダーの代替クライアント ID フィールドに T パイプ名を指定することで、IMS Connect の任意のインスタンスを介して出力をリトリブできます。

代替クライアント ID を使用して出力をリトリブしている場合、/DISPLAY TMEMBER TPIPE コマンドは、RESUME TPIPE 要求をサブミットしたクライアントの実際の ID の代わりに代替クライアント ID を表示します。その結果、問題を診断する必要がある場合に、どのクライアントが特定の RESUME TPIPE 要求をサブミットしたかを特定するのが難しいことがあります。

ただし、ターゲット T パイプが並列 RESUME TPIPE 要求をサポートしている場合、/DISPLAY TMEMBER TPIPE は RESUME TPIPE トークンを表示します。これを使用して、IMS Connect コマンドの QUERY IMSCON TYPE(CLIENT) によって表示された出力で真のクライアント ID を特定できます。

### 手順

- ユーザー提供のクライアント・アプリケーションは、RESUME TPIPE 呼び出しの IRM に以下を指定することにより、代替 T パイプから非同期出力をリトリブすることができます。

固定 IRM セクションの IRM\_ARCH フィールドに X'01' を指定

IRM のユーザー定義セクションの IRM\_RT\_ALTCID フィールドに、ターゲット T パイプの名前を指定

- IMS TM リソース・アダプター・クライアント・アプリケーションの場合、代替 T パイプから非同期でリトリブするには、クライアント・アプリケーションは OTMA ヘッダーに以下の値を指定する必要があります。

OTMA ヘッダーの OMUSR\_RT\_ALTCID フィールドに、ターゲット T パイプの名前を指定

OMUSR\_ARCLEV フィールドに OMUSR\_AL02 を指定

## 共用非同期出力のグループ定義

データ・ストア接続、IMS Connect インスタンス、またはその両方のグループを定義し、グループのメンバーが、グループの他のメンバーの非同期出力をリトリブできるようにすることができます。これらのグループは、OTMA スーパーメンバー 関数を使用します。

### このタスクについて

スーパーメンバー・グループが定義されていない場合、非同期出力のリトリブは、発信元クライアントおよび発信元 IMS Connect インスタンスのクライアント ID を使用することによってのみ可能です。例えば、発信元クライアント ID が自動的に生成されたことなどが原因でその ID が不明である場合、または z/OS Sysplex Distributor などの製品が使用されていることなどが原因で IMS Connect の元のインスタンスが不明である場合、発信元のクライアント ID と IMS Connect インスタンスに従属関係があると問題が発生する可能性があります。

クライアント ID と IMS Connect インスタンスが既知である場合でも、IMS Connect インスタンスが Single Point of Failure となることによって生じるリスクがあります。つまり、IMS Connect インスタンスが失敗すると、IMS Connect インスタンスが元に戻るまで、非同期出力をリトリブすることはできません。

IMS Connect インスタンスと個々のデータ・ストア接続を、スーパーメンバー・グループの参加プログラムとして定義して、スーパーメンバー・グループを作成します。同じスーパーメンバー・グループ内の各参加プログラムは、IMS.PROCLIB データ・セットの IMS Connect 構成メンバーにある SMEMBER パラメーター上で、同じスーパーメンバー名を指定する必要があります。IMS Connect の場合、SMEMBER パラメーターは HWS ステートメント上にあります。データ・ストア接続の場合、SMEMBER パラメーターは DATASTORE ステートメント上にあります。

デフォルトでは、IMS Connect のインスタンスに定義されているデータ・ストア接続は、そのインスタンスが参加プログラムになっているスーパーメンバー・グループ (もしあれば) の参加プログラムです。ただし、SMEMBER パラメーターを DATASTORE ステートメントに指定することで、データ・ストア接続を、異なるスーパーメンバー・グループに属するようになり、また、いずれのスーパーメンバー・グループにも属さないようにしたりできます。DATASTORE ステートメントに行われたスーパーメンバー・グループに対する指定は、HWS ステートメントに行われたすべてのスーパーメンバー・グループの指定をオーバーライドします。

DATASTORE ステートメントに SMEMBER パラメーターは含まれないが、HWS ステートメントにはそのパラメーターが含まれる場合、データ・ストア接続の属性を表示すると、デフォルトでデータ・ストアが参加プログラムになっているスーパーメンバー・グループ名がそれらの属性に含まれなくなることに注意してください。HWS ステートメントの属性を表示して、スーパーメンバー・グループ名を確認する必要があります。

### 手順

- IMS Connect インスタンスを、スーパーメンバー・グループの参加プログラムとして定義するには、HWS 構成ステートメントの SMEMBER パラメーターに、スーパーメンバー名を指定します。  
HWS 構成ステートメントに指定されたスーパーメンバー名は、自身のスーパーメンバー・グループを指定しない IMS Connect のインスタンスに定義されるすべてのデータ・ストア接続に対する、デフォルトのスーパーメンバー名になります。
- データ・ストア接続を、HWS 構成ステートメントに指定されたスーパーメンバー・グループ以外のスーパーメンバー・グループに参加プログラムとして定義するには、DATASTORE 構成ステートメントの SMEMBER パラメーターに、スーパーメンバー名を指定します。  
DATASTORE ステートメントに指定されたスーパーメンバー名は、このデータ・ストア接続に関して、HWS 構成ステートメントに定義されたすべてのスーパーメンバー名をオーバーライドします。
- IMS Connect インスタンスによって定義されたデフォルトのスーパーメンバー・グループに、データ・ストア接続が参加しないようにするには、データ・ストア接続を定義する DATASTORE 構成ステートメントに、SMEMBER='####' を指定します。

## 例

以下の IMS Connect 構成ステートメントの例には、スーパーメンバー・グループの指定が含まれています。HWS 構成ステートメントの SMEMBER パラメーターは、スーパーメンバー・グループ SHRD に参加するものとして IMS Connect インスタンス HWS1 を定義し、この構成メンバーに定義されるすべてのデータ・ストア接続に対するデフォルトのスーパーメンバー・グループとして SHRD を設定します。

この例では、IMS Connect インスタンス HWS1 に定義されたデータ・ストアは、それぞれ、以下のスーパーメンバー・グループに入ります。

- IMS1 は、HWS 構成ステートメントに定義された SHRD スーパーメンバー・グループに入ります。これは、その DATASTORE ステートメントに SMEMBER パラメーターが含まれないからです。
- IMSY と IMSN は両方とも、スーパーメンバー・グループ SHR1 に入ります。
- IMSX は、おそらく IMS Connect の別のインスタンスからのその他のデータ・ストア接続とともに、スーパーメンバー SHR2 に入ります。
- IMSA は、いずれのスーパーメンバー・グループにも入りません。

```
HWS=(ID=HWS1,XIBAREA=100,RACF=Y,SMEMBER=SHRD)
TCPIP=(HOSTNAME=TCPIP,PORTID=(9999,LOCAL),RACFID=GOFISHIN,TIMEOUT=0,
IPV6=N,SSLPORT=(9998),SSLENVAR=HWSCFSSL,
EXIT=(HWSSMPL0,HWSSMPL1,HWSCSLO0,HWSCSLO1))
DATASTORE=(ID=IMS1,GROUP=XCFGRP1,MEMBER=HWS1,TMEMBER=IMS1)
DATASTORE=(ID=IMSY,GROUP=XCFGRP1,MEMBER=HWSY,TMEMBER=IMS1,SMEMBER=SHR1)
DATASTORE=(ID=IMSN,GROUP=XCFGRP1,MEMBER=HWSN,TMEMBER=IMS1,SMEMBER=SHR1)
DATASTORE=(ID=IMSX,GROUP=XCFGRP1,MEMBER=HWSX,TMEMBER=IMS1,SMEMBER=SHR2)
DATASTORE=(ID=IMSA,GROUP=XCFGRP1,MEMBER=HWSA,TMEMBER=IMSA,SMEMBER=#####)
IMSPLEX=(MEMBER=HWSPLEX1,TMEMBER=PLEX1)
IMSPLEX=(MEMBER=HWSPLEX2,TMEMBER=PLEX2)
```

## 関連タスク

[875 ページの『非同期コミット後送信出力の共用: OTMA スーパーメンバー 機能』](#)

保留キュー対応 OTMA クライアント (IMS Connect など) は、OTMA のスーパーメンバー 機能を有効にすることにより、非同期コミット後送信 (CMO) 出力メッセージを共用できます。OTMA スーパーメンバー 機能は、z/OS Sysplex Distributor 環境で IMS Connect の複数のインスタンスをサポートするために、特に設計されたものです。

## 関連資料

[IMS PROCLIB データ・セットの HWSCFGxx メンバー \(システム定義\)](#)

## 非同期出力メッセージ・フロー

非同期出力サポートをインプリメントすると、コミット後送信 (コミット・モード 0) のメッセージ・フローになります。このフローでは、クライアントからの確認応答 (ACK/NAK) が必要です。

コミット後送信メッセージのフローで実行される IMS トランザクションがメッセージをクライアントに送ったとき、そのメッセージが送達できないと、OTMA は NAK が IMS Connect から OTMA に送られたかのように反応し、メッセージは OTMA 保留キューに入れられます。OTMA は、NAK が送られることになった原因 (例えば、z/OS システム間カップリング・ファシリティ接続が利用できなかった、IMS Connect が終了していた、IMS Connect が TCP/IP との通信を切断していた) には関係なく、常にこの方法で処理します。

## 関連タスク

[301 ページの『配信不能なコミット後送信出力のページ』](#)

トランザクションを開始した OTMA クライアント・アプリケーションにコミット後送信 (コミット・モード 0) IOPCB 出力を返せない場合に、その出力をページするように OTMA を構成することができます。

## 関連資料

[249 ページの『メッセージ出口からクライアントへの出力メッセージ』](#)

メッセージ構造のフォーマットは、IMS Connect クライアントのタイプとクライアントのサポートに使用されるユーザー・メッセージ出口によって異なります。

[261 ページの『IMS Connect で使用される OTMA ヘッダー・フィールド』](#)

IMS Connect は、OTMA に送信されるメッセージのヘッダー内のフィールドを使用して、処理オプションおよびその他の情報を IMS に伝えます。

## IMS Connect クライアント呼び出しのフロー

以下の例は、IMS Connect クライアントの会話型トランザクションと非会話型トランザクションのフローを示しています。

どのサンプル・フローも、持続 TCP/IP ソケットとトランザクション TCP/IP ソケットの両方に適用されます。また、すべてのフローで、コミット・モード 1 (送信後コミット)、同期レベル = CONFIRM、ACK および NAK 付き、のプロトコルを使用しています。

次のサンプル・フローを示します。

- 非会話型で、ACK を使用して正常終了まで実行する場合
- 会話型で、ACK を使用して正常終了まで実行する場合
- 非会話型で、クライアントがメッセージに応答して NAK を送る場合
- 会話型で、クライアントがメッセージのいずれかに応答して NAK を送る場合
- 非会話型で、トランザクションが正常終了前にホスト・アプリケーション によって終了した場合
- 会話型で、トランザクションが正常終了前にホスト・アプリケーション によって終了した場合

### 非会話型で、ACK を使用して正常終了まで実行する場合

次の例は、コミット・モード = 1、同期レベル = CONFIRM、および ACK (トランザクションは正常終了まで実行される) での非会話型フローを示しています。

```

CLIENT REQUEST          FLOW          IMS CONNECT REQUEST

SEND----->IRM/TRAN/DATA ----->RECEIVE
RECEIVE<-----DATA/CSM<-----SEND
SEND----->IRM/ACK----->RECEIVE

RECEIVE<-----RSM<-----SEND DEALLOCATE CONFIRM
RSM reason code = DEALLOCATE CONFIRM X'61' (97)
(97 = IMS Host application has committed the transaction)

```

### 会話型で、ACK を使用して正常終了まで実行する場合

次の例は、コミット・モード = 1、同期レベル = CONFIRM、および ACK (トランザクションは正常終了まで実行される) での会話型フローを示しています。

```

CLIENT REQUEST          FLOW          IMS CONNECT REQUEST

SEND----->IRM/TRAN/DATA ----->RECEIVE
RECEIVE<-----DATA/CSM<-----SEND
SEND----->IRM/ACK----->RECEIVE
SEND----->IRM/DATA----->RECEIVE
RECEIVE<-----DATA/CSM<-----SEND
SEND----->IRM/ACK----->RECEIVE
.
.
SEND----->IRM/DATA ----->RECEIVE
RECEIVE<-----DATA/CSM<-----SEND

SEND----->IRM/ACK----->RECEIVE

RECEIVE<-----RSM<-----SEND DEALLOCATE CONFIRM
RSM reason code = DEALLOCATE CONFIRM X'61' (97)
(97 = IMS Host application has committed the transaction)

```

## 非会話型で、クライアントがメッセージに回答して NAK を送る場合

次の例は、コミット・モード = 1、同期レベル = CONFIRM、および NAK (トランザクションはクライアント・アプリケーションからの NAK で終了する) での非会話型フローを示しています。

CLIENT REQUEST	FLOW	IMS CONNECT REQUEST
SEND----->	IRM/TRAN/DATA	----->RECEIVE
RECEIVE<-----	DATA/CSM<-----	-----SEND
SEND----->	IRM/NAK	----->RECEIVE
RECEIVE<-----		IMS MESSAGE "DFS555.."

## 会話型で、クライアントがメッセージのいずれかに回答して NAK を送る場合

次の例は、コミット・モード = 1、同期レベル = CONFIRM、および NAK (トランザクションはクライアント・アプリケーションからの NAK で終了する) での会話型フローを示しています。

CLIENT REQUEST	FLOW	IMS CONNECT REQUEST
SEND----->	IRM/TRAN/DATA	----->RECEIVE
RECEIVE<-----	DATA/CSM<-----	-----SEND
SEND----->	IRM/ACK	----->RECEIVE
SEND----->	IRM/DATA	----->RECEIVE
RECEIVE<-----	DATA/CSM<-----	-----SEND
SEND----->	IRM/ACK	----->RECEIVE
	:	
	:	
SEND----->	IRM/DATA	----->RECEIVE
RECEIVE<-----	DATA/CSM<-----	-----SEND
SEND----->	IRM/NAK	----->RECEIVE
RECEIVE<-----		IMS MESSAGE "DFS555.."

## 非会話型で、トランザクションが正常終了前にホスト・アプリケーションによって終了した場合

次の例は、コミット・モード = 1、同期レベル = CONFIRM、および ACK (トランザクションは正常終了する前にホスト・アプリケーションによって終了される) での非会話型フローを示しています。

CLIENT REQUEST	FLOW	IMS CONNECT REQUEST
SEND----->	IRM/TRAN/DATA	----->RECEIVE
RECEIVE<-----	Host Application abnormally terminates	-----SEND
RECEIVE<-----		IMS MESSAGE "DFS555.."

## 会話型で、トランザクションが正常終了前にホスト・アプリケーションによって終了した場合

次の例は、コミット・モード = 1、同期レベル = CONFIRM、および NAK (トランザクションはホスト・アプリケーションによって終了される) での会話型フローを示しています。

CLIENT REQUEST	FLOW	IMS CONNECT REQUEST
SEND----->	IRM/TRAN/DATA	----->RECEIVE
RECEIVE<-----	DATA/CSM<-----	-----SEND
SEND----->	IRM/ACK	----->RECEIVE
SEND----->	IRM/DATA	----->RECEIVE
RECEIVE<-----	DATA/CSM<-----	-----SEND
SEND----->	IRM/ACK	----->RECEIVE
	:	
	:	

```
SEND----->IRM/DATA----->RECEIVE
RECEIVE<-----DATA/CSM<-----SEND
SEND----->IRM/ACK----->RECEIVE
```

Host Application abnormally terminates

```
RECEIVE<----- IMS MESSAGE "DFS555.."
```

## IMS DFS メッセージおよび IMS コマンド出力に関する IMS Connect クライアント・メッセージの Protokol・シーケンス

350 ページの表 66 と 350 ページの表 67 は、いろいろな IMS DFSnnnnn メッセージまたは IMS コマンド出力が IMS Connect クライアントに送られた場合に取られる必要なアクションを示しています。これらの 2 つの表は、同期レベル CONFIRM と同期レベル NONE、およびコミット・モード 0 (CM0) とコミット・モード 1 (CM1) の両方について、クライアントが IMS DFS メッセージまたは出力を IMS コマンドから受け取った時に ACK を送信する必要があるかどうかを示すものです。

注：クライアント・コードは、CSM\_FLG1 バイトに CSM\_ACK\_NAK フラグがあるかどうかを検査します。また、RSM\_FLG1 バイトに RSM\_ACK\_NAK フラグがあるかどうかを検査します。このテストは、ACK または NAK が必要かどうかを判別するために実施されます。必要でない場合は、350 ページの表 66 および 350 ページの表 67 に概説された分析が行われます。

350 ページの表 66 と 350 ページの表 67 は、また、クライアントが、IMS Connect から「割り振り解除の打ち切り」応答 (RSM) を受け取るために READ を出す必要があるかどうかを示しています。両方の表の注は、350 ページの表 67 の直後にまとめてあります。

表 66. 持続ソケット: IMS DFS メッセージおよび IMS コマンド出力に対するクライアント・メッセージ・Protokol・シーケンス

クライアントへのメッセージ出力	CM1、同期レベル Confirm (確認)	CM1、同期レベル None (なし)	CM0、同期レベル Confirm (確認)	CM0、同期レベル None (なし)
トランザクション・コードが無効 DFS064	DFS064 <sup>1</sup>	DFS064 <sup>1</sup>	N/A	N/A
トランザクションが停止 DFS065	DFS065 <sup>1</sup>	DFS065 <sup>1</sup>	N/A	N/A
トランザクションが異常終了 DFS555	DFS555 <sup>7</sup>	DFS555 <sup>1</sup>	N/A	N/A
出力 DFS2082	DFS2082 <sup>2</sup>	DFS2082 <sup>1</sup>	N/A	N/A
IMS コマンド出力	コマンド出力 <sup>1</sup>	コマンド出力 <sup>1</sup>	N/A	N/A
セキュリティ障害 DFS1292	DFS1292 <sup>1</sup>	DFS1292 <sup>1</sup>	N/A	N/A
セグメントが 32 K より大	DFS1294 <sup>5</sup>	DFS1294 <sup>5</sup>	N/A	N/A

表 67. トランザクション・ソケット: IMS DFS メッセージおよび IMS コマンド出力に対するクライアント・メッセージ・Protokol・シーケンス

クライアントへのメッセージ出力	CM1、同期レベル Confirm (確認)	CM1、同期レベル None (なし)	CM0、同期レベル Confirm (確認)	CM0、同期レベル None (なし)
トランザクション・コードが無効 DFS064	DFS064 <sup>1</sup>	DFS064 <sup>1</sup>	DFS064 <sup>1</sup>	N/A

表 67. トランザクション・ソケット: IMS DFS メッセージおよび IMS コマンド出力に対するクライアント・メッセージ・プロトコル・シーケンス (続き)

クライアントへのメッセージ出力	CM1、同期レベル Confirm (確認)	CM1、同期レベル None (なし)	CM0、同期レベル Confirm (確認)	CM0、同期レベル None (なし)
トランザクションが停止 DFS065	DFS065 <sup>1</sup>	DFS065 <sup>1</sup>	DFS065 <sup>1</sup>	N/A
トランザクションが異常終了 DFS555	DFS555 <sup>7</sup>	DFS555 <sup>1</sup>	DFS555 <sup>7</sup>	N/A
出力 DFS2082	DFS2082 <sup>2</sup>	DFS2082 <sup>1</sup>	出力なし <sup>3</sup>	N/A
IMS コマンド出力	コマンド出力 <sup>1</sup>	コマンド出力 <sup>1</sup>	コマンド出力 <sup>4</sup>	N/A
セキュリティ障害 DFS1292	DFS1292 <sup>1</sup>	DFS1292 <sup>1</sup>	DFS1292 <sup>1</sup>	N/A
セグメントが 32 K より大	DFS1294 <sup>5</sup>	DFS1294 <sup>5</sup>	DFS1297 <sup>6</sup>	N/A

注:

- DFS メッセージに対する ACK は必要ない。
- DFS メッセージに対する ACK と、割り振り解除応答を入手するための 2 番目の READ の両方が必要。
- トランザクション出力を受信するための READ はタイムアウトになる。データは何も受信されない。OTMA は、コミット・モード 0、同期レベル CONFIRM を非同期出力として処理する。IMS ホスト・アプリケーションがメッセージ (IOPCB への ISRT) を返さない場合は、OTMA は割り振り解除を送信しない。切断の完了は、IMS Connect 構成ファイルに指定された TIMEOUT= 値が満了した後でなければならない。
- コマンド出力に対する ACK が必要。割り振り解除応答を入手するための 2 番目の READ は必要ない。このコマンド出力は、非同期出力として処理される。
- DFS1294 出力に対する ACK は必要ない。DFS555 メッセージを受信するために 2 番目の RECEIVE が必要。
- クライアントは、DFS1294 ではなく DFS1297 を受信する。DFS1294 メッセージは ACK を必要としない。DFS555 メッセージは送られず、したがって、2 番目の RECEIVE は必要ない。アプリケーションはコミットされるが、アプリケーション出力は、セグメントが 32 K よりも大きいため、廃棄される。
- DFS メッセージに対する ACK は必要ない。

### コミット・モード= 1、同期レベル= CONFIRM の理由コード

CM1 では、ゼロ (0) の戻りコードに関して 3 つの理由コードがあり、また、戻りコード X'04' には 2 つの理由コードがあり、クライアント・アプリケーションに情報を提供します。これらのコードの使われ方については、サンプルのフローに示してあります。コードの意味は、以下の表にリストされています。

表 68. CM1、同期レベル CONFIRM の通知理由コード

戻りコード	理由コード	説明
X'00'	94	応答 - 非会話型でのホストからの出力のみ
X'00'	95	会話 - 会話中のホストからの最後の出力
X'00'	96	会話 / 応答 - 会話中
X'04'	97	割り振り解除コミット - ホスト・アプリケーションの正常終了
X'04'	98	割り振り解除の打ち切り - ホスト・アプリケーションの異常終了



## IMS Connect デッド・レター・キュー (HWS\$DLQ)

一部のインスタンスでは、OTMA が IMS Connect からの NAK 応答を受信すると、OTMA は未配布メッセージを IMS Connect のデッド・レター・キューに保管します。IMS Connect のデッド・レター・キューは、T パイプ名 HWS\$DLQ によって識別されます。

OTMA が HWS\$DLQ にメッセージを保管するインスタンスを以下に示します。

- IMS Connect が、OTMA ヘッダーのユーザー・データ・セクションが欠落したメッセージに対して、OTMA に NAK 応答を返す場合。この場合、IMS はメッセージ HWS\$D0255W を出します。
- IMS Connect が非同期コールアウト要求を処理できなかったために、IMS Connect が OTMA に NAK 応答を返す場合。この場合、IMS はメッセージ HWS\$P1510E を出します。

メッセージのユーザー・データ・セクションが欠落している場合を除いては、RESUME TPIPE 呼び出しで代替クライアント ID として HWS\$DLQ を指定することで、HWS\$DLQ のメッセージをリトリブできます。HWS\$DLQ に OTMA ヘッダーのユーザー・データ・セクションが欠落したメッセージが存在する場合、キューにある後続のメッセージをリトリブするには、このメッセージをデキューする必要があります。

メッセージを HWS\$DLQ からデキューするには、次のようにします。

1. コマンド `/STOP TMEMBER tmembername TPIPE hws$dLq` を発行して、HWS\$DLQ を停止します。
2. コマンド `/DEQUEUE TMEMBER tmembername TPIPE hws$dLq PURGE1` を発行して、メッセージをデキューします。
3. コマンド `/START TMEMBER tmembername TPIPE hws$dLq` を発行して、HWS\$DLQ を始動します。

HWS\$DLQ のキュー・カウントを表示するには、コマンド `/DISPLAY TMEMBER tmembername TPIPE hws$dLq` を発行します。

## IMS Connect の ping サポート

IMS Connect が使用可能かどうかを判別するために、ping 要求を IMS Connect に送信できます。ping サポートは、トランザクションのように作動し、見た目はトランザクション・コードとデータです。要求 PING IMS\_CONNECT を送信すると、PING RESPONSE が応答されます。

ユーザー・メッセージ出口 HWSJAVA、HWSSMPL1、HWSSMPL0、および HWSSOAP1 は ping サポートを提供します。ユーザー・メッセージ出口 HWSCSLO0 および HWSCSLO1 は ping 機能をサポートしません。ユーザー独自のメッセージ出口を作成する場合、その出口に ping 機能サポートを追加するように選択できます。

ping 要求の通信シーケンスは、4 つのステップで構成されます。HWSSOAP1 ユーザー・メッセージ出口を使用する場合、3 番目と 4 番目のステップである受信と切断ステップが、HWSJAVA0、HWSSMPL0、および HWSSMPL1 ユーザー・メッセージ出口のステップとは異なります。

HWSJAVA0、HWSSMPL0、および HWSSMPL1 ユーザー・メッセージ出口の場合の ping 要求の通信シーケンスを以下に示します。

1. 接続する。
2. PING IMS\_CONNECT を送信する (大文字で送信する必要がある)。
3. 受信する。
  - HWSSMPL0 または HWSSMPL1 を使用している場合、ユーザー作成のクライアント・アプリケーションは、HWSC0030I \*PING RESPONSE\* \*CSMOKY\* を受信します。
  - HWSJAVA0 を使用している場合、IMS TM Resource Adapter のアプリケーションは、OTMA ヘッダーのアプリケーション・データ部分に HWSC0030I \*PING RESPONSE\* を受信し、OTMA ヘッダーのユーザー・データ部分に戻りコード 48 と理由コード ICONSUCC を受信します。
4. ソケット接続を解決する。

IMS Connect は、以下の状況のときを除いて、ソケット接続を維持します。

- ソケットが持続ソケットではない場合、IMSConnect はソケットを切断します。



- IMS Connect がクライアント・アプリケーションからの ACK または NAK 応答を想定している場合、IMS Connect はソケットを切断します。
- クライアント・アプリケーションが IMS の会話方式である場合、IMS Connect はソケットを切断します。

HWSSOAP1 ユーザー・メッセージ出口の場合の ping 要求の通信シーケンスを以下に示します。

1. 接続する。
2. PING IMS\_CONNECT を送信する (大文字で送信する必要がある)。
3. PING\_RESPONSE を受信する。
4. 切断する。

#### **関連資料**

[255 ページの『PING 応答のフォーマット』](#)

PING IMS\_CONNECT (ping 要求) を IMS Connect に送信すると、PING 応答が IMS Connect クライアントに送信されます。

[IMS TM Resource Adapter ユーザー・メッセージ出口ルーチン \(HWSJAVA0\) \(出口ルーチン\)](#)

[ユーザー・メッセージ出口ルーチン HWSSMPL0 および HWSSMPL1 \(出口ルーチン\)](#)

[SOAP Gateway 出口ルーチン \(HWSSOAP1\) \(出口ルーチン\)](#)



## 第 19 章 IMS Connect 2 フェーズ・コミット・サポート

IMS Connect は、IMS TM トランザクションと IMS DB データベース要求の両方に対して、2 フェーズ・コミットをサポートします。

2 フェーズ・コミットの参加プログラムである IMS TM トランザクションおよび IMS DB データベース要求は、z/OS リソース・リカバリー・サービスまたは外部コーディネーター (例えば、IBM WebSphere Application Server) によって調整されます。

IMS Connect を介して IMS TM にアクセスする場合、外部コーディネーターは IMS TM Resource Adapter をリソース・アダプターとして使用しなければなりません。IMS TM Resource Adapter および IMS Connect は、共同して 2 フェーズ・コミット処理のデータ・フローを処理します。

IMS Connect を介して IMS DB にアクセスする場合、外部コーディネーターは、以下のリソース・アダプターおよび API のいずれも使用できます。

- IMS Universal Database リソース・アダプター
- IMS Universal JDBC ドライバー
- IMS Universal DL/I ドライバー
- 分散リレーショナル・データベース体系 (DRDA)

IMS DB サポートに応じて、2 フェーズ・コミット処理のデータ・フローを処理するために、IMS Connect は、CSL Open Database Manager (ODBM) コンポーネントおよび IMS Universal ドライバー を処理します。

このトピックでは、2 フェーズ・コミットの概要と、IMS Connect がサポートするいくつかの主要なシナリオを説明します。

### 2 フェーズ・コミット・プロトコルの概要

2 フェーズ・コミット・プロトコルは、複数のデータベースに関連するトランザクションが同期の取れていない更新を行わないように保証する、一組のアクションから構成されます。

2 フェーズ・コミットを使用すると、複数の異なるデータ・ソース上にある一連のデータベース対話を単一のトランザクションとしてグループ化し、完了またはロールバックすることができます。複数のデータ・ソース上にある一連のデータベース対話を表す 2 フェーズ・コミット・トランザクションは、グローバル・トランザクションと呼ばれます。

グローバル・トランザクションの開始時に、グローバル・トランザクション ID (XID) が生成されます。外部トランザクション・マネージャーはこの ID を使用して、関連するすべてのリソース・マネージャー全体に渡って 2 フェーズ・コミット処理を進めます。グローバル・トランザクションの範囲内の各データベース対話は、それに関連するリソース・マネージャーで実行されます。その後、対話の結果は、処理用にアプリケーションに戻されます。グローバル・トランザクションの範囲内のデータベース対話が完了すると、グローバル・トランザクションがアクセスした各リソース・マネージャーに準備呼び出しが送信されます。準備呼び出しによって、各リソース・マネージャーは、グローバル・トランザクションの一部として実行した作業をコミットできるかを判別し、それに関して報告する機会が得られます。各リソース・マネージャーがその作業をコミットできるという確認を受け取ると、トランザクション・マネージャーは各リソース・マネージャーにコミット呼び出しを送信します。

コミット呼び出しを送信する前であれば、任意の時点で、2 フェーズ・コミット・トランザクションをロールバックすることができます。トランザクションがロールバックされる場合、そのトランザクションに関連する各リソース・マネージャーにロールバック呼び出しが送られて、一時的な変更が除去または廃棄されます。

コミット・フェーズ中に何らかのデータベース障害が発生した場合、外部トランザクション・マネージャーは、失敗したリソース・マネージャーとの接続の再確立を試行し、リソース・マネージャーがコミットするための呼び出しを再開します。

コミット・フェーズでトランザクション・マネージャーに障害が発生すると、トランザクション・マネージャーは再始動時にリカバリー処理を実行し、関連するすべてのリソース・マネージャーとの接続の再確

立を試行します。トランザクション・マネージャーが接続を再確立すると、リソース・マネージャーがコミットするための呼び出しを再開します。

## 分散 2 フェーズ・コミット・サポート

分散 2 フェーズ・コミット・プロトコルは、TCP/IP を使用して、さまざまなプラットフォーム (例えば、Windows、AIX®、Solaris、Linux® など) 間のトランザクションと通信します。

分散 TCP/IP トランザクションは、通常、以下のコンポーネントと関係します。

- アプリケーション・コンポーネント
- WebSphere Application Server などのアプリケーション・サーバー
- IMS Universal Database リソース・アダプター、または IMS TM Resource Adapter の 1 つのようなリソース・アダプター
- リソース・マネージャー
- IMS TM などのトランザクション・マネージャー
- IMS DB などのエンタープライズ情報システム (EIS)

分散 2 フェーズ・コミット・プロトコルでは、アプリケーション・サーバーにデプロイされるクライアントは、トランザクションを発行します。アプリケーション・サーバーは、外部トランザクション・マネージャー (外部コーディネーター) としての役割を果たして、1 つ以上のリソース・マネージャーにまたがるトランザクションを管理します。エンタープライズ情報システムのリソース・マネージャーにアクセスするには、外部コーディネーターはリソース・アダプターを使用しなければなりません。IMS Connect クライアントは、TCP/IP を使用して、IMS Connect を介してリソース・マネージャー (IMS) にアクセスします。

IMS は、z/OS リソース・リカバリー・サービス (RRS) を介して X/Open XA プロトコルをサポートします。IMS はこれを使用して、z/OS 上の 2 フェーズ・コミット処理に参加します。その結果、IMS Connect は RRS と通信し、RRS に IMS Connect クライアントからのトランザクション・コンテキストを渡します。次に、同期点コーディネーターとしての RRS は、IMS に対してすべての更新が行われるか、またはまったく更新が行われないように、変更を調整します。RRS では、トランザクション有効範囲内で行われるかまたは行われない一連の変更はリカバリー単位 (UR) と呼ばれます。

IMS Connect は、2 フェーズ・コミット処理では二重の役割を果たします。IMS Connect は、RRS の拡張としての役割を果たし (同期点マネージャー)、サーバー分散同期点リソース・マネージャー (SDSRM) と見なされます。SDSRM として、IMS Connect は、RRS が必要に応じて他の同期点マネージャーと通信して、アプリケーションがアクセスする分散リソースを確実に調整できるようにします。また、IMS Connect は、通信リソース・マネージャー (CRM) でもあります。CRM として、IMS Connect は、アプリケーション・コンポーネントが他のアプリケーション・コンポーネントおよび別のシステム上にあるリソース・マネージャーと通信するのを可能にすることにより、分散リソースへのアクセスを制御します。また、CRM として、IMS Connect は、同期点イベントの処理を支援し、そのイベントを分散同期点マネージャーに伝達します。

IMS 分散トランザクション処理は、2 つのタイプのトランザクション、すなわち、2 フェーズ・コミット・プロトコルを使用するグローバル・トランザクションと、1 フェーズ・コミット最適化を使用するトランザクションに分けることができます。1 フェーズ・コミット最適化は、トランザクションに 1 つのリソース・マネージャーしか含まれていない場合に使用されます。

## IMS Universal ドライバー のサポート

IMS Connect は、IMS Universal ドライバー から受け取る IMS DB データベース要求について 2 フェーズ・コミットおよび 1 フェーズ・コミットをサポートします。

## IMS Universal ドライバー を使用するグローバル (XA) トランザクション

IMS Universal ドライバー 用のグローバル 2 フェーズ・コミット・トランザクションに対する IMS サポートは、z/OS リソース・リカバリー・サービス (RRS) マルチシステム・カスケード・トランザクションを使用して、IMSplex 内で IMS Connect、CSL Open Database Manager (ODBM)、および IMS 間のデータベース・アクセス要求を調整します。

IMS Universal ドライバー は、IMS Connect との通信に、LocalTransaction インターフェースと XAResources インターフェースを使用します。IMS Connect は、XID を保持し、それを作業コンテキスト・トークンおよび親リカバリー単位 (UR) に関連付けます。これは、IMS Universal ドライバー によってサブミットされたトランザクションを表しています。

IMS Connect は、2 フェーズ・コミットを必要とするいずれかの IMS Universal ドライバー からデータベース・アクセス要求を受け取ると、RRS を呼び出し、マルチシステム・カスケード・トランザクションのコーディネーター となります。IMS Connect は、親リカバリー単位 (UR) を作成し、CSL Open Database Manager (ODBM) に対するデータベース・アクセス要求で親 UR トークンを渡します。

ODBM は、マルチシステム・カスケード・トランザクションに従属 となり、IMS Connect から受け取った親 UR トークンを使用してカスケード子 UR を作成します。

クライアント・アプリケーションがコミットの準備を要求すると、IMS Universal ドライバー が IMS Connect に準備シグナルを送信します。次に、IMS Connect が RRS に準備要求を発行し、ODBM が 2 フェーズ・コミット・プロセスのフェーズ 1 を完了するのを待ちます。IMS リソース・マネージャーがコミットする準備ができている場合、ODBM は、確認をコミットする準備を RRS に返し、RRS は、結果を IMS Connect に送ります。次に、IMS Connect は、準備コマンドの結果を、IMS Universal ドライバー を通してトランザクション・マネージャーに送り返します。この時点で、IMS Connect および ODBM によって維持されている UR は、外部トランザクション・マネージャーからコミット・コマンドまたはロールバック・コマンドが送り込まれるまで未確定状態となります。

グローバル・トランザクションに関連付けられたすべてのリソース・マネージャーでコミット可能な場合、トランザクション・マネージャーは、コミット決定を確定し、IMS Universal ドライバー に変更をコミットさせます。IMS Universal ドライバー はコミット・シグナルを IMS Connect に送り、IMS Connect は全体の決定がすべてのリソースのコミットであることを RRS に知らせます。RRS は、変更をコミットするように IMS に依頼します。IMS が変更をコミットした後、RRS は、ローカル・リソースがコミットされたという情報を IMS Connect に戻します。IMS Connect は、ログ・レコードを削除するように RRS に依頼します。

以下の図は、IMS Universal ドライバー から IMS Connect にサブミットされる分散 2 フェーズ・コミット・グローバル・トランザクションのフローの最初の 17 ステップを示しています。トランザクションは、1 つの IMS システムに関係します。RRS は、各 z/OS イメージ上でアクティブであることが必要です。しかし、IMS Connect は、ODBM および IMS と同じ z/OS イメージ上に存在する必要はありません。

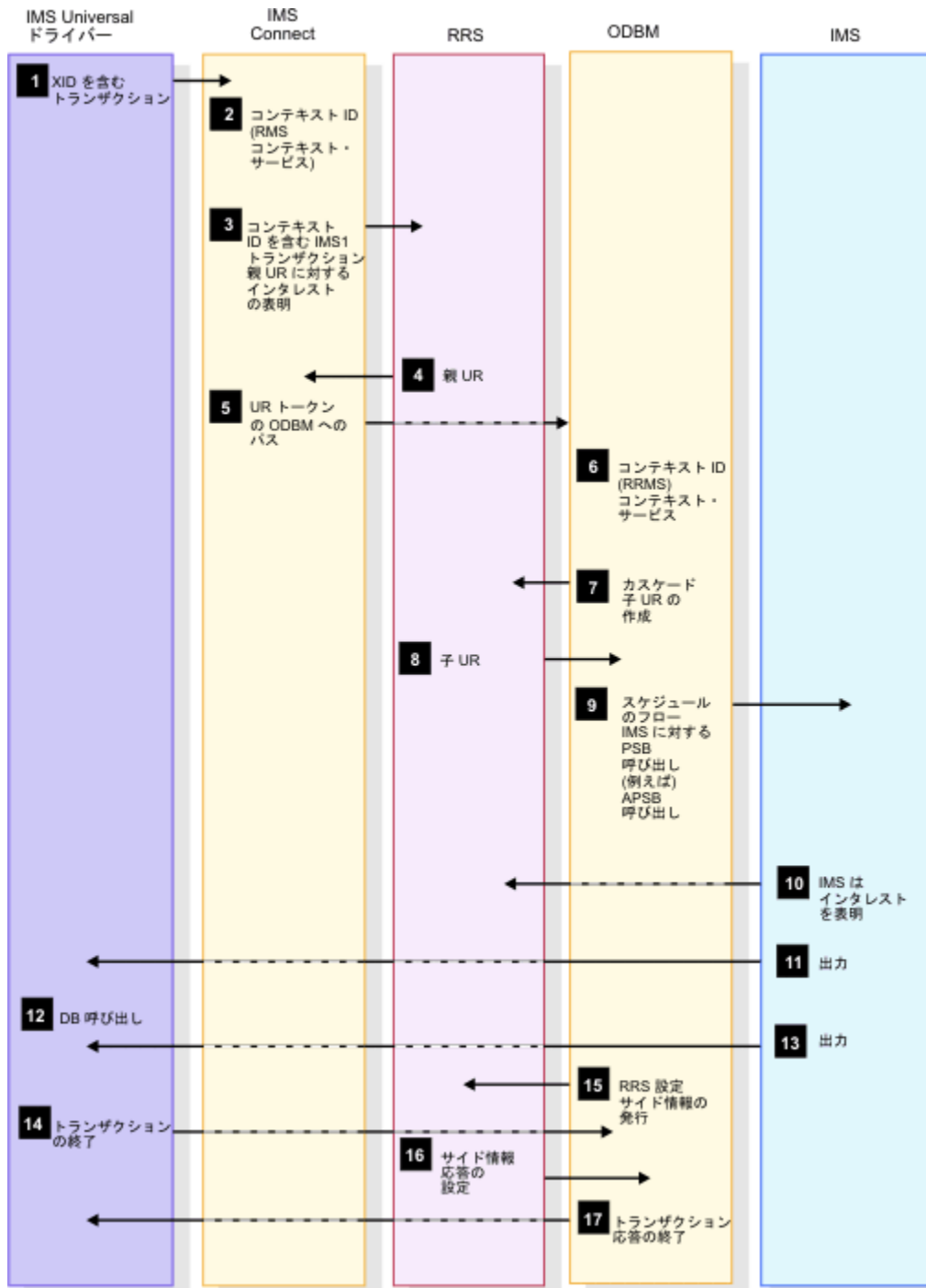


図 34. IMS Universal ドライバー に関する分散 2 フェーズ・コミット・グローバル・トランザクション・クライアントのフロー (1/2)

以下の図は、前の図に示した分散 2 フェーズ・コミット・グローバル・トランザクションのフローの残りのステップを示しています。

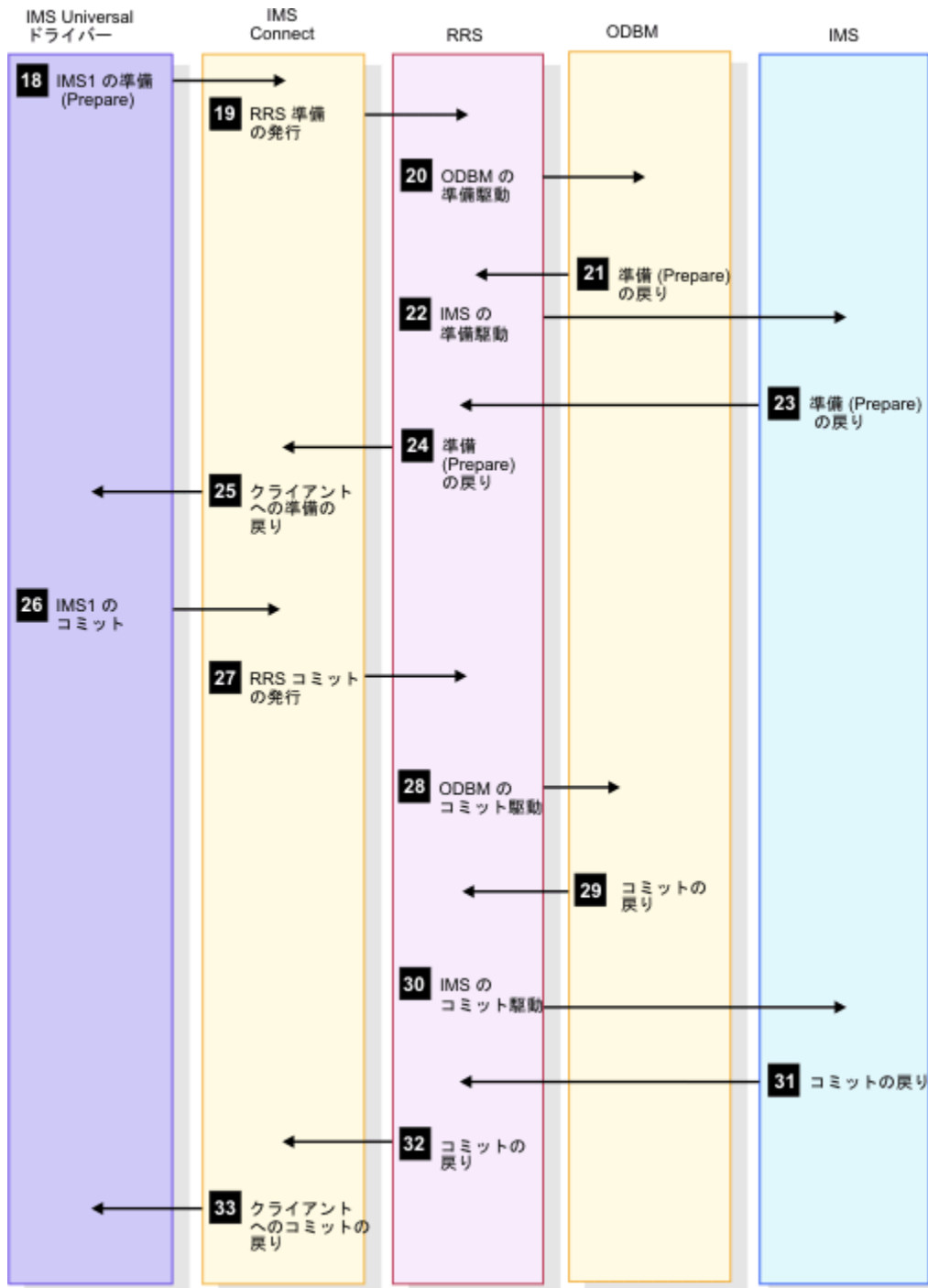


図 35. IMS Universal ドライバー に関する分散 2 フェーズ・コミット・グローバル・トランザクション・クライアントのフロー (2/2)

## IMS Universal ドライバー を使用する 1 フェーズ・コミット・グローバル・トランザクション

外部トランザクション・マネージャーが、指定されたトランザクションにただ 1 つのリソース・マネージャーが登録されていることを検出した場合、トランザクション・マネージャーは 1 フェーズ・コミット最適化を実行できます。

1 フェーズ・コミット最適化では、2 フェーズ・コミット操作の最初のフェーズである準備フェーズが省略されており、トランザクション・マネージャーは、最初に「コミット準備」なしで「コミット要求」のみをリソース・マネージャーに送信して変更をコミットします。

また、1 フェーズ・コミットを使用する z/OS リソース・リカバリー・サービス (RRS) トランザクションは、RRS ローカル・トランザクションとも呼ばれます。ローカル・トランザクションには、XID は含まれません。

IMS Connect が IMS Universal ドライバー から RRS ローカル・トランザクションとしてデータベース・アクセス要求を受け取った場合、IMS Connect は、RRS を呼び出さず、そのローカル・トランザクションを直接 ODBM に渡します。次に、ODBM は RRS を呼び出し、データベース要求のコミットまたはロールバックを調整します。

以下の図は、分散 1 フェーズ・コミット・グローバル・トランザクションのフローを示しています。RRS は、各 z/OS イメージ上でアクティブであることが必要です。しかし、IMS Connect は、ODBM および IMS と同じ z/OS イメージ上に存在する必要はありません。

以下の図は、IMS Universal ドライバー から IMS Connect にサブミットされる分散 1 フェーズ・コミット・グローバル・トランザクションのフローの最初の 17 ステップを示しています。トランザクションは、1 つの IMS システムに関係します。RRS は、すべての z/OS イメージ上でアクティブであることが必要です。しかし、IMS Connect は、ODBM および IMS と同じ z/OS イメージ上に存在する必要はありません。



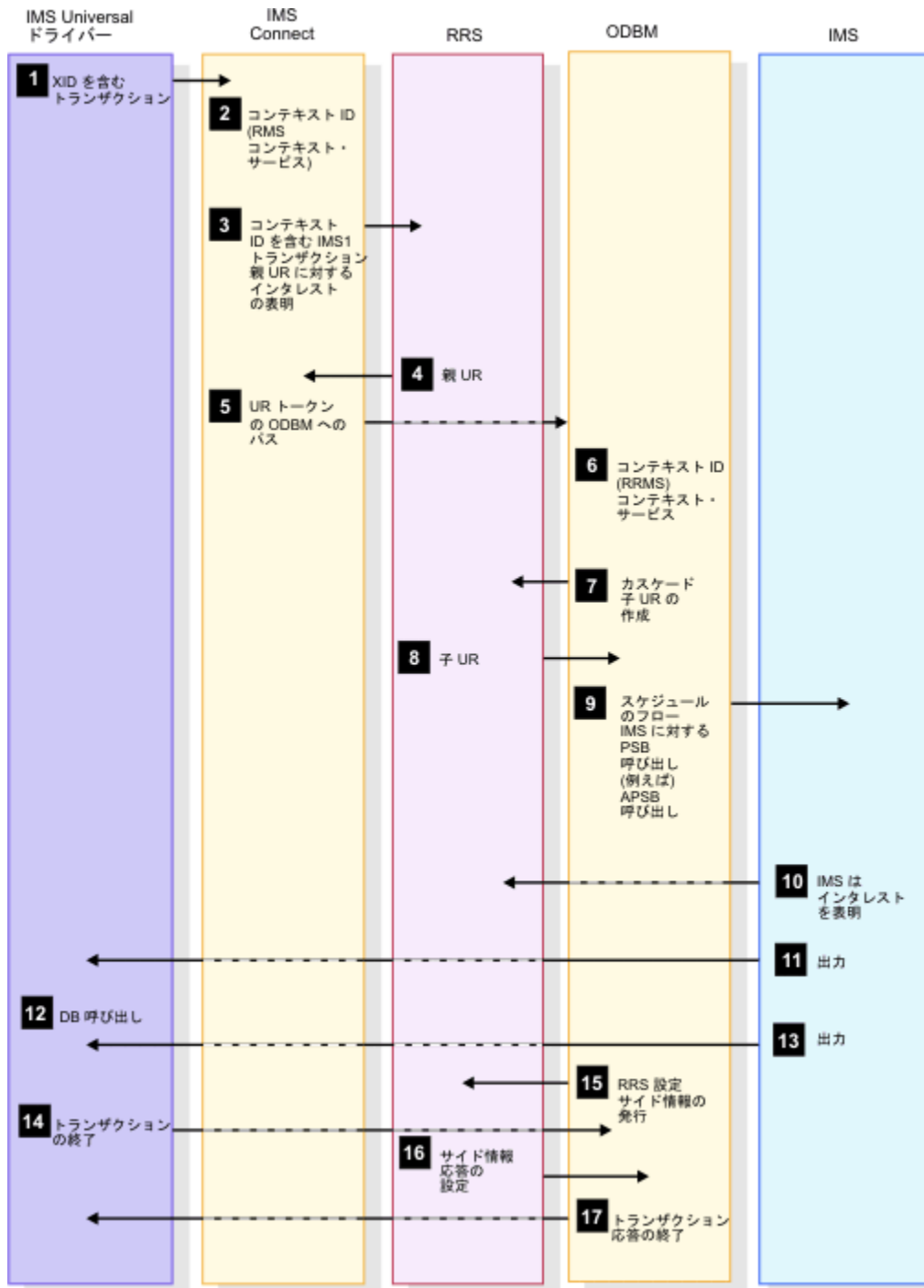


図 36. IMS Universal ドライバー に関する分散 1 フェーズ・コミット・グローバル・トランザクション・クライアントのフロー (1/2)

以下の図は、前の図に示した分散 1 フェーズ・コミット・グローバル・トランザクションのフローの残りのステップを示しています。

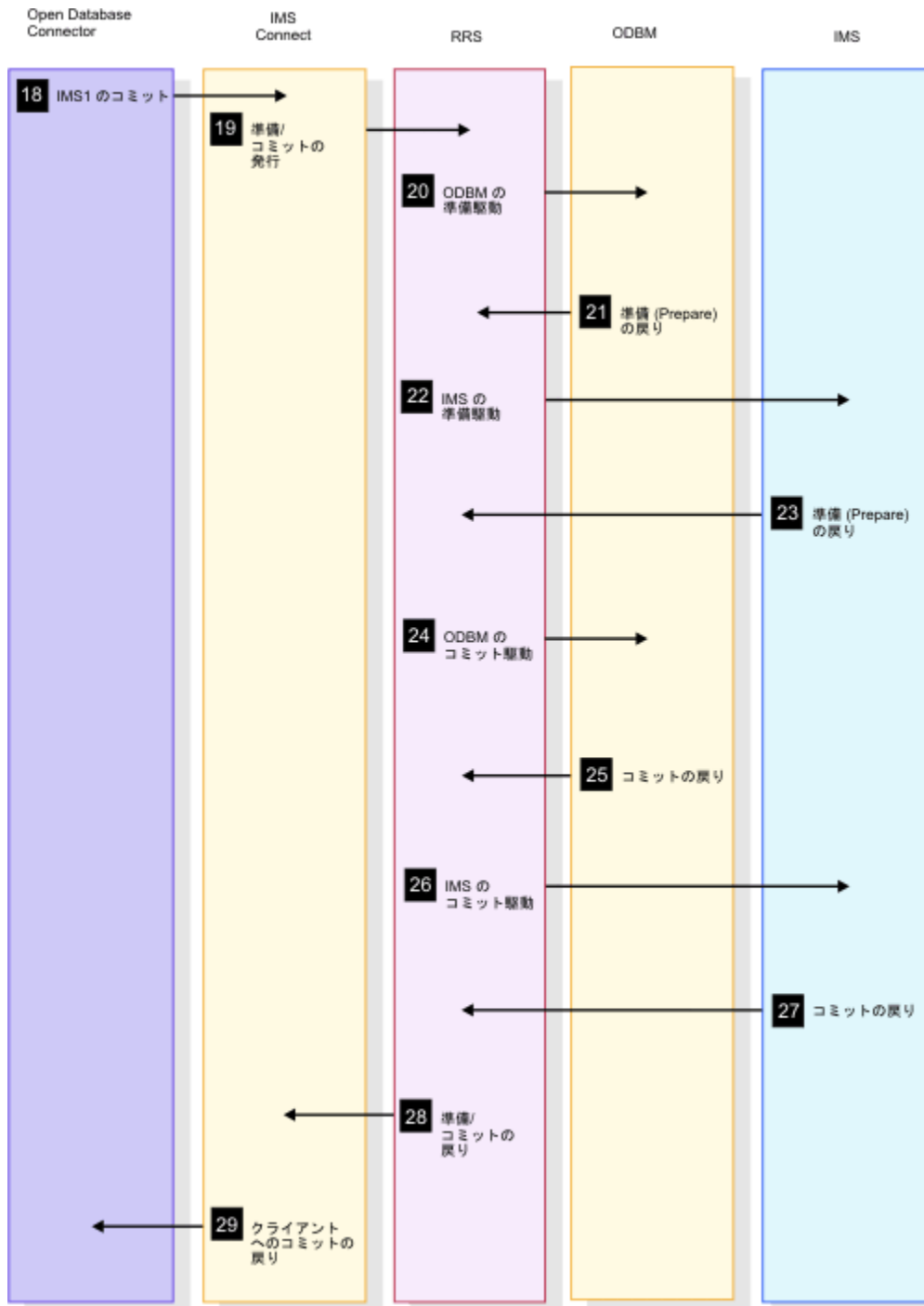


図 37. IMS Universal ドライバー に関する分散 1 フェーズ・コミット・グローバル・トランザクション・クライアントのフロー (2/2)

## IMS TM Resource Adapter のサポート

IMS Connect は、IMS TM Resource Adapter から受け取る IMS TM トランザクションについて 2 フェーズ・コミットおよび 1 フェーズ・コミットをサポートします。

### IMS TM Resource Adapter を使用するグローバル (XA) トランザクション

グローバル (XA) トランザクションは、リソース・マネージャーに対する外部トランザクション・マネージャー (外部コーディネーター) によって制御および調整が行われます。

トランザクションは、通常、別々のプラットフォーム上にある複数のリソース・マネージャーをまたがる調整を必要とします。トランザクションは、SYNCPPOINT の同期レベルを指定した送信後コミット (コミット・レベル 1) として IMS Connect に送信する必要があります。

エンタープライズ情報システムをアクセスするために、外部コーディネーターは、X/Open XA 標準で定義されている XID をリソース・アダプターに送信します。長さおよび FormatID のフィールドの他に、XID には、グローバル・トランザクション ID (GTRID) と分岐クォリファイヤー (BQUAL) の 2 つの部分があります。IMS は X/Open XA プロトコルをサポートしないため、IMS TM Resource Adapter は、LocalTransaction および XAResources インターフェースを使用して、IMS Connect と通信するために外部コーディネーターによって調整されたトランザクションを扱います。IMS Connect は、XID を保持し、それを作業コンテキスト・トークンおよび IMS 名に関連付けます。次に、IMS Connect はコンテキスト・トークンを z/OS リソース・リカバリー・サービス (RRS) に渡します。

IMS Connect は、トランザクション出力を IMS TM Resource Adapter に返し、そこから出力データがクライアントに返されます。IMS TM Resource Adapter への出力メッセージの送信が正常に終了すると、IMS Connect は ACK を IMS に送ってメッセージに回答します。IMS への要求後、アプリケーション・コンポーネントは IMS TM Resource Adapter に、変更をコミットする準備ができたことを知らせます。この時点で、IMS TM Resource Adapter は、準備シグナルを IMS Connect に送ります。IMS Connect は、次に、RRS に準備フェーズを開始するように依頼します。IMS リソース・マネージャーがコミットする準備ができている場合、RRS は、リソース・マネージャーから確認をコミットする準備を受け取り、結果を IMS Connect に送ります。次に、IMS Connect は「コミット要求」シグナルを IMS TM Resource Adapter に送って、変更をコミットするように要求します。

IMS TM Resource Adapter は、「コミット要求」シグナルを受け取ると、IMS システム上のリソースをコミットできることを外部コーディネーターに知らせます。トランザクション・マネージャーは、全体の結果を判別します。すべてのリソース・マネージャーでコミット可能な場合、トランザクション・マネージャーは、コミット決定を確定し、IMS TM Resource Adapter に変更をコミットさせます。IMS TM Resource Adapter はコミット・シグナルを IMS Connect に送り、IMS Connect は全体の決定がすべてのリソースのコミットであることを RRS に知らせます。RRS は、変更をコミットするように IMS に依頼します。IMS が変更をコミットした後、RRS は、ローカル・リソースがコミットされたという情報を IMS Connect に戻します。IMS Connect は、ログ・レコードを削除するように RRS に依頼します。

IMS Connect システム構成または IMS Connect データ・ストア定義のいずれかで CASCADE=Y を指定することによってグローバル・トランザクションのカスケードに対するサポートが使用可能になっている場合、IMS Connect と IMS は、それぞれ異なる z/OS イメージ (LPAR) 上で稼働しているときにグローバル・トランザクションを処理することができます。グローバル・トランザクションのカスケードに対するサポートが使用不可である場合、IMS Connect、RRS、および IMS すべてが同じ LPAR 上になければなりません。

**要件 :** 同じグローバル・トランザクションから IMS にサブミットされた複数の別個のトランザクションを含むことができるグローバル・トランザクションのすべてのメッセージ・フローは、同じ IMS と IMS Connect のペアによって処理される必要があります。この要件の理由から、z/OS Sysplex Distributor など IMS Connect の複数のインスタンス間でワークロードを分散するソフトウェアを使用する場合、グローバル・トランザクションを IMS に送信できない可能性があります。そのようなワークロード分散ソフトウェアは、同じグローバル・トランザクションからのすべてのフローが同じ IMS および IMS Connect のペアに経路指定されることを保証しない可能性があります。

以下の一連の図は、異なる構成シナリオの IMS TM Resource Adapter と IMS との間での分散 2 フェーズ・コミット・グローバル・トランザクションのフローを示しています。

以下の図は、IMS TM Resource Adapter と単一の IMS システムとの間の単一 2 フェーズ・コミット・トランザクションのフローを示しています。図では、IMS Connect と IMS が同じ LPAR 上で実行されています。

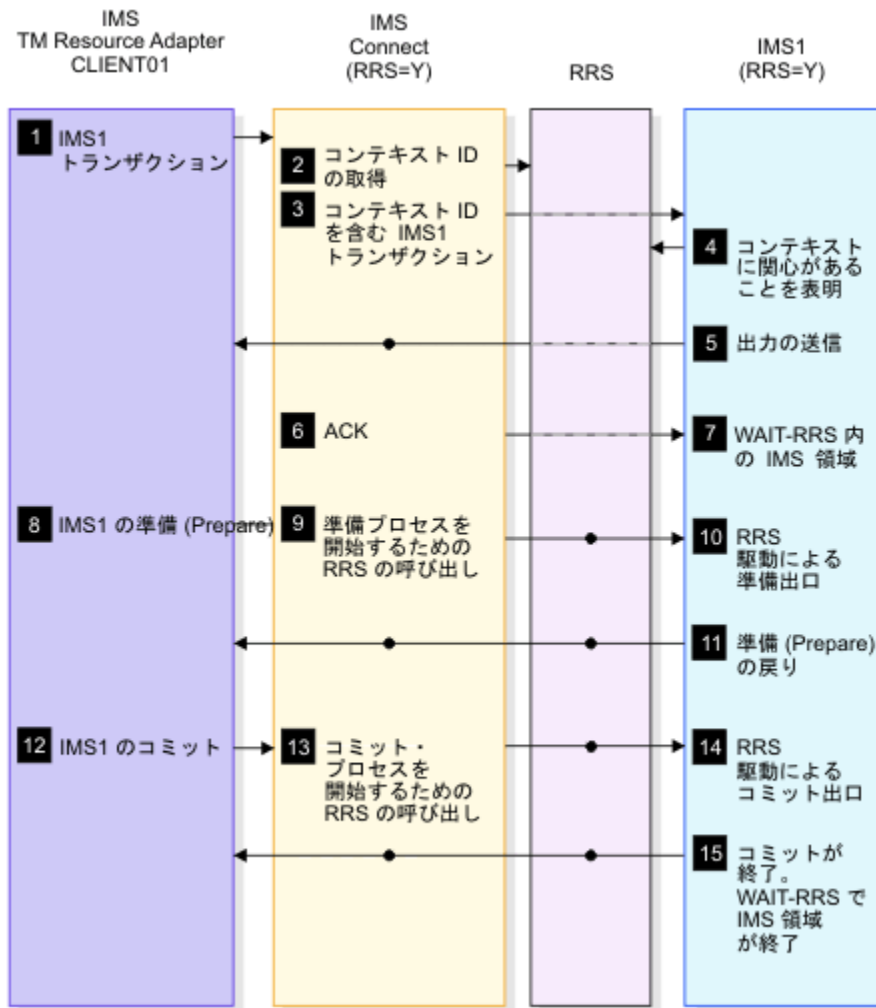


図 38. 単一の分散 2 フェーズ・コミット・グローバル・トランザクションのフロー

以下の図は、IMS TM Resource Adapter と単一の IMS システムとの間の単一 2 フェーズ・コミット・トランザクションのフローを示しています。図では、IMS Connect と IMS が異なる LPAR 上で実行しています。

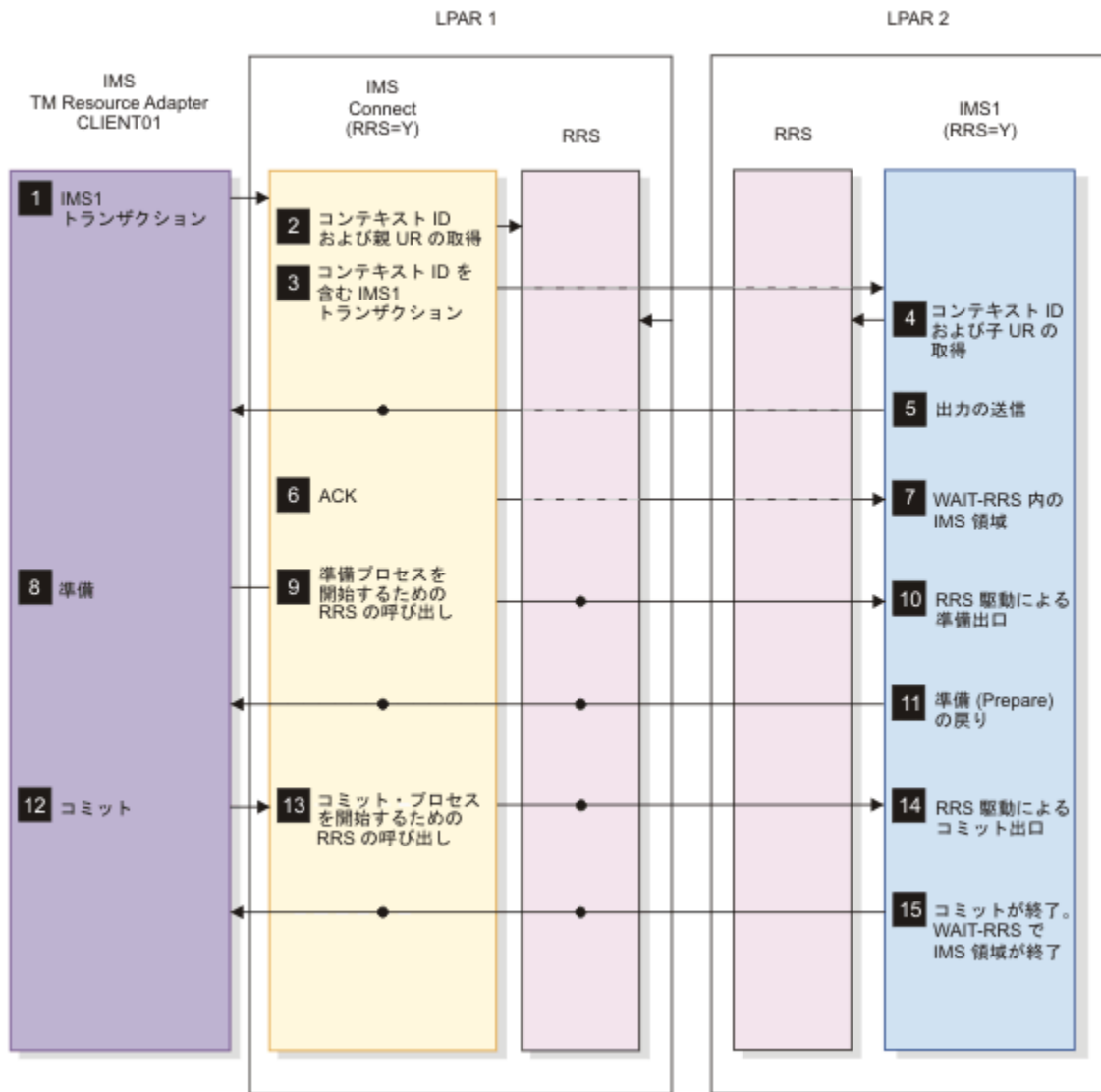


図 39. カスケードされた単一分散 2 フェーズ・コミット・グローバル・トランザクションのフロー

以下の図は、分散 2 フェーズ・コミット・グローバル・トランザクションのフローを示します。トランザクションは、2つの IMS システムに関係します。

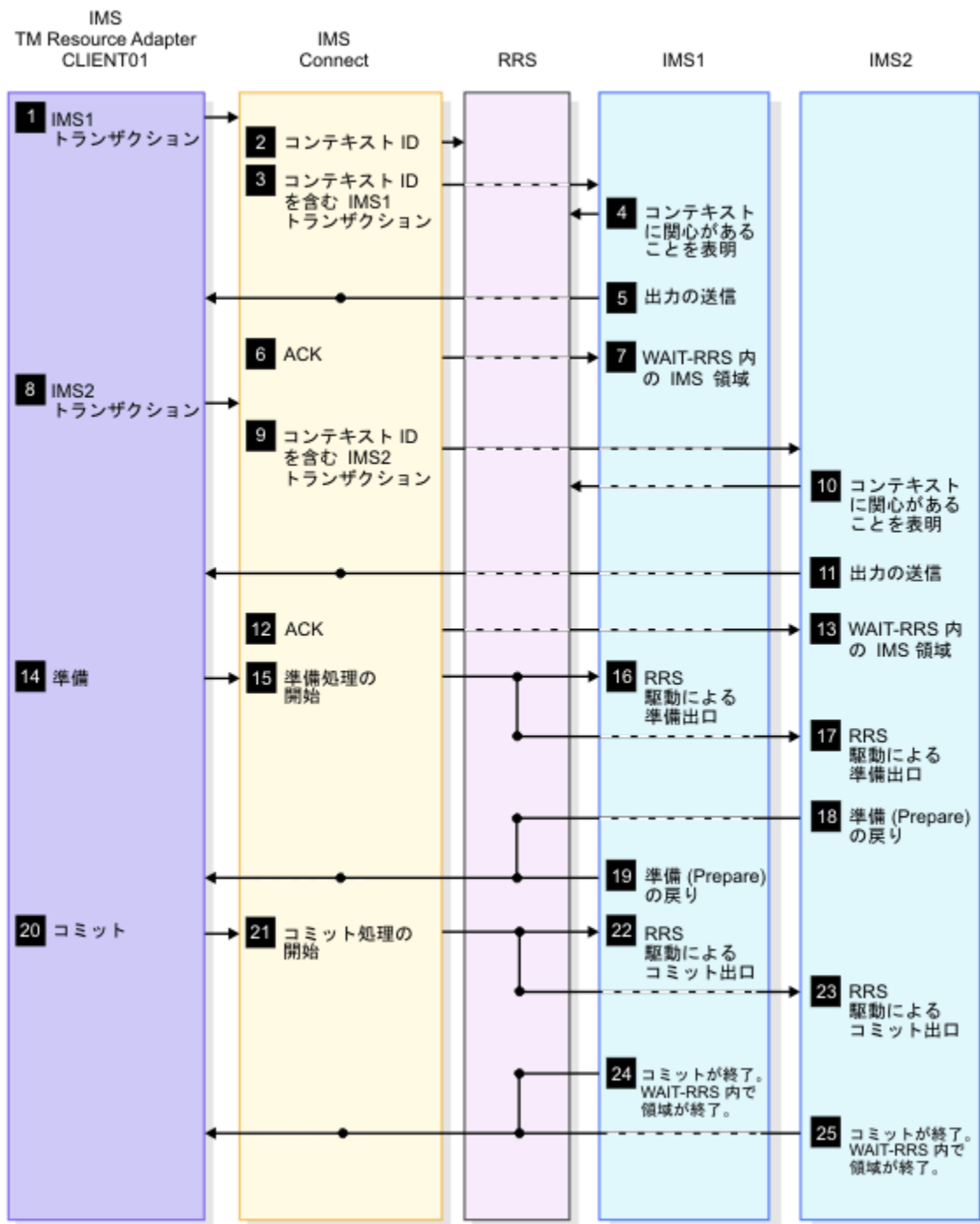


図 40. 分散 2 フェーズ・コミット・グローバル・トランザクションのフロー

## IMS TM Resource Adapter から異なる z/OS イメージ上の IMS システムへのグローバル・トランザクションのカスケード

IMS Connect は、IMS TM Resource Adapter により TCP/IP 接続でサブミットされたグローバル (XA) トランザクションを、異なる z/OS イメージ (LPAR) 上で実行している IMS TM システムにカスケードできます。

### このタスクについて

デフォルトでは、IMS TM リソース・アダプターから別の LPAR 上の IMS システムへのグローバル・トランザクションのカスケードに対するサポートは使用不可になります。トランザクションのカスケードに対す

るサポートは、データ・ストア接続の定義または IMS Connect システム構成の定義のいずれかで CASCADE パラメーターを指定することにより使用可能になります。

グローバル・トランザクションのパフォーマンスは、IMS と IMS Connect が同じ LPAR 上で実行されている場合に最高になるため、パフォーマンスが主要な課題でない限り、IMS Connect と同じ LPAR 上にある IMS システムが使用不可である場合のフェイルオーバー保護として一時的にのみ、LPAR 間でのグローバル・トランザクションをカスケードしてください。

以下のいずれかの方法により、異なる LPAR 上にある IMS TM データ・ストアにグローバル・トランザクションをカスケードするための IMS Connect サポートを使用可能にすることができます。

## 手順

1. 以下のいずれかを行って、IMS Connect システムのデフォルト・サポートを設定します。
  - IMS タイプ 2 コマンド UPDATE IMSCON TYPE(CONFIG) SET(CASCADE(ON)) を発行する
  - IMS PROCLIB データ・セットの HWSCFGxx メンバー内の HWS 構成ステートメントに CASCADE=Y を指定する
2. 以下のいずれかの方法で、個々のデータ・ストア接続に合わせてサポート・オプションを設定します。
  - IMS タイプ 2 コマンド CREATE IMSCON TYPE(DATASTORE) SET(CASCADE(ON)) を発行する
  - IMS タイプ 2 コマンド UPDATE IMSCON TYPE(DATASTORE) SET(CASCADE(ON)) を発行する
  - IMS PROCLIB データ・セットの HWSCFGxx メンバー内の DATASTORE 構成ステートメントに CASCADE=Y を指定する
3. 既存のデータ・ストア接続を更新した場合は、CASCADE オプションを設定してから、データ・ストア接続を再始動してください。

## IMS TM Resource Adapter を使用する 1 フェーズ・コミット・グローバル・トランザクション

共用リソースを変更するトランザクションでただ 1 つのリソース・マネージャーが登録されている場合、トランザクション・マネージャーは 1 フェーズ・コミット最適化を実行できます。外部コーディネーターは必要ありません。

トランザクション・マネージャーは、変更をコミットするためにフェーズ 2 の「コミット要求」をリソース・マネージャーに直接送ることができます。IMS Connect は、フェーズ 1、すなわち 2 フェーズ・コミット・プロトコルの「コミット準備」を実行する必要がなく、フェーズ 2、すなわち「コミット要求」に直接進むことができます。

IMS Connect システム構成または IMS Connect データ・ストア定義のいずれかで CASCADE=Y を指定することによってグローバル・トランザクションのカスケードに対するサポートが使用可能になっている場合、IMS Connect と IMS は、それぞれ異なる z/OS イメージ上で実行しているときにグローバル・トランザクションを処理することができます。グローバル・トランザクションのカスケードに対するサポートが使用不可である場合、IMS Connect、RRS、および IMS すべてが同じ z/OS イメージ上になければなりません。

以下の図は、分散 1 フェーズ・コミット・グローバル・トランザクションのフローを示しています。

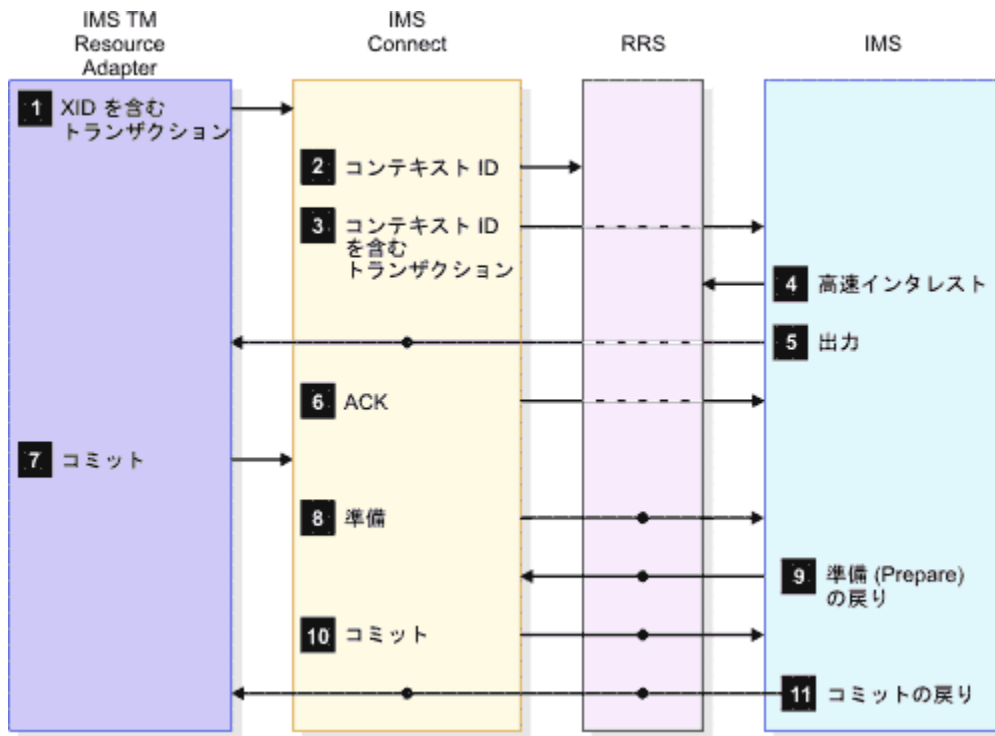


図 41. 分散 1 フェーズ・コミット最適化クライアントのフロー



## 第 20 章 IMS Connect のユニコードに関する考慮事項

IMS Connect は、ユニコード・データが IMS Connect クライアント・アプリケーションとの間で送受信できるようにユニコードをサポートします。特定の環境では、IMS Connect はユニコード・データを変換します。

IMS Connect のユニコード・サポートには、クライアント・アプリケーションと IMS ホスト・アプリケーションの両方が以下をサポートすることが求められます。

- ユニコード・データ
- 送受信されるメッセージの構造および内容と同じユニコード・エンコード・スキーマ (UTF8、UTF16、または UCS-2 のいずれか)

IMS Connect は、クライアント・アプリケーションとの間の両方向で、ASCII および EBCDIC データ・ストリームをサポートします。クライアント・アプリケーションが ASCII データを IMS Connect に送信する場合、ASCII は EBCDIC に変換されます。IMS Connect からクライアント・アプリケーションへの後続の出力は、逆に EBCDIC から ASCII に変換されます。クライアント・アプリケーションが EBCDIC データを IMS Connect に送信する場合は、変換は必要ありません。ユニコード・サポートを使用すると、IMS クライアント・アプリケーションは、ユニコード・データ (具体的には、UTF8、UTF16、または UCS-2 データ・ストリーム) を IMS Connect との間で送受信することもできます。

IMS Connect は言語グループ 1、2、および 3 をサポートします。

クライアント・アプリケーションは、IMS 要求メッセージ (IRM) を以下の目的で使用します。

- 送信するデータがユニコードかどうか、IMS トランザクション・コードがユニコードとして送信されるかどうかを IMS Connect に伝達する。トランザクション・コードをユニコードとして送信する場合、IMS Connect はトランザクション・コードを変換してから、その変換したコードとユニコード・データを IMS に送信します。
- IMS Connect に、使用するユニコード・エンコード・スキーマ (UTF8、UTF16、または UCS-2) を伝達する。

トランザクション・コードは、ユニコード、ASCII、または EBCDIC のいずれでも送信可能です。ただし、トランザクション・コードは、8 バイト・フィールドを占める、最大 8 バイトまでの有効な IMS トランザクション・コードでなければなりません。このフィールドでは、コードは左揃えにし、8 バイトより短い場合は空白を埋め込む必要があります。8 バイトのトランザクション・コード・フィールドの後に空白が続く場合、これはユニコード・データの一部と見なされます。

### メッセージ変換

すべての IMS エラー・メッセージ (例えば、DFS555) は、ASCII または EBCDIC のいずれかで送信されます。クライアント・アプリケーションは、IMS 要求メッセージ (IRM) ヘッダーの IRM\_ID フィールドを使用して、どちらのタイプを送信するかを IMS Connect に伝えます。IMS Connect は、メッセージをユニコードに変換しません。

例えば、IRM\_ID が EBCDIC の場合、IMS エラー・メッセージ (DFSnnnn) は EBCDIC で送信され、IRM\_ID が ASCII の場合、IMS エラー・メッセージ (DFSnnnn) は EBCDIC から ASCII に変換されます。

IRM\_ID は、OTMA ヘッダーのコード・タイプも識別します。

IMS クライアント・アプリケーションは、IMS トランザクション・コードを ASCII、EBCDIC、またはユニコードのいずれでも送信できます。IMS クライアント・アプリケーションがトランザクション・コードをユニコードで送信する場合、IMS Connect ユーザー・メッセージ出口 (HWSSMPLO および HWSSMPL1) は、トランザクション・コードをユニコードから EBCDIC に変換します。クライアント・アプリケーションがトランザクション・コードを ASCII で、残りのデータをユニコードで送信する場合、トランザクション・コードのみが EBCDIC に変換されます。有効な 8 バイトの IMS トランザクション・コードは、以下の文字から構成され、英字で始まらなければなりません。

- A から Z まで (大文字のみ)
- 0 から 9 まで

- 特殊文字 #、\$、@

ユニコードをサポートする IMS ホスト・アプリケーションは、トランザクション・コードを含めるように、入力メッセージ定義で 8 バイト・フィールドを定義する必要があります。この 8 バイト・フィールドをブランクで埋め込むと、このフィールドは EBCDIC のブランクとして送信されます。

クライアント・アプリケーションがユニコード・データを送信する場合、出力メッセージは変換されず、ユニコードとして扱われます。RESUME TPIPE 呼び出しの場合、クライアント・アプリケーションは、出力をユニコードとして扱うかどうかを IRM で指定しなければなりません。メッセージ通信の間、IMS ホスト・アプリケーションは、宛先に応じて出力メッセージを正しくフォーマットする (特定のユニコード・スキーマまたは EBCDIC を使用して) ようにしなければなりません。

## クライアントが送信する入力メッセージのフォーマット

以下の表では、クライアントが送信する入力メッセージのメッセージ構造を対比し、有効な ASCII、EBCDIC、およびユニコードのフォーマットを定義します。

表 69. 入力メッセージ構造 - クライアントが送信するメッセージ

EBCDIC IRM	ASCII IRM	クライアントにより OTMA ヘッダーが渡される場合	トランザクション・コード	データ
Y	N/A	EBCDIC	EBCDIC	ユニコード
Y	N/A	EBCDIC	ユニコード	ユニコード
N/A	Y	ASCII	ASCII	ユニコード
N/A	Y	ASCII	ユニコード	ユニコード

## クライアントが受信する出力メッセージのフォーマット

以下の表では、クライアントがユニコード・データを送信する場合の有効な出力メッセージ・エレメントを定義します。

表 70. 出力メッセージ構造 - クライアントが受信するメッセージ

入力メッセージが EBCDIC IRM であった場合	入力メッセージが ASCII IRM であった場合	RMM	RSM	出力 CSM	出力データ
Y	N/A	EBCDIC	EBCDIC	EBCDIC	ユニコード
N/A	Y	ASCII	ASCII	ASCII	ユニコード

### 関連資料

241 ページの『メッセージ構造と IMS Connect ユーザー・メッセージ出口ルーチン』

IMS Connect は、254 までのユーザー出口を構成ファイルに定義することができます。IMS Connect がサポートする入力メッセージ構造は 2 つあります。また、ユーザー出口から戻るときにも 2 つのメッセージ構造がサポートされます。

## 第 21 章 IMS Connect の TCP/IP 設定

さまざまな TCP/IP 値を選択して、IMS Connect の環境設定を最大限に活用することができます。

TCP/IP に対する z/OS Communications Server 構成設定は、z/OS PROFILE.TCPIP データ・セットにあります。

以下の TCP/IP パラメーター値は、IMS Connect に影響を与えます。

### TCPNODELAY=ENABLE

- データは、クライアントの SEND ごとに TCP/IP で伝送される。
- TCP/IP は、伝送当たり 1 ミリ秒待機する。
- 複数のクライアント TCP/IP SEND の結果は、複数の TCP/IP 伝送になる。

### TCPNODELAY=DISABLE

- 伝送の前に、TCP/IP によってデータはクライアント TCP/IP SEND から収集される。
- TCP/IP は、バッファがいっぱいになるまで待ってから、伝送する。
- 複数のクライアント SEND の結果は、IMS Connect への、1 回から  $n$  回の TCP/IP 伝送になる。

### SO\_LINGER=Y, VALUE=0

- クライアント・コードに即時に戻る。
- ソケットをクローズするクライアント要求は、前のクライアント TCP/IP SEND 要求によるデータ送信をバイパスすることができるが、その結果、クライアント SEND データが失われることもある。

### SO\_LINGER=N

- クライアント・コードに即時に戻る。
- ソケットをクローズするクライアント要求は、前のクライアント TCP/IP SEND 要求によるデータ送信をバイパスすることができるが、その結果、クライアント SEND データが失われることもある。

### SO\_LINGER=Y, VALUE=10

- ACK がホストから受信されたときにクライアント・コードに戻るか、または 10 秒間待ってからクローズを送信する。
- ソケットをクローズしても、データ送信をバイパスしない。

### DELAYACK

DELAYACK は、ホストからのデータ以外の伝送を最小化するために使用されます。DELAYACK が使用されると、z/OS TCP/IP は 200 ミリ秒待ってから、ACK をリモート・サーバー TCP/IP に送信します。ただし、IMS Connect から送信されるデータに ACK を付加する場合は、遅延が発生しません。

クライアント・アプリケーションが 1 つの SEND とそれに続く READ を実行する場合、DELAYACK が推奨されます。

DELAYACK は TCP/IP の PORT ステートメントまたは GATEWAY ステートメントで設定できます。

### NODELAYACK

NODELAYACK は、ホストからのデータ以外の伝送でデータなしのフローを可能にするのに使用されます。NODELAYACK が使用されると、z/OS TCP/IP は即時に ACK をリモート・サーバー TCP/IP に送信します。IMS Connect から送信されるデータに ACK は付加されません。

クライアント・コードが、NODELAYACK を設定して、ホストに対する 1 つの SEND とそれに続く READ を実行する場合、ACK は別に送信されます。

クライアント・コードが、ホストに対する複数の SEND とそれに続く READ を送信する場合、ホスト TCP/IP は、受信されたデータに対して即時に ACK を送信します。これにより、クライアントからのデータの次の SEND フローが可能になります。

クライアント・アプリケーションが複数の SEND とそれに続く READ を送信する場合、NODELAYACK が推奨されます。

NODELAYACK は TCP/IP の PORT ステートメントまたは GATEWAY ステートメントで設定できます。

## **SOMAXCONN**

SOMAXCONN ステートメントは、IMS.PROCLIB データ・セットの IMS Connect HWSCFGxx メンバーの TCPIP ステートメントの TCPIPQ パラメーターと共に使用できます。PROFILE.TCPIP の SOMAXCONN ステートメントは、LPAR レベルでの listen ソケットの TCP/IP キュー項目数を制御します。TCPIPQ パラメーターを使用して、IMS Connect のインスタンスの SOMAXCONN の値をオーバーライドすることができます。TCPIPQ の値は、ホスト LPAR の SOMAXCONN の値より低い場合にのみ使用されます。

**推奨事項:** SOMAXCONN のデフォルト値 (10) は TCPIPQ の最小値 (50) より低いため、TCPIPQ パラメーターは、SOMAXCONN の値が 50 より大きくなるように変更された場合にのみ使用してください。

---

## 第 6 部 IMS VTAM ネットワーク管理

このでは、IMS ネットワークを管理するために役立つ情報を記載しています。



## 第 22 章 IMS Transaction Manager ネットワークの紹介

以下のトピックでは、IMS Transaction Manager (TM) ネットワークの主なコンポーネントを紹介し、IMS と論理装置間でどのようにして通信が確立され、メッセージが伝送され、通信が終了されるかを説明します。

### IMS TM ネットワーク概説

IMS は汎用のデータベース/トランザクション・マネージャー・システムであり、拡張通信ネットワークに必要なサポートをします。仮想記憶通信アクセス方式 (VTAM) は、いろいろな論理装置から IMS へまたは IMS から適切な論理装置へデータを宛先指定して、ネットワーク内のデータの伝送を物理的に制御します。

IMS 通信ネットワークには、以下のコンポーネントを組み込む必要があります。

- IMS
- VTAM (IMS は VTAM のネットワーク機能を使用しますが、基本順次アクセス方式 (BSAM) などの非 VTAM 装置も制御できます。IMS の優先アクセス方式は VTAM です。)
- 通信ハードウェア (制御装置など)
- 端末

ネットワークには、オプションで以下のコンポーネントを組み込むことができます。

- 以下の IMS Transaction Manager (IMS TM) サービス
  - 拡張端末オプション (ETO) (Extended Terminal Option (ETO))
  - 高速機能 (Fast Path)
  - メッセージ形式サービス (MFS)
  - システム間連絡 (ISC)
  - 複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC))
  - 拡張プログラム間通信機能 (APPC) (Advanced Program-to-Program Communication (APPC))
- 以下の構造のいずれかを持つ Common Queue Server (CQS) およびカップリング・ファシリティー
  - 共通キュー構造
  - 共通データ構造
  - リソース構造
- VTAM 汎用リソース・グループ
- Open Transaction Manager Access (OTMA)
- 以下を含む共通サービス層 (CSL)
  - Operations Manager (OM)
  - Resource Manager (RM)
  - 構造化呼び出しインターフェース (SCI) (Structured Call Interface (SCI))

さらに、IMS 通信ネットワークは、以下のいずれかのフレームワーク内で操作することができます。

- 複数の製品を統一された設計概念の基で統合する IBM システム・ネットワーク体系 (SNA)。また、SNA は、それぞれのネットワーク・コンポーネントの機能的役割を形式的に定義します。
- 拡張プログラム間通信 (APPC) または OTMA を使用するクライアント/サーバー環境。

#### 定義:

- 論理装置とはアドレス可能なリソースであり、アプリケーション・プログラム、端末、または顧客情報管理システム (CICS) のようなサブシステムにすることができます。論理装置はまた、次のものからなる汎用端末システムのコンポーネントの 1 つにすることもできます。すなわち、プログラマブル・コントローラー、それに接続されたオペレーター端末、プリンター、および補助制御装置の構成です。

- 端末という用語は、本書では装置を記述するために使用されており、また、コントローラーやリモート・サブシステムにも適用されます。オペレーター端末とは、キーボード・プリンター、キーボード付きディスプレイ装置、通信端末装置などを示すか、またはこれらの装置の組み合わせの構成のことを指します。

IMS とプログラマブル論理装置から構成されるネットワークでは、ユーザーが各機能をネットワーク内のコンポーネントに分散させて持たせることができます。このような機能の分散によって、ホストとしての中央処理装置への処理要件を軽減したり、1つのコンポーネントに問題がある場合、ネットワークの他の部分への影響を軽減することができます。

以下の図は、完全な通信ネットワーク・システムのコンポーネントを示します。図の中の矢印は、コンポーネント間で行われる通信を示します。図は、以下のコンポーネントを示しています。

- IMS およびそのアプリケーション・プログラム
- 仮想記憶通信アクセス方式 (VTAM)
- Tivoli NetView for z/OS
- z/OS オペレーティング・システム (APPC/IMS が使用されている場合は APPC/MVS を含む)
- IBM 37x5 通信コントローラーおよびネットワーク制御プログラム (NCP)
- 端末

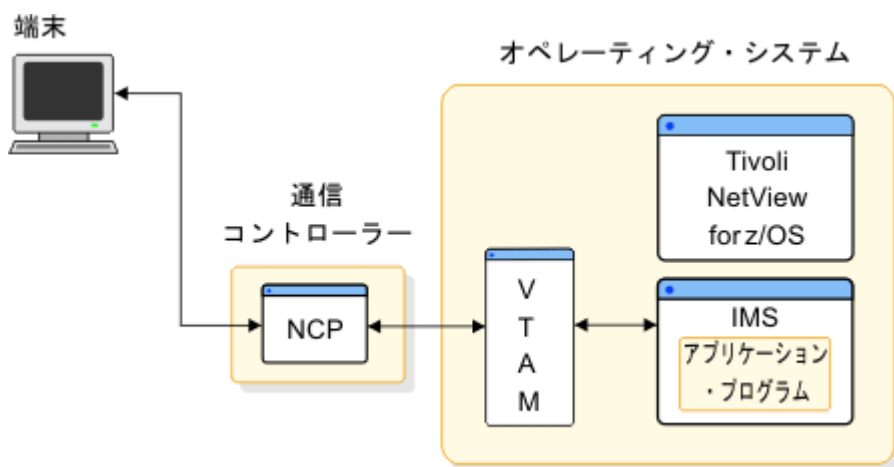


図 42. ネットワークのコンポーネント

前の図の各コンポーネントがネットワークにどのように参加しているかを、以下のリストに要約します。

### IMS

- トランザクション・セキュリティーの検査
- 適切なアプリケーション・プログラムのスケジューリング
- 適切な端末への出力の指示
- チェックポイントとリカバリー能力の提供

### アプリケーション・プログラム

- 端末からのデータの読み取り
- プロセッサへのデータの書き込み
- プロセッサからのデータの読み取り
- 端末へのデータの書き込み

### VTAM

- 端末のネットワークからの接続および切断



- 端末から IMS へのデータの送信
- ネットワークのモニターと変更の許可
- IMS から端末へのデータの送信
- 物理的ネットワークの管理 (Tivoli NetView for z/OS を使用)

### 通信コントローラー

- 回線制御文字の追加
- データの伝送
- データの受信
- 回線制御文字の取り外し
- 伝送エラーの検査
- バッファリングの制御

### ネットワーク制御プログラム (NCP)

- 通信回線 / アダプターからのデータの送信および受信
- エラーの検査および記録

IMS ネットワークを計画する場合は、ネットワークのそれぞれのコンポーネントを理解することと、他のコンポーネントとの関係を知っておく必要があります。

以下のトピックでは、追加情報を提供します。

## プログラマブル論理装置 (LU)

### 定義:

- プログラマブル論理装置 (LU) は IMS とセッション接続されている入出力装置です。リモート論理装置のアプリケーション・プログラムは、複数の端末を制御するように設計することができます。
- 論理装置が IMS との通信を望んでいることを VTAM に通知すると、VTAM は VTAM ログオン 出口ルーチンを使用して、IMS に通知します。次に、IMS が要求を受け入れると、VTAM は論理装置を IMS に論理的に接続します。この論理接続をセッションと呼びます。

このリモート・アプリケーション・プログラムの機能の代表的なものは以下のとおりです。

- 関連する端末の読み取りと書き込み
- 端末から受信するデータの編集と確認
- リモート論理装置内のディスク装置の読み取り / 書き込み
- IMS が稼働しているホストの読み取り / 書き込み
- IMS が稼働中のホストから受信したデータの編集と確認
- 他のネットワークの論理装置との通信
- 表示装置とプリンターのフォーマット設定
- ホスト、VTAM、IMS、NCP が使用不能の時のオフライン操作

システム・タイプにより、論理端末はすべて 1 つ以上の端末で構成されます。複数の端末またはコンポーネントから構成される LU を制御するアプリケーション・プログラムは、出力を特定の装置に送ることができなければなりません。このため、アプリケーション・プログラムでは、何らかの形でデータの問い合わせを行い、適切な装置選択ができなければなりません。IMS はこの過程でアプリケーション・プログラムの以下の項目をサポートします。

- LU のコンポーネントを個々に設定し、アドレッシングすることができる。
- 出力メッセージのヘッダーでどのコンポーネントに出力するかの識別ができる。

## 通信コントローラーとネットワーク制御プログラム (NCP)

VTAM は、37x5 通信コントローラー内で稼働する NCP の機能を使用します。VTAM が NCP を使用する目的を以下に示します。

- コントローラーに接続されている回線および装置の制御
- 論理装置とホスト CPC 間のデータの伝送
- エラー・リカバリーの実行
- ネットワークに関する統計の収集

## 仮想記憶通信アクセス方式 (VTAM)

VTAM は、ネットワーク・リソースの割り振りを制御して、VTAM ユーザーがこのリソースを共用できるようにします。VTAM に対して、IMS は 1 つの VTAM ユーザーです。VTAM は、IMS アプリケーション・プログラムを認識しません。

IMS アプリケーション・プログラムは、IMS 呼び出しインターフェースを使用して、IMS のサービスを要求し、VTAM 機能を起動するために、IMS は VTAM マクロを使用します。

APPC/IMS がアクティブである場合、VTAM はそれを別のユーザーと見なします。

### IMS での VTAM USERVAR の使用

IMS は、XRF 複合システムで VTAM ユーザー変数 (USERVAR) を使用して、アクティブ IMS サブシステムに障害が起きたときにセッションを保守し、管理します。限定された事例では、VTAM USERVAR を使用して、XRF 複合システムの一部ではない IMS の VTAM アプリケーション ID をポイントすることもできます。ただし、IMS が VTAM USERVAR のこの使用方法をサポートするのは、ポイントされた IMS がセッションの基本論理装置 (PLU) である場合のみです。

例えば、IMS システム のアプリケーション ID を変更したが、LU が元のアプリケーション ID を使用して IMS システム に継続して接続することを一時的に許可する場合に、非 XRF コンテキストで VTAM USERVAR を使用することができます。

注：ISC または MSC 環境内では、XRF 複合システムの一部ではない IMS サブシステム のポイント時に、VTAM USERVAR が予測できない動作をすることがあります。この予測できない動作の原因は、ISC 環境および MSC 環境での 2 つの IMS サブシステム間のセッション内では、IMS サブシステム は PLU または 2 次論理装置 (SLU) のどちらの場合もあり得るという事実にあります。

### 関連資料:

- VTAM の説明と使用方法について詳しくは、「*z/OS Communications Server: SNA Programming*」を参照してください。
- VTAM USERVAR について詳しくは、「*z/OS Communications Server: SNA ネットワーク・インプリメンテーション・ガイド*」を参照してください。

## IMS 製品の機能

IMS は以下の 2 つの主な製品の機能で構成されています。

- データベースを制御するデータベース・マネージャー (IMS DB)
- データ通信およびアプリケーション・プログラムを制御する Transaction Manager (IMS TM)

IMS には以下の機能があります。

- アプリケーション・プログラムに必要な標準機能
- 多数のオンライン・ユーザーがアプリケーション・プログラムを並行して実行するための実行環境
- 全機能および高速機能データベースの制御

IMS および IMS が制御するアプリケーション・プログラムは、z/OS で実行されます。

## 関連概念

拡張回復機能の概要 (システム管理)

# IMS Transaction Manager サービス

このトピックでは、オプションの Transaction Manager サービスのいくつかについて説明します。

## APPC/IMS および LU 6.2 装置

APPC/IMS は IMS TM の一部で、これにより、会話用共通プログラミング・インターフェース (CPI-C) を使用して CPI アプリケーション・プログラムを作成することができます。APPC/IMS は、APPC/MVS の機能で APPC をサポートします。

IMS は暗黙的、明示的に APPC サポートのアプリケーション・プログラム・インターフェース (API) をサポートします。APPC サポートの暗黙的 API は、他のリモート装置と同じ技術を使用して、IMS アプリケーション・プログラムが他の APPC 装置と通信できるようにします。この結果、APPC LU 6.2 装置について、何の知識もないプログラマーが書いたアプリケーション・プログラムでも以下のことができます。

- APPC LU 6.2 装置からの始動
- 元の LU 6.2 装置からの入力メッセージの受信
- 発信 LU 6.2 装置への出力メッセージの送信
- リモート LU 6.2 装置でのトランザクション・プログラムの始動

同様に、このアプリケーション・プログラムは、すべての装置タイプ (LU 6.2 非 LU 6.2 の別にかかわらず) で新しいコーディングや変更なしに使用できます。

APPC をサポートする明示的アプリケーション・プログラミング・インターフェース (API) は、CPI-C インターフェースであり、IMS のどのアプリケーション・プログラムにでも使えます。IMS アプリケーション・プログラムは、このインターフェースを介して APPC/MVS に対する呼び出しを発行できます。<sup>9</sup>

## 高速機能

高速機能は、トランザクションおよびデータベースの処理を高速に実行することができます。ユーザーのシステム要件のトランザクション・ボリュームが大きい場合、全機能処理のすべての機能を必要とするわけではなければ、高速機能の使用が有益です。このようなアプリケーションの例としては、銀行用のテラー・トランザクション および小売業の販売時点 (在庫更新) トランザクションがあります。高速機能の入出力メッセージに急送メッセージ・ハンドラー (EMH) を使用すると、メッセージ・キューイングや優先順位スケジューリングをバイパスします。

## 拡張端末オプション (ETO) (Extended Terminal Option (ETO))

IMS 拡張端末オプション (ETO) は、IMS に、動的端末やローカルおよびリモート論理端末 (LTERM) のサポートを提供します。

**定義:** 論理端末 (LTERM) はユーザーの宛先です。静的に定義された端末の場合、1つの LTERM は1つの物理端末に関連付けられます。ETO 端末の場合、1つの LTERM は1人のユーザーに関連付けられ、ユーザーが物理端末にサインオンした後でのみ、物理端末に関連付けられます。

すなわち、最初に定義していなくても、IMS に端末ハードウェア および LTERM (ユーザー) を追加することができます。ETO を使用すると、オプションで、VTAM 端末および LTERM の IMS システム定義内のマクロ・ステートメントを除去することができます。ETO は、端末や LTERM を追加する時にシステムをシャットダウンする必要がないので IMS の可用性が高まります。

さらに、ETO は、すべての出力を、装置ではなく IMS TM の特定のユーザーに関連付けることによって、ユーザーのセキュリティを高めます。ETO は、ユーザーがサインオンする必要があります。

<sup>9</sup> z/OS ATBxxx 呼び出しサービスを使用することもできます。これらの呼び出しサービスについては、「z/OS MVS プログラミング: APPC/MVS トランザクション・プログラムの書き方」を参照してください。

ETO は、端末ネットワークが動的に定義されているシステムに関する、IMS システム 定義時間を削減します。

## システム間連絡 (ISC)

IMS と他の外部サブシステム間でデータを交換することができます。

**定義:** 外部サブシステム は、IMS が使用できるが制御できないデータベース・リソースのセットを提供するサブシステムです。IMS が接続できる外部サブシステム の例を次に示します。

- 顧客情報管理システム (CICS)
- その他の IMS システム
- ユーザー作成サブシステム

IMS と外部サブシステム間で作成されるセッションは、アプリケーション間セッションと呼ばれます。このセッション内で作動する IMS は、SNA プロトコルを使用するシステム間連絡機能 (ISC) という機能を使用します。

## メッセージ形式サービス (MFS)

IMS メッセージ形式サービス (MFS) は、アプリケーション・プログラムが入出力装置のデータに左右されることなく、端末からまたは端末へのメッセージをフォーマット設定する機能です。アプリケーション・プログラムは、装置によって入力と出力が異なる場合でも、単一の編集論理を使用して、様々な装置タイプに合わせてメッセージをフォーマット設定することができます。

IMS MFS MFS は、リモート制御装置やサブシステム内のユーザー作成プログラムとのメッセージもフォーマット設定するので、アプリケーション・プログラムは、リモート制御装置の伝送特性に対応する必要はありません。

## 複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC))

複数システム結合機能 (MSC) では、ある IMS にトランザクションを入力し、他の IMS で処理することができます。IMS からの応答は、トランザクションを入力した端末または他の端末のどちらにでも戻すことができます。

### 関連概念

[390 ページの『高速機能急送メッセージ・ハンドラー』](#)

このトピックでは、高速機能急送メッセージ・ハンドラー (EMH) およびこのハンドラーが高速機能メッセージを処理する方法について簡単に説明します。

[396 ページの『IMS メッセージとスケジューリング』](#)

IMS メッセージのタイプは、IMS がプロセスを制御する 4 つのタイプのデータ通信 (トランザクション、LTERM へ送信されるメッセージ、IMS コマンド、DFSAPPC) のうちの 1 つです。

[395 ページの『論理端末 \(LTERM\)』](#)

論理 端末 (LTERM) はユーザーの宛先です。静的に定義された端末の場合、1 つの LTERM は 1 つの物理端末に関連付けられます。ETO 端末の場合、1 つの LTERM は 1 人のユーザーに関連付けられ、ユーザーが物理端末にサインオンした後でのみ、物理端末に関連付けられます。

[435 ページの『メッセージ形式サービス』](#)

メッセージ形式サービス (MFS) は、IMSS アプリケーション・プログラムが入力メッセージまたは出力メッセージ内の装置固有の特性に対応せずにすむように、端末との間でやりとりするメッセージをフォーマット設定する IMS 機能です。

### 関連タスク

[23 ページの『CPI 通信および APPC/IMS』](#)

これらのトピックでは、CPI 通信と APPC/IMS を紹介します。このトピックでは、CPI 通信ドリブンのアプリケーション・プログラムの機能する方法、APPC/IMS を管理し、CPI アプリケーション・プログラムを構築するために CPI 通信インターフェース付きの APPC/IMS を使用する方法を説明します。

#### 61 ページの『拡張端末オプション (ETO) (Extended Terminal Option (ETO))』

これらのトピックでは、拡張端末オプション (ETO) を紹介し、ETO の概要および IMS TM ネットワーク内の ETO 端末の管理に必要な情報を説明します。

#### 453 ページの『システム間連絡 (ISC)』

これらのトピックでは、システム間連絡 (ISC) を紹介し、ISC を使用して IMS サブシステムと ISC プロトコルをサポートする他のタイプのサブシステムを接続する際に必要なすべての情報を提供します。

#### 681 ページの『複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC))』

これらのトピックでは、複数システム結合機能 (MSC) を紹介します。MSC を使用して、複数の IMS サブシステムを接続することができます。これらのトピックでは、MSC の概要と MSC ネットワークの設計、実装、および管理で必要となる情報を説明します。

## データ通信制御機能 (DCCTL) 環境

データ通信制御機能 (DCCTL) 環境では、IMS データベース・マネージャーから独立して、IMS Transaction Manager を使用することができます。

DCCTL は、スループット、システム使用可能性、保全性の点で、システム・パフォーマンスを改善します。DCCTL は、現在の IMS アプリケーション・プログラムやインストール済み端末と共存して使うことができます。

DCCTL を使用する以下のシナリオでは、既存の環境への変更を必要としません。

- 外部データベース・マネージャーにアクセスするアプリケーション・プログラムは、変更なしに DCCTL を使用することができます。例えば、DCCTL は、Db2 for z/OS のフロントエンド・トランザクション・マネージャーとしての機能を持っています。Db2 for z/OS サブシステムは DCCTL で稼働するために変更する必要はありません。
- DCCTL の環境下では、外部サブシステム・リソースにアクセスする IMS 出口ルーチン および IMS アプリケーション・プログラムを変更する必要はありません。
- グローバル・リソース管理は、DCCTL 環境内では、DB/DC 環境内にあるときと同じです。

以下のプロシージャーは DCCTL 環境に合わせて変更しなければならない場合があります。

- 操作プロシージャー
- システム定義によって生成される汎用プロシージャー
- 外部サブシステムと IMS データベースの両方にアクセスする呼び出しが、アプリケーション・プログラムに混ざっている場合には、変更が必要となります。DL/I 呼び出しをすると、状況コードが AD になります。

既存のシステム定義プロシージャーは、DCCTL システムの生成をサポートします。DCCTL は、データ処理環境を表す制御ブロック群を使って実行されます。この制御ブロックは、システム、データ通信、トランザクション・マネージャーのコンポーネントを記述しています。

DCCTL 環境クラス・システム定義を使用すると、TM バッチ環境を生成することができます。TM バッチを使用すれば、IMS バッチ端末シミュレーター (BTS) を利用するか、または Db2 for z/OS システムにアクセスすることができます。TM バッチ・サポートは、DBB と DL/I 領域に限られます。DL/I データベース機能は、サポートしません。

**関連資料:** Db2 for z/OS へのアクセスについて詳しくは、「DB2 for z/OS アプリケーション・プログラミングおよび SQL 解説書」を参照してください。

### 関連概念

[DCCTL 環境 \(システム管理\)](#)

## IMS ネットワークの操作

基本的な IMS ネットワークの操作には、いくつかのタスクが含まれます。

操作タスクには以下のものがあります。

- 論理装置と IMS 間の通信セッションの設定
- 論理装置と IMS 間のデータの送信
- 論理装置と IMS 間のセッションの終了
- セッションに障害が起こった時の再始動

### APPC/IMS によるネットワークの操作

APPC/IMS では、ネットワーク操作に IMS コマンドが使用できますが、LU 6.2 装置で、セッション開始、トランザクション開始、マスター端末オペレーター (MTO) 介在なしのエラー処理などの通常操作を処理します。

### IMS によるセッションの開始

セッションの開始は、論理装置、VTAM ネットワーク・オペレーター、IMS マスター端末オペレーター (MTO)、VTAM で自動的に、または IMS 自体のいずれからでもできます。論理装置は、IMS に接続した後、以下のいずれかのアクションが行われるまで、接続されたままです。

- 論理装置自身がシャットダウンを要求する
- IMS MTO が切断を要求する
- 他の VTAM アプリケーション・プログラムが該当端末に接続を要求する
- IMS、VTAM、NCP、論理装置またはネットワーク全体が停止される

コントローラーと VTAM 間に物理接続が設定されると、LU 間セッションが開始されます。セッションを要求している LU は、IMS との通信が必要である旨を VTAM に伝えます。VTAM はその要求を IMS に VTAM ログオン出口ルーチンを通じて伝えます。IMS が、その要求の受け入れを示すと、VTAM は、論理的に LU を IMS につなげます。LU と IMS の通信が成立する前に、セッションが必要です。セッションを開始するには、通信パス (IMS、NCP、回線、およびステーション) のノードがすべてアクティブ状況である必要があります。セッションが設定されると、VTAM は、論理装置と IMS 間のすべてのデータの管理を指示します。

IMS では、拡張回復機能 (XRF) 複合システムにおいて、SNA 通信リンクもサポートされています。

### IMS へのログオンとサインオン

以下の定義は、IMS へのログオンと IMS へのサインオンに適用されます。

#### 定義:

- 端末へのログオンによって、その端末と IMS 間のセッションが設定されます。
- 端末へのサインオンによって、ユーザーが IMS に対して識別されます。

### IMS からのログオフとサインオフ

以下の定義は、IMS からのログオフと IMS からのサインオフに適用されます。

#### 定義:

- 端末からのログオフによって、その端末と IMS 間のセッションが終了します。
- 端末からのサインオフによって、IMS に対する特定のユーザーの識別が終了します。

#### 関連概念

[拡張回復機能の概要 \(システム管理\)](#)



## 共用キュー環境

共用キュー環境における操作によって、シスプレックス環境の複数のIMSシステムがIMSメッセージ・キューおよびEMHメッセージ・キューを共用できるようになります。IMSシステムは、複数のIMSシステムの単一イメージ・ビューを提供するIMSplexとしてともに作業を行います。

共用キュー環境はIMSplex内のIMSシステム間の処理ロードを分散します。1つのIMSで入力したトランザクションは、トランザクションを処理できる他の任意のIMSによって共用キュー上で使用可能になります。これらのトランザクションの結果は、入力端末に戻されます。エンド・ユーザーはこれらのアクティビティについて認識する必要はありません。つまり、エンド・ユーザーはこれらの処理を単一システムの操作のように見なすことができます。

### 定義:

- 共用キューとは、同じキュー名によって関連付けられているメッセージの集合体です。共用キューは、Common Queue Server (CQS) によって管理され、IMSplex内のCQSクライアントによって共用することができます。
- Common Queue Serverとは、そのクライアントのカップリング・ファシリティのリスト構造に常駐する共用キューからデータ・オブジェクトを受信し、保守し、分散します。
- CQSクライアントとは、固有のCQSを介して共用キューにアクセスするIMS DB/DCまたはDCCTLシステムです。
- カップリング・ファシリティとは、シスプレックス環境で高速のキャッシング、リスト処理、およびロック機能を提供する特殊な論理区画です。
- シスプレックス環境とは、作業負荷を処理するために特定のマルチシステムのハードウェア・コンポーネントとソフトウェア・サービスを介して相互に通信し、連携するz/OSシステムのセットです。
- IMSplexは、ユニットとして共同作業を行う、1つ以上のIMS制御領域、管理機能、またはサーバーです。IMSplex内のIMSシステムは一般的に以下のことを行います(ただし、常にではありません)。
  - データベース、リソース、またはメッセージ・キュー(あるいはいずれかの組み合わせ)の共用
  - z/OS Parallel Sysplex®(並列シスプレックス)環境での実行
  - IMS共通サービス層(CSL)の組み込み

一般的に、IMSはメッセージを以下の方法で処理します。

1. IMSシステムが、処理できるメッセージのインタレストを各自のキューに登録する。
2. IMSがメッセージを受信し、共用キューに置くと、そのキューにインタレストに登録したすべてのIMSシステムに通知される。
3. 1つのIMSがメッセージをリトリブし、処理する。
4. メッセージを処理したIMSが、応答をキューに置く。
5. 元のメッセージを受け渡したIMSに、応答メッセージがキューに置かれたことが通知される。
6. 元のメッセージを受け渡したIMSが、応答メッセージを発信元端末に送信する。

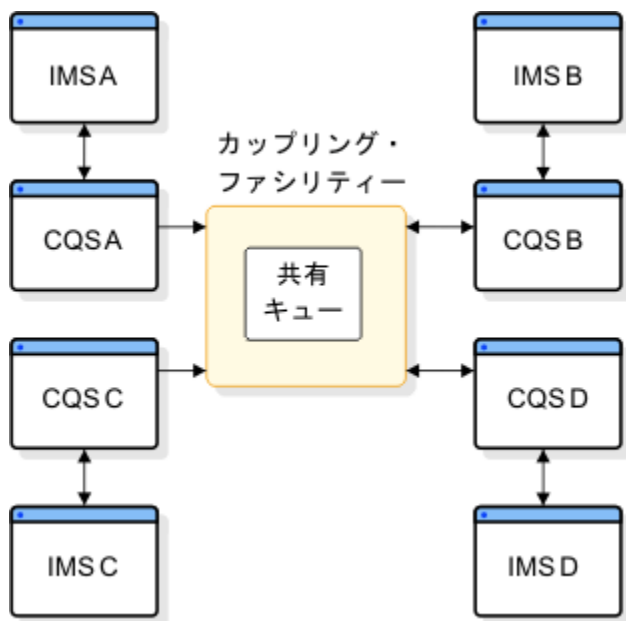


図 43. 基本的な共有キュー環境

#### 関連資料

[キュー・タイプ \(システム管理\)](#)

## 共有キュー使用の利点

共有キュー環境で操作する主な利点は、作業負荷の自動的な平衡化、増加への対応、および信頼性の向上です。

共有キューがこれらの利点をどのように実現するかを、以下のリストで説明します。

#### 作業負荷の自動的な平衡化

共有キューに置かれたメッセージは、作業を行える任意の参加 IMS によって処理することができます。

#### 増加への対応

作業負荷の増加に応じて、新しい IMS システムを追加することができます。

#### 信頼性の向上

ある IMS で障害が発生しても、共有キューに置かれた作業は、他の IMS システムによって処理することができます。

#### 推奨:

- 共有キューでは汎用リソース・グループを使用してください。
- IMSplex 内のすべてのデータを IMSplex 全体で共有してください。

#### 関連概念

[387 ページの『汎用リソースによるセッションの平衡化』](#)

シスプレックス環境で複数の IMS システムを操作している場合は、汎用リソース・グループの名前を使用して、セッションを開始することができます。VTAM は汎用リソース・グループ内の汎用リソース・メンバー間でセッションを平衡化します。

[IMS 環境でのデータ共有 \(システム管理\)](#)

## 共有キュー環境に必要なコンポーネント

共有キュー処理には、さまざまなコンポーネントが必要です。

共有キュー環境は、多くのさまざまな構成で操作できますが、共有キュー処理に必要なコンポーネント ([386 ページの図 44](#) を参照) には、以下のものが含まれます。



## Common Queue Server (CQS)

クライアントごとに1つのCQSが必要ですが、複数のIMSシステムがCQSを共用できます。各CQSは、カップリング・ファシリティのリスト構造に常駐する共用キューにアクセスします。

### CQS クライアント

CQS クライアント要求を使用して共用キューにアクセスできる1つ以上のIMS DB/DC または DCCTL サブシステム。

### z/OS カップリング・ファシリティ・リスト構造

共用キューを保持するカップリング・ファシリティ構造のタイプ。

#### 定義:

- リスト構造とは、カップリング・ファシリティ内のストレージ域の1つであり、シスプレックス環境内のマルチシステム・アプリケーションが、リストまたはキューのセットとして編成された情報を共用できるようにするものです。リスト構造は、リストのセットとオプションのロック・テーブルから構成されます。
- CQS は、基本リスト構造とオーバーフロー・リスト構造からなる構造の対と呼ばれる対でリスト構造を維持します。
- 基本リスト構造には共用キューが入ります。
- オーバーフロー・リスト構造(定義した場合)には、基本リスト構造が事前定義しきい値に到達した後オーバーフローする共用キューが入れられます。

### z/OS システム・ログ

それぞれの構造ペアごとに1つのz/OS システム・ログが使用されます。CQS は、処理した作業およびリスト構造ペアに関するリカバリー情報をz/OS ログ・ストリーム内に置きます。これらのログ・ストリームは、リスト構造の対にアクセスするすべてのCQSによって共用されます。

### CQS チェックポイント・データ・セット

各CQSの構造の対ごとに1つのCQS チェックポイント・データ・セットが保持されます。CQS チェックポイント・データ・セットには、CQS システム・チェックポイント情報が入ります。

### CQS 構造リカバリー・データ・セット (SRDS)

CQS は、構造上の共用キューを回復できるように、構造の対ごとに2つのSRDSを保持します。SRDSは共用キューに関する構造チェックポイント情報を保持します。

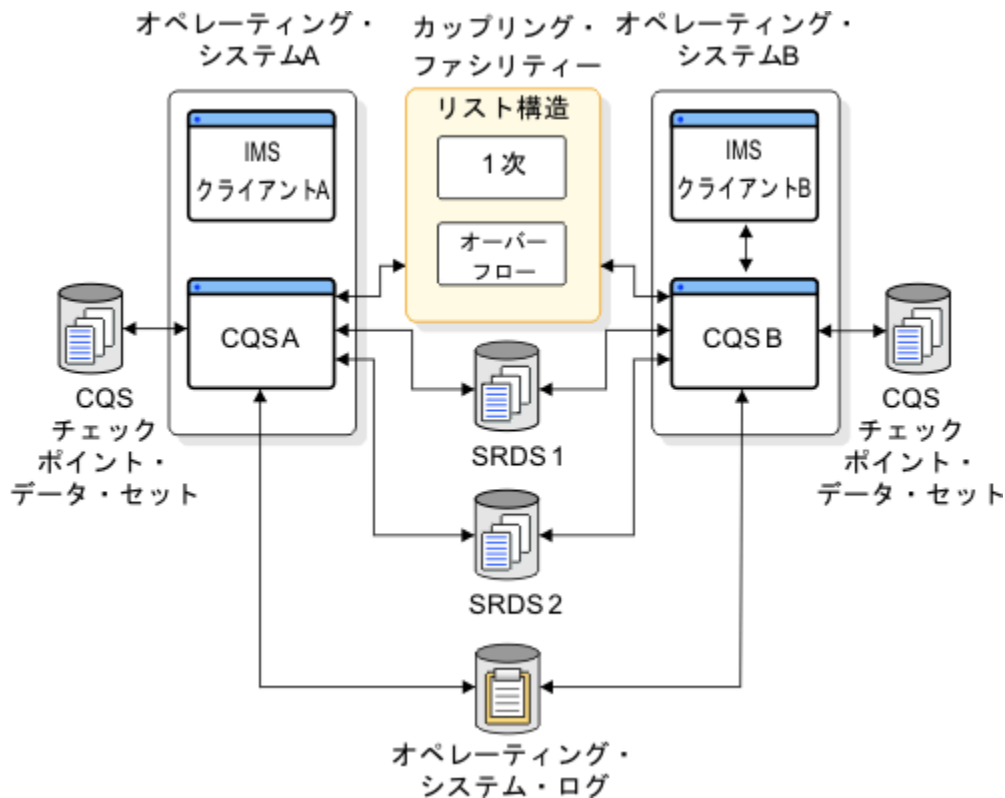


図 44. 共用キュー環境のコンポーネント

#### 関連概念

386 ページの『Common Queue Server の概説』

Common Queue Server (CQS) は IMS が共用メッセージ・キューと通信するための内部インターフェースです。CQS 要求を開始するには、IMS コマンドを使用します。

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

## Common Queue Server の概説

Common Queue Server (CQS) は IMS が共用メッセージ・キューと通信するための内部インターフェースです。CQS 要求を開始するには、IMS コマンドを使用します。

CQS アドレス・スペースは IMS によって開始されます。

CQS は、以下のサービスを提供します。

- 共用キューに作業が存在している場合に、登録済みクライアントに通知する。
- 共用キューおよび CQS にアクセスするためのインターフェースをクライアントに提供する。
- CQS システム・チェックポイント情報を CQS チェックポイント・データ・セットに書き込む。
- 共用キュー・リスト構造リカバリーのために、構造チェックポイント情報を SRDS に書き込む。
- 共用キュー・リスト構造に関する構造リカバリーおよびオーバーフロー処理を提供する。
- 以下の CQS クライアント 出口ルーチンを実行する。
  - クライアント CQS イベント出口ルーチンは、CQS 異常終了および CQS 再始動完了などのシステム・イベントについてクライアントに通知します。
  - クライアント構造イベント出口ルーチンは、構造コピー、構造リカバリー、構造オーバーフロー、構造チェックポイント、および構造再同期などの構造イベントについて、クライアントに通知します。
  - クライアント構造通知出口ルーチンは、共用キューに作業が存在している場合に、クライアントに通知します。
- 以下の CQS ユーザー提供の出口ルーチンを実行する。

- CQS キュー・オーバーフロー・ユーザー提供の出口ルーチンによって、CQS がオーバーフロー用に選択したキューを、出口ルーチンが受け入れるかどうかを決定することができます。
- CQS 初期設定 / 終了ユーザー提供の出口ルーチンは、CQS が初期設定したとき、および (正常終了条件の下で) すべての構造から切断した後で CQS が終了したときに通知されます。
- CQS クライアント接続ユーザー提供の出口ルーチンは、構造へのクライアントの接続、および構造からのクライアントの切断に関与します。
- CQS 構造統計ユーザー提供の出口ルーチンは、CQS システム・チェックポイント時に構造統計を収集します。
- CQS 構造イベント・ユーザー提供の出口ルーチンは、構造チェックポイント、構造再作成、構造オーバーフロー、および構造状況変更などの構造アクティビティを追跡します。さらに、CQS が構造に接続した時期または構造から切断した時期も追跡します。
- サンプル JCL 付きのログ印刷ユーティリティを提供します。サンプル JCL を使用すると、z/OS ログからのログ・レコードを印刷することができます。

## 関連概念

[CQS の管理 \(システム管理\)](#)

[共用キューの使用可能化 \(システム管理\)](#)

## 関連資料

[CQS クライアント出口ルーチン \(出口ルーチン\)](#)

# 汎用リソースによるセッションの平衡化

シスプレックス環境で複数の IMS システムを操作している場合は、汎用リソース・グループの名前を使用して、セッションを開始することができます。VTAM は汎用リソース・グループ内の汎用リソース・メンバー間でセッションを平衡化します。

特定の IMS のサービスを必要としない場合、特定の IMS のアプリケーション ID 名ではなく、汎用リソース名を使用してセッションを開始してください。VTAM 汎用リソースはシスプレックス環境で実行されることを想定されていますが、シスプレックス環境は必要ではありません。

このトピックでは、VTAM 汎用リソースの利点および用語に関する概要を説明します。

## 汎用リソース・グループの利点

汎用リソース・グループを使用する利点には以下のものがあります。

### セッションの作業負荷の自動的な平衡化

汎用リソースと共用キューの使用は互いに補足し合います。汎用リソースはネットワーク・トラフィックを複数の IMS システムに分散します。一方、共用キューはバックエンド・アプリケーション作業負荷を分散します。

### 単一イメージ・リソース

複数の IMS システムにアクセスするために、単一の汎用リソース名を使用できるので、多くの IMS システムのリソースを使用する場合に、単一システム・イメージを提供できます。

### 拡張 IMS システムの可用性

一般的に、1つの IMS が失敗しても、その汎用リソース・グループ内の別の IMS にログオンすることができます。

**例外:** 端末が LOCAL 状況リカバリー・モードである場合、ログオンできない可能性があります。

### 共用グローバル・メッセージ

汎用リソース・グループ内の任意の IMS から、共用キュー上のメッセージを入手できます。

**推奨事項:** 会話モードまたは高速機能の応答モードの端末からメッセージを入手するには、入力が最初にサブミットされたシステムに前もって再ログオンしておく必要があります。状況リカバリー・モードが LOCAL の場合、入力が最初にサブミットされたシステムに前もって再ログオンしておかなければなりません。状況リカバリー・モードが GLOBAL または NONE の場合、汎用リソース・グループ内の任意の IMS にログオンすることができます。

## 汎用リソース・グループ定義

### 定義:

- 汎用リソース・グループとは、同じ汎用リソース名を持つ IMS サブシステムのセットであり、VTAM はこれらのサブシステム間で端末セッションを分散することができます。
- 汎用リソース・メンバーとは、汎用リソース・グループに属する IMS サブシステムのことを言います。
- 汎用リソース名とは、VTAM が、汎用リソース・グループに属するすべての IMS サブシステムを認識する際に使用する共通名です。
- アプリケーション ID 名とは、VTAM が IMS サブシステムを認識する際に使用する固有のアプリケーション・プログラム名です。汎用リソース・グループ内では、VTAM はアプリケーション ID 名を使用して各 IMS サブシステムを識別します。
- 汎用リソースのコンテキストでは、類似性とは、VTAM 論理装置と汎用リソース・グループ内の特定の IMS サブシステムとの間の関連付けです。

## 汎用リソース類似性

VTAM は、以下の環境内で、端末と特定の IMS の間で類似性を設定します。

- IMS によって開始されるすべてのセッションに対して。
- 汎用リソース名を使用して IMS にログオンするとき。

類似性がリセットされるまで、その後のログオンは同じ IMS に行われます。

次のいずれかの場合には、類似性は自動的にリセットされます。

- VTAM 管理類似性がセッションに対して確立されている場合はセッション終了時に VTAM によって
- IMS 管理類似性がセッションに対して確立されていて、かつ端末がローカル IMS (状況リカバリー・モードが LOCAL) でエンド・ユーザー有効状態でない場合は、セッション終了時に IMS によって

### 関連概念

[VTAM 汎用リソース・グループの計画 \(システム管理\)](#)

## IMSplex 端末管理

リソース・マネージャー (RM) およびリソース構造を使用することでユーザーの機能が拡張され、IMSplex 全体に渡る TM リソースの管理、および DB/DC または DCCTL 環境内の端末関連情報の共用が可能になります。

共通サービス層 (CSL)、特に RM によって操作している場合、リソース構造は、IMSplex 全体に渡るリソースを統合的に定義および保守する方法を提供します。IMSplex 全体のリソース情報を共用することで、端末およびユーザーに対する透過性および状態を取得することができます。

このトピックでは、RM およびリソース構造を使用して TM リソースを管理することの利点に関する概要を説明します。

注：リソース構造を使用せずに IMSplex を定義した場合は、端末管理は、ローカル・モードのみで稼働する IMS システムの場合と同じです。

### 関連概念

[CSL RM の管理 \(システム管理\)](#)

[CSL の管理 \(システム管理\)](#)

[IMSplex 内のトランザクション・マネージャーのリソースの計画 \(システム管理\)](#)

### 関連資料

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

## リソース構造によるリソースの管理の利点

リソース・マネージャー (RM) およびリソース構造を使用して IMS TM リソースを管理すると、さまざまな利点があります。

RM およびリソース構造を使用する利点には、以下のものがあります。

### リソース・タイプ整合性の制約

同じ名前が複数のリソース・タイプに対して使用されることを防止します。

### リソース名の一意性の制約

任意のリソースは IMSplex 内で一回のみアクティブになることを保証します。

### グローバル呼び出し可能サービス

出口ルーチンが LTERM、ノード、および IMSplex 全体のユーザー・リソース 情報を取得することを可能にします。

### 端末およびユーザー状況リカバリー

端末およびユーザーがどの IMS に対しても類似性を持っていなくても作業を再開できるようにします。

### VTAM で汎用リソース類似性を管理可能

GLOBAL を状況リカバリー・モードとして指定すると、IMS の代わりに VTAM で 汎用リソース類似性を管理することができます。IMS システムが障害を起こした場合、たとえ端末装置が障害を起こした IMS システムへの類似性を持っていたとしても、VTAM はその端末装置を別の IMS システムに再割り当てすることができます。

### IMS コールド・スタートをまたいだ、グローバル IMS 機能のイネーブルメント値のリトリブ

**UPDATE IMSFUNC** コマンドを使用してグローバルに IMS 機能を使用可能にし、RM およびリソース構造を使用する場合、機能イネーブルメント値はリソース構造に保管され、IMS コールド・スタートをまたいで値をリトリブすることができるようになります。

### 関連概念

[RM およびリソース構造の IMS アクティビティへの影響 \(システム管理\)](#)

### 関連資料

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

## 共用 TM リソース

リソース構造がリソース・マネージャー (RM) を使用する IMSplex 内にあるときは、TM リソース共用は自動的に行われます。

TM リソースを共用する場合、以下の点が適用されます。

- リソース構造に接続されている各 IMS は、リソース構造に接続されている他のすべての IMS システムの TM リソースを使用することができる。
- RM は、リソース構造に接続されている IMS システムの間でリソース名の一意性を強制する。
- RM は、リソース構造に接続されている IMS システムの間でリソース・タイプ整合性を強制する。

リソース構造と IMSplex 内の IMS システム間の接続をすべて維持している際に、TM リソース共用を使用不可にする必要がある場合は、DFSDCxxx PROCLIB メンバー内で STM=NO を指定してください。

### 関連資料

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

## リソース名の一意性

リソース・マネージャー (RM) がアクティブであり、リソース構造が定義されている場合、IMS は自動的に特定の TM リソースに対して名前の一意性を強制します。

IMS が名前の一意性を強制する TM リソースには、以下のものがあります。

- VTAM LTERM
- VTAM 単一セッション・ノード

- ユーザー ID
- ユーザー

DFSDCxxx PROCLIB メンバーで STM=NO と指定することによって、リソース名の制約、および一般には、TM リソース共用を使用不可に設定できます。

#### 関連資料

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

## リソース・タイプ整合性

RM がアクティブでリソース構造がカップリング・ファシリティ内で定義されている場合、IMS は TM リソースに対するタイプの整合性を自動的に制約します。

IMS は以下のメッセージ宛先に対するリソースのタイプを確認します。

- LTERM
- APPC 記述子として定義された LTERM
- MSNAME
- CPI-C トランザクション
- 非 CPI-C トランザクション

リソース構造がある場合に、DFSDCxxx PROCLIB メンバー内で STM=NO と指定することによって、リソース・タイプ整合性の制約を使用不可にすることができます。STM=NO と指定すると、リソース・タイプ整合性は、非 CPI-C トランザクションおよび CPI-C トランザクションのみに強制されます。

#### 関連資料

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

## 高速機能急送メッセージ・ハンドラー

このトピックでは、高速機能急送メッセージ・ハンドラー (EMH) およびこのハンドラーが高速機能メッセージを処理する方法について簡単に説明します。

### 高速機能メッセージ・スケジューリング

高速機能は、入力メッセージをロード・バランシング・グループに関連付けることによってスケジュールします。

**定義:** ロード・バランシング・グループは、高速機能プログラムの 1 つ以上のコピーによるロード・バランシングが作動可能な高速機能入力メッセージのグループです。それぞれの固有の高速機能メッセージ・ドリブン・アプリケーション・プログラムごとに 1 つのロード・バランシング・グループが存在します。

各ロード・バランシング・グループのメッセージは、同じ高速機能プログラムで処理されます。EMH は、高速機能メッセージを以下のようにして制御します。

- メッセージの完全な実行を先入れ先出し式で管理する
- 補助記憶機構や入出力操作を使用せずに、制御プログラム・ストレージに受信したメッセージを保持する
- 並列スケジューリングのために、複数のプログラムのコピーをサポートする
- 入力待ちモードで動くプログラムを必要とする

### 宛先コードと平衡化グループ

IMS は、入力メッセージをメッセージ・キューに置く場合、トランザクション・コードを使用して、システム定義の範囲内で 1 つのキューがなくなるまで、1 つずつ処理しようとしています。メッセージ処理は、負荷平衡化のためにグループ化され、データベースの保全性とリカバリーのために同期化されます。

#### 定義:



- メッセージ・キューは、メッセージを、アプリケーション・プログラムによる処理または端末への送信前に置くデータ・セットです。
- トランザクション・コードは、IMS メッセージ処理プログラムを呼び出す 1 文字から 8 文字の英数字コードです
- 宛先コードは、EMH がトランザクションをロード・バランシング・グループ内のプログラムに経路指定するために使用するユーザー定義コードです。宛先コードによって平衡化グループに、メッセージを割り当てます。宛先コードは、トランザクション・コードと同じにするか、または、他の名前を選択することもできます。

着信メッセージは、事前に割り当てられた宛先コードを持っているか、または、入力メッセージに対して後で動的にセットされます。宛先コードはシステム定義時に宣言してから、各アプリケーション・プログラムに関連付けます。

## 高速機能入力編集/経路指定出口ルーチン (DBFHAGU0)

高速機能専用のメッセージとその可能性のあるメッセージはすべて、高速機能入力編集/経路指定出口ルーチンに送られます。したがって、アプリケーション・プログラムでは、入力域内の入力メッセージを検査、変更、または編集することができます。

## 高速機能可能システムによるメッセージ・バッファ

IMS バッファ管理は、入力メッセージが以下のすべての特性を持っている場合、単一の EMH バッファを使用します。

- ソース端末は VTAM
- 端末はフロントエンド切り替え (FES) が可能でない
- 入力は単一セグメント
- 入力が MFS で編集されていない
- 入力が IMS コマンドでない
- 最低 1 トランザクションは高速機能で実行

IMS が上記のすべての特性を持った最初のメッセージを受信すると、急送メッセージ・ハンドラー (EMH) バッファがその端末に動的に割り振られます。ユーザーがサインオフしたり、セッションが終了するまで、EMH バッファは、それに続く同じ特性を持った入力メッセージのために割り振られたままになっています。その後バッファは、EMH プールにリリースされます。

## 高速機能 EMH と共用キュー

高速機能が組み込まれた共用キュー環境では、EMH キュー (EMHQ) を使用して高速機能 EMH メッセージを共用するオプションがあります。EMHQ は、EMH メッセージの処理を複数の IMS システムに分散します。

EMH メッセージを共用しない場合は、EMHQ ステートメントを DFSSQxxx PROCLIB メンバーから削除して、IMS が EMHQ 構造とデータ・セットを割り振りしないようにできます。

## EMH を使用する高速機能アプリケーション・プログラムの基準

EMH を使用するアプリケーション・プログラムは、以下のすべての基準を満たす必要があります。

- 各トランザクションが、単一のセグメントの応答タイプ・メッセージで始まっている。
- 各入力メッセージはすべて、セグメントの応答メッセージを 1 つだけ必要とするかまたは、応答を必要としない。
- 入出力メッセージ長は、EMH バッファ・サイズを超えない。

- ETO ユーザーと LU 6.2 ユーザーから、端末関連キーを使用して 主記憶データベース (MSDB) にアクセスできない。
- IMS 会話型処理が実行されない。
- 高速機能プログラムは自動化操作プログラムの役割をしたり IMS コマンドを出す処理はできない。
- 他の IMS での処理のために、高速機能トランザクションを、複数システム結合機能 (MSC) 物理リンクの 4 つのどのタイプにも送ることはできない。

#### **関連概念**

[共用キューの使用可能化 \(システム管理\)](#)

[高速機能 \(システム管理\)](#)

#### **関連資料**

[高速機能入力編集/経路指定出口ルーチン \(DBFHAGU0\) \(出口ルーチン\)](#)



## 第 23 章 ネットワークの設計

以下のトピックでは、IMS ネットワークを設計するための情報を記載します。

### ネットワーク管理の設計

IMS ネットワーク管理者は、以下の表にリストされた各アクティビティについて計画する必要があります。この表には、各アクティビティに関する追加情報の検索先も記載されています。

#### このタスクについて

表 71. ネットワーク管理アクティビティ

アクティビティ	関連情報
オンライン・システムに必要な端末と他の装置を識別し、これらの装置を IMS がサポートすることを確認し、非互換性があればそれを識別する。	IMS がサポートする端末装置、端末ごとに IMS がサポートする通信モード、および各端末、制御装置、または CPC ごとの機能については、「 <a href="#">IMS 15 でサポートされる端末および装置 (リリース計画)</a> 」を参照してください。
ネットワーク内で適切なセキュリティ機能を設定する。	セキュリティの設定については、 <a href="#">412 ページの『セキュリティ設計』</a> を参照してください。
メッセージ編集要件を判別する。	メッセージの編集およびフォーマット設定については、 <a href="#">435 ページの『第 25 章 IMS メッセージの編集とフォーマット設定』</a> を参照してください。
出口ルーチンの要件を判別する。	IMS ネットワークをカスタマイズするための出口ルーチンの使用方法については、「 <a href="#">出口ルーチン (出口ルーチン)</a> 」を参照してください。
適切な命名規則を実施する。	命名規則の設定については、 <a href="#">命名規則の設定 (システム管理)</a> を参照してください。
IMS システム定義でネットワークを定義し、システム定義をホストに合わせて調整する。	IMS ネットワーク・システム定義については、「 <a href="#">データ通信マクロによる端末の定義 (システム定義)</a> 」を参照してください。
ETO 記述子と MFS 装置特性テーブル項目を定義する。	ETO および MFS 装置特性テーブルについては、 <a href="#">71 ページの『第 5 章 拡張端末オプションの管理』</a> を参照してください。
LU 6.2 記述子を APPC/IMS に定義する。	APPC/IMS および LU 6.2 については、 <a href="#">33 ページの『第 3 章 APPC/IMS および LU 6.2 装置の管理』</a> を参照してください。
ネットワークの作動可能状況をテストする。	ネットワークのテストについては、 <a href="#">ネットワークの作動可能性の確認 (システム管理)</a> を参照してください。
ネットワークのパフォーマンスをモニターし、トランザクション経路指定やメッセージ・キュー・サイズなどのメッセージの負荷を分析する。	パフォーマンスのモニターについては、 <a href="#">容量計画の実行 (システム管理)</a> を参照してください。

## ネットワークと端末要件の文書化

IMS インストール・システムと端末プロファイルのサイズによって、ネットワーク要件を文書化する方法が決まります。

### このタスクについて

#### 端末プロファイル

システム定義に必要な情報を収集する最も適切な方法は、端末プロファイルの作成です。端末プロファイルは、インストール要件の調整を行う場合にも役立ちます。これらのプロファイルによって、システム・プログラム担当者に対し、各端末をどのように使用するかを示すことができます。

アプリケーション・プログラムの仕様またはハードウェア計画を検討して、使用する端末を確認してください。

#### VTAM を使用して接続されている端末

VTAM で接続されている端末は端末タイプにより定義されています。通信および接続のモードには以下のものがあります。

- 2進データ同期通信 (BSC)
- 同期データ・リンク制御 (SDLC)
- ローカル接続 (チャンネルに直接接続)

IMS を定義するか、ETO ログオン記述子を選択すると、選択した端末タイプは、IMS が使用する通信プロトコルと MFS フォーマット設定を判別します。特定の物理端末に複数の端末タイプを使用することができます。ただし、選択は IMS が実行するアプリケーション・プログラムと互換性を持つ必要があります。

## IMS 端末ネットワーク

IMS ネットワークは、物理端末と他の装置によって特徴付けられます。IMS はディスプレイ装置、プリンター、およびリモート・インテリジェント・コントローラーを含む、多くの端末をサポートします。

IMS は VTAM のようなアクセス方式を使用して物理端末と通信します。以下の相違について理解する必要があります。

- ホストに接続されている端末に関する VTAM の視点
- これらの接続に関する IMS の監視

VTAM の責任には以下のものが含まれます。

- 入力メッセージの受信と処理
- データの伝送
- 回線と端末の通信状況の管理

IMS の責任には以下のものが含まれます。

- メッセージ・トラフィックのモニター
- アプリケーション・プログラムへのメッセージの経路指定
- メッセージの保管、スケジューリング、および記録の処理

IMS はネットワーク機能である VTAM を使用しますが、BSAM を使用する装置も制御します。

IMS の場合、優先アクセス方式は VTAM です。非 VTAM ネットワークには OTMA を使用することができます。

物理端末構成を IMS に示すために、端末接続およびその特性 (通信タイプ、機能、オプションなど) について記述する TYPE マクロなどのシステム定義マクロを作成します。ETO および LU 6.2 記述子も定義します。

## IMS への端末接続

IMS は、さまざまなモードの通信あるいは接続機構を、通常は VTAM を使用してサポートします。リモート接続機構は交換回線か非交換回線のいずれかです。

VTAM で接続されている端末には、端末タイプが割り当てられます。IMS が使用する回線と端末のアドレスは、IMS システム定義で指定されていないので、IMS にとって透過的です。その代わりに、すべての VTAM リソースには名前が割り当てられます。

**定義:** 各論理装置には、ご使用のシステムで定義したノード名が割り当てられます。

VTAM 定義のため選択されたノード名は、IMS システム定義とコマンドでも使用されます。VTAM を使用すると、IMS ユーザーは他の VTAM アプリケーションを使用するネットワーク・リソースを共用できます。

## XRF 端末サポート

XRF は、IMS がサポートするすべての端末をサポートします。XRF 端末サポートは、3つのカテゴリー（クラス 1、クラス 2、およびクラス 3）に分割されます。

### 関連概念

[418 ページの『XRF 複合システムの端末』](#)

XRF は、IMS ユーザーが現在使用しているすべての端末をサポートします。XRF は、ユーザーの最小の関与で、緊急サポートを提供します。

[拡張回復機能の概要 \(システム管理\)](#)

## 論理端末 (LTERM)

論理端末 (*LTERM*) はユーザーの宛先です。静的に定義された端末の場合、1つの LTERM は1つの物理端末に関連付けられます。ETO 端末の場合、1つの LTERM は1人のユーザーに関連付けられ、ユーザーが物理端末にサインオンした後でのみ、物理端末に関連付けられます。

各論理端末はインストール時に定義した名前があり、*LTERM* 名と呼ばれます。

LTERM を1つ予約して、それを IMS のマスター端末としてください。この論理端末は、オンライン IMS の制御点になります。LTERM に関連するコマンドは、システムの開始と停止、システム・リソースの制御を行い、制御しているシステム・リソースを表示します。/ASSIGN コマンドによって、マスター端末オペレーター (MTO) は、LTERM に関連する物理装置を変更することができます。ETO ユーザーがサインオンすると、LTERM に関連する物理装置は自動的に更新されます。装置は、ノード名、あるいは回線番号と物理端末 ID によって識別されます。

トランザクションを入力すると、論理端末名は入力メッセージと関連付けられ、入力メッセージのトランザクション・コードによりキューに入ります。出力メッセージのキュー名は、実際には、LTERM 名そのものです。この名前は、しばしば入力用と出力用が同じですが、プログラムは、出力を代替 LTERM に行わせる場合があります。入力端末では、メッセージの宛先として LTERM 名を指定することもできます。メッセージ通信 (入力) 編集ルーチンは、入力している LTERM 名をメッセージに追加することができます。

### 関連概念

[401 ページの『論理端末ネットワークの設計』](#)

IMS システム定義は、通信回線、静的端末、および論理端末 (LTERM) の特性と関係を記述します。

### 関連資料

[トランザクション・マネージャー 出口ルーチン \(出口ルーチン\)](#)

## APPC/IMS と LU 6.2 の端末サポート

APPC/IMS は入力に LTERM を使用しません。APPC/IMS は、出力宛先の判別で使用される、LTERM に似た2つの機能 (LU 6.2 記述子とサイド情報) を備えています。

LU 6.2 記述子は IMS LTERM 名が LU 6.2 の宛先 (LUNAME、TPNAME、その他 LU 6.2 宛先の特性の指定) を定義できるようにしています。これらの LU 6.2 LTERM は、IMS アプリケーション・プログラムで使用されている通常の LTERM と同じ方法で使用されます。

サイド情報には、CPI コミュニケーション (CPI-C) によって提供されるシステム定義値が含まれており、LU 6.2 宛先情報を提供します。APPC/IMS がパートナー・プログラムを使用して会話を設定するには、CPI-C は、パートナー・ノードにおけるパートナー・プログラムと LU 名のような初期設定情報が必要です。CPI-C は、これらの必須フィールドにシステム定義値を使用する方法を提供します。これらのシステム定義値はサイド情報に指定され、シンボリック宛先名によってアクセスされます。シンボリック宛先は、partner\_LU\_name、mode\_name、TP\_name を含むサイド情報にある項目に相当します。

**関連資料:** サイド情報、シンボリック宛先名、およびサイド情報内の項目については、「z/OS MVS 計画: APPC/MVS 管理」を参照してください。

## IMS メッセージとスケジューリング

IMS メッセージのタイプは、IMS がプロセスを制御する 4 つのタイプのデータ通信 (トランザクション、LTERM へ送信されるメッセージ、IMS コマンド、DFSAPPC) のうちの 1 つです。

### トランザクション

トランザクションはアプリケーション・プログラムによって処理される予定の入力メッセージです。トランザクションは、1 文字から 8 文字のトランザクション・コードによって識別されます。トランザクションは端末から入力するか、アプリケーション・プログラムによって生成されます。複数システム環境では、トランザクションはリモート IMS から、または別のサブシステム (Db2 for z/OS など) から発信することができます。

### LTERM へ送信されるメッセージ

LTERM へ送信されるメッセージは LTERM 名によって識別されます。これらのメッセージのタイプは以下のいずれかです。

- ・ 論理端末と通信するアプリケーション・プログラムから送信される出力メッセージ。出力メッセージは通常作業結果の受信確認をするか出すかして、もとの端末に送られます。アプリケーション・プログラムは、入力メッセージを生成した論理端末以外の論理端末にも出力を送付します。
- ・ 宛先が論理端末の入力メッセージ。このメッセージ・タイプは、通常 端末間メッセージ通信をしています。メッセージのテキストは、宛先の LTERM への出力メッセージになります。

### IMS コマンド

IMS コマンドは、IMS に表示を要求するか、1 つ以上の IMS リソースの状況を更新する端末ユーザーによって入力されます。ほとんどのコマンドはマスター端末オペレーター (MTO) から入力しますが、アプリケーション・プログラムと同様に、他の端末からも入力できます。

### DFSAPPC

DFSAPPC は IMS サービスで、LU 6.2 装置間、または LU 6.2 と非 LU 6.2 装置の組み合わせにおいてもメッセージの交換を行うために使用されます。メッセージは、指定された LU 6.2 の宛先装置との新しい会話を割り振ることにより送達されます。割り振られた会話が失敗すると、出力は将来の送達に備えて格納されます。メッセージは、正常に送達されるまで IMS メッセージ・キューに保留されます。

## メッセージ・キューイング

共用キューを使用しない IMS システムでは、すべての入力メッセージと出力メッセージは IMS 制御の仮想記憶域にキューとして入れられます。ほとんどのコマンド出力は仮想記憶域のキューにあります。TRANSACT マクロで SERIAL=YES が指定された場合、アプリケーション・プログラムはメッセージを順次処理します。

仮想記憶域が制限を超えた場合、メッセージは直接アクセス記憶域に保管されます。このようにしてメッセージは、処理するのに必要なリソースがすぐには利用できなくても、入力用として受信されるか、あるいは出力用として保管できます。

端末の並行入出力要求の最大数とメッセージを保留する仮想記憶域の容量は IMS システム定義時に指定されます。どちらの変数も IMS の開始時に変更可能です。

共用キュー環境では、入力メッセージと出力メッセージはカップリング・ファシリティ内のキュー構造へキューとして入れられます。共用キュー環境については、「IMS V15 システム管理」に説明があります。

## メッセージ・セグメント

端末が IMS オンライン・システムにデータを入力するのに使用される場合、入力は単一あるいは複数のセグメントのいずれかで構成されます。セグメントの順序で、アプリケーション・プログラムで解釈されるメッセージを指定する必要があります。装置の操作はセグメントの終了を判別します。

入力の場合については、IMS によるセグメントの終了条件を検出して、メッセージが完了したことを示します。

最初のセグメント終了条件が検出されると、IMS はそのセグメントの始まりを調べて、宛先を検索します。宛先が単一セグメント・メッセージの場合、セグメントの終了がメッセージの終了です。

編集の後、メッセージの最初 (あるいはただ 1 つ) のセグメントは以下のような構造を持ちます。

<b>LL</b>	<b>ZZ</b>	<b>DEST-CODE</b>	<b>b</b>	<b>MMMM</b>
2 bytes	2 bytes	1 to 8 bytes	1 byte	message text

これらのパラメーターについて、次の表で説明します。

表 72. メッセージ・セグメントのフォーマット

メッセージ・セグメントの部分	Bytes	説明
LL	2	LL および ZZ 部分を含む、セグメントの全長
ZZ	2	予約済みハーフワード
DEST-CODE	1 から 8	宛先コード (通常トランザクション・コード)
MMMM	さまざま	メッセージ・テキスト

各入力メッセージは、宛先コードによって固有に識別されます。宛先は、通常トランザクション・コードですが、メッセージ通信の場合、IMS コマンドあるいは LTERM になることもあります。

### 無効な宛先

宛先コードが通常のリソースの命名規則に従ったとしても、入力した宛先が誤っている場合があります。

- ETO がアクティブでない場合、IMS が識別しない宛先は無効と見なされ、入力メッセージは受け入れられません。
- ETO がアクティブの場合、IMS はこれらの宛先が LTERM 名であると想定し、動的ユーザー構造を作成します。宛先が無効な場合、特定の送達不能キューが作成されます。MTO が送達不能キューを処理するための、/DISPLAY コマンドを含む、複数のコマンドが用意されています。

### トランザクション・コード

トランザクション・コードには IMS 制御プログラムによって プロセス・プログラムのスケジューリングの適格性に影響を及ぼす任意の属性があります。

アプリケーション・プログラムは関連マクロ命令以外は別途定義されます。特定のトランザクション・コードを処理する目的で指定したアプリケーション・プログラムは、IMS によって、別のトランザクション属性であると見なされます。アプリケーション・プログラムは、複数のトランザクション・コードを処理しますが、1 つのトランザクション・コードは、1 つのプログラムだけと関連しています。

### メッセージ・スケジューリング

**定義:** メッセージ・スケジューリング は、完全に受信した入力トランザクションを関連するアプリケーション・プログラムと結び付けて処理するプロセスです。

メッセージ・スケジューリング・プロセスに影響を与える要素には次のものがあります。

- トランザクション・コードに関連する可変属性
- その他のトランザクション・コードの数とその相対的重要性
- 受信したが未処理のメッセージの数
- 処理されるデータに対する関連アプリケーション・プログラムの目的
- 制御ブロックのストレージ・プールとバッファにおける使用可能スペースの量

さらに、以下の各アクティビティーがメッセージ・スケジューリング・プロセスに影響を与えます。

- システム定義オプションの選択
- データベースの設計および使用
- バッファ・サイズの指定
- トランザクション・コードの優先順位の宣言および選択

上記のアクティビティーを適切に組み合わせることによって、メッセージの処理順序を制御し、システム・パフォーマンスを向上することができます。

## CPI-C トランザクション

CPI 通信ドリブン・アプリケーション・プログラムは、APSB (Allocate\_PSB) 呼び出しを出すと、IMS リソースを使用できるようになります。CPI 通信ドリブン・アプリケーション・プログラムでは、SAA リソース・リカバリー・コミット (SRRCMIT) およびバックアウト (SRRBACK) 呼び出しを使用して、IMS 同期点またはバックアウトを開始できます。CPI 通信ドリブン・アプリケーション・プログラムは、プログラムの終了の前に SAA リソース・リカバリー呼び出しを使用して、IMS 同期点を開始する必要があります。

**定義:** 同期点 (*synchronization point*) (本書全体を通じて同期点 (*sync point*) とも呼ばれる) とは、プログラムまたはサブシステム内でのオカレンスであり、ここでリソースのコミットや後続の再始動のための参照点の設定が必要に応じて行われます。

**推奨:** 非 CPI トランザクションに使用されるメッセージ・クラスとは別のメッセージ・クラスを使用して、CPI トランザクションを定義してください。

### 関連概念

[DFSAPPC メッセージ通信 \(アプリケーション・プログラミング\)](#)

[容量計画の実行 \(システム管理\)](#)

### 関連タスク

[108 ページの『無効な宛先への非同期出力』](#)

IMS では、送達できないデータを「送達不能」と言います。

[23 ページの『CPI 通信および APPC/IMS』](#)

これらのトピックでは、CPI 通信と APPC/IMS を紹介します。このトピックでは、CPI 通信ドリブンのアプリケーション・プログラムの機能する方法、APPC/IMS を管理し、CPI アプリケーション・プログラムを構築するために CPI 通信インターフェース付きの APPC/IMS を使用する方法を説明します。

## IMS オンライン・システム内のメッセージ・フロー

このトピックでは、ある端末で入力されたメッセージが IMS オンライン・システムを通して流れる時に起こるイベントを説明します。イベントの順番あるいはプロセスには例外もありますが、大多数のトランザクションは以下の方法で処理されます。

IMS では、基本編集 (デフォルト) と MFS という 2 つの方法でプロセスをカスタマイズできます。

### 端末からプログラムへのメッセージ・フロー



端末からの最初の入力 は IMS で非同期に処理されるメッセージです。端末からメッセージを受信した後、そのメッセージは、基本編集、MFS、または出口ルーチンを使用してフォーマット設定されます。

IMS メッセージ・スケジューラーはメッセージ・キューの状況と優先順位付けアルゴリズムを使用して、スケジュールすべき次のトランザクションを選択します。従属領域にスケジューリングするためにプログラムを選択すると、メッセージの最初のセグメントがプログラムに対して使用可能になります。

## プログラムから出力端末へのメッセージ・フロー

アプリケーション・プログラムは、実行中に、DL/I 呼び出しを使用してデータベースへアクセス できます。DL/I のデータ通信サポートは、最初のメッセージ・セグメントに対して GU 機能を、その後のメッセージ・セグメント・リトリブに GN 機能を使用して、メッセージを要求するためプログラムで使用 されます。プログラムは、入出力 PCB に応答を挿入して入力している端末に送信します。入力している 端末の LTERM は、正常な GU 呼び出し後の入出力 PCB です。

メッセージが MFS で処理される場合、メッセージはプログラム出力形式から装置出力形式に変換 されます。他の方法では、メッセージは変換されず装置に渡されます。

端末装置あるいはプログラムがすべて出力メッセージを受け取るとキューが解除されます。メッ セージ送達で起こるすべての障害では、IMS が後の送達に使用するためリカバリー可能メッセー ジ・キューとして保持できるようにします。

## プログラムから代替宛先へのメッセージ・フロー

アプリケーション・プログラムは出力を代替宛先に送信することもできます。代替 PCB を使用すると プログラムは、他の LTERM にメッセージを挿入することができます。この場合、メッセージ処理は同じで す。メッセージの置かれるキューは別の LTERM 名を持ちます。

## プログラムからプログラムへのメッセージ・フロー

アプリケーション・プログラムは代替 PCB を使用して、プログラムにメッセージを送信すること もできま す。つまり、メッセージ・キューに置かれる 2 次トランザクションを生成することができます。2 次ト ランザクションの処理は、スケジューリング・アルゴリズムが選択するトランザクションによります。

### 関連タスク

435 ページの『IMS メッセージの編集とフォーマット設定』

IMS、メッセージ形式サービス (MFS) および基本編集ルーチンの 2 つの方法によって、端末でやり取りす るメッセージの編集およびフォーマット設定を行います。

## 会話型トランザクション

会話型トランザクション処理を使用すると、会話を処理するプログラムがその会話中にストレージ内に保 持されていなくても、特定の端末からのメッセージの連続性を保つことができます。

トランザクションが会話型として定義されると、アプリケーション・プログラムはスクラッチパッド域 (SPA) を使用することによって、特定の端末装置からのメッセージを内部的に関連付けることができます。

スクラッチパッド域 (SPA) は、アプリケーション・プログラムからの情報をプログラムの実行間で保持する ために、会話型処理で使用される作業域です。固有な SPA が会話ごとに作成されます。

### SPA の内容

典型的な SPA の内容は会話のやり取りの間、保管する必要がある、端末装置から入力されたデータ項目と データベースからリトリブされるデータです。オペレーター・プロシージャとアプリケーションの設 計を単純化するために、データベースの更新は、会話の最後のやり取りまで SPA に保持されます。

会話型モードで操作中の端末からの後続のデータ入力によって、メッセージ処理プログラム (MPP) は、SPA と入力端末データの両方の内容を受信します。各入力メッセージは、個々の作業単位として見なされます。

一般的に、同一プログラムが使用されたとしても、各対話は異なるプログラムを使用する可能性があります。

## 会話型トランザクションのメッセージ処理

メッセージが会話型トランザクションである場合、以下の一連のイベントが発生します。

- IMS は、トランザクション・コードを除去し、それをメッセージ・セグメントの先頭に置きます。メッセージ・セグメントの長さは、システム定義でこのトランザクションに関して定義されている SPA の長さと同じです。これは、プログラムで使用できる入力メッセージの最初のセグメントです。端末から取得した 2 から  $n$  番目のセグメントから、トランザクション・コードを削除したものが、アプリケーション・プログラムに渡されるメッセージの残りとなります。
- 会話型プログラムは、応答を作成すると、IMS に SPA を挿入します。次に、プログラムは応答の実際のテキストを、出力メッセージのセグメントとして挿入します。
- IMS は、SPA を保管し、メッセージを入力 LTERM に送ります。
- SPA の挿入によって、別のプログラムが同じ会話を継続することを指定した場合、すべての応答 (SPA を含む) は、次のプログラムへの入力としてメッセージ・キューに保持されます。次に、このプログラムは類似の形式のメッセージを受信します。
- 会話型プログラムを入力交換ごとにスケジューリングする必要がある。その他のプロセスは入力端末のオペレーターが応答を調べて新規入力メッセージを作成している間も、継続することに注意してください。
- 会話を終了するために、プログラムは SPA のトランザクション・コード・フィールドにブランクを置き、その SPA を IMS に渡します。
- SPA のトランザクション・コードが非会話型プログラムのトランザクション・コードによって置き換えられ、その SPA が IMS に渡された場合も、会話は終了します。次の端末入力後、IMS はそのメッセージを通常の方法で別のプログラムのキューへ送付します。

## ETO 会話

ETO 会話は、端末ではなくユーザーとの関連を保持します。会話はユーザーがサインオンしている間、端末と関連付けられます。会話は他のどの ETO 端末からも再始動できます。

## 会話型プログラムのプログラム間通信

会話型プログラムは、据え置きプログラム間通信または即時プログラム間通信を使用することができます。

### 定義:

- 据え置きプログラム間通信では、プログラムは発信元端末に回答しますが、その端末からの次の入力を、別の会話型プログラムに渡します。
- 即時プログラム間通信では、プログラムは、発信元端末に回答せずに、SPA (およびオプションでメッセージ) を別の会話型プログラムに渡します。この場合、発信元端末への回答は、次のプログラムの責任です。

### 関連タスク

61 ページの『[拡張端末オプション \(ETO\) \(Extended Terminal Option \(ETO\)\)](#)』



これらのトピックでは、拡張端末オプション (ETO) を紹介し、ETO の概要 および IMS TM ネットワーク内の ETO 端末の 管理に必要な情報を説明します。

## メッセージ通信

メッセージ通信が行われるのは、入力メッセージでトランザクション・コードの代わりに LTERM 名が指定された場合です。IMS はメッセージをフォーマット設定し、指定された LTERM の出力メッセージ・キューに置きます。

## 論理端末ネットワークの設計

IMS システム定義は、通信回線、静的端末、および論理端末 (LTERM) の特性と関係を記述します。

非交換端末の場合、IMS 内での物理端末と論理端末の関係は、システム定義時に定義される静的な関係です。

1 人のユーザーが 1 つの物理端末を操作している場合、その物理端末に関連付けられるのは、1 つの論理端末だけです。複数のユーザーが 1 つの物理端末を操作している場合、多くの論理端末がその物理端末に関連付けられます。

特定の静的端末のユーザーごとに、別個の論理端末が割り当てられるように、IMS システム定義を構築することができます。

**定義:** このトピックの情報では、物理端末 という語を使用して、VTAM 端末のノードを表します。

論理端末は、入出力用に物理端末に割り当てることができます。1 つの論理端末が出力用として 1 つの物理端末に割り当てられると、その論理端末へ送信されたすべてのメッセージは関連する物理端末へ伝送されます。複数の論理端末が出力用として特定の物理端末に割り当てられます。1 つだけの物理端末が特定の論理端末への出力を受信することができます。以下の図は、出力に関する物理端末と論理端末の関係を示しています。

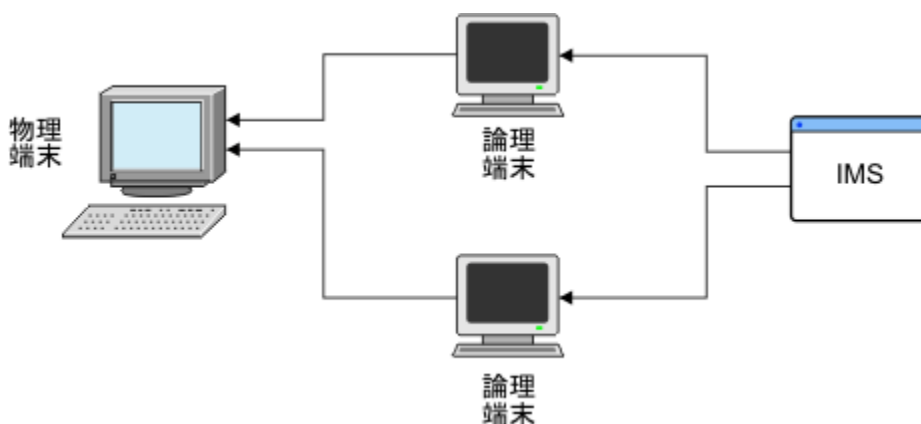


図 45. IMS からの出力に関する物理端末と論理端末の関係

## 論理端末チェーン

論理端末が入力用として物理端末に割り当てられると、物理端末から入力されるメッセージはすべてその論理端末で発信されたと見なされます。複数の論理端末が入力用として物理端末に割り当てられると、入力の論理端末チェーンが形成されます。物理端末からの入力は、すべてチェーン上の最初の論理端末が起点であると見なされます。

以下の図は、入力に関する物理端末と論理端末の関係を示しています。

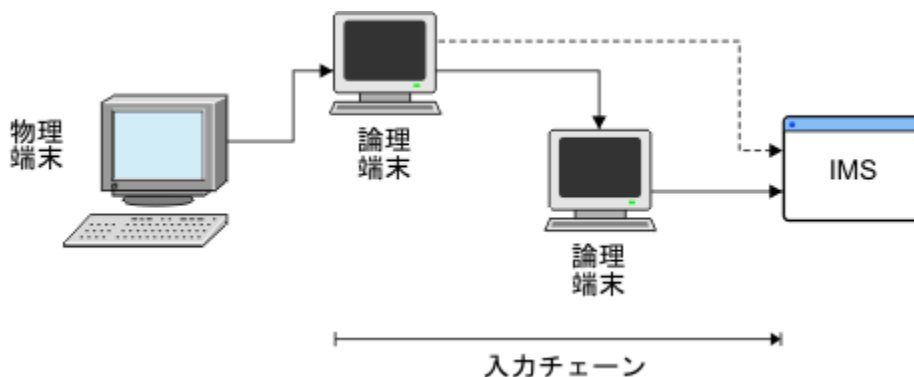


図 46. IMS への入力に関する物理端末と論理端末の関係

最初の論理端末がメッセージを送信できなかった場合（許可されていない場合あるいは停止された論理端末である場合）、入力チェーンにあるすべての論理端末は、メッセージを送信できるかどうか、チェーンの順序で調べられます。物理端末が金融機関、SLU 1、SLU P、または LU タイプ 6.1 の場合、入力コンポーネントに関連した論理端末だけがスキャンされます。最初に検索された論理端末がメッセージの発信元と見なされます。適切な論理端末が見つからない場合は、メッセージはエラー・メッセージ付きで拒否されます。

## 論理端末キュー

受信済み入力メッセージまたは保留中の出力メッセージに関してメッセージ・キューを使用すると、アプリケーション・プログラムを、到着時間およびメッセージ伝送から独立させることができます。このメッセージ・キューは、物理端末ではなく論理端末に関連付けられているので、アプリケーション・プログラムと無関係に、装置間を移動することができます。このメッセージ・キューは、装置クラス間も移動することができます。

論理端末は、アプリケーション・プログラムに安定性を提供します。物理端末ネットワークの変更が行われても、アプリケーションには影響がありません。

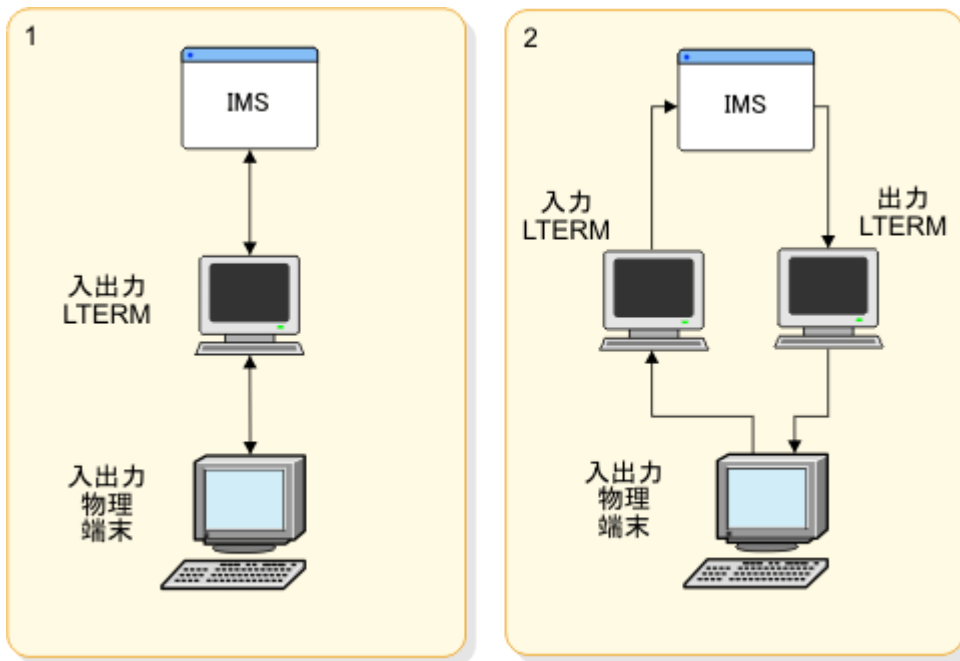
論理端末へのアプリケーション・プログラム・インターフェースは、概念的にデータベース・システムと同じです。

- GU 呼び出しは、キューからメッセージ・セグメントを取り出します。
- ISRT 呼び出しは、キューにメッセージ・セグメントを挿入します。

## 入出力装置の分離

特定のアプリケーション・プログラムにおいては、入力で使用された装置ではなく、異なる物理装置を出力用に関連付けることが必要になる場合があります。物理端末タイプが入力専用装置で、出力が必要な場合、別の装置を出力用に関連付ける必要があります。

IMS システム定義とコマンドは、入力装置とは異なる出力装置の割り当てをサポートします。例えば、表示装置からのメッセージを処理するアプリケーション・プログラムでは、ある種の出力をプリンターに送信しなければならないことがあります。考えられる物理端末と論理端末の関係を以下の図に示します。



#### アプリケーションに関する図の注

- 1つ以上の論理端末または物理端末の通常の割り当て。出力は入力端末に送信されます。アプリケーションは影響を受けません。
- アプリケーションは出力用の特定の論理端末を使用します。アプリケーションは入力端末の影響を受けません。

図 47. 考えられる物理端末と論理端末の関係

### 論理端末と物理端末の関係

以下の図は、非交換通信ネットワークでの端末ユーザー、物理端末、通信回線、論理端末の間の通信フローを示しています。



図 48. 非交換通信ネットワーク

IMS システム定義は物理端末、通信回線、および論理端末の特性と関係を記述しています。非交換回線において、一方の物理端末と他方の IMS にある論理端末の関係は、システム定義時または ETO 端末へのサインオン時に定義される継続的關係です。

VTAM を使用している端末 (LU 6.1 を除く) には、非交換回線の端末と物理端末 / 論理端末 (PTERM/LTERM) 関係があります。

システム定義時に物理端末と1つ以上の論理端末の間で設定した関係は、コマンドを使用するか、または新しいシステム定義を作成することによって変更することができます。**/ASSIGN** コマンドは、論理端末と物理端末の関係を動的に変更します。このコマンドは、通常、マスター端末からのみ実行することができます。

## マスター端末

マスター端末はIMSの制御センターです。VTAMマスター端末が定義されている場合、IMSは始動時にこのマスター端末と接続します。

**推奨事項:**いくつかの装置タイプがサポートされますが、マスター端末として望ましいタイプはVTAM SLU 2装置です。

**制約事項:**ETO 端末またはLU 6.2 端末のどちらも、基本マスター端末または2次マスター端末として定義することはできません。

マスター端末オペレーター (MTO) はシステム運用の局面をすべて理解する必要があります。コンピューター・コンソールとの関係でマスター端末の配置されている場所は重要です。MTOとマスター端末が、セキュリティの理由で近くに配置されない場合は、電話連絡ができるようにしておいてください。

## XRF 複合システムにおけるマスター端末

XRF 複合システムにおいては、各IMSはそれぞれでマスター端末を持つ必要があります。また2次マスター端末を持つことも可能です。

一般的に、2次マスター端末はプリンターのような出力専用の装置タイプです。アクティブおよび代替サブシステムのシステム・メッセージはそれぞれのマスター端末に行きます。

テークオーバーの間は、コンソールのオペレーターは両方のシステムのテークオーバーの状況を理解する必要があります。代替サブシステムのオペレーターは、障害を起こしたアクティブ・サブシステムの入出力防止がいつ完了するかを知る必要があります。これは、クラス3 端末の端末切り替えと、障害を起こしたクラス2の切り替えの試みを処理するためです。

### 関連概念

[IMS マスター端末の選択 \(システム定義\)](#)

## NTO 端末

NTO ライセンス製品を使用する場合、VTAMはNTO装置とされている特定のスタート・ストップ装置とテレタイプ装置をサポートすることができます。

メッセージの受信は、入力の後キーボードを「ロック」するトランザクション応答モードを使用して確認することができます。IMSが特定のトランザクションに対する出力を特定の端末に返すと、キーボードの「ロックが解除」されて、データが送信されます。

NTO装置が主に照会トランザクションを対象に使用されたとしても、トランザクションをリカバリー可能にすることができます。入力が確認応答されないため、アプリケーションに出力応答の原因になった照会トランザクションを表示させることが賢明です。

NTO装置を使用するトランザクションには、入力編集ルーチンと出力編集ルーチン (ご使用のシステムで提供され、MFS処理の後入力されます) の両方を使用可能にすることができます。

NTOはVTAMを使用しているため、IMSは端末ユーザーがセッションをログオンしているか、ログオフしているのか分かりません。したがって、ある1人の端末ユーザーがセッションを終了し、別の端末ユーザーがセッションを再開した場合に、既存のセッションの状況によって問題が生じる可能性があります。この問題は、すべてのNTO装置を応答モードに定義することによって最小にすることができます。

継続するセッション状況に関する問題を回避するために、端末ユーザーは、**/SET** コマンドを使用して以前に設定したトランザクション・コードを、以下のいずれかの時期に **/RESET** コマンドを入力してクリアすることができます。

- NTOセッションの終了時に **/RCLSDST** コマンドを出す前
- NTOセッションの開始時に **/SIGN ON** コマンドを出す前

**/RCLSDST** を入力する端末ユーザーかあるいは **/CLSDST** を入力する MTO はセッションを終了しますが、そのセッション中に設定された条件はリセットしません。

### 関連概念

[NTO 装置からの IMS との通信方法 \(オペレーションおよびオートメーション\)](#)

## リソース・モードおよび状態

IMS では、ネットワーク内の端末について、3 種類の状況情報を保持します。

IMS が保持する状況情報の種類は以下のとおりです。

- 操作のモード
- 動作不能な端末の状態 (一時的に使用不能など)
- リカバリー状況

### 端末とユーザーの操作モード

端末は、同時に複数のモードになる場合があります。

端末またはユーザーのモードには以下の種類があります。

#### 応答モード。

応答モードは、静的端末、ETO ユーザー、またはトランザクション仕様を通じて設定されます。応答モードは全機能あるいは高速機能処理で使用されます。応答モードでは、入力メッセージの入力後、端末は応答を受信するまでロックされ、その後の入力データは受け付けません。

#### 定義:

- 3270 のようなオペレーター・ドリブンの端末の場合でロック状態というのは、キーボードがロックされることです。
- LUP のようなプログラマブル端末の場合でロック状態というのは、IMS が、出力メッセージの送信が作動可能になるまで入力メッセージの確認応答を遅らせることです。

LU 6.2 の場合、トランザクションの発信元が応答メッセージ (同期出力) を受信するには、**Receive** コマンドを出す必要があります。そうしないとプロトコル違反として扱われます。システムによりですが、アプリケーション・プログラムが失敗して、IMS がエラー・メッセージを送信することがあります。

端末応答モードは、応答が送信され、キューが解除されると終了します。高速機能が使用されている場合、静的端末がログオフするとき、または ETO ユーザーがサインオフするときに、端末応答モードは IMS 障害後自動的に継続されます。全機能オペレーションが使用されている場合、端末応答モードは IMS が再始動したとき、静的端末がログオフしたとき、または ETO ユーザーがサインオフしたときに終了します。ただし、端末またはユーザーが SRMDEF=LOCAL と RCVYRESP=YES の両方を指定して定義されている場合は除きます。その場合、全機能応答モードは端末ログオフの後でも、ユーザー・サインオフの後でも、リカバリー可能です。

応答モードになるのは ETO ユーザーで、端末ではありません。ユーザーがサインオフし、かつ高速機能が使用されている場合、応答モードがそのユーザーに対してリカバリーされますが、端末は応答モードではありません。ユーザーは最初のトランザクションの応答を受信するまで、他のトランザクションを入力できません。

#### 会話モード

TRANSACT マクロで SPA= パラメーターを使用して定義したトランザクションを入力した後、端末は会話モードになります。端末が会話モードの場合、他のトランザクションをその端末から入力することはできません。しかし、この端末は、応答モードが端末をロックする場合と同じ意味でロックされているわけではありません。この端末は、会話が終了する時 (メッセージが送信され、キューが解除されて、次にアプリケーション・プログラムがスクラッチパッド 域 (SPA) のトランザクション・コード・フィールドをブランクにする時) まで会話モードのままです。

会話は、以下の条件で異常終了する可能性があります。

- アプリケーション・プログラムが異常終了した時。
- IMS MTO が /EXIT コマンド、/START NODE コマンド、あるいは /START USER を出した時。
- IMS と MSC 間の定義の矛盾がアプリケーション・プログラムにある時。



LU 6.2 の場合、IMS 会話のすべての反復では、同じ LU 6.2 の会話を使用する必要があります。なお、IMS 会話の各反復は、LU 6.2 CMPTR (Prepare\_To\_Receive) 呼び出しを使用して区切られます。LU 6.2 会話が、IMS 会話が終了する前に終了した場合、IMS 会話は異常終了します。

会話モードになるのは ETO ユーザーで、端末ではありません。

### 排他モード

端末は、/EXCLUSIVE コマンドが発行されると、排他モードになります。排他モードには、以下の機能があります。

- 端末が受信する出力を制限します。
- ETO ユーザーのサインオフ後も、そのユーザーと共に残ります。ETO ユーザーが排他モードの時にサインオフすると、そのユーザーが次にサインオンした時も排他モードになっています。
- /END または /START NODE コマンドで終了します。

### ロック・モード

ロック・モードは、端末がメッセージを送信したり受信したりできないようにします。端末、ノード、論理端末 (LTERM) は、オペレーターが /LOCK コマンドを発行した時、ロック・モードになります。ロック・モードは、/UNLOCK コマンドを発行した時リセットされます。

### テスト・モード

テスト・モードでは、端末に入力した入力メッセージがすべてその端末に返送されることを確認します。ノードは /TEST コマンドを使用することによってテスト・モードになります。テスト・モードは /END コマンドあるいは /START コマンドを使用することによってリセットされます。テスト・モードは、再始動および ETO サインオフ/サインオンによって有効ではなくなります。

### 関連タスク

61 ページの『[拡張端末オプション \(ETO\) \(Extended Terminal Option \(ETO\)\)](#)』

これらのトピックでは、拡張端末オプション (ETO) を紹介し、ETO の概要および IMS TM ネットワーク内の ETO 端末の管理に必要な情報を説明します。

### 関連資料

[IMS コマンド \(コマンド\)](#)

## 端末とユーザーの状態

端末またはユーザーは、いくつかの状態のときに動作不能になります。

端末が動作不能になる状態には以下のものがあります。

### 停止状態

停止状態では、物理端末に対応している論理端末のどの出力キューも送達できないようにします。

/STOP NODE コマンドは、IMS とノード間のセッションを終了します。この終了はほとんどの装置に対してすぐ起こりますが、これは、3270 装置と SLU 2 装置の場合、メッセージの終わりに限られます。/STOP NODE コマンドは、また /START コマンドあるいは /RSTART コマンドが出されるまで新規セッションが始まらないようにします。

また /STOP コマンド、/PSTOP コマンド、/MONITOR コマンドは、端末が停止状態になるようにします。この状態は、/START コマンドあるいは /RSTART コマンドでリセットされます。/STOP NODE または USER コマンドでは、ETO ユーザーがサインオフあるいはログオフした後でもその状況を保存することができます。

### QERROR 状態

論理端末は、メッセージ・キューの読み取りまたは書き込みを行おうとしている間に入出力エラーが起きた場合、停止状態になります。この条件は、MTO オペレーターが /START コマンドを出した時にリセットされます。

### QLOCK 状態

論理端末は、LU 6.1 セッションで受信された特定の要求によってロック状態がリセットされるまで、追加の出力を送信することができなくなります。また LTERM は、追加の出力を作成するような入力を受

信しないようにします。この条件は、MTO が /START コマンドを出した時にリセットされます。 / PSTOP あるいは /PURGE コマンドは、QLOCK 状態にある LTERM に対しては無視されます。

### INOP 状態

物理端末は、エラーが検出されて、INOP 状態にされた時はいつでも IMS 装置サポートによって動作不能であると見なされます。また、この物理端末に関連するすべての論理端末は、動作不能であると見なされます。 /START または /RSTART コマンドは動作不能条件をリセットします。

### COMPINOP 状態

コンポーネントの操作不能 (作動不能) な状態は、次の 2 つのどちらかの場合に設定されます。

- 端末のコンポーネントから分離されるエラーが検出された場合
- VTAM に定義された端末に対して /COMPT または /RCOMPT コマンドが出された場合

このコンポーネントに関連したすべての論理端末はメッセージ出力用としては不適合です。コンポーネントの操作不能な状態は、オペレーターが /START LINE PTERM コマンド、 /START NODE コマンド、他の /COMPT コマンド、あるいは /RCOMPT コマンドを発行した時にリセットされます。装置からの特別の信号 (例えば、3270 装置からの装置終了、または SLU 1 からの"コンポーネント使用可能"状況など) もリセットを引き起こします。

### PAGE、SCREEN、および COMPONENT PROTECTION 状態

これは、ビデオ端末、SLU P、金融機関、および LU 6.1 装置がサポートされる状態です。これらの物理端末に関連した論理端末は、出力の選択には適切ではありません。

### SNA QUIESCE 状態

IMS が VTAM のプログラム式ノードへ出力メッセージを送信したが、そのノードが受信を停止したい場合には、そのノードはチェーンの終わりが送信された後に伝送を停止するよう IMS に信号を出します。IMS は、端末が SNA 静止完了 (RELQ) コマンドを送信するまで出力を送信しません。

### 関連概念

[3270 端末画面保護 \(オペレーションおよびオートメーション\)](#)

### 関連タスク

[963 ページの『IMS からの出力の延期』](#)

コントローラー・アプリケーション・プログラムで IMS からそれ以上のどんな出力も受信したくない場合、あるいは受信できない場合に、プログラムは IMS に、VTAM チェーン終了時静止 (QEC) コマンドを送信することができます。IMS は DR1 および VTAM 静止完了 (QC) コマンドを戻します。IMS は、VTAM 解放静止 (RELQ) コマンドを受信するまでワークステーションに出力をそれ以上送信しません。

[61 ページの『拡張端末オプション \(ETO\) \(Extended Terminal Option \(ETO\)\)』](#)

これらのトピックでは、拡張端末オプション (ETO) を紹介し、ETO の概要および IMS TM ネットワーク内の ETO 端末の管理に必要な情報を説明します。

## リソース状況リカバリー

リソース状況リカバリーは、端末またはユーザーがどのようにリカバリーされるかを定義します。

リカバリー情報は RM を使用して共用することができます。したがって、リソースは、IMS ログ・レコードを使用することなくリカバリーすることができます。IMS 障害では、端末またはユーザーは障害システムへの類似性がなくても、あるいは IMS 再始動を待つ必要なく作業を再開することができます。リソースをローカル・システム上のみでリカバリーするか、そのリソース状況を削除するかを選択することもできます。

このトピックでは、リソース状況種別およびリカバリー・モードを定義します。また、高速機能および XRF によってリカバリー・モード状況がどのように使用されるかを説明します。

## リソース状況種別

IMS では、リソース状況を分類して、リソース構造またはローカル・ログ・レコードに保管する必要のある情報の量を判断します。

### リカバリー不能状況

リソースがアクティブのとき、状況は存在するだけです。リソースが非アクティブになり、かつ端末ログオフ、ユーザー・サインオフ、または IMS 再始動時にリカバリーされない場合、状況は削除されます。

### リカバリー可能状況

状況はリカバリーされますが、リソースが、サインオフ、ログオフ、または IMS の再始動時に削除されるのを回避することはできません。

### 有効な状況

状況はリカバリーされ、リソースは、サインオフ、ログオフ、または IMS 再始動間に削除されません。状況がどこに保守されるかは、その状況がコマンドかエンド・ユーザーかにより異なります。

### コマンド有効状況

状況は、STOP、TRACE、および MFSTEST といったリソース・コマンドに関係します。状況リカバリーは、定義されている場合、リソース構造内で RM によって常にグローバルに保守されます。状況は、状況リカバリー・モードによっては影響されません。コマンドによって設定された状況がすべて有効であるわけではありません。

### エンド・ユーザー有効状況

状況は、会話、STSN、および高速機能などのリソース作業に関係します。状況はたびたび変化し、パフォーマンスに影響を与える場合があります。したがって、状況リカバリー・モードを、GLOBAL、LOCAL、または NONE に指定することができます。

## 関連資料

411 ページの『TM リソースの状況リカバリー』

IMS、LTERM、ノード、ユーザー、およびユーザー ID のリカバリー可能状況を保管します。このトピックでは、これらの保管された状況をリストします。

## エンド・ユーザー有効状況に対する状況リカバリー・モード

状況リカバリー・モードは、エンド・ユーザー有効状況によってリソースに対する効力範囲およびリカバリーのロケーションを定義します。

各 IMS に対するデフォルト・モードを設定することができます。これは、ログオンまたはサインオン時にオーバーライドしない限り、すべてのリソースに対して使用されます。その場合、どのエンド・ユーザー有効状況をリカバリーするかを設定することができます。

以下のリストでは、TM リソースに対する状況リカバリー・モードの設定方法が説明されています。

- DFSDCxxx で IMS デフォルトを指定する。
- ユーザー記述子を変更する。
- 静的端末ログオンおよび動的 STSN 端末ログオンに対して DFSLGNX0 オーバーライドを設定する。
- 動的非 STSN ユーザー・サインオンに対して DFSSGNX0 オーバーライドを設定する。

以下のリストでは、エンド・ユーザー有効状況に対する 3 つのリカバリー・モードを説明しています。

### GLOBAL 状況リカバリー・モード

すべてのリカバリー可能状況は IMS レコード・ログ内でローカルに保管されますが、RM を使用してこれらのログの代わりにその状況をリカバリーします。状況は次のログオンまたはサインオン時にリストアップされ、IMSplex 内のどの IMS に対しても使用可能です。リソースがアクティブになったとき、状況はローカル・システムにコピーされます。リソースが非アクティブになったとき、状況はローカル・システムから削除されます。

RM、カップリング・ファシリティー・リソース構造、および共用キューが必要です。キューを共用しており、リソース構造を持っている場合は、GLOBAL がデフォルトです。デフォルトは、DFSDCxxx、ユーザー記述子、またはログオン/サインオン 出口ルーチンによってオーバーライドされます。



## LOCAL 状況リカバリー・モード

すべてのリカバリー可能状況はローカル制御ブロックおよびログ・レコード内にローカルで保管されます。状況は次のログオンまたはサインオン時にリストアされ、ユーザーまたはノードがアクセスしていた同一IMSに対して使用可能です。また、ユーザーまたはノードは、ローカル状況が存在する場所でIMSにアクセスすることができます。この類似性は、RM類似性と呼ばれ、エンド・ユーザー有効状況が存在するときに強制されます。ユーザーまたは端末が別のIMSとのRM類似性を持つことをRMが示した場合、IMS RM類似性により端末またはユーザーがIMSにログオン/サインオンすることを許可しません。この類似性は、エンド・ユーザー有効状況(会話、STSN、または高速機能)が別のIMS上でリカバリーされているために起こります。

RM類似性を持つIMSに障害が起こったとき、RM類似性はまだ存在しています。ログオン出口ルーチン(DFSLGNX0)またはサインオン出口ルーチン(DFSSGNX0)が許可する場合、ユーザーまたはノードは別のIMSに即時にアクセスすることが可能ですが、リソース状況はリカバリーされず、ローカル状況は再始動時に障害IMS上で削除されます。

リソース構造または共用キューが使用されていない場合は、LOCALがデフォルトです。

RMは必要ではありません。RMがアクティブではない場合、RM類似性は強制されません。

## NONE 状況リカバリー・モード

RMまたはローカル・ログ・レコードに保管された状況はありません。ログオンまたはサインオン時に有効な状況はコールドです。STSN状況、会話状況、および高速機能状況は自動的にリカバリー不能になります。

RMおよびリソース構造は必要ではありません。

## 特定リソース・タイプのリカバリー可能性

特定のリソース状況(会話状況、STSN状況、および高速機能状況)のリカバリー可能性を指定する方法の1つとして、DFSDCxxx PROCLIBメンバー内でパラメーターを使用する方法があります。

会話状況、STSN状況、および高速機能状況のリカバリー可能性に適用されるパラメーターは以下のとおりです。

### RCVYCONV= YES | NO

会話状況がリカバリーされるかどうかを指定します。このパラメーターは出力メッセージに影響を与えません。会話状況がリカバリーされない場合、出力は引き続きリカバリーされ、非同期に送信されます。SRMDEF=GLOBALまたはLOCALのときは、YESがデフォルトです。SRMDEF=NONEのときは、NOがデフォルトでYESは無効です。

### RCVYFP= YES | NO

高速機能状況および高速機能出力がリカバリーされるかどうかを指定します。SRMDEF=GLOBALまたはLOCALのときは、YESがデフォルトです。SRMDEF=NONEのときは、NOがデフォルトでYESは無効です。

STSN状況がリカバリーされるよう指定する場合、YESを指定する必要があります。

### RCVYRESP= YES | NO

全機能応答モードが、端末ログオフまたはユーザー・サインオフの後にリカバリーされるかどうかを指定します。デフォルトはRCVYRESP=NOです。SRMDEF=GLOBALまたはNONEの場合、またはCSLリソース・マネージャー(RM)がアクティブの場合、RCVYRESP= YESは無効です。

### RCVYSTSN= YES | NO

STSN端末(SLU P、金融機関、およびISC)に対して、STSN状況がリカバリーされるかどうかを指定します。このパラメーターはSTSNのシーケンス番号のリカバリーにのみ影響を与え、出力メッセージには影響を与えません。SRMDEF=GLOBALまたはLOCALのときは、YESがデフォルトです。SRMDEF=NONEのときは、NOがデフォルトでYESは無効です。

## 関連資料

410 ページの『高速機能リカバリー』

高速機能トランザクションのリカバリーは、状況リカバリー・モード、高速機能リカバリー、STSN リカバリー、および高速機能トランザクションが実行されている場所により決まります。

## 高速機能 リカバリー

高速機能トランザクションのリカバリーは、状況リカバリー・モード、高速機能リカバリー、STSN リカバリー、および高速機能トランザクションが実行されている場所により決まります。

以下の表に、これらの基準に基づいた高速機能トランザクションのリカバリーと状況をリストします。トランザクションが EMH キューを bypass せずにローカルで実行される場合、IMS は高速機能状況を RM 内でグローバルにリカバリーすることはできません。状況リカバリー・モードが GLOBAL で、トランザクションがローカルで実行されている状態では、状況リカバリー・モードは RM 類似性を持つ端末またはユーザーにより一時的に LOCAL に変更されます。

表 73. 高速機能リカバリー可能性の判断

状況リカバリー・モード	高速機能リカバリー (RCVYFP)	STSN リカバリー (RCVYSTSN)	ローカル高速機能 (DBFHAGUO)	共用 EMH キュー (DBFHAGUO)
LOCAL または GLOBAL	NO	NO	状況およびメッセージが廃棄された。	状況およびメッセージが廃棄された。
LOCAL または GLOBAL	NO	YES	INVALID	INVALID
LOCAL	YES	NO	状況および出力がローカルでリカバリーされた。STSN がコールド・スタートされた。	状況および出力がローカルでリカバリーされた。STSN がコールド・スタートされた。
GLOBAL	YES	NO	状況および出力がローカルでリカバリーされた。STSN がコールド・スタートされた。	状況および出力がグローバルにリカバリーされた。STSN がコールド・スタートされた。
LOCAL	YES	YES	状況および出力がローカルでリカバリーされた。STSN がリカバリー可能。	状況および出力がローカルでリカバリーされた。STSN がリカバリー可能。
GLOBAL	YES	YES	状況および出力がローカルでリカバリーされた。STSN がリカバリー可能。	状況および出力がグローバルにリカバリーされた。STSN がリカバリー可能。

## XRF 端末に対するリカバリー・モード

XRF 環境でのテークオーバー時は、状況リカバリー・モードの影響が拡張回復機能 (XRF) 端末クラスごとに異なります。

XRF 環境でのオペレーティングは状況リカバリー・モードには影響を与えません。ただし、シスプレックス環境で状況リカバリー・モードを使用している場合、RM を使用していない環境と比較すると、クラス 1 端末のリカバリーは少し異なります。

## クラス 1 端末

IMS は、たとえ状況リカバリー・モードが GLOBAL の場合でも、XRF テークオーバーのためのローカル・ログ・レコードのみにより決まります。セッション・テークオーバーが完了した場合、IMS は、RM および新規アクティブ・システム間の状況の相違をすべて調和させます。GLOBAL モードは、正常終了後の

IMSplex 内の別の IMS に ログオンすることができるユーザーまたは端末の利点を提供しますが、クラス 1 端末に関しては、LOCAL モードの方がより適切な場合があります。

状況リカバリー・モードが LOCAL で、RCVYRESP=YES であり、RM が使用されていない場合、全機能応答モードがクラス 1 端末に対してのみリカバリーされます。

## クラス 2 端末

IMS は、状況リカバリー・モードが LOCAL の場合、テークオーバー時にローカル・ログ・レコードを使用し、状況リカバリー・モードが GLOBAL の場合、RM を使用します。新規アクティブ・システムはアクティブ・ターミナルのログを記録しますが、ユーザー・サインオンが必要です。IMS はローカル・ログ・レコードを使用してログオンを開始しますが、ユーザーまたは端末の状況リカバリー・モードを基にしたリカバリーを決定します。

## クラス 3 端末

IMS はこれらの端末を自動的にテークオーバーしません。リカバリーは状況リカバリー・モードが基になります。

### 関連概念

416 ページの『[拡張回復機能 \(XRF\) の計画](#)』

XRF 複合システムでは、障害が発生したときに、エンド・ユーザー・サービスを迅速に再開することができます。

## TM リソースの状況リカバリー

IMS、LTERM、ノード、ユーザー、およびユーザー ID のリカバリー可能状況を保管します。このトピックでは、これらの保管された状況をリストします。

### LTERM リカバリー状況

IMS は以下の LTERM リカバリー可能状況を保管します。

- LTERM 名
- EDIT=UC (大文字変換指定)
- ノードまたはユーザー所有者

IMS は、以下の LTERM コマンド有効状況を保管します。

- /ASSIGN SAVE
- STOP

### ノード・リカバリー状況

IMS は、以下のリカバリー可能状況を保管します。

- ノード名
- 装置タイプ
- 割り振られた LTERM 名
- 割り振られたユーザー名

IMS は、以下のコマンド有効状況を保管します。

- EXCLUSIVE
- MFSTEST
- STOP

- TRACE

IMS は、以下のエンド・ユーザー有効状況を保管します。

- 会話
- 高速機能 (Fast Path)
- STSN
- 端末ログオフまたはユーザー・サインオフの後の全機能応答モード。ただし、RM が使用されておらず、SRMDEF=LOCAL で RCVYRESP=YES の場合に限り。全機能応答モードは、IMS の終了または障害の後にはリカバリーされません。

### ユーザー・リカバリー状況

IMS は以下のユーザー・リカバリー可能状況を保管します。

- ユーザー名
- ユーザー ID
- 割り振られた LTERM 名
- 割り振られたノード名
- 自動ログオン・パラメーター

IMS は以下のユーザー・コマンド有効状況を保管します。

- EXCLUSIVE
- MFSTEST
- STOP
- /CHANGE AUTOLOGON SAVE

IMS は以下のユーザー・エンド・ユーザー有効状況を保管します。

- 会話
- 高速機能 (Fast Path)
- 端末ログオフまたはユーザー・サインオフの後の全機能応答モード。ただし、RM が使用されておらず、SRMDEF=LOCAL で RCVYRESP=YES の場合に限り。全機能応答モードは、IMS の終了または障害の後にはリカバリーされません。

### ユーザー ID リカバリー可能状況

IMS は以下の LTERM リカバリー可能状況を保管します。

- ユーザー ID
- 端末名

## セキュリティ設計

---

IMS ネットワーク内の端末の無許可使用を防止するために、RACF (または同等の製品) を使用できます。

RACF は、z/OS オペレーティング・システムの下で使用可能なライセンス・プログラムです。

RACF では以下の項目に対するアクセス管理を行うことができます。

- 物理端末
- 論理端末
- トランザクション
- コマンド

RACF セキュリティーを使用しない場合、IMS はユーザー端末 (マスター端末は除きます) で特定のコマンドのみの入力を許可します。これをデフォルト端末セキュリティと言います。

RACF を使用すると、ユーザー ID に基づいてセキュリティー・プロファイルを設計でき、ネットワークに次の2つのレベルのセキュリティーを定義できます。

- ネットワークに接続している端末の使用を管理することができます。
- 端末からアクセスして利用できるリソースを管理することができます。

端末の使用は、サインオン検査セキュリティーによって制御します。例えば、端末ユーザーが、/SIGN コマンドにパラメーターとして、あるいは DFS3649 メッセージの応答に、ID を入力したとします。RACF、出口ルーチン、あるいは両方を使用してサインオンの妥当性検査をすることができます。ユーザー ID は、各入出力メッセージおよび各データベース変更時にログに記録されます。

#### 関連概念

[IMS コマンドの使用可能性 \(オペレーションおよびオートメーション\)](#)

#### 関連資料

[IMS タイプ 1 コマンドに関する端末セキュリティーのデフォルト \(コマンド\)](#)

## TM ネットワーク内でのトランザクション許可

トランザクション許可中に、RACF は、端末ユーザーがトランザクションを入力する許可があるかどうかを検査します。

### このタスクについて

#### 関連概念

[IMS セキュリティー \(システム管理\)](#)

### トランザクションをセキュアにするための RACF の使用

どのユーザーがどのトランザクションを出せるかを制御するには、制御対象のトランザクションを RACF に TIMS クラスとして定義し、RACF で定義されたユーザーまたはユーザー・グループに権限を付与します。

### このタスクについて

トランザクション許可が成功したか失敗したかを追跡するのにシステム・モニター機能 (SMF) のロギングを使用することもできます。RACF を使用して、トランザクション・セキュリティー・プロファイルで監査機能を要求してください。

### トランザクション許可出口ルーチン (DFSCTRNO) の使用

トランザクション許可出口ルーチン (DFSCTRNO) を使用すれば、/SIGN コマンドを使用してサインオンしたユーザーが入力したトランザクションの妥当性を検査することができます。

### このタスクについて

トランザクション許可出口ルーチンは、RACF と結合して使用できます。

**関連情報:** トランザクション許可出口ルーチン (DFSCTRNO) の使用法の詳細については、「IMS V15 出口ルーチン」を参照してください。

## TM ネットワーク内でのコマンド許可

コマンド許可には、RACF、およびコマンド許可出口ルーチン (DFSCCMD0) を使用できます。

### このタスクについて

#### 関連概念

[IMS セキュリティー \(システム管理\)](#)

## コマンドをセキュアにするための RACF の使用

どのユーザーが被制御コマンドを出せるかを制御するには、被制御コマンドを RACF CIMS クラスとして定義し、RACF 定義ユーザーあるいはユーザーのグループに認可を与えてください。

### このタスクについて

コマンド許可が成功したか失敗したかを追跡するには SMF のロギングを使用することができます。RACF を使用して、コマンド・セキュリティー・プロファイルに監査機能を要求してください。

## コマンド許可出口ルーチン (DFSCCMD0) の使用

コマンド許可出口ルーチン (DFSCCMD0) を使用すれば、入力コマンド・バッファーを編集することによって、キーワードとパラメーターを制限することができます。

### このタスクについて

コマンド許可出口ルーチンはコマンドの妥当性を検査します。この出口ルーチンは、RACF で使用することができます。

#### 関連資料

[コマンド許可出口ルーチン \(DFSCCMD0\) \(出口ルーチン\)](#)

## トランザクション・コマンド・セキュリティー

トランザクション・コマンド・セキュリティーは、もう 1 つのレベルのアクセス制御を提供できます。

トランザクション・コマンド・セキュリティーは端末と間接的にかかわっています。IMS コマンドを出すプログラムを開始するために、端末ユーザーがトランザクションを入力する場合は、プログラムから出すことができるコマンドのセットとトランザクション・コードの両方に許可を与える必要があります。

## パスワード・セキュリティー

パスワード・セキュリティーには、端末で入力されるトランザクションまたはコマンドがパスワードを持っている必要があります。これにより、特定のトランザクションあるいはコマンドを出すユーザーがそうすることを許可されているかどうかを検査する 2 番目のセキュリティー機能が実行されます。

## ユーザー ID を基本にした RACF とセキュリティー・プロファイルの使用

RACF には REVERIFY オプションがあり、コマンドまたはトランザクションの入力時にユーザーのパスワードを入力する必要があります。サインオン許可に RACF を使用し、RACF にトランザクションまたはコマンドを定義する時、REVERIFY パラメーターを APPLDATA として含める必要があります。REVERIFY を使用するには、RVFY=Y を実行パラメーターとして指定しなければなりません。

## APPC/IMS のセキュリティー

APPC/IMS には、RACF へのシステム許可機能 (SAF) インターフェースを使用するセキュリティー、または同等のセキュリティー環境が必要です。RACF は、LU 6.2 アプリケーション・プログラムからのリモート・トランザクションのオプション項目です。

APPC/IMS はトランザクション許可出口ルーチン (DFSCTRNO) とコマンド許可出口ルーチン (DFSCCMD0) の両方をサポートします。

#### 制約事項:

- APPC/IMS は /SIGN コマンドをサポートしません。ユーザー ID の妥当性検査にこのコマンドが必要なからです。RACF を使用する場合、z/OS はユーザー ID の妥当性を検査するため、各 APPC/IMS は有効なユーザー ID を持っています。
- リモート LU 6.2 アプリケーション・プログラムから入力された IMS コマンドについては、RACF およびコマンド許可出口ルーチン (DFSCCMD0) を使用しない場合、デフォルトのコマンド・セキュリティーは次の 4 つのコマンドのみを許可します。



- /BROADCAST
- /LOG
- /RDISPLAY
- /RMLIST

他のコマンドを許可するには、DFSCCMD0 または RACF を使用してください。

#### 関連資料

[トランザクション管理のための DL/I 呼び出し \(アプリケーション・プログラミング API\)](#)

## ETO のセキュリティ

RACF セキュリティーは、静的端末と ETO 端末の両方でコマンドおよびトランザクション許可に使用できます。

#### 関連概念

81 ページの『ETO を用いた高セキュリティ環境の計画』

ETO は、IMS システムのセキュリティを強化します。ETO のセキュリティ機能は、ご使用のシステムのニーズに応じてカスタマイズできます。

## 高速機能端末の計画

高速機能が可能なシステムにするためには、FPCTRL マクロで高速機能サポート を指定します。

トランザクション率が非常に高い IMS システムは、高速機能の急送メッセージ・ハンドラー (EMH) 機能を使用します。EMH は、メッセージの長さやセグメント化に制約を課してメッセージ処理を促進するパフォーマンス・オプションです。LU 6.2 端末は、定義とか仕様の指定をしなくても EMH を使用できます。

高速機能を使用した場合は、高速機能トランザクションに関する最初の適格入力メッセージが受信された時点で、EMH バッファが EMH バッファ・プールから取得されます。バッファは、セッションが終了するかユーザーがサインオフするまでその後の使用に備えて端末に割り振られたままになります。EMH バッファが端末に割り振られている場合、それは、次の入力呼び出しの要件に合致する限り再利用されます。メッセージがより大きなバッファを必要とする場合、高速機能はそれを大きなバッファと交換します。

入力メッセージが IMS への入力時に下記の基準すべてを満たす場合、入力メッセージは全機能メッセージ・キュー・バッファ内ではなく高速機能 EMH バッファで編集されます。

- 端末は FES 可能ではない
- 端末は MTO ではない
- 入力は単一セグメント
- 入力が MFS で編集されていない
- 入力が IMS コマンドでない
- 少なくとも 1 つの高速機能トランザクションの実行が高速機能を使用してスケジュールされている

**例外:** メッセージが高速機能トランザクションでない場合には、メッセージは全機能メッセージ・キュー・バッファに移動されます。

上記のすべての特性を持つメッセージを受信すると、EMH バッファが端末に割り振られます。EMH バッファは、ユーザーがサインオフするかセッションが終了するまで、端末に割り振られたままになります。より大きな EMH バッファが特別なアプリケーションで必要になるまで、バッファは高速機能トランザクションのために繰り返し使用されます。より大きなバッファが必要な時は、現行の EMH バッファはより大きいサイズの EMH バッファと交換されます。小さい EMH バッファは解放されて、EMH プールに戻されます。

IMS 制御領域初期設定の EMHL パラメーター (始動パラメーターとも呼ばれる) によって、すべての高速機能適格端末のすべての高速機能トランザクションについてのデフォルトの EMH バッファ・サイズを指定します。高速機能トランザクションでは、デフォルトより大きい EMH バッファを指定することができます。より大きなアプリケーションの EMH バッファ・サイズは、システム生成時に FPATH=size パラメ

ーターを使用して、APPLCTN または TRANSACT stage\_1 マクロで指定してください。EMH バッファークラスタが使い尽くされた場合には、メッセージ DFS3971 が入力端末に送信されます。

バッファークラスタは、動的に拡大したり、縮小したりする EMH バッファークラスタから取り出されます。クラスタのサイズは、高速機能トランザクションを並行して入力する端末の数と、各要求を満たすのに必要なバッファークラスタサイズにより決まります。

APPLCTN および TRANSACT マクロで FPATH=No / Yes を指定するか、または FPATH=size (size は、トランザクションを実行するのに必要な EMH バッファークラスタサイズ) を指定することができます。FPATH=size は FPATH=Yes を暗黙指定します。最小の EMH バッファークラスタサイズは 12 バイトで、最大は 30720 バイトです。

端末関連のキーを持っている場合を除いて、MSDB は ETO 端末で使用可能です。

TERMINAL マクロの OPTIONS キーワードあるいは ETO 記述子は、以下のために使用してください。

- 高速機能適格で使われる端末が応答モードで操作できるよう FORCRESP あるいは TRANRESP を宣言する。
- 適切なところで自動的にページ削除するように PAGDEL を指定する。

非 VTAM 端末を使用して、高速機能トランザクションを入力することができます。非 VTAM 端末の場合は、LINE マクロの RESP=TERM パラメーターを使用して応答モードを指示してください。

#### 関連資料

[IMS 15 でサポートされる端末および装置 \(リリース計画\)](#)

## 拡張回復機能 (XRF) の計画

XRF 複合システムでは、障害が発生したときに、エンド・ユーザー・サービスを迅速に再開することができます。

XRF を使用して、全トランザクションとデータベース処理を定期保守のために、あるいはオフピーク時に、1 つの IMS システム から別のシステムに切り替えることもできます。

XRF を使用していない場合に IMS 障害が起きた時には、エンド・ユーザー・サービスの非常に短い中断だけで高水準の可用性を達成することは困難です。終了処理を完了し、障害の原因を判別しなければ、再始動を行うことはできません。IMS 領域の再始動が必要で、かつ端末セッションの再設定が必要な場合、再始動自体に時間がかかります。

XRF 構成は、アクティブ IMS と呼ばれる 1 次システムと代替 IMS と呼ばれる 2 次システム で構成されます。代替 IMS は、同一、あるいは異なる中央演算処理複合システム (CPC) 内でアクティブ IMS のアクティビティを追跡します。アクティブ IMS は XRF がない場合と同じ方法で IMS 作業負荷を処理します。また余分な処理もロギングも行いません。アクティブ IMS は、そのアクティビティを、ログを使用して代替 IMS に知らせます。監視メカニズムは、アクティブ IMS で起きた問題を代替 IMS に警告します。アクティブ IMS には、代替 IMS のアクティビティは認識されません。

代替 IMS はどのようなトランザクションも処理しません。つまり、代替 IMS は、アクティブ IMS の状況を保持し、障害の兆候をモニターします。代替 IMS は、アクティブ IMS におけるリソースの変更を追跡し、自身の制御ブロックとバッファークラスタを更新して、継続的に作動可能状況を変更します。このようにして、作業を中断せずにアクティブ IMS から代替 IMS に迅速にシフトすることができます。アクティブ IMS から代替 IMS へのこの作業のシフトは、テークオーバー処理と呼ばれます。以下の図は、単純な XRF 複合システムの例を示しています。



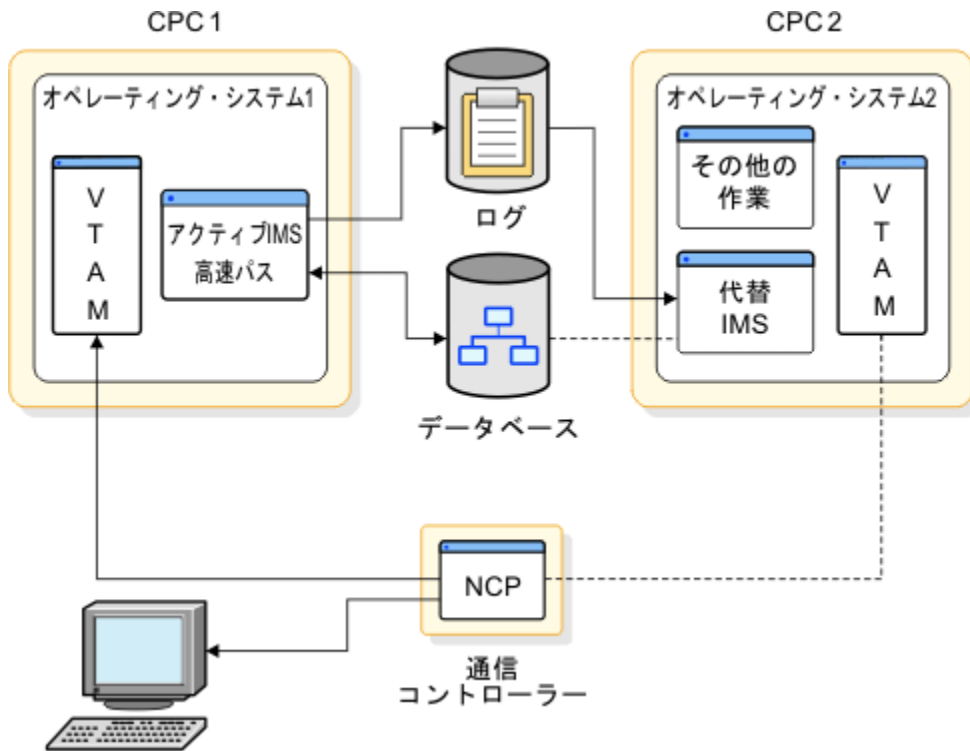


図 49. サンプル XRF 複合システム

#### 関連概念

拡張回復機能の概要 (システム 管理)

## XRF 複合システムにおける MNPS または USERVAR の使用

IMS マルチノード持続セッション (MNPS) サポートを XRF に使用できます。パフォーマンスの制約のために、これは推奨される解決策ではありません。

以下の 2 つのタイプの XRF 複合システムの 1 つをインプリメントすることができます。

- VTAM MNPS と MNPS ACB を使用する XRF 複合システム。
- ユーザー変数 (USERVAR) と USERVAR テーブルを使用する XRF 複合システム。

どちらのタイプの XRF 複合システムも IMS への同様に向上された可用性を備えていますが、端末セッションおよびテークオーバー処理の管理方法が異なります。その他の相違点には、以下のものが含まれます。

- USERVAR を使用する XRF 複合システムには、追加のハードウェア要件があります。
- それぞれのタイプの複合システム内で、VTAM とネットワーク端末装置は、以下のように IMS との通信に異なる名前を使用します。
  - MNPS を使用する XRF 複合システムでは、IMS は、VTAM とネットワーク端末装置に MNPS ACB (MNPS 名) という名前で識別される。
  - USERVAR を使用する XRF 複合システムでは、IMS は、VTAM にはアプリケーション ID ACB (アプリケーション ID 名) という名前で、ネットワーク端末装置には USERVAR という名前で識別される。

MNPS を使用する XRF 複合システムには、やはりアプリケーション ID ACB が存在しますが、XRF の目的にはアプリケーション ID ACB は使用されません。

#### 関連概念

拡張回復機能の概要 (システム 管理)

## XRF 複合システムの端末

XRF は、IMS ユーザーが現在使用しているすべての端末をサポートします。XRF は、ユーザーの最小の関与で、緊急サポートを提供します。

XRF では、システム定義のマクロ・ステートメントでキーワードを指定するか、あるいは ETO 記述子の指定が必要です。端末の切り替え優先順位をデフォルトから変更するか、あるいは端末タイプ・セッションを変更する必要がある限り、端末定義マクロを変更する必要はありません。

XRF はインストール・システムの端末の特性が許す最大限のサービスを端末に自動的に与えます。このトピックでは、これらの特性について説明します。

XRF は、以下の 3 つのタイプの端末へのサポートを提供します。

### クラス 1

テークオーバー時、セッションは持続します。セッションは、途切れることなく代替 IMS に透過的に切り替えられます。

### クラス 2

テークオーバー時、セッションは終了します。代替 IMS はセッションの再設定を自動的に行おうとします。ユーザーは、しばらくの間サービスを受けられません。

### クラス 3

テークオーバー時、セッションは終了します。端末が物理的に接続されていれば、オペレーターまたはエンド・ユーザーは新規アクティブ IMS で新規セッションを設定することができます。

APPC/IMS 装置はクラス 3 なので、障害時の LU 6.2 会話は、テークオーバー中保持されることはありません。新しい LU 6.2 会話は、テークオーバーをした後、新しくアクティブになった IMS から割り振られなければなりません。新しくアクティブになった IMS は、テークオーバーが起こる前の IMS とまったく同じ APPC/IMS アクティビティを処理します。新しくアクティブになった IMS 前にアクティブであった IMS と同じ LU 名を持ちます。

## クラス 1 端末

XRF を最大限に活用するには、複合システムのほとんどの端末装置をクラス 1 端末として定義します。クラス 1 端末は、TYPE マクロの UNITYPE キーワードで、次のいずれかとして定義された端末です。

- SLUTYPE1
- SLUTYPE2
- SLUTYPEP
- FINANCE

クラス 1 端末の取り扱い方法は、VTAM マルチノード持続セッション (MNPS) サポートを使用するかどうかにより異なります。

### MNPS を使用する XRF システムでのクラス 1 端末のリカバリー

MNPS を使用する XRF を使用する場合、クラス 1 端末は代替 IMS 上にバックアップ・セッションを持ちません。その代わりに、アクティブ IMS 内でクラス 1 セッションが開始される時に、IMS は、VTAM に対してセッション状態とデータ・トラフィック情報の両方を追跡するように伝えます。

XRF テークオーバー時に、IMS は、VTAM にクラス 1 端末セッションを検査し、データ・トラフィック情報を使用してセッションを透過的に復元することを要求します。高い優先順位を持つ端末装置は、低い優先順位の端末装置より前に復元されます。

420 ページの『[端末サポートの指定](#)』の表にリストされている端末タイプは、MNPS 使用可能な XRF 対応 VTAM が所有する場合、クラス 1 サービスに適格です。VTAM を MNPS に対して使用可能にするには、APPL 定義に PERSIST=MULTI を指定します。

### USERVAR を使用する XRF システムでのクラス 1 端末のリカバリー

USERVAR を使用する XRF システムでは、代替 IMS は、アクティブ IMS 上にオープン・セッションを持つすべてのクラス 1 端末のバックアップ・セッションを維持します。これらのセッションは、トラッキング・フェーズ中にアクティブ IMS でセッションが開始した時に設定されます。テークオーバーの場合、代替 IMS は、すべてのクラス 1 端末をバックアップ・セッションに切り替えます。

端末ユーザーから見ると、IMS 1つのセッションしかありません。NCPから見ると、2つのホスト・セッションがあります。実際のメッセージは、ノードとアクティブ IMS の間でやりとりするだけです。代替 IMS は、IMS ログをモニターしてメッセージのアクティビティ 状況を追跡します。

アクティブ IMS がクラス 1 端末との端末セッションを設定または終了すると、代替 IMS はその端末とのバックアップ・セッションを設定または終了します。アクティブ IMS が失敗すると、代替 IMS は設定されたバックアップ・セッションのモードを BACKUP から ACTIVE に変更して端末セッションを(ノードの視点から制御を失うことなしに)引き継ぎます。セッションが切り替えられると、NCP は新規アクティブ IMS に端末の状況ビューを送信します。IMS は、これと IMS の状況レコードを比較して、必要な場合は、どのようなリカバリー・アクションを行うかを決定します。REVERIFY オペランドが RACF で使用される場合、ユーザーはサインオンをやり直す必要があります。

420 ページの『[端末サポートの指定](#)』の表にリストされている端末タイプは、以下の両方の条件を満たしている場合、クラス 1 サービスに適合です。

- XRF 機能がある VTAM が所有している
- 37x5 通信コントローラー内の XRF 対応 NCP に接続している

## クラス 2 端末

クラス 2 端末は、代替 IMS にバックアップ・セッション・サポートを持っていませんが、テークオーバー後、IMS により自動的に再始動されます。

クラス 2 端末セッションがアクティブ IMS で設定されると、代替 IMS は、ログ・レコードを使用してセッション開始および終了を追跡します。アクティブ IMS が異常終了した時、代替 IMS は障害が起こる前にアクティブであったネットワーク・リソースで新規セッションの設定を試みます。

IMS が、利用可能なネットワーク・パスをもたない端末のセッションを再構築しようとした時には、試みは失敗します。IMS がローカル接続端末でセッションを再設定する前に、オペレーターはその端末を新規アクティブ IMS に切り替える必要があります。複合システムがこの種の端末を多く持っている場合、テークオーバーが実際に行われる前に、オペレーターによる制御を考慮しておく必要があります。この時、オペレーターは該当端末のセッションを早急にリカバリーするため切り替えを実行できなければなりません。オペレーターにこの制御を与えるには、IMS.PROCLIB 内のメンバー DFSHSBxx の AUTO パラメーターを使用してください。

クラス 2 として適切な端末については、420 ページの『[端末サポートの指定](#)』の表を参照してください。

XRF 複合システムにプログラマブル・コントローラーと通信するアプリケーション端末がある場合、利用可能な最高レベルの透過性を達成するためには、通信部分にコードを加える必要があります。さらに、ISC 端末の場合、ISC 端末を XRF IMS に接続するリンク用セッションを定義する必要があります。

## クラス 3 端末

クラス 3 端末は、代替 IMS にバックアップ・セッション・サポートを持っていません。テークオーバー後の新規アクティブ IMS で、端末セッションを MTO またはユーザー・ログオンによって、手動で再始動する必要があります。

すべての IMS 端末はクラス 3 サービスを提供できます。切り替えを行ってはならないと示されたクラス 1 およびクラス 2 端末は、クラス 3 として適合です。これを行うには、システム定義マクロまたは ETO ログオン記述子に BACKUP=NO を指定します。

テークオーバー後、クラス 3 端末は既存のプロシージャを使用して、新規のアクティブ IMS と接続します。新規 IMS システムへのパスがない場合は、セッションを再設定することができません。障害を起こしている IMS で使用されているすべての LU 6.2 会話はテークオーバー後、再割り振りされる必要があります。

### 関連概念

[XRF 複合システムの端末 \(システム管理\)](#)

### 関連資料

410 ページの『[XRF 端末に対するリカバリー・モード](#)』

XRF 環境でのテークオーバー時は、状況リカバリー・モードの影響が拡張回復機能 (XRF) 端末クラスごとに異なります。

#### 84 ページの『ETO 記述子のコーディング』

IMS は ETO 記述子を使用して、端末構造とユーザー構造を動的に構築します。

IMS 環境で使用されるマクロ (システム定義)

## 端末サポートの指定

以下の表では、各種の装置の優先順位または端末タイプを変更するために使用できるキーワードを要約しています。

IMS がサポートしているすべての端末は XRF 複合システムで使用することができます。端末特性は、XRF 複合システムで端末が持っているサポートのタイプを、クラス 1 が最高のサポートを持ち、クラス 3 は何もサポートしないと判断します。

表 74. 端末サポートの指定

装置	クラス	TERMINAL マクロあるいは ETO ログオン記述子の BACKUP キーワード
SLUTYPE1 (3286) SLUTYPE2 (3278,3279,...) SLUTYPEP (4700,8100, SERIES 1,...) FINANCE 3601	クラス 1、2、3 として使用可能	クラス 1 サービスの場合 BACKUP=(1-7,YES) クラス 2 サービスの場合 BACKUP=(1-7,NO) クラス 3 サービスの場合 BACKUP=NO
LUTYPE6 (ISC) MSC (VTAM) NTO (専用回線) 3270 (VTAM) SPOOL 回線グループ ローカル接続装置	クラス 2、3 として使用可能	クラス 2 サービスの場合 BACKUP=(1-7,NO) クラス 3 サービスの場合 BACKUP=NO
LU 6.2	クラス 3 として使用可能	適用できない

基本と 2 次のマスター端末は特別なタイプの端末で、アクティブ IMS と代替 IMS の両方に同時に存在します。これらは、クラス 1 またはクラス 2 端末としては扱われません。代替 IMS のマスター端末はテークオーバーによって混乱することはありません。テークオーバーの後、代替 IMS のマスター端末が、アクティブ IMS のマスター端末からの未配布メッセージを処理します。

システム定義ステートメントまたは ETO ログオン記述子におけるいくつかのマクロの BACKUP キーワードでは、端末が受け取る実際のサポートを指定します。インストール・システムは端末の切り替え優先順位を判別します。優先順位は一番低いレベル 1 から一番高いレベル 7 まであります。VTAM 要求は IMS によって優先順位の順に並べられますが、ネットワークにおけるペーシングや、回線速度、輻輳 (ふくそう) といった予測不能な要因のため、要求が完了する順序は変動します。

BACKUP=(1-7,YES) の場合は、端末はセッション制御を失うことなく自動的にテークオーバーされます。端末が XRF 対応 VTAM に接続されている場合は、クラス 1 サポートを構成します。

クラス 1 とクラス 2 の両方の端末では、デフォルトの切り替え優先順位は 4 です。

BACKUP=NO の場合、この端末は、テークオーバー後に手で再始動しなければなりません。これは、クラス 3 サポートを強制します。

BACKUP パラメーターのオプションが誤って設定された場合、IMS は誤ったオプションをセッション初期設定時にリセットします。例えば、ISC 端末に BACKUP=(1-7,YES) を指定した場合、IMS は YES を NO に



リセットします。これは、ISC 端末が受け取ることができる最高位クラスのサポートは、クラス 2 であるからです。

## USERVAR を使用する XRF と BACKUP パラメーター

USERVAR を使用する XRF システムの場合は、BACKUP=(1-7,YES) によっても 代替 IMS にすべてのクラス 1 端末のバックアップ・セッションを維持するように伝えられます。代替 IMS で設定できる端末バックアップ・セッションの最大数は、NCP BUILD ステートメントの BACKUP=n オプションで指定されます。(n +1) のクラス 1 端末ユーザーがアクティブ IMS にログオンしようとする、ログオンは拒否されます。

BACKUP=(1-7,NO) の場合、この端末はテークオーバー後に自動的に再始動されますが、BACKUP セッションは設定されません。これは、クラス 2 サポートを強制します。

## 高速ネットワーク再接続 (RNR) の計画

高速ネットワーク再接続 (RNR) は、障害 (IMS、z/OS、CPC、または VTAM) の発生後 IMS VTAM 端末セッションを自動的に再接続し、その後の IMS は IMSplex 内の同じまたは別の CPC 上で再始動します。RNR を使用すると、VTAM セッションの可用性が向上し、障害発生後にセッションの終結処理と IMS の再始動を行う必要がなくなります。

注：以下のトピックは RNR システムのみに適用されます。同じく VTAM マルチノード持続セッション (MNPS) を使用する XRF システムと混同しないようにしてください。

## サポート・レベルの指定

RNR サポートは、IMS 実行パラメーター、ETO ログオン記述子の OPTIONS= パラメーター、ログオン出口ルーチン (DFSLGNX0) を使用したセッション別の 3 つのレベルで指定できます。

### このタスクについて

RNR サポートは、RNR=ARNR、RNR=NRNR、または RNR=NONE を指示して DFSDCxxx PROCLIB メンバー内で指定します。ARNR は RNR を活動化し、IMS または VTAM の再始動後の /START DC 処理時に IMS が基本セッション再接続を自動的に実行することを指定します。NRNR は、自動セッション再接続が活動化されるよう指定しますが、ログオン記述子またはログオン・ユーザー出口ルーチンによってオーバーライドされない限り、使用されません。オーバーライドされない限り、IMS または VTAM の再始動後、セッションは終了します。NONE は RNR を活動化しません。

RNR がパラメーターなし (ARNR または NRNR なしの RNR=) で指定されている場合、NRNR がデフォルトです。RNR が指定されていない (RNR= なし) 場合、RNR は、RNR=NONE が指定されている場合と同様、活動化されません。

RNR サポートは、ETO ログオン記述子で OPTIONS= ARNR|NRNR パラメーターを使用して指定することもできます。このレベルで指定された RNR サポートは、IMS 実行パラメーターを使用してシステム・レベルで設定された値をオーバーライドします。RNR が指定されずに IMS が初期設定された場合には、ETO ログオン記述子の RNR サポートの指定は無視されます。

RNR サポートは、ログオン出口ルーチン (DFSLGNX0) で LGOPT=LGOARNR パラメーターを使用して指定することができます。このレベルで指定された RNR サポートは、ETO ログオン記述子またはシステム・レベル実行パラメーターを使用した設定をオーバーライドします。RNR が指定されずに IMS が初期設定された場合は、DFSLGNX0 での RNR サポートの指定は無視されます。

DFSDCxxx PROCLIB メンバーを使用して、IMS または VTAM 障害後の最大セッション持続時間を指定することもできます。この時間を指定するためのパラメーターは PSTIMER です。PSTIMER は、1 から 86400 秒までの値に設定できます。デフォルト設定は 3600 (1 時間) です。0 が指定された場合は、タイマーは使用されず、セッション持続性は無期限に続きます。

注：MNPS の VTAM START オプション HPRPST が PSTIMER より小さい値に設定されると、VTAM START は PSTIMER 設定をオーバーライドします。

## サポート・レベルの変更

RNR サポートは、セッション間または IMS 再始動間で オン/オフにすることができ、IMS PROCLIB メンバー、ETO ログオン記述子、およびログオン出口の中で任意のレベルで変更することができます。

### このタスクについて

RNR は、オンまたはオフに設定されると、次のセッション・コールド・スタートまで適用されません。IMS 端末 (非 ISC、金融機関および SLUP) の場合は、最後に選択された RNR オプションがログオンごとにアクティブになります。ISC、金融機関、および SLUP 端末の場合は、最後に選択された RNR が複数のログオンまたは IMS 再始動にまたがり、次のセッション・コールド・スタートのときだけしかアクティブになりません。

## 持続セッション・トラッキング

VTAM 持続セッション・トラッキングは、単一ノード持続セッション (SNPS) および マルチノード持続セッション (MNPS) の両方に用意されています。

VTAM に必要な IMS 持続セッション・サポートのレベルは、APPL 定義ステートメントで PERSIST=MULTI | SINGLE パラメーターを使用して指定されます。

VTAM SNPS には、次のような特性があります。

- 再接続は、元の IMS と同じ CPC で行わなければなりません。
- IMS の障害と再接続だけがサポートされます。

VTAM MNPS には、次のような特性があります。

- 再接続は、IMSplex 内の別の CPC で行うこともできます。
- IMS、VTAM、z/OS、および CPC の障害と再接続がサポートされます。

VTAM MNPS を使用する場合は、VTAM エンド・ノードが APPN/HPR (高性能経路指定) ネットワーク環境で実行されなければならないことに注意してください。さらに、すべての VTAM インスタンスを、ISTMNPS 構造を使用して並列シスプレックス・カップリング・ファシリティーに接続しなければなりません。

APPC 持続セッション・サポートは APPC/MVS によって提供されます。ただし、障害の後で APPC 会話が自動的に再始動されることはありません。

## 持続セッション・トラッキングの終了

IMS が VTAM アクセス制御ブロック (ACB) をクローズする前にセッションが終了すると、VTAM 持続セッション・トラッキングおよび IMS の再接続機能が、すべての端末で終了します。

IMS 開始のシャットダウンを実行することができます (例えば、/CHECKPOINT FREEZE または /STOP DC または /CLSDST コマンドを出すことによって)。

VTAM セッションは、VTAM 持続セッション・トラッキングが、非 IMS 開始のセッションがクローズされた (そのセッションが IMS の再始動の完了前にクローズされた) ために早期終了した場合にも切断されます。これは、次のいずれかの理由で起こることがあります。

- リモート・オペレーター開始セッションの終了。例えば、/RCL コマンドの実行。
- ネットワーク・オペレーター開始セッションの終了。例えば、VARY NET または INACT コマンドの実行。
- PSTIMER 値によって指示された持続セッション・トラッキングのタイムアウト。

MNPS で RNR に対して IMS が定義された場合、VTAM ネットワーク・オペレーターは、VTAM DISPLAY ID コマンドを使用して、持続性が現在アクティブになっているかどうかを判別することができます。

VTAM をシャットダウンする場合、HALT NET または HALT NET, QUICK コマンドはセッション持続性を終了し、HALT NET, CANCEL コマンドはセッション持続性を維持します。

## IMS シャットダウンと RNR

IMS に対する **MVS MODIFY** コマンドを使用して IMS がシャットダウンされる場合、RNR は迅速に IMS ネットワークを終了するため、VTAM 持続性は保持されます。

**/START DC** コマンドが出されると、IMS は、ARNR で指定されたセッションのうち、ABEND 時にまだアクティブであったすべてのセッションを自動的に再接続します。NRNR で指定されたセッションは終了されます。

**/STOP DC** コマンドとその後に **/CHECKPOINT** コマンドを続けて使用して IMS がシャットダウンされた場合は、VTAM 端末はそれぞれのセッションごとに切断されるため、IMS 再始動後に自動セッション再接続が起こることはありません。

## XRF または VGR との RNR の併用

RNR は、XRF を使用して構成されたシステムではサポートされません。XRF が構成されたシステムで RNR を活動化しようとしても、RNR を使用不可にして実行が継続されることを示す警告メッセージが作成されます。RNR は、VGR と一緒に使用することができます。ただし、IMS RNR のサポートの方が VGR サポートに優先します。

IMS 再始動時に **/START DC** コマンドが入力されたとき、RNR=NRNR であれば、セッションは終了し、VGR (アクティブの場合) は、適切なレベルの類似性および端末状況管理を実行します。IMS 再始動時に **/START DC** コマンドが入力されたとき、RNR=ARNR であれば、そのセッションは再接続のためにスケジューラされ、呼び出し側 VGR が、別の IMS との新規セッションを設定することはできません。IMS がロード・スタートされた場合は、アクティブ VTAM セッションはすべて終了し、アクティブの VGR 類似性および状況はすべて削除されます。

**制約事項:** RNR は OTMA 接続をサポートしません。

## 端末再接続プロトコル

RNR が活動化された端末に使用される再接続プロトコルは、処理中の作業の種類および障害発生時に使用中であった端末のタイプによって異なります。

以下の表は、セッション障害後にそれぞれの端末タイプに使用される再接続プロトコルを示しています。すべての端末で RNR が活動化されていると想定しています。セッションを再接続できない場合には、エラー・メッセージ DFS2050 または DFS2055 がマスター端末オペレーターに送信されます。

表 75. 高速ネットワーク再接続プロトコル

端末タイプ	接続プロトコル	端末サブタイプ	生成されるメッセージ
SLU1	SNA CLEAR	• 静的セッション	DFS3650 <sup>1</sup>
	SNA SDT	• ETO プリンター	
		ETO 非プリンター	DFS3649
	SNA CLEAR	静的セッション	DFS3650
SLU2	SNA SDT	静的セッション	DFS3649
	SNA CLEAR	静的セッション	DFS3650
NTO	SNA SDT	ETO セッション	DFS3649
	SNA UNBIND	静的セッション	DFS3650
SLU0 (非 SNA 3284 および 3286)	SNA BIND	ETO セッション	DFS3649

表 75. 高速ネットワーク再接続プロトコル (続き)

端末タイプ	接続プロトコル	端末サブタイプ	生成されるメッセージ
SLU0 (金融機関および SLUP) <sup>2</sup>	SNA CLEAR SNA STSN SNA SDT	ETO セッション	メッセージなし
ISC (1 次ハーフセッション)	SNA UNBIND SNA BIND SNA STSN SNA SDT		メッセージなし
ISC	SNA UNBIND		メッセージなし
ETO ALOT=0 <sup>3</sup> 付き	UNBIND UNBIND		メッセージなし

注:

1. メッセージ DFS3649 は、サインオンが必要であることを示します。メッセージ DFS3650 は、サインオンが不要であることを示します。
2. 単一ノード持続セッションの場合のみ、再接続が可能です。
3. サインオン・データを、ログオン・ユーザー・データとして、または DFSLGNX0 出口によって提供しなければなりません。

## サインオン・セキュリティ

RNR が活動化されているかいないかに応じて、および使用中の端末のタイプに応じて、IMS の再始動後に、VTAM セッションのサインオンまたはログオンが必要になる場合があります。

RNR が活動化されていない場合には、セッションのログオンとサインオンの両方が必要になります。RNR が活動化されている場合、以下のタイプの端末では、セッション再接続の後にサインオンを行う必要があります。

- SLU0 (非プリンター、3270、非 SNA)
- SLU1 (非プリンターのみ)
- SLU2
- NTO

RNR が活動化されている場合、以下のタイプの端末は、セッション再接続の後に自動的にサインオンします。

- SLU0 (金融機関/3600)
- SLU0 (SLUP)
- SLU1 (3284、3286、非 SNA)
- ISC (LU 6.1)



## 第 24 章 ネットワークの定義

IMS 端末ネットワークに関するシステム定義の実行には、いくつかのタスクが含まれます。

### このタスクについて

タスクには以下のものがあります。

- 端末の名前の定義
- 端末装置タイプの定義
- バッファ・サイズなどのオプション・パラメーターの指定

IMS ネットワークは静的端末と動的端末 (ETO と LU 6.2 装置) より構成されます。IMS に ETO または LU 6.2 装置を定義する必要はありません。

IMS ネットワークにおける静的端末だけがシステム定義マクロで定義されたとしても、静的端末と動的端末の両方を VTAM に定義する必要があります。VTAM ネットワーク生成時に指定するパラメーターによって IMS でのいろいろな要件を調整する必要があります。以下のトピックでは、そのような調整アクティビティーを中心に説明しますが、独立した IMS システム定義を一連の VTAM 定義および NCP 定義と詳しく関連付けることは行いません。

### 関連概念

[データ通信マクロによる端末の定義 \(システム定義\)](#)

## 操作可能ネットワークの準備

ネットワークの設定において、IMS システム定義要件を対応する VTAM 生成に組み入れる必要があります。さらに、全体的な進み具合とハードウェア・インストール・スケジュールを管理する必要があります。

### このタスクについて

IMS ネットワークを設定するための本質的な作業としては以下のような項目があります。

- 物理端末に必要な IMS 要件の収集。

さらに、端末の要件に関する以下の情報を収集する必要があります。

- 静的端末に必要なステージ 1 IMS システム定義の入力データを編成する。
- 動的に割り振られた端末に必要な ETO 記述子と表示画面特性を定義する。
- LU 6.2 記述子を定義する。
- IMS システム設計に必要な文書化の一部として 端末プロファイルを使用する。

- インストール・システムでいろいろな用途に利用可能な端末と IMS 用に指定した端末を突き合わせる。

高機能端末のユーザーが多数いるが、それらの端末が IMS オンライン・システム専用になっていない可能性があります。ネットワーク生成担当者に、それらの端末を知らせて、エンド・ユーザーに望む使用法を明確にする必要があります。

- IMS システム定義パラメーターをネットワーク生成のなかで対応するものと突き合わせる。

装置を正しく機能させるには、IMS システム定義パラメーターを VTAM および NCP のネットワーク・パラメーターと突き合わせる必要があります。バッファ・サイズを定義する時は特に注意を払ってください。IMS とネットワーク・バッファ・サイズは互換であることが必要です。ETO と LU 6.2 装置に関するシステム定義は IMS では必要ありませんが、VTAM には定義する必要があります。

- ハードウェア・インストール計画のモニター

ハードウェアのインストールとネットワークの生成の進行状況をトラッキングする計画を作成して、実動モードへの変更が計画どおりに実施されることを確認します。このトラッキングには、端末のハードウェアをインストールし保守するハードウェア・スペシャリストによる助言が必要になる場合があります。

- 端末装置インストールの予想

インストールを予想して端末を IMS に事前設定する場合があります。

### 関連タスク

71 ページの『拡張端末オプションの管理』

IMS 拡張端末オプション (ETO) を使用すると、VTAM 端末とユーザーをシステム定義時に事前定義しなくても、IMS に動的に追加できるようになります。

23 ページの『CPI 通信および APPC/IMS』

これらのトピックでは、CPI 通信と APPC/IMS を紹介します。このトピックでは、CPI 通信ドリブンのアプリケーション・プログラムの機能する方法、APPC/IMS を管理し、CPI アプリケーション・プログラムを構築するために CPI 通信インターフェース付きの APPC/IMS を使用する方法を説明します。

## IMS 定義とネットワーク定義の調整

以下のトピックで説明するネットワーク要件に関する入力を、VTAM と NCP に提供します。

### ホスト・サブシステムとしての IMS の使用

IMS 制御領域をアプリケーションとして VTAM に定義する必要があります。インストール先で名前を決めてください。この名前は SYS1.VTAMLST システム・データ・セットに追加されるアプリケーション・ノードの定義で使用されます。

#### このタスクについて

**例:** 制御領域定義の例を以下に示します。

```
IMS  APPL  AUTH=(ACQ),PRTCT=password
```

この例では、サブシステムとの通信を開始するためにはパスワードが必要であることを示しています。COMM マクロの PASSWD キーワードを使用して、IMS システム定義のために対応するパスワードを指定してください。VTAM 定義で IMS サブシステムの名前として IMS を使用していない場合は、IMS していない場合は、IMS 用に選択したサブシステム名を COMM マクロに APPLID キーワードを使用してコーディングしてください。デフォルトの APPLID 名は IMS です。

### VTAM ノードの定義

ローカル・ノード名も SYS1.VTAMLST に追加されます。

#### このタスクについて

IMS システム定義で使用されている正確な端末名が、SYS1.VTAMLST データ・セットのローカル・リスト・メンバーで繰り返されることに注意してください。端末のその他の特性は VTAM 項目で指定されるので、IMS パラメーターと VTAM パラメーターの間に矛盾がないことを確認する必要があります。ノード名と関連付けられるものは、次のとおりです。

- プログラム機能キーか選択ペンの使用
- キーボード特性
- 画面サイズ
- 型式番号
- LU タイプ
- 伝送サービス・レベル

IMS は、動的端末の画面サイズ、型式、LU タイプ、および伝送サービスを検査します。それらが間違っている場合、セッションは設定されません。

**推奨:** DC 管理者は、VTAM 定義が正しいことを確認してください。これらの定義は ETO 記述子と MFS 形式特性を選択するために使用されます。静的端末の場合は、IMS システム定義がこれらの定義を設定し、VTAM 定義は無視されます。

## 関連概念

[VTAM 端末の定義 \(システム定義\)](#)

## VTAM ストレージ要件の見積もり

IMS ネットワークのために計画しているすべての端末オプションを調べて、VTAM ストレージ要件に影響がある通信オプションを選択してください。

### このタスクについて

例えば、次のようなオプションを選択します。

- プロトコル (BID または NOBID)
- 応答パターンのタイプ
- アクティブ・ノードの数

端末プロファイルを使用することによって、IMS サポートに必要な記憶容量を見積もるシステム・プログラミング情報の要求に応えることができます。

## VTAM バッファ・プール値の決定

IMS 入出力メッセージ・サイズを VTAM システム・プログラマーに知らせてください。COMM マクロの RECAN Y キーワードで指定する、受信用バッファの数とサイズに選択する値は非常に重要です。

### このタスクについて

VTAM ネットワークをサポートするために必要な受信用バッファの最大数を指定しなければなりません。サイズは、実行パラメーター RECASZ で、数は、実行パラメーター RECAN Y でオーバーライドすることができます。システムに追加された端末が、現行で指定されている受信用バッファ・サイズより大きいサイズを必要とする場合、それらの端末からのセッション開始は失敗します。端末が IMS とのセッションを開始できるようにするには、もっと大きなバッファ・サイズを指定して IMS システムを再始動する必要があります。

## NCP バッファ・プール値の決定

NCP バッファ・プール・サイズとしきい値は、VTAM バッファ・プール情報に基づいています。

### このタスクについて

システム・プログラマーと一緒に、アプリケーション・プログラムの予測トラフィック量を評価してください。NCP システム定義パラメーターに対する特定の IMS 要件はありません。

## 静的および動的な端末サインオン要件の決定

VTAM が定義するすべての端末は、サインオンの要、不要にかかわらず、セッション開始要求と同時に入力されたユーザー・データによってサインオンすることができます。

### このタスクについて

複数のサインオンを許可する SGN=G、M、または Z を始動プロシージャで指定しない限り、1人のユーザーを複数の端末に同時にサインオンすることはできません。そうでないと、IMS はメッセージ DFS3649A または DFS2467I を出します。

ユーザーは、ログオンあるいは /SIGN コマンドを使用して出力専用装置からサインオン・データを入力できません。サインオンを必要とする出力専用装置の場合は、サインオン・データがセッション開始時に省略されていると、メッセージ DFS2085I が MTO に送信されます。

## 複数のサインオン

システム定義時に定義された静的 LTERM と同一の名前を、動的 LTERM に使用することはできません。

IMS プロシージャまたは DCC プロシージャ のいずれかで SGN=M を指定すると、ユーザーは 1 つ以上の端末 (静的端末または動的端末) に並行してサインオンできます。

サインオンが拒否された場合、IMS マスター端末はセキュリティー違反メッセージ DFS286 を受信します。このメッセージを受け取りたくない場合は、COMM マクロ、IMSGEN マクロ、または SECCNT 初期設定パラメーターで SECCNT を 0 に設定してください。

## セッション状況メッセージ DFS3650I

サインオンが実行されたとき、またはサインオンが不要な場合は、IMS とのセッションの状況を示すメッセージ DFS3650I がユーザーに送信されます。

**例外:** 以下の端末はこのメッセージを受け取りません。

自動ログオン端末  
不在モードで実行中の SLU-1 端末  
ISC  
SLU P  
3600/ 金融機関

また、TERMINAL マクロか ETO ユーザー記述子で NOTERM が指定された場合、DFS3650I メッセージは受信されません。

DFS3650I メッセージ・パネルの情報は、ユーザーが会話モードかどうかを表示します。この状況表示パネルはユーザー出力セキュリティーがその端末に設定してあるかどうか也表示しています。

## 関連タスク

61 ページの『[拡張端末オプション \(ETO\) \(Extended Terminal Option \(ETO\)\)](#)』

これらのトピックでは、拡張端末オプション (ETO) を紹介し、ETO の概要および IMS TM ネットワーク内の ETO 端末の管理に必要な情報を説明します。

## 関連資料

[COMM マクロ \(システム定義\)](#)

[IMSGEN マクロ \(システム定義\)](#)

## 関連情報

[DFS3650I \(メッセージおよびコード\)](#)

# LOGON MODE テーブルに関する要件の検査

静的 VTAM 端末が IMS に接続され、セッションが開始するとき、VTAM は LOGON MODE ID を知っている必要があります。

## このタスクについて

VTAM LOGON MODE テーブルには、セッションのための SNA バインド・パラメーターと該当端末の ID (ログオン時に使用される LOGMODE パラメーター値) が含まれています。各 SDLC 接続端末ごとにデフォルトの ID も存在しています。

これらの考慮事項は静的 VTAM 端末に適用されます。

## LOGON MODE ID の調整

LOGON MODE ID を決定する場合、以下のことが必要になります。

## 手順

- インストール・システムの VTAM ネットワークで使用されている命名規則と一致させる。名前の選択は、この規則に従う必要があります。

- IMS リモート端末の自動開始を用意するか、あるいはVTAM オペレーターに制御させる。あるいは、リモート端末オペレーター (RTO) が VTAM LOGON コマンドを入力してセッションを開始する場合には、リモート端末オペレーターに内容が分かるような名前を選択してください。
- セッションが /OPNDST コマンドで開始する場合でも、マスター端末オペレーターに分かるような名前を選択する。端末が異なる SNA プロトコルで操作される場合でも、その名前が操作モード間でも判別できるようになっている必要があります。
- MODETBL キーワードを TERMINAL マクロまたは ETO ログオン記述子を使用して、以下の状況で使用するデフォルトの ID を指定する。
  - デフォルト ID が使用されない場合。
  - 端末オペレーターが LOGON コマンドで ID を指定しない場合。
  - マスター端末オペレーターが /OPNDST コマンドで ID を指定しない場合。

## 例

下記コードは、3270 SDLC 接続または 4730 装置に関して VTAM に記述される LOGON MODE テーブル項目の性質を示しています。IMS では、これらをそれぞれ SLU 2 および SLU P と定義しています。

```
SE3270  MODETAB
IBMS3270  MODEEENT LOGMODE=S3270,FMPROF=X'02',TSPROF=X'02',      X
          PRIPROT=X'71',SECPROT=X'40',COMPROT=X'2000'
MT4730  MODEEENT LOGMODE=MT4730,FMPROF=X'04',TSPROF=X'04',      X
          PRIPROT=X'B1',SECPROT=X'B1',COMPROT=X'6080',      X
          RUSIZES=X'0000'
MODEEEND
```

オペレーターは、VTAM LOGON コマンドを入力するか、あるいはインストール・システムで設定した ログオン構文の VTAM USSTAB を使用することができます。

次の例では、KEYWORD=*value* の形式を指定しています。

```
U3270  USSTAB
        USSCMD      CMD=IMS,REP=LOGON
        USSPARM     PARM=APPLID,DEF=IMS
        USSEND
        END
```

## 次のタスク

### マスター端末でのログオン要件

IMS は、IMS START コマンドが出された後に、マスター端末に自動的にログオンするので、VTAM デフォルトを使用すべきでない場合は、MTO として使用する物理装置の LU ステートメントで MODETBL キーワードをコーディングしなければなりません。

MTO が /OPNDST コマンドを使用すると、MODE オペランドはコマンドで与えられたすべてのノード名に適用されます。

### 関連タスク

61 ページの『[拡張端末オプション \(ETO\) \(Extended Terminal Option \(ETO\)\)](#)』

これらのトピックでは、拡張端末オプション (ETO) を紹介し、ETO の概要および IMS TM ネットワーク内の ETO 端末の管理に必要な情報を説明します。

## 初期 VTAM 構成の指定

IMS 要件を基にして、開始リスト (ATCSTRyy) と構成メンバー (ATCCONxx) は、SYS1.VTAMLST ライブラリに配置されます。この情報は、VTAM 始動時に活動化すべき端末、および始動時に VTAM に対して認識させる VTAM ノードを反映していなければなりません。



## このタスクについて

ISC (LU 6) 装置および MSC VTAM リンクでは、セッションが別のサブシステムから開始されると、セッションの自動的な回復は行われません。セッション障害メッセージが MTO に送信され、セッションを手動で再始動する必要があることを示します。

IMS の柔軟性を生かして、ただちに必要となるネットワーク・リソースを少なくするように努めてください。これにより、開始リストと構成メンバーに関してより詳しい始動命令が必要になるかもしれません。このことは、/OPNDST コマンドを使用する範囲および MTO 命令にも反映してください。ネットワーク・セクションの始動を遅らせて、IMS 始動を早めるために IMS 処理とオーバーラップさせることができます。また、ネットワークの一連の立ち上げは、IMS 内で実行される自動化操作プログラムにより行うことができます。

## IMS における SON/COS サポートの使用

セッション停止通知 (SON) とサービス・クラス (COS) は、IMS がセッション停止を認識できるようにする VTAM と SNA の機能です。

IMS はリカバリー可能なセッション障害のセッションを再始動しようとします。SON と COS の両方は、IMS で利用可能にするには VTAM に指定する必要があります。さらに、システム定義処理または /CHANGE コマンドを使用して、IMS ノードに対して ASR オプションを指定する必要があります。

e IMS SON および COS サポートはすべての VTAM SNA 端末と MSC VTAM リンクに利用可能です。複数の仮想経路が IMS と他の LU の間に存在するときは、代替経路を使用して、ネットワーク停止を回避してください。

SON 機能によって、VTAM はセッション障害が起きたことを IMS に知らせることができます。リカバリー可能セッション障害がネットワークに起きた場合、SON 機能は、インストール・システムに IMS システムに IMS が自動的にセッションを再始動するオプションを与えることができます (この時、エンド・ユーザーによる介入は不要)。

セッション停止の後に自動セッション再始動を活動化するには、IMS アプリケーションの VTAM APPL 定義ステートメントに SONSCIP=YES をコーディングする必要があります。IMS に SONSCIP 定義を含めない場合、VTAM と IMS は自動セッション再始動をせずにセッション終了を続行します。

COS 機能は、トラフィックの速度、データ・タイプ、セキュリティの考慮事項に基づいた仮想通信経路のセットを指定することによって、VTAM が実際の経路選択を制御できるようにします。これらの経路のセットは VTAM COS テーブルに定義されます。セッション設定に使用されるモード・テーブル項目で、COS 項目名を指定することができます。

端末に関連するセッション BIND パラメーターのセットに COS 名が含まれている場合は、これらの指定された仮想経路がスキャンされ、どの仮想経路をこのセッションに使用できるかが判別されます。選択された仮想経路が失敗した場合は、セッションの再始動によって COS セットが再びスキャンされ、セッション再設定に別の仮想経路を使用できるかどうか調べられます。

プログラム LU (金融機関、LU P、MSC、および ISC と同じように) の場合、伝送中のメッセージは必要に応じて回復され再送されます。これによって、再始動を通じてのメッセージ通信の安全性が保たれます。

装置 LU の場合、経過中のメッセージは失われるか重複する可能性があります。

SON および COS サポートは、新規セッションを再設定するために、同じモード・テーブル項目名を使用することによって、失敗したセッションに関連するサービス・クラスと共にセッション・セットアップ・オプション (BIND パラメーター) を保存します。

**制約事項:** 境界ノード NCP はアップストリーム経路の障害と ホスト CPC の障害とを区別できないので、バックアップ・セッションを持つ XRF 構成の端末では、SON および COS サポートは使用できません。

### 関連資料

[/CHANGE コマンド \(コマンド\)](#)

## IMS ネットワークの開始

IMS が VTAM ログオン (セッション初期設定) 要求を受信できるようにするには、DC キーワードを指定した IMS **/START** コマンドを使用します。

### 始める前に

IMS でセッションが設定される前には、VTAM と NCP はアクティブにする必要があります。

### このタスクについて

**/START DC** コマンドは、IMS に属すると VTAM に定義されている論理装置に対するログオン要求と共に、IMS に対する待機 VTAM ログオン要求を IMS に渡すよう VTAM に伝えます。

**/START DC** コマンドは以下のプロセスを活動化します。

- IMS データ通信を開始する。
- VTAM アクセス方式制御ブロックをオープンする。
- IMS VTAM ログオン出口ルーチンを使用可能にする。

IMS **/START DC** コマンド実行の前でかつ、IMS VTAM アクセス方式制御 ブロック (ACB) がオープン後に、VTAM が受信したすべてのログオン要求は、**/START DC** が完了するまで VTAM のキューに保管されます。IMS の初期設定時に VTAM がアクティブであると、IMS VTAM ACB がオープンされます。VACBOPN=DELAY が指定された場合、VTAM ACB のオープンは、**/START DC** コマンドが入力されるまで遅延されます。

APPC/IMS を開始するには、**/START APPC** コマンドを使用してください。

## セッション開始

セッションとは、論理装置 (端末 など) と VTAM アプリケーション・プログラム (IMS など) の間の論理接続のことです。データを論理装置と IMS 間で伝送するには、その前に論理装置と IMS 間のセッションを設定する必要があります。

### このタスクについて

セッション開始は以下の 5 つの方法のうちのどれか 1 つで要求されます。

- 端末オペレーターはログオン・シーケンスを入力します。VTAM がコマンドを検証し、IMS にその要求を渡します。

端末オペレーターは、以下の名前を使用して、ログオン・シーケンスで IMS を識別することができます。

- アプリケーション ID 名 (端末オペレーターが特定の IMS とのセッションを要求している場合)。
- MNPS ACB 名 (端末オペレーターが MNPS を使用する XRF システムとのセッションを要求している場合)。
- USERVAR (端末オペレーターが USERVAR を使用する XRF システムとのセッションを要求している場合)。
- 汎用リソース名 (端末オペレーターが汎用リソース・グループとのセッションを要求している場合)。
- z/OS VTAM ネットワーク・オペレーターは、LOGON オプションを指定した VTAM **VARY** コマンドを使用して、論理装置に代わってセッション開始を要求します。VTAM はその要求を処理し IMS に渡します。
- VTAM は、IMS に属すると VTAM に定義されている各論理装置ごとに、ログオン要求を IMS に渡します。
- IMS マスター端末オペレーターは、IMS **/OPNDST** コマンドを入力して論理装置に対するセッション開始を要求します。
- ETO 自動ログオンは、ユーザー記述子および出口ルーチンに基づいて、セッションを開始し、適切なユーザー・データを提供します。

LU 6.2 の場合は、セッション開始は会話が割り振られると自動的に始まります。IMS MTO では /**ALLOCATE** コマンドを発行してキューにある LU 6.2 の出力の送達を要求することができます。LU 6.2 を使用する時は、LU 6.2 会話の IMS アプリケーション名 (LU 名) は、非 LU 6.2 タイプに IMS が使用する LU 名とは異なること、ならびに /**START DC** コマンドと /**START APPC** コマンドが独立して機能することを覚えておいてください。

セッション開始がどのように要求されたかにかかわらず、IMS が要求を受信すれば同一の処理が始まります。

#### 関連概念

451 ページの『システム間でのプリンターの共用』

マスター以外のプリンターを共用して使用する場合は、IMS では、オンライン IMS システムと他のサブシステムと (他の IMS も可) が交互にプリンターを使用できるようになっています。

102 ページの『ETO を使用したプリンターの共用』

同じ出力端末を使用することにより、複数のユーザーがプリンターを共用できます。

## IMS トランザクション・タイプおよびトランザクション状態

トランザクションは、論理装置から IMS へ送られる最も一般的な データ・タイプです。

### このタスクについて

IMS は、2 種類のトランザクション (更新と照会) をサポートします。

#### 定義:

- 更新トランザクションでは、データベースを変更することができます。
- 照会トランザクションでは、データベースのデータを見ることはできますが、変更あるいは更新はできません。

トランザクションは、IMS システム 定義で 更新あるいは照会のいずれかに定義してください。

照会トランザクションについては、リカバリー可能かリカバリー不能かが追加属性として定義されます。

#### 定義:

- リカバリー可能と定義した照会トランザクションは、どのネットワークの要素で失敗してもいつでもリカバリー可能です。
- リカバリー不能と定義した照会トランザクションは、入出力エラー状態または IMS システム 再始動の後では回復されません。

更新トランザクションは、すべてリカバリー可能です。高速機能トランザクションは、すべてリカバリー可能として定義する必要がありますが、照会か更新かということについては、どちらでも構いません。

LU 6.2 トランザクションがリカバリー可能であるのは、非同期プロトコルを使用してそのトランザクションを開始した場合だけです。LU 6.2 プログラムからの IMS 会話型トランザクションは、ローカル・システムではリカバリー可能ではありませんが、MSC リンクをまたがる場合にはリカバリー可能です。

IMS は、リカバリー可能性を実現するために必要な処理を行わないことを除いて、リカバリー不能トランザクションをリカバリー可能トランザクションと同じように取り扱います。この結果、リカバリー不能トランザクションの処理時間は短縮されますが、ネットワーク障害 (例えば、回線障害、CPC 障害、またはキュー障害など) が発生した場合にそのトランザクションが失われる可能性があります。

宛先作成出口ルーチン (DFSINSX0) によってキューイング専用で作成されたトランザクションの状況は、DYN です。キュー専用トランザクションの唯一の目的は、共用キューにメッセージを入れることです。キュー専用トランザクションは、停止されたか、まだチェックポイントが指定されていなかった場合を除いて、再始動時にリカバリーされません。

### トランザクション状態の判別

トランザクションは、多数の状態のいずれかを取ります。

可能なトランザクション状態には、次のものがあります。



## LOCK

LOCK 状態のトランザクションはメッセージを受信しません。このトランザクション・コードを含むスケジュールされたメッセージは停止されます。

**制約事項:** /LOCK TRANSACTION コマンドは、高速機能専用トランザクションでは使用できませんが、高速機能利用可能トランザクションでは使用できます。

## MODSTOPPED

MODSTOPPED 状態のトランザクションは、オンライン変更処理が進行中のため 入力を受信できません。/MODIFY COMMIT コマンドは、このような状況を設定します。CPI 通信ドリブン・トランザクションには MODSTOPPED のマークを付けることはできません。

## PSTOP

PSTOP 状態のトランザクションはスケジュールできませんが、限界カウントに到達するまでトランザクションの処理を継続します。限界カウントが大きい場合、処理間隔も長くなります。トランザクション状況を確認するには、/DISPLAY コマンドを使用してください。トランザクション状況を変更するには、/ASSIGN コマンドを使用してください。

## PURGE

PURGE 状態のトランザクションは、入力メッセージが停止されています。

## QSTOP

QSTOP 状態のトランザクションは、/MODIFY PREPARE コマンドの完了とそれに対応する /MODIFY COMMIT または /MODIFY ABORT コマンドの完了の間に入力できません。オンライン変更の内容に影響されるトランザクションは受け入れられません。すなわち、トランザクションが変更または削除されるか、あるいは変更または削除されるデータベースかプログラムをトランザクションがアクセスできるかです。/DISPLAY MODIFY コマンドを使用することで、現行のオンライン変更の影響を受けるトランザクション・リストを表示することができます。端末では、そのようなトランザクションはメッセージ DFS3470 で拒否されます。トランザクションが変更あるいは削除される高速機能の宛先コードを使用した場合、トランザクション拒否メッセージは DFS3471 です。CPI 通信ドリブン・トランザクションは QSTOP 状態になることはありません。

## STOP

STOP 状態のトランザクションは停止されます。宛先がトランザクションあるいはトランザクションのクラスであるメッセージのキューイングとスケジューリングは停止されます。しかし、出力がアプリケーション・プログラムで作られている場合はキューに加えられます。

## USTOPPED

USTOPPED は、アプリケーション・プログラムが IMS DL/I データベース・リソースを使用しようとしたが、そのリソースが利用不能な場合に設定される状況値で、この場合、プログラムは ABEND U3033 で異常終了します。USTOPPED 状態は、CPI 通信ドリブン・プログラムには設定されません。CPI 通信ドリブン・アプリケーション・プログラムは ABEND U0125 で異常終了します。

## 高速ネットワーク再接続 (RNR) についての VTAM の定義

高速ネットワーク再接続 (RNR) を VTAM セッションで使用するには、IMS に対する APPL ステートメントを使用して、望ましい持続セッション・サポートのレベルについて VTAM を定義する必要があります。また、ユーザーのセキュリティー要件および露出を評価し、端末の RNR サポートの適切なレベルを定義する必要があります。

### 関連概念

421 ページの『[高速ネットワーク再接続 \(RNR\) の計画](#)』

高速ネットワーク再接続 (RNR) は、障害 (IMS、z/OS、CPC、または VTAM) の発生の後 IMS VTAM 端末セッションを自動的に再接続し、その後の IMS は IMSplex 内の同じまたは別の CPC 上で再始動します。RNR を使用すると、VTAM セッションの可用性が向上し、障害発生後にセッションの終結処理と IMS の再始動を行う必要がなくなります。

## 持続サポート・レベルの定義

VTAM 持続セッション・トラッキングは、単一ノード持続セッション (SNPS) および マルチノード持続セッション (MNPS) の両方に用意されています。どちらのレベルの持続サポートが RNR に必要であるかを示さなければなりません。

### このタスクについて

MNPS を使用した場合は、必要があれば、VTAM セッションを シスプレックス内の別の CPC に再接続することができます。SNPS を使用した場合は、VTAM セッションを、停止が起こった時に接続されていたのと同じ CPC に再接続しなければなりません。

VTAM に必要な IMS 持続セッション・サポートのレベルは、APPL 定義ステートメントで PERSIST=MULTI | SINGLE パラメーターを使用して指定されます。デフォルト設定は PERSIST=SINGLE です。PERSIST 指定が入力されなかった場合は、PERSIST=SINGLE と見なされます。

## RNR サポート・レベルの定義

RNR サポートは、システム・レベル、端末レベル、およびセッション・レベルの 3 つのレベルで定義することができます。

### このタスクについて

3 つのレベルすべてを使用して RNR サポートを制御するには、以下を行ってください。

- IMS 実行パラメーターを更新して、RNR を活動化します。
- 動的端末サポートのために適切な IMS ETO ログオン記述子を更新します。
- RNR オプションの動的オーバーライドをセッション単位で行うように、DFSLGNX0 ログオン出口サポートを更新します。

システム・レベル (かつデフォルト) で RNR を活動化するには、DFSDCxxx IMS.PROCLIB メンバーに RNR=ARNR (RNR の活動化) または RNR=NRNR (RNR なし) のいずれかを指定して、RNR を活動化するかどうかを指示しなければなりません。

DFSDCxxx IMS.PROCLIB メンバー内の PSTIMER パラメーターを使用して、IMS または VTAM 障害後の最大セッション持続時間を指定することもできます。PSTIMER は、1 から 86400 秒までの値に設定できます。デフォルト設定は 3600 (1 時間) です。0 が指示された場合は、タイマーは使用されず、セッション持続性は無期限に続きます。

注: MNPS の VTAM START オプション HRPST が PSTIMER より小さい値に設定されると、VTAM START は PSTIMER 設定をオーバーライドします。

ETO 動的端末の RNR サポートを制御するには、ETO ログオン記述子で OPTIONS=ARNR | NRNR パラメーターを使用します。このレベルで指定された RNR サポートは、IMS 実行パラメーターを使用してシステム・レベルで設定された値をオーバーライドします。

RNR サポートをセッション単位で制御するには、ログオン出口ルーチン (DFSLGNX0) で LGOPT=LGOARNR | LGONRNR パラメーターを使用します。このレベルで指定された RNR サポートは、ETO ログオン記述子またはシステム・レベル実行パラメーターを使用した設定をオーバーライドします。

DFSLGNX0 のパラメーターは、ログオン出口ルーチンのソース・コードで文書化されています。

## 第 25 章 IMS メッセージの編集とフォーマット設定

IMS、メッセージ形式サービス (MFS) および基本編集ルーチンの 2 つの方法によって、端末でやり取りするメッセージの編集およびフォーマット設定を行います。

### このタスクについて

IMS では、以下を編集するように設計できるユーザー作成出力ルーチンのサンプルを提供しています。

- 端末からの入出力
- トランザクション・コード
- 入力メッセージ・フィールド
- 入力メッセージ・セグメント
- メッセージ通信

このトピックでは、MFS の利点を概説し、メッセージのフォーマット設定のための MFS 制御ブロックについて紹介し、MFS がサポートする各種装置の特性および MFS の管理者の責任について要約しています。

**制約事項:** MFS では、LU 6.2 装置または OTMA はサポートされません。暗黙 API のサポートが使用されている場合、LU 6.2 装置からの入出力メッセージ用に、LU 6.2 編集出力ルーチン (DFSLUEE0) が提供されます。OTMA からの入出力メッセージ用には、OTMA 入出力編集 ユーザー出口 (OTMAIOED) が提供されます。

### 関連資料

[OTMA 入出力編集ユーザー出口 \(DFSYIOE0 およびその他の OTMAIOED タイプの出口\) \(出力ルーチン\)](#)  
[LU 6.2 編集出力ルーチン \(DFSLUEE0\) \(出力ルーチン\)](#)

## メッセージ形式サービス

メッセージ形式サービス (MFS) は、IMSS アプリケーション・プログラムが入力メッセージまたは出力メッセージ内の装置固有の特性に対応せずにすむように、端末との間でやり取りするメッセージをフォーマット設定する IMS 機能です。

MFS は、リモート・コントローラーやサブシステム内のユーザー作成プログラムとの間のメッセージもフォーマット設定するので、ホスト・アプリケーション・プログラムは、リモート・コントローラーの端末固有の特性に対応する必要はありません。

MFS は、制御ブロックを使用します。制御ブロックは、入出力メッセージの配置方法を IMS に指示するためにユーザーが指定するものです。

- 入力メッセージの場合、MFS 制御ブロックは、装置がアプリケーション・プログラムに送ったメッセージをアプリケーション・プログラムの入出力域でどのように配置するかを定義します。
- 出力メッセージの場合、MFS 制御ブロックは、アプリケーション・プログラムによって装置に送られたメッセージを画面上またはプリンターでどのように配置して出力するかを定義します。また、プログラムの入出力域ではなく画面上に出るリテラルのようなデータも定義することができます。

IMS システムでは、アプリケーション・プログラムと端末またはリモート・プログラム間で渡されるデータは、MFS あるいは基本編集で編集できます。MFS の提供する機能は、ネットワークが使用する端末タイプや 2 次論理装置 (SLU) によって異なります。

MFS を使えば、アプリケーション・プログラマーが装置依存のデータの代わりに論理メッセージで処理することができます。したがって、アプリケーション開発が簡略化されます。装置入出力は装置タイプにより異なりますが、同じアプリケーション・プログラムで、1 つの論理を使用して異なる装置タイプを処理することができます。アプリケーション・プログラムを変更することなく装置上のデータの表示やオペレーターの入力を変更することができます。IMS のディスプレイ装置には、全ページング機能があります。入力メッセージを、複数のデータ画面から作成することができます。

MFS を使用するプログラムは、特定の装置機構を使用する場合を除いて、入出力メッセージに使用される装置の物理的特性を考慮して設計する必要がありません。特定の装置機構を使用する場合でも、プログラムや装置への表示の援助を MFS に対して要求することができます。

MFS は、SLU タイプの装置 SLU-1、SLU-2、SLU-P、金融機関、および LU 6.1 をサポートします。また、MFS は IBM 3270、および 3600 などの古い装置でも使用可能です。

IBM 3270 や SLU 2 装置の場合、装置制御文字や命令を MFS バイパス機能を使用して、プログラムから直接送信したりまたはプログラムで直接受信することができます。これにより、アプリケーション・プログラムはより直接的にデータ・ストリームの制御ができます。プログラムは予約されたフォーマット名を使用し、これにより、MFS は以下のものの編集をバイパスします。

- 出力メッセージ
- ディスプレイ端末から受け取る次の入力メッセージ

IBM 3270 および 3604 ディスプレイ装置には、論理ページングと物理ページングの両方の機能があります。この機能により、アプリケーション・プログラムでは大量データを書き込むことができ、MFS ではデータを端末上で複数の表示画面に分割することができます。端末オペレーターは、メッセージ内で別の表示画面の上方、下方へそれぞれのページ送りができます。

## MFS コンポーネント

MFS にはいくつかのコンポーネントがあります。

MFS コンポーネントには以下のものがあります。

- MFS 言語ユーティリティ  
メッセージ・エディター
- MFS プール・マネージャー
- MFS サービス・ユーティリティ
- MFSTEST プール・マネージャー
- メッセージ形式サービス装置特性テーブル (MFS DCT) ユーティリティ (DFSUTB00)

MFS 言語ユーティリティはオフラインで実行され、制御ブロックを生成して、それらを IMS.FORMAT という名前の形式制御ブロック・データ・セットに入れます。制御ブロックは、メッセージ入出力操作中に起こるメッセージ形式を記述します。制御ブロックは、1 組のユーティリティ制御ステートメントによって生成されます。

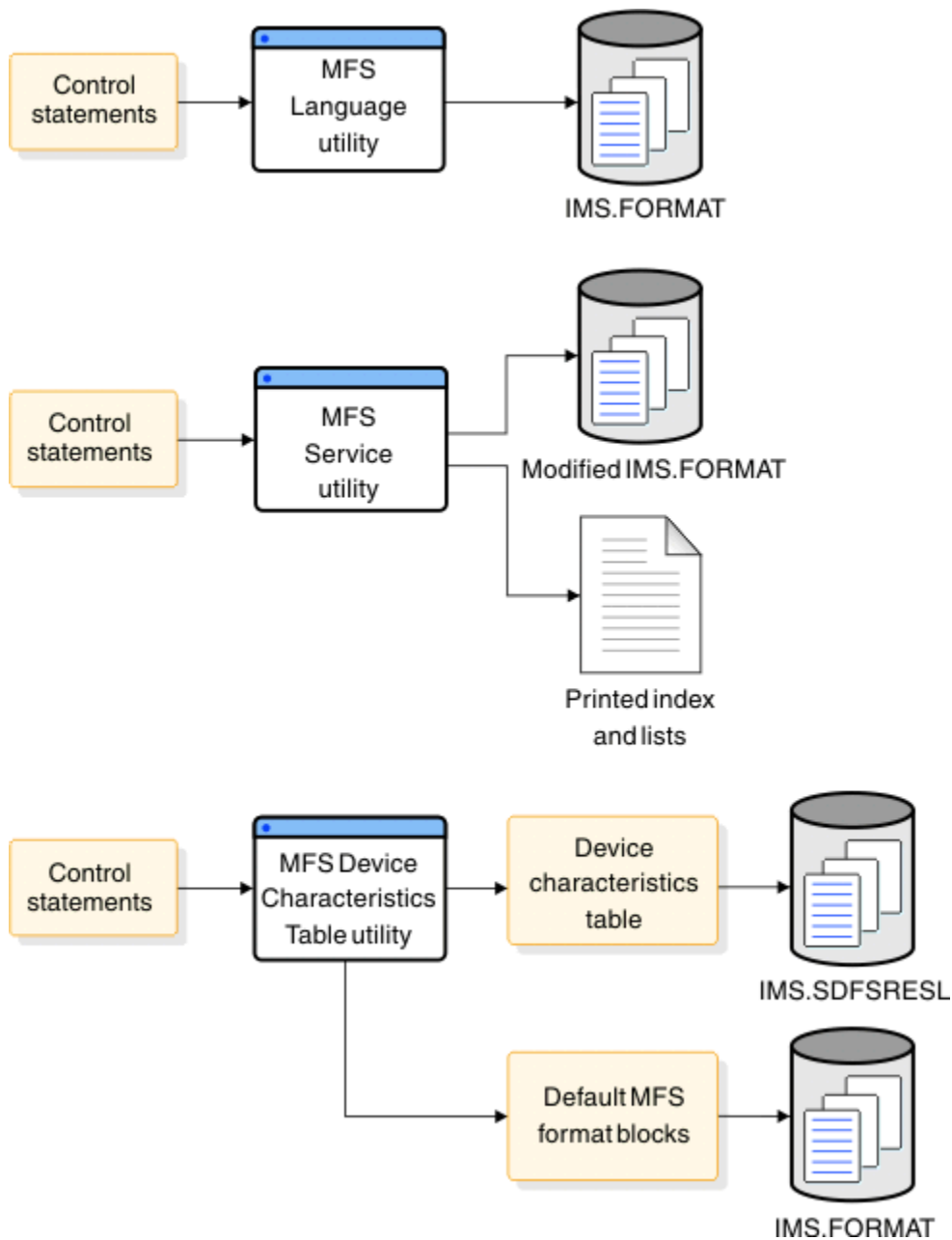


図 50. MFS ユーティリティとその出力

注：MFS テスト・モードでは、MFS 言語ユーティリティは オンライン IMS 制御領域として同時に実行できます。ただし、IMS のテスト・モードでないときは、MFS フォーマットを変更するにはオンライン変更プロシージャを使用する必要があります。

以下の図は、MFS オンライン環境を示します。図の後に リストされているステップは、図中の番号に対応しています。

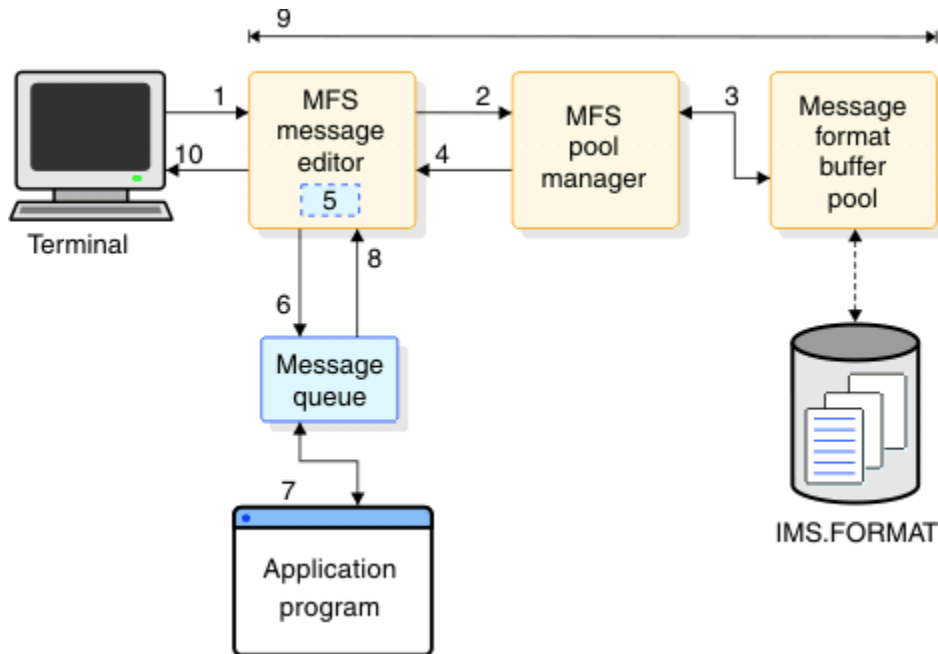


図 51. MFS オンライン環境の概要

1. 入力メッセージは端末から MFS メッセージ・エディターに送信される。
2. MFS メッセージ・エディターは、MFS プール・マネージャーから MFS ブロック へのポインターを要求する。
3. MFS プール・マネージャーはプールにブロックが存在しているかを見るためメッセージ・フォーマット・バッファ・プールをチェックする。ブロックが存在しない場合、MFS プール・マネージャーはブロックを IMS.FORMAT からバッファ・プールに読み込む。
4. MFS プール・マネージャーは、MFS ブロックのアドレスを MFS メッセージ・エディターに送信する。
5. MFS メッセージ・エディターは、アプリケーション・プログラム用に入力メッセージをフォーマット設定する。
6. MFS メッセージ・エディターは、フォーマット済み入力メッセージを処理するためにメッセージ・キューに送信する。
7. アプリケーション・プログラムは、メッセージを処理し、出力メッセージをメッセージ・キューに送信する。
8. 出力メッセージは、メッセージ・キューから MFS メッセージ・エディターに送信される。
9. MFS は、入力メッセージを処理したのと同様に (ステップ 438 ページの『2』から 438 ページの『6』) 端末の出力メッセージを処理する。
10. フォーマット済みの出力メッセージは端末に送信される。

以下の図は、MFS テスト環境を示します。図の後にリストされているステップは、図中の番号に対応しています。

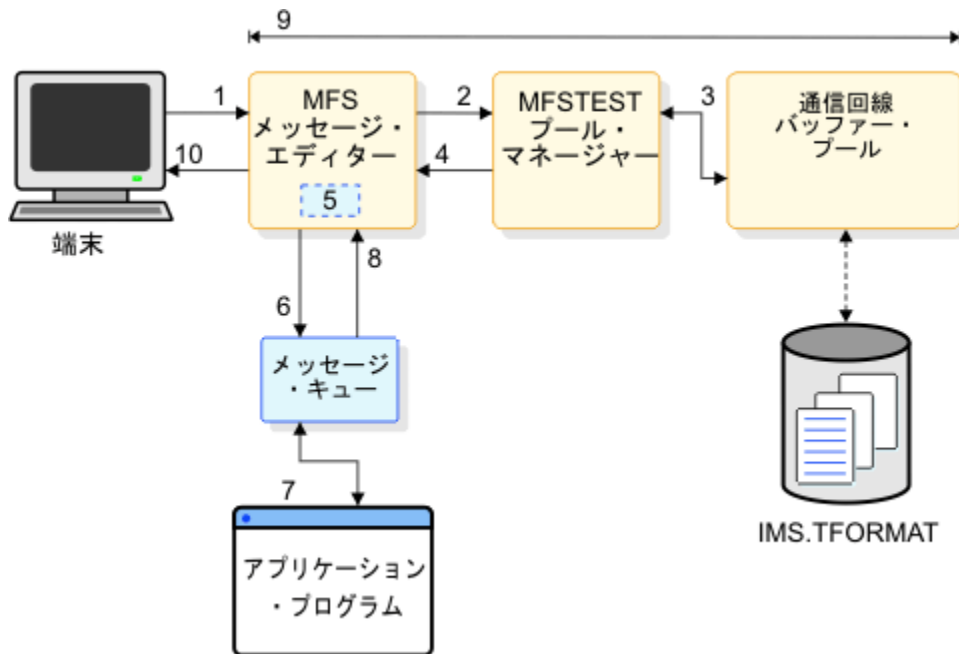


図 52. MFS テスト環境の概要

注: MFS テスト・モードを開始するには、**/TEST MFS** コマンドを使用する必要があります。

1. 入力メッセージは端末から MFS メッセージ・エディターに送信される。
2. MFS メッセージ・エディターは、MFSTEST プール・マネージャーから MFS ブロック へのポインターを要求する。
3. MFSTEST プール・マネージャーはプールにブロックが存在しているかを見るため通信回線バッファプールをチェックする。ブロックが存在しない場合、MFS プール・マネージャーはブロックを IMS.TFORMAT からバッファプールに読み込む。
4. MFSTEST プール・マネージャーは、MFS ブロックのアドレスを MFS メッセージ・エディターに送信する。
5. MFS メッセージ・エディターは、アプリケーション・プログラム用に入力メッセージをフォーマット設定する。
6. MFS メッセージ・エディターは、フォーマット済み入力メッセージを処理するためにメッセージ・キューに送信する。
7. アプリケーション・プログラムは、メッセージを処理し、出力メッセージをメッセージ・キューに送信する。
8. 出力メッセージは、メッセージ・キューから MFS メッセージ・エディターに送信される。
9. MFS は、入力メッセージを処理したのと同様に (ステップ 439 ページの『2』から 439 ページの『6』) 端末の出力メッセージを処理する。
10. フォーマット済みの出力メッセージは端末に送信される。

メッセージ・エディターと MFS プール・マネージャーは、標準操作モードではオンラインで実行されません。メッセージ・エディターは、操作制御ブロック仕様を使用して実際のメッセージ・フォーマット設定操作を実行します。

他の 2 つの MFS のコンポーネント、MFS サービス・ユーティリティーと MFSTEST プール・マネージャーは、オプションの MFS 操作のサポートに利用できます。

MFS サービス・ユーティリティーは、形式制御ブロック・データ・セットの追加制御を行うためのメソッドを提供します。これはオフラインで実行され、MFS プール・マネージャーがオンラインで使用するための制御ブロックの索引を作成および保守することができます。

オプションの MFSTEST 操作モードをサポートするときは、MFSTEST プール・マネージャーが、MFS プール・マネージャーに取って代わります。MFS キーワードを持つ /TEST コマンドは、論理端末を MFSTEST



モードにします。各端末が MFSTEST モードにある場合は、一時的な形式ブロックをすでに実動モードにある他のブロックと結合して使用することができます。そうすると、新しいアプリケーションや既存のアプリケーションに対する変更を、実動アクティビティーを混乱させずにテストすることができます。

## 関連資料

[MFS サービス・ユーティリティー \(DFSUTSA0\) \(システム・ユーティリティー\)](#)

[MFS 言語ユーティリティー \(DFSUPAA0\) \(システム・ユーティリティー\)](#)

[MFS 装置特性テーブル・ユーティリティー \(DFSUTB00\) \(システム・ユーティリティー\)](#)

## MFS の管理

MFS が持つ柔軟なメッセージ・フォーマット設定のオプションの利点を十分に利用したり、MFS を効果的に操作するためには、MFS 管理担当者を決めてください。

### このタスクについて

MFS 管理担当者は、MFS の実施と管理に対し責任を持ち、インストール・システムのプログラミングと MFS アプリケーション設計の調整をします。

MFS 管理担当者の責任は、次のとおりです。

- アプリケーション開発要員が MFS 制御ブロックの依頼を提出する時のプロシージャの設定。
  - プロシージャを設定して IMS.TFORMAT ライブラリーに対するアプリケーションの変更を管理する。
  - 特定アプリケーションやシステム全体の要求に応じて最も効率的に MFS 制御ブロックを定義する。
  - 装置の文字伝送を最小限に抑え、MFS 制御ブロックを共用し、アプリケーションの要求事項やオペレーターの考慮事項を損なうことなく、最も効率的な MFS の使用を確実にする。
  - MFS のオペレーター用指針やシステム標準を設定し文書化する。MFS には多くのオプションがあるので、標準のプロシージャを設定してそれに従わないと、操作に混乱を招く可能性があります。端末オペレーターの訓練やエラー率を少なくするために、フォーマット設定の設計の特定の面を標準化してください。
  - どんな場合に、どのように、オプションの索引ディレクトリーを使用するかを決定し、バッファー・プール要件を決める。
  - IMS /DISPLAY と IMS モニター報告書出力の報告書で、MFS 制御ブロックと MFS バッファー・プールの使用をモニターし、必要に応じて MFS パラメーターを修正する。
  - エンド・ユーザーに、各種の装置タイプおよび端末サブシステムの操作特性に対する注意を促す。
  - 各種区分形式の違いについてユーザーに知らせる。
  - 命名規則や指針を設定し、ユーザーに知らせる。特に、MFS 管理担当者は、区画記述子ブロックの命名規則や表示画面のサイズ、ビューポート、表示文字などについて、相談にのることができるようにしておく。
  - MFS 形式上の制限や規則の情報を伝達する。
  - IMS ステージ 1 システム定義に含まれていない画面サイズと機能の組み合わせを定義する。
  - MFSDCT ユーティリティー (DFSUTB00) による処理のための MFS 装置特性テーブル 制御ステートメントを作成する。MFS 装置特性テーブルの項目およびデフォルトの形式制御ブロックは ETO 端末用で使用されます。
  - 入力メッセージのフィールド編集ルーチンやセグメント編集ルーチンを定義する。MFS および MFS がサポートするすべての装置は、メッセージ編集ルーチンを使用することができます。数値の妥当性検査やブランクをゼロに変換するなどの共通編集機能にこの出口ルーチンを使用することができます。
- IMS には、フィールド編集ルーチンとセグメント編集ルーチンのサンプルがあります。
- 3270 装置および SLU2 装置によって返された保護フィールドを MFS が検証するかどうかの判別。MFS は、装置から返された保護フィールドの内容が、装置へ送信されたフィールドの内容と異なることを検出した場合、返されたフィールドを無視します。MFS 管理者は、IMS PROCLIB データ・セットの DFSDCxxx メンバーを構成して、MFS が保護データ・フィールドを検証するかどうかを指定します。

MFS 管理者は、MFS 関連の IMS の局面すべてについて技術的な能力を有する必要があります。

- オンライン・トランザクション処理
- メッセージ処理のための IMS API
- リモート・コントローラーによる操作
- MFS の実行、装置特性、および能力
- MFS 統計および関連する IMS モニターの報告機能の出力の解釈

管理者はまた、ハードウェアおよび SLUP 用のリモート・プログラム、金融機関リモート・プログラムや ISC サブシステムについても、このようなプログラムが分散表示管理を使用して MFS で稼働する場合には、よく知っておく必要があります。

さらに、管理責任の 1 つは、装置文字伝送を最小化することです。管理者は、端末のハードウェアの特性についてもよく知っている必要があります。

MFS 管理担当者は、プログラマブル・ワークステーション開発者やエンド・ユーザーと同様、IMS システム管理担当者やアプリケーション開発者とも連絡をとる必要があります。管理担当者は、インストール標準の実施を促し、全体のシステム・パフォーマンスに必要な場合には、MFS 制御ブロックのアプリケーションの仕様を修正できなければなりません。関連プログラミング・グループのプロシージャで、MFS 管理担当者のこの権限について、確認しておく必要があります。

### 関連概念

[入力メッセージのフィールド編集ルーチンとセグメント編集ルーチン \(アプリケーション・プログラミング API\)](#)

## MFS 使用の利点

MFS 使用には、端末向けアプリケーション・システムの開発や保守の単純化と、オンライン・パフォーマンスの向上という 2 つの主要な利点があります。

IMS アプリケーションの開発や保守を単純化するため、MFS はアプリケーションにとって共通の多くの機能を実行し、アプリケーション・プログラムに、特定の装置やリモート・プログラムから独立性を持たせます。

MFS のこの装置独立性のため、1 つのアプリケーション・プログラムで、それぞれの異なった機能を生かしながら複数の装置タイプとの間でデータを処理できます。このようにして、MFS は、端末タイプを新しく追加した場合でも、アプリケーション・プログラムの変更なしにあるいは、変更を最小限に抑えることができます。

MFS を使用すると、アプリケーション・プログラムは、メッセージの読み取りと作成の方法を変更しなくとも、別のタイプの端末と通信することができるようになります。アプリケーション・プログラムが端末からメッセージを受信する場合、プログラムの入出力域でどのように現れるかは、送信した端末の種類に関係ありません。メッセージの外観は、プログラムに指定された MFS オプションによります。そのアプリケーション・プログラムが次のメッセージを違うタイプの端末から受け取る場合、ユーザーはアプリケーション・プログラムに対してなにもする必要がありません。MFS は、メッセージを送っている物理装置からアプリケーション・プログラムを切り離します。これは、データベース・プログラム連絡ブロック (PCB) が、データベース内のデータおよびデータの格納方法からプログラムを切り離すのと同じです。

他に、データの左右の位置調整、埋め込み、妥当性検査の出口ルーチン、日付と時刻のスタンプ、ページとメッセージの番号付け、およびデータの順序付けとセグメント化なども、MFS の共通機能です。MFS がこのような機能を実行する場合、アプリケーション・プログラムは、メッセージ・データの実際の処理だけを扱います。

以下の図は、IMS に表示するための装置やリモート・プログラムからの入力データや、出力装置やリモート・プログラムに表示するためのアプリケーション・プログラム・データをフォーマット設定することによって、MFS がアプリケーション・プログラムに装置独立性を持たせる方法を示しています。

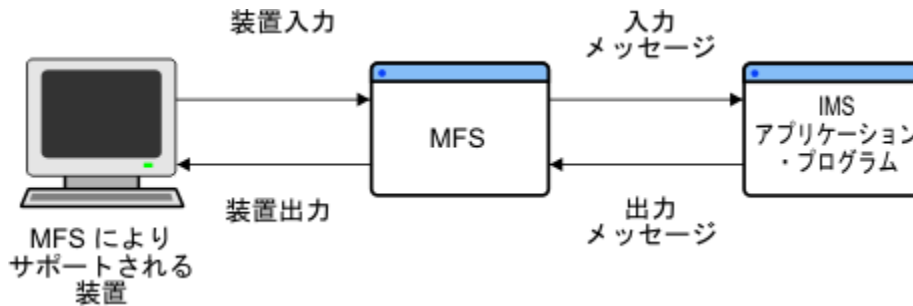


図 53. MFS を使用したメッセージのフォーマット設定

MFS はまた、オンライン処理用に設計された制御ブロックを使用して、端末向け IMS のオンライン・パフォーマンスを向上します。MFS 制御ブロックは、IMS が実行されていない時に、ソース言語による定義をオフラインでコンパイルして作成されます。MFS は、制御ブロックの妥当性を (自身で) 検査し、オフラインで判断して、オンライン処理を削減します。さらに、オンライン処理中には、MFS は、CPU 使用や入出力アクティビティーのチャンネル・コストを削減するため、MFS 制御ブロックのルックアサイド・バッファリングを使用します。

MFS の制御ブロックは再入可能で、複数のアプリケーションに使用できるため、オンライン記憶装置の必要性が少なくなります。オプションの制御ブロックの主記憶域指標付けと先行読み取りでも、応答時間を削減することができます。MFS 形式ライブラリーから形式ブロックをロードする際に複数の入出力操作を並行して実行できるので、IMS のパフォーマンスは良くなります。

さらに、MFS は、z/OS ページング・サービスを使用するので、IMS 制御領域タスクでのページ不在を削減できます。

最後に、MFS で通信回線の使用の削減もできます。必要なデータだけを圧縮し伝送することにより、回線の負荷を軽減し、応答時間と装置パフォーマンスを向上します。

#### 関連資料

[トランザクション・マネージャー 出口ルーチン \(出口ルーチン\)](#)

## MFS 制御ブロック

アプリケーション・プログラムと端末またはリモート・プログラムの入出力をフォーマット設定するために、4 つのタイプの MFS 制御ブロックを指定することができます。

#### 定義:

- メッセージ出力記述子 (MOD) は、アプリケーション・プログラムから MFS が受け取るメッセージのレイアウトを定義します。
- 装置出力形式 (DOF) には、MFS がどのようにしてプログラムが通信する装置に応じてメッセージをフォーマット設定するかを記述します。
- 装置入力形式 (DIF) には、プログラムが通信する各装置から MFS が受信するメッセージの形式を記述します。
- メッセージ入力記述子 (MID) には、アプリケーション・プログラムが処理できるよう、MFS のメッセージをどのようにフォーマット設定するかを記述します。
- メッセージ記述子は MID と MOD の両方を指します。
- 装置形式は DIF と DOF の両方を指します。

MOD、DOF、DIF、MID は、それぞれ特定のメッセージを扱うので、プログラムが送信する固有のメッセージごとに MOD と DOF が必要ですし、また逆に、受信する固有のメッセージごとに DIF と MID が必要です。

## MFS コンポーネントと操作の概説

MFS には、以下のコンポーネントが含まれています。

- MFS 言語ユーティリティー。これは、ユーザー作成の制御ステートメントから制御ブロックを生成して、それを IMS.FORMAT というライブラリーに置きます。
- MFS サービス・ユーティリティー。これは、IMS.FORMAT 内の制御ブロックの保守に使用されます。
- MFS メッセージ・エディター。これは、言語ユーティリティーで生成した制御ブロック仕様に応じてメッセージをフォーマット設定します。
- MFS プール・マネージャー。これは、主記憶域の MFS バッファ・プールの中に、メッセージ・エディターが必要とする MFS 制御ブロックを保持します。
- MFSTEST プール・マネージャー。これは、言語ユーティリティーがテスト・モードで使用されているときに MFS プール・マネージャーに取って代わります。

また、IMS オンライン変更は、MFS の一部ではありませんが、MFS ライブラリーの更新において重要な役割を果たします。簡単に言うと、IMS 制御領域が実行中も、IMS オンライン変更で制御ブロック・ライブラリーを修正することができるのです。

## 関連概念

[オンライン変更機能 \(システム管理\)](#)

## SDF II での MFS 形式の作成

SDF II は、MFS 形式を設計し、生成する対話式ツールです。

### このタスクについて

SDF II は、MFS の代わりにするのではなく、MFS 形式の開発や保守を容易にします。SDF II は、形式を設計したりテストするのに、パネル・エディターを使用するため、プログラマーは MFS ソース・ステートメントのコーディング作業の一部から解放されます。

SDF II を使用すると、アプリケーション・プログラマーおよびアナリストは、MFS の要件を特に知らなくても、以前なら特別な知識が必要であったプログラミング・ジョブの一部分を実行することができます。SDF II では、以下のスクリーン・ショットに示すようなパネル・エディターを使用して、パネルの定義やテストを行います。

```

DEFINE FORMAT
Format . . . . .Positions 1-75 or 80, Lines 1-24 of 24
Marks: V - C . L , S +           Contents: FORMAT
001
002          *****
003          ** EMPLOYEE PAYROLL **
004          *****
005
006 LAST NAME:                      FIRST NAME:
007
008 EMPL NO:
009
010 SOC SEC NO:
011
012 RATE OF PAY:
013
014
015
016 INPUT:
017
018
019
020
021
022
023
024
PF1=HELP      2=SPLIT      3=END      4=RETURN      5=RFIND      6=CHANGE
PF7=UP        8=DOWN       9=SWAP    10=LEFT      11=RIGHT     12=CURSOR

```

パネルをテストした後、SDF II のユーザーは、形式定義に必要な MFS ソース・コード (以下のサンプル・コードを参照) を自動的に生成することができます。MFS だけを使用する場合、プログラマーはこれらのステートメントを手作業でコード化する必要があります。

```

DOF
PAYF      FMT          TYPE=(3270,2),FEAT=IGNORE,DSCA=X'00A0'
          DEV
          DIV          TYPE=INOUT
          DPAGE        CURSOR=((5,15))
          DFLD         '*****',POS=(1,21)
          DFLD         '* EMPLOYEE PAYROLL *',POS=(2,21)
          DFLD         '*****',POS=(3,21)
          DFLD         'LAST NAME:',POS=(5,2)
LNAME     DFLD         POS=(5,15),LTH=16
          DFLD         'FIRST NAME:',POS=(5,36)
FNAME     DFLD         POS=(5,48),LTH=16
          DFLD         'EMPL. NO:',POS=(7,2)
EMPNO     DFLD         POS=(7,11),LTH=6
          DFLD         'SOC SEC NO:',POS=(9,2)
SSN       DFLD         POS=(9,14),LTH=11
          DFLD         'RATE OF PAY: $',POS=(11,2)
RATE      DFLD         POS=(11,16),LTH=9
          DFLD         'INPUT:',POS=(16,2)
INPUT     DFLD         POS=(16,9),LTH=30
          FMTEND

MID
PAYIN     MSG          TYPE=INPUT,SOR=(PAYF,IGNORE)
          SEG
          MFLD         'PAYUP ' SUPPLIES TRANCODE
          MFLD         LNAME,LTH=16
          MFLD         FNAME,LTH=16
          MFLD         EMPNO,LTH=6
          MFLD         SSN,LTH=11
          MFLD         RATE,LTH=9
          MFLD         INPUT,LTH=30,JUST=R,FILL=C'0'
          MSGEND

MOD
PAYDAY    MSG          TYPE=OUTPUT,SOR=(PAYF,IGNORE)
          SEG
          MFLD         LNAME,LTH=16
          MFLD         FNAME,LTH=16
          MFLD         EMPNO,LTH=6
          MFLD         SSN,LTH=11
          MFLD         RATE,LTH=9
          MFLD         INPUT,LTH=30,JUST=R,FILL=C'0'
          MSGEND

```

SDF II は、3270 ディスプレイ 上で使用するよう設計されていますが、SDF II を使って、MFS がサポートするすべての装置についての形式を作成することができます。

**関連情報:** SDF II についての詳細は、「SDF II 概要説明」を参照してください。

## 基本編集

MFS を使用しない場合は、IMS の基本編集機能でメッセージを編集します。

入力メッセージの場合、基本編集は以下のことを行います。

- システム定義 TRANSACT マクロの EDIT=UC パラメーターで指定された場合は、メッセージを大文字に変換する。
- 各メッセージの最初のセグメントから先行制御文字を除去する。メッセージが会話の続きや、事前設定モードの端末からのメッセージでない場合、先行ブランクも最初のセグメントから除去されます。
- メッセージ・タイプがトランザクションまたはコマンド (/BROADCAST コマンドを除いて) である場合には、後続のすべてのメッセージ・セグメントから先行制御文字をすべて除去する。
- すべてのセグメントから回線制御文字を除去する。
- トランザクションのセグメントから末尾キャリッジ・リターンとブロック終了文字をすべて除去する。
- 入力端末上で、バックスペースの入力や伝送が、正常な訂正プロシージャである場合に、すべてのセグメントから 1 対 1 でバックスペースを除去する。



- トランザクション・コード、論理端末、あるいはコマンド verb とそれに続く データの間に区切りが必要な場合、パスワードを除いて空白と置き換える。
- メッセージの最初のセグメントのデータの前に、/SET コマンドで事前に定義されたトランザクション・コードまたは論理端末名を挿入する。挿入するトランザクション・コードと入力したデータの間に区切りが必要な時は、トランザクション・コードに続いて空白が挿入されます。
- 端末が会話モードで、非会話型トランザクション・コードを SPA に挿入することでアプリケーションが会話を終了させた場合、非会話型トランザクション・コードを次の入力メッセージの最初のセグメントに追加する。
- VTAM サポート装置の最初の伝送のチェーンのはじめに機能管理ヘッダー (FMH) があれば除去する。
- それぞれのレコード間分離 (IRS) 制御文字のところでメッセージ・セグメントを非ブロック化し、SLU 1 カード読取装置、送信データ・セット (TDS)、またはユーザー・データ・セットから、入力のための IRS 制御文字を破棄する。
- SLU 1 コンソールに MFS 編集のオプションを選択していない場合は、それぞれの改行または用紙送り制御文字のところでメッセージ・セグメントを非ブロック化する。この文字は、上記の基準によって廃棄される場合があります。
- TRN (X'35') 文字と次のバイトにある長さを透過データとして扱う。

出力メッセージの場合、基本編集は以下のことを行います。

- 出力メッセージ中の非グラフィック文字をデータが装置に行く前に変更する。
- 改行文字、行送り文字、およびタブ文字の後に必要なアイドル文字を挿入する。
- 通信回線の操作のために、回線制御文字を追加する。

SLU-1 透過データの基本編集サポートでは、透過処理が要求された場合、基本編集は、宛先およびパスワード・フィールドの後の文字を変更したり削除することはありません。透過処理は、以下を指定することによって指示します。

- VTAM タイプ SLU-1 端末のバインド・イメージ上で BINPDSB1=BINTRNDS を指定する。
- 端末についての IMS TERMINAL マクロまたは ETO 記述子の COMPT1 パラメーターで編集オプション BASIC=TRN を指定する。

### 関連概念

399 ページの『会話型トランザクション』

会話型トランザクション処理を使用すると、会話を処理するプログラムがその会話中にストレージ内に保持されていなくても、特定の端末からのメッセージの連続性を保つことができます。

## システム間連絡機能 (ISC) の IMS 編集

IMS は、ISC (LU 6.1) (システム間連絡機能) と通信する際にメッセージごといくつかのオプションを認識します。

IMS は、以下のオプションを認識します。

- 基本編集の使用

宛先プロセス名 (DPN) は、予約 IMS 名である BASICEDT です。

- データを透過データとして扱う。

/SET コマンドで設定された IMS 事前設定宛先モードが使用されます。このオプションが使用されるのは、/SET コマンドが特定のセッションで有効であり、1 次リソース名 (PRN) が ATTACH または SCHEDULER FMH で指定されていない場合です。このオプションが有効である間は、IMS はこの入力データ・ストリーム中の入力コマンドを認識することができません。

- トランザクション・コードと入力パスワードのフィールドの次にくるデータを編集しない。

入力宛先とセキュリティー検査は、基本編集におけるものと同じです。先頭文字の / は IMS コマンドであることを示し、これにより IMS 事前設定宛先モードをバイパスまたは終了させます。入力トランザクション・コード、LTERM 名、オプションのパスワード・フィールドに続く入力メッセージに対しての追加編集はされません。これは、以下の条件のもとでは、ISC に対するデフォルトの入力編集です。

ATTACH または SCHEDULER FMH が入力に含まれていない  
ATTACH または SCHEDULER FMH が DPN を指定していない  
ATTACH または SCHEDULER FMH は、DPN を指定しているが PRN を指定していない

- 何もしない

IMS は、単一入力メッセージの処理中、入力 PRN ATTACH パラメーターをトランザクション・コードまたは LTERM 名として使用します。その後、元の IMS 事前設定値を再び使うことができます。基本編集は、入力をパスワード・セキュリティーで検査済みの、透過データとして扱います。

選択するオプションは、宛先プロセス名 (DPN) と IMS で受け取る データによって違います。DPN は、ATTACH または SCHEDULER FMH で指定されます。

ISCEDT の DPN またはその別名が指定された場合は、2 番目から 4 番目のオプションが選択されます。ISCEDT の別名は、COMM マクロの EDTNAME キーワードを使用して定義されます。ISCEDT もまた、IMS の予約名です。

#### 関連概念

469 ページの『メッセージの編集』

IMS 内では、通信インターフェースはアプリケーション・プログラムにとっては透過的です。

## 透過オプション

透過データには、プログラム式シンボル、拡張フィールド属性、およびコマンドが含まれます。

このような印刷不能な文字は、3270 の画面では、ハイフンや印刷不能文字として別の表示で現れる場合があります。

IMS が透過データを、それを受信することができない IBM 3270 端末に送ろうとすると、エラーが IMS に戻り、その端末はサービスの対象から外されます。端末を再始動するには、マスター端末のオペレーターが /START コマンドを出さなければなりません。

機能管理ヘッダー (FMH) の 1 次リソース名 (PRN) がトランザクションを含む場合は、LU 6.1 の入力は、基本編集の透過オプションを使用して処理されます。それ以外は、データ・ストリームは、トランザクション・コードとオプションのパスワードを探してスキャンされるだけです。残りのデータは、透過データとして扱われます。

#### 関連概念

出力装置の拡張フィールド属性 (アプリケーション・プログラミング API)

## 無保護画面オプション

画面が無保護状態の場合、IMS は端末からの入力がなくとも、出力をいつでも端末に送ることができます。このオプションは、端末ごとまたはメッセージごとに使用できます。

## MFS 編集のバイパス

IMS では、予約されたフォーマット名 DFS.EDT または DFS.EDTN を指定することによって、IMS アプリケーション・プログラムが IBM 3270 または SLU-2 装置宛ての出力メッセージの MFS によるフォーマット設定をバイパスすることができます。このバイパスは、IMS 下で実行するサブシステムまたはインストール・サポート・プログラム用に意図されたものです。

**推奨事項:** このバイパスを使用すると、アプリケーション・プログラムは MFS に関連した生産性、装置独立性、およびマイグレーションの利点を失うことになるため、IMS アプリケーション・プログラムにはこのバイパスを使用しないでください。

MFS を出力時にバイパスした場合、アプリケーション・プログラムは、コマンド・コードで始まり最後のデータ・バイトで終わる、3270 データ・ストリーム全体を構成します。アプリケーション・プログラムが MFS または基本編集の編集なしで、3270 の入力データ・ストリームを端末から受信することができるよう、MFS 出力編集機能をバイパスさせたい場合もあります。



## 端末キーボードのロックとアンロック

アプリケーションで MFS バイパス機能と予約形式名 DFS.EDTN.EDTN を使用する場合、IMS ではアプリケーション・プログラムで、端末キーボードをロックまたはアンロックできるようになっています。

### このタスクについて

ロック・オプションを指定した場合は、処理が完了した後のキーボードのアンロックは、プログラムで行わなければなりません。アンロック・オプションを指定すると、IMS がキーボードのロックとアンロックを制御します。

#### 関連概念

[MFS メッセージ形式 \(アプリケーション・プログラミング API\)](#)

## IMS 非グラフィック・メッセージ・データに対するセンシティブティ

以下のトピックでは、ユーザーが IMS メッセージの中で非グラフィック・データを送受信しようとする場合の、特定の文字に対する IMS センシティブティについて説明します。

### 出力メッセージ・セグメントの編集

MFS が編集する出力メッセージ・セグメントの場合、装置に送られる出力メッセージに入れられるのは、グラフィック・データ (X'40' から X'FE') だけです。出力メッセージに非グラフィック文字があれば、データが装置に送られる前に MFS が変更します。

装置制御文字 HT、CR、LF、NL、および BS は、3270 データ・ストリームの場合 X'00' に変わります。それ以外の装置タイプの場合、すべての非グラフィック文字もまた空白に変わります。

MFS の分散表示管理 (DPM) オプションが SLU-P または ISC に使用された場合、ユーザーは SEG ステートメントに GRAPHIC=NO を指定することができます。非グラフィック文字は、GRAPHIC=NO を指定された出力セグメントにある場合は、リモート・プログラムにはそのまま渡されます。

VTAM を介してサポートされるプログラマブル・ワークステーションの場合、IMS は、単一の IMS セグメントを複数の伝送に分割する際に、装置制御シーケンスに機能管理ヘッダー (FMH) を挿入したり、追加編集を実行したりすることができます。

### MFS による入力メッセージ・セグメントの編集

MFS が装置に定義されている場合は、特定の考慮事項に注意してください。

考慮事項には以下のものがあります。

- システム定義 TRANSACT マクロの EDIT=UC を使用して宛先で大文字変換を要求する場合は、セグメントでの大文字変換を避けるため、SEG ステートメントで GRAPHIC=NO を指定してください。
- 最初の入力レコードが 3600、SCS1、または SCS2 の装置から、あるいは DPM-An から送られたもので、セグメントの最後の文字が以下のいずれかである場合には、そのセグメントは廃棄されます。

2 つのアスタリスク (\*\*)

2 つのアスタリスクの後に NL が付いたもの (\*\*X'15')

2 つのアスタリスクの後に IRS が付いたもの (\*\*X'1E')

- メッセージ・セグメントの最初にある 2 つのスラッシュ (//) は、エスケープ・シーケンスと見なされません。
- カード機能が SCS1 装置 (DEV ステートメントの CARD= オペランド) に指定されている場合、データが入力 MFLD に表示される前に、制御文字は磁気カード入力からすべて除去されます。

MFS サポートのある装置 (IBM 3270 を除く) に対する MFS 削除文字 (DEV ステートメントの LDEL= オペランド) およびフィールド・タブ文字 (DEV ステートメントの FTAB= オペランド) の定義によって、入力メッセージ・セグメントの編集を指示することができます。

入力を MFS で処理する場合、実行される編集は、メッセージ形式サービス言語ユーティリティーを介して提供される記述によって異なります。MFS 編集後は、入力メッセージ・セグメントと装置からの入力メッセージは 何の関係もなくなるので、装置からの入力セグメントは、ユーザー作成編集ルーチンで使用できなくなります。MFS 編集後の入力メッセージ・セグメントがユーザー作成編集ルーチンで使用できます。

## 関連概念

[入力メッセージ・フォーマット設定 \(アプリケーション・プログラミング API\)](#)

## 基本編集による入力メッセージ・セグメントの編集

基本編集を非グラフィック・メッセージ・データに使用する場合、以下の編集が行われます。

- 端末が会話モードでない場合、入力メッセージの最初のセグメントについては、X'41' より小さい先行文字が除去されます。最初のセグメント以外か、端末が会話モードである場合は、X'40' より小さい先行文字が除去されます。
- 端末が会話モードあるいは事前設定モードでない場合、最初のセグメントの 9 桁以内に左括弧がある場合は、パスワードがあることを示しています。左右の括弧とパスワードは除去され、セグメントは圧縮されます。
- 非 SNA 装置では、セグメントの最後の文字として現れる文字 X'26' が除去されます。
- セグメントの最後の文字として、アスタリスク 2 つ (\*\*) または アスタリスク 2 つの後に NL (X'15') がある場合、セグメント全体が廃棄されます。
- バッファなしのキーボード装置では、バックスペース (X'16') 文字を、文字削除標識として扱います。バックスペース文字自身と先に入力した文字は入力セグメントから除去されます。
- 入力メッセージの宛先が、トランザクションである場合、セグメントの最後にある NL (X'15') 文字は除去されます。
- 装置が事前設定モードである場合は、トランザクション・コードが最初のセグメントに追加されます。
- IBM 3270 装置からの入力の場合、アテンション ID とカーソル・アドレスは除去され、開始バッファ・アドレス・シーケンスはすべてブランクに変わります。
- セグメントの先頭文字がスラッシュ (/) の場合は、入力メッセージ全体がコマンドとして扱われます。
- 入力メッセージが NDS 装置から受信したものか、または、システム間連絡機能 (ISC) を使用している場合、データ・ストリームは、トランザクション・コードとパスワードの編集の場合を除いて、透過データまたは透過として扱われます。

## 出力装置の制御

メッセージ接頭語 (Z2) の Z2 フィールドの 2 バイト目に制御文字を使用することによって、出力装置を制御することができます。

### このタスクについて

以下の規則に従ってください。

#### • SLU-1:

セグメントに構造化フィールド・データを含む場合は、Z2 フィールドにビット X'80' を設定する必要があります。

セグメントが新規 LPAGE シリーズの最初のセグメントである場合は、Z2 フィールドにビット X'40' を設定する必要があります。

#### • 交換回線装置:

##### **制約事項:** IBM 3270

メッセージ・フォーマット設定の Z2 フィールドを使用して、現行メッセージの送信後の回線の切断を要求できます。出力が物理的にこの機能がない装置に送られる場合、このフィールドは無視されます。切断要求は、X'80' を Z2 フィールド値として使用することによって指示されます。この要求は、どのセグメントに存在する場合でも認識されます。

- IBM 3270 プリンター (3270P 装置タイプ)

Z2 フィールド、ビット値 X'80' を使用して、MFS 編集をバイパスする場合に使用する、データ・ストリーム内のコマンド・コードおよび WCC 文字の存在を指定することができます。

- プリンター・コンポーネント

プログラムのメッセージのテキスト部分に復帰文字や改行記号を組み込むことができます。出力がローカル・プリンター (SYSOUT) に行く場合は、メッセージの最初の 2 文字は、キャリア制御文字にすることができます。

## 小バッファ装置

端末装置によっては、処理できる最大バッファ・サイズに、ハードウェア上の制限があります。さらにこの制限が、装置かネットワークのどちらかのソフトウェアにもかかる可能性があります。

この制限を超えると、エラーになり、また処理する装置に対してメッセージが大きすぎるため送ることができなくなります。この制限がどのくらいか、また、アプリケーション・プログラムでとれる代替方法は何かなどに、注意しておく必要があります。

装置タイプにより制限の指定の仕方が異なります。入力の場合、IMS の SEGSIZE または BUFSIZE 指定、出力は、OUTBUF 指定を使用します。VTAM 端末の場合、BIND RU サイズはこれらの値に基づいています。IMS は、入力と出力のどちらについてもメッセージ・チェーンをサポートしますが、端末によってはチェーンがサポートされない場合もあります。この場合、メッセージ長はサイズで制限されます。

アプリケーション・プログラムは、出力メッセージの長さを直接、またはメッセージ・フォーマット設定に使用する MFS 定義を介して制御することができます。ターゲット装置でサポートされる最大長を超えるメッセージを送る必要がある場合は、メッセージを複数のメッセージに分割して、各メッセージが送信先の最大長より短くなるようにしてください。これを行うためには、アプリケーション・プログラムから DL/IPURG 呼び出しを出さなければなりません。IMS は、アプリケーション・プログラムが挿入した順序に従ってメッセージを伝送します。これは、メッセージ境界を感知しない装置にとってよい解決方法です。

### 関連資料

[PURG 呼び出し \(アプリケーション・プログラミング API\)](#)

[495 ページの『TERMINAL マクロ』](#)

TERMINAL マクロのいくつかのシステム定義キーワード・パラメーターは、ISC セッションを定義するために重要です。

[LINE マクロ \(システム定義\)](#)

## 出力制御

IMS オンライン・システムの出力の大部分は、入力端末に送られます。プリンター出力の作成には、いくつかの選択があります。

### このタスクについて

- ハードウェア対応の装置サポートを介してのローカル印刷の使用。
- 応答を入力端末のプリンター・コンポーネントに出すアプリケーション・プログラムの作成。
- プリンター・コンポーネントに応答 LTERM を割り当てるための **/ASSIGN** コマンドの使用。
- 出力 (SYSOUT) を IMS システム・プリンターとしての LTERM に送るアプリケーション・プログラムの作成。
- オフライン印刷の代替装置である LTERM に出力を送るアプリケーション・プログラムの作成。

アプリケーション設計に応じて選択をしてください。**/ASSIGN** コマンドの使用とスプール・データ・セットの制御を選択する場合は、マスター端末のオペレーターの介入が必要となる場合があります。

ETO を使用すると、出力はユーザーと関連付けられます。ユーザーは端末から端末へと移動することができます。出力はユーザーの後について行きます。出力が使用可能になると、自動ログオン機能を使用して、指定した端末に出力を送るセッションを生成することができます。

## プリンター・コンポーネントの使用

端末のプリンター・コンポーネントが、いつも連続稼働状態にあるわけではありません。時には、プリンターに、リモート・オペレーターによる用紙補給または特殊用紙の監視が必要になることがあります。

### このタスクについて

リモート・オペレーターがVTAMのコンポーネントである場合は、**/COMPT** コマンドを使用して、装置を作動可能または作動不能にすることができます。オペレーターは **/ASSIGN** コマンドを使用して、出力を特定のコンポーネントに送ることができます。

#### 関連資料

[/COMPT コマンド \(コマンド\)](#)

[/ASSIGN コマンド \(コマンド\)](#)

## スプール出力制御

IMSのスプール出力の要件には注意を払う必要があります。それは、エンド・ユーザーが出力を即時に必要としたり、割り振りスペースを使いきってしまった場合、それ以降の出力が妨げられる可能性があるからです。

スプール回線グループの出力に使用するデータ・セットの数に注意を払ってください。お勧めする定義は少なくとも2つで、1つは印刷用のデータ・セットで、もう1つは次の入力を受信するためのものです。回線番号および PTERM 参照も必要です。

**/STOP LINE n PTERM nn** コマンドを使用すれば、いつでもスプール出力をプリンターにスケジュールすることができます。

このコマンドは、データ・セットをクローズして、それ以降の出力を次のスプール・データ・セットに送ります。これがセットの最後である場合は、出力は最初のデータ・セットに送られます。印刷ユーティリティがスケジュールされた時に、出力データ・セットが1つしかない時は、LTERMのメッセージはすべてキューに並びます。これでメッセージ・キューの内容に即座に追加することができます。メッセージ・キューは、スプールが2次マスター・コンソールである場合は特に、すぐに満杯になる可能性があります。データ・セットが満杯になると、印刷ユーティリティを実行する IMSWTnnn プロシージャが自動的にスケジュールされます。また、データ・セットの書き込みエラーでもプロシージャが開始されます。

印刷プロシージャはシステム定義により、回線グループ・データ・セットに合わせて調整されます。このプロシージャは、**/START REGION IMSWTnnn** コマンドを使用していつでも呼び出すことができます。データ・セットがクローズしていない場合、現行の回線出力はキューに入れられます。

再始動すると、スプール出力の取り扱いに影響します。出力データ・セットが前の実行で印刷されなかった場合、**/NRESTART** コマンドはそれ以降の出力メッセージを出さないようにします。**/START LINE** を入れると、自動的に IMSWTnnn プロシージャをスケジュールし、現行のシステム実行から出力のデータ・セットを解放します。

## IBM 3270 情報表示システムのプリンター・コンポーネントの使用

プリンター・コンポーネントを IBM 3270 情報表示システムの一部として、定義することができます。これで、ビデオ出力(または入力)の印刷コピーを、プリンターのコンポーネントに送ることができます。

サポートされる装置は、IBM 3284、3286、3287、3288、3289 のプリンター・コンポーネントです。

システム定義入力で、接続方式をポーリング方式の BSC にするか SDLC 回線を通すかを指示します。印刷コピーは自動的に作成することも、オペレーター制御にすることもできます。入力メッセージの定義または出力メッセージの制御接頭部でこの自動印刷をすることができます。プリンター・コンポーネントは、コピーする情報を持つディスプレイ・ワークステーションと同じ 3270 制御装置に接続する必要があります。ディスプレイ・ワークステーションは、IBM 3271/3274 または 3275/3276 です。

**制約事項:** ローカル接続 3270 は、コピー機能をサポートしません。



## 候補プリンターの指定

BSC 3270 で、画面内容があるプリンターへコピーする 要求は、VTAM を使用して画面と同じ制御装置上にある最初に使用可能なプリンター に向けられます。

**定義:** 最初に使用可能なプリンター とは、その装置に対する TERMINAL マクロの順序付けでのことです。事前に決められた順番によって、メッセージ出力のために選択されるプリンター・コンポーネントの順序が制御されます。以下の図は、2つのディスプレイ・ワークステーションのプリンター・グループを示しています。

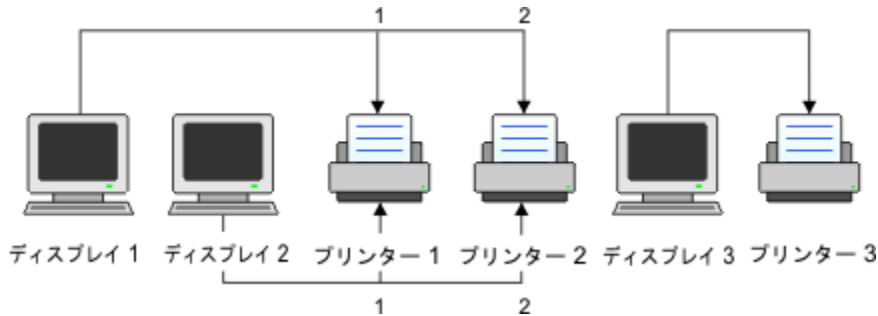


図 54. IMS TERMINAL マクロの順序

ディスプレイの後に指定される最初のプリンターがそのディスプレイに対するプリンターの候補となります。また、最初のプリンター候補との間になにもディスプレイの定義がなければ、それに続く指定プリンターもプリンター候補となります。上記の図では、ディスプレイ 1 またはディスプレイ 2 の印刷コピーが要求された場合、IMS は 3270 コピー・コマンドをプリンター 3 (使用可能な場合) に送信します。プリンター 3 が利用可能でない場合、IMS はプリンター 4 を考慮します。プリンター 6 は、ディスプレイ 1 またはディスプレイ 2 の候補ではなく、ディスプレイ 5 の候補となります。

コピー機能は、3270 制御装置の境界を渡って使用することはできません。IBM 3274 と 3276 に対しては、SNA プロトコルで、IMS が印刷要求をディスプレイに送り、そのコントローラーがプリンターを選択します。IMS にプリンターを定義する必要はありません。

**制約事項:** ETO 端末については、3270R (非 SNA) コピー機能はサポートされません。

## 操作上の考慮事項

オペレーターやアプリケーション・プログラムがコピー操作の要求を送信すると、最初に使用可能な候補プリンターが出力に使用されます。

検索する順序は、システム定義の順序に従います。プリンターが停止していたり、すでに他のメッセージを印刷している、専有的状態にある、または作動不能などの場合は、次のプリンター候補を選びます。オペレーターから印刷要求があり、プリンターがすべて使用中の場合は、プリンターが使用可能になるまでキーボードはロック状態になります。使用しようとするプリンターが使用不可の場合は、コピー機能の出力メッセージは送られません。表示ワークステーションは、エラー・メッセージを受け取ります。メッセージ先送り機能 (PA2 キー) を使用して、画面からエラー・メッセージをクリアすると、再試行できます。メッセージのフォーマット設定記述で、コピー・アクションの指定もできます。MFS では、DEV ステートメントの DSCA オペランドを使用して、コピー・アクションを指定します。

### 関連概念

[システム制御域 \(SCA\) とデフォルト SCA \(DSCA\) \(アプリケーション・プログラミング API\)](#)

### 関連資料

[DEV ステートメント \(システム・ユーティリティー\)](#)

## システム間でのプリンターの共用

マスター以外のプリンターを共用して使用する場合は、IMS では、オンライン IMS システムと他のサブシステムと (他の IMS も可) が交互にプリンターを使用できるようになっています。

この IMS サポートは VTAM 3270 プリンター用で、SNA 環境の場合、LU-1 および LU-4 プリンター用です。/OPNDST NODE Q コマンドで要求プリンターに対する IMS へのログオンをシミュレートします。プ

リナーは、所有しているサブシステムがプリンターを解放すると、自動的に使用可能になります。出力は、プリンターの可用性に関係なく、いつでもキューに入れることができます。

システム定義要件は、TERMINAL マクロの OPTIONS キーワードで指定します。下記を指定してください。

#### **OPNDST**

**/OPNDST** コマンドがこの端末に有効になります。

#### **SHARE**

データが出力のためにキューに入れられると、IMS は VTAM マクロの SIMLOGON を出して、プリンターを獲得します。

#### **RELRO**

要求があれば、IMS は他の VTAM サブシステムに端末を解放します。

ETO 端末と同一機能を使用可能にするには、ETO ログオン記述子で RELRO を指定してください。

共用プリンターのキューに入れられた各出力メッセージが、セッションを設定しようとします。

即時ログオン・シミュレーションの代替として、およびメッセージ獲得に繰り返し失敗した場合には、共用プリンター出口ルーチン (DFSSIMLO) を使用することができます。この出口ルーチンは、端末 (またはトランザクション) 状況をモニターし、以下の 3 つの処置のうちの 1 つを実行します。

- 要求を無視する
- **/OPNDST** コマンドをシミュレートする
- 自動化操作プログラムのためにトランザクションをキューに入れる

プリンターが接続されていれば、共用プリンター出口ルーチンは呼び出されません。共用プリンター出口ルーチンを呼び出すには、IMS 始動パラメーターで SIMEXIT キーワードを指定します。DFSSIMLO は、IMS.SDFSRESL の前に連結されている JOBLIB、STEPLIB、または LINKLIST ライブラリー内の許可ライブラリーに入れる必要があります。

**推奨事項:** 2 次マスター端末を共用しないでください。

#### **関連概念**

401 ページの『[論理端末ネットワークの設計](#)』

IMS システム定義は、通信回線、静的端末、および論理端末 (LTERM) の特性と関係を記述します。

#### **関連資料**

[共用プリンター出口ルーチン \(DFSSIMLO\) \(出口ルーチン\)](#)

---

## 第 7 部 システム間連絡 (ISC)

これらのトピックでは、システム間連絡 (ISC) を紹介し、ISC を使用して IMS サブシステムと ISC プロトコルをサポートする他のタイプのサブシステムを接続する際に必要なすべての情報を提供します。





## 第 26 章 システム間連絡機能の概要

ISC は、IMS Transaction Manager の一部です。これは、複数のサブシステムを接続する 1 つの方法です。別の接続方法としては複数システム結合機能 (MSC) があります。

SNA の下で定義されているように、ISC は下記を行う LU 6.1 セッションです。

- 別のサブシステムと接続し、アプリケーション・レベルで通信する。
- あるサブシステムの端末ユーザーまたはアプリケーションが別のサブシステムの端末またはアプリケーションと通信を行い、オプションとして応答を受信できるようにする、分散トランザクション処理を提供する。アプリケーションはユーザー作成のものである場合もあれば、サブシステム自体がアプリケーションとして働く場合もあります。
- 分散サービスを提供する。したがって、1 つのサブシステムの中のアプリケーションは、別のサブシステムの中のサービス (例えば、メッセージ・キューやデータベースなど) を使用することができます。

ISC の IMS 実装環境は、TCP/IP (ISC TCP/IP) または SNA VTAM (ISC VTAM) によって管理されている接続をサポートします。動的に定義された ISC ノードと静的に定義された ISC LU 6.1 端末は両方とも、TCP/IP または VTAM のサポートを使用することができます。

ISC TCP/IP は、IMS と CICS 間の接続のみをサポートします。ISC TCP/IP サポートは、CICS によって定義されている専用プロトコルである IP 相互接続 (IPIC) を使用します。IPIC の使用は、通常、ISC VTAM に対して SNA によって定義されているプロトコルと整合しており、ISC を使用するアプリケーション・プログラムからは認識されません。

ISC VTAM は、異種のサブシステム間の通信をサポートし、SNA 定義のセッション制御プロトコル、データ・フロー制御プロトコル、および経路指定パラメーターを含みます。これらの各プロトコルおよびパラメーターが提供する機能については、以下に要約してあります。また、それらに対する IMS サポートについても、このトピックと各サブトピックで説明します。

セッション制御 (SC) は、以下の事項から成ります。

- サブシステム間のセッションの開始
- セッションの回復と再同期、セッション状態の保全およびセッションとサブシステム両方の障害に及ぶリカバリー可能リソースの保守
- IMS と別サブシステム間のいずれか、または、すべてのセッション・パスの終了

データ・フロー制御 (DFC) には、以下の事項が含まれます。

- セッション内でのプロトコルの送信と受信を制御する。
- セッション内でのトランザクション開始の競合を解決する。
- エラー・リカバリー処理をモニターする。
- 対称シャットダウンのモニター
- リソース同期化の制御。同期点制御は、すべてのリソースが同期的にコミットまたはバックアウトされるようにします。

ISC VTAM 経路指定は、次の事項から成ります。

- SNA が定義した機能管理ヘッダーのパラメーターを使用して、着信メッセージに必要なプロセスに接続し、そして応答メッセージを転送します。

APPC/IMS は、DFSAPPC サービスを通して LU 6.2 宛先へメッセージ通信することができます。

### 関連概念

54 ページの『DFSAPPC システム・サービス』

DFSAPPC は IMS システム・サービスであり、LU 6.2 アプリケーション・プログラム間 (LU 6.2 から LU 6.2)、および LU 6.2 アプリケーション・プログラムと IMS が管理する LTERM 間のメッセージ通信を行います。メッセージ送達是非同期であり、メッセージは送達されるまで IMS メッセージ・キューに保留されます。

110 ページの『ETO と LU 6.1 (ISC) 端末』

LU 6.1 (ISC) 端末の場合、IMS は同じノード名への並列セッションをサポートします。この場合、別の構造がセッションごとに構築されます。ただし、各セッションとその関連構造は、別個の端末のように独立して機能します。

#### 関連タスク

499 ページの『VTAM 接続用の ISC プロトコル』

IMS では ISC プロトコルを使用して、ISC VTAM 接続経由のセッション、データ・フロー、およびメッセージ経路指定を制御します。以下のトピックには、ISC リンクでのデータの送受信に必要な特定のプロトコル情報が含まれています。

## ISC と MSC の比較

複数システム結合機能 (MSC) とシステム間連絡機能 (ISC) は、どちらも複数の IMS サブシステムを結合するために使用できます。MSC と ISC はどちらも、トランザクションの経路指定、トランザクション処理の分散、および 1 つの IMS システムの容量を超える拡張を可能にします。

MSC は、IMS システム間結合を他の IMS システムだけに許可する IMS プロトコルです。しかし、ISC を使用すれば、IMS サブシステムを、ISC プロトコルをサポートする他のサブシステムと接続することができます。ここで言う他のシステムとは、別の IMS、CICS、またはユーザー作成のシステム のことです。

MSC は、IMS システム間の 4 つのタイプのリンクをサポートします。すなわち、チャンネル間 (CTC)、メモリー間 (MTM)、TCP/IP、および VTAM LU 6.1 です。ISC は VTAM LU 6.1 または TCP/IP を使用しますが、ISC TCP/IP は、IMS と IBM CICS Transaction Server for z/OS の間の接続にのみサポートされます。

次の図は、MSC と ISC を比較したものです。

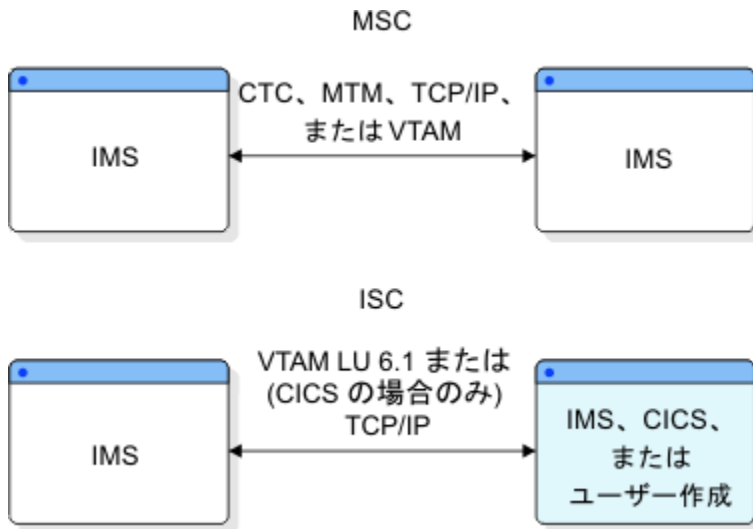


図 55. IMS 複数システム結合とシステム間連絡機能

ISC、MSC VTAM、および MSC TCP/IP はすべて、並列セッション・サポートを提供します。ただし、いくつかのキーに差があります。次の表では MSC と ISC の主要機能を中心に説明し、そのサポートの違いを示します。

表 76. MSC と ISC 機能の比較

MSC 機能	ISC 機能
MSC は、複数の IMS システムだけを相互に接続します。これらの IMS システムは、すべてが 1 つのプロセッサに常駐することも、あるいは別のプロセッサに常駐することもできます。	接続されたサブシステムの両方が ISC を導入していない限り、類似または非類似サブシステムに接続することができます。ISC は IMS サブシステムを以下のものに結合することができます。 <ul style="list-style-type: none"> <li>• 別の IMS サブシステム</li> <li>• CICS サブシステム (VTAM または TCP/IP サポートを使用)</li> <li>• 1 つのユーザー作成サブシステム</li> </ul>
MSC 環境における通信は、サブシステム間で行われます。	通信は、2 つのサブシステムの中のアプリケーション・プログラム間で行われます。サブシステム自体は、セッション・パートナーで、アプリケーション間の論理フローをサポートします。
処理はユーザーには透過的です。言い換えれば、ユーザーには、MSC 処理が、単一システムで起きているように見えます。	ISC は非類似サブシステムの結合を許可するために、メッセージ経路指定では、端末ユーザーまたはアプリケーションによってメッセージの宛先を判別することが必要になります。ISC VTAM 接続に対して指定されている経路指定パラメーターは、メッセージ形式サービス (MFS) によってオーバーライド、変更、または削除することができます。
MSC 直接経路指定が使用されない限り、経路指定はシステム定義パラメーターに基づいて自動的に行われます。端末オペレーターまたはアプリケーション・プログラムは、経路指定情報を認識している必要はありません。	ISC は、独自のメッセージ通信機能を備えているため、ユーザー・アプリケーションと かかわり合いを持たずにメッセージ経路指定を許可します。
MSC は、ソース端末オペレーターおよび各会話型ステップ (アプリケーション) の両方に対しては透過的に、会話のステップが複数の IMS サブシステムに分散されるのをサポートします。	VTAM によってサポートされている ISC 接続は、IMS サブシステムで MFS をサポートし、サブシステム間のメッセージの経路指定とフォーマット設定を支援します。ISC TCP/IP 接続は、MFS をサポートしていません。
MSC は、高速機能急送メッセージ・ハンドラー (EMH) の使用をサポートしません。	VTAM 接続が使用された場合、ISC は、IMS サブシステム間の高速機能急送メッセージ・ハンドラー (EMH) の使用をサポートします。

### 関連タスク

701 ページの『複数システム結合機能の管理』

以下のトピックでは、MSC を使用してネットワーク内で複数の IMS オンライン・システムを接続する際に必要なシステム管理アクティビティについて説明します。

## ISC に使用可能な IMS 機能

システム間連絡 (ISC) ではさまざまな IMS 機能を使用できます。

### 分散トランザクション処理

分散トランザクション処理をサポートするとき、IMS はフロントエンドまたはバックエンドのいずれかのプロセッサになることができます。フロントエンド・サブシステムは、バックエンド・サブシステムで処理される通信トラフィック (トランザクション、コマンド、またはメッセージ通信) を発信します。メッセージは、一般的に端末で入力されるデータの結果です。ただし、メッセージは、フロントエンド・サブ

システム内のアプリケーションで生じた処理の結果である場合もあります。バックエンド・サブシステムは、別のサブシステムからの入力メッセージを処理します。

最も一般的な構成では、IMS は、バックエンド・システムであり、別のサブシステムにより入力される IMS トランザクションを処理します。

さらに、フロントエンド切り替え出口ルーチンは、ISC VTAM が使用された場合、フロントエンドまたはバックエンドのシステムを利用するための特別なサポートを提供します。ISC TCP/IP では、フロントエンド切り替え出口ルーチンはサポートされません。

IMS フロントエンドに入力されるトランザクションは、ISC メッセージ通信で別のサブシステムに送信されることがあります。この場合には、端末オペレーターまたは MFS がメッセージ経路指定情報を提供します。トランザクションは、ユーザー作成のアプリケーション・プログラムまたは MFS 形式定義が経路指定情報を提供した代替プログラム連絡ブロック (PCB) 出力として送信することもできます。

IMS が接続待ちサブシステムとなる場合や、IMS がフロントエンド・プロセッサとして作動する場合には、特に、IMS が提供するどの ISC 機能を使用するかを決めることが重要となります。フロントエンド IMS の端末オペレーターまたはアプリケーションによって作成されたメッセージは、受信側サブシステムへの伝送のキューに入れられ、そのメッセージを送信した端末またはアプリケーションとは非同期式に送信されます。IMS フロントエンドに接続された端末は、ISC VTAM およびフロントエンド切り替え出口ルーチン (DFSFEJ0) の両方を使用しない限り、受信 (バックエンド) サブシステムの処理中は応答モードで保留されません。

## 分散サービス

IMS は IMS メッセージ・キューにリモート・アクセスすることにより分散サービスを提供します。IMS は分散 DL/I 呼び出しをサポートしませんが、ある 1 つのサブシステムの DL/I データベースを別のサブシステムで更新することはできます。このアクションには、受信側サブシステムの更新アプリケーションを呼び出す送信側サブシステムが必要です。

## IMS トランザクション・タイプ

IMS は、以下のトランザクション・タイプをサポートします。

- リカバリー可能な更新 (高速機能を含む)
- リカバリー可能な照会
- リカバリー不能な照会

ISC を TCP/IP で使用している場合、前のリストに記載されているトランザクション・タイプに対する ISC サポートにいくつかの制限が適用される可能性があります。詳細については、[603 ページの『ISC TCP/IP セッションで使用可能な機能』](#)を参照してください。

## IMS 実行モード

IMS は、以下の実行モードをサポートします。

- 応答モード (高速機能を含む)
- 会話型モード
- 排他モード
- トランザクション事前設定モード
- テスト・モード
- 非応答または非会話モード

ISC を TCP/IP で使用している場合、非応答モードまたは非会話モードのみがサポートされます。詳細については、[603 ページの『ISC TCP/IP セッションで使用可能な機能』](#)を参照してください。

## IMS 編集機能

ISC セッションで伝送され、かつ IMS サブシステム内で処理されるメッセージは、下記の編集機能により編集されます。

ISC 編集 (デフォルト・エディター)  
メッセージ形式サービス (MFS)  
基本編集

これらのエディターは、メッセージごとに選択できます。

## ISC メッセージの保水性

メッセージの保水性は、セッション再始動中の IMS と別のハーフセッション間の入出力メッセージの消失または重複を予防するためにあります。

メッセージの保水性とリカバリーは、ログ先行書き込み (LWA) で強化されます。この機能は、システム定義中に TRANSACT マクロのオプションによって呼び出されます。LWA は、IMS がメッセージを確認する前に、同期点情報をログに書き込む (その結果、IMS 再始動プロシージャに使用できる) ようにします。メッセージの保水性について詳しくは、『セッションの再同期』および『同期点および応答要件』のトピックを参照してください。

## ISC セキュリティー

IMS セキュリティー機能は、別のサブシステムによる IMS リソースへのアクセスを制御します。IMS で ISC セキュリティーを実施する前に、サブシステムおよび関連するオペレーティング・システムのコンポーネントのデータ保護機能を調べてください。ここでは、重要な ISC 環境のセキュリティ要素について説明します。

端末オペレーターの検証と許可は、その端末接続を制御するサブシステムによって行われます。

IMS 端末とパスワード・セキュリティはともに、リソース・アクセス制御機能 (RACF) を使用して、静的 ISC 端末に対して定義することができます。パスワード・セキュリティが定義されている場合は、入力メッセージにパスワードを規定することが必要です。

## XRF 複合システムにおける ISC

ISC VTAM セッションは、クラス 2 または 3 の端末として扱われます。アクティブ IMS システムに障害が発生すると、すべての LU 6.1 セッションは終了します。代替 (現在アクティブ) IMS システムは、複合システムの制御を引き継いだ後、そのセッションを再設定します。ユーザーは、短時間の間、サービスを受けられない場合があります。

ISC TCP/IP セッションは、XRF ではサポートされません。

### 関連概念

[501 ページの『XRF 複合システムとの接続の確立』](#)

XRF 複合システムを、セッション・パートナーにすることができます。

[412 ページの『セキュリティ設計』](#)

IMS ネットワーク内の端末の無許可使用を防止するために、RACF (または同等の製品) を使用できます。

[110 ページの『ETO と LU 6.1 \(ISC\) 端末』](#)

LU 6.1 (ISC) 端末の場合、IMS は同じノード名への並列セッションをサポートします。この場合、別の構造がセッションごとに構築されます。ただし、各セッションとその関連構造は、別個の端末のように独立して機能します。

[504 ページの『セッションの再同期』](#)

サブシステム障害およびセッション障害において、IMS 内のリカバリー可能リソース、メッセージおよびキューの保水性を保守するには、両方のハーフセッションで再同期プロセスに必要なセッション情報を保守しなければなりません。

[471 ページの『ISC と IMS の実行モードの関係』](#)

「同期」と「非同期」という用語は、IMS と ISC では少し意味が異なります。以下のトピックでは、これらの実行モードの関係について説明します。

[拡張回復機能の概要 \(システム管理\)](#)

[IMS セキュリティー \(システム管理\)](#)



## 関連タスク

435 ページの『IMS メッセージの編集とフォーマット設定』

IMS、メッセージ形式サービス (MFS) および基本編集ルーチンの 2 つの方法によって、端末でやり取りするメッセージの編集およびフォーマット設定を行います。

432 ページの『IMS トランザクション・タイプおよびトランザクション状態』

トランザクションは、論理装置から IMS へ送られる最も一般的なデータ・タイプです。

## 関連資料

522 ページの『同期点および応答要件』

IMS 入出力メッセージ・フローを順次キュー・データ・セットからの入出力のフローとして示すことができます。

フロントエンド切り替え出口ルーチン (DFSFEBJ0) (出口ルーチン)

## サンプルのシステム構成

これらの図は、サンプルの IMS システム間連絡 (ISC) 構成を示しています。最初の 2 つの図では、ISC ノードの例として IBM CICS Transaction Server for z/OS を使用しています。

### CICS からの呼び出される既存または新規の IMS トランザクション

次の図は、ISC の分散トランザクション処理機能、すなわち既存または新規の IMS トランザクションを CICS アプリケーションから、または (トランザクション経路指定を使用して) CICS に接続された端末に代わって CICS アプリケーションから呼び出す機能を示しています。この図の中のリンクは、ISC TCP/IP リンクまたは ISC VTAM リンクのいずれかの場合があります。

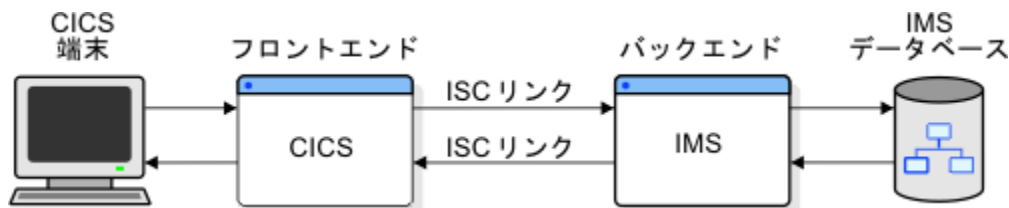


図 56. CICS からの呼び出される既存または新規の IMS トランザクション

上記の図において、2 つの相補トランザクション (一方は IMS システムで、もう一方は CICS システムで実行する) として 1 つのアプリケーション・プログラムを作成することができるため、ISC はトランザクション間の機能を備えています。ユーザー機能は、必要に応じてシステム間で分散することができます。ISC TCP/IP リンクが使用された場合は、非同期 CICS トランザクションのみを使用でき、他の制限が適用されます。

### CICS の BMS に分散される IMS MFS DPM マッピング機能

以下の図は、IMS メッセージ形式サービス (MFS) システムから CICS 基本マッピング・サポート (BMS) システムに装置マッピング機能を分散するための ISC VTAM 機能を示しています。MFS は、CICS アプリケーションに送信されるデータ・ストリームを部分的にマップします。そのアプリケーションは、入力データ・ストリームを BMS にとって受容可能な書式に処理する責任があります。BMS は、その時に装置マッピングを完了させるために使用することができます。

MFS では、ISC TCP/IP 通信はサポートされません。

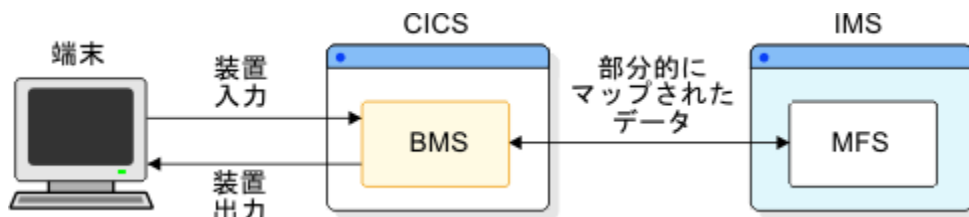


図 57. CICS の BMS に分散される IMS MFS DPM マッピング機能



## ISC VTAM で IMS から IMS へ

以下の図は、2つのIMSシステムがISC VTAMリンクを使用してどのように接続できるかを図示したものです。2つのIMSシステムの相互接続を使用することにより、一方のIMSシステムの新規または修正済みトランザクション (TRAN1) が、他方のIMSシステムの既存のトランザクション (TRAN2) を呼び出すことができます。MFSを使用して、応答を、開始IMSシステムの新規トランザクション (TRAN3)、または発信元トランザクション (TRAN1) の新規インスタンスに経路指定することができます。MFSがない場合には、応答はメッセージ通信として扱われ、ソース端末に返送されます。

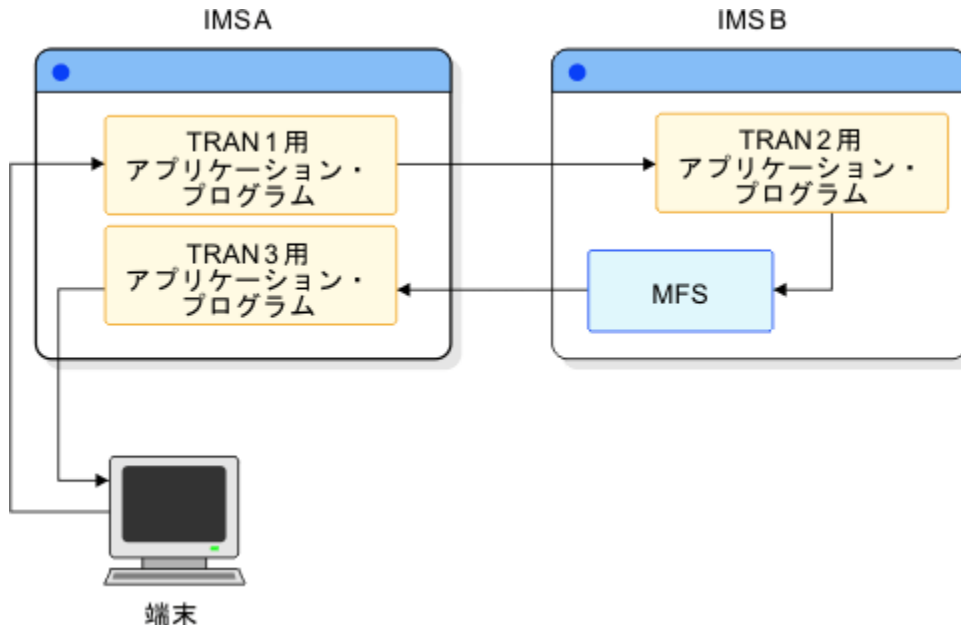


図 58. ISC VTAM で IMS から IMS へ

## ISC を使用して CICS から IMS MSC へ

以下の図は、CICS サブシステムが、別の IMS サブシステムへの MSC リンクを持つ IMS サブシステムとどのように通信できるかを図示したものです。この場合、CICS から IMS へのセッションが ISC によってサポートされている間は、MSC を使用してこれらの IMS サブシステムを結合することができます。CICS サブシステムは、それが通信できる複数の IMS サブシステムを単一システムと見なします。MSC は、複数 IMS サブシステム間のロードを分散します。

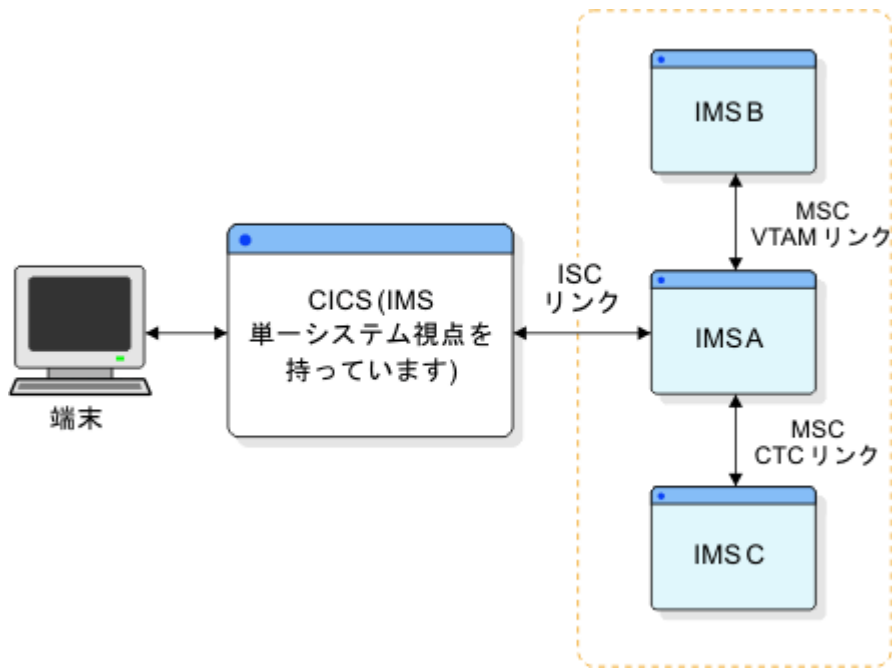


図 59. ISC を使用して CICS から IMS MSC へ

#### 関連資料

673 ページの『IMS-CICS ISC のサンプル・プログラム』

以下のトピックには、IMS と CICS の間での ISC の使用例を示したサンプル・プログラムが記載されています。

## TCP/IP の ISC サポート

IMS サブシステムと CICS サブシステムの間での ISC 通信をサポートするために、TCP/IP を使用することができます。

### このタスクについて

TCP/IP 接続に対する ISC サポートは、仮想記憶通信アクセス方式 (VTAM) によって提供される接続サポートの代替手段を提供します。DFSDCxxx PROCLIB メンバーに ISCTCPIP パラメーターをコーディングすることにより、静的に定義された既存の ISC 端末を TCP/IP サポートに対して使用可能にできます。ETO ログオン記述子を使用して、TCP/IP をサポートする動的 ISC 端末を定義することができます。

ISC TCP/IP 接続で CICS と通信するために、IMS は、セッション制御とデータ・フローに、専用の CICS IP 相互接続 (IPIC) プロトコルを使用します。ただし、アプリケーション・プログラムは、IPIC の使用、または ISC TCP/IP あるいは ISC VTAM の使用の影響を受けません。アプリケーション・プログラムが、ISC TCP/IP によってサポートされていない機能を使用しない限り、リンクのどちら側でもアプリケーション・プログラムを変更せずに ISC VTAM リンクから ISC TCP/IP リンクに切り替えることができます。

#### 関連タスク

145 ページの『IMS Connect および TCP/IP 通信』

IMS の IMS Connect 機能は、TCP/IP 対応の環境から IMS DB および IMS TM の両方へのアクセスを提供します。

607 ページの『TCP/IP を介する CICS との ISC 通信』

IMS サブシステムと IBM CICS Transaction Server for z/OS サブシステムの間の ISC 接続をサポートするために、TCP/IP を使用できます。

## IMS と CICS との間での ISC

次の表に、CICS および IMS が提供する ISC 機能と、CICS **START** または **RETRIEVE**、および **SEND** または **RECEIVE** のユーザーが使用できる IMS 機能を示します。

表 77. CICS START RETRIEVE および SEND RECEIVE ユーザーに使用可能な機能

機能	フロントエンド・システム	バックエンド・システム
分散 トランザクション 処理	CICS <sup>463</sup> ページの 『1』	CICS <sup>463</sup> ページの 『1』
リモートの開始 トランザクション (transaction)	IMS の全トランザクション・タイプ <sup>463</sup> ページの 『2』	IMS 非応答タイプ CICS <sup>463</sup> ページの 『1』
分散 トランザクション 処理	IMS の、会話型を除く全トランザク ション・タイプ <sup>463</sup> ページの 『2』	IMS の、会話型を除く全トランザク ション・タイプ <sup>463</sup> ページの 『2』
分散 トランザクション 処理 アプリケーション アプリケーション 会話 <sup>463</sup> ページの 『5』	CICS <sup>463</sup> ページの 『3』	CICS <sup>463</sup> ページの 『3』
分散 表示 管理 <sup>463</sup> ページの 『5』	IMS MFS マップ	IMS MFS マップ
分散 トランザクション 処理	CICS ユーザー・アプリケーション <sup>463</sup> ページの 『4』 IMS MFS マップ	IMS MFS マップ CICS ユーザー・ アプリケーション <sup>463</sup> ページの 『4』

### 注:

1. CICS START または RETRIEVE コマンドを使用します。
2. 処理サブシステムのトランザクションが応答モードまたは高速機能トランザクションの場合、発信元システムで処理サブシステムからの応答を処理するためには、非応答トランザクション・タイプが必要となります。
3. CICS SEND または RECEIVE コマンドを使用します。
4. ユーザー・アプリケーションは、オプションで BMS を呼び出して、CICS と 端末の間の機能のマッピングを援助することができます。
5. この機能は、TCP/IP を使用する ISC リンクではサポートされていません。

### 関連概念

601 ページの 『IMS および CICS が ISC インターフェースを使用する方法』

IMS と CICS の両方のノードを含む ISC ネットワークを設計およびインプリメントするときには、関連する問題に留意する必要があります。

## 端末の装置依存データ

IMS で ISC を使用することは、各種サブシステム (IMS、CICSIMS、CICS、およびユーザー作成) のプログラムが、他のサブシステムのアプリケーション・プログラムとメッセージを交換できるようにする 1 つの

技法です。これらのプログラムを変更する必要性を最小限にするか、あるいは除去するためには、交換されるメッセージに、発信元端末または宛先端末に固有の装置制御文字を含めてはなりません。

ISC メッセージの作成プログラムが開始される前に、IMS (MFS) または CICS (BMS) の装置制御編集機能を使用して、これらの装置制御コードを除去しなければなりません。同様に、端末に応答を送信するプログラムでは、固有のサブシステムのフォーマット設定サービスを使用して、メッセージの送信前に、そのメッセージを適した形式にする必要があります。このようにして、発信元端末と宛先端末とは関係なく、すべてのプログラム間メッセージ通信を実行することができます。これは、各端末固有のシステムが、そのシステムの端末で入力、または受信される可能性のあるトランザクションの形式をすべて認識していなければならないことを意味します。

## IMS への CICS データの受け渡し

ユーザーは、フロントエンド・アプリケーション・プログラムに、接続端末からの 3270 データ・ストリームを受信して、このデータ・ストリーム (装置制御文字) をバックエンド IMS に単純にパススルーするように要求することができます。

IMS からの結果の出力は、フロントエンド・アプリケーションからパススルーして、3270 に出力することが可能な 3270 データ・ストリームとなるはずですが。

### 入力を IMS にパススルー

3270 データ・ストリームをフロントエンド・アプリケーションから IMS に送信することができます。ただし、IMS はその ISC セッションを 3270 装置として認識せず、また、データ・ストリームからの装置制御文字の編集も行いません。ユーザーの IMS アプリケーション・プログラムが、この編集機能を備えていることが必要です。

### IMS からの出力

同様に、IMS は、ISC LTERM 宛ての出力データ・ストリームには 3270 装置制御文字を入れません。これは、ISC LTERM が 3270 装置ではないためです。ユーザーのアプリケーション・プログラムでは、3270 データ・ストリームが、最小限の編集で、接続端末にこのデータ・ストリームを送信できるフロントエンド・アプリケーションに送信されるように構成する必要があります。

この作業は、各装置および各アプリケーション・プログラムごとに繰り返す必要があり、また、装置形式を変更したときは再度必要となります。

## 第 27 章 ISC 接続に使用される VTAM 機能

VTAM によってサポートされる ISC 接続の場合、VTAM は、IMS と論理装置間の物理的なデータ伝送を制御します。

**制約事項:** ここで述べられている VTAM 機能は、ISC TCP/IP 接続には適用されません。

VTAM アプリケーション (IMS など) と他のサブシステムは両方とも、VTAM 論理装置と見なすことができます。論理装置は、アプリケーション・プログラムやサブシステムのような、アドレス可能なリソースです。また、論理装置は、汎用端末システムのコンポーネントでもかまいません。

ISC に特に関係のある、IMS に使用される VTAM の概念と機能には、次のものがあります。

- 接続、切断およびログオン・モードの設定
- メッセージと応答
- 確定応答要求 1 (RQD1) または確定応答要求 2 (RQD2)、および関連する確定応答 (DR1 または DR2)。LU 6.1 プロトコルでは、確定応答 1 と確定応答 2 が分離され、再定義されていることに注意してください。RQD2 要求および DR2 応答は、現在では同期点要求および応答として認識されており、DR1 に関連した応答に関係なく行われます。
- 順序付けとチェーン
- ブラケット開始の停止 (SBI) および停止ブラケット開始 (BIS) コマンドからの順序セッションの切断
- ブラケットおよび方向変換標識の使用を含む、順序正しい通信を確認するための機能
- シーケンス番号のリカバリー
- 入力を受信とメッセージの送信
- 条件付きブラケット終了
- 拡張エラー・リカバリー・プロシージャ (EERP)
- ISC ノード間の並列セッションの使用
- 交渉可能セッション・バインド・パラメーターおよび交渉不能セッション・バインド・パラメーター双方のサポート

IMS は、サービス・クラス (COS) およびセッション障害通知 (SON) 機能もサポートします。

**関連資料:** VTAM アプリケーション・プログラム (IMS など) と他のサブシステムとの間におけるデータ伝送を管理する通信の概念および機能について詳しくは、「*z/OS Communications Server: SNA Programming*」を参照してください。

ISC システム・プログラマーおよびシステム・アナリストは、IMS と SNA LU 6.1 プロトコルを使用するリモート・サブシステムとの間における通信インターフェースを設計してインプリメントするために、これらの概念および機能に精通する必要があります。

### 関連概念

[504 ページの『セッションの再同期』](#)

サブシステム障害およびセッション障害において、IMS 内のリカバリー可能リソース、メッセージおよびキューの健全性を保守するには、両方のハーフセッションで再同期プロセスに必要なセッション情報を保守しなければなりません。

[430 ページの『IMS における SON/COS サポートの使用』](#)

セッション停止通知 (SON) とサービス・クラス (COS) は、IMS がセッション停止を認識できるようにする VTAM と SNA の機能です。

### 関連資料

[554 ページの『LU 6.1 \(SBI および BIS\) の対称セッション・シャットダウン』](#)

停止ブラケット開始 (SBI) とブラケット開始の停止 (BIS) という 2 つのデータ・フロー制御コマンドにより、対等レベルの LU 6.1 ハーフセッションが対称的かつ正常に終了することができます。

## VTAM コマンドと標識

IMS アプリケーション・プログラムと他の VTAM 論理装置間のデータ伝送には、VTAM コマンドおよび標識 (通信制御情報) が必要です。

次の表は、IMS が IMS セッション中に送受信する VTAM コマンドと標識を示しています (X=サポートされます)。サポートされないコマンドまたは標識を使用した場合は、その結果を予測することはできません。

表 78. IMS によって送受信される VTAM コマンドおよび標識

VTAM コマンド / 標識	IMS は 1 次ハーフセッションとして送信し、2 次ハーフセッションとして受信します。	IMS は 2 次ハーフセッションとして送信し、1 次ハーフセッションとして受信します。
<b>セッションから独立したもの</b>		
セッション開始		注 467 ページの『4』
プロシージャ・エラー	注 467 ページの『1』	注 467 ページの『1』
セッション終了		注 467 ページの『5』
<b>セッション制御コマンド</b>		
バインド	X	
バインド応答		X
STSN	X	
STSN 応答		X
開始データ通信 (SDT)	X	
SDT 応答		X
アンバインド	X	X
<b>NORMAL (同期) FLOW インディケータ</b>		
ブラケット開始 (BB)	X	X
ブラケット終了 (EB)	X	X
方向変換 (CD)	X	X
<b>コマンド</b>		
BID	注 467 ページの『3』	X
CANCEL	X	X
CHASE	X	X
論理装置状況 (LUSTATUS)	X	注 467 ページの『2』
受信準備済み (RTR)		X
ブラケット開始の停止 (BIS)	X	
<b>EXPEDITED (非同期) FLOW コマンド</b>		
要求のシャットダウン (RSHUT)	X	注 467 ページの『2』
信号	X	X

表 78. IMS によって送受信される VTAM コマンドおよび標識 (続き)

VTAM コマンド / 標識	IMS は 1 次ハーフセッションとして送信し、2 次ハーフセッションとして受信します。	IMS は 2 次ハーフセッションとして送信し、1 次ハーフセッションとして受信します。
停止ブラケット開始 (SBI)		X

注:

1. VTAM による送信。
2. IMS によっては送信されない。ただし IMS が 1 次ハーフセッションである場合は、オプションとして、他のサブシステムにより送信され、IMS により受信される。
3. IMS によっては送信されない。ただし IMS が 2 次ハーフセッションである場合は、オプションとして、他のサブシステムにより送信され、IMS により受信される。
4. CINIT (LOGON 出口) の内部 VTAM 形式によりサポートされる。
5. CTERM (NS および LOSTERM 出口) の内部 VTAM 形式によりサポートされる。

## VTAM アプリケーション・プログラミング・インターフェースの使用

IMS によって ISC に使用される VTAM アプリケーション・プログラミング・インターフェース (API) は、マクロ命令と制御ブロックからなっています。

これらのマクロ命令と制御ブロックにより、IMS は以下のことができます。

- 接続の設定または接続の切断
- IMS と他の論理装置との間におけるデータ転送の要求および制御

アプリケーション間通信を行うために、VTAM API は基本および 2 次セッション・アプリケーション・インターフェースを提供します。基本インターフェースは、他の VTAM ノードをサポートする IMS に使用されるものと本質的に同一です。ただし、インターフェースは、一部の新規マクロ、および既存マクロに追加されたパラメーターを組み込んでいます。

VTAM API は、単一セッションに加え、2 つの論理装置間の並列 (複数で、かつ同時にアクティブにする) セッションもサポートします。このサポートの IMS インターフェースは、基本的に、LOGON および SCIP 出口ルーチンを介して利用可能なセッション修飾子フィールドの CINIT および BIND によるものです。単一セッション・サポートでは、IMS システム定義時にメッセージ・キューの静的定義が必要ですが、並列セッション・サポートでは、セッション開始時にメッセージ・キューを動的に割り振ることができます。

VTAM の詳細については、「z/OS Communications Server: SNA リソース定義解説書」を参照してください。

### 関連概念

[VTAM インターフェースの考慮事項 \(システム管理\)](#)

## 接続確立時のログオン・モードの指定

接続を確立するとき、IMS では、論理装置が IMS と通信するときに従わなければならない規則をセッション・パラメーターで定義する必要があります。これらのセッション・パラメーターは VTAM モード・テーブルに含まれています。

### このタスクについて

次の 2 つの方法のいずれかで、デフォルト・ログオン・モード・テーブル項目の使用をオーバーライドすることができます。

- システム定義時に、MODETBL=キーワードを使用する TERMINAL マクロ、もしくは ETO ログオン記述子でログオン・モード項目を指定することができます。
- /OPNDST コマンドの MODETBL キーワードは、ログオン・モード・テーブル項目を指定して、システム定義時に定義されているテーブル項目を置き換えることができます。



### 関連資料:

- MODETBL キーワードの指定方法についての詳細は、「IMS V15 システム定義」を参照してください。
- /OPNDST コマンドの指定方法についての詳細は、「IMS V15 コマンド 第2巻: IMS コマンド N-V」を参照してください。

ログオン・モード・テーブル項目の指定にどちらの方式も使用されない場合、VTAM ノード・モード・テーブル項目定義によってデフォルト・ログオン・モード・テーブルがオーバーライドされていなければ、VTAM は、デフォルト・ログオン・モード・テーブルの最初の項目を使用します。ログオン・モード・テーブル項目が指定されている場合、VTAM は、指定された項目をデフォルトまたはユーザー指定のログオン・モード・テーブルで検索します。

IMS は、指定された VTAM モード・テーブル項目内のセッション・パラメーターのセットを調べ、IMS が依存するパラメーターのみをオーバーレイします。残りのバイトは変更されません。IMS は、セッション・パラメーターで伝送されたすべての不定形式ユーザー・データを無視します。

**関連資料:** ログオン・モード・テーブルの設定と、ログオン・モード・テーブル項目の定義について詳しくは、「z/OS Communications Server: SNA リソース定義解説書」を参照してください。

### 関連資料

[SLUP と LU 6.1 のバインド・パラメーター \(システム・プログラミング API\)](#)

[IMS 環境で使用されるマクロ \(システム定義\)](#)

[/OPNDST コマンド \(コマンド\)](#)

## 2 次論理装置の設計考慮事項

IMS と他のサブシステムが ISC セッションを確立した場合、一方のセッション・パートナーがビッダー、そして他方がファースト・スピーカー (競合勝者) になります。

IMS ISC セッションでは、ビッダーは常に基本論理装置 (バインド送信側) であり、ファースト・スピーカーは常に 2 次論理装置 (バインド受信側) です。ISC ネットワーク内の IMS と通信させる 2 次論理装置にシステム設計を提供するときは、以下の点を考慮してください。

- ファースト・スピーカー (2 次論理装置) は、ブラケット・コンテンションを解決し、常時、割り込み (例えば、VTAM コマンドまたはブラケット標識など) を受信できる状態にしておかなければなりません。ファースト・スピーカーがこのように準備されていないと、デッドロックが生じることがあります (例えば、基本論理装置と 2 次論理装置の両方が VTAM SEND マクロを使用し、POST=SCHED ではなく POST=RESPONSE を発行する場合)。
- VTAM TERMSESS マクロを使用してセッションを終了する時は、システム設計者は、CONDITIONAL を発行した場合と UNCONDITIONAL を発行した場合の違いを認識していなければなりません。CONDITIONAL を送信すると、基本論理装置を終結処理し、セッション終了を制御できます。UNCONDITIONAL を送信すると、基本論理装置の通知前に VTAM によってセッションが即時終了されます。例えば、応答を待っているときに、基本論理装置が延期状態にある場合には、システム設計者は、UNCONDITIONAL セッション終了が予測不能な結果をもたらす可能性があることを知っておく必要があります。

「2 次論理装置」という用語は「ファースト・スピーカー」と、「基本論理装置」という用語は「ビッダー」とそれぞれ同義語です。

## 第 28 章 ISC によって影響を受ける IMS 機能

以下のトピックでは、IMS 機能が ISC 処理から受ける影響について説明します。

### メッセージの編集

IMS 内では、通信インターフェースはアプリケーション・プログラムにとっては透過的です。

これをサポートするために、IMS は次のような共通の編集機能を提供します。

- プロトコルと機能管理ヘッダーをアウトバウンド・メッセージに付加する。
- 着信メッセージから機能管理ヘッダーを取り除き、メッセージを IMS 制御ブロック内の適切なポイントに移動する。

この共通編集の後に、IMS が受信するメッセージは、入力 ATTACH または SCHEDULER DPN パラメーターに基づいて、ISC 編集、メッセージ形式サービス (MFS)、または基本編集によって編集されます。IMS が送信するメッセージは、送信前にメッセージ形式サービスで編集されます。ISC 編集および基本編集の編集機能は固有なものです。しかしながら、関連プロトコルは同一です。

- ISC 編集は、デフォルト・エディターです。ISC 編集が要求され、その要求が、入力データの後ろに SNA 定義の PRN パラメーターを含んでいない場合、メッセージは部分的入力データ透過性で編集されます。つまり、メッセージの最初の部分に存在する必要がある IMS 必須宛先コードおよびオプション・パスワードだけが ISC 編集機能によって編集されます。入力メッセージの残りは、トランザクションが大文字に変換することを定義しない限り、処理前に編集されません。全入力データの透過性は、ATTACH または SCHEDULER FM ヘッダーにある SNA 定義の PRN パラメーターを使用した ISC 編集で得られます。このことにより、IMS トランザクション・コードをメッセージとは別に扱うことができ、また IMS が入力を受け取り、その入力を入力メッセージ自体の検証なしに、適切な宛先 (端末またはトランザクション) に経路指定することができます。トランザクション・コードは、データ・ストリームとは別にあるので、オプション・パスワードは、そのセッションに使用できません。したがって、パスワード・セキュリティを定義する必要はありません。パスワード・セキュリティが定義される場合には、エラーとなります。
- メッセージ形式サービスの分散表示管理 (DPM) オプションは、MFS と別のサブシステムに常駐するプログラム間のメッセージ形式処理を分割します。DPM-Bn を使用する場合、物理形式の端末特性を MFS に定義しません。その代わりに、IMS は別のサブシステムのプログラムに MFS 形式のデータを送信します。そのプログラムは、必要に応じてその形式処理を完了し、必要なら、物理装置にそのデータを渡します。DPM は、またユーザーに入出力の ISC メッセージ経路指定パラメーターを指定することを許可しています。また、ATTACH または SCHEDULER FM ヘッダーの PRN パラメーターを使用して、結果の MFS 形式の入力データ・ストリームに関して、IMS 宛先を外部から指定することもできます。
- 基本編集では、端末入力用の標準編集ができます。基本編集は、装置とオペレーターの制御文字の編集に使用され、また、既存のアプリケーションに対する入力の互換性を維持することが重要である場合に ISC セッションで使用することができます。基本編集用のプロトコルは、ISC 編集用のものと同じです。基本編集が使用される場合には、ATTACH または SCHEDULER FM ヘッダーの PRN パラメーターの使用は効果がありません。メッセージの宛先は、入力データ・ストリームの最初のフィールドで判別されます。

IMS では、基本編集、ISC 編集または MFS を呼び出して、トランザクション、コマンド、およびメッセージ通信の通信を処理することができます。MFS DPM 機能は、コンポーネントごとにオプションとして定義されます。使用可能なこれらのプロセスはいずれも、ATTACH または SCHEDULER FM ヘッダーを使用してメッセージごとに選択することができます。

ISC 入出力は、大部分の IMS データ通信出口ルーチンで編集することができます。例外は、MSC 関連出口ルーチンとハードウェア必須ルーチンです。

#### 関連タスク

435 ページの『IMS メッセージの編集とフォーマット設定』

IMS、メッセージ形式サービス (MFS) および基本編集ルーチンの 2 つの方法によって、端末でやり取りするメッセージの編集およびフォーマット設定を行います。

#### 関連資料

479 ページの『メッセージ経路指定の FM ヘッダー』

ISC メッセージ経路指定情報は、SNA 定義の機能管理 (FM) ヘッダー内で提供されます。

556 ページの『ISC 編集を呼び出すための FM ヘッダーの使用』

以下の状況では IMS は、トランザクションを編集するための ISC 編集 (ISCEDT)、コマンド、および IMS - ISC セッション用の LTERM 間でのメッセージ通信を使用することができます。

MFS による入力メッセージのフォーマット設定方法 (アプリケーション・プログラミング API)

## ISC セッションからの IMS コマンドの発行

IMS オペレーター・コマンドは、ISC セッションにも使用できますが、本来は IMS マスター端末および IMS システムに直接接続されているリモート端末の、適切な許可を与えられたオペレーターが使用するよう意図したものです。

他の (非 IMS) サブシステムの端末オペレーターは IMS と直接通信せずに、そのサブシステム内の制御プログラムまたはユーザー提供のアプリケーションと通信します。そのサブシステムでは、端末オペレーターおよびアプリケーション・プログラマーが IMS と通信する際に従うプロシージャーを決定し提供する必要があります。非 IMS サブシステム内のアプリケーションが、ISC セッションを介して IMS コマンドを発行することを許可される場合は、IMS 従属関係がそのアプリケーションに組み込まれます。ISC ハーフセッションは、基本または 2 次マスター端末としては定義できません。

## 並列セッションへの影響

許可された IMS コマンドを NODE キーワードと一緒に使用する場合は、USER キーワードを組み込んでください。

特定の並列セッション・インスタンスが USER キーワードで IMS に対して識別されない場合は、示されたノードのすべてのアクティブまたは潜在的にアクティブな並列セッションに対して同等のアクションが取られます。

例えば、USER を指定せずに **/CLSDST NODE x** コマンドを出すと、NODE x のすべてのアクティブ並列セッションの終了がスケジュールされます。USER を指定せずに **/STOP NODE x** コマンドを出すと、NODE x のすべてのアクティブ並列セッションの終了がスケジュールされ、NODE x での新規並列セッションの開始も行われません。

## ISC VTAM セッション用 IMS テスト・モードの使用

バックエンド IMS システムをテスト・モードにすることによって、ISC VTAM 通信プロトコルと編集機能をテストすることができます。

### このタスクについて

**制約事項:** IMS テスト・モードは、ISC TCP/IP セッションについてはサポートされません。

テスト・モードでは、IMS が入力メッセージを受信し、入力プロトコルと FM ヘッダー・パラメーターのチェック、入出力編集の実行、適切な出力プロトコルと FM ヘッダー・パラメーターの挿入を行い、そしてそのメッセージをフロントエンド・サブシステムに戻します。非通信エラー分析、トランザクション・コード検証およびトランザクション・スケジューリングは、バイパスされます。また、テスト・モード中は、IMS は非同期出力を送信しません。

IMS **/TEST** コマンドが ISC セッションで受信されると、特定のサブシステムの中の 1 つまたはすべての ISC セッションをテスト・モードに置くことができます。**/TEST** コマンドは、テスト・モードに置かれている端末のみ入力することができます。テスト・モードは、SNA **LUSTATUS** または **CHASE** コマンドでブレット終了 (EB) 標識を送信することで終了することができます。またテスト・モードは、**/END** コマンドが ISC セッションで受信された場合、あるいは、このコマンドが許可端末オペレーターによってバックエンド IMS プロセッサでローカルに入力された場合にも終了することができます。

IMS 間環境では、ISC メッセージ通信機能を使用すれば、バックエンド IMS システムをテスト・モードに置き、メッセージを送信し、そのエコー応答を受信し、テスト・モードを終了することができます。この機能は、各サブシステムにユーザー・アプリケーションを必要としない有効なテスト方法を提供します。

## ISC 並列セッションでの IMS 制御ブロック・ストレージ

潜在的セッションを表す IMS 制御ブロック構造に必要なストレージは、入出力メッセージ・キューを表す制御ブロックに必要なストレージより大きくなります。

必要な論理装置タイプ 6.1 がごくわずかであるインストール済み環境では、それぞれにおける関連並列セッションの数が比較的少数であるが、そこに動的に割り振られる ISC ユーザー (サブプールまたは LTERM のセット) の数が多数である場合に、ストレージ要件はより低くなる可能性があります。静的 ISC 端末では、これらのユーザー (サブプール) も静的に定義されます。ISC 単一セッション (静的に定義され割り振られる LTERM) 用の IMS 制御ブロック構造は、IMS 内の他の静的 VTAM 端末タイプの場合と同じです。

ETO-ISC 間セッションの最大数を定義する必要はありません。ストレージおよびプロセッサの容量を超えない限り、セッションの追加を継続することができます。そのため、SESSION= キーワードは ETO 記述子でサポートされるキーワードの 1 つではありません。静的 ISC セッション用のセッションの最大数を定義するときは、TERMINAL マクロの SESSION キーワードを使用します。

### 関連資料

659 ページの『ISC データ・フロー制御の例』

以下のトピックでは、ISC データ・フロー制御の例を示します。

## ISC と IMS の実行モードの関係

「同期」と「非同期」という用語は、IMS と ISC では少し意味が異なります。以下のトピックでは、これらの実行モードの関係について説明します。

### 実行モードの外部指定

ISC VTAM におけるメッセージは、SNA 定義の機能管理ヘッダーを使用します。VTAM によって送信される ISC メッセージは、同期または非同期で処理することができます。これは、最初に FM ヘッダーによって決定され、次に、メッセージと共に送信される VTAM ブラケット・プロトコルによって決定されます。

IMS と IBM CICS Transaction Server for z/OS との間の通信に ISC TCP/IP を使用するメッセージは、IPIC メッセージ・フォーマットを使用します。これは、CICS によって定義された内部プロトコルです。ISC TCP/IP メッセージは、非同期でのみ処理することができます。

以下の表に、サポートされる VTAM プロトコルの要約を示します。メッセージ・モード (同期または非同期) を設定するために FM ヘッダー および VTAM ブラケット・プロトコルを使用することは、そのメッセージの範囲外と見なされます。

表 79. FM ヘッダーで要求される処理モード

VTAM ブラケット・プロトコルを使用した FM ヘッダー	同期	非同期
CD 付き ATTACH	X	
EB 付き ATTACH		X
EB または CD なし ATTACH	X	
CD 付き SCHEDULER	X	X
EB 付き SCHEDULER		X
EB または CD なし SCHEDULER	X	X

ATTACH FM ヘッダーによって IMS が受信するメッセージは、以下のように処理されます。

- メッセージを受信したあとにセッション状態を「ブラケット中」にしておく場合は、同期に処理されません。



- 次のいずれかの場合には、非同期に処理されます。
  - メッセージを受信したあと、セッション状態を「ブラケット間」にしておく。
  - メッセージが、先頭チェーンまたは単独チェーン上 EB 付きで送信される。

同期処理中、非送信請求または非同期出力は送信されません。

SCHEDULER FM ヘッダーによって IMS が受信するすべてのメッセージは、セッションに関連して受信側で非同期に処理されます。

#### 関連概念

518 ページの『IMS 応答モードまたは会話型出力エラーの処理』

このトピックでは、ISC セッション中の応答モード・エラーと会話型エラーを IMS がどう処理するか、およびハーフセッションの同期を維持する方法について説明します。

#### 関連資料

555 ページの『機能管理ヘッダー』

SNA では、機能管理 (FM) ヘッダーはリンクに対し送信される要求単位のオプションの部分です。このトピックでは、ISC セッション上の IMS によってサポートされる FM ヘッダーについて説明します。

## 実行モードの内部定義

IMS 内部実行モードは、IMS がトランザクションをどのように処理するかを決定します。

例えば、応答モードおよび会話型モードは、特定の入力メッセージとその関連 応答間の同期性を確認するために使用されます。トランザクションは、セッションとアプリケーション間の IMS 入出力メッセージ・キューを使用しますが、これらのモードは、関連メッセージが常にソース・セッションと関連して同期に処理されることを確認します。これらのメッセージの処理中、IMS はそのセッションで非同期または非送信請求出力を送信しません。

内部 IMS 定義は同期または非同期のいずれかです。

次のモードは同期です。特定のコマンドを除き、ISC TCP/IP セッションでは以下のモードはサポートされません。

- 応答モード
- 会話型モード
- 高速機能 (Fast Path)
- コマンド
- テスト・モード

ISC TCP/IP セッションがサポートする IMS コマンドとしては、**/DISPLAY** および **/RDISPLAY** があります。

次のモードは非同期です。

- 非応答モード
- 非会話型モード
- メッセージ通信

## ISC VTAM 通信中の結果としての処理モード

外部的に要求された実行モードと内部で了解されている処理モードとの関係に矛盾がない場合、ISC VTAM 通信時のメッセージは要求どおりに処理されます。2つの指定 (内部対外部) が整合しない場合、そのセッションに関するメッセージの実行は同期で行われます。

いずれの場合も、生成された応答が、要求と同じセッションで戻される場合、その応答は、入力メッセージと一緒に受信された同じタイプの FM ヘッダーを使用して送信されます。応答が入力メッセージを受信したセッション以外のセッションに返される場合、その応答は非送信請求非同期出力と見なされ、SCHEDULER FM ヘッダーと共に送信されます。応答を受信するサブシステムが SCHEDULER ヘッダーの受信をサポートしていない場合、その応答は、IMS メッセージの終わりにブラケットを終了する ATTACH ヘッダーを付けて送信されます。そのため、最後でない会話、最後でない MFS 要求時ページング・メッセージ、またはテスト・モード出力を除いては、ATTACH SCHEDULER の代わりに ATTACH が使用され、メッ

ページの最初または唯一のチェーン上に EB が示されます。最後でない会話、最後でない MFS 要求時ページング・メッセージ、およびテスト・モード 出力は、メッセージの最初または唯一のチェーンに表示される CD 付き ATTACH を伴って送信されます。

ただし、メッセージがフロントエンド・サブシステムからバックエンド・サブシステムに送信され、外部的に同期処理を要求する場合、そのセッションに対する出力応答のタイミングまたは他の出力の可用性に関して、フロントエンド・サブシステムが前提となつてはなりません。その結果、要求と応答は、バックエンド・サブシステムにおける実行が同期に行われた場合であっても、フロントエンド・サブシステムでは相互に関連することはありません。これを要約すると、次の表のようになります。

フロントエンド切り替え出口ルーチンによって提供される、フロントエンド/バックエンド・システム使用のための特別なサポートがあります。

2つの IMS システム間の通信は、常に非同期形式で送信されます。バックエンド・システム内のトランザクションの内部定義が同期である場合、通信は同期に処理され、その ISC セッションは同期出力用に保持されます。ただし、この同じ関係がソース端末に適用されるわけではありません。フロントエンド IMS は、フロントエンド切り替え出口ルーチンを使用していない限り、バックエンド・サブシステムから応答を受信するまで、その端末を 応答モードに保持しません。

表 80. 内部対外部の実行モード仕様

外部指定 (FMH)		
内部指定 (システム定義)	同期 473 ページの『1』	非同期 473 ページの『2』
同期	同期	非同期 473 ページの『3』
非同期	同期 473 ページの『4』, 473 ページの『5』	非同期

注:

1. 同期指定は、メッセージの最初または唯一のチェーン上に EB が示された ATTACH を含みません。
2. 非同期指定は、メッセージの最初または唯一のチェーン上に EB が示された ATTACH を含みます。
3. IMS における同期操作は、SNA 定義の外部非同期形式が IMS をメッセージ受信側として独自の解釈でインバウンド・メッセージを実行することができるので、この場合で行うことができます。このように、IMS の内部同期定義は、外部非同期形式をオーバーライドします。ただし、IMS の内部 (システム定義) 実行モードが同期で、かつメッセージを ATTACH EB と共に受信した場合、IMS は、エラー・メッセージを生成してそのセッションを終了します。
4. このサポートでは、メッセージが内部的に非同期 (非応答モード) になるように定義されていても、応答モードを動的に (メッセージごとに) 確立することができます。
5. OPTIONS=NORESP が IMS TERMINAL マクロ または ETO ユーザー記述子で内部的に指定されている場合、メッセージは、外部的に同期として定義されないことがあります。システム定義と FM ヘッダーの意図との、この対立は、セッション終了を引き起こします。

2つの IMS システム間の通信は、常に非同期形式で送信されます。バックエンド・システム内のトランザクションの内部定義が同期である場合、通信は同期に処理され、その ISC セッションは同期出力用に保持されます。ただし、この同じ関係がソース端末に適用されるわけではありません。フロントエンド IMS は、フロントエンド切り替え出口ルーチンを使用していない限り、バックエンド・サブシステムから応答を受信するまで、その端末を 応答モードに保持しません。

#### 関連資料

フロントエンド切り替え出口ルーチン (DFSFEBJ0) (出口ルーチン)

## LTERM ユーザー (サブプール) とコンポーネント

IMS ユーザー・ブロックは、IMS システム定義時に SUBPOOL マクロによって定義される、もしくは ETO ユーザー記述子から動的に作成される IMS 論理端末 (LTERM) のセットです。

システム定義時に定義されるサブプールは ETO LU 6.1 端末では使用できません。ISC に定義されるユーザー (サブプール) は、ダイヤル式端末用に定義されるユーザーとは別で、ISC 並列セッション・サポートと関連するときのみ許可されます。

**定義:** すべての静的 ISC ユーザーの集合を VTAMPOOL と言います。VTAMPOOL 内で定義された LTERM は、VTAMPOOL 内のユーザー間のみで再割り当てすることができます。VTAMPOOL 内で定義されていない LTERM は、VTAMPOOL の中に割り当てることができません。

セッション開始の結果、ISC ユーザーは、ISC セッション・インスタンスに動的に割り当てられます。すなわち、これらのパラメーターは、特定のセッション・インスタンスに使用可能なユーザーを定義します。このユーザーは、セッションおよびサブシステムの障害が起こっても、IMS と他のサブシステムの双方合意で正常終了して解放されるまで指定の並列セッションに割り振られたままになります。単一の非並列セッションの場合、1セットの LTERM の割り振りは、システム定義中に固定されます。

各 IMS LTERM は、1つの入力 IMS コンポーネントおよび 1つの出力の IMS コンポーネントと関連付けられます。入力と出力のコンポーネントは、同じコンポーネントでもよく、また、別のコンポーネントを指定することもできます。逆に、IMS は、複数の入力および出力の LTERM を単一のコンポーネントと関連付けることはできません。そのようにすると、入力コンポーネントの判別または出力表示に問題を引き起こすおそれがあります。

IMS は、入力コンポーネント ID を使用して、入力メッセージと関連付けられる LTERM を識別します。他の端末サポートの場合、IMS、すべての入力が、必要な操作上の検査およびセキュリティ検査をパスするリストの最初の LTERM から行われるものと想定します。しかし、ISC ノードからの入力の場合、入力コンポーネントは、ATTACH FM ヘッダーで示されたコンポーネントに基づいて判別されます。FM が付かないヘッダーを受信すると、IMSIMS はその入力を入力コンポーネント (ICOMPT) として LTERM と関連付けるものと見なします。コンポーネント値の決定後は、関連 LTERM が検出できなかったり、停止したり、作動不能である場合、あるいはその LTERM がセキュリティ検査をパスできなかった場合、そのメッセージは受け入れられません。

出力が送信される場合、出力は出力 LTERM で識別されたコンポーネント (COMPT) に送られます。メッセージ通信、特定の LTERM に対するブロードキャスト・メッセージ、およびトランザクションからのデータ応答は、指定した出力 LTERM に関連付けられたコンポーネントに送られます。ユーザー作成 MPP は、入出力 PCB に挿入することができ、デフォルトとして、選択した入力 LTERM と関連する出力コンポーネントに送ることができます。また、代替の PCB に挿入して適切な LTERM 名により特定のコンポーネントにアドレッシングすることもできます。

入力コンポーネントと出力コンポーネント間の適切な関係は、IMS システム定義時に NAME マクロを使用して、または ETO ユーザー記述子上の LTERM キーワードにより設定することができます。これにより、論理装置はその入力コンポーネントを指示することができ、出力を IMS システム定義時または ETO ユーザー記述子で指示された関連出力コンポーネントに戻すことができます。コンポーネントを適切に定義して使用することにより、LTERM 命名規則、DL/I CHNG 呼び出し、および代替 PCB への挿入の必要性を削減または除去することができます。

**推奨:** ISC ユーザー (サブプール) の定義時に、メッセージ削除システム定義パラメーターの MSGDEL あるいは ETO ユーザー記述子を誤って指定すると、セッションが開始されない可能性があります。ISC セッションは、TERMINAL または ETO ユーザー記述子と SUBPOOL マクロの MSGDEL 指定が一致した場合にのみ開始することができます。

さらに、ISC メッセージ通信サポートまたはローカル・トランザクションからの代替 PCB 挿入を使用して、メッセージを他のサブシステムに経路指定し、応答をソース端末オペレーターに返す場合には、ISC セッションとソース端末の両方に関連する IMS システム定義 TERMINAL マクロまたは ETO ログオン記述子に MSGDEL=SYSINFO を指定する必要があります。ISC セッションに MSGDEL=NONIOPCB を指定すると、ISC メッセージを他のサブシステムに経路指定できなくなります。ソース端末に MSGDEL=NONIOPCB を指定すると、メッセージ応答をソース端末オペレーターに経路指定できなくなります。

IMS が使用する出力ブラケットと送受信プロトコルは、システム定義出力コンポーネントの指定、IMS へのメッセージ入力に使用するプロトコル、および受信された IMS メッセージのタイプの組み合わせによって決まります。例えば、IMS 同期メッセージ・タイプには、出力応答用の事前定義プロトコルがあります。非同期出力に対する送信 / 受信プロトコルの定義は、出力コンポーネントと関連付けられます。

## 関連概念

480 ページの『IMS による経路指定パラメーターの使用』

フロントエンド切り替え出口ルーチンなしの IMS ISC は、IMS フロントエンド・サブシステムである場合は自動トランザクション経路指定を行いません。

VTAM 端末の定義 (システム定義)



[ETO 記述子 \(システム定義\)](#)

**関連資料**

[532 ページの『IMS 出力のブラケット・プロトコル』](#)

IMS によって使用される出力ブラケットおよび送信/受信プロトコル、およびブラケットごとに送信される出力メッセージ数はさまざまな要因に応じて異なります。

[VTAMPOOL マクロ \(システム定義\)](#)

[SUBPOOL マクロ \(システム定義\)](#)

[TERMINAL マクロ \(システム定義\)](#)



## 第 29 章 ISC プロトコルを使用した通信の設計

以下のトピックでは、ISC プロトコルを使用したサブシステム間通信の概要を説明します。

### 出力プロトコルの判別

ISC セッションの通信は、メッセージに関連した VTAM プロトコルによって制御されます。

使用される基本ブラケット・プロトコルは次のものです。

- ブラケット開始 (BB)

ブラケットの開始信号。ブラケット開始は、ファースト・スピーカー (2 次ハーフセッション) によって出された時には無条件要求であり、ビッダー (1 次ハーフセッション) によって出された時には条件付き要求です。

- ブラケット終了 (EB)

ブラケットの終了信号であり、コンテンション状態のセッションに配置され、いずれのセッション・パートナーにも新規ブラケットの要求を許可します。

- 方向変換 (CD)

セッションの制御をセッション・パートナーに変更し、セッション・パートナーにセッション通信の送信を許可します。

ISC セッション間のトラフィックは、セッションに対して同期的に送受信される可能性があります。メッセージの処理は、セッションで同期的にも非同期にも実行されます。つまり、送信サブシステムは、メッセージを開始して、応答を待つ (同期) こともあり、待たない (非同期) こともあります。

ISC では、(FM ヘッダーで指定されるような) 外部同期モードの操作には、伝送メッセージおよび関連した応答が必要であり、適切な場合には、単一ブラケット内で発生します。同期モードの終了は EB で信号されます。同期モードでは IMS は必要ならば、同一のブラケット内で (EB が受信される前に)、メッセージ通信の例外を伴う未解決のメッセージ・トラフィックに対して応答しなければなりません。入力メッセージプロトコルにより、IMS で同一ブラケット内で必要な応答を送信できない場合は、入力メッセージは拒否されます。

IMS では、ATTACH を指定して送信しなければならない (他のサブシステムに SCHEDULER サポートがないため) 出力に使用されるプロトコルは、発信側入力トランザクションと結果出力応答が同期 (ATTACH) であるか非同期 (SCHEDULER) であるかに関係なく、事前定義されます。この出力には、次のものが含まれます。

- MFS ページ出力の最後でないチェーン (ページ)
- 応答モード、会話モード、テスト・モード、および IMS コマンド応答の最後 (MFS ページ) または単一のチェーン
- 非同期出力

別の非同期出力の最後 (MFS ページ) または単一チェーンで使用されるプロトコルを IMS に定義する必要があります。これらは、TERMINAL マクロまたは ETO ログオン記述子の COMPTn キーワードで定義されます。以下の 4 つのパラメーターが提供されています。

#### **SINGLE1**

このコンポーネントの非同期出力には、ブラケットごとに 1 つのメッセージが送信されます。

#### **SINGLE2**

このコンポーネントの非同期出力には、VTAM 開始ブラケットで一度に 1 つのメッセージ (必要な場合) と、受信中のサブシステムでそのトラフィックを任意で送信することができる方向変換標識が送信されます。

#### **MULT1**

LTERM のすべての非同期メッセージは、ブラケットの終了前に送信されます。

## MULT2

LTERM のすべての非同期メッセージは、方向変換の送信前に送信されます。

IMS 間セッションでは、SINGLE1、SINGLE2、MULT1、または MULT2 の選択がトランザクション・タイプおよび受信 IMS サブシステムの特性に関連していることがあります。次の事項も考慮してください。

## SINGLE1

このプロトコルは、受信 IMS へのメッセージ通信、非応答トランザクション、あるいは非会話型トランザクションの送信に適しています。

## SINGLE2

同期通信では、入力で正しい出力が生成されることを前提としているため、このプロトコルは、応答モード (高速機能を含む) のトランザクション、コマンド、およびテスト・モードの送信に適しています。IMS 間セッションでは、会話型モードはサポートされません。

## MULT1 または MULT2

これらのプロトコルは、SINGLE1 または SINGLE2 での同期通信の取り扱い方と同様に非同期通信を取り扱います。これは、大量の非同期通信が一方 (MULT1) または両方 (MULT2) のサブシステムのキューにある場合に、コンテンツの抑制に有用です。

## 関連概念

[471 ページの『ISC と IMS の実行モードの関係』](#)

「同期」と「非同期」という用語は、IMS と ISC では少し意味が異なります。以下のトピックでは、これらの実行モードの関係について説明します。

## 関連資料

[528 ページの『ブラケットおよび半二重プロトコル』](#)

IMS では、SNA ブラケット・プロトコルを使用してコンテンツを解決し、半二重プロトコルの方向変換標識を使用して、ブラケット状態である間の通常フローの送信/受信モードを制御します。

[532 ページの『IMS 出力のブラケット・プロトコル』](#)

IMS によって使用される出力ブラケットおよび送信/受信プロトコル、およびブラケットごとに送信される出力メッセージ数はさまざまな要因に応じて異なります。

# ISC での既存のアプリケーション・プログラムへのアクセス

ISC セッション中に、ISC サポートのインストール前に書き込まれたアプリケーション・プログラムにアクセスすることができます。

これらのプログラムへのアクセスを計画する場合は、以下のトピックの情報を考慮してください。

## MFS を使用するプログラムへのアクセス

MFS プログラムを使用するプログラムが MFS MODname に左右される場合は、ISC セッションから入力を受信するときに MFS を使用する必要があります。

例えば、プログラムが IOPCB の MODname を論理の一部としてテストする場合、あるいは MODname を呼び出しの一部として組み込んでいる IOPCB に、ISRT 呼び出しを使用する場合は、MFS MODname はプログラムにとって重要なものです。

プログラムでカーソル位置が重要となる場合は、ISC セッション入力の受信に使用することはできません。ISC LTERM はハードウェア装置ではないので、カーソル位置には意味がありません。

出力では、MFS は属性バイトおよび拡張属性バイトをサポートします。

ISC は、次の方法でシステム制御域 (SCA) およびデフォルト・システム制御域 (DSCA) をサポートしています。3270 表示装置に使用できるすべての機能は、SCA として定義されたメッセージ・フィールド (MFLD) 内のアプリケーション・プログラムによって指定することができます。装置フィールド (DFLD ステートメント) は、装置出力形式 (DOF) に SCA として定義されることがあります。ISC サブシステムでは、MFS は SCA 仕様を解釈せず、単にリモート・プログラムまたは ISC サブシステムに送信される、ユーザー定義の装置フィールド SCA の仕様を伝送するだけです。受信アプリケーション・プログラムは、ISC セッション出力で MFS がサポートする SCA または DSCA を使用して、サブシステムに接続された端末装置を正しく操作することができます。

アプリケーション・プログラムで MFS フォーマット設定が重要でない場合、ISC 接続のサブシステムは、MFS を呼び出さずに入力データ・ストリームをプログラムに送信することができます。これを行うためには、送信サブシステムは入力データ・ストリームをプログラムの入出力域と完全に同じになるように構成しなければなりません。この MFS の「バイパス」で、MFS に関連したプロセッサ・オーバーヘッドは縮小される結果となります。この方法で MFS をバイパスすると、送信サブシステムからのデータ・ストリームの伝送が不十分になる可能性があり、また、送信サブシステムではプログラムの入出力域の変更が重要となるので注意してください。

実施の観点からは、MFS がプログラムの入力にすでに使用されている時には、ISC を伴う MFS を使用することが最も容易と考えられます。この場合には、ISC (DPM-Bn) 装置入力形式 (DIF) および DOF を作成するだけです。

## 関連資料

[DFLD ステートメント \(システム・ユーティリティー\)](#)

## MFS を使用しないプログラムへのアクセス

MFS を使用しない既存のアプリケーション・プログラムには 2 つの入力編集選択項目があり、それは ISC 編集と基本編集です。

ISC 編集は、IMS への ISC 入力のデフォルト編集プロセスです。既存のアプリケーション・プログラムでは、入力メッセージの開始点に トランザクション・コードがあり、その後ろにブランクが続いていると考えています。これが真である時には、送信サブシステムでは、データ・ストリームの開始点に トランザクション・コードがあり、その後ろにブランクがあることを確認する必要があります。パスワード・セキュリティは静的定義の端末専用で、データ・ストリームの トランザクション・コードの後にパスワードがあり、FM ヘッダーの SNA 定義 PRN がいないことが確認された場合にしか呼び出すことができません。ISC 編集が使用される時には、IMS に送信される入力データ・ストリームがアプリケーション・プログラムの入出力域と厳密に一致していなければなりません。

ISC 入力の基本編集は、入力が端末から行われているかのように、標準編集を行います。装置およびオペレーター制御文字の編集で使用されます。装置が IMS に直接接続されている場合には、送信サブシステムは、通常基本編集を使用する装置からのデータ・ストリームを単純に IMS に渡します。この状態では、IMS の ISC 基本編集サポートは、送信サブシステムにおけるアプリケーション提供の編集を除去します。文字「BASICEDT」が FM ヘッダーの DPN フィールドにある時には、送信サブシステムは基本編集を選択します。

## メッセージの経路指定

以下のトピックでは、SNA 定義 FM ヘッダー内の経路指定フィールドを紹介し、IMS がこれらのフィールドを使用してメッセージを経路指定する (また、それによって分散処理をサポートする) 方法について説明します。

### メッセージ経路指定の FM ヘッダー

ISC メッセージ経路指定情報は、SNA 定義の機能管理 (FM) ヘッダー内で提供されます。

IMS は、入力時に自動的にこれらの FM ヘッダーの情報を処理し、入力メッセージから取り除きます。次に IMS は、適切な入力編集プロセスにメッセージを渡します。IMS は出力時に、IMS システム定義、セッション・バインド・パラメーター、入力 FM ヘッダー、あるいは MFS によって提供される情報に基づいて、必要な経路指定 FM ヘッダーを自動的に構築します。

分散トランザクション処理についてのこの説明の場合、対象の ATTACH および SCHEDULER FM ヘッダーの一部は、それが帯びる経路指定パラメーターです。4 つのフィールドを使用して、メッセージを適切な制御ブロックに経路指定します。双方のヘッダーにあるこれらの経路指定フィールドは次のものです。

#### DPN (宛先プロセス名)

DPN パラメーターは、セッションと同期的に呼び出される入力プロセスを命名します。IMS では、呼び出すことのできるプロセスは基本編集、ISC 編集、または MFS です。MFS では、入力 DPN 値が MFS メッセージ入力記述子 (MID) 名です。その他のサブシステムでは、名前付きプロセスは (トランザクション・コードなどの) メッセージ宛先になることができます。

### PRN (1 次リソース名)

PRN パラメーターは、付加されたプロセスと関連したリソースを名付けます。このパラメーターは、リモート・サブシステムでのメッセージ処理の結果として戻される応答で現れます。

IMS では、入力 PRN は、端末出力 (IMS メッセージ通信) の LTERM メッセージ・キュー、もしくは入力トランザクション・メッセージ・キューを命名することができます。入力時に PRN が利用不能な場合、IMS でのデフォルトでは、最初のデータ・フィールドに含まれる (あるいは MFS フォーマット設定で任意に挿入された) メッセージ宛先か、または IMS オペレーター・コマンドで設定された事前設定トランザクション・コードを使用する通常の方法が想定されます。

### RDPN (戻り宛先プロセス名)

RDPN パラメーターは、出力メッセージに関連して提示される戻り宛先リソース名を定義します。これは、発信元サブシステムに戻される応答の処理を容易にするために、結果のメッセージ応答の DPN として戻されなければなりません。応答またはメッセージ通信出力が別のセッションで送信された時には、入力 RDPN は廃棄されます。RDPN は、MFS 形式出力の場合にのみ IMS によって送信されます。その中で、RDPN は、MFS メッセージ出力記述子 (MOD) で指定された、オプションのチェーン MID 名です。

### RPRN (戻り 1 次リソース名)

RPRN パラメーターは、出力メッセージに関連して提示される戻り 1 次リソース名を定義します。これは、発信元サブシステムでの応答の処理を容易にするために、メッセージの発信元であるサブシステムへの結果のメッセージ応答として戻されなければなりません。応答はまたメッセージ通信出力が別のセッションで送信された時には、入力 RPRN は廃棄されます。RPRN が MFS MOD で指定された場合には、MFS 形式出力の使用によりオプションで設定することができます。

### 関連資料

#### 555 ページの『機能管理ヘッダー』

SNA では、機能管理 (FM) ヘッダーはリンクに対し送信される要求単位のオプションの部分です。このトピックでは、ISC セッション上の IMS によってサポートされる FM ヘッダーについて説明します。

#### 575 ページの『ATTDPN』

宛先プロセス名 (ATTDPN) パラメーターは、明示的または暗黙的に、ハーフセッションに付加される入力プロセス名を指定します。

#### 577 ページの『ATTPRN』

ATTACH 1 次リソース名 (ATTPRN) パラメーターは、受信サブシステムの入力メッセージの宛先を表します。

#### 577 ページの『ATTRDPN および ATTRPRN』

戻り宛先プロセス名 (ATTRDPN) および戻り 1 次リソース名 (ATTRPRN) パラメーターは、ソース・セッション内の戻りプロセスと戻り 1 次リソースを定義し、ソース・セッション内の応答を戻すための経路指定を可能にするために、結果の応答上でソース・セッションに戻す必要があります。

#### 578 ページの『ATTDQN および ATTDQ』

宛先キュー名 (ATTDQN) パラメーターは、特定のメッセージ・インスタンスの名前を示します。このパラメーターは、出力 MFS 要求時ページ化メッセージでのみ有効です。

## IMS による経路指定パラメーターの使用

フロントエンド切り替え出口ルーチンなしの IMS ISC は、IMS フロントエンド・サブシステムである場合は自動トランザクション経路指定を行いません。

これらのトランザクションは、IMS 内のユーザー作成アプリケーション・プログラム、または ISC メッセージ通信の使用による代替 PCB 出力として送信することができます。後者の場合には、発信元端末オペレーターは、IMS メッセージ通信の最初のデータ・フィールドに必要な ISC 宛先 LTERM 名によって出力経路指定情報を提供します。

出力が MFS によって (メッセージ通信に関連した入力 MID から チェーニングされた MOD の使用によって) フォーマット設定されている場合を除き、ISC の IMS メッセージ経路指定サポートは、メッセージ・テキストの残りの部分をバックエンド・サブシステムに送信する前に、最初のデータ・フィールドを削除します。MFS を使用してメッセージ通信出力のフォーマット設定を行っている場合は、MFS 編集を使用して出力 LTERM 名を削除する必要があります。



さらに、IMS の ISC メッセージ経路指定サポートは、メッセージとともに送信された出力 FM ヘッダー内の RPRN のデフォルトとして、メッセージ通信の発信元端末に関連する LTERM 名を採用します。代替 PCB 出力の場合、RPRN は入出力 PCB 名に設定されます。他方のサブシステムが応答時に RPRN パラメーターとして RPRN パラメーターを戻した場合は、RPRN により、IMS でメッセージ通信の形式でリモート・サブシステムから発信元端末に戻されるメッセージ応答を自動的に経路指定できます。MFS DPM を使用すると、FM ヘッダー宛先を変更し、経路指定パラメーターを戻すことができます。IMS をフロントエンド・サブシステムとして使用して、別のサブシステムにメッセージを経路指定する、あるいは応答を経路指定して発信元端末に戻す場合は、ISC セッションと発信元端末の両方に関連する、IMS システム 定義 TERMINAL マクロ・ステートメントまたは ETO ユーザー記述子で MSGDEL=SYSINFO を指定する必要があります。

## 経路指定の例

次の経路指定の例では、ISC を使用する 2 サブシステム間で実現される一般的な機能と、これらの機能を実現するための経路指定パラメーターの使用法を示します。

例では、次のようになっています。

- ISC 編集別名は ISCE です。
- 図の左方の端末に関連した入出力の LTERM 名は T です。
- 図の右方の端末に関連した入出力の LTERM 名は T2 です。
- 2つのサブシステム間の ISC セッションの LTERM 名は LTISC1 です。

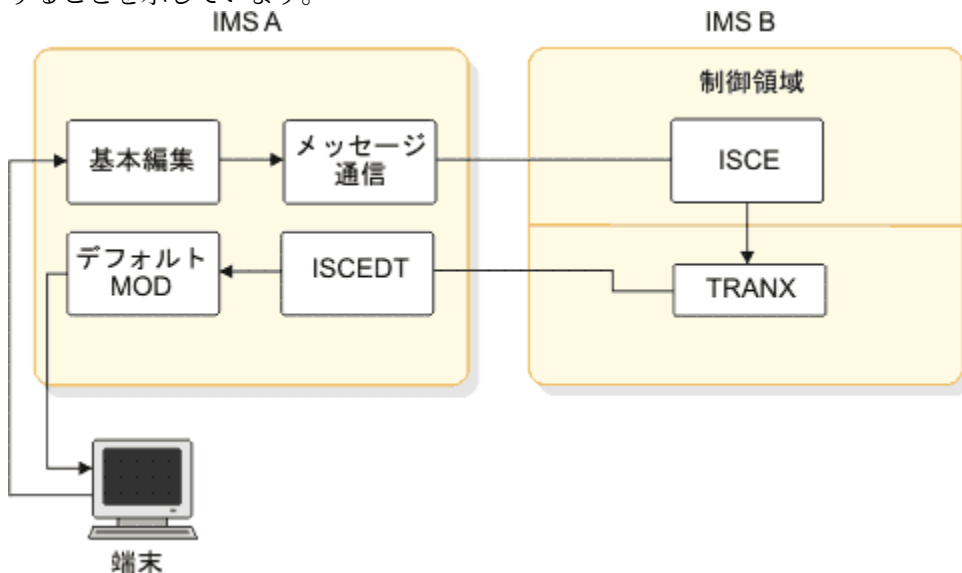
**関連資料:** フロントエンド切り替え出口ルーチンの例については、「IMS V15 出口ルーチン」を参照してください。

### 関連資料

フロントエンド切り替え出口ルーチン (DFSFEJ0) (出口ルーチン)

### 例 1. IMS 間メッセージ通信経路指定

次の図は、IMSA に接続された端末が ISC メッセージ通信を使用して、IMSB で実行される TRANX を入力することを示しています。



**図の注:** 名前 ISCE は、両サブシステムの COMM マクロで EDTNAME=ISCE を指定し、システム定義時に ISC 編集用に定義したものです。

図 60. IMS 間メッセージ通信経路指定の ISC 例

IMSB の MPP は TRANX を処理し、応答はメッセージ通信の発信元である IMSA の端末に戻るよう経路指定されます。出力端末応答は、出力メッセージ通信として表示されます。



1. 端末のタイプは 3270 表示装置を想定しています。この場合は、基本編集はクリアな画面から入力した場合のみ呼び出すことができます。この例では端末オペレーターは次のように入力します。

```
LTISC1 | TRANX | Data...
```

LTISC1 は IMSA と IMSB 間の ISC セッションに関連した IMSA 論理端末名です。

2. 基本編集は端末からの入力データ・ストリームを編集し、メッセージは宛先 LTISC1 で IMSA メッセージ・キューに入れられます。
3. LTISC1 への出力では、IMSA の ISC サポートは次のことを行います。
  - a. データ・ストリームからの宛先 LTERM 名 (LTISC1) をストリップ
  - b. トランザクションの IMSB への送信に必要な FMH を構築
4. IMSB に送信されるデータ・ストリームは次のようになります。

```
FMH: DPN=SCHEDULER
```

```
FMH: DPN=ISCE,PRN=,RDPN=,RPRN=T | TRANX | Data...
```

- a. 出力が別のサブシステムに送信される時に DPN が与えられない場合には、IMS ISC サポートがデフォルトとしてこの値を提供するために DPN=ISCE です。
  - b. PRN= は提供されず、この例では必要ありません。TRANX はユーザー・データの一部です。MFS の使用を通してしか PRN= は提供されません。
  - c. RDPN= は提供されず、この例では必要ありません。MFS の使用を通してしか RDPN= は提供されません。
  - d. RPRN=T は、IMS ISC に取り入れられたメッセージ通信論理のデフォルト機能としての IMSA によって自動的に挿入されます。
5. TRANX データ・ストリームは、FMH で DPN=ISCE を指定したため、IMSB への入力時に ISC 編集 (ISCE) で編集されます。編集後、データ・ストリームはメッセージ・キューに入れられ、次のようになります。

```
TRANX | Data...
```

6. スケジュール時には、MPP はメッセージ・キューから TRANX をリトリブし、トランザクションを処理します。MPP で起点入力端末へ挿入される出力 (ISRT) は次のようになります。

```
Data...
```

この出力は、MPP によって (同一並列セッションの) 入出力 PCB または代替 PCB に挿入 (ISRT) できます。TRANX が応答モードのトランザクションである場合、入出力 PCB または代替応答 PCB を使用しなければなりません。TRANX が非応答トランザクションの場合、代替 PCB を使用できます。出力 LTERM が割り当てられたセッションに応じて、代替 PCB 出力は、同一または異なる並列セッションに送信されます。

7. IMSB から IMSA への出力では、FMH がデータとともに構築され、送信されます。

```
FMH: DPN=SCHEDULER
```

```
FMH: DPN=ISCE,PRN=T,RDPN=,RPRN | Data...
```

出力は、入力を受信されたセッションと同じセッションに送信されます。

- a. 非同期出力が別のシステムに送信される時に入力から、もしくは MFS を通じての DPN が利用可能でない場合には、ISC サポートがデフォルトとしてこの値を提供するために DPN=ISCE が指定されます。
- b. IMSB への入力 FMH で RPRN=T が指定されたため、PRN=T が提供されます。IMS ISC サポートは、出力 FMH PRN フィールドへの入力 FMH RPRN 値を自動的にラップします。

c. RDPN= は必要ではありませんが、MFS DPM で追加することもできます。応答が入力トランザクションと同じセッションに戻されるため、RPRN は提供されません。ただし、MFS を使用して RPRN を設定することもできます。

8. IMSA への入力では、FMH が DPN=ISCE を指定したため、IMSB からの応答は ISC 編集 (ISCE) で編集されます。

PRN が入力 FMH に提供されるために、ISC 編集は IMS 宛先として PRN を使用し、入力データに PRN 値を追加します。ISC 編集後、メッセージは入力メッセージ・キューに入れられ、次のようになります。

T | Data...

IMSA 内の応答を処理するために MFS が使用されている場合には、LTERM 名の挿入を抑制できます。例 6 を参照してください。

9. IMS デフォルト MOD は、端末への出力用データのフォーマット設定に使用されます。端末 T へのデータ表示は次のようになります。

T | Data...

## 例 2. IMS 間アプリケーション経路指定

次の図は、両方の IMS サブシステムでアプリケーション・プログラムを使用した IMS 間経路指定を示しています。

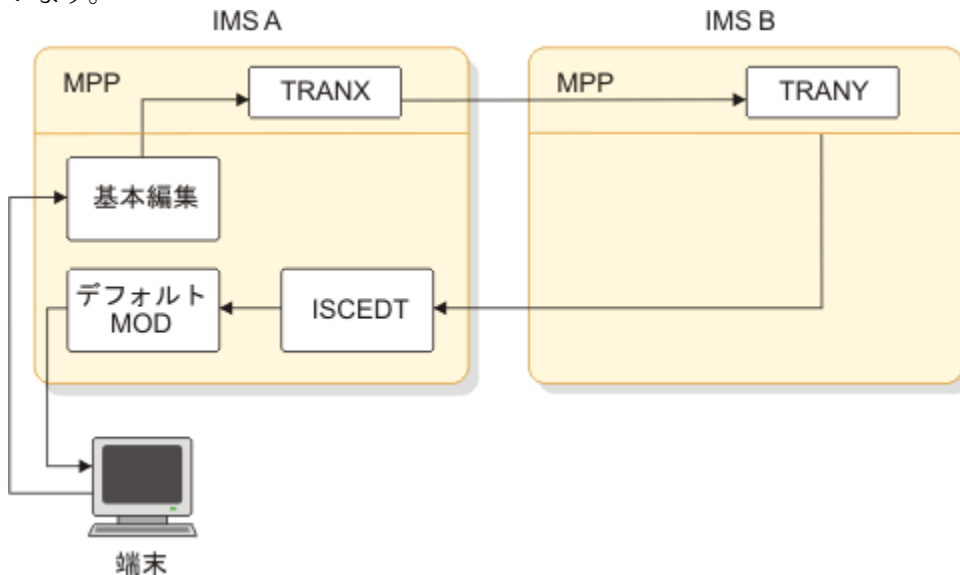


図 61. IMS 間アプリケーション経路指定の ISC 例

端末オペレーターからの入力形式は、端末オペレーターではなく IMSA の TRANX によって提供される経路指定である例 1 の入力形式と異なります。IMSB およびセッションでの処理は、例 1 で説明した処理と同じです。経路指定シナリオでの MFS の相互作用を理解するために、このトピックの後編にある例 6 と例 7 も参照してください。

### 例 3. IMS とその他のサブシステム間のメッセージ通信経路指定

次の図に示されている例で実現される機能は、例 1 の機能と同じです。ただし、バックエンド・サブシステムは CICS かユーザー作成サブシステムです。

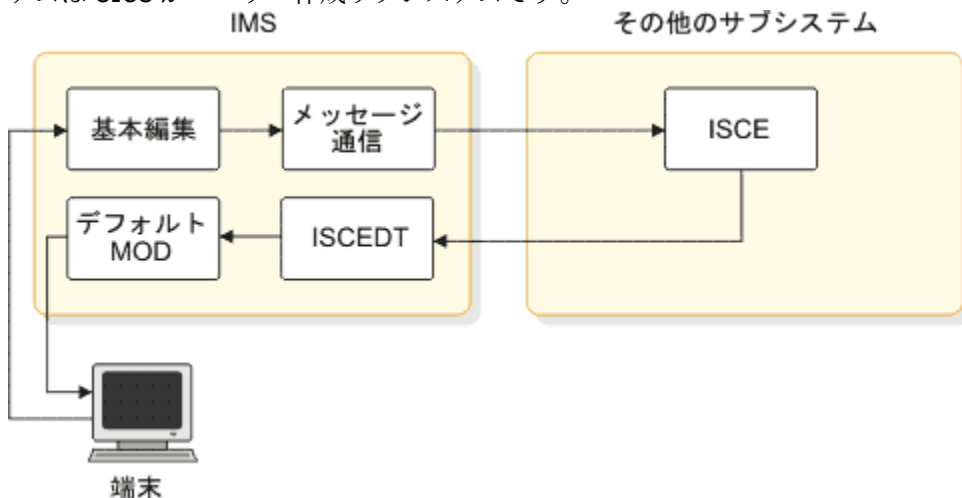


図 62. IMS とその他のサブシステム間のメッセージ通信経路指定の ISC 例

この例の IMS で行われるアクティビティーについての説明は、例 1 の場合と同様です。「その他の」サブシステムが重要であり、それについて以下に説明します。

1. その他のサブシステムに CICS を想定しています。例 1 では、IMS から送信されるデータ・ストリームは次のようになっていました。

```
FMH: DPN=ISCE,PRN=,RDPN=,RPRN=T | Data...
```

データは、ISCE が受信を予期する方法でフォーマット設定されます。トランザクションを入力する IMS 端末オペレーターは、そのフォーマット設定について理解する必要があります。

2. DPN は CICS のトランザクション・コードを示しているために、CICS にはトランザクション・コード ISCE で定義されたユーザー作成 トランザクションがなければなりません。CICS は、項目 1 で説明されているデータ・ストリームの受信後に、ISCE という名前のトランザクションを呼び出します。これは、ISC 環境用に特別に作成する必要があります。このトランザクションは提供された FMH 値を検査し、RETRIEVE インターフェース を使用して、ユーザー・データを獲得し、TRANX を処理しなければなりません。

CICS が最大 4 文字のトランザクション・コードをサポートしているため、ISC 編集には別名 ISCE が付けられています。

3. CICS のトランザクションからの出力は、例 1 で IMSB が IMSA に送り返した出力のようではありません。すなわち、

```
FMH: DPN=ISCE,PRN=T,RDPN=,RPRN | Data...
```

(IMS ISC サポートが FMH を構築する例 1 とは対照的に) CICS トランザクション ISCE は、出力 FMH の構築に使用するために、CICS を実際にフォーマット設定します。この例では、CICS トランザクションは CICS に DPN=ISCE および PRN=T を提供して適切な FMH を構築できるようにする必要があります。

4. IMS が CICS からこの出力データ・ストリームを受信する時には、すべてのアクティビティーは例 1 と同様です。

CICS 機能および、それとシステムで実現される機能との関連についての前述の説明を理解することにより、ユーザーは、ユーザー作成サブシステムでのこの例の機能を実現することができます。

## 例 4. IMS 端末間メッセージ通信経路指定

以下の図は、IMSB に接続された 端末 T2 へのメッセージ通信の結果として IMSA に接続された 端末 T を表示しています。

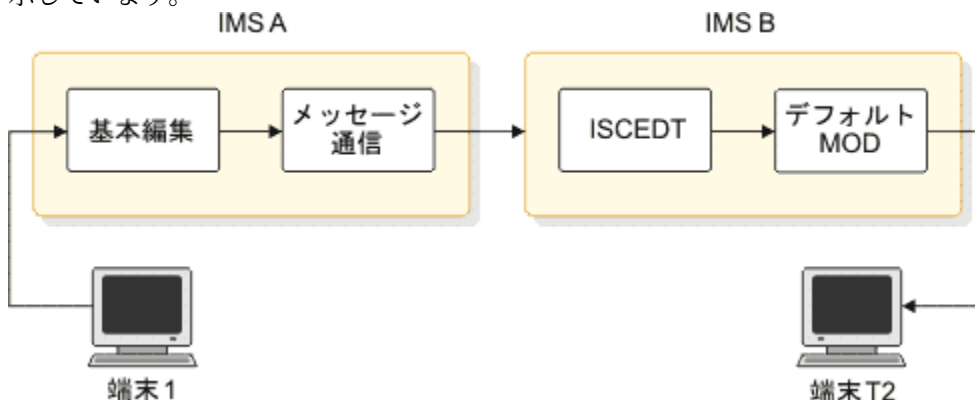


図 63. IMS 端末間メッセージ通信経路指定の ISC 例

1. 端末 T は次のデータ・ストリームを入力します。

```
LTISC1 | T2 | Data...
```

LTISC1 は、IMSA が IMSB の ISC セッションと関連している LTERM 名です。端末オペレーター (T) は、IMSB に接続された宛先端末 (T2) の LTERM 名と同様に、この ISC LTERM 名について理解してなければなりません。

2. IMSB に ISC セッションのメッセージの残りの部分を送信する前に、ISC メッセージ通信サポートは LTERM 名 (LTISC1) を除去します。ISC セッションで送信されるデータ・ストリームは次のようになります。

```
FMH: DPN=SCHEDULER
```

```
FMH: DPN=ISCE, PRN=, RDPN=, RPRN=T | T2 | Data...
```

- a. 出力が別のサブシステムに送信される時に IMS が DPN を提供しない場合には、IMS ISC サポートがデフォルトとしてこの値を提供するため、DPN=ISCE が指定されます。
  - b. 宛先端末 T2 がデータ・ストリームの一部であるために PRN= は必要ありませんが、MFS で提供または変更することもできます。
  - c. RDPN= は提供されず、この例では必要もありません。この値は MFS によって提供されることがあります。
  - d. 例 1 と同様、RPRN=T は、IMS ISC に取り入れられたメッセージ通信論理のデフォルト機能として IMSA によって自動的に挿入されます。応答が戻らないために、このパラメーターは IMSB によって廃棄されます。IMSB で MFS を使用して、端末 T2 のオペレーターがこのパラメーターを利用することができます。
3. IMSB のメッセージ・キューに入れられる前に、データ・ストリームは ISC 編集で編集されます。IMSB では、(入力 FMH で PRN が利用可能でないために) ISC 編集はデータを検査して宛先を判別します。T2 が宛先として検出され、入力メッセージは宛先 T2 を指定してメッセージ・キューに入れられます。
  4. IMSB から端末 T2 への出力では、この時点のデータ・ストリームは次のようになります。

```
T2 | Data...
```

5. T2 が必要な MFS 装置である場合には、デフォルト・システム MOD が出力用に使用されます。

## 例 5. IMS 端末と他の端末との間のメッセージ通信経路指定

次の図は「例 4」によく似ていますが、バックエンド・サブシステムは CICS サブシステムかユーザー作成サブシステムです。

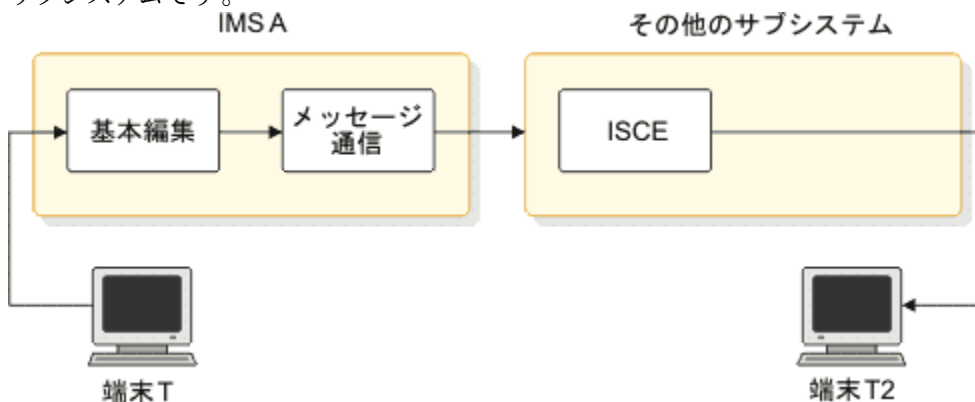


図 64. IMS 端末と他の端末との間のメッセージ通信経路指定の ISC 例

この例の IMS で起きるアクティビティーについての説明は、例 1 の説明とまったく同じです。その他のサブシステムについての事項を説明します。

1. その他のサブシステムに CICS を想定しています。例 1 では、IMS から送信されるデータ・ストリームは次のようになっていました。

```
FMH: DPN=ISCE,PRN=,RDPN=,RPRN=T | Data...
```

2. 例 3 と同様に、CICS には ISCE として定義されたトランザクション・コードがなければなりません。

データは ISCE が受信するための方式でフォーマット設定されています。トランザクションを入力する IMS 端末オペレーターは、そのフォーマット設定について理解する必要があります。データ・ストリームの受信では、CICS は ISC 環境用に特に書き込まれる ISCE というトランザクションを呼び出さなければなりません。このトランザクションは入力 FMH のデータを使用し、RETRIEVE インターフェースを通して入力データを入手します。入力 FMH の PRN フィールドが IMSA によって T2 と等しくなるように初期設定されていないため、CICS アプリケーションは、一時データ・キューから端末 T2 へのメッセージ通信を開始するか、もしくは 1 次リソースとして T2 を命名する新規トランザクションを開始しなければなりません。

この例では、出力端末 T2 は入力データ・ストリームで識別される必要があります。IMSA が T2 への入力 FMH 内の PRN フィールドを初期設定している場合には、CICS はトランザクション ISCE を 1 次リソースとして T2 に付加し、端末 T2 への SEND は直接実行されます。

3. IMSA で PRN フィールドを設定するためには MFS が必要です。

## 例 6. MFS での IMS 間メッセージ通信経路指定

次の図で、端末 T はバックエンド IMS サブシステムで処理する必要があるトランザクションを入力します。

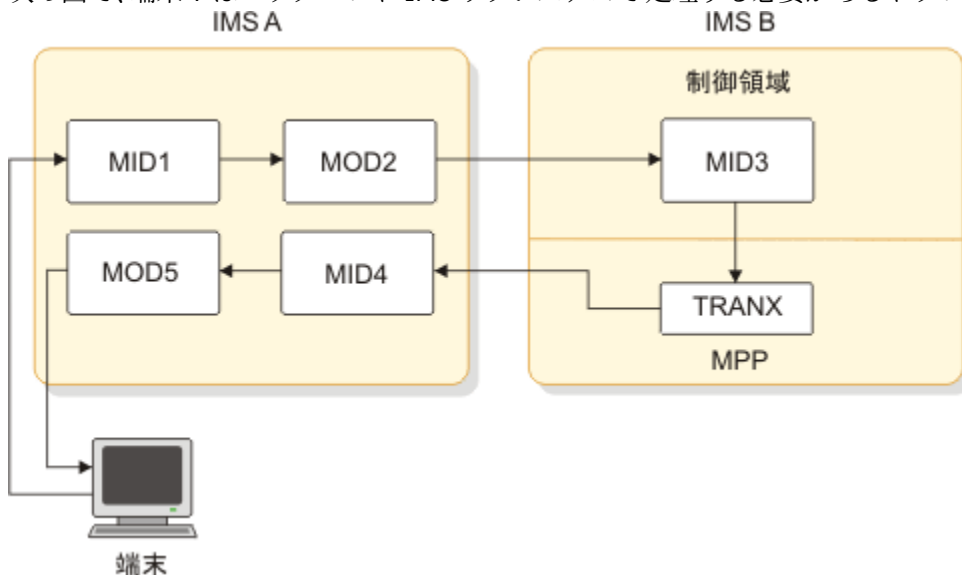


図 65. MFS での IMS 間メッセージ通信経路指定の ISC 例

バックエンド・サブシステムからの応答は経路指定され、トランザクション (T) を入力した元の端末に戻されます。

1. 端末 T はフォーマット設定された画面から入力します。端末が必要とするのは、データの入だけです。
2. MID1 による編集後、メッセージはメッセージ・キューに入れられ、次のようになります。

```
LTISC1 | TRANX | Data...
```

値 LTISC1 および値 TRANX は、MFS によってデータに追加されます。

3. IMSB に送信されるデータ・ストリームを編集するために使用される MOD2 は、MID1 から IMSA 内の別の MID (MID4) までチェーニングされます。IMSB に送られるデータ・ストリームは以下のようになります。

```
FMH: DPN=SCHEDULER
```

```
FMH: DPN=MID3, PRN=, RDPN=MID4, RPRN=T
```

```
TRANX | Data...
```

- a. DPN=MID3 は、MOD2 によって、あるいは MOD2 に関連した DOF のリテラルから提供されます。
  - b. TRANX がすでにデータ・ストリームにあるため、PRN= は必要ありません。MOD2 またはそれに関連した DOF から提供されることがあります。
  - c. MID4 が MOD2 にチェーニングされているため、RDPN=MID4 が指定されます。
  - d. RPRN=T は、メッセージ通信のデフォルト機能として IMSA によって再び自動的に挿入され、MFS によってオーバーライドされることはありません。
  - e. LTISC1 は、MFS MOD によって出力メッセージから再び取り除かれます。
4. IMSA からのデータ・ストリームの受信では、FMH の DPN が MID3 を指定しているために、IMSB は MID3 を使用してデータを編集します。
  5. アプリケーションは、次のものを受信して処理します。

```
TRANX | Data...
```

6. アプリケーション・プログラムからの出力は、IMSB によって次の形式で IMSA に送信されます。

```
FMH: DPN=SCHEDULER
```

```
FMH: DPN=MID4,PRN=T,RDPN=,RPRN= | Data...
```

7. DPN=MID4 および PRN=T は、IMS によって元の FMH の RDPN 値および RPRN 値から自動的に循環されます。

8. このデータ・ストリームの受信では、IMSA の MID4 は入力を編集し、端末 T への最終出力のために次の形式でメッセージをキューに入れます。

```
Data...
```

PRN=T が提供されているために、IMS はそれを宛先として使用します。MFS が入力応答のフォーマット設定にも使用されるために、PRN はデータに追加されません。

9. MOD5 が MID4 にチェーニングされているために、MOD5 は端末 T への出力のフォーマット設定に使用されます。

この例に関して、いくつか注目しなければならないことがあります。

- IMSB の MID3 は MOD にチェーニングされません。このため、IMSB のアプリケーション・プログラムが参照する入出力 PCB は、出力に使用する MOD 名を含んでいません。MID3 が MOD にチェーニングしている必要がある場合には、アプリケーション ISRT 呼び出しは、出力で MFS 編集を否定するためのブランク MOD 名を持っていなければなりません。
- IMSB のアプリケーションは、出力の MOD 名にチェーニングされていません。この場合には、追加の MFS 形式設計が必要です。
- IMSB の MPP は既存のプログラムであることがあり、また、IMSB に接続された端末からのトランザクションを取り扱います。MFS を使用することで、ISC セッションの使用をアプリケーションにとっては透過的にすることができます。
- 元の入出力端末は応答モードのままではありません。

### 例 7. MFS での IMS 間アプリケーション経路指定

次の図は、前述の例 (IMSB の入力の編集に ISC 編集が使用される事例、または IMSB の入力の編集に MFS DPM-Bn が使用される事例) を理解したうえで、2 つの文脈において理解できます。

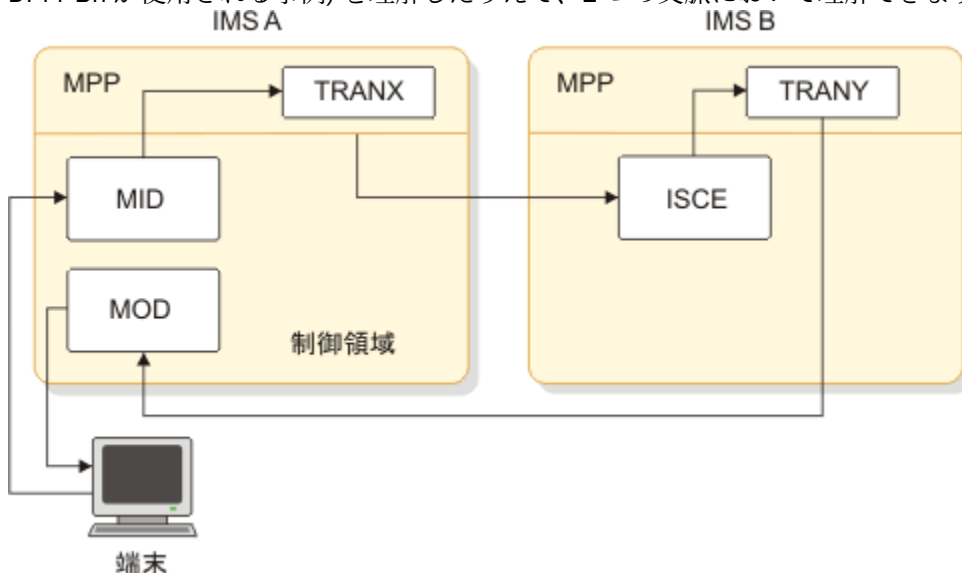


図 66. MFS での IMS 間アプリケーション経路指定の ISC 例

例 7 は次のいくつかの点を強調します。



- 端末 T は、フロントエンド・サブシステムによって完全に操作されます。このため、この例では、端末 T は IMSA での応答モードのままです。
- IMSA から送信されるデータは、代替 PCB によって挿入 (ISRT) され、応答モード・トランザクションに非同期です。
- IMSA は複数の IMS サブシステムに接続され、入力トランザクションに基づいて ISC 通信を適切なバックエンド IMSB に経路指定します。
- IMSA の MPP は、代替 PCB への ISRT を通しての適切なバックエンド・サブシステムなどの、必要なメッセージ経路指定情報を提供できます。
- MFS は、一方または両方のサブシステムでのデータ・ストリームのフォーマット設定および経路指定の提供に使用することができます。

## MSC を介した ISC LTERM へのメッセージの経路指定

ISC ハーフセッションへの MSC リンクを介してメッセージを経路指定することができます。

### このタスクについて

次の図に示すようなセットアップを使用しているとします。IMS A および IMS B は MSC リンクで接続されています。SYS C は ISC リンクによって IMS B に接続されています。IMS A と SYS C の間の入力メッセージおよび出力応答を経路指定する必要がある場合には、ISC 経路指定パラメーターをメッセージの IMS 接頭部においてください。MSC リンク処理機能がこれらの経路指定パラメーターを保持します。

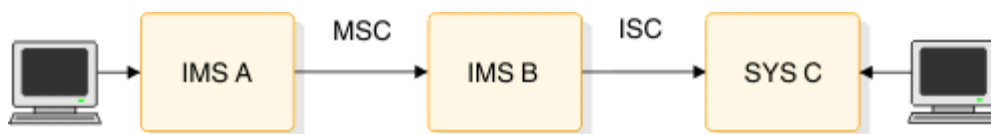


図 67. ISC LTERM への MSC の経路指定

## IMS 間 ISC セッションの考慮事項

静的端末を使用する 2 つの IMS サブシステム間の ISC セッションを構成する場合は、その前に以下の情報を考慮してください。

### 1 つの IMS サブシステム内での ISC と MSC VTAM の共存

IMS システム定義時に、一方の IMS サブシステムの TERMINAL、MSPLINK マクロ・ステートメント、動的 CREATE MSPLINK コマンド、または ETO ログオン記述子を、他方の IMS システムの COMM (ACB 名) マクロで定義されている名前と同じ名前前で定義する必要があります。

IMS システム定義では、動的 CREATE MSPLINK コマンド (またはステージ 1 システム定義 MSPLINK マクロ) と、TERMINAL マクロまたは ETO ログオン記述子の両方で同じリモート・サブシステム名を指示できます。ただし、セッション修飾子情報 (SUBPOOL マクロで使用される名前と、動的 CREATE MSPLINK コマンドまたはステージ 1 システム定義 MSPLINK マクロのパートナー ID パラメーター) は、固有でなければなりません。また、MSC と IMS のセッションを定義し、2 つの IMS サブシステム間で同時にアクティブにすることができます。2 つの IMS サブシステム間で同時にアクティブにすることができる ISC 並列セッションの数は、その 2 つの IMS サブシステムの TERMINAL マクロ・ステートメントの SESSION= パラメーターに定義されている小さい方の値です。(SESSION= パラメーターは静的定義端末専用です。) 2 つの IMS サブシステム間で同時にアクティブにすることができる MSC 並列セッションの数は、その 2 つの IMS サブシステムで動的 CREATE MSPLINK コマンド (またはステージ 1 システム定義 MSPLINK マクロ) の SESSION= パラメーターに定義されている小さい方の値です。同時にアクティブにすることができる MSC 並列セッションと ISC 並列セッションの総数は、上記の 2 つの小さい方の数値の合計です。

### 単一セッションおよび並列セッションの定義

IMS システム定義中に、IMS サブシステム間の ISC セッションを定義する TERMINAL マクロ・ステートメントまたは ETO ログオン記述子を、単一セッションまたは並列セッションのどちらかとして定義する必要

があります。ただし、両方のサブシステムを並列セッションに定義する場合には、現在アクティブのセッションの数、もしくは SUBPOOL マクロ・ステートメントの数を同じにする必要はありません。

ETO がない場合には、4,095 並列セッションまで定義することができます。ETO 付きの場合には、ISC 並列セッションの数は、主としてプロセッサ容量および制御域として使用される領域を除く仮想記憶域の容量で制限されます。

## 互換性のあるバッファ・サイズの確認

ある 1 つの IMS サブシステムの TERMINAL マクロ・ステートメントまたは ETO ログオン記述子で定義される VTAM 出力バッファ・サイズは、他のサブシステムの COMM マクロ・ステートメントで定義される"受信"バッファ・サイズとの互換性が必要です。

## ISC 経由 IMS のリモート制御

IMS ISC メッセージ通信サポートまたはユーザー作成のアプリケーションを使用すると、ある 1 つの IMS サブシステム内の端末オペレーターが別の IMS サブシステムの操作を効果的に制御することができます。IMS 自動化操作プログラム・インターフェース (AOI) 機能は、どちらか一方の IMS サブシステム の制御における補助機能としても使用できます。

静的に定義された端末の場合、リモート IMS サブシステム内で ISC セッション へのコマンドまたはトランザクションを許可し、ISC セッションでのデータの送信を許可されたローカル IMS サブシステム内のすべての端末オペレーターおよびアプリケーションがリモート・サブシステムにアクセスできるようにすることができます。そのため、パスワード・セキュリティを使用して、リモート・サブシステムへのアクセスは、IMS サブシステム内のマスター端末のように特別に許可された個々のオペレーター とアプリケーションだけに限定してください。

ISC セッションでのすべてのコマンドを両方のサブシステムの 自動化操作プログラム・インターフェース間に出すか、バックエンド・サブシステムの 単一簡易操作インターフェースに出すよう指定することによって、追加のセキュリティ を実行できます。

また、IMS システム定義時に、タイプ 6.1 の論理装置 を IMS マスター端末として定義することはできず、IMS /ASSIGN コマンドを使用してマスター端末として割り当てる こともできません。

## IMS 間の会話モードの制限事項

2 つの IMS サブシステム 間の会話モードは、サポートされていません。1 つのハーフセッションだけが会話型モードにある場合、ISC を使用して接続された IMS サブシステム 間の会話は、予測不能なセッション・プロトコルと 会話型トランザクションの入力となるためです。また、両方のハーフセッションが会話モードにある場合には、会話は終了します。

## バックエンド IMS への経路指定トランザクション

IMS ISC サポートは、以下によるバックエンド IMS サブシステムへの トランザクションの経路指定を可能にします。

- ISC メッセージ通信を使用する端末オペレーター
- フロントエンド IMS サブシステム内で代替 PCB 挿入を使用するアプリケーション・プログラム

どちらの場合でも、ISC サポートによるデフォルト・アクションは、結果としてのトランザクション応答をメッセージ通信の形式で、発信元端末オペレーターまたはトランザクションに経路指定し直すことです。これらの条件の下では、メッセージ通信の応答に必要な応答もリカバリー可能であるため、バックエンド IMS サブシステム内でアクセスされるトランザクション も、通常はリカバリー可能として定義してください。

リカバリー不能トランザクションは、アクセス可能ですが、MFS DPM-Bn を使用して、ISC メッセージ経路指定パラメーターで設定されたデフォルト宛先を、発信元端末オペレーターに関連付けられた LTERM からフロントエンド IMS 内のリカバリー不能トランザクションに変更する必要があります。これにより、リカバリー不能応答メッセージを送信するときに生ずる プロトコル・エラーを防止します。MFS DPM により宛先をフロントエンド IMS 内では、バックエンド IMS への トランザクションで送信したデフォルト RPRN パラメーターを修正または削除、あるいはバックエンド IMS 内では、フロントエンド IMS への応答

で送信した折り返し PRN パラメーターを修正または削除するように変更することができます。このリカバリー不能トランザクションは、さらに修正可能な代替 PCB を挿入することにより、応答を適切な端末オペレーターに転送することができます。

フロントエンド切り替え出口ルーチンによって提供される、フロントエンド/バックエンド・システム使用のための特別なサポートもあります。

## サブシステム間のメッセージの送信

IMS フロントエンドからの通信は、常に非同期です。したがって、メッセージは、ATTACH ヘッダーと SCHEDULER FM ヘッダーの両方を付けて送受信することが必要です。例外は、システム・メッセージは ATTACH FM ヘッダーだけを付けて送信されるということです。

## IMS 間セッションに関するプロトコル制約事項

ISC セッションが 2 つの IMS サブシステム間にあるとき、あるタイプのプロトコルは、そのサブシステム間で送信されません。ここでは、それらのプロトコルについて記載します。IMS 間セッションを設計している場合には、499 ページの『[第 30 章 VTAM 接続用の ISC プロトコル](#)』に記載されているこれらのプロトコルおよびヘッダーについての情報は読み飛ばしてかまいません。

- データ・フロー制御コマンドの BID、RTR、および RSHUT は、IMS のサブシステム間で送信されません。
- リセット付加プロセス (RAP) の FM ヘッダーは IMS サブシステム間では送信されません。
- キュー・モデル (QMODEL) ヘッダーは、IMS サブシステム間で送信されません。これにより、IMS 出力要求時ページングおよび関連するすべての入力 QMODEL ページング要求は除外されます。さらに、ATTACH FM ヘッダーの ATTDQN パラメーターと SCHEDULER FM ヘッダーの SCDDQN パラメーターは IMS 間のセッションではサポートされていません。
- 以下のエラー・コードは、2 つの IMS サブシステム間で送信されません。
  - LUSTATUS コミット、機能打ち切り X'0864' および X'0866'
  - X'0813'、X'0846' を除くすべての送信側 ERP センス・コード
  - 545 ページの『[選択受信側 ERP](#)』にリストされたものを除く選択された受信側の ERP センス・コード

### 関連資料

559 ページの『[システム・メッセージ・プロセス \(SYSMSG\) および関連 FM ヘッダー](#)』システム・メッセージ・プロセスは、ATTACH FM ヘッダーのプロセス名で指示されます。

### [CREATE MSPLINK コマンド \(コマンド\)](#)

[フロントエンド切り替え出口ルーチン \(DFSFEBJ0\) \(出口ルーチン\)](#)

## IMS への ISC ノードの静的定義

IMS のシステム定義で ISC ノードとして静的に定義されているサブシステムは、IMS では端末に、そして、IMS アプリケーション・プログラムでは 1 つ以上の論理端末 (LTERM) になります。このように、1 つの ISC ノードは、他の VTAM 端末定義にも使用できるシステム定義 マクロで定義されます。

以下の表は、ISC ノードを定義するマクロ・ステートメントを示しています。

表 81. ISC ノード定義用のマクロ・ステートメント

マクロ・ステートメント	用途
COMM マクロ	VTAM に対して IMS サブシステムの名前を付ける。ISC 編集に名前を付け、どんなバッファ・サイズでも受け取れる (RECAN) ことを定義します。
TYPE マクロ	ISC ノード (UNITYTYPE=LUTYPE6) として装置タイプを識別する。

表 81. ISC ノード定義用のマクロ・ステートメント (続き)

マクロ・ステートメント	用途
TERMINAL マクロ	IMS への他の ISC ノードを識別する。また、出力および高速機能バッファのサイズを指定し、入力セグメント・サイズ、および処理オプションを限定する。2つの ISC ノード間のセッション数を設定する。コンポーネントの特性を定義する。
VTAMPOOL マクロ	並列セッションで使用される ISC LTERM ユーザー (サブプール) の定義を開始する。
SUBPOOL マクロ	VTAMPOOL 内での LTERM のセットを定義する。
NAME マクロ	単一セッションのある端末に関連する LTERM あるいは、並列セッションのあるユーザー (サブプール) に関連する LTERM に名前を付ける。

このトピックは、2つのシステム間の ISC セッションを定義するのに重要なマクロについて、また、あるシステム定義オプションを他のオプションの代わりに選択する上での関連事項について詳しく説明しています。

以下の図は、New York データ・センター内の NYIMS と、San Francisco データ・センター内の SFIMS の2つの IMS システムを示しています。NYIMS と SFIMS は、それぞれがお互いに対して ISC ノードとして定義されています。以下の段落で、IMS システム定義でのこれらの IMS システムの定義方法を詳細に説明します。理解しやすくするために、この説明では、IMS SFIMS を IMS NYIMS に対して定義する場合に焦点を合わせます。IMS NYIMS を IMS SFIMS に対して定義する場合にも、同様のシステム定義が必要となります。ここでは、COMM、TERMINAL、および NAME マクロ・ステートメントの関連パラメーターについてだけ説明します。

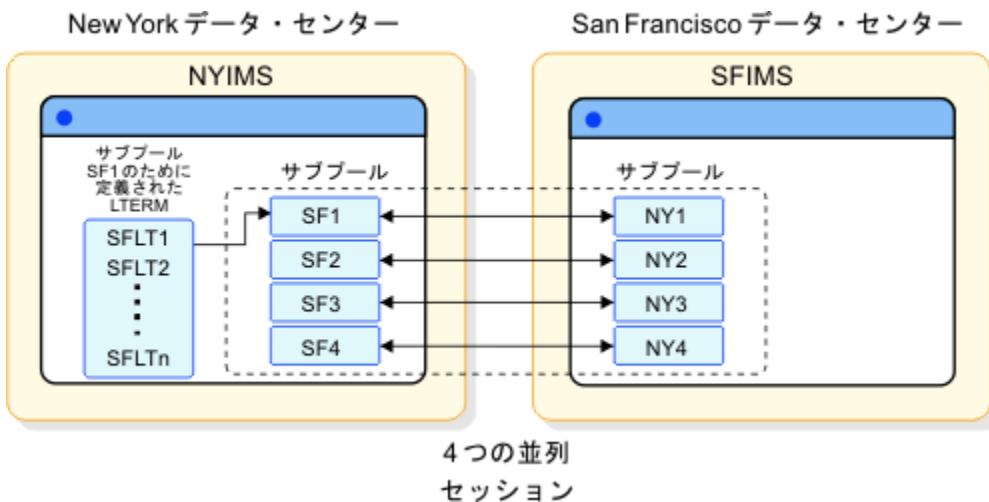


図 68. ISC ノードとして、お互いに定義された2つの IMS システム

NYIMS システム定義の COMM マクロ上でパラメーター APPLID=NYIMS を指定すると、そのサブシステムは、ノード名 NYIMS で VTAM、そのサブシステム自体、およびそのサブシステムへのアクセスが必要な他のすべてのサブシステムによって認識されます。San Francisco IMS SFIMS についても同様のシステム定義が必要です。

NYIMS システム定義の TERMINAL マクロ・ステートメント上の NAME=nodename は、SFIMS システムのノード名を表示するようにコード化されています (NAME=SFIMS)。SFIMS が XRF 複合システムである場合、ノード名は、SFIMS システムに関連した USERVAR または MNPS ACB であることが必要です。

SESSION= キーワードは、単一セッションではなく並列セッションに関連しています。(SESSION= キーワードは、静的定義の端末装置のみに適用されます。ETO 環境には適用されません。) SESSION= は、NYIMS から SFIMS に可能な並列セッションの最大数を指定します。1つの "並列" セッションを指定することは可能です。単一セッションと1つの並列セッションは異なるものです。

SESSION= を 1 より大としてコーディングすると、NYIMS は、SESSION= パラメーターで 指定された数ま で、SFIMS との複数並行セッションを保持することができます。上の図では、2つの IMS システム間に 4 つのパスがあります。これはそれぞれ、並列セッションです。NYIMS で 4 つの並列セッションを可能にするには、SESSION=4 と指定します。

並列セッション・サポートの主な利点は、IMS /OPNDST コマンド・パラメーターでそのセッションに LTERM を動的に割り振ることができることです。

各並列セッションは、SUBPOOL マクロ・ステートメントを使用して NYIMS に対して識別される必要があります。例のように、同時に 4 つのアクティブ並列セッションを可能にするには、4 ユーザー (サブプー ル) を NYIMS システム定義時に、次のように定義する必要があります。

```
VTAMPOOL
SUBPOOL  NAME=SF1
SUBPOOL  NAME=SF2
SUBPOOL  NAME=SF3
SUBPOOL  NAME=SF4
```

VTAMPOOL マクロ・ステートメントは、ISC サブプールの 定義を開始し、並列セッションを使用する場 合には必ずコード化する必要があります。VTAMPOOL マクロ・ステートメントには、オペランドがありません。

492 ページの図 68 の SFLT1、SFLT2、SFLT3、... SFLTn は、最初の SUBPOOL、SF1 に定義してある LTERM の名前を表しています。特定のサブプールの LTERM の定義は、NAME マクロで指定します。複数の LTERM を 1 つのサブプールに定義すると、いくつかの利点があります。最初の利点は、特定のサブプー ルに定義された 1 つの LTERM または複数の LTERM を (/ASSIGN コマンドを使用して) 他の任意のサブプー ルに再割り当てできることです。これにより、ISC のネットワークを柔軟に操作することができます。

2 番目の利点は、TERMINAL マクロ・ステートメントで COMPTn= キーワードを使用できることです。この キーワードは、指定された LTERM にある特性を定義または割り振りするのに使用できます。COMPT= と ICOMPT= は NAME マクロ・ステートメントのキーワードで、コンポーネント (COMPTn) と LTERM の関連 付けに使用します。

NYIMS に対する SFIMS の定義は、以下のコードのようになります。

```
TYPE      UNITYPE=LUTYPE6
TERMINAL  NAME=SFIMS
          SESSION=4

VTAMPOOL
SUBPOOL  NAME=SF1
          NAME SF1LT1
          NAME SF1LT2
          :
          NAME SF1LTN
SUBPOOL  NAME=SF2
          :
SUBPOOL  NAME=SF3
          :
SUBPOOL  NAME=SF4
          :
```

## 関連概念

473 ページの『[LTERM ユーザー \(サブプール\) とコンポーネント](#)』

IMS ユーザー・ブロック は、IMS システム定義時に SUBPOOL マクロによって定義される、もしくは ETO ユーザー記述子から動的に作成される IMS 論理端末 (LTERM) のセットです。

## 関連資料

[IMS 環境で使用されるマクロ \(システム定義\)](#)



## パラメーター選択: システム設計の考慮事項

以下のトピックでは、COMM、TERMINAL、NAME、およびSUBPOOL マクロ・ステートメントのパラメーターについて、詳しく説明します。

### COMM マクロ・ステートメント

COMM マクロ・ステートメント上のキー・パラメーターは、APPLID、RECANY、および EDTNAME です。

APPLID パラメーターで IMS に VTAM ACB 名を定義します。APPLID= で指定する名前は、VTAM APPL 定義ステートメントの ACBNAME= パラメーターで指定する名前と同じでなければなりません。ACBNAME= パラメーターが指定されていない場合、APPLID= 名前は、IMS を VTAM に定義される時に使用される APPL 定義ステートメントの名前と同じでなければなりません。他のサブシステムもまた IMS である場合は、APPLID= で指定する名前は、その IMS の TERMINAL マクロ・ステートメントの NAME= のキーワード・パラメーターで提供されたものと同じである必要があります。

COMM マクロ・ステートメントのアプリケーション ID パラメーターを定義せず、代わりに IMS 制御領域の開始時に EXEC ステートメントのジョブ・ステップ名を使用する場合、VTAM は、上記の段落での説明のように、その名前を ACBNAME= パラメーターまたは VTAM APPL 定義ステートメントの名前のいずれかに一致させようとしています。

COMM マクロで受信バッファ・サイズ (RECANY) を指定する場合、ISC セッションでは 28 バイトのオーバーヘッドが必要であることを覚えておいてください。したがって、NYIMS システム定義で指定される RECANY バッファ・サイズは、NYIMS に対する SFIMS システム定義の TERMINAL マクロの OUTBUF= で指定されるサイズよりも少なくとも 28 バイト大きくなければなりません。SFIMS の定義の場合も、同様の考慮事項が適用されます。

他の要素も RECANY バッファのサイズに影響します。指定された RECANY バッファ・サイズはすべての VTAM 装置に適用されるので、RECANY に指定されるサイズは、他の VTAM 装置の要求のために、ISC が必要とするサイズよりも大きくなる場合があります。さらに、ISC RECANY のサイズを決定するには両システムの最大データ・レコード・サイズに FM ヘッダー・サイズを追加する必要があります。

EDTNAME パラメーターは、IMS システムの ISCEDT に使用できる別名を指定するものです。このシステム定義では、デフォルト名の ISCEDT が使用されているため、EDTNAME= パラメーターの指定は必要ありません。

#### 関連資料

[COMM マクロ \(システム定義\)](#)

### NAME マクロ

NAME マクロで ISC セッションを定義するためのキー・システム定義オプションは、COMPT、OUTPUT、EDIT、および ICOMPT です。

COMPT= と ICOMPT= は、定義されている LTERM と関連する入出力の各コンポーネントをそれぞれ指定します。このシステム定義の例では、COMPT=1 および COMPT=2 と定義されたコンポーネントを使用しています。

OUTPUT= は、ISC セッション上で使用しないでください。

**推奨事項:** セッションへの ISC 出力の変換が必要ない場合には、ULC を指定してください。セッション出力が、大文字または小文字のデータ、あるいは 2 進データでもかまわない場合には、ULC を使用してください。

### SUBPOOL マクロ

SUBPOOL マクロは、システムに対して ISC サブプールを指名します。

SUBPOOL マクロのキー・システム定義オプションは、次のとおりです。

- NAME
- MSGDEL

NAME パラメーターは、ISC サブプールを指名します。

この例では、MSGDEL の指定はデフォルトの SYSINFO です。SUBPOOL マクロで MSGDEL を指定する場合、TERMINAL マクロの MSGDEL 指定と一致していることが必要です。

### 関連概念

473 ページの『LTERM ユーザー (サブプール) とコンポーネント』

IMS ユーザー・ブロックは、IMS システム定義時に SUBPOOL マクロによって定義される、もしくは ETO ユーザー記述子から動的に作成される IMS 論理端末 (LTERM) のセットです。

## TERMINAL マクロ

TERMINAL マクロのいくつかのシステム定義キーワード・パラメーターは、ISC セッションを定義するために重要です。

TERMINAL マクロで ISC セッションを定義するための基本システム定義キーワード・パラメーターは、次のとおりです。

- OUTBUF
- MSGDEL
- COMPT1
- COMPT2
- COMPT3
- COMPT4

指定の必要がある追加パラメーターには、以下のものがあります。

- SINGLE または MULT
- VLVB または DPM-B1...DPM-B15

### OUTBUF= キーワード・パラメーターの指定

IMS は、ISC セッションの開始時にバッファー・サイズのネゴシエーションを行わず、RU 短縮形もサポートしません。IMS は、バインドに (OUTBUF=) と受信サイズ (RECANY=) を設定します。

バインドの応答があった場合、IMS は以下を検証します。

- (OUTBUF=) で設定した送信サイズが他のサブシステムによって減らされていないか。
- (RECANY=) で設定した受信サイズが他のサブシステムの (OUTBUF=) で設定した送信サイズを超えていないかどうか。

決められたバインドを受信する場合、IMS は、以下を実行します。

- 他のサブシステムの送信 (OUTBUF=) サイズが、IMS の受信 (RECANY=) サイズを超えていないか検証する。
- バインド応答に (OUTBUF=) で送信サイズを挿入する。

OUTBUF で指定するサイズは、最大出力セグメントのサイズに、メッセージ・ヘッダー用のオーバーヘッドを加えたものを組み込めるだけの十分な大きさにする必要があります。SFIMS システムの設定でも、同様の考慮が必要です。

### MSGDEL= キーワード・パラメーターの指定

SYSINFO や NONIOPCB の選択の効果については、473 ページの『LTERM ユーザー (サブプール) とコンポーネント』に説明してあります。そのトピックの例では、TERMINAL と SUBPOOL の両マクロに対して SYSINFO のデフォルトが使用されています。

### COMPTn= キーワード・パラメーターの指定



キーワード・パラメーター COMPT1= から COMPT4= を使用して、TERMINAL マクロでコンポーネントを 4 つまで、1 つの ISC ノードに設定することができます。COMPTn= は、LTERM に特性を割り当てる際に柔軟性を与えます。システム定義の例では、コンポーネント 1 は、SINGLE1、DPM-B1、および IGNORE の特性を持つものと指定されます。コンポーネント 2 は、SINGLE2、DPM-B2、および IGNORE の特性を持つものと指定されます。

## SINGLE または MULT パラメーターの指定

SINGLE1、SINGLE2、MULT1、および MULT2 の特性については、[477 ページの『出力プロトコルの判別』](#)に説明があります。

## VLVB または DPM-B1...DPM-B15 パラメーターの指定

VLVB または DPM-Bn を選択することによって、このコンポーネントが MFS の分散表示管理機能を使用できるかどうか決定されます。DPM-Bn の指定によって MFS がコンポーネントに指定されても、使用できるかどうかは、メッセージごとのオプションです。このシステム定義の例では DPM-Bn が使用されています。

VLVB を選択すると、コンポーネントに対する MFS-DPM の使用が妨げられ、入出力双方で MFS の代わりに可変長、可変ブロック形式が使用されることを示します。

## IGNORE または 1, 2,...10 パラメーターの指定

このパラメーターは、MFS DPM-Bn に、ユーザー定義の機能コードを指定するのに使用します。指定された機能は、突き合わせ機能の指定を含む MFS 形式 (DPM-B1...DPM-B15) を選択するために使用されます。IGNORE は、DEV ステートメントの FEAT=IGNORE で MFS 形式 (DPM-Bn) を選択するために指定するとき使用します。IGNORE はこのシステム定義の例で使用されています。

### 関連資料

[494 ページの『COMM マクロ・ステートメント』](#)

COMM マクロ・ステートメント上のキー・パラメーターは、APPLID、RECAN、および EDTNAME です。

## システム定義の要約

前の例にある SFIMS は、次の例に示すように、NYIMS に対して定義できます。

```
TYPE          UNITYPE=LUTYPE6
TERMINAL NAME=SFIMS
              COMPT1=(SINGLE1, DPM-B1, IGNORE)
              COMPT2=(SINGLE2, DPM-B2, IGNORE)
              SESSION=4
VTAMPOOL
SUBPOOL NAME=SF1
  NAME SF1LT1, COMPT=1
SUBPOOL NAME=SF2
  NAME SF2LT1, COMPT=2
SUBPOOL NAME=SF3
  NAME SF3LT1, COMPT=2
SUBPOOL NAME=SF4
  NAME SF4LT1, COMPT=2
```

この定義は、必ずしも [491 ページの『IMS への ISC ノードの静的定義』](#)内の図に正確に対応してはいません。この図において、SUBPOOL SF1 には複数の LTERM が含まれています (複数の LTERM を SUBPOOL に割り当てることができることを示します)。SINGLE1、SINGLE2、MULT1、および MULT2 に基づいて、1 つの SUBPOOL ごとに定義される 1 つの LTERM だけを示すように定義を単純化することができます。

最初の SUBPOOL、SF1 は、COMPT=SINGLE1 で定義された LTERM SF1LT1 です。NYIMS から SFIMS への非同期出力の場合には、LTERM SF1LT1 を下記のすべてに対する宛先 LTERM として使用します。

- SFIMS で非同期トランザクションを生成する NYIMS の中のアプリケーション・プログラムからの代替 PCB 出力。
- NYIMS に接続する端末によって開始されるメッセージ通信。

- SFIMS から NYIMS で受信される非応答トランザクション に対する応答。

SINGLE1 (BB/ メッセージ /EB) の特性では、1つの並列セッションが NYIMS と SFIMS 間のすべての非同期入出力を扱うことができるようにする必要があります。

3つの SUBPOOLS (SF2、SF3、SF4) が単一の LTERM に定義され、それぞれが SINGLE2 として定義された COMPT とともに定義されます。これらの3つの SUBPOOLS (セッション) を使用して、SFIMS に 応答モード・トランザクションを送信することができます。 応答モード・トランザクション (トランザクション起点に 応答の受信まで関連付けておくセッション) の性質上、3つの並列セッションは、ユーザー が NYIMS と SFIMS 間の 応答モードの障害を最小化できるように定義します。

大部分の IMS 定義において、1つの LTERM を各 SUBPOOL に割り当てれば、2つのノード間通信を適切に行えます。

示されていませんが、SFIMS からの着信 SINGLE1 通信を扱うために NYIMS 内で 5 番目の SUBPOOL を定義することも可能です。設計によって、NYIMS は 5 番目の SUBPOOL で定義されたセッションで SFIMS からすべての SINGLE1 入力を受信することができます。そして、最初の SUBPOOL (SF1) で定義されたセッションですべての SINGLE1 出力を送信することができます。このように計画すれば、2つのシステム間の非同期出力の競合を最小にすることができます。

#### **関連概念**

VTAM 端末の定義 (システム定義)



## 第 30 章 VTAM 接続用の ISC プロトコル

IMS では ISC プロトコルを使用して、ISC VTAM 接続経由のセッション、データ・フロー、およびメッセージ経路指定を制御します。以下のトピックには、ISC リンクでのデータの送受信に必要な特定のプロトコル情報が含まれています。

### このタスクについて

**制約事項:** ISC VTAM 接続の ISC プロトコルは、ISC TCP/IP 接続には適用されません。

## ネットワーク操作

ISC は論理装置間の複数セッションを許可するので、ISC についてのこの説明の残りの部分では、論理装置、セッション、およびハーフセッションの各用語を区別しています。

### このタスクについて

#### 定義:

- 2つのセッション・パートナー間のすべてのセッションに共通する特性を参照する場合、あるいは個別の各セッションを区別する必要がない場合には、論理装置の用語を使用しています。
- 論理装置間の特定の接続(セッション・インスタンス)に固有の特性を説明する場合は、セッションの用語を使用しています。
- セッション・パートナーの一方に固有の特性について説明する場合は、ハーフセッションの用語を使用します。

## IMS を作動可能にする

IMS で VTAM ログオンまたは BIND SCIP (セッション初期設定) 要求の受信を可能にするためには、DC キーワードで IMS /START コマンドを使用してください。

### このタスクについて

DC キーワードは、IMS データ通信処理を開始し、VTAM アクセス方式制御ブロック (ACB) がオープンされていない場合にはそれをオープンし、IMS VTAM ログオン出口を使用可能にします。IMS /START DC コマンドが出される前にかつ、ACB がオープンされた後に VTAM が受信したすべてのログオン要求は、/START DC コマンドが完了するまで VTAM のキューに保管されます。IMS の初期設定時に VTAM がアクティブであり、DFSDCxxx PROCLIB メンバーのキーワード VACBOPN=INIT が指定されていると、IMS VTAM ACB がオープンされます。DFSDCxxx PROCLIB メンバーのキーワード VACBOPN=DELAY が指定された場合は、IMS VTAM ACB のオープンは、/START DC コマンドが処理されるまで遅延されます。

また、/START DC コマンドによって、VTAM はキュー VTAM ログオン要求を IMS に渡すようにさせます。

## ISC の IMS ネットワークの開始

セッションが設定される前には、VTAM と NCP はアクティブにする必要があります。また、すべての論理装置は (VARY コマンドで活動化される) オンラインでなければなりません。

### このタスクについて

コールド・スタート後にはメッセージ再同期を実行する必要はありません。

#### 関連概念

[トランザクション・マネージャーの開始 \(オペレーションおよびオートメーション\)](#)

## 関連タスク

500 ページの『セッションの制御 (セッション制御プロトコル)』

セッション開始には、セッション・インスタンスの開始、セッションのバインド、必要な場合にはハーフセッションの同期確認などがあります。

## ISC の IMS ネットワークのシャットダウン

ネットワークを終了させ、IMS をシャットダウンするには、IMS /CHECKPOINT コマンドを使用します。

### このタスクについて

/CHECKPOINT コマンドの形式によって、即時にネットワークを終了させるか、あるいは処理が完了するのを待つかが決まります。

- /CHECKPOINT FREEZE|DUMPQ|PURGE は、次のように、すべての論理装置のセッションを即時に終了させます。

#### FREEZE

現行入出力メッセージの直後

#### DUMPQ

制御ブロックがチェックポイントを過ぎた後

#### PURGE

すべてのキューが空になった後

- /CHECKPOINT FREEZE|DUMPQ|PURGE QUIESCE は、シャットダウンの前に、すべてのネットワーク・ノードが通常の処理を完了できるようにします。

IMS コマンド /STOP DC も、処理の完了時に IMS ネットワークをシャットダウンします。

また、VTAM HALT NET などの VTAM コマンドもネットワークをシャットダウンします。

## 関連タスク

IMS ネットワークのシャットダウン (オペレーションおよびオートメーション)

## セッションの制御 (セッション制御プロトコル)

セッション開始には、セッション・インスタンスの開始、セッションのバインド、必要な場合にはハーフセッションの同期確認などがあります。

### このタスクについて

両ハーフセッションがバインドに合意しており、同期している場合、VTAM 開始データ通信 (SDT) コマンドを使用して、通常のトラフィック・フローが開始されます。

## ISC VTAM セッションの開始

セッションは、他の論理装置タイプ 6.1 と IMS の間でデータが伝送される前に、設定する必要があります。メッセージ再同期が必要な場合には、バインド後に実行されます。

次の方法のうちのいずれか 1 つを使用して、IMS にセッションの開始を要求することができます。

- ISC 論理装置が「自己開始 (initiate-self)」コマンドを送信して、セッション開始を要求します。VTAM がコマンドを検証し、IMS にその要求を渡します。
- ISC 論理装置が BIND を送信して、IMS のセッションを開始します。
- z/OS VTAM ネットワーク・オペレーターは、LOGON オプションを指定した z/OS VARY コマンドを使用して、論理装置に代わってセッション開始を要求します。VTAM はその要求を処理し IMS に渡します。VARY コマンドは、並列セッションの開始に使用することはできません。
- VTAM は IMS に、ログオン要求または BIND SCIP 要求 (IMS に属すると VTAM に定義されている各論理装置に対するもの) を渡します。(この方式は、並列セッションの開始に使用することはできません。)

- 権限がある IMS 端末オペレーターが、IMS /OPNDST コマンドを入力して、ISC 論理装置のセッション開始を要求します。

次のすべての条件に合う場合には、IMS はセッションを開始 (BIND 送信) または受け入れる (BIND 受信) ことができます。

- IMS マスター端末オペレーターが /START DC コマンドを出した。
- 論理装置名が IMS で認識されている。
- 論理装置が IMS で停止 (/STOP コマンド) されていない。
- 論理装置が IMS に定義された最大許容セッション数に達していない。(これは、静的定義端末のみに適用されます。)
- セッションが並列である場合には、CINIT または BIND セッション修飾子フィールドは、使用するメッセージ・キュー・セットを定義する有効な LTERM サブプール名を含んでいなければならない。有効なサブプール名とは、停止または割り振りされていないものです (セッション再始動中を除く)。IMIMS に単一セッションとして定義された単位装置でのセッション開始の要求時に、セッション修飾子フィールドが提供されてはなりません。
- 静的定義端末の場合、SUBPOOL マクロで指定された LTERM サブプールの MSGDEL オプションが、TERMINAL マクロでセッションに指定された MSGDEL オプションと矛盾していない。
- CINIT、BIND、または交渉可能 BIND 応答のいずれでも、無効な、あるいは矛盾するパラメーターが指示されていない。

#### 関連概念

491 ページの『IMS への ISC ノードの静的定義』

IMS のシステム定義で ISC ノードとして静的に定義されているサブシステムは、IMS では端末に、そして、IMS アプリケーション・プログラムでは 1 つ以上の論理端末 (LTERM) になります。このように、1 つの ISC ノードは、他の VTAM 端末定義にも使用できるシステム定義 マクロで定義されます。

#### 関連資料

[SLU P と LU 6.1 のバインド・パラメーター \(システム・プログラミング API\)](#)

## XRF 複合システムとの接続の確立

XRF 複合システムを、セッション・パートナーにすることができます。

XRF 複合システムには、2 つのタイプがあります。すなわち、XRF テークオーバー中のセッション管理に USERVAR を使用する XRF 複合システムと、XRF テークオーバー中のセッション管理に MNPS ACB を使用する XRF 複合システムです。

どちらの場合も、XRF 複合システムのアクティブ IMS サブシステムに障害が起きた場合は、代替 IMS サブシステムがテークオーバー中にセッションを再確立します。前のアクティブ IMS システムとリモート・システムとの ISC セッションが正常に終了しないと、セッションは開始されません。テークオーバーが完了すると、代替 IMS サブシステムは アクティブ・サブシステムとなり、セッションが正常に確立されるまで 30 秒ごとに (最大 20 回まで) セッション開始を再試行します。

### MNPS ACB を使用する XRF 複合システムへの接続

MNPS ACB を使用する XRF 複合システムであるセッション・パートナーとの接続を確立するには、単に、他の非 XRF IMS システムに接続する場合と同様に MNPS ACB 名を指定して、その XRF 複合システムとのセッションを開始します。

## USERVAR を使用する XRF 複合システムへの接続

USERVAR を使用する XRF 複合システムであるセッション・パートナーとの接続を確立するために実行する手順は、セッションが XRF 複合システムから開始しているか、XRF 複合システム外の IMS システムから開始しているかにより異なります。

### 手順

- 非 XRF ハーフセッションが IMS でセッションを開始した場合には、非 XRF ハーフセッションは次のことを行う必要があります。

- 次のマクロを発行して、現行のアクティブ IMS システムの APPLID を識別します。

```
INQUIRE OPTCD=USERVAR,AREA=area,AREALN=8
```

VTAM ネットワークがリリース 3.2 以降であり、USERVAR 管理機能拡張がインストールされている場合、INQUIRE マクロを出す必要はありません。

この要求によって、AREA パラメーターが、アクティブ XRF システムの USERVAR 名をもつ区域のアドレスを指定します。その後で、USERVAR がその時点でのアクティブ IMS システムの APPLID と置き換えられます。

- アクティブ IMS システムの APPLID を指定して、セッション開始要求を発行します。

VTAM ネットワークがリリース 3.2 以降であり、USERVAR 管理機能拡張がインストールされている場合、XRF システムの USERVAR を指定してセッション開始要求を出します。

- XRF 複合システムの IMS システムがセッションを開始する場合には、IMS システムは次のことを行います。
  - ユーザー・データ・フィールドの BIND に割り当てられた USERVAR を追加して、セッション開始要求を発行します。
  - 要求の終わりにあるユーザー・データ・フィールドの USERVAR を参照して、セッション開始要求の妥当性検査をします。開始はユーザーからは認識できません。

## セッションのバインド

ISC セッションでは、IMS は 1 次側ハーフセッションの役割 (BIND の送信)、または 2 次側ハーフセッションの役割 (BIND の受信) のどちらかを想定します。ただし、セッションが再始動し、リカバリーが必要な場合には、セッションの極性を保守する必要があります。

**定義:** セッション極性とは、同じセッション役割 (1 次または 2 次) が障害発生時に有効であり、セッション開始要求によって再確立されることを意味します。そうしないと、要求は拒否されます。セッションが IMS /OPNDST コマンドを使用して開始され、セッションの再始動とリカバリーが要求されない場合には、IMS は 1 次ハーフセッションであることを要求します。

## 交渉可能 BIND 対交渉不能 BIND

1 次ハーフセッションとして、IMS は VTAM モード表のパラメーターに応じて、交渉可能または交渉不能 BIND を送信します。

モード表項目は、VTAM CINIT または IMS /OPNDST コマンドで指示されるか、あるいは IMS システム定義中に TERMINAL マクロで定義されます。

### 定義:

- 交渉可能 BIND を送信すると、IMS はバインド・パラメーターを設定します。2 次ハーフセッションがパラメーターの一部を変更する場合があるので、交渉可能 BIND 応答の受信時に、IMS はすべてのパラメーターの妥当性を検査します。
- 交渉不能 BIND の送信前に、IMS はモード表項目からのすべてのパラメーターの妥当性を検査します。IMS が非互換パラメーターを検出すると、セッションが終了します。



2次ハーフセッションとして、IMSは交渉可能形式および交渉不能形式の両方のBINDを受信します。交渉可能BINDの受信時に、IMSは2次ネットワーク・アドレス可能単位 (NAU) プロトコル・フィールド以外の、バインド・パラメーターの妥当性を検査します。その後、IMSは、交渉可能BIND 応答の2次NAU プロトコル・フィールドを設定します。

交渉不能BINDを受信する時に、IMSは"必要なSTSN"と"送信BIS"以外のバインド・パラメーターの妥当性を、BINDを受け入れる前に検査します。その後、IMSはバインドに提供された2次NAUプロトコル定義下で稼働しなければなりません。

## 関連資料

[1次ハーフセッションとしてのIMS \(システム・プログラミングAPI\)](#)

[SLUPとLU 6.1のバインド・パラメーター \(システム・プログラミングAPI\)](#)

## 単一または並列セッションのバインド

単一セッションと並列セッションではセッションのバインド要件が異なります。

### このタスクについて

単一セッションをバインドするには、交渉可能または交渉不能BINDが送信されます。IMSと通信を行うISC論理装置は、IMSシステム定義時にLTERMの静的集合で定義されていなければなりません。IMSは、CINITパラメーターおよびBINDセッション修飾子ペア (SQP) フィールドを無視します。

並列セッションをバインドするためには、IMSと論理装置の間の特定の並列ハーフセッション・インスタンスを識別するために、CINITおよびBINDパラメーターがSQPフィールドを組み込んでいなければなりません。(ETO機能を使用して動的に作成されたセッションはすべて、並列セッションです。)バインド・パラメーターのこのフィールドには、基本および2次セッション修飾子があります。IMSと通信するISCノードのハーフセッション名は、論理装置名とそれに関連したセッション修飾子からなります。IMSハーフセッション名は、IMSACB名とIMSに関連したセッション修飾子を連結したものです。IMSセッション修飾子はサブプール名です。ハーフセッション名は、セッションおよびIMSサブシステム障害の発生前後を通して保管され、セッションに使用されるLTERMサブプールの割り振りか、ウォーム・スタート時にはその妥当性検査に使用されます。

### バインド競争の解決

IMSと別の論理装置が同時にBIND要求を送信し合い、2つのハーフセッション名がミラー・イメージになると、競争が発生します。

つまり、一方の2次論理装置名およびセッション修飾子と連結する基本論理装置名およびセッション修飾子が、他方の2次論理装置名およびセッション修飾子と連結する基本論理装置名およびセッション修飾子と等しくなっています。競争に勝って基本論理装置となる論理装置は、名前 (各ハーフセッション名から取られた基本論理装置名) が標準照合シーケンスの最上部にある論理装置です。このテストは、双方の側が互いに拒否し合い、セッションが確立されなくなることを防止します。

### 正常なセッション・バインドの後

正常なセッション・バインドに続いて、双方のハーフセッションは再同期プロセスを行う必要があります。IMSがコールド・スタートされているか、あるいはIMSと他の論理装置の間の直前のセッション・インスタンスが、ブラケット開始の停止/停止ブラケット開始 (SBI/BIS) プロシーチャーを使用して正常にシャットダウンされている場合には、メッセージ再同期は実行されません。

#### 関連概念

504 ページの『[セッションの再同期](#)』

サブシステム障害およびセッション障害において、IMS内のリカバリー可能リソース、メッセージおよびキューの保全性を保守するには、両方のハーフセッションで再同期プロセスに必要なセッション情報を保守しなければなりません。

#### 関連タスク

513 ページの『[ISC VTAM セッションの終了](#)』

ISC VTAM セッションを終了すると、論理装置を VTAM アプリケーション・プログラムとの現行接続から解放し、他の VTAM アプリケーションとのセッションにその LU を利用できるようにしたり、または通信と一緒に終了させたりします。

### 関連資料

554 ページの『LU 6.1 (SBI および BIS) の対称セッション・シャットダウン』

停止ブラケット開始 (SBI) とブラケット開始の停止 (BIS) という 2 つのデータ・フロー制御コマンドにより、対等レベルの LU 6.1 ハーフセッションが対称的かつ正常に終了することができます。

## セッションの再同期

サブシステム障害およびセッション障害において、IMS 内のリカバリー可能リソース、メッセージおよびキューの健全性を保守するには、両方のハーフセッションで再同期プロセスに必要なセッション情報を保守しなければなりません。

サブシステムの一方または両方でエラーが発生した場合、あるいはユーザー再始動プロシージャがこの情報を破棄した場合には、ユーザーの介入なしでメッセージの健全性を保守することはできません。

リカバリー可能作業単位がフロー上 (1 次から 2 次、または 2 次から 1 次) で未確定の可能性がある場合、再同期が必要です。

**定義:** 作業単位には、以下の図で示すように、ブラケット内の同期点間のすべての伝送が含まれます。同期点は、ハーフセッションの一方 (または両方) によって確認なしに要求されることがあります。これらの状態は、サブシステム障害やセッション障害、またはシャットダウン順序の異常終了によって生じる可能性があります。

(非交渉 BIND が送信された場合を除き) 通常のシャットダウン順序に続いてメッセージ再同期を実行する必要はありません。これは、通常のメッセージ・トラフィックあるいは同期点の要求が、セッションの終了前にこれ以上追加発生しないことを、両方のハーフセッションに相互に了解させるよう制御されるからです。2 次ハーフセッションが検出したエラー条件を 1 次ハーフセッションに通知できるようにするため、交渉不能 BIND の後に続いて常にメッセージ再同期が必要になります。

ハーフセッション・ペアは、VTAM BIND、設定およびテスト・シーケンス番号 (STSN)、およびデータ通信開始 (SDT) コマンドと再同期化されます。STSN コマンドを使用して、双方のハーフセッションは、それらが保守する同期点情報 (セッション・シーケンス番号) を再設定できます。

メッセージ再同期が必要な場合、ハーフセッションのどちらかが通常のデータ伝送を再開する前に、メッセージ再同期が正常に完了していなければなりません。

以下の図は、2 つの作業単位の例を示しています。

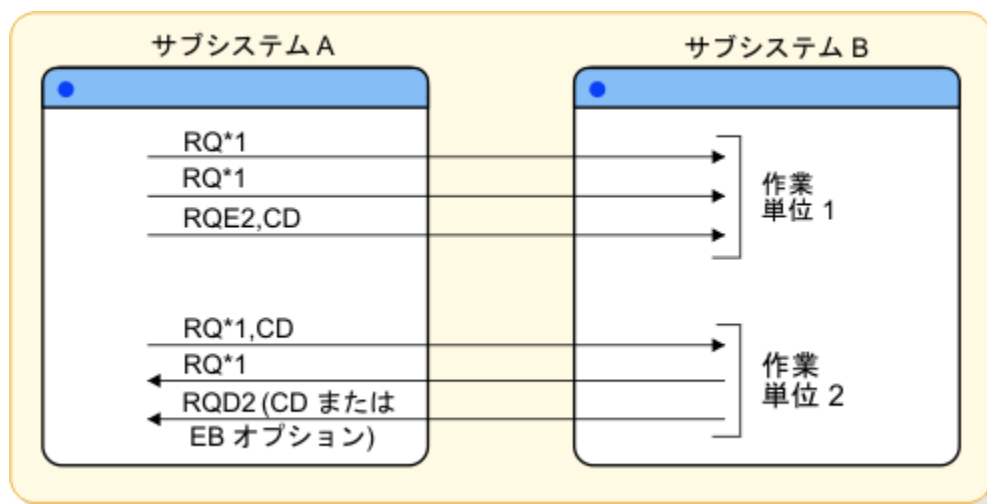


図 69. 作業単位の例

上記の図の作業単位 1 では、例外応答要求および CD への応答が暗黙の同期点を作成します。作業単位 2 では、DR2 応答を RQD2 に送信することによって同期点が作成されます。

## 関連概念

520 ページの『[ハーフセッションの同期の維持](#)』

同期点応答 (DR2) は ISC セッション・パートナー間で使用され、両方のパートナーの同期点マネージャーが、リカバリー可能リソースを同期させてコミットまたはバックアウトできることを確認するために使用されます。

## 関連資料

522 ページの『[同期点および応答要件](#)』

IMS 入出力メッセージ・フローを順次キュー・データ・セットからの入出力のフローとして示すことができます。

## 再始動/再同期プロシージャーの設計

IMS の再始動時、およびセッションの確立または再確立時には、再同期が必要なことも、必要でないこともあります。このトピックでは、IMS IMS 再始動時とセッション再始動時の再同期についての考慮事項を説明しています。

IMS 再始動は、次の方法でメッセージ・リカバリーに備えます。

- 通常の IMS サブシステム 再始動プロシージャーでは、IMS メッセージ・キューとセッション再始動情報を最後のまたは指定のチェックポイントへ戻す。
- 緊急の IMS サブシステム 再始動プロシージャーでは、IMS メッセージ・キューとセッション再始動情報を障害直前の状態に戻す。

障害が IMS と他の論理装置の間の通常シャットダウン順序の前に発生した場合には、必ず障害時にセッションにあったものと同じハーフセッション・ペア (ハーフセッション名) を使用してセッションを再始動する必要があります。また、セッションが再開された場合には、ハーフセッション・ペアは直前の関係を保持しなければなりません。つまり、以前の 1 次ハーフセッションは再び 1 次となり、以前の 2 次ハーフセッションは現行の 2 次にならなければなりません。

## シーケンス番号の保守

セッション障害とサブシステム障害において、セッションのリカバリーとメッセージ再同期を可能にするためには、両方のハーフセッションが、シーケンス番号と他の標識のチェックポイントを保持する必要があります。

特に、以下の項目についてチェックポイントを保持する必要があります。

- セッションが最後にアクティブであった時に作成された 3 つのシーケンス番号  
3 つのシーケンス番号とは、ハーフセッションが送信したフローの潜在 (保留) シーケンス番号とコミットされたシーケンス番号、およびハーフセッションが受信したフローの最後にコミットされたシーケンス番号です。
- セッション障害中に保留にされていた作業単位をバックアウトするかコミットするかという決定を、片方のハーフセッションが一方的に行った時に作成された標識
- 決定の方向を示す指示

シーケンス番号の不一致およびリカバリー可能な作業単位をコミットまたはバックアウトするという誤った決定は、チェックポイント再同期情報の VTAM 設定 およびテスト・シーケンス番号 (STSN) コマンドで受信されたシーケンス番号を比較することによって検出されます。各ハーフセッションは、アウトバウンド・フローに一致しない無効なシーケンス番号を検出し、再同期要求を拒否することができます。各ハーフセッションは、作業単位をコミットまたはバックアウトするという相手方ハーフセッションの一方向的な決定に対して、同意または不同意することもできます。この一致または不一致は、インバウンド・フローのシーケンス番号の不一致の検出に基づき、誤った決定が STSN 受信側のインバウンド・フローでなされた時には、2 番目の STSN を含みます。STSN コマンドの受信側は、(最初の STSN の TEST NEGATIVE の受信に応答して送信される) 2 番目の STSN コマンドに対する応答の前にこれを行います。この 2 番目の STSN は、インバウンド・フローで誤った決定がなされたことを STSN 受信側に知らせます。

## コールド・スタートでのセッションの回復

無効なシーケンス番号の不一致、および一方的な決定がなされなかった場合の不一致によって、誤ったログ、またはログ内の誤ったチェックポイントからサブシステムの再始動が指示される場合があります。この状態では、一方または両方のハーフセッションをコールド・スタートして強制的に ISC セッションをリカバリーする必要がある可能性があります。

IMS では、認可端末オペレーターが、必要な時、セッションの開始を試みる前に **IMS /ASSIGN** (VTAMPOOL のサブプール) コマンドを使用して、リカバリー可能からコールド・スタートへ、IMS ISC セッション状態を変更することができます。セッションをコールド・スタート状態に変更すると、そのセッションに関連したサブプールは、任意のノードとの任意の ISC コールド・スタート可能並列セッションに割り振り可能になります。

片方または両方のハーフセッションがコールド・スタートされた場合のセッション・リカバリーは、重大なエラーとは見なされません。なぜなら、定義により、ハーフセッションをコールド・スタートすると、リカバリーまたは再同期の交渉 (同意または不同意) を行う対象となる直前のアクティブ・セッションからの情報がないからです。

## 保留状態作業単位についての一方的決定の制御

リソースが誤ってサブシステム間で同期化されないのを防ぐために、片方のハーフセッションが一方的に保留状態の作業単位のコミットまたはバックアウトを決定した場合は、セッション障害の後で、セッション再同期を継続するかどうかを指定することができます。

### このタスクについて

IMS ではシステム定義仕様で、インバウンド・シーケンス番号の不一致にかかわらず常に再同期できるようにする (TERMINAL マクロ・ステートメントの **OPTIONS=FORCSESS** または ETO ユーザー記述子) か、またはインバウンド・シーケンス番号が一致する場合に限り再同期できるようにする (**OPTIONS=SYNCSESS**) ことができます。IMS /CHANGE コマンドでキーワードを使用すると、認可端末オペレーターがシステム定義仕様をオーバーライドして、セッションを一度開始することができます。CHANGE コマンドは、セッションが一度開始されると、元のシステム定義仕様にリセットされます。IMS では、一方的な決定でセッション障害中にバックアウトできます。ただし、出力メッセージ同期点が保留状態のセッション障害中に、セッション (端末およびオプションとしてサブプール・パラメーター) または LTERM に対して IMS /DEQUEUE コマンドを使用することは、許可された端末オペレーターが、保留状態の出力をコミットするという一方的な決定をしたと見なされます。

保留状態の出力同期点応答は、IMS /DISPLAY コマンドを使用して判別することができます。2つの IMS サブシステムが ISC セッションによって接続されている場合は、/DEQUEUE コマンドを使用して保留状態の出力メッセージがデキュー (コミット) された IMS サブシステムの反対側のサブシステムにおいて、FORCSESS オプションが有効でなければなりません。そうでないと、メッセージ DFS2065 が出力されてセッションの再同期は失敗します。SYNCSESS でなく FORCSESS を使用した場合には、2つの IMS サブシステム間に他の影響はありません。

## イン・ブラケット障害からの回復

イン・ブラケットの間にセッションが失敗した時には、セッションの再始動時に失敗したブラケットを再始動しなければならないことがあります。

### このタスクについて

ブラケット状態マネージャーおよびバインド内の半二重リセット・フラグは、BIND 要求または各ハーフセッションの交渉可能バインド応答のどちらかに設定できます。これらのフラグは、イン・ブラケット / SEND または RECEIVE に設定され、事前に付加されたプロセスを再始動しなければならない場合があることを示します。

## 再始動で利用可能な応答または会話型出力

セッション開始時に、IMS が応答モードまたは会話型出力が送信可能な場合、あるいは IMS が同期点応答保留状態である場合には、ブラケット状態マネージャーはリセットされ、バインドまたはバインド応答にある IMS 部分内の半二重フラグは、IMS によってイン・ブラケット /SEND に設定されます。

IMS が会話型モードであるが、利用可能または保留状態の会話型出力がない場合には、会話を継続するためには入力が必要になるので、これらと同じバインドまたはバインド応答フラグはイン・ブラケット /RECEIVE に設定されます。会話または応答モード出力が利用可能な時には、IMS は、非交渉 BIND や、IMS がイン・ブラケット /RECEIVE であることを示す交渉 BIND 応答を受け入れません。直前のセッションから同期点が保留状態にある場合を除き、セッションは再開され、送信される STSN は、処理を継続するために SET AND TEST オプションに対して TEST POSITIVE 応答を受信しなければなりません。

IMS が会話または応答モードにあり、非交渉 BIND またはブラケット間を示す交渉 BIND に対する応答を受信した場合には、IMS は会話または応答モードを終了しようとします。この終了は、IMS が通常データ・フロー・アクティブ状態 (SDT 後) で会話または応答モードである場合に、FMH7 または LUSTATUS でブラケット終了が受信されると発生する終了と同じです。

会話または応答モードの終了は、出力応答メッセージが送信可能な場合、あるいはハーフセッションが次の会話型入力を待っている場合にのみ保証されます。このため、出力応答が送信可能な場合、あるいは会話型入力が不要な場合は、IMS 制限はブラケット間のバインドによって会話または応答モードを終了しようとします。出力応答が送信不能であるか、会話型入力が不要な場合には、後でセッション開始を再試行する指示とともに、警告メッセージが IMS マスター 端末オペレーターに送信されます。

IMS がイン・ブラケット /SEND を示す交渉 BIND を受信し、会話または応答モードでない場合には、ブラケット間を示す BIND 応答が送信されます。IMS が会話または応答モードでない場合、および非交渉 BIND またはイン・ブラケット /SEND を示す交渉 BIND 応答を受信した場合には、IMS は SDT の直後にブラケット終了を示す LUSTATUS - NO-OP を送信します。このような状況ではトランザクション再始動が可能ではないために、LUSTATUS - NO-OP が送信されます。

会話または応答モード出力同期点応答が直前のセッションから保留状態であるために、BIND または BIND 応答後、IMS がイン・ブラケット /SEND のままである場合があります。STSN 処理が、応答が送信済みであることを示していると、出力メッセージはデキューされる可能性があります。この場合、出力メッセージが応答モード応答であったならば、SDT 後にブラケット終了を示す LUSTATUS - NO-OP が送信されます。出力メッセージが会話型応答であった場合、入力は会話を継続する必要があるため、方向変換を示す LUSTATUS - NO-OP が送信されます。

STSN が会話モードまたは応答モード・メッセージが確認されていないことを示している場合には、IMS は、メッセージが方向変換で送信され、例外同期点応答が要求された場合と同様に、メッセージを再び同期点応答保留状態にします。その後では、通常フロー応答は保留状態出力メッセージの暗黙の確認と見なされ、応答が LUSTATUS - Abort または ATTACH ATTDPN=SYMSMSG であった場合でも IMS によってそのメッセージはデキューされます。これらの応答は、メッセージのスケジューリングまたは実行に関連した追加のエラー条件を示しています。セッションが失敗しなかった場合には、これらのエラー条件は LUSTATUS - Abort または ATTACH ATTDPN=SYMSMSG ではなく、同期点要求に対する例外応答により反映されます。使用されたセンス・コードによっては、セッション障害とアプリケーション障害が同時に発生した際に、LUSTATUS - Abort または ATTACH ATTDPN=SYMSMSG で行われるように、メッセージをキューから除去(コミット)せずに、例外応答はメッセージを再伝送のためにメッセージ・キューに戻す(バックアウトする)ことがあります。

## IMS 障害を伴わないセッション障害

IMS が応答モードまたは会話型入力メッセージに要求された同期点応答を戻す前に、セッション障害 (IMS サブシステム障害を含まない) が発生することがあります。

セッションが再設定される時に IMS がバインドを使用してセットする DFC 状態は、IMS イン・ブラケット /SEND です。続く STSN シーケンス番号のリカバリーは、入力メッセージがコミットされている (トランザクションが応答メッセージの挿入後に同期点に達した) こと、あるいはまだ入力メッセージがコミットされていないことのどちらかを反映することがあります。いずれの場合にも、イン・ブラケット /SEND としてバインドされている IMS は、入力メッセージが受信され、他のハーフセッションが応答メッセージを

待機しなければならないことを示しています。セッション再始動により元の同期点要求がリセットされるため、応答メッセージはその時点で元の入力メッセージの暗黙の同期点応答になります。

IMS は、IMS 障害全体でセッションが会話型モードであった状態を回復します (また、関連した入力メッセージと出力応答も回復します)。障害が、応答モード応答を利用可能にするトランザクション同期点の後に発生した場合には、セッションが応答モードであったという事実 (および関連した出力応答すべて) は、IMS 障害のみで回復されます。

## IMS 障害によるセッション障害

IMS が応答モード (メッセージがリカバリー可能な場合) または会話型入力の同期点要求に応答する前に、IMS サブシステム障害をも伴うセッション障害が起きることがあります。

このセッションが再設定される時に、IMS がバインドを使用してセットする DFC 状態は、IMS がイン・ブレット /SEND でバインドされたことを示します。いずれの場合にも、サブシステム・ログからの IMS 再始動プロセスのために、まだコミットが実際には発生していない場合でも、続く STSN シーケンス番号リカバリーは入力メッセージがコミットされていることを反映します。ただし、再始動後にトランザクションが異常終了した場合に限り、矛盾する結果が生じます。この場合、この状態によって、応答モード出力応答メッセージは、回復した ISC セッションで非同期 (ATTACH EB または ATTACH ATDPN=SCHEDULER) 送達が可能になります。

## コマンドおよび実行モードのリカバリー可能性

IMS コマンド (/DIS、/RDIS、および /FOR を除く) およびテスト・モード入力は、入力メッセージ・キューに置かれずに即時に実行されます。これらのコマンドは、トランザクション同期点を使用して出力を使用可能にする前に、すべての処理を完了します。障害によって、コマンドまたはテスト・モード・トランザクション入力、および関連した出力メッセージがバックアウトおよび破棄されます。

IMS コマンド /DIS、/RDIS、および /FOR は、非同期キュー出力を作成します。障害が発生する前にキューに入れられた出力は、セッションの再確立時に回復され、非同期伝送が可能になります。

障害がトランザクション同期点の前に発生した場合には、高速機能、リカバリー可能応答モード、およびリカバリー可能会話型トランザクションは、バックアウトされ破棄されます。セッションの再確立時には、応答モードは回復されません。

## 再始動プロセスの調整

ハーフセッションは、セッション障害後の再始動を調整するため、規則を使用します。

ハーフセッションで使用される規則には、以下のものがあります。

- 再始動は、常に最後に完了した同期点から起こります。
- セッション・バインドは、現行の同期点で半二重状態を設定します。
- 1 次ハーフセッションが再始動を必要とする場合は次のことが行われます。
  - 適切な半二重状態を選択するために、BIND 要求を送信します。
  - 2 次ハーフセッションが再始動を望まない場合を除き、2 次ハーフセッションは BIND 応答でこの状態を変更することはできません。
  - 2 次ハーフセッションが再始動を望まない場合には、2 次ハーフセッションは BIND 応答を送信し、ブレット状態をブレット間に設定することによって再始動しないようにします。
- 1 次ハーフセッションが再始動を望まない場合は次のことを行います。
  - 1 次ハーフセッションは、ブレット状態をブレット間に設定する BIND 要求を送信します。
  - 2 次ハーフセッションが再始動を望む場合には、2 次ハーフセッションは BIND 応答を送信し、適切な半二重状態を設定し、ブレット状態をブレット間に設定します。
  - 2 次ハーフセッションが再始動を望まない場合には、BIND に対する 2 次ハーフセッションの応答は変更されません (つまり、1 次ハーフセッションがセットした状態と一致します)。
- 同期点応答の保留状態の要求をすべて再同期するために、STSN が使用されます。メッセージと応答



- 不一致が発生しなかった場合には、現行の同期点は一致していると同意されています。データ通信開始 (SDT) の送信後に、再始動が試みられます。
  - 半二重状態を STSN に基づいて現行の同期点に合わせるために、送信状態にあるハーフセッションが LUSTATUS - NO-OP、CD を送信しなければならないことがあります。
  - 不一致が発生しても、セッションが継続する場合には、送信状態のハーフセッションはセッションをコンテンションに置くために、LUSTATUS - NO-OP、EB を送信します。
- 最後の再始動点で送信状態のセッションは、明示の ATTACH を送信してセッションを再確立します。

2 次側の役割を想定するハーフセッションは、次の状態でのセッション・バインドを拒否します。

- セッション・バインド・パラメーターがイン・ブラケットを示しており、「他のハーフセッションが最初に話し」、さらに応答モードまたは会話型出力 (またはそれに相当するもの) がデータ通信アクティブ状態で即時利用可能になっている。これは、セッション再始動論理エラーです。
- 異常な障害またはシャットダウン順序に続く再始動時に、セッション・バインド・パラメーターは、IMS と他の論理装置間の直前のセッションで有効な名前とは異なるハーフセッション名を示します。

ISC セッションでは、どちらのサブシステムも 1 次ハーフセッションの役割または 2 次ハーフセッションの役割になることができ、どちらのサブシステムもメッセージの再同期を要求することができます。このため、これらのハーフセッション・ペアがメッセージまたは同期点の保全性を保守できるようにするための規則が必要になります。

以下のトピックでは、STSN の形式および、再同期とリカバリーを完了するための STSN の使用方法について説明します。

#### 関連資料

542 ページの『LUSTATUS プロトコル』

IMS は、以下の図で要約されているように LUSTATUS を送受信します。この図で示された DFC ブラケット、送信 / 受信、および応答要件についての考慮事項は、このトピックの残りの部分で詳細に説明します。

## STSN を使用するセッション同期の判別

再同期が必要な場合は、それをバインド交渉中に伝えることができます。再同期し、ハーフセッションをセッション終了時に存在した状態に戻すための要件を判別するために、BIND 要求の 2 つのフラグが使用されます。

両方のハーフセッションが正常に終了した場合、つまり、どちらにも処理しなければならない未解決の通信がない場合には、あたかも新しいセッションのバインド・パラメーターが送信されたかのようにセッションが再始動され、セッション・パートナーの相対位置を保守する必要がなく、また同一のハーフセッション名を使用する必要もありません。この判別を行うための 2 つのフラグは、次のものです。

- シーケンス番号標識
  - 1 = シーケンス番号利用可能
  - 0 = シーケンス番号利用不能
- ブラケット開始の停止 (BIS) 標識
  - 1 = BIS 送信済み
  - 0 = BIS 未送信

次の表は、バインド送信側 (1 次ハーフセッションまたは PHS) とバインド受信側 (2 次ハーフセッションまたは SHS) の両方に関して、これらの 2 つのフラグで設定された状態を説明するマトリックスです。

両方のハーフセッションが「COLD START」状態にある場合は、シーケンス番号が使用できないか、または送信してもらう必要があります。セッションのシャットダウンはこのようであるので、再同期は必要ありません。バインドが交渉され、セッションはバインド・パラメーターに一致して開始されます。交渉不能 BIND では、両方のハーフセッションに必要ないくつかの情報が、STSN フローが存在する状態になるまでは利用できません。そのため、交渉不能 BIND では、STSN は常に送信されます。



表 82. BIND アクション / 応答マトリックス

	バインド受信側 (SHS): 番号が使用不可	バインド受信側 (SHS): 番号が使用可能、BIS 送信済み	バインド受信側 (SHS): 番号が使用可能、BIS 未送信 (514 ページの表 83 を参照)
バインド送信側 (PHS): 番号が利用不能	COLD STSN 不要 RESET を NOBB に送信	COLD STSN 不要 RESET を NOBB に送信	COLD/WARM 不一致 STSN 送信
バインド送信側 (PHS): 番号が使用可能、BIS 送信済み	COLD STSN 不要 RESET を NOBB に送信	COLD STSN 不要 RESET を NOBB に送信	WARM STSN 送信
バインド送信側 (PHS): 番号が使用可能、BIS 未送信 (515 ページの表 84)	WARM/COLD 不一致 STSN 送信	WARM STSN 送信	WARM STSN 送信

IMS は、次のように、「BIS 送信済み」および「シーケンス番号利用可能」フラグを 交渉可能 BIND と BIND 応答 (または、それぞれ PHS と SHS) に設定します。

- IMS コールド・スタートまたは正常なセッション終了に続くセッションの コールド・スタートは、次のようになります。

BIS 未送信

シーケンス番号は使用不能

- セッション再始動

– 直前のセッションが正常に終了し、IMS による正常終了の試行がそれに続く場合には、次のようになります。

BIS 送信済み

シーケンス番号使用可能

- 直前のセッションが正常に終了した (IMS による正常終了が試みられない) 場合には、次のようになります。

BIS 未送信

シーケンス番号使用可能

#### 関連資料

[511 ページの『再同期の実行』](#)

セッションが再確立された (あるいは、初めて確立された) 場合には、個々のフローの状態は、各ハーフセッション内の同期に利用可能なシーケンス番号で示されます。

[554 ページの『LU 6.1 \(SBI および BIS\) の対称セッション・シャットダウン』](#)

停止ブラケット開始 (SBI) とブラケット開始の停止 (BIS) という 2 つのデータ・フロー制御コマンドにより、対等レベルの LU 6.1 ハーフセッションが対称的かつ正常に終了することができます。

[516 ページの『STSN コマンドの形式』](#)

STSN コマンドと STSN 応答の両方が、5 バイトのデータ・フィールドを含んでいます。

## 再同期の実行

セッションが再確立された (あるいは、初めて確立された) 場合には、個々のフローの状態は、各ハーフセッション内の同期に利用可能なシーケンス番号で示されます。

個々のハーフセッション内の同期に利用可能なシーケンス番号は、フローごとの以下の状態を示すことができます。

- COLD

このセッションには、送信または比較するシーケンス番号はありません。1 次ハーフセッションは、BIND 要求のシーケンス番号標識ビットをリセットしてこれを示します。BIND が交渉可能である場合には、2 次ハーフセッションは、BIND 応答内と同じビット設定でこれを示します。STSN が (交渉不能 BIND 用に) 必要な場合には、1 次ハーフセッションは、シーケンス番号値としてゼロを持つ SET AND TEST アクション・コードを送信します。

- NOT PENDING

このセッションには、未確定の (つまり、コミットを待っている) 作業単位はありません。

- PENDING

ある作業単位は未確定 (コミットを待っている) です。セッションに利用可能なシーケンス番号は、コミット済み番号 (コミットされた最後のメッセージのシーケンス番号) および潜在番号 (発行された最後のメッセージのシーケンス番号) です。

- DECISION TO COMMIT

未確定作業単位は、セッション障害中にコミットされます。潜在およびコミット済みの数字は、まだ利用可能です。(コミットの決定は、セッションの故障中に発行された /DEQ コマンドの結果としてのみ発生します。)

- DECISION TO BACK OUT

未確定作業単位は、セッション障害中にバックアウトされます。潜在およびコミット済みの数字は、まだ利用可能です。(IMS は作業単位をバックアウトしませんが、他のセッション・パートナーがそれを行う可能性があります。)

- INVALID

無効なシーケンス番号が検出されました。(無効なシーケンス番号とは発生すべきでない番号であり、例えば、2 次から 1 次フロー上で送信されたシーケンス番号が、STSN 受信側が保管したコミット済みまたは潜在番号と一致しないシーケンス番号である場合など)。このタイプの状態は、通常、誤ったログ・データ・セットを示し、マスター端末オペレーターはセッションの回復に介入する必要があります。

- RECEIVED PENDING (同期点応答が失われた)

このフローの受信側は、送信側が送信した作業単位全体を受信し、ネットワークで失われた可能性のある DR2 を生成しました。

- NOT RECEIVED PENDING (同期点応答は失われていない)

このフローの受信側は、送信側が送信した RQ\*2 を受信していないので、送信側が保留状態にした作業単位をバックアウトしました。

ハーフセッションは、STSN 要求と応答内のバインド交渉とアクション・コードを使用して、それぞれの状態に関する情報を交換します。

### 関連資料

[514 ページの『セッションの再同期のための STSN の使用』](#)

1次から2次(P-S)へのフローおよび2次から1次(S-P)へのフローに STSN を使用した場合の結果とアクションが、以下のトピックの表に示されています。

## セッション開始の完了

---

ハーフセッションが同期した後で、いずれのハーフセッションにおいても、セッションの受け入れまたは拒否を決定することができます。

- 1次ハーフセッションがセッションの継続を望まない場合、1次ハーフセッションは UNBIND を送信します。
- 1次ハーフセッションがセッションの継続を望む場合、1次ハーフセッションはデータ通信開始 (SDT) を送信します。
  - 2次ハーフセッションが肯定応答をする場合には、セッションは正式にバインドされ、セッションの通信が開始されます。
  - 2次ハーフセッションがセッションの中止を望む場合には、2次ハーフセッションは1次ハーフセッションの SDT を拒否し、拒否されたことに対して1次ハーフセッションは UNBIND で応答しなければならず、セッションは終了します。

IMS が1次ハーフセッションであれば開始データ・トラフィック (SDT) 応答の受信時に、あるいは IMS が2次ハーフセッションであれば SDT 応答の送信時に、LTERM サブプールの割り振りを含めてセッション開始が完了します。両方のハーフセッション相互の同意によってセッションが終了されるまでは (双方対称のシャットダウン)、その間にセッション障害またはサブシステム障害が発生したとしても、LTERM サブプールは他方のハーフセッション名に割り振られたままです。このことにより、いったん割り振りが行われると、正常なセッション終了によってサブプールが解放されるまでは、同じハーフセッションの後続のセッション・バインドはすべて、セッション異常終了時にアクティブであったのと同じバインド・セッション修飾子を指定する必要があります。

セッションが SDT の完了前に終了した場合には、セッションのコールド・スタートによって新たに試みられたサブプール割り振りはすべてバックアウトされ、サブプールは割り振り可能状況に戻ります。

IMS では、マスター端末オペレーターには常に、データ通信開始 (SDT) の完了前に拒否されたセッションを通知する必要があります。ISC セッションでは、通常の開始および終了順序の場合、IMS マスター端末オペレーターへの通知はオプションです。オペレーターへの通知は、TYPE と TERMINAL マクロの OPTIONS キーワード・パラメーター MTOMSG と NOMTOMSG、または ETO ログオン記述子で指定します。

## セッションの実行

---

データ・フロー制御プロトコルを使用して、ISC セッションのデータ・フローを制御します。

### 関連概念

518 ページの『IMS 応答モードまたは会話型出力エラーの処理』

このトピックでは、ISC セッション中の応答モード・エラーと会話型エラーを IMS がどう処理するか、およびハーフセッションの同期を維持する方法について説明します。

### 関連資料

528 ページの『データ・フロー制御プロトコルの関連情報』

以下のトピックで、データ・フロー制御 (DFC) のバイト・レベルのプロトコルについて説明します。プロトコルを以下に示します。

## ISC VTAM セッションの終了

ISC VTAM セッションを終了すると、論理装置を VTAM アプリケーション・プログラムとの現行接続から解放し、他の VTAM アプリケーションとのセッションにその LU を利用できるようにしたり、または通信と一緒に終了させたりします。

### このタスクについて

**定義:** セッション終了の 2 つのタイプは、「正常」と「異常」です。正常終了では、セッション終了前に両方のハーフセッションが通常の処理を完了できます。異常終了では、セッションは無条件に終了を強制されます。

セッション終了は、IMS マスター端末オペレーター、VTAM ネットワーク・オペレーター、またはどちらかのハーフセッションによって呼び出されます。

セッション終了には様々な方法があるために、各 IMS ネットワークのインストールは、セッションを終了するための特定のプロシーチャーを判別しなければなりません。これらのプロシーチャーを開発する時には、セッション終了処理の要件を考慮する必要があります。

### 正常終了

ISC セッションの正常終了は、データ・フロー制御標識のブラケット開始の停止 (SBI) および停止ブラケット開始 (BIS) のフローで発生します。正常終了はどちらのハーフセッションからでも開始できます。

また、正常なセッション終了は、MTO が **FREEZE**、**PURGE**、または **DUMPQ** パラメーターおよび **QUIESCE** パラメーターを持つ **IMS /CHECKPOINT** コマンドによって IMS ネットワークの終了を呼び出すときにも発生します。**QUIESCE** パラメーターは、セッション終了前にメッセージ・キューが空になっていることを確認します。

シャットダウンが完了したことをすべての端末が示すと、IMS は次のことを行います。

- 1 次ハーフセッションとして働いている場合には、チェックポイント処理を実行してから VTAM **CLSDST** マクロ命令を出します。
- 2 次ハーフセッションとして働いている場合には、**UNBIND** を待ちます。

**CLSDST** により、VTAM は **UNBIND** コマンドを送信できます。**CLSDST** または **UNBIND** は、IMS とのセッションから論理装置を解放します。論理装置へこれ以上のデータは伝送できません。

セッション終了の処理中に、IMS マスター端末オペレーターは、**IMS /CLSDST**、**/STOP**、または **/CHECKPOINT** コマンドを使用して、無条件にネットワークを終了することができます。

### 異常終了

セッション異常終了は、伝送エラーやプロトコル・エラー、あるいは受信側メッセージ処理プログラムでデータを受け入れられないようなデータのエラーが原因で発生する可能性があります。ISC セッションには 2 つの対等レベルのシステムがあるために、エラー・リカバリー処理とセッション異常終了プロセスは異なることがあります。セッション異常終了を必要とするエラー条件を IMS が検出すると、IMS が 1 次ハーフセッションの場合は VTAM **CLSDST** マクロを出し、IMS が 2 次ハーフセッションの場合は **OPTCD=UNCOND** を持つ **TERMSESS** を出します。

### 関連資料

554 ページの『[LU 6.1 \(SBI および BIS\) の対称セッション・シャットダウン](#)』

停止ブラケット開始 (SBI) とブラケット開始の停止 (BIS) という 2 つのデータ・フロー制御コマンドにより、対等レベルの LU 6.1 ハーフセッションが対称的かつ正常に終了することができます。

528 ページの『[データ・フロー制御プロトコルの関連情報](#)』

以下のトピックで、データ・フロー制御 (DFC) のバイト・レベルのプロトコルについて説明します。プロトコルを以下に示します。

## セッションの再同期のための STSN の使用

1 次から 2 次 (P-S) へのフローおよび 2 次から 1 次 (S-P) へのフローに STSN を使用した場合の結果とアクションが、以下のトピックの表に示されています。

### 1 次から 2 次へのフロー・マトリックス

以下の表は、セッションの再同期時に取られるアクションと、1 次ハーフセッションから 2 次ハーフセッションへの STSN コマンドのフローを示しています。

表の各枠の最初の項目は STSN コマンド処置であり、2 番目の項目は STSN コマンドに戻される応答です。これらは斜線 (/) で区切られます。2 つの STSN が送信された場合にはコンマで区切られます。

表 83. STSN 1 次から 2 次へのフロー

1 次ハーフセッション	2 次ハーフセッション、セッション・コールド	2 次ハーフセッション、未受信: 保留中	2 次ハーフセッション、受信済み: 保留中
保留中	<b>1</b> 設定およびテスト / リセット	<b>2</b> 設定およびテスト / 否定	<b>3</b> 設定およびテスト / 肯定
コミットの判断	<b>4</b> 設定およびテスト / リセット	<b>5</b> 設定およびテスト / 否定、 セット / 肯定	<b>6</b> 設定およびテスト / 肯定
バックアウトの判断	<b>7</b> 設定およびテスト / リセット	<b>8</b> 設定およびテスト / 肯定	<b>9</b> 設定およびテスト / 否定、 セット / 肯定
保留中でない	<b>10</b> 設定およびテスト / リセット	<b>11</b> 設定およびテスト / 肯定	適用できない

### 上記の表に関する注記

**1** STSN 送信側は PENDING です。STSN 受信側にはシーケンス番号がありません。COLD/WARM の不一致が発生しました。セッションは続行できます。STSN 送信側は、セッションを継続しなければなりません。

**2** STSN 送信側は PENDING です。STSN 受信側は、保留状態の RQ\*2 を受信しておらず、STSN 応答に TEST NEGATIVE を戻します。STSN 送信側は、保留状態の作業単位を最後のコミット・ポイントにバックアウトします。

**3** STSN 送信側は PENDING です。STSN 受信側は、保留状態の RQ\*2 を受信し、+ DR2 を送信しました。STSN 受信側は TEST POSITIVE を応答し、STSN 送信側は保留状態の作業単位をコミットします。

**4** STSN 送信側はロックを解除しリリースをコミットする決定をしますが、RQ\*2 が未解決です。STSN 受信側にはシーケンス番号がありません。COLD/WARM の不一致が発生しました。セッションは続行できます。STSN 送信側は、セッションを継続しなければなりません。

**5** STSN 送信側はロックを解除しリリースをコミットする決定をしますが、RQ\*2 が未解決です。STSN 受信側は、保留状態のチェーンを受信しておらず、STSN 応答に TEST NEGATIVE を示します。STSN 送信

側は、誤った決定がなされたことを STSN 受信側に知らせるために、SET アクション・コードといっしょに 2 番目の STSN を送信します。STSN 受信側は TEST POSITIVE を送信してセッションを継続するか、あるいは INVALID を送信してセッションの継続を拒否することができます。

**6** STSN 送信側はロックを解除しリリースをコミットする決定をしますが、RQ\*2 が未解決です。STSN 受信側は保留状態のチェーンを受信しそれをコミットし、STSN 応答に TEST POSITIVE を示します。

**7** STSN 送信側は保留状態の作業単位のバックアウトを決定しましたが、RQ\*2 が未解決です。STSN 受信側にはシーケンス番号がありません。COLD/WARM の不一致が発生しました。セッションは続行できません。STSN 送信側は、セッションを継続しなければなりません。

**8** STSN 送信側は保留状態の作業単位のバックアウトを決定しましたが、RQ\*2 が未解決です。STSN 受信側は、シーケンス番号が一致するように、保留状態のチェーンを受信していません。TEST POSITIVE が STSN 応答に戻されます。

**9** STSN 送信側は保留状態の作業単位のバックアウトを決定しましたが、RQ\*2 が未解決です。STSN 受信側は保留状態のチェーンを受信し、リソースをコミットします。STSN 送信側が以前にコミットされたシーケンス番号を送信し、STSN 受信側がチェーンを受信したために、TEST NEGATIVE が戻されました。STSN 送信側は SET アクション・コードといっしょに 2 つ目の STSN を送信し、誤った決定がなされたことを STSN 受信側に知らせます。STSN 受信側は TEST POSITIVE を送信してセッションを継続するか、あるいは INVALID を送信してセッションの継続を拒否することができます。

**10** STSN 送信側は P-S フローで保留状態にありません。STSN 受信側にはシーケンス番号がありません。セッションは継続します。

**11** STSN 送信側は P-S フローで保留状態にありません。STSN は STSN のシーケンス番号に一致しており、TEST POSITIVE を戻します。

## 2 次から 1 次へのフロー・マトリックス

以下の表は、セッションの再同期化時に取られるアクションと、2 次ハーフセッションから 1 次ハーフセッションへの STSN コマンドのフローを示しています。

表の各枠の最初の項目は STSN コマンド処置であり、2 番目の項目は STSN コマンドに戻される応答です。これらは斜線 (/) で区切られます。2 つの STSN が送信された場合にはコンマで区切られます。

表 84. STSN 2 次から 1 次へのフロー

主	2 次、セッション・コールド	2 次、保留中	2 次、コミットの判断	2 次、バックアウトの判断	2 次、無効	2 次、保留中ではない
未受信：保留中	<b>1</b> 設定およびテスト / リセット	<b>2</b> 設定およびテスト / 肯定	<b>3</b> 設定およびテスト / 否定	<b>4</b> 設定およびテスト / 肯定	<b>5</b> 設定およびテスト / 無効	<b>6</b> 設定およびテスト / 肯定
受信済み：保留中	<b>7</b> 設定およびテスト / リセット	<b>8</b> 設定およびテスト / 肯定	<b>9</b> 設定およびテスト / 肯定	<b>10</b> 設定およびテスト / 否定	<b>11</b> 設定およびテスト / 無効	適用できない

### 上記の表に関する注記

**1** STSN 送信側は COLD ではありません。STSN 受信側にはシーケンス番号がありません。セッションは継続します。

**2** STSN 受信側には、保留状態の作業単位があり、ロックを固守し、リソースをコミットしません。SET AND TEST アクション・コードで送信されたシーケンス番号は、STSN 送信側が保留状態の RQ\*2 を受



信しなかったことを示します。このため、STSN 受信側は保留状態の作業単位を最後のコミット・ポイントまでバックアウトし、STSN 応答で TEST POSITIVE を戻します。

**3**

STSN 受信側は保留状態の作業単位をもち、リソースをコミットしてロックを解放することを決定しました。STSN 送信側は、STSN 受信側によって送信された RQ\*2 を受信しませんでした。STSN 受信側は STSN 応答に TEST NEGATIVE を戻し、誤った決定がなされたことを STSN 送信側に知らせます。STSN 送信側は SDT を送信してセッションを継続するか、あるいは UNBIND を送信してセッションの継続を拒否することができます。

**4**

STSN 受信側は保留状態の作業単位をもち、保留状態の作業単位をバックアウトすることを決定しました。STSN 送信側は保留中の作業単位を受け取らなかったため、STSN 受信側は STSN 応答に TEST POSITIVE を戻します。

**5**

STSN 受信側が同期の重大な損失を検出し(可能性として、ログ・データ・セットの不一致など)、STSN 応答に INVALID を戻します。

**6**

STSN 受信側には保留状態の作業単位がなく、STSN で送信されたシーケンス番号に同意するので、TEST POSITIVE で応答します。

**7**

STSN 送信側は COLD ではありません。STSN 受信側にはシーケンス番号がありません。セッションは継続します。

**8**

STSN 受信側には、保留状態の作業単位があり、リソース・ロックを保持しています。STSN 送信側が S-P SET AND TEST で送信した番号は、STSN 送信側が保留作業単位を受信したことを STSN 受信側に示します。このため、STSN 受信側は保留状態の作業単位をコミットし、TEST POSITIVE を応答します。

**9**

STSN 受信側は保留状態の作業単位をもち、保留状態の作業単位をコミットすることを決定しました。STSN 送信側が S-P フローで送信した番号は、STSN 送信側が RQ\*2 を受信し、STSN 応答に TEST POSITIVE が送信されたことを示します。

**10**

STSN 受信側に保留状態の作業単位がありません。受信側はその作業単位をバックアウトし、ロックを解放することを決定しました。STSN 送信側が S-P フローで送った番号は、STSN 送信側が RQ\*2 を受信し処理したことを STSN 受信側に示します。STSN 受信側は、STSN 応答の TEST NEGATIVE によって誤った決定であることを示します。STSN 送信側は SDT を送信してセッションを継続するか、あるいは UNBIND を送信してセッションの継続を拒否することができます。

**11**

STSN 受信側が、同期の重大な損失を検出し(可能性として、ログ・データ・セットの不一致)、STSN 応答の INVALID によって、STSN 送信側にこれを示します。

## STSN コマンドの形式

---

STSN コマンドと STSN 応答の両方が、5 バイトのデータ・フィールドを含んでいます。

STSN コマンドの形式は次のとおりです。

**バイト 0**

アクション・コード

**バイト 1、2**

PHS に送られた最終インバウンド同期点メッセージのシーケンス番号

**バイト 3、4**

PHS から送られた最終アウトバウンド同期点メッセージのシーケンス番号



1次ハーフセッションはアクション・コードを使用して、2次ハーフセッションにVTAMシーケンス番号の検査を依頼します。アクション・コード・バイトのビットは次のとおりです。

#### ビット0および1

インバウンド・シーケンス番号フィールドを参照します。

#### ビット2および3

アウトバウンド・シーケンス番号フィールドを参照します。

#### ビット4から7

予約済み

次の値は、STSN コマンド・アクション・コードのビット0、1、2、3に受け入れ可能です。

#### 00 IGNORE

値を更新しません。IMSは、PHSとしては、このアクション・コード値を送信しません。IMSは、SHSとしては、再同期が必要な時にはこのアクション・コードにINVALID 応答コードを戻し、再同期が必要でない時にはこのアクション・コードにTEST POSITIVE 応答コードを戻します。

#### 01 SET

適切なシーケンス番号を、シーケンス番号フィールドに指示された値に設定します。ISCでは、このコードは、保留状態の作業単位をコミットまたはバックアウトするために一方的決定がなされたことを示すために、PHSが2つ目のSTSNを送信する必要がある場合にのみ発生します。

#### 10 SENSE

値を更新しません。SHSは、コマンド応答にシーケンス番号のバージョンを戻す必要があります。IMSは、SHSとして、このアクション・コードにINVALID 応答を戻します。

#### 11 SET AND TEST

適切なシーケンス番号を、シーケンス番号フィールドに指示された値に設定します。ISCでは、このコードは常に、PHSが送信した最初のSTSNの両方のフローに発生します。SHSは、シーケンス番号値が受け入れ可能であるかどうかをコマンド応答で示す必要があります。

2次ハーフセッションがSTSN コマンドを受信した場合には、次のことができなければなりません。

- SHS アウトバウンド (PHS アウトバウンド) シーケンス番号を検査して、必要な場合には PHS へメッセージを再送できるようにします。
- SHS インバウンド (PHS インバウンド) シーケンス番号を検査して、その番号が SHS が保管した番号と一致しているかどうかを PHS に知らせます。また、直前のセッションからの保留状態の作業単位をコミットするかバックアウトするかについての PHS の一方的な決定に、SHS が同意するか、または同意しないかも知らせます。
- PHS に、DR1 および STSN コマンドへの 5 バイト・データ応答を戻します。

STSN 応答の形式は、STSN コマンドの形式と同じです。

#### バイト0

アクション・コード

#### バイト1、2

SHS が PHS に送信した最後のリカバリー可能メッセージのシーケンス番号

#### バイト3、4

SHS が PHS から受信した最後のリカバリー可能メッセージのシーケンス番号

2次ハーフセッション (SHS) は、アクション・コードを使用してテスト結果を示します。アクション・コードは1次ハーフセッション (PHS) に戻されなければなりません。シーケンス番号 (バイト1から4) を戻すことはオプションです。しかし、リカバリーをデバッグし、問題から再始動するため、SHS で保守されている番号と STSN 番号が一致しない場合は必ず、シーケンス番号を PHS に戻してください。アクション・コード・バイトのビットは次のとおりです。

#### ビット0および1

SHS インバウンド・シーケンス番号を示します。

#### ビット2および3

SHS アウトバウンド・シーケンス番号を示します。

## ビット 4 から 7

予約済み

次の値は、STSN 応答アクション・コードのビット 0、1、2、3 に受け入れ可能です。

### 00 RESET

SHS がコールド・スタートされなければならない、シーケンス番号情報がないことを示すために、SET AND TEST オプションに回答して両方のシーケンス番号フローに戻されます。PHS はセッション開始を継続しなければならない、この回答をエラーとして取り扱ってはなりません。IMS はコールド・スタートおよび STSN に対する返答時にこのコードを送信し、これを受信する場合はセッションを継続します。

### 01 TEST POSITIVE

シーケンス番号が SHS のチェックポイントのシーケンス番号と一致していることを示すために、SET AND TEST オプションに対する応答内に戻されます。SHS が、セッション故障中にアウトバウンド・フローの保留状態の作業単位をコミットまたはバックアウトすることについての PHS の一方的な決定に続くセッション開始の継続に同意する時には、このコードは SET オプションに対する応答に戻す必要があります。

### 10 INVALID

STSN SHS アウトバウンド・シーケンス番号が SHS のチェックポイントのシーケンス番号と一致していないこと(メジャー・セッションの再始動の不一致)を示すために、SET AND TEST オプションに対する応答内に戻されます。INVALID もまた、アウトバウンド・フローの保留状態の作業単位をコミットまたはバックアウトすることについての PHS の一方的な決定に同意しないことを指示するために、SET オプションに対する応答に戻されます。どちらかのフローへの INVALID 応答を受信した場合には、PHS はセッションを継続してはなりません。

### 11 TEST NEGATIVE

シーケンス番号が SHS のチェックポイントのシーケンス番号と一致していないことを示すために、SHS インバウンド・フローの SET AND TEST オプションに対する応答に戻されます。PHS は、不一致の理由が、PHS が送信したけれども SHS が受信しなかった保留状態の作業単位である場合には、SDT を使用して応答します。PHS が保留状態の作業単位をコミットまたはバックアウトする一方的な決定をした場合には、両フローに対する SET オプションを示す 2 つ目の STSN を使用して応答し、あるいは保留状態の作業単位がなく、一方的な決定がなされなかった(メジャー・セッション再始動の不一致)場合には、UNBIND します。SHS が、アウトバウンド・フロー上の保留状態の作業単位をコミットまたはバックアウトする一方的な決定をしたことを示すために、SET AND TEST オプションに対する応答に TEST NEGATIVE を戻すこともできます。この場合 PHS はオプションとして、SET を送信することによりセッション開始を継続することに同意することも、UNBIND を送信してセッション開始を継続しないこともできます。

## 関連タスク

### 506 ページの『保留状態作業単位についての一方的決定の制御』

リソースが誤ってサブシステム間で同期化されないのを防ぐために、片方のハーフセッションが一方的に保留状態の作業単位のコミットまたはバックアウトを決定した場合は、セッション障害の後で、セッション再同期を継続するかどうかを指定することができます。

## IMS 応答モードまたは会話型出力エラーの処理

このトピックでは、ISC セッション中の応答モード・エラーと会話型エラーを IMS がどう処理するか、およびハーフセッションの同期を維持する方法について説明します。

いったん受信され、キューに入れられた(スケジューリングできるようになった) IMS メッセージは、セッション終了またはプロトコルで取り消すことはできません。ただし、入力メッセージがエンキューされたからは、どちらかのハーフセッションでエラーを検出することができます。トランザクションの処理中に、応答メッセージがコミットされる前に IMS がエラーを検出した場合(例えば、DFS555 トランザクション異常終了の発生など)、どちらのハーフセッションもバックアウトすることができますが、その理由は、応答メッセージが出力に利用可能になるまで IMS が入力同期点応答を保持しているからです。例外応答および適切なエラー・リカバリー・プロセス(ERP)メッセージが戻されます。応答メッセージが出力に利用可能になった後でエラーが発生した場合は、(応答メッセージを利用可能にした)最後のアプリケーション同期点にバックアウトします。これらのエラーは、もう一方のハーフセッションには伝えられません。

DFC コマンドに対する例外応答の後では、送受信の前に除去を行わなければならないことがあります。

**関連資料:** DFC プロトコルについて詳しくは、「z/OS Communications Server: SNA Programming」を参照してください。

## 関連資料

528 ページの『データ・フロー制御プロトコルの関連情報』

以下のトピックで、データ・フロー制御 (DFC) のバイト・レベルのプロトコルについて説明します。プロトコルを以下に示します。

538 ページの『ERP ページ』

例外応答の送信後で送信または受信を継続する前に、例外応答の送信側は、ハーフセッションの両方の DFC 状態マネージャが同期化されるまで、エラー・リカバリー・プロセス (ERP) PURGE モードに入れなければならない場合があります。

## 応答モード・エラー

他方のハーフセッションで検出されたエラーは、IMS が応答の送信を試みるまで IMS には伝えられません。

この時点で、要求された同期点応答が戻される前に IMS にエラーを伝える必要がありますが、その理由は、すべての応答モード出力が RQD および EB を示して送信されるからです。エラーは例外応答を戻すことで反映され、またこの例外応答はオプションとして、該当するセンス・データとプロトコルを含む ERP メッセージを持つことがあります。

例外応答または ERP FM ヘッダー・センス・コードは、出力メッセージがバックアウト (デキュー) または再送される原因になるか、あるいはまだメッセージがキューにある状態でセッションが終了する原因になります。ERP メッセージ・プロトコルは、EB の使用を通してセンス・コードに関係なくメッセージがバックアウト (デキュー) されるか、あるいは IMS が CD の使用を通して送信状態のままになる原因となる場合があります。後者の場合は、使用されたセンス・コードによって、出力メッセージがデキューまたは再送される場合にのみ許可されます。

ERP メッセージに CD が使用されており、センス・コードが応答モード出力メッセージをキューから除去する結果となる場合には、セッションの次の出力として、EB を持つ LUSTATUS - abort が送信されます。ただし、いずれの場合も、エラーが他のハーフセッションによって検出される場合には、IMS がバックアウトできる唯一のリソースは応答メッセージだけです。IMS は、データベースの更新および応答モード・トランザクションで開始されるその他のメッセージをバックアウトすることはできません。

## 会話モード・エラー

他方のハーフセッションで検出されたエラーは、IMS が出力応答の送信を試みるまで IMS には伝えられません。

エラーは、要求された同期点応答が完了する前に IMS に伝えなければなりません。非継続会話型出力では、CD を伴う RQE2 を使用して同期点が要求されるか、任意の通常フロー・データが戻されると、同期点は暗黙に完了します。継続会話型出力メッセージは、EB を伴う RQD2 の要求で送信されます。エラーは、例外応答 (例外応答はオプションとして、適切なセンス・データとプロトコルを含む場合がある ERP メッセージを持つ) を戻すことによって、あるいは該当のセンス・コードとプロトコルを持つ LUSTATUS を戻すことによって反映されます。

センス・コードはメッセージのバックアウト (キューから除去) または再送をもたらすか、あるいはメッセージをキューに入れたままセッションを終了させることができます。ERP と LUSTATUS メッセージ・プロトコルは、EB を使用してメッセージをバックアウト (キューから除去) するか、または CD を使用して IMS を送信状態のままにしておくことができます。後者は、センス・コードによってメッセージが再送されるか、あるいはセンス・コードの最後の会話型出力によって、出力メッセージがキューから除去される場合にのみ許可されます。LUSTATUS または ERP メッセージに CD が使用されており、センス・コードによって最後の会話型出力メッセージがキューから除去される場合には、セッションの次の出力として、EB を持つ LUSTATUS - abort が送信されます。

他方のハーフセッションでエラーが検出された場合、IMS が自動的にバックアウトできるリソースは出力応答メッセージだけです。IMS は、データベースの更新、会話型 SPA、および会話型 トランザクションで開始されるその他のメッセージを自動的にバックアウトすることはできません。ただし、会話の場合、内部 IMS /EXIT コマンドはユーザーの会話型異常終了出口ルーチンを呼び出すようにスケジュールされ、

このルーチンはオプションとして、会話型 SPA の内容に基づいて必要なデータベースの変更を元に戻すトランザクションをスケジュールすることができます。出口ルーチンへのインターフェースは、セッションで /EXIT コマンドを受信した場合と同様です。

#### 関連資料

[会話型異常終了出口ルーチン \(DFSCONE0\) \(出口ルーチン\)](#)

## ISC による拡張会話の正常終了

IMS 非 ISC 環境では、IMS 会話型モードの正常終了は、IMS トランザクションが、応答メッセージをコミットする前に会話型 SPA のトランザクション・コード・フィールドをブランクにした場合に起こります。応答メッセージが正常にキューから除去されると、会話は終了します。

ISC では、他のハーフセッションの対等レベルのアプリケーションが同時に会話を終了する可能性があるために、通常の正常終了に拡張が行われました。ただし、IMS が正常終了のこの要求をサポートするのは、SNA 定義のコミット要求、つまり EB を伴う LUSTATUS X'0006' を受信することでのみ行われます。これは、リモート・アプリケーションによる、そのサブシステムに対するスタンドアロンのコミット要求 (または正常終了) により発生します。この形式の会話型終了では、完了した会話型ステップによって変更内容のコミットが遅れる可能性があるため、会話型異常終了出口ルーチン (レジスター 1 のバイト 3 にある新しい入力ベクトル X'28') を使用してユーザーに通知する必要があります。これは、会話型 SPA 内の遅延情報を記録することによって実行できます。出口ルーチンは、新規ベクトルでの呼び出し時に、遅延された変更内容をコミットするために、適切なトランザクションをスケジュールすることができます。

**制約事項:** IMS は、EB を持つ入力メッセージによる会話の終了をサポートしません。

## ハーフセッションの同期の維持

同期点応答 (DR2) は ISC セッション・パートナー間で使用され、両方のパートナーの同期点マネージャーが、リカバリー可能リソースを同期させてコミットまたはバックアウトできることを確認するために使用されます。

IMS ISC セッションで送信または受信されるすべてのメッセージは、メッセージ・タイプによってリカバリー可能またはリカバリー不能のどちらかとして定義されます。ISC セッションの両側が互いに各メッセージに関連したリカバリー可能属性を理解していることを確認するため、セッション応答プロトコルが使用されます。使用される応答プロトコルは IMS メッセージ・タイプと整合性がなければなりません。

ISC 同期点には明示的または暗黙的があります。

#### 明示的

入力が RQD2 応答を要求する。

#### 暗黙的

入力が例外応答 (RQE2) の送信および、トランザクションにより可能になった出力の方向変換を要求するか、または、トランザクションからの出力データの代わりに LUSTATUS の送信を要求する。

保全性とリカバリー可能性をさらに高めるために、情報をロギングすることによって、IMS は同期点機能を補足することができます。ログ先行書き込みの使用によって、同期点指示が確実にログに記録されます。これによって、同期点応答が実際に送信されるかまたは方向変換が (暗黙の同期点に) 返答される前に、IMS 再始動プロシージャで同期点情報が利用可能になります。

次のトピックを読む前に、それらの段落に記載されている ISC 同期点と関連のコミットおよびバックアウト処理の定義、および、それらと IMS アプリケーション・プログラムとの関係を理解しておく必要があります。

#### 関連情報

[VTAM および SNA の参照情報 \(システム・プログラミング API\)](#)

## IMS への入力で要求される同期点

どのような入力のタイプでも、完全な入力メッセージが正常に受信されるまでは、IMS は意図的なトランザクションをスケジュールしません。

送信側で検出されたエラー、IMS 入力の処理によって発生したエラー、および完全な入力メッセージの受取前に発生したセッション障害によって、メッセージ全体が破棄またはバックアウトされます。ただし、完全なメッセージがいったん受信され、スケジュールに利用可能になると、ISC セッション障害またはプロトコルで入力メッセージを取り消すことはできません。

IMS 入力プロセス中または同期トランザクションの実行中に検出されたエラーにより、入力メッセージがバックアウトされた場合には、セッション終了または ISC 入力への例外応答のどちらかによってセッション・パートナーに通知されます。このバックアウトでは次のイベントが発生します。

- バックアウトによって、関連した DFC および ATTACH 状態は最後の同期点の状態にリセットされます。
- トランザクションの実行中のバックアウトは、最後のアプリケーション同期点以降に作成された特急メッセージを除き、アプリケーションの更新とメッセージをバックアウトします。
- アプリケーション同期点と ISC 同期点は必ずしも同一ではありません。
- IMS への入力中のバックアウトは、データベースなどのその他のリカバリー可能 IMS リソースには影響を与えません。これは、入力メッセージが完全に、かつエラーのない状態で受信されるまでは、入力メッセージはスケジューリングや実行に利用できないためです。
- 最後の入力同期点が IMS から要求された後で、複数の連続入力 (リカバリー不能) メッセージが受信され実行された、あるいはスケジューリングのためにキューに入れられた場合でも、バックアウトされるのは現在の入力メッセージだけです。

IMS アプリケーション同期点に対する正常な ISC 入力同期点の定義および関係は、ISC 入力が同期かまたは非同期かによって異なります。非同期入力では、同期点が反映されるのは、IMS がメッセージ・リカバリーにかかわっているということだけです。トランザクションのスケジューリングや実行、またはトランザクション出力が利用できるかということとの関連はありません。メッセージが正常にキューに入れられると、DFC および ATTACH 同期点情報が更新され、要求された同期点応答がセッション・パートナーに戻されます。

同期入力に対する IMS 例外はいくつかありますが、同期点は、IMS トランザクションの実行と同期点が成功したことの反映を意図したものです。IMS は適切に DFC および ATTACH 同期点情報を更新し、すべての関連トランザクション・リソース (例えば、DL/I データベース) をコミットし、また ISC 接続入力同期点要求への応答時であれば、出力トランザクション・メッセージ応答をコミットします。

次の例外は、IMS への同期入力の同期点に適用されます。

- トランザクション挿入応答モードまたは会話型応答メッセージ (入出力 PCB または代替応答 PCB に挿入された最初のメッセージ) がトランザクション同期点を介して出力可能である場合、ISC 同期点応答は、IMS によって戻されます。追加のトランザクション処理および同期点は、ISC セッションに反映されません。トランザクション同期点のための ISC 同期点応答は、プログラム間通信の場合のように、トランザクションが応答メッセージを挿入する前には行われません。入出力 PCB または代替応答 PCB に挿入された 2 番目以降のメッセージ、非応答 PCB に挿入されたメッセージ、および ISC セッション用の IMS 特急メッセージは、応答メッセージを正常に送った後で、その後の非同期送付に備えてキューに入れられます。
- 別の会話型トランザクションへのプログラム間通信中を除き、IMS では、最初に出力会話型応答メッセージを挿入せずにトランザクション同期点を引き起こそうとした、会話型トランザクションの異常終了の結果としてサブシステム・エラー・メッセージを生成します。サブシステム・エラー・メッセージによって、例外応答が入力同期点要求に送信され、入力メッセージがバックアウトされます。

## IMS による出力で要求された同期点

要求された同期点応答が別のセッション・パートナーによって戻される場合、IMS は出力メッセージをコミットします。メッセージは、要求された同期点応答への例外応答で戻されたセンス・コードの結果としてコミットされる場合もあります。

**定義:** コミットという用語は、メッセージが正常に送信され、デキューされ、同期点情報が適切に更新されたことを意味します。



出力同期点応答が保留状態で、セッション故障が発生した場合に、IMS /DEQUEUE コマンドを使用することは、送信出力メッセージをコミットすることを認可された端末オペレーターによる一方的な決定であると見なされます。

同期点要求に例外応答が戻されるか、あるいはIMS が同期点応答を要求する前に、許可された端末オペレーターがIMS /DEQUEUE コマンドを出すと、IMS は出力メッセージを(使用されたセンス・コードにしたがって)バックアウトします。

**定義:** バックアウト という用語は、リカバリー可能メッセージが、その後の再伝送のためにメッセージ・キューに戻される (/DEQUEUE コマンドでキューから除去されない場合) ことを意味します。リカバリー不能メッセージは、エラーのタイプにより、メッセージ・キューから除去されるか、またはその後で再送するためにメッセージ・キューに戻されるかのどちらかです。通常、例外応答またはIMS 障害によってリカバリー不能メッセージはキューから除去されます。サブシステム 障害から発生したのでない一部のセッション障害では、リカバリー不能メッセージは再同期化後の最初の機会に再送されます。バックアウトによって、関連した DFC および ATTACH 状態は最後の同期点の状態にリセットされます。

## 同期点および応答要件

IMS 入出力メッセージ・フローを順次キュー・データ・セットからの入出力のフローとして示すことができます。

このキュー・データ・セットの保全性およびリカバリー可能性を保持するために、IMS では、リカバリー可能な入力および出力メッセージがそれぞれ、フローの継続前に両方のハーフセッション間で同期点確立することが必要です。これによって、両方のハーフセッションの同期点マネージャーが同期のリソースをコミットまたはバックアウトすることができます。

ISC の同期点機能は DR1 と DR2 の要求および応答を再定義し分離します。DR2 要求および応答は同期点要求および応答として認識され、機能的に DR1 と関連した要求および応答から独立しています。

## リカバリー可能メッセージ

リカバリー可能トランザクションが確実に回復できるようにするために、IMS では、送信または受信された各リカバリー可能メッセージに以下の応答プロトコルが必要です。

- MFS がページ化したメッセージでないメッセージ

例外 DR2 (RQE2) は、SNA チェーンの、最後以外の(またはそれが唯一の RU でない場合)各 RU で要求しなければなりません。

方向変換が送信される時を除き、SNA チェーンの最後または唯一の RU はそれぞれ DR2 (RQD2) を要求しなければなりません。方向変換が指示された場合、例外 DR2 (RQE2) または RQD2 を要求できます。

SNA チェーンの最後または唯一の RU への同期点応答は、常にメッセージの終了を示します。メッセージの終了は、単一チェーン(非 MFS ページ化)メッセージのチェーンの終了時に発生します。

- 非同期 (ATTACH SCHEDULER) の要求時ページ化メッセージの最初のチェーン

非同期の要求時ページ化メッセージの最初のチェーン (OIC) は SCHEDULER モデルの ATTACH です。この OIC はブラケット終了の DR2 (RQD2) を要求します。これによって、アプリケーションはメッセージを受信するために非同期にスケジュールされ、セッションはスケジュールされたアプリケーションに割り振られたままの状態になります。

- MFS 要求時ページ化の最後でないページ (ATTACH または ATTACH SCHEDULER を使用しての出力)、および MFS オペレーター論理ページ化 (OLP) 出力の最後のページ

例外 DR1 (RQE1) は、SNA チェーンの最後でない、または唯一でない各 RU (MFS 要求時ページ) で要求されます。

SNA チェーンの最後または唯一の各 RU (MFS 要求時ページ) はそれぞれ、方向変換付きの例外 DR1 (RQE1) を要求します。

- MFS 要求時ページ化(出力)メッセージの最後のページ

例外 DR2 (RQE2) は、SNA チェーンの最後でない、または唯一でない各 RU (MFS 要求時ページ) で要求されます。

方向変換が最後のページで送信される時以外は、SNA チェーン (MFS 要求時ページ) の最後のまたは唯一の各 RU は、それぞれ DR2 (RQD2) を要求します。方向変換付きの例外 DR2 (RQE2) が要求されます。

SNA チェーンの最後または唯一の RU への同期点応答は、常にメッセージの終了を示します。メッセージの終了を確認するために、MFS 要求時ページ化メッセージの最後のページでは必ず同期点を要求する必要があります。

- MFS 自動ページ化メッセージの最初であって最後でないページ

例外 DR1 (RQE1) は、IMS に送信された SNA チェーンの最後でない、または唯一でない各 RU (MFS 自動ページ化) で要求され、IMS が受信した SNA チェーンの最後でない、または唯一でない各 RU (MFS 自動ページ化) で要求する必要があります。

RQD1 は、IMS が送信した MFS 自動ページ化出力メッセージの最初の SNA チェーンの最後または唯一の RU で要求されます。

RQD1 は、他のハーフセッションが 1 次ハーフセッション (ビッド元) として初期設定されている場合には、IMS が受信した MFS 自動ページ化入力メッセージの最初の SNA チェーンの最後または唯一の RU で要求しなければなりません。他のハーフセッションが 2 次ハーフセッション (ファースト・スピーカー) として初期設定されている場合には、例外 DR1 (RQE1) または DR1 (RQD1) のいずれかを、この SNA チェーンの最後または唯一の RU で要求することができます。方向変換は送信されないため、MFS 自動ページ化入力および出力メッセージの最後でないページで要求してはなりません。

- MFS 自動ページ化メッセージの最初でなく、かつ最後でないページ

例外 DR1 (RQE1) は、IMS が送信した SNA チェーン (MFS 自動ページ化) の各 RU (最後または唯一を含む) で要求されます。例外 DR1 (RQE1) は、IMS が受信した SNA チェーンの最後でない、または唯一でない各 RU で要求されなければなりません。例外 DR1 (RQE1) または RQD1 のいずれかを、IMS が受信した SNA チェーンの最後または唯一の各 RU で要求することができます。方向変換は送信されないため、MFS 自動ページ化入力および出力の最後でないページで要求してはなりません。

- MFS 自動ページ化メッセージの最後のページ

例外 DR2 (RQE2) を、IMS が送信した SNA チェーン (MFS 自動ページ化) の最後でない各 RU で要求し、IMS が受信した SNA チェーン (MFS 自動ページ化) の最後でない各 RU で要求する必要があります。

方向変換が最後のページで送信される時以外は、SNA チェーン (MFS 自動ページ化) の最後のまたは唯一の各 RU は DR2 (RQD2) を要求します。方向変換が送信された場合、例外 DR2 (RQE2) が要求されます。

方向変換を最後のページで受信した時以外は、MFS 自動ページ化入力の SNA チェーンの最後のまたは唯一の各 RU は、DR2 (RQD2) を要求する必要があります。方向変換が指示された場合は、例外 DR2 (RQE2) または確定応答 2 (RQD2) を要求できます。

SNA チェーンの最後または唯一の RU への、あるいは LUSTATUS - commit への同期点応答は常にメッセージの終了を示します。メッセージの終了を確認するため、最後のページでは、または自動ページ化メッセージの最後のページの後の LUSTATUS - commit では、必ず同期点を要求しなければなりません。

MFS 自動ページ化入力または出力で方向変換を示していない SNA チェーンの最後または唯一の RU 上で RQE1 を許す場合、セッション・バインドが DEFINITE RESPONSE CHAINS を示す時に、前述のプロトコルへの例外が起きます。確定応答チェーン・パラメーターが IMS ハーフセッション用に設定された場合、定義された最後または唯一の各 RU (方向変換を示さないもの) が、RQD1 または RQD2 を要求して送信されます。RQD1 または RQD2 は、そのセッション・バインドが DEFINITE RESPONSE CHAINS を指示した場合には、その他のハーフセッションによってこれらと同じ条件下で要求する必要があります。方向変換が確定応答チェーン規則のもとで示された場合には、例外および確定応答 (DR1 および DR2) の両方が有効です。

## 関連資料

542 ページの『LUSTATUS プロトコル』



IMS は、以下の図で要約されているように LUSTATUS を送受信します。この図で示された DFC ブラケット、送信 / 受信、および応答要件についての考慮事項は、このトピックの残りの部分で詳細に説明します。

## リカバリー不能メッセージ

IMS は、リカバリー可能性を達成するために必要なすべての処理が省かれる点を除いて、リカバリー不能メッセージをリカバリー可能メッセージと同様に扱います。この結果、リカバリー不能メッセージの処理時間は短縮されますが、障害時には失われる可能性があります。

リカバリー不能な非 MFS 入力および出力メッセージには、オプションとして DR1 と例外 DR1 をそれぞれ DR2 と例外 DR2 の代わりに要求できることを除いて、基本的にはリカバリー可能メッセージと同じ要件があります。リカバリー不能 MFS ページ化入力および出力メッセージには、リカバリー可能メッセージと同じ要件があります。

メッセージの送信時には、一度に 1 つのメッセージだけが未解決になりえます。これは、送信側は 1 つのメッセージだけを送信でき、応答または応答を受け取るまでは別のメッセージを送れないということです。

応答あるいは同期点プロトコルが要求される場合は、ハーフセッションの同期点マネージャーに、メッセージの受け入れ(コミット)時か拒否(バックアウト)時を相互に理解させることでメッセージの保全性を保持できます。また、これにより必要に応じて、SIGNAL RCD を使用して方向変換を要求することができます。これらの機能が自動的であるために、方向変換が示されている場合には、RQE1 または RQE2 が推奨されます。RQD1 または RQD2 は有効ですが、不要な応答によりパフォーマンスが低下することがあります。方向変換および RQE1 または RQE2 を示して送信されたメッセージに対する応答を受信すると、応答または同期点が暗黙に指定されます。すなわち、応答は暗黙の DR1 または DR2 応答ということです。方向変換付きの確定応答 (RQD1 または RQD2) あるいは例外応答 (RQE1 または RQE2) の要求は、セッション・バインド・オプション DEFINITE RESPONSE CHAINS では有効です。

リカバリー可能メッセージの送信と同期点応答(または応答メッセージ)の受信との間に障害が発生することがあります。セッションの再始動プロシージャ中に、STSN コマンドを使用して、送信または受信された最後の同期点メッセージのシーケンス番号が、両方のハーフセッションに通知されます。どちらかのハーフセッションがメッセージを完全に受信していなかった場合には、そのメッセージを再送することができます。

## メッセージ上の同期点標識

以下のトピックでは、IMS の入力および出力メッセージで送信される同期点標識を説明します。

### IMS 入力メッセージでの要求

以下の表は、IMS への入力メッセージ用の応答と同期点要求を要約しています。

表内の「X」は、そのエントリーが IMS によりサポートされることを示します。表内の「S」は、対応するエントリーが推奨されることを示します。

表 85. IMS 入力メッセージ用の応答と同期点要求

入力メッセージのタイプを示す。	メッセージの VTAM 標識							
	RQE1 526 ページの『1』		RQD1		RQE2 526 ページの『1』		RQD2	
	CD	-CD	CD	-CD	CD	-CD	CD	-CD
最後でない MFS ページ (自動ページ化)		S		X				
最後の MFS ページ (自動ページ化)								
リカバリー可能とリカバリー不能					S		X	S

表 85. IMS 入力メッセージ用の応答と同期点要求 (続き)

入力メッセージのタイプを示す。	メッセージの VTAM 標識							
	RQE1 <a href="#">526 ページの『1』</a>		RQD1		RQE2 <a href="#">526 ページの『1』</a>		RQD2	
	CD	-CD	CD	-CD	CD	-CD	CD	-CD
高速機能会話型リカバリー可能とリカバリー不能トランザクション								
応答モード・トランザクション	S		X	X	S		X	X
リカバリー不能、非応答、非会話モード・トランザクション	S	<a href="#">S526 ページの『2』</a>	X	X	S	<a href="#">注 526 ページの『2』</a>	X	X
リカバリー可能、非応答、非会話モード・トランザクション						S	X	S
MFS ページング制御要求: SNA 形式 QMODEL FMH	S							
MFS ページング制御要求: SNA RAP FMH					S		X	
IMS メッセージ通信								
ATTACH SCHEDULER を使用した IMS メッセージ通信					S		X	S
ATTACH (SCHEDULER なし) を使用した IMS メッセージ通信								<a href="#">注 526 ページの『3』</a>
ATTACH SYMSMG <a href="#">526 ページの『4』</a>								<a href="#">注 526 ページの『3』</a>
IMS コマンド	S		X	X	S		X	X
VTAM コマンド <a href="#">526 ページの『5』</a>								
LUSTATUS - コミット					S		X	S
その他	X		X	X				
テスト・モード <a href="#">526 ページの『6』</a>	S		X	X	S		S	X
FMH7 (ERP) メッセージ	X	<a href="#">注 526 ページの『3』</a>	X	<a href="#">注 526 ページの『3』</a>				

注:

1. 方向変換標識への応答をもって応答と見なします。したがって、方向変換を送信する時には RQE1 または RQE2 を推奨します。応答を要求するどちらの方法も、DEFINITE RESPONSE CHAINS のセッション BIND オプションとしてサポートされます。IMS が 2 次システムとして稼働中の場合には、1 次システムとしてのもう一方のハーフセッションは、BB-CD チェーンを送信中の場合、RQD1 に適切な DFC ブラケットと送信 / 受信同期を許可するように指示しなければなりません。
2. EB または BB/EB を要求するリカバリー不能照会入力のみサポートされます。
3. EB で送信されます。BB も必要に応じて送信されます。
4. IMS SYMSMSG はメッセージ通信と同じ扱い方をされるので、同一のプロトコルが適用されます。
5. LUSTATUS は、EB または BB/EB を持つ RQE1 を示すことがあります。CHASE、LUSTATUS、および CANCEL コマンドは、オプションとして、CD の RQE1 を要求できます。LUSTATUS - commit は、特定の状態の RQE2 または RQD2 を要求することもできます。通常の他のフローの VTAM コマンドはすべて RQD1 を要求しなければなりません。
6. テスト"エコー"モードにのみ適用されます。/TEST MFS には適用されません。

### 関連資料

528 ページの『ブラケットおよび半二重プロトコル』

IMS では、SNA ブラケット・プロトコルを使用してコンテンツョンを解決し、半二重プロトコルの方向変換標識を使用して、ブラケット状態である間の通常フローの送信 / 受信モードを制御します。

## IMS 出力メッセージでの要求

以下の表は、IMS 出力メッセージ用の応答と同期点要求を要約しています。

表内の「X」は、対応するエントリーが IMS によりサポートされることを示します。

表 86. IMS 出力メッセージ用の応答と同期点要求

出力メッセージの タイプ	メッセージの VTAM 標識							
	RQE1 <sup>527 ページの『1』</sup>		RQD1		RQE2 <sup>527 ページの『1』</sup>		RQD2	
	CD	-CD	CD	-CD	CD	-CD	CD	-CD
更新、リカバリー可能					X			X
照会、リカバリー可能					X			X
照会、 リカバリー不能	X			X				
高速機能 (リカバリー可能)								X
最後でない MFS ページ (自動ページ化)		注 527 ページの『2』 を参照		注 527 ページの『2』 を参照				
最初のチェーン要求 時ページ化出力								
ATTACH (SCHEDULER なし)	X							
ATTACH SCHEDULER								X

表 86. IMS 出力メッセージ用の応答と同期点要求 (続き)

出力メッセージの タイプ	メッセージの VTAM 標識							
	RQE1 <sup>527</sup> ページの『1』		RQD1		RQE2 <sup>527</sup> ページの『1』		RQD2	
	CD	-CD	CD	-CD	CD	-CD	CD	-CD
最後でない MFS ページ (要求ページ)	X							
最後の MFS ページ (自動ページ化または、OLP なし) リカバリー可能、リカバリー不能					X			X
最後の MFS ページ (OLP あり) <sup>527</sup> ページの『3』	X							
IMS コマンド応答: / DISPLAY、 / RDISPLAY、 / FORMAT					X			X
IMS コマンド応答: / TEST	X							
他の IMS コマンド応答				X				
テスト・モード出力 <sup>528</sup> ページの『4』	X			X				
ブロードキャスト出力 (ATTACH SYSMMSG)					X			X
システム・エラー・メッセージ (ATTACH SYSMMSG)	X			X	X			X
メッセージ通信出力					X			X
VTAM コマンド <sup>528</sup> ページの『5』	X			X				
FMH7 (ERP) メッセージ	X			注 <sup>528</sup> ページの『6』を参照				
QSTATUS (QMODEL) <sup>528</sup> ページの『7』	X				X			X

注:

1. 方向変換標識への応答をもって応答と見なします。したがって、方向変換標識が送信される時には IMS は RQE1 または RQE2 を示します。応答を要求するどちらの方法も、DEFINITE RESPONSE CHAINS のセッション BIND オプションとしてサポートされます。
2. IMS が「RQD 専用」としてバインドされたときに、RQE1 ではなく RQD1 が各チェーンで発生し、それは、エラーまたは競合が受信側で検出された場合には、不要な ERP オーバーヘッドを防ぐために MFS 自動ページ化出力の最初のページに現れます。
3. MFS オペレーター論理ページングは、MOD が PAGE=YES を指定し、自動ページ化出力が MFS システム制御域に示されていない場合に有効です。

4. テスト"エコー"モードにのみ適用されます。 /TEST MFS には適用されません。
5. CHASE には常に RQE1/CD が送信されます。
6. EB で送信されます。
7. CD/RQE1 は QSTATUS で送信され、ATTACH SCHEDULER として送信された出力の MFS DPM 要求時ページ要求における無効カーソルの結果として生じます。CD/RQE2 または EB/RQD2 は QSTATUS で送信され、要求時ページ化出力中の QPURGE の受信によって生じます。CD/RQE2 は次の場合に送信されます。
  - IMS 会話モードで、またはテスト・モードでの操作時
  - ATTACH SCHEDULER を使用して入力送信され、関連する出力のコンポーネントが SINGLE2 または MULT2 の場合その他の場合はすべて EB/RQD2 になります。

#### 関連資料

542 ページの『LUSTATUS プロトコル』

IMS は、以下の図で要約されているように LUSTATUS を送受信します。この図で示された DFC ブラケット、送信 / 受信、および応答要件についての考慮事項は、このトピックの残りの部分で詳細に説明します。

536 ページの『CANCEL プロトコル』

IMS は、SNA **CANCEL** コマンドを送受信します。これにより、送信側は、送信側検出エラーまたは受信側検出エラーの後にすべてのチェーンを送信せずに、出力 SNA チェーンを終了することができます。

528 ページの『ブラケットおよび半二重プロトコル』

IMS では、SNA ブラケット・プロトコルを使用してコンテンションを解決し、半二重プロトコルの方向変換標識を使用して、ブラケット状態である間の通常フローの送信 / 受信モードを制御します。

## データ・フロー制御プロトコルの関連情報

以下のトピックで、データ・フロー制御 (DFC) のバイト・レベルのプロトコルについて説明します。プロトコルを以下に示します。

### BID プロトコル

IMS は SNA BID コマンドを送信しませんが、2 次ハーフセッションとして機能している場合にはこのコマンドを受信します。

BID コマンドをブラケット間状態で受信している場合、IMS は DR1 に応答し、受信状態を保留状態入力に入力します。IMS がイン・ブラケットの時に、BID コマンドが受信された場合には、IMS は例外 DR1 応答 (X'08130000') の付いた BID を拒否します。

### ブラケットおよび半二重プロトコル

IMS では、SNA ブラケット・プロトコルを使用してコンテンションを解決し、半二重プロトコルの方向変換標識を使用して、ブラケット状態である間の通常フローの送信 / 受信モードを制御します。

IMS が別の ISC 論理装置とのセッションにある場合には、次のオプションを使用します。

- ブラケット・リセット状態はデータ通信アクティブ状態で選択することができます。この状態はブラケット間 (BETB) またはイン・ブラケット (INB) にセットすることができます。
- 半二重の送信または受信状態はデータ通信アクティブ状態で選択することができます。
- どちらのハーフセッションもブラケット終了 (EB) を送信できます。IMS は常に EB を許可するようにバインドされていなければなりません。
- 半二重フリップフロップは通常フローの送信 / 受信状態に使用されます。
- ブラケット終了規則 1 (条件付き規則) が使用されます。

#### 関連資料:

- ブラケット終了規則の情報については、「z/OS Communications Server: SNA Programming」を参照してください。

## 関連資料

SLUP と LU 6.1 のバインド・パラメーター (システム・プログラミング API)

## IMS 入力のブラケット・プロトコル

1つのブラケット内に1つ以上の入力メッセージを受信することができます。

以下に例を示します。

- 任意の数の非応答、非会話、非コマンドまたは非テスト・モードの入力メッセージをブラケット内に送信することができます。入力、後続の非同期出力、または LUSTATUS によりブラケットを終了できます。また、単一の非応答、非会話、非コマンド、または非テスト・モードの ATTACH EB の入力メッセージを送信することができます。
- /TEST を除き、上記の同じブラケット内のどの入力メッセージの後にでも、1つの応答モードまたはコマンド・メッセージを、直前の黒丸のリストに示しているオプションとして入れることができます。応答モードまたはコマンド・メッセージ出力によりブラケットが終了します。
- オプションとして、同一ブラケット内で、最初の黒丸のリストに記載した入力メッセージの後に、会話出力またはテスト・モード出力が続く任意の数の会話入力またはテスト・モード入力を開始することができます。ブラケットは最後の会話出力メッセージ (または テスト・モードの /END からのコマンド完了メッセージ) で終了します。ブラケットは、他方のハーフセッションが EB 付き入力 LUSTATUS または CHASE を送信して終了することもできますが、EB 付きで送信された入力データでは終了しません。
- ブラケットは、エラーの結果 FMH7 または LUSTATUS で生じた EB により終了する場合があります。

以下の一連の表は、IMS に対するさまざまなメッセージ・タイプ入力で受け入れ可能なブラケットおよび送信/受信標識を要約しています。

表内の「X」は、対応するエントリーが IMS によりサポートされることを示します。

これらの表で、VTAM 標識は以下の省略形で示されています。

- BB: ブラケット開始
- EB: ブラケット終了
- CD: 方向変換。最終チェーンまたは単独チェーンで許可。
- BB/EB: ブラケット開始および終了。先頭チェーンまたは単独チェーンで許可。

表 87. IMS へのメッセージと共に送信される VTAM ブラケットおよび送信標識/受信標識: ATTACH SCHEDULER を使用した入力メッセージ・タイプ

ATTACH SCHEDULER を使用した入力メッセージのタイプ	BB	EB	BB/EB	BB/CD	CD	-BB、 -EB、-CD
MFS 自動ページ化入力: 先頭ページ (複数ページ入力の)	X					X
MFS 自動ページ化入力: 先頭および末尾でないページ						X
MFS 自動ページ化入力: 末尾ページ: トランザクション・タイプを参照						
非応答トランザクション						
非会話モード・トランザクション	注 530 ページの 『1』	注 530 ページの 『2』	注 530 ペ ージの『1』、 530 ページ の『2』	注 530 ペ ージの『1』	X	X



表 87. IMS へのメッセージと共に送信される VTAM ブラケットおよび送信標識/受信標識: ATTACH SCHEDULER を使用した入力メッセージ・タイプ (続き)

ATTACH SCHEDULER を使用した入力メッセージのタイプ	BB	EB	BB/EB	BB/CD	CD	-BB、 -EB、 -CD
応答モード (高速機能を含む) トランザクション	注 530 ページの『1』			注 530 ページの『1』、 530 ページの『3』	注 530 ページの『3』	X
会話型トランザクション	注 530 ページの『1』			注 530 ページの『1』、 530 ページの『3』	注 530 ページの『3』	X
IMS メッセージ通信	X	X	X	X	X	X
IMS コマンド	X	注 530 ページの『4』	注 530 ページの『4』	注 530 ページの『3』	注 530 ページの『3』	X
テスト・モード中の入力 530 ページの『4』					注 530 ページの『3』	X

注:

1. MFS 自動ページ化入力の最後のページでは無効。
2. MFS 自動ページ化入力の最初のページでは無効。
3. IMS が SNA SIGNAL を使用した方向変換を要求しないように、適したメッセージ終了標識を示します。
4. /DIS、/RDIS、および /FOR コマンドのみで有効。またこれらのコマンドの場合、EB 付きの LUSTATUS -NO-OP (X'0006') を送信して、IMS がブラケット間状態の即時強制をしないようにすることが最適です。これらのコマンドからの出力は常にキューに入れられ、非同期に送信されます。
5. EB を示している FMH7 メッセージおよび LUSTATUS NO-OP は、会話、応答モード、およびテスト・モードの終了を強制し、関連した IMS 出力メッセージがデキューされる原因になります。また、会話型異常終了出口ルーチンが呼び出されます。

表 88. ATTACH (SCHEDULER なし) を使用して IMS への入力メッセージと共に送信される VTAM ブラケットおよび送信標識/受信標識

ATTACH (SCHEDULER なし) を使用した入力メッセージのタイプ	BB	EB	BB/EB	BB/CD	CD	BB、EB、 CD
MFS 自動ページ化入力: 先頭ページ (複数ページ入力の)	X					X
MFS 自動ページ化入力: 先頭および末尾でないページ						X
MFS 自動ページ化入力: 末尾ページ: トランザクション・タイプを参照						
非応答、非会話モード・トランザクション		X	X			
応答モード (高速機能を含む) トランザクション 531 ページの『6』	注 530 ページの『1』			注 530 ページの『1』、 530 ページの『3』	注 530 ページの『3』	X



表 88. ATTACH (SCHEDULER なし) を使用して IMS への入力メッセージと共に送信される VTAM ブラケットおよび送信標識/受信標識 (続き)

ATTACH (SCHEDULER なし) を使用した入力メッセージのタイプ	BB	EB	BB/EB	BB/CD	CD	BB、EB、CD
会話型トランザクション	注 530 ページの『1』			注 530 ページの『1』、 530 ページの『3』	注 530 ページの『3』	X
IMS メッセージ通信		X	X			
ATTACH SYSMMSG		X	X			
IMS コマンド	X	注 530 ページの『4』	注 530 ページの『4』	注 530 ページの『3』	注 530 ページの『3』	X
テスト・モード中の入力 <a href="#">531 ページの『5』</a>					注 530 ページの『3』	X

注：

1. MFS 自動ページ化入力の最後のページでは無効。
2. MFS 自動ページ化入力の最初のページでは無効。
3. IMS が SNA SIGNAL を使用した方向変換を要求しないように、適したメッセージ終了標識を示します。
4. /DIS、/RDIS、および /FOR コマンドのみで有効。またこれらのコマンドの場合、EB 付きの LUSTATUS - NO-OP (X'0006') を送信して、IMS がブラケット間状態の即時強制をしないようにすることが最適です。これらのコマンドからの出力は常にキューに入れられ、非同期に送信されます。
5. テスト"エコー"モードにのみ適用されます。/TEST MFS には適用されません。
6. ISC では、TERMINAL マクロ・ステートメントまたは ETO ユーザー記述子が TRANRESP または FORCRESP のどちらか示す場合は、(TRANSACT マクロ・ステートメントでのこれらの定義に関係なく) 応答モード操作は EB を示さない接続されたトランザクションのために強制されます。TERMINAL マクロ・ステートメントまたは ETO ユーザー記述子が NORESP を示す場合には、入力は拒否されます。

表 89. IMS に送信される他の入力メッセージと共に送信される Vtam ブラケットおよび送信標識/受信標識

その他の入力メッセージのタイプ	BB	EB	BB/EB	BB/CD	CD	BB、EB、CD
VTAM 通常フロー・コマンド/標識 <a href="#">531 ページの『1』</a> 、 <a href="#">532 ページの『2』</a>	X				X	X
FMH7 (ERP) メッセージ		注 530 ページの『5』	X	X	X	
IMS MFS ページング制御要求 (QMODEL FMH)				X	X	
RAP FMH					X	

注：

1. LUSTATUS、CHASE、および CANCEL は EB または CD を示すことができます。その他の VTAM 通常フロー・コマンドまたは標識は EB または CD のどちらも示してはいけません。

IMS が会話型入力を待機中の場合、LUSTATUS および CHASE がブラケット終了を示していない場合には、IMS に送信することはできません。EB は会話型出力メッセージをキューから除去し、会話モード

を終了させ、会話型異常終了出口ルーチン呼び出します。他のプロトコルで IMS が LUSTATUS または CHASE を受信した場合には、セッションが終了します。

2. EB を示している FMH7 メッセージおよび LUSTATUS NO-OP は、会話、応答モード、およびテスト・モードの終了を強制し、関連した IMS 出力メッセージがデキューされる原因になります。また、会話型異常終了出口ルーチンが呼び出されます。

## 関連概念

### 518 ページの『IMS 応答モードまたは会話型出力エラーの処理』

このトピックでは、ISC セッション中の応答モード・エラーと会話型エラーを IMS がどう処理するか、およびハーフセッションの同期を維持する方法について説明します。

## 関連資料

### 528 ページの『データ・フロー制御プロトコルの関連情報』

以下のトピックで、データ・フロー制御 (DFC) のバイト・レベルのプロトコルについて説明します。プロトコルを以下に示します。

## IMS 出力のブラケット・プロトコル

IMS によって使用される出力ブラケットおよび送信/受信プロトコル、およびブラケットごとに送信される出力メッセージ数はさまざまな要因に応じて異なります。

IMS によって使用される出力ブラケットおよび送信/受信プロトコル、およびブラケットごとに送信される出力メッセージ数は以下の 1 つ以上の要因に応じて異なります。

- IMS メッセージ・タイプ
- メッセージが MFS ページ化されるかどうか
- 他のハーフセッションが SCHEDULER モデルをサポートするかどうか
- IMS システム定義中に指定された出力コンポーネント
- 元の入力トランザクションと関連したブラケットおよび ATTACH プロトコル

IMS 内では、MFS ページ化出力の最後でないチェーン (ページ) で使用されるプロトコル、および、応答モード、会話モード、テスト・モード、またはコマンド応答の最後の (MFS ページ化) チェーンまたは唯一のチェーンで使用されるプロトコルは、元の入力トランザクションおよび結果の出力応答が同期か (ATTACH) 非同期 (SCHEDULER) かにかかわらず、前もって定義されます。このことは、ATTACH EB で送信された入力によって生じる結果の出力にも当てはまり、別のハーフセッションに SCHEDULER サポートがないために ATTACH を送信しなければならない、非同期出力にも当てはまります。

CD は、MFS 要求時ページ化メッセージの最後でないすべてのチェーン、テスト・モードの最後の (MFS ページ化) または唯一のチェーン、および最後でない会話型モード・メッセージで使用されます。CD によって、IMS と別のハーフセッション間のページング操作または同期イベントを続行することができます。

EB は、応答モード応答の最後の (MFS ページ化) または唯一のチェーン、最後の会話型モード応答、およびコマンド応答に対して、セッションをリセット状態に置くことによって、同期イベントの終了が確実に起こるようにします。このことは、ATTACH EB で送信される入力によって生じる応答にも当てはまり、別のハーフセッションに SCHEDULER サポートがないために ATTACH を送信しなければならない、非同期出力にも当てはまります。MFS 自動ページ化メッセージの最後でないすべてのチェーン (ページ) は、CD または EB なしで送信されます。

各インストール・システムは IMS に対し、ATTACH SCHEDULER を使用して、他の非同期出力の最後の (MFS ページ化) または唯一のチェーンに使用されるプロトコルを定義しなければなりません。IMS では、セッションに定義されたコンポーネントごとに単一または複数のメッセージの指定ができ、方向変換またはブラケット終了を示す前に、IMS が複数の非同期メッセージを送信すべきかどうかを指定できます。コンポーネントの定義は次のとおりです。

## SINGLE1

このコンポーネントの非同期出力には、ブラケットごとに 1 つのメッセージが送信されます。各メッセージはブラケットを開始し (必要な場合)、常にブラケットを終了します。

## **SINGLE2**

このコンポーネントの非同期出力には、VTAM 開始ブラケットで一度に 1 つのメッセージ (必要な場合) と、受信中のサブシステムでそのトラフィックを任意で送信することができる方向変換標識が送信されます。

## **MULT1**

LTERM のすべての非同期メッセージは、ブラケットの終了前に送信されます。トラフィックには、BB (必要な場合)、メッセージ 1、メッセージ 2、...メッセージ N、EB が送信されます。EB は LTERM の最後のメッセージが確認されデキューされた後で発生します。

## **MULT2**

LTERM のすべての非同期メッセージは、方向変換の送信前に送信されます。トラフィックには、BB (必要な場合)、メッセージ 1、メッセージ 2、...メッセージ N、CD が送信されます。CD は LTERM の最後のメッセージが確認されデキューされた後で行われます。

ISC コンポーネントを非同期メッセージの伝送に使用するためのプロトコルの定義 (IMS システム定義 TERMINAL マクロ・ステートメントまたは ETO ログオン記述子での)、および使用について考慮する必要があります。これは、特に IMS 間の環境で重要です。なぜなら 2 つのサブシステム間のほとんどのメッセージが非同期だからです。これらの ISC プロトコルを誤って定義または使用すると、次のような問題が起きます。

- フローを獲得するため、またはブラケットを終了させるため、余分な伝送が必要になります。
- 不要なブラケット・コンテンション・エラー・リカバリー操作が必要になります。
- 受信サブシステムが受け入れられない出力プロトコルが作成されます。

詳しい説明は次のとおりです。

## **SINGLE1**

SINGLE1 を使用すると、各メッセージで EB が生じることになり、他のサブシステムに送信データがある場合は、ブラケット・コンテンションを引き起こすことがあります。

トランザクションが開始ブラケット内で同期に応答しなければならない場合には、各メッセージ上の EB が他のサブシステム内でプロセス・エラーを起こすことがあります。SINGLE1 コンポーネントを使用して応答モード、会話、またはテスト・モード・メッセージを送信する場合、あるいは IMS サブシステム間で IMS コマンドを送信する場合に、受信サブシステムで、このタイプのエラーが発生することがあります。

## **SINGLE2 または MULT2**

他のサブシステム内で利用可能な出力も応答もない場合は、SINGLE2 として定義されたコンポーネントからのメッセージで送信された方向変換は、ブラケットを終了させるための不要な伝送をもたらします。

## **MULT1 または MULT2**

MULT1 または MULT2 として定義されたコンポーネントで送信されたメッセージは、キューの最後のメッセージがデキューされた後にだけ方向変換またはブラケット終了を示します。したがって、他のサブシステムが同期応答を戻すためにフローに信号を出す必要がある場合には、追加の伝送が発生する場合があります。

MULT1 および MULT2 として定義されたコンポーネントでは、IMS は通常出力メッセージ間で生じるキューの回転を抑制します。この抑制によって、出力のために選択されたキューからのすべてのメッセージを、IMS が他のキューで出力を開始する前に送信することができます。使用可能な出力がある次のキューの選択は、前のキューが空になっているときか、アクティブ出力キューを変更する入力があったときに行われます。LUSTATUS - 空キューかまたは方向変換を示す別の入力と同じキューに対して発生した場合には、キューは回転されません。

以下の表は、IMS 出力メッセージで受け入れ可能なブラケットおよび送信/受信標識を要約しています。

これらの表で、VTAM 標識は以下の省略形で示されています。

## **INB**

同期または非同期出力が、直前の入力と同じブラケット内で発生したことを示します。

## **BETB**

ブラケット間状態にある時に IMS が出力を開始することを示します。

- BB** 開始ブラケット
- EB** ブラケット終了
- CD** 方向変換 (チェーン内の最後またはチェーン内の唯一において可能)
- N** ブラケットなし、または送信 / 受信標識なし

表 90. ATTACH SCHEDULER を使用して出力メッセージと共に送信される VTAM ブラケットおよび送信標識/受信標識

Attach Scheduler を使用して送信されたメッセージのタイプ	SINGLE1 msg.init INB BETB	SINGLE2 msg.init INB BETB	MULT1/2 <sup>534 ページ</sup> の『1』 msg.init INB BETB
MFS 要求時ページ化出力: 最初の SNA チェーン (ATTACH SCHEDULER) <sup>534 ページの『2』</sup>	EBBB/EB	EBBB/EB	EBBB/EB
MFS 要求時ページ化出力: 最後でないページ			
MFS 要求時ページ化出力: OLP では最後のページ <sup>535 ページの『3』</sup>	CD	CD	CD
MFS 要求時ページ化出力: OLP のない 末尾ページ (該当メッセージ・タイプを参照)			
MFS 自動ページ化出力: 最初のページ	NBB	NBB	NBB
MFS 自動ページ化出力: 最初でない、最後でないページ	N	N	N
MFS 自動ページ化出力: 最後のページ (該当メッセージ・タイプを参照)			
応答モード出力 (高速機能を含む)	EB	EB	EB
最後ではない会話型出力メッセージ	CD	CD	CD
最後の会話型出力メッセージ	EB	EB	EB
メッセージ通信出力	EBBB/EB	CDBB/CD	NBB
コマンド出力: /FOR、/DIS、/RDIS	EBBB/EB	CDBB/CD	NBB
コマンド出力: /TEST <sup>535 ページの『4』</sup>	CD	CD	CD
その他のコマンド出力	EB	EB	EB
テスト・モード出力 <sup>535 ページの『4』</sup>	CD	CD	CD
上記のメッセージ・タイプでない - 非同期	EBBB/EB	CDBB/CD	NBB

**注:**

1. LUSTATUS - 空キューは、MULT1 の EB、MULT2 の CD を送信するためにキューの終わりで使用されます。EB は、ERP バックアウトが DFC および ATTACH 状態をブラケット間にリセットし、コンポーネント 1 (COMPT1) が IMS システム定義中に定義されなかった場合に送信されます。また、BETB 列の BB 標識は、最初のメッセージあるいは出力ブラケットを非同期に開始しなければならない MFS ページのみに発生します。
2. 応答モード、高速機能、または会話型要求時ページ化出力用の独立型 ATTACH SCHEDULER を送信した後、IMS はブラケット間にありますが、先行する出力が正常に伝送され、該当のページング要求を使用してデキューされるまで、IMS は入力を受け入れません。

3. MFS オペレーター論理ページング (OLP) は、MOD が PAGE=YES を指定し、自動ページ化出力が MFS システム制御域 (SCA) で指示されていない場合に有効です。
4. テスト"エコー"モードにのみ適用されます。 /TEST MFS には適用されません。

表 91. ATTACH (SCHEDULER なし) を使用して出力メッセージと共に送信される VTAM ブラケットおよび送信標識/受信標識

ATTACH (SCHEDULER なし) を使用して送信されたメッセージのタイプ	SINGLE1 msg.init INB BETB	SINGLE2 msg.init INB BETB	MULT1/2 <sup>534</sup> ページの 『1』 msg.init INB BETB
MFS 要求時ページ化出力: 最初の SNA チェーン (ATTACH)	CD	CD	CD
MFS 要求時ページ化出力: 最後でないページ、OLP では最後のページ <sup>535</sup> ページの 『3』	CD	CD	CD
MFS 要求時ページ化出力: OLP のない 末尾ページ (該当メッセージ・タイプを参照)			
MFS 自動ページ化出力: 最初のページ	N	N	N
MFS 自動ページ化出力: 最初でない、最後でないページ	N	N	N
MFS 自動ページ化出力: 最後のページ (該当メッセージ・タイプを参照)			
応答モード出力 (高速機能を含む) <sup>535</sup> ページの 『5』	EB	EB	EB
最後ではない会話型メッセージ	CD	CD	CD
最後の会話型メッセージ	EB	EB	EB
非応答、非会話型出力 (他のハーフセッションの SCHEDULER モデルが 定義されていない)	EB BB/EB	EB BB/EB	EB BB/EB
ATTACH SYMSG	BB/EB	BB/CD	BB
コマンド出力			
コマンド出力: /TEST <sup>535</sup> ページの 『4』	CD	CD	CD
その他のコマンド出力 (/FOR、/DIS、および /RDIS を除く) <sup>536</sup> ページの 『6』	EB	EB	EB
テスト・モード出力 <sup>535</sup> ページの 『4』	CD	CD	CD

注:

1. LUSTATUS - 空キューは、MULT1 の EB、MULT2 の CD を送信するためにキューの終わりで使用されます。EB は、ERP バックアウトが DFC および ATTACH 状態をブラケット間にリセットし、コンポーネント 1 (COMPT1) が IMS システム定義中に定義されなかった場合に送信されます。また、BETB 列の BB 標識は、最初のメッセージあるいは出力ブラケットを非同期に開始しなければならない MFS ページのみに発生します。
2. 応答モード、高速機能、または会話型要求時ページ化出力用の独立型 ATTACH SCHEDULER を送信した後、IMS はブラケット間にありますが、先行する出力が正常に伝送され、該当のページング要求を使用してデキューされるまで、IMS は入力を受け入れません。
3. MFS オペレーター論理ページング (OLP) は、MOD が PAGE=YES を指定し、自動ページ化出力が MFS システム制御域 (SCA) で指示されていない場合に有効です。
4. テスト"エコー"モードにのみ適用されます。 /TEST MFS には適用されません。
5. 応答モード操作は、直接接続された入力トランザクションに対しては常に行われます。

6. /DISPLAY、/RDISPLAY、および /FORMAT コマンド出力 について詳しくは、542 ページの『LUSTATUS プロトコル』を参照してください。

表 92. 他出力メッセージと共に送信される VTAM ブラケットおよび送信標識/受信標識

その他の出力メッセージのタイプ	SINGLE1 msg.init INB BETB	SINGLE2 msg.init INB BETB	MULT1/2 <sup>534</sup> ページの 『1』 msg.init INB BETB
VTAM コマンド/標識 <sup>536</sup> ページの『2』			
FMH7 (ERP) メッセージ: 会話中	CD	CD	CD
FMH7 (ERP) メッセージ: テスト・モード中	CD	CD	CD
FMH7 (ERP) メッセージ: その他 <sup>536</sup> ページの『3』	EB BB/EB	CD BB/CD	注 <sup>536</sup> ページの 『4』
QSTATUS (QMODEL)	注 <sup>536</sup> ページ の『5』	注 <sup>536</sup> ページ の『5』	注 <sup>536</sup> ページの 『5』

注:

- LUSTATUS - 空キューは、MULT1 の EB、MULT2 の CD を送信するためにキューの終わりで使用されます。EB は、ERP バックアウトが DFC および ATTACH 状態をブラケット間にリセットし、コンポーネント 1 (COMPT1) が IMS システム定義中に定義されなかった場合に送信されます。また、BETB 列の BB 標識は、最初のメッセージあるいは出力ブラケットを非同期に開始しなければならない MFS ページのみに発生します。
- BIS は CD または EB なしで送信されます。CHASE は常に CD を受信します。
- FMH7 は、ERP 操作のために DFC および ATTACH 状態がバックアウトになった後、最後にコミットされた入力コンポーネントの結果のブラケット間リセット状態と関連したプロトコルを要求します。
- FMH7 は MULT1 の EB、MULT2 の CD で送信されます。
- CD/RQE1 は出力の MFS DPM 要求時ページ要求における無効カーソルによって生じる QSTATUS で送信されます。CD/RQE2 または EB/RQD2 は QSTATUS で送信され、要求時ページ化出力中の QPURGE の受信によって生じます。CD/RQE2 は次の場合に送信されます。1) IMS 会話またはテスト・モードの場合 または 2) 入力が ATTACH SCHEDULER で送信され、関連した出力コンポーネントが SINGLE2 または MULT2 である場合。その他の場合はすべて EB/RQD2 になります。

#### 関連資料

659 ページの『ISC データ・フロー制御の例』

以下のトピックでは、ISC データ・フロー制御の例を示します。

542 ページの『LUSTATUS プロトコル』

IMS は、以下の図で要約されているように LUSTATUS を送受信します。この図で示された DFC ブラケット、送信 / 受信、および応答要件についての考慮事項は、このトピックの残りの部分で詳細に説明します。

536 ページの『CANCEL プロトコル』

IMS は、SNA **CANCEL** コマンドを送受信します。これにより、送信側は、送信側検出エラーまたは受信側検出エラーの後にすべてのチェーンを送信せずに、出力 SNA チェーンを終了することができます。

## CANCEL プロトコル

IMS は、SNA **CANCEL** コマンドを送受信します。これにより、送信側は、送信側検出エラーまたは受信側検出エラーの後にすべてのチェーンを送信せずに、出力 SNA チェーンを終了することができます。

送信側または受信側検出エラーのために送信される **CANCEL** コマンドは、IMS により、以下の表に示される VTAM 標識と共に送信されます。

送信側または受信側検出エラーのために IMS が受信する **CANCEL** コマンドは、メッセージの送信側に適切な (現行 DFC 状態と整合性がある) どのプロトコルも指定することができます。



この表で、**CANCEL** コマンドと共に送信される VTAM 標識は以下の省略形で示されます。

- EB: ブラケット終了
- CD: 方向変換

表 93. CANCEL コマンドで送信される VTAM 標識

出力 (次の順序で判別される)	-CD または -EB	CD	EB
/DEQ 後 (LUSTATUS ABORT が次)	X		
1 次ハーフセッションと FIC/OIC が ブラケット間の場合 (条件付き BB)	X		
選択受信側 ERP センス・コード		X	
SIGNAL 受信後		X	
会話型または応答モード出力が まだ利用可能な場合	X		
FIC/OIC が EB RQD* (条件付き EB) である場合			X
テスト (エコー) モード中		X	
SINGLE1 コンポーネントの非 MFS 要求時ページ化出力	FIC/OIC 用標識がブラケット間にあるか FIC/OIC が EB であることを確認すること		
SINGLE1 コンポーネントの MFS 要求時ページ化出力			X
SINGLE2 コンポーネントの出力		X	
MULT1 または MULT2 コンポーネントの出力	X		

## プロトコルのチェーン

1 次ハーフセッション、または 2 次ハーフセッションのいずれかで操作するかにかかわらず、IMS は、単一と複数の両方の RU チェーンを送信します。ハーフセッションはバインド・パラメーターでの指定にしたがって、単一または複数のどちらかの RU チェーンで操作できます。

IMS メッセージは通常、単一の SNA チェーンとして送信および受信され、ここでは単独チェーン (OIC) または先頭チェーン (FIC) はメッセージの開始を示し、OIC またはチェーン内での最後 (LIC) はメッセージの終了を示します。ただし、MFS ページ化入力または出力メッセージの各ページは、単一の SNA チェーンとして送信または受信されます。メッセージの開始は最初のページが送信または受信された時に起こり、メッセージの終了は、最後の入力または出力ページ (MFS オペレーターの論理ページングを除く) の終了で要求された同期点 (RQD2 または RQE2, CD) で起こります。

オプションとして、メッセージの終了は、MFS DPM 自動ページ化入力の最後のページの直後の LUSTATUS - commit を使用して要求することができます。

複数の RU 入力チェーンでエラーが検出された場合には、CANCEL コマンドを使用してください。

### 関連資料

542 ページの『LUSTATUS プロトコル』

IMS は、以下の図で要約されているように LUSTATUS を送受信します。この図で示された DFC ブラケット、送信 / 受信、および応答要件についての考慮事項は、このトピックの残りの部分で詳細に説明します。

538 ページの『ERP パージ』



例外応答の送信後で送信または受信を継続する前に、例外応答の送信側は、ハーフセッションの両方の DFC 状態マネージャーが同期化されるまで、エラー・リカバリー・プロセス (ERP) PURGE モードに入れなければならない場合があります。

## CHASE プロトコル

IMS は SNA CHASE を送信および受信します。

IMS は例外応答の受信後、同期イベントを起こすために RQE1/CD を要求する CHASE を送信しますが、この時 IMS は MFS 自動ページ化出力のページ間 (チェーン) にあり、確定応答チェーン専用でバインドされていません。

CHASE の受信時、IMS は必要に応じて DR1 で応答します。IMS はブラケット内の任意の 2 つのメッセージの間に CHASE を受信することができます。IMS は、MFS 自動ページ化入力に例外応答を送信した後に、同期化イベントとして CHASE を受信することもできます。(MFS 自動ページ化入力の送信側は、出力ページ間にある時または RQE1 を示すチェーンの後、例外応答を受信します。)

EB を伴う CHASE だけが、IMS が会話型入力の待機中に受け入れられます。EB によって会話型出力メッセージはキューから除去され、会話モードが終了し、会話型異常終了出口ルーチンが呼び出されます。これは /EXIT コマンドを受信した場合と同様に行われます。EB 以外のプロトコルの CHASE が受信された場合には、セッションが終了します。

## ERP パージ

例外応答の送信後で送信または受信を継続する前に、例外応答の送信側は、ハーフセッションの両方の DFC 状態マネージャーが同期化されるまで、エラー・リカバリー・プロセス (ERP) PURGE モードに入れなければならない場合があります。

また、ERP PURGE は FMH7 ERP メッセージの送信以前に完了しなければなりません。ERP パージは単一または複数 SNA チェーンのどちらでも起こせます。単一のチェーン・パージは DFC 状態が例外応答を受信する同じチェーン内で同期化される場合に行われます。複数のチェーン・パージは、DFC 状態マネージャーが同期化される前に、複数の SNA チェーンをパージしなければならない場合に行われます。(CANCEL コマンドおよび CHASE コマンドは、例外応答の原因となった SNA チェーンの直後に実行される場合には、論理的に同一チェーンの一部と見なされます。)

単一チェーンおよび複数チェーンのパージは、データ・フロー制御サポート層内で行われ、MFS などの SNA 表示層機能からは独立しています。

ハーフセッションは、ブラケットの終了または同期化されている両方のハーフセッションの DFC 状態の原因ではない RU (OIC、MIC、または LIC) に例外応答を送信する場合に、ERP PURGE を入力しなければなりません。つまり、ERP PURGE は、ハーフセッションが以下の RU に例外応答を送信したときに開始されます。

FIC RQE\* (EB には適用されない)  
MIC RQE\* (CD または EB には適用されない)  
LIC RQE\* -EB および -CD  
OIC RQE\* -EB および -CD

ERP PURGE は、パージを行っているハーフセッションが、ブラケットを終了する RU、または両方のハーフセッションの DFC 状態を再同期化させる RU を受信すると終了します。つまり ERP PURGE は、パージを行っているハーフセッションが次の RU を受信すると終了します。

### **OIC RQD\***

これには CANCEL および CHASE コマンドが含まれ、CD または EB には適用されません。

### **OIC RQE\* CD または EB**

これには CANCEL コマンドが含まれます。

### **LIC RQD\***

CD または EB には適用されません。

### **LIC RQE\* CD または EB**

ERP PURGE は終了する前に複数の SNA チェーンにまたがることができます。複数のチェーンが除去される場合には、ブラケット終了時点または DFC 状態の同期化 (CANCEL または CHASE コマンド以外) の時点で、さらに例外応答が必要になります。これにより、送信側ハーフセッション (応答受信側) によるメッセージの後処理について、あいまいに、または誤って理解することを防ぐことができます。この送信側 ERP 例外応答 (X'0867') は、直前のチェーンで発生したエラーによりこのチェーンが除去されたこと、および、メッセージの送信側が次の可能な機会にメッセージを送信し直す必要があることを示しています。X'0867' 例外応答は、ハーフセッションが、元の例外応答の原因となった RU の後で、次に示すチェーンの RU を受信した後で、ERP PURGE を終了する場合に、送信されなければなりません。

**OIC RQD\***

これには CANCEL および CHASE は含まれず、また CD または EB には適用されません。

**OIC RQE\* CD または EB**

CANCEL 用を除く

**LIC RQD\***

CD または EB には適用されません。

**LIC RQE\* CD または EB**

FMH7 ERP メッセージと共に受信することができる DFC プロトコルは、方向変換またはブラケット終了のどちらかでなくてはならず、適切な場合、ブラケット開始を含むこともできます。IMS によって FMH7 ERP メッセージで送信される DFC プロトコル (CD または EB) は、ERP バックアウト後に生じる ATTDSP 値によって判別されます。バックアウトによりブラケット間状態になる場合には、ATTDSP 値はコンポーネント 1 です。バックアウトによりブラケット間以外の状態になる場合には、ATTDSP 値は最後にコミットされた入力コンポーネントです。プロトコルは結果のコンポーネント用にセットされます。ブラケット終了は、ERP バックアウトが DFC および ATTACH 状態をブラケット間にリセットし、コンポーネント 1 (COMPT1) が IMS システム定義中に定義されなかった場合に送信されます。

**関連資料**

532 ページの『IMS 出力のブラケット・プロトコル』

IMS によって使用される出力ブラケットおよび送信/受信プロトコル、およびブラケットごとに送信される出力メッセージ数はさまざまな要因に応じて異なります。

**送信側 ERP パージ後の結果の DFC 状態**

以下の表は、両方のハーフセッションが、選択受信側 ERP 以外の送信側 ERP センス・コードを示す例外応答の後に、同期点に達した結果生じる有効なブラケットおよび送信/受信状態をすべて示しています。

この表では、以下の省略形が使用されます。

**BETB**

ブラケット間

**INB**

イン・ブラケット

**PHS**

1 次ハーフセッション

**SHS**

2 次ハーフセッション

表 94. 送信側 ERP パージ後の結果の DFC 状態

エラー内のチェーンのタイプ	PHS がデータを送信、SHS が例外を送信: ハーフセッションの状態		SHS がデータを送信、PHS が例外を送信: ハーフセッションの状態	
	PHS	SHS	PHS	SHS
BB/- FIC				
RQ** CD OIC,LIC	540 ページの『1』	540 ページの『1』	INB.SEND	INB.RCV

表 94. 送信側 ERP ページ後の結果の DFC 状態 (続き)

エラー内のチェーンのタイプ	PHS がデータを送信、SHS が例外を送信: ハーフセッションの状態		SHS がデータを送信、PHS が例外を送信: ハーフセッションの状態	
	PHS	SHS	PHS	SHS
RQD* OIC,LIC	BETB	BETB	INB.RCV	INB.SEND
RQD* CANCEL	BETB	BETB	INB.RCV	INB.SEND
RQ** CD CANCEL	<a href="#">540 ページの『1』</a>	<a href="#">540 ページの『1』</a>	INB.SEND	INB.RCV
RQD* EB CANCEL	<a href="#">540 ページの『1』</a>	<a href="#">540 ページの『1』</a>	BETB	BETB
BB/EB FIC				
RQE* OIC,LIC	BETB	BETB	BETB	BETB
RQD* OIC,LIC	BETB	BETB	BETB	BETB
RQD* CANCEL	BETB	BETB	BETB	BETB
RQ** CD CANCEL <sup>1</sup>				
RQD* EB CANCEL <sup>1</sup>				
--/-- FIC				
RQE* OIC,LIC	INB.SEND	INB.RCV	INB.RCV	INB.SEND
RQ** CD OIC,LIC	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* OIC,LIC	INB.SEND	INB.RCV	INB.RCV	INB.SEND
RQD* CANCEL	INB.SEND	INB.RCV	INB.RCV	INB.SEND
RQ** CD CANCEL	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* EB CANCEL	BETB	BETB	BETB	BETB
--/EB FIC				
RQ** CD OIC,LIC	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* OIC,LIC	INB.SEND	INB.RCV	INB.RCV	INB.SEND
RQD* CANCEL	INB.SEND	INB.RCV	INB.RCV	INB.SEND
RQ** CD CANCEL	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* EB CANCEL	BETB	BETB	BETB	BETB

1. CD および EB は、 DFC BETB 状態の時には送信されないことがあります。

## 選択受信側 ERP パージ後の結果の DFC 状態

以下の表は、両方のハーフセッションが、SNA 選択受信側 ERP センス・コードを示す例外応答の後で、かつエラーを検出したハーフセッションが FMH7 を送信する前に同期点に達した結果生じる、有効なブラケットおよび送信/受信状態をすべて示しています。

表 95. 選択受信側 ERP パージ後の結果の DFC 状態

エラー内のチェーンのタイプ	PHS がデータを送信、SHS が例外を送信: ハーフセッションの状態		SHS がデータを送信、PHS が例外を送信: ハーフセッションの状態	
	PHS	SHS	PHS	SHS
BB/-- FIC				
RQ** CD OIC,LIC	541 ページの 『1』	541 ページの 『1』	INB.SEND	INB.RCV
RQD* OIC,LIC	ERP.RCV	ERP.SEND	INB.SEND	INB.RCV
RQD* CANCEL	ERP.RCV	ERP.SEND	INB.SEND	INB.RCV
RQ** CD CANCEL	541 ページの 『1』	541 ページの 『1』	INB.SEND	INB.RCV
RQD* EB CANCEL	541 ページの 『1』	541 ページの 『1』	ERP.SEND	ERP.RCV
BB/EB FIC				
RQE* OIC,LIC	ERP.RCV	ERP.SEND	ERP.SEND	ERP.SEND
RQD* OIC,LIC	ERP.RCV	ERP.SEND	ERP.SEND	ERP.SEND
RQD* CANCEL	ERP.RCV	ERP.SEND	ERP.SEND	ERP.SEND
RQ** CD CANCEL				
RQD* EB CANCEL				
--/-- FIC				
RQE* OIC,LIC	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQ** CD OIC,LIC	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* OIC,LIC	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* CANCEL	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQ** CD CANCEL	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* EB CANCEL	ERP.RCV	ERP.SEND	ERP.SEND	ERP.RCV
--/EB FIC				
RQ** CD OIC,LIC	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* OIC,LIC	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* CANCEL	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQ** CD CANCEL	INB.RCV	INB.SEND	INB.SEND	INB.RCV
RQD* EB CANCEL	ERP.RCV	ERP.SEND	ERP.SEND	ERP.RCV

注:

1. CD および EB は、ブラケット間状態にある間は送信できません。

## LUSTATUS プロトコル

IMS は、以下の図で要約されているように LUSTATUS を送受信します。この図で示された DFC ブラケット、送信 / 受信、および応答要件についての考慮事項は、このトピックの残りの部分で詳細に説明します。

以下の表にリストされていない LUSTATUS センス・コードはすべて、ISC セッションを終了させます。表内の S は、該当の LUSTATUS に推奨される 標識設定を示しています。表内の X は、IMS は、指定された 標識設定も該当の LUSTATUS 用にサポートすることを示します。

表 96. LUSTATUS で送信される VTAM 標識

LUSTATUS	LUSTATUS の VTAM 標識								
	RQE1		RQD1			RQE2	RQD2		
	EB	CD	--	CD	EB	CD	--	CD	EB
受信センス・コード 542 ページの『1』: コミット -X'0006'						S	S	X	S
受信センス・コード 542 ページの『1』: ノーオペレーション -X'0006'	X	S		X	S				
受信センス・コード 542 ページの『1』: 空キュー -X'0007'		S		X	S				
機能異常終了 -X'0864'		S	S	X	S				
機能異常終了 -X'0865'		S	S	X	S				
機能異常終了 -X'0866'		X	X	X	X				
送信センス・コード: コミット -X'0006'									
送信センス・コード: ノーオペレーション -X'0006'		X			X				
送信センス・コード: 空キュー -X'0007'		X			X				
機能異常終了 -X'0865'		X	X		X				

### 注:

- IMS が会話型入力を待っている間、これは EB のみを伴う LUSTATUS を受け入れます。EB は会話型出力メッセージをキューから除去し、会話モードを終了させ、会話型異常終了出口ルーチンをスケジュールします。その他のプロトコルをもつ LUSTATUS が受信された場合には、セッションは終了します。

IMS は、LUSTATUS RQD1 または RQD2 を受信した時に DR1 または DR2 に応答します。前の表で示したように、IMS は、DR1、DR2、または例外 DR1 か DR2 を要求します。LUSTATUS に例外が発生した場合、セッションは終了します。

- LUSTATUS - 空キュー。

IMS は LUSTATUS - 空キューを送信し、受信します。

IMS は次の場合に LUSTATUS - 空キューを送信します。

- 非同期 (ATTACH SCHEDULER) 入力メッセージを受信し、すぐに使用可能な出力がない場合。
- 与えられたキュー上の使用可能なすべての出力を送信した場合。

IMS が会話または応答モードではなく、イン・ブラケット / SEND のままであり、SINGLE1 または MULT1 として定義されたコンポーネントから送信可能な出力がない場合、送信された LUSTATUS 上で、RQD1 およびブラケット終了が示されます。RQE1 および方向変換は、IMS が会話または応答モードではなく、

イン・ブラケット /SEND のままで、かつ SINGLE2 または MULT2 として定義されたコンポーネント から送信可能な出力がない場合に送信された LUSTATUS 上で指示されます。

- LUSTATUS - 機能打ち切り。サポートされるセンス・コードは次のとおりです。

**X'0864'**

再実行でループが発生する。送信側は同じデータを再度送信してはなりません。

**X'0865'**

データ送信側はループの検出と防止に責任があります。

**X'0866'**

データ受信側はループの検出と防止に責任があります。

次の状態でのみ IMS は LUSTATUS - 機能打ち切り X'0865' を送信します。

- MFS 要求または自動ページ化出力メッセージの最後のページを IMS が送信する前に、認可された IMS 端末オペレーターが IMS /DEQUEUE コマンドを発行した場合。

LUSTATUS 上で示された DFC ブラケットおよび送信 / 受信プロトコルは、メッセージを正常に送信した最後のチェーン (ページ) のものと同じになります。

IMS が LUSTATUS - 機能打ち切りを受信した場合に、次のことが行われます。

- 会話型モード中を除き、RQE/CD を送信した出力メッセージ (MFS 要求時ページ化または自動ページ化出力の最後のページを含む) の後に受信された機能打ち切りはすべて、出力メッセージをコミットし、セッション終了を引き起こしません。LUSTATUS コマンドとして処理される前に通常フロー入力として処理される LUSTATUS もメッセージのコミットの原因となります。
- 機能打ち切りによって、不完全な入力 MFS 自動ページ化メッセージは破棄されます。
- 活動出力 MFS 要求時ページ化メッセージの最後のページが送信される前に機能打ち切り X'0864' を受信すると、出力が会話型であり要求時ページである場合には、正常な入力 / 出力操作を継続する前に、メッセージはキューから除去されます。セッションで /EXIT コマンドを受信したのと同様に、会話型異常終了出口ルーチンが呼び出されます。
- アクティブ出力 MFS 要求時ページ化メッセージの最後のページが送信される前に受信された機能打ち切り X'0865' によって、メッセージが出力メッセージ・キューに戻され、セッションが終了することになります。
- アクティブ出力 MFS 要求時ページ化メッセージの最後のページが送信される前に受信された機能打ち切り X'0866' によって、メッセージがメッセージ・キューに戻され、正常な入力 / 出力操作を継続する前に再送が可能になります。
- 機能打ち切りは ATTACH 状態を最後の同期点で有効な状態にリセットします。

LUSTATUS - 機能打ち切りは、RQD2 または RQE2 を示す RU に対する応答で IMS に送信されることはありません。送信されると、セッションは終了します。

- LUSTATUS - コミット

IMS は LUSTATUS - コミットを送信しません。自動ページ化入力の最後のページの直後のメッセージ終了指示として、会話型入力を待機中の EB 付きのメッセージ終了指示として IMS は LUSTATUS - コミットを受信します。これにより、X'28' の新しい入力ベクトルを使用して会話型異常終了出口ルーチンが呼び出され、会話の「正常」終了が作成されます。これによって、出口ルーチンは、スクラッチパッド区域 (SPA) に反映された保留状態のリソースをコミットするトランザクションをスケジュールすることができます。

- LUSTATUS - NO-OP

IMS は LUSTATUS - NO-OP を送信し、受信します。

IMS は LUSTATUS - NO-OP を送信します。

- SIGNAL RCD 後に、RQE1 および方向変換 (CD) のフローを可能にする場合。
- 出力を生成しない同期入力メッセージを受信後にブラケットを終了する場合。IMS では、同期的に接続された場合に FORMAT、/RDISPLAY、および /DISPLAY コマンドでのみこれが行われます。LUSTATUS - NO-OP は RQD1 および EB を示して送信されます。

- イン・ブラケット /SEND がバインドされ、同期出力または再始動が可能でない時にブラケットを終了する場合。つまり、IMS はセッションの再始動後には会話または応答モードではありません。LUSTATUS が、RQD1 および EB を示して送信されます。
- セッション再始動でイン・ブラケット /SEND でバインドされ、保留状態の会話型メッセージが STSN プロトコルを使用してキューから除去される際に、必要な入力を可能にする場合。LUSTATUS は RQE1 および CD を示して送信されます。
- 応答モードまたは IMS 会話の最後の出力メッセージに続いて、例外または ERP FMH7 (EB なし) センス・コード X'0864' を受信した後に、ブラケットを終了する場合。

LUSTATUS - NO-OP は同期または非同期入力処理中に受信でき、必要に応じて CD または EB のどちらかを示します。IMS 会話または応答モード中、あるいはテスト・モード時に LUSTATUS EB を受信すると、IMS 会話、応答、またはテスト・モードを終了させ、関連した出力メッセージがキューから除去されます。会話型の場合、セッションで /EXIT コマンドを受信したのと同様に、会話型異常終了出口ルーチンが呼び出されます。

IMS が 2 次ハーフセッションの場合には、1 次ハーフセッションが送信した BID コマンドに対して DR1 で応答した後、LUSTATUS BB/EB RQ\*1 の特殊なケースを受信します。この LUSTATUS は、前述の DR1 が送信され、1 次ハーフセッションに送信可能なその他のメッセージがなくなった後で送信された DFC イン・ブラケット / 保留状態から、IMS (2 次) を除去します。この状態は、BID の送信を起こした 1 次ハーフセッション内のアプリケーションが、BID の送信と BID 応答の受信の間に異常終了した場合などに起こることがあります。

LUSTATUS-CD の後には別の LUSTATUS-CD を続けてはなりません。そのようにすると、ハーフセッションの間に前後 (back-and-forth) 効果が作り出されます。LUSTATUS-CD を受信し、送信可能な出力がない場合は、IMS は常に LUSTATUS-EB に応答します。

#### 関連概念

518 ページの『IMS 応答モードまたは会話型出力エラーの処理』

このトピックでは、ISC セッション中の応答モード・エラーと会話型エラーを IMS がどう処理するか、およびハーフセッションの同期を維持する方法について説明します。

## ページ化メッセージ ERP

IMS 複数チェーン要求または自動ページ化出力メッセージの伝送が開始された後で送信側検出エラー (/DEQ コマンド) が検出された場合は、IMS は (必要な場合) CANCEL コマンドを LUSTATUS - 打ち切りと一緒に送信しなければなりません。

CANCEL コマンドは 1 つの複数 RU チェーン、またはページを終了するのに使用され、一方 LUSTATUS は複数チェーンまたはページ化メッセージの受信プロセスを終了するのに使用されます。入力自動ページ化メッセージ中に、次の入力 RU 上で後続の LUSTATUS 打ち切りを受信せずに IMS が CANCEL を受信した場合には、IMS は、打ち切りセンス・コード X'0865' および適切な ERP メッセージ付きの FMH7 が後に続く選択受信側 ERP を示す例外応答を戻します。チェーン間にある時にエラーが検出された場合に、送信または受信が必要なのは LUSTATUS - 打ち切りだけです。

自動ページ化入力メッセージの最初のページで検出されたエラーにより、IMS は、コンテンション・センス・コードか、あるいは非機能打ち切りセンス・コードの 1 つおよび適切なエラー・メッセージを示す FMH7 の前の選択受信側 ERP センス・コードのどちらかを送信することになります。自動ページ化入力メッセージの最初でないページで検出されたエラー、または IMS 要求ページ化出力メッセージ中に受信された入力ページ要求により、IMS は、X'08650000' 機能打ち切りセンス・コードおよび適切なエラー・メッセージを示す FMH7 が後に続く選択受信側 ERP センス・コードを送信することになります。機能打ち切りセンス・コードは、単一のチェーンまたはページではなく、(自動ページ化入力または要求時ページ化要求の) 送信プロセスを終了する場合に使用されます。



## 受信準備済みプロトコル

IMS は SNA 受信準備済み (RTR) コマンドを送信しませんが、1 次ハーフセッションとして活動中の場合にはこのコマンドを受信します。

RTR コマンドの受信時には、出力が使用不能または送信不能であれば、IMS は使用可能な出力メッセージが後に続く DR1、あるいは例外 DR1 応答 (X'08190000') のいずれかで応答します。

## RSHUT プロトコル

IMS は、RSHUT コマンドを送信しませんが受信します。

RSHUT の受信時、IMS は DR1 応答を送信し、次の適当な時点でセッションの終了をスケジュールします。セッションは通常、現行の入力または出力のチェーンの終わりで終了します。

## 選択受信側 ERP

選択受信側のエラー・リカバリー・プロシージャ (ERP) が X'08460000' 送信側 ERP センス・コードを介して開始されます。

このプロシージャには次のような特性があります。

- プロシージャは両方のハーフセッションに対して対称です。
- 選択受信側 ERP を指示する例外応答の送信側によって ERP メッセージを送信することができます。ERP メッセージは IMS エラー・メッセージの前の ERP FMH7 ヘッダーです。各選択受信側 ERP メッセージの前には、ERP FMH7 機能管理ヘッダーがあります。
- ERP メッセージのコンテンションは例外応答送信側では起こりません。
- ERP メッセージを任意の長さのチェーン全部で構成することができますが、選択受信側 ERP を指示する例外応答の直後になければなりません。
- ERP メッセージは特殊処理として受信側によって認識されます。
- 応答送信側には機能管理レベル・ブラケットおよび送信 / 受信状態をリセットする機会があります。

IMS によって送信された ERP メッセージには、関連したデフォルト IMS メッセージ出力記述子 (MOD) およびディスプレイ出力形式 (DOF) があります。エラー・メッセージは、MFS で定義されたコンポーネントに送信される場合でも、SNA 文字ストリング (SCS) 制御を使用してフォーマット設定されます。

必要な場合、例外応答の送信側は、両方のハーフセッションの送信 / 受信状態が再び同期化されるまで、ERP PURGE モードに入ります。例外応答の受信側は再同期化イベントを起こす必要があります。

### 関連概念

[SNA 文字ストリング制御 \(システム・プログラミング API\)](#)

### 関連資料

557 ページの『[エラー・リカバリー・プロシージャ FM ヘッダー](#)』

IMS は FMH7 ERP ヘッダーおよびその関連メッセージを、選択受信側 ERP 例外応答 (X'0846') の後に続けて送信します。

538 ページの『[ERP ページ](#)』

例外応答の送信後で送信または受信を継続する前に、例外応答の送信側は、ハーフセッションの両方の DFC 状態マネージャが同期化されるまで、エラー・リカバリー・プロセス (ERP) PURGE モードに入れなければならない場合があります。

550 ページの『[送信側 ERP](#)』

SNA CANCEL 要求は、エラーのあるチェーンを終了させ、ハーフセッション間を同期させるために、必要に応じて送信側エラー・リカバリー・プロセス時に使用されることがあります。CANCEL は、(例外応答の送信側により) 要求されたものである場合も、(受信側が検出したエラーが例外応答をもたらすため) 要求されたものでない場合もあります。

## サポートされる選択受信側 ERP センス・コード

ハーフセッションのどちらかの役割を想定している場合、IMS は出力時に SNA 定義済み選択受信側 ERP システム・センス・コードの受信を認識し、メッセージをキューに戻し、他方のハーフセッションの ERP

のページ・サイクルを終了させるための同期化イベントが起こるようにし、また他方のハーフセッションからの入力 (ERP メッセージ) によってのみ可能な受信状態に入ります。

この ERP メッセージの受信中に、IMS は FMH7 センス・コードおよび、受信した ERP メッセージと関連した DFC プロトコルに基づき、次の 3 つの処置のうちの 1 つを実行します。

- ERP メッセージ (DFS2083) は、エラー状態が ISC セッションで回復されたことの警告として、IMS マスター端末オペレーターに経路指定されます。エラーのある ISC 出力メッセージはキューから除去されるか、または再送され、正常入力または出力操作は継続します。FMH7 がセンス・コード X'0866' 付きで EB なしで受信された時に、メッセージが再送されます。FMH7 がテスト、応答、または会話型モード出力中に EB 付きで受信された場合、メッセージはデキューされます。
- エラーのある ISC メッセージ通信出力の発信元 (IMS メッセージ通信を使用) である IMS 端末オペレーターと関連した IMS LTERM に、ERP メッセージ (DFS2073) が経路指定されます。エラーのある ISC 出力メッセージはデキューされ、正常入力または出力操作は継続されます。
- ERP メッセージ (DFS2073) は IMS マスター端末オペレーターに経路指定され、ISC セッションが終了します。エラーのある ISC 出力メッセージは IMS メッセージ・キューに残ります。

ただし、ERP KPSSES=Y であり、送信側 ERP センス・コードが X'08460000' である場合は、元のメッセージがデキューされ、セッションはアクティブのままです。この場合、どの端末に ERP メッセージ (DFS2073I) が渡されるかは、COMPTn パラメーターの指定と、どちら側がエラーのあるメッセージを開始したかによって異なります。元のメッセージがセッションの 2 次側から開始されており、COMPTn=SINGLE2 | MULT1 | MULT2 である場合、ERP メッセージは元の入力端末に経路指定されます。それ以外の場合、ERP メッセージは MTO に経路指定されます。

すべての ERP メッセージは EB または CD を伴わなければなりません。そうでない場合には、セッションが終了します。さらに、ERP メッセージで受信された EB は、IMS の会話、応答、またはテスト・モードの終了を強制します。会話モードが終了する場合には、IMS /EXIT コマンドの受信時と同様に、会話異常終了出口ルーチンも呼び出されます。IMS は、次のサブピックに示すように、FMH7 センス・コードおよび、メッセージと関連した DFC プロトコルに基づいて、受信した FMH7 ERP メッセージを処理します。

## X'0864xxxx': 機能打ち切り

再送でループが起きます。送信側はデータを再送してはなりません。

以下の表は、センス・コード X'0864xxxx' を持つ FMH7 ERP メッセージを IMS が処理する方法を示しています。

表 97. センス・コード X'0864xxxx' に対する IMS の処理

メッセージ・タイプ	MTO DFS2083	MTO DFS2073	TERMINAL DFS2073
応答モード			
FMH7 W/EB	X		
FMH7 W/CD <sup>547</sup> ページの『1』	X		
最後でない会話			
FMH7 W/EB <sup>547</sup> ページの『2』	X		
FMH7 W/CD		X	
最後の会話			
FMH7 W/EB <sup>547</sup> ページの『2』	X		
FMH7 W/CD <sup>547</sup> ページの『1』, <sup>547</sup> ページの『2』	X		
メッセージ通信			X
テスト・モード			

表 97. センス・コード X'0864xxxx' に対する IMS の処理 (続き)

メッセージ・タイプ	MTO DFS2083	MTO DFS2073	TERMINAL DFS2073
FMH7 W/EB	X		
FMH7 W/CD		X	
その他のメッセージ・タイプ			
FMH7 W/EB	X		
FMH7 W/CD <sup>547</sup> ページの 『1』	X		

注:

1. エラーのある出力が応答モードの応答、最後の会話型出力、または SCHEDULER なしの ATTACH である場合には、CD 付きで受信された FMH7 は、EB 付きの LUSTATUS-NO-OP (X'0006') をスケジュールして、セッション上で次に戻されるようにします。
2. 会話異常終了出口ルーチンが、/EXIT で行われるように呼び出されます。

### X'0865xxxx': 機能打ち切り

送信側がループを検出する責任があります。

以下の表は、センス・コード X'0865xxxx' を持つ FMH7 ERP メッセージを IMS が処理する方法を示しています。

表 98. センス・コード X'0865xxxx' に対する IMS の処理

メッセージ・タイプ	MTO DFS2083	MTO DFS2073	TERMINAL DFS2073
応答モード			
FMH7 W/EB	X		
FMH7 W/CD		X	
最後でない会話			
FMH7 W/EB <sup>548</sup> ページの 『1』	X		
FMH7 W/CD		X	
最後の会話			
FMH7 W/EB <sup>548</sup> ページの 『1』	X		
FMH7 W/CD		X	
メッセージ通信			X
テスト・モード			
FMH7 W/EB	X		
FMH7 W/CD		X	
その他のメッセージ・タイプ			
FMH7 W/EB		X	
FMH7 W/CD		X	

注:

1. 会話異常終了出口ルーチンが、 /EXIT で行われるように呼び出されます。

## 関連概念

SNA 文字ストリング制御 (システム・プログラミング API)

## X'0866xxxx': 機能打ち切り

受信側がループを検出する責任があります。

以下の表は、センス・コード X'0866xxxx' を持つ FMH7 ERP メッセージを IMS が処理する方法を示しています。

表 99. センス・コード X'0866xxxx' に対する IMS の処理

メッセージ・タイプ	MTO DFS2083	MTO DFS2073	TERMINAL DFS2073
応答モード			
FMH7 W/EB	X		
FMH7 W/CD	X		
最後でない会話			
FMH7 W/EB <sup>548</sup> ページの 『2』	X		
FMH7 W/CD	X		
最後の会話			
FMH7 W/EB <sup>548</sup> ページの 『2』	X		
FMH7 W/CD	X		
メッセージ通信	X		
テスト・モード			
FMH7 W/EB	X		
FMH7 W/CD		X	
その他のメッセージ・タイプ			
FMH7 W/EB	X		
FMH7 W/CD <sup>548</sup> ページの 『1』	X		

### 注:

1. エラーのある出力が応答モードの応答、最後の会話型出力、または SCHEDULER なしの ATTACH である場合には、CD 付きで受信された FMH7 は、EB 付きの LUSTATUS-NO-OP (X'0006') をスケジュールして、セッション上で次に戻されるようにします。
2. 会話異常終了出口ルーチンが、 /EXIT で行われるように呼び出されます。

IMS では、入力時に、未定義トランザクション・コード、誤ったトランザクション形式、セキュリティ侵害、SNA プロトコルとデータ構造エラー、および同期点要求エラーなどの様々な入力エラーおよび内部処理状態を検出することができます。これらのエラーは、ほとんどメッセージの SNA 要求エレメントで発生し、これにより、IMS は選択受信側 ERP センス・コードを示す例外応答を送信します。つまり、同期化する CANCEL コマンド、RQD1 または RQD2、方向変換標識、または CHASE コマンドを受信するまで入力メッセージを除去し、その後、選択受信側 ERP メッセージを送信します。

IMS では、ERP メッセージで送信された ERP FMH7 のセンス情報が利用できます。

**X'08260000'**

正常な IMS ERP メッセージが送信された場合に検出される 入力および内部処理エラー (このリストの残りの部分に示すエラーを除く) に使用されます。

**X'10030000'**

ATTACH ATTPRN 値が IMS で 認識されない場合に送信されます。

**X'08650000'**

データ送信側がループの検出および防止を行う機能打ち切りのセンス・コード。機能打ち切り X'08650000' は、DFC ブラケット間状態をもたらさない MFS 自動ページ化入力メッセージ中に、要求されていない CANCEL コマンド (送信側検出エラー) を受信した場合に、IMS により送信されます。また、IMS 要求時ページ化出力メッセージ中に受信された入力ページ要求でエラーが検出された場合、および自動ページ化入力メッセージの最初でないページで検出された場合にも、IMS は機能打ち切り X'08650000' を送信します。

**X'084B0000'**

接続 ATTDPN 値がデータを入力するコンポーネントで利用不能の場合に送信されます。これは、入力 DPN 値が、ISCEDT、任意の ISC 編集別名、または基本編集ではないか、あるいは MFS がコンポーネントで利用不能であることを示します。ただし、MFS は利用可能であるが DPN 値が有効な MID でなかった場合には、センス・コード X'08260000' ではなく X'084B0000' が送信されます。

**X'080F0000'**

ATTACH データ・ストリーム・プロファイル (DSP) 値が、IMS 内のハーフセッション名に有効なまたは定義済みのコンポーネント番号を定義しない場合に送信されます。

**X'1008xxxx'**

ユーザー・フィールド (xxxx) は、SNA によって次のように定義されます。

X'6001': 無効な ATTACH FM ヘッダー ATTDDBA 値

X'1204': FMH4 の無効なバージョン値

セッションの即時終了の原因となる、入力または出力操作中に検出される他のタイプのエラーは次のとおりです。

- ブラケット・プロトコル違反
- 送信 / 受信プロトコル違反
- サポートされない応答要求
- サポートされないまたは無効な FMH タイプおよび形式
- LOSTERM、バッファ・プールまたは消耗 RPL スペースのコピー、または入力 RU 短縮形など、VTAM で検出されるその他の重要なエラーまたは状態。

**関連資料**

544 ページの『ページ化メッセージ ERP』

IMS 複数チェーン要求または自動ページ化出力メッセージの伝送が開始された後で送信側検出エラー (/DEQ コマンド) が検出された場合は、IMS は (必要な場合) CANCEL コマンドを LUSTATUS - 打ち切りと一緒に送信しなければなりません。

**データ・フロー・リセット状態時の受信側検出エラー**

セッション・フローがデータ・フロー・アクティブ状態でないとき、例えば、バインド時、再同期化 (STSN) 時またはデータ通信開始 (SDT) 時などには、次のセンス・コードが送信および受信されます。

**X'0845xxxx'**

無効なユーザー・データにより拒否されたバインド。

**X'0847xxxx'**

BIND1 STSN または SDT の受信時に 2 次ハーフセッションが検出した再始動モードの不一致。

**X'08210000'**

無効なセッション・パラメーターにより拒否されたバインド。

**X'080Dxxxx'**

バインド・レースにより拒否されたバインド。



## 関連概念

503 ページの『バインド競争の解決』

IMS と別の論理装置が同時に BIND 要求を送信し合い、2つのハーフセッション名がミラー・イメージになると、競争が発生します。

## 送信側 ERP

SNA CANCEL 要求は、エラーのあるチェーンを終了させ、ハーフセッション間を同期させるために、必要に応じて送信側エラー・リカバリー・プロセス時に使用されることがあります。CANCEL は、(例外応答の送信側により) 要求されたものである場合も、(受信側が検出したエラーが例外応答をもたらすため) 要求されたものでない場合もあります。

チェーン RU または自動ページ化出力時に必要な場合には、両方のハーフセッションの送信 / 受信状態が再び同期化されるまで、例外応答の送信側は ERP PURGE モードに入ります。例外応答の受信側は再同期化イベントを起こす必要があります。このイベントは次のいずれかになることがあります。

- RQD1 または RQD2
- 方向変換の受信
- 例外応答が出力チェーン中に行われた場合には、CANCEL 要求 (RQD1 または RQE1/CD によって生じた)
- IMS が自動ページ化出力メッセージの RQE1 出力チェーン (ページ) 間にある時に例外応答が受信された場合には、CHASE 要求 (CHASE 上の RQD1 による)

**関連資料:** 送信側エラー・リカバリー・プロシージャー (ERP) について詳しくは、「z/OS Communications Server: SNA Programming」を参照してください。

## 送信側 ERP センス・コード

2つのIMSサブシステム間に送信される、送信側ERPセンス・コードはX'08130000'およびX'0846xxxx'だけです。

IMSが受信する有効なコンテンション・センス・コード、および実行される追加の操作は次のとおりです。

- X'08130000' - ブラケット拒否 (RTR は送信されない)

受信準備済み (RTR) 状態が設定されていません。疑似待機が満たされた場合には、メッセージが初めから再送されます。このセンス・コードはIMSが送信するSNA BISコマンドを拒否するために使用することもできます。

- X'08140000' - ブラケット拒否 (RTR は送信される)

メッセージはキューに戻され、RTR 保留状態に入ります。IMSがブラケット間にある時には、RTRが受信されるまで出力は送信されません。ただしIMSが方向変換を受信した後に再び、イン・ブラケット / SEND 状態にされると、メッセージ・フローは、メッセージ・タイプまたはコンポーネントに定義されたブラケットおよび送信 / 受信プロトコルに従います。

上記のいずれかのセンス・コードの後に、IMSは出力メッセージの送信を試みます。

IMSは1次ハーフセッションの役割の場合、競合システム・センス・コードX'08130000'の受信を認識し、出力メッセージをメッセージ・キューへ戻し、以下により可能な疑似受信状態に入ります。

- 2次ハーフセッションからの入力を受信
- IMS マスター端末オペレーター・コマンドや、別の論理装置からのメッセージ通信、またはIMSアプリケーション・プログラムで挿入された追加のメッセージの結果による出力通知

上記のいずれかの処置の後に、IMSは出力メッセージの送信を試みます。

1次ハーフセッションに対するブラケットの開始後、2次ハーフセッションの役割を想定して、入力データまたは通常フロー・コマンド (BID および BIS) を受信する場合は、IMSは単にコンテンションを示すだけです。このような場合には、IMSはX'08130000'センス・コードを送信してコンテンションを示しません。X'08140000'競合センス・コードはIMSは送信しません。

送信および受信される追加の例外応答の送信側ERPセンス・コードは次のとおりです。

- X'08190000' - 出力は使用不能

1 次ハーフセッションの役割を想定している場合、IMS はこのコードを送信しますが、受信準備済み (RTR) 標識の受信直後に、出力は使用できません。IMS は RTR を送信しないので、IMS はこのセンス・コードを受信しません。

- X'0846xxxx' - 選択受信側 ERP

IMS SNA 選択受信側 ERP センス・コードを送信し、受信します。IMS はこのセンス・コードを、応答モード時、会話型トランザクション時、またはアプリケーション異常終了時に送信します。xxxx はユーザー・センス・フィールドであり、受信時に IMS はこれを無視します。ただし、IMS マスター端末もしくはメッセージの発信元である端末オペレーターに送信するメッセージに、このセンス・コードを含めません。出力中に、通常、センス・コードは IMS エラー・メッセージ番号の 2 進値にセットされ、これは応答を送信する時に ERP メッセージとして送信されます。ただし、ユーザー・センス・コードは、例えばアプリケーションの異常終了などのエラーが応答モードまたは会話型トランザクションで発生した場合、X'0000' にセットすることができます。

- X'0864xxxx' - 機能打ち切り。再送でループが起きます。送信側はデータを再送してはなりません。

IMS はこの機能打ち切りセンス・コードを受信しますが、送信はしません。このコードの受信時、IMS は関連した出力メッセージをデキューしてから (まだアクティブな場合)、正常な入力または出力操作を継続します。xxxx はユーザー・センス・フィールドであり、IMS はこれを無視します。LUSTATUS - abort は、IMS MFS ページ化出力メッセージの終了前に入力が受信された場合に、IMS によって送信されます。

最後でない IMS 会話型出力中には、機能打ち切りセンス・コード X'0864xxxx' を IMS に送信することはできません。送信すると、出力メッセージはキューに戻されます。マスター端末オペレーターに通知され、セッションは終了します。機能打ち切りセンス・コード X'0864xxxx' が最後の会話型出力メッセージの後で送信される場合、メッセージはデキューされ、会話型異常終了出口ルーチンは、/EXIT で行われる場合と同様に呼び出されます。

- X'0865xxxx' - 機能打ち切り。送信側がループの検出を行う責任があります。

IMS は送信側 ERP 打ち切りセンス・コードを受信しますが、送信はしません。センス・コードを受信した場合には、IMS はメッセージをキューに戻し、セッションをクローズします。IMS はセッションの再始動後、次の機会にメッセージを初めから再送します。xxxx はユーザー・センス・フィールドであり、IMS はこれを無視します。

- X'0866xxxx' - 機能打ち切り。受信側がループの検出を行う責任があります。

このセンス・コードによって、IMS は次の機会にメッセージを初めから再送します。このセッションは終了しません。xxxx はユーザー・センス・フィールドであり、IMS はこれを無視します。

- X'08670000' - マルチチェーン ERP ページ。

IMS は、ERP ページの終了時に複数チェーン・ページのセンス・コードを送信します。MFS 出力自動ページ化メッセージの最初でない、または最後でないページに対して例外応答を受信した場合のみ、IMS は複数チェーン ERP ページ・センス・コードを受信します。非 MFS 自動ページ化出力および MFS 出力自動ページ化メッセージの最初および最後のページが確定応答を要求して送信されます。これらの SNA チェーンに当てはまらないものは、単一チェーンの除去のみの結果となります。

他の送信側 ERP センス・コードによって、IMS はマスター端末オペレーターに通知し、セッションを終了します。

#### 関連資料

545 ページの『[選択受信側 ERP](#)』

選択受信側のエラー・リカバリー・プロシージャ (ERP) が X'08460000' 送信側 ERP センス・コードを介して開始されます。

538 ページの『[ERP ページ](#)』



例外応答の送信後で送信または受信を継続する前に、例外応答の送信側は、ハーフセッションの両方の DFC 状態マネージャーが同期化されるまで、エラー・リカバリー・プロセス (ERP) PURGE モードに入れなければならない場合があります。

## 非ページ化メッセージ上の送信側検出エラー

単一 SNA 非ページ化、複数 RU メッセージの伝送を開始した後でエラーが検出された場合には、送信側は SNA 非送信請求 **CANCEL** コマンドを送信することができます。

伝送中のメッセージ (単一の SNA チェーン) の最後のセグメント前に入力されたオペレーター **/DEQUEUE** コマンドで現行の出力メッセージが終了した場合にだけ、IMS は非送信請求 **CANCEL** を送信します。

IMS によるデータの受信中に、IMS が非送信請求 **CANCEL** を受信した場合には、メッセージは破棄されるか、バックアウトされます。

## IMS が受信するセンス・コード

センス・コードとは、発生した特定の例外条件に対し定義されたカテゴリと修飾子を含む 2 バイトのフィールドです。センス・コードの後の 2 バイトは、オプションのユーザー・データを含むことができ、すべてのセンス・コードにサポートされているわけではありません。

以下の表は、IMS が受信する入力センス・コードを示しています。

入力センス・コード名	入力センス・コード	送信側 ERP	FMH7	LUSTATUS
Commit/NO-OP	- X'00060000'			X
空キュー	- X'00070000'			X
Bind 競争	- X'080D0000'	X		
コンテンツョン	- X'08130000'	X		
コンテンツョン	- X'08140000'	X		
無効パラメーター	- X'08210000'	X		
選択受信側 ERP	- X'08460000'	X		
再始動拒否	- X'08470000'	X		
打ち切り	- X'08640000'	X	X	X
打ち切り	- X'08650000'	X	X	X
打ち切り	- X'08660000'	X	X	X
ERP パージ	- X'0867'	X		
その他		注 1	注 1	注 1

注：

1. その他のセンス・コードが出されると、IMS はマスター端末オペレーターに通知し、セッションを終了させます。

**関連資料:** センス・コードについて詳しくは、「z/OS Communications Server IP and SNA Codes」を参照してください。

## IMS が送信するセンス・コード

以下の表は、IMS が送信する出力センス・コードを示しています。

表 101. IMS が送信するセンス・コード

出力センス・コード名	出力センス・コード	送信側 ERP	FMH7	LUSTATUS
NO-OP	- X'00060000'			X
空キュー	- X'00070000'			X
Bind 競争	- X'080D0000'	X		
ATTDSP	- X'080F0000'		X	
コンテンツョン	- X'08130000'	X		
出力なし	- X'08190000'	X		
無効パラメーター	- X'08210000'	X		
サポートされない	- X'08260000'		X	
選択受信側 ERP	- X'08460000'	X		
再始動拒否	- X'08470000'	X		
ATTPDN	- X'084B0000'		X	
打ち切り	- X'08650000'	X		X
ERP パージ	- X'0867'	X		
ATTPRN	- X'10030000'		X	
FMH4 バージョン ID	- X'10081204'		X	
ATTDBA	- X'10086001'		X	

## SIGNAL プロトコル

IMS は SIGNAL コマンドを送信し、入力メッセージ・チェーンの終わりで方向変換 (SIGNAL RCD - X'00010000') を要求します。この入力メッセージ・チェーンは、方向変換を示さず、後続の入力前に送信しなければならない出力を作成します。

例えば、IMS は、テスト・モードで、入力コマンド、応答モード・トランザクション、会話型トランザクション、または入力に応答します。SIGNAL RCD は、IMS が入力メッセージに必要な DR1 または DR2 応答を送信する前に、送信されます。

IMS 応答プロトコルおよび IMS が SIGNAL RCD を送信する点を基にして、IMS は SIGNAL RCD の後の次の入力が CD または EB を示す LUSTATUS あるいは CHASE であることを要求します。IMS によってどちらも受信されなかった場合には、セッションは終了します。方向変換によって IMS は保留状態の出力メッセージを送信することができます。つまり、ブラケット終了により、メッセージがデキューされ、会話モード、応答モード、またはテスト・モードが終了します。保留状態の出力メッセージが正常に伝送されるかまたはキューから除去されるまで、その他の入力を処理することはできません。

ブラケット状態にある SIGNAL RCD の受信後、IMS は方向変換を現行出力メッセージの終わり、または LUSTATUS の受信の直後に送信します。ブラケット間の時に受信された SIGNAL RCD によって、IMS は通常フロー入力 (データまたは SNA コマンド) が行われるまで、出力メッセージをこれ以上送信することはできません。

### 関連資料

667 ページの『シグナル・プロトコルの例』

次の例は、SIGNAL RCD の使用例を示しています。

## LU 6.1 (SBI および BIS) の対称セッション・シャットダウン

停止ブラケット開始 (SBI) とブラケット開始の停止 (BIS) という 2 つのデータ・フロー制御コマンドにより、対等レベルの LU 6.1 ハーフセッションが対称的かつ正常に終了することができます。

SBI は VTAM 急送フロー・コマンドで、BIS は通常フロー・コマンドです。

これらのコマンドは、新規のブラケットを開始する通常フローの要求だけを制御します。また、これらのコマンドは既存のブラケット内でフローすることができる応答は排除しません。受信ハーフセッションで今後のブラケットの開始を抑制することを要求するために、ハーフセッションはいつでも SBI を送信できます。SBI の受信側が、シャットダウンを受け入れられるブラケット間の点に達すると、受信側は SBI を開始した側に BIS コマンドを戻します。BIS 送信側は、その後、「開始ブラケットなし」または「NOBB」状態になり、その間そのハーフセッションはブラケット開始要求を受信し、ブラケット間にある時には通常フローで要求や応答を送信することはできますが、自分自身のブラケットを開始することはできません。両方のハーフセッションがこの状態になった場合は、NOBB 状態にあるハーフセッションの指示を保持し、セッションを終了することによって、1 次ハーフセッションは SBI レース状態を検出し、解決しなければなりません。

シャットダウン順序の正常終了とは、両方のハーフセッションが正常に NOBB 状態に入ることを指し、つまり正常なセッション終了のことを言います。

IMS /QUIESCE コマンドが入力されるか、または静止オプションが /CHECKPOINT コマンドで指定される場合に、IMS は SBI コマンドを送信します。他のサブシステムから BIS が戻される場合には、IMS は QUIESCE または CHECKPOINT コマンド処理で続行します。/QUIESCE コマンドが入力されており、CHECKPOINT QUIESCE が進行中でない場合には、IMS は次のトランザクションで BIS をブラケット間状態に送信します。IMS が BIS を受信した時に CHECKPOINT QUIESCE が進行中だった場合には、IMS は他の /CHECKPOINT パラメーター (FREEZE、PURGE、および DUMPQ) にしたがって出力の送信を継続し、シャットダウン・プロセス完了時に BIS を送信します。

SBI または BIS の受信後に IMS が BIS を送信するのは、CHECKPOINT QUIESCE が進行中でない時の、次のブラケット間状態の時か、あるいは CHECKPOINT QUIESCE が進行中のシャットダウン・プロセスの終了時です。

IMS が 1 次ハーフセッションの場合には、BIS の送信または受信後、IMS は両方のハーフセッションが NOBB 状態にあることを検査します。両方が NOBB 状態にあった場合には、IMS はすべてのメッセージ・カウントをリセットし、セッションを終了します。これ以外の場合には、IMS は通常の処理で続行します。IMS が 2 次ハーフセッションである場合は、IMS はセッションを終了するために 1 次ハーフセッションを待機します。LTERM サブプールがセッションに割り振られている場合は、両方のハーフセッションが NOBB 状態にある場合には、セッションの終了前に割り振り解除され、関連したメッセージ・カウントはリセットされます。この割り振り解除/終了プロセスは、後続のセッションのコールド・スタートを許可し、他のセッションによる使用のために割り振られたサブプールを解放します。このプロセスによって、セッションの再始動時に、再同期またはリカバリーが要求されません。

いったん NOBB 状態に入るとブラケットの開始は許可されないため、単一ブラケット・チェーン (開始ブラケットとブラケット終了) 入力メッセージが、ブラケットを開始する即時の応答 (システム・メッセージやエラー・メッセージなど) を要求すると問題が起きます。NOBB 状態の時に、これらの入力メッセージのブラケット終了を除外しないで、IMS はすべての入力を受け入れて処理を続行します。NOBB 状態の時に IMS にブラケットの開始を要求する応答が発生すると、セッションは終了します。

IMS が応答モードか、会話型モード、または高速機能応答を待機中の場合は、サブプール割り振り解除は行えず、また BIS を送信することもできません。これは、EB が出力または入力のどちらにも流れなかった (つまり、IMS がブラケット間でない) ためです。

### 関連資料

[665 ページの『SBI/BIS の例』](#)

IMS は、どの掲載例の場合でも、1 次ハーフセッション (PHS) であることもできますし、2 次ハーフセッション (SHS) であることもできます。したがって、PHS および SHS のために表示された機能およびコマンドのすべては IMS 機能およびコマンドです。

## 機能管理ヘッダー

SNA では、機能管理 (FM) ヘッダーはリンクに対し送信される要求単位のオプション の部分です。このトピックでは、ISC セッション上の IMS によってサポートされる FM ヘッダーについて説明します。

このトピックで説明するヘッダーには次のものがあります。

- **ATTACH 機能管理ヘッダー。**このタイプ 5 ヘッダーは、セッション入力を受信できるようにプロセスを付加するために使用されます。これには、付加されたプロセスによって使用される他のパラメーターだけでなく、同期に付加されるプロセスの名前も伴われます。すべてのメッセージは明示または暗黙の ATTACH FM ヘッダーをもちます。実際にメッセージで送信されない場合には、ヘッダーは暗黙です。暗黙の ATTACH FM ヘッダーは、リセット・パラメーター値またはこのトピックで後に定義する直前の ATTACH の値を想定します。
- **エラー・リカバリー・プロシージャ (ERP) 機能管理ヘッダー。**このタイプ 7 ヘッダーは、エラー・メッセージの送信を要求するエラー条件が検知された後で送信されます。
- **リセット付加プロセス (RAP) 機能管理ヘッダー。**このタイプ 5 ヘッダーは IMS に対して入力され、付加されたプロセスおよび関連したデータ・フロー制御状態をリセットさせます。

次のヘッダー・タイプは ATTACH FM ヘッダーに従属し、ATTACH FM ヘッダーが以前にそのブラケット内で関連プロセスを付加するために送信されていない場合には、使用することができません。

- **SCHEDULER 機能管理ヘッダー。**このタイプ 6 ヘッダーは、SCHEDULER プロセスが付加された後送信されます。これには、SCHEDULER プロセスによって要求される他のパラメーターだけでなく、IMS 内で非同期にスケジュールされるプロセスの名前も含まれます。
- **SYSMSG 機能管理ヘッダー。**タイプ 6 システム・メッセージ・ヘッダーは、IMS により送信または受信されたシステム・メッセージに先行します。
- **データ記述子機能管理ヘッダー。**このタイプ 4 ヘッダーは、MFS と一緒に使用され、データ構造名、バージョン ID または現行の出力フィールド・タブ分離文字を送信するために使用されます。
- **QMODEL 機能管理ヘッダー。**これらのタイプ 6 ヘッダーは、IMS メッセージ形式サービスによって送信される要求時ページ化メッセージに使用されます。

すべての機能管理ヘッダーは先頭チェーンまたは単独チェーンになければなりません。以下の表では、IMS が送受信するヘッダーのタイプを要約しています (Y の場合サポートされます)。これらのヘッダー・タイプについては、後のトピックで説明します。

表 102. 機能管理ヘッダーのタイプ

機能管理ヘッダーのタイプ	IMS によって送信された	IMS によって受信された
FMH4 - データ記述子	Y	Y
FMH5		
ATTACH	Y	Y
リセット付加プロセス (RAP)		Y
FMH6 - QMODEL		
QGET		Y
QGETN		Y
QPURGE		Y
QSTAT	Y	
QXFR	Y	

表 102. 機能管理ヘッダーのタイプ (続き)

機能管理ヘッダーのタイプ	IMS によって送信された	IMS によって受信された
FMH6 - SCHEDULER	Y	Y
FMH6 - SYSMMSG		
SYSSTAT (デフォルト)		Y
SYSSTAT	Y	Y
SYSERROR	Y	Y
FMH7 - ERP	Y	Y

ヘッダーの受信時、影響されやすい最後のパラメーターを超えてコード化されたパラメーターを IMS は無視します (つまり、IMS がサポートする)。その他の LU 6.1 サブシステムは、IMS が送信するヘッダーを受信する時にはこの手順に従う必要があります。

## ISC 編集を呼び出すための FM ヘッダーの使用

以下の状況では IMS は、トランザクションを編集するための ISC 編集 (ISCEDT)、コマンド、および IMS - ISC セッション用の LTERM 間でのメッセージ通信を使用することができます。

- 接続マネージャーはリセット状態 (ブラケット間または RAP に続いて) にあり、ATTACH FM ヘッダーを受信しないか、または宛先プロセス名 (ATTPDN) を指定しない ATTACH FM ヘッダーを受信します。
- ATTACH ヘッダーまたは SCHEDULER FM ヘッダー内の宛先プロセス名 (ATTPDN/SCDDPN) は ISCEDT と等しいかまたは COMM マクロ・ステートメント 上でのシステム定義中に定義された ISC 編集にユーザー定義の別名を指示します。
- 接続マネージャーはリセット状態ではなく、アクティブなプロセスは ISC 編集です。ATTPDN を指定しない ATTACH FM ヘッダーは受信されるかあるいは受信されないかのどちらかです。

ATTPRN/SCDPN パラメーターを指定しない入力が入力がトランザクション・コード およびメッセージの始めのパスワードに対してだけ編集されます。オプションの IMS パスワードおよび X'41' 未満の先行文字が編集中に削除されます。IMS トランザクション・コードとパスワードの形式および要件は、基本編集の場合と同じです。残りのメッセージに対しては、どのような編集も行われません。

接続マネージャーまたは SCHEDULER (提供された場合) から渡された入力パラメーターは戻り宛先プロセス名 (ATTPDN/SCDRPN)、戻り 1 次リソース名 (ATTPRN/SCDRPN)、および 1 次リソース名 (ATTPRN/SCDPN) です。

557 ページの『プロセスの開始: ATTACH FM ヘッダー』で定義されているように、入力 ATTPRN/SCDPN パラメーターは、接続マネージャーまたは SCHEDULER から、IMS トランザクション・コードとして、またはメッセージ通信上の LTERM 名として渡されます。ATTPRN/SCDPN パラメーターが提供される場合には、入力に対してどのような編集も行われません。ATTPRN/SCDPN は IMS 宛先を定義しますが、IMS トランザクション・セキュリティーに 入力パスワードを提供しません (このセキュリティーが ISC ノードに定義されている場合)。

### 関連概念

444 ページの『基本編集』

MFS を使用しない場合は、IMS の基本編集機能でメッセージを編集します。

### 関連資料

591 ページの『ISC 編集 ATTACH パラメーターの使用の例』



以下のトピックでは、ISC 編集 ATTACH パラメーターの使用例を示します。

## プロセスの開始: ATTACH FM ヘッダー

ATTACH FM ヘッダーを使用して、セッション入力を受信するプロセスを付加します。これには、付加されたプロセスが使用する他のパラメーターだけでなく、非同期式に付加されるプロセスの名前も含まれます。

ATTACH が同期実行で定義されるので、付加されるプロセス、IMS メッセージ・タイプと実行モード、および SCHEDULER プロセス間の関係を理解する必要があります。

IMS は次の手段で SCHEDULER FM ヘッダーなしの ATTACH FM ヘッダーを送信します。

- 入力 ATTACH から生じる会話型、応答モード、または IMS コマンド出力メッセージの応答。
- ATTACH EB と同じセッションで受信されたメッセージから生じた非同期応答。
- 他方のハーフセッションが SCHEDULER モデル・サポートなしでバインドされる時。
- SYSMMSG を通した IMS によって送信されたシステム・メッセージ。

IMS メッセージは、ATTACH ATTIU パラメーター内で示されるように、単一 SNA チェーンまたは複数 SNA チェーンのどちらかで構成できます。ATTACH FM ヘッダーは各入力または出力 IMS メッセージに一度だけ存在することができます。出力 MFS 要求時ページ化メッセージでは、ATTACH FM ヘッダーは最初の出力 SNA チェーンにだけあります。要求時ページ化出力に対する最初の入力ページング要求には、ATTACH (DPN=QMODEL の場合) が含まれていなければなりません。同じ要求時ページ化出力メッセージの後続の入力ページ要求は、オプションとして ATTACH (これも DPN=QMODEL を示す必要がある) を含むことができます。

### 関連概念

[471 ページの『ISC と IMS の実行モードの関係』](#)

「同期」と「非同期」という用語は、IMS と ISC では少し意味が異なります。以下のトピックでは、これらの実行モードの関係について説明します。

### 関連資料

[571 ページの『ATTACH FM ヘッダーの形式』](#)

ATTACH FM ヘッダーの形式が、以下の表に定義されています。

## エラー・リカバリー・プロシージャ FM ヘッダー

IMS は FMH7 ERP ヘッダーおよびその関連メッセージを、選択受信側 ERP 例外応答 (X'0846') の後に続けて送信します。

ERP 例外応答を FMH7 ERP ヘッダーおよびその関連メッセージの受信以前に IMS が受信する必要はありません。IMS によって送信された FMH7 メッセージは、それぞれ RQE1/CD または RQD1/EB のどちらかを示します。IMS により送信または受信された ERP メッセージは、SCS 文字の単一の SNA チェーンに限定され、VLVB 形式のレコードを入れてはいけません。IMS によって受信された FMH7 メッセージは CD または EB のどちらかを指示する必要があります。RQE1 または RQD1 はどちらかで許可されます。

IMS 内で作成される単一セグメントのエラー・メッセージの長さは、79 文字またはそれ以下です。IMS 内で作成される複数セグメントのメッセージは 79 文字より長くても構いませんが、各セグメントの最大長は 79 文字です。

ISC エラー・メッセージ・プロセスへの入力により作成され、マスター端末または発信元オペレーター端末へ出力されるエラー・メッセージは、単一セグメントと複数セグメントの出力メッセージ (DFS2073 または DFS2083) を作成します。

IMS はデフォルトの MFS MOD 名を使用して、エラー・メッセージを IMS メッセージ・キューに挿入します。MFS DPM で定義されたコンポーネントへの出力中に MOD によって定義されたようにこれらのメッセージはフォーマット設定されます。これらの MOD および関連した DOF は MFS 形式ライブラリーに入っています。

各選択受信側 ERP メッセージは、特定のシステムおよびユーザー・エラー・センス・コードを含む ERP メッセージ・ヘッダーに先行されます。

## 関連資料

545 ページの『[選択受信側 ERP](#)』

選択受信側のエラー・リカバリー・プロシージャ (ERP) が X'08460000' 送信側 ERP センス・コードを介して開始されます。

580 ページの『[エラー・リカバリー・プロシージャ \(ERP\) FM ヘッダー](#)』

以下の表には、ERP FM ヘッダーの形式が示されています。

## アクティブ・プロセスのリセット : RAP FM ヘッダー

リセット付加プロセス (RAP) FM ヘッダーは、ブラケットおよび送信 / 受信状態を除き、受信ハーフセッションのアクティブ・プロセスおよびすべてのセッション状態をブラケット間に対応する状態にリセットするために使用されます。

IMS は RAP FM ヘッダーを受信しても、送信はしません。RAP 要求の受信後、送信可能な出力がない場合には、IMS は ATTACH SCHEDULER、または LUSTATUS - 空キューで応答します。IMS 要求時ページ化出力メッセージ中に RAP 要求が受信された場合には、これ以上の出力の検査が行われる前に、メッセージがデキュー (コミット) されます。RAP 要求に対し実行された IMS 操作は、MFS 入力で提供された NEXTMSG オペレーター制御要求、または 3270 装置からの PA2 要求に対するものと同じです。

RAP 要求はデータなしで送信する必要があり、方向変換、RQD2、または RQE2 を指示しなければなりません。

以下の図は、RAP FM ヘッダーの使用方法を示します。

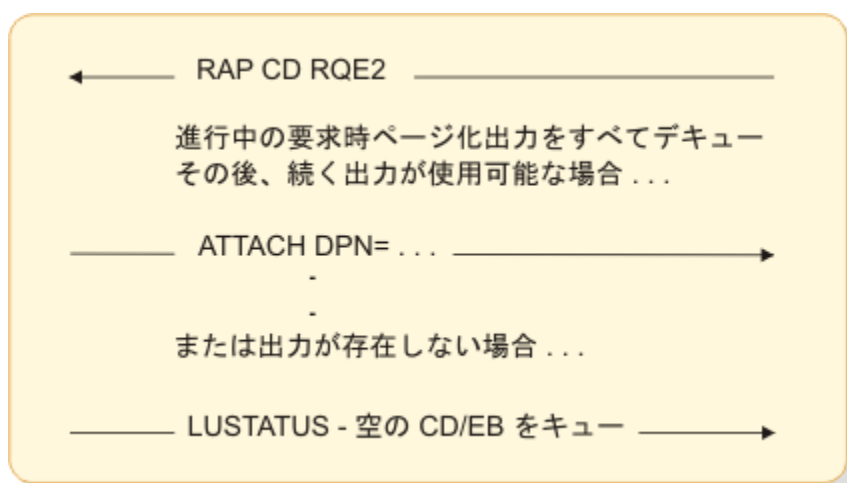


図 70. RAP FM ヘッダーの例

## 関連資料

588 ページの『[リセット付加プロセス \(RAP\) FM ヘッダーの形式](#)』

以下の表には、リセット付加プロセス (RAP) FM ヘッダーの形式が示されています。

## 非同期トランザクション処理の要求 : SCHEDULER FM ヘッダー

ATTACH は同期スケジューリングおよび実行で定義されます。つまり、SCHEDULER は非同期スケジューリングおよび実行で定義されます。

SCHEDULER プロセスは非同期に処理されるメッセージを送信するために附加することができます。つまり、出力が発生するかどうかにかかわらず、入力と結果の応答間の同期的関係は無視して、IMS はメッセージをスケジューリングまたは実行タイミングと無関係に処理中と認識します。論理装置間の各ハーフセッションは、セッション・バインド・パラメーターを使用して、SCHEDULER 処理がサポートされ、非同期メッセージの受信に使用可能かどうかを示すことができます。IMS ハーフセッションは常に、SCHEDULER プロセスが使用可能であると示します他のハーフセッションは、SCHEDULER プロセスが非同期入力を受信可能または受信不能であると示すことができます。

バインドによって、他のハーフセッションが SCHEDULER プロセスをサポートしないと指示した場合には、IMS はデフォルトとして ATTACH FM ヘッダーを含んだ各出力を送信します。



SCHEDULER プロセスは、SNA 名 (X'02') を ATTACH DPN パラメーターとして示し、SCHEDULER FM ヘッダーを ATTACH FM ヘッダーに連結することによって付加されます。SCHEDULER が付加されている時は、非同期にスケジュールされる後続の各メッセージは、少なくとも SCHEDULER FM ヘッダーと一緒に送信する必要がありますが、別の ATTACH FM ヘッダーは必要としません。SCHEDULER に渡された可変長 ATTACH パラメーターは入力時に無視され、IMS による出力と一緒に送信されません。SCHEDULER FM ヘッダーには、スケジューリングおよび戻り応答の経路指定に必要な情報だけでなく、スケジュールされるプロセスの名前 (SCDDPN) も含まれます。SCDDPN は SCHEDULER の必須パラメーターであり、IMS SCHEDULER 入力で提供する必要があります。IMS によって送信されたメッセージ上の SCDDPN は、直前の入力 SCDDPN からセットされるか、またはオプションとして、MFS により出力メッセージ形式記述子を使用して送信されます。他の値が使用可能でない場合には、SCDDPN はデフォルトとして "ISCEDT" にセットされます。

IMS 入力では、SCHEDULER プロセスが常に使用可能です。次のような条件下では、IMS は ATTPDN パラメーターを X'02' (SCHEDULER) に設定した ATTACH を、SCHEDULER FM ヘッダーと連結して送信します。

- (バインド・パラメーターで定義されたように) 他のハーフセッション が SCHEDULER モデルをサポートする場合。
- セッションで前に入力され、SCHEDULER プロセスを使用してスケジュールされた入力からの結果の非同期応答を送信する場合。
- 非送信請求の非同期メッセージを送信する場合。このようなメッセージの例は、IMS メッセージ通信またはセッション上の入力の結果として直接スケジュールされないアプリケーションから生じる出力応答です。

IMS 内では SCHEDULER は、単一チェーン入力および複数チェーン入力と、ATTACH ATTIU パラメーターを使用する出力メッセージの両方をサポートします。IMS は、ATTACH 複数チェーン標識を使用して複数の入力または出力メッセージをスケジュールすることを許可しません。

SCHEDULER FM ヘッダーは、各入力または出力 IMS メッセージごとに一回のみ存在することができます。出力 MFS 要求時ページ化メッセージでは、SCHEDULER FM ヘッダーは最初の出力 SNA チェーン (最初の出力 MFS ページ) にだけあります。IMS は SCHEDULER SCHEDSTAT、PURGE、または PURGSTAT FM ヘッダーを送信または受信しません。

#### 関連資料

[528 ページの『データ・フロー制御プロトコルの関連情報』](#)

以下のトピックで、データ・フロー制御 (DFC) のバイト・レベルのプロトコルについて説明します。プロトコルを以下に示します。

[588 ページの『SCHEDULER FM ヘッダーの形式』](#)

以下の表には、SCHEDULER FM ヘッダーの形式が示されています。

## システム・メッセージ・プロセス (SYSMSG) および関連 FM ヘッダー

システム・メッセージ・プロセスは、ATTACH FM ヘッダーのプロセス名で指示されます。

ATTPRN が ATTACH 内または SYSMSG の SYSERROR FM ヘッダーで提供されない場合には、入力システム・メッセージはマスター端末オペレーターに経路指定されます。この経路指定は、着信 SYSMSG を IMS システム・メッセージ DFS2072 に変換することによって実行されます。ATTPRN 値が提供された場合には、これは IMS 入力メッセージの宛先トランザクション・コードまたは LTERM 名になります。システム・メッセージ・プロセスを IMS コマンド・プロセッサにアクセスするために使用することはできません。

提供された ATTPRN 値が IMS トランザクション用だった場合には、応答が入力 SYSMSG から生じることがあります。出力コンポーネントの定義によって、この応答は ISC 編集または MFS を介し戻されます。

IMS はデフォルト MFS MOD 名を使用して、システム・メッセージを IMS メッセージ・キューに挿入します。MFS DPM で定義されたコンポーネントへの出力中に MOD によって定義されたようにこれらのメッセージはフォーマット設定されます。これらの MOD および関連した DOF は MFS 形式ライブラリーに入っています。

IMS は次の場合のみ SYSMSG を送信します。

- IMS ブロードキャスト・メッセージ

- 入力に応答を送信した後で IMS が送信するシステム・メッセージ。これらのシステム・メッセージは、応答を送信するアプリケーションに代わって出されます (例えば、異常終了が起こった場合など)。IMS コマンドによって直接送信請求される IMS システム・メッセージは、通常の応答として送信されます。他のシステム生成メッセージはエラー条件の結果として生じる場合があります。これらのメッセージは入力トランザクションへの例外応答に変換され、エラー・リカバリー・プロセス (ERP) を使用してエラー・メッセージとして送信されます。

出力コンポーネントの定義および応答が MFS MOD 名で挿入されたかどうかによって、MFS が出力 SYSMMSG を編集する場合があります。

SYSMMSG によって送受信されるメッセージには、SYSSTAT または SYSERROR FM ヘッダーのどちらかを入れることができます。送信側によってヘッダーが提供されない場合には、受信側は SYSSTAT FM ヘッダーを想定します。IMS はどちらのヘッダーでも受信でき、IMS メッセージ番号 DFS2072 を使用して各ヘッダーに接頭部を付けます (メッセージがマスター端末オペレーターに送信される場合)。IMS は SYSSTAT ヘッダーを使用してブロードキャスト出力を送信します。SYSMMSG として送信されるその他のすべての IMS システム・メッセージは、SYSERROR ヘッダーを使用します。SYSMMSG 出力をもたらす入力メッセージ上で ATTACH ATTRDPN または ATTRPRN パラメーターが提供されると、これらのパラメーターは受信側の SYSMMSG プロセスで使用される出力 SYSERROR FM ヘッダーに組み込まれます。ATTACH ATTRDPN および ATTRPRN パラメーターは IMS 出力 SYSMMSG では送信されません。

IMS 内で作成される単一セグメントのシステム・メッセージの長さは、79 文字またはそれ以下です。IMS 内で作成される複数セグメントのメッセージは 79 文字より長くても構いませんが、各セグメントの最大長は 79 文字です。

ISC SYSMMSG プロセスへの入力によって作成され、マスター端末に出力されるシステム・メッセージは、単一セグメントおよび複数セグメントの出力メッセージを作成します。ATTRPRN および ATTRDPN が指定されている ISC SYSMMSG プロセスへの入力により作成されるシステム・メッセージは、少なくとも 5 つのセグメントを持つ複数セグメント・メッセージを処理できる MID を持つ必要があります。

#### 関連資料

589 ページの『SYSMMSG プロセス・ヘッダー』

以下の表では、SYSMMSG プロセス・ヘッダーの形式である SYSERROR FM ヘッダーおよび SYSSTAT FM ヘッダーが示されています。

## 第 31 章 ISC での MFS の使用

MFS DPM のメッセージ・フォーマット設定は、IMS システム定義 TERMINAL マクロで、パラメーター DPM-Xn を介して指定します (ここで X は A または B です)。ISC ノードの場合は、DPM-Bn の形式が常に使用されます。

### このタスクについて

このトピックでは、次の項目を説明します。

- IMS 入出力をフォーマット設定するための MFS DPM-Bn の呼び出し方法
- MFS が ATTACH、SCHEDULER、およびリセット付加プロセス (RAP) 機能管理ヘッダーを使用する方法
- SNA 定義 QMODEL およびデータ記述子機能管理ヘッダー (MFS 処理を制御) の MFS サポート

**制約事項:** MFS は、ISC TCP/IP リンクではサポートされません。

いくつかの入力 MFS フォーマット設定エラー (例えば、無効カーソル、誤った出力フォーマット、または出力フォーマットがない) は、IMS が入力メッセージに要求された応答を送信するまでは、IMS 内で検出することができません。このような場合には、IMS は ATTACH ATTDPN=SYSMSG を使用してエラー・メッセージを送信します。この ATTACH は、入力メッセージが方向変換および例外応答を指示したとしても生じ、ATTACH はメッセージに対して暗黙確認応答です。

要求時ページ化出力中に IMS によって受信される ATTACH SYSMSG は、無効 MFS ページング要求として処理され、セッション終了の原因になります。受信された SYSMSG が廃棄されます。

### 関連資料

571 ページの『FM ヘッダー形式の参照』

各ヘッダーの長さは、1 バイトの長さフィールドによって定義されます。このフィールドの値には長さフィールド自体が含まれます。連結フラグは追加のヘッダーが続くかどうかを指示します。

## MFS 入力フォーマット設定の活動化

MFS を使用する場合、入力メッセージはメッセージ記述子とフォーマット設定記述子により処理することができます。

### このタスクについて

IMS が入力メッセージを ISC 論理装置から受信する場合には、MFS が入力コンポーネント (ATTDSP) で定義され、MID 名がメッセージを伴うのでなければ、基本編集または ISC 編集が実行されます。MID 名はこれを、ATTACH 機能管理ヘッダーの ATTDPN パラメーターとして、または SCHEDULER FM ヘッダーの SCDDPN パラメーターとして組み入れることによって提供することができます。

MFS エスケープ文字 (//) は ISC ではサポートされません。MID 名がある時には、指定された MID および関連した装置入力形式 (DIF) を使用して、MFS がメッセージを編集します。

## MFS 出力フォーマット設定の活動化

MFS 出力フォーマット設定は、出力メッセージに関連したメッセージ出力記述子 (MOD) がある時に行われます。

### このタスクについて

MOD は、以下の方法のいずれか 1 つでセッションを提供します。

- アプリケーション・プログラムが出力メッセージに MOD 名を提供している。
- 入力メッセージがメッセージ入力記述子 (MID) により処理され、その定義が出力のフォーマット設定に MOD 名を指定している。

- 出力メッセージが MID により作成されるメッセージ通信であり、その定義が MOD 名を出力編集に指定している。

出力メッセージと関連した MOD がない場合には、MFS 編集は行われません。ATTACH および SCHEDULER パラメーターの値は、これらのそれぞれのヘッダー・タイプで定義されたデフォルト値です。

MOD を使用して、出力の結果としての入力をフォーマット設定するのに使用される、次の MID 名を指定することができます。ISC セッションでは、IMS はフォーマット設定された IMS 出力メッセージと後続の入力メッセージの関係性を制御することはできません。したがって、適切な MID 名が入力 MFS フォーマット設定に確実に使用されるようにするために、リモート・サブシステムまたはユーザー・アプリケーション・プログラムによる介入が必要になります。ATTACH および SCHEDULER RDPN/DPN パラメーターは、出力の次の MID を外部化し、後続の入力の MFS フォーマット設定を呼び出す ISC 機能を実行します。

#### 関連概念

MFS 制御ブロック間の関係 (アプリケーション・プログラミング API)

#### 関連資料

575 ページの『ATTPDN』

宛先プロセス名 (ATTPDN) パラメーターは、明示的または暗黙的に、ハーフセッションに付加される入力プロセス名を指定します。

## MFS 分散表示管理 (DPM) メッセージ

IMS MFS DPM メッセージは、ATTACH パラメーターで示されるように、単一または複数の SNA チェーンのどちらかで構成することができます。

ページング OPTIONS=MSG で定義される装置フォーマットを使用する MFS DPM 入力または出力では、IMS はメッセージ全体を 1 つ以上の関連伝送の単一のチェーンとして送信または受信します。ページング OPTIONS=DPAGE または PPAGE で定義される装置フォーマットを使用する MFS DPM 出力では、IMS は各論理または表示ページを 1 つ以上の関連伝送の単一のチェーンとして送信します。

MFS 出力 DPM フォーマット定義により、ページ化出力 (OPTIONS=PPAGE または DPAGE) のオプションを要求時ページ化または自動ページ化として送信することもできます。要求時ページ化出力には、各出力ページの他のハーフセッションに対するページング要求が必要です。自動ページ化出力は、他のハーフセッションに対するページング要求なしの一連の連続ページ (チェーン) として送信されます。

ページング OPTIONS=DPAGE で定義される装置フォーマットを使用する MFS DPM 入力によって、IMS は任意で各論理ページを 1 つ以上の関連伝送の単一チェーンとして受信することができます。自動ページ化入力のすべてのチェーン (複数の関連入力チェーンとしての複数の論理ページ) は IMS によるページング要求なしで連続して受信することができます。

FM ヘッダーは、メッセージの各入出力チェーンの最初のまたは唯一の伝送の一部として発生できます。

## MFS ページ削除機能

MFS ページ削除機能は ISC セッションによって使用することはできません。

QMODEL- 体系ページング要求以外のページング要求がサポートされないため、OPTIONS=NPGDEL は IMS システム定義 TERMINAL マクロ・ステートメントまたは ETO ユーザー記述子で強制されます。これには MFS オペレーター制御テーブル要求、および出力要求時ページング中に発生することがあるその他の入力データの形式が含まれます。要求時ページ化出力中に、有効な QMODEL ページング要求以外のデータが受信された場合には、エラー (例外応答および該当の ERP メッセージ) が起こります。エラー状態の結果として出力メッセージはキューに戻され、次の機会に再送されます。出力要求時ページ化メッセージが会話型または応答モード出力である場合には、即時に再送され、出力メッセージが正常に伝送されるまで、または有効な QMODEL ページング要求ないし適切な ERP 処置によってキューから除去されるまで、入力は許可されません。



## MFS オンライン・エラー検出

このトピックでは、IMS が MFS の入力エラー、出力エラー、およびページング・エラーを検出する方法を説明します。

### 出力エラー

IMS がすでに応答を以前の入力に送信した後で、MFS 出力エラーが検出されます。MFS MOD または DOF ブロック選択中にエラーが検出された場合には、エラー・メッセージが ATTACH SYSMSG として送信され、IMS メッセージは再送のためにメッセージ・キューに戻されます。MFS テスト・モードが有効な場合には、これがリセットされます。

無効なページ要求は、エラー・メッセージが送信されます。

### 入力エラー

エラー・メッセージが送信され、以下の 1 つが起こった場合には入力メッセージは拒否されます。

- MFS MID または DIF ブロック選択中にエラーが検出された。
- データ記述子 FM ヘッダーにゼロでないバージョン ID があり、この ID が MFS 記述子内のバージョン ID と一致しない。
- DPAGE 選択中にエラーが検出された (すなわち、DPAGE ラベルと DD FM ヘッダーの DSN、または COND= とデータを突き合わせて条件が満足されない)。

次のエラーのうちの 1 つが複数の DPAGE 入力中に検出された場合には、エラー・メッセージが他のサブシステムに送信され、入力メッセージは拒否されます。

- 複数の伝送チェーン。選択済み DPAGE に定義されたより多くのデータがチェーン中にあります。
- マップされた入力 LPAGE にデータ・セグメントがないものがある (例えば、すべてのセグメントを取り消すセグメント・ルーチンの結果として)。

次のエラーのうちの 1 つが単一の DPAGE 入力中に検出された場合には (つまり、複数の DPAGE 入力 MFS 定義で要求されない)、エラー・メッセージが送信され、入力メッセージが拒否されます。

- 単一の伝送チェーンが受信され、選択済み DPAGE で定義されたより多くのデータが含まれます。
- 複数の伝送チェーンが伝送されます。
- マップされた入力メッセージにデータ・セグメントがない (例えば、すべてのセグメントを取り消すセグメント・ルーチンの結果として)。

入力メッセージがユーザー・セグメント編集出口ルーチンによって取り消された場合、あるいはユーザー・セグメント編集出口ルーチン障害が検出された場合には、エラー・メッセージは他のサブシステムに送信されます。後者の場合には、入力メッセージは拒否されます。

### ページング・エラー

以下のような無効ページング要求 (QMODEL FM ヘッダー) が検出された場合には、エラー・メッセージが他のサブシステムに送信されます。

- QGETN または QGET 機能管理ヘッダーの QNAME パラメーターが、IMS によって送信される ATTACH または SCHEDULER FM ヘッダーの ATTDQN/SCDDQN パラメーターと一致しない。付加された要求時ページ化出力メッセージは再伝送のためにメッセージ・キューに戻されます。スケジュールされた要求時ページ化出力メッセージは、適切な QNAME が指定された QGETN を引き続き待ちます。
- QGETN または QGET FM ヘッダーの QORG パラメーターが無効である。出力メッセージは再伝送のためにメッセージ・キューに戻されます。
- QGETN FM ヘッダーが受信され、進行中の出力メッセージがない。

- QGETN 以外のページング要求が、スケジュールされた要求時ページ化出力 (ATTACH SCHEDULER) の後の最初の入力として受信されるか、あるいはページング要求での無効カーソルのために、QSTATUS FM ヘッダーが IMS によって送信される。
- QGETN FM ヘッダーが OLP 要求時ページ化出力のために受信され、現行カーソル位置が最後のページになっている。

次のような場合には、QGET FM ヘッダーによって QSTATUS またはエラー・メッセージが送信されることとなります。

- メッセージ間。
- 非 OLP 要求時ページ化出力メッセージが進行中の場合。エラー・メッセージが送信され、出力メッセージが再伝送のためにキューに入れられます。
- カーソル値に、有効な 2 バイトの 2 進数が含まれていない。エラー・メッセージが送信され、出力メッセージが再伝送のためにメッセージ・キューに入れられます。
- カーソル値が出力メッセージの範囲外である。

#### 関連資料

569 ページの『QSTATUS FM ヘッダー』

QSTATUS FM ヘッダーは、入力 QPURGE FM ヘッダーへ応答して IMSS により送信されるか、あるいは無効カーソルが要求時ページ化出力のページ要求で検出されたために送信されます。

557 ページの『エラー・リカバリー・プロシージャ FM ヘッダー』

IMS は FMH7 ERP ヘッダーおよびその関連メッセージを、選択受信側 ERP 例外応答 (X'0846') の後に続けて送信します。

## MFS での ATTACH および SCHEDULER FM ヘッダー

すべての入出力メッセージには、暗黙または明示の ATTACH FM ヘッダーが含まれ、任意で SCHEDULER FM ヘッダーが含まれます。

MFS DPM は、入力データ・ストリームに入力経路指定パラメーター (ATTRDPN/SCDRDPN、ATTRPRN/SCDRPRN、および ATTPRN/SCDPRN) を挿入させるためにフォーマット定義オプションを提供します。入力 ATTDPN/SCDDPN は入力メッセージをフォーマット設定するために使用される MFS MID です。MFS DPM は、これらのヘッダーにユーザー定義情報を挿入できるようにするための出力フォーマット定義オプションを提供しています。

ATTACH FM ヘッダーおよび SCHEDULER FM ヘッダーが必要な場合には、要求時ページ化出力の最初の伝送チェーンの単独エレメントとしてデータなしで送信されます。要求時ページ化出力に対する最初の入力ページング要求には、ATTACH (DPN=QMODEL の場合) を含める必要があり、同じ要求時ページ化出力メッセージの後続の入力ページ要求は、オプションとして ATTACH (これも DPN=QMODEL を示す必要がある) を含むことができます。その他のすべての MFS 入出力メッセージでは、ATTACH (および必要な場合には SCHEDULER) FM ヘッダーがデータ付きでメッセージの最初または唯一の伝送チェーンの最初のまたは単独のエレメントとして送信されます。

MFS は次のような ATTACH および SCHEDULER FM ヘッダー・パラメーターを使用します。

- ATTDP および ATTIU。これらの ATTACH パラメーターは、メッセージが要求時ページ化かどうか、あるいはそれが単一チェーンか複数のチェーンかを示します。
- ATTDDBA。この ATTACH パラメーターは、出力データ・ブロック化アルゴリズムまたは入力データ非ブロック化アルゴリズムを IMS によって使用することを指示します。入力時に非ブロック化されたデータ・エンティティは、MFS に対する入力データ・レコードになります。MFS は次にページおよびセグメントで構成される標準 IMS メッセージを作成します。出力時に、MFS レコードは出力 VLVB ブロック化用のデータ・エンティティになります。例外は MFS DPM ストリーム・モードで、これはチェーン (ATTDDBA) 出力として送信されます。
- ATTDPN/SCDDPN および ATTPRN/SCDPRN。ATTDPN/SCDDPN パラメーターは、入力時に MFS 入力フォーマット設定を活動化するために使用されます。ATTPRN/SCDPRN は入力データ・ストリーム中に挿入することができ、したがってアプリケーションで使用できます。IMS からの出力メッセージ応答には、ATTDPN/SCDDPN または ATTPRN/SCDPRN を、MFS フォーマット設定記述で指定したように、MFS によって処理される出力メッセージの ATTACH ヘッダーまたは SCHEDULER FM ヘッダーに挿入すること

ができます。これらは、ソースの入力メッセージ ATTACH ヘッダーまたは SCHEDULER FM ヘッダーから折り畳まれた可能性がある、ATTRDPN/SCDRDPN および ATTRPRN/SCDRPRN を上書きします。

- ATTRDPN/SCDRDPN。ATTRDPN/SCDRDPN はオプションとして入力データ・ストリーム中に挿入することができ、したがって、アプリケーションで MFS DPM によって処理される入力メッセージに使用できます。戻り宛先プロセス名は、MFS によって処理される出力メッセージの出力 ATTACH ヘッダーまたは SCHEDULER FM ヘッダーに挿入することができます。出力メッセージへの応答と関連した MFS 提示戻り宛先プロセスは、MOD の NXT= オペランドで指定された MID 名です。
- ATTRPRN/SCDRPRN。戻り 1 次リソース名 (RPRN) は、MFS によって処理される出力メッセージの ATTACH ヘッダーまたは SCHEDULER FM ヘッダーに挿入することができます。これは MFS 定義を使用して指定することができます。これが送信されたメッセージに対する応答とともに IMS に戻された場合は、これは入力応答を受信するための宛先 (トランザクションまたは LTERM) を決定します。
- ATTDQN/SCDDQN。宛先キュー名はメッセージ ID として要求時ページ化出力時に IMS によって送信され、要求時ページ化メッセージと関連したすべての要求で戻されなければなりません。

非同期要求時ページ化出力メッセージの最初のチェーン (ATTACH SCHEDULER) が正常に送信された場合には、出力キュー (IMS LTERM) は入力または出力に対して自動的にロックされます。この機能はまた、要求時ページ化要求でのカーソル・エラーのために、IMS によって QSTATUS FM ヘッダーの後に自動的に送信されます。ページ要求 (ATTACH QMODEL、QNAME 付きの QGETN) が受信され、データの最初のページが ATTACH ヘッダーまたは SCHEDULER FM ヘッダーの再送なしに送信される時には、キュー・ロック状態がリセットされます。キュー・ロック状態はまた、セッション障害および後続の再始動を通してリセットされます。IMS がこのページ要求 (QGETN) を待っている間、会話モードまたは応答モードではなかったとして、IMS は他の LTERM のメッセージを送信または受信することができます。出力 LTERM がロックされている間、ロックされたキューに出力を作成する ISC セッションからの入力は受け入れられません。その理由は、IMS システム定義 NAME マクロ・ステートメントまたは ETO ログオン記述子を使用して、ロックされたキューに定義されている入力 / 出力コンポーネントの関係のためです。他の LTERM の出力を作成する入力は受け入れられます。

## データ記述子 FM ヘッダー

入出力データ記述子 FM ヘッダーを使用して、データ構造名およびバージョン ID またはフィールド・タブ分離文字を受信および送信します。

### 入力データ記述子 FM ヘッダー

IMS への入力では、このヘッダーは ATTACH ヘッダーまたは SCHEDULER FM ヘッダーの後に続けることができ、データ構造名 (DPAGE 選択が DPAGE 名による場合) および、バージョン ID または入力フィールド・タブ分離文字を受信するのに使用されます。

入力 DPAGE 選択を DPAGE 名で実行する場合は、入力データ記述子 FM ヘッダーを IMS に送信する必要があります。これはつまり、OPTIONS=DNM を DIV ステートメント (TYPE=INPUT) で指定した場合です。

OPTIONS=DNM をセットし、DPAGE 名を指定しないか、誤った DPAGE 名を指定するとエラーが起きます。

MFS が、データをマップするのに正しい定義が使用されていることを検査する場合には、入力メッセージの最初のまたは単独の FM ヘッダーで IMS にバージョン ID が送信されます。バージョン ID が送信され、X'0000' である場合には、検査は行われません。MFS が FTAB 機能で最大 8 個の分離文字を提供するため、フィールド・タブ分離文字は入力時には必要ありません。入力時に受信された場合には、フィールド・タブ分離文字は現行の伝送用の MFS FTAB 仕様 (もしあれば) の代わりに使用されます。

#### 関連資料

579 ページの『データ記述子 FM ヘッダーの形式』



以下の表では、入出力のデータ記述子 FM ヘッダーの形式を示します。

## 出力データ記述子 FM ヘッダー

IMS からの出力では、データ記述子 FM ヘッダーは、DIV または DPAGE ステートメントの OFTAB= パラメーターが指定される場合に、データ構造名およびバージョン ID または出力フィールド・タブ分離文字を送信するのに使用されます。

このヘッダーは、非ページ化出力メッセージの単独の伝送チェーンまたはページ化出力メッセージの各伝送チェーンで送信されます。非要求時ページ化メッセージ (OPTIONS=MSG) では、このヘッダーは ATTACH ヘッダーまたは SCHEDULER FM ヘッダーの後にあります。自動ページ化メッセージでは、このヘッダーは ATTACH ヘッダーまたは SCHEDULER FM ヘッダーの後にあり、最初の論理または表示ページのデータの前にあります。データ記述子 FM ヘッダーは出力の各追加ページのデータの前にあり、伝送チェーンにおける単独 FM ヘッダーです。要求時ページ化出力の場合、このヘッダーは QFXR ヘッダーの後にあります。バージョン ID は、出力メッセージの最初のまたは単独の FM ヘッダーの各メッセージに対し一度だけ送信されます。さらに、OFTAB が指定され、OPTIONS=DNM が要求される場合には、現行の伝送に使用される出力フィールド・タブ分離文字が FM ヘッダーに組み込まれます。OPTIONS=NODNM が指定された場合には、OFTAB が定義されていたかどうかにかかわらずデータ記述子 FM ヘッダーは送信されません。

### 関連資料

579 ページの『データ記述子 FM ヘッダーの形式』

以下の表では、入出力のデータ記述子 FM ヘッダーの形式を示します。

## 要求時ページ化メッセージの制御 : QMODEL FM ヘッダー

キュー・モデル (QMODEL) ヘッダーは、要求時ページ化メッセージを制御するために送信および受信されます。すべての要求時ページング要求 (RAP 以外) は、QMODEL 定義済み FM ヘッダーを使用して行わなければならないなりません。

**制約事項:** MFS DPM-Bn は、要求時ページ化出力の送信中にはオペレーター制御テーブル要求または入力メッセージをサポートしません。

IMS がサポートする QMODEL ヘッダーを以下の表にリストしています。

QMODEL ヘッダー	IMS <sup>566</sup> ページの『1』によって送信された	IMS <sup>566</sup> ページの『2』によって受信された
QXFR	X	
QGETN		X
QGET		X
QPURGE		X
QSTATUS	X	

注:

- QMODEL ヘッダーは 2 つの IMS サブシステム間では送信されません。
- QMODEL FM ヘッダーは、最初のページの QMODEL ページ要求の前に置かれなければならない。これらの FM ヘッダーの前に、値 X'03' を含む DPN パラメーターを持つ QMODEL ATTACH FM ヘッダーを置くことができます。

QXFR は QMODEL 応答です。他のすべてのヘッダーは QMODEL 要求です。論理装置間の各ハーフセッションは、セッション・バインド・パラメーターを使用して、QMODEL 要求を受信するために QMODEL プロセスが使用できるかどうかを示すことができます。バインド・パラメーターは QMODEL 応答に影響を与えません。IMS は常に、「QMODEL 使用可能」を指示し、IMS 要求時ページ化出力に対する QMODEL (ページング) 要求を受信する準備ができています。もう一方のハーフセッションは「QMODEL 使用可能」を示す

場合も、示さない場合もあります。IMS によって送信できるのは QMODEL 応答だけなので、IMS は QMODEL 要求の他のハーフセッション・バインド指示を無視します。

キュー・モデルは LU 6.1 プロトコルを使用して次のようなメッセージを処理します。

- SNA は複数のメッセージをアクティブにすることができますが、IMS メッセージは逐次にアクティブになります。つまり、与えられた時間にアクティブにされるのは 1 つのメッセージだけです。アクティブ・メッセージの処理が完了すると、別のメッセージを処理することができます。同期 (SCHEDULER なしの ATTACH) 出力 MFS DPM 要求時ページ化メッセージは、最初のチェーン (ATTACH) の送信時に即時にアクティブになったと見なされます。ATTACH SCHEDULER を使用して送信される要求時ページ化出力は、最初のページ要求が IMS に戻されるまで、アクティブになりません。
- MFS 要求時ページ化出力は、線形 (OPTIONS=DPAGE) および階層 (OPTIONS=PPAGE) という 2 つのタイプのメッセージ編成のために提供されます。両方のタイプでは、ページを順次に (次を獲得する) または線形に (カーソルによって獲得する) リトリブすることができます。しかし、線形リトリブはオペレーター論理ページング (OLP) が定義されている場合にだけ許可されます。階層検索はサポートされません。

#### 関連資料

[528 ページの『データ・フロー制御プロトコルの関連情報』](#)

以下のトピックで、データ・フロー制御 (DFC) のバイト・レベルのプロトコルについて説明します。プロトコルを以下に示します。

## 要求 (入力) QMODEL FM ヘッダー

以下のトピックでは、QGETN、QGET、および QPURGE FM ヘッダーについて説明します。

### QGETN FM ヘッダー

IMS によって受信される QGETN FM ヘッダーによって、MFS DPM 要求時ページ化出力の単一論理または表示ページが順次転送されます。これはチェーンの単独 QMODEL FM ヘッダーで、機能管理データが後に続くことはありません。

QGETN FM ヘッダーは IMS によって送信されません。

QGETN FM ヘッダーは次の場合に、IMS に送信できます。

- 要求時ページ化出力の最初のページを検索する場合。QGETN は常にスケジュールされた要求時ページ化出力の受信後の最初の要求でなければなりません。この前に ATTACH QMODEL FM ヘッダーがなければなりません。
- スケジュールされた要求時ページ化出力の無効カーソルを指示する QSTATUS FM ヘッダーに続いている場合。
- 非 OLP 要求時ページ化出力メッセージが進行中の時。
- OLP 要求時ページ化出力メッセージが進行中で、現行カーソル位置がメッセージの最後のページにない時。

上記のそれぞれの事例は、QXFR FM ヘッダーが応答で戻される原因となります。

QGETN FM ヘッダーは次の場合に、IMS に送信できません。

- メッセージ間。エラー・メッセージが戻されます。
- OLP 要求時ページ化出力メッセージが進行中で、現行カーソル位置が最後のページにある時。

#### 関連資料

[569 ページの『QSTATUS FM ヘッダー』](#)

QSTATUS FM ヘッダーは、入力 QPURGE FM ヘッダーへ応答して IMSS により送信されるか、あるいは無効カーソルが要求時ページ化出力のページ要求で検出されたために送信されます。

[582 ページの『QGETN FM ヘッダーの形式』](#)

以下の表には、QGETN FM ヘッダーの形式が示されています。

## QGET FM ヘッダー

QGET FM ヘッダーは、カーソルにしたがって、次のページ、最後のページ、または任意の論理ページに移るために、IMS への入力として使用します。

QGET FM ヘッダーが有効なのは、定義済みオペレーター論理ページングの要求時ページ化出力の場合だけです。これは入力チェーンの単独 FM ヘッダーで、機能管理データが後に続くことはありません。

QGET FM ヘッダーは IMS によって送信されません。

QGET FM ヘッダーは次の場合に、IMS に送信できません。

- メッセージ間。エラー・メッセージが戻されます。
- スケジュールされた要求時ページ化出力メッセージの ATTACH の受信後。出力のこのタイプの最初のページング要求は QGETN でなければなりません。QGET は、QGETN を使用して最初の出力ページを要求した後で使用することができます。
- 非 OLP 要求時ページ化出力メッセージが進行中の時。エラー・メッセージが戻されます。出力メッセージが再伝送のためにメッセージ・キューに入れられます。
- カーソル値に、有効な 2 バイトの 2 進数が含まれない時。エラー・メッセージが戻され、出力メッセージが再伝送のためにメッセージ・キューに置かれます。
- カーソル値が出力メッセージ用の範囲外の時。

### 関連資料

569 ページの『QSTATUS FM ヘッダー』

QSTATUS FM ヘッダーは、入力 QPURGE FM ヘッダーへ応答して IMSS により送信されるか、あるいは無効カーソルが要求時ページ化出力のページ要求で検出されたために送信されます。

581 ページの『QGET FM ヘッダーの形式』

以下の表には、QGET FM ヘッダーの形式が示されています。

## QPURGE FM ヘッダー

IMS が QPURGE ヘッダーを受信した場合には、進行中の要求時ページ化出力メッセージ上の処理を停止します。

QPURGE は MFS 要求時ページ化出力メッセージがアクティブな時にだけ、IMS により受信することができます。QPURGE FM ヘッダーは、チェーンの単独 FM ヘッダーで、機能管理データが後に続くことはありません。IMS は QSTATUS で QPURGE FM ヘッダーに応答します。要求時ページ化出力メッセージを削除するために IMS が QPURGE ヘッダーを受信した場合、IMS が QSTATUS への応答で同期点応答を受信した後でのみ、このメッセージは削除されます。

QPURGE FM ヘッダーは IMS によって送信されません。

### 関連資料

583 ページの『QPURGE FM ヘッダーの形式』

以下の表には、QPURGE FM ヘッダーの形式が示されています。

## 応答 (出力) QMODEL FM ヘッダー

以下のトピックでは、QXFR および QSTATUS FM ヘッダーについて説明します。

## QXFR FM ヘッダー

QMODEL ヘッダーによる有効なページング要求へ応答して、要求時ページ化出力メッセージの各出力伝送チェーンで QXFR ヘッダーが送信されます。

QXFR FM ヘッダーは後にデータ記述子 (DD) FM ヘッダー (MFS 形式定義を基にした) を続けることができ、論理または表示ページ・データを含みます。このヘッダーを IMS に送信してはいけません。

#### 関連資料

585 ページの『QXFR FM ヘッダーの形式』

以下の表には、QXFR FM ヘッダーの形式が表示されています。

## QSTATUS FM ヘッダー

QSTATUS FM ヘッダーは、入力 QPURGE FM ヘッダーへ応答して IMSS により送信されるか、あるいは無効カーソルが要求時ページ化出力のページ要求で検出されたために送信されます。

QSTATUS が要求時ページ化出力中の QPURGE の結果生じた場合には、同期点を要求して送信されます。同期点応答が受信された時に、メッセージはキューから除去されます。例外同期点応答を受信した場合には、メッセージはキューに戻され、再送に使用できます。

スケジュールされた要求時ページ化出力中の無効カーソルの結果生じた QSTATUS に RQE1/CD が送信され、後続の QGETN がメッセージの最初のページを再要求できるようにするか、QPURGE がメッセージをデキューできるようにします。

QSTATUS FM ヘッダーは、チェーンの単独 FM ヘッダーで、機能管理データが後に続くことはありません。QSTATUS FM ヘッダーを IMS に送信してはいけません。

#### 関連資料

584 ページの『QSTATUS FM ヘッダーの形式』

以下の表には、QSTATUS FM ヘッダーの形式が示されています。

## MFS での RAP FM ヘッダー

MFS によって使用される時には、リセット付加プロセス (RAP) FM ヘッダーを IMS に送信して進行中の要求時ページ化出力メッセージを削除することができます (これ以上の処理を妨げる)。

このヘッダーは、NEXTMSG の MFS オペレーター制御機能と等価です。このヘッダーを IMS に送信して、進行中のオペレーター論理ページ化メッセージを削除することもできます。RAP FM ヘッダーの後に機能管理データを続けることはできません。RAP FM ヘッダーは IMS によって送信されません。

#### 関連資料

588 ページの『リセット付加プロセス (RAP) FM ヘッダーの形式』

以下の表には、リセット付加プロセス (RAP) FM ヘッダーの形式が示されています。



## 第 32 章 FM ヘッダー形式の参照

各ヘッダーの長さは、1 バイトの長さフィールドによって定義されます。このフィールドの値には長さフィールド自体が含まれます。連結フラグは追加のヘッダーが続くかどうかを指示します。

ヘッダー形式内では、可変長および固定長パラメーターはコマンド・コード別に定位置になっています。各可変長定位置パラメーターの前に 1 バイトの長さフィールドがあります。この長さフィールドに含まれる値は X'00' から X'08' で、長さフィールド自体は含みません。長さフィールドに X'00' が含まれている場合には、変数パラメーターが省略され、次の定位置可変長パラメーターの長さフィールドが現れ、その可変長パラメーター・フィールドが後に続きます。

ヘッダーの終わりにある末尾定位置パラメーター長さフィールドの X'00' は、IMS への入力から除去することができ、さらに IMS からの出力では送信されません。また、IMS は、上記の名前のどれについても末尾ブランクを送信しません。

ヘッダーの受信時、影響されやすい最後のパラメーターを超えてコード化されたパラメーターを IMS は無視します (つまり、IMS がサポートする)。その他の LU 6.1 サブシステムは、IMS が送信するヘッダーを受信する時にはこの手順に従う必要があります。

### 関連資料

659 ページの『ISC データ・フロー制御の例』

以下のトピックでは、ISC データ・フロー制御の例を示します。

591 ページの『ISC 編集 ATTACH パラメーターの使用の例』

以下のトピックでは、ISC 編集 ATTACH パラメーターの使用例を示します。

## ATTACH FM ヘッダーの形式

ATTACH FM ヘッダーの形式が、以下の表に定義されています。

バイト	ビット	名前	内容
0		FMHL	
1	0	FMHC	
	1-7	FMHT	B'0000101'
2-3		FMH5CMD	X'0202'

表 104. ATTACH FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容		
4	0-3	FMH5MOD			
	4		予約済み		
	5	ATTDP	セッション・ローカル・フラグ <b>B'0'</b> 要求時ページングでない <b>B'1'</b> 要求時ページング		
6-7		ATTIU	交換単位コード <b>B'00'</b> 複数チェーン <b>B'01'</b> 単一チェーン <b>B'10'</b> 予約済み <b>B'11'</b> 予約済み		
			5	FMH5FXCT	固定長パラメーターの長さは X'02' でなければなりません。
			6	ATTDSP	データ・ストリーム・プロファイル中の 1 バイトの 16 進数入力値。(IMS は、X'00' から X'03' までの値を要求してコンポーネント 1-4 をそれぞれに選択します。接続マネージャーがリセット状態の時に ATTACH FM ヘッダーが受信されなかった場合には、値 X'00'、またはコンポーネント 1 と見なされます。)
			7	ATTDBA	アルゴリズムを処理するアプリケーション・データ <b>X'00'</b> 未定義 (IMS のみへの入力。接続マネージャーがリセット状態の時に ATTACH FM ヘッダーが受信されなかった場合には、IMS もまた、これをデフォルトと見なします。) <b>X'01'</b> 可変長、可変ブロック化 (IMS の入出力) <b>X'02'</b> SCS の文書サブセット (IMS によってサポートされない) <b>X'03'</b> SCS のカード・サブセット IMS によってサポートされない) <b>X'04'</b> RU のチェーン (IMS での入力 / 出力) <b>X'05'</b> RU (IMS に対する入力のみ) <b>X'06' から X'FF'</b> 予約済み



表 104. ATTACH FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容
8-m		ATDDPN <sup>573</sup> ページの『1』	開始されるプロセスの名前。SNA 定義プロセスの場合、DPN は 1 バイト、非図形 16 進文字です。サポートされる SNA プロセスは次のとおりです。 <b>X'01'</b> SYMSMG <b>X'02'</b> SCHEDULER <b>X'03'</b> QMODEL  IMS 処理については、ATDDPN パラメーターの項で説明されています。
m+1-n		ATTPRN <sup>573</sup> ページの『1』	開始されているプロセスの 1 次リソースの名前
n+1-p		ATTRDPN <sup>573</sup> ページの『1』	提示された戻りプロセス名の名前
p+1-q		ATTRPRN <sup>573</sup> ページの『1』	戻りプロセスのために提示された 1 次リソースの名前
q+1-r		ATTDQN <sup>573</sup> ページの『1』	付加プロセスと関連したキューの名前
r+1-s		ATTACC	アクセス・コード IMS により無視される。

**注:**

1. ATDDPN が SCHEDULER (X'02') にセットされた場合には、残りの ATTACH パラメーターを指定してはなりません。この場合は、アスタリスク付きの各パラメーターの等値が、連結された SCHEDULER FM ヘッダーに現れることがあります。これらの SCHEDULER パラメーターの接頭部は、「ATT」でなく「SCD」ですが、ATTACH と同じ定義および規則に従います。

**ATTIU**

交換単位コード (ATTIU) は、単一の入力または出力 IMS メッセージが、1 つ以上の SNA チェーンで構成されるかを示します。

MFS 自動ページ化メッセージでないすべての入力メッセージは、単一または複数のチェーンのどちらかを示しますが、メッセージ終了を示すのは 1 つの実際の伝送チェーンに制限されます。MFS 自動ページ化入力メッセージは、メッセージが実際に 1 つ以上の SNA チェーンかどうかに関係なく、複数のチェーンを示すことができます。MFS は、単一のチェーン・メッセージが複数の DPAGE を作成できるオプションも提供します。自動ページ化ではない非 MFS および MFS 出力は、常に単一チェーンを指示する ATTIU で単一伝送チェーンとして送信されます。MFS 自動ページ化出力メッセージは常に、複数チェーンを示す ATTIU で送信されます。

**制約事項:** IMS は、連続入力または出力メッセージ間の関係をサポートしません。したがって、複数チェーンを示す交換単位コードを使用して、SCHEDULER プロセスを使用してスケジュールされる IMS バッチ入力メッセージや連続メッセージを送信することはできません。

IMS は、少なくとも EB が指定した 1 つのチェーンになる付加された (同期) 入力メッセージ通信を要求します。

**制約事項:** IMS は、複数の SNA チェーンの MFS 自動ページ化同期入力メッセージ通信をサポートしません。

ATTIU パラメーターは、自動的に入力または出力メッセージ間で保存されるわけではありません。出力メッセージが複数チェーンで、MFS によってフォーマット設定される場合には、このパラメーターは各入力

メッセージで、または MFS により明示的に提供されなければなりません。ATTIU の入力リセット状態は、"複数チェーン" です。

## ATTDSP

データ・ストリーム・プロファイル (ATTDSP) パラメーターは、QMODEL を付加する時には無視されます (ATTDPN=X'03')。その他の入力中、ATTDSP フィールドは特定の IMS 入力コンポーネントを指示します。

入力コンポーネントを使用して、セッションに割り振られた LTERM のセット (LTERM サブプール) から入力 IMS LTERM を識別します。入力 DSP 値は、有効な入力コンポーネント (その入力コンポーネント値を使用して定義された IMS LTERM) を示す必要があります。そうでない場合には、入力は拒否されます。

いくつかの内部 IMS 機能は入力コンポーネントに定義される LTERM と関連しています。例えば、LTERM は IMS 内で特定の発信元 (入力) および宛先 (出力) パスを定義します。この出力宛先は、入力発信元 LTERM と同じかまたは別のものにでき、IMS 内の安定したアプリケーションとオペレーター参照点を表します。

静的に定義された端末では、入力 LTERM に入力端末セキュリティを定義することができます。このセキュリティ許可は各 LTERM で固有にすることも、IMS 内のセキュリティ・レベル、またはセキュリティ・グループを表すこともできます。入力端末セキュリティは IMS 端末、トランザクション、およびコマンドへのアクセスを認可します。

入力 ATTACH DSP パラメーターは次の状態になるまで有効です。

- 別の ATTACH によって変更された
- システム障害によってリセットされた
- ERP によって最後の同期点でその値にバックアウトされた
- ブラケット間状態でリセットされた
- RAP FM ヘッダーによってリセットされた

ATTDSP パラメーターの入力リセット状態は、X'00' または「コンポーネント 1」です。

IMS は IMS システム障害にまたがってアクティブ入力 ATTDSP を保管していないため、ATTACH FM ヘッダーを使用して ATTDSP を提供し、"イン・ブラケット" (再始動可能) を示すバインドまたは交渉可能バインド応答の後の受信プロセスを復元する必要があります。

出力 LTERM は、メッセージ通信およびブロードキャスト・メッセージに関しては、入力用に選択された LTERM またはメッセージ送信側のいずれかにより決定されます。また、IMS では、受信メッセージ処理プログラム (MPP) は CHNG 呼び出しを行って、出力を変更可能代替 PCB に挿入することによって、結果の出力の宛先を変更することもできます。入力 (例えばコマンドなど) からの直接の結果としてのシステム・メッセージについては、選択される出力 LTERM は、入力が発信された LTERM に定義された通常の宛先 LTERM です。

出力中に、DSP フィールドは出力 LTERM と関連した出力コンポーネントの値に設定されます。

IMS メッセージ形式サービスは、コンポーネント 1 個ずつの単位で使用可能として定義することができます。出力コンポーネントは、ブラケットおよび送信/受信プロトコルも指定し、これは "SCHEDULER" を示す DPN パラメーターをもつ ATTACH を使用して送信される非同期出力で使用されます。

### 関連概念

[508 ページの『再始動プロセスの調整』](#)

ハーフセッションは、セッション障害後の再始動を調整するため、規則を使用します。

## ATTDBA

非ブロック化アルゴリズム (ATTDBA) は、使用するデータ・ブロック化および非ブロック化アルゴリズムを決定します。

このアルゴリズムは、MFS が使用されているかどうかに応じて、単一の GET 操作または PUT 操作の処理で渡されるデータの量を決定します。

IMS がメッセージを送信中で、MFS を使用中でない場合には、メッセージ・キューのメッセージの各セグメントが RU (または RU のチェーン) 内でデータ・エンティティになります。メッセージのセグメントが

メッセージ・キュー・レコード (LRECL) にまたがる場合には、またがったセグメントの各部分がデータ・エンティティになります。MFS が使用される場合には、MFS レコードの定義に従ってセグメントがブロック化され、データ・エンティティとして送信されます。

IMS がメッセージを受信した場合には、入力メッセージの各データ・エンティティがメッセージ・キューにメッセージのセグメントとして挿入されます。ISC が入力のスパン・キュー・セグメントをサポートしないために、最大のメッセージ・キュー LRECL は最大のデータ・エンティティまたは受信済み MFS データ・レコードを処理するのに十分な大きさでなくてはなりません。この LRECL 定義が超過した場合にはエラーが検出され、適切なエラー・メッセージが作成されます (DFS074)。また、メッセージ・キュー中の実際の使用可能スペースは、メッセージの変数接頭部項目によって減少します。

IMS は SNA によって定義された次の 4 つのアルゴリズムを実行します。

#### **UNDEFINED**

IMS の場合、以下の RU と同じです。

#### **RU**

入力 RU は付加プロセスに渡されるエンティティです。IMS はこの DBA 値を受信しますが、送信はしません。

#### **VARIABLE LENGTH, VARIABLE BLOCKED (VLVB)**

送信される各データ・エンティティの前には 2 バイトの長さフィールドがあり、RU サイズや境界に依存しないですむようになっています。複数のデータ・エンティティを単一の RU にブロック化でき、あるいはデータ・エンティティは複数の RU にまたがることができます。長さには長さフィールド自体の 2 バイトが含まれます。

#### **RU の CHAIN**

各データ・エンティティは、単一の SNA チェーンとして送信または受信されます。

IMS が受信する各メッセージは、ATTACH FM ヘッダーの ATTDDBA フィールドを基に非ブロック化されます。入力 ATTDDBA フィールド値はサポートされるアルゴリズムを示さなければならず、そうしないと、入力は拒否されます。

入力 ATTACH DBA パラメーターは次の状態になるまで有効です。

- 別の ATTACH によって変更された
- ERP によって最後の同期点でその値にバックアウトされた
- ブラケット間状態でリセットされた
- RAP FM ヘッダーによってリセットされた
- セッション障害によってリセットされた

ATTDDBA パラメーターの入力リセット状態は、「UNDEFINED」(「RU」と等値)です。

IMS は、IMS システム障害の後までアクティブ入力 ATTDDBA を記憶しないため、ATTACH FM ヘッダーを使用して ATTDDBA を提供し、「イン・ブラケット」(再始動可能)を示すバインドまたは交渉可能バインド応答の後の受信プロセスを復元する必要があります。

MFS ストリーム・モードを使用する出力を除いて、IMS からの出力は自動的に可変長、ATTACH ヘッダーまたは SCHEDULER FM ヘッダーでの可変ブロック (VLVB) アルゴリズムを指示しながら送信されます。この出力はチェーン・アセンブリーを指示して送信されます。

#### **関連概念**

[508 ページの『再始動プロセスの調整』](#)

ハーフセッションは、セッション障害後の再始動を調整するため、規則を使用します。

## **ATTDPN**

宛先プロセス名 (ATTDPN) パラメーターは、明示的または暗黙的に、ハーフセッションに付加される入力プロセス名を指定します。

付加されたプロセスの 1 つの責任は、受信サブシステムのために、そのサブシステム内の入力メッセージの宛先を決定することです。場合によっては、名前を指定されたプロセスがメッセージ宛先であることもあります。IMS 内では受信プロセス (基本編集以外) は、入力 1 次リソース名 (ATTPRN) が提供された場

合、これをメッセージ宛先トランザクションのキューまたは LTERM として使用します。ATTPRN が提供されなかった場合には、メッセージ宛先の判別に標準 IMS アルゴリズムが使用されます。

**関連資料:** メッセージ宛先を決定する IMS アルゴリズムの詳細については、「IMS V15 システム管理」の"IMS メッセージとスケジューリング"を参照してください。

付加されたすべてのプロセスはセッションと同期式に実行されます。しかし一部のプロセスは、IMS 内で非同期的に行われる追加の作業をスケジュールすることができます。付加されたプロセスは次の状態になるまで、付加されたままです。

- 別の ATTACH によって変更された
- ERP によって最後の同期点でその値にバックアウトされた
- ブラケット終了標識によってリセットされた
- RAP FM ヘッダーによってリセットされた
- システム障害によってリセットされた

ATTPDN パラメーターの入力リセット状態は「ISCEDT」です。

IMS は IMS システム障害の後までアクティブ入力 ATTPDN を記憶しないため、ATTACH FM ヘッダーを使用して ATTPDN を提供し、「イン・ブラケット」(再始動可能)を示すバインドまたは交渉可能バインド応答の後の受信プロセスを復元する必要があります。

ATTPDN パラメーターは、付加されるプロセスが次のいずれかであることを示す場合があります。

- ISC 編集 ('ISCEDT' またはユーザーが付けた別名)
- IMS 基本編集 ('BASICEDT')
- MFS フォーマット設定 (MFS MID 名)
- システム・メッセージ (SYSMSG)、X'01'
- SCHEDULER モデル、X'02'
- キュー・モデル (QMODEL)、X'03'

MFS および QMODEL を除いて、上記のプロセスは常に使用可能です。MFS は IMS システム定義中に入力コンポーネント (ATTDSP) に定義された場合にだけ使用可能です。ATTPDN が MFS MID 名で、MFS が使用不能の場合には、入力メッセージが拒否されます。QMODEL は、MFS 要求時ページ化出力メッセージに続く場合にのみ使用可能です。

ISCEDT (そして、IMS システム定義中にユーザーによって ISCEDT に定義された別名も) および BASICEDT は予約名であり、MFS MID 名または MOD 名として使用することはできません。これらの名前の使用の結果、MFS プロセスではなく名前付きプロセスが使用されます。

接続マネージャーがリセット状態の時に、ATTACH FM ヘッダー (または ATTPDN) が提供されなかった場合には、ISC 編集が選択されます。接続マネージャーがリセット状態でない時に、ATTACH FM ヘッダー (または ATTPDN) が提供されなかった場合には、「アクティブ」プロセスが使用されます。

IMS は戻り応答上の出力 ATTPDN を、直前の入力要求の ATTACH 内でオプションとして提供される ATTPDN パラメーターに含まれる値にセットします。システム・メッセージの送信時に、IMS は SYSMSG ATTPDN を必要な場所へ自動的に挿入します。MFS DPM はオプションとして、出力メッセージ形式記述子を使用して、出力 ATTPDN を指定、オーバーライド、または削除する手段を提供しています。ATTPDN パラメーターが出力で使用できない場合には、X'00'を含む1バイト・フィールドが出力 ATTACH FM ヘッダーに含まれます。

## 関連資料

[577 ページの『ATTPRN』](#)

ATTACH 1 次リソース名 (ATTPRN) パラメーターは、受信サブシステムの入力メッセージの宛先を表します。

[577 ページの『ATTRDPN および ATTRPRN』](#)

戻り宛先プロセス名 (ATTRDPN) および戻り 1 次リソース名 (ATTRPRN) パラメーターは、ソース・セッション内の戻りプロセスと戻り 1 次リソースを定義し、ソース・セッション内の応答を戻すための経路指定を可能にするために、結果の応答上でソース・セッションに戻す必要があります。



## ATTPRN

ATTACH 1 次リソース名 (ATTPRN) パラメーターは、受信サブシステムの入力メッセージの宛先を表します。

このパラメーターは、メッセージ要求で送信された ATTACH ATTPRN パラメーターの 応答メッセージを戻した結果として、リモート側で実行されたメッセージからの戻り応答上で送られます。しかし、メッセージ要求に入力 ATTPRN が含まれていない状況では、出力 ATTPRN パラメーターは、MFS を使用して、ユーザー定義情報に基づいて作成することができます。

IMS は IMS コマンド verb の仕様を PRN として許可しません。

この宛先は通常、アプリケーション・プログラムではなく端末 (LTERM) を表します。要求 ATTPRN/ 応答 ATTPRN がアプリケーション・プログラムを表す場合には、ソースのアプリケーション・プログラムとリモート・トランザクション実行の間の同期関係と非同期関係を比べて考慮してください。

さらに、応答のプロセスと経路指定に関する入力トランザクションのセキュリティーについても検討する必要があります。IMS への入力で提供された場合には、ATTPRN は IMS トランザクション・コードまたはメッセージ通信用の LTERM 名として使用されます。データ・ストリームは宛先およびセキュリティー情報には編集されません。ATTPRN がトランザクションを表している場合、およびパスワード・セキュリティー (これは静的に定義された端末にのみ適用される) がノードに定義されている場合には、入力パスワードで使用できるソースがないため、入力セキュリティー・エラーが起きます。

**関連資料:** パスワード・セキュリティーについては、「IMS V15 システム管理」を参照してください。

アプリケーションが以前にトランザクション・コードを SPA に挿入していた場合には、ATTPRN は IMS 会話モード中に拒否されます。ATTPRN は単一のメッセージ・インスタンスにのみ適用され、IMS /SET コマンドを使用して設定された「事前設定」宛先をオーバーライドはしますが破棄はしません。ATTACH FM ヘッダーがないか、または ATTPRN パラメーターがない ATTACH FM ヘッダーを持つ 後続のメッセージは、IMS /SET コマンドによって、または IMS トランザクション・コード、LTERM 名、あるいはデータまたは MFS フォーマット設定を通して提供されたコマンド verb によって、以前に設定された事前設定宛先を使用することができます。

IMS は戻り応答上の出力 ATTPRN を、直前の入力要求の ATTACH 内でオプションとして提供される ATTPRN パラメーターに含まれる値にセットします。

MFS DPM はオプションとして、出力メッセージ形式記述子を使用して、出力 ATTPRN を指定、オーバーライド、または削除する手段を提供しています。このパラメーターがない場合には、X'00' を含む 1 バイト・フィールドが出力 ATTACH FM ヘッダーと一緒に送信されます。

ATTPRN は自動的に入力または出力メッセージ間で保存されるわけではありません。出力メッセージが MFS によって処理される場合には、このパラメーターを各入力メッセージで、または出力 MFS 記述子を介して明示的に提供しなければなりません。

### 関連資料

577 ページの『ATTRDPN および ATTRPRN』

戻り宛先プロセス名 (ATTRDPN) および戻り 1 次リソース名 (ATTRPRN) パラメーターは、ソース・セッション内の戻りプロセスと戻り 1 次リソースを定義し、ソース・セッション内の応答を戻すための経路指定を可能にするために、結果の応答上でソース・セッションに戻す必要があります。

## ATTRDPN および ATTRPRN

戻り宛先プロセス名 (ATTRDPN) および戻り 1 次リソース名 (ATTRPRN) パラメーターは、ソース・セッション内の戻りプロセスと戻り 1 次リソースを定義し、ソース・セッション内の応答を戻すための経路指定を可能にするために、結果の応答上でソース・セッションに戻す必要があります。

入力で IMS に提供され、出力 MFS フォーマット設定記述子によって変更または削除されなかった場合には、これらのパラメーターは、出力 ATTACH ATTRDPN および ATTRPRN パラメーターとしてそれぞれ 応答 ATTACH FM ヘッダーで変更されない発信元セッションに戻されます。これらのパラメーターは処理されるメッセージと関連しており、セッション全体のメッセージおよびサブシステム障害で回復されます。

入力 ATTRDPN および ATTRPRN は自動的に入力または出力メッセージ間で保存されるわけではありません。MFS DPM が出力に使用される場合には、ATTRDPN および ATTRPRN は、各入力メッセージで、また

は MFS 出力フォーマット設定記述子を介して明示的に提供されなければなりません。次の MID が MOD で指定された場合には、ATTRDPN は MFS の結果として生じます。

入力メッセージを発生したセッションと異なるセッションで応答が戻される場合には、異なるプロシージャが続きます。出力 ATTRPRN が MFS によってセットされず、出力メッセージのソースが同一のセッションと関連していない場合には、IMS は入力 ATTRDPN および ATTRPRN をそれぞれ出力 ATTDPN および ATTPRN として折り返しませんが、この場合 IMS は、入力メッセージ通信またはランザクション (代替 PCB 出力の場合) を入力した端末のソース LTERM 名を、出力 ATTRPRN パラメーターとして自動的に挿入することもあります。

ATTRDPN パラメーターが出力で使用できない場合には、X'00' を含む 1 バイト・フィールドが出力 ATTACH FM ヘッダーに含まれます。

以下の表は、メッセージと一緒に送信される ATTACH ヘッダーと SCHEDULER ヘッダー内の DPN、PRN、RDPN、および RPRN フィールドに関連した IMS 処置を要約しています。

表 105. DPN、PRN、RDPN、および RPRN フィールドに対する IMS 変換処理

入力 FMH	IMS 変換処理
DPN	MFS MID 名または入力メッセージ・エディター名 (ISC 編集または基本編集)。提供されなかった場合には、デフォルトの値として ISCEDT が使用されます。
PRN	入力ランザクション・コードまたは LTERM 名、オーバーライド・データ・ストリーム、および会話中以外の事前設定モード。
RDPN	デフォルト応答 DPN フィールドとして保管されます。
RPRN	デフォルト応答 PRN フィールドとして保管されます。

表 106. DPN、PRN、RDPN、および RPRN フィールドに対する IMS の処置

出力 FMH フィールド	入力と同じセッション上に送信された IMS 処置応答メッセージ	IMS 処置応答メッセージまたは別のセッション上に送信された非同期出力
ATTDPN	折り返された入力 ATTRDPN (提供された場合)。オプションとして MFS により変更される。  ATTACH のみの出力の場合、パラメーターが使用できなければ DPN は送信されません。SCHEDULER 出力の場合、デフォルトは ISCEDT です。	MFS を使用してのみ提供される。  ATTACH のみの出力の場合、パラメーターが使用できなければ DPN は送信されません。SCHEDULER 出力の場合、デフォルトは ISCEDT です。
ATTPRN	折り返された入力 ATTRPRN。オプションとして、MFS によって変更される。	MFS を使用してのみ提供される
ATTRDPN	MFS を使用してのみ提供される	MFS を使用してのみ提供される
ATTRPRN	MFS を使用してのみ提供される	自動的にデフォルトの値としてソース LTERM 名が使用される。オプションとして、MFS によって変更される。

## ATTDQN および ATTDQ

宛先キュー名 (ATTDQN) パラメーターは、特定のメッセージ・インスタンスの名前を示します。このパラメーターは、出力 MFS 要求時ページ化メッセージでのみ有効です。

MFS 要求時ページ化出力では、ATTDQN は出力メッセージ ID として IMS によって送信されます。その後この名前は、他のハーフセッションで、ページング要求内の QNAME パラメーターとして使用し、IMS 要求時ページ化出力メッセージにアクセスすることができます。IMS、内では、あるセッションには、任意の 1 時点で、1 つだけのメッセージがアクティブになることができます。これは、1 つのメッセージで出力ページングが始まると、ページング操作が終了するまで、すべての入力は同じ QNAME (ATTDQN から) 値を示すページング要求でなければならないことを意味します。ATTACH FM ヘッダーのセッション・ロ

ーカル・フラグ (ATTDQ) は 1 に設定されます。 ATTDQ および ATTDQ はその他の状態では送信または受信されません。

## ATTACC

IMS は出力時にはアクセス・コード (ATTACC) パラメーターを送信せず、 入力時にはこのパラメーターを無視します。

## データ記述子 FM ヘッダーの形式

以下の表では、入出力のデータ記述子 FM ヘッダーの形式を示します。

表 107. 入力データ記述子 FM ヘッダーの形式

バイト	ビット	名前	内容
0	0-7	FMHL	長さ
1	0	FMHC	
	1-7	FMHT	B'0000100'
2	0-7	FMH4FXCT	固定長パラメーター (X'03')
3	0-7	FMH4DTYP	
		FMH4UNDF	X'00' 予約済み
		FMH4FIX	X'40' フィールド・フォーマット済みレコード (FFR) - 固定
		FMH4FXSP	X'41' FFR - 分離文字 (注 1) によって終了されたすべてのフィールド
3	0-7	FMH4MXSP	X'42' FFR - 分離文字またはマップにより定義された長さによって終了したフィールド X'43' - X'FF' 予約済み (注 1)
4		FMH4SEP	分離文字 (注 2)
5	0-7	FMH4PCTL	表示制御バイト (注 3)
6-m		FMH4DSN	データ構造名 (注 4)
m+1-n		FMH4BDT	ブロック・データ・タイプ (注 3)
n+1-p		FMH4VERS	バージョン ID (注 5)

注:

1. DIV または DPAGE ステートメントから。
2. これは MFS フォーマット設定記述で受信されるので、必要でない。
3. 送信された場合には、IMS によって無視される。
4. DPAGE 選択がデータ構造名で実行されることになっている場合には、DPAGE 名。OPTIONS=MSG の場合には、FMT 名。必須フィールド。
5. 正しい定義がデータをマップするのに使用されていることを検査するために MFS によって使用されるバージョン ID。入力メッセージを作成するために複数の伝送チェーンが使用された場合には、このフィールドは各入力メッセージごとに一度しか存在してはなりません。各メッセージに一度だけ存在している場合には、これをメッセージの最初の伝送チェーンで送信しなければなりません。バージョン ID が送信されないか、または X'0000' である場合には、MFS 記述子検査は実行されません。

表 108. 出力データ記述子 FM ヘッダーの形式

バイト	ビット	名前	内容
0	0-7	FMHL	長さ



表 108. 出力データ記述子 FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容
1	0	FMHC	
	1-7	FMHT	B'0000100'
2	0-7	FMH4FXCT	固定長パラメーター (X'03')
3	0-7	FMH4DTYP	
		FMH4UNDF	X'00' 予約済み
		FMH4FIX	X'40' フィールド・フォーマット済みレコード (FFR) - 固定
		FMH4FXSP	X'41' FFR - 分離文字 (注 1) によって終了されたすべてのフィールド
		FMH4MXSP	X'42' FFR - 分離文字、またはマップによって定義された長さによって終了されたフィールド X'43' - X'FF' 予約済み
4		FMH4SEP	分離文字
5	0-7	FMH4PCTL	表示制御バイト (注 1)
6-m		FMH4DSN	データ構造名 (注 2)
m+1-n		FMH4BDT	ブロック・データ・タイプ (注 3)
n+1-p		FMH4VERS	バージョン ID (注 4)

注:

- 出力で 0 に設定。
- FMH4DSN は次のとおりです。
  - OPTIONS=MSG の場合には、FMT 名
  - OPTIONS=DPAGE の場合には、DPAGE 名
  - OPTIONS=PPAGE の場合には、PPAGE 名
- バージョン ID がない場合には、省略されます。バージョン ID がある場合には、0 に設定されます。
- デフォルトが使用された場合には、FMT 定義で指定された、あるいは MFS によって計算されたバージョン ID。このフィールドは、メッセージの最初または単独のデータ記述子 FM ヘッダーにだけ存在しません。

## エラー・リカバリー・プロシージャ (ERP) FM ヘッダー

以下の表には、ERP FM ヘッダーの形式が示されています。

表 109. エラー・リカバリー・プロシージャ (ERP) FM ヘッダー

バイト	ビット	名前	内容
0		FMHL	長さ
1	0	FMHC	
	1-7	FMHTYPE	X'07'

表 109. エラー・リカバリー・プロシージャー (ERP) FM ヘッダー (続き)

バイト	ビット	名前	内容
2-5		ERPSENSE	エラー応答に現れる SNA センス・コード
6-7		ERPSEQ	エラーが検出されたチェーンのシーケンス番号

## QMODEL FM ヘッダー

各ヘッダーの始めは固定長です。この固定長域の後に固定長サブフィールドの変数番号が続き、その後に可変長サブフィールドの変数番号が続きます。

## QGET FM ヘッダーの形式

以下の表には、QGET FM ヘッダーの形式が示されています。

表 110. QGET FM ヘッダーの形式

バイト	ビット	名前	内容
0		FMHL	長さ
1	0	FMHC	
	1-7	FMHT	タイプ: B'0000110'
2-3		FMH6CMD	コマンド・コード: X'0A10'
4		FMH6MOD	修飾子
	0	FMH6LNSZ	B'0' 1 バイトの長さフィールド
	1-6		予約済み
	7	QGETLAST	B'1' コード化要求、最後に取得 <a href="#">582 ページの『1』</a>
5		FMH6FXCT	固定長パラメーターの長さ (=X'01')
6		QORG	指定されないページング <b>X'00'</b> キュー編成が指定されていない。IMS への入力で有効。 <b>X'01'</b> 順次。無効。 <b>X'02'</b> 線形。IMS により送信された QXFER 内の QORG に一致する場合、IMS への入力で有効。 <a href="#">582 ページの『2』</a> <b>X'03'</b> 階層型。IMS によってサポートされない。 <b>X'04' から X'FF'</b> 予約済み
7-m		QNAME	レコードがそこから検索される Q。IMS メッセージ ID (ATTDQN/SCDDQN) <a href="#">582 ページの『3』</a>
m+1-n		QCURSOR	目標キュー内の カーソル・ベクトル <a href="#">582 ページの『4』</a>

表 110. QGET FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容
n+1-p		QTRNSZ	返される最大レコードのサイズ (バイナリー)。送信された場合には、IMS によって無視される。このヘッダーを使用する場合は、すべての固定長パラメーターが存在しなければなりません。

注:

- QGETLAST は、メッセージの最後の論理ページと等しい値をもつ論理ページを選択します。  
OLP 要求時ページ化出力メッセージが進行中で、要求されたカーソル値が出力メッセージの範囲内にある場合にだけ、QGET FM ヘッダーは有効で、QXFR 応答が出されることとなります。
- QORG が無効の場合にはエラー・メッセージが送信され、出力メッセージが再伝送のためにメッセージ・キューに戻されます。
- QNAME (メッセージ ID) はスケジュールされた要求時ページ化出力の各ページでは必須で、同期要求時ページ化出力ではオプションです。QNAME パラメーターがある場合には、これは IMS によって送信された ATTACH FM ヘッダーの ATTDQN/SCDDQN パラメーターと一致しなければなりません。QNAME が同期要求時ページ化出力で指定されなかった場合には、IMS はこれが ATTDQN パラメーター名と同一であると想定します。QNAME パラメーターが ATTDQN/SCDDQN パラメーターと一致しない場合には、エラー・メッセージが送信され、出力メッセージが再伝送のためにメッセージ・キューに戻されます。
- カーソル・ベクトルは、論理ページ番号を表す 1 つの 2 バイト 2 進数で構成されます。IMS は階層リトリブをサポートしません。したがって、2 レベルのカーソル (後ろに表示ページ番号が続く論理ページ番号) は無効で、結果としてエラー・メッセージが出されます。QCURSOR の長さの値は 2 です。次の 2 バイトには論理ページ番号 (絶対) 要求が含まれています。

## QGETN FM ヘッダーの形式

以下の表には、QGETN FM ヘッダーの形式が示されています。

表 111. QGETN FM ヘッダーの形式

バイト	ビット	名前	内容
0		FMHL	長さ
1	0	FMHC	
	1-7	FMHT	タイプ: B'0000110'
2-3		FMH6CMD	コマンド・コード: X'0A10'
4		FMH6MOD	修飾子
	0	FMH6LNSZ	B'0' 1 バイトの長さフィールド
	1-7		予約済み
5		FMH6FXCT	固定長パラメーターの長さ (=X'01')

表 111. QGETN FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容
6		QORG	ページング要求のタイプ X'00' キュー編成が指定されていない。この形式は OLP と非 OLP 出力のどちらでも有効。 <b>X'01'</b> 順次。583 ページの『1』 <b>X'02'</b> 線形。583 ページの『1』 <b>X'03'</b> 階層型。IMS によってサポートされない。 <b>X'04' から X'FF'</b> 予約済み。
7-m1		QNAME	ページがそこから検索される Q。 メッセージ ID を指定する。 (ATTDQN/SCDDQN)583 ページの『2』
m+1-n		QTRNSZ	戻される最大レコードのサイズ (バイナリー)。送信された場合には、IMS によって無視される。

注:

1. QORG が指定された場合には、IMS によって送信された結果の QXFR FM ヘッダーで指定された QORG はこれと一致しなければなりません。そうでない場合には、エラー・メッセージが送信され、出力メッセージが再伝送のためにメッセージ・キューに戻されます。
2. QNAME (メッセージ ID) はスケジュールされた要求時ページ化出力の各ページでは必須で、同期要求時ページ化出力ではオプションです。QNAME パラメーターがある場合には、これは IMS によって送信された ATTACH FM ヘッダーの ATTDQN/SCDDQN パラメーターと一致しなければなりません。QNAME が同期要求時ページ化出力で指定されなかった場合には、IMS はこれが ATTDQN パラメーター名と同一であると想定します。QNAME パラメーターが ATTDQN/SCDDQN パラメーターと一致しない場合には、エラー・メッセージが送信され、出力メッセージが再伝送のためにメッセージ・キューに戻されません。

## QPURGE FM ヘッダーの形式

以下の表には、QPURGE FM ヘッダーの形式が示されています。

表 112. QPURGE FM ヘッダーの形式

バイト	ビット	名前	内容
0		FMHL	長さ
1	0	FMHC	
	1-7	FMHT	タイプ: B'0000110'
2-3		FMH6CMD	コマンド・コード: X'0A06'
4		FMH6MOD	修飾子
	0	FMH6LNSZ	B'0' 1 バイトの長さフィールド
	1-7		予約済み
5		FMH6FXCT	固定長パラメーターの長さ (=X'01')

表 112. QPURGE FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容
6		QORG	キュー除去機能のタイプ (複数の DPAGE 入力の場合) またはページング要求 (要求時ページ化出力の場合) <sup>584</sup> ページの『1』
7-m		QNAME	IMS メッセージ ID の名前 (ATTDQN/SCDDQN) <sup>584</sup> ページの『2』

注:

1. 要求時ページ化出力中に IMS が QPURGE を受信した場合には、QORG は指定されていなければ (つまり、QORG は 0 でない)、IMS によって送信された QFXR FM ヘッダーで指定された QORG と一致しなければなりません。
2. QNAME (メッセージ ID) はスケジュールされた要求時ページ化出力の各ページでは必須で、同期要求時ページ化出力ではオプションです。要求時ページ化出力中に IMS が QPURGE を受信した場合に、QNAME が指定されると、IMS によって送信された ATTACH FM ヘッダーの ATTDQN/SCDDQN パラメーター (メッセージ ID) と一致している必要があります。QNAME が同期要求時ページ化出力で指定されなかった場合には、IMS は ATTDQN パラメーター名と同一の名前を想定します。QNAME パラメーターが ATTDQN/SCDDQN パラメーターと一致しない場合には、エラー・メッセージが送信され、出力メッセージが再伝送のためにメッセージ・キューに戻されます。

## QSTATUS FM ヘッダーの形式

以下の表には、QSTATUS FM ヘッダーの形式が示されています。

表 113. QSTATUS FM ヘッダーの形式

バイト	ビット	名前	内容
0		FMHL	長さ
1	0	FMHC	
	1-7	FMHT	タイプ: B'0000110'
2-3		FMH6CMD	コマンド・コード: X'0A0A'
4		FMH6MOD	修飾子
	0	FMH6LNSZ	B'0' 1 バイトの長さフィールド
	1-7		予約済み
5		FMH6XCT	固定長パラメーターの長さ (=X'02')
6		QORG	キューの追加/リトリブ要求のタイプ。 <sup>585</sup> ページの『1』
7	0-4		IMS によってサポートされない。ビット値は B'0' でなければなりません。
	5	QINVCUR	B'0' メッセージが QPURGE に対する応答の場合。 B'1' 無効カーソル OLP 出力で受信された無効論理ページ 要求を指示します。
	6-7		IMS によってサポートされない。

表 113. QSTATUS FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容
8			予約済み
9-12		QCURSOR	現行カーソル値は 0585 ページの『2』
13-16		QSENSE	センス・データ 585 ページの『3』
17-n		QNAME	メッセージ ID (ATTDQN/SCDDQN)

注:

1. QSTATUS が QPURGE への応答で IMS によって送信された場合には、QORG は指定されません (つまり、QORG にゼロの値は含まれません)。QSTATUS がカーソル要求による QGET への応答で IMS によって送信された場合には、QORG は線形に設定されます (つまり、QORG に 2 の値が含まれます)。
2. このパラメーターは QINVCUR 値が B'1' に設定される場合にだけ存在します。カーソル値は 0 に設定されます。QGETN 要求は、このエラー (無効カーソル) が検出されたメッセージの最初のページをリトリブします。
3. このパラメーターは QINVCUR 値が B'1' に設定される場合にだけ存在します。これは 2 進数として DFS223 エラー・メッセージで構成されます。

## QXFR FM ヘッダーの形式

以下の表には、QXFR FM ヘッダーの形式が表示されています。

表 114. QXFR FM ヘッダーの形式

バイト	ビット	名前	内容
0		FMHL	長さ
1	0	FMHC	
	1-7	FMHT	タイプ: B'0000110'
2-3		FMH6CMD	コマンド・コード: X'0A08'
4		FMH6MOD	修飾子
	0	FMH6LNSZ	B'0' 1 バイトの長さフィールド
	1-7		予約済み
5		FMH6FXCT	固定長フィールドの長さ (=X'02')

表 114. QXFR FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容
6		QORG	このメッセージに有効なページング要求のタイプ。 <b>X'01'</b> ページの順次リトリブ。オペレーター論理ページング (OLP) は定義されません。最後のページの正常な伝送後、メッセージはキューから除去されます。 <b>X'02'</b> ページの線形リトリブ。オペレーター論理ページング (OLP) またはブラウズは、許可されています。メッセージは、他のサブシステム (RAP または QPURGE FM ヘッダーは受信される) からの明示的アクションによってのみデキューされる。 <b>X'03'</b> ページの階層リトリブ。IMS によってサポートされない。 <b>X'00', X'04'-X'FF'</b> 予約済み。
7	0-4	予約済み	
	5	QDISP	B'0' 後処理が保管される。
	6		予約済み。
	7	QEMSG	B'0' メッセージの終わりでない B'1' メッセージの終わり メッセージの最後の論理ページ (OPTIONS=DPAGE) または最後の表示ページ (OPTIONS=PPAGE) が伝送された場合、B'1' にセットされる。
8-n		QCURSOR <sup>587 ページ</sup> の『1』	現行メッセージのカーソル・ベクトル。 OPTIONS=PPAGE の場合、このフィールドには値 4 の 1 バイトのフィールドと、その後に 2 バイトの現行論理ページ番号および、2 バイトの現行表示ページ番号が含まれます。 OPTIONS=DPAGE の場合、このフィールドには値 2 の長さフィールドと、その後に現行論理ページ番号が含まれます。 <sup>587 ページの『2』</sup>
n+1-n		QCOUNT <sup>587 ページ</sup> の『1』	カーソルの最低レベルでのページのカーソルのレベル。 存在する場合、このフィールドには値 2 の長さバイトと、その後に現行論理ページに定義されたページ数が含まれます。 QCOUNT フィールドは、現行論理ページに定義された表示ページ数が 1 より大きい場合のみ存在します。



表 114. QXFR FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容
n+1-p		QRECLNG	切り捨ての前のレコードの長さ。0 の場合には、レコードが切り捨てられなかったか、QGET(N) の QTRNSZ が無視されました。このフィールドは IMS によりサポートされず、FM ヘッダーに含まれません。

注:

1. オペレーター論理ページング (OLP) が許可されている場合には、受信側がカーソル要求によって後続の QGET を公式化するのに使用できる情報が、QCURSOR および QCOUNT フィールドに含まれます。
2. 次の出力メッセージを OPTIONS=PPAGE で想定してください。

論理ページ 1 3 定義済み表示ページ  
 論理ページ 2 1 定義済み表示ページ  
 論理ページ 3 2 定義済み表示ページ

以下の表は、順次リトリブ要求で QCURSOR および QCOUNT フィールドに入れられた値を示しています。

表 115. 順次リトリブ要求を使用した QCURSOR および QCOUNT 値

伝送	カウント / LP	番号 / PP 番号	Qcursor カウント / PP 番号	Q カウント・コメント
1	4 / 1	1	2 / 3	論理ページ 1
2	4 / 1	2	2 / 3	論理ページ 1
3	4 / 1	3	2 / 3	論理ページ 1
4	4 / 2	1	(省略された)	論理ページ 2
5	4 / 3	1	2 / 2	論理ページ 3
6	4 / 3	2	2 / 2	論理ページ 3

注:

OPTIONS=DPAGE の出力メッセージは次のとおりです。

論理ページ 1  
 論理ページ 2  
 論理ページ 3

以下の表は、順次リトリブ要求で QCURSOR フィールドに入れられた値を示しています。

表 116. 順次リトリブ要求を使用した QCURSOR 値

伝送	Qcursor カウント / LP	Q カウント	コメント
1	2 / 1	なし	論理ページ 1
2	2 / 2	なし	論理ページ 2
3	2 / 3	なし	論理ページ 3

## リセット付加プロセス (RAP) FM ヘッダーの形式

以下の表には、リセット付加プロセス (RAP) FM ヘッダーの形式が示されています。

表 117. リセット付加プロセス (RAP) FM ヘッダーの形式

バイト	ビット	名前	内容
0		FMHL	
1	0	FMHC	
	1-7	FMHT	B'0000101'
2-3		FMH5CMD	X'0204'
4		FMH5MOD	
	0	FMH5LNSZ	B'0': 1 バイトのパラメーター長さフィールド
	1-7		パラメーター長さフィールドの長さ B'0': 1 バイトのパラメーター長さフィールド B'1': 予約済み
5		FMH5FXCT	固定長パラメーターの長さ (X'00')

## SCHEDULER FM ヘッダーの形式

以下の表には、SCHEDULER FM ヘッダーの形式が示されています。

表 118. SCHEDULER FM ヘッダーの形式

バイト	ビット	名前	内容
0		FMHL	
1	0	FMHC	
	1-7	FMHT	B'0000110'
2-3		FMH6CMD	X'0802'
4		FMH6MOD	
	0	FMH6LNSZ	B'0': 1 バイトのパラメーター長さフィールド
	1	FMH6RPLY	<b>B'0'</b> 無応答 <b>B'1'</b> 応答 <a href="#">589 ページの『1』</a>
	2	FMH6PROT	B'0' 保護なし <a href="#">589 ページの『2』</a> B'1' 保護
	3	FMH6DELY	<b>B'0'</b> タイマー・オプション <b>B'1'</b> タイマー必要 <a href="#">589 ページの『1』</a>
4-7			予約済み
5		FMH6FXCT	固定長パラメーターの長さ - X'01'

表 118. SCHEDULER FM ヘッダーの形式 (続き)

バイト	ビット	名前	内容
6			スケジュール要求制御
	0	SCDTIME	開始遅延仕様 <b>B'0'</b> 間隔 <b>B'1'</b> 時間 IMS によってサポートされない
	1-7		予約済み
7-m		SCDDPN <sup>589 ページの『3』</sup>	開始されるプロセスの名前。(必須フィールド)
m+1-n		SCDPRN <sup>589 ページの『3』</sup>	開始されるプロセスの 1 次リソースの名前
n+1-p		SCDRDPN <sup>589 ページの『3』</sup>	戻りプロセスの提示名
p+1-q		SCDRPRN <sup>589 ページの『3』</sup>	戻りプロセス名の 1 次リソースの提示名
q+1-r		SCDDQN <sup>589 ページの『3』</sup>	プロセスと関連したキューの名前
r+1-s		SCDREQN	このプロセス要求インスタンスの名前 (IMS によってサポートされない)
s+1-t		SCDDELY	宛先処理の開始前に 減少させる時間間隔。 (IMS によってサポートされない)

注:

1. 入出力の両方で 0 でなければなりません。
2. 入力で、このフィールドは無視されます。出力でこのフィールドは、リカバリー不能出力の場合は 0、リカバリー可能出力の場合は 1 に設定されます。
3. これらのパラメーターの詳細な説明については、[575 ページの『ATDPN』](#)、[577 ページの『ATPRN』](#)、[577 ページの『ATTRDPN および ATTRPRN』](#)、および [578 ページの『ATDQN および ATDTP』](#) を参照してください。ATTACH のトピックでは、これらのパラメーターの接頭部は、「SCD」でなく「ATT」です。これらの SCHEDULER パラメーターの操作の規則は、ATTACH パラメーターで定義された規則と同じです。

## SYMSG プロセス・ヘッダー

以下の表では、SYMSG プロセス・ヘッダーの形式である SYSERROR FM ヘッダーおよび SYSSTAT FM ヘッダーが示されています。

### SYSERROR FM ヘッダーの形式

以下の表には、SYSERROR FM ヘッダーの形式が示されています。

表 119. SYSERROR FM ヘッダーの形式

バイト	ビット	名前	内容
0		FMHL	
1	0	FMHC	
	1-7	FMHT	B'0000110'
2-3		FMH6CMD	X'0404'
4		FMH6MOD	
	0	FMH6LNSZ	B'0': 1 バイトのパラメーター長さフィールド
	1-7		予約済み
5		FMH6FXCT	固定長パラメーターの長さ (X'00')
6-m		ATDPN	この SYSMMSG が結果になる入力メッセージで提供された RDPN フィールド
m+1-n		ATTPRN	この SYSMMSG が結果になる入力メッセージで提供された RPRN フィールド

### SYSSTAT FM ヘッダー

以下の表には、SYSSTAT FM ヘッダーの形式が示されています。

表 120. SYSSTAT FM ヘッダー

バイト	ビット	名前	内容
0		FMHL	長さ
1	0	FMHC	
	1-7	FMHT	B'0000110'
2-3		FMH6CMD	X'0402'
4		FMH6MOD	
	0	FMH6LNSZ	B'0' 1 バイトの長さフィールド
	1-7		予約済み
5		FMH6FXCT	固定長パラメーターの長さ

## 第 33 章 ISC 編集 ATTACH パラメーターの使用の例

以下のトピックでは、ISC 編集 ATTACH パラメーターの使用例を示します。

### ISC 編集での ATTACH および SCHEDULER パラメーター

以下の一連の図は、IMS 入出力メッセージに ISC 編集を使用する場合の ATTACH パラメーターの使用例を示しています。

ISC 編集は、以下の 1 つが起こったときの通常のトランザクション、コマンド、および LTERM 間のメッセージ通信に使用されます。

- ブラケット間状態にあるときに、宛先プロセス名 (DPN) を指定していない ATTACH FM ヘッダーが受信された。
- DPN が、IMS システム 定義時に定義されたとおりに IMS ISC 編集に名前を付けている。
- 接続マネージャーがブラケット間リセット状態になっているとき、またはアクティブ・プロセスが ISC 編集であるときに、ATTACH FM ヘッダーが受信されなかった。
- DPN=ISCEDT

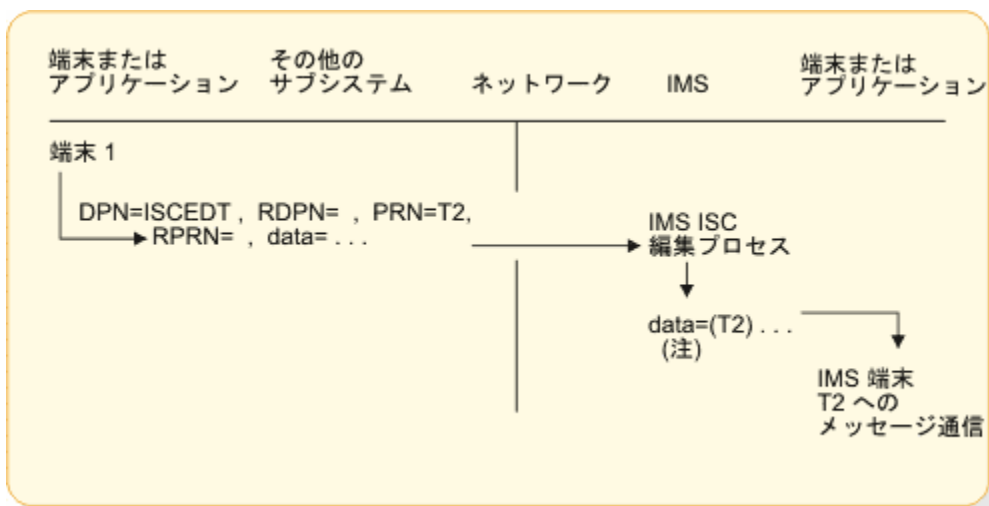


図 71. 別のサブシステム端末から IMS 端末へのメッセージ通信の例

注：端末 T2 の IMS LTERM 名は、入力基本リソース名 (PRN) として、またはデータ中の最初のフィールドとして、提供することができます。

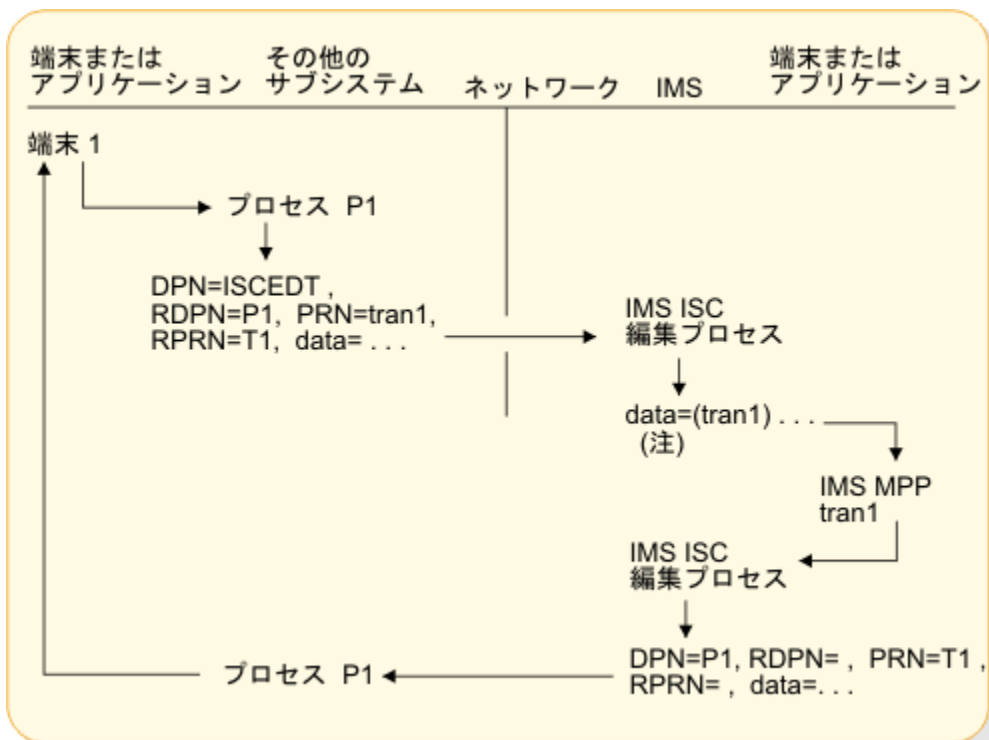


図 72. 別のサブシステムが IMS アプリケーション・プログラムにアクセスし、サブシステム端末に応答が戻される例

注: IMS トランザクション・コードは、入力 PRN として、またはデータの中に組み込んで、提供することができます。入力 RDPN および RPRN パラメーターは、MFS によって処理されないときのアプリケーション・プログラムからの結果の応答として、それぞれ DPN および PRN パラメーターになります。

以下の図では、メッセージは、メッセージ通信あるいは MPP または高速機能メッセージ経路指定アプリケーションを介して IMS 端末 1 から ISC 編集プロセスへ経路指定されます。ISC 編集プロセスは、他のサブシステム上のトランザクションへのメッセージの送信に SCHEDULER パラメーターを使用します。次に、他のサブシステムは同じ SCHEDULER パラメーターを使用してメッセージを IMS および ISC 編集プロセスへ戻します。ISC 編集プロセスは、メッセージを端末 1 へ戻します。

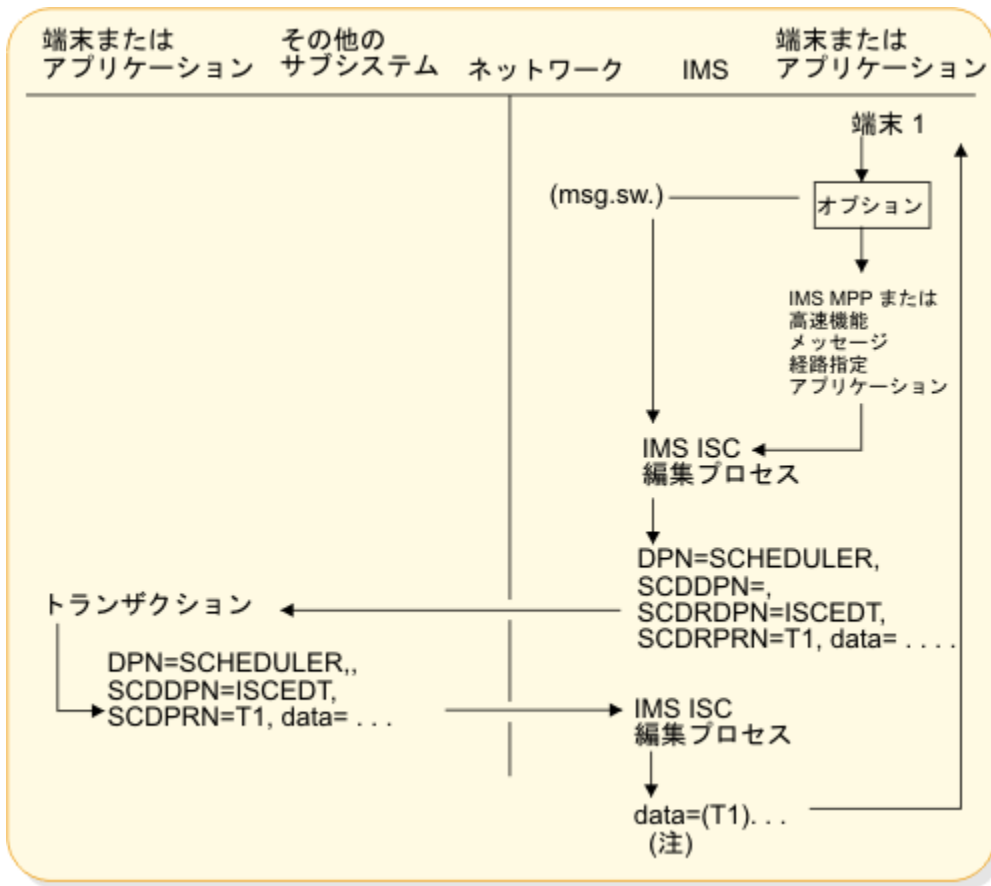


図 73. ISC 編集でのメッセージ経路指定の例

注: IMS LTERM 名は、入力 SCDDPN として、または最初のデータ・フィールドとして、提供することができます。

## IMS SYSMMSG プロセスにおける ATTACH パラメーター

システム・メッセージ・プロセス (SYSMMSG) は、ATTACH ヘッダー内の特別な SNA プロセス名によって指示されます。

PRN が入力 ATTACH パラメーターで提供されない場合には、IMS ISC セッション用の入力システム・メッセージは、ログに記録された上で直接 IMS マスター端末オペレーターに経路指定されます。IMS ISC セッション用の PRN 値が提供された場合には、この値は IMS 入力メッセージ宛先トランザクション・コードまたは LTERM 名 (メッセージ通信) となり、入力システム・メッセージが ISC 編集に渡されます。SYSMMSG は、IMS コマンド・プロセッサへのアクセスを提供しません。

提供された PRN 値が IMS トランザクション用のものであった場合には、応答は結果的に入力 SYSMMSG からなされる可能性があります。この応答は、ちょうど他の任意の (SYSMMSG 以外の) 非同期出力メッセージのようには送信されます。

IMS は、IMS ブロードキャスト・メッセージおよびその他のシステム・メッセージを送信するために SYSMMSG プロセスの接続を要求します。これらのシステム・メッセージは、入力 IMS コマンドによって直接に送信請求されることはありません。

**例外:** 入力 IMS 応答モード時または会話型トランザクション時に IMS がエラー条件を検出すると、その入力トランザクションに対する例外応答 (選択受信側 ERP) が発生し、システム・メッセージは ERP メッセージとして送信されます。

以下の図は、IMS 入出力メッセージ用の SYSMMSG プロセスを接続する際の ATTACH パラメーターの使用について説明しています。



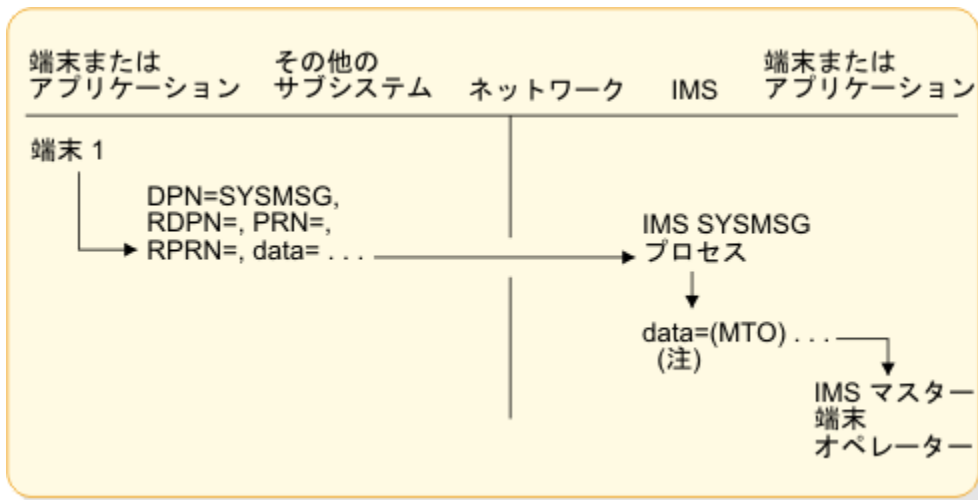


図 74. 他のサブシステム端末から IMS A への PRN のない SYSMSG の例

注：入力 ATTACH PRN パラメーターが提供されないときには、マスター端末オペレーターの IMS LTERM 名と見なされます。

以下の図では、端末 1 からの入力メッセージは、ATTACH パラメーターを使用して IMS 以外のサブシステムから IMS 内の IMS SYSMSG プロセスへ経路指定されます。IMS SYSMSG プロセスはメッセージを IMS 編集プロセスに渡し、IMS 編集プロセスはデータを IMS MPP 領域内の tran1 へ送信します。tran1 の処理後に、応答が再度 IMS 編集プロセスに渡されます。IMS 編集プロセスは、SCHEDULER パラメーターを使用して、応答を他のサブシステムのプロセス 1、そしてその後端末 1 へと戻します。

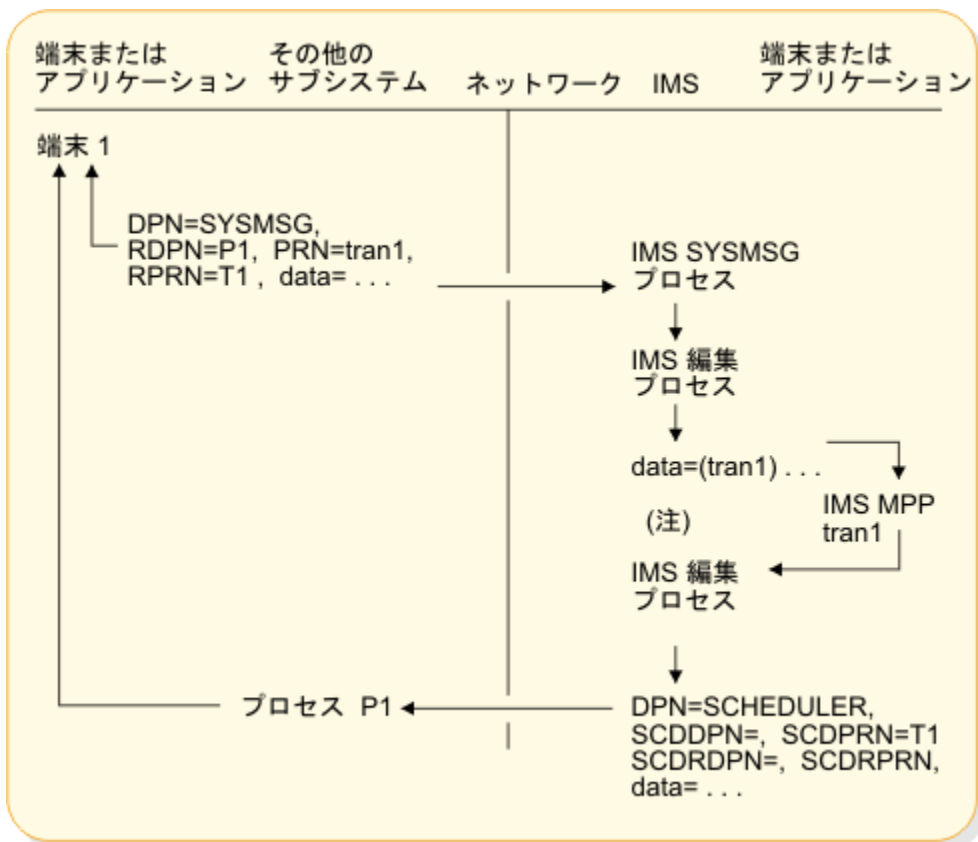


図 75. 別のサブシステム端末から IMS への PRN 付きの SYSMSG の例

注: PRN パラメーターは IMS 宛先になります。入力 RDPN パラメーターおよび入力 RPRN パラメーターは、MPP からの結果の応答として、それぞれ出力 DPN パラメーターおよび 出力 PRN パラメーターになります。

## IMS MFS における ATTACH および SCHEDULER パラメーター

以下の一連の図は、IMS 入出力メッセージに IMS MFS を使用する場合の ATTACH および SCHEDULER パラメーターの使用例を示しています。

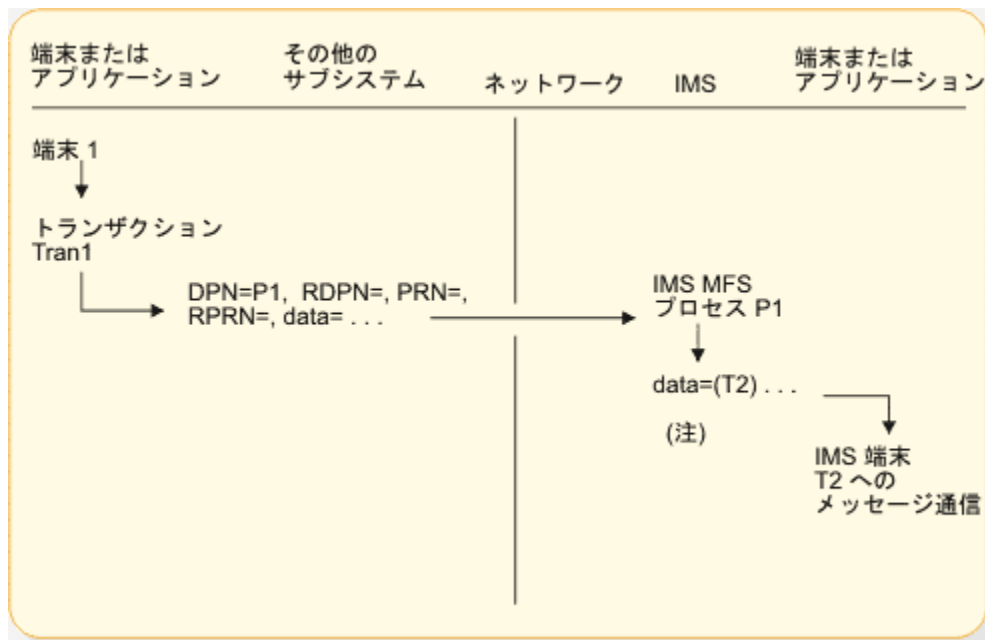


図 76. 他のサブシステム端末から IMS 端末へのメッセージ通信の例

注: 端末 T2 の IMS LTERM 名は、入力 1 次リソース名 (PRN)、データ・フィールド、または MFS 定義のリテラルとして提供することができます。提供された場合には、RPRN および RDPN は、MFS によって IMS 端末に表示されるデータにオプションで組み込むことができます。MFS を呼び出すためには、DPN パラメーターを入力 ATTACH FM ヘッダーに提供しなければなりません。

以下の図では、トランザクション・メッセージが端末 1 から発信され、IMS 以外のサブシステム上のプロセス P1 へ送信されます。次に ATTACH パラメーターを使用して、メッセージが IMS サブシステム上の MFS プロセス P2 へ送信されます。MFS プロセス P2 は、メッセージを MPP 領域内の Tran2 へ送信します。MPP 領域よりの応答は、MFS プロセスに戻されます。MFS プロセスは、ATTACH パラメーターを使用して応答を他のサブシステムへ送信します。他のサブシステムは、端末 1 をトランザクション Tran1 のリソースとして割り振り、端末へメッセージが戻されます。

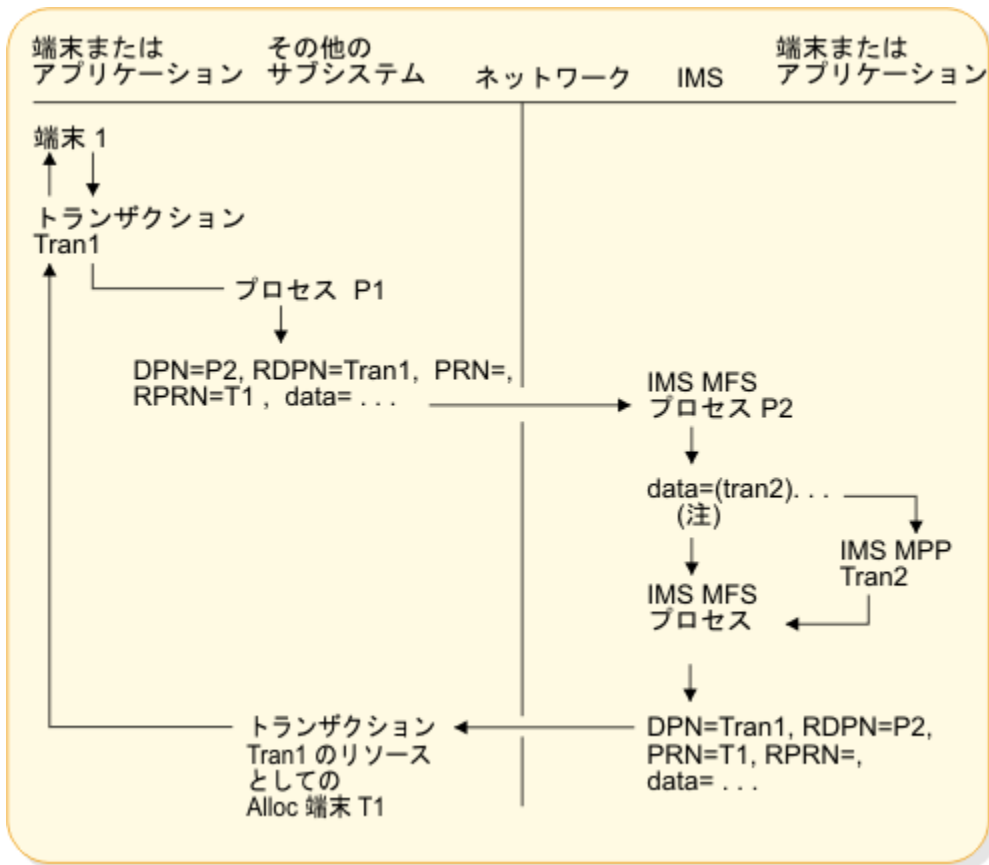


図 77. 他のサブシステムの端末が IMS アプリケーション・プログラムにアクセスし、応答が戻される例

注：IMS トランザクション・コードは、入力 PRN、データ・フィールド (基本編集の場合は最初のフィールド)、または MFS 定義のリテラルとして提供することができます。IMS MFS DPM を呼び出すために、DPN パラメータを提供しなければなりません。MPP によって挿入された応答を MFS が処理する場合は、DPM プロセスによって出力 ATTACH パラメータのいずれかが作成またはオーバーライドされることがあります。入力 RDPN パラメータおよび入力 RPRN パラメータは、DPM プロセスによってオーバーライドされない場合には、アプリケーション・プログラムからの結果の応答として、それぞれ出力 DPN パラメータおよび出力 PRN パラメータになります。

以下の図では、トランザクション・メッセージが端末 1 から発信され、IMS 以外のサブシステム上のプロセス P1 へ送信されます。次に ATTACH パラメータを使用して、メッセージが IMS サブシステム上の MFS DPM プロセス P2 へ送信されます。MFS DPM プロセス P2 は、メッセージ・データを MPP 領域内の Tran2 へ送信します。MPP 領域よりの応答は、MFS DPM プロセスに戻されます。MFS DPM プロセスは、ATTACH パラメータを使用して、IMS 以外のサブシステムに Tran3 として応答を送信します。Tran3 は保管され、後に端末 1 によって取り出されます。

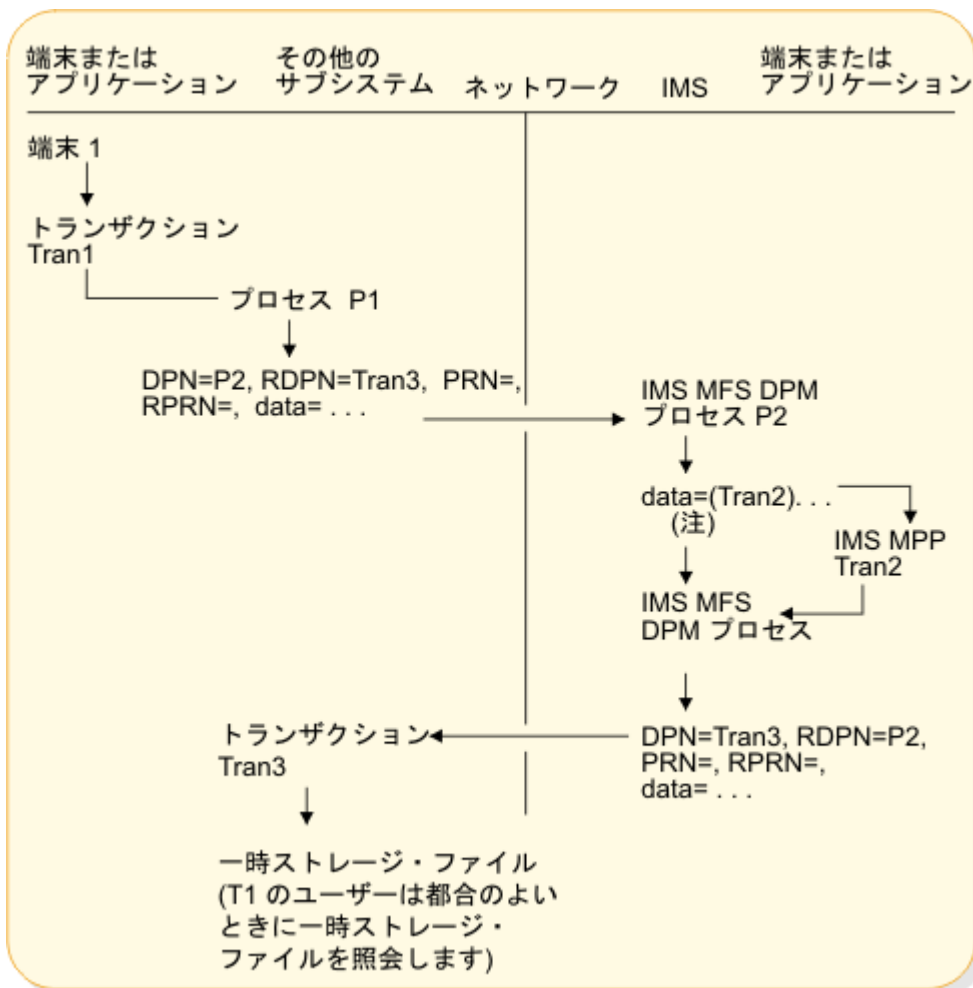


図 78. IMS MPP が他のサブシステム端末からアクセスされ、他のサブシステムの一時記憶域ファイルへの応答が返される例

注: IMS トランザクション・コードは、入力 PRN、データ・フィールド、または MFS 定義のリテラルとして提供することができます。IMS MFS DPM を呼び出すために、DPN パラメーターを提供しなければなりません。MPP が応答を挿入した場合には、DPM プロセスによって出力 ATTACH パラメーターのいずれかがオーバーライドまたは作成されることがあります。入力 RDPN パラメーターおよび入力 RPRN パラメーターは、DPM プロセスによってオーバーライドされない場合には、アプリケーション・プログラムからの結果の応答として、それぞれ出力 DPN パラメーターおよび出力 PRN パラメーターになります。

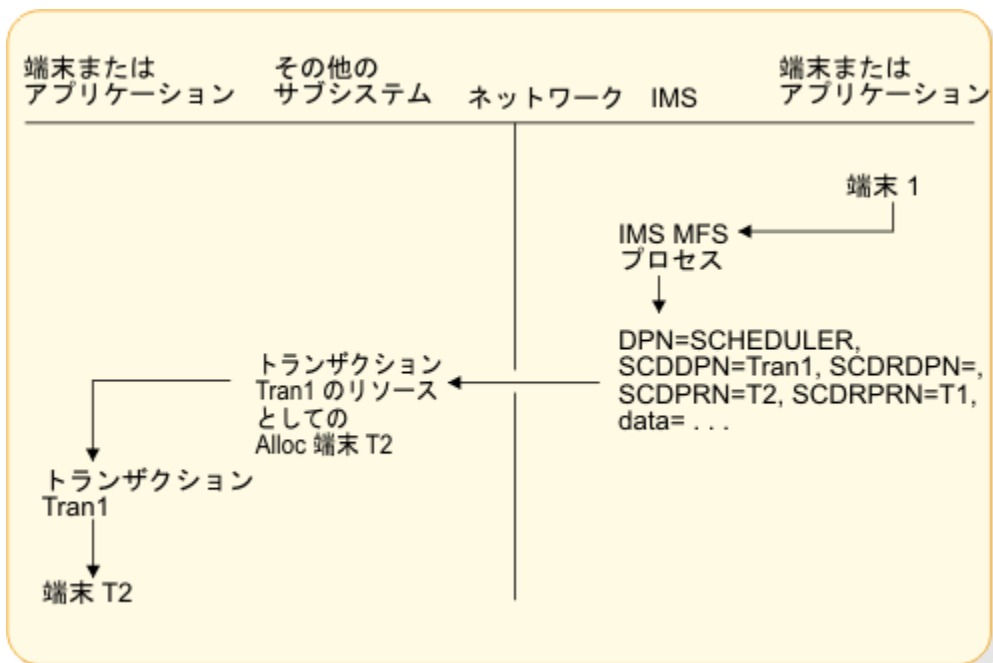


図 79. IMS 端末から別のサブシステム上の端末へのメッセージ通信の例

MFS DPM は、必須出力 SCHEDULER パラメーターを作成するために必要です。端末またはアプリケーション・プログラムから、データとして SCDDPN および SCDPRN を入力することができます。応答が別の IMS の端末またはアプリケーション・プログラムへ戻される場合には、IMS 出力メッセージの SCHEDULER FM ヘッダー内のデフォルト SCDPRN パラメーターをオーバーライドして、推奨される IMS 端末またはアプリケーションをその応答の受信先として指定することができます。当該の応答は、その SCDPRN フィールドにこの値を入れられて、戻されます。

以下の図では、メッセージは、メッセージ通信を介してあるいは MPP または高速機能メッセージ経路指定アプリケーションを介して IMS 端末 1 から IMS MFS プロセスへ経路指定されます。MFS プロセス P2 は、SCHEDULER パラメーターを使用して、メッセージを他のサブシステム上のトランザクションへ送信します。次に、他のサブシステムは、同じ SCHEDULER パラメーターを使用して応答メッセージを IMS および MFS プロセス P2 へ戻します。MFS 編集プロセスは、メッセージを端末 1 へ戻します。

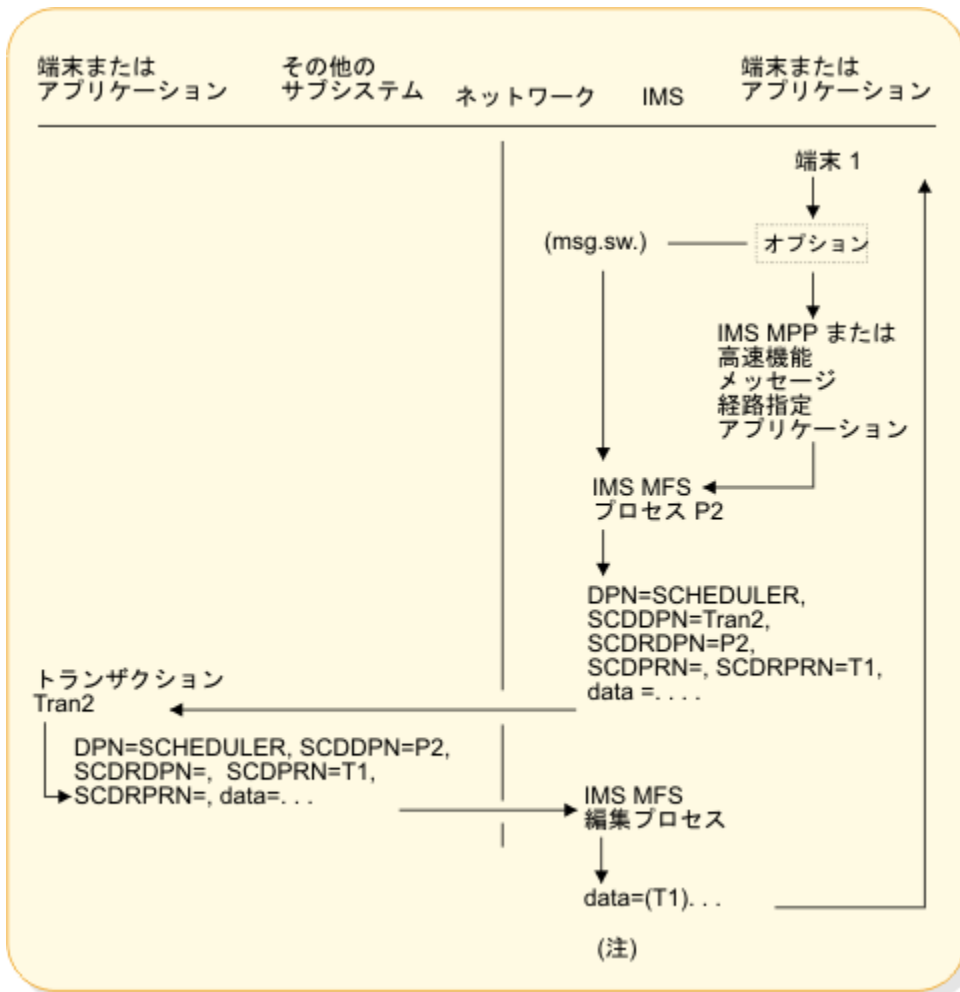


図 80. IMS MFS でのメッセージ経路指定の例

注：IMS LTERM 名は、入力 SCDPRN、最初のデータ・フィールド、または MFS 定義のリテラルとして、提供することができます。IMS MFS を呼び出すために、SCDDPN パラメーターを提供しなければなりません。MFS は、任意の出力 SCHEDULER パラメーターをオーバーライドまたは作成することができます。





# 第 34 章 IMS および CICS が ISC インターフェースを使用する方法

IMS と CICS の両方のノードを含む ISC ネットワークを設計およびインプリメントするときには、関連する問題に留意する必要があります。

VTAM によってサポートされる ISC 接続の場合、VTAM 環境で稼働する IMS と CICS 両方の基本的なシステム設計概念についてもよく理解しておく必要があります。

TCP/IP によってサポートされる ISC 接続の場合、TCP/IP サポートは IMS Connect によって提供され、VTAM プロトコルのほとんどは適用されません。IMS と IMS Connect の間の通信は、IMS Common Service Layer (CSL) の Structured Call Interface (構造化呼び出しインターフェース) (SCI) によって使用可能にされます。

ここに記載されている CICS 情報は、計画目的のためであり、CICS 資料で、または CICS システム・プログラマーに確認してください。

**関連資料:** LU 6.1 ISC を通して IMS と通信する CICS アプリケーションの設計とコーディングについて詳しくは、「*CICS Transaction Server for z/OS CICS Intercommunication Guide*」を参照してください。

## ISC セッションで使用可能な機能

IMS と CICS の間の ISC セッションで使用可能な機能は、接続が TCP/IP によって提供されているのか、それとも VTAM によって提供されているのかと、さらに、処理が同期か非同期かによって異なります。

### 関連概念

471 ページの『ISC と IMS の実行モードの関係』

「同期」と「非同期」という用語は、IMS と ISC では少し意味が異なります。以下のトピックでは、これらの実行モードの関係について説明します。

## ISC の CICS 同期処理および非同期処理の概要

CICS は、コマンド・レベル (EXEC) アプリケーション・プログラミング・インターフェース (API) を使用して ISC を実装します。CICS API では、同期も、非同期も可能です。

同期 API は、SEND EXEC コマンドと RECEIVE EXEC コマンドを使用します。ISC セッションの同期モードでは、要求が入力された瞬間から応答が戻される瞬間までセッションが保留される。そのため、入力要求と出力応答とを直接的に相関させることができます。

ISC TCP/IP 接続では、SEND EXEC コマンドと RECEIVE EXEC コマンドはサポートされていません。

非同期 API は、START EXEC コマンドと RETRIEVE EXEC コマンドを使用します。ISC セッションの非同期モードでは、入力要求と出力応答とを相関させることは一切できない。出力応答のタイミングについても、そのセッションに対して他の出力が使用できるかどうかについても、前提事項とすることができるものは何もありません。

CICS は、START EXEC コマンドと RETRIEVE EXEC コマンドでのみ ISC TCP/IP セッションをサポートします。

CICS アプリケーション・プログラムのコンテキスト内で使用された場合、EXEC コマンドは、同期処理または非同期処理のためにセッション上のメッセージと関連したデータ・フロー制御プロトコルを作成します。

CICS の実施は IMS とはまったく異なっています。IMS では、メッセージと一緒に送信されるデータ・フロー制御プロトコルは、システムによって作成され、以下の要因に基づいています。

- メッセージ用の (リカバリー可能性およびセグメンテーションなどの) 属性は、IMS システム定義時にマクロ・ステートメント TRANSACT、TERMINAL、および SUBPOOL で定義されたものか、それとも ETO ログオン記述子で定義されたものか
- VTAM 接続の場合、特性がバインド・パラメーターに定義されているかどうか

- TCP/IP 接続の場合、特性が CICS IPCONN リソース定義と TCPIP SERVICE リソース定義に定義されているかどうか
- メッセージは、関連した要求と同じセッション上に戻された応答であるか、あるいは関連した要求と異なるセッション上に戻された応答であるか
- メッセージは非送信請求非同期出力であるかどうか

非同期メッセージが CICS から IMS へ送信されたときには、送信トランザクションが終了することがあります。戻された応答は、新規に開始された CICS トランザクションによって処理される可能性があります。このプロセスは、IMS がメッセージに付加して戻す機能管理ヘッダー内のパラメーターに従って行われます。

以下の図は、SEND と RECEIVE の同期処理の概念を示しています。

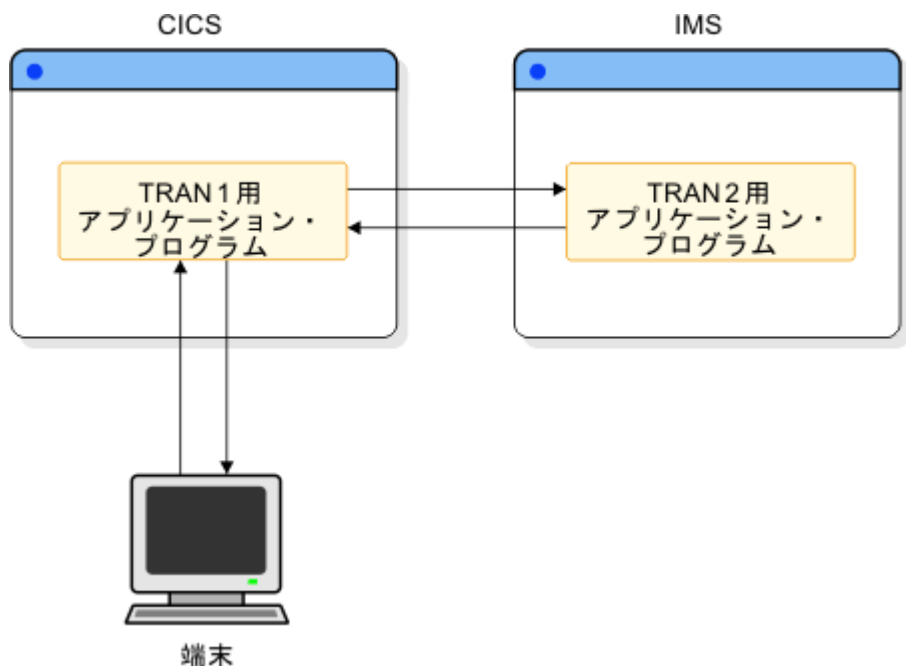


図 81. SEND および RECEIVE での同期処理

以下の図は、START と RETRIEVE の非同期処理の概念を示しています。

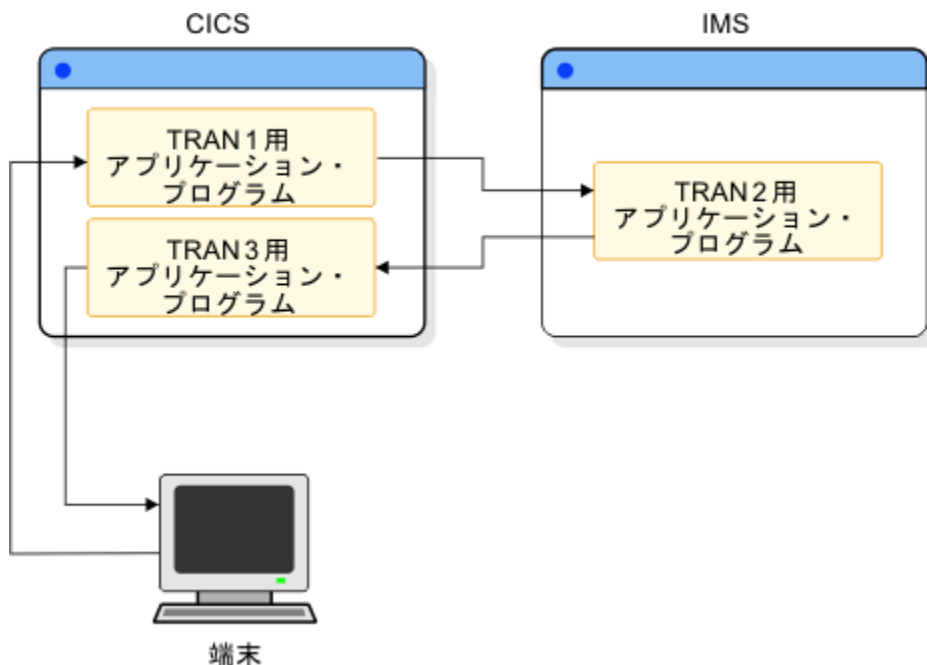


図 82. START および RETRIEVE での非同期処理

#### 関連概念

CICS: [CICS-IMS 間アプリケーション](#)

## ISC TCP/IP セッションで使用可能な機能

CICS は、プライベート IP 相互接続 (IPIC) プロトコルおよび非同期コマンド・レベル (EXEC) アプリケーション・プログラミング・インターフェース (API) を使用する ISC TCP/IP セッションをサポートします。

CICS により ISC TCP/IP セッションに対して提供される非同期 API は、**START EXEC** コマンドと **RETRIEVE EXEC** コマンドを使用します。CICS アプリケーション・プログラムのコンテキスト内で使用された場合、**START EXEC** コマンドと **RETRIEVE EXEC** コマンドは、セッション上のメッセージに関連付けられた非同期処理のための IPIC データ・フロー制御プロトコルを作成します。

**START** コマンドを実行すると、メッセージと共に IPIC ヘッダーが発行されます。**START** コマンドによって IMS に送信されるメッセージは 1 チェーンのみ長さの可能性があります、BB/EB と共に送信されます。CICS は **RETRIEVE** コマンドを使用して、CICS トランザクション処理のためにセッションから非同期メッセージを入手します。非同期メッセージで発行される他の IPIC データ・フロー制御プロトコルは、発行されたメッセージのタイプによって異なります。

ISC TCP/IP セッションでの CICS 非同期 **START EXEC** コマンドおよび **RETRIEVE EXEC** コマンドの使用では、CICS が同期フローを必要とする機能の使用は除外されます。例えば、**START EXEC** コマンドと **RETRIEVE EXEC** コマンドでは、単一チェーンのメッセージのみがサポートされます。さらに、ISC VTAM セッションによってサポートされる特定の機能は、ISC TCP/IP セッションによってサポートされません。

以下の表は、CICS での ISC TCP/IP セッションで使用可能な機能を示しています。

この表では、リストされている組み合わせでのリカバリーについては一切考慮していません。「はい」だけが、アプローチが可能であることを示します。しかし、システム設計者は、容易なりカバリーおよび再始動の必要性について考慮する必要があります。

表 121. CICS での ISC TCP/IP セッションで使用可能な機能: CICS フロントエンド

関数	ISC TCP/IP サポート
非応答または非会話モード IMS 入力トランザクション	あり
非応答または非会話モード IMS 出力トランザクション	あり

表 121. CICS での ISC TCP/IP セッションで使用可能な機能: CICS フロントエンド (続き)

関数	ISC TCP/IP サポート
応答モード IMS トランザクション (高速機能を含む)	なし
会話型モード非継続 IMS 入力トランザクション	なし
会話型モード継続 IMS 入力トランザクション	なし
回復可能 <a href="#">604 ページの『1』</a> IMS トランザクション	N/A
回復不能 <a href="#">604 ページの『1』</a> IMS トランザクション	N/A
IMS メッセージ通信	あり
フロントエンド切り替え (FES) 出口ルーチン (DFSFEBJO)	なし
IMS オペレーター・コマンド	<b>/DISPLAY</b> および <b>/RDISPLAY</b> のみ
メッセージ形式サービス (MFS) <a href="#">604 ページの『2』</a>	なし
単一セグメント入出力 IMS トランザクション	あり
複数セグメント入出力 IMS トランザクション	なし

表 122. ISC TCP/IP セッションで使用可能な機能 (CICS EXEC コマンドの場合): IMS フロントエンド

関数	ISC TCP/IP サポート
CICS トランザクション	あり
CICS オペレーター・コマンド (システム・トランザクション)	なし
MFS <a href="#">604 ページの『2』</a>	なし

#### CICS EXEC 表の注:

1. トランザクションは、IMS システム定義時にリカバリー可能またはリカバリー不能と定義され、同期点プロトコルおよびトランザクション定義が一貫しているときだけ IMS に受け入れ可能です。詳しくは、[471 ページの『ISC と IMS の実行モードの関係』](#) および [520 ページの『ハーフセッションの同期の維持』](#) を参照してください。
2. MFS は、セッションと IMS アプリケーション・プログラム間でデータをマップします。CICS では、基本マッピング・サポート (BMS) が、CICS アプリケーション・プログラムの要求によりデータをマップします。BMS はセッションと対話しませんが、IMS とやりとりするデータのマッピングを処理するために使用することができます。IMS の MFS と CICS の BMS の間では通信は行われません。

#### 関連タスク

[462 ページの『TCP/IP の ISC サポート』](#)

IMS サブシステムと CICS サブシステムの間での ISC 通信をサポートするために、TCP/IP を使用することができます。

## ISC VTAM セッションで使用可能な機能

VTAM 接続上の CICS を使用する ISC セッションは、同期および非同期の CICS API の両方をサポートします。

このトピックでは、ISC VTAM セッションで CICS アプリケーションが使用できる同期および非同期のコマンド・シーケンスについて説明します。

IMS フロントエンド切り替え出口ルーチンが提供するフロントエンド/バックエンドのシステム利用に対するサポートは、非同期モードで ISC を使用します。

CICS **SEND** および **RECEIVE** コマンドは、**SEND** が INVITE オプションで使用されたときに同期処理に関連した事前定義の SNA プロトコルを作成します。**SEND INVITE** によって、SNA ATTACH 機能管理ヘッダーお

よび方向変換が、メッセージで発行されます。メッセージは、プロセスのために CICS からリモート IMS サブシステムへ送信されます。これは、メッセージ応答が戻されたときに、最初のメッセージが発行されたトランザクションと同一の CICS トランザクション内で発行される **RECEIVE** コマンドによって処理されます。

**例外:** 再始動されたトランザクションは処理が異なります。

**SEND** コマンドに **LAST** オプションを指定して使用すると、当該セッションで **ATTACH** 機能管理ヘッダーを伴うメッセージが送信され、ブラケット開始状況とブラケット終了状況の両方 (**BB/EB**) がこの **SEND** コマンドと共に送信されます。IMS は、この特別な場合の同期処理を、IMS 非同期トランザクションの場合と同じ方法で実行します。**SEND LAST** で生成された入力に対して IMS が生成して戻す出力応答は、結果的に、新規に開始された CICS トランザクションで処理される可能性があります。この IMS 出力応答も **ATTACH BB/EB** を付加されて送信されます。CICS は **RECEIVE** を使用して、その応答を処理します。

**START** および **RETRIEVE** コマンドは、非同期処理と関連した SNA プロトコルを作成します。**START** によって、SNA **ATTACH** 機能管理ヘッダーおよび連結した **SCHEDULER** 機能管理ヘッダーがメッセージで発行されます。**START** コマンドによって IMS に送信されるメッセージは 1 チェーンのみ長さの可能性があります、**BB/EB** と共に送信されます。CICS は **RETRIEVE** コマンドを使用して、CICS トランザクション処理のためにセッションから非同期メッセージを入手します。非同期メッセージで発行される他の SNA データ・フロー制御プロトコルは、発行されたメッセージのタイプによって異なります。

以下の一連の表には、IMS と CICS の間の ISC VTAM セッションで使用可能な機能の要約が示されています。当該のセッションでどの機能が使用可能であるかを判別するためには、セッションにおけるフローがどの場合に同期となり、どの場合に非同期となるのかを知る必要があります。詳しい説明については、表の後にある注を参照してください。

この表では、リストされている組み合わせでのリカバリーについては一切考慮していません。「はい」だけが、アプローチが可能であることを示します。しかし、システム設計者は、容易なりカバリーおよび再始動の必要性について考慮する必要があります。

表 123. ISC VTAM セッションで使用可能な機能 (CICS EXEC コマンドの場合): CICS フロントエンド

使用可能な機能	SEND(INVITE)/ RECEIVE (同期)	SEND(LAST) (同期)	START/ RETRIEVE <sup>606 ページ の『1』</sup> (非同期)
非応答または非会話モード IMS 入力トランザクション	なし	あり	あり
非応答または非会話モード IMS 出力トランザクション	はい <sup>606 ページの『2』</sup>	N/A	あり
応答モード IMS トランザクション (高速機能を含む)	はい <sup>606 ページの『3』</sup>	なし	なし
会話型モード非継続 IMS 入力トランザクション	はい	いいえ <sup>606 ページの『4』</sup>	なし
会話型モード継続 IMS 入力トランザクション	なし	いいえ <sup>606 ページの『4』</sup>	あり
IMS メッセージ通信	なし	あり	あり
IMS オペレーター・コマンド	はい <sup>606 ページの『5』</sup>	はい <sup>606 ページの『6』</sup>	はい <sup>606 ページの『6』</sup>
リカバリー可能 <sup>607 ページの『7』</sup> IMS トランザクション	N/A	N/A	N/A
リカバリー不能 <sup>607 ページの『7』</sup> IMS トランザクション	N/A	N/A	N/A

表 123. ISC VTAM セッションで使用可能な機能 (CICS EXEC コマンドの場合): CICS フロントエンド (続き)

使用可能な機能	SEND(INVITE)/ RECEIVE (同期)	SEND(LAST) (同期)	START/ RETRIEVE <sup>606 ページ</sup> の『1』 (非同期)
単一セグメント入出力 IMS トランザクション	あり	あり	あり
複数セグメント入出力 IMS トランザクション	あり	あり	なし

表 124. ISC VTAM セッションで使用可能な機能 (CICS EXEC コマンドの場合): IMS フロントエンド

使用可能な機能	SEND(INVITE)/ RECEIVE (同期)	SEND(LAST) (同期)	START/ RETRIEVE <sup>606 ページ</sup> の『1』 (非同期)
CICS トランザクション	なし	なし	あり
CICS オペレーター・コマンド (システム・トランザクション)	なし	なし	なし

表 125. ISC VTAM セッションで使用可能な機能 (CICS EXEC コマンドの場合): MFS および BMS サポート

使用可能な機能 <sup>607 ページの『8』</sup>	SEND(INVITE)/ RECEIVE (同期)	SEND(LAST) (同期)	START/ RETRIEVE <sup>606 ページ</sup> の『1』 (非同期)
IMS MFS なし	あり	あり	あり
ページングなし MFS	あり	あり	あり
MFS 自動ページ化入力	はい <sup>607 ページの『9』</sup>	はい <sup>607 ページの『9』</sup>	なし
MFS 自動ページ化出力	はい <sup>607 ページの『10』</sup> , <sup>606 ページの『2』</sup>	N/A	なし
MFS 要求時ページ化出力	あり	N/A	いいえ <sup>607 ページの『11』</sup>
BMS サポート <sup>607 ページの『8』</sup>	N/A	N/A	N/A

**CICS EXEC 表の注:**

1. 単一チェーン・メッセージのみ
2. SEND LAST で送信されたトランザクションに対する 非応答モード応答は、ATTACH BB/EB を送信されます。CICS は (SEND を使用せずに) RECEIVE を使用して、セッションからメッセージを入手します。(623 ページの『SEND LAST EXEC コマンドを使用して CICS から IMS へ』を参照してください。)
3. トランザクションが外部的に (機能管理ヘッダーを使用して) 同期として指定されている場合には、IMS はシステム定義に関係なく応答モードを強制します。471 ページの『ISC と IMS の実行モードの関係』を参照してください。
4. IMS 会話型モードでは、IMS が EB を伴う最後の応答を送信することによって会話を終了する必要があります。これに関する例外については 642 ページの『CICS 対 IMS 会話モード』で説明しています。
5. IMS /DISPLAY、/FORMAT、および /RDISPLAY コマンドから出される応答は、IMS が入力コマンドに対して同期応答を行った後で非同期に送信されます。
6. /DISPLAY、/RDISPLAY、および /FORMAT のためにだけサポートされています。



7. トランザクションは、IMS システム 定義時にリカバリー可能またはリカバリー不能と定義され、同期点プロトコルおよびトランザクション定義が一貫しているときだけ IMS に受け入れ可能です。詳しくは、471 ページの『ISC と IMS の実行モードの関係』および 520 ページの『ハーフセッションの同期の維持』を参照してください。
8. MFS は、セッションと IMS アプリケーション・プログラム間でデータをマップします。BMS は、CICS アプリケーション・プログラムの要求でデータをマップします。BMS はセッションと対話しませんが、IMS とやりとりするデータのマッピングを処理するために使用することができます。IMS の MFS と CICS の BMS との間では通信は行われません。
9. 単一メッセージ・チェーンと複数メッセージ・チェーンの両方を使用できますが、SEND 1 つにつき使用可能なチェーンは 1 つだけです。
10. IMS からの同期自動ページ化出力のプロセス時にセッションがタイアップするのを避けるためには、処理を開始する前に CICS が IMS 出力のすべてのページを読み取ることが必要です。
11. IMS と CICS との間で要求時ページングを行うことは可能ですが、これには複雑な CICS アプリケーション・コーディングと複雑な端末ユーザー・インターフェースが必要であるため、推奨しません。それよりも、自動ページ化出力を作成して IMS から CICS へ送信し、ローカル・システムで CICS ページ・リトリブを使用してページングを行う方法がよいでしょう。

### 関連概念

520 ページの『ハーフセッションの同期の維持』

同期点応答 (DR2) は ISC セッション・パートナー間で使用され、両方のパートナーの同期点マネージャーが、リカバリー可能リソースを同期させてコミットまたはバックアウトできることを確認するために使用されます。

623 ページの『SEND LAST EXEC コマンドを使用して CICS から IMS へ』

CICS がフロントエンド・サブシステムで IMS がバックエンド・サブシステムであるシステムでは、IMS は IMS 非同期トランザクションへのアクセスをできるようにするために CICS 同期 API の使用をサポートします。SEND LAST は、IMS 非応答モードと非会話型トランザクション、メッセージ通信、および / DISPLAY、/RDISPLAY および /FORMAT などのコマンドを処理するために使用されます。

### 関連資料

フロントエンド切り替え出口ルーチン (DFSFEJ0) (出口ルーチン)

## TCP/IP を介する CICS との ISC 通信

IMS サブシステムと IBM CICS Transaction Server for z/OS サブシステムの間の ISC 接続をサポートするために、TCP/IP を使用できます。

### このタスクについて

ISC は、IMS と CICS 間の接続についてのみ TCP/IP をサポートします。ISC TCP/IP サポートは、CICS によって定義されている専用プロトコルである IP 相互接続 (IPIC) を使用します。IPIC の使用は、通常、ISC VTAM に対して SNA によって定義されているプロトコルと整合しており、ISC を使用するアプリケーション・プログラムからは認識されません。

IMS Connect は、ISC 用の TCP/IP 接続サポートを提供します。IMS Common Service Layer (CSL) の Structured Call Interface (構造化呼び出しインターフェース) (SCI) は、IMS と IMS Connect の間の通信パスを提供します。

動的に定義された ISC 端末と静的に定義された ISC 端末はどちらも TCP/IP を使用できます。ただし、TCP/IP をサポートするために新規に ISC 端末を定義する場合は、動的端末を使用してください。動的端末は、システム定義も IMS のコールド・スタートも必要とせず、静的 ISC 端末に比べてはるかに柔軟性が大きくなります。動的 ISC 端末は、IMS.PROCLIB データ・セットの DFSDSCMx メンバーまたは DFSDSCTy メンバーにログオン記述子をコーディングすることで定義します。

### 関連タスク

145 ページの『IMS Connect および TCP/IP 通信』



IMS の IMS Connect 機能は、TCP/IP 対応の環境から IMS DB および IMS TM の両方へのアクセスを提供します。

## ISC TCP/IP サポートの概要

ISC TCP/IP 接続は、IMS サブシステムと CICS サブシステム間のセッション制御およびデータ・フロー制御に、CICS IP 相互接続プロトコルを使用します。IMS Connect が TCP/IP サポートを提供します。

ISC TCP/IP は、SNA VTAM、または VTAM で使用されるプロトコル、コマンド、ヘッダーなどを使用しません。ただし、一貫性を保つために、一部の VTAM 用語は、意味が同じ場合、ISC TCP/IP コンテキストで使用されます。

その他の点では、操作、アプリケーション・プログラミング、およびハイレベル構成のパーспекティブから、ISC TCP/IP 通信と ISC VTAM 通信は一般に同じです。

同じ IMS コマンドを使用して、IMS から ISC セッションを制御します。IMS アプリケーション・プログラムは、ISC TCP/IP または ISC VTAM の使用に敏感である必要はありません。静的に定義された ISC LU 6.1 端末は、同じ IMS ステージ 1 システム定義マクロを使用して定義され、動的に定義された端末は、同様の ETO ログオン記述子を使用します。

静的に定義された ISC LU 6.1 端末と動的に定義された ISC ノードは、TCP/IP を使用して CICS サブシステムに接続することができます。

動的に定義された端末を使用する場合は、拡張端末オプション (ETO) が必要です。

以下の図は、IMS と CICS の間の ISC TCP/IP 通信をサポートするメイン・コンポーネント、つまり IMS と IMS Connect との間の通信のための Structured Call Interface (構造化呼び出しインターフェース) (SCI)、および TCP/IP での CICS との通信のための IMS Connect を示しています。

この図は、これらのコンポーネントを、ISC VTAM 通信をサポートするメイン・コンポーネントと比較します。これらの 2 つの通信タイプの間の主な違いは、ISC TCP/IP の場合、IMS と CICS の間のパスは SCI、IMS Connect、および TCP/IP によって提供される点です。ISC VTAM の場合、IMS と CICS の間のパスは、VTAM のみによって提供されます。

両方の通信タイプで、IMS ユーザーまたは CICS ユーザーから行われる入力がサポートされます。いずれの場合も、入力はパートナー・サブシステムに送信され、そこでトランザクションが処理され、応答が返されます。

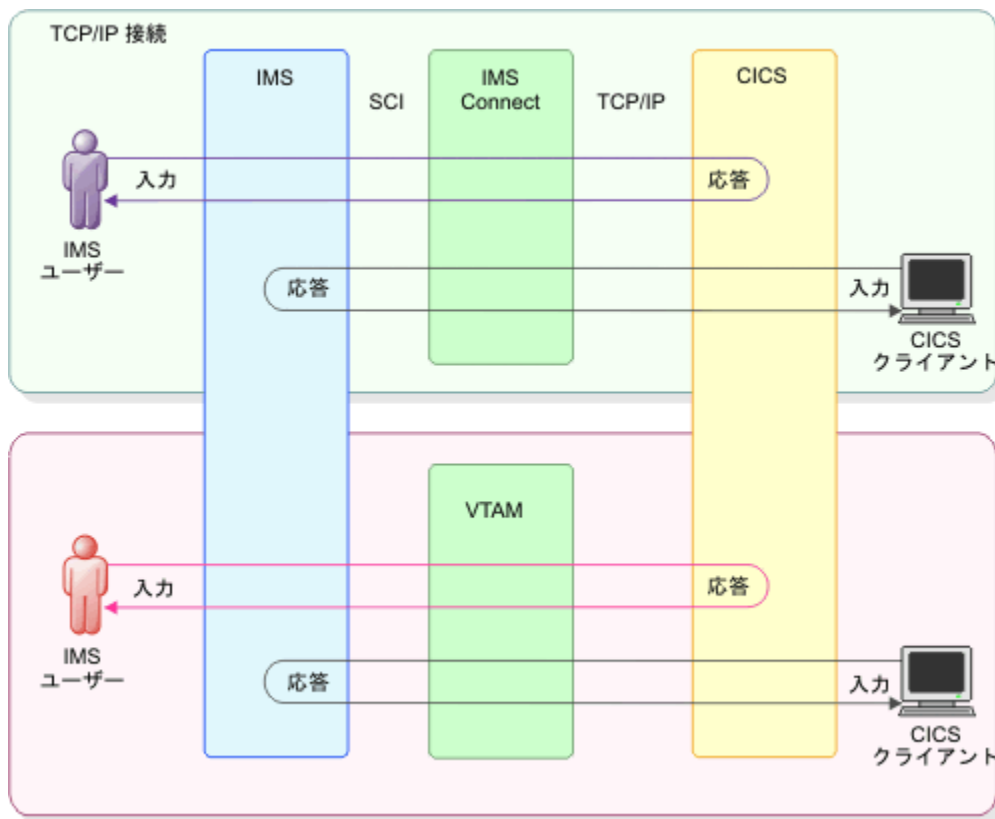


図 83. ISC TCP/IP と ISC VTAM 通信フローの比較

### 関連タスク

61 ページの『[拡張端末オプション \(ETO\) \(Extended Terminal Option \(ETO\)\)](#)』

これらのトピックでは、拡張端末オプション (ETO) を紹介し、ETO の概要 および IMS TM ネットワーク内の ETO 端末の 管理に必要な情報を説明します。

145 ページの『[IMS Connect および TCP/IP 通信](#)』

IMS の IMS Connect 機能は、TCP/IP 対応の環境から IMS DB および IMS TM の両方へのアクセスを提供します。

## ISC TCP/IP サポートの要件

ISC TCP/IP 機能の要件には、CICS、ETO、IMS Connect、 および最小限の CSL に関する要件が含まれます。

- IBM CICS Transaction Server for z/OS バージョン 5.1 以降を使用する必要があります。
- IMS に対して TCP/IP ソケット接続サポートを提供するために、IMS Connect が必要です。
- 少なくとも Structured Call Interface (SCI) および Operations Manager (OM) を備えた Common Service Layer (CSL) が必要です。IMS と IMS Connect の間で通信するために SCI が必要であり、タイプ 2 コマンドのサポートのために OM が必要です。
- OM API または REXX SPOC API に対してタイプ 2 コマンドを発行するには、IMS TSO SPOC などの単一制御点 (SPOC) プログラムを使用する必要があります。
- 動的に定義された端末を ISC TCP/IP で使用する IMS サブシステムごとに、IMS 拡張端末オプション (ETO) が必要です。

### 関連概念

[Common Service Layer の概要 \(システム管理\)](#)

[CSL での単一制御点 \(SPOC\) プログラム \(システム管理\)](#)

### 関連タスク

145 ページの『[IMS Connect および TCP/IP 通信](#)』

IMS の IMS Connect 機能は、TCP/IP 対応の環境から IMS DB および IMS TM の両方へのアクセスを提供します。

61 ページの『[拡張端末オプション \(ETO\) \(Extended Terminal Option \(ETO\)\)](#)』

これらのトピックでは、拡張端末オプション (ETO) を紹介し、ETO の概要 および IMS TM ネットワーク内の ETO 端末の管理に必要な情報を説明します。

## ISC TCP/IP サポートの制約事項

以下の機能は、ISC TCP/IP によってサポートされていません。

- 同期通信に SEND(INVITE)/RECV プロトコルを使用する CICS トランザクション
- SEND(LAST)/RECV プロトコルを使用する CICS トランザクション
- 拡張回復機能 (XRF)
- /DISPLAY および /RDISPLAY を除く IMS オペレーター・コマンド
- IMS 会話型モード・トランザクション
- 高速機能を含めた IMS 応答モード・トランザクション
- IMSplex 端末管理 (STM)
- フロントエンド切り替え (FES)
- メッセージ形式サービス (MFS)
- VTAM 汎用リソース (VGR)

## ISC TCP/IP 接続のセキュリティー

CICS との ISC TCP/IP 接続の保護は、ユーザー名またはサブプール名の認証、トランザクション許可、サインオン出口ルーチン、セキュリティー製品 (RACF など)、およびメッセージの暗号化によって行えます。

並列セッションが開始されると、IMS は、開始要求に関連付けられたユーザー ID を認証します。IMS は、ユーザー ID を /OPNDST コマンドの USER キーワードの値から取得するか、静的に定義された端末の場合は SUBPOOL マクロの NAME パラメーターから取得します。パスワードは使用されません。

接続で送信されるトランザクションごとに、IMS は、そのトランザクションをスケジュールに入れるためのユーザー ID の権限を検査します。

静的に定義された ISC TCP/IP 端末の場合、IMS は、セッション開始中にユーザー ID を認証し、AUTOSIGN オプションが使用可能になっている場合にのみユーザー ID のトランザクション権限を検査します。

IMS の場合、ISC TCP/IP 接続でのメッセージの暗号化は、z/OS TCP/IP スタックの Application Transparent Transport Layer Security (AT-TLS) によって提供できます。AT-TLS は、IMS Connect から独立して、メッセージの暗号化および暗号化解除を行うことができます。IMS Connect によって送受信されるメッセージは暗号化されません。IMS Connect は、並列セッションに別々の送信ソケットと受信ソケットを使用するため、AT-TLS 管理者は両方のソケットをサポートするように AT-TLS を構成する必要があります。

AT-TLS を使用して、IMS によって受信される ISC TCP/IP 接続要求でのクライアント認証を実行することもできます。CICS は、ISC TCP/IP 接続を開始する時にセキュリティー証明書を提供するように構成されている必要があります。接続要求と共に証明書が提供されないと、AT-TLS はメッセージを発行し、接続要求をリジェクトします。IMS Connect はエラー・メッセージ HWSV5000E を発行します。

IMS での AT-TLS 暗号化およびクライアント認証サポートの構成は、CICS での SSL サポートの構成と調整する必要があります。CICS では、IPCONN リソース定義と TCP/IP SERVICE リソース定義に IPIC バインド時セキュリティーを実装することにより、ISC TCP/IP 接続に対して SSL サポートが使用可能になっています。

**制約事項:** CICS との ISC TCP/IP 接続では、IMS の /SIGN コマンドはサポートされていません。

AT-TLS について詳しくは、「[z/OS Communications Server: IP 構成ガイド](#)」を参照してください。

### 関連資料

[/OPNDST コマンド \(コマンド\)](#)

## CICS との ISC TCP/IP 接続のセットアップ

CICS との ISC TCP/IP 接続のセットアップには、IMS および IMS Connect でのステップに加えて、CICS 管理者との調整が必要です。

### 始める前に

**前提条件:** CICS との ISC TCP/IP 接続を定義するためには、その前に、IMS で以下の機能が使用可能になっている必要があります。

- Common Service Layer (CSL) の構造化呼び出しインターフェース (SCI) コンポーネントと Operations Manager (OM) コンポーネント
- 動的に定義された ISC 端末またはノードを使用している場合、拡張端末オプション (ETO)

### 次のタスク

以下のトピックは、CICS との ISC TCP/IP 接続の定義方法について説明しています。

## ISC TCP/IP 接続の端末の定義

動的に定義された ISC 端末と静的に定義された ISC 端末は両方とも TCP/IP を使用できますが、新しい端末を定義する場合は動的端末を使用してください。動的端末は、システム定義も IMS のコールド・スタートも必要とせず、静的 ISC 端末に比べて柔軟性が大幅に向上します。

### このタスクについて

動的 ISC 端末は、IMS.PROCLIB データ・セットの DFSDSCMx メンバーまたは DFSDSCTy メンバーにログオン記述子をコーディングすることで定義します。

静的 ISC TCP/IP 端末と動的 ISC TCP/IP 端末は、以下に示すように多くの点で同等です。

- 接続は、IMS コマンドを使用して開始または終了できる。
- セッション状況は、IMS コマンドを使用して表示できる。
- 入出力メッセージは、静的端末または動的端末を使用して転送できる。

静的端末と動的端末とで異なる重要な側面は、静的端末が IMS ステージ 1 システム定義マクロを使用して定義しなければならない、IMS の再起動が必要であるのに対して、動的端末は、IMS.PROCLIB データ・セットの拡張端末オプション (ETO) ログオン記述子によって定義されている端末属性を使用して、必要に応じて実行時に作成される点です。

### ISC TCP/IP セッションの動的端末の定義

IMS 拡張端末オプション (ETO) が使用可能であるとき、ETO ログオン記述子を使用して、ISC TCP/IP LU 6.1 セッションの端末を動的作成できます。

### このタスクについて

IMS で ISC TCP/IP 端末の動的な作成を可能にするには、少なくとも 1 つのログオン記述子が定義されている必要があります。動的に定義された ISC TCP/IP ノードをサポートする IMS Connect の各インスタンスに対して少なくとも 1 つのログオン記述子が定義されている必要があります。

ログオン記述子は、次に示す IMS.PROCLIB データ・セット内の 2 つのメンバーに保管されます。

#### DFSDSCMx

コールド・スタート中に IMS によって作成されるデフォルトのログオン記述子用。

#### DFSDSCTy

ご使用のシステムによって作成されるログオン記述子用。

**推奨事項:** 独自のログオン記述子を作成し、ステージ 1 システム定義処理全体でそれらを保存する場合は、ログオン記述子を DFSDSCTy PROCLIB メンバーに保管してください。

DFSDSCMx メンバー内の記述子は追加または変更することができます。ただし、ステージ 1 システム定義の処理中、IMS は DFSDSCMx メンバーの内容を削除し、すべてのデフォルト記述子を再作成します。

IMS は、DFSTCP という名前の、ISC TCP/IP 端末用のデフォルト・ログオン記述子を作成します。デフォルト ISC TCP/IP 記述子を、IMS で開始されたセッションに使用するには、/OPNDST コマンドの LOGOND キーワードを使用してそれを明示的に指定する必要があります。CICS から開始された ISC TCP/IP セッションはデフォルト・ログオン記述子を使用できますが、それは、他のどのログオン記述子も、CICS によって指定されたノード名に一致しない場合に限定されます。

## 手順

1. ISC TCP/IP 端末用のログオン記述子を作成するには、UNITYTYPE=ISCTCPIP と LCLICON=*lcl\_imsconnect\_name* を指定する ETO ログオン記述子をコーディングします。  
LCLICON キーワードに指定された IMS Connect インスタンスは、CICS サブシステムのネットワーク・アドレスおよびポートを含め、TCP/IP サポートを提供します。
2. オプション: ETO ユーザー記述子を定義します。

## 例

例えば、次のようにします。

```
L ISCTCP1 UNITYTYPE=ISCTCPIP LCLICON=HWS1
L ISCTCP2 UNITYTYPE=ISCTCPIP LCLICON=HWS1
L ISCTCP3 UNITYTYPE=ISCTCPIP LCLICON=HWS1
L ISCTCP4 UNITYTYPE=ISCTCPIP LCLICON=HWS1
L ISCTCP5 UNITYTYPE=ISCTCPIP LCLICON=HWS1
L TERMA UNITYTYPE=3270 UNIT=3284
L DFS327P UNITYTYPE=3270 UNIT=3284
U DFSUSER OPTIONS=(TRANRESP)
```

## 関連概念

### 65 ページの『ETO 記述子』

記述子は、IMS がログオンまたはサインオンに必要な動的リソースを構築する際に、IMS に情報を提供します。ETO 記述子には、ログオン記述子、ユーザー記述子、MSC 記述子、および MFS 装置記述子の 4 つのタイプがあります。

### ログオン記述子 (システム定義)

### ETO 記述子 (システム定義)

### ETO 記述子のオーバーライド (システム定義)

## 関連タスク

### 85 ページの『ログオン記述子の作成』

ログオン記述子は、ログオン・セッションを設定した端末の物理的特性についての情報を IMS に提供します。これらの特性は、VTAM ログオン BIND 特性と一致している必要があります。

### 88 ページの『ユーザー記述子の作成』

ユーザー記述子はユーザー・オプションとユーザー構造に関連している情報を提供します。IMS は、ユーザーが ETO 端末を使用できるようにする制御ブロック作成のためにユーザー記述子を必要とします。

## 静的に定義された LU 6.1 ISC 端末で TCP/IP を使用するための構成

静的に定義された ISC 端末で TCP/IP を使用するように構成するには、IMS のコールド・スタートとステージ 1 システム定義マクロおよび DFSDCxxx PROCLIB メンバーの更新を行う必要があります。

## このタスクについて

デフォルトにより、静的に定義された ISC 端末は通信に VTAM を使用します。DFSDCxxx PROCLIB メンバーの ISCTCPIP パラメーターは、静的に定義された ISC 端末に対して TCP/IP を使用可能にします。

**要件:** TCP/IP をサポートするには、並列セッションをサポートするように ISC を定義する必要があります。ISC TCP/IP 通信では、単一セッション ISC 端末はサポートされません。

静的に定義された ISC 端末は、TYPE ステージ 1 システム定義マクロに UNITYTYPE=LUTYPE6 を指定する必要があります。

静的に定義された LU 6.1 ISC 端末で TCP/IP を使用するように構成するには、以下のようにします。

## 手順

1. 既存の端末を VTAM から TCP/IP に切り替える場合は、端末を定義するシステム定義マクロの TYPE マクロで UNITTYPE=LUTYPE6 が指定されていることを確認します。

UNITTYPE=LUTYPE6 が指定されていない場合、DFSDCxxx メンバー内の ISCTCPIP パラメーターの処理時にエラー・メッセージが発行され、その端末に対して TCP/IP は使用可能にされません。

2. 新規の端末を定義する場合は、適切なシステム定義マクロをコーディングして端末を定義します。TYPE マクロは UNITTYPE=LUTYPE6 を指定する必要があるため、VTAMPOOL マクロと SUBPOOL マクロをコーディングすることにより並列セッションを定義する必要があります。例えば、次のようにします。

```
TYPE UNITTYPE=LUTYPE6
TERMINAL NAME=LU6NDPA,MSGDEL=SYSINFO,EDIT=(NO,NO), X
SESSION=3,OPTIONS=NOMTOMSG, X
COMPT1=(SINGLE1,VLVB), X
COMPT2=(SINGLE2,VLVB), X
COMPT3=MULT1
VTAMPOOL
SUBPOOL NAME=LU6SPA,MSGDEL=SYSINFO
NAME LU6LTPA1,COMPT=2,ICOMPT=1
SUBPOOL NAME=LU6SPB
NAME LU6LTPB1
SUBPOOL NAME=LU6SPC
NAME LU6LTPC1
```

UNITTYPE=LUTYPE6 が指定されていない場合、IMS は DFSDCxxx メンバー内の ISCTCPIP パラメーターの処理時にエラー・メッセージを発行し、その端末に対して TCP/IP は使用可能にされません。

3. DFSDCxxx PROCLIB メンバーで、端末の名前と TCP/IP サポートを提供する IMS Connect インスタンスの名前を ISCTCPIP キーワードに指定することにより、TCP/IP 通信を使用する端末を定義します。例えば、次のように指定します。ISCTCPIP=(LU6NDPA,ICON1)
4. IMS をコールド・スタートします。

IMS は、ウォーム再始動中または緊急時再始動中は、ISCTCPIP キーワードへの変更を読み取りません。

## 関連資料

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

[IMS 環境で使用されるマクロ \(システム定義\)](#)

## 静的 ISC TCP/IP 端末の TCP/IP から VTAM へのフォールバック

静的 LU 6.1 ISC 端末で TCP/IP から VTAM へフォールバックするには、DFSDCxxx PROCLIB メンバーの更新と IMS のコールド・スタートを行う必要があります。

## このタスクについて

LU 6.1 ISC 端末のデフォルトの通信プロトコルは VTAM です。

VTAM へのフォールバックは、以下のようにして行います。

## 手順

1. DFSDCxxx PROCLIB メンバーから、端末のノード名を指定する ISCTCPIP パラメーターを削除します。
2. IMS をコールド・スタートします。

ウォーム再始動または緊急時再始動が行われた場合は、ISCTCPIP パラメーターは有効のままです。

## 関連資料

[IMS PROCLIB データ・セットの CSLDCxxx メンバー \(システム定義\)](#)



## ISC TCP/IP リンクのための IMS Connect サポートの定義

ISC TCP/IP のための IMS Connect サポートは、IMS.PROCLIB データ・セットの IMS Connect 構成メンバーの ISC、RMTICICS、および TCPIP ステートメントによって定義されます。

### このタスクについて

ISC ステートメントは、IMS Connect に対する ISC TCP/IP リンクのほとんどの属性を定義します。

RMTICICS ステートメントは、CICS サブシステムのネットワーク・アドレス指定情報を定義します。

TCPIP ステートメントは、IMS Connect がトランザクションおよびデータ応答を CICS から受信するポートを指定します。

IMS サブシステムと CICS サブシステム間に 1 つ以上の TCP/IP リンクを定義することができます。

次のように、同じ CICS サブシステムに接続する ISC リンクは、同じ RMTICICS 接続定義を使用するか、別々に定義された RMTICICS 接続を使用することができます。

- 同じ RMTICICS 接続定義を使用する ISC リンクは、RMTICICS パラメーターに同じ値を指定する必要があります。
- 同じ CICS サブシステムに対して別々に定義された RMTICICS 接続を使用する ISC リンクは、RMTICICS パラメーターに異なる ID を指定する必要があります。ただし、それぞれの ISC ステートメントが参照する別々の RMTICICS ステートメントは、同じホスト名または IP アドレスとポート番号を指定している必要があります。

ISC TCP/IP に対する IMS Connect サポートを定義するには、以下のステートメントをコーディングしてください。

### 手順

- ISC ステートメント。  
ISC ステートメントをコーディングするには、以下の値が必要です。
  - ローカル IMS システムの IMS ID
  - リモート CICS サブシステム CICS の APPLID
  - IMS への ISC リンクを定義するノード名。
  - IMS および IMS Connect が IMSplex によって認識される名前。
  - IMS Connect が CICS サブシステムからトランザクションおよびデータ 応答を受信するポートの CICSSPORT 番号。
  - CICS サブシステムのネットワーク・アドレスとポートを定義する RMTICICS ステートメントの ID。
- RMTICICS ステートメント。  
RMTICICS ステートメントをコーディングするには、以下の値が必要です。
  - CICS サブシステムのホスト名。この値は、TCPIP ステートメントの CICSSPORT キーワードにも指定されます。
  - CICS サブシステムが IMS からトランザクションおよびデータ 応答を受信するポートのポート番号。
- TCPIP ステートメント。  
ISC TCPIP サポートを既存の TCPIP ステートメントに追加するには、CICSSPORT パラメーターを指定して、IMS Connect が CICS からトランザクションおよびデータ 応答を受信するポートを定義します。この値は、ISC ステートメントの CICSSPORT キーワードにも指定されます。

### 例

以下の例は、ISC TCP/IP に対する IMS Connect サポートを提供する IMS Connect 構成メンバーを示しています。完全を期するために、HWS ステートメントが含まれています。

```
HWS=(ID=HWS1,XIBAREA=100,RACF=N)
TCPIP=(HOSTNAME=TCPIP,PORTID=(9998,19998,LOCAL),RACFID=GOFISHIN,
```



```

TIMEOUT=000,
IPV6=Y,SSLP0RT=(8899),SSLENVAR=HWSCFSSL,
PORT=(ID=15554),
CICSPORT=(ID=1111,KEEPAV=1000),
CICSPORT=(ID=3333,KEEPAV=1000),
CICSPORT=(ID=6666,KEEPAV=1000),
CICSPORT=(ID=7777,KEEPAV=1000),
EXIT=(HWSSMPL0,HWSSMPL1,HWSCSL00,HWSCSL01,HWSOAP1)
ISC=(ID=CICSA1,NODE=LU6NDPA,
IMSPLEX=(MEMBER=HWS1,TMEMBER=PLEX1),
LCLIMS=IMS1,RMTCICS=CICS1,CICSAPPL=CICS1,CICSPORT=1111)
ISC=(ID=CICSA2,NODE=LU6NDPB,
IMSPLEX=(MEMBER=HWS1,TMEMBER=PLEX1),
LCLIMS=IMS1,RMTCICS=CICS1,CICSAPPL=CICS1,CICSPORT=3333)
ISC=(ID=CICSA5,NODE=ISCTCP1,
IMSPLEX=(MEMBER=HWS1,TMEMBER=PLEX1),
LCLIMS=IMS1,RMTCICS=CICS1,CICSAPPL=CICS1,CICSPORT=6666)
ISC=(ID=CICSA6,NODE=ISCTCP2,
IMSPLEX=(MEMBER=HWS1,TMEMBER=PLEX1),
LCLIMS=IMS1,RMTCICS=CICS1,CICSAPPL=CICS1,CICSPORT=7777)
RMTCICS=(ID=CICS1,HOSTNAME=ABC.EXAMPLE.COM,PORT=23456)

```

## 関連資料

[RMTCICS ステートメント \(システム定義\)](#)

[ISC ステートメント \(システム定義\)](#)

[TCPIP ステートメント \(システム定義\)](#)

## CICS での ISC TCP/IP リンクの定義

2 つの CICS リソース定義 (IPCONN リソースおよび TCPIP SERVICE リソース) により、CICS で ISC TCP/IP リンクが IP 相互接続 (IPIC) 接続として構成されます。

### このタスクについて

CICS IPCONN リソースは、CICS が IPIC 接続でメッセージを送信するために必要な属性を定義します。CICS パースペクティブから、IPCONN ステートメントは IPIC 接続のアウトバウンド属性を定義します。

各並列セッションに 1 つの IPCONN リソースが必要です。CICS では、IPCONN リソースは、CICS 自動インストール・プログラムを使用して動的に作成するか、ユーザー自身で定義することができます。

IPCONN リソース定義で、以下のパラメーターは、IMS に定義された値に一致している必要があります。

- IPCONN(*ipconnname*) は、IMS /OPNDST コマンドに指定されたユーザー名に一致しているか、静的に定義された ISC TCP/IP 端末の場合は、SUBPOOL IMS システム定義マクロの NAME キーワードに一致している必要があります。
- HOST は、IMS.PROCLIB データ・セットの IMS Connect HWSCFGxx メンバー内の TCPIP 構成ステートメントの HOSTNAME キーワードに指定されている、IMS Connect のホスト名に一致している必要があります。
- PORT は、IMS.PROCLIB データ・セットの IMS Connect HWSCFGxx メンバー内の RMTCICS 構成ステートメントと TCPIP 構成ステートメント両方の CICSPORT パラメーターに指定されている値に一致している必要があります。IMS Connect は、このポートで CICS からのメッセージを受信します。

CICS TCPIP SERVICE リソースは、CICS が IPIC 接続でメッセージを受信するために必要な属性を定義します。CICS パースペクティブから、TCPIP SERVICE リソースは、IPIC 接続のインバウンド属性を定義します。

TCPIP SERVICE リソース定義で、以下のパラメーターは、IMS に指定されている値に一致している必要があります。

- HOST は、TCPIP 構成ステートメントの HOSTNAME キーワードに指定されている、IMS Connect のホスト名に一致している必要があります。
- PORT は、ISC 構成ステートメントの PORT パラメーターで IMS Connect に対して指定されている値に一致している必要があります。

## 例

次の例は、CICS IPCONN および TCPIP SERVICE リソース用の定義を示しています。

```
DELETE GROUP(TCPIPGRP)
DEFINE TCPIP SERVICE(TSA)
      PORT(23456)
      PROTOCOL(IPIC)
      GROUP(TCPIPGRP)

DEFINE IPCONN(LU6SPA)
      APPLID(LU6SPA)
      AUTOCONNECT(YES)
      HOST(XYZ.EXAMPLE.COM)
      PORT(1111)
      SENDCOUNT(5)
      RECEIVECOUNT(5)
      TCPIP SERVICE(TSA)
      GROUP(TCPIPGRP)

DEFINE IPCONN(LU6SPB)
      APPLID(LU6SPB)
      AUTOCONNECT(YES)
      HOST(XYZ.EXAMPLE.COM)
      PORT(3333)
      SENDCOUNT(5)
      RECEIVECOUNT(5)
      TCPIP SERVICE(TSA)
      GROUP(TCPIPGRP)

DEFINE IPCONN(LU6SPC)
      APPLID(LU6SPC)
      AUTOCONNECT(YES)
      HOST(XYZ.EXAMPLE.COM)
      PORT(3333)
      SENDCOUNT(5)
      RECEIVECOUNT(5)
      TCPIP SERVICE(TSA)
      GROUP(TCPIPGRP)

DEFINE IPCONN(USER01)
      APPLID(USER01)
      AUTOCONNECT(YES)
      HOST(XYZ.EXAMPLE.COM)
      PORT(6666)
      SENDCOUNT(5)
      RECEIVECOUNT(5)
      TCPIP SERVICE(TSA)
      GROUP(TCPIPGRP)

DEFINE IPCONN(USER02)
      APPLID(USER02)
      AUTOCONNECT(YES)
      HOST(XYZ.EXAMPLE.COM)
      PORT(6666)
      SENDCOUNT(5)
      RECEIVECOUNT(5)
      TCPIP SERVICE(TSA)
      GROUP(TCPIPGRP)

DEFINE IPCONN(USER03)
      APPLID(USER03)
      AUTOCONNECT(YES)
      HOST(XYZ.EXAMPLE.COM)
      PORT(7777)
      SENDCOUNT(5)
      RECEIVECOUNT(5)
      TCPIP SERVICE(TSA)
      GROUP(TCPIPGRP)

DEFINE IPCONN(USER04)
      APPLID(USER04)
      AUTOCONNECT(YES)
      HOST(XYZ.EXAMPLE.COM)
      PORT(7777)
      SENDCOUNT(5)
      RECEIVECOUNT(5)
      TCPIP SERVICE(TSA)
      GROUP(TCPIPGRP)

DEFINE TRANSACTION(SR1A)
```

```
PROGRAM(IMSSRT1A)
GROUP(DFHPPTDR)

DEFINE PROGRAM(IMSSRT1A)
GROUP(DFHPPTDR)
LANGUAGE(COBOL)

ADD GROUP(TCPIPGRP) LIST(DRVRLIST)
```

## ISC TCP/IP リンクでの CICS とのセッションの開始

ISC TCP/IP リンクでの ISC 並列セッションは、IMS から、またはパートナー CICS システムから開始できます。

### IMS からの ISC TCP/IP リンクでの並列セッションの開始

IMS から、**/OPNDST NODE** コマンドを発行して、CICS との ISC TCP/IP 並列セッションを開始します。

#### 始める前に

##### 前提条件：

- CICS で ISC TCP/IP セッションを開始するためには、その前に、611 ページの『[CICS との ISC TCP/IP 接続のセットアップ](#)』に記載されているように、IMS、IMS Connect、および CICS に接続が定義されている必要があります。
- IMS で、ISC 端末は TCP/IP を使用するように定義されている必要があります。端末が TCP/IP を使用するように定義されていない場合、IMS は VTAM を使用してセッションを開始しようとします。

#### このタスクについて

TCP/IP で CICS との ISC 並列セッションを開始するには、以下の情報が必要です。

- ターゲット CICS サブシステムを識別するノード名。
- 並列セッションのユーザー名。
- 動的に定義された端末を使用するセッションの場合、接続の端末属性を定義し、TCP/IP サポートを提供する IMS Connect インスタンスを識別する ETO ログオン記述子の名前。

ノード名は、IMS Connect へのターゲット CICS サブシステムを定義する ISC ステートメントの **NODE** キーワードで見つかります。

また、静的に定義された端末を使用するセッションの場合、ノード名は、**DFSDCxxx PROCLIB** メンバーの **ISCTCPIP** キーワード (端末を、TCP/IP を使用するとして定義する) と **TERMINAL** システム定義マクロの **NAME** キーワードで IMS に定義されています。

並列セッションのユーザー名を指定するための要件は、並列セッションおよび ISC 端末が静的に定義されているか動的に定義されているかによって異なります。

IMS または CICS のいずれかで静的定義が使用されている場合、**/OPNDST NODE** コマンドに指定するユーザー名は、静的定義内の対応する値に一致している必要があります。IMS で、静的に定義された並列セッションのユーザー名は、**SUBPOOL** システム定義マクロの **NAME** キーワードによって定義されます。CICS では、ユーザー名は、**IPCONN** 名、または **IPCONN** リソース定義の **APPLID** キーワードによって定義されます。

IMS で ISC 端末が動的に定義されている場合、ユーザー名は、IMS システムの **/OPNDST NODE** コマンドの **USER** キーワードにのみ指定されます。

CICS 自動インストール機能が CICS サブシステムで使用可能になっており、IMS から並列セッションが開始された場合、CICS は IMS に定義されたユーザー名を使用して **IPCONN** リソースを指定します。ただし、ユーザー名の長さが 4 文字を超える場合、CICS は最後の 4 文字のみを使用します。

**/OPNDST NODE** コマンドの **USER** キーワードに指定されるユーザー名には、最大 8 文字を使用できます。ただし、CICS で **IPCONN** 自動インストール・プログラムがアクティブになっている場合、CICS はユーザー名の最後の 4 文字のみを使用します。

ETO ログオン記述子は、TCP/IP サポートを提供する IMS Connect インスタンスの名前を含め、CICS への ISC 接続に使用される動的端末の属性を定義します。

**/OPNDST NODE** コマンドでのログオン記述子の指定はオプションです。LOGOND キーワードが指定されず、並列セッションがまだオープンされていない場合は、NODE キーワードの値を使用してログオン記述子が検索されます。あるいは、ログオン出口ルーチン (DFSLGNX0) によってログオン記述子の名前を提供することもできます。

並列セッションがノードで既にオープンされており、最初の並列セッションを開始するために使用されたログオン記述子とは異なるログオン記述子が指定されている場合、**/OPNDST NODE** コマンドはリジェクトされます。

**/OPNDST NODE** コマンドに指定されたログオン記述子が TCP/IP サポートを指定していない場合、IMS は、VTAM ノードとしてセッションをオープンしようとします。

**/OPNDST NODE** コマンドは、CICS との ISC TCP/IP 並列セッションを IMS から開始するための唯一の方法です。

**制約事項:** ISC は、TCP/IP 接続での並列セッションのみをサポートします。

## 手順

CICS との ISC 並列セッションは、**/OPNDST NODE** コマンドを発行することによって開始できます。

## タスクの結果

セッションが正常に開始されると、IMS はマスター端末にメッセージ DFS2064I を発行し、IMS と CICS は機能交換を実行して、セッションおよびサーバーの属性を検査します。

### 関連概念

65 ページの『[ETO 記述子](#)』

記述子は、IMS がログオンまたはサインオンに必要な動的リソースを構築する際に、IMS に情報を提供します。ETO 記述子には、ログオン記述子、ユーザー記述子、MSC 記述子、および MFS 装置記述子の 4 つのタイプがあります。

[ログオン記述子 \(システム定義\)](#)

### 関連タスク

85 ページの『[ログオン記述子の作成](#)』

ログオン記述子は、ログオン・セッションを設定した端末の物理的特性についての情報を IMS に提供します。これらの特性は、VTAM ログオン BIND 特性と一致している必要があります。

### 関連資料

[/OPNDST コマンド \(コマンド\)](#)

## CICS からの ISC TCP/IP リンクでの並列セッションの開始

IBM CICS Transaction Server for z/OS から、CICS コマンド SET IPCONN(*ipconnnm*) ACQUIRE を発行して IMS との ISC TCP/IP 並列セッションを開始します。

## 始める前に

**前提条件:** CICS から IMS との ISC TCP/IP セッションを開始するためには、その前に、以下のことを実行する必要があります。

- 611 ページの『[CICS との ISC TCP/IP 接続のセットアップ](#)』に記載されているように、IMS、IMS Connect、および CICS に接続が定義されている必要があります。
- IMS で、ISC 端末は TCP/IP を使用するように定義されている必要があります。TCP/IP を使用するように IMS に端末が定義されていない場合、IMS はセッションの開始要求をリジェクトします。

## このタスクについて

CICS から ISC TCP/IP セッションを開始するには、CICS に ISC 接続を定義する IPCONN リソースからの 1 文字から 4 文字の IPCONN 名が必要です。この値は、IMS で並列セッションを識別する USER 値と同じです。

LUTYPE 6.1 リンクを ISC VTAM から ISC TCP/IP にマイグレーションした場合、IPCONN 名は、ISC VTAM リンクを定義した CONNECTION リソースの名前と同じである可能性が高くなります。

SET IPCONN ACQUIRE コマンドは、以下を含め、さまざまな CICS インターフェースから発行することができます。

- CICS Explorer®
- CICS システム・プログラミング・インターフェース (SPI)
- The CICS マスター端末トランザクション CEMT
- CICS コマンド・レベル・インタープリター (CECI) トランザクション

## 手順

例えば、CEMT を使用して CICS から IMS との並列セッションを開始するには、以下のコマンドを発行します。

```
➡ CEMT SET IPCONN( ipconnnm ) ACQUIRED ➡
```

## タスクの結果

セッションが正常に開始されると、IMS はマスター端末にメッセージ DFS2064I を発行し、IMS と CICS は機能交換を実行して、セッションおよびサーバーの属性を検査します。

セッションが開始されたことを確認するには、いずれかの CICS コマンド・インターフェースを使用して、INQUIRE IPCONN(*ipconnnm*) コマンドを発行します。

## 次のタスク

セッションが開始された後、CICS アプリケーション・プログラムは、CICS の START コマンドと RETRIEVE コマンドを使用して IMS 非応答モードおよび非会話型トランザクション、メッセージ通信、および IMS の /DISPLAY、/RDISPLAY、および /SIGN オペレーター・コマンドを処理することができます。

### 関連資料

[CICS Transaction Server for z/OS](#)

## ISC TCP/IP セッションの終了

TCP/IP を使用する ISC セッションの終了は、一般に、VTAM を使用する ISC セッションの終了と同じです。

## このタスクについて

終了が完了する前に進行中の作業が終了した場合は、適正な方法で ISC TCP/IP セッションを終了することができます。または、終了が即時で、終了が完了する前に進行中の作業を終了できない場合は、ISC TCP/IP セッションを無条件に終了することができます。

セッションが異常終了した場合、進行中の作業は、通常、終了が完了する前に終了できません。

### 関連資料

[/CLSDST コマンド \(コマンド\)](#)

[/QUIESCE コマンド \(コマンド\)](#)

[/STOP NODE コマンド \(コマンド\)](#)

[/CHECKPOINT コマンド \(コマンド\)](#)

## ISC TCP/IP セッションの正規手順での終了

正規手順での終了では、IMS とパートナー・サブシステムの両方が通常の処理を完了してからセッションが終了されます。作業が保留中のままになることはなく、接続はコールド状態で終了されます。

### このタスクについて

接続の再開時に、IMS とパートナー・サブシステムが接続を再同期する必要はありません。

### 手順

- 既に進行中の作業を中断せずに ISC TCP/IP セッションを終了するには、**/QUIESCE IMS** タイプ 1 コマンドを発行します。  
パラメーター **FREEZE**、**PURGE**、または **DUMPQ** を指定した IMS の **/CHECKPOINT** コマンド、および **QUIESCE** パラメーターも、正規手順での終了を開始します。**QUIESCE** パラメーターは、セッション終了前にメッセージ・キューが空になっていることを確認します。すべての端末が、シャットダウンが完了したことを示すと、IMS はチェックポイント処理を完了します。

## ISC TCP/IP セッションの無条件終了

ISC TCP/IP セッションを無条件で終了する場合、終了時点で進行中の作業は、セッション終了前に完了しません。接続を再開した場合、IMS とパートナー・サブシステムが接続を再同期してからでないと、保留中の作業を終了できません。

### このタスクについて

IMS MTO は、IMS の **/CLSDST** コマンド、**/STOP** コマンド、または **/CHECKPOINT** コマンドを発行して、適正なセッション終了の処理中に、即時ネットワークを終了することができます。**/STOP** コマンドは、セッションを STOPPED 状態にします。

IMS Connect コマンドを使用してセッションを終了しても、セッションは即時シャットダウンされます。

### 手順

- ISC TCP/IP セッションを即時終了するには、**/CLSDST IMS** タイプ 1 コマンドを発行します。これにより、すべての処理が即時終了されます。

## ISC TCP/IP セッションの異常終了

ISC TCP/IP セッションの異常終了は、伝送エラーやプロトコル・エラー、あるいは受信側メッセージ処理プログラムでデータを受け入れられないようなデータのエラーが原因で発生する可能性があります。

ISC セッションでは 2 つの対等レベルのシステムが使用されるため、エラー・リカバリー処理とセッションの異常終了処理は、どちらのシステムが接続を開始したかによって異なる可能性があります。

異常終了時に進行中だった作業は、セッションが終了するまで完了されません。接続を再開した場合、IMS とパートナー・サブシステムが接続を再同期してからでないと、保留中の作業を終了できません。

## ISC TCP/IP セッションの再始動

**/OPNDST** コマンドを発行して、TCP/IP を使用する ISC セッションを再始動します。

### このタスクについて

CICS AUTOCONNECT 機能は、CICS と IMS が初めて通信する際にのみ自動的に接続を獲得します。CICS 領域をシャットダウンして再始動した場合、CICS は、ローカル CICS カタログから IPCONN リソースをリカバリーしますが、ISC TCP/IP セッションを自動的に再獲得することはありません。通信を再確立するには、静的に定義された端末と動的に定義されたノードの両方に対して、IMS **/OPNDST** コマンドを発行する必要があります。CICS AUTOCONNECT 機能は、リソース定義を含む CICS DFHCSD データ・セット内の IPCONN ステートメントで設定されます。



IMS Connect 内で ISC TCP/IP 接続が停止した場合は、IMS で ISC 通信を再始動する前に、必ず IMS Connect 内で接続を再始動してください。IMS Connect では、IMS と IMS Connect の間の ISC 接続、または IMS Connect と CICS の間の TCP/IP 接続、あるいはその両方で通信が停止する可能性があります。

## TCP/IP を介する場合に ISC によってサポートされる CICS フロントエンド・トランザクション・タイプ

フロントエンド・システムとして、CICS は、IMS との ISC TCP/IP 接続用に非同期 START および RETRIEVE インターフェースのみをサポートします。

したがって、CICS アプリケーション・プログラムは、IMS 非応答モードまたは非会話型モードのトランザクションを使用する必要があります。CICS アプリケーション・プログラムは、TCP/IP に同期または非同期の SEND インターフェースおよび RECEIVE インターフェースを使用できません。CICS 非同期 SEND インターフェースおよび RECEIVE インターフェース、または CICS 同期分散トランザクション処理 (DTP) を ISC に使用するには、ISC 接続は VTAM を使用する必要があります。

IMS がフロントエンド・システムの場合、ISC TCP/IP 接続での IMS から CICS へのトランザクション・フローも、ISC VTAM 接続と同様に非同期になります。

## CICS アプリケーション内での CICS EXEC コマンドの一般的なフロー

CICS EXEC コマンド・レベルのアプリケーション・プログラミング・インターフェースを使用する CICS アプリケーション・プログラムをどのように設計するかは、ISC セッションで送信中のトランザクションの処理を SEND/RECEIVE、SEND LAST、および START/RECEIVE のうちのどれを使用して行うのかによって決まります。

発行された EXEC コマンドの順序は、トランザクションがリカバリー可能として定義されたか、リカバリー不能として定義されたかによって判別されます。プログラム設計も、CICS が (トランザクションを開始する) フロントエンド・システムであるか、(IMS トランザクションに返答する) バックエンド・システムであるかによって異なります。

このトピックでは、CICS 内部の同期および非同期トランザクション処理フローの概要について記述されています。ATTACH FM ヘッダーおよび SCHEDULER FM ヘッダーの内容と機能に関する知識は、このトピックを理解する上で役立ちます。

### 関連概念

601 ページの『ISC セッションで使用可能な機能』

IMS と CICS の間の ISC セッションで使用可能な機能は、接続が TCP/IP によって提供されているのか、それとも VTAM によって提供されているのかと、さらに、処理が同期か非同期かによって異なります。

### 関連資料

645 ページの『CICS のための機能管理ヘッダーのコーディング』

CICS では、IMS によって使用されるのと同じ SNA 定義機能管理ヘッダー・フィールドのいくつかが使用されます。

## SEND/RECEIVE EXEC コマンドを使用して CICS から IMS へ

CICS がフロントエンド・サブシステムで、IMS がバックエンド・サブシステムであるシステムでは、**SEND/RECEIVE** コマンドは、(高速機能を含む) IMS 応答モードと会話型トランザクション、および IMS コマンドを処理するために使用されます。

次の図に、CICS から IMS への同期メッセージの一般的な CICS アプリケーション・プログラムのフローを示します。



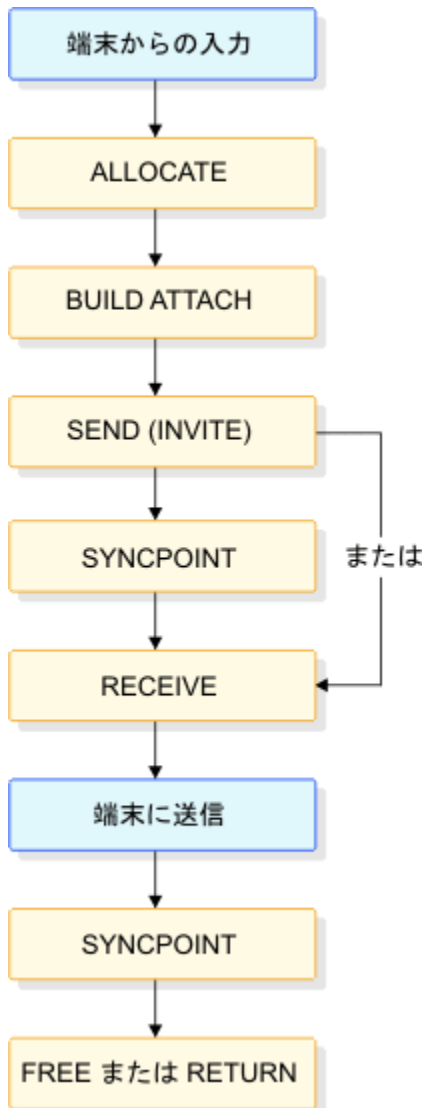


図 84. SEND/RECEIVE を使用した CICS から IMS へのアプリケーション・プログラムのフロー

この例では、CICS アプリケーションは端末から入力メッセージを読み取り、**ALLOCATE** コマンドを使用して CICS と IMS 間のセッションを設定します。

CICS アプリケーション・プログラムは、**BUILD ATTACH EXEC** コマンドを使用して、必須 ATTACH 機能管理ヘッダー（これは同期アプリケーションなので、唯一の必須ヘッダーです）を構築します。再始動に備えて、RPROCESS および RRESOURCE フィールドを **BUILD ATTACH** に指定しなければなりません。**SEND** を発行すると、出力メッセージが構成されます。**SEND INVITE** によって、方向変換 (CD) が出力メッセージに付加されます。

出力メッセージは、次のコマンドが実行されるまで、セッションの相手側に送信されません。発行される次の EXEC コマンドは、当該のトランザクションが IMS 内部でどのように定義されているかによって異なります。なぜなら、このコマンドは、同期点要求を必要に応じてメッセージに付加するだけでなく、さらにそのメッセージを ISC セッションで IMS に送信しなければならないからです。このトランザクションが IMS に対してリカバリー可能と定義されている場合には、次のコマンドは **SYNCPOINT** でなければなりません。**SYNCPOINT** によって、アウトバウンド・メッセージは RQD2 で送信されます。トランザクションがリカバリー不能と定義されている場合は、次に **SYNCPOINT** コマンドまたは **RECEIVE** コマンドを発行することができます。**SEND** コマンドの直後に（介入コマンド **SYNCPOINT** を挟まずに）**RECEIVE** コマンドを発行すると、RQE1 を付加されたアウトバウンド・メッセージが発行されます。

**SYNCPOINT** コマンドまたは **SEND** コマンドの後に発行された **RECEIVE** が IMS 応答を読み取ります。IMS からの応答は、(ATTACH FM ヘッダーを使用して) セッション上に同期に戻されます。トランザクションの

応答が、(高速機能を含む) 応答モード・トランザクション、(/DISPLAY、/RDISPLAY、および /FORMAT を除く) コマンド、または最後の会話型応答に対するものである場合には、リカバリー可能では RQD2 EB が、リカバリー不能では RQD1 EB が付加されて、CICS に戻されます。トランザクションが最後以外の会話型応答である場合には、RQE2 および CD (会話が継続するため) を付加された応答が CICS に戻されます。

当該のトランザクションが IMS 内部でリカバリー不能照会と定義されている場合には、**CONVERSE** コマンドを使用することができます。**CONVERSE** は **SEND** コマンドが出されたかのように行動し、その直後に **RECEIVE** コマンドが続きます。**CONVERSE** で送信されたトランザクションは、BB/CD と例外応答 (RQE1) に対する要求を繰り返します。リカバリー不能と定義されたもの以外のトランザクションが **CONVERSE** を使用して送信された場合には、結果的にエラーが発生します。

**RECEIVE** を発行した後で、アプリケーション・プログラムは、追加処理を実行する前に **EXEC** インターフェース・ブロック (EIB) の値を保管し、検査しなければなりません。EIB を検査して判別したとおりに正常にトランザクション応答を受信した場合は、アプリケーションはここで同期点 (SYNCPPOINT) を発行します。この同期点を発行すると、IMS トランザクション応答に対して DR2 応答が戻されます。出力メッセージを端末に送信することも含んで、CICS が IMS 応答メッセージの内容に基づいてアプリケーション処理を行う必要がある場合には、同期点が発行される前にその処理は実行されます。これによって、確実に CICS リソースおよび IMS リソースが同期にコミットされます。

トランザクションが正常に完了した場合には、アプリケーション・プログラムはコマンド **FREE** または **RETURN EXEC** のいずれかを使用してそのセッションを解放しなければなりません。

### 関連概念

[652 ページの『リカバリーおよび再始動の概念』](#)

このトピックでは、セッション、システム、またはアプリケーションの障害の後で ISC セッションを回復するために実行しなければならないシステム機能およびユーザー機能について説明されており、読者がセッション再同期での STSN コマンドの役割を理解していることを前提としています。

### 関連資料

[645 ページの『CICS のための機能管理ヘッダーのコーディング』](#)

CICS では、IMS によって使用されるのと同じ SNA 定義機能管理ヘッダー・フィールドのいくつかが使用されます。

## SEND LAST EXEC コマンドを使用して CICS から IMS へ

CICS がフロントエンド・サブシステムで IMS がバックエンド・サブシステムであるシステムでは、IMS は IMS 非同期トランザクションへのアクセスをできるようにするために CICS 同期 API の使用をサポートします。SEND LAST は、IMS 非応答モードと非会話型トランザクション、メッセージ通信、および /DISPLAY、/RDISPLAY および /FORMAT などのコマンドを処理するために使用されます。

SEND LAST を使用した CICS から IMS への一般的な CICS アプリケーション・プログラムのフローを、次の図に示します。

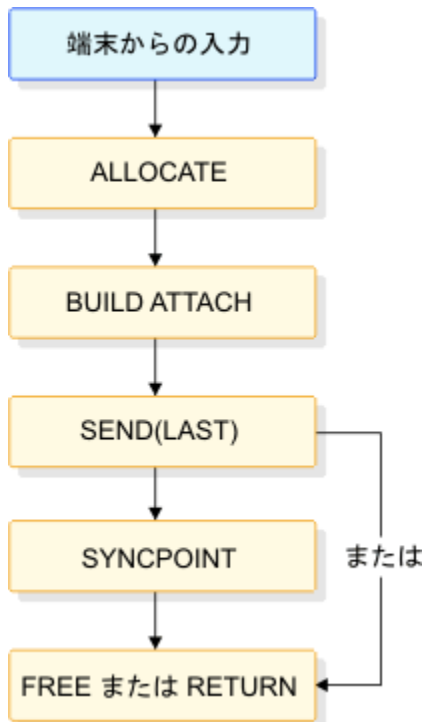


図 85. SEND LAST を使用した CICS から IMS へのアプリケーション・プログラムのフロー

この例では、CICS アプリケーションは端末から入力メッセージを読み取り、ALLOCATE コマンドを使用して CICS と IMS 間のセッションを設定します。

CICS アプリケーション・プログラムは、BUILD ATTACH EXEC コマンドを使用して、必須 ATTACH 機能管理ヘッダー (これは同期アプリケーションなので、唯一の必須ヘッダーです) を構築します。応答を処理するために使用される端末およびトランザクションを識別するために、RPROCESS フィールドおよび RRESOURCE フィールドを BUILD ATTACH に指定しなければなりません。SEND コマンドを発行すると、出力メッセージが構成されます。SEND LAST によって、ブラケット終了がメッセージに追加されます。

出力メッセージは、次のコマンドが実行されるまでセッション上に送信されません。発行される次の EXEC コマンドは、当該のトランザクションが IMS 内部でどのように定義されているかによって異なります。なぜなら、このコマンドは、メッセージへの同期点要求の付加と ISC セッションでのメッセージの発行の両方を行わなければならないからです。このトランザクションが IMS に対してリカバリー可能と定義されている場合には、次のコマンドは SYNCPOINT でなければなりません。SYNCPOINT によって、アウトバウンド・メッセージは RQD2 で発行されます。トランザクションがリカバリー不能と定義されている場合は、次に発行できるコマンドは SYNCPOINT、FREE、または RETURN のいずれかです。SEND の直後に (SYNCPOINT を挟まずに) FREE または RETURN を発行すると、RQE1 を付加されたアウトバウンド・メッセージが発行されます。

このトランザクションを受信した結果として IMS が応答メッセージを生成した場合には、IMS から戻されるその応答には ATTACH FM ヘッダーが付加されています。セッションから IMS 応答メッセージを入手するためには、CICS は RECEIVE を発行しなければなりません (応答メッセージは ATTACH だけを繰り返すからです)。

#### 関連資料

645 ページの『CICS のための機能管理ヘッダーのコーディング』

CICS では、IMS によって使用されるのと同じ SNA 定義機能管理ヘッダー・フィールドのいくつかが使用されます。

## RECEIVE EXEC コマンドを使用して IMS から CICS へ

IMS に、SEND LAST (ATTACH BB/EB) を使用して CICS によって送信されたメッセージから生じた CICS に送信される応答が入っているときには、IMS は ATTACH 機能管理ヘッダーと BB/EB を使用してそのメッセージを送信します。セッションからメッセージを受信するためには、CICS は、この目的のために

RECEIVE EXEC コマンドを使用する新規のトランザクション (または元のトランザクションの新しいインスタンス) を開始しなければなりません。

このための一般的な CICS アプリケーション・プログラムのフローを、次の図に示します。

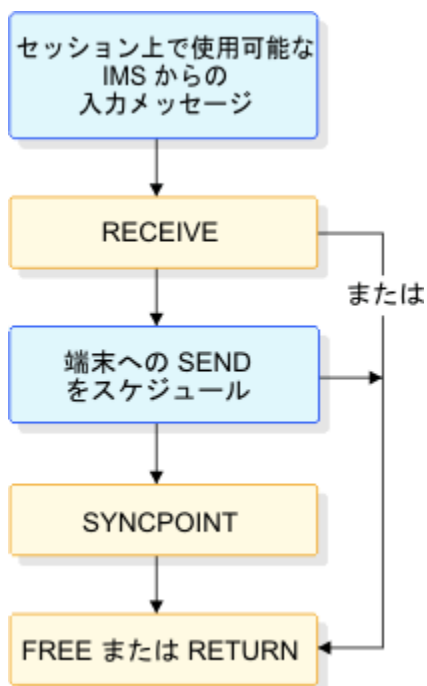


図 86. IMS から CICS への RECEIVE を使用したアプリケーション・プログラムのフロー

IMS からの応答は、(ATTACH FM ヘッダーを使用して) セッション上に同期に戻され、BB/EB を繰り返します。リカバリー可能な応答は RQD2 を繰り返す、リカバリー不能な応答は RQD1 を繰り返します。IMS に送信された RPROCESS (RDPN) および RRESOURCE (RPRN) フィールドは、自動的に IMS によってアウトバウンド ATTACH FMH の DPN フィールドおよび PRN フィールドに折り返されます。

戻された応答を処理することができるトランザクションを判別するために、CICS は戻された応答を調べて DPN を使用するか、あるいはデータの最初の 4 文字を使用します。RECEIVE はセッションから IMS 応答を入手するために使用されます。PRN フィールドは開始されたトランザクションに渡され、戻された応答が送信される端末を示します。

RECEIVE を発行した後で、アプリケーション・プログラムは、追加処理を実行する前に EXEC インターフェース・ブロック (EIB) の値を保管し、検査しなければなりません。

戻された応答を端末へ送信するために、CICS は START コマンドを発行して非同期トランザクションをスケジュールしなければなりません。開始されるトランザクションが構成され、応答の送信先となる端末を所有します。この非同期トランザクションは、端末と通信するために START インターフェースを使用し、このトランザクション内のすべての処理は、これを開始したトランザクションに対して非同期に行われます。端末に出力を送信するトランザクションが正常にスケジュールされた場合には、発信元アプリケーションは、必要に応じて同期点 (SYNCPOINT) を発行します。この同期点を発行すると、該当する応答が IMS トランザクション応答に戻されます。トランザクションが正常にスケジュールされなかった場合には、この同期点を発行すると、該当する例外応答が IMS トランザクション応答に戻されます。IMS 応答メッセージの内容に基づいて CICS がアプリケーション・プロセスを実行する必要がある場合には、その処理は同期点が発行される前に実行されます。これによって、確実に CICS リソースおよび IMS リソースが同期にコミットされます。

処理が正常に完了した場合には、アプリケーション・プログラムは EXEC コマンド FREE または RETURN のいずれかを使用してそのセッションを解放しなければなりません。

## 非同期メッセージのコーディング

IMS から (ATTACH および SCHEDULER 機能管理ヘッダーで送信された) 非同期メッセージを受信したときには、CICS は、セッションからこれらの着信メッセージを入手するために CICS 提供のミラー・トランザクションを呼び出します。

ミラー・トランザクションは、SCHEDULER プロセスのための CICS の名前です。このミラー・トランザクションは着信データ・ストリームを調べ、(START を使用して) 着信 DPN の結果として開始されるトランザクションまたはデータの最初の 4 文字をスケジュールします。受信 CICS トランザクションは、次の 1 つと見なされます。

- ISC 編集 (ISCE)、IMS によって設定されたデフォルトの DPN
- 着信 RDPN (CICS START 上に指定された RTRANSID) の IMS の折り返しによって ID が ATTACH FM ヘッダーの DPN フィールドに入れられたトランザクション
- その ID を MFS MOD によって提供されたトランザクション

CICS は、着信メッセージの PRN を START コマンドの TERMID パラメーターとして使用します。これにより、このメッセージの書き込み先となる CICS 端末 (適用可能な場合) が START コマンドで開始されるトランザクションによって所有されます。

IMS に該当する応答を戻し、非同期トランザクションのスケジュールリングを完了するために、ここで SYNCPOINT が発行されます。同期点に続き、トランザクションを終了してセッションを解放するために RETURN コマンドが発行されます。IMS に戻された応答によって、出力非同期メッセージは IMS 出力キューからデキューされます。

### START/RETRIEVE EXEC コマンドを使用して CICS から IMS へ

CICS は、**START/RETRIEVE** コマンドを使用して、IMS 非応答モードおよび非会話型トランザクション、メッセージ通信、および **/DISPLAY**、**/RDISPLAY**、および **/FORMAT** コマンドを処理します。

次の図に、CICS から IMS への非同期メッセージの一般的な CICS アプリケーション・プログラムのフローを示します。

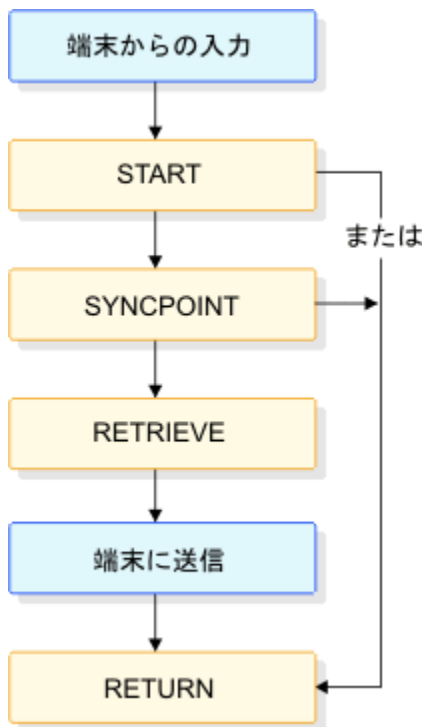


図 87. START/RETRIEVE を使用した CICS から IMS へのアプリケーション・プログラムのフロー

端末からの入力メッセージの受信後に、トランザクションが **START** コマンドを使用してアセンブルされます。ATTACH および SCHEDULER 機能管理ヘッダー DPN、PRN、RDPN、および RPRN フィールドに含まれる値は、**START** コマンドでパラメーターとして指定されます。開始される編集および着信メッセージの宛先を判別するために、DPN および PRN は IMS によって使用されます。RDPN および RPRN は、応答が戻されるべき CICS 端末および CICS トランザクションを識別するために使用されます。**START** コマンドはまた、これが IMS リカバリー不能トランザクションの場合に **NOCHECK** を指定し、IMS リカバリー可能トランザクションの場合に **NOCHECK PROTECT** を指定しなければなりません。(PROTECT は、リカバリー不能トランザクションの場合にオプションで使用することができます。)

**NOCHECK** は、IMS を使用した ISC セッションの場合には必須パラメーターです。CICS-CICS ISC セッションでは、**START** を使用することによって、戻りコード応答が受信 CICS から開始 CICS へ送信されます。この戻りコードは、受信 CICS がそれが受信したメッセージの結果実行されるトランザクションをスケジュールしたことを示します。この戻りコードは、IMS-CICS セッションでは生成されません。**NOCHECK** は、そのような応答が実行されないことを送信 CICS に通知します。**START NOCHECK** は、BB/EB RQE1 を繰り上げるメッセージを作成します。

リカバリー可能トランザクションに指定された **PROTECT** は、CICS が正常に SYNCPOINT を取るまでセッション上のトランザクションの送信を遅らせます。**START** を指定すると、開始ブラケットとブラケット終了 (BB/EB) および 機能管理ヘッダー ATTACH と SCHEDULER がアウトバウンド・メッセージに付加されます。**NOCHECK PROTECT** は確定応答 (RQD2) を要求します。

該当する同期点プロトコルを付加してメッセージをセッションで送信するために、ここで **SYNCPOINT** コマンドが発行されます。このトランザクションが応答の待機を選択した場合には、これは **RETRIEVE** を発行します。このトランザクションが応答の待機を選択しなかった場合には、**START** コマンドまたは **SYNCPOINT** コマンドの後に **RETURN** を発行してこれを終了することができます。応答を受信すると、CICS ミラー・トランザクションはこのトランザクションの新しいインスタンスをスケジュールします。

トランザクションが応答の待機を選択した場合には、**WAIT** パラメーターを使用して、あるいは使用しないで **RETRIEVE** を発行することができます。**WAIT** パラメーターを使用せずに **RETRIEVE** を発行すると、CICS はこのトランザクションと端末のキュー非同期入力を検査します。このような入力が即時に使用可能な場合には、これは **RETRIEVE** コマンドの条件を満たしています。このような即時に使用可能な入力がない場合には、該当する指示が **RETRIEVE** コマンドに戻されます。その後では、アプリケーションは他の処理を実行するか、**RETURN** を発行して終了することができます。

**RETRIEVE WAIT** が発行された場合には、このトランザクションおよびこの端末に予定された非同期メッセージが IMS から送信されるまで、CICS は制御をアプリケーション・プログラムに戻しません。**WAIT** を使用すると、端末は IMS 応答が受信されるまで保留されます。

上記のいずれの場合 (**RETRIEVE** コマンドに **WAIT** パラメーターを指定した場合と指定しなかった場合) においても、非同期処理を要求してメッセージを CICS から IMS へ送信したときには、出力応答のタイミグまたは他の出力の可用性に関し、発信元アプリケーションが前提事項とすることができるものは何もありません。つまり、CICS に対して保留されている非送信請求非同期出力がある場合には、この非同期トランザクションへの応答が戻される前に、この出力が CICS に送信される可能性があるということです。したがって、非同期環境の CICS 内部では要求と応答は関連しません。**RETRIEVE** を発行すると、指示されたトランザクションと端末識別名をもつあらゆるセッション上の不特定の IMS 応答が入手されます。



## RETRIEVE EXEC コマンドを使用して IMS から CICS へ

次の図は、IMS から CICS までの、非同期モードでの一般的な CICS アプリケーション・プログラムのフローを示しています。

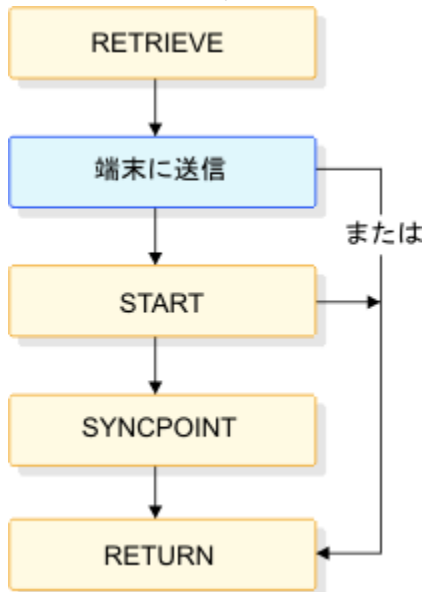


図 88. IMS から CICS への RETRIEVE を使用したアプリケーション・プログラムのフロー

直前の非同期入力トランザクションの結果として生成される応答および IMS フロントエンドから CICS バックエンドへの非送信請求非同期出力は、CICS に対して非同期に (すなわち、ATTACH FM ヘッダーおよび SCHEDULER FM ヘッダーを付加されて) 送信されます。IMS がフロントエンド・サブシステム有的时候には、非同期モードはサポートされた唯一のフローです。CICS ミラー・トランザクションは入力を入力し、着信メッセージを分析し、(START を使用して) 着信 DPN の結果として開始されるトランザクションまたはデータの最初の 4 文字をスケジュールします。このトランザクションが端末に対して出力を送信しようとしている場合には、ミラー・トランザクションは TERMID パラメーター (この値として着信 FMH の PRN パラメーターの値を使用) を指定した START を発行します。これにより、出力の送信先となる端末は、スケジュールされる非同期トランザクションによって所有されます。

IMS に対する応答が必要な場合には、CICS アプリケーションは着信 RTERMID および RTRANSID を後続の START で使用される TERMID フィールドおよび TRANSID フィールドに折り返さなければなりません。これらのフィールドは、それぞれ、IMS 内部の宛先 LTERM またはトランザクション、およびメッセージを受信する IMS エディターまたは MFS MID を示しています。この応答を IMS へ送信するための START コマンドが処理されます。

IMS に対する応答が必要でない場合には、RETURN を発行してセッションを終了することができます。

### 関連概念

626 ページの『START/RETRIEVE EXEC コマンドを使用して CICS から IMS へ』

CICS は、**START/RETRIEVE** コマンドを使用して、IMS 非応答モードおよび非会話型トランザクション、メッセージ通信、および **/DISPLAY**、**/RDISPLAY**、および **/FORMAT** コマンドを処理します。

### 関連資料

645 ページの『CICS のための機能管理ヘッダーのコーディング』

CICS では、IMS によって使用されるのと同じ SNA 定義機能管理ヘッダー・フィールドのいくつかが使用されます。

## ISC セッション上で使用してはならないコマンド

ISC セッションでは、**WAIT SIGNAL** コマンドまたは **WAIT TERMINAL** コマンドを使用しないでください。

**WAIT SIGNAL** コマンドは CICS-IMS セッションで使用することはできません。



WAIT TERMINAL コマンドは通常、CICS-IMS ISC セッションで使用されません。

## ISC のための該当する CICS インストール・オプションの選択

このトピックでは、IMS-CICS ISC 環境に関して固有の考慮事項のあるシステム定義オプションおよびリソース定義オプションについてのみ説明しています。

### このタスクについて

CICS インストール・プロセス時にコード化されたパラメーターの一部は、IMS システム定義マクロ上でコード化された IMS システム定義パラメーターの一部と互換性がなければなりません。630 ページの『[互換性のある IMS ノードと CICS ノードの定義](#)』のトピックでは、ここで提供されている定義と 491 ページの『[IMS への ISC ノードの静的定義](#)』で提供されている IMS システム定義情報が一緒に掲載されています。

## CICS システム定義オプションのコーディング

配布テープで提供される CICS 標準事前生成システムには、必須のシステム定義オプションをすべて含むシステム間連絡環境をサポートするのに必要な各 CICS 管理モジュールの事前生成バージョンが入っています。

### 関連資料:

- システム定義については、「*CICS Transaction Server for z/OS CICS Intercommunication Guide*」を参照してください。この資料には、IMS-CICS システム間連絡機能に参加する CICS サブシステムを生成するのに必要な仕様について記載されています。
- リカバリー・オプションのチェックリストについては、「*CICS Transaction Server for z/OS Recovery and Restart Guide*」を参照してください。

## CICS リソース定義の準備

CICS ユーザー作成テーブルとオンライン・リソース定義 (RDO) には、データベースとデータの通信環境およびこの環境内のエレメントに対する処理が記述されています。

これらには、端末、ファイルとデータベース、プログラム、トランザクション、一時データ宛先、および一時記憶域データ ID についての情報が入っています。これらはシステム定義とは独立して作成されますが、当該システムを操作可能にするにはこれらが存在していなければなりません。CICS のシステム・ログおよび動的ログは、一部の CICS 環境ではオプションですが、IMS-CICS ISC の場合は必須です。

### 関連資料:

- テーブルの準備および RDO オプションについて詳しくは、「*CICS Transaction Server for z/OS CICS リソース定義ガイド*」を参照してください。
- IMS-CICS システム間連絡機能に必要なパラメーターの要約については、「*CICS Transaction Server for z/OS CICS Intercommunication Guide*」を参照してください。
- リカバリー関連のオプションのチェックリストについては、「*CICS Transaction Server for z/OS Recovery and Restart Guide*」を参照してください。

## IMS-CICS LU 6.1 リンクの定義

IMS システムに CICS をリンクするには、接続(またはシステム)の定義または各セッションの別々の定義が必要です。

IMS に対する ISC リンクの定義には、オンライン・リソース定義またはマクロ・レベル・リソース定義を使用することができます。IMS-CICS ISC リンクは、LU 6.1 プロトコルを使用して実行することができます。

次の表に、個別の LU 6.1 セッションに対する RDO 定義の書式、およびオペランドを関連させるマクロの書式が示されています。

ユーザーが定義する各セッションごとに TRMTYPE、TRMIDNT、SYSIDNT、NETNAMQ、および SESTYPE などのオペランドをコーディングしなければなりません。定義されたすべてのセッションにデフォルトを提供するために、TYPE=SYSTEM マクロ上で TYPE=TERMINAL の残りのオペランドを任意にコーディングすることができます。さらに、必要な場合には、個別のセッションに対して、TYPE=SYSTEM の CONNECT、DATASTR、および RECFM オペランドもコーディングすることができます。

表 126. 個別のセッションを使用した LU 6.1 リンクの定義

RDO 定義	マクロ・レベルの定義
<pre>DEFINE CONNECTION(sysidnt) GROUP(groupname) NETNAME(name) ACCESSMETHOD(VTAM) PROTOCOL(LU61) DATASTREAM(USER 3270             SCS STRFIELD             LMS) RECORDFORMAT(U VB) AUTOCONNECT(NO YES) SECURITYNAME(name) INSERVICE(YES)</pre>	<pre>DFHTCT TYPE=SYSTEM ,SYSIDNT=sysidnt) ,NETNAME=name ,ACCMETH=VTAM) ,DATASTR=( {USER 3270             SCS STRFIELD             LMS} ) ,RECFM={U VB} [,CONNECT=AUTO ALL] ,XSNAME=name</pre>
<p>その後で、各個別のセッションは以下のとおり定義されます</p> <pre>DEFINE SESSIONS(csdname) GROUP(groupname) SESSNAME(name) CONNECTION(sysidnt) NETNAMQ(name) PROTOCOL(LU61) SENDCOUNT(0 1) RECEIVECOUNT(1 0) SENDSIZE(size) RECEIVESIZE(size) BUILDCHAIN(N Y) OPERID(operator-id) OPERPRIORITY(number) OPERRSL(number) OPERSECURITY(number) IOAREALEN(value) SESSPRIORITY(number)</pre>	<p>その後で、各個別のセッションは以下のとおり定義されます</p> <pre>DFHTCT TYPE=TERMINAL ,TERMIDNT=name ,SYSIDNT=sysidnt ,NETNAMQ=name ,TRMTYPE=LUTYPE6 ,SESTYPE= SEND RECEIVE ,BUFFER=size ,RUSIZE=size ,CHNASSY={NO YES} ,OPERID=operator-id ,OPERPRI=number ,OPERRSL=number ,OPERSEC=number ,TIOAL=value ,TRMPRTY=number ,TRMSTAT=TRANSCEIVE</pre>

## 互換性のある IMS ノードと CICS ノードの定義

IMS-CICS ISC リンクを定義するためには、リモート・システムとセッションが CICS で定義される方法と、それらが IMS で定義される方法との関連を理解する必要があります。

互換性の要件に関する以下の説明では、オンライン・リソース定義 (RDO) 用語が使用されています。同等のマクロ・レベル・オペランドについては、629 ページの『IMS-CICS LU 6.1 リンクの定義』を参照してください。

### システム名

CICS システムのネットワーク名は、DFHSIT マクロの APPLID オペランドに指定されています。

この名前は、CICS の始動時、または DFHTCT TYPE=INITIAL マクロの APPLID オペランドの中でオーバーライドとして提供できます。この名前は、TERMINAL マクロの NAME オペランドで、または CICS システムを定義する ETO ログオン記述子で指定しなければなりません。IMS と CICS 間のセッションを参照する IMS マスター端末オペレーター・コマンド (例えば、/OPNDST) 内部に指定する必要があるときに、この名前は IMS マスター端末オペレーターに認識されていなければなりません。

IMS システムのネットワーク名は、IMS COMM マクロの APPLID オペランドに指定されています。IMS システムを定義する **DEFINE CONNECTION** コマンドの NETNAME オペランドにこの名前を指定しなければなりません。

## セッションの数

IMS に静的に定義された端末の場合には、CICS システムと IMS システム間で必要な並列セッションの数は、IMS TERMINAL マクロの SESSION オペランドに指定しなければなりません。

その後で、各セッションは IMS VTAMPOOL の SUBPOOL 項目によって表示されます。CICS では、これらの各セッションは個別のセッション定義によって表されます。

## セッション名

各 CICS-IMS セッションは、CICS セッション名と IMS セッション名の両方から構成されるセッション修飾子ペアによってそれぞれ固有に識別されます。

CICS セッション名は、**DEFINE SESSIONS** コマンドの SESSNAME オペランドで指定されます。IMS で開始する予定のセッションの場合、この名前はセッションの IMS /OPNDST コマンドの ID パラメーターに対応していなければなりません。CICS によって開始されたセッションの場合には、名前は CICS /OPNDST コマンドで提供され、IMS によって保管されます。

IMS セッション名は、IMS SUBPOOL マクロの NAME オペランドに指定されています。対応する **DEFINE SESSIONS** コマンドの NETNAMEQ オペランドにこの名前をコーディングして、セッション名間の関係を明示しなければなりません。

**推奨事項:** 操作を簡便にするために、CICS セッション名と IMS セッション名に同じ名前を使用してください。

## 他のセッション・パラメーター

**DEFINE CONNECTION** コマンドと **DEFINE SESSIONS** コマンドの以下の追加オペランドも、CICS-IMS セッションにとって重要です。

### SENDSIZE

- CICS が基本ハーフセッションの場合、CICS がリモート IMS システムに送信する最大要求ユニット (RU) を指定します。この値は、IMS COMM マクロ内の RECANY パラメーターによって指定された値より小さいか等しくなければいけません。
- CICS が 2 次ハーフセッションの場合、CICS がリモート IMS システムから受信する最大 RU を指定します。この値は、IMS TERMINAL マクロ内の OUTBUF パラメーターによって指定された値より小さいか等しくなければいけません。

### RECEIVESIZE

- CICS が基本ハーフセッションの場合、CICS がリモート IMS システムから受信する最大要求ユニット (RU) を指定します。この値は、IMS TERMINAL マクロ内の OUTBUF パラメーターによって指定された値より小さいか等しくなければいけません。
- CICS が 2 次ハーフセッションの場合、CICS がリモート IMS システムへ送信する最大 RU を指定します。この値は、IMS COMM マクロ内の RECANY パラメーターによって指定された値より小さいか等しくなければいけません。

### BUILDCHAIN(N|Y)

複数の RU チェーンを、アプリケーション・プログラムに渡す前にアセンブルするかどうかを指定します。Y を指定すると、RECEIVE コマンド一つ一つに対する応答で完全なチェーンがアプリケーション・プログラムに渡されるため、アプリケーションは必要なブロック解除を実行しなければなりません。N を指定した場合は、各 RECEIVE コマンドに対する応答では単一 RU がアプリケーション・プログラムに渡されます。

**推奨事項:** IMS レコード・モード (VLVB) を使用している場合でも、BUILDCHAIN(Y) が推奨されます。その理由は、IMS 出力として作成された論理レコードが RU 境界と一致しないことがあるからです。

## DATASTREAM(USER)

CICS が START コマンド (非同期処理) を使用して IMS と通信するときに、データ・ストリーム・プロファイルを指定します。START コマンドによって生成される CICS メッセージによって常に、IMS はデータ・ストリーム・プロファイルをコンポーネント 1 の入力として解釈します。このパラメーターは必須です。

分散トランザクション処理用のデータ・ストリーム・プロファイルは、アプリケーション・プログラムで BUILD ATTACH コマンドの DATASTR オプションを使用して指定することができます。

## RECORDFORMAT(U|VB)

特定のセッションにおいて START コマンド (非同期処理) によって開始される伝送のために CICS が使用するチェーンのタイプを指定します。

2つのタイプのデータ処理アルゴリズムは、CICS と IMS 間でサポートされます。

- チェーン

メッセージは SNA チェーンとして送信されます。ユーザーは、専用のブロック化およびブロック解除のアルゴリズムを使用することができます。この形式は RECORDFORMAT (U) に対応しています。IMS が受信側であるときには、RU のチェーンは完了した単一セグメント・メッセージと解釈されます。

- 可変長可変ブロック・レコード (VLVB)

メッセージは、各レコードの前にハーフワードの長さフィールドの可変長可変ブロック形式で送信されます。この形式は、RECORDFORMAT (VB) に対応しています。IMS が受信側であるときには、入力非ブロック化エレメントは入力セグメントとして扱われます。

分散トランザクション処理用のデータ・ストリーム・フォーマットは、アプリケーション・プログラムで BUILD ATTACH コマンドの RECFM オプションを使用して指定することができます。

## SENDCOUNT および RECEIVECOUNT

セッションが SEND セッションであるか RECEIVE セッションであるかを指定するために、これらのオペランドが使用されます。(マクロ・レベルの定義では、これは SESTYPE=SEND|RECEIVE オペランドに指定されます。)

SEND セッションは、ローカル CICS が 2 次であり競合勝者であるセッションです。これは以下によって指定されます。

```
SENDCOUNT(1)
RECEIVECOUNT(0)
```

RECEIVE セッションは、ローカル CICS が基本であり競合敗者であるセッションです。これは以下によって指定されます。

```
SENDCOUNT(0)
RECEIVECOUNT(1)
```

**推奨事項:** SEND セッションは、すべての CICS-IMS セッションに推奨されます。

ユーザーは SENDPFX または RECEIVEPFX を指定する必要はありません。セッションの名前は SESSNAME オペランドから取られます。

次の表は、ISC リンクの CICS オンライン・リソース定義と IMS 定義の関係を示しています。関連したオペランドは番号で表示しています。CICS が XRF IMS と通信する場合、NETNAME(SYSIMS) がその XRF IMS と関連付けられている USERVAR または MNPS ACB でなければなりません。

表 127. 互換性のある CICS と IMS ノードの定義: RDO

CICS	IMS
DFHIST TYPE=CSECT ,SYSIDNT=CICL ,APPLID=SYSCICS <b>1</b>	COMM APPLID=SYSIMS <b>2</b> RECANY= <i>mmm</i> + 22 <b>7</b> EDTNAME=ISCEDT
DEFINE CONNECTION(IMSR) <b>3</b> GROUP( <i>group_name</i> ) NETNAME(SYSIMS) <b>2</b> ACCESSMETHOD(VTAM) PROTOCOL(LU61) DATASTREAM(USER)	TYPE UNITYPE=LUTYPE6 <b>4</b>
DEFINE SESSIONS( <i>csdname</i> ) GROUP( <i>group_name</i> ) PROTOCOL(LU61) <b>4</b> SESSNAME(IMS1) CONNECTION(IMSR) <b>3</b> NETNAMEQ(CIC1) <b>5</b> SENDcount(1) RECEIVECOUNT(0) SENDSIZE( <i>mmm</i> ) <b>7</b> RECEIVESIZE( <i>nnn</i> ) <b>8</b> IOAREALEN( <i>nnn</i> ,16364)	VTAMPOOL SUBPOOL NAME=CIC1 <b>5</b> NAME CICLT1 COMPT=1 NAME CICLT1A
DEFINE SESSIONS( <i>csdname</i> ) GROUP( <i>group_name</i> ) PROTOCOL(LU61) <b>4</b> SESSNAME(IMS2) CONNECTION(IMSR) <b>3</b> NETNAMEQ(CIC2) <b>9</b> SENDcount(1) RECEIVECOUNT(0) SENDSIZE( <i>mmm</i> ) <b>7</b> RECEIVESIZE( <i>nnn</i> ) <b>8</b> IOAREALEN( <i>nnn</i> ,16364)	SUBPOOL NAME=CIC2 <b>9</b> NAME CICLT2 COMPT=2

次の表は、ISC リンクの CICS マクロ・レベル定義と IMS 定義の関係を示しています。関連したオペランドは番号で表示しています。CICS を XRF IMS セッションに対して定義する場合、NETNAME=SYSIMS は、IMS システムと関連付けられた USERVAR または MNPS ACB でなければなりません。

表 128. 互換性のある CICS と IMS ノードの定義: マクロ・レベル

CICS	IMS
DFHIST TYPE=CSECT ,SYSIDNT=CICL ,APPLID=SYSCICS <b>1</b>	COMM APPLID=SYSIMS <b>2</b> RECANY=mmm + 22 <b>7</b> EDTNAME=ISCEDT
DFHTCT TYPE=SYSTEM ,ACCMETH=VTAM ,SYSIDNT=IMSR <b>3</b> ,NETNAME=SYSIMS <b>2</b>	TYPE UNITYPE=LUTYPE6 <b>4</b>
DFHTCT TYPE=TERMINAL ,TRMTYPE=LUTYPE6 <b>4</b> ,TRMIDNT=IMS1 ,SYSIDNT=IMSR <b>3</b> ,NETNAMEQ=CIC1 <b>5</b> ,SESTYPE=SEND ,BUFFER=mmm <b>7</b> ,RUSIZE=nnn <b>8</b> ,TIOAL=(nnn,16364) ,DATASTR=USER	TERMINAL NAME=SYSCICS <b>1</b> SESSION=2 COMPT1= COMPT2= OUTBUF=nnn <b>8</b>
DFHTCT TYPE=TERMINAL ,TRMTYPE=LUTYPE6 <b>4</b> ,TRMIDNT=IMS2 ,SYSIDNT=IMSR <b>3</b> ,NETNAMEQ=CIC2 <b>9</b> ,SESTYPE=SEND ,BUFFER=mmm <b>7</b> ,RUSIZE=nnn <b>8</b> ,TIOAL=(nnn,16364) ,DATASTR=USER	VTAMPOOL SUBPOOL NAME=CIC1 <b>5</b> NAME CICLT1 COMPT=1 NAME CICLT1A
	SUBPOOL NAME=CIC2 <b>9</b> NAME CICLT2 COMPT=2

## 関連概念

473 ページの『LTERM ユーザー (サブプール) とコンポーネント』

IMS ユーザー・ブロックは、IMS システム定義時に SUBPOOL マクロによって定義される、もしくは ETO ユーザー記述子から動的に作成される IMS 論理端末 (LTERM) のセットです。

## IMS システムとの複数のリンクの定義

CICS システムと IMS システムとの間に複数のシステム間リンクを定義することができます。これは、複数の接続 (システム) を、それらと関連付けられているセッション定義 (NETNAME は同一だが SYSID はそれぞれ異なる) を使用して定義することによって行います。

### このタスクについて

すべてのシステム定義は同じ NETNAME に解決され、したがって同じ IMS システムに解決されますが、CICS 内で SYSID 名を使用することにより、指定された SYSIDNT をもつリンクからセッションが CICS によって割り振られます。

**推奨事項:** CICS システムと IMS システムとの間に定義するリンク (すなわち、セッションのグループ) の数は最大でも 3 つまでとし、この範囲内でいくつ定義するのかは、インストール先でのアプリケーション要件によって決めてください。

### 手順

- CICS 開始の分散トランザクション処理 (同期処理) のためのセッションのグループ  
SEND/RECEIVE インターフェースを使用する CICS アプリケーションは、セッションをリモート・システムに割り振るためにこのグループの SYSIDNT を使用することができます。セッションは、会話が終了するまで保留状態 (使用中) にされます。
- CICS 開始の非同期処理のためのセッションのグループ  
START コマンドを使用する CICS アプリケーションは、このグループの SYSIDNT に名前を指定することができます。CICS は、最初の「非使用中」セッションを使用して開始要求を送ります。

セッションが比較的短期間使用されるように、IMS は開始要求をキューイングするとすぐに CICS に肯定応答を送信します。したがって、グループの最初のセッションの使用頻度が最も高く、グループの最後のセッションに近づくにつれて使用頻度は低くなります。

• IMS 開始の非同期処理のためのセッションのグループ

このグループは、CICS 開始の非同期処理で発生することがある実行問題に対する解決の一部としても役立ちます。特定のセッションで START コマンドが送られた結果として開始された IMS トランザクションは、CICS に対して「応答」START コマンドを送るのに同じセッションを使用します。(2) で示した理由により、CICS START コマンドは最も使用頻度の高いセッションで送られた可能性が高く、また、CICS は競合勝者であるために、IMS からの応答がそのセッションを使用できる機会を待っている間に、バックログが作成される可能性があります。

しかし、IMS には、トランザクションがそのデフォルトの出力セッションを別のセッションで代替するための機能が用意されているため、この 3 番目のグループのセッションへの切り替えを行うことによって、バックログ問題を軽減することができます。

例

以下の一連の表は、IMS ノードでの 3 つのセッション・グループの定義例を提供します。これらの表には、CICS オンライン・リソース定義 (RDO) フォーマットと CICS マクロ・フォーマットの両方でのセッション・グループの定義例が含まれています。

表 129. CICS システム初期設定パラメーター

RDO 定義	マクロ・レベルの定義
DFHSIT TYPE=CSECT ,SYSIDNT=CICL ,APPLID=SYSCICS	DFHSIT TYPE=CSECT ,SYSIDNT=CICL ,APPLID=SYSCICS

表 130. CICS 開始分散トランザクション処理

RDO 定義	マクロ・レベルの定義
DEFINE CONNECTION(IMSA) ACCESSMETHOD(VTAM) NETNAME(SYSIMS)	DFHTCT TYPE=SYSTEM ,ACCMETH=VTAM ,SYSIDNT=IMSA ,NETNAME=SYSIMS
DEFINE SESSIONS(csdname) PROTOCOL(LU61) SESSNAME(IMS1) CONNECTION(IMSA) NETNAMEQ(DTP1)	DFHTCT TYPE=TERMINAL ,TRMTYPE=LUTYPE6 ,TRMIDNT=IMSA ,SYSIDNT=IMSA ,NETNAMQ=DTP1
DEFINE SESSIONS(csdname) ⋮	DFHTCT TYPE=TERMINAL ⋮

表 131. CICS 開始の非同期処理

RDO 定義	マクロ・レベルの定義
DEFINE CONNECTION(IMSB) ACCESSMETHOD(VTAM) NETNAME(SYSIMS)	DFHTCT TYPE=SYSTEM ,ACCMETH=VTAM ,SYSIDNT=IMSB ,NETNAME=SYSIMS
DEFINE SESSIONS(csdname) PROTOCOL(LU61) SESSNAME(IMS1) CONNECTION(IMSB) NETNAMEQ(ASP1)	DFHTCT TYPE=TERMINAL ,TRMTYPE=LUTYPE6 ,TRMIDNT=IMSA ,SYSIDNT=IMSA ,NETNAMQ=DTP1
DEFINE SESSIONS(csdname) ⋮	DFHTCT TYPE=TERMINAL ⋮



表 132. IMS 開始の非同期処理

RDO 定義	マクロ・レベルの定義
<pre>DEFINE CONNECTION(IMSC)   ACCESSMETHOD(VTAM)   NETNAME(SYSIMS)</pre>	<pre>DFHTCT TYPE=SYSTEM   ,ACCMETH=VTAM   ,SYSIDNT=IMSC   ,NETNAME=SYSIMS</pre>
<pre>DEFINE SESSIONS(csdname)   PROTOCOL(LU61)   SESSNAME(IMS1)   CONNECTION(IMSC)   NETNAMEQ(IST1)</pre>	<pre>DFHTCT TYPE=TERMINAL   ,TRMTYPE=LUTYPE6   ,TRMIDNT=IMS1   ,SYSIDNT=IMSC   ,NETNAMQ=IST1</pre>
<pre>DEFINE SESSIONS(csdname)   :</pre>	<pre>DFHTCT TYPE=TERMINAL   :</pre>

## IMS-CICS ISC の CICS トランザクションの定義

このトピックでは、IMS-CICS ISC に関与するトランザクションに固有の考慮事項について説明しています。

### このタスクについて

**関連資料:** CICS トランザクションの定義の完全な詳細については、「*CICS Transaction Server for z/OS CICS* リソース定義ガイド」を参照してください。

## CICS バックアウト未確定処理の定義

IMS に対する同期点要求の送信と肯定応答の受信の期間には、CICS はリモート・システムがコミットしたかどうかを認識しません。この期間は未確定期間として知られています。

### このタスクについて

未確定期間中の CICS 処理は、DEFINE TRANSACTION IN-DOUBT パラメーター (または DFHPCT DTB= パラメーター) によって制御されます。

ISC ネットワーク内部の IMS と CICS サブシステム間の整合性を確認するために、DEFINE TRANSACTION IN-DOUBT (WAIT) または DTB=(YES,WAIT) を指定しなければなりません。

## IMS に対する非同期通信用 CICS トランザクションの定義

DEFINE TRANSACTION REMOTENAME (または DFHPCT TYPE=REMOTE RMTNAME) は、CICS **START** コマンドの結果作成されたトランザクションによって参照されるすべての IMS 編集名または MID 名に必要です。

### このタスクについて

**START** コマンドで使用された 4 文字の CICS トランザクション ID は、IMS エディターの名前または MFS MID 名に変換されます。どちらの名前も最大 8 文字とすることができます。

## CICS からのセッションの開始および割り振り

CICS では、いくつかある方法の 1 つでセッションを開始できます。

### このタスクについて

- DEFINE CONNECTION マクロで AUTOCONNECT YES を (または DFHTCT TYPE=TERMINAL マクロで CONNECT=AUTO を) 使用することによって、セッションを明示的に開始することができる。CICS は、そ

の開始時に、AUTOCONNECT が指定されているすべてのセッションを設定しようと試みます。セッション開始が行われるためには、IMS は CICS の開始時にアクティブになっていなければなりません。

- マスター端末オペレーターは、次のコマンドを入力することによって、セッションを開始することができる。

```
CEMT SET TERMINAL(#####) ACQUIRED|COLDACQ
```

ここで、##### は DEFINE SESSIONS 上の SESSNAME または DFHTCT TYPE=TERMINAL 上の TRMIDNT です。

ACQUIRED が指定されている場合には、IMS との通常の再同期が試みられます。

COLDACQ が指定されている場合には、再同期は実行されません。

CICS ハーフセッションの状況が CEMT INQUIRE|SET TERMINAL コマンドの表示域に上書きされます。セッションが正常に開始された場合には、状況は REL から ACQ に変更されます。セッションを開始しようとする試みが正常に行われなかった場合には、エラー・メッセージが一時データ宛先 CSMT に書き込まれます。

- ALLOCATE コマンドを使用するアプリケーション・プログラムによって、セッションを暗黙的に開始することができる。ALLOCATE コマンドは、SEND/RECEIVE インターフェースでのみ使用されます。

**推奨事項:** このコマンドを ALLOCATE SYSID という書式で使用してください。その理由は、このように指定すると、CICS が使用可能なセッションを選択できるようになるからです。

即時に使用できるセッションがないときに、制御がアプリケーション・プログラムに戻される場合を以下に示します。

- この条件に対する HANDLE CONDITION が当該のアプリケーション・プログラムによって発行された場合。
- ALLOCATE コマンドに NOQUEUE が指定されている場合。

そうでない場合には、コマンドはキューに入れられます。

次の条件によって、セッションは "即時に使用できない" ようになります。

- 指定されたシステムに対するすべてのセッション (または指定されたセッション) が使用中である。
- 使用可能な唯一のセッションが結合されていない。
- 使用可能な唯一のセッションが競合敗者である。

- CICS 自動タスク開始 (ATI) によってセッションを開始することができる。

CICS のバインド・パラメーターがハードコーディング・モデルから構築されます。DEFINE CONNECTION/SESSION 上の RECEIVESIZE および SENDSIZE パラメーター (または DFHTCT TYPE=TERMINAL/SYSTEM 上の RUSIZE および BUFFER パラメーター) は、基本および 2 次 RU サイズを判別するために使用されます。

**関連資料:** ALLOCATE コマンドおよび後続の処理の詳細については、「*CICS Transaction Server for z/OS CICS Intercommunication Guide*」を参照してください。

## セッション開始の他の方法

CICS によって開始されないセッションは、IMS、VTAM オペレーター、Tivoli NetView for z/OS のいずれかによって開始できます。

## CICS からのセッションの終了

CICS によってセッションを終了することができるのは、マスター端末オペレーター・コマンドを使用する場合だけです。

### このタスクについて

マスター端末オペレーターは、CEMT コマンドを使用して次のように入力することにより、セッションを解放することができます。

```
CEMT SET TERMINAL (tttt) [RELEASED|OUTSERVICE]
```

この例では、*tttt* は、DEFINE SESSIONS 上の SESSNAME または DFHTCT TYPE=TERMINAL 上の TRMIDNT です。

マスター端末オペレーターが RELEASED を指定した場合には、セッションは、アクティブ・トランザクションのいずれかが完了してセッションがブラケット間状態になったときに終了します。

マスター端末オペレーターが OUTSERVICE を指定した場合には、同じくアクティブ・トランザクションが完了し、セッションがブラケット間状態のときにセッションは終了します。しかし、この場合には、セッションはサービス範囲外になり、マスター端末オペレーターがこれをサービス範囲内に戻すまで使用することはできません。

RELEASED が指定された場合には、これは IMS と CICS 間の正常終了を開始します。セッションは、CICS の視点からはウォーム・スタート状態になっているように見えたとしても、SBI/BIS フローの結果、実際にはコールド・スタート状態になっています。

OUTSERVICE が指定された場合には、セッションはウォーム・スタート状態になったままです。つまり、CICS は再始動時に再同期を要求するということです。セッションをコールド・スタート・モードで開始するためには、CICS マスター端末オペレーターは次のように指定する必要があります。

```
CEMT SET TERMINAL (tttt) COLDACQ.
```

RELEASED と OUTSERVICE を一緒に指定することができます。

**推奨事項:** CICS アプリケーション・プログラムで EXEC CICS DISCONNECT を発行すると、結果的にセッションを正常に終了できますが、この方法は通常のアプリケーションでは推奨されません。ただし、オペレーター制御タイプのアプリケーションの場合は、この機能を使用するように作成しても構いません。

セッションの終了に関連したメッセージは、一時データ宛先 CSMT に送信されます。

### 関連タスク

636 ページの『[CICS からのセッションの開始および割り振り](#)』

CICS では、いくつかある方法の 1 つでセッションを開始できます。

### 関連資料

554 ページの『[LU 6.1 \(SBI および BIS\) の対称セッション・シャットダウン](#)』

停止ブラケット開始 (SBI) とブラケット開始の停止 (BIS) という 2 つのデータ・フロー制御コマンドにより、対等レベルの LU 6.1 ハーフセッションが対称的かつ正常に終了することができます。

## ISC の CICS アプリケーションの設計

CICS は、ISC セッションで行わなければならないフローの多くが CICS アプリケーション・プログラムの制御下で行われる点で、IMS とは異なります。

CICS アプリケーションが行わなければならない作業を以下に示します。

- アウトバウンド ATTACH 機能管理ヘッダーの作成
- SCHEDULER 機能管理ヘッダー用のフィールドの提供 (必要な場合)
- インバウンド ATTACH 機能管理ヘッダーの検査およびそれらの内容の処理
- アプリケーション・プログラムの処理時における、適切な間隔での同期点の発行

CICS が生成するメッセージに関連する SNA ブラケット、SEND/RECEIVE、および同期点プロトコルは、アプリケーション・プログラムによって使用される EXEC コマンド (この EXEC コマンドは、同期の場合は SEND/RECEIVE で、非同期の場合は START/RETRIEVE です) と、アプリケーション・フロー中に同期点が発行されるタイミングに基づきます。

情報は、IMS システム・コードによるメッセージの編集ならびに経路指定のために、機能管理ヘッダー ATTACH および SCHEDULER のフィールド DPN、RDPN、PRN、および RPRN に挿入されます。この情報は、IMS メッセージ形式サービス (MFS) を使用する場合のみオーバーライド可能です。CICS の場合には、ATTACH および SCHEDULER 機能管理ヘッダーへの情報の挿入は、CICS アプリケーションの制御に基づいて実行されます。

## アプリケーション関連の概念

このトピックでは、CICS と IMS の両方のユーザーのために参照の共通のフレームを設定するための、アプリケーション関連のいくつかの概念について説明しています。

### サブシステム設計: 直接制御とキュー

CICS は直接制御サブシステムですが、IMS はキュー・サブシステムです。すなわち、CICS は端末から入力したデータを受け入れ、結果的にそのデータを処理するために該当するアプリケーション・プログラムを呼び出します。

端末と他のシステム・リソースは、アプリケーション・タスクがその処理を完了するまで呼び出されたアプリケーションによって所有されます。処理の結果発生する情報は、キューに入れられるのではなく主記憶域に保留されます。ISC の含意は、CICS アプリケーション・プログラムが、該当する SNA データ・フロー制御、同期点、および応答プロトコルの生成、およびシステム・サービスのほとんどを制御することに直接に関連しているということです。

それとは対照的に、IMS はキュー・サブシステムです。この場合には、すべての入出力トランザクションとメッセージ通信は、関連した IMS アプリケーションと端末に代わって IMS 制御領域によってキューに入れます。したがって、端末との間のメッセージのメッセージの入出力は、メッセージの処理とは非同期です。(しかし、メッセージが IMS に対してどう定義されているにより、処理が端末に対して同期しているように見えることがあります。例えば、応答モードのメッセージまたは会話は、同期しているように見える場合があります。) ISC の含意は、SNA プロトコルおよび多くのシステム・サービスが IMS システム・コードの制御のもとで処理されるということです。したがって、アプリケーション・プログラムがそれらを提供する必要はありません。

### ISC VTAM リンクでの同期処理と非同期処理

メッセージ伝送は、メッセージを入力する端末とメッセージを受信するサブシステムとの間で同期方式と非同期方式のどちらでも行うことができます。メッセージはまた、ISC セッションに関しては同期または非同期のいずれかとすることができます。

注: このトピックは、ISC VTAM リンクで送信されるメッセージに適用されます。ISC TCP/IP リンクに適用される情報については、607 ページの『TCP/IP を介する CICS との ISC 通信』を参照してください。

CICS アプリケーションの観点から、CICS SEND/RECEIVE EXEC コマンドを使用するとセッションでは同期処理が行われ、CICS START/RETRIEVE EXEC コマンドを使用すると処理は結果的に非同期となります。CICS がフロントエンド・サブシステム有的时候には、両方の処理タイプがサポートされます。一方、IMS がフロントエンド・サブシステム有的时候には、IMS フロントエンド切り替え出口ルーチンによって提供されるフロントエンド/バックエンド・システム使用状況に関する特別なサポートを使用しない限り、実行できるのは非同期処理だけです。

次の表は、CICS START/RETRIEVE および SEND/RECEIVE EXEC コマンドが CICS の観点からどのようにサポートされるのかを示しています。

表 133. SEND/RECEIVE コマンドと START/RETRIEVE コマンド

メッセージ・タイプ	CICS から IMS へ	IMS から CICS へ
SEND/RECEIVE (同期)	YES	YES
START/RETRIEVE (非同期)	YES	YES

次の表は、SEND/RECEIVE または START/RETRIEVE の使用時における CICS アプリケーション・プログラム・インターフェースの相違を示しています。

表 134. SEND/RECEIVE 用および START/RETRIEVE 用の CICS API

属性	SEND/RECEIVE	START/RETRIEVE
モード	応答あり (SEND INVITE) または 応答なし (SEND LAST) 同期	非同期
FMH	ATTACH	ATTACH SCHEDULER
名前の長さ	8 バイトの名前をサポートする	4 バイトの名前をサポートする <a href="#">641 ページの『1』</a>
方向変換 (CD)	サポートされる	CICS フロントエンドは CD を送信できません。COMPT=SINGLE2 または MULT2 の場合、IMS は CD を CICS に送信します。
1 メッセージに対して複数のチェーン (IUTYPE)	サポートされる <a href="#">641 ページの『2』</a>	サポートされない
IMS コンポーネント選択 (DATASTREAM)	サポートされる	入力の場合はサポートされない。 入力コンポーネント = 1; 出力コンポーネント = 任意
RECORDFORMAT: <a href="#">641 ページの『3』</a> 未定義 (U)	サポートされる <a href="#">641 ページの『4』</a>	サポートされない
RECORDFORMAT: <a href="#">641 ページの『3』</a> RU	サポートされる <a href="#">641 ページの『4』</a>	サポートされない
RECORDFORMAT: <a href="#">641 ページの『3』</a> VLVB	サポートされる	サポートされる
RECORDFORMAT: <a href="#">641 ページの『3』</a> CHAIN <a href="#">641 ページの『5』</a>	サポートされる	サポートされる
構築された FMH	アプリケーション・プログラム (BUILD ATTACH)	システム : EXEC START コマンドで指定された DPN、PRN、RDPN、RPRN <a href="#">641 ページの『6』</a>
FMH へのアクセス	はい (EXTRACT 接続)	提供されたパラメーターはどれも EXEC RETRIEVE コマンドで使用可能 <a href="#">641 ページの『6』</a>
ISC セッションとの関係	直接に関連した (ALLOCATED)	独立

表 134. SEND/RECEIVE 用および START/RECEIVE 用の CICS API (続き)

属性	SEND/RECEIVE	START/RETRIEVE
サポートされる IMS トランザクション・タイプ	応答モード 会話モード メッセージ通信 <a href="#">641 ページの『7』</a> コマンド 非応答、 非会話型 <a href="#">641 ページの『7』</a>	非応答モード 非会話 メッセージ通信 コマンド <a href="#">641 ページの『8』</a>
リカバリー可能入出力	サポートされる	サポートされる
リカバリー不能入出力	サポートされる	サポートされる
複数セグメントの入力	サポートされる	単一セグメントのみ
複数セグメント出力	サポートされる	単一セグメントのみ
IMS 編集	基本編集	基本編集
使用可能	ISC 編集	ISC 編集
MFS: ページング なし	サポートされる	サポートされる
MFS: 自動ページ化入力: 単一チェーン	サポートされる	サポートされる
MFS: 自動ページ化入力: 単一チェーン: 複数チェーン	サポートされる	サポートされない
MFS: 自動ページ化出力	サポートされる	サポートされない
MFS: 要求時ページ化出力	サポートされる	推奨されない

**注:**

1. メッセージ・テキストに名前を組み込むことができます。DEFINE TRANSACTION REMOTENAME (または DFHPCT TYPE=REMOTE) を使用して、DPN フィールドを 4 文字名から 8 文字名に変換することができます。
2. MFS 自動ページ化入力または出力メッセージを送信または受信するためにだけ使用されます。
3. ATTACH パラメーターは、CONNECTION 定義で定義されます。しかし、START/RETRIEVE を使用するアプリケーションでは、アプリケーション内部の正しいデータ形式を理解し、またそれを提供しなければなりません。
4. CICS によって予約済みとマークが付けられましたが、BUILD ATTACH で指定することができます。
5. IMS からのチェーン出力には、MFS の使用が必要です。
6. RDPN と RPRN は 4 文字名に限定されています。
7. 非応答、非会話式、およびメッセージ通信入力は、SEND LAST しか使用しません。CICS は、RECEIVE を使用して非応答および非会話式トランザクションに対する応答を獲得します。
8. /DIS、/RDIS、/FOR のみ。

**関連概念**

[471 ページの『ISC と IMS の実行モードの関係』](#)

「同期」と「非同期」という用語は、IMS と ISC では少し意味が異なります。以下のトピックでは、これらの実行モードの関係について説明します。

**関連資料**

[571 ページの『ATTACH FM ヘッダーの形式』](#)

ATTACH FM ヘッダーの形式が、以下の表に定義されています。

[フロントエンド切り替え出口ルーチン \(DFSFEJ0\) \(出口ルーチン\)](#)



## 基本および代替機能

CICS 同期 EXEC 端末管理コマンド (SEND、RECEIVE、CONVERSE など) の機能は、基本機能または代替機能に対して反対の作用をします。

CICS では、タスクの基本機能は、タスクが開始されたときにアプリケーション・プログラムに使用できる端末または ISC セッションです。代替機能は、アプリケーション・プログラムが必要とするときに割り振られる ISC セッションです。基本機能に対して発行されたコマンドは、SESSION (名) オプションを使用せずに発行されます。代替機能に対して発行されるコマンドは、このオプションを指定された上で発行されます。

CICS がフロントエンド・サブシステム有的时候には、ユーザー端末が基本機能です。IMS に割り振られたセッションは代替機能です。しかし、再始動トランザクションが ISC セッションから接続されたときには、セッションは基本機能です。再始動トランザクションには、ユーザー端末への直接アクセスはありません。

### 関連概念

652 ページの『リカバリーおよび再始動の概念』

このトピックでは、セッション、システム、またはアプリケーションの障害の後に ISC セッションを回復するために実行しなければならないシステム機能およびユーザー機能について説明されており、読者がセッション再同期での STSN コマンドの役割を理解していることを前提としています。

## CICS 対 IMS 会話モード

IMS では、会話モードはトランザクションの属性で、ここでは会話は、スクラッチパッド区域 (SPA) の転送、および端末とトランザクション・メッセージ・キュー間の関連したメッセージによって続行されます。

情報をトランザクション・メッセージ・キューへ転送し、また端末キューへ戻すと、そのたびに、結果的に同期点になり、リソースのコミットメントおよびロック状態のリソースの解放の原因になります。また、IMS 会話のすべてのステップを全部ブラケット内部で行わなければなりません。会話内の中間点において EB を送信することは一切できません。その理由は、EB を受信すると、IMS は出力メッセージを廃棄し、会話を終了し、さらに会話型異常終了出口を呼び出すからです。会話時に、1 次リソース名 (PRN) パラメーターが入力機能管理ヘッダーに提供された場合には、これはトランザクション・コードが SPA に接続されるので無視されます。

CICS では、会話は端末とアプリケーション間のブラケット内部の一連の対話です。アプリケーションの終了時に、ブラケット終了が起こります。CICS はまた、端末とアプリケーション間の一連のトランザクションである「疑似会話」を許可します。各トランザクションには、それ自身のブラケット内部に端末入力が 1 つと端末応答が 1 つ入っています。これらの会話タイプのどちらの場合でも、要求と応答の合間に CONVERSE EXEC コマンドを使用することによって ISC セッションで IMS と対話することができます。CICS によってブラケット終了を含むメッセージが端末に対して送信されると、それ以前に IMS がブラケット終了状況を応答時に送信していない場合は、IMS に対してもブラケット終了メッセージが ISC セッションで戻されます。

CICS と IMS で会話が定義される方法の違いには、次の ISC についての意味が含まれています。

- CICS トランザクションが CICS 検知の会話型である場合には、ISC セッションと IMS トランザクション処理で CICS に接続された端末だけを同期に保留することができます。
- ISC セッションにおいて単一ブラケット内部で入出力が 2 回以上発生する場合は、CICS トランザクションは会話型と定義されている IMS トランザクションと対話することができます。ただし、セッション障害が発生した場合には、その会話を再始動することができないことがあります。
- ISC セッションにおいて単一ブラケット内部で入出力が 1 回だけ発生する場合は、CICS トランザクションは応答モードと定義されている IMS トランザクションと対話することができます。CICS 疑似会話を構成する一連のトランザクションは、応答モードと定義されている一連の IMS トランザクションと 1 つの ISC セッションで対話することができます。

IMS 会話型モードでは、IMS が RQD2,EB を含む最後の応答を送信することによって会話を終了する必要があります。中間会話型メッセージに RQE2,CD が送信されます。CICS アプリケーションは、SYNCPOINT または RETURN を発行することにより、会話が正常に終了するように決めることができます。このようにすると、CICS は、IMS からの最後の会話型出力に対する応答として LUSTATUS RQD2,EB を送信することによって会話を終了できるようになります。この場合には、LUSTATUS は IMS によって送信された RQE2,CD に対する確認応答として振る舞い、その論理作業単位をクローズします。上記の RQD2,EB を受信



すると、IMS は DR2 で応答して、ブラケットをクローズし、出力メッセージをデキューし、さらに会話のリモート・プログラムによって正常に終了されたことを会話型異常終了出口ルーチンに通知します。

#### 関連概念

[652 ページの『リカバリーおよび再始動の概念』](#)

このトピックでは、セッション、システム、またはアプリケーションの障害の後で ISC セッションを回復するために実行しなければならないシステム機能およびユーザー機能について説明されており、読者がセッション再同期での STSN コマンドの役割を理解していることを前提としています。

## CICS からの IMS コマンドの送信

IMS コマンドは、それらのコマンドの使用を許可されたすべての端末から入力できます。

静的に定義されている端末の場合は、許可にリソース・アクセス管理機能 (RACF) または同等の製品が使用されます。ETO を使用して動的に定義される端末の場合、この許可の妥当性は RACF を使用して検査されます。

しかし、CICS アプリケーションが ISC セッションを通して IMS コマンドを発行できると、IMS 依存要素が CICS アプリケーション・プログラムに紹介されます。この機能を使用する前に、この決定の結果について慎重に考慮する必要があります。

CICS アプリケーション・プログラムは、SEND/RECEIVE アプリケーション・プログラム・インターフェースを使用して IMS コマンドを発行することができます。

**例外:** CICS アプリケーション・プログラムは、IMS コマンドの **/DISPLAY**、**/RDISPLAY**、**/FORMAT**、および **/TEST** を発行できません。

IMS コマンド verb はメッセージに組み込まれています。これは、BUILD ATTACH コマンドの PROCESS パラメーターに指定することはできません。

**/DISPLAY**、**/RDISPLAY**、および **/FORMAT** は、START/RETRIEVE または SEND/RECEIVE を使用して送信することができます。これらのコマンドは、これらのコマンドに対する応答が即時に処理されるのではなく IMS によってキューに入れられる点で、他の IMS コマンドと異なっています。

これらのコマンドが SEND で (LAST ではなく) 発行された場合には、IMS は LUSTATUS NO-OP をブラケット終了を強制する返信として戻します。IMS は、ATTACH BB/EB を送信することによって、これらのコマンドに応答します。CICS は RECEIVE を使用して、応答を入手しなければなりません。コマンドが SEND LAST で発行された場合には、応答メッセージは、戻されたときに、名前が事前に BUILD ATTACH の RPROCESS フィールドに指定された CICS トランザクションによって処理されます。

コマンドが START/RETRIEVE を使用して発行された場合には、応答は ATTACH と SCHEDULER の両方の FM ヘッダーと一緒に戻されます。

IMS が CICS フロントエンドに対するバックエンドになっている場合は、ISC セッションにおける通信プロトコルのテストおよび機能の編集に **/TEST** コマンドを使用することができます。テスト・モードでは、SEND/RECEIVE アプリケーション・プログラム・インターフェースを使用する必要があります。

#### 関連概念

[470 ページの『ISC セッションからの IMS コマンドの発行』](#)

IMS オペレーター・コマンドは、ISC セッションにも使用できますが、本来は IMS マスター端末および IMS システムに直接接続されているリモート端末の、適切な許可を与えられたオペレーターが使用するよう意図したものです。

[63 ページの『拡張端末オプションの概説』](#)

IMS の拡張端末オプション (ETO) を使用すると、システム定義中に VTAM 端末と ISC TCP/IP 端末およびユーザーを事前定義せずに、それらを IMS に追加することができます。

[IMS セキュリティー \(システム管理\)](#)

#### 関連タスク

[470 ページの『ISC VTAM セッション用 IMS テスト・モードの使用』](#)

バックエンド IMS システムをテスト・モードにすることによって、ISC VTAM 通信プロトコルと編集機能をテストすることができます。

## 同期点

ISC セッション・プロトコルは、RQ\*2 要求およびそれらの関連した応答 (DR2 または例外 DR2) を同期点要求および応答として定義します。

これらの要求および応答は、機能的に RQ\*1 要求およびその関連応答 (DR1 および例外 DR1) から独立しています。その理由は、これら後者の要求と応答は同期点の原因にならないからです。同期点応答 (DR2) は、両方のパートナーのサブシステム同期点マネージャーが同期にリカバリー可能リソースをコミットまたはバックアウトできることを確認するために ISC セッション・パートナー間で使用されます。

ISC セッション上で送信または受信されたすべてのメッセージは、メッセージ・タイプによってリカバリー可能またはリカバリー不能と定義されます。セッション応答プロトコルは、ISC セッションの両方のパートナーがお互いに理解し合い、各メッセージと関連したリカバリー可能性属性に同意していることを確認するために使用されます。使用された応答プロトコルはメッセージ・タイプと一貫していなければならず、送信および受信サブシステム同期点マネージャーによって実施されます。

一方のセッション・パートナーがメッセージをコミットすると、そのことが他方のセッション・パートナーに肯定セッション同期点応答によって通知されます。一方のセッション・パートナーが入力または出力のプロセス時に検出されたエラーの結果メッセージをバックアウトしたときには、それは、セッション終了、受信サブシステムによって送信される例外応答、または送信サブシステムによって送信される LUSTATUS 機能打ち切りセンス・コードのいずれかによって、他方のセッション・パートナーに通知されます。バックアウトは、結果的に現在アクティブなメッセージを廃棄し、関連した ISC データ・フロー制御 (DFC) および ATTACH 状態を最後の同期点の状態にリセットします。

CICS は直接制御サブシステムなので、同期トランザクションは ISC セッションを直接に制御します。したがって、同期トランザクション処理時のバックアウトもまた、結果的に最後の同期点以降に行われたアプリケーションの更新をバックアウトします。

キュー・サブシステムおよび非同期アプリケーション出力は、トランザクションに ISC セッションの直接制御を提供しません。したがって、コミットまたはバックアウトは、ISC セッションでメッセージに対して非同期に行われるアプリケーションの更新に影響を与えません。IMS はキュー・サブシステムなので、セッション同期点はアプリケーション同期点から分離していて、常に実行アプリケーションの制御外で行われます。

## IMS 入力での同期点

どのような入力のタイプでも、完全な入力メッセージが正常に受信されるまでは、IMS は意図的なトランザクションをスケジュールしません。完全な入力メッセージを受信する前のプロセス・エラーおよびセッション障害によって、メッセージ全体が廃棄されるか、あるいはバックアウトされます。これはデータベースなどの他のリカバリー可能 IMS リソースに影響を与えません。ただし、完全なメッセージが受信され、エンキューされ、スケジュールリングに使用可能になった後では、ISC セッション障害またはプロトコルによって入力メッセージを取り消すことはできません。複数の連続入力 (リカバリー不能) メッセージが受信されて、エンキューされている場合であっても、直前の同期点及要求されたため、現在の入力メッセージだけがバックアウトされます。アプリケーション同期点に対する ISC 入力同期点の定義および関係は、その入力についての IMS の内部実行モードが同期であるか非同期であるかによって異なります。

CICS からの同期トランザクションは、CD 付き ATTACH を生成する SEND/RECEIVE を使用して送信されます。次の IMS トランザクション・タイプは同期です。

- 応答モード・トランザクション
- 会話型モード・トランザクション
- IMS コマンド
- テスト・モード入力

CICS からの非同期トランザクションは、EB 付き ATTACH を生成する SEND LAST、または EB 付き ATTACH SCHEDULER を生成する START/RETRIEVE を使用して送信されます。次のトランザクション・タイプは非同期です。

- 非応答モード・トランザクション
- 非会話モード・トランザクション
- IMS メッセージ通信

IMS の場合には、CICS に戻される同期点応答は、メッセージを正常に受信したことを示すこともあり、また IMS 処理の結果を示すこともあります。IMS に対する非同期入力の場合には、CICS へ戻された非同期点応答は、入力メッセージが正常にエンキューされていて、IMS にはメッセージ・リカバリーの責任があることのみを示します。IMS に対する同期入力の場合には、CICS へ戻された同期点応答は、トランザクションの実行がセッションから独立している場合であっても、トランザクションが正常に実行され、同期点がとられたことを示します。IMS は、アプリケーション同期点が正常にとられた結果としてアプリケーション挿入の応答メッセージが出力に使用可能になった後にだけ入力同期点応答を戻して、これを反映します。

## IMS 出力での同期点

出力の場合には、IMS は要求された同期点応答が CICS によって戻されたときに出力メッセージをコミットします。これは、メッセージが正常に送信およびデキューされ、セッション同期点情報が適切に更新されたことを意味します。(使用されたセンス・コードによって) IMS は、例外応答が CICS によって戻されたときに出力メッセージをバックアウトします。この場合、メッセージは再伝送するためにメッセージ・キューに戻され、関連した DFC 状態と接続状態は最後の ISC 同期点の状態にリセットされます。ISC セッションのバックアウトは、IMS アプリケーション・プログラム、またはデータベース更新などの他のリカバリー可能リソースに影響を与えません。ほとんどの場合、IMS 会話型モード出力のバックアウトが起こると、IMS 会話は終了し、会話型異常終了出口ルーチンが呼び出されます。会話型 SPA の内容に基づいて、ユーザー提供の出口論理が別の IMS トランザクションをスケジュールして、データベース変更をバックアウトすることがあります。

## CICS 同期点

CICS の場合には、同期点はアプリケーション・プログラムの制御に基づいて行われ、いつでも発行することができます。同期処理の場合には、同期点要求がアプリケーション・プログラムによって発行されたときに、CICS は論理作業単位の完了をログに記録します(すなわち、リソースがコミットされます)。さらに、SYNCPPOINT コマンドが CICS アプリケーション内部で発行された場合には、CICS には直前の処理の結果として作成された ISC メッセージを IMS が予期する SNA 応答要求 (RQD1 または RQD2) に付加する責任があります。しかし、IMS 入力の結果として CICS 非同期トランザクションがスケジュールされている場合は、ミラー・トランザクションによってすでに同期点が発行されており、これによって該当する応答が IMS に戻されます。アプリケーションによってその後が続いて発行された同期点は、ISC セッションに影響を与えません。

### 関連概念

520 ページの『[ハーフセッションの同期の維持](#)』

同期点応答 (DR2) は ISC セッション・パートナー間で使用され、両方のパートナーの同期点マネージャーが、リカバリー可能リソースを同期させてコミットまたはバックアウトできることを確認するために使用されます。

471 ページの『[ISC と IMS の実行モードの関係](#)』

「同期」と「非同期」という用語は、IMS と ISC では少し意味が異なります。以下のトピックでは、これらの実行モードの関係について説明します。

652 ページの『[論理作業単位](#)』

論理作業単位とは、同期点間で発生するあらゆる処理のことです。

## CICS のための機能管理ヘッダーのコーディング

CICS では、IMS によって使用されるのと同じ SNA 定義機能管理ヘッダー・フィールドのいくつかが使用されます。

IMS の場合とは異なり、CICS アプリケーション・プログラムは、これらの機能管理ヘッダーを作成して自分で IMS に送信する必要があります。CICS 自体はそれらを送信しません。

ISC セッションの両端には、FMH で要求された機能を実行する "接続マネージャー" が付いています。

## 関連資料

555 ページの『機能管理ヘッダー』

SNA では、機能管理 (FM) ヘッダーはリンクに対し送信される要求単位のオプションの部分です。このトピックでは、ISC セッション上の IMS によってサポートされる FM ヘッダーについて説明します。

## ATTACH 機能管理ヘッダー

このトピックでは、ATTACH 機能管理ヘッダーのフィールドについて説明しています。CICS が主として使用するフィールドは、ATTPDN、ATTPRN、ATTRDPN、および ATTRPRN です。

### ATTPDN

CICS トランザクションは、BUILD ATTACH コマンドの PROCESS フィールドにアウトバウンド宛先プロセス名フィールド (ATTPDN) を指定します。

CICS によって IMS に送信された場合は、このフィールドにはインバウンド・メッセージを受信する IMS エディター (基本編集または ISC 編集) の名前あるいは MFS MID 名が入っています。IMS からの応答に含まれる形で CICS によって受信された場合は、このフィールドには、CICS が発信元アウトバウンド・メッセージで戻り宛先プロセス名フィールド (ATTRDPN) として IMS に送信した値が入っています。これが SEND LAST を使用して IMS に送信された非応答または非会話式メッセージに対する応答である場合には、このフィールドはこの応答を処理するために開始されるトランザクションを識別します。その他の場合では、この値が再始動トランザクションを識別する CICS トランザクション・コードであることもあります。この値を使用することができない場合は、CICS は着信データ・ストリームの最初の 4 文字をトランザクション・コードとして使用します。着信メッセージを受信したセッションと同じセッションに応答が戻された場合には、IMS システム・コードは自動的に着信 RDPN フィールドを応答メッセージの DPN フィールドに折り返します。

IMS から戻された応答に入っている ATTPDN の場合は、受信した CICS はそれを無視します。

**例外:** セッション再始動の後、または SEND LAST で送信されたメッセージへの応答の後には、ATTPDN を使用して CICS トランザクションを接続します。

SCHEDULER で使用されたときには、ATTPDN には SCHEDULER モデルの DPN が含まれます。この場合には、IMS は残りの ATTACH パラメーターを送信しませんが、その代わりに、その情報はこれらのパラメーターと関連付けられている SCHEDULER モデルのパラメーターで提供されます。

## 関連資料

649 ページの『SCHEDULER 機能管理ヘッダー』

IMS と CICS との間で非同期メッセージを送信するために SCHEDULER 機能管理ヘッダーを使用することができます。

### ATTPRN

CICS トランザクションは、BUILD ATTACH コマンドの RESOURCE フィールドに 1 次リソース名フィールド (ATTPRN) を指定します。

CICS によって IMS に送信されたときには、このフィールドは IMS LTERM またはメッセージ宛先であるトランザクション・コードに名前を付けます。このパラメーターが省略されたときには、IMS はメッセージ宛先を識別するためにデータ・ストリームの最初の 8 文字を使用します。

**推奨事項:** IMS と CICS との間の整合性を維持するために、このフィールドを使用して IMS メッセージ通信のための LTERM に名前を付け、トランザクション・コードをデータ・ストリームの最初の 8 文字に入れてください。

IMS からの応答に含まれる形で CICS によって受信された場合は、このフィールドには CICS がアウトバウンド・メッセージで戻り 1 次リソース名 (ATTRPRN) として IMS に送信した値が入っています。IMS から戻された応答に入っている ATTPRN の場合は、受信した CICS はそれを無視します。

**例外:**

- セッション再始動の後では、ATTPRN は、当該の端末がこのトランザクションを発生させた端末と一致するかどうかを判別するために使用される。再始動の間に同一の端末が獲得される必要があります。

- 当該の応答が、CICS が SEND LAST を使用して発行した非応答型または非会話型のトランザクションに対する応答である場合は ATTPRN が使用される。

着信メッセージを受信したセッションと同じセッションに応答が戻される場合には、IMS は着信 RPRN フィールドに受信された値を PRN フィールドに自動的に折り返します。MFS はこの値を IMS 出力に設定またはオーバーライドするために使用することができます。

## ATTRDPN

CICS トランザクションは、BUILD ATTACH コマンドの RPROCESS フィールドに、戻り宛先プロセス名 (ATTRDPN) を指定します。

このフィールドを CICS が IMS に送信する場合は、このフィールドには、セッション再始動が必要となったとき、あるいは SEND LAST を使用して CICS が IMS に送信したメッセージに対して応答が戻されるときに実行されるトランザクション・コードが入っています。次の MID が MFS メッセージ出力記述子 (MOD) の内部に指定されたときには、IMS はこれを CICS に送信します。CICS が入力メッセージで ATTRDPN フィールドを受信した場合は、このフィールドは EXTRACT ATTACH コマンドによって調べることができます。この値は、保管して継続の応答があったときに DPN として IMS に戻さなければなりません。

## ATTRPRN

CICS トランザクションは、ATTRPRN フィールドに発信元端末の ID を入れることができます。

セッション再始動が必要な場合、あるいは SEND LAST を使用して CICS が IMS に送信したメッセージに対する応答が必要な場合には、この識別は IMS によって PRN フィールドに折り返されます。CICS トランザクションは、BUILD ATTACH の RRESOURCE フィールドを使用してフィールドにこの値を入れ、EXTRACT ATTACH コマンドを使用してこのフィールドを調べます。

非同期非送信請求出力については、応答が必要な場合は IMS はデフォルトとして送信元 LTERM 名を使用します。任意のタイプの出力でこの値を設定またはオーバーライドするために MFS を使用することができます。CICS アプリケーションがこの値を受信した場合は、この値を保管し、後続の応答の PRN フィールドとして戻す必要があります。

## ATTDQN および ATTDQ

要求時ページ化出力メッセージの ATTACH で IMS から CICS へ送信されたときは、ATTDQN に固有のメッセージ ID が含まれます。

このフィールドは、IMS 要求時ページングにだけ使用されます。CICS トランザクションは、戻された QMODEL FM ヘッダーの QNAME 値として後続の要求時ページ要求に対して宛先メッセージ ID (ATTDQN) を指定します。CICS が IMS から ATTDQN を受信したときには、ATTDQ フィールドは 1 に設定されます。これらのフィールドは、他の条件では IMS によって設定されず、また入力時に IMS によって受け入れられません。

## ATTIU

CICS トランザクションは、BUILD ATTACH コマンドの IUTYPE フィールドに交換単位コード (ATTIU) を指定します。

2 つの値を指定することができます。すなわち、すべての非 MFS 入力または出力に適用される単一チェーン、および MFS 自動ページ化入出力にだけ適用される複数チェーンです。IMS に対する入力が複数ページの MFS 自動ページ化入力でない限り、単一チェーンを使用しなければなりません。

## ATTDSP

CICS トランザクションは、IMS コンポーネントを判別するために BUILD ATTACH コマンドの DATASTR フィールドにデータ・ストリーム・プロファイル (ATTDSP) を指定します。

このフィールドに以下の値を入れることができます。

### X'00'

IMS コンポーネント 1 を示します



### X'01'

IMS コンポーネント 2 を示します

### X'02'

IMS コンポーネント 3 を示します

### X'03'

IMS コンポーネント 4 を示します

IMS によって受信されたときには、この入力コンポーネントは、デフォルト出力コンポーネント、入力セキュリティ要件、および入力に MFS を使用できるかどうかを判別します。出力コンポーネントは、MFS が出力に使用されるかどうか、および、非同期出力時に送信されるプロトコルが前の送信と同様に ATTACH SCHEDULER かどうかを判別します。出力時に、IMS はこの値を出力 LTERM の値に設定します。

## ATTDBA

CICS トランザクションは、BUILD ATTACH コマンドの RECFM フィールドにブロック解除アルゴリズム (ATTDBA) を指定します。

## ATTACC

ATTACC パラメーターは ISC セッションではサポートされず、送信された場合にはセッション終了の原因になります。

次の表では、ATTACH FM ヘッダーのフィールドに入れられる値の発信元が要約されています。

表 135. ATTACH FM ヘッダーのフィールドに入れられる値の発信元

主要フィールド	A <sup>648 ページの『1』</sup>	B <sup>648 ページの『2』</sup>	C <sup>648 ページの『3』</sup>
ATTDPN <sup>648 ページの『4』</sup>	PROCESS	DPN	
ATTPRN	RESOURCE	PRN	
ATTRDPN	RPROCESS	RDPN	
ATTRPRN	RRESOURCE	RPRN	
ATTDQN	QUEUE	メッセージ ID <sup>648 ページの『5』</sup>	
ATTIU	IUTUPE <sup>648 ページの『6』</sup>	ページング <sup>648 ページの『6』</sup>	
ATTDSP	DATASTR <sup>648 ページの『7』</sup>		COMPTn <sup>648 ページの『8』</sup>
ATTDBA	RECFM <sup>648 ページの『7』</sup>		VLVB
			DPM-Bn <sup>649 ページの『9』</sup>

### 注:

1. CICS EXEC BUILD/EXTRACT ATTACH
2. IMS MFS によって設定されます。あるいは、MFS が使用されず、出力が非同期非送信請求出力の場合には、RPRN のデフォルトの値として発信元 LTERM 名が使用されます。
3. TERMINAL/NAME マクロの IMS COMPTn
4. 非同期プロセスの場合は、このフィールドには値 X'02' が入り、それによって「SCHEDULER が次に送信される」ことが示されます。この場合には、残りのヘッダー・フィールドは ATTACH FM ヘッダーには含まれず、その代わりに、連結された SCHEDULER FM ヘッダーに含まれます。
5. MFS 要求時ページングのメッセージ ID
6. MFS 自動ページ化によって値が設定されます。
7. これらのハーフワード 2 進数フィールドの 2 番目のバイトが使用されます。
8. 1 から 4 まで番号が付けられたコンポーネント

9. DPM-BnによってMFSが使用されます。出力時にMFS MODE=STREAMを使用すると、ATTDBAがチェーン・モードに変更されます。これ以外の場合は、IMSはデフォルトVLVBを使用します。

## SCHEDULER 機能管理ヘッダー

IMSとCICSとの間で非同期メッセージを送信するためにSCHEDULER機能管理ヘッダーを使用することができます。

入力時にSCHEDULER FMHを受信すると、CICSはミラー・トランザクションを呼び出して、その入力を処理する非同期トランザクションをスケジュールします。出力時の場合は、CICSがSTARTコマンドを、以下に示すように、このヘッダーのフィールドと関連するフィールドと共に使用した結果としてこのヘッダーが生成されます。

### SCDDPN

CICSトランザクションは、STARTコマンドのTRANSIDフィールドにアウトバウンド宛先プロセス名フィールド(SCDDPN)に入れる値を指定します。

CICSによってIMSに送信された場合は、このフィールドにはインバウンド・メッセージを受信するIMSエディター(MFS、基本編集、またはISC編集)の名前が入っています。SYSIDがSTARTコマンドでコード化されない場合には、このTRANSIDは、プログラム管理テーブル(PCT)でRMTNAMEオプションの結果として(例えば、8文字の名前を入れること)変更されることがあります。

IMSからの応答に含まれる形でCICSによって受信された場合は、このフィールドには、CICSがアウトバウンド・メッセージで戻り宛先プロセス名フィールド(SCDRDPN)としてIMSに送信した値が入っています。IMSは自動的に着信RDPNフィールドを応答メッセージのDPNフィールドに折り返します。代わりに、IMS MFSがこのフィールドを設定した場合もあり得ます。この値はCICSトランザクション・コードです。このフィールドが入力時にCICSに提供されなかった場合には、CICSはトランザクション・コードとして入力データ・ストリームの最初の4文字を使用します。

### SCDPRN

CICSトランザクションは、STARTコマンドのTERMIDフィールドに1次リソース名フィールド(SCDPRN)を指定します。

CICSによってIMSに送信されたときには、このフィールドはIMS LTERMまたはメッセージ宛先であるトランザクション・コードに名前を付けます。CICSがIMSにメッセージを送信したときにこのフィールドが省略された場合には、IMSはメッセージ宛先を抽出するために着信メッセージの最初の8バイトを調べます。

**推奨事項:** IMSとCICSとの間の整合性を維持するために、このフィールドを使用してIMSメッセージ通信のためのLTERMに名前を付け、トランザクション・コードをデータ・ストリームの最初の8文字に入れてください。

IMSからの応答に含まれる形でCICSによって受信された場合は、このフィールドには、CICSがアウトバウンド・メッセージで1次リソース名(SCDRPN)としてIMSに送信した値またはMFSがこのフィールドに入れた値が入っています。このフィールドには、基本機能として使用される、応答の送信先となるべき端末を識別する値が含まれています。

### SCDRDPN

CICSトランザクションは、STARTコマンドのRTRANSIDフィールドに、戻り宛先プロセス名(SCDRDPN)を指定します。

CICSがこのフィールドをIMSに送信するときには、これには応答受信時にCICSミラー・トランザクションによってスケジュールされる非同期トランザクションの名前が入っています。

非送信請求非同期出力の場合には、後続の応答で使用される次のメッセージ入力記述子(MID)を示すために、MFSがRDPNを設定することができます。このパラメーターがCICSによって入力として受信された場合には、このパラメーターを保管して、STARTコマンドのTRANSID(DPN)中のIMSに戻さなければなりません。



## SCDRPRN

CICS トランザクションは、START コマンドの RTERMID フィールドに戻り 1 次リソース名 (SCDRPRN) を指定します。

CICS が SCDRPRN を IMS に送信するときには、これには戻される応答の送信先として経路指定されるべき端末の識別が入っています。IMS から CICS への出力時には、IMS は SCDRPRN に送信元の LTERM の名前を設定します。MFS も SCDRPRN を設定することができます。CICS は SCDRPRN の値を保管して、以後 IMS に対する応答の TERMID (PRN) には必ずこの値を使用するようにしなければなりません。

## SCDDQN と SCDDP

非同期要求時ページングは推奨ではないため、SCDDQN と SCDDP フィールドは CICS-IMS ISC セッションでは使用されません。

SCHEDULER ヘッダーの前に、ATTACH ヘッダーがきます。ATTACH FM ヘッダーの必須フィールドのいくつかは、非同期メッセージにも適用されます。これらのフィールドは、SCHEDULER ヘッダー上には置かれず、ATTACH ヘッダー上に保存されます。これらのフィールドは ATTIU、ATTDSP、および ATTDDBA フィールドです。

非同期環境でのこれらの値は以下のとおりです。

- ATTIU

CICS は、それ自体と IMS 間の単一チェーン・メッセージだけをサポートします。

- ATTDSP

START コマンドを使用して送信されたときには、ATTDSP の値は 端末管理テーブル生成によって判別され、X'00' (コンポーネント 1) でなければなりません。入出力の場合、IMS についての ATTDSP フィールドの定義は、646 ページの『ATTACH 機能管理ヘッダー』で説明されているものと同じです。

- ATTDDBA

CICS トランザクションは、BUILD ATTACH コマンドの RECFM フィールドにブロック解除アルゴリズム (ATTDBA) を指定します。

以下の表では、SCHEDULER FM ヘッダーのフィールドに入れられる値の発信元が要約されています。

主要フィールド	A <sup>651 ページの『1』</sup>	B <sup>651 ページの『2』</sup>	C <sup>651 ページの『3』</sup>	D <sup>651 ページの『4』</sup>
SCDDPN	TRANSID <sup>651 ページの『5』</sup>		DPN	
SCDRPRN	TERMID <sup>651 ページの『5』</sup>		PRN	
SCDRDPN	RTRANSID <sup>651 ページの『5』</sup>		RDPN	
SCDRPR	RTERMID <sup>651 ページの『5』</sup>		RPRN	
SCDDQN	QUEUE		注 651 ページの『6』	
ATTIU <sup>651 ページの『7』</sup>			注 651 ページの『6』	
ATTDSP <sup>651 ページの『7』</sup>		DATASTR <sup>651 ページの『8』</sup>		COMPTn <sup>651 ページの『8』</sup>
ATTDBA <sup>651 ページの『7』</sup>		RECFM		VLVB

表 136. SCHEDULER FM ヘッダーのフィールドに入れられる値の送信元 (続き)

主要フィールド	A 651 ページの『1』	B 651 ページの『2』	C 651 ページの『3』	D 651 ページの『4』
				DPM-Bn 651 ページの『9』

注:

1. CICS EXEC START/RETRIEVE
2. CICS DEFINE CONNECTION または DFHTCT TYPE=SYSTEM
3. IMS MFS によって設定されます。あるいは、MFS が使用されない場合には、非送信請求非同期出力である出力に関しては、RPRN のデフォルトの値として発信元 LTERM 名が使用されます。
4. TERMINAL/NAME マクロ中の IMS COMPTn または ETO ログオン記述子
5. 4 バイト名; MFS は 8 バイト名をサポートします。
6. MFS ページングは、IMS と CICS 間の非同期セッションでは推奨されません。
7. SCHEDULER FMH の前にある ATTACH FMH フィールド
8. DATASTR を USER として指定しなければなりません。これによって、X'00' が IMS コンポーネント 1 を指定して、入力コンポーネントとして指示されます。IMS 出力コンポーネントは、1 から 4 までのコンポーネント (X'00' から X'03') とすることができます。
9. DPM-Bn によって MFS が使用されます。出力時に MFS MODE=STREAM を使用すると、ATTDBA がチェーン・モードに変更されます。これ以外の場合は、IMS はデフォルト VLVB を使用します。

#### 関連資料

646 ページの『ATTACH 機能管理ヘッダー』

このトピックでは、ATTACH 機能管理ヘッダーのフィールドについて説明しています。CICS が主として使用するフィールドは、ATTPDN、ATTPRN、ATTRDPN、および ATTRPRN です。

## キュー・モデル機能管理ヘッダー

CICS は QMODEL ヘッダーを使用して、IMS MFS 要求時ページ化メッセージをアクセスします。

QMODEL ヘッダーは IMS によって CICS へ送信されません。SEND/RECEIVE の場合には、CICS アプリケーション・プログラムは、IMS から要求時ページ化出力を受信するために、QMODEL 機能管理ヘッダーを作成しなければなりません。START/RETRIEVE で一時記憶機能 SHIPPING を使用することができます。QMODEL FM ヘッダーのコーディングをサポートする EXEC コマンドは 1 つもありません。以下の QMODEL ヘッダーは、MFS 要求時ページングに使用されます。

#### QGET

直接ページを要求します

#### QGETN

次の順次ページを要求します

#### QPURGE

要求時ページングの終了を要求します

#### QXFR

要求されたページで送信します

#### QSTAT

QPURGE が受信されたとき、あるいは要求されたページ番号が無効のときに送信します

## データ記述子機能管理ヘッダー

IMS は、データ記述子機能管理ヘッダーを使用して、装置ページ (DPAGE) を指定します。

このヘッダーは CICS には使用されませんが、CICS トランザクションへ渡すことができます。CICS 同期トランザクションはデータ記述子 FM ヘッダーを構築し、これを該当する IMS に送信することができます。ただし、データ記述子 FM ヘッダーのコーディングをサポートする EXEC コマンドは 1 つもありません。

## システム・メッセージ・プロセス (SYSMSG) 機能管理ヘッダー

IMS は、システム・メッセージ・プロセス (SYSMSG) 機能管理ヘッダーを使用して、システム・メッセージを送信します。

システム・メッセージは、DPN=SYSMSG を繰り上げる ATTACH に追加されます。セッションがそのトランザクションに、またはそれ以外のときに宛先 CSMT に割り振られたときには、CICS はこれらのメッセージを同期アプリケーションに経路指定します。トランザクションに経路指定された場合には、そのトランザクションは SYSMSG 機能管理ヘッダーのプロセスに責任があります。CICS が SYSMSG を送信しませんが、BUILD ATTACH に該当するプロセス名を設定すると、同期アプリケーション・プログラムによって SYSMSG は送信されます。アプリケーション・プログラムは、ATTACH に追加される適切な SYSMSG FMH を構築することになります。

ブロードキャスト・メッセージの場合、または入力に対する応答が送信されたが、その後に処理エラーがあった場合に、SYSMSG は IMS によって送信されます。非同期入力に応答した後で、SYSMSG は応答メッセージの代わりに送信されます。同期処理時に、SYSMSG は以下のとおり送信されます。

- 現在行われているトランザクションがセッション故障時に異常終了した場合には、セッション再始動の後で
- MFS 出力形式ブロックが使用不能であるか、または無効の場合には、MFS 出力時に

## エラー・リカバリー・プロシージャ機能管理ヘッダー

CICS は、エラー・リカバリー・プロシージャ機能管理ヘッダーを送受信して、プロセス間でエラー情報を伝送します。

このヘッダーは、例外応答 (X'0846') の後で送信され、エラーの性質についてハーフセッション・パートナーに通知するために使用されるセンス・コードを送ります。この後にはエラー・メッセージが続きます。CICS は、受信された IMS エラー・メッセージを一時データ宛先 CSMT に書き込みます。IMS は、受信された CICS エラー・メッセージをマスター端末または入力発信元ノードに書き込みます。

## リカバリーおよび再始動の概念

このトピックでは、セッション、システム、またはアプリケーションの障害の後で ISC セッションを回復するために実行しなければならないシステム機能およびユーザー機能について説明されており、読者がセッション再同期での STSN コマンドの役割を理解していることを前提としています。

同期点、コミット、バックアウト、および論理作業単位（Logical Work Unit）の概念は、IMS と CICS に共通しています。しかし、それらの意味は IMS と CICS では少し異なっています。

### 関連概念

509 ページの『[STSN を使用するセッション同期の判別](#)』

再同期が必要な場合は、それをバインド交渉中に伝えることができます。再同期し、ハーフセッションをセッション終了時に存在した状態に戻すための要件を判別するために、BIND 要求の 2 つのフラグが使用されます。

644 ページの『[同期点](#)』

ISC セッション・プロトコルは、RQ\*2 要求およびそれらの関連した応答 (DR2 または例外 DR2) を同期点要求および応答として定義します。

## 論理作業単位

論理作業単位とは、同期点間で発生するあらゆる処理のことです。

アプリケーションが CICS などのセッションを直接に制御するサブシステムの内部では、アプリケーション同期点と ISC セッション同期点は、単一の明示的または暗黙的なアプリケーション・コマンドの結果として同時に起こります。IMS などのキュー・サブシステム内部では、アプリケーション処理と同期点は、ISC セッションのそれらから独立しています。IMS は、アプリケーション同期点を ISC セッション同期点要求にマップしなければなりません。

同期の場合 (SEND/RECEIVE) は、CICS がフロントエンド・サブシステムとなります。1 つの論理作業単位には、CICS アプリケーションが同期点を最後に発行した点から、次の同期点が発行され、さらに同期点応答が戻される点までの間に、IMS サブシステムと CICS サブシステムの両方で行われる処理のすべてが組み込まれています。

トランザクションがリカバリー可能な場合には、同期点が CICS アプリケーションによって要求されなければならない前に、メッセージを 1 つだけ IMS に送信することができます。しかし、CICS 同期点要求と IMS 応答の間に一連のリカバリー不能要求および応答が行われる可能性があります。

リカバリー可能メッセージの場合には、IMS がこの応答を戻すのは、IMS アプリケーションが応答または会話型モード応答メッセージを挿入して、アプリケーション同期点の原因になった後だけです。その後で、IMS は同期点を要求する応答メッセージを送信して、次の論理作業単位を開始します。応答メッセージは、CICS トランザクションによって発行された RECEIVE を満たしています。CICS トランザクションは、コマンド SYNCPOINT または RETURN (暗黙同期点) を発行する前に、必要なプロセスをすべて完了しなければなりません。

アプリケーション同期点によって、CICS は同期点応答を IMS に戻します。その後で、IMS は出力メッセージ・キューから応答メッセージをデキューして論理作業単位を完了します。

CICS がフロントエンド・サブシステムである 非同期の場合には、IMS に対するリカバリー可能入力に関して、入力要求と戻された応答メッセージは、分離した論理作業単位です。CICS アプリケーションが SEND LAST または START を発行して、同期点を要求したときには、IMS は、トランザクションがエンキューされた (スケジューリング使用可能になった) 後すぐに同期点応答を戻します。IMS は、トランザクションによって使用できるようになったときにリカバリー可能非同期応答メッセージを送信し、それぞれに関して同期点を要求します。応答メッセージが CICS ミラー・トランザクションによって読み取られたときには、該当する応答が自動的に IMS に戻されます。直前の SEND LAST の結果、応答メッセージが RECEIVE を使用して読み取られた場合には、同期点応答は、後続の SYNCPOINT、FREE、または RETURN コマンドが発行されるまで IMS に戻されません。

それぞれの後に RETRIEVE が続く複数の START コマンドを使用して、1 つ以上のリカバリー不能トランザクションを CICS 同期点と CICS 同期点との間に IMS に送信することができます。応答メッセージが CICS ミラー・トランザクションによって読み取られたときには、該当する応答が自動的に IMS に戻されます。

論理作業単位 (または作業単位) の概念は、セッション開始時および最大 1 つの作業単位が 1 次ハーフセッションから 2 次ハーフセッションへのフローまたは 2 次ハーフセッションから 1 次ハーフセッションへのフローで未確定であると見なされる場合には、セッション再同期を 実行しなければならないという点で重要です。未確定作業単位は、セッション BIND および STSN フローの結果に基づいてコミットまたはバックアウトを待機中の作業単位です。

## 障害後の未解決メッセージ・トラフィックのリカバリー

いくつかある状態のいずれかが発生した場合には、CICS と IMS とのセッションが失敗または終了する可能性があります。

CICS と IMS とのセッションが失敗または終了する可能性がある状態は、以下のとおりです。

- 通信コンポーネント (例えば VTAM または NCP) の障害
- CICS サブシステムまたは IMS サブシステムの障害
- 直接 VTAM またはサブシステム・コマンド (例えば、IMS /CLSDST または /STOP コマンド) が入力された
- CICS トランザクションまたは IMS トランザクションの障害 (後続のエラー処理を間接的な原因とする)

これらのいずれの場合においても、このセッション障害は、残りの動作可能サブシステムの障害と同じように見えます。しかし、障害後にユーザーが行わなければならないリカバリー・プロセスおよび再同期プロセスはそれぞれ異なります。このトピックでは、通信コンポーネントの障害時、IMS サブシステムまたは CICS サブシステムの障害時、あるいはこの 2 つのサブシステム間のセッションに対して直接コマンドを発行した場合に何が起るかについて説明しています。

### 関連概念

[656 ページの『トランザクションの異常終了の処理』](#)

CICS リカバリー・トランザクションを設計する際には、セッションおよびサブシステムの障害に関する考慮事項に加え、以下のトピックで説明するトランザクション終了の場合に行うアクションについても考慮する必要があります。

## セッションの再確立

障害が起こったセッションは、どちらのハーフセッション・パートナーでも再始動できます。

### このタスクについて

セッションを再始動した場合、ハーフセッション同士の関係は、セッション障害時におけるものと同じです。すなわち、1次ハーフセッションであった方は基本のままであり、2次ハーフセッションであった方は2次のままです。IMSとCICSの両方とも、セッション極性の標識でセッションを開始するために使用されたセッション修飾子の対(SQP)をロギングすることで、この方向付けを忘れずにいます。

CICS マスター端末オペレーターは、以下のコマンドを使用してセッションを再開することができます。

```
CEMT SET TERMINAL (tttt) ACQUIRED
```

ここで、*tttt* は、端末管理テーブルで定義された CICS セッションの TERMIDNT です。その後で、CICS はセッションを開始して、CICS システム定義パラメーターによって固定されたセッション極性を維持します。認可された IMS 端末オペレーターが /OPNDST コマンドを発行することによってセッションが開始中の場合には、そのコマンドは同じサブプール名 (IMS ローカル・セッション修飾子名) と障害時にセッションに割り振られたセッション ID (CICS ローカル・セッション修飾子名) を指定しなければなりません。オペレーターは、IMS /DISPLAY コマンドを用いることにより、必要に応じてこれらを手入して使用することができます。IMS は、セッション障害時に有効であったのと同じセッション極性を維持しつつ、自動的にセッションを再設定します。

セッションがコールド・スタート・モードの IMS によって再始動中の場合、および CICS が 1 次ハーフセッションとなるように定義されている場合には、CICS が再始動を開始する必要があります。

セッションが再確立されたときには、IMS はデータ・フロー制御ブラケット状態をセッション障害の前の最後の同期点の状況にリセットしようと試みます。CICS によって、セッションを SEND 状態の IMS を使用してブラケット間またはブラケット内状態として再確立することができます。

## セッションの再同期化

セッションがアクティブなときには、IMS と CICS の両方ともそのセッションのために一組の SNA メッセージ・シーケンス番号を維持します。障害発生後の再始動試行時には、サブシステム間でセッションの同期点の合計が STSN コマンドによって検査されます。

### このタスクについて

CICS/IMS セッションは、以下の条件が両方とも真である場合に再開します。

- セッションはコールド・スタート中です。
- ハーフセッションのシーケンス番号は、一致するか、または受け入れ可能な限度内にあります。

セッションに障害が起こった場合には、そのセッションに対して保留状態のアクティビティーがある CICS トランザクションにも障害が起こります。CICS は動的トランザクション・バックアウト (DTB) を使用して、リカバリー可能リソースの保全性を確認します。ISC ネットワーク内の IMS サブシステムと CICS サブシステム間の整合性を保証するために、DTB 未確定パラメーターを IN-DOUBT(WAIT) または DTB=(YES, WAIT) に指定しなければなりません。

WAIT は、再同期が行われるまでリカバリー可能リソースに対するロックを保留します。

この状態が好ましくない場合には、IMS-CICS ISC トランザクションに IN-DOUBT(BACKOUT) または DTB=YES を指定することができます。ただし、この指定を行うと、セッション再同期後に IMS に対して重複メッセージが送信されることがあり、IMS ユーザー・トランザクション内にこれらを解決するための論理が必要となる可能性があります。



DTB 未確定パラメーターに WAIT が指定されていない場合は、IMS TERMINAL マクロ・システム定義の OPTIONS パラメーターに FORCSESS を指定するか、あるいは認可された IMS 端末オペレーター用の / CHANGE コマンドを使用する必要があります。

## 関連概念

### 509 ページの『STSN を使用するセッション同期の判別』

再同期が必要な場合は、それをバインド交渉中に伝えることができます。再同期し、ハーフセッションをセッション終了時に存在した状態に戻すための要件を判別するために、BIND 要求の 2 つのフラグが使用されます。

### 960 ページの『シーケンス番号の設定およびテスト (STSN)』

IMS がワークステーションにシーケンス番号の設定およびテスト (STSN) コマンドを送信すると、メッセージ再同期が開始されます。

## 関連タスク

### 636 ページの『CICS バックアウト未確定処理の定義』

IMS に対する同期点要求の送信と肯定応答の受信の期間には、CICS はリモート・システムがコミットしたかどうかを認識しません。この期間は未確定期間として知られています。

## 未解決のトラフィックの処理

IMS では、トランザクション処理はセッション状況から独立しています。つまり、メッセージが受信され、正常にキューに入れられた場合には、そのメッセージは常に処理されるということです。

しかし、IMS が後続の応答を処理して会話処理を継続する方法は、メッセージのタイプ、発生したエラーのタイプ、および IMS サブシステムに障害が起こったかどうかによって異なります。

一般的に、セッションだけに障害が起こり (IMS には障害が起こっていない)、同期処理が IMS と CICS との間で 1 つも発生していないときには、以下のとおりになります。

- セッションは、再開したときにブラケット間結合になっています。
- IMS は常に、セッションが正常に再始動した後で、非同期応答または非送信請求非同期出力を送信または再送信します。
- IMS は、/DISPLAY、/RDISPLAY、および /FORMAT などのコマンドから発生するキュー非同期応答を送信または再送信します。
- IMS 非同期トランザクションがセッションの故障時に異常終了した場合には、例外応答はもはや不可能であるため、IMS は ATTACH SYSMSG を使用してエラー・メッセージを送信します。

IMS によって同期に処理された応答の場合には、以下のとおりになります。

- 応答モードまたは会話型出力が保留状態の場合には、セッションは IMS ブラケット内 /SEND で結合され、保留状態の応答を送信または再送信することができます。
- IMS が会話モードになっており、入力が必要である場合には、IMS はイン・ブラケット /RECEIVE を結合しようと試みる。CICS は、このバインドをブラケット間状態で行うように折衝します。その結果、IMS は会話を終了し、出力メッセージを廃棄し、さらに会話型異常終了出口ルーチン呼び出します。この出口ルーチンは、直前の会話型プロセスの結果発生したデータベースの変更点を SPA の内容に基づいてすべてバックアウトするために、ユーザー・プロセスを呼び出して IMS トランザクションをスケジュールすることができます。

セッションが再確立されて、前述のパラグラフで説明されたブラケット間に送信されたときに、IMS トランザクションがまだ会話型または応答モード入力の処理を完了していない (すなわち、出力がまだ使用できない) 場合には、IMS は、(出力が使用可能になったときに) セッションを再開するように要求するメッセージをマスター端末オペレーターに送信してセッションを終了します。IMS がブラケット間バインドに必要なプロセス中トランザクションを終了できないために、これが起こります。

- 上で記述した場合を除いて、IMS は障害時に受信したコマンドをすべて取り消し、IMS コマンドの結果として発生する保留状態の同期応答メッセージを廃棄します。
- IMS 同期トランザクションがセッションの故障時に異常終了した場合には、例外応答はもはや不可能であるため、IMS は ATTACH SYSMSG を使用してエラー・メッセージを送信します。

IMS はこれらのトランザクションを以下の方法で処理します。

- 緊急時再始動の後で、IMS はリカバリー不能トランザクションからの結果として発生した保留状態の応答メッセージを廃棄します。
- IMS が入力トランザクションを受信して、それがエンキューされた後では、セッション・プロトコルまたは障害によってそのトランザクションを取り消すことはできません。
- これは、CICS 同期トランザクション (応答モードまたは会話) に対する応答メッセージがエンキューされた後では、直前の IMS 処理の結果はすべてコミット済みであることを示します。これらの更新は、応答メッセージが廃棄されたとしてもバックアウトされません。

CICS トランザクションを STSN 処理によって判別されたとおりに IMS に再送信しなければならない場合には、CICS サブシステム自体ではなく、CICS アプリケーションがこの判別を行い、当該のトランザクションを再構成するか、あるいはそのトランザクションを再入力するように要求する必要があります。

セッションに障害が起こった後では、CICS は元のトランザクションが実行していた環境を再確立することはできません。すなわち、当初トランザクションを入力した端末との接続はもはや存在しません。その上、当該のトランザクションは異常終了しています。したがって、CICS アプリケーション・プログラマーは「トランザクションの再始動」が CICS によって呼び出されるように定義し、再始動したセッションにおいて CICS が受信する可能性のあるすべての IMS 出力を CICS が処理できるようにする必要があります。このトランザクションが呼び出されたときには、ここではむしろ、端末ではなくセッションが基本機能になります。端末に送信する必要のある出力が存在する場合には、今度はこのトランザクションが非同期トランザクションを呼び出して、その端末を獲得しなければなりません。

呼び出される再始動トランザクションのトランザクション・コードは、入力メッセージと一緒に CICS に送信される機能管理ヘッダーの DPN フィールドに入れられて送信されます。このトランザクション・コードは、アウトバウンド・メッセージと一緒に CICS によって送信された BUILD ATTACH コマンドの RPROCESS フィールドに指定されたものです。これは、IMS によって DPN フィールドに自動的に折り返される RDPN パラメーターになります。DPN は、IMS MFS によって変更されていない限り、IMS によってアウトバウンド FMH に入れられて送信されます。

CICS のあらゆる出力応答の送信先となる端末の ID は、着信機能管理ヘッダーの PRN フィールドに入っています。これは、アウトバウンド・メッセージで CICS によって送信された BUILD ATTACH の RRESOURCE フィールドに指定された値です。これは、MFS によって変更されない限りアウトバウンド FMH の PRN フィールドに自動的に IMS によって折り返される RPRN パラメーターになります。この値は、端末を獲得するために再始動トランザクションによってスケジュールされた (START) 非同期トランザクションの TERMIID として使用されます。

再始動トランザクションは、TRANSID フィールドと TERMIID フィールドの両方を繰り上げる RETRIEVE を発行します。TRANSID によって指定されたトランザクションの開始時には、TERMIID によって指定された端末はこのトランザクションによって基本機能として所有されています。

### 関連概念

#### 656 ページの『トランザクションの異常終了の処理』

CICS リカバリー・トランザクションを設計する際には、セッションおよびサブシステムの障害に関する考慮事項に加え、以下のトピックで説明するトランザクション終了の場合に行うアクションについても考慮する必要があります。

#### 657 ページの『再始動のための CICS アプリケーションのコーディング』

セッションまたはサブシステムの障害の結果、ISC セッションで障害が発生した場合は、セッションの再始動および再同期を試みることができます。

## トランザクションの異常終了の処理

CICS リカバリー・トランザクションを設計する際には、セッションおよびサブシステムの障害に関する考慮事項に加え、以下のトピックで説明するトランザクション終了の場合に行うアクションについても考慮する必要があります。

### IMS トランザクション異常終了

IMS トランザクションが異常終了したときには、このトランザクションによって実行された IMS リソースに対するすべての変更は、IMS によってバックアウトされます。トランザクションが同期 (CICS SEND/ SYNCPOINT) の場合には、IMS は否定応答 (例外応答 DR2) を CICS に送信します。これによって、CICS ト



ランザクションも異常終了します。この否定応答とそれに続く CICS トランザクションの異常終了の結果、CICS は直前の CICS 同期点以降にリカバリー可能リソースに対して行われた変更をすべてバックアウトします。エラー・メッセージも CICS に送信されます。CICS は、このメッセージをマスター端末宛先 CSMT に経路指定します。MFS 出力形式が形式ライブラリーに存在しないとき、あるいは入出力エラーを招いたときに、例外が起きます。この場合 IMS は、肯定同期点応答を送信済みであるため、ATTACH SYMSMG を使用して CICS に MFS エラーを指示する必要があります。

トランザクションが非同期 (CICS START または SEND LAST) の場合には、IMS はエラー・メッセージを CICS に送信します。CICS はこのメッセージをそのマスター端末宛先へ経路指定します。エラー・メッセージの前に、発生したエラーのタイプによって、SYMSMG ヘッダーまたは ERP ヘッダーのいずれかがきます。IMS が同期点応答を入力に送信する前に、エラーが起こった場合には、例外応答および ERP が使用されます。同期点応答の後にエラーが起こった場合には、ATTACH SYMSMG が使用されます。

## CICS トランザクション異常終了

トランザクションが IMS のためのフロントエンドとして作用する CICS サブシステムによって開始され、その CICS トランザクションが同期点に達する前に異常終了した場合には、実行される処理は、DEFINE TRANSACTION IN-DOUBT パラメーター (または DFHPCT DTB= パラメーター) に関する仕様に従って処理されます。

非同期トランザクションが IMS フロントエンドによって開始され、その CICS トランザクションがトランザクションの受信時に異常終了した場合には、IMS は通知されません。これは、メッセージを受信すると、CICS ミラー・トランザクションが即時 DR2 を非同期入力に戻し、これを処理するためにトランザクションをスケジュールするからです。このスケジュールされたトランザクションに障害が起こった場合には、IMS に対する ISC リンクはもはやアクティブではなく、通知は不可能です。しかし、CICS は CICS 宛先 CSMT にトランザクションの障害について通知します。

同期プロセス時に、CICS が 2 次ハーフセッションで、ALLOCATE コマンドが正常に発行され、(例えば、バックアウトの原因になる DTB のために) IMS に送信されるメッセージが使用できない場合には、CICS は LUSTATUS BB/EB を IMS に送信して自動的にセッションを割り振り解除します。

同期トランザクションが SEND/SYNCPPOINT コマンドを使用して IMS にメッセージを送信した後で異常終了したときに、そのメッセージがすでに IMS 入力キューでエンキューされている場合には、IMS はこのメッセージを処理します。IMS が出力メッセージを CICS に送信しようと試みたときには、そのメッセージは CICS から否定応答を受信します。その結果、そのメッセージは、再送信が可能になるまで、あるいは IMS マスター端末オペレーターによってデキューされるまで、IMS 出力キューに入ったままになる可能性があります。

### 関連資料

645 ページの『CICS のための機能管理ヘッダーのコーディング』

CICS では、IMS によって使用されるのと同じ SNA 定義機能管理ヘッダー・フィールドのいくつかが使用されます。

652 ページの『エラー・リカバリー・プロシージャ機能管理ヘッダー』

CICS は、エラー・リカバリー・プロシージャ機能管理ヘッダーを送受信して、プロセス間でエラー情報を伝送します。

## 再始動のための CICS アプリケーションのコーディング

セッションまたはサブシステムの障害の結果、ISC セッションで障害が発生した場合は、セッションの再始動および再同期を試みることができます。

セッション再同期 (STSN 処理) が正常に行われず、介入なしでセッションを再始動できない場合には、両方のサブシステムのマスター端末オペレーターに通知されます。この状態では、セッションの状況および行うべきアクションについて、端末ユーザーにも通知しなければならなくなる可能性があります。端末ユーザーが行うべきアクションとは、障害が起こったトランザクションの再入力や、修正アクションが MTO によって行われるまで待機することなどです。

リカバリー可能と定義された ISC トランザクションが、IMS によって受信されてログに記録された後では、セッションおよびサブシステムの障害を回復することができます。再始動後、セッションが回復したときに、IMS に保留状態の出力がある場合には、その出力の送信は次のように行われます。

- IMS が入力メッセージを受信したのと同じセッションに出力を送信する場合は、その出力メッセージは、最初に入力と一緒に送信されたヘッダーと同じタイプのヘッダーと一緒に送信される。したがって、SEND/RECEIVE または SEND LAST で送信されたメッセージに対する応答は、ATTACH FM ヘッダーと一緒に戻され、START/RETRIEVE で送信されたメッセージに対する応答は ATTACH ヘッダーおよび SCHEDULER FM ヘッダーと一緒に戻されます。
- IMS が入力メッセージを受信したセッションとは別のセッションに出力を送信する場合は、その出力メッセージは非送信請求出力と見なされて、非同期に (ATTACH SCHEDULER と一緒に) 送信される。

再始動の後で、IMS が出力メッセージを送信したときには、CICS は入力メッセージを読み取り、トランザクションを開始します。

- アプリケーションの内部で、EXEC ASSIGN STARTCODE パラメーターは調べられて、このトランザクションが同期に開始されたのか、非同期に開始されたのかを判別されます。
- 再始動トランザクションが非同期に開始された場合
  - ISC 応答を入手するために RETRIEVE が発行される。EIB 標識を保管する必要があります。CICS は、エンド・ユーザーの端末を基本機能としてこの CICS プログラムに接続します。これにより、CICS EXEC SEND コマンドを使用して IMS 出力を直接に端末に発行することができます。
  - RETRIEVE の実行後に保管された EIB 標識をテストして、このトランザクションを終了するために SYNCPOINT または RETURN を発行することができるかどうかを判別する必要があります。
- トランザクションが SEND/RECEIVE によって同期にまたは SEND LAST によって非同期に開始された場合
  - 入力メッセージは RECEIVE を使用して入手される。この場合では、セッションは現在基本機能になっています。したがって、RECEIVE コマンドは SESSION パラメーターの仕様を必要としません。
  - RECEIVE の後で、CICS アプリケーションは、順番に EIB フィールド EIBSYNC、EIBFREE、および EIBRECV を保管してから、検査します。
  - ここで、アプリケーションは EXTRACT ATTACH を発行して、入力データの形式、および最初にトランザクションを提出した端末の ID を判別します。
  - ここで、再始動ロジックは、追加 CICS トランザクションを START して、基本機能としてエンド・ユーザーの端末を獲得しなければなりません。START 要求は、EXTRACT ATTACH を使用して PRN フィールドで検出した端末 ID を START の TERMID パラメーターとして使用することによって行います。
  - ここで EIB 標識をテストして、同期点が必要かどうか、およびセッションを解放できるかどうかを判別する。これを実行できる場合には、アプリケーションは RETURN を発行して、暗黙に同期点を発行し、セッションを解放できるようにします。

再始動トランザクションのコンテキスト内では、照会のためにデータをリトリブしたり、データベースやファイルを更新したりすることができます。

前述のロジックの他に、CICS 再始動トランザクションには以下を実行するためのロジックを組み込むことができます。

- 一時記憶域に発信元端末から受信した入力を保管する入力トランザクションを再作成する必要がある場合には、この情報を入手して IMS に再送信することができます。
- ロジックを作成して、トランザクションの再入力、または処理前のマスター端末オペレーター処置の待機など、実行される特定の処理について、端末ユーザーに通知してください。

## 第 35 章 ISC データ・フロー制御の例

以下のトピックでは、ISC データ・フロー制御の例を示します。

### MFS 以外のブラケットおよび半二重プロトコルの例

以下の一連の例では、ISC データ・フロー制御プロトコルを示します。

これらの例で、小括弧内の項目はフロー上でのオプションです。

簡単にするために、本トピックの例では、方向と応答の両方のフローの単独チェーン SCHEDULER メッセージ、ATTACH、および SCHEDULER プロトコルは除外されているものと見なします。さらに、1 次ハーフセッション (PHS) または 2 次ハーフセッション (SHS) のいずれも図示のとおりブラケットを開始できるという点でブラケット・プロトコルは対称であるとみなされます。

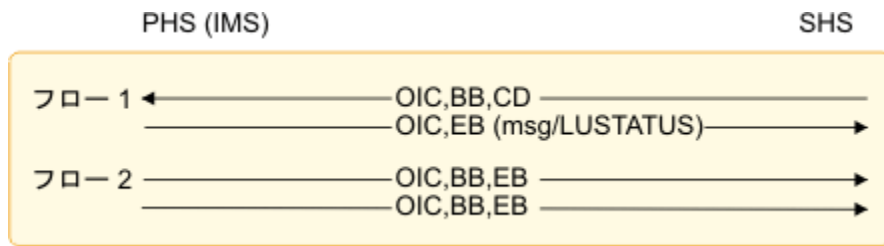


図 89. SINGLE1 として IMS に対して定義された PHS コンポーネントのブラケット・プロトコルの例

サンプルのフロー 1 は、方向変換を受信した後に起こる最初の出力で PHS がブラケットを終了しているところを示しています。出力が存在せず、入力が入力を保証できないときには、ブラケットは独立型 LUSTATUS によって終了します。

サンプルのフロー 2 は、PHS がブラケット間状態時に起こる出力を送信しているところを示しています。

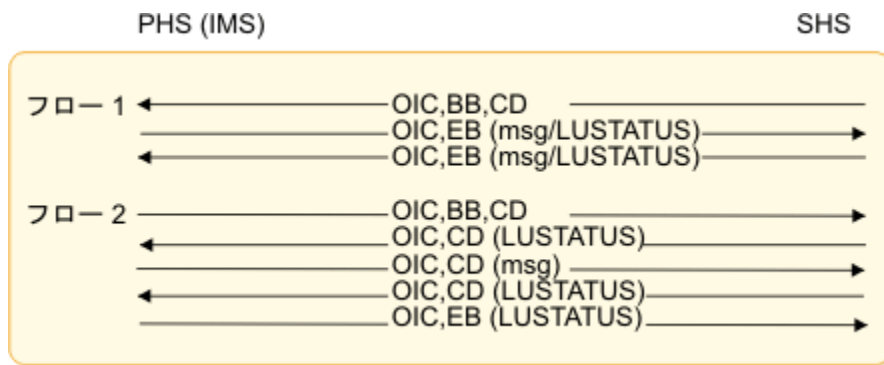


図 90. SINGLE2 として IMS に対して定義された PHS コンポーネントのブラケット・プロトコルの例

サンプルのフロー 1 は、方向変換を受信した後に PHS がフローをブラケット・イニシエーターに戻しているところを示しています。ブラケット・イニシエーターは、入力の継続またはブラケットの終了を任意に選択することができます。出力が存在せず、入力が入力を保証できないときには、独立型 LUSTATUS を使用してフローがブラケット・イニシエーターに戻されます。

サンプルのフロー 2 は、PHS の非同期出力の送信を示しています。また、IMS は、潜在的な LUSTATUS CD ループを検出し、LUSTATUS を使用してブラケット終了を強制しています。

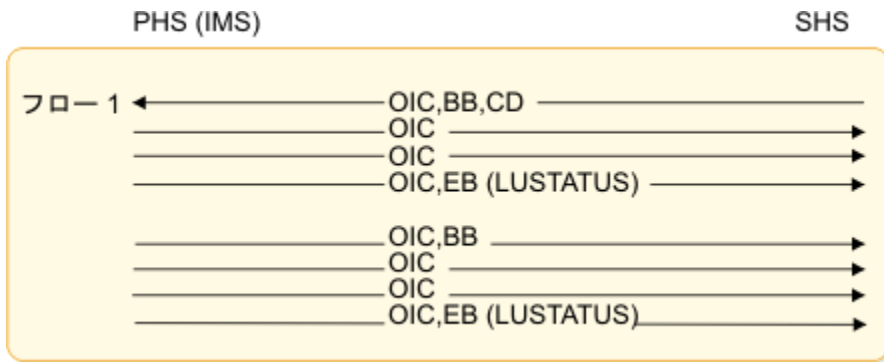


図 91. MULT1 として IMS に対して定義された PHS コンポーネントのブラケット・プロトコルの例

サンプルのフロー 1 は、PHS が LUSTATUS を使用してブラケットを終了しているところを示しています。方向変換を受信した後でキューからの最後の出力に続いて、ブラケットが終了します。出力が存在せず、入力が出力を保証できないときには、ブラケットは独立型 LUSTATUS によって即時に終了します。

サンプルのフロー 2 は、PHS がキューからの最後の出力の後で LUSTATUS を使用してブラケットの開始および終了の両方を行っているところを示しています。他のキューから使用可能な追加出力によって、後続のブラケットが開始することになります。

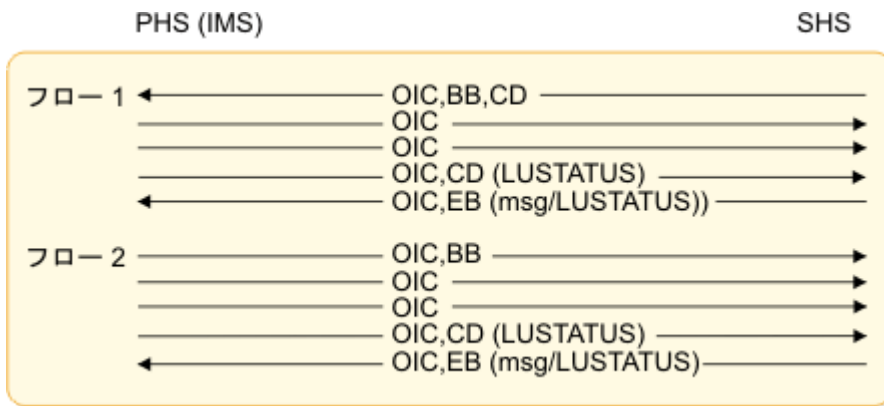


図 92. MULT2 として IMS に対して定義された PHS コンポーネントのブラケット・プロトコルの例

サンプルのフロー 1 は、PHS がフローをブラケット・イニシエーターに LUSTATUS を使用して戻しているところを示しています。方向変換を受信した後でキューからの最後の出力に続いて、ブラケットが終了します。ブラケット・イニシエーターは、入力の継続またはブラケットの終了を任意に選択することができます。出力が存在せず、入力が出力を保証できない（つまり、入力が会話型でも応答モードでもない）ときには、独立型 LUSTATUS を使用することによってフローが戻されます。

サンプルのフロー 2 は、PHS がブラケット間状態で起こる出力を送信しているところを示しています。

## MFS ブラケットと半二重プロトコルの例

以下のトピックでは、MFS 出力と MFS 入力の両方の例を示します。

### MFS 出力の例

下記の図は、MFS 出力の例を示しています。

以下の図に示された例では、IMS メッセージは次のような 3 つの表示ページからなっています。

- 2 つの表示ページからなる論理ページ 1
- 1 つの表示ページからなる論理ページ 2

すべての例で、DPN=X'03' を示している ATTACH FM ヘッダーはオプションです。

**例外:** 最初のページング要求の場合は、DPN=X'03' を示している ATTACH FM ヘッダーは必須です。

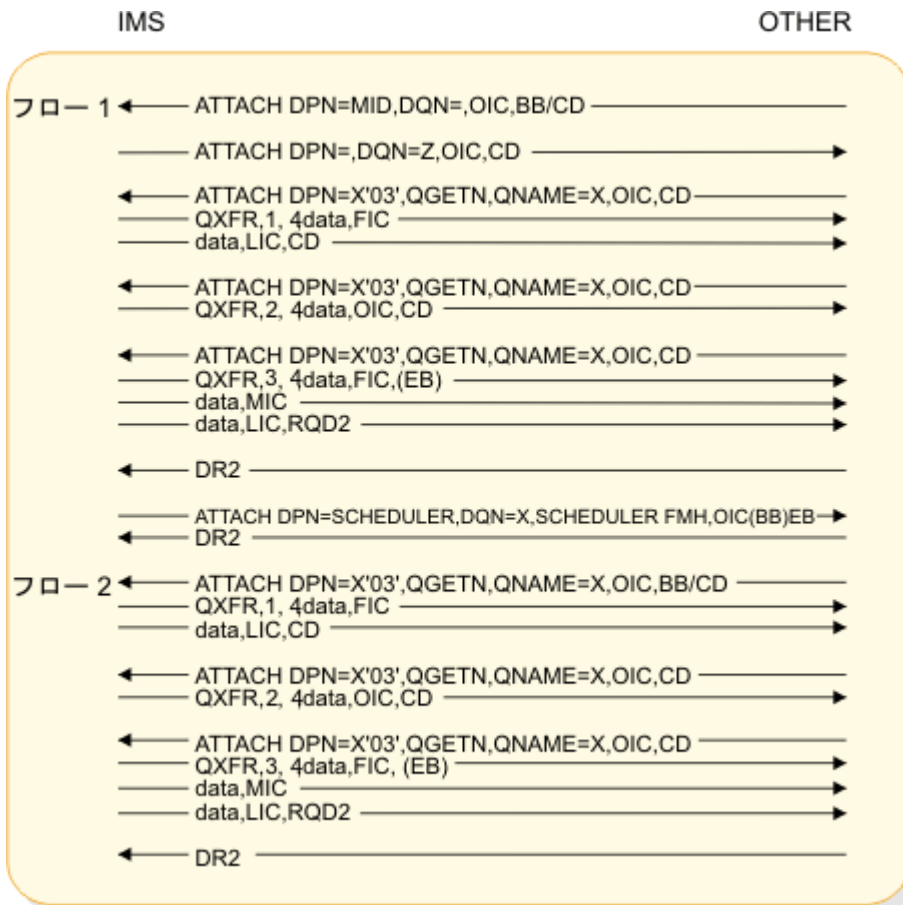


図 93. 例 1: 要求時ページ化出力、定義されていないオペレーター論理ページング (OLP)、QGETN FM ヘッダーによる順次リトリーブ

IMS 処置: メッセージ・キューからメッセージを除去します

注:

1. QCURSOR=0400010001、QCOUNT=020002
2. QCURSOR=0400010002、QCOUNT=020002
3. QCURSOR=0400020001、QCOUNT 存在しない
4. DD FM ヘッダーをデータの前に置くことができます

ページング要求で戻された QNAME は、出力 ATTACH DQN で送信されたものと同じ値でなければなりません。

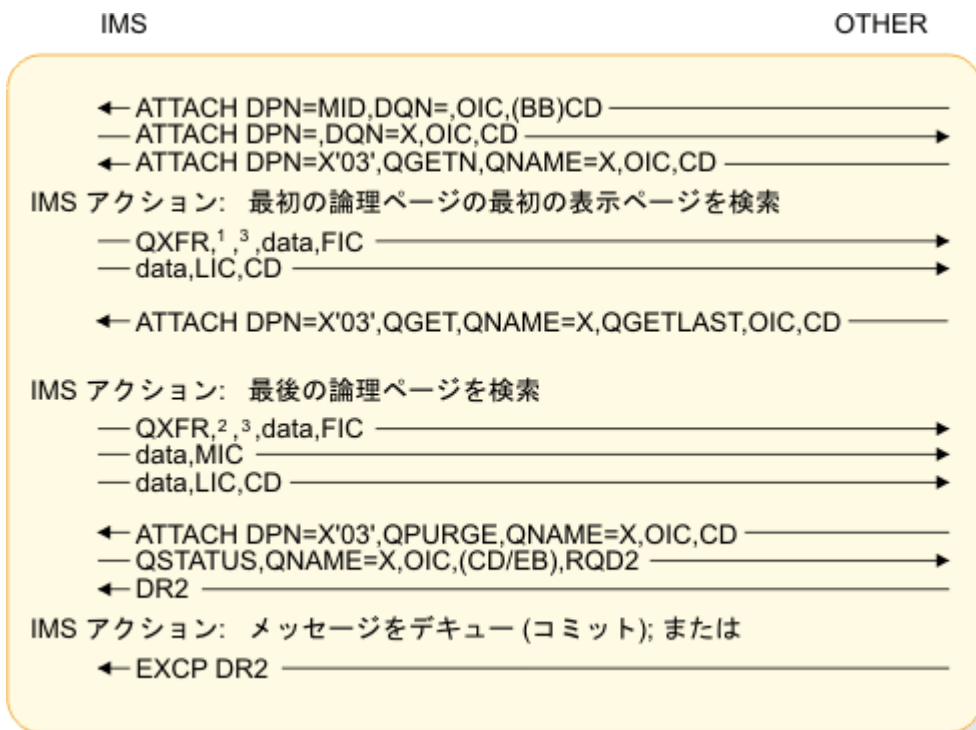


図 94. OLP が定義され、QGET (最後のページ要求であり、カーソルによる) が使用されている要求時ページ化出力の例。メッセージをデキューするために使用される QPURGE

注:

1. QCURSOR=0400010001、QCOUNT=020002
2. CURSOR=0400020001、QCOUNT 存在しない
3. データ記述子 FM ヘッダーをデータの前に置くことができます

SCHEDULER 以外の要求時ページ化出力に対するこれらの同じ条件によって、結果的に例外応答および後続の ERP FMH7 が送信されることとなります。

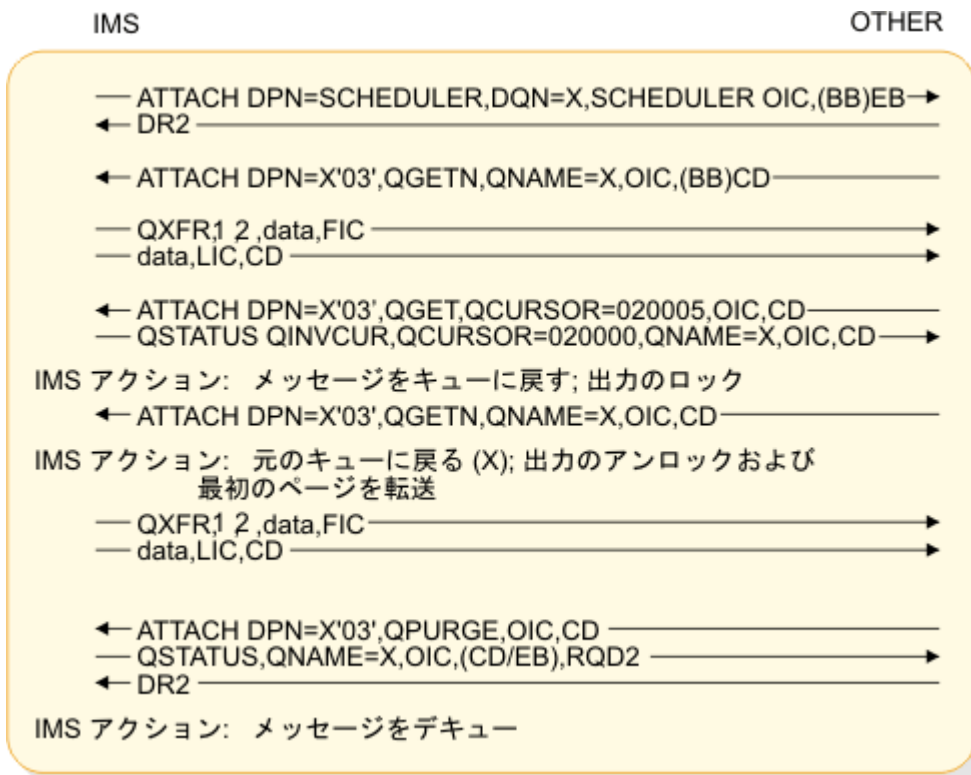


図 95. OLP が定義され、出力メッセージの範囲内がないページに対するカーソル要求による QGET が行われた要求時ページ化 SCHEDULER 出力の例

注:

1. QCURSOR=0400010001、QCOUNT=020002
2. データ記述子 FM ヘッダーをデータの前に置くことができます

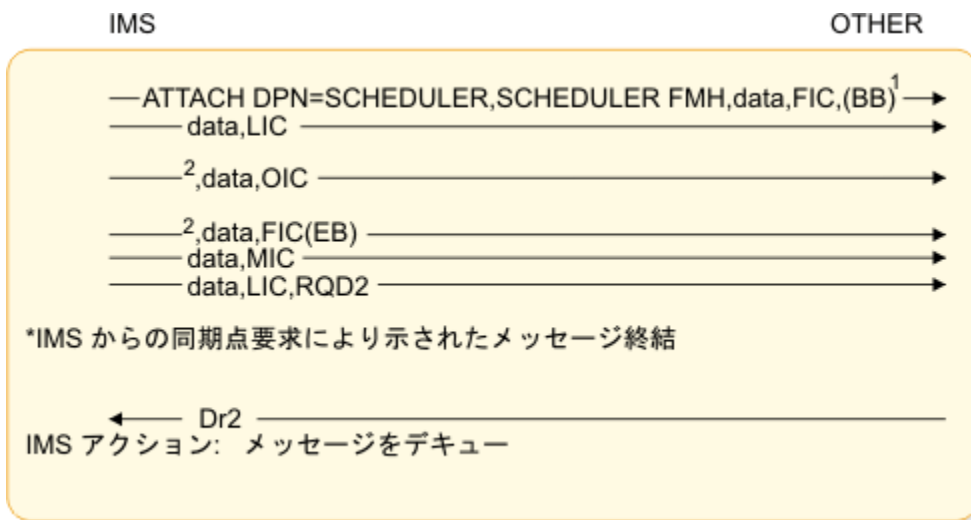


図 96. 自動ページ化出力の例

注:

1. 自動ページ化出力要求 DR1 (RQD1) で BB によるチェーンの先頭の送信。
2. データ記述子 FM ヘッダーを各チェーン内のデータの前に置くことができます。ATTACH は、複数チェーン・メッセージを示しています。



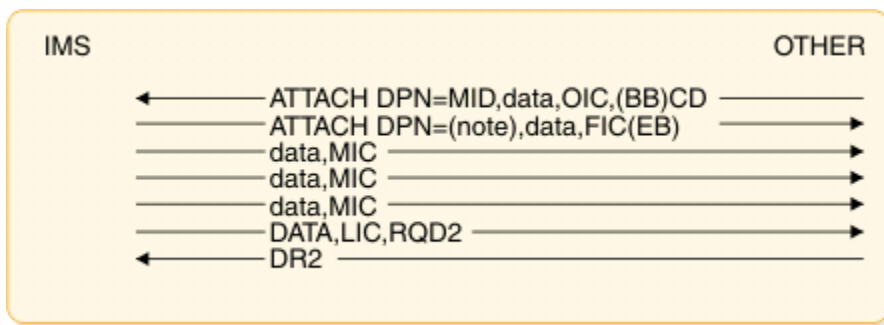


図 97. 非ページ化出力メッセージの例

注：データ記述子 FM ヘッダーをデータの前に置くことができます。

## MFS 入力の場合

下記の図は、MFS 入力の例を示しています。

以下の例では、3つの DPAGE が定義されています。

DPAGE 1 は、2つのセグメント用のデータを定義します

DPAGE 2 は、1つのセグメント用のデータを定義します

DPAGE 3 は、3つのセグメント用のデータを定義します

メッセージ用の3つの LPAGE が作成されます。DSN に各チェーンを提供されたデータ記述子機能管理ヘッダーは、各 DPAGE を選択するために使用されます。最初の DSN 名が DPAGE 1 を選択し、2番目の DSN 名が DPAGE 2 を選択し、3番目の DSN 名が DPAGE 3 を選択します。

\*最初の DPAGE 用の ATTACH とデータ:

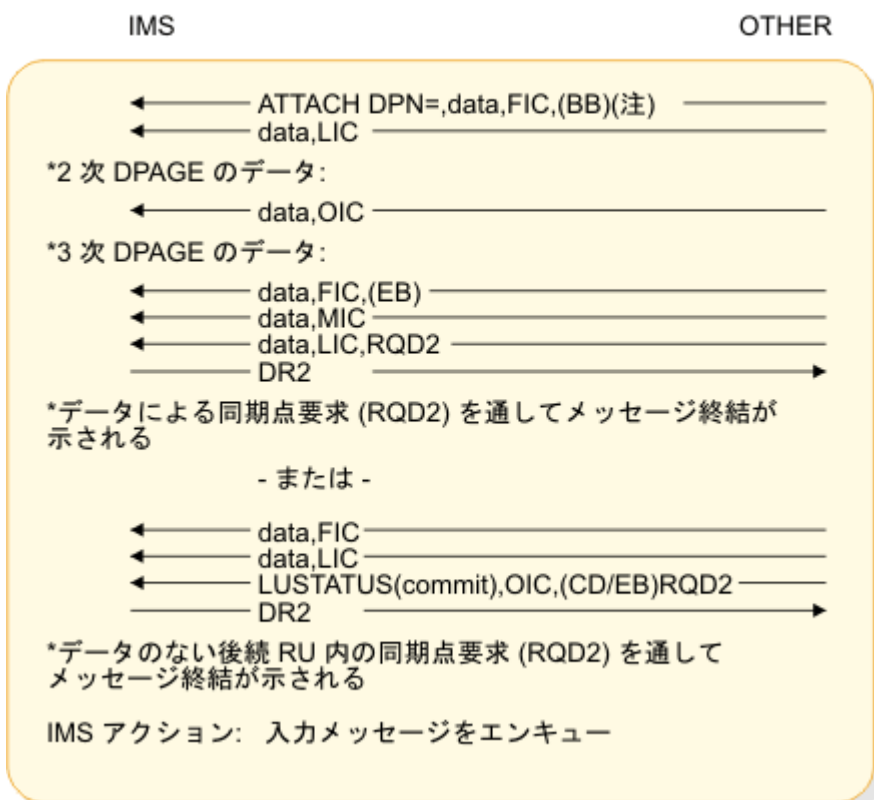


図 98. 自動ページ化入力の例 (3つのチェーン)

データ記述子 FM ヘッダーをデータの前に置くことができます。

メッセージ用の3つのLPAGEが作成されます。各DPAGEを選択するために、データに関する条件付きテストが実行されます。最初のテストの結果がDPAGE 1を選択し、2番目のテストの結果がDPAGE 2を選択し、3番目のテストの結果がDPAGE 3を選択します。

\*最初のDPAGE用のATTACHとデータ:

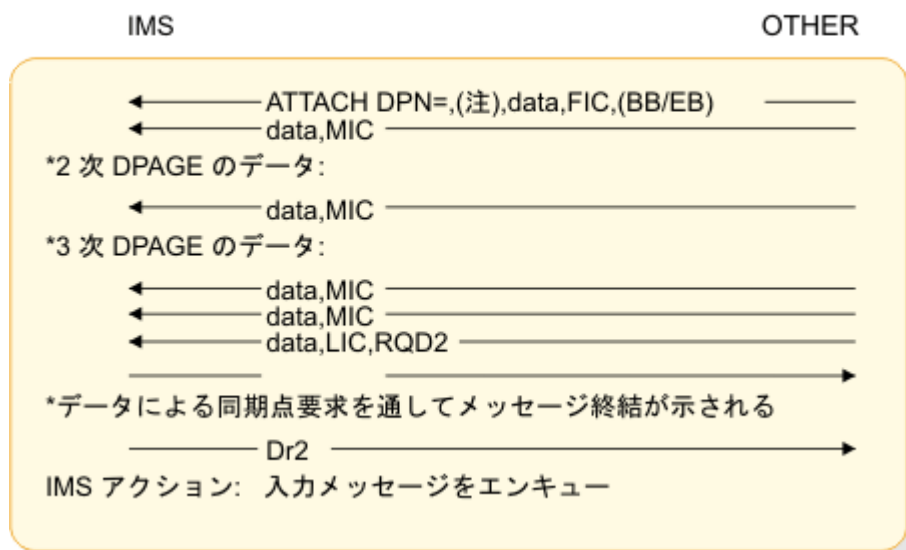


図 99. 自動ページ化入力の例 (単一の伝送チェーン)

データ記述子 FM ヘッダーを送信して、DPAGE を1つだけ選択することができます。ATTACH は、複数チェーン・メッセージを示しています。

## SBI/BIS の例

IMS は、どの掲載例の場合でも、1次ハーフセッション (PHS) であることもできますし、2次ハーフセッション (SHS) であることもできます。したがって、PHS および SHS のために表示された機能およびコマンドのすべては IMS 機能およびコマンドです。

すべての SBI および BIS 順序は、LU 6.1 プロトコルを使用するあらゆるセッションに対して有効ですが、以下の例では1次ハーフセッションおよび2次ハーフセッションが両方ともIMSであることを前提としています。ここでは、IMS 以外のサブシステムがこれらの順序を送信または受信することができる時点または方法について、一切考慮していません。



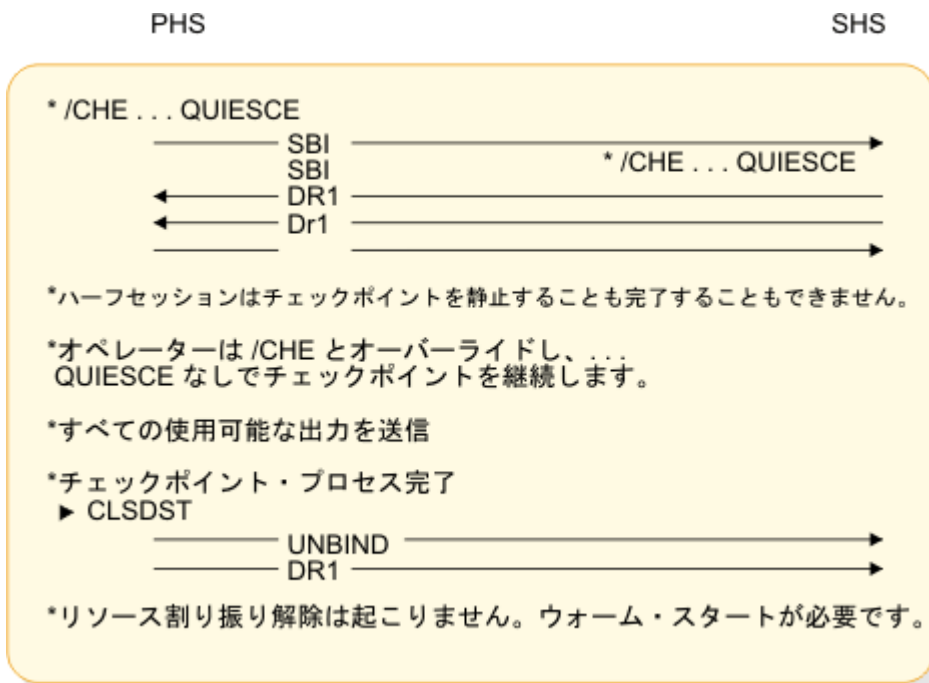


図 102. オペレーターによるオーバーライドが原因で障害が起こる、SBI/BIS を使用した PHS と SHS の同時シャットダウンの例

## シグナル・プロトコルの例

次の例は、SIGNAL RCD の使用例を示しています。

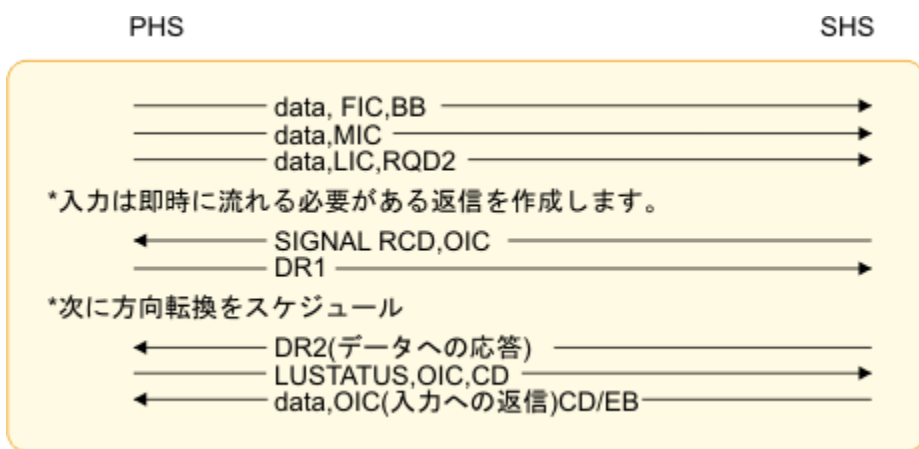


図 103. シグナル・プロトコルの例



# 第 36 章 ISC エラー・リカバリー・プロシージャの例

以下のトピックでは、ISC エラー・リカバリー・プロシージャの例を示します。

## 送信側検出エラーの例

以下の図は、ページ化出力 (または入力) 時の送信側検出エラーの例を示しています。IMS は、両方の図で送信側です。

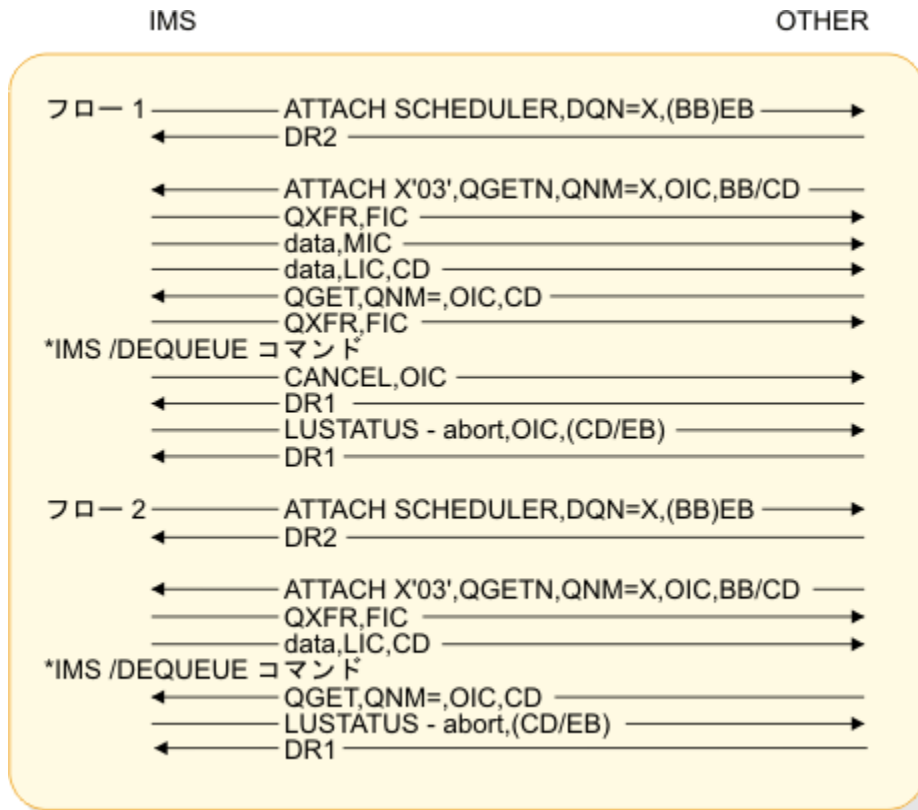


図 104. 要求時ページ化出力メッセージの送信中の送信側検出エラーの例

図 96 のフロー 1 は、2 番目の出力要求ページの間が発生する IMS /DEQUEUE コマンドを示しています。CANCEL は、現在行われているチェーン (ページ) を終了し、LUSTATUS は受信プロセスを終了します。ページ化出力メッセージがブラケットを開始した別の入力メッセージまたは出力メッセージの後に続く場合には、BB は最初のページの OIC では送信されません。

サンプルのフロー 2 は、CANCEL コマンドが必要にならないように IMS /DEQUEUE コマンドが出力ページ間で発生することを除いて、前述のフローと同じです。

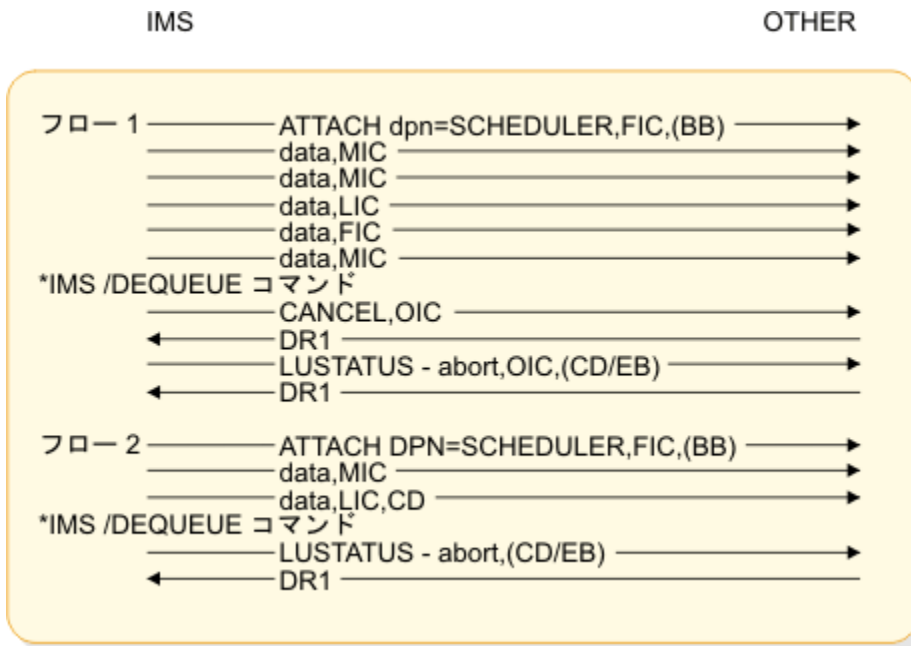


図 105. 自動ページ化出力メッセージの送信中の送信側検出エラーの例

再び、CANCEL は、現在行われているチェーン (ページ) を終了し、LUSTATUS は受信プロセスを終了します。初期開始ブラケットおよび結果のブラケット送信 / 受信プロトコルは、669 ページの図 104 の要求時ページ化出力に対するものと同じです。

フロー 2 は、CANCEL コマンドが必要にならないように IMS /DEQUEUE コマンドが出力ページ間で発生することを除いて、図 97 のフロー 1 と同じです。

## 受信側検出エラーの例

説明を簡単にするために、以下の一連の図では、すべての例で両方向のフローの単独チェーン (OIC) メッセージを想定しています。さらに、ブラケット・プロトコルと ERP プロトコルは対称であると考えられています。それは、どちらのハーフセッションも、図示されているとおり、ブラケットまたは ERP プロシージャを開始することができるからです。

システムの故障に関連しないすべてのセッション終了の場合には、ATTACH 情報は回復し、最後の同期点までバックアップされます。以下の例では、IMS は受信側です。



図 106. SINGLE1 として定義された IMS コンポーネントの受信側 ERP の例





IMS

OTHER

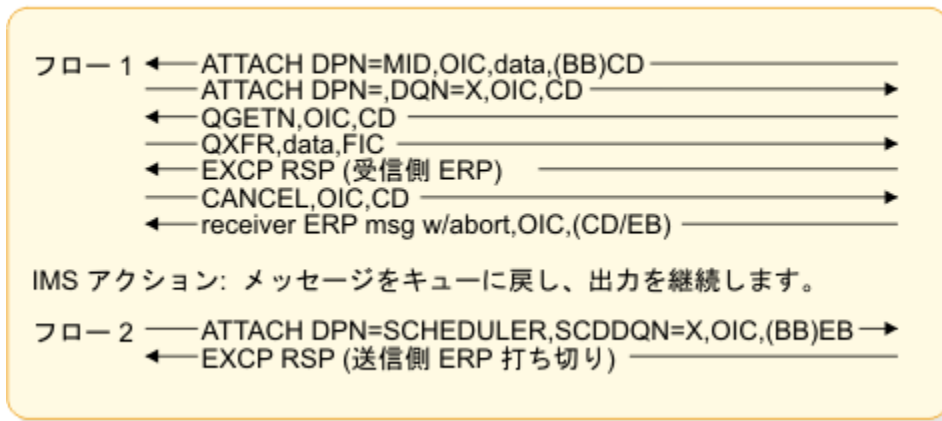


図 110. 受信側検出エラーがある要求時ページ化出力の例

IMS アクション: メッセージをキューに戻し、出力を継続します。

この例では、フロー 1 は、要求時ページ化メッセージを拒否するために受信側 ERP を使用している受信側を示しています。フロー 2 は、同じ終了結果をもつが、受信側からの ERP メッセージに関連しない代替方式を示しています。

## 第 37 章 IMS-CICS ISC のサンプル・プログラム

以下のトピックには、IMS と CICS の間での ISC の使用例を示したサンプル・プログラムが記載されています。

このサンプル・プログラムを「*CICS Transaction Server for z/OS CICS Intercommunication Guide*」の対応するサンプル・プログラムと結合することにより、両方の製品で ISC 機能を使用できます。

サンプル・プログラムは、オブジェクトおよびソース・コード・データ・セット DFSISCO0 として出荷されます。オブジェクト・コードが入っているロード・ライブラリーは IMS.ADFSLOAD です。ソース・コードが入っているソース・ライブラリーは IMS.ADFSSRC です。

### インストール手順

サンプル・プログラムを使用して IMS と CICS の間で ISC 機能を実行するには、事前にいくつかのステップを実行する必要があります。

#### このタスクについて

実行する必要があるステップは、以下のとおりです。

#### 手順

1. サンプル COBOL プログラムをコンパイルし、バインドする。
2. システム定義プロシージャを使用して、IMS に対してトランザクション・コードを定義する。
3. システム定義プロシージャを使用して、IMS と CICS 間に ISC セッションを定義する。
4. 必要な書式をオンライン・ライブラリーに入れるために、MFS ユーティリティー・ステートメントをコンパイルする。
5. まず PSB を生成し、次に ACB を生成する。
6. IMS システムを初期設定する。
7. IMS と CICS 間にセッションを設定する。
8. 対応する CICS サンプル・プログラムを実行する。

#### タスクの結果

関連トピックで、ステップ 1 から 5 を実行するためのサンプル・コードが提供されています。

#### 関連資料

##### IMS サンプル・プログラム (DFSISCO0)

COBOL で書かれている以下の IMS サンプル・プログラムは、CICS との ISC セッションから入力メッセージを受け入れ、同じメッセージを CICS へ戻します。

##### サンプル・プログラムのジョブ制御ステートメント

サンプル・プログラムをコンパイルし、バインドするには、以下の JCL を使用してください。

##### IMS システム定義ステートメント

以下のステートメントを使用して、サンプル・プログラムがサポートする 2 つのトランザクション・コードを IMS に対して定義してください。

##### MFS 形式

以下のステートメントは、サンプル・プログラムで使用される MFS 形式を生成するために必要な JCL および MFS ステートメントから構成されています。

##### サンプル・プログラムのプログラム仕様ブロック (PSB) 生成

以下の JCL およびプログラム仕様ブロック (PSB) 生成ステートメントは、サンプル・プログラムを実行するために必要な IMS 制御ブロックを生成します。

##### アプリケーション制御ブロック (ACB) 生成

以下の JCL およびアプリケーション制御ブロック (ACB) ユーティリティ・ステートメントを使用して、PSB を該当する IMS オンライン・ライブラリーに入れてください。

## IMS サンプル・プログラム (DFSISCO0)

COBOL で書かれている以下の IMS サンプル・プログラムは、CICS との ISC セッションから入力メッセージを受け入れ、同じメッセージを CICS へ戻します。

このプログラムには、SAMPLA1 と SAMPLA2 の 2 つのトランザクション・コードがあります。SAMPLA1 を使用して呼び出されたときには、出力メッセージは MFS によってフォーマット設定されません。SAMPLA2 を使用して呼び出されたときには、分散表示管理 (DPM) を使用すると、出力メッセージは MFS によってフォーマット設定されます。

```
CBL APOST
IDENTIFICATION DIVISION.
PROGRAM-ID. SAMPLA.
AUTHOR:
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER. IBM-370.
OBJECT-COMPUTER. IBM-370.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
DATA DIVISION.
FILE SECTION.
EJECT
WORKING-STORAGE SECTION.
77 BEGIN-LIT PIC X(16) VALUE 'BEGIN 77 ENTRIES'.
77 FILLER PIC X(45) VALUE
'****DATE LAST COMPILED: xxx xx, 19xx****'.
77 CALL-FUNCTION PIC XXXX.
77 MOD-NAME PIC X(08).
EJECT
01 INPUT-AREA.
02 IN-LL PIC S9999 COMP.
02 IN-ZZ PIC S999 COMP.
02 TRAN-CODE PIC X(08).
02 INPUT-DATA PIC X(79).
02 INPUT-DATA-D REDEFINES INPUT-DATA.
05 INPUT-DATA-1 OCCURS 3 TIMES
PIC X(20).
05 INPUT-DATA-2 PIC X(19).
01 OUTPUT-AREA.
02 OUT-LL PIC S9999 COMP.
02 OUT-ZZ PIC S999 COMP.
02 OUTPUT-DATA PIC X(79).
01 OUTPUT-AREA-1.
02 OUT-LL-1 PIC S9999 COMP.
02 OUT-ZZ-1 PIC S999 COMP.
02 OUTPUT-DATA-1 PIC X(20).
EJECT
01 DLI-FUNCTIONS.
02 GET-UNIQ PIC XXXX VALUE 'GU '.
02 GET-NEXT PIC XXXX VALUE 'GN '.
02 ISRT PIC XXXX VALUE 'ISRT'.
EJECT
01 STATUS-ERROR-SEG.
02 FILLER PIC S999 COMP VALUE +83.
02 FILLER PIC S999 COMP VALUE +0.
02 FILLER PIC X(28) VALUE
'***** STATUS ERROR *****'.
02 FILLER PIC X(10) VALUE ' TRANCODE: '.
02 ERROR-TRAN PIC X(8).
02 FILLER PIC X(17) VALUE
' STATUS RECEIVED: '.
02 ERROR-STATUS PIC XX.
02 FILLER PIC X(10) VALUE ' FUNCTION: '.
02 ERROR-FUNCTION PIC XXXX.
EJECT
LINKAGE SECTION.
01 IOTP-PCB.
02 IOTP-LTERM PIC X(8).
02 FILLER PIC XX.
02 IOTP-STATUS PIC XX.
02 IOTP-PREFIX.
03 IOTP-DATE PIC S9(7) COMP-3.
03 IOTP-TIME PIC S9(7) COMP-3.
```

```

03 IOTP-MSG-NUMBER PIC S999 COMP.
03 FILLER PIC XX.
02 IOTP-MOD-NAME PIC X(8).
EJECT

PROCEDURE DIVISION.
ENTRY 'DLITCBL' USING IOTP-PCB.
100-RETRIEVE-MESSAGE-SEGMENT.
MOVE GET-UNIQ TO CALL-FUNCTION.
CALL 'CBLTDLI' USING CALL-FUNCTION IOTP-PCB INPUT-AREA.
IF IOTP-STATUS = 'QC'
    GO TO 800-GOBACK-ROUTINE.
IF IOTP-STATUS NOT = SPACES
    GO TO 700-INVALID-STATUS-CODE.
200-CHECK-TRAN-CODE.
*****
* THE ONLY DIFFERENCE BETWEEN THESE TWO TRANSACTIONS *
* IS THE ABSENCE OR PRESENCE OF MFS. SAMPLA1 DOES NOT *
* CONTAIN MFS. SAMPLA2 CONTAINS MFS. *
*****
IF TRAN-CODE = 'SAMPLA2'
    PERFORM 400-SAMPLA2-ROUTINE
        THRU 450-SAMPLA2-ROUTINE-EXIT,
ELSE
    PERFORM 300-SAMPLA1-ROUTINE
        THRU 350-SAMPLA1-ROUTINE-EXIT.
GO TO 100-RETRIEVE-MESSAGE-SEGMENT.
EJECT

300-SAMPLA1-ROUTINE.
MOVE ISRT TO CALL-FUNCTION.
SUBTRACT 8 FROM IN-LL GIVING OUT-LL.
MOVE IN-ZZ TO OUT-ZZ.
MOVE INPUT-DATA TO OUTPUT-DATA.
CALL 'CBLTDLI' USING CALL-FUNCTION IOTP-PCB OUTPUT-AREA.
IF IOTP-STATUS NOT = SPACES
    GO TO 700-INVALID-STATUS-CODE.
350-SAMPLA1-ROUTINE-EXIT.
EXIT.
400-SAMPLA2-ROUTINE.
MOVE 'MODA' TO MOD-NAME,
MOVE +0 TO OUT-ZZ-1.
MOVE 24 TO OUT-LL-1.
MOVE INPUT-DATA-1 (1) TO OUTPUT-DATA-1.
PERFORM 500-INSERT-ROUTINE
    THRU 550-INSERT-ROUTINE-EXIT.
SUBTRACT 20 FROM IN-LL.
IF IN-LL IS LESS THAN 13 GO TO 450-SAMPLA2-ROUTINE-EXIT.
MOVE INPUT-DATA-1 (2) TO OUTPUT-DATA-1.
PERFORM 600-INSERT-ROUTINE
    THRU 650-INSERT-ROUTINE-EXIT.
SUBTRACT 20 FROM IN-LL.
IF IN-LL IS LESS THAN 13 GO TO 450-SAMPLA2-ROUTINE-EXIT.
MOVE INPUT-DATA-1 (3) TO OUTPUT-DATA-1.
PERFORM 600-INSERT-ROUTINE
    THRU 650-INSERT-ROUTINE-EXIT.
SUBTRACT 20 FROM IN-LL.
IF IN-LL IS LESS THAN 13 GO TO 450-SAMPLA2-ROUTINE-EXIT.
MOVE INPUT-DATA-2 TO OUTPUT-DATA-1.
PERFORM 600-INSERT-ROUTINE
    THRU 650-INSERT-ROUTINE-EXIT.
450-SAMPLA2-ROUTINE-EXIT.
EXIT.
EJECT
500-INSERT-ROUTINE.
MOVE ISRT TO CALL-FUNCTION.
CALL 'CBLTDLI' USING CALL-FUNCTION IOTP-PCB OUTPUT-AREA-1
MOD-NAME.
IF IOTP-STATUS NOT = SPACES
    GO TO 700-INVALID-STATUS-CODE.
550-INSERT-ROUTINE-EXIT.
EXIT.
600-INSERT-ROUTINE.
MOVE ISRT TO CALL-FUNCTION.
CALL 'CBLTDLI' USING CALL-FUNCTION IOTP-PCB OUTPUT-AREA-1.
IF IOTP-STATUS NOT = SPACES
    GO TO 700-INVALID-STATUS-CODE.
650-INSERT-ROUTINE-EXIT.
EXIT.
700-INVALID-STATUS-CODE.
MOVE CALL-FUNCTION TO ERROR-FUNCTION.
MOVE IOTP-STATUS TO ERROR-STATUS.

```

```
MOVE TRAN-CODE TO ERROR-TRAN.  
MOVE ISRT TO CALL-FUNCTION.  
CALL 'CBLTDLI' USING CALL-FUNCTION IOTP-PCB STATUS-ERROR-SEG.  
800-GOBACK-ROUTINE.  
GOBACK.
```

## 関連タスク

### インストール手順

サンプル・プログラムを使用して IMS と CICS の間で ISC 機能を実行するには、事前にいくつかのステップを実行する必要があります。

## 関連資料

### サンプル・プログラムのジョブ制御ステートメント

サンプル・プログラムをコンパイルし、バインドするには、以下の JCL を使用してください。

### IMS システム定義ステートメント

以下のステートメントを使用して、サンプル・プログラムがサポートする 2 つのトランザクション・コードを IMS に対して定義してください。

### MFS 形式

以下のステートメントは、サンプル・プログラムで使用される MFS 形式を生成するために必要な JCL および MFS ステートメントから構成されています。

### サンプル・プログラムのプログラム仕様ブロック (PSB) 生成

以下の JCL およびプログラム仕様ブロック (PSB) 生成ステートメントは、サンプル・プログラムを実行するために必要な IMS 制御ブロックを生成します。

### アプリケーション制御ブロック (ACB) 生成

以下の JCL およびアプリケーション制御ブロック (ACB) ユーティリティ・ステートメントを使用して、PSB を該当する IMS オンライン・ライブラリーに入れてください。

## サンプル・プログラムのジョブ制御ステートメント

サンプル・プログラムをコンパイルし、バインドするには、以下の JCL を使用してください。

この JCL は、z/OS COBOL 2.4 コンパイラーを使用し、結果のロード・モジュールを IMS.PGMLIB に入れます。

```
//IMSCOBOL JOB ACCT,NAME,CLASS=A,MSGLEVEL=(1,1),MSGCLASS=A  
//          PROC MBR=,PAGES=60,RGN=512K,  
//          SOUT=A  
//C          EXEC PGM=IKFCBL00,REGION=&RGN,  
//          PARM='SIZE=130K,BUF=10K,LINECNT=50,APOST,BATCH'  
//SYSLIN DD   DSN=&&LIN,DISP=(MOD,PASS),UNIT=SYSDA,  
//          DCB=(USER.PROCLIB),  
//          SPACE=(3520,(40,10),RLSE,,ROUND)  
//SYSPRINT DD SYSOUT=&SOUT,DCB=(LRECL=121,BLKSIZE=605,RECFM=FBA),  
//          SPACE=(605,(&PAGES.0,&PAGES),RLSE,,ROUND)  
//SYSUT1 DD  UNIT=SYSDA,DISP=(,DELETE),  
//          SPACE=(3520,(100,10),RLSE,,ROUND)  
//SYSUT2 DD  UNIT=SYSDA,DISP=(,DELETE),  
//          SPACE=(3520,(100,10),RLSE,,ROUND)  
//SYSUT3 DD  UNIT=SYSDA,DISP=(,DELETE),  
//          SPACE=(3520,(100,10),RLSE,,ROUND)  
//SYSUT4 DD  UNIT=SYSDA,DISP=(,DELETE),  
//          SPACE=(3520,(100,10),RLSE,,ROUND)  
//L          EXEC PGM=IEWL,REGION=&RGN,PARM='XREF,LET,LIST',  
//          COND=(4,LT,C)  
//          DSN=IMS.SDFSRESL,DISP=SHR  
//*STEPLIB DD DSN=IMS.SDFSRESL,DISP=SHR  
//SYSLIB DD  DSN=SYS1.COBLIB,DISP=SHR  
//SDFSRESL DD DSN=IMS.SDFSRESL,DISP=SHR  
//SYSLIN DD  DSN=&&LIN,DISP=(OLD,DELETE),VOL=REF=*C.SYSLIN  
//          DD DSN=IMS.PROCLIB(CBLTDLI),DISP=SHR  
//          DD DDNAME=SYSIN  
//SYSLMOD DD DSN=IMS.PGMLIB(&MBR),DISP=SHR  
//SYSPRINT DD SYSOUT=&SOUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=605),  
//          SPACE=(605,(&PAGES.0,&PAGES),RLSE,,ROUND)  
//SYSUT1 DD  UNIT=(SYSDA,SEP=(SYSLMOD,SYSLIN)),DISP=(,DELETE),  
//          SPACE=(3520,(100,10),RLSE,,ROUND)  
//          PEND  
//IMSCOBOL EXEC IMSCOBOL,SOUT=A,MBR=SAMPLA  
//C.SYSIN DD *
```

サンプル・プログラムを IMS.ADFSLOAD から直接 IMS.PGMLIB にバインドすることもできます。

## 関連タスク

### インストール手順

サンプル・プログラムを使用して IMS と CICS の間で ISC 機能を実行するには、事前にいくつかのステップを実行する必要があります。

## 関連資料

### IMS サンプル・プログラム (DFSISCO0)

COBOL で書かれている以下の IMS サンプル・プログラムは、CICS との ISC セッションから入力メッセージを受け入れ、同じメッセージを CICS へ戻します。

### IMS システム定義ステートメント

以下のステートメントを使用して、サンプル・プログラムがサポートする 2 つのトランザクション・コードを IMS に対して定義してください。

### MFS 形式

以下のステートメントは、サンプル・プログラムで使用される MFS 形式を生成するために必要な JCL および MFS ステートメントから構成されています。

### サンプル・プログラムのプログラム仕様ブロック (PSB) 生成

以下の JCL およびプログラム仕様ブロック (PSB) 生成ステートメントは、サンプル・プログラムを実行するために必要な IMS 制御ブロックを生成します。

### アプリケーション制御ブロック (ACB) 生成

以下の JCL およびアプリケーション制御ブロック (ACB) ユーティリティ・ステートメントを使用して、PSB を該当する IMS オンライン・ライブラリーに入れてください。

## IMS システム定義ステートメント

以下のステートメントを使用して、サンプル・プログラムがサポートする 2 つのトランザクション・コードを IMS に対して定義してください。

**制約事項:** 以下のコードで指定されている DPM-Bn は、MFS 形式の DEV ステートメントで指定された DPM-Bn に一致する必要があります。

```
APPLCTN PSB=SAMPLA
  TRANSACT CODE=SAMPLA1,INQ=(YES,NORECOV)
  TRANSACT CODE=SAMPLA2,INQ=NO
  TYPE UNITYPE=LUTYPE6,OPTIONS=(TRANRESP,OPNDST,NOMTMSG,          X
    SYNCSESS),                                                    X
    MSGDEL=SYSINFO,                                              X
    OUTBUF=256,                                                  X
    SEGSIZE=256
*
**CICSA1 PARALLEL SESSION NODE
*
  TERMINAL NAME=CICSA1,                                          X
    SESSION=5,                                                  X
    COMPT1=(SINGLE1,DPM-B1,IGNORE),                               X
    COMPT2=(SINGLE2,DPM-B1,IGNORE),                               X
    COMPT3=(MULT1,DPM-B1,IGNORE),                                X
    COMPT4=(MULT2,DPM-B1,IGNORE)
  VTAMPOOL
*
  SUBPOOL NAME=PS01
    NAME PSLT01,COMPT=1,ICOMPT=1
    NAME PSLT02,COMPT=2,ICOMPT=2
    NAME PSLT03,COMPT=3,ICOMPT=3
    NAME PSLT04,COMPT=4,ICOMPT=4
```

## 関連タスク

### インストール手順

サンプル・プログラムを使用して IMS と CICS の間で ISC 機能を実行するには、事前にいくつかのステップを実行する必要があります。

## 関連資料

### IMS サンプル・プログラム (DFSISCO0)



COBOL で書かれている以下の IMS サンプル・プログラムは、CICS との ISC セッションから入力メッセージを受け入れ、同じメッセージを CICS へ戻します。

#### サンプル・プログラムのジョブ制御ステートメント

サンプル・プログラムをコンパイルし、バインドするには、以下の JCL を使用してください。

#### MFS 形式

以下のステートメントは、サンプル・プログラムで使用される MFS 形式を生成するために必要な JCL および MFS ステートメントから構成されています。

#### サンプル・プログラムのプログラム仕様ブロック (PSB) 生成

以下の JCL およびプログラム仕様ブロック (PSB) 生成ステートメントは、サンプル・プログラムを実行するために必要な IMS 制御ブロックを生成します。

#### アプリケーション制御ブロック (ACB) 生成

以下の JCL およびアプリケーション制御ブロック (ACB) ユーティリティ・ステートメントを使用して、PSB を該当する IMS オンライン・ライブラリーに入れてください。

## MFS 形式

以下のステートメントは、サンプル・プログラムで使用される MFS 形式を生成するために必要な JCL および MFS ステートメントから構成されています。

これは、IMS.ADFSMAC 中のデータ・セット MFSSISCO のコピー・コードとして出荷されます。

```
//MFSUTL JOB ACCT,NAME,CLASS=A,MSGLEVEL=(1,1)
//JOB LIB DD DSN=IMS.SDFSRESL,DISP=SHR
//      PROC RGN=360K,SOUT=A,SNODE=IMSVS,
//          SOR=NOLIB,MBR=NOMBR,PXREF=NOXREF,
//          PCOMP=NOCOMP,PSUBS=NOSUBS,PDIAG=NODIAG,
//          COMPR=NOCOMPRESS,COMPR2=COMPRESS,
//          LN=55,SN=8,DEVCHAR=0
//S1 EXEC PGM=DFSUPAA0,REGION=&RGN,
//     PARM=(&PXREF,&PCOMP,&PSUBS,&PDIAG,&COMPR,
//     'LINECNT=&LN,STOPRC=&SN,DEVCHAR=&DEVCHAR')
//SYSLIB DD DSN=IMS.SDFSMAC,DISP=SHR
//SYSIN DD DSN=&SNODE. &SOR. (&MBR),DISP=SHR
//REFIN DD DSN=IMS.REFERAL,DISP=OLD
//REFOUT DD DSN=IMS.REFERAL,DISP=OLD
//REFRD DD DSN=IMS.REFERAL,DISP=OLD
//SYSTEXT DD DSN=&&TXTPASS,UNIT=SYSDA,
//          SPACE=(CYL,(1,1)),DCB=BLKSIZE=800
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//UTPRINT DD SYSOUT=&SOUT
//SYSPRINT DD SYSOUT=&SOUT,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330)
//SYSUDUMP DD SYSOUT=&SOUT
//SEQBLKS DD DSN=&&BLKS,DISP=(NEW,PASS),
//          UNIT=SYSDA,SPACE=(CYL,(1,1))
//S2 EXEC PGM=DFSUNUB0,REGION=&RGN,
//     PARM='&COMPR2,DEVCHAR=&DEVCHAR',
//     COND=(8,LT,S1)
//SEQBLKS DD DSN=&&BLKS,DISP=(OLD,DELETE)
//UTPRINT DD SYSOUT=&SOUT,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330)
//SYSUDUMP DD SYSOUT=&SOUT
//FORMAT DD DSN=IMS.FORMAT,DISP=OLD
//SYSPRINT DD SYSOUT=&SOUT
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
// PEND
//MFSUTL EXEC MFSUTL,SOUT=A
//S1.SYSIN DD *
PRINT ON,NOGEN
FM TA      FMT
          SPACE 3
          DEV TYPE=DPM-B1,FEAT=IGNORE,DSCA=X'00A0'
          SPACE 2
          DIV TYPE=OUTPUT,OPTIONS=(DPAGE,NODNM)
          SPACE 3
DPAGE01   DPAGE FILL=NULL
PPAGE01   PPAGE
          DFLD 'PAGE'
LPGNO     DFLD LTH=04
          DFLD ' '
          SPACE 2
```

```

DATA01  DFLD  LTH=20
        SPACE 3
        FMTEND
        EJECT
MODA    MSG    TYPE=OUTPUT,SOR=(FMFTA,IGNORE),PAGE=YES
        SPACE 3
        SEG
        SPACE 2
        MFLD  (LPGNO,LPAGE0)
        MFLD  DATA01,LTH=20
        SPACE 3
        MSGEND
        END
/*

```

## 関連タスク

### インストール手順

サンプル・プログラムを使用して IMS と CICS の間で ISC 機能を実行するには、事前にいくつかのステップを実行する必要があります。

### 関連資料

#### IMS サンプル・プログラム (DFSIS00)

COBOL で書かれている以下の IMS サンプル・プログラムは、CICS との ISC セッションから入力メッセージを受け入れ、同じメッセージを CICS へ戻します。

#### サンプル・プログラムのジョブ制御ステートメント

サンプル・プログラムをコンパイルし、バインドするには、以下の JCL を使用してください。

#### IMS システム定義ステートメント

以下のステートメントを使用して、サンプル・プログラムがサポートする 2 つのトランザクション・コードを IMS に対して定義してください。

#### サンプル・プログラムのプログラム仕様ブロック (PSB) 生成

以下の JCL およびプログラム仕様ブロック (PSB) 生成ステートメントは、サンプル・プログラムを実行するために必要な IMS 制御ブロックを生成します。

#### アプリケーション制御ブロック (ACB) 生成

以下の JCL およびアプリケーション制御ブロック (ACB) ユーティリティ・ステートメントを使用して、PSB を該当する IMS オンライン・ライブラリーに入れてください。

## サンプル・プログラムのプログラム仕様ブロック (PSB) 生成

以下の JCL およびプログラム仕様ブロック (PSB) 生成ステートメントは、サンプル・プログラムを実行するために必要な IMS 制御ブロックを生成します。

```

//PSBGEN EXEC PSBGEN,SOUT=A,MBR=SAMPLA
//C.SYSIN DD *
        PSBGEN LANG=COBOL,PSBNAME=SAMPLA
        END
/*

```

## 関連タスク

### インストール手順

サンプル・プログラムを使用して IMS と CICS の間で ISC 機能を実行するには、事前にいくつかのステップを実行する必要があります。

### 関連資料

#### IMS サンプル・プログラム (DFSIS00)

COBOL で書かれている以下の IMS サンプル・プログラムは、CICS との ISC セッションから入力メッセージを受け入れ、同じメッセージを CICS へ戻します。

#### サンプル・プログラムのジョブ制御ステートメント

サンプル・プログラムをコンパイルし、バインドするには、以下の JCL を使用してください。

#### IMS システム定義ステートメント

以下のステートメントを使用して、サンプル・プログラムがサポートする 2 つのトランザクション・コードを IMS に対して定義してください。

#### MFS 形式

以下のステートメントは、サンプル・プログラムで使用される MFS 形式を生成するために必要な JCL および MFS ステートメントから構成されています。

#### アプリケーション制御ブロック (ACB) 生成

以下の JCL およびアプリケーション制御ブロック (ACB) ユーティリティ・ステートメントを使用して、PSB を該当する IMS オンライン・ライブラリーに入れてください。

## アプリケーション制御ブロック (ACB) 生成

以下の JCL およびアプリケーション制御ブロック (ACB) ユーティリティ・ステートメントを使用して、PSB を該当する IMS オンライン・ライブラリーに入れてください。

```
//ACBGEN JOB ACCT,NAME,CLASS=A,MSGLEVEL=(1,1)
//ACBGEN EXEC ACBGEN,SOUT=A
//G.SYSIN DD *
BUILD PSB=SAMPLA
BUILD PSB=SAMPLB
/*
```

### 関連タスク

#### インストール手順

サンプル・プログラムを使用して IMS と CICS の間で ISC 機能を実行するには、事前にいくつかのステップを実行する必要があります。

### 関連資料

#### IMS サンプル・プログラム (DFSISCO0)

COBOL で書かれている以下の IMS サンプル・プログラムは、CICS との ISC セッションから入力メッセージを受け入れ、同じメッセージを CICS へ戻します。

#### サンプル・プログラムのジョブ制御ステートメント

サンプル・プログラムをコンパイルし、バインドするには、以下の JCL を使用してください。

#### IMS システム定義ステートメント

以下のステートメントを使用して、サンプル・プログラムがサポートする 2 つのトランザクション・コードを IMS に対して定義してください。

#### MFS 形式

以下のステートメントは、サンプル・プログラムで使用される MFS 形式を生成するために必要な JCL および MFS ステートメントから構成されています。

#### サンプル・プログラムのプログラム仕様ブロック (PSB) 生成

以下の JCL およびプログラム仕様ブロック (PSB) 生成ステートメントは、サンプル・プログラムを実行するために必要な IMS 制御ブロックを生成します。

---

## 第 8 部 複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC))

これらのトピックでは、複数システム結合機能 (MSC) を紹介します。MSC を使用して、複数の IMS サブシステムを接続することができます。これらのトピックでは、MSC の概要と MSC ネットワークの設計、実装、および管理で必要となる情報を説明します。



## 第 38 章 複数システム結合機能の概要

複数システム結合機能 (MSC) は、ある IMS に入力されたトランザクションを別の IMS で処理することを可能にします。

その場合の応答は、トランザクションを入力した端末、または、他の端末でもどちらにでも戻すことができます。IMS は MSC を使用して、接続された IMS システム間のメッセージ・トラフィックを経路指定し、制御します。

### 複数システム結合機能の概念

MSC には、1 つの IMS のプログラムおよびオペレーターを、接続された IMS システムのプログラムおよびオペレーターにアクセス可能にするという方法で、地理的に分散した IMS システムを接続する機能があります。

サポートされるオペレーティング・システムの組み合わせで稼働する複数 (最大 2036) の IMS システム間で通信が行えます。

MSC により、IMS のスループットをシングル CPU の能力以上に引き出すことも可能です。この拡張は、IMS アプリケーションが IMS システム間で次の 2 つの方法で区分できる場合に可能です。

- 水平区分化: アプリケーションは、データベースの内容が IMS システム間で分割されている複数の IMS 内で実行されます。
- 垂直区分化: アプリケーションは、そのアプリケーションが参照する完全なデータベースが接続している 1 つの IMS 内で実行されます。トランザクションはどの IMS でも発信することができます。

リンクとは、2 つの IMS システム間の接続のことです。各 IMS ごとの IMS システム定義時に、すべてのリンクを定義する必要があります。2 つのリンクのタイプがあります。物理リンクと論理リンクです。

- 物理リンクは、2 つの IMS システム間のアクセス方式接続またはハードウェア接続です。
- 論理リンクは、物理リンクをその物理リンクを利用するトランザクションおよび端末と関連付けるメカニズムです。

物理リンクへの論理リンクの割り当ては、システム定義時に行うか、CREATE MSLINK コマンドを使用して動的に行うことができます。また、IMS タイプ 2 UPDATE MSLINK コマンド、またはタイプ 1/MSASSIGN LINK コマンドのいずれかを使用して、後で動的に定義を更新することができます。

### MSC 物理リンク

物理リンクとは、IMS システムがアクセス方式またはハードウェアを通して相互に接続する方法です。

オンライン IMS システムで **CREATE MSPLINK** コマンドを使用して物理リンクを定義できます。

また、IMS システム定義時に MSPLINK ステージ 1 システム定義マクロを使用して、物理リンクを定義することもできます。

オンライン IMS システムで物理リンクを更新するには、**UPDATE MSPLINK** コマンドを使用します。

タイプ 2 コマンドを使用して作成または更新された物理リンクを、IMS コールド・スタートをまたいで保管するには、その物理リンク定義を IMSRSC リポジトリにエクスポートするか、MSC リソースに対する変更をステージ 1 システム定義マクロの中にコーディングします。

複数システム結合機能 (MSC) ネットワーク内の各 IMS では、最大 1018 の物理リンクが許可されます。

複数システム結合機能は、以下のタイプの物理リンクをサポートします。

#### チャンネル間 (CTC) アダプター

通常、IMS システムが同じデータ・センター内にある場合にのみ使用されます。

CTC アダプターはチャンネル間ハードウェア接続です。CTC 接続タイプを使用する物理リンクには、1 つだけ論理リンクを割り当てることができます。

## メモリー間 (MTM)

IMS システムが同じ論理区画内にある場合に使用されます。

MTM リンクは、同じ論理区画で稼働している IMS サブシステム間のソフトウェア・リンクです。MTM 接続タイプを使用する物理リンクには、1つだけ論理リンクを割り当てることができます。

## TCP/IP

通常、IMS システムが異なるデータ・センターにある場合に使用されます。IMS Connect が、物理リンクの TCP/IP 接続およびプロトコルを管理します。

TCP/IP 接続プロトコルおよびネットワーク・プロトコルは、インターネットで使用されるプロトコルです。IMS Connect が、MSC の TCP/IP 接続およびプロトコルを管理します。MSC と IMS Connect の間の通信は、IMSplex 内の IMS 共通サービス層 (CSL) の構造化呼び出しインターフェース (SCI) コンポーネントによって管理されます。TCP/IP 接続タイプを使用する物理リンクには、複数の論理リンクを割り当てることができます。TCP/IP 物理リンクは常に MSC 帯域幅モードで動作し、他の物理リンク・タイプより幾分大きいバッファ・サイズが必要です。

## VTAM

通常、IMS システムが異なるデータ・センターにある場合に使用されます。

VTAM は、通常、テレプロセシング・メディア接続を使用するアクセス方式です。VTAM 接続タイプを使用する物理リンクには、複数の論理リンクを割り当てることができます。

MSC の観点からは、TCP/IP 物理リンクと VTAM 物理リンクの動作は類似しています。システム定義要件およびバッファ・サイズ要件の相違は別にして、各接続タイプをサポートする IMS コンポーネントと z/OS コンポーネントが、動作上の最も重要な相違を示します。

TCP/IP 物理リンク・タイプと VTAM 物理リンク・タイプは両方とも、1次物理リンク・タイプとして使用することも、何らかの理由で他方のリンク・タイプが故障した場合のバックアップ・リンク・タイプとして使用することもできます。

ネットワーク・トラフィックや接続された 2 つの IMS システム間の距離など、さまざまな要因によって異なりますが、TCP/IP 物理リンクは VTAM 物理リンクよりも優れたパフォーマンスを実現する可能性が高くなります。

以下の図は、物理リンクのタイプを示しています。



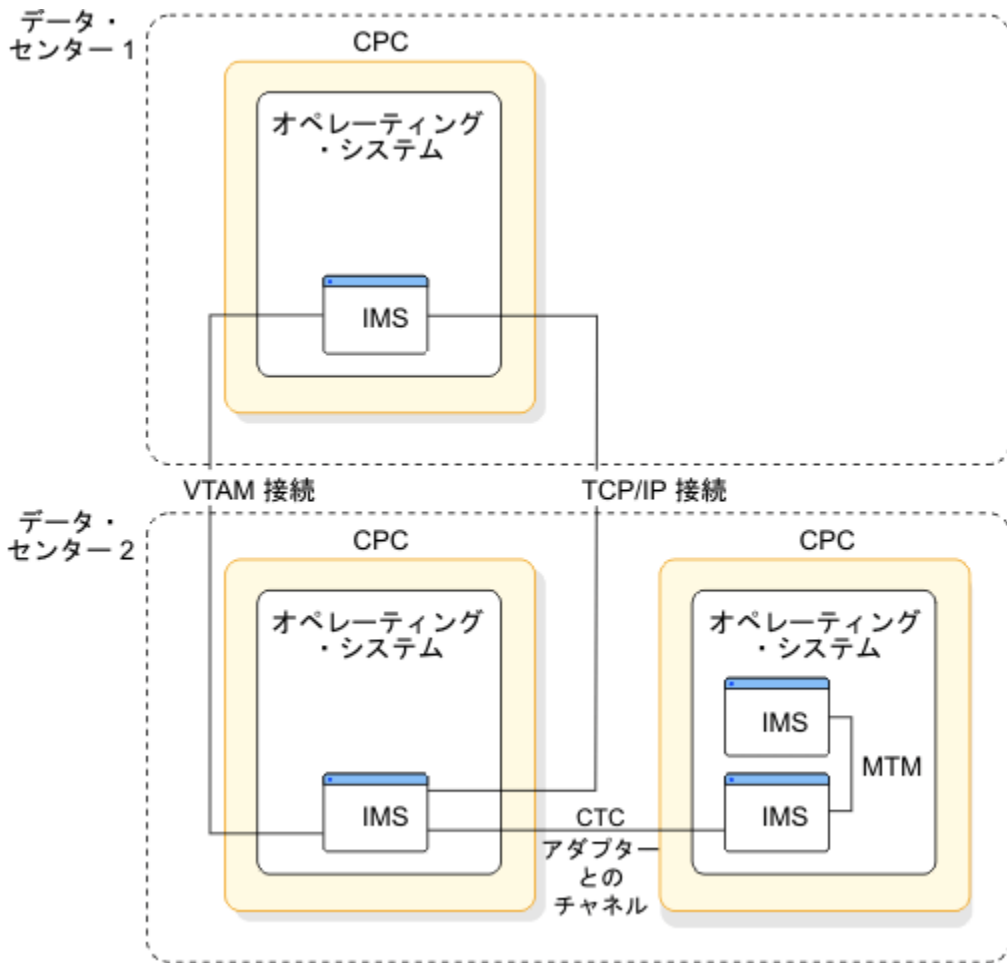


図 111. MSC 物理リンク・タイプ

#### 関連資料

[CREATE MSPLINK コマンド \(コマンド\)](#)

[UPDATE MSPLINK コマンド \(コマンド\)](#)

[QUERY MSPLINK コマンド \(コマンド\)](#)

[MSPLINK マクロ \(システム定義\)](#)

## MSC 論理リンク

論理リンクは、物理リンクをその物理リンクを使用することができるトランザクションおよび端末に関係付けます。

MSC ネットワーク内の各 IMS は、1 つ以上の定義済み論理リンクを持っています。

MSC ネットワーク内の各 IMS に対しては、最大 1018 の論理リンクが許可されます。

それぞれが特定の論理リンクを通して相互に通信するように定義された 2 つの IMS システムは、パートナー・システムと呼ばれます。

2 つの IMS システム間の接続を確立するには、各パートナーが論理リンク定義を持つ必要があります。2 つの論理リンク定義は、同じパートナーの識別を指定し、かつ同じ物理リンクに割り当てられる必要があります。IMS システム定義プロセスでは、定義済みの各論理リンクに番号が割り当てられます。論理リンク番号は、そのリンクが定義される順序で、1 から順番に割り当てられます。論理リンクは、別の物理リンクに再割り当てすることができますが、2 つの IMS システムは、常に論理リンクのパートナー関係を介して通信する必要があります。

MSLINK ステージ 1 システム定義マクロを使用した場合、IMS はデフォルト名を各論理リンクにも割り当てます。ただし、MSLINK マクロでラベルフィールドに別の論理リンク名を指定した場合は除きます。デフ

ォルトの論理リンク名は DFSLxxxx (xxxx は論理リンク番号) です。タイプ 2 コマンドの QUERY MSLINK および UPDATE MSLINK を使用して、論理リンク名を表示または変更できます。

タイプ 2 **CREATE MSLINK** コマンドを使用する場合は、MSPLINK キーワードを指定して、論理リンクを物理リンクに割り当てる必要があります。そのようにしないと、論理リンクは物理リンクに割り当てられません。

タイプ 1 コマンドでは、ターゲット論理リンクを識別するために論理リンク番号を指定する必要があります。タイプ 2 コマンドでは、ターゲット論理リンクを識別するために論理リンク名を指定する必要があります。

タイプ 2 コマンドを使用して作成または更新された論理リンクを、IMS コールド・スタートをまたいで保管するには、その論理リンク定義を IMSRSC リポジトリにエクスポートするか、MSC リソースに対する定義をステージ 1 システム定義マクロの中にコーディングします。

IMSRSC リポジトリを使用して動的に定義された MSC リソースを保管する場合は、MSC リソースに対するコマンドを発行する自動化手順と運用手順で、リンク番号を指定するタイプ 1 コマンドの代わりに、リンク・ネームを指定するタイプ 2 コマンドが使用されていることを確認してください。例えば、**/RSTART LINK 10** コマンドを使用してリンクを開始する代わりに、UPDATE MSLINK NAME(logicallinkname) START (COMM) コマンドを使用します。ステージ 1 のシステム生成時に、IMS システムはリンクが生成された順序で論理リンクに番号を割り当てます。ただし、リンクの番号は IMSRSC リポジトリに保管されません。論理リンクがリンク番号を使用して参照されていて、リンクが IMSRSC リポジトリから自動的にインポートされる場合は、次の IMS コールド・スタート時にリンクの番号が変更される可能性があります。

TCP/IP 物理リンク・タイプおよび VTAM 物理リンク・タイプでは、複数の論理リンクが 1 つの物理リンクを使用することができます。各物理リンクを定義する際に、SESSIONS キーワードを使用して、いくつの論理リンクが 1 つの物理リンクを共用できるかを指定できます。セッション という用語は VTAM に由来しており、通常は論理リンクという用語と同義です。VTAM リンク・タイプの場合、1 つのアクティブ・セッションは、パートナー・システム間の 1 つの論理リンクです。

IMS システム定義プロセスで、論理リンクごとに物理リンクを指定する必要はありません。タイプ 1 IMS コマンド /MSASSIGN LINK またはタイプ 2 IMS コマンド UPDATE MSLINK NAME (linkname)SET (MSPLINK(msplinkname)) を使用して、物理リンクを論理リンクにオンラインで割り当てることができます。パートナー間の通信は、割り当てが行われるまでできません。

#### 関連資料

[CREATE MSLINK コマンド \(コマンド\)](#)

[UPDATE MSLINK コマンド \(コマンド\)](#)

[QUERY MSLINK コマンド \(コマンド\)](#)

[MSLINK マクロ \(システム定義\)](#)

## MSC 論理リンク・パス

メッセージは、論理リンク・パスを使用して、MSC ネットワーク内で経路指定されます。論理リンク・パスは、メッセージを送達する必要のあるリモートの IMS システムと、メッセージの送達元であるローカルの IMS システムを識別します。

入力 IMS システム の端末でトランザクションをサブミットし、宛先 IMS システム でそのトランザクション処理し、入力 IMS システムの端末に応答を返すには、少なくとも 2 つの対応する論理リンク・パスを定義する必要があります。1 つの論理リンク・パスは入力 IMS システムに定義し、もう 1 つは宛先 IMS システムに定義します。

入力 IMS システムに定義される論理リンク・パスでは、入力 IMS システムはローカルとして定義され、宛先 IMS システムはリモートとして定義されます。

宛先 IMS システムに定義される論理リンク・パスでは、宛先 IMS システムはローカルとして定義され、入力 IMS システムはリモートとして定義されます。

論理リンク・パスの定義では、各 IMS システムは、割り当てられているシステム ID (SYSID) (1 から 2036 の数値) のいずれかによって識別されます。

IMS は、入力 IMS システムの SYSID を、入力トランザクション・メッセージとそれに対応する応答に関連付けて、応答を入力端末に返せるようにします。

使用する前に、論理リンク・パスを論理リンクに割り当てておく必要があります。複数の論理リンク・パスを単一の論理リンクに割り当てることができます。

タイプ 2 コマンド CREATE MSNAME を発行するか、MSNAME ステージ 1 システム定義マクロをコーディングすることにより、論理リンク・パスを定義できます。このマクロ名とコマンド名により、MSNAME という用語は、多くの場合、論理リンク・パスという用語の同義語として使用されます。

タイプ 2 コマンドを使用して作成または更新された論理リンク・パスを、IMS コールド・スタートをまたいで保管するには、変更された論理リンク・パス定義を IMSRSC リポジトリにエクスポートするか、MSC リソースに対する変更をステージ 1 システム定義マクロの中にコーディングします。

## 例

次の 2 つの論理リンク・パスの MSNAME マクロ定義を考えてみましょう。

```
MSNAME SYSID=(2,1)
MSNAME SYSID=(3,1)
```

最初の定義は、この論理リンク・パスを使用するメッセージは、ローカル SYSID が 2 のリモート・システムで処理されることを示します。2 番目の定義は、この論理リンク・パスを使用するメッセージは、ローカル SYSID が 3 のリモート・システムで処理されることを示します。これらの定義を使用して、IMS システム定義プロセスは、定義中の IMS に SYSID 1 を割り当て、SYSID 2 と SYSID 3 を持つ 2 つのリモート・システムを認識します。3 番目のパスが SYSID=(5,4) で定義されていた場合には、IMS はローカル・システムに SYSID 4 も割り当てます。

トランザクションは、APPLCTN マクロ定義において論理リンク・パスに割り当てられます。

次の例では、それぞれ 1 つのトランザクション・コードが定義されている、次のようなアプリケーション定義を考えてみましょう。

```
APPLCTN PSB=A
TRANSACT CODE=A
APPLCTN PSB=B,SYSID=(2,1)
TRANSACT CODE=B
APPLCTN PSB=C,SYSID=(3,1)
TRANSACT CODE=C
```

SYSID キーワードは、アプリケーションに関連するトランザクションに使用すべき論理リンク・パスを識別します。トランザクション A はローカル・トランザクションと見なされます。というのは、SYSID キーワードの不在は、トランザクション A が定義中の IMS によってのみ処理されることを示すからです。トランザクション B と C はリモート・トランザクションです。IMS は、アプリケーション定義を MSNAME 定義に関連付けるため、アプリケーション・プログラムで応答の代替宛先を指定していない限り、トランザクション B および C からの応答を SYSID 1 として定義された IMS に戻します。

ローカル・システムで発信されるメッセージがリモート・システム内の論理端末を参照する場合、直接経路指定が使用されるのでない限り、論理リンク定義には、それらのリモート論理端末を識別する NAME マクロも含める必要があります。

## 関連資料

[CREATE MSNAME コマンド \(コマンド\)](#)

[MSNAME マクロ \(システム定義\)](#)

## MSC ネットワークおよび経路指定

以下のトピックで、MSC の管理に必要な概念について説明します。

### リモート・システムとローカル・システム

MSC ネットワークでは、ローカル・システムとは、メッセージが入力される特定の IMS を指します。その他のすべての IMS システムは、特定のローカル・システムに対してはリモート・システム と見なされま

す。ローカル・トランザクション は、同じ IMS で入力され処理されるトランザクションです。リモート・トランザクション とは、端末またはリンクから IMS に入力され、その IMS では処理されないトランザクション のことです。

以下の図は、ローカル・トランザクションとリモート・トランザクション およびローカル・システム とリモート・システムを示します。トランザクション A が IMS A (ローカル・システム) に入力されると、トランザクション A はローカルで処理されます。トランザクション A が IMS B (今度はこれがローカル・システム) に入力されると、それは MSC リンクを越えてリモート・システム IMS A に伝送され、リモートで処理されます。

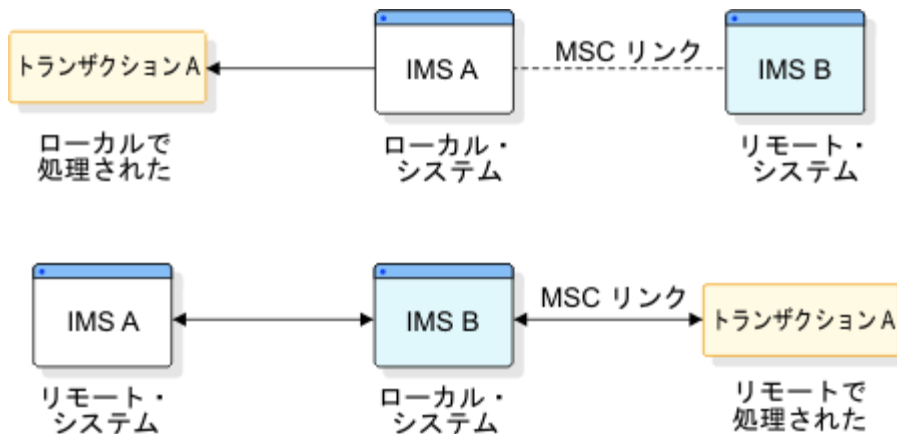


図 112. リモート・トランザクションとローカル・トランザクションおよびリモート・システムとローカル・システム

上記の図で、トランザクション A が IMS A から入力され、IMS A で処理される場合、トランザクション A はローカル・トランザクションと見なされます。トランザクション A が IMS B から入力され、MSC リンクを通して送信され、IMS A で処理される場合、リモート・トランザクションと見なされます。

### 複数システム内のデータ・フロー

MSC ネットワーク内でのトランザクションの流れには、1つの IMS と比較して、追加のステップが必要となります。

一般的な手順を以下の図に示します。図の説明については以下を参照してください。

- ローカル・システムにおいて、LTERM から入力されたリモート・トランザクションは、リモート・トランザクション名の宛先を持つローカル・システムのメッセージ・キューに置かれます **(1)**。(そのメッセージは、指定したリモート宛先と関連した MSNAME のキューに入れられます。)
- MSC はメッセージ・キューからメッセージを取り出し **(2)**、MSC リンクを通してそのメッセージを送信し **(3)**、リモート・システムのメッセージ・キューに置きます **(4)**。
- リモート・システムは、そのメッセージをメッセージ・キューからアプリケーション・プログラムに送信し、処理します **(5)**。アプリケーション・プログラムがメッセージを処理した後、プログラムが返信を送信します。
- リモート・システムは、その応答メッセージに出力 LTERM の宛先を付け、メッセージ・キューに置きます **(6)**。

- MSC は、リモート・システムのメッセージ・キューからメッセージを取り出し (7)、MSC リンクを通してそのメッセージを送信します (8)。
- MSC は、ローカル・システムのメッセージ・キューにそのメッセージを置き (9)、出力 LTERM に送信します (10)。

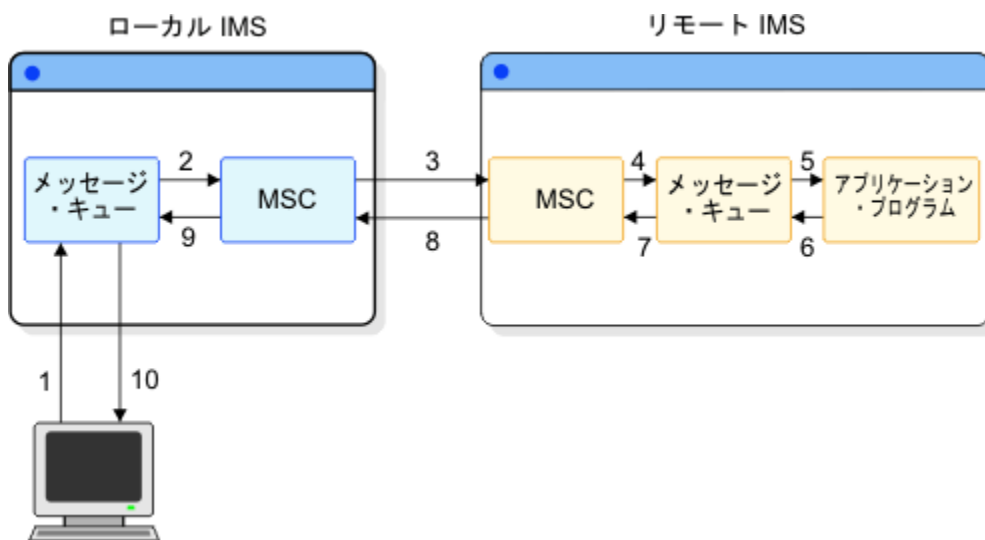


図 113. リモート・トランザクションの流れ

## メッセージ経路指定

MSC のメッセージ経路指定機能は、いくつかのタイプのメッセージ経路指定をサポートしています。

MSC がサポートするメッセージ経路指定のタイプには以下のものがあります。

- ある IMS 内の端末から別の IMS のアプリケーション・プログラムへのトランザクション・メッセージの経路指定。トランザクションは、リカバリー可能、リカバリー不能、応答モード、または会話型として定義することができます。

**制約事項:** MSC リンクを越える高速機能トランザクションはサポートされていません。ただし、ローカル・システム内の高速機能トランザクションはサポートされています。

- ある IMS 内の LTERM から別の IMS 内の LTERM へのメッセージ通信の経路指定および、同一 IMS における LTERM 間のメッセージ通信。このサポートは、**/BROADCAST** コマンド指定により送信されたメッセージを含みます。
- アプリケーション・プログラムから、トランザクションを送信する端末への応答メッセージの経路指定、あるいはアプリケーション・プログラムから代替端末へのメッセージ。代替リモート端末 (トランザクションと LTERM が異なる IMS システム内にある) へメッセージを経路指定するには、代替 LTERM をリモート LTERM として定義する必要があります。直接経路指定または TM および MSC メッセージ経路指定および制御ユーザー出口ルーチンが使用されている場合には、代替 LTERM をリモートとして定義する必要はありません。
- 異なる IMS システム内または同一 IMS 内のトランザクション間でのプログラム間通信。

## 経路指定パス

IMS はメッセージをローカル・システムから経路指定パス上のリモート・システムに渡します。1つ以上の IMS システムを、1つの経路指定パス内に組み込むことができます。

以下の図では、IMS B は、IMS A との間に往復のパス (パス BA) を持っています。同様に、往復のパスが、IMS B と IMS D の間 (パス BD)、IMS B と IMS C の間 (パス BC)、および IMS A と IMS D の間 (パス AD) に存在します。パスは、IMS C と IMS D の間のパス CAD のように、ある IMS を通って別の IMS に到達することができます。同じ 2 つの IMS システム間に複数のパスが存在することができます。IMS C と IMS D には、2 つの直接パス CD1 と CD2 のほか、間接パス CAD があります。

MSC ネットワーク内では、合計で 2036 個のパスが認められています。



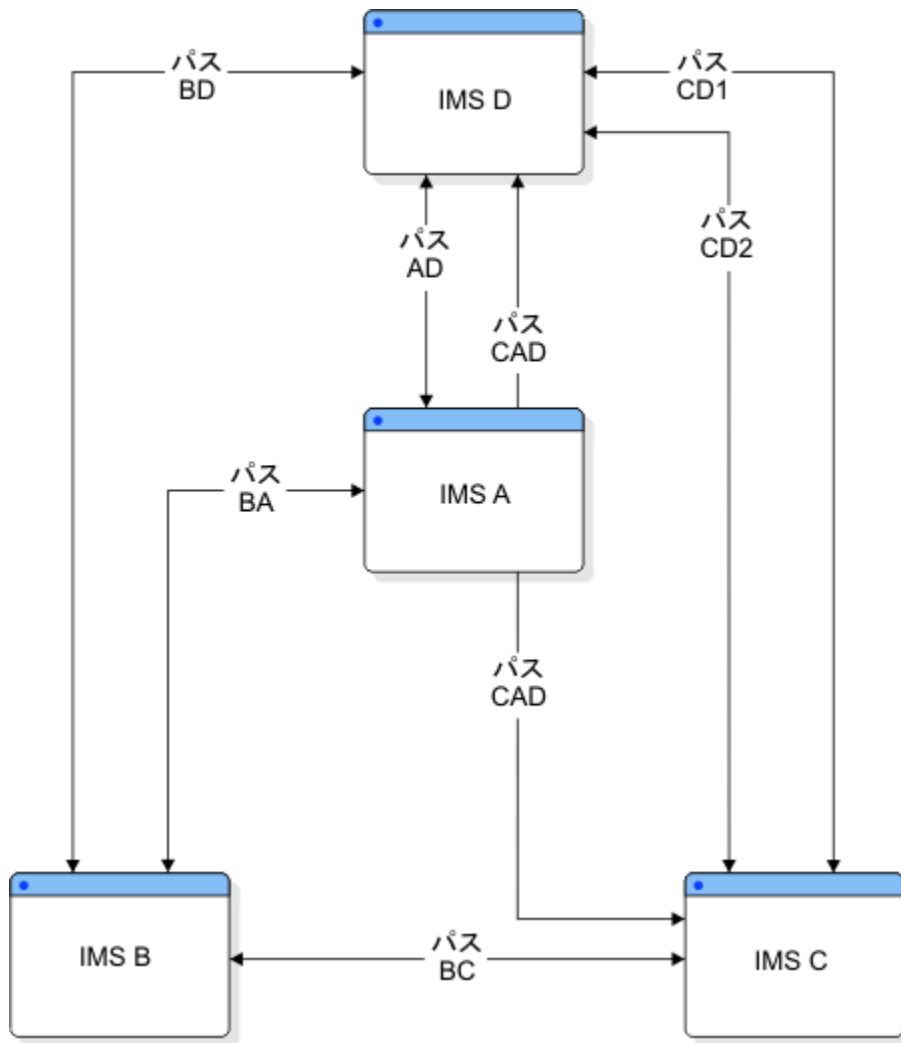


図 114. 経路指定パス

## 論理宛先

MSC ネットワーク用のメッセージ経路指定は、単一システム環境の場合と同じように論理宛先を使用します。

宛先は、LTERM または トランザクション・コードのいずれかです。ローカル宛先はローカル・システムに常駐し、リモート宛先はリモート・システムに常駐します。各ローカル・システム内で、すべてのローカルとリモートの宛先は、固有な名前前で定義される必要があります。以下の図では、すべてのローカル宛先とリモート宛先は、各ローカル・システム内で固有に定義されています。リモートに定義される宛先は、同一 IMS 内でローカルにも定義されることはありません。同様に、ローカルに定義される宛先は、リモートには定義されません。

MSC で接続されている異なる IMS システムで、同一の宛先名をローカル宛先として使用することができます。宛先名が、MSC ネットワーク内の宛先の全体的な意図と矛盾してはいけません。例えば、以下の図では、TRANAB は IMS B と IMS A ではローカル・トランザクションで、IMS C ではリモート・トランザクションです。IMS C は、ローカル TRANAB を IMS A でのみ参照し、IMS B では参照しません。IMS C は、IMS B においてリモートで TRANAB を参照することはできません。リモート宛先で参照される宛先システムは、システム識別 (SYSID) 値により決まります。

直接経路指定および TM/MSC メッセージ経路指定 / 制御ユーザー出口ルーチンを使用すると、アプリケーション・プログラムからのメッセージを、宛先システムで明示的に定義されていないリモート宛先に送信することができます。トランザクション処理システムにおける宛先の妥当性検査は、メッセージがローカル・システムに着信するまで遅延されます。この妥当性検査の遅延により、リソース名を決める要件がより柔軟になります。

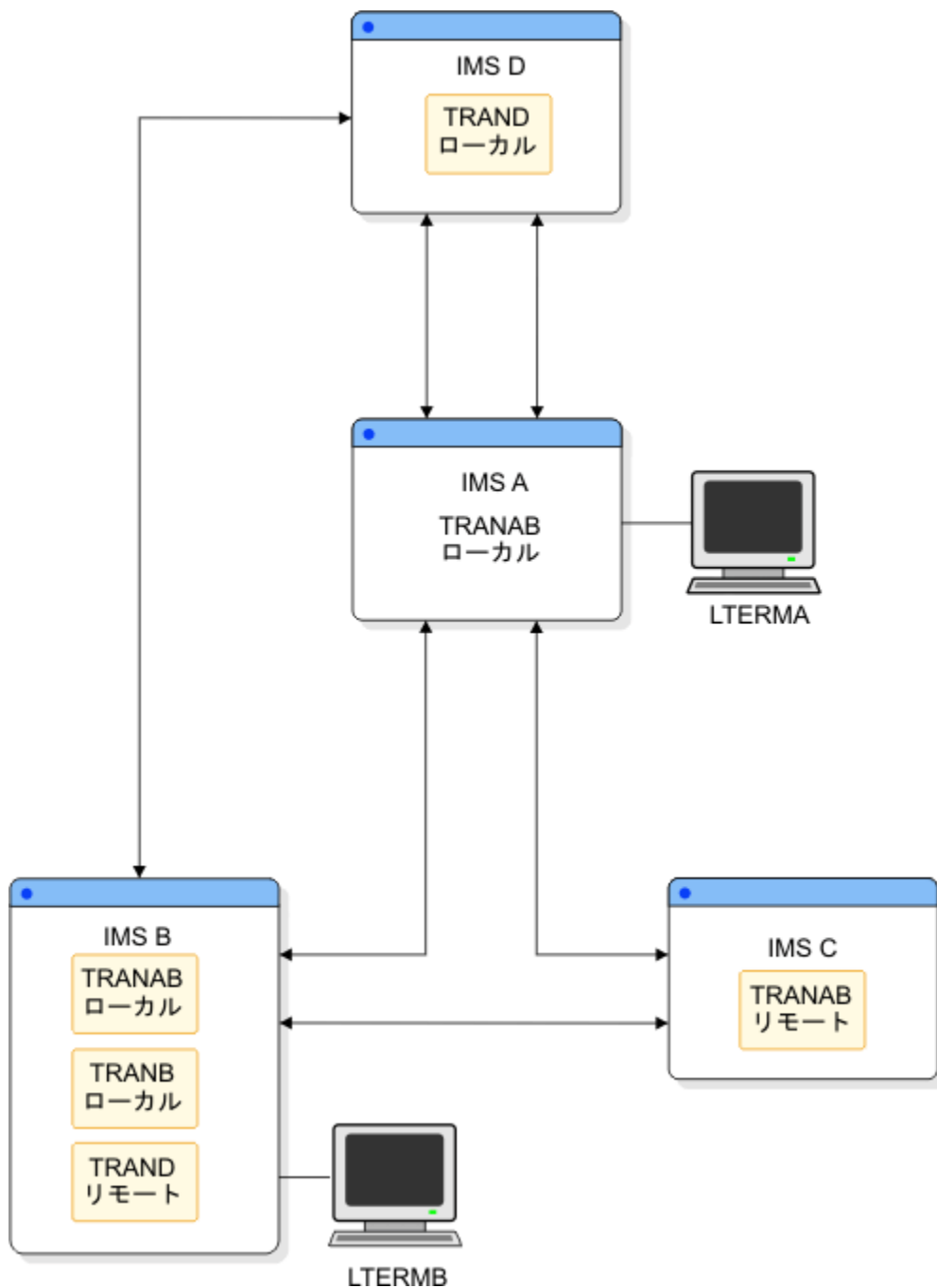


図 115. 論理宛先

#### 関連概念

693 ページの『システム ID (SYSID)』

MSC は、システム ID (SYSID) (1 から 2036 までの 2 バイトの数字) を使用して、MSC ネットワークの IMS システムを識別します。

699 ページの『複数システム結合機能 (MSC) 直接経路指定』

MSC 直接経路指定は MSC の機能の 1 つであり、これを使用すると、アプリケーション・プログラムは、LTERM またはアプリケーション・プログラムへのメッセージに対し、IMS 名 (MSNAME) とその IMS 内の宛先を指定することができます。

#### 関連資料

「TM および MSC メッセージ経路指定および制御」ユーザー出口ルーチン (DFSMSCEO) (出口ルーチン)



## 入力システム、宛先システム、および中間システム

メッセージにより、IMS は、入力システム、宛先システム、または中間システムのいずれかに することができます。

例えば、以下の図では、トランザクション・メッセージは IMS B の LTERMB から発信され、パス BD を通じて IMS D の TRAND に経路指定されます。IMS B が入力システムで、IMS D が宛先システムです。このメッセージは、基本メッセージと呼ばれます。メッセージが IMS D のアプリケーション・プログラムによって処理され、応答が IMS B の入力端末に送り返された場合、この応答メッセージは、応答と呼ばれます。この応答メッセージの場合、宛先と入力システムは IMS B です。メッセージの宛先は IMS B 内の LTERMB です。この応答メッセージが出される原因となった入力も、同様に、IMS B 内の LTERMB (基本トランザクション) から行われています。

メッセージが、ある IMS を通じて送信され、処理のために別の IMS に直接回送される場合には、その経路指定システムは、中間システムと呼ばれます。例えば、TRAND が IMS B から IMS A を経由して IMS D (パス BAD) に送信される場合、IMS A は中間システム、IMS D は宛先システムになります。同様に、TRANC が IMS B から IMS D と IMS A を経由して IMS C (パス BDAC) に送信される場合には、IMS A と IMS D は TRANC の中間システムとなります。

リモート宛先名は入力システムではリモートとして、そして宛先システムでは ローカルとして定義します。ただし、中間システムでは、回送されるメッセージの入力、宛先名、または戻りパス(ローカル SYSID)を定義する必要はありません。宛先システムへのパスだけが、定義を必要とします。戻りパスは、中間システムを通らない場合もあります。メッセージが中間システムを経由して回送される場合には、メッセージの入力および宛先名は検査されません。

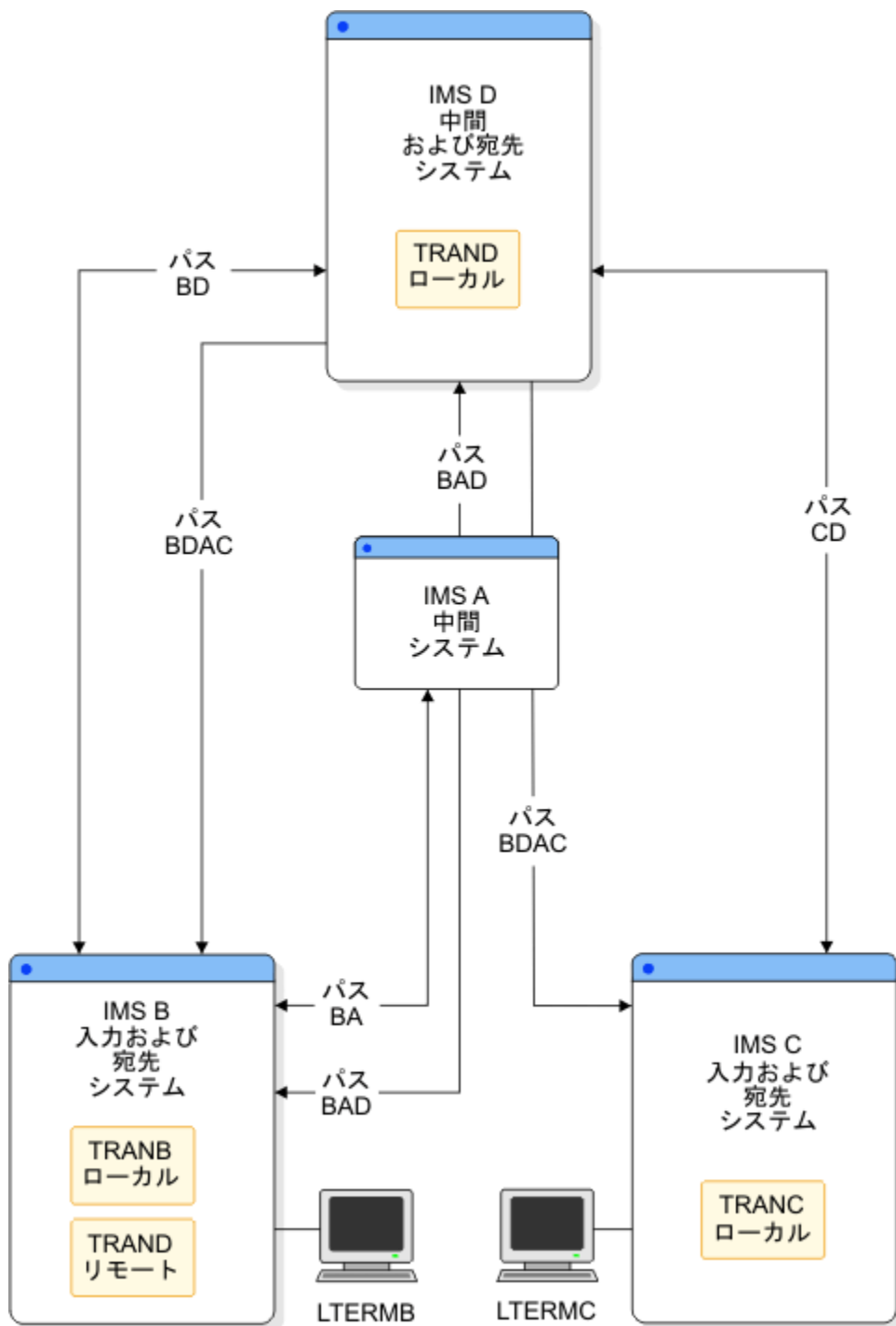


図 116. 入力システム、宛先システム、および中間システム

## システム ID (SYSID)

MSC は、システム ID (SYSID) (1 から 2036 までの 2 バイトの数字) を使用して、MSC ネットワークの IMS システムを識別します。

MSC ネットワーク内の各 IMS システムには、1 つ以上の SYSID が割り当てられます。SYSID は、割り当て先の IMS システムにローカルであり、MSC ネットワーク内の他の IMS システムに対してリモートです。

ローカル SYSID は、MSC ネットワーク内で固有でなければなりません。共有キューを持つ IMSplex に MSC ネットワークを接続すると、共有キュー・グループ内の IMS システムは、ローカル SYSID を共有できます。

ローカル SYSID とリモート SYSID をペアにして、論理リンク・パスを定義します。論理リンク・パスは、ローカル SYSID とリモート SYSID が表す 2 つの IMS システム間の論理リンクに割り当てられます。MSNAME という用語は、多くの場合、論理リンク・パスという用語の同義語として使用されます。

メッセージが MSC ネットワークを通過して伝送されるたびに、ローカル SYSID はメッセージの起点を識別し、リモート SYSID はメッセージの宛先を識別します。一般に、リモート SYSID によって識別される IMS システムは、トランザクション・メッセージが処理される場所であり、ローカル SYSID は、トランザクションに対する応答を返す必要がある場所です。ただし、リモート SYSID は、トランザクションを処理せずにネットワーク内の別の IMS に渡す中間 IMS システムを表すこともあります。

ローカル SYSID は、次の方法で IMS システムに割り当てることができます。

- MSNAME ステージ 1 システム定義マクロ
- IMS PROCLIB データ・セットの DFSDFxxx メンバーの MSC セクション
- IMS タイプ 2 コマンド **CREATE MSNAME**

IMS タイプ 2 コマンド **UPDATE MSNAME NAME(msname) SET(SIDL|SIDR)** を使用して、オンラインで SYSID を変更できます。

以下の図では、IMS B はローカル SYSID 1、2 および 3 を持ちます。IMS A はローカル SYSID 4 を持ちます。IMS C はローカル SYSID 5 を持ちます。IMS D はローカル SYSID 6 および 7 を持ちます。ローカル SYSID が 1、2、および 3 である 3 つの MSNAME マクロがあるため、IMS B はローカル SYSID 1、2、および 3 を持ちます。

この図では、IMS B は、リモート SYSID 6 (IMS D)、7 (IMS D)、および 5 (IMS C) への 3 つのパスを持っています。IMS D と IMS B は、SYSID 4 (IMS A) へのパスを持っていないため、IMS A にメッセージを送ることはできません。ただし、IMS A には、SYSID 7 (IMS D) および SYSID 2 (IMS B) へのパスがあります。IMS A は、IMS A で発信されたメッセージを SYSID 7 または SYSID 2 に送信することはできません。というのは、ソース SYSID (SYSID 4) は IMS B または IMS D では認識されないからです。この構成では、IMS A は、IMS B および IMS D の中間システムとしてしか機能しません。IMS B を IMS A および IMS D と通信させるためには、別のパス (MSNAME) を定義する必要があります。

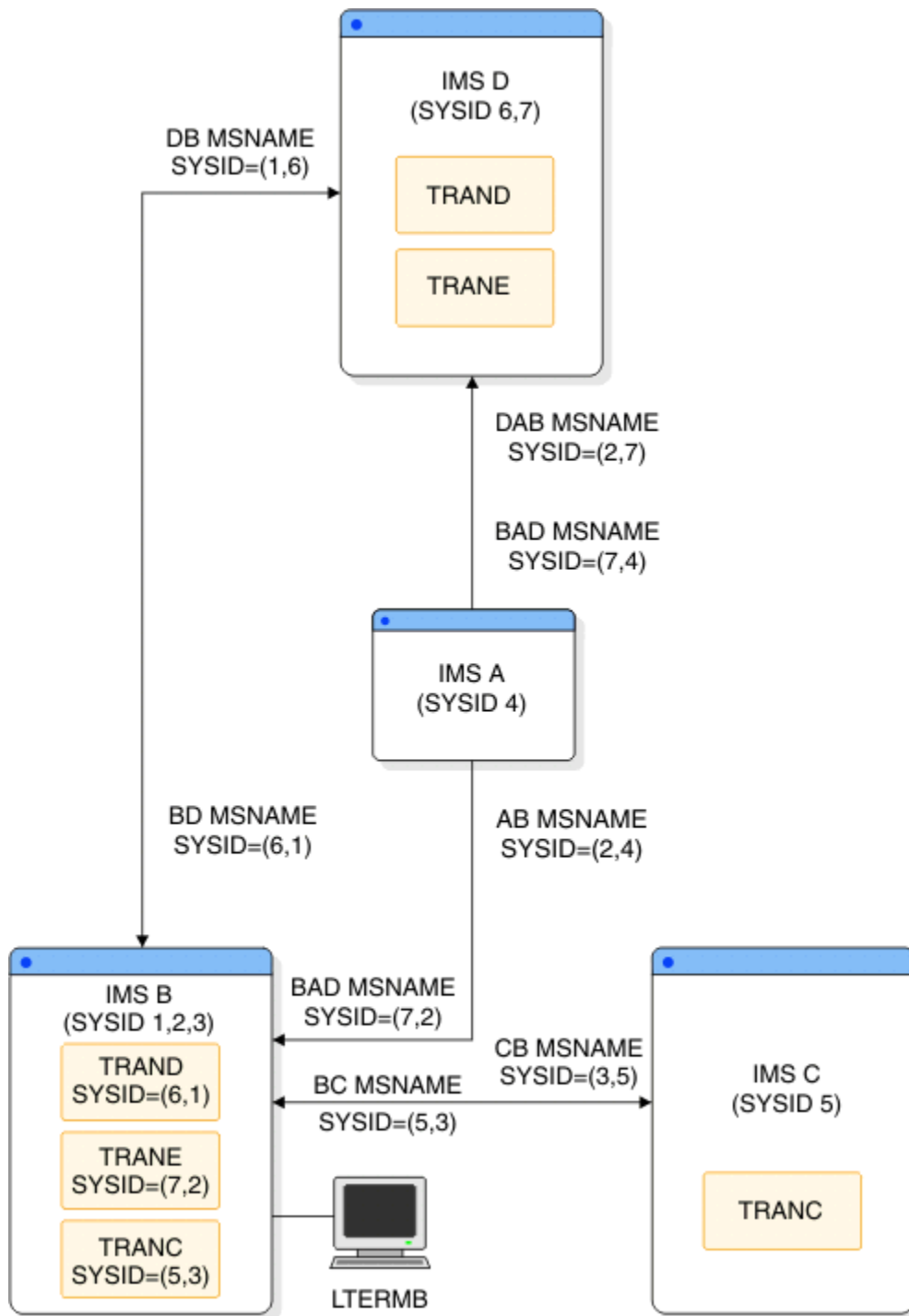


図 117. システム ID (SYSID)

### 関連概念

686 ページの『MSC 論理リンク・パス』

メッセージは、論理リンク・パスを使用して、MSC ネットワーク内で経路指定されます。論理リンク・パスは、メッセージを送達する必要があるリモートの IMS システムと、メッセージの送達元であるローカルの IMS システムを識別します。

### 関連タスク

709 ページの『SYSID の定義』

システム ID (SYSID) は、1 から 2036 までの 2 バイトの番号で、MSC ネットワークの IMS システムを識別します。

#### 関連資料

[MSNAME マクロ \(システム定義\)](#)

[DFSDFxxx メンバーの MSC セクション \(システム定義\)](#)

[CREATE MSNAME コマンド \(コマンド\)](#)

## 宛先名および SYSID によるメッセージの経路指定

MSC ネットワーク内のメッセージには、IMS システム間でメッセージの経路指定を可能にする情報が含まれています。

リモート・トランザクションが LTERM から入力される場合には、トランザクション・メッセージが作成され、メッセージ・キューに入れられます。このメッセージには、そのメッセージをリモート宛先に経路指定するのに必要な情報が入っています。

- リモート宛先名 (トランザクション・コード)
- トランザクションを入力するローカルまたはソースの LTERM 名
- リモート SYSID 値
- ローカルまたはソース SYSID 値

トランザクションがアプリケーション・プログラムで処理され、2 次メッセージが他のトランザクションに送信される (プログラム間通信) 場合、これらのメッセージは、宛先名と通信先トランザクションの SYSID を持ちます。そのソース (発信元) 名と SYSID は変わりません。すなわち、ソース SYSID と名前はどんな場合にも変わりません。このため、そのトランザクションがどこで処理されるかに関係なく、応答メッセージを容易に入力 LTERM に送り返すことができます。

リモート・システムから受け取ったメッセージをローカルに処理する IMS は、そのメッセージのリモート SYSID と同一のローカル SYSID をシステムに定義する必要があります。また、メッセージのソース (発信元) に戻りパスを持つ必要があります。以下の図では、TRANA が IMS B の LTERMB から入力されると、それはパス MSNAME=(4,2) を通して IMS A に送信されます。メッセージの宛先名および SYSID は TRANA および 4 です。ソース名および SYSID は LTERMB および 2 です。IMS A がメッセージを受け入れて処理します。これはローカルとして定義された SYSID=4 を持ち、宛先 SYSID=2 の IMS B への戻りパスを持っています。IMS A の TRANA が IMS D の TRAND に対してプログラム間通信を出す場合、宛先名および SYSID は TRAND および 5 です。ソース名および SYSID は LTERMB および 2 のままです。IMS D がトランザクションを受け入れて処理します。このトランザクションは、ローカルに定義された SYSID=5 と、宛先 SYSID=2 の IMS B への戻りパスを持ちます。

IMS D の TRAND が入力 LTERMB に応答を送り返す場合、応答メッセージは LTERMB と 2 の宛先名と SYSID を持ち、ソース名と SYSID も LTERMB および 2 になります。

入力 LTERM である LTERMB が、IMS A または IMS D でリモートとして定義されることはありませんが、応答メッセージは IMS B の LTERMB に戻されます。IMS は、発信元の LTERM 名と SYSID を 1 次メッセージおよびすべての 2 次メッセージに入れて運びます。IMS は、アプリケーション・プログラムが入力 LTERM に応答する場合に、応答メッセージをどの経路で返すか認識しています。

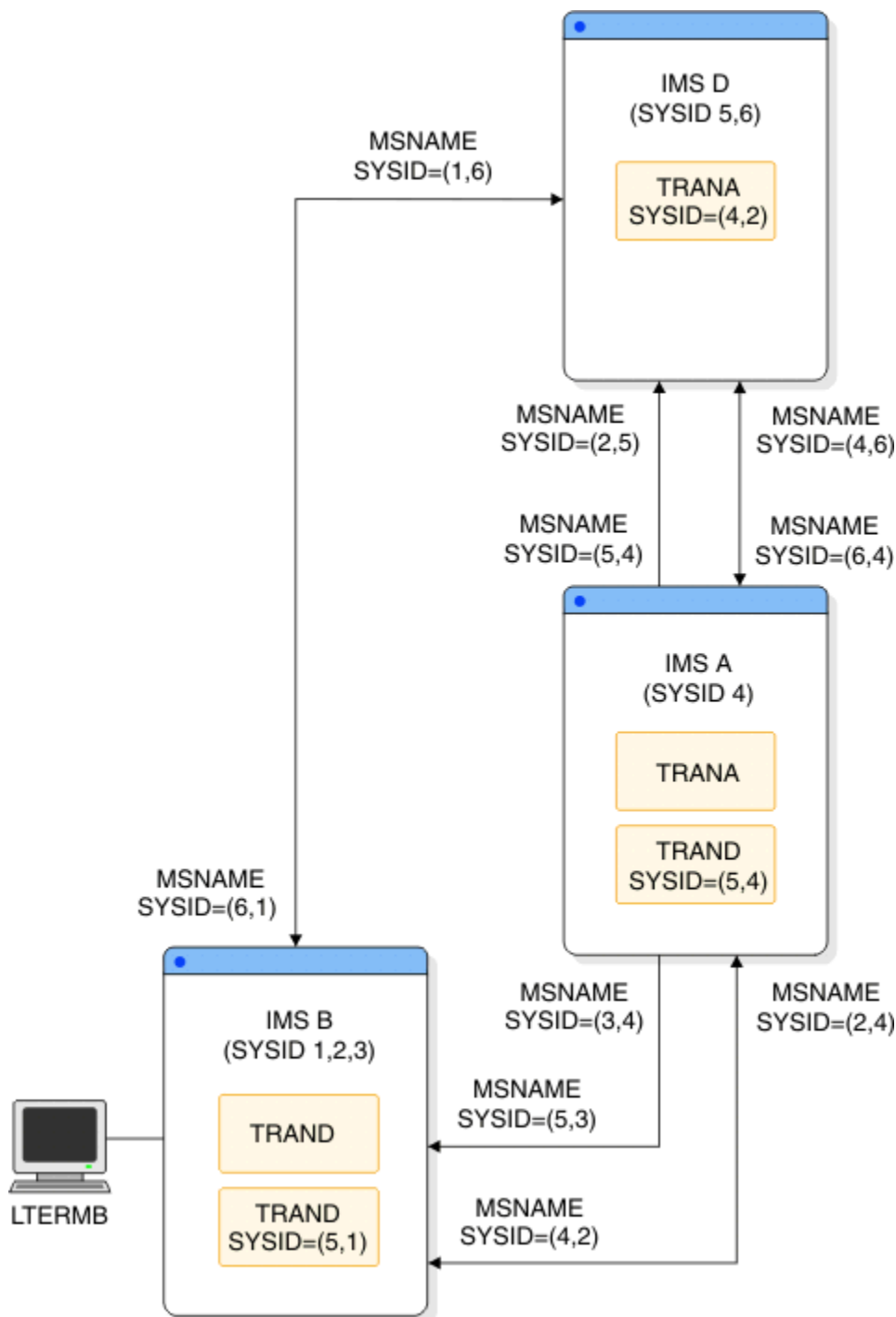


図 118. メッセージ経路指定

## リモート LTERM

リモート *LTERM* とは、ローカル・システムに常駐しない論理端末のことです。

**推奨事項:** リモート IMS システムでは、メッセージ通信、リモート・ブロードキャスト、またはメッセージをアプリケーション・プログラムから代替リモート端末 (代替 PCB) に送信する場合にのみ、ローカル IMS システムから入力 LTERM をリモート LTERM として定義してください。それ以外の場合、IMS は入力 LTERM を記憶し、リモート・システムに入力 LTERM を定義しなくてもリモート・システムから入力 LTERM に応答メッセージを返すことができます。

ETO MSC 記述子を使用すれば、リモート LTERM を定義することができます。ETO MSC 記述子は、リモート LTERM を静的に定義された MSC リンクと関係付けます。

タイプ 2 CREATE LTERM コマンドを使用して、リモート LTERM を動的に定義することもできます。

応答メッセージを入力(ソースまたは発信元) LTERM に戻すために、IMS はソース LTERM 名と SYSID をリモート・メッセージに入れて運びます。ソース LTERM 以外の LTERM に応答メッセージを送信するには、リモート LTERM を定義する必要があります。

- NAME マクロを使用する場合は、リモート・システムにおいて、ローカル・システムのローカル端末の名前を持つ NAME マクロを定義します。この NAME マクロを、ローカル・システムの宛先 SYSID を定義する MSNAME と関連付けてください。
- CREATE LTERM コマンドを使用してリモート・システム内にリモート LTERM を動的に作成する場合は、ローカル・システムのローカル端末と同じ名前を持つリモート LTERM 名を定義します。このリモート LTERM を、ローカル・システムの宛先 SYSID を定義している論理リンク・パス (MSNAME) と関連付けてください。

例えば、以下の図では、IMS A はローカル LTERMA を持ち、IMS B はローカル LTERMB を持ち、そして IMS D はローカル LTERMD を持ちます。LTERMD は IMS B ではリモートで定義されているので、IMS B は LTERMB から LTERMD へのメッセージ通信およびリモート・ブロードキャストを送信することができます。LTERMD は実際には SYSID=(5,2) で定義されてはいませんが、メッセージが出される時点でこれらの SYSID を想定します。LTERMD は、MSNAME BAD (IMS B において SYSID=(5,2) で定義される) と関連付けられます。この関連は、LTERMD に関する NAME マクロを次の MSNAME マクロの後に置くことによって設定されます。

```
BAD MSNAME SYSID=(5,2)
NAME LTERMD
```

この命名は、リモートおよびローカル SYSID で明示的に定義される リモート・トランザクション定義とは異なります。

さらに、IMS A の LTERMA は、メッセージ通信またはリモート・ブロードキャストを IMS D の LTERMD に送信することができます。同様に、IMS D の LTERMD は、メッセージ通信またはリモート・ブロードキャストを IMS A の LTERMA に送信することができます。IMS A と IMS D はともに、他方のシステムのローカル LTERM についてのリモート LTERM 指定を持っています。

IMS D の TRAND は、代替メッセージを IMS B の LTERMB または IMS A の LTERMA に送信することができます。MSNAME は、以下の目的を表す名前前で定義されます。

- その名前の付いた論理リンク・パスに対するキューの数を表示する。
- 会話を継続しているメッセージを除くすべてのメッセージを、端末から送信するのを停止する。
- 論理リンク・パスを開始する。
- 入力停止された MSC ネットワークにおいて論理リンク・パスを除去する。
- アプリケーション・プログラムが直接経路指定を使用することを許可する。
- タイプ 1 コマンドの /MSASSIGN MSNAME *msname* LINK *link#* またはタイプ 2 コマンドの UPDATE MSNAME NAME(*msname*) SET(MSLINK(*linkname*)) のいずれかを使用して、それらを異なるリンクに再度割り当てる。

以下の図で、各 MSNAME の名前は、MSNAME の前の 2 文字の名前です。



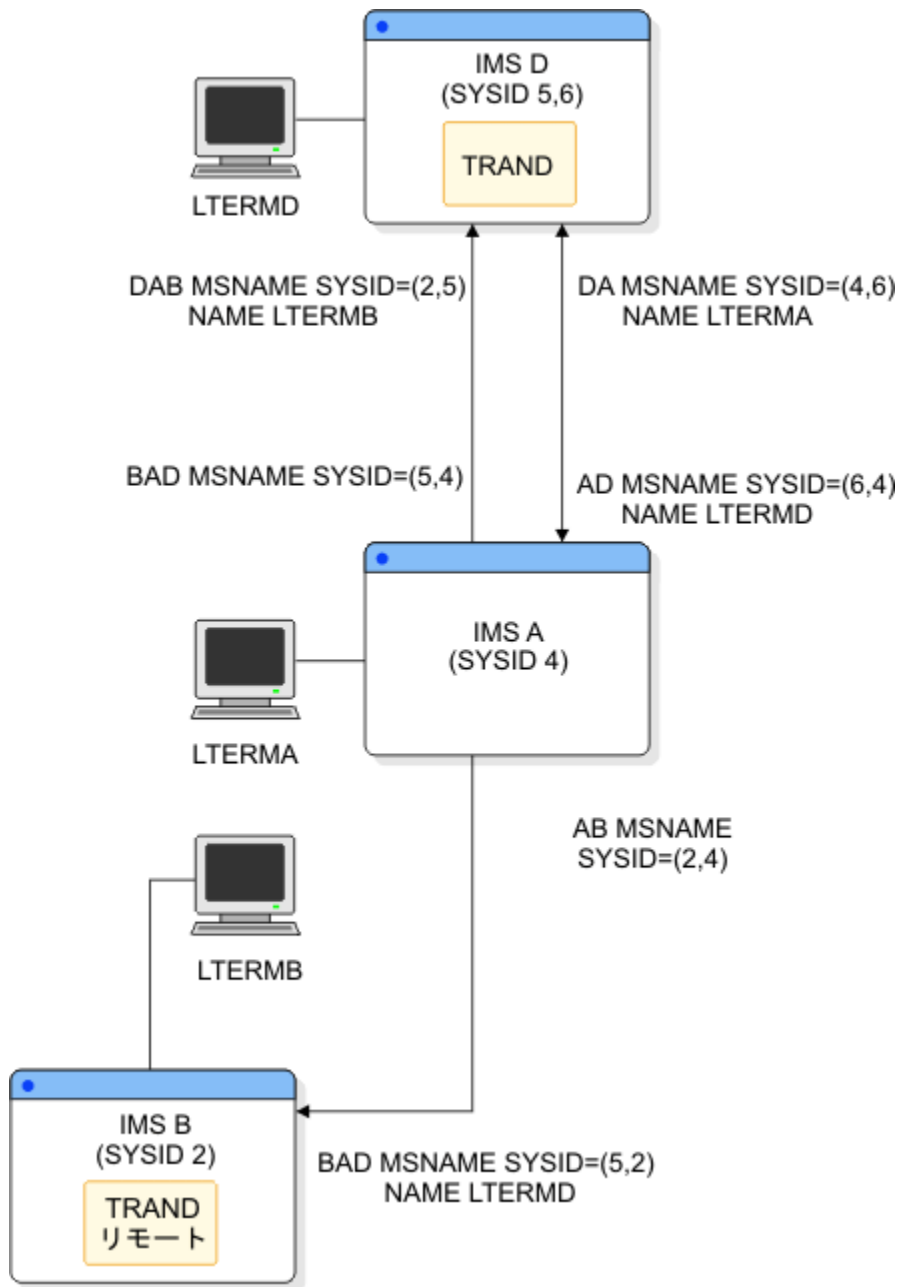


図 119. リモート LTERM

#### 関連資料

[MSC 記述子の形式とパラメーター \(システム定義\)](#)

[CREATE LTERM コマンド \(コマンド\)](#)

## 複数システム結合機能 (MSC) 直接経路指定

MSC 直接経路指定は MSC の機能の 1 つであり、これを使用すると、アプリケーション・プログラムは、LTERM またはアプリケーション・プログラムへのメッセージに対し、IMS 名 (MSNAME) とその IMS 内の宛先を指定することができます。

受信側アプリケーション・プログラムは、最初にスケジュールした IMS の MSNAME を判別することができます。直接経路指定により、別の IMS 内の指定済みリモート宛先 (トランザクションまたは LTERM) は、送信 IMS の IMS システム定義内で明示的に宣言する必要はありません。端末用のこれらの論理 (ローカル) 名は、MSC ネットワーク内の異なる IMS システムが、端末とトランザクション・コード用に同じ論理名を使用することを可能にします。名前は、指定された IMS 内でこれまでどおり固有であることが必要で

す。複数システム検査ユーティリティー (DFSUMSV0) では、MSC 直接経路指定に関連するエラーを検出することができません。

#### 制約事項:

- MSC 直接経路指定は、会話型トランザクション間のプログラム間通信をサポートしません。
- MSC 直接経路指定は、非会話型トランザクションから会話型トランザクションへのプログラム間通信をサポートしていません。例えば、システム A における会話型トランザクションは、直接経路指定を使用してプログラム間通信を実行し、システム B における会話型トランザクションを呼び出すことはできません。
- 直接経路指定トランザクションの DL/I ISRT 呼び出しで、応答モードを伝搬することはできません。

#### 関連概念

[複数システム結合機能を使用した他の IMS TM システムとの通信 \(アプリケーション・プログラミング\)](#)

## リモート宛先検査

システム保全性を維持しエラーを防止するために、MSC ネットワーク内の IMS は、MSC 直接経路指定が使用されない限りすべての指定された宛先を検査します。

MSC 直接経路指定が使用される場合、IMS は、プログラム間通信が、非会話型トランザクションから会話型トランザクションに行われていないことを保証するだけです。端末からメッセージを受信したとき、またはリモート宛先がメッセージに指定されている場合には、アプリケーション・プログラム応答を受信次第、リモート宛先検査が行われます。宛先検査は、次のように行われます。

#### 宛先

##### 検査対象

#### LTERM

宛先タイプ: 元の宛先は、論理端末であることが必要です。

#### トランザクション

宛先タイプ: 元の宛先は、トランザクションであることが必要です。

トランザクション属性: 下記の属性は、入力と宛先システムにおいてトランザクション定義が一致していることが必要です。

- 単一セグメントまたは複数セグメント
- リカバリー可能またはリカバリー不能
- 会話型または非会話型
- 照会または更新

リモート・システムからの会話用の SPA サイズは、リモート・システムでの挿入による場合以外変更することはできません。

無効な宛先が認識されると、IMS はメッセージを取り消し、入力端末とローカル・システムのマスター端末にエラー・メッセージを送信し、無効な要求をログに記録します。メッセージが会話型の場合には、会話異常終了出口ルーチン (DFSCONE0) が、入力システムで呼び出されて、会話は終了します。

## 第 39 章 複数システム結合機能の管理

以下のトピックでは、MSC を使用してネットワーク内で複数の IMS オンライン・システムを 接続する際に必要なシステム管理アクティビティについて説明します。

### このタスクについて

LU 6.2 アプリケーション・プログラムにより、リモート IMS システムでトランザクションが 処理されます。また、ETO はリモート LTERM をサポートしています。

#### 関連概念

697 ページの『リモート LTERM』

リモート LTERM とは、ローカル・システムに常駐しない論理端末のことです。

## 複数システムの設計考慮事項

MSC の設計の主要な目標は、システム間に適切な接続を定義してリソース消費を最小化し、システム間で機能を分散してリソースの要求を平衡化して受け入れ可能なパフォーマンスを得て、マルチシステム会話トランザクションについて設計上の考慮事項をプログラムすることです。

単一 IMS システムに適用できる設計とチューニングの推奨事項は、MSC 環境における各 IMS システムに適用できます。MSC 構成の一部であるシステムを定義するときは、リソースの要求と消費量を考慮に入れる必要があります。

ローカル・システムにおいて処理される IMS トランザクションは、非 MSC 環境の場合と同じハードウェアおよびソフトウェア・リソースを使用します。リモート・システムで処理されるトランザクションには、追加のリソースが必要になります。トランザクションを物理リンクを通してリモート・プロセッサに伝送するために使用されるリソースや、リモート・プロセッサから応答を受け取るために使用されるリソースに加え、メッセージのキューイングとロギングのためのリソースが必要になります。パフォーマンスの考慮は、リモート処理によって消費されるリソースの最小化と MSC 構成における、複数のプロセッサ間のリソース要求を平衡化することに直接関係します。

## リソース消費の最小化

MSC システムを設計する際には、さまざまな方法でリソースの消費を最小化することができます。

### このタスクについて

リソース消費を最小化するために、以下の各項目を実行してください。

- できるだけ多くのトランザクションをローカルに処理できるよう、環境を設計する。
- ローカル・システムからリモート・システムに直接行く物理リンクを提供する。トランザクション経路指定過程で中間システムを組み込まないでください。中間システムを通るトランザクションには、追加のプロセッサ・アクティビティ、メッセージ・キュー・アクティビティおよびロギング・アクティビティが必要となります。
- 不要なメッセージ・キュー入出力アクティビティを除去するために、各プロセッサにメッセージ・キュー・バッファ・プールを設ける。
- トランザクション入力メッセージおよび出力応答メッセージを単一のキュー・バッファに保持できるだけの大きさを持つように、キュー・バッファのサイズを設計してください。
- 物理リンクのバッファ・サイズを、メッセージ接頭語に加え、大部分のメッセージのすべてのセグメントを保持可能な十分な大きさに定義する。物理リンク・バッファ・サイズは、物理リンクに割り当てられるすべての論理リンクのデフォルト・バッファ・サイズを定義します。 **UPDATE MSLINK** `NAME(linkname) SET(BUFSIZE(new_bufsize))` を使用して、論理リンクごとに異なるバッファ・サイズを指定することができますが、ある所定の論理リンクのバッファ・サイズの指定は、その論理リンクが接続する両方の IMS システムで同じものにする必要があります。

## MSC リンクの帯域幅の制御

MSC リンクのバッファ・サイズを増やしたり、MSC 帯域幅モードをオン/オフにしたりすることによって、CTC、MTM、およびVTAMの各MSC リンク・タイプの帯域幅を制御できます。

### このタスクについて

TCP/IP 物理リンク・タイプは、常に帯域幅モードで動作します。TCP/IP リンクの場合、帯域幅モードをオフにすることはできません。

帯域幅モードがオフの場合、MSC は 1 回の入出力送信操作または書き込み操作ごとに最大 1 つのメッセージまたは応答を送信します。帯域幅モードがオンである場合、IMS は以下のメッセージおよび応答を同じバッファに集約します。

- キューに入れられ、送信の準備ができていないすべてのメッセージ
- 受信したメッセージに対する応答

リンク・バッファ・サイズを増やすと、同時に送信するメッセージおよび応答の数を増やすことができます。有効な MSC リンク・バッファ・サイズは、1024 バイトから 65536 バイトです。物理リンクは、そのリンクに関連付けられた論理リンクのデフォルト・バッファ・サイズを定義します。

デフォルト値は、システム定義時に MSPLINK マクロで BUFSIZE キーワードを指定することによって設定するか、または、CREATE MSPLINK コマンドで BUFSIZE キーワードを指定することによって動的に設定することができます。その後、UPDATE MSLINK コマンドまたは UPDATE MSPLINK コマンドを使用すると、オンライン IMS システムで論理リンクごとにバッファ・サイズを変更できますが、ある所定の論理リンクについては、その論理リンクの両端にある両方の IMS システムで、同じバッファ・サイズを指定する必要があります。

帯域幅モードおよびリンク・バッファ・サイズを変更するには、タイプ 2 コマンドの UPDATE MSLINK またはタイプ 1 コマンドの /UPDATE MSLINK のいずれかを使用します。帯域幅モードの状況とバッファのサイズを表示するには、タイプ 2 コマンドの QUERY MSLINK またはタイプ 1 コマンドの /DISPLAY LINK OPTION BUFSIZE を使用します。

システム定義時に帯域幅モードを設定することはできません。コマンドで設定する必要があります。帯域幅モードおよびバッファ・サイズを設定した後は、リンクを再始動しても、IMS のウォーム・スタートを実行しても、選択されたモードとサイズが保持されます。IMS のコールド・スタートが実行されると、帯域幅モードはオフに設定され、バッファ・サイズは SYSDEF 値に復元されます。

### 関連資料

[MSPLINK マクロ \(システム定義\)](#)

[CREATE MSPLINK コマンド \(コマンド\)](#)

[UPDATE MSLINK コマンド \(コマンド\)](#)

[UPDATE MSPLINK コマンド \(コマンド\)](#)

[QUERY MSLINK コマンド \(コマンド\)](#)

## リソース要求の平衡化

複数のプロセッサで構成された MSC 環境では、1 つのプロセッサの使用率が極端に高くなるのを避ける意味で、作業負荷を分散するようにしてください。

### このタスクについて

使用可能なプロセッサ間で IMS アプリケーションとその関連トランザクションおよび端末を分散することにより、作業負荷を分散することができます。アプリケーションの複雑さとプロセッサの能力に応じて、1 つのプロセッサへの過負荷を回避できます。

現在のデータベース設計が、データベースと関連アプリケーションを、使用可能なプロセッサ間で分散できないようになっている場合には、次のことが可能です。

- 照会専用データベースの 2 重化。これにより、複数のシステムがデータベース全体を参照できるようになります。(これは垂直区分化と呼ばれます。)

- データベースをいくつかのコンポーネント・データベースに分割する。(これは水平区分化と呼ばれます。)コンポーネント・データベースは、使用可能な複数のプロセッサ間の分散に対し完全に独立していることが必要です。例えば、データベースをキー範囲間隔で分割することが可能です。新規データベースとそれらの関連アプリケーションは、既存の IMS システム間で分散でき、さらに端末経路指定出口ルーチンを使用して、着信トランザクションを正しい IMS システムに経路指定することができます。別の可能性は、データベースを地理的区域で分割することです。IMS システムは、自分の地理的区域のデータベースを参照するトランザクションについては処理を行い、リモートの地理的区域のデータベースを参照するトランザクションについては経路指定することができます。

プロセッサを通しての作業負荷の平衡化に加え、物理リンクの作業負荷も平衡化が必要になる場合があります。これは、2つのシステム間の物理リンクが SDLC タイプである場合や、複数の物理リンクがインストールされている場合に起こります。以下のことにより、物理リンクの作業負荷を平衡化することができます。

- IMS システム定義時に、リモート・アプリケーションごとに適切な論理リンク・パスと論理リンクを指定する。
- ユーザー作成の TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン (DFSMSCEO) を使用して、トランザクション 負荷を各代替物理リンクに分散する。

## 会話型処理の設計

会話型処理は、MSC ネットワークの任意の IMS に接続された端末においても、単一システム環境の場合と同じ程度まで使用可能です。

MSC ネットワーク内の会話型処理には、以下の相違点があります。

- 会話に使用されるすべてのトランザクションは、MSC ネットワークの各 IMS において会話型として定義される必要があります。
- 入力システムは、会話期間中、会話型リソースを制御します。入力システムは、会話型トランザクションを受信するとき、最初のメッセージ・セグメントとしてスクラッチパッド域 (SPA) を挿入し、そのメッセージを宛先アプリケーション・プログラムに回送します。
- MSC ネットワーク内のどのシステムも、会話の任意のステップを処理できます。
- プログラム間通信は、システム間で回送可能です。
- マルチシステム会話で使用される SPA は、以下の規則に従う必要があります。
  - SPA ISRT の場合、会話型プログラム間通信は、同サイズか、大きいまたは小さいサイズの SPA を持つトランザクションを引き起こすことができます。
  - SPA の最小サイズは 16 バイト (X'10')、最大サイズは 32767 バイト (X'7FFF') です。

一般には、端末オペレーターやアプリケーション・プログラムは、会話がマルチシステムであるかどうかには気が付きません。

### 例外:

- 会話型プログラムが、リモート・システムにおける応答代替 PCB にメッセージを挿入すると想定します。暗黙指定により、この宛先は入力システムにあり、入力システムによって検査されます。この場合、宛先検査には、指定された論理端末が入力端末に割り当てられたままであることを確認することができます。論理端末が再割り当てされている場合には、入力システムは会話異常終了出口ルーチンを活動化し、会話を終了します。アプリケーション・プログラムに戻される状況コードは、正常終了を示すブランクです。前述の条件のために実際の結果が正常終了でない場合でも、この状況コードはブランクです。
- 入力システム以外のシステムで実行するアプリケーション・プログラムが、SPA を使用してトランザクション・コードを指定し、それによって会話制御を別のプログラムに渡すとします。指定されたトランザクション・コードが無効である場合、入力システムは会話異常終了出口ルーチンを活動化して、会話を終了します。アプリケーション・プログラムに状況コードは戻されません。

APPC/IMS および OTMA に対する MSC サポートには、以下の IMS トランザクション・タイプが組み込まれています。

### 会話型



非会話型  
応答モード  
非応答モード

**制約事項:** MSC は、高速機能をサポートしません。

## 会話を伴う経路指定出口ルーチン

TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン (DFSMSCEO) のプログラム経路指定エントリー・ポイントおよびリンク経路指定エントリー・ポイントを使用して、会話を開始するメッセージを入力することができます。

### このタスクについて

会話継続のための宛先を提供するのは入力端末ではなくアプリケーション・プログラムであるため、このルーチンは他の会話型ステップには適用できません。

## 会話のリモート宛先検査

プログラム間通信の宛先は、MSC 直接経路指定が使用される場合を除いて、切り替えを要求するプログラムが実行されるシステムで検査されます。

MSC 直接経路指定が使用されず、宛先が有効な場合、システムは、SPA とメッセージを宛先トランザクションに送信します。宛先が無効の場合、システムは、状況コードをアプリケーション・プログラムに返します。

MSC 直接経路指定が使用される場合、IMS は、非会話型トランザクションから会話型トランザクションへのプログラム間通信が実行されていないことだけを保証します。宛先が有効な場合、システムは、SPA とメッセージを宛先トランザクションに送信します。宛先が無効の場合、システムは SPA を経路指定せず、会話は終了します。

入力端末へのメッセージに対する宛先検査は、入力システムによって実行されます。指定された論理端末が入力端末に割り当てられたままになっていなければなりません。入力システムは、MSC 直接経路指定が使用される場合を除いて、SPA に指定された次のトランザクションも検査します。宛先が無効の場合、入力システムは、会話異常終了出口ルーチン呼び出し、会話を終了します。アプリケーション・プログラムに状況コードは戻されません。

## 切り捨てられたデータの SPA への保管

TRANSACT マクロの SPA=STRUNC オプションは、サイズの異なる SPA を持つ他のトランザクションへのプログラム間通信を使用する会話に適用されます。

- SPA=STRUNC オプションを使用すると、より小さいサイズで定義されたトランザクションへのプログラム間通信が行われた場合であっても、IMS はすべてのデータを SPA に保存します。小さい SPA を持つトランザクションは、切り捨てられたデータを参照しません。しかし、このトランザクションがもっと大きな SPA を持つトランザクションに切り替えられると、切り捨てられたデータが使用されます。IMS は、SPA に挿入された最長データを追跡して、切り捨てられたデータの長さを判別します。

**例:** 次の3つのトランザクションがあるとします。

```
TRANA SPA=100  
TRANB SPA=50  
TRANC SPA=150
```

TRANA 向けのアプリケーション・プログラムが TRANB に切り替わると、TRANA 用の SPA の 50 バイトは TRANB に送信されません。その後 TRANB 向けのアプリケーション・プログラムが TRANC に切り替わると、TRANC が受け取る SPA には、以下の 150 バイトが含まれます。

- TRANB によって挿入された SPA の最初の 50 バイト
- 2 番目の 50 バイトは、TRANA によって挿入された 2 番目の 50 バイト
- 3 番目の 50 バイトは 2 進ゼロです。

- SPA=STRUNC オプションを使用しないと、切り捨てられたデータは失われます。上の例では、TRANC が受け取る 2 番目の 50 バイトは 2 進ゼロになります。

**制約事項:** 前のトランザクションからの切り捨てられたデータは、MSC を使用して IMS 5.1 (またはそれ以前の) システムに送信される場合には、失われます。

## 会話の終了

会話は、アプリケーション・プログラムまたは端末オペレーターのいずれかによって終了することができます。

アプリケーション・プログラムは、トランザクション・コードがブランクまたは非会話型トランザクションのトランザクション・コードいずれかを有する SPA と共に入力端末にメッセージを挿入することにより、会話を終了します。

会話型トランザクションは、**/EXIT** コマンドによっても終了することができます。

IMS シャットダウンでは、会話は終了しません。会話は IMS コールド・スタートが起こるか、会話を **/EXIT** のようなオペレーター・コマンドによって終了しない限り、IMS が再始動した後に継続されます。入力システムがシャットダウンされた後、続いてコールド・スタートすると、その入力システムが制御するすべての会話が失われます。リソース・マネージャー (RM) を使用していて、かつ状況リカバリー・モードが GLOBAL のとき、状況が RM に保管されているので、IMS 会話は IMS コールド・スタートの後も失われません。入力システムは、以前にアクティブもしくは保留中であった会話に関連していた入力端末について受信するあらゆる会話型メッセージを取り消します。IMS は、再始動後も会話を保存します。IMS コールド・スタートは正規のプロシーチャーではなく、会話状態を失う以上により重大な問題を引き起こす恐れがあります。

会話型ステップが処理中またはキューに入っているときに、リモート・システムがシャットダウンされ、続いてコールド・スタートすると、会話に対するすべての参照は、失われます。この方法で失われた会話は、**/EXIT** コマンドを用いて入力システムで明確に取り消す必要があります。

## 会話の異常終了

複数のさまざまなイベントにより、会話が異常終了する可能性があります。

下記のいずれか 1 つが起こると、会話は異常終了します。

- 会話型アプリケーション・プログラムが異常終了する。
- 入力システムまたはリモート・システムで (会話型応答、プログラム間通信、または SPA で)、無効な宛先が認識される。
- 会話型メッセージが、会話が終了した端末に挿入される。
- 会話型メッセージの宛先検査に失敗する。
- アプリケーション・プログラムで出力が生成されない。

会話の SPA (終了の原因を示すものと一緒に) は、入力システムの 会話異常終了出口ルーチンに渡されます。

## 複数システム結合リソースの定義

複数システム結合機能 (MSC) リソース (物理リンク、論理リンク、システム ID など) は、IMS の実行時に (IMS タイプ 2 コマンドを使用して) 動的に定義することも、IMS システム定義時に (マクロを使用して) 静的に定義することもできます。

### このタスクについて

MSC リソースを動的に定義するためには、その前に、MSC と、MSC 用の動的定義の両方が IMS システムで使用可能になっている必要があります。MSC は、IMS 始動プロシーチャーに MSC= 実行パラメーターを指定するか、または IMS システム定義時にステージ 1 システム定義マクロを使用して少なくとも 1 つの MSC リンクを定義すると、使用可能になります。MSC の動的定義は、DFSDFxxx PROCLIB メンバーの MSC セクションに MSCRSCS=DYN を指定すると、使用可能になります。



MSC と動的定義が使用可能になると、IMS タイプ 2 コマンド (CREATE MSPLINK、UPDATE MSPLINK、および DELETE MSPLINK) を使用して、MSC リソースを作成、更新、および削除することができます。MSC リソースに対して動的に行われた変更をコールド・スタートをまたいで保管するには、定義を IMSRSC リポジトリにエクスポートするか、MSC リソースに対する変更をステージ 1 システム定義マクロにコーディングします。IMSRSC リポジトリからリソースを削除するには、**DELETE DEFN** コマンドを発行します。これらを行わない場合、動的に行われた MSC リソースに対する変更は、ウォーム・リスタートおよび緊急リスタートの場合にのみ再始動をまたいで保管されます。MSC は、リソース定義データ・セット (RDDS) をサポートしません。

IMS システム定義時に MSC リソースを静的に定義する場合は、MSC 始動パラメーターを指定する必要はありません。ただし、これを指定した方が混乱は防げます。静的定義のみを使用して IMS オンライン・システムで MSC を使用可能にするには、ご使用の IMS システム定義に 3 つのマクロ、すなわち MSPLINK、MSLINK、および MSNAME を含める必要があります。

トランザクション入力または処理に対して、何らかの役割を持つ IMS システムごとにトランザクション・コードを定義する必要もあります。トランザクション・コードは、動的に定義することも、システム定義マクロを使用して定義することもできます。個々のシステムは、いくつかの役割を果たすことができます。それは、入力システムである場合もあれば、経路指定トランザクションに責任を持つ中間システムである場合も、あるいはトランザクションが処理される宛先システムである場合もあります。

MFS の使用法は、MSC ネットワークでも単一システム環境でも同じです。ある IMS において他の IMS に接続された端末に対するメッセージが作られた場合、必要なメッセージと形式記述はその端末が接続されている IMS において使用可能でなければならず、同じ名前を使用した定義がそれぞれの IMS で等しく定義されなければなりません。

## IMS システムでの MSC の使用可能化

IMS システムで複数システム結合機能を使用可能にするには、開始プロシージャで MSC=Y 実行パラメーターをコーディングするか、あるいは、MSC=N が指定されていない限り、IMS システム定義時に少なくとも 1 つの MSC リンクで定義することによって設定できます。

### このタスクについて

**推奨事項:** 動的 MSC が使用可能に設定され MSC リソースがもはやシステム定義プロセスを使用しては定義されなくなった際の操作を容易にするために、MSC リソースがシステム定義プロセスで定義される場合でも MSC=Y 実行パラメーターを指定するようにしてください。

## MSC に対する DRD の使用可能化

MSC リソースの動的定義を使用可能にするには、IMS PROCLIB データ・セットの DFSDFxxx メンバーの MSC セクション内で MSCRSCS=DYN を指定します。

### 始める前に

MSC リソースの動的リソース定義を使用可能にするには、少なくとも Structured Call Interface (SCI) と Operations Manager (OM) を備えた Common Service Layer (CSL) を前もって使用可能にしておく必要があります。

### このタスクについて

MSC リソースの動的リソース定義を使用可能にした後、IMS タイプ 2 の CREATE、UPDATE、DELETE の各コマンドを使用して、オンライン IMS システム内の MSC リソースの作成、変更、および削除を行うことができます。

MSC リソースに加えたすべての動的変更は、コールド・スタートを実行する前に、MSC リソースに対する変更をステージ 1 システム定義マクロ内にコーディングするか、変更を IMSRSC リポジトリにエクスポートした場合を除き、コールド・スタートをまたいで保管されません。

### 手順

動的定義を使用可能にするには、以下の手順を実行します。

1. IMS PROCLIB データ・セットの DFSPBxxx メンバー内で MSC=Y を指定するか、IMS システム 定義時に少なくとも 1 つの MSC リンクを定義することにより、MSC を使用可能にします。
2. IMS PROCLIB データ・セットの DFSDfxxx メンバーの MSC セクションで、MSCRSCS=DYN を指定します。
3. IMS をコールド・スタートします。
4. QUERY MEMBER TYPE(IMS) コマンドを発行して、MSC リソースの動的定義が使用可能であることを確認します。  
MSC リソースの動的リソース定義が使用可能である場合は、コマンド出力のローカル属性に DYNMSC が含まれています。

### 関連タスク

#### 707 ページの『MSC リソースに対する IMSRSC リポジトリの使用可能化』

IMSRSC リポジトリを使用可能にして、IMSplex のすべてのメンバーに対応する単一の中央保管場所すべての MSC リソース定義を保管できます。MSC リソースに対して IMSRSC リポジトリを使用可能にすると、動的に作成および更新された MSC リソースを IMS コールド・スタートの後にも以前のまま保管できます。

#### 706 ページの『IMS システムでの MSC の使用可能化』

IMS システムで複数システム結合機能を使用可能にするには、開始プロシージャで MSC=Y 実行パラメータをコーディングするか、あるいは、MSC=N が指定されていない限り、IMS システム 定義時に少なくとも 1 つの MSC リンクで定義することによって設定できます。

### 関連資料

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

DFSDfxxx メンバーの MSC セクション (システム定義)

## MSC リソースに対する IMSRSC リポジトリの使用可能化

IMSRSC リポジトリを使用可能にして、IMSplex のすべてのメンバーに対応する単一の中央保管場所すべての MSC リソース定義を保管できます。MSC リソースに対して IMSRSC リポジトリを使用可能にすると、動的に作成および更新された MSC リソースを IMS コールド・スタートの後にも以前のまま保管できます。

### 始める前に

MSC リソース定義に対する IMSRSC リポジトリを使用可能にする前に、IMSRSC リポジトリが定義されて使用可能になっていることを確認してください。

### 手順

1. 動的に定義された MSC リソースに対する IMSRSC リポジトリを使用可能にするには、DFSDfxxx proclib メンバーの MSC セクションで MSCREPO=Y を指定します。
2. IMS をコールド・スタートします。

### タスクの結果

動的に定義された MSC リソースに対する IMSRSC リポジトリを使用可能にした後、オンライン・リソースは、IMS のチェックポイント時に IMSRSC リポジトリに自動的にエクスポートされ、IMS のコールド・スタート時に IMSRSC リポジトリから IMS システムに自動的にインポートされます。

### 関連概念

#### 754 ページの『IMSRSC リポジトリ内の MSC リソースの保守』

IMSRSC リポジトリ内で MSC リソース定義を保守することにより、IMSplex のすべてのメンバーに対応する単一の中央保管場所に定義を保管します。また、IMSRSC リポジトリ内で MSC リソース定義を保守することにより、IMS コールド・スタートの後にも以前の定義を保管できます。

### 関連タスク

IMSRSC リポジトリの定義 (システム定義)

## 関連資料

[DFSDFxxx メンバーの MSC セクション \(システム定義\)](#)

[DFSDFxxx メンバーの DYNAMIC\\_RESOURCES セクション \(システム定義\)](#)

[DFSDFxxx メンバーの COMMON\\_SERVICE\\_LAYER セクション \(システム定義\)](#)

[DFSDFxxx メンバーの REPOSITORY セクション \(システム定義\)](#)

## MSC= 実行パラメーターによる MSC の使用可能化

IMS システムの複数システム結合 (MSC) 機能を使用可能にするには、IMS PROCLIB データ・セットの DFSPBxxx メンバー内で MSC=Y を指定します。

### 始める前に

MSC リソースの動的定義を使用可能にする場合は、CSL Operations Manager (OM) や Structured Call Interface (SCI) などの機能を使用可能にして、動的定義をサポートするよう IMS システムを構成する必要があります。TSO SPOC などのタイプ 2 コマンド・インターフェースも必要です。IMSRSC リポジトリを使用する場合は、リソース・マネージャー (RM)、リポジトリ・サーバー、および Common Queue Server も使用可能にする必要があります。

### このタスクについて

MSC=Y 実行パラメーターをコーディングして、IMS システム内で MSC が使用可能に設定されている場合は、IMS の開始前にステージ 1 システム定義マクロを使用して MSC リソースを定義する必要はありません。MSC リソースが IMS システム定義時に定義されている場合、MSC リソースは IMS の始動時にロードされます。

**推奨事項:** 動的 MSC が使用可能に設定され MSC リソースがもはやシステム定義プロセスを使用しては定義されなくなった際の操作を容易にするために、MSC リソースがシステム定義プロセスで定義される場合でも MSC=Y 実行パラメーターを指定するようにしてください。

以下の場合には、システム定義時に MSC リソースを定義せずに MSC を使用可能にすることができます。

- MSC リソースが既に IMSRSC リポジトリ内に定義されている
- IMS タイプ 2 CREATE コマンドを使用して MSC リソースを作成する予定である。この場合は、MSC リソースの動的定義を使用可能にする必要があります。
- IMS システムが、MSC リンクを必要としない共用キュー・グループに含まれるバックエンド IMS システムである。MSC は、以前に MSC リンクで送信された共用キュー上のメッセージを処理するためにだけ使用可能にされる。

MSC を使用可能にするには、以下のようにします。

### 手順

1. IMS PROCLIB データ・セットの DFSDFxxx メンバーの MSC セクションで、SYSID パラメーターを使用して、IMS システムのローカル・システム ID (SYSID) を 1 つ以上定義します。  
IMS システム定義に 1 つ以上の MSNAME マクロが組み込まれている場合、SYSID パラメーターのコーディングはオプションです。
2. オプションとして、DFSDFxxx メンバーの MSC セクションに MSCRSCS=DYN を指定することにより、MSC リソースの動的定義を使用可能にします。  
IMS タイプ 2 コマンドを使用して MSC リソースの作成や削除を行うには、MSC リソースの動的定義を使用可能にする必要があります。
3. MSCRSCS=DYN の場合は、DFSDFxxx メンバーの DYNAMIC\_RESOURCES セクションの AUTOIMPORT パラメーターで、使用する MSC リソース定義をインポートするためのオプションを指定します。
  - リソース定義の定義方法や格納場所に関係なく、始動時に IMS に MSC リソース定義を自動的に検出させてロードさせるには、AUTOIMPORT=AUTO を指定します。デフォルトは、AUTOIMPORT=AUTO です。

- IMS に IMSRSC リポジトリからのみ MSC リソースをロードさせるには、AUTOIMPORT=REPO を指定します。AUTOIMPORT=REPO の場合、IMS は IMSRSC リポジトリから MSC と MODBLKS の両方のリソースをインポートします。
  - IMS に、IMS システム定義時に定義された MSC リソースだけをロードさせるには、AUTOIMPORT=MSCGEN を指定します。
  - 始動時にどの MSC リソースもロードしないようにするには、AUTOIMPORT=NO を指定します。MSC リソース定義は、後で IMPORT DEFN コマンドを使用して IMSRSC リポジトリからインポートするか、IMS タイプ 2 CREATE コマンドを使用して定義する必要があります。
4. 始動プロシージャで MSC=Y 実行パラメーターを指定して、MSC を使用可能にします。
  5. IMS をコールド・スタートします。

システム定義時に IMS.SDFSRESL データ・セットの DFSCLL3x メンバー内に MSC リソースが生成されている場合、IMS は、始動時にそれらのリソースをオンライン・システム内にロードします。

#### 関連資料

[プロシージャの MSC= パラメーター \(システム定義\)](#)

[IMS PROCLIB データ・セットの DFSDFxxx メンバー \(システム定義\)](#)

## システム定義時の MSC の使用可能化

ステージ 1 システム定義マクロ MSPLINK、MSLINK、および MSNAME で少なくとも 1 つの MSC リンクを定義することにより、システム定義時に MSC を使用可能にすることができます。

### このタスクについて

MSC リソースがステージ 1 システム定義時にマクロで定義されている場合は、MSC= 実行パラメーターを指定する必要はありません。

**推奨事項:** 単に MSC リンクからフロントエンド上で受け取ったメッセージを処理するために、共用キュー環境のバックエンド IMS システム内で MSC を使用可能にする場合は、バックエンド IMS システム内でステージ 1 システム定義マクロを使用して MSC を使用可能にしないでください。代わりに、IMS システムの始動プロシージャで MSC=Y 実行パラメーターを指定してください。ステージ 1 システム定義マクロの代わりに MSC 実行パラメーターによって MSC を使用可能にすると、バックエンド IMS システムのクローン作成が容易になります。

#### 関連資料

[プロシージャの MSC= パラメーター \(システム定義\)](#)

[MSPLINK マクロ \(システム定義\)](#)

[MSLINK マクロ \(システム定義\)](#)

[MSNAME マクロ \(システム定義\)](#)

## SYSID の定義

システム ID (SYSID) は、1 から 2036 までの 2 バイトの番号で、MSC ネットワークの IMS システムを識別します。

### このタスクについて

ローカル SYSID は、次の方法で IMS システムに割り当てることができます。

- MSNAME ステージ 1 システム定義マクロ
- IMS PROCLIB データ・セットの DFSDFxxx メンバーの MSC セクション
- IMS タイプ 2 コマンド **CREATE MSNAME**

IMS タイプ 2 コマンド UPDATE MSNAME NAME(*msname*) SET(SIDL|SIDR) を使用して、オンラインで SYSID を変更できます。

## 関連概念

693 ページの『システム ID (SYSID)』

MSC は、システム ID (SYSID) (1 から 2036 までの 2 バイトの数字) を使用して、MSC ネットワークの IMS システムを識別します。

## 関連資料

[MSNAME マクロ \(システム定義\)](#)

[DFSDFxxx メンバーの MSC セクション \(システム定義\)](#)

[CREATE MSNAME コマンド \(コマンド\)](#)

## MSC= 実行パラメーターによる MSC の使用不可化

IMS 実行パラメーターで MSC=N を指定し、IMS のコールド・スタートを実行することによって、IMS システムの複数システム結合 (MSC) 機能を使用不可にすることができます。

### このタスクについて

MSC=N が指定されている場合、IMS の初期設定時に MSC リソースや制御ブロックはロードされません。DFSDFxxx メンバーや DFSDCxxx PROCLIB メンバーで指定されている MSC パラメーターがあっても、無視されます。ステージ 1 システム定義入力内の MSC 定義も無視されます。

あるいは、IMS システム定義のステージ 1 入力からすべての MSC 定義を削除し、MSC= 実行パラメーターを省略することによっても、MSC を使用不可にすることができます。MSC ステージ 1 システム定義入力が存在しない場合は、MSC=N がデフォルトです。

MSC を使用不可にするには、以下のようにします。

### 手順

1. IMS 実行パラメーターに MSC=N を指定します。  
MSC= パラメーターは、IMS PROCLIB データ・セットの DFSPBxxx メンバー内で、IMS 始動プロシージャまたは DCC 始動プロシージャに指定できます。
2. IMS をコールド・スタートします。
3. 始動操作の終わりに発行される最終的な DFS1929I メッセージで MSC= の指定を調べて、MSC が使用不可に設定されていることを確認します。

## ローカル・システムの定義

MSC ネットワークの定義では、ローカル・システムごとにトランザクション、論理端末、および物理接続と論理接続を定義する必要があります。これらのリソースは、システム定義マクロを使用して定義でき、また定義するリソース・タイプ用の IMS タイプ 2 CREATE コマンドを使用して動的に定義することもできます。

### このタスクについて

具体的には、個々のローカル・システムに以下の定義が必要です。

- そのシステムによって入力または処理されるすべてのトランザクション。
- そのシステムに接続されているすべての論理端末と、そのシステムで処理されるトランザクションによってまたは端末オペレーターによって参照されるリモート・システム内のすべての論理端末。ただし、プログラム経路指定出口ルーチンまたは MSC 直接経路指定が使用される場合を除きます。
- そのシステムと、指定トランザクションの処理中に共用されるリモート・システムとの間の、物理接続および論理接続。

各システムに用意する必要のあるシステム定義マクロは、以下の表に要約されています。使用する最初のマクロは、IMSCTRL です。MSVID キーワードを使用して、1 から 255 の範囲内の数値をそのシステムの固有の ID として割り当てます。これにより、制御ブロック DFSMSxxx が作成されます。この制御ブロックは、複数システム検査ユーティリティで使用されます。(3 桁の接尾部、xxx は、MSVID パラメーターと一致します。MSC ネットワーク内のそれぞれのシステムごとに 1 つの固有の制御ブロックが存在しま



す。)このオフライン・ユーティリティーは、すべてのパートナー・システム用のシステム定義が一致していることを検査するのに役立ちます。このユーティリティーを実行するために、プロシージャー IMSMSV が IMS.SDFSPROC に生成されます。

リソース	識別	マクロ	コーディングされる数
System	システム ID	IMSCTRL	1
プログラム	PSB 名	APPLCTN	PSB 当たり 1
ローカル・トランザクション	トランザクション・コード	TRANSACT	APPLCTN 当たり 1 またはそれ以上
リモート・トランザクション	トランザクション・コード	TRANSACT	リモート・システムにおいて APPLCTN 当たり 1 またはそれ以上
物理接続	物理リンク名	MSPLINK	接続当たり 1
論理接続	論理リンク名	MSLINK	物理リンク・タイプに応じて、MSPLINK 当たり 1 つ以上。 <a href="#">711 ページの『1』</a>
経路指定名	論理バス名	MSNAME	MSLINK 当たり 1 またはそれ以上
ローカル端末	LTERM 名	NAME	ローカル端末当たり 1
リモート端末	LTERM 名	NAME <a href="#">711 ページの『2』</a>	リモート端末当たり 1

#### 注:

1. 複数セッション TCP/IP 物理リンクおよび VTAM 物理リンクには、追加の MSLINK マクロがあります。チャンネル間 (CTC) 物理リンク・タイプおよびメモリー間 (MTM) 物理リンク・タイプは、物理リンク当たり 1 つだけ論理リンクを持つことができます。
2. 対応するローカル端末が ETO 端末の場合には、ETO MSC 記述子を用いて定義します。

#### 関連資料

IMS 環境で使用されるマクロ (システム定義)

複数システム検査ユーティリティー (DFSUMSV0) (システム・ユーティリティー)

## パートナー・システムの定義

2 つのパートナー・システム間で使用可能なリンクを定義するには、MSPLINK、MSLINK、および MSNAME を両側で定義する必要があります。これらの MSC リソースは、CREATE MSPLINK、CREATE MSLINK、および CREATE MSNAME の各コマンドを使用して動的に定義でき、またステージ 1 システム定義マクロ MSPLINK、MSLINK、および MSNAME を使用して静的に定義することもできます。

### このタスクについて

システム間の接続名およびコマンドで使用される論理名を定義する 3 つの MSC リソース・タイプ MSPLINK、MSLINK、および MSNAME の間には、関連があります。

## 物理リンクの定義

IMS システムに接続するさまざまなタイプの物理リンクを定義できます。各リンクに必要なハードウェアに従って選択してください。

### このタスクについて

物理リンクの選択項目は、以下のとおりです。

チャンネル間 (CTC)

メモリー間 (MTM)  
TCP/IP  
VTAM

CREATE MSPLINK タイプ 2 コマンドまたは MSPLINK ステージ 1 システム定義マクロを使用して、物理リンクとその属性を定義できます。

CREATE MSPLINK コマンドを使用するには、ご使用の IMS システムで動的リソース定義が使用可能になっている必要があります。CREATE MSPLINK コマンドはログに記録されます。

IMS システムで動的定義が使用可能になっていない場合は、MSPLINK マクロを使用して物理リンクを定義する必要があります。いくつかの物理接続がバックアップ用であったり、継続して使用する意図がない場合であっても、システム定義時に、使用する可能性のある物理接続をすべて定義してください。

パートナー・システムの物理リンクも利用法のシステム定義で宣言する必要があります。TYPE キーワードを用いて、使用する物理リンクの種類を宣言してください。

物理リンクに名前を割り当てます。物理リンクの名前は、システム間の接続を、使用される物理装置や伝送技法と一致させるために、論理リンクの定義に使用されます。また、この名前は、コマンドの発行時にリンクを識別するためにも使用されます。

あるシステムから別のシステムに会話の一部が渡される場合、SPA サイズがメッセージ・セグメントのサイズを超える場合には、最大のスクラッチパッド域 (SPA) サイズを使用してください。

タイプ 2 **UPDATE MSPLINK** コマンドを使用して、既存の物理リンクの属性を変更できます。物理リンクを IMSRSC リポジトリに保管しない限り、変更は IMS システムの次のコールド・スタートまでの間のみ有効です。

#### 関連資料

[MSPLINK マクロ \(システム定義\)](#)  
[IMS 環境で使用されるマクロ \(システム定義\)](#)  
[CREATE MSPLINK コマンド \(コマンド\)](#)  
[QUERY MSPLINK コマンド \(コマンド\)](#)  
[UPDATE MSLINK コマンド \(コマンド\)](#)  
[UPDATE MSPLINK コマンド \(コマンド\)](#)

## 物理リンクのバッファ・サイズ

物理リンク・バッファ・サイズは、物理リンクに割り当てられるすべての論理リンクのデフォルト・バッファ・サイズを定義します。

### このタスクについて

少なくとも、この物理リンクを通して伝送される最大のメッセージ・セグメントのサイズを使用してください。バッファ・サイズは、物理リンクの両端で等しいことが必要です。

CREATE MSPLINK コマンドまたは MSPLINK マクロで BUFSIZE キーワードを使用して、バッファ・サイズを指定します。

IMS タイプ 2 コマンド **UPDATE MSLINK NAME(linkname) SET(BUFSIZE(new\_bufsize))** または IMS タイプ 1 コマンド **/UPDATE MSLINK NAME(linkname) SET(BUFSIZE(new\_bufsize))** を使用して、個別の論理リンクに異なるバッファ・サイズを指定できます。

物理リンクに割り当てられているすべての論理リンクのバッファ・サイズを更新するには、コマンド **UPDATE MSPLINK NAME(linkname) SET(BUFSIZE(new\_bufsize))** を使用します。

リンク・バッファに指定できる最小サイズおよび最大サイズについては、BUFSIZE キーワードの説明を参照してください。

MSC 統計を表示すると、最適なバッファ・サイズを判別するのに役立ちます。MSC 統計は、IMS タイプ 2 コマンド **QUERY MSLINK NAME(linkname) SHOW(STATISTICS)** で表示します。統計 Hi\_Msg\_Send\_SZ および Hi\_Msg\_Rec\_SZ を使用して、送信および受信されるメッセージの最大サイズを確認します。送信および受信された入出力要求の数と、送信および受信されたメッセージの数を比較することもできます。IMS



が1つのメッセージを送信するために複数の入出力要求の送受信を必要としている場合は、バッファが小さすぎます。送信および受信された入出力要求数は、統計 Tot\_Send\_CT および Tot\_Rec\_CT で収集されます。送信および受信されたメッセージの数は、統計 Tot\_Msg\_Send\_CT および Tot\_Msg\_Rec\_CT で収集されます。

物理リンク・タイプに応じて、リンク・バッファ・サイズ用のストレージは、異なるプールから割り振られます。TCP/IP および VTAM リンク・バッファは、高入出力プール (HIOP) から割り振られます。CTC リンク・バッファは、通信入出力プール (CIOP) から割り振られます。MTM リンク・バッファは、共通ストレージ域 (CSA) のサブプール 231 から割り振られます。

#### 関連資料

[MSPLINK マクロ \(システム定義\)](#)

[IMS 環境で使用されるマクロ \(システム定義\)](#)

[CREATE MSPLINK コマンド \(コマンド\)](#)

[QUERY MSLINK コマンド \(コマンド\)](#)

[UPDATE MSLINK コマンド \(コマンド\)](#)

[UPDATE MSPLINK コマンド \(コマンド\)](#)

## CTC 物理リンクの定義

CTC アダプターは、チャンネル間ハードウェア接続です。

### このタスクについて

CTC リンクは通常、IMS システムが同じデータ・センター内にある場合にのみ使用されます。CTC 接続タイプを使用する物理リンクには、1つだけ論理リンクを割り当てることができます。

CREATE MSPLINK タイプ 2 コマンドまたは MSPLINK ステージ 1 システム定義マクロを使用して、物理リンクとその属性を定義できます。

CREATE MSPLINK コマンドを使用するには、ご使用の IMS システムで動的リソース定義が使用可能になっている必要があります。CREATE MSPLINK コマンドはログに記録されます。

IMS システムで動的定義が使用可能になっていない場合は、MSPLINK マクロを使用して物理リンクを定義する必要があります。いくつかの物理接続がバックアップ用であったり、継続して使用する意図がない場合であっても、システム定義時に、使用する可能性のある物理接続をすべて定義してください。

CTC リンクの場合は、アドレス・パラメーター値を追加する必要があります。各 CTC リンクは、固有のアドレスを持っていなければなりません。

MSPLINK システム定義マクロを使用して CTC リンクを作成する場合は、DD 名も追加する必要があります。

#### 関連資料

[MSPLINK マクロ \(システム定義\)](#)

[IMS 環境で使用されるマクロ \(システム定義\)](#)

[CREATE MSPLINK コマンド \(コマンド\)](#)

[QUERY MSPLINK コマンド \(コマンド\)](#)

[UPDATE MSLINK コマンド \(コマンド\)](#)

[UPDATE MSPLINK コマンド \(コマンド\)](#)

## MTM 物理リンクの定義

MTM リンクは、同じ論理区画で稼働している IMS サブシステム間のソフトウェア・リンクです。

### このタスクについて

MTM 接続タイプを使用する物理リンクには、1つだけ論理リンクを割り当てることができます。

CREATE MSPLINK タイプ 2 コマンドまたは MSPLINK ステージ 1 システム定義マクロを使用して、物理リンクとその属性を定義できます。

CREATE MSPLINK コマンドを使用するには、ご使用の IMS システムで動的リソース定義が使用可能になっている必要があります。CREATE MSPLINK コマンドはログに記録されます。

IMS システムで動的定義が使用可能になっていない場合は、MSPLINK マクロを使用して物理リンクを定義する必要があります。いくつかの物理接続がバックアップ用であったり、継続して使用する意図がない場合であっても、システム定義時に、使用する可能性のある物理接続をすべて定義してください。

#### 関連資料

[MSPLINK マクロ \(システム定義\)](#)

[IMS 環境で使用されるマクロ \(システム定義\)](#)

[CREATE MSPLINK コマンド \(コマンド\)](#)

[QUERY MSPLINK コマンド \(コマンド\)](#)

[UPDATE MSPLINK コマンド \(コマンド\)](#)

## TCP/IP 物理リンクの定義

TCP/IP 物理リンクでは、2つの IMS システム間の接続に、インターネットで使用される TCP/IP 接続プロトコルおよびネットワーク・プロトコルを使用します。

### このタスクについて

通常、IMS システムが異なるデータ・センターにある場合に使用されます。IMS Connect が、物理リンクの TCP/IP 接続およびプロトコルを管理します。

MSC と IMS Connect の間の通信は、IMSplex 内の IMS 共通サービス層 (CSL) の構造化呼び出しインターフェース (SCI) コンポーネントによって管理されます。

TCP/IP 接続タイプを使用する物理リンクには、複数の論理リンクを割り当てることができます。TCP/IP 物理リンクは常に MSC 帯域幅モードで動作し、他の物理リンク・タイプより幾分大きいバッファ・サイズが必要です。

CREATE MSPLINK タイプ 2 コマンドまたは MSPLINK ステージ 1 システム定義マクロを使用して、物理リンクとその属性を定義できます。

CREATE MSPLINK コマンドを使用するには、ご使用の IMS システムで動的リソース定義が使用可能になっている必要があります。CREATE MSPLINK コマンドはログに記録されます。

IMS システムで動的定義が使用可能になっていない場合は、MSPLINK マクロを使用して物理リンクを定義する必要があります。いくつかの物理接続がバックアップ用であったり、継続して使用する意図がない場合であっても、システム定義時に、使用する可能性のある物理接続をすべて定義してください。

TCP/IP を使用する物理リンクの場合、MSPLINK マクロの NAME キーワードの指定は、リモート IMS システムの IMSCTRL システム定義マクロの IMSID キーワードの指定と一致している必要があります。

SESSION パラメーターは、TCP/IP および VTAM の物理リンク・タイプに対してアクティブにできる並列セッション (つまり、論理リンク) の数を示します。

#### 関連資料

[MSPLINK マクロ \(システム定義\)](#)

[IMS 環境で使用されるマクロ \(システム定義\)](#)

[CREATE MSPLINK コマンド \(コマンド\)](#)

[QUERY MSPLINK コマンド \(コマンド\)](#)

[UPDATE MSPLINK コマンド \(コマンド\)](#)

[UPDATE MSPLINK コマンド \(コマンド\)](#)

## VTAM 物理リンクの定義

VTAM 物理リンクは、SNA VTAM によってサポートされています。

### このタスクについて

CREATE MSPLINK タイプ 2 コマンドまたは MSPLINK ステージ 1 システム定義マクロを使用して、物理リンクとその属性を定義できます。

CREATE MSPLINK コマンドを使用するには、ご使用の IMS システムで動的リソース定義が使用可能になっている必要があります。CREATE MSPLINK コマンドはログに記録されます。

IMS システムで動的定義が使用可能になっていない場合は、MSPLINK マクロを使用して物理リンクを定義する必要があります。いくつかの物理接続がバックアップ用であったり、継続して使用する意図がない場合であっても、システム定義時に、使用する可能性のある物理接続をすべて定義してください。

VTAM を使用する物理リンクの場合は、リモート・システムの VTAM APPL ステートメントのラベルに一致する NAME キーワードを指定する必要があります。NAME キーワードと VTAM APPL ステートメントではともに、VTAM ノード名を指定します。

VTAM APPL ステートメントのラベルは、同じステートメント上の ACBNAME パラメーターのデフォルト値としても使用されます。ACBNAME パラメーターにデフォルト値を使用するかどうかに関係なく、この値は IMS COMM マクロの APPLID パラメーターと一致していなければなりません。

SESSION パラメーターは、TCP/IP および VTAM の物理リンク・タイプに対してアクティブにできる並列セッション (つまり、論理リンク) の数を示します。

### 関連資料

[MSPLINK マクロ \(システム定義\)](#)

[IMS 環境で使用されるマクロ \(システム定義\)](#)

[CREATE MSPLINK コマンド \(コマンド\)](#)

[QUERY MSPLINK コマンド \(コマンド\)](#)

[UPDATE MSLINK コマンド \(コマンド\)](#)

[UPDATE MSPLINK コマンド \(コマンド\)](#)

## 論理リンクの定義

論理リンク定義は、MSLINK システム定義マクロを使用して静的に定義でき、またタイプ 2 CREATE MSLINK コマンドを使用して動的に定義することもできます。どちらの方式でも、リンクに名前を付けたり、論理リンクをパートナー・システムで定義された論理リンクと関連付けたり、論理リンクと一緒に使用できる物理リンクのタイプを定義したりすることができます。

### このタスクについて

マクロのラベル・フィールドを使用して論理リンクに名前を割り当てることができます。論理リンク名は、タイプ 2 コマンド **QUERY MSLINK** または **UPDATE MSLINK** が、ターゲット論理リンクを識別するために使用します。マクロを使用して論理リンク名を定義しなかった場合、IMS はデフォルトの論理リンク名 DFSLxxxxx を割り当てます。ここで、xxxxxx は論理リンク番号です。

PARTNER キーワードで 2 文字の英数字のパートナー ID を指定することにより、論理リンクをパートナー・システムに定義されている論理リンクと関連付けます。指定するパートナー ID は、パートナー・システム内の MSLINK マクロで指定された対応するパートナー ID と一致していなければなりません。

PARTNER キーワード値が一致していると、2 つのパートナー・システムの間に論理接続があることを表します。例えば、ある論理リンクで B1 がパートナー ID として定義されている場合、この同じ B1 というパートナー ID が、パートナー・システムの対応する MSLINK マクロで指定していなければなりません。

MSPLINK キーワードを指定して、使用される可能性のある物理接続のタイプをこの論理リンクと一致させます。指定した物理リンクのタイプに応じて、1 つ以上の MSLINK マクロを物理リンクに割り当てることができます。

- 物理リンク・タイプが CTC または MTM の場合には、物理接続の各種類 ごとに 1 つの MSLINK マクロが使用されます。
- 物理リンク・タイプが TCP/IP または VTAM の場合、1 つの物理リンクに複数の論理リンクを割り当てることができます。例えば、System\_B と System\_C は、MTM を使用する論理リンクと CTC を使用する論理リンクという 2 つの論理リンクを持つことが可能であり、System\_C と System\_D は両方とも TCP/IP または VTAM 物理リンクを使用する 2 つの論理リンクを持つことが可能です。

各 IMS システムに対して、最大 999 の論理リンクを定義することができます。

マクロを使用せずに、タイプ 2 CREATE MSLINK コマンドを使用して、論理リンクを定義することもできます。

**UPDATE MSLINK** コマンドを使用して、MSLINK マクロによって定義された論理リンク属性を変更できます。変更可能な属性には、名前、物理リンクの割り当て、パートナー ID、バッファ・サイズなどがあります。変更は、IMS システムを次にコールド・スタートするまで有効です。

#### 関連資料

[CREATE MSLINK コマンド \(コマンド\)](#)

[UPDATE MSLINK コマンド \(コマンド\)](#)

[MSLINK マクロ \(システム定義\)](#)

## 論理パスの定義

オペレーターが 2 つのシステムを接続するときは、物理接続タイプの選択が可能な場合もあります。したがって、論理リンク名が使用されます。その論理パス名は、MSNAME システム定義マクロを使用して定義するか、タイプ 2 CREATE MSNAME コマンドを使用して動的に定義することができます。

### このタスクについて

SYSID キーワードを使用して、パスで結合する 2 つのシステムを宣言します。1 から 2036 までの範囲から 1 桁の ID を選択します。1 つはリモート・システム用で、もう 1 つはローカル・システム用です。例えば、(1,3) は、このパスを使用するメッセージがリモート・システム番号 1 に送信されていて、しかもローカル・システム番号は 3 であることを表します。

タイプ 2 コマンドの **UPDATE MSNAME** を使用して、論理パスのリモート・システム ID とローカル・システム ID を変更できます。変更は、IMS システムを次にコールド・スタートするまでの間のみ有効です。

MSNAME マクロの後には、リモート・システムにある端末の LTERM 名を指定することができる一連の NAME マクロを続けることができます。トランザクションを入力している、リモート・システムの各端末をすべて宣言する必要はありません。このローカル・システム宛てのトラフィックを入力する端末だけを宣言します。このシステムに宛てるトランザクションを入力するリモート・システムの LTERM が、ETO 端末用である場合には、NAME マクロの代わりに ETO MSC 記述子を使用して LTERM を定義します。

#### 関連資料

[CREATE MSNAME コマンド \(コマンド\)](#)

[MSNAME マクロ \(システム定義\)](#)

## リモート・トランザクション用リンク優先順位の設定

宛先システムのシステム定義時に、TRANSACT マクロの PRTY= キーワードを使用して、リモート・トランザクションに優先順位を割り当てることができます。優先順位の割り当ては、タイプ 2 コマンド CREATE TRAN および UPDATE TRAN を使用して動的に行うこともできます。

### このタスクについて

リモート入力トランザクションは、宛先システムにおいては、発信元システムのシステム定義時に定義された場合とは異なるスケジューリング優先順位を持つ場合があるため、優先順位の割り当てが必要なことがあります。これは、TRANSACT マクロの SYSID= パラメータまたは CREATE TRAN コマンドの NPRI キーワードで定義されたときに、リモート・トランザクションが高い優先順位を持っていたとしても、応答を長時間待つ可能性があることを意味します。

例えば、以下の図では、端末 1 は、システム B にリモート・トランザクション (ASMB1) を送信しています。

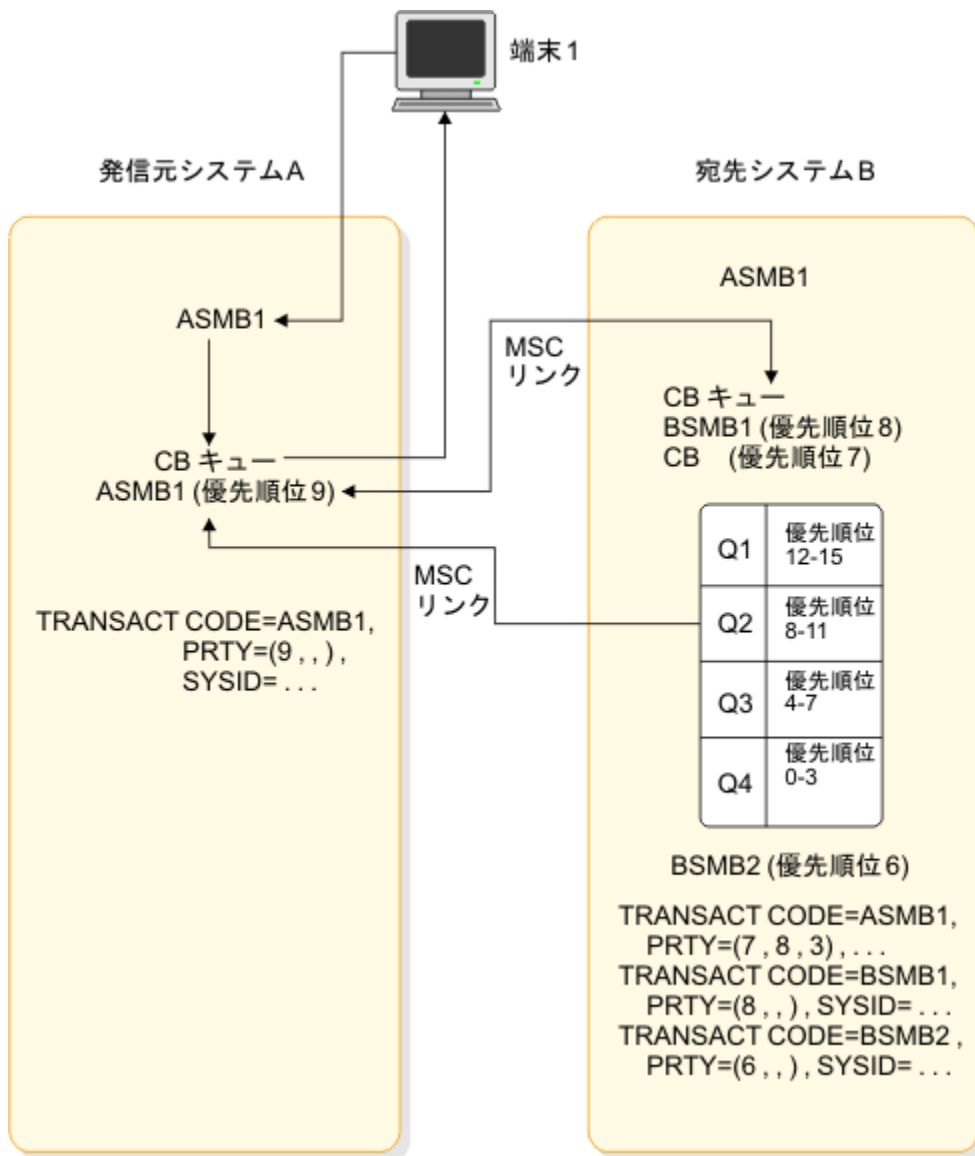


図 120. リモート・トランザクションのリンク優先順位

IMS システム A では、このトランザクションは、制御ブロック (CB) へのキューに入れられ、優先順位は 9 (SYSID= パラメーターで定義済み) の状態にあります。ASMB1 は、システム B に送信されると、ローカル・トランザクションとなり、処理されます。メッセージ・ルーターは、応答を送信するために呼び出されると、MSC リンクを表す制御ブロックへの応答を待機させます。ASMB1 が優先順位 7 でシステム B に定義されている場合、12 - 15 (キュー 1) の優先順位を持つすべてのトランザクションが最初に処理され、次に 8 - 11 (キュー 2) のトランザクションが処理されます。ASMB1 (キュー 3 で優先順位 7) は、その次に処理されます。しかし、図に示されたように、限界優先順位を 8 に設定し、限界カウントを 3 に設定することができます。その結果、処理されるメッセージ・キューの数が 3 に達すると、ASMB1 の優先順位は 8 に変更され、その後の応答はキュー 2 に入れられます。システム B では、BSMB1 のすべてのメッセージが送信されてから、応答が送信されます。この例では、BSMB2 のメッセージが最後に送信されます。

ただし共用キュー環境では、優先順位は、MSC リンクから受信された、処理されるメッセージにのみ適用されます。優先順位は、カップリング・ファシリティーがそれら 4 つのキューを持っていないため、応答メッセージ (入力システムに戻るメッセージ) には適用されません。その代わりに、すべての応答メッセージが FIFO 送信されます。



## 関連資料

[CREATE TRAN コマンド \(コマンド\)](#)

[UPDATE TRAN コマンド \(コマンド\)](#)

[TRANSACTION マクロ \(システム定義\)](#)

## MSC ネットワーク内での逐次トランザクション処理

逐次トランザクションは、同タイプの他のトランザクションと関連して受信された順序で処理されます。

以下のステップを実行することにより、リモート MSC IMS システム内でトランザクションを確実に逐次処理することができます。

- ローカルとリモート MSC IMS システムの両方で、トランザクションを逐次と定義する。
- トランザクションを、ローカルとリモート MSC IMS システム間の単一論理リンク・パスに制限する。
- すべての同タイプの逐次トランザクションを、処理用に同じリモート MSC IMS に送信する。

シリアライゼーションは、以下のトランザクションに関しては保持されません。

- 別の論理リンク・パスを通して送信されている
- 別の MSC IMS システムから発信されている
- 別のリモート MSC IMS システムで処理されている

## MSC ネットワークでの逐次トランザクションの PRTY= キーワードと出力メッセージ

優先順位の *normal* パラメーターと *limit* パラメーター (CREATE TRAN SET(NPRI(*normalpriority*),LPRI(*limitpriority*)), または TRANSACTION マクロの PRTY= キーワードで定義される) が等しくない場合、MSC ネットワーク内での逐次トランザクションの出力メッセージの逐次処理は予測不能になることがあります。

*normal* パラメーターと *limit* パラメーターが等しくない場合に、メッセージ・キュー上の出力メッセージ数が PRTY= キーワードの *limit\_count* パラメーターの値より大きいか等しくなると、*limit\_count* 値に到達した後にリモート MSC IMS システムが受信した出力メッセージは、リモート MSC IMS システムが受信していたメッセージより前に処理される可能性があります。

## 関連資料

[CREATE TRAN コマンド \(コマンド\)](#)

[UPDATE TRAN コマンド \(コマンド\)](#)

[TRANSACTION マクロ \(システム定義\)](#)

## 出口ルーチンの指定

以下のトピックでは、MSC を使用した出口ルーチンの指定方法について説明します。

### 関連概念

750 ページの『TM/MSC メッセージ経路指定および制御ユーザー出口ルーチンの概要』

メッセージ経路指定は、経路指定オプションまたはメッセージの制御を提供する TM/MSC メッセージ経路指定 / 制御ユーザー出口ルーチン (DFSMSCEO) を使用しない限り、定義済みスキームに基づいて自動的に行われます。

## DFSMSCEO によるメッセージの経路指定

TM および MSC メッセージ経路指定、および制御ユーザー出口ルーチン (DFSMSCEO) は、システム定義処理には組み込まれていません。したがって、DFSMSCEO を IMS.SDFSRESL に組み入れてください。

端末経路指定エントリー・ポイントは、宛先を変更しながら、入力システム内で呼び出されます。トランザクションが MSC リンクを渡って IMS に到達した場合、リンク受信エントリー・ポイントが呼び出される場合があります。これらのエントリー・ポイントは、宛先を再び変更する場合があります。宛先はさまざまなポイントで変更される可能性があるため、宛先がどのように変更されるかを明確に文書化し、処理に使用される新規宛先を説明してください。キュー状況の表示を解釈できるように、マスター端末オペレーター用の情報としてトランザクション・コード変更を関連付けることが必要な場合もあります。

DFSMSCEO 出口ルーチンが共用キューを使用する IMSplex 環境で使用される場合、DFSMSCEO 出口ルーチンを、IMSplex 内の IMS システムで類似性経路指定のために使用できます。これで、トランザクション・メッセージとバックエンド IMS システム間に類似性が確立されます。類似性経路指定により、共用キューでのトランザクション・メッセージの共用が効果的に抑制されるため、トランザクションが共用キューに到着した場合には、類似性を持つ IMS システムのみにそのことが通知されます。

DFSMSCEO IMSplex 類似性経路指定は、RRS=N または AOS=N の場合の処理で、APPC 同期トランザクションまたは OTMA 送信後コミット (CM1) トランザクションを IMSplex 内のバックエンド IMS システムに経路指定するための方法でもあります。類似性を割り当てないと、これらの APPC トランザクションと OTMA トランザクションは、処理のためにバックエンド・システムに経路指定できません。APPC 同期トランザクションまたは OTMA 送信後コミット (CM1) トランザクションと DFSMSCEO IMSplex 類似性経路指定を併用するには、MSC が IMSplex で使用できなければなりません。

トランザクションが、全体の処理の中の異なる ステージでいかに経路指定されるかを念頭に置いておくください。IMS A からの入力が入力端で処理されている場合、処理プログラムは出口ルーチンと呼び出して、その出力を元の入力端末に送信するか、別のロケーションに送信するかを決定することができます。この出口ルーチンの文書では、代替メッセージ宛先のパターンを示す必要があります。エンド・ユーザーは、出力を予期していながら、出力が別のコンポーネントに送信されていることに気付いていない可能性があります。

### 関連概念

750 ページの『[TM/MSC メッセージ経路指定および制御ユーザー出口ルーチンの概要](#)』

メッセージ経路指定は、経路指定オプションまたはメッセージの制御を提供する TM/MSC メッセージ経路指定 / 制御ユーザー出口ルーチン (DFSMSCEO) を使用しない限り、定義済みスキームに基づいて自動的に行われます。

### 関連資料

[「TM および MSC メッセージ経路指定および制御」ユーザー出口ルーチン \(DFSMSCEO\) \(出口ルーチン\)](#)

## DFSCMUX0 によるエラー・メッセージの管理

メッセージ制御 / エラー出口ルーチン (DFSCMUX0) により、エラー状態にある MSC リンクまたはメッセージ・キュー上のメッセージを管理し、制御することができます。この出口ルーチンは、MSC リンク開始、リンク終了、送信エラー、または受信エラーで活動化されます。

メッセージ制御/エラー出口ルーチンは、以下の状態の場合に APPC/IMS に呼び出されます。

- LU 6.2 プログラムに出力メッセージを送信中に、LU 6.2 セッションが失敗した。
- LU 6.2 アプリケーション・プログラムへの送信が拒否され、送信エラーによる 割り振り解除 というメッセージが出された。
- **/DEQUEUE** *luname tpname* が入力された。

メッセージ制御/エラー出口ルーチンをカスタマイズして、IMS に以下のアクションを取るよう に 要求することができます。

- メッセージを破棄し、**/DEQUEUE** コマンドを続行するという デフォルト・アクションを採用する。
- エラーのメッセージを廃棄する。
- エラーのメッセージを廃棄し、MTO または発信元端末にこのエラーを通知する。
- エラーのメッセージを、異なるローカルまたはリモートのトランザクション、ローカルまたはリモートの LTERM、あるいは LU 6.2 アプリケーション・プログラム に再経路指定する。

IMS.ADFSSRC で提供されるサンプル出口ルーチンは、デフォルトのアクションを使用します。

### 関連資料

[メッセージ制御/エラー出口ルーチン \(DFSCMUX0\) \(出口ルーチン\)](#)

## 複数システムによるネットワーク定義への影響

それぞれのシステム定義には、接続されているすべての端末が定義されている必要があります。これらの端末は、システム定義時に TERMINAL マクロまたは NAME マクロを使用して定義するか、タイプ 2 コマンド CREATE LTERM および CREATE MSNAME を使用して動的に定義します。しかし、他のシステムの端末



が、定義されたシステムあてのメッセージを送信できる場合、それらの端末は、論理的にはそのシステムのネットワークの一部分です。

MSC 物理リンクおよび論理リンクは、CREATE MSPLINK コマンドおよび CREATE MSLINK コマンドで動的に定義するか、システム定義プロセスを通じて静的に定義することができます。MSC リモート LTERM は、CREATE LTERM コマンドを使用して動的に定義するか、ステージ 1 システム定義 NAME マクロを使用して、または IMS の初期設定時に拡張端末機能 (ETO) を使用して、静的に定義することができます。

トランザクションが、リモート・システムで処理される場合には、ローカル・システムの入力 LTERM 名は、メッセージの一部として送信されます。処理プログラムが代替 PCB を使用して、メッセージを入力端末以外の端末に送る場合は、直接経路指定を使用する場合を除いて、それらの宛先をリモートとして宣言する必要があります。NAME マクロを使用して、または CREATE LTERM コマンドを使用して、すべての入力に対する LTERM 名を定義します。MSNAME マクロの後に、NAME マクロをグループで置きます。これで、いくつかのシステム定義デッキにおいて集散的に発生する一連の LTERM を持つことになります。例えば、TERMA は、入力システム、中間システム、および処理システムに存在することができます。

ネットワークを計画する場合には、入力システムまたは中間システムに対するメッセージ・キューを、待機中のリモート・トランザクションに使用できるようにしなければならないことに留意してください。メッセージ・キュー用のスペースを割り振るときは、メッセージ長と予期される負荷を考慮に入れる必要があります。同様に、これらのメッセージがそのシステムで処理されない場合であっても、入出力バッファにこれらのメッセージが存在することも考慮しなければなりません。

### 関連タスク

71 ページの『[拡張端末オプションの管理](#)』

IMS 拡張端末オプション (ETO) を使用すると、VTAM 端末とユーザーをシステム定義時に事前定義しなくても、IMS に動的に追加できるようになります。

### 関連資料

[CREATE LTERM コマンド \(コマンド\)](#)

[CREATE MSNAME コマンド \(コマンド\)](#)

[NAME マクロ \(システム定義\)](#)

[TERMINAL マクロ \(システム定義\)](#)

## システム間のトランザクション定義の検査

複数システム・ネットワークの一部となるいくつかのシステムのシステム定義が完了すると、これで、トランザクション・コード、LTERM 名、およびシステム識別番号はすべて制御ブロックで指定できます。

### 複数システム検査ユーティリティの使用

IMS 複数システム検査ユーティリティをオフラインで使用すれば、使用した名前がシステム定義を通して一貫しているかどうかを検査することができます。

#### このタスクについて

**例:** 2つの定義では TRANX をコードとして使用していたのに、3番目のシステムでは TRANXX を使用した場合、複数システム検査ユーティリティは、TRANXX への単一参照を強調表示したメッセージをユーザーに送信します。

複数システム検査ユーティリティは、IMS.PROCLIB 内の IMSMSV プロシージャーを使用して実行することができます。

また、複数システム検査ユーティリティは、トランザクション・コードが複数のシステムでローカルとして定義されていないかどうかの確認もします。

この複数システム検査ユーティリティは、ユーザーが検査に含めたいシステム識別番号が指定された入力制御ステートメントを使用します。その出力は、マルチシステム・パス・マップの書式になります。

**制約事項:** 複数システム検査ユーティリティは、MSC 直接経路指定が誤って使用されたことから生じるエラーを検出することはできません。

## 関連資料

複数システム検査ユーティリティー (DFSUMSV0) (システム・ユーティリティー)

## システム定義状況のオンライン検査

ネットワークの各コンポーネント・システムが操作可能な場合には、**/MSVERIFY** コマンドを使用して、システム定義または ETO MSC 記述子の整合性をチェックすることができます。

### このタスクについて

**推奨事項:** **/MSVERIFY** コマンドは、実動時ではなく、システム・テスト・フェーズで使用してください。**/MSVERIFY** コマンドは、かなりのトラフィックを生成します。

**/MSVERIFY** は以下の妥当性検査をします。

- 定義されたリモート LTERM 用に論理端末が存在する。
- トランザクションが定義され、そしてリモート・システムの属性がローカル・システムの属性と同じである。
- 論理パス定義が、整合性があり使用可能である。

**推奨事項:** リモート LTERM は、ETO MSC 記述子を使用して定義してください。そうしないと、複数システム検査ユーティリティーはリモート LTERM は認識しますが、ターゲット・システムにおける対応するローカル LTERM は認識しません。

**/MSVERIFY** コマンドは、各コマンドごとにローカル・システムと1つのリモート・システムとの間の定義の整合性を検査します。検査は、ローカル SYSID から始まり、その次にリモート SYSID に進みます。処理の各セグメントが始まると、検査されるローカルおよびリモートの SYSID が MTO に通知されます。このメッセージ (DFS2234I または DFS2236I) もタイム・スタンプを取られます。特定のローカルおよびリモートの SYSID が検査されると、MTO は、対応するタイム・スタンプの付いた検査管理用通知を受け取ります。定義または割り当てにエラーが見つかったと、エラー・メッセージが戻されます。

コマンド処理中に、ローカル・システムは、独自にローカルで定義したすべての MSC エlement (SYSID) を、**/MSVERIFY** コマンドの中で指定されたリモート IMS システムに送信します。リモート・システムは、ローカル・システムに属するすべての宛先と属性 (リモート・システム定義または割り当てがこれらを記述しているため) を戻すことで応答します。コマンド入力システムは、ローカルとリモート定義の整合性をチェックします。

検査すべき SYSID のリスト (メッセージ DFS2234I または DFS2236I IN PROGRESS で与えられた) および応答のタイム・スタンプの対応を使用して、応答リストが完全なものかどうかを検査しなければなりません。

ローカル・システムは、その次にすべての MSC エlement をそれらが対応する属性 (ローカル・システムは属性が定義されているので) でリモート・システムに定義されていることを期待してリモート・システムに送信します。リモート・システムは、整合性検査を実行し、エラー・メッセージをコマンド入力端末に戻します。

MSC パス整合性は、論理リンクへの論理リンク・パスの現在操作可能な割り当て、および物理リンクへの論理リンクの現在操作可能な割り当てについてのみ検査されます。

**推奨事項:** 良好な応答を受け取った (エラーが戻されない) 場合でも、MSC 定義と割り当てが正確であることを確認してください。定義エラーや割り当てエラーがあると、一部の SYSID からコマンド応答が戻されない可能性があります。特定の SYSID から応答がないということは大変な事態であると見なしてください。

**/MSVERIFY** コマンドからの下記のエラー応答を検討すると、これらの定義エラーおよび割り当てのエラーの回避に役立ちます。

```
DFS2235I SYSID __ is defined as local in both systems
DFS2241I __ is defined as remote transaction in both systems
DFS2242I __ is not defined as LTERM in both systems
DFS2243I __ is not defined as transaction in both systems
DFS2245I Multisegment transaction flag for __ not consistent
DFS2246I Non-inquiry only flag for __ not consistent
```

```
DFS2247I Conversational flag for __ not consistent
DFS2248I Irrecoverable flag for __ not consistent
DFS2249I Fixed length SPA flag for __ is not consistent
DFS2250I The SPA length for __ is not the same
```

**/MSVERIFY** コマンドは、MSC 直接経路指定の不適切な使用により起きたエラーを検出することができません。

## MSC に対するセキュリティの考慮事項

別のシステムで処理されるトランザクションについては、基本メッセージで必要とされるのと同程度のセキュリティ検査を行ってください。RACF は、MSC ネットワーク内の IMS リソースの保護に使用できません。

トランザクション許可とパスワード検査とを組み合わせたサインオン検査では、入力時にその処理を制御することができます。RACF のリソース定義では、セキュリティ・テーブルが作成されたシステムでトランザクションが処理されない場合であっても、トランザクション名を宣言する必要があります。

MSC ネットワークでのセキュリティ管理は、それぞれのローカルおよびリモート IMS で個別に実行されます。該当のトランザクションに関してローカルとリモートのどちらでもない、MSC 環境内の中間 IMS は、そのトランザクションにいかなるセキュリティ検査も実行しません。

RACF は、宛先システムが処理すべきメッセージを受信した際にトランザクション許可検査を行うことができます。RACF が提供する検査の量は、DFSDCxxx IMS.PROCLIB メンバー内の MSCSEC= パラメーターおよび DFSMSCEO 出口ルーチンからのフィードバックによって決まります。DFSMSCEO ユーザー出口は、オプションで、システム DFSDCxxx メンバー・セキュリティをメッセージごとにオーバーライド または受け入れることができます。

MSC リンク上のリモート IMS で受信されるトランザクションは、許可検査のためにトランザクション許可モジュールに渡されますが、パスワードはリンクを通して渡されないため、パスワードが必須である場合にはトランザクション許可検査は失敗します。パスワードを必要としないトランザクションは、受け入れられます。

トランザクションをリモート宛先 IMS でスケジュールできるようにするため、リソース・アクセス・セキュリティ (RAS) を使用して、その処理を許可することができます。RAS セキュリティを使用するには、トランザクションを従属領域による使用が許可されているものとして RACF に定義する必要があります。

宛先システムで RACF セキュリティ環境が利用できない場合 (/SIGN ON コマンドが RACF を用いて入力されている場合)、セキュリティ環境が動的に作成されて、トランザクション許可を進めることができます。

### 関連概念

[IMS セキュリティ \(システム管理\)](#)

## 複数システム結合機能の操作

MSC ネットワーク内の各システムは 操作上独立した装置です。各システムは、独自の通信リソースを排他的に所有し、それぞれの通信リソースは独自のマスター端末によって管理されます。

### このタスクについて

多くの場合、1 つの IMS システムで実行される MSC 操作は、対応する操作がパートナー IMS システムで実行されることを必要とします。

一般的な操作タスクには、MSC リンク上での通信の開始と停止、リンク割り当ての変更、およびリカバリが含まれます。

### 関連概念

[MSC 操作 \(オペレーションおよびオートメーション\)](#)

## MSC リンク統計

IMS は、各 MSC 論理リンクの統計を保持します。MSC メッセージをさらに効率的に処理するように MSC システムをチューニングするために、リンク統計を使用できます。

### このタスクについて

IMS は、以下の 3 つのタイプの MSC 論理リンク統計を記録します。

#### 一般統計

統計開始時刻、MSC ITASK ディスパッチ・カウント、MSC ITASK 処理時間、およびロガー検査書き込みのレートと回数

#### 送信統計

送信されたメッセージ、送信されたバイト数、送信メッセージ・サイズ、キュー・マネージャーの取得数と回数、および送信入出力回数

#### 受信統計

受信したメッセージ、受信したバイト数、受信メッセージ・サイズ、QMGR 挿入数と回数、および受信入出力回数

送信および受信の入出力回数は、MSC 帯域幅モードが使用可能な場合のみ保持されます。MSC 帯域幅モードが使用不可の場合、入出力統計の値はゼロです。

タイプ 2 コマンドの `QUERY MSLINK NAME(linkname) SHOW(STATISTICS)` を発行することによって、論理リンクの現在の統計情報を表示できます。

IMS は、論理リンクごとに統計のコピーを 2 つ保持します。IMS を最後に始動または再始動してから現在までの累計と、リンク統計の最後のチェックポイントまたは人手によるリセット以降の現在までの累計です。コマンド `QUERY MSLINK NAME(linkname) SHOW(STATISTICS)` では、最後のチェックポイントまたは人手によるリセット以降の統計情報のみを表示します。ただし、最後に IMS が再始動されてからリンク統計がリセットされていない場合を除きます。リセットされていない場合は、X'4513' ログ・レコードを調べることによってのみ、最後の IMS 再始動以降の現在までの合計を確認することができます。

システム・チェックポイントごとに、リンクの統計を手動でリセットするか、IMS に自動的に統計をリセットさせることができます。リンク統計のリセットは、パフォーマンス・テストを実行し、結果を比較するときに役立ちます。リンクの統計を手動でリセットするか、リンク統計の自動チェックポイント・リセットをオンまたはオフにするには、タイプ 2 コマンドの `UPDATE MSLINK NAME(linkname) START(STATISTICS) OPTION(reset_option)` を、適切なリセット・オプションと共に使用します。リンク統計をリセットするか、または自動リセット・モードから手動リセット・モードに変更したり元に戻したりする場合に、論理リンクを停止したり、アイドル状態にする必要はありません。

MSC 論理リンク統計をリセットしても、MSC 論理リンク統計のロギングには影響しません。MSC 論理リンク統計は、常にログに記録され、リセットすることはできません。

#### 関連タスク

[MSC リンク統計の使用によるリンクの問題の診断 \(診断\)](#)

## ベンチマーク・リンク・アクティビティー

リンク統計を照会して、MSC 論理リンクでのメッセージ・アクティビティーの通常レベルおよびハイレベルを文書化することによって、リンク処理のベンチマークを取ることができます。

### このタスクについて

リンク・アクティビティー・レベルを文書化すると、現在のアクティビティー・レベルが文書化されたレベルと異なる場合に、文書化された情報を使用して問題の認識に役立てることができます。または、以前のレベルと現在のレベルを比較して、容量制限またはボトルネックを軽減するために論理リンクまたはネットワーク容量を増やす必要があるかどうかを判断できます。

## 最適な MSC リンク・タイプの判別

論理リンク統計をリセットし、1つの標準的なメッセージを送信し、統計を記録することによって、メッセージの具体的なリソース要件を判別することができます。

### このタスクについて

単一のメッセージは、MSC 作業単位を表し、単一メッセージに対して生成された統計は、MSC 作業単位が必要とするリソース (時間、呼び出しなど) を表します。

複数のメッセージを送信してから、生成された累積統計を送信メッセージ数で除算することにより、MSC 作業単位の平均リソース要件を判別することもできます。

特定のリンク・タイプで作業単位が必要とするリソースの量が分かった後では、ご使用のインストール済み環境が複数の物理リンク・タイプ (すなわち CTC、MTM、TCP/IP、および VTAM) を使用する場合、それぞれのリンク・タイプでの作業単位のリソース要件を比較してから、最も少ない量のリソースを使用するリンク・タイプを使用することができます。

## システム・チェックポイントでの定期的な統計のリセット

MSC 論理リンク統計を手動でリセットする具体的な理由がある場合を除き、タイプ 2 コマンドの UPDATE MSLINK でリセット・オプション RESET,CHKPT を指定することによって、システム・チェックポイントで定期的にリンク統計をリセットするように IMS を構成する必要があります。

### このタスクについて

システム・チェックポイントでの自動リセットが、IMS 再始動時に有効なデフォルトのリセット・オプションです。

リセットの間隔が長くなるほど、統計が問題の判別やその他の分析のために有用ではなくなります。IMS システム・チェックポイントの間隔よりも長い間隔で、ベンチマークを実行するか、統計を収集する必要がある場合は、UPDATE MSLINK コマンドを NORESET,CHKPT パラメーターと共に発行して、チェックポイントでの自動リセットをオフにすることができます。

チェックポイントでの自動リセットをオフにすると、IMS はリンク統計をリセットしないため、新しい記録間隔を開始するには、統計を手動でリセットする必要があります。統計を手動でリセットするには、タイプ 2 コマンド UPDATE MSLINK NAME(linkname) START(STATISTICS) OPTION(RESET) を発行します。

## リンク・バッファ・サイズをメッセージのサイズに調整する

MSC 論理リンク統計を使用して、MSC リンクでのメッセージ・サイズを判別し、単一の入出力操作と 2 回の QGET (1 回の GU および 1 回の DEQ) 呼び出しでほとんどのメッセージを受信できるように、リンクおよびメッセージ・キューのバッファ・サイズを調整します。

### このタスクについて

タイプ 2 コマンドの QUERY MSLINK で返される高、低、および平均メッセージ・サイズの統計は、送信メッセージ・キューおよび受信メッセージ・キュー内のメッセージの合計サイズから導出されます。これには、X'01' および X'03' のログ・レコードに記録されている接頭部データ・セグメントおよびユーザー・データ・セグメントが含まれます。

IMS が、MSC リンクでメッセージを送信または受信するために単一の入出力操作と 2 つの QMGR 呼び出しのみを使用するように、バッファ・サイズを調整します。2 回の QGET 呼び出し (1 回の GU と 1 回の DEQ) および 2 回の QPUT 呼び出し (1 回の ISRT と 1 回の ENQ) を使用します。

MSC リンクの送信側での QPUT 呼び出しは、MSC リンクの受信側での QGET 呼び出しとは異なった動作をします。各 QPUT 呼び出しはメッセージ・セグメント用であり、各 QGET GU 呼び出しはメッセージ・キュー・バッファ用です。1 つのキュー・バッファ内に 2 つのユーザー・セグメントを持つメッセージでは、3 回の QPUT 呼び出し (2 回の ISRT 呼び出しと 1 回の ENQ 呼び出し) および 2 回の QGET 呼び出し (1



回の GU 呼び出しと 1 回の DEQ 呼び出し) が必要です。単一セグメント・メッセージの場合、IMS は MSC リンクの両側で同じ数の QPUT および QGET 呼び出しを使用します。

## MSC 帯域幅モードの論理リンク容量の調整

MSC 帯域幅モードでは、IMS は単一の送信または受信アクションで複数のメッセージを処理します。IMS が一度に複数のメッセージを処理すると、送信および受信の回数の合計は、送受信された個別のメッセージの合計数よりも少なくなります。

### このタスクについて

単一の送信または受信アクションで複数のメッセージを処理することは、処理効率の点では適切ですが、メッセージが送信されたとき、複数メッセージに対応するサイズのリンク・バッファが常にいっぱいの場合、リンクは帯域幅容量(またはそれに近いレベル)で動作している可能性があり、バッファが再度クリアされるまで待機する間にメッセージが積み重なっていく可能性があります。

論理リンクが帯域幅容量またはそれに近いレベルで動作している場合、追加の論理リンク間でメッセージ作業負荷を分散させることによって、リンクの負荷を軽減できることがあります。

## 最適な MSC リンク・バッファ・サイズの決定

リンク・バッファの最適サイズを決定するための基本的な方法は、MSC リンク統計を使用して、バッファの内容が送信される回数を実際に送信されたメッセージ数と比較することです。

### このタスクについて

帯域幅モードを使用している場合、送信されたメッセージの数が、送信されたバッファの数以上でなければなりません。帯域幅モード用に最適化されたバッファ・サイズでは、出力キュー上に複数のメッセージがある場合、バッファ内容の 1 回の送信で複数のメッセージを送信することができます。

非帯域幅モードを使用している場合、送信されるメッセージの数は、送信されるバッファの数とほぼ等しくなければなりません。非帯域幅モード用に最適化されたバッファ・サイズでは、バッファ内容の各送信で、1 つのメッセージ全体を送信することができます。

バッファ・サイズの設定が適切かどうかを調べるには、コマンド **QUERY MSLINK NAME(linkname) SHOW(STATISTICS)** を発行して、Tot\_Send\_CT (バッファ送信の総数) と Tot\_Msg\_Send\_CT (送信されたメッセージの総数) を比較します。

帯域幅モードを使用している場合は、Tot\_Msg\_Send\_CT 値が Tot\_Send\_CT 値以上でなければなりません。Tot\_Msg\_Send\_CT 値が Tot\_Send\_CT 値より大きい場合、バッファ内容の各送信で複数のメッセージが送信されています。

非帯域幅モードを使用している場合は、Tot\_Msg\_Send\_CT 値が Tot\_Send\_CT 値とほぼ等しくなければなりません。これは、バッファ内容の各送信に 1 つのメッセージ全体が含まれることを示します。Tot\_Msg\_Send\_CT 値が Tot\_Send\_CT 値より小さい場合、バッファ・サイズが 1 つのメッセージ全体を含めるのに十分な大きさではなく、1 つのメッセージ全体を送信するために複数のバッファ送信が必要になります。

### 関連資料

[QUERY MSLINK コマンド \(コマンド\)](#)

## 帯域幅モードの MSC リンク・バッファ・サイズの判別

MSC リンクが帯域幅モードで稼働する場合、リンク・バッファのサイズを判別する際に、いくらかの追加バイトを考慮する必要があります。

### このタスクについて

帯域幅モードでは、バッファ内容の各送信に複数の応答およびメッセージが含まれることがあります。しかし、出力キュー上に他に何も無い場合は、バッファ内容の送信に 1 つの応答または 1 つのメッセージしか含まれないことがあります。

バッファ作業フィールドは、バッファから送信される内容には含まれません。

表 137. 帯域幅モードの MSC BUFMS バッファのバイト・フォーマット

リンク・タイプ	バッファ作業フィールド	バッファ・ヘッダー	各応答	各メッセージ	VTAM 作業域
CTC	96	128	96	メッセージ・データのサイズ (X'01' または X'03' ログ・レコードに表 示)、プラス、144 バイトの帯域幅接 頭部	N/A
MTM	96	128	96	メッセージ・データのサイズ (X'01' または X'03' ログ・レコードに表 示)、プラス、144 バイトの帯域幅接 頭部	N/A
TCP/IP	96	240	96	メッセージ・データのサイズ (X'01' または X'03' ログ・レコードに表 示)、プラス、240 バイトの帯域幅接 頭部	N/A
VTAM	96	128	96	メッセージ・データのサイズ (X'01' または X'03' ログ・レコードに表 示)、プラス、144 バイトの帯域幅接 頭部	196

帯域幅モードの MSC リンクで使用するバッファ・サイズを計算するには、以下の手順を実行します。

## 手順

1. バッファ作業フィールド用の 96 バイトを含めます。  
バッファ作業フィールドは、常に各バッファの最初の 96 バイトです。バッファ作業フィールドは送信されません。
2. 送信される各バッファの前にあるバッファ・ヘッダー用のバイト数を含めます。バッファ・ヘッダーのサイズは、リンク・タイプによって決まります。
  - CTC リンクの場合、バッファ・ヘッダー用に 128 バイトを含めます。
  - MTM リンクの場合、バッファ・ヘッダー用に 128 バイトを含めます。
  - TCP/IP リンクの場合、バッファ・ヘッダー用に 240 バイトを含めます。
  - VTAM リンクの場合、バッファ・ヘッダー用に 128 バイトを含めます。
3. VTAM リンクの場合のみ、VTAM 作業域用の 196 バイトを含めます。
4. 応答用の 96 バイトを含めます。  
各応答は 96 バイトです。応答の数は、送信されるメッセージの数によって異なります。応答は最初にバッファに挿入されます。
5. 送信されるメッセージのサイズを判別します。

このサイズには、メッセージ接頭語とデータ、および 144 バイトの帯域幅接頭部の両方が含まれます。

送信されたメッセージのサイズを確認するには、**QUERY MSLINK NAME(linkname) SHOW(STATISTICS)** を発行し、Hi\_Msg\_Send\_SZ および Avg\_Msg\_Send\_SZ の値を調べます。QUERY MSLINK コマンドで返されるサイズには、メッセージ接頭語とデータのみが含まれ、帯域幅接頭部は含まれません。返されるサイズは、タイプ X'01' およびタイプ X'03' のキュー・マネージャー・メッセージ・サイズと同じです。

Hi\_Msg\_Send\_SZ および Avg\_Msg\_Send\_SZ の値は、送信されたメッセージの実際のサイズ (X'01' および X'03' のログ・レコードに記録されているサイズ) から導出されます。X'01' および X'03' のログ・レコード内のサイズには、各メッセージの接頭部データ・セグメントおよびユーザー・データ・セグメントが含まれます。X'01' および X'03' のログ・レコードは QLOGMSGP マクロによってマップされます。



## 関連資料

QUERY MSLINK コマンド (コマンド)

ログ・レコード (診断)

## 非帯域幅モードの MSC リンク・バッファ・サイズの判別

非帯域幅モードでは、MSC リンク・バッファのサイズを判別する際に、応答を考慮する必要はありません。

### このタスクについて

また、帯域幅モードが使用されている場合に各メッセージで送信される帯域幅ヘッダーも考慮する必要はありません。

非帯域幅モードでは、バッファ内容の各送信には、単一のメッセージのみが含まれます。応答はバッファ内容の個別オブジェクトとして送信されません。

バッファ作業フィールドは、バッファから送信される内容には含まれません。

TCP/IP リンク・タイプは、非帯域幅モードをサポートしません。

表 138. 非帯域幅モードの MSC BUFMS バッファのバイト・フォーマット

リンク・タイプ	バッファ作業フィールド	バッファ・ヘッダー	応答	メッセージ	VTAM 作業域
CTC	96	6	適用されない	メッセージ・データのサイズ (01 ログ・レコードまたは 03 ログ・レコードに表示) から 124 バイトを差し引く	適用されない
MTM	96	6	適用されない	メッセージ・データのサイズ (01 ログ・レコードまたは 03 ログ・レコードに表示) から 124 バイトを差し引く	適用されない
TCP/IP	適用されない	適用されない	適用されない	適用されない	適用されない
VTAM	96	23	適用されない	メッセージ・データのサイズ (01 ログ・レコードまたは 03 ログ・レコードに表示) から 124 バイトを差し引く	196

非帯域幅モードの MSC リンクで使用されるバッファ・サイズを計算するには、以下の手順を実行します。

### 手順

1. バッファ作業フィールド用の 96 バイトを含めます。  
バッファ作業フィールドは、常に各バッファの最初の 96 バイトです。バッファ作業フィールドは送信されません。
2. バッファ・ヘッダー用のバイト数を含めます。これは、以下のようにリンク・タイプによって決まります。
  - CTC リンクの場合は、バッファ・ヘッダー用に 6 バイトを含めます。
  - MTM リンクの場合は、バッファ・ヘッダー用に 6 バイトを含めます。
  - VTAM リンクの場合は、バッファ・ヘッダー用に 23 バイトを含めます。
3. VTAM リンクの場合のみ、VTAM 作業域用の 196 バイトを含めます。

VTAM 作業域内のデータは、メッセージと共に送信されません。

#### 4. 送信されるメッセージのサイズを計算して含めます。

非帯域幅モードでは、送信されるメッセージのサイズは、X'01' および X'03' のログ・レコードに示されたメッセージ・サイズから 124 バイトを差し引いた値です。非帯域幅モードでは、特定のメッセージ接頭語はメッセージ・データと共に送信されず、代わりに受信側 IMS システムによって再作成されます。

送信されたメッセージのサイズを確認するには、**QUERY MSLINK NAME(linkname) SHOW(STATISTICS)** を発行し、Hi\_Msg\_Send\_SZ および Avg\_Msg\_Send\_SZ の値を調べます。

Hi\_Msg\_Send\_SZ および Avg\_Msg\_Send\_SZ の値は、送信されたメッセージの実際のサイズ (X'01' および X'03' のログ・レコードに記録されたサイズから 124 バイトを減算) から導出されます。X'01' および X'03' のログ・レコード内のサイズには、各メッセージの接頭部データ・セグメントおよびユーザー・データ・セグメントが含まれます。X'01' および X'03' のログ・レコードは QLOGMSGP マクロによってマップされます。

### 例

#### 関連資料

[QUERY MSLINK コマンド \(コマンド\)](#)

[ログ・レコード \(診断\)](#)

## 高位値のリンク統計を使用した MSC リンクの問題の診断

高位値のリンク統計を使用して、MSC リンク上のメッセージのバックアップにつながる 特定の問題の診断に役立てることができます。タイプ 2 コマンドである QUERY MSLINK コマンドの出力では、高位値統計のフィールドは、すべて HI で始まります。

### このタスクについて

例えば、リンク上でメッセージのバックアップが行われた場合、Hi\_SendIO\_Time に著しく高い値が記録されたかどうかを確認します。メッセージ送信の際に著しく入出力時間がかかった場合、それは 2 つの IMS システムを接続するネットワークで障害が発生したことを示す可能性があり、これにより原因を追跡できます。

確認できる高位値の統計には、以下のものがあります。

#### Hi\_Proc\_Time

リンクが処理のためにディスパッチされた最高 (最長) の時間

#### Hi\_MSG\_Send\_SZ

送信された最大メッセージ・サイズ (タイプ X'01' または X'03' のメッセージ・レコード)

#### Hi\_QGET\_Time

送信メッセージを処理するための 最高 (最長) QMGR 呼び出し (GU または DEQ)

#### Hi\_SendIO\_Time

メッセージを送信するための最高 (最長) 入出力時間

#### Hi\_MSG\_Rec\_SZ

受信した最大メッセージ・サイズ (タイプ X'01' または X'03' のメッセージ・レコード)

#### Hi\_QPUT\_Time

受信メッセージを処理するための 最高 (最長) QMGR 呼び出し (ISRT または ENQ)

#### Hi\_RecIO\_Time

メッセージを受信するための最高 (最長) 入出力時間

## 複数システムのモニターとチューニング

MSC ネットワークの一部である IMS オンライン・システムの 統計データとパフォーマンス・データの両方  
を取得する計画を立ててください。

### このタスクについて

単一 IMS システム用のパフォーマンス・データを生成するために使用するのと同じモニター・ツールを使用  
することができます。

- IMS モニターは、いくつかのシステムで並行して実行することができます。個々の IMS システムごとに  
IMS モニター報告書を入手し、処理分析を調整します。
- 統計分析ユーティリティーは、個々のシステムごとのトランザクション・トラフィックの 要約を作成し  
ます。再度、統計学を使って図式化します。
- IMS トランザクション分析ユーティリティーでは、複数システムを通してのトランザクションをトレ  
ースし、さまざまなアクティブ物理リンクを使用してトラフィックを 調べることができます。

### パフォーマンス情報の調整

可能な場合、SYSID の数と、物理リンクおよび論理リンクの数を増やすことによって、MSC ネットワーク  
を拡張してください。

### このタスクについて

最大 2036 個の SYSID を指定できます。最大 999 個の物理リンクおよび 999 個の論理リンクを定義でき  
ます。

MSC ネットワークを拡張することにより、以下が可能となります。

- 多数の他の IMS サブシステムから、ある 1 つの IMS サブシステムに アクセスする。
- トランザクションを転送する。
- トランザクション処理を分散させる。
- ネットワークのスループットを増大する。
- 1 つの IMS システムの能力を増大する。
- 容量制約または応答時間制約に応える。

MSC に参加している各システムごとの IMS システム・ログには、そのシステムで 起きたイベントのレコー  
ドだけが入ります。ただし、リンク上のトラフィックを記録するためにロギングは実行されます。すべて  
の結合システムの SYSID を、チェックポイント間隔を記録する システム・ログの文書に追加してくださ  
い。これは、報告書を解釈するのに役立ちます。メッセージ・キューに存在はするが処理されないトラン  
ザクションがわかったり、リモート・ソースからのトランザクション・ロードの追加を予測することがで  
きるからです。

分析プロシージャには、別のシステムで発生したトランザクションによって開始される処理を分離させ  
るための方法を含める必要があります。非同期通信処理用のバッファをチューニングする上での考慮事  
項には、システム間トラフィックから例外条件が起こらないような基準を含める必要があります。

システム間処理を含む典型的なアクティビティーでモニターの 要求を満たすために、DC モニターと他のト  
レースのスケジューリングを、マスター端末オペレーター間で調整する必要があります。モニターのスパ  
ンは、まったく同じである必要はありませんが、大きく違う場合、報告書要約を平均化すると、システム  
間メッセージによって引き起こされた処理の影響を解釈することがより困難になる場合があります。

## MSC 用の IMS モニターによって生成される報告書

IMS モニター報告書印刷プログラムには、システム結合によって生じたメッセージ・イベントを強調表示する3つの報告書が含まれます。

### このタスクについて

報告書は以下のとおりです。

#### MSC トラフィック報告書

指定された間隔について、各種のリンク・パスを使用するメッセージの数を表示します。この報告書は、システム間キューイングを評価するために使用できます。

#### MSC 要約報告書

入力トランザクション名、宛先名、リンク番号、および宛先システムごとに、トラフィック・キューの要約を表示します。MSC 要約報告書は、リンクの負荷を評価するために使用できます。

#### MSC キューイング要約報告書

メッセージがキュー上にあった時間の長さ、キュー上のメッセージの数を表示します。MSC キューイング要約報告書は、システム間メッセージが宛先システムに送信される前に、ローカル・システムにキューイングされたときに生成されます。ローカル・システムは、中間システムであることが必要です。

3つの報告書はすべて、Distribution Appendix に記入項目を持つことができ、異常な伝送パターンが予想される場合に、トラフィックの頻度分布を調べることができます。分布付録には、イベントに関連する値(例えば、待ち時間)の一定の範囲内のイベントの複数のオカレンスが表示されます。例えば、待ち時間のケースで、分布付録を使用すると、特定の待ち時間を持つイベントが起こる頻度、および、ある範囲の待ち時間内に、そのイベントがどのように分散しているかが簡単にわかります。

### 関連概念

[IMS モニター報告書 \(システム管理\)](#)

[MSC トラフィック報告書 \(システム管理\)](#)

[MSC 要約報告書 \(システム管理\)](#)

[MSC キューイング要約報告書 \(システム管理\)](#)

[分布付録の解釈 \(システム管理\)](#)

## 複数システム・トランザクション統計の抽出

ログ・トランザクション分析ユーティリティーを使用して、ローカル・システム内およびシステム間の両方のメッセージ・トラフィックを得ることができます。

### このタスクについて

異なるタイプの物理リンクを通る伝送も調べることができます。(そのアクティビティーは、論理リンク・パスのステップごとに合計されます。)すべてのパートナー・システム・アクティビティーを反映したシステム・ログ入力をIMSに提供する必要があります。すなわち、各MSCシステムごとにシステム・ログのセットが必要です。個々のシステム・ログのセットを調整するには、ログ・マージ・ユーティリティーを使用します。9個の別々のシステム・ログをマージすることができます。各ログは、MSCがインストールされている、固有に識別されたIMSシステムの出力です。

### 関連資料

[ログ・トランザクション分析ユーティリティー \(DFSILTA0\) \(システム・ユーティリティー\)](#)

## ログ・マージの制御

ログ出力を制御するには、複数のステップを実行する必要があります。

### このタスクについて

ログ出力を制御するために、下記のことが必要です。

- ・ 調査したい論理リンク・パスに関係する必要なシステムを選択します。
- ・ システムごとに一連の入力ログが類似のタイム・スパンをカバーするよう調整します。
- ・ 特定の時間間隔に対するシステム間処理のサンプルを取りたい場合には、ログ・マージ・ユーティリティー制御ステートメントに開始と停止時刻を指定します。

開始日(ユリウス)と時刻の両方または時刻だけを指定することができます。これらの時間が適用されるのは、最初のシステム・ログ(LOG01 DD ステートメントによって指定される)です。他のログ・アクティビティーは、最初のシステム・ログの最初と最後のイベント間で発生する場合に収集されます。

- ・ トランザクション分析ステップに適切なログ・レコードを選択するために MSG を指定します。(ALL レコードがデフォルトですが、これは、いくつかのシステムに対する DL/I アクティビティーが、ユーティリティー入力に含められることを意味し、処理時間の延長につながる可能性があります。)

#### 関連資料

[ログ・トランザクション分析ユーティリティー \(DFSILTA0\) \(システム・ユーティリティー\)](#)

[ログ・マージ・ユーティリティー \(DFSMTMG0\) \(システム・ユーティリティー\)](#)

## トランザクション分析報告書の解釈

IMS トランザクション分析ユーティリティーによって作成されるログ分析報告書を使用すると、任意のシステムで処理される個々のトランザクションについて統計を得ることができます。

ログ分析報告書には以下の統計が含まれます。

総応答時間  
入出力キュー時間  
処理時間

ログ分析報告書には、IMS トランザクション分析ユーティリティーによって作成される詳細な報告書レコードの形式の定義、および処理タイプ・コードのリストもこの資料に収められています。報告書の行でメッセージ GU 呼び出しまたは MPP 終了に関して時間が示されていないということは、入力ソースまたは中間システム報告書の行であることを示します。

処理タイプ・フィールドは、明細報告書の行を解釈する上で重要です。コード S は、この行がトランザクションの送受信イベントであることを示します。コード C、D、P、X および Y を用いて、システム間会話の進捗よくをトレースすることができます。

報告書の見出しには、メッセージ・キュー時間に対する GU についての列の後に、ID で始まる列があります。ID 見出しの下の報告書行に表示される数は、ログ入力ログ・マージ・ユーティリティーに送られた順序と一致します。このフィールドは、明細報告書レコードでは、開始位置が 102 の、SYSTEM ID という名前の 3 桁のフィールドに相当します。

ソート・ステップを使用することにより、報告書レコードの順序を任意の順序、例えば、トランザクション・コード内のシステム ID 順などに変更することができます。デフォルトの順序は、入力順です。

#### 関連概念

[統計分析報告書](#)、[ログ・トランザクション報告書](#)、および[ログ・レコード分析 \(システム管理\)](#)

#### 関連資料

[ログ・トランザクション分析ユーティリティー \(DFSILTA0\) \(システム・ユーティリティー\)](#)

## MSC および共用キューを使用する IMSplex

以下のトピックでは、複数システム結合機能 (MSC) ネットワークと共用キューを使用する IMSplex の共存について説明します。MSC と IMSplex の共存には、MSC ネットワークから IMSplex 構成へマイグレーションした時などの一時的な共存もあれば、MSC リンクが IMSplex を IMSplex 外の IMS システムに接続した時などの永続的な共存もあります。

MSC ネットワークと IMSplex が共存する際の主要な考慮事項は、MSC 環境と IMSplex 環境の両方にわたるトランザクション・メッセージの適正な経路指定と処理です。これは、それぞれの環境で異なる経路指定方式が使用されるからです。



通常、MSC ネットワークは SYSID を使用して、トランザクションを特定の IMS システムに 経路指定しますが、一方、共用キューを使用する IMSplex は、トランザクションを共用キュー上でトランザクションに関するインタレストを登録した 任意の IMS システムに対して使用可能にすることにより、トランザクションを経路指定します。MSC ネットワークと共用キューを使用する IMSplex が共存する場合、これらの経路指定方式は両方ともトランザクションに適用されます。

## MSC 環境と IMSplex 環境にわたるメッセージ経路指定

MSC は、ローカル SYSID とリモート SYSID および宛先名を使用して、MSC ネットワーク内の ローカル、中間、およびリモート IMS システム にわたってメッセージを経路指定します。共用キューを使用する IMSplex は、宛先名登録、IMSID、共用キュー、および IMS システムの 登録済みインタレストを使用して、IMSplex 内のフロントエンドおよびバックエンド IMS システム間でメッセージを経路指定します。IMSplex ネットワークと MSC ネットワークが共存する場合は、両方の経路指定方式が使用できます。

MSC ネットワークと共用キューを使用する IMSplex では、IMS がメッセージを作成してローカル・キューまたは共用キューに置く際に、IMS は SYSID、宛先名、および IMSID 値をメッセージ接頭語に保管します。メッセージ接頭語に保管された SYSID、宛先名、または IMSID は、IMSplex または MSC ネットワーク内のいずれかにある IMS システム の SYSID、宛先名、または IMSID に一致していなければなりません。それらが一致しない場合、共用キュー上あるいは MSC ネットワーク内で未完了のメッセージはすべて、IMS システムが メッセージ処理を試みた際に、疑似異常終了 U0830 などの経路指定エラーをトリガーします。この状態は、例えば、SYSID、宛先名、または IMSID を所有する IMS システムが 停止したか、またはその値を変更した場合などに発生することがあります。

## MSC ネットワーク内でのメッセージの経路指定方法

MSC ネットワークでは、IMS は、 リモート・トランザクション、リモート論理端末 (LTERM) 名、および SYSID を使用して、MSC ネットワークを通してメッセージを経路指定します。

リモート・トランザクションは、CREATE TRAN コマンドを使用して動的に定義するか、TRANSACT マクロ内でリモートとローカルの SYSID を指定して静的に定義します。リモート LTERM は、CREATE LTERM コマンドを使用して動的に定義するか、ステージ 1 システム定義内の MSNAME マクロの直後に NAME マクロで静的に定義します。

メッセージが、実際の宛先名を含む IMS システムに 到達する前に中間 IMS システムをパススルー しなければならない場合は、MSNAME ラベルを中間宛先名として使用することもできます。この場合、MSNAME ラベルも メッセージ接頭語に保管されます。

### 関連概念

696 ページの『[宛先名および SYSID によるメッセージの経路指定](#)』

MSC ネットワーク内のメッセージには、IMS システム間でメッセージの経路指定を可能にする情報が含まれています。

### 関連タスク

705 ページの『[複数システム結合リソースの定義](#)』

複数システム結合機能 (MSC) リソース (物理リンク、論理リンク、システム ID など) は、IMS の実行時に (IMS タイプ 2 コマンドを使用して) 動的に定義することも、IMS システム定義時に (マクロを使用して) 静的に定義することもできます。

### 関連資料

[MSNAME マクロ \(システム定義\)](#)

[NAME マクロ \(システム定義\)](#)

[TRANSACT マクロ \(システム定義\)](#)

[CREATE TRAN コマンド \(コマンド\)](#)

[CREATE LTERM コマンド \(コマンド\)](#)

## 共用キューを使用する IMSplex 内でのメッセージの経路指定方法

共用キューを使用する IMSplex 内では、IMS はカップリング・ファシリティに登録された発信元名と宛先名を使用して、フロントエンドとバックエンドの IMS システム間でメッセージを経路指定します。

発信元名と宛先名は、論理端末名、トランザクション・コード、または APPC か OTMA クライアント名にすることができます。ユーザーは、システム定義時に静的発信元名および静的宛先名を定義します。ユーザーは、記述子ライブラリー内またはユーザー出口内に、動的 LTERM、ETO 端末、または動的トランザクションである発信元名と宛先名を定義します。APPC または OTMA クライアントの発信元名と宛先名は、APPC または OTMA クライアントによる会話の割り振り時に 8 バイトのトークン形式で IMS に送信されます。

トランザクションが開始され、従属領域または MSC リンクが開始されてメッセージの処理が作動可能になると IMS はトランザクションを登録します。LTERM が開始されるか、またはサインオンしてメッセージの処理が作動可能になると、IMS は LTERM 名をカップリング・ファシリティに登録します。APPC または OTMA 会話が割り振りされると、IMS は APPC および OTMA トークンを登録します。APPC または OTMA 会話が割り振り解除されると、IMS は APPC および OTMA トークンを登録解除します。

メッセージは、これらの発信元名と宛先名をメッセージ接頭語内に持っています。カップリング・ファシリティ上の空の共用キューがフロントエンド IMS システムから登録済み宛先名用のメッセージを受信すると、カップリング・ファシリティはインタレストを登録したすべての IMS システムに、処理すべきメッセージがあることを通知します。

## IMSplex と MSC ネットワークが共存する場合のメッセージ経路指定

共用キューを使用する IMSplex と MSC ネットワークが共存する場合、IMS は、発信元名と宛先名、MSC SYSID、および IMSID を使用して、フロントエンド、バックエンド、およびリモート IMS システムの間でメッセージを経路指定します。

IMSplex 内では、IMS は、カップリング・ファシリティと共用キューを使用してメッセージを経路指定します。IMS が MSC リンクを使用するのは、メッセージを IMSplex 外の IMS システムに経路指定する場合だけです。IMSplex 内の IMS システム間に存在する MSC リンク定義はすべて使用されません。

MSC ネットワーク内での経路指定用に定義されたトランザクションは、共用キューを使用する IMSplex 内でも経路指定できますが、経路指定に使用されるのは IMSplex のメソッドであり MSC のメソッドではありません。ただし、IMSplex はトランザクションの MSC SYSID 属性を認識します。IMSplex は、SYSID を認識することにより、リモート・トランザクションを MSC ネットワークに対するのと同様の方法で処理することができます。また、このことにはローカル処理類似性に関する含みがあります。

MSC ネットワーク内での経路指定用に明確に定義されていないトランザクションは、MSC リンクを使用して IMSplex の外側で経路指定することはできません。

### 関連概念

733 ページの『IMSplex 内での類似性の処理』

メッセージを IMSplex 内の特定の IMS システム上で処理する必要がある場合、IMS は、IMS システムの IMSID を宛先名に付加して、その IMS システムにメッセージ類似性を割り当てます。これで、その IMS のみがメッセージを処理できます。

## IMSplex 内での類似性の処理

メッセージを IMSplex 内の特定の IMS システム上で処理する必要がある場合、IMS は、IMS システムの IMSID を宛先名に付加して、その IMS システムにメッセージ類似性を割り当てます。これで、その IMS のみがメッセージを処理できます。

TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン (DFSMSCEO) を使用して、IMSplex 内のトランザクション・インスタンスと IMS システム間に類似性を確立することもできます。

## IMSplex-MSC 共存構成内の類似性処理

MSC ネットワーク内では、リモート・トランザクションは、そのトランザクションを入力として受信する IMS システムに対してトランザクションが持つ可能性があるいかなるローカル類似性もバイパスします。



IMSplex 内でも、リモート・トランザクションはローカル類似性のバイパスを引き起こします。他の点では、IMSplex で類似性処理を規定している規則に変更はありません。

### 関連概念

750 ページの『[TM/MSC メッセージ経路指定および制御ユーザー出口ルーチンの概要](#)』

メッセージ経路指定は、経路指定オプションまたはメッセージの制御を提供する TM/MSC メッセージ経路指定 / 制御ユーザー出口ルーチン (DFSMSCE0) を使用しない限り、定義済みスキームに基づいて自動的に行われます。

### 関連タスク

738 ページの『[MSC と IMSplex が共存する場合の APPC および OTMA のリモート・トランザクション管理](#)』

以下のトピックには、同期 APPC および OTMA メッセージが MSC ネットワークでリモートに処理された場合の MSC ネットワークから IMSplex へのマイグレーションと、IMSplex と MSC ネットワークが共存する場合 IMSplex 外のリモート IMS システム上での APPC および OTMA メッセージの処理に関する情報が記載されています。

## MSC ネットワークから IMSplex ネットワークへのマイグレーション

既存の MSC ネットワークを共用キューを持つ IMSplex にマイグレーションあるいは変換する場合、共用キュー上に MSNAME 定義を使用する トランザクション・メッセージがある間は、それらの定義を IMS システムに保存する必要があります。

### このタスクについて

また、IMSplex 外の IMS システムに対する MSC リンク定義もすべて保存する必要があります。IMSplex のテストが完了し、IMSplex 内のどの IMS システムも MSC のみのネットワークに戻す必要がないことが確実な場合は、IMSplex メンバー・システム間の MSC リンク定義はすべて削除することができます。

IMSplex 内では、1 つの IMS システムが MSC 対応の場合は、IMSplex 内のすべての IMS システムが MSC 対応でなければなりません。IMS システムを MSC 対応にするには、システム定義時に IMS システムに対して MSC リンクを定義する必要があります。この MSC リンク定義は、機能させる必要はありません。

### IMSplex 内の MSC リンク定義

MSC ネットワークから IMSplex へのマイグレーション・プロセスの間、必要とする限り MSC リンク定義を保存することができます。これにより、ユーザーは、IMSplex を共用キュー環境でテストし、その後 MSC ネットワークに戻って調整を行うことができます。

共用キューを使用する IMSplex が確立されたら、IMSplex 内の IMS システム間に新規の MSC リンクを追加することはできません。共用キューを使用する同じ IMSplex 内の 2 つの IMS システム間で MSC リンクを開始しようとすると、IMS はメッセージ DFS2149 を出してリンクの開始を打ち切ります。

### IMSplex 内の MSNAME 定義と SYSID の共用

共用キューを使用する IMSplex 内で MSNAME ステートメントが組み込まれた IMS システムを始動する時に、IMSplex は、結合する IMS システムの MSNAME ステートメントとローカル SYSID を、既に IMSplex 内に存在する他の IMS システムと共用します。

結合する IMS システム内の各 MSNAME ステートメントごとに、IMSplex は、IMSplex 内の他のそれぞれの IMS システム内に、重複動的 MSNAME ステートメントを作成します (ただし、MSNAME ステートメントが既に存在する場合は除きます)。

結合する IMS システムが所有する各ローカル SYSID ごとに、IMSplex は、重複ローカル SYSID を、IMSplex 内の他のそれぞれの IMS システムの SYSID テーブルに追加します。これにより、結合する IMS システムの SYSID が IMSplex 全体のローカル SYSID として有効になります。MSNAME ステートメントを持つ IMS システムが IMSplex に結合または再結合する場合、IMSplex は、常に IMSplex 内の各 IMS システムの SYSID テーブルを更新します。

動的 MSNAME ステートメントを生成し、IMSplex 全体でローカル SYSID を共用することにより、IMSplex は、MSC ネットワーク上の単一の IMS システムとして機能することができます。また、MSC ネットワー

クで実行されるように定義されたトランザクションが、共用キューを使用する IMSplex の分散処理を利用できるようになります。

動的 MSNAME と共用 SYSID を説明するために、以下の図では、共用キューを使用する IMSplex を導入する前の単純な MSC ネットワークを示します。次に 735 ページの図 122 では、2 つの IMS システム間で共用キューを使用する IMSplex が作成された後の、同じ MSC ネットワークを示します。

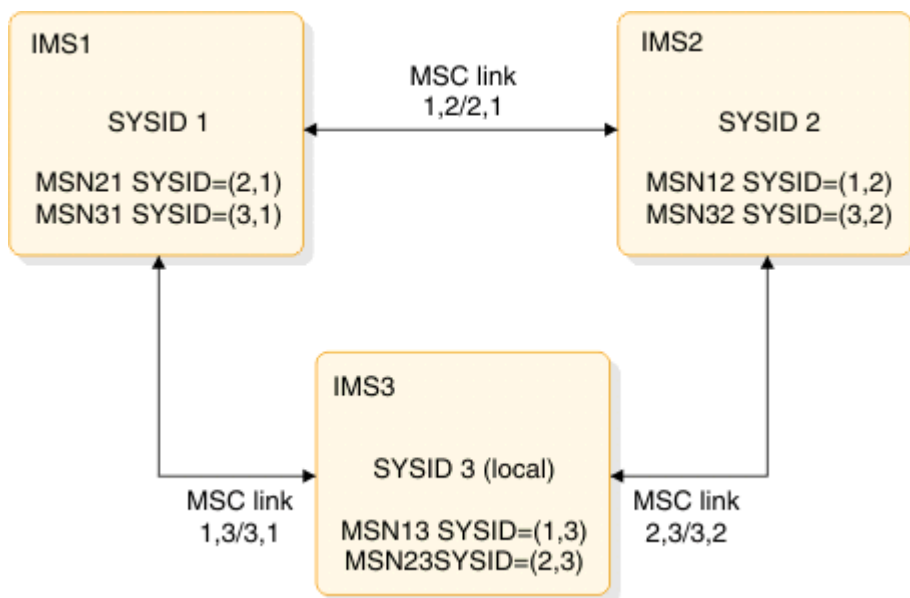


図 121. IMSplex を持たない MSC ネットワーク

上記の図では、IMS1、IMS2、および IMS3 は、それぞれ MSC ネットワークのメンバーです。それぞれのメンバーのローカル SYSID は固有であり、それらの MSNAME ステートメントは、各 IMS システムでローカルに定義されているリンクのみを定義します。

#### IMSplex with shared queues

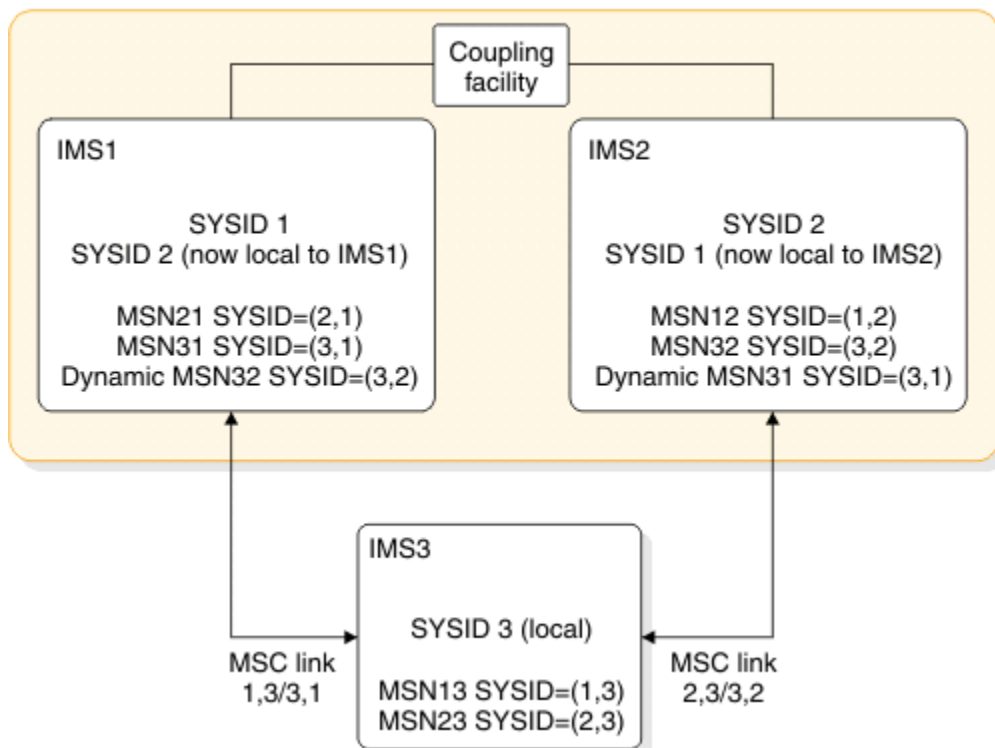


図 122. 共用キューを使用する IMSplex と共存する MSC ネットワーク

上記の図では、IMS1 および IMS2 は共用キューを使用する IMSplex の一部になっています。このため、SYSID 1 は、IMS2 に対してローカルとなり、SYSID 2 は IMS1 に対してローカルとなります。同様に、動的 MSNAME MSN31 が IMS2 内に、そして動的 MSNAME MSN32 が IMS1 内に現われます。また、IMS1 および IMS2 は、MSC ネットワーク内の単一の IMS システム またはノードとして機能するようになりましたが、それでも IMS1 および IMS2 は MSC ネットワークおよび IMS3 への接続を保ちます。

以下の表では、735 ページの図 121 に表示された IMSplex 導入前の IMS システムの SYSID テーブルを示します。

表 139. IMSplex を持たない MSC ネットワーク内の SYSID 所有権

SYSID	IMS1	IMS2	IMS3
1	ローカル	RMT/MSN12	RMT/MSN13
2	RMT/MSN21	ローカル	RMT/MSN23
3	RMT/MSN31	RMT/MSN32	ローカル

以下の表では、735 ページの図 122 に示される MSC ネットワーク への IMSplex 導入の結果としての、各 IMS の SYSID テーブルを示します。

表 140. IMSplex と共存する MSC ネットワーク内の SYSID 所有権

SYSID	IMS1	IMS2	IMS3
1	ローカル	ローカル	RMT/MSN13
2	ローカル	ローカル	RMT/MSN23
3	RMT/MSN31 動的 MSN32	RMT/MSN32 動的 MSN31	ローカル

IMS システムが IMSplex を離脱する場合に、IMSplex が、残った IMS システム内の SYSID テーブル を変更したり、動的 MSNAME ステートメントを削除したりすることはありません。これにより、離脱した IMS の MSNAME 定義 を使用するメッセージが共用キュー上に存在する場合に、IMSplex がそれらのメッセージを処理できる状態が保たれます。IMS システムが IMSplex に再結合する場合、IMSplex は常に、MSNAME ステートメントを再交換し、再結合される SYSID を IMSplex 内のすべての SYSID テーブルで再検証します。

IMS システムが共用キューを使用する IMSplex に正常に結合、あるいはその IMSplex から正常に離脱するときは、IMSplex は、IMSplex 内の各 IMS システムの マスター端末にメッセージ DFS0778I を出します。

## 共用キューを使用する IMSplex 内での MSNAME 重複

MSC ネットワークと共用キュー・グループの両方に参加する各 IMS システム については、MSNAME および関連したリモート SYSID およびローカル SYSID が、IMS の初期設定時に、共用キュー・グループ内の他の IMS システムと 交換されます。

各 IMS システム内に、共用キュー・グループ内の他の IMS システムで定義された MSNAME に対する 動的 MSNAME が作成されます。

共通 MSC SYSID 経路指定テーブルが、各 MSNAME に定義されるリモート SYSID 値 およびローカル SYSID 値で指定される各 IMS 内に作成されます。各 IMS システムで ローカルな SYSID は、共用キュー・グループ内の他の IMS に対してもローカル となります。共用キュー・グループ内のすべての IMS システム上に同一のローカル SYSID が存在することにより、共用キュー・グループ外の IMS システム から MSC リンクを通して受信されたメッセージを、共用キュー・グループ内の任意の IMS システムに 経路指定して処理することができます。

共用キュー・グループ内の IMS システムの 重複 MSNAME は、同一のリモート SYSID を持つ必要があります。これは、MSNAME とリモート SYSID が同義語であるからです。同一の MSNAME は同一のリモート SYSID に 関連する必要があります。同一の MSNAME に対して異なるローカル SYSID を持つことは可能です。

同一の MSNAME が、2つの IMS システム内で異なるリモート SYSID を使用して定義された場合、この MSNAME は、共用キュー・グループ内での MSNAME の交換時に無視されます。同一の MSNAME を持つ IMS システム内では 動的 MSNAME は作成されず、各 IMS は、それぞれの独自バージョンの MSNAME を保持します。

次の 2つの例はそれぞれ、共用キュー・グループ内で同一名を持つ MSNAME の有効な組み合わせを示します。

- 有効な MSNAME ペア:
  - (IMSA 上) MSN12345 MSNAME SYSID=(1,2)
  - (IMSB 上) MSN12345 MSNAME SYSID=(1,2)
- 同様に有効な MSNAME ペア:
  - (IMSA 上) MSN12345 MSNAME SYSID=(1,2)
  - (IMSB 上) MSN12345 MSNAME SYSID=(1,3)

次の例は、共用キュー・グループ内で同一名を持つ MSNAME の無効な組み合わせの例です。

- (IMSA 上) MSN12345 MSNAME SYSID=(1,2)
- (IMSB 上) MSN12345 MSNAME SYSID=(2,1)

## 関連概念

[リソース・タイプ整合性 \(システム管理\)](#)

## IMSplex 内の SYSID テーブルからの MSNAME 定義の削除

IMSplex から SYSID または MSNAME ステートメントを削除するには、それを所有する IMS システムから、まずそれを削除し、次に IMS システムをコールド・スタートして IMSplex 内に戻す必要があります。

### このタスクについて

SYSID または MSNAME ステートメントを、それを所有するオリジナルの IMS システムから 削除した後に、IMSplex 内の他の IMS システムをそれぞれウォーム・スタートまたはコールド・スタートし、それらの SYSID テーブルを再作成します。

## MSC から IMSplex へのマイグレーション完了時の MSPLINK 定義と MSLINK 定義の除去

MSC ネットワークから共用キューを使用する IMSplex へのマイグレーションが完了し、どの IMS システムも MSC のみの状態に戻さないことが 確実な場合は、使用しなくなった MSC リンクの MSPLINK 定義と MSLINK 定義を削除する必要があります。

### このタスクについて

注: IMSplex 全体にわたる MSC 使用可能性を維持するために、IMSplex の各 IMS システム内に少なくとも 1つの MSC リンク定義を 保持する必要があります。このリンクは、IMSplex 外の IMS システムへの 機能する MSC リンクでも、MSNAME ステートメントを付加し MSC を使用可能にするためのみに定義された機能しないリンクのどちらでも 構いません。

## MSC と IMSplex が共存する時の SYSID の管理

ローカル SYSID は、MSC ネットワーク内で 固有でなければなりません。ローカル SYSID は、IMSplex 内で 固有である必要はありません。ただし、IMSplex 外の MSC ネットワークに リンクされた IMSplex メンバーが所有するローカル SYSID は、その MSC ネットワーク内で 固有である必要があります。

また、IMSplex の複数メンバーが IMSplex 外の同一 IMS システムに リンクする場合、各 IMSplex メンバーは、その外部 IMS システムに 固有のローカル SYSID を使用する必要があります。

同一のリモート SYSID を使用して、MSC ネットワーク内の異なる IMS システムを 指すことはできません。

## IMSplex 内の MSC SYSID 複製

IMSplex は、IMSplex 内の各ローカル SYSID を IMSplex 内の各 SYSID テーブルに追加しますが、IMSplex 内 MSC リンクを除去する際には、IMSplex 内の各 IMS システムにあるすべてのローカル SYSID を複製する必要があります。ローカル SYSID を複製するには、それを含む MSNAME ステートメントを複製します。

### このタスクについて

ローカル SYSID を複製することにより、SYSID を所有する IMS システムが利用不能で、さらに、その SYSID が IMSplex 内の SYSID テーブルに含まれていない場合でも、IMSplex は確実に共用キュー上のメッセージを経路指定し、処理することができます。MSC SYSID は、IMSplex 内のカップリング・ファシリティーには保管されません。アプリケーション・プログラム・メッセージの処理時に IMS システムが共用キュー上のメッセージの SYSID をローカルとして認識できない場合、IMSplex は疑似異常終了 U0830 を発行します。

## IMSplex と MSC が共存する時の IMSID

IMSplex は、IMS システムの識別、および特定の IMS システムへの類似性を持つすべてのメッセージの経路指定に、IMSID を使用します。IMSID は、IMSplex 内の各 IMS で固有でなければなりません。

IMSplex-MSC ネットワーク内では、IMSID は MSC ネットワーク全体 (IMSplex 外のリモート MSC IMS システムも含む) にわたって固有でなければなりません。

IMSID は、IMSplex を含まない MSC ネットワーク内で固有である必要はありません。MSC ネットワークを共用キューを使用する IMSplex にマイグレーションする時は、必ず、各 IMS システムの IMSID を固有のものにしてください。

## MSC と IMSplex が共存する場合の APPC および OTMA のリモート・トランザクション管理

以下のトピックには、同期 APPC および OTMA メッセージが MSC ネットワークでリモートに処理された場合の MSC ネットワークから IMSplex へのマイグレーションと、IMSplex と MSC ネットワークが共存する場合 IMSplex 外のリモート IMS システム上での APPC および OTMA メッセージの処理に関する情報が記載されています。

### MSC ネットワーク内 APPC および OTMA メッセージのリモート処理

MSC ネットワーク内では、同期 APPC および OTMA メッセージをリモート MSC トランザクションにキューイングすることにより、それらのメッセージに対するローカル類似性をバイパスすることができます。

### このタスクについて

リモート IMS システムは、APPC および OTMA メッセージを非同期に、かつ APPC または OTMA 会話とは別個に処理します。IMS は、リモート IMS システムに APPC または OTMA 会話を伝搬あるいはカスケードしません。IMS は、APPC または OTMA トークンをメッセージ接頭語に保管します。応答またはプログラム間通信が、発信元の IMS システムに戻ると、IMS は、同期 APPC または OTMA 会話型モードをメッセージに復元しようとします。

MSC ネットワーク内の APPC または OTMA メッセージのリモート処理を使用可能にするには、メッセージをリモート MSC トランザクションのキューに入れます。

### 共用キューを使用する IMSplex 内の APPC または OTMA トランザクション・メッセージのバックエンド処理

共用キューを使用する IMSplex では、リモート MSC トランザクションを使用して、APPC または OTMA メッセージを処理のためにバックエンド IMS システムに経路指定することができます。

リモート MSC トランザクションを使用すると、APPC または OTMA クライアントから受信したトランザクション・メッセージに適用されるローカル類似性制約を回避することができます。その結果、トランザク

ションを領域に割り当てられるローカル MSC トランザクションとして定義する IMSplex 内の IMS システムはすべて、そのトランザクションを処理できるようになります。

MSC ネットワークから IMSplex にマイグレーションする場合、既存のリモート MSC トランザクションを使用して、IMSplex 内のバックエンド IMS システム上で APPC または OTMA メッセージを処理し、APPC および OTMA 類似性制約をバイパスすることができます。トランザクションがローカル IMSplex トランザクションに変換される場合は、RRS=Y および AOS=Y を指定しない限り、APPC および OTMA 類似性制約が引き続き適用されます。

APPC または OTMA メッセージが、フロントエンド IMS システムで、リモート MSC トランザクションとして定義されているトランザクションへのキューに入れられる場合、IMS は、どの IMS システムへの類似性も指定せずにトランザクション・メッセージを共用キューに挿入します。

バックエンド IMS システムは、共用キューから APPC および OTMA トランザクション・メッセージを取得すると、APPC または OTMA 会話トークンをメッセージ接頭語に保管し、トランザクション・メッセージを APPC または OTMA 会話から切り離して、その会話とは非同期に処理します。IMS は、APPC または OTMA 会話を、バックエンド IMS システムにも、プログラム間通信後に APPC または OTMA メッセージを処理する可能性のある他のどの IMS システムにも (フロントエンド IMS システムを含む)、伝搬あるいはカスケードしません。

**注:** OTMA 会話から独立して処理されている OTMA トランザクションが、変更可能な PCB への CHNG 呼び出しを発行し、IMS が OTMA 宛先解決ユーザー出口 (OTMAYPRX) を発行すると、OTMAYPRX ユーザー出口は、OTMA クライアントが最初にトランザクションをサブミットした IMSplex 内でそのトランザクションが処理されている場合にのみ、そのトランザクション・メッセージを OTMA メッセージとして認識します。

発信元フロントエンド IMS システムが最初のあるいは唯一の応答を受信すると、クライアントが引き続き同期モードで接続されていると想定して、応答は APPC または OTMA 同期モードでクライアントに返されます。IMS は、同じ対話についてのクライアントへの後続の応答はすべて非同期で返します。最初のあるいは唯一の応答が返される前にクライアントが終了していた場合、応答は破棄されるのではなく、代わりに非同期でクライアントへのキューに入れられます。

例えば、MSC トランザクションをフロントエンド IMS システムで、開始済み MSC リンクに割り当てずにリモートとして定義し、次に、同じ MSC トランザクションをバックエンド IMS でローカルとして定義して、トランザクションを開始済み領域に割り当てた場合、APPC または OTMA クライアントがフロントエンド IMS システムで同期モードでトランザクションを開始すると、フロントエンド IMS システムはクライアントを識別するためにメッセージ接頭語にトークンを保管し、フロントエンド IMS システムへの類似性を何も指定せずに、トランザクションを共用キューに入れます。

バックエンド IMS システムが共用キューからトランザクションをリトリブすると、トランザクションは非同期で処理され、APPC/OTMA クライアントから切り離されます。

以下の一連の図は、リモート MSC トランザクションを使用して、APPC または OTMA トランザクションを IMSplex 内のバックエンドで処理するため、および MSC リンクを使用して IMSplex の外部で処理するための各種の可能なシナリオを示しています。

次の図は、リモート MSC トランザクションを使用して、IMSplex 内のバックエンドで APPC または OTMA トランザクションを処理する単純な構成を示しています。トランザクション TRAN1 は IMS 1 に対してリモート MSC トランザクションとして定義されていますが、TRAN1 は IMS 1 内の MSC リンクに割り当てられていないため、IMS 1 はこれを共用キューに入れます。

トランザクションを IMS 1 にサブミットした後、APPC または OTMA クライアントは受信状態で同期応答を待ちます。ただし、IMS 1 と IMS 2 による TRAN1 のキューイングおよび処理は、クライアントで維持される同期通信モードから独立して IMS によって処理されます。



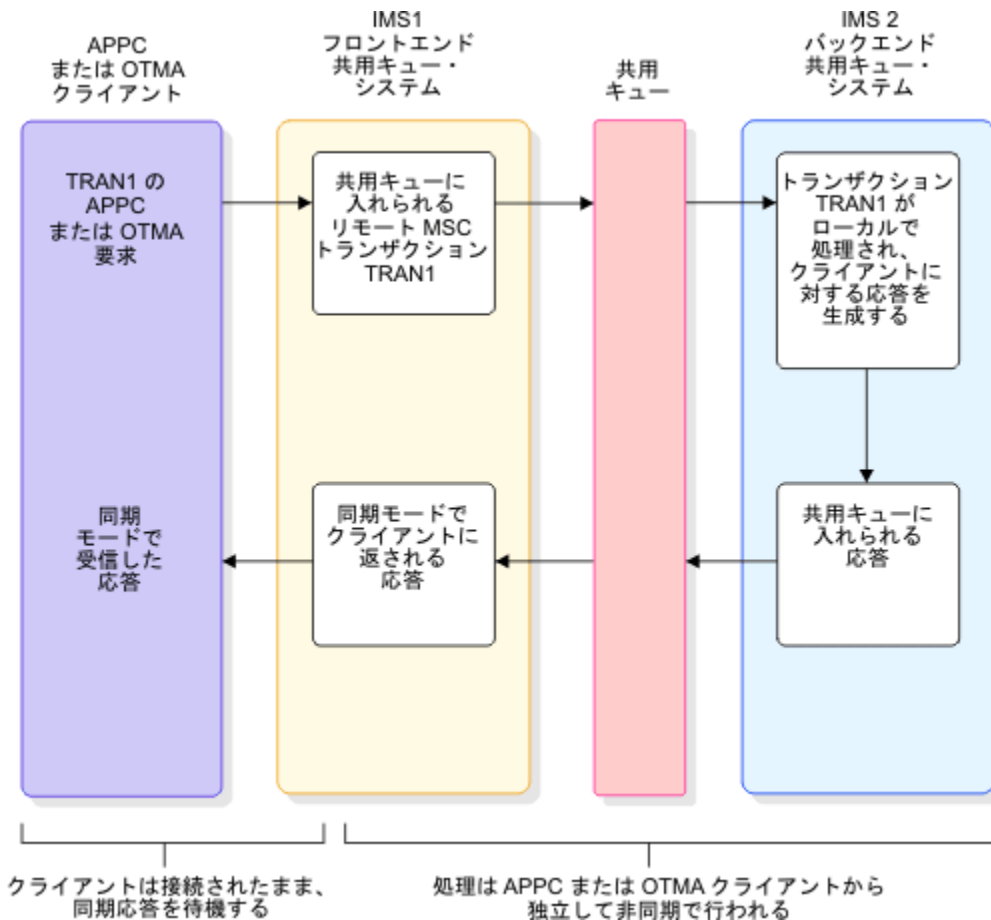


図 123. リモート MSC トランザクションを使用した、APPC または OTMA トランザクションのバックエンド IMS システムへの経路指定

次の図は、リモート MSC トランザクションを使用して APPC または OTMA トランザクションを経路指定する、別のシナリオを示しています。このシナリオには 3 つの IMS システムが含まれており、TRAN1 は、共用キューと MSC リンクの両方を経由してリモート MSC IMS システムに経路指定されます。

IMS 1 がクライアントからトランザクションを受け取ると、IMS 1 はそのトランザクションを共用キューに入れます。IMS 2 でトランザクション TRAN1 はリモート MSC トランザクションとして定義され、IMS 3 への MSC リンクに割り当てられます。IMS 3 は応答を生成します。応答は MSC リンクを介して IMS 2 に返され、次に共用キューを経由して IMS 1 に返されます。IMS 1 は、クライアントが予期する同期モードで、応答を APPC または OTMA クライアントに返します。



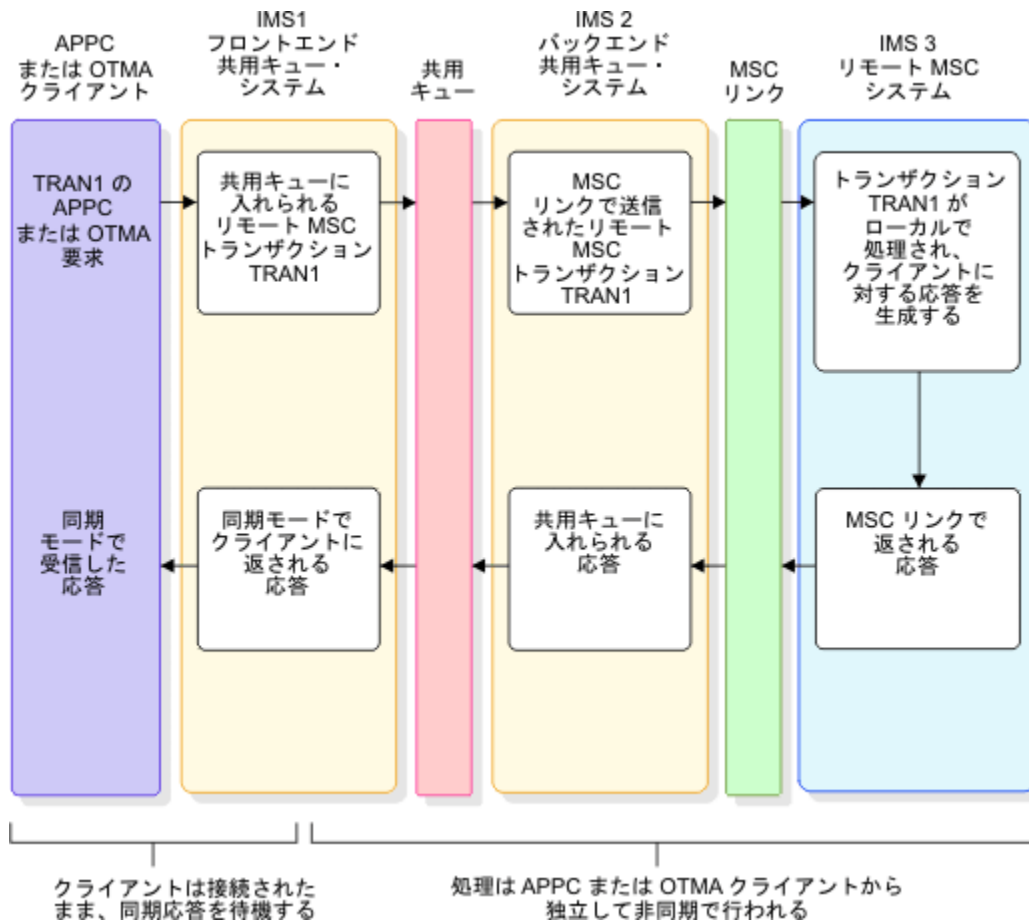


図 124. 共用キューと MSC リンクを経由する APPC または OTMA トランザクションの経路指定

次の図に示されたシナリオでは、APPC または OTMA トランザクション TRAN1 が、共用キュー・グループ内のバックエンド IMS システムである IMS 2 に経路指定されます。ここで、TRAN2 へのプログラム間通信が行われ、処理は共用キューを経由して IMS 1 に送信されます。TRAN2 は MSC トランザクションではありません。次に、TRAN2 は TRAN3 への別のプログラム間通信を実行し、処理を IMS 2 に送信します。TRAN3 も非 MSC トランザクションです。TRAN3 は共用キューに入れられ、IMS 2 によってリトリブされます。ここで、TRAN3 は処理され、クライアントに対する応答を生成します。応答は共用キューを経由して IMS 1 に返されます。IMS 1 は、クライアントが予期する同期モードで応答をクライアントに返します。

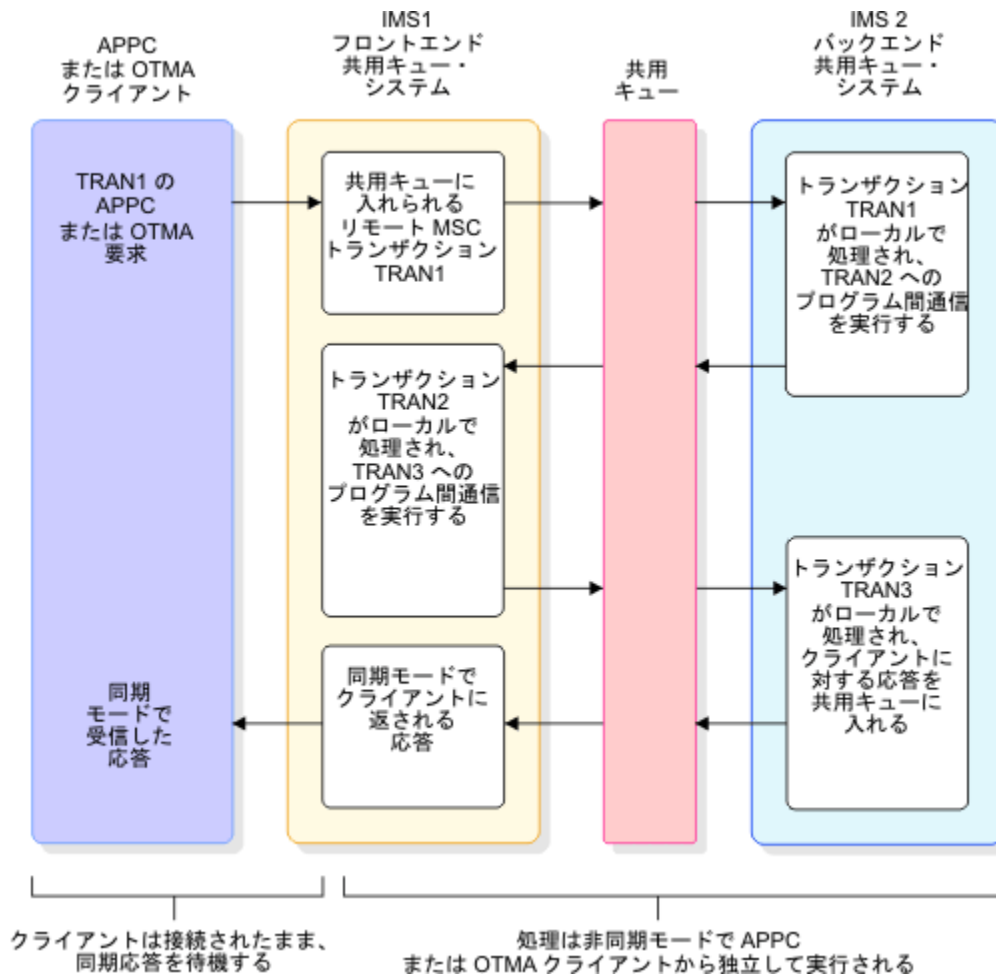


図 125. 複数のプログラム間通信を伴う、リモート MSC トランザクションと共用キューを使用した、APPC または OTMA トランザクションの経路指定

次の図に示されているシナリオには、3つのIMSシステムが含まれます。IMS1は、共用キューを経由して、TRAN1をIMS2にリモートMSCトランザクションとして経路指定します。TRAN1はIMS2でローカルに処理され、TRAN2へのプログラム間通信を実行します。TRAN2はIMS2でリモートMSCトランザクションとして定義されます。IMS2は、MSCリンクを介してTRAN2をIMS3に送信します。TRAN2はIMS3によって処理され、クライアントに対する応答を生成します。IMS3は、MSCリンクを介して応答をIMS2に返します。IMS2は、応答を共用キューに入れます。IMS1は応答をリトリブし、クライアントが想定する同期モードでAPPCまたはOTMAクライアントに応答を返します。

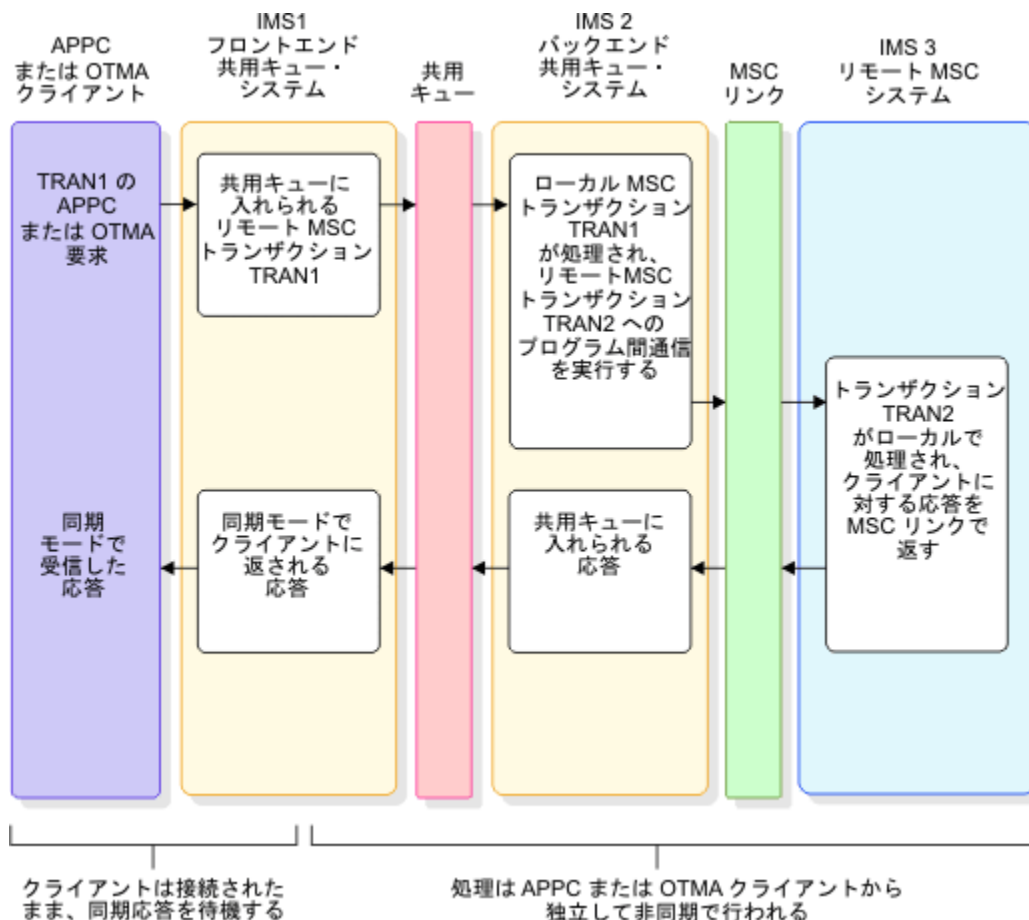


図 126. 1つのプログラム間通信を伴う、MSCと共用キューを使用して経路指定される APPC または OTMA トランザクション

### 関連概念

シプレックス環境での APPC および OTMA メッセージの管理 (システム管理)

## リモート・トランザクションを使用する APPC メッセージおよび OTMA メッセージのバックエンド処理の有効化

リモート・トランザクションを使用して IMSplex 内のバックエンド IMS システム上で APPC メッセージおよび OTMA メッセージを処理するには、以下のステップを実行します。

### 手順

- IMSplex 内の各 IMS システムに少なくとも 1 つの MSC リンクを定義して、IMSplex 内のすべての IMS システムで MSC を使用可能にします。この MSC リンクは、IMSplex 外のリモート IMS システムへの機能するリンクとするか、あるいは、そのようなシステムが 1 つも存在しない場合は、機能しない MSC リンクとすることができます。
- APPC クライアントまたは OTMA クライアントに接続された IMS システムで、TRANSACT マクロの SYSID キーワードを指定して、APPC メッセージまたは OTMA メッセージ用のリモート・トランザクションを定義します。  
SYSID キーワードでは、リモート SYSID には任意のリモート SYSID を指定できます。ローカル SYSID は、APPC クライアントまたは OTMA クライアントに接続された IMS システムの SYSID と一致する必要があります。
- すべてのバックエンド IMS システム内に、APPC メッセージまたは OTMA メッセージを処理するためのローカル・トランザクションを定義する。

4. トランザクションとそれを処理する従属領域を開始して、リモート・トランザクションを IMSplex に登録する。

## IMSplex 外部の MSC システムへのトランザクションの送信

MSC リンクを使用して、IMSplex から IMSplex 外部のリモート IMS システムに APPC トランザクションまたは OTMA トランザクションを送信するには、以下の手順を実行します。

### 手順

1. IMSplex 内の各 IMS システムに少なくとも 1 つの MSC リンクを定義して、IMSplex 内のすべての IMS システムで MSC を使用可能にします。この MSC リンクは、IMSplex 外のリモート IMS システムへの機能するリンクとするか、あるいは、そのようなシステムが 1 つも存在しない場合は、機能しない MSC リンクとすることができます。
2. IMSplex とリモート IMS システムとの間に MSC リンクを定義する。
3. IMSplex 内の任意の IMS システムで、リモート MSC トランザクションを定義し、TRANSACT マクロの SYSID キーワードにリモート SYSID とローカル SYSID を指定してそのリモート・トランザクションを MSC リンクに割り当てる。  
リモート SYSID は、リモート MSC IMS システムの SYSID と一致する必要があります。ローカル SYSID は、APPC クライアントまたは OTMA クライアントに接続された IMS システムの SYSID と一致していません。
4. リモート MSC IMS システム内に、APPC または OTMA トランザクション・メッセージを処理するためのローカル・トランザクションを定義する。

## 疑似異常終了 U0830 の回避

MSC ネットワークと共用キューを使用する IMSplex が共存する場合、メッセージが共用キューに残っているか MSC ネットワーク内で未完了の間に、IMSplex または MSC ネットワークに対して IMS システムを削除または追加すると、エラーが発生することがあります。

### このタスクについて

メッセージの経路指定に使用する定義 (SYSID、IMSID、または宛先名など) を変更した場合にも、エラーが発生することがあります。

MSC を使用する IMSplex 内では、IMS システムが トランザクション・メッセージを処理のために共用キューから取得する際に、IMS は、SYSID、IMSID、および宛先名の妥当性検査を行います。妥当性検査が失敗した場合、IMS は トランザクション・メッセージを共用キュー上に残し、疑似異常終了 U0830 を発行します。以下の予防措置を実行すると、ほとんどの U0830 疑似異常終了を回避することができます。

- IMS システムのマイグレーションまたは移動の前に、そのマイグレーションまたは移動する IMS システムを宛先とする全トランザクション・メッセージを共用キューおよび MSC ネットワークからデキューする。
- SYSID、IMSID、または MSNAME を変更する前に、それらの ID を使用する全トランザクション・メッセージを共用キューおよび MSC ネットワークからデキューする。
- 新規 IMS システムの IMSplex へのマイグレーション後、すべての IMS システムが IMSplex に結合し、すべての SYSID テーブルが更新されるまで待機してから従属領域を開始してメッセージを処理する。
- IMSplex 内で MSC 定義を持つ複数の IMS システムを再始動する時に、IMSplex 内のすべてのローカル SYSID が各 IMS システム内に複製されていない場合は、すべての IMS システムが再始動を完了するまで待機してから従属領域を開始する。再始動が完了するまでは、一部の SYSID テーブル内に、複製されていない SYSID のすべてが組み込みされていない場合があります。
- IMSplex 内の各 IMS システムに、MSC が定義されていることを確認する。MSC が定義されていない IMS によってキューに入れられたメッセージは、MSC 拡張接頭部を持たず、SYSID 値はゼロになります。このメッセージが、MSC 機能を持つ IMSplex 内の別の IMS システムによって処理された場合、疑似異常終了 U0830 が発生し、戻りコードは 08 です。

ある IMSplex 内で、1 つの IMS システムが MSC 対応の場合は、その IMSplex 内のすべての IMS システムが MSC 対応でなければなりません。IMS システムを MSC 対応にするには、システム定義時に IMS システムに対して MSC リンクを定義する必要があります。この MSC リンク定義は、機能させる必要はありません。

疑似異常終了 U0830 に関連付けられた戻りコードに関する詳細情報またはその他の診断情報については、「IMS V15 メッセージおよびコード 第 3 巻: IMS 異常終了コード」を参照してください。

## MSC TCP/IP 汎用リソース

TCP/IP 汎用リソースを使用すると、リモート MSC 対応 IMS システムは、特定の IMS システムの代わりに TCP/IP 汎用リソース・グループと MSC リンクを確立することができます。

特定の IMS システムに接続する代わりに TCP/IP 汎用リソース・グループに接続すると、リモート IMS システムに大きな影響を与えることなく、IMSplex 内の 1 つの IMS システムから別のシステムに MSC 物理リンクを簡単に移動できるようになります。

例えば、XRF テークオーバーまたは定期保守のために IMSplex 内の物理リンクのパートナーを切り替える必要がある場合、リンクが短時間停止することを除いて、切り替えは IMSplex 外部の IMS システムには透過的に行われます。新しいリンク・パートナーを参照するようにリモート MSC 定義を変更する必要はありません。

IMSplex 内の IMS システムは、IMS.PROCLIB データ・セット内のそれぞれの DFSDCxxx メンバーの GENIMSID パラメーターで、共用の汎用 IMS ID を指定することによって、TCP/IP 汎用リソース・グループに参加します。

リンクの両側にある IMS システム内の物理リンクは、TCP/IP 汎用リソースを使用しない場合と同じ方法で定義します。唯一の相違点は、リモート IMS システムが、MSPLINK システム定義マクロまたは CREATE MSPLINK コマンドの NAME パラメーターで、特定の IMS システムの IMS の代わりに、共用の汎用 IMS ID を指定することです。

TCP/IP 汎用リソースは、IMSplex 環境または XRF 環境でサポートされます。

MSC TCP/IP 汎用リソースは、MSC VTAM 汎用リソース (VGR) と同様に機能します。ただし、VTAM VGR では、共用の汎用 IMS ID の代わりに、共通の汎用 APPLID 名を使用する点を除きます。

物理リンク上の最初の論理リンクが TCP/IP 汎用リソース・グループに接続すると、論理リンクは、リンク要求を受け入れるために、グループ内の最初の IMS システムとの類似性を確立します。物理リンク上の後続のすべての論理リンクも同じ IMS システムへの類似性を持ちます。

リンク属性を表示すると、類似性に従う論理リンクには AFFIN という状況が表示されます。類似性は、その IMS システムへのすべての論理リンクが正常に終了するまで、各論理リンクに対して有効のままです。リンクが ERE 状況で終了した場合、類似性状況は残ります。類似性状況は、IMS のウォーム・スタート (NRE または ERE) 後も残ります。

IMS Connect は、TCP/IP 汎用リソース・グループに参加している IMS システム間のリンクの経路指定を管理します。IMS Connect は、どの IMS システムがどの MSC 物理リンクとの類似性を持っているかを追跡します。

### 関連タスク

[MSC の TCP/IP 汎用リソース・グループの定義 \(システム定義\)](#)

### 関連資料

[CREATE MSPLINK コマンド \(コマンド\)](#)

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

[MSC ステートメント \(システム定義\)](#)

## TCP/IP 汎用リソース・グループ内の MSC リンクの管理

MSC で TCP/IP 汎用リソースが使用される場合、TCP/IP 汎用リソース・グループに接続する各リンクは、グループ内の特定の IMS システムへの類似性を持ちます。

### このタスクについて

類似性は、1つの物理リンク上のすべての論理リンクの並列セッションを1つの IMS システムに制限します。論理リンクが正常に終了すると、類似性は IMS 内でクリアされます。

リンクが類似性を持っている場合、/DISPLAY AFFIN LINK または QUERY MSLINK コマンドを使用すると、そのリンクは AFFIN の状況で表示されます。QUERY MSLINK コマンドは、類似性を持つ IMS システムの IMS ID も表示します。

IMS Connect は、TCP/IP 汎用リソース・グループへのリンク要求の経路指定を管理します。IMS Connect 内の MSC リンクの定義で、TCP/IP 汎用リソース・グループの共用の汎用 IMS ID を、グループに参加している個別の IMS システムと関連させます。

リモート IMS システムで TCP/IP 汎用リソース・グループへのリンクが開始されると、開始要求は、IMS Connect によって、グループ内のすべての使用可能な IMS システムに経路指定されます。IMS Connect は、リンクを受け入れる最初の IMS システムにリンク要求を渡し、リンクの類似性が設定されます。IMS Connect は、その後、同じ物理リンク上のすべての論理リンクの並列セッションを、その IMS システムに経路指定します。

TCP/IP 汎用リソース・グループに1つだけ IMS システムが含まれている場合には、どちら側からリンクが開始されたかに関係なく、初期リンク要求は常にそのシステムによって受け入れられます。

**制約事項:** 物理リンクが複数の IMS システム内で定義されている TCP/IP 汎用リソース・グループでは、グループ内のいずれかの IMS システム上で論理リンクが PSTOPPED ERE 状態の場合、グループ内の他のいずれかの IMS システムで同じ物理リンクを使用している論理リンクを始動しないでください。別のシステム上の同じ物理リンクで論理リンクを始動するには、その前に正常にシャットダウンするか COLD 状態にリセットすることによって、ERE リンクのアフィニティをクリアする必要があります。

予防措置として、別の IMS システム上で再始動してリンクを移動する前に、現行の IMS システム上で MSC リンクのアフィニティの状況を表示するようにします。アフィニティがまだアクティブな場合は、リンクを正常にシャットダウンして、アフィニティの状況をリセットします。リンクを正常にシャットダウンできない場合は、タイプ 1 コマンド **/CHANGE LINK linknum FORCSESS | SYNCSESS COLDSESS** またはタイプ 2 コマンド **UPDATE MSLINKNAME(linkname) SET(SYNCOPT(COLDSESS))** のいずれかを実行することによって、リンクを COLD 状態に設定します。

**重要:** リンクの状況を ERE から COLD に変えるか、ERE 状況にあるリンクを移動して別の IMS システム上で再始動すると、始動時のメッセージ順序番号の同期が妨げられ、メッセージが重複したり消失したりすることがあります。

TCP/IP 汎用リソース・グループに複数の IMS システムが参加している場合は、以下の方法を使用して、物理リンク上の最初の論理リンクを開始するときに、どの IMS システムが類似性を取得するかを制御することができます。

### 手順

- TCP/IP 汎用リソース・グループ内の、類似性を確立する必要がある IMS システムからリンクを開始します。
- IMS Connect 内で、汎用リソース・グループ内の、類似性を必要としない IMS システムへの MSC 物理リンク・パスを停止します。
  - a) IMS Connect で物理リンク・パスを停止するには、以下のいずれかの IMS Connect コマンドを発行します。
    - IMS タイプ 2 フォーマット・コマンド **UPDATE IMSCON TYPE(MSC) NAME(lclPlkid) STOP(COMM)**
    - WTOR フォーマット・コマンド **STOPMSC lclPlkid**
    - z/OS MODIFY コマンド **UPDATE MSC NAME(lclPlkid) STOP(COMM)**

- b) 類似性を必要とする IMS システムを除くすべてのシステムへのリンクが停止された後、リモート IMS システムからリンクを開始します。
- リンク接続を防止する各 IMS システム内で、物理リンク上の汎用リソース・グループへのログオンを停止します。
  - a) リンク接続を防止する必要がある各 IMS システムに対して、以下のいずれかの IMS コマンドを発行することで、IMS システム内のログオンを停止することができます。
    - IMS タイプ 2 コマンド **UPDATE MSPLINK NAME(msplink\_name) STOP(GENLOGON)**
    - IMS タイプ 1 コマンド **/PSTOP MSPLINK msplink\_name**
  - b) リンクの接続先となる IMS システムを除くグループ内のすべての IMS システムに対して汎用リソース・グループへのログオンが停止された後、リモート IMS システムからリンクを開始します。

#### 関連資料

[UPDATE IMSCON TYPE\(MSC\) コマンド \(コマンド\)](#)

[UPDATE MSLINK コマンド \(コマンド\)](#)

[/RSTART コマンド \(コマンド\)](#)

[STOPMSC コマンド \(コマンド\)](#)

[STARTMSC コマンド \(コマンド\)](#)

[IMS Connect UPDATE MSC コマンド \(コマンド\)](#)

## TCP/IP 汎用リソース・グループ内の MSC リンク類似性の持続性

TCP/IP 汎用リソース・グループ内の特定の IMS システムへの MSC リンクの類似性は、リンクの開始時に確立され、リンクが正常に停止して NRE 状況または COLD 状況と見なされた時点でクリアされます。

類似性を持つ IMS システムに障害が起きた場合、IMS Connect は通常、構造化呼び出しインターフェース (SCI) から通知を受け取ります。障害が通知されると、IMS Connect は障害が起きた IMS システムへのリンク・パスを終了し、リンク類似性をクリアします。

IMS Connect が障害の通知を受け取らなかった場合は、リンク類似性が持続することがあります。持続しているリンク類似性は、別の IMS システムでその物理リンクを再始動する前に、IMS Connect 内で IMS Connect コマンドを発行して手動でクリアする必要があります。

#### 関連タスク

746 ページの『[TCP/IP 汎用リソース・グループ内の MSC リンクの管理](#)』

MSC で TCP/IP 汎用リソースが使用される場合、TCP/IP 汎用リソース・グループに接続する各リンクは、グループ内の特定の IMS システムへの類似性を持ちます。

747 ページの『[TCP/IP 汎用リソース・グループ内の MSC リンク類似性のクリア](#)』

MSC 論理リンクの並列セッションは、TCP/IP 汎用リソース・グループ内の特定の IMS システムへの類似性を持っています。類似性が持続している間、リンクを別の IMS システムに再割り当てすることはできません。TCP/IP の汎用リソース・グループ内のリンクの類似性をクリアするには、物理リンク上のすべての論理リンク並列セッションを停止する必要があります。

#### 関連資料

[UPDATE IMSCON TYPE\(MSC\) コマンド \(コマンド\)](#)

[STOPMSC コマンド \(コマンド\)](#)

[STARTMSC コマンド \(コマンド\)](#)

[IMS Connect UPDATE MSC コマンド \(コマンド\)](#)

## TCP/IP 汎用リソース・グループ内の MSC リンク類似性のクリア

MSC 論理リンクの並列セッションは、TCP/IP 汎用リソース・グループ内の特定の IMS システムへの類似性を持っています。類似性が持続している間、リンクを別の IMS システムに再割り当てすることはできません。TCP/IP の汎用リソース・グループ内のリンクの類似性をクリアするには、物理リンク上のすべての論理リンク並列セッションを停止する必要があります。



## このタスクについて

TCP/IP 汎用リソース・グループ内の特定の IMS システムへの MSC 論理リンクの類似性をクリアするには、以下の手順を実行します。

### 手順

- 以下のいずれかの IMS コマンドまたはコマンド・シーケンスを発行して、論理リンクを停止します。
  - IMS タイプ 2 フォーマット・コマンド **UPDATE MSPLINK** NAME(*mslink\_name*) STOP(COMM)
  - IMS タイプ 2 フォーマット・コマンド **UPDATE MSPLINK** NAME(*mslink\_name*) STOP(COMM) OPTION(FORCE) SET(SYNCOPT(COLDSSESS))
  - IMS タイプ 1 フォーマット・コマンド **/PSTOP LINK** *link\_num*
  - IMS タイプ 1 フォーマット・コマンド **/PSTOP LINK** *link\_num* と、後続の SYNCSESS COLDSSESS または FORCSESS COLDSSESS キーワードを指定したタイプ 1 コマンド **/CHANGE LINK** *link\_num* のコマンド・シーケンス。

## 次のタスク

IMS でリンクが停止されると、リンク類似性はクリアされます。

### 関連タスク

746 ページの『[TCP/IP 汎用リソース・グループ内の MSC リンクの管理](#)』

MSC で TCP/IP 汎用リソースが使用される場合、TCP/IP 汎用リソース・グループに接続する各リンクは、グループ内の特定の IMS システムへの類似性を持ちます。

### 関連資料

[QUERY IMSCON TYPE\(MSC\) コマンド \(コマンド\)](#)

[/RSTART コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(MSC\) コマンド \(コマンド\)](#)

[VIEWMSC コマンド \(コマンド\)](#)

[STOPMSC コマンド \(コマンド\)](#)

[STARTMSC コマンド \(コマンド\)](#)

[IMS Connect QUERY MSC コマンド \(コマンド\)](#)

[IMS Connect UPDATE MSC コマンド \(コマンド\)](#)

## IMS Connect 内の MSC リンク類似性のクリア

通常、IMS で物理リンク上のすべての MSC 論理リンクが正常に終了すると、IMS Connect は自動的に MSC TCP/IP リンク類似性をクリアします。IMS Connect 内に類似性が持続している場合は、IMS Connect 内で物理リンク・パスを停止して再始動することで、類似性をクリアできます。

## このタスクについて

IMS Connect 内で、IMS システムへの MSC 物理リンクの持続する類似性をクリアするには、以下の手順を実行します。

### 手順

- 何らかの理由で IMS Connect 内のリンク類似性がクリアされない場合は、以下のいずれかの IMS Connect コマンドを発行して、IMS Connect 内で物理リンク・パスを停止することができます。
  - IMS タイプ 2 フォーマット・コマンド **UPDATE IMSCON** TYPE(MSC) NAME(*lclPlkid*) STOP(COMM)
  - WTOR フォーマット・コマンド **STOPMSC** *lclPlkid*
  - z/OS MODIFY フォーマット・コマンド **UPDATE MSC** NAME(*lclPlkid*) STOP(COMM)

## 次のタスク

リンクを再使用するには、以下のいずれかのコマンドを発行して、IMS Connect 内でリンクを再始動する必要があります。

- IMS タイプ 2 フォーマット・コマンド **UPDATE IMSCON TYPE(MSC) NAME(lclPlkid) START(COMM)**
- WTOR フォーマット・コマンド **STARTMSC lclPlkid**
- z/OS MODIFY フォーマット・コマンド **UPDATE MSC NAME(lclPlkid) START(COMM)**

## 関連タスク

746 ページの『TCP/IP 汎用リソース・グループ内の MSC リンクの管理』

MSC で TCP/IP 汎用リソースが使用される場合、TCP/IP 汎用リソース・グループに接続する各リンクは、グループ内の特定の IMS システムへの類似性を持ちます。

## 関連資料

[QUERY MSLINK コマンド \(コマンド\)](#)

[/PSTOP コマンド \(コマンド\)](#)

## XRF、MSC、および TCP/IP 汎用リソース

アクティブ IMS システムと代替 IMS システムの XRF ペアが TCP/IP 汎用リソースを使用して MSC をサポートしている場合、MSC 物理リンクとそのすべての並列セッションは、アクティブ IMS システムへの類似性を持ちます。

TCP/IP 汎用リソースをサポートするには、XRF ペア内のアクティブ IMS システムと代替 IMS システムの両方の DFSDCxxx PROCLIB メンバーに、同じ GENIMSID パラメーターが指定されていなければなりません。

テークオーバーが発生すると、IMS は、論理リンクの類似性をアクティブ IMS システムから代替 IMS システムに切り替え、これが新しいアクティブ IMS システムになります。

IMS は、IMSplex 内のアクティブ IMS システムの障害を IMS Connect にも通知します。IMS Connect は、障害の発生時にアクティブであったソケットを終了し、MSC 物理リンクと障害が起きた IMS システム間の類似性を解放します。

障害が起きた IMS システムへの類似性が解放されたことを確認するには、物理リンクの状況を表示するための IMS Connect コマンドを発行します。IMS Connect が類似性状況を解放していない場合 (この状態は、IMS Connect がテークオーバーを通知されていなかった場合などに起きることがあります)、MSC リンクを停止して再始動することによりソケットをクリーンアップするために、該当する IMS Connect コマンドを発行します。

IMS Connect 内の類似性状況が削除されたら、新しいアクティブ IMS システム上で MSC リンクを開始することができます。新しいアクティブ IMS システムでリンクが開始された後は、リンクは新しいアクティブ IMS システムとの類似性を持ち、IMS Connect は新しい類似性状況を表示します。

XRF の代替 IMS システムは、IMSplex 外部のリモート IMS システムからの再始動要求に応答しません。テークオーバーが完了し、代替 IMS システムが新しいアクティブ IMS システムになるまでは、代替 IMS システム上の MSC リンクは開始できません。

## 関連タスク

[MSC の TCP/IP 汎用リソース・グループの定義 \(システム定義\)](#)

## 関連資料

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

[QUERY IMSCON TYPE\(MSC\) コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(MSC\) コマンド \(コマンド\)](#)

[VIEWMSC コマンド \(コマンド\)](#)

[STOPMSC コマンド \(コマンド\)](#)

[STARTMSC コマンド \(コマンド\)](#)

[IMS Connect QUERY MSC コマンド \(コマンド\)](#)

[IMS Connect UPDATE MSC コマンド \(コマンド\)](#)

## VTAM 汎用リソース (VGR) および MSC

IMSplex 外部の MSC IMS システムは、IMSplex 内で特定の IMS システムの APPLID を使用するのではなく、VTAM 汎用リソース・グループ名 (GRSNAME) を使用して、IMSplex によって MSC リンク・セッションを確立できます。

VTAM GRSNAME を使用すると、IMSplex 内部の IMS システム間で MSC リンク・セッションを移動できます。そのとき、IMSplex の外部にある IMS システムごとに、CREATE MSPLINK コマンドや MSPLINK ステージ 1 システム定義マクロ内の NAME パラメーターで VTAM ノード名を変更する必要はありません。

VTAM GRSNAME を使用すると、IMSplex 内部の IMS システムの複製を容易にすることができます。それは、IMSplex 内の各 IMS システムが、同じ MSC リンク定義を使用して、リモート IMS システムに接続できるためです。IMSplex 内の各並列リンクは、共通の GRSNAME を使用して、IMSplex の外部にある IMS システムとのセッションを確立できます。パートナー ID、SYSID、などのその他のリンク属性も、IMSplex 内にある IMS システム間で複製することができます。

IMSplex 内の VTAM 汎用リソースの使用は、外部 MSC システムに対して透過的です。外部 MSC システムは IMSplex が使用する GRSNAME を指定するだけで済み、CREATE MSPLINK コマンドや MSPLINK マクロの NAME パラメーターで特定の IMS システム APPLID 名を指定する必要はありません。IMSplex では、IMSplex 内の各 IMS システムの DFSDCxxx PROCLIB メンバーに MSCVGR=Y (MSCVGR=N がデフォルト) を指定することによって、MSC VGR サポートが使用可能になります。IMSplex では、VTAM 汎用リソースを使用可能にする必要もあります。

MSC VGR サポートが使用可能である場合は、DFSDCxxx PROCLIB メンバーで GRMESTAE=Y (デフォルト) も指定して、IMS 異常終了が発生した場合に IMS が VTAM の類似性を確実に削除するようにします。タイプ 1 コマンド **/DISPLAY AFFIN LINK** およびタイプ 2 コマンド **QUERY MSLINK SHOW(AFFIN)** は、MSC リンクと IMSplex 内の IMS システム間の現在の類似性を表示します。

### 関連概念

[VTAM 汎用リソース・グループの計画 \(システム管理\)](#)

### 関連資料

[IMS PROCLIB データ・セットの DFSDCxxx メンバー \(システム定義\)](#)

[CREATE MSLINK コマンド \(コマンド\)](#)

[QUERY MSLINK コマンド \(コマンド\)](#)

[/DISPLAY AFFIN コマンド \(コマンド\)](#)

## TM/MSC メッセージ経路指定および制御ユーザー出口ルーチンの概要

メッセージ経路指定は、経路指定オプションまたはメッセージの制御を提供する TM/MSC メッセージ経路指定 / 制御ユーザー出口ルーチン (DFSMSCEO) を使用しない限り、定義済みスキームに基づいて自動的に行われます。

DFSMSCEO 出口ルーチンは、単一パラメーター・リストと、メッセージ経路指定およびセキュリティーをカスタマイズするためにオプションのユーザー接頭部をメッセージに追加する機能を提供します。

DFSMSCEO 出口ルーチンは、共用キューを使用する IMSplex の IMSplex 類似性経路指定オプションも提供します。類似性経路指定は、出口ルーチンで指定された IMS システムとトランザクション・インスタンスとの間に類似性を確立します。DFSMSCEO 出口ルーチンが類似性をトランザクションに割り当て、そのトランザクションが共用キューに挿入されると、宛先 IMS システムのみがその通知を受け取ります。

**注:** この出口ルーチンの使用に際して、複数システム結合機能 (MSC) は必要ではありませんが、そのオプションの多くは MSC が使用可能な場合のみ機能します。

この経路指定出口ルーチンはメッセージ宛先が確認される前に呼び出されます。出口ルーチンのエンター・ポイントは次のとおりです。

- **端末経路指定 (TR):** メッセージが端末から受信されたとき、制御を受信します。
  - VTAM メッセージ (TRVTAM)
  - APPC メッセージ (TRAPPC)
  - OTMA メッセージ (TROTMA)

- **リンク受信 (LR):** メッセージが MSC リンク上で受信されたとき、制御を受信します。
  - ローカル・トランザクション・メッセージ (LRTRAN)
  - ローカル LTERM メッセージ (LRLTERM)
  - ローカル直接経路指定メッセージ (LRDIR)
  - 中間メッセージ (LRINT)
- **プログラム経路指定 (PR):** アプリケーション・プログラムが CHNG または ISRT 呼び出しを発行してメッセージを挿入した時に、制御を受信します。
  - アプリケーション・プログラム CHNG 呼び出し (PRCHNG)
  - アプリケーション・プログラム INST 呼び出し (PRINST)

TM/MSC メッセージ経路指定 / 制御ユーザー出口ルーチンは、IMS.SDFSRESL の前に連結される JOBLIB、STEPLIB、または LINKLIST ライブラリー内に存在していれば、IMS 初期設定時にロードされます。この出口ルーチンを呼び出すために必要なシステム定義または始動パラメーター・モジュールはありません。

オプションのユーザー定義の接頭部をメッセージに追加することができます。メッセージ接頭語は、例えば、メッセージ・セキュリティ、ユーザー・アカウント、および統計要件をカスタマイズしたり、出口ルーチン相互間で通信を許可して経路指定制御を増大したりするために使用できます。このユーザー・メッセージ接頭語セグメントは、それぞれの出口ルーチン・エンタリー・ポイントの呼び出し時、メッセージが MSC/TM ネットワークを介して経路指定されるときに、メッセージに追加および更新することができます。これらの他の出口ルーチンで接頭部セグメントを読み取ったり更新することができます。ユーザー接頭部セグメントはオフラインで使用できます。ユーザー接頭部のサイズは 512 バイトに制限されており、合計のメッセージ接頭語サイズは大きなメッセージ・キュー LRECL に制限されています。

TM/MSC メッセージ経路指定 / 制御ユーザー出口ルーチンは、エンタリー・ポイントのすべてに対して、共通パラメーター・リスト・インターフェース (DFSMSCEP) を使用します。すべてのパラメーター・フィールドを参照するための DSECTS が用意されています。ユーザーは、ユーザー出口モジュールの前にある DFSMCCSV マクロ内のポインターを変更して再アセンブルすることによって、出口ルーチンが制御を受け取るエンタリー・ポイントを選択することができます。IMS は、エンタリー・ポイントが存在していれば、その出口ルーチンを呼び出します。

**推奨事項:** パラメーター・フィールドの共通参照のために DSECT DFSMSCEP マクロを用意すると、追加データ (経路指定を決定する際に使用するパフォーマンス・データやパス可用性データなど) を将来、出口ルーチンに簡単に渡せるようになります。SIO 率、応答時間、キュー・カウントをデータとして含めることができます。

## 端末経路指定

TM/MSC メッセージ経路指定 および制御ユーザー出口ルーチンの端末経路指定エンタリー・ポイントは、入力システムに存在します。出口ルーチンは、メッセージが端末から受信された時に呼び出されます。この出口ルーチンは指定宛先 (LTERM または トランザクション・コード) を検査し、指定されている場合は、その宛先をリジェクトする、あるいはローカルまたはリモートの宛先に変更することができます。この出口ルーチンが宛先を変更しない場合には、最初に指定した宛先が経路指定に使用されます。この出口ルーチンは、代わりに、ローカル LTERM または トランザクションを オーバーライドしてリモート IMS へのメッセージを経路指定することもできます。

IMS は、会話を継続している端末やコマンドに対しても、出口ルーチンを呼び出しません。

水平区分化を用いる構成において、出口ルーチンを使用してすべての入力メッセージの評価を行い、その入力メッセージの最初のセグメントの情報に基づいて適切な処理システムにそれらの入力メッセージを回送することが可能です。トランザクションとリンクが適切に定義されている場合、出口ルーチンを使用して入力メッセージ・セットの共通宛先を設定することもできます。宛先システムに着信するとすぐに、メッセージは、それらの個々のトランザクション・コードに基づいて処理されます。

## リンク受信

TM/MSC メッセージ経路指定 / 制御ユーザー出口ルーチン (DFSMSCE0) のリンク受信エンタリー・ポイントは、IMS が MSC リンクからメッセージを受信したときに、そのメッセージのトランザクション・コード

または LTERM 名を変更することができます。DFSMSCEO は、メッセージが現在の IMS システムでローカルに処理されるように要求することも、メッセージを異なるリモート IMS システムに転送することもできます。DFSMSCEO は、宛先として使用されるトランザクション・コードまたは LTERM 名を検査し、指定されている場合はリジェクトする、または同じ属性を持つ別のトランザクション・コードに変更することができます。出口ルーチンはメッセージの最初のセグメントを検査し、新しいトランザクション・コードが何であるべきかを決定します。出口ルーチンがトランザクション・コードを変更しない場合、宛先は元のトランザクション・コードで指定された場所のままです。DFSMSCEO は、メッセージごとに使用するセキュリティ (ある場合) を決定できます。

## プログラム経路指定

TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン (DFSMSCEO) のプログラム経路指定ポイントを使用して、アプリケーション・プログラムが CHNG または ISRT 呼び出しを発行するときにメッセージの経路指定を制御することができます。これらのエントリー・ポイントによって、メッセージまたは要求の宛先を変更してメッセージをリジェクトすることができます。また、DFSMSCEO 出口ルーチンは、メッセージが現在の IMS システムでローカルに処理されるように要求することも、メッセージを異なるリモート IMS システムに転送することもできます。

この出口ルーチンを使用して、リモート LTERM およびトランザクションに固有の名前を定義しないようにできます。この出口ルーチンでは、MSC ネットワークの各端末に同一の名前を付けることができます。この出口ルーチンを使用して、アプリケーション・プログラムの DL/I CHNG または ISRT 呼び出しを実行し、宛先をローカルからリモートへ変更することができます。

### 関連資料

[「TM および MSC メッセージ経路指定および制御」ユーザー出口ルーチン \(DFSMSCEO\) \(出口ルーチン\) MSGQUEUE マクロ \(システム定義\)](#)

## DFSMSCEO 出口ルーチンの IMSplex 類似性経路指定オプション

共用キューを使用する IMSplex では、TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン (DFSMSCEO) の IMSplex 類似性経路指定オプションによって、トランザクションが類似性オプションを指定して開始される IMS システムとトランザクション・インスタンスとの間に類似性を確立することができます。

DFSMSCEO 出口ルーチンの IMSplex 類似性経路指定オプションを使用する IMS システムにおいて、MSC が使用可能である必要があります。

DFSMSCEO 出口ルーチンが類似性をトランザクションに割り当て、そのトランザクションが共用キューに挿入されると、DFSMSCEO 出口ルーチンによって指定された IMS システムのみがその通知を受け取ります。

DFSMSCEO 出口ルーチンの IMSplex 類似性経路指定オプションによって、IMSplex 内のバックエンド IMS システムが APPC 同期トランザクションと OTMA 送信後コミット (CM1) トランザクションの両方を処理することができるようになります。IMSplex 類似性経路指定オプションは、このオプションが指定されなければ APPC トランザクションと OTMA CM1 トランザクションが IMSplex 内のバックエンド IMS システムで処理ができなくなるという制約事項を無効にします。類似性オプションが呼び出されると、APPC トランザクションと OTMA トランザクションは、クライアントからトランザクション・メッセージを切断するモードで実行されます。これは MSC がメッセージを MSC ネットワーク内のリモート IMS システムに経路指定する際のモードと同様です。

DFSMSCEO 出口ルーチンを呼び出して、以下の出口点に類似性を確立することができます。

### 端末経路指定 (TR)

DFSMSCEO 出口ルーチンは制御を受け取り、IMSplex 内の IMS システムにより端末から受信されるメッセージに対して類似性を設定できます。

### リンク受信 (LR)

DFSMSCEO 出口ルーチンは制御を受け取り、IMSplex 内の IMS システムにより MSC リンクから受信されるメッセージに対して類似性を設定できます。



## プログラム経路指定 (PR)

DFSMSCEO 出口ルーチンは制御を受け取り、IMSplex 内の IMS システムで実行中のアプリケーション・プログラムが、メッセージを挿入するために CHNG 呼び出しまたは ISRT 呼び出しを発行するときそのメッセージに対して類似性を設定できます。

DFSMSCEO 出口ルーチンは、メッセージ接頭語の共用キュー名 (SQNAME) に IMS ID を、または XRF 複合システムでは、宛先 IMS システムの RSENAME を追加することにより、トランザクション・インスタンスに対する類似性を確立します。

トランザクション・インスタンスは、それが類似性を持つ IMS システムにより処理できるようにするにはその前に、トランザクション・タイプに IMS システムに類似性があるものとして登録しておく必要があります。

トランザクションを IMS システムで類似性オプションを指定して開始するには、そのトランザクションを処理する IMS システムで以下のコマンドのいずれかを指定します。

- タイプ 2 コマンド **UPDATE TRAN NAME**(*trancode*) START(SCHD) OPTION(AFFIN)
- タイプ 1 コマンド **/START TRAN** *trancode* AFFINITY

IMS は、ウォーム・スタートおよび緊急時再始動では、IMS システムでトランザクション・タイプの類似性状況を維持しますが、コールド・スタートまたはトランザクション定義のリソース定義データ・セットへのエクスポートおよびリソース定義データ・セットからのインポートでは、類似性状況を維持しません。

IMS システムに対して類似性を持つ共用キューにあるトランザクションを表示するには、以下のコマンドのいずれかを適切なキーワードを指定して発行することができます。

- タイプ 2 コマンド **QUERY TRAN QCNT**(GT,0) SHOW(AFFIN)
- タイプ 1 コマンド **/DIS TRAN ALL QCNT**

IMS システムが共用キューから類似性を持つトランザクションを取り出すと、そのトランザクションの類似性状況は LclStat の下に AFFIN として表示されます。IMS システムでどのトランザクションにローカル類似性状況があるのかを確認するには、以下のタイプ 2 コマンドを発行してください。

- **QUERY TRAN NAME**(*trancode* | \*) SHOW(STATUS)

例えば、**QUERY TRAN NAME**(APOL11 APOL12) SHOW(STATUS) と発行すると、コマンド出力は以下のようになります。

Trancode	MbrName	CC	LclStat
APOL11	IMS1	0	
APOL12	IMS1	0	AFFIN

## 関連資料

[「TM および MSC メッセージ経路指定および制御」 ユーザー出口ルーチン \(DFSMSCEO\) \(出口ルーチン\)](#)

## MSC での IMSRSC リポジトリの使用

複数システム結合機能 (MSC) 対応の IMS システムの場合、IMS システム間で IMSRSC リポジトリを定義することができます。

## IMSRSC リポジトリ定義および MSC

MSC リンクが IMSplex 内にある場合、すべての IMS システムがその IMSplex 用に定義された同じリポジトリを共有します。MSC リンクが IMSplex 間にあり、IMSplex が同じ z/OS システム内にある場合、各 IMSplex は独自のリポジトリを持つことも、同じリポジトリを共有することもできます。これらの異なる IMSplex 内のリポジトリは、同一のアクティブ・リポジトリ・サーバー (RS) アドレス・スペースで管理することも、別のアクティブ RS アドレス・スペースで管理することもできます。

以下の MSC リソースをリポジトリ内で保守することができます。

- リモート・トランザクションおよびトランザクション記述子
- 物理リンクと論理リンク
- リンク・パス

- リモート論理端末 (LTERM)

リモート・トランザクションまたはトランザクション記述子が、それに関連付けられたプログラム名を持っている場合、そのプログラム・リソースはリポジトリ内に存在する必要はありません。ローカルとして定義されているトランザクションまたはトランザクション記述子に関連付けられているプログラム・リソースは、リポジトリ内に存在する必要があります。

**推奨事項:** MSC リンクが同じ z/OS シスプレックス内の IMSplex 間にある場合は、1つの RS と、IMSplex ごとに個別のリポジトリを定義してください。個別のリポジトリの方が管理が容易だからです。

MSC リンクが IMSplex 間にあり、IMSplex がそれぞれ異なる z/OS シスプレックス内にある場合は、各 IMSplex が独自のリポジトリと RS アドレス・スペースを持つ必要があります。これらの異なる IMSplex のリポジトリは、それぞれ異なる RS アドレス・スペースで管理する必要があります。

IMS システム間で異なる属性 (各 MSC 対応システムの SIDR 値や SIDL 値など) を持つリソース定義の場合、リポジトリに保管されたリソース定義は、汎用セクション (共通の属性用) と IMS 固有のセクション (IMS システムごとに異なる属性用) で構成されます。

#### 関連概念

[IMSRSC リポジトリの概要 \(システム定義\)](#)

## リモート・トランザクションおよび記述子の SIDR 値と SIDL 値の格納方法

IMS システム間で異なる属性 (各 MSC 対応システムの SIDR 値や SIDL 値など) を持つリソース定義の場合、リポジトリに保管されたリソース定義は、汎用セクション (共通の属性用) と IMS 固有のセクション (IMS システムごとに異なる属性用) で構成されます。

リモート・トランザクションおよびトランザクション記述子の場合、SIDR 値と SIDL 値が同じではありません。各 IMS システムの SIDR 値および SIDL 値は、リポジトリ内の IMS 固有のセクションに維持されます。各 IMS が、SIDR 値および SIDL 値のための独自の固有セクションを持っています。ローカル・トランザクションおよびトランザクション記述子の場合、リポジトリの汎用セクション内の SIDR 値と SIDL 値は 0 に設定されます。AUTOIMPORT 処理時または **IMPORT** コマンドの処理時に、保管されたリソース定義がリポジトリからインポートされる場合、SIDR 値と SIDL 値は、ランタイム・リソース定義が作成された IMS システムの最低のローカル SID 値に設定されます。

リモート・トランザクションおよびトランザクション記述子のリポジトリ内の SIDR 値および SIDL 値は、SET(IMSID(\*)) キーワードを指定した **EXPORT** コマンドが発行された場合、IMS 固有のセクション内に残り、更新されたり、縮小して汎用セクションに入れられたりすることはありません。リモート・トランザクションおよびトランザクション記述子の SIDR 値および SIDL 値を変更するには、SET(IMSID) キーワードで特定の IMS ID を指定するか、デフォルトの SET(IMSID()) キーワードを指定して **EXPORT** コマンドを発行し、定義をエクスポートする IMS にコマンドを経路指定する必要があります。

#### 関連概念

[IMSRSC リポジトリの概要 \(システム定義\)](#)

## IMSRSC リポジトリ内の MSC リソースの保守

IMSRSC リポジトリ内で MSC リソース定義を保守することにより、IMSplex のすべてのメンバーに対応する単一の中央保管場所に定義を保管します。また、IMSRSC リポジトリ内で MSC リソース定義を保守することにより、IMS コールド・スタートの後にも以前の定義を保管できます。

IMSRSC リポジトリを使用して動的に定義された MSC リソースを保管する場合は、MSC リソースに対するコマンドを発行する自動化手順と運用手順で、リンク番号を指定するタイプ 1 コマンドの代わりに、リンク・ネームを指定するタイプ 2 コマンドが使用されていることを確認してください。例えば、**/RSTART LINK 10** コマンドを使用してリンクを開始する代わりに、UPDATE MSLINK NAME(*logical linkname*) START (COMM) コマンドを使用します。ステージ 1 のシステム生成時に、IMS システムはリンクが生成された順序で論理リンクに番号を割り当てます。ただし、リンクの番号は IMSRSC リポジトリに保管されません。論理リンクがリンク番号を使用して参照されていて、リンクが IMSRSC リポジトリから自動的にインポートされる場合は、次の IMS コールド・スタート時にリンクの番号が変更される可能性があります。



IMSRSC リポジトリ内に MSC リソースを作成して保守する前に、以下のすべての前提条件が満たされていることを確認してください。

- IMSRSC リポジトリが定義されます。
- MSC リソースの動的リソース定義が使用可能である。
- IMSRSC リポジトリが MSC リソース用に使用可能である。
- MODBLKS=DYN は、以下のいずれかまたは両方の場所で指定されます。
  - DFSDFxxx PROCLIB メンバーの COMMON\_SERVICE\_LAYER セクション
  - IMS PROCLIB データ・セットの DFSCGxxx メンバー
- DFSDFxxx PROCLIB メンバーの DYNAMIC\_RESOURCES セクションで、AUTOEXPORT=AUTO または AUTOEXPORT=REPO が指定されている。
- DFSDFxxx PROCLIB メンバーの DYNAMIC\_RESOURCES セクションで、AUTOIMPORT=AUTO または AUTOIMPORT=REPO が指定されている。

## IMSRSC リポジトリにおける MSC リソースの作成

IMSRSC リポジトリ内に MSC リソースを作成するには、作成したい MSC リソースのタイプに対して **CREATE** コマンドを発行します。MSC リンクを使用可能にするには、次の順序で MSC リソースを作成する必要があります。

1. CREATE MSPLINK
2. CREATE MSLINK
3. CREATE MSNAME
4. CREATE LTERM

リソース定義をリポジトリにエクスポートする各 IMSplex 内の IMS ごとに、**CREATE** コマンドを発行します。OM API で ROUTE キーワードを使用して、コマンドを経路指定する先の IMS を指定できます。

作成した MSC リソース定義は、次の IMS チェックポイントの終わりか IMS シャットダウン・チェックポイントの前に、IMSRSC リポジトリに自動的にエクスポートされます。

IMSRSC リポジトリへの自動エクスポート処理の終わりに、X'22' マップ・バイト X'51' 自動エクスポート完了ログ・レコードが書き込まれます。

## IMSRSC リポジトリにおける MSC リソースの更新

IMSRSC リポジトリ内の MSC リソースを更新するには、更新するリソース定義のタイプに対して適切な **UPDATE** コマンドを使用します。

リソース定義をリポジトリにエクスポートする各 IMSplex 内の IMS ごとに、**UPDATE** コマンドを発行する必要があります。OM API で ROUTE キーワードを使用して、コマンドを経路指定する先の IMS を指定できます。

自動エクスポート処理は、各 IMS 正常チェックポイントの終わり、および IMS シャットダウン・チェックポイントの前に実行されます。IMS は、前回の自動エクスポート以降に MSC リソースのリソース定義に変更があった場合、それらの変更を自動的にエクスポートします。

## IMSRSC リポジトリからの MSC リソースの削除

IMSRSC リポジトリからリソース定義および記述子定義を削除するには、**DELETE DEFN** コマンドを発行します。**DELETE DEFN** コマンドを発行すると、コールド・スタート後も IMS™ 内で定義が削除されたままになります。また、コールド・スタートの処理時に定義がリポジトリからインポートされません。

IMSRSC リポジトリからの MSC リソース定義の削除は、以下の順序で行います。

1. リモート・トランザクション定義
2. リモート論理端末 (LTERM) 定義
3. MSC 論理リンク・パス (MSNAME)

#### 4. MSC 論理リンク (MSLINK)

#### 5. MSC 物理リンク (MSPLINK)

**重要:** あるタイプの MSC リソースを削除し、そのリソースを別のタイプとして再作成する場合は、最初にローカル IMS システムからリソースを削除し、次に、IMSRSC リポジトリからリソースを削除します。元のタイプの MSC リソースを最初にローカル側で削除しておかずに IMSRSC リポジトリから削除すると、IMS システムは元のリソースを別のリソース・タイプとして再作成できない場合があります。

**DELETE** コマンドを使用してランタイム・リソース定義を削除してから IMS をコールド・スタートしても、リソース定義が、最新の定義を含んでいる IMSRSC リポジトリからではなく、IMS.SDFSRESL データ・セットの元の DFSCLL3x メンバーから自動的にインポートされる場合は、削除したリソース定義が再表示されます。

### IMSRSC リポジトリへの MSC リソースのエクスポート

IMS は、次の IMS 正常チェックポイントの終わりか IMS シャットダウン・チェックポイントの前に、前回の自動エクスポート以降に新規に作成されるか更新された MSC リソース定義を IMSRSC リポジトリに自動的にエクスポートします。IMS チェックポイントは、**/CHECKPOINT** コマンドを発行することによって開始するか、または IMS システムによって自動的に開始することができます。

**/CHECKPOINT** コマンドを使用して IMS チェックポイントを開始する場合は、MSC リソースが定義されている各 IMSplex 内の各 IMS にコマンドを経路指定する必要があります。

### IMSRSC リポジトリからの MSC リソースのインポート

MSC リソース定義が IMSRSC リポジトリ内に存在する場合は、次の IMS コールド・スタートのときに、それらの定義がリポジトリからインポートされます。

チャンネル間 (CTC) リンクを使用する場合は、MSC リソースを IMSRSC リポジトリからインポートする前に、IMS JCL から CTC リンクの DD 定義を削除することを検討してください。これにより、IMSRSC リポジトリからインポートされる CTC リンクに対して、定義された CTC アドレスが使用されるようになります。

チャンネル間リンクが作成されてエクスポートされた z/OS システムとは異なる z/OS 上で IMS システムがコールドスタートされた場合、新規 z/OS システム上にチャンネル間アドレスが定義されていないと、CTC リンクのオープンが失敗し、DFS2168I CONNECTION ESTABLISHED ON LINK x メッセージは発行されません。ただし、自動インポートは続行されます。UPDATE MSPLINK SET(ADDR(addr)) コマンドを使用して、MSC 物理リンクのアドレスを、z/OS システムに有効なアドレスに変更することができます。

### DFSCLL3x メンバーおよび DFSCLR0x メンバーの除去、および IMS.SDFSRESL データ・セットの DFSCLC0x メンバーの更新

動的に定義された MSC リソースを保管するために IMSRSC リポジトリを使用する場合、IMS.SDFSRESL データ・セットの DFSCLL3x メンバーおよび DFSCLR0x メンバーは不要になりました。DFSCLC0x メンバーには非 MSC リソースが含まれることがあるため、DFSCLC0x は引き続き必要になる場合があります。MSC リソースに関する DRD 環境のセットアップを問題なく完了し、DRD 環境が正常に実行され、MSC リソースが IMSRSC リポジトリにエクスポートされた後、DFSCLL3x および DFSCLR0x メンバーを削除できます。DFSCLC0x メンバーについては、メンバーを更新して、MSC 論理リンク・パス定義を削除してください。自動インポートが有効になっている場合は、IMS のコールド・スタート時に、最新のデータを含むリポジトリから MSC リソース定義がインポートされます。ただし、DFSCLL3x メンバー、DFSCLR0x メンバー、および DFSCLC0x メンバーをリポジトリの代わりに MSC リソース定義のソースとして引き続き使用することもできます。DFSCLL3x メンバー、DFSCLR0x メンバー、および DFSCLC0x メンバーを引き続き使用する場合は、DRD コマンドを使用して動的に行った変更と、システム定義マクロの同期を保ってください。

### IMS.SDFSRESL データ・セットの DFSCLL3x メンバーの同期化

動的に定義された MSC リソースに対しては IMSRSC リポジトリの使用に移行する一方で、IMS.SDFSRESL データ・セットの DFSCLL3x メンバー、DFSCLR0x メンバー、および DFSCLC0x メンバー

を引き続き使用する場合は、リポジトリ内のリソース定義とメンバー内のリソース定義の同期を保ってください。この同期によって、MSC リソースに対して DRD を無効にしてシステム生成プロセスの使用にフォールバックする必要がある場合に、DFSCLL3x メンバー、DFSCLR0x メンバー、および DFSCLC0x メンバーを使用可能な状態に維持できます。DFSCLL3x メンバー、DFSCLR0x メンバー、および DFSCLC0x メンバーとオンライン定義の同期を保つには、タイプ 2 コマンドを使用して動的に実施する変更に合わせて、静的マクロ定義を更新します。変更を動的に実施するときに、MSC システム定義を実行して、DFSCLL3x メンバー、DFSCLR0x メンバー、および DFSCLC0x メンバーのリソースの追加、変更、または削除を行います。

### 関連概念

[ランタイム・リソース定義および記述子定義の削除 \(システム定義\)](#)

### 関連タスク

[MSC リソースに対する IMSRSC リポジトリの使用可能化 \(コミュニケーションおよびコネクション\)](#)

[IMSRSC リポジトリの定義 \(システム定義\)](#)

[IMSRSC リポジトリでの MSC リソース定義の作成 \(システム定義\)](#)

[MSC リソースの動的定義の使用可能化 \(システム定義\)](#)

[IMSRSC リポジトリへの MSC リソース定義のエクスポート \(システム定義\)](#)

[IMSRSC リポジトリでの MSC リソース定義の更新 \(システム定義\)](#)

### 関連資料

[CREATE LTERM コマンド \(コマンド\)](#)

[CREATE MSLINK コマンド \(コマンド\)](#)

[CREATE MSNAME コマンド \(コマンド\)](#)

[CREATE MSPLINK コマンド \(コマンド\)](#)

[/CHECKPOINT コマンド \(コマンド\)](#)

[DELETE DEFN コマンド \(コマンド\)](#)

[UPDATE MSLINK コマンド \(コマンド\)](#)

[UPDATE MSNAME コマンド \(コマンド\)](#)

[UPDATE MSPLINK コマンド \(コマンド\)](#)

[DFSDFxxx メンバーの MSC セクション \(システム定義\)](#)

[DFSDFxxx メンバーの DYNAMIC\\_RESOURCES セクション \(システム定義\)](#)

## リポジトリ内の MSC リソースの作成または更新

IMSplex 用の IMSRSC リポジトリ内で、MSC リソースを作成または更新することができます。

### 始める前に

IMSRSC リポジトリで MSC リソース定義を作成または更新する前に、以下の前提条件が満たされていることを確認してください。

- IMSRSC リポジトリが定義されます。
- MSC リソースの動的リソース定義が使用可能である。
- IMSRSC リポジトリが MSC リソース用に使用可能である。
- IMS PROCLIB メンバーの DFSDFxxx メンバーの DYNAMIC\_RESOURCES セクションで、AUTOEXPORT=AUTO または REPO が指定されている。
- IMS PROCLIB データ・セットの DFSDFxxx PROCLIB メンバーまたは DFSCGxxx メンバーの COMMON\_SERVICE\_LAYER セクションで、MODBLKS=DYN が指定されている。

### このタスクについて

IMSplex 用のリポジトリ内で MSC リソースを作成または更新するには、以下のようになります。

## 手順

1. 適切な CREATE コマンドまたは UPDATE コマンドを使用して、IMS システムで MSC リソースをローカルに作成または更新します。  
次の IMS チェックポイント時に、最後の自動エクスポート以降に作成または更新された MSC リソース定義が、IMSRSC リポジトリにエクスポートされます。IMS チェックポイントは、/CHECKPOINT コマンドを発行することによって開始するか、または IMS システムによって自動的に開始することができます。
2. リモート・トランザクションおよびリモート・トランザクション記述子の場合、**EXPORT DEFN TARGET(REPO)** コマンド (デフォルトは SET(IMSID()) キーワード) を発行して、リソース定義がエクスポートされる IMS システムにコマンドを送信することができます。
3. これらのステップを、MSC リソース定義が定義されているか定義される予定の各 IMS システムで繰り返します。リポジトリは、各 IMS の MSC リソースの属性を維持します。

### 関連資料

[EXPORT コマンド \(コマンド\)](#)

[DFSDFxxx メンバーの DYNAMIC\\_RESOURCES セクション \(システム定義\)](#)

## リポジトリを使用したリモートからローカルへのトランザクションの更新

IMSRSC リポジトリを使用して、IMSpdex 内で、リモートからローカルにトランザクションまたはトランザクション記述子を更新することができます。

### このタスクについて

リポジトリを使用して、IMSpdex 内のトランザクションおよびトランザクション記述子をリモートからローカルに更新するには、以下の手順を実行します。

## 手順

1. REMOTE キーワード、SIDR キーワード、および SIDL キーワードを指定して、**UPDATE TRAN** コマンドまたは **UPDATE TRANDESC** コマンドを発行します。コマンドでは、REMOTE(N) を指定し、SIDR 値と SIDL 値を同じローカル・システム ID (SYSID) に設定します。
2. **EXPORT DEFN TARGET(REPO)** コマンド (デフォルトでは、SET(IMSID()) キーワードを使用) を発行して、リソース定義をエクスポートする IMS システムにコマンドを経路指定します。あるいは、AUTOEXPORT=AUTO または REPO で IMSRSC リポジトリへの自動エクスポートが使用可能に設定されている場合は、AUTOEXPORT\_IMSID が THIS\_IMS に設定されていることを確認します。
3. トランザクションまたはトランザクション記述子をローカルとして定義する各 IMS システムで、上記のステップを繰り返します。リポジトリは、リポジトリ内の各 IMS のリモート属性を削除します。

### 関連資料

[EXPORT コマンド \(コマンド\)](#)

[UPDATE TRAN コマンド \(コマンド\)](#)

[UPDATE TRANDESC コマンド \(コマンド\)](#)

[DFSDFxxx メンバーの DYNAMIC\\_RESOURCES セクション \(システム定義\)](#)

## リポジトリを使用したローカルからリモートへのトランザクションの更新

IMSRSC リポジトリを使用して、IMSpdex 内で、ローカルからリモートにトランザクションまたはトランザクション記述子を更新することができます。

### このタスクについて

リポジトリを使用して、IMSpdex 内のトランザクションおよびトランザクション記述子をローカルからリモートに更新するには、以下の手順を実行します。

## 手順

1. REMOTE キーワードおよび、SIDR キーワードと SIDL キーワードあるいは MSNAME キーワードのいずれかを指定して、**UPDATE TRAN** コマンドまたは **UPDATE TRANDESC** コマンドを発行します。コマンドでは、REMOTE(Y) を指定し、SIDR 値をリモート SYSID に、また SIDL 値をローカル SYSID に設定するか、あるいは MSNAME キーワードを使用して論理リンク・パスの名前を指定します。
2. **EXPORT DEFN TARGET(REPO)** コマンド (デフォルトでは、SET(IMSID()) キーワードを使用) を発行して、リソース定義をエクスポートする IMS システムにコマンドを経路指定します。あるいは、**AUTOEXPORT=AUTO** または **REPO** で IMSRSC リポジトリへの自動エクスポートが使用可能に設定されている場合は、**AUTOEXPORT\_IMSID** が **THIS\_IMS** に設定されていることを確認します。どちらのパラメーター (**AUTOEXPORT** および **AUTOEXPORT\_IMSID**) も、DFSDFxxx メンバーの **DYNAMIC\_RESOURCES** セクションで見つけることができます。
3. トランザクションまたはトランザクション記述子をリモートとして定義する各 IMS システムで、上記のステップを繰り返します。リポジトリは、各 IMS のリモート属性を維持します。

## 関連資料

[EXPORT コマンド \(コマンド\)](#)

[UPDATE TRAN コマンド \(コマンド\)](#)

[UPDATE TRANDESC コマンド \(コマンド\)](#)

[DFSDFxxx メンバーの DYNAMIC\\_RESOURCES セクション \(システム定義\)](#)



---

## 第 9 部 ODBA および DRA 接続

Open Database Access (ODBA) およびデータベース・リソース・アダプター (DRA) インターフェースは、IMS と同じ z/OS イメージで実行されているアプリケーションから、IMS データベースへの直接アクセスを提供します。

### このタスクについて

DRA、または ODBA および DRA を使用するアプリケーション・プログラムには、以下があります。

- CICS Transaction Server for z/OS
- Db2 for z/OS ストアード・プロシージャなどの、ODBA z/OS アプリケーション





# 第 40 章 CICS を使用した IMS データベースへのアクセス

CICS アプリケーション・プログラムは、DBCTL を介して DL/I データベースにアクセスできます。これは、DBCTL が、CICS-DBCTL インターフェースを使用して CICS DL/I 要求を満たすことを意味します。

## このタスクについて

CICS/ESA および CICS Transaction Server for z/OS は、IMS DBCTL をサポートしています。

### CICS とのデータ共有のインストール

CICS ユーザーは、オプションで IMS データ共有機能を使用できます。ブロック・レベルのデータ共有には Internal Resource Lock Manager (IRLM) が必須ですが、データベース・レベルの共有には必須ではありません。

### IMS システム間連絡とともに使用する CICS のインストール

CICS をシステム間連絡ノードとして定義する方法と、システム間連絡ネットワーク内で CICS テーブルが IMS との互換性を持つように定義する方法は、453 ページの『第 7 部 システム間連絡 (ISC)』に記載されています。

### 関連概念

[IMS 環境でのデータ共有 \(システム管理\)](#)

## PSB に関するコーディング上の考慮事項

このトピックでは、CICS を使用する場合に、PSB をコーディングする上でのいくつかのガイドラインを示します。

### オンライン・トランザクション用の PSB

- PSB は、DL/I データベースにアクセスするオンライン・プログラムごとに必要です。
- PSBGEN マクロの PSBNAME= キーワードで指定された PSB の名前は、PDIR の項目のいずれかと正確に一致している必要があります。
- PSB の名前も、オンライン・トランザクションが発行するスケジューリング呼び出しで指定された名前と同じでなければなりません。オンライン・トランザクションのスケジューリング呼び出しで名前が指定されていない場合、PSB の名前は、CICS プログラム管理テーブル (PCT) の中でトランザクションに関連付けられたプログラムの名前と正確に一致している必要があります。例えば、仮に、トランザクション X がプログラム Y に関連付けられているとします。プログラム Y が、PSB 名を指定せずにスケジューリング呼び出しを発行するプログラム Z にリンクしていたり、制御をプログラム Z に転送したりすると、PSB のデフォルト名は Y になります。これは、PCT でトランザクション X に関連付けられているのがプログラム Y であるためです。
- PSBGEN ステートメントの CMPAT=YES オプションは省略できます。
- CICS/ESA で使用できる個々の PSB には、最大サイズは決まっていません。ただし、DBCTL の PSB プール・サイズにより制限が設定されています。

### 追加処理インテント・オプション (PROCOPT)

PCB または SENSEG ステートメントの PROCOPT= キーワードで指定できる処理インテントには、さらに 2 つのタイプがあります。この 2 つの追加オプションは、O と E です。

PROCOPT=O は、「保全性なしの状態で読み取る」と指定します。データベースに対する呼び出しではリソース・ロック管理による動的エンキューは行われません。インテント・オプション G を使用すると、GO、GOP、GON、または GOT を指定できます。このオプションは、PCB ステートメントに対してのみ有効です。詳しくは、[注意の 1 番](#)をお読みください。

PROCOPT=E では、このデータベースまたはセグメントを、オンライン・トランザクションが強制的に排他使用します。このデータベース/セグメントを参照する PSB をスケジューリングするその他のアプリケーション・プログラムは、それ自体のスケジューリング・プロセスの間は待機しています。リソース・ロック管理による動的エンキューは行われませんが、データベース更新の動的ロギングは行われます。PROCOPT E は、G、I、D、R、および A とともに指定できます。



#### 重要:

1. PROCOPT=O オプションは、コミットされていないデータの読み取りと使用における保全性に影響します。このオプションを指定すると、IMS は、戻されるセグメントの所有権を検査しません。このことは、読み取り専用ユーザーが、別のユーザーによって更新されたセグメントへのアクセス権を持つ可能性があるということを意味します。更新しているユーザーが異常終了してバックアウトされた場合、読み取り専用ユーザーに対しては、データベース内にもはや同じ形では存在していないセグメントが表示されていたこととなります。このため、このオプションを指定する場合は、どのデータベース内でも、レコードを更新するための基礎として読み取られるデータは使用しないでください。
2. このプログラムがポインターをたどっているときに別のプログラムがそのポインターを更新すると、PROCOPT=GO で異常終了が発生することがあります。ポインターは、挿入、削除、およびバックアウトの各操作の間に更新されます。
3. PCB ステートメントで PROCOPT=O となっている場合、SENSESEG ステートメントは I、R、D、H、または A の PROCOPT を指定してはなりません。
4. O または E のオプションを使用する場合は、関連する機能コードの直後にコーディングする必要があります (例えば OG ではなく GO)。

#### 関連資料

[プログラム仕様ブロック \(PSB\) 生成ユーティリティー \(システム・ユーティリティー\)](#)

## 順次バッファリングの使用

---

順次バッファリングは、CICS とともに使用できます。

### このタスクについて

CICS で順次バッファリングを使用するには以下の 2 つを行う必要があります。

- //DFSVSMxx ファイルに SBONLINE 制御ステートメントを入れる。SBONLINE により、順次バッファリングが使用できるようになります。
- 順次バッファリングを使用するプログラムを指定する。これは、PSBGEN のときに PCB マクロの SB= キーワードをコーディングすることによって行えます。

#### 関連資料

[全機能または高速機能データベースの PCB ステートメント \(システム・ユーティリティー\)](#)

[オンライン・システムでの順次バッファリングの指定 \(システム定義\)](#)

## DL/I に接続された CICS

---

CICS は、IMS を使用して、DL/I データベースをサポートすることができます。

DL/I サポート を CICS で使用する方法には、以下の 2 つがあります。

- CICS リモート DL/I サポート経由。機能シップとも呼ばれます。
- DBCTL 経由。

CICS/ESA では、リモート DL/I サポートおよび DBCTL サポートは、あらかじめ生成される CICS に含まれています。

## IMS DBCTL システムへの CICS CCTL 接続の構成

IMS DB によって管理されるデータベースに CICS を接続するには、CICS リソースを定義し、IMS DBCTL システムを初期設定する必要があります。

### 手順

#### • CICS リソースの定義

DBCTL を排他的に使用する場合は、以下のいずれかの方法を使用して、IMS で PSB および DMB を定義します。

- APPLCTN マクロおよび DATABASE マクロを使用します。
- 動的リソース定義 (DRD) が使用可能になっているシステムで、コマンド CREATE DB および CREATE PGM を使用します。
- 動的リソース定義 (DRD) が使用可能になっているシステムで、プログラム作成ユーザー出口ルーチン (PGMCREAT) を使用して、BMP および JBP 従属領域内でスケジュールされたプログラムに関連付ける PSB を定義します。

リモート CICS システム (この中ではデータベース・マネージャーとして DBCTL またはリモート DL/I が使用可能 (機能シップ)) への要求を機能シップする場合、生成する必要があるのは PDIR のみです。

CICS は、DL/I 要求を、指定された PSB に従ってリモート DL/I または DBCTL に経路指定します。PSB が CICS PDIR に出現せず、CICS が DBCTL に接続されている場合、CICS はこの要求を DBCTL に経路指定します。

**関連資料:** PDIR および DDIR の定義について詳しくは、「*CICS Transaction Server for z/OS CICS システム定義ガイド*」および「*CICS Transaction Server for z/OS CICS リソース定義ガイド*」を参照してください。

#### • IMS DBCTL の初期設定

DBCTL サブシステムを初期設定するには、プロシージャー・ライブラリー・メンバー DBRC を使用します。また、DBRC および DL/I 用のプロシージャーも生成されます。これらは、DBRC と DL/I のアドレス・スペースを初期設定するために使用されます。DBRC および DL/I のプロシージャーは、DBCTL アドレス・スペースの初期設定中に DBCTL によって自動的に開始されます。これら 3 つのプロシージャーはすべて、EXEC ステートメントで 定位置パラメーターを使用します。

```
PARM='region type, parm1, parm2, parm3,...'
```

指定される領域タイプは次のとおりです。

```
PARM='DBC' for DBCTL  
PARM='DRC' for DBRC  
PARM='DLS' for DLISAS
```

3 つのアドレス・スペースがすべて正常に初期設定されると、DBCTL は以下のメッセージを発行します。

```
DFS989I DBCTL READY
```

「READY」メッセージを受信すると、IMS コンソール・オペレーターは /START コマンドを入力します。このコマンドのオプションは、以下のとおりです。

- /NRESTART CHECKPOINT 0。直前にシャットダウンしなかった場合のコールド・スタートに使用します。
- /NRESTART。ウォーム・スタートに使用します。
- /ERESTART。障害発生後の緊急時再始動に使用します。

**関連資料:** 詳しくは、「*CICS Transaction Server for z/OS IMS Database Control Guide*」を参照してください。

## CICS のタスク

データベース・リソース・アダプター (DRA) は、DBCTL とトランザクション管理サブシステムの間インターフェースです。CICS STEPLIB データ・セットの中に、DRA 始動パラメーター・テーブル (DFSPZPxx) および DRA 始動ルーター・プログラム (DFSPRRC0) を入れる必要があります。

### このタスクについて

このトピックでは、DBCTL および CICS の使用に関連する CICS のタスクについて説明します。概略のタスクには、以下があります。

### 手順

#### 1. DRA リソースの定義

**例:**

```
//STEPLIB DD DSN=CICSTSxx.CICS.SDFHAUTH,DISP=SHR
// DD DSN=IMS.SDFSRESL,DISP=SHR
```

**関連資料:** DRA 始動テーブルについて詳しくは、「IMS V15 アプリケーション・プログラミング API」または「IMS V15 出口ルーチン」の"CCTL 出口ルーチン"を参照してください。

#### 2. DBCTL への CICS の接続

CICS の始動後に、CICS オペレーターは、CDBC トランザクションを介して CONNECT コマンドを DBCTL に発行することができます。このトランザクションは、DBCTL からの切断にも使用されます。CDBI トランザクションは、接続の状況を提供します。

**関連資料:** これらのトランザクションについて詳しくは、「CICS Transaction Server for z/OS IMS Database Control Guide」を参照してください。

## 第 41 章 ODBA インターフェースを介した IMS データベースへのアクセス

Open Database Access (ODBA) は、z/OS の リカバリー可能な、リソースにより管理された z/OS アドレス・スペースがある場合、それが DL/I データベース呼び出しを IMS DB サブシステムに対して行うための呼び出し可能インターフェースを提供します。

### このタスクについて

このインターフェースは、z/OS アプリケーション・プログラム (以後 ODBA アプリケーションと呼びます) による 全機能 DL/I データベースおよび高速処理データベース (DEDB) へのアクセスを可能にします。ODBA アプリケーションと IMS は、同一の z/OS イメージ上に共存している必要があります。

ODBA インターフェースを使用するアプリケーション・プログラムは、APPLCTN マクロを使用して、あるいは、動的リソース定義が使用可能な場合は **CREATE PGM** コマンドまたはプログラム作成ユーザー出口ルーチン (PGMCREAT) を使用して、ODBA アプリケーションが必要とする PSB 名を定義できます。

ODBA インターフェースを使用すると、IMS DB および ODBA アプリケーション・プログラムを独立して開発し、インストールし、保守することができます。この独立性により、z/OS リソース・リカバリー・サービス (RRS) を使用して障害を分離でき、リソースをリカバリーすることができます。

CSL Open Database Manager (ODBM) を使用するように ODBA を構成することにより、障害をさらに分離できます。ODBM では、DL/I 処理中に ODBA アプリケーションが予想外に終了した場合に、IMS 異常終了 0113 の発生を回避できます。他の IMS データベース・リソース・アダプター (DRA) インターフェース始動パラメーターと共に DFSxxxx0 メンバーに ODBMNAME および IMSPLEX パラメーターを指定することで、ODBM を使用するように ODBA を構成できます。ODBA インターフェースは、DRA インターフェースを使用して、IMS DB と通信します。

ODBA インターフェースは z/OS アドレス・スペースに常駐し、IMS によりアプリケーション領域 (以後 z/OS アプリケーション領域と呼びます) として認識されます。

### 関連概念

28 ページの『RRS および分散同期点/保護会話』

SYNCLVL の設定が NONE、CONFIRM、または SYNCPOINT のいずれであるかに関係なく、RRS=Y の場合は、z/OS リソース・リカバリー・サービスが同期点マネージャーであり、複数の保護リソースの更新およびリカバリーを調整します。RRS は、RRS に登録されたリソース・マネージャー (IMS など) と調整しながら、保護リソースをいつ、どのようにコミットするかを制御します。

## ODBA DRA 始動テーブルの作成

DFSFRP マクロを使用して、ODBA DRA 始動テーブルを作成します。

### このタスクについて

始動テーブルは DSNAME を使用して、その他の ODBA インターフェース・ルーチンを含むデータ・セットを動的に割り振ります。同一の z/OS アプリケーション領域から IMS に対して複数の接続ができるように、DDNAME が生成されます。DFSFRP マクロにこの DDNAME を指定しても無視されます。

デフォルトの DSNAME は IMS.SDFSRESL です。これは、IMS 定義プロセスによって設定されたデフォルトの名前です。このデータ・セットは、APF 許可されています。

注: IMS.SDFSRESL ライブラリーには、DRA 始動テーブルは含まれていません。DFSFRP マクロを使用して、独自のテーブルを生成する必要があります。以下の命名規則に従って、ロード・モジュールに名前を付けます。

- 文字 1 から 3 = DFS
- 文字 4 から 7 = 指定された 4 バイトの ID

4 バイトの ID は、接続先となる IMS システムの IMSID あることが望まれます。ただしこれは必須要件ではありません。

- 文字 8 = 0 (ゼロ)

DRA 始動テーブル・モジュール名が、既存の IMS モジュールの名前とは同じでないことを確認してください。モジュールが不用意に重なることを防止するために、モジュールは、IMS 領域ではなく z/OS アプリケーション領域がアクセス可能なロード・ライブラリーに置いてください。

ご使用のバージョンに別のライブラリーを使用する場合は、そのライブラリーが APF 許可されていることを確認してください。DRA 呼び出し可能インターフェースは、DRA 始動テーブルで指定されたデータ・セット名を使用して、ライブラリーを動的に割り振ります。同一の z/OS アプリケーション領域から IMS に対して複数の接続ができるように、DDNAME が生成されます。

DFSPRP マクロについては、「IMS V15 システム・プログラミング API」に記載されています。

## z/OS アプリケーション領域での ODBA モジュールと DRA モジュールのロードおよび実行

---

次の ODBA モジュールと DRA モジュールを、z/OS アプリケーション領域の STEPLIB または JOBLIB に置きます。これらのモジュールは、IMS とともに IMS.SDFSRESL で出荷されます。

### このタスクについて

#### DFSCDLIO

このモジュールは、アプリケーション・プログラムによりバインドまたはロードされます。DFSCDLIO には、ALIAS 名 AERTDLI も含まれています。

#### DFSAERGO

このモジュールは、DFSCDLIO によりロードされます。

#### DFSAERMO

このモジュールは、z/OS アプリケーション領域の DFSAERGO により接続されます。

#### DFSAERA0

このモジュールは、DFSAERMO により、初期設定を行うため、指定された IMS DB サブシステムに接続されます。

## アプリケーション・プログラムのバインディング

---

ODBA アプリケーション・プログラムを DFSCDLIO (AERTDLI) とバインドします。また、Register 1 の AIB 呼び出しリストを受け渡すロードおよび分岐コマンドを発行する方法もあります。

### このタスクについて

#### 関連資料

AERTDLI インターフェース (アプリケーション・プログラミング)

## セキュリティの確立と定義

---

IMS では、ODBA インターフェースを使用するアプリケーション・プログラム用に、セキュリティを確立し定義するオプションを複数用意しています。

選択するオプションは、使用するセキュリティ環境のタイプと許可方式によって異なります。一般に、IMS が PSB のセキュリティ確保に使用するプロセスには、以下のタイプのセキュリティ検査のいずれかが必要です。

- リソース・アクセス・セキュリティ (RAS)

RACF により、ユーザーが PSB を使用する許可を得ているかどうかを判別するためのセキュリティ検査が実行されます。RACF は、従属領域に定義された RACF セキュリティ・クラス・プロファイルを調べ、許可されているかどうかを判別します。



- APSB セキュリティー

ユーザーが PSB を使用する許可を得ている場合、セキュリティ検査が実行されます。

以下の表は、特定のセキュリティ実装に対するデータ・アクセスを制御するために指定する必要がある値を示します。この表では、指定の各セットについて、実行されるセキュリティ検査のタイプも示しています。

表 141. ODBA を使用するアプリケーションのデータ・アクセスを制御するオプション

セキュリティ・インプリメンテーション	許可方式	ISIS= 指定	ODBASE= 指定	接続セキュリティ	PSB セキュリティー
リソース・アクセス・セキュリティ	RACF	R	N		X
	リソース・アクセス・セキュリティ・ユーザー出口 (RASE)	C	N		X
	RACF および リソース・アクセス・セキュリティ・ユーザー出口 (RASE)	A	N		X
	ありません。	0   N	N		
APSB セキュリティー	RACF	適用されない	Y		X

ODBA インターフェースを使用するアプリケーション・プログラムのデータ・アクセスを制御するには、以下のトピックに示す技法に従って、接続セキュリティと PSB セキュリティー検査を設定します。

**関連概念**

[IMS セキュリティー \(システム管理\)](#)

**関連資料**

[RASE: リソース・アクセス・セキュリティ・ユーザー出口 \(DFSRAS00 およびその他の RASE 出口\) \(出口ルーチン\)](#)

## RAS セキュリティー

z/OS アプリケーション領域に対して PSB を使用する許可を付与するかどうかを制御するには、ISIS 実行パラメーターと ODBASE 実行パラメーターを使用します。

### このタスクについて

以下の表は、特定のオプションを選択した場合にセキュリティをセットアップするために必要なアクションを示します。

表 142. ODBA を使用するアプリケーションの RAS セキュリティーを定義するオプション

指定	実行するアクション
ISIS=0   N および ODBASE=N	アクションは不要です。PSB セキュリティー検査は行われません。

表 142. ODBA を使用するアプリケーションの RAS セキュリティーを定義するオプション (続き)

指定	実行するアクション
ISIS=R および ODBASE=N	RACF により保護しようとする PSB を IIMS または Ixxxxxxx リソース・クラスに定義してから、PSB にアクセスする許可を付与しようとする従属領域のユーザー ID を定義します。IMS の ODBA サポートは、従属領域のタスク (TCBSENV) で受け渡されるセキュリティ環境 (ACEE) が存在する場合はこれを使用し、ACEE がタスク・レベルで存在しない場合は従属領域のアドレス・スペース (ASXBSERV) を使用します。
ISIS=C および ODBASE=N	リソース・アクセス・セキュリティ・ユーザー出口 (RASE) を作成します。このルーチンは、ユーザーが PSB を使用する許可を持っているかどうかを判別できるものでなければなりません。
ISIS=A および ODBASE=N	<ol style="list-style-type: none"> <li>1. RACF により保護しようとする PSB を IIMS または Ixxxxxxx リソース・クラスに定義してから、PSB にアクセスする許可を付与しようとする従属領域のユーザー ID を定義します。IMS の ODBA サポートは、従属領域のタスク (TCBSENV) で受け渡されるセキュリティ環境 (ACEE) が存在する場合はこれを使用し、ACEE がタスク・レベルで存在しない場合は従属領域のアドレス・スペース (ASXBSERV) を使用します。</li> <li>2. リソース・アクセス・セキュリティ・ユーザー出口 (RASE) を作成します。このルーチンは、ユーザーが PSB を使用する許可を持っているかどうかを判別できるものでなければなりません。</li> </ol> <p>RACF が最初に呼び出され、次に出口ルーチンが呼び出されます。</p>

## APSB セキュリティーの定義

ユーザーに対して、PSB を使用する許可を付与するかどうかを制御するには、ODBASE 実行パラメーターを使用します。

### このタスクについて

ODBASE=Y の時に APSB セキュリティーをセットアップするには、以下の手順を実行してください。

### 手順

1. RACF によって保護する PSB を AIMS または Axxxxxxx 一般リソース・クラス (ここで、xxxxxxx は、RCLASS= 初期設定 EXEC パラメーターに指定された値) に定義します。
2. IMS システム定義時に RCF= T | N | および ISIS= R | C を指定します。

---

## 第 10 部 Open Transaction Manager Access (OTMA)

Open Transaction Manager Access (OTMA) に関する以下のトピックは、OTMA のサーバーまたはクライアントに対して、インストール、設計、カスタマイズ、操作、およびリカバリー手順を行う責任者である IMS システムおよび Transaction Manager の管理者のためのものです。



## 第 42 章 OTMA の概要

IMS Open Transaction Manager Access (OTMA) は、トランザクション・ベースの、コネクションレス・クライアント/サーバー・プロトコルです。

これは汎用的なものですが、インプリメンテーションは z/OS シスプレックス環境では IMS 固有です。このプロトコルのドメインは、z/OS システム間カップリング・ファシリティ (XCF) のドメインに制限されません。

OTMA は、クライアントがハイパフォーマンスを維持しながら、大規模ネットワーク (あるいは多数のセッション) をサポートするために、どのようにクライアントをサーバーに接続するかという問題に対処します。

今日使用可能なその他のソリューションは、システム・ネットワーク体系 (SNA) といったネットワーク・ベースのプロトコルを使用します。しかし、これらのプロトコルは、トランザクション・ベースではないので、非常に大きなオーバーヘッドがかかります。

### OTMA とは？

OTMA は、ネットワーク・プロトコルと類似性があります。

ネットワークにはいくつかの体系モデルが存在します。以下の図は、そのうちの 2 つを示しています。右方に表示された単純化 4 層モデルは、UNIX ネットワークの説明によく使用されます。左方に表示されているように、オープン・システム間相互接続 (OSI) モデルでは、OTMA はセッション層です。どちらのモデルも、トランスポート、ネットワーク、およびデータ・リンク層を持っています。OSI モデルは、アプリケーション、プレゼンテーション、およびセッション層も含み、単純化モデルはプロセス層も含んでいます。4 層モデルでは、OTMA はプロセス層です。

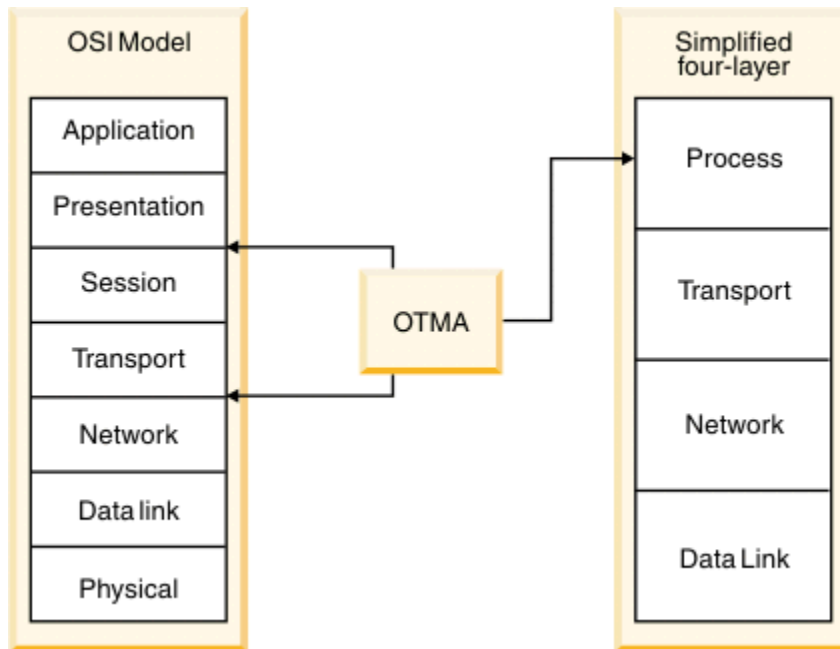


図 127. ネットワーク体系モデル

ただし、OTMA は正確に OSI モデルに準拠しません。以下が真の場合、OTMA は、単一トランスポート接続を使用して同時に複数のセッションを処理できるからです。

- z/OS システム間カップリング・ファシリティ (XCF) は、トランスポート層である。
- セッションは、IMS とクライアント間の接続である。
- クライアントあるいはサーバーは、単一 XCF 接続のみを作成する。

OTMA は、OSI トランスポート層の基本機能のいくつか (XCF では行わない) を実行するので、簡単に言えば、OTMA を XCF と OTMA の両方で構成された トランスポート層を持つ、結合したセッションとトランスポート層と考えることができます。

OTMA をネットワーク体系モデルのセッションとトランスポート層として考えることはできますが、OTMA は z/OS プログラムが IMS アプリケーションにアクセスすることを許可する、高性能の 包括的プロトコルとして設計されています。

**定義:** この場合の z/OS プログラム は、IMS を含む XCF グループのメンバーである 任意の z/OS アプリケーションを意味します。IMS が通信する XCF グループ・メンバーは、OTMA クライアント と呼ばれます。

OTMA を使用することによって、各クライアント (z/OS アプリケーション) は、IMS にトランザクションをサブミットするか、IMS コマンドを発行して、IMS アプリケーション・プログラムと IMS 自身から出力を受信します。

**定義:** IMS は、多くの OTMA クライアントと通信できる (サービスを提供できる) ため、IMS はサーバー と呼ばれます。しかし、OTMA は以下の IMS 環境でのみ操作します。

- IMS TM および DB (IMS DB/DC 環境)
- IMS TM と Db2 for z/OS (IMS DCCTL 環境)

### 関連概念

847 ページの『[OTMA クライアント](#)』

OTMA 環境には、サーバーと 1 つ以上のクライアントが含まれます。

## OTMA の機能

OTMA の機能には、IMS アプリケーション、IMS および OTMA コマンド、リソース・モニターなどのサポートが含まれます。

OTMA クライアントは、タイプ 1 コマンドやタイプ 2 コマンドなど、ほぼすべての IMS コマンドを発行でき、これらのコマンドの結果として応答を受信できます。

各 OTMA クライアントについて、個別にセキュリティ・レベルを指定できます。あるいは、クライアントのメッセージについてセキュリティ検査を行わないよう指示して、セキュリティ処理のオーバーヘッドを最小化することもできます。

OTMA メッセージ・フローおよび同期点プロトコルは、トランザクションごとに OTMA クライアントによって修正可能です。言い換えれば、使用されるトランザクション処理プロトコルは、現行セッションに従属しないということです。

OTMA は、IMS の着信トランザクションの処理状況をモニターして、IMS システムの処理機能が低下した場合に、クライアントにアラートを出すことができます。

OTMA はトランザクションをモニターして、処理が完了する前に指定されたトランザクション有効期限を超過した場合に、そのトランザクションを取り消すことができます。

処理されると同時に待機することができる OTMA クライアントについて、1 つの OTMA クライアントからの入力メッセージ数に限度を定義できます。これにより、OTMA メッセージに使用されるストレージの量が多すぎるために起きる異常終了が防止されます。OTMA は、メッセージ・カウントが限度に近づくと、警告メッセージを発行します。メッセージ・カウントが限度に到達すると、OTMA はメッセージあふれ状態を識別し、メッセージ・カウントが低下するか新しい限度が設定されるまで、OTMA クライアントからの新規メッセージをすべて拒否します。

接続を停止したり、すでに IMS が処理しているトランザクションまたはコマンドに影響を与えずに、個々の OTMA クライアントから着信するトランザクションとコマンドを停止できます。

OTMA は、ローカル IMS システムの従属領域で実行されている IMS アプリケーション・プログラムからトランザクション・メッセージを、処理のために TCP/IP ネットワーク経由で別の IMS システムに送信することをサポートします。IMS Connect は TCP/IP 接続を管理する必要があります。

また、OTMA は、IMS 従属領域で実行されている IMS アプリケーション・プログラムからのコールアウト要求をサポートします。コールアウトは、DL/I ICAL 呼び出しを使用して実行されます。コールアウト要求を使用すると、IMS アプリケーション・プログラムから、IMS インストール済み環境の外部のサーバーに

あるデータまたはサービスを要求できます。コールアウト要求が同期処理されるか非同期処理されるかによって、コールアウト要求の OTMA サポートが異なります。

OTMA は、IMS Connect とそのクライアント (IMS Enterprise Suite SOAP Gateway および IMS TM Resource Adapter など) と IBM MQ にコールアウト・メッセージを経路指定することができます。IMS は OTMA を内部で使用して (OTMA が使用可能になっているかどうかに関係なく)、IMS アプリケーションに同期プログラム間通信要求を経路指定します。

IMS /DISPLAY TRANSACTION コマンド出力は、メッセージ接頭語のアプリケーション・データ・セクションにあるクライアントに戻される OTMA メッセージの形式です。

OTMA 開始トランザクションは、IMS と OTMA 間の論理接続を識別する OTMA トランザクション・パイプ名を使用して、z/OS ワークロード・マネージャーに識別されます。

OTMA で使用されたことがない既存の IMS アプリケーション・プログラムは、修正や OTMA クライアントとの対話なしに実行できます。IMS SETO 呼び出しを使用する APPC/IMS アプリケーション・プログラムは、多少の修正が必要になることがあります。

## 関連概念

[877 ページの『コールアウト要求の OTMA サポート』](#)

IMS 従属領域で実行されている IMS アプリケーション・プログラムは、IMS の Open Transaction Manager Access (OTMA) コンポーネントを介して、データまたはサービスを要求する目的で、IMS インストールの外部にあるサーバーに対してコールアウトすることができます。同期コールアウト・インターフェースを使用して、同期プログラム間通信により別の IMS アプリケーションからのサービスを要求することもできます。

[827 ページの『OTMA のトランザクション有効期限サポート』](#)

トランザクションに有効期限を指定して、クライアントが使用できなくなったトランザクションを IMS に処理させないようにすることで、処理コストを削減できます。

[815 ページの『OTMA リソース・モニター』](#)

OTMA は、OTMA トランザクションの処理に使用される IMS システム・リソースをモニターし、IMS システムの OTMA トランザクション処理の状況を OTMA クライアントに通知します。

## 関連資料

[814 ページの『OTMA の制約事項および要件』](#)

OTMA には一般の制約事項と要件がいくつか適用されます。

## OTMA を使用する利点

OTMA を使用しない IMS アプリケーションでは、OTMA を使用するための変更は不要です。ただし、SETO 呼び出しを使用する場合を除きます。その場合は、OTMA を使用するための変更が必要になることがあります。

SETO 呼び出しは、APPC/IMS および SPOOL/API 処理に適用されます。

OTMA 発信元トランザクションごとに、SETO 呼び出しが状況コードを戻します。必要な場合、SETO 呼び出しと INQY 呼び出しを一括して扱うこともできます。

全二重処理により、トランザクションと出力メッセージの送信および処理が並列的に行われる環境が提供されます。

IMS 装置サポートを IMS の外側でインプリメントします。IMS が提供するものとは違う IMS サブシステム用の装置サポートをインプリメントするか、または IMS が提供しない装置サポートを使用可能にすることもできます。次の図は、IMS が OTMA クライアント内でインプリメントされている装置サポートを使用して、装置 (ここではワークステーションとして表示されています) と通信する方法を示します。仮想記憶通信アクセス方式 (VTAM) を使用する IMS 装置サポートを比較のために示します。



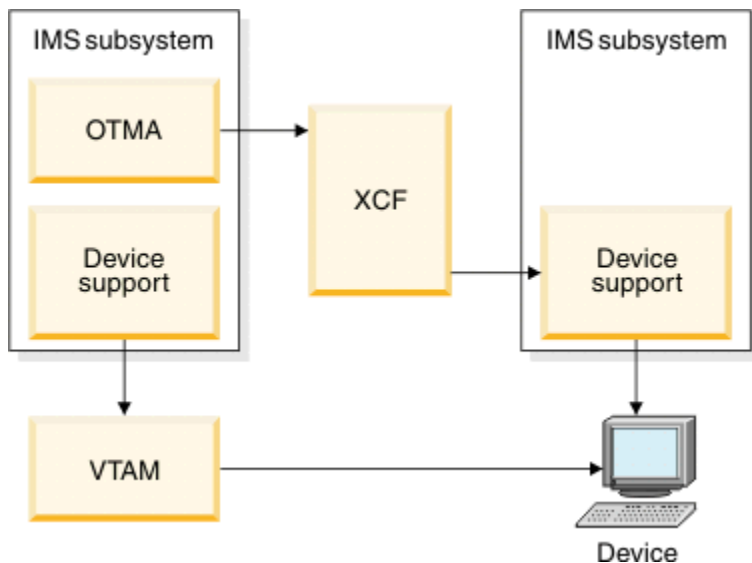


図 128. IMS は、OTMA クライアント内でインプリメントされている装置サポートを使用している 装置を使用して通信を行います。

フロー制御とトランザクション処理属性は、トランザクションに動的に結合されます。

クライアントは、IMS にハイパフォーマンス・アクセスができます。

- OTMA は、z/OS システム間カップリング・ファシリティー アプリケーション・プログラミング・インターフェース (API) を使用。
- OTMA は VTAM および IMS の装置依存サポートを使用しない。

異なったプロトコル (つまり、リカバリー可能あるいはリカバリー不能であるような異なった処理要求を持つ) に基づいたトランザクションは、単一トランザクション・パイプと関連付けることができます。

OTMA グループ当たり 255 クライアントまで接続することができます。

メッセージ接頭語のユーザー・データ・セクションを使用して、メッセージを拡張でき、トランザクションで追加のユーザー情報を送信することができます。

ユーザー情報とトランザクション・パイプ名は、メッセージ自身に組み込まれます。

一意的な名前付きリソースを使用しなくても、異なったクライアントが、同じトランザクション・パイプ名を指定することができます。

SNA (システム・ネットワーク体系) のような ネットワーク体系を使用する必要はありません。

#### 関連概念

[783 ページの『OTMA でのトランザクション・パイプの使用』](#)

IMS トランザクションは、何らかの処理を行うための、IMS に対する要求を表します。多くのトランザクションは、IMS が処理を完了した後、応答も要求します。したがって、各トランザクションは、ソース (リクエスター) と (応答の) 宛先を持ちます。

#### 関連資料

[841 ページの『OTMA 環境での DL/I 呼び出しの使用』](#)

一部の DL/I 呼び出しには、OTMA で使用する場合に特別な考慮事項があります。

## OTMA プロトコルの利点

OTMA は、データ・オブジェクトをトランザクションとして扱います。それらのデータ・オブジェクトは、アプリケーション層、セッション層、あるいはトランスポート層の考慮事項から独立した属性を持っています。つまり、OTMA は、トランザクション層で、他の層から独立しています。

OTMA は固有の層なので、他のソリューションが提供しないような柔軟性、単純性、およびパフォーマンスが得られます。このセクションでは、OTMA がクライアントに提供するトランザクション特有のサービスの概要を示します。

トランザクション・パイプを使用したトランザクションのグループ化。

セキュリティー・オプション (例えば、クライアントがセキュリティーを検査することができ、あるいはユーザー ID をサーバーに検査させることができる)。

動的結合フロー制御と処理。クライアントは、トランザクション要求と応答をサーバーに処理させる方法を決めることができます。

クライアントが、サーバーがサポートするトランザクションを照会する機能。

オブジェクトをトランザクションとして扱うこと。クライアントは、トランザクションに関連ユーザー・データを組み込むことができ、そのデータを、そのトランザクションによって生成されたすべてのメッセージに含めることができます。

クライアントが、入力と出力を関連させるために各トランザクションでクライアント・トークンを指定する機能。

クライアントが、サーバーが行うトランザクション処理を制御する機能。これには以下のものがあります。

- クライアントは、所属する OTMA グループのセキュリティー・レベルと異なるセキュリティー・レベルを設定できます。
- パフォーマンスを向上させるために、クライアントは、サーバーが配信するメッセージに対して行うセキュリティー検査をなくすことができます。
- トランザクション・パイプ・トークンを使用したトランザクションのグループ化
- synclevel=confirm または synclevel=synchpt を使用する OTMA クライアントは、送信後コミット・メッセージのタイムアウト値を指定できます。

クライアント・ルーティング。代替 PCB に挿入される出力メッセージをいずれかの OTMA クライアントまたは IMS に転送するために、DFSYDTx PROCLIB メンバーで OTMA 宛先記述子を容易にコーディングすることができます。代わりに、代替 PCB に挿入される出力メッセージを転送するために、IMS 出口ルーチンをコーディングすることもできます。

設計済みコマンド出力。クライアントは、IMS /DISPLAY TRANSACTION コマンドを使用して、サーバーのトランザクション属性を照会し、構造化されたフォーマットで応答を受信することができます。それ故、処理を制御するための自動化操作プログラム・スクリプトの必要性は減少します。

APPC と違って、トランザクション・パイプを介したメッセージ・フローを使用するとき、すべてのトランザクション用のフロー制御パラメーターおよびセッション用の関連出力データを含むセッションの概念は存在しません。

## OTMA 環境での IMS メッセージ・フロー

OTMA のメッセージ・フローに重要なのは、トランザクション・パイプ、すなわちサーバーと OTMA クライアント間の論理接続です。

OTMA クライアントは、入力メッセージ用のメッセージ接頭語のメッセージ制御情報セクションに、トランザクション・パイプ名を組み込んでいます。そこで IMS は、OTMA クライアントのアプリケーション出力と特定のトランザクション・パイプを関連付けます。

### 関連概念

[783 ページの『OTMA でのトランザクション・パイプの使用』](#)

IMS トランザクションは、何らかの処理を行うための、IMS に対する要求を表します。多くのトランザクションは、IMS が処理を完了した後、応答も要求します。したがって、各トランザクションは、ソース (リクエスター) と (応答の) 宛先を持ちます。

## 基本的な OTMA メッセージ・フロー

OTMA クライアントの基本メッセージ・フローは、比較的単純です。

基本メッセージ・フローは以下のとおりです。

1. クライアントは、トランザクションまたはコマンドを IMS にサブミットします。
2. IMS は IMS トランザクションをクライアントからの入力として受け入れます。

IMS トランザクション・コードは、入力メッセージのアプリケーション・データ・セクションに指定されます。

クライアントが IMS コマンドをサブミットしている場合、コマンドは、入力メッセージのアプリケーション・データ・セクションに組み込まれています。

3. 入力メッセージが処理されます。

IMS トランザクションは、IMS スケジューラー・メッセージ・ブロック (SMB) を使用して、適切なアプリケーション・プログラムにエンキューされます。

IMS コマンドは、IMS によって処理されます。出力が、要求タイプに従って、クライアントに同期的あるいは非同期的に送信されます。

4. アプリケーション出力がクライアントに送信されます。

出力の生成とコミットは、入力メッセージ用のメッセージ接頭語の状態データ・セクションに指定されているコミット・モードに基づいて調整されます。

アプリケーション出力は、クライアントに送信される前に、動的に作成された IMS トランザクション・パイプ構造(そのクライアントに特定のもの)にエンキューされます。

OTMA サブミットのトランザクションに対して、IOPCB 出力が、OTMA クライアントに戻されます。デフォルトでは、代替 PCB 出力のすべても、OTMA クライアントに送信されます。これは、OTMA DFSYDTx PROCLIB メンバーに OTMA 宛先記述子をコーディングするか、OTMA 宛先解決ユーザー出口 (OTMAYPRX) またはクライアントの OTMA ユーザー・データ・フォーマット設定ユーザー・ルーチン (DFSYDRU0) をコーディングすることによって変更できます。また、これらの出口ルーチンを使って、代替 PCB 出力を非 OTMA サブミット・トランザクションから OTMA クライアントに経路指定することもできます。

z/OS システム間カップリング・ファシリティ (XCF) ではメッセージの順次送達は保証されませんが、IMS はセグメント化されたメッセージを順番に送達します。

以下の図は、OTMA 環境でのメッセージ・フローの例を示します。この例では 2 つのクライアントが横並びに示されていますが、これらは TCP/IP クライアント、IBM MQ Queue Manager クライアント、あるいは他のネットワーク・タイプのクライアントが可能です。メッセージ・フローはクライアントから開始され、XCF グループを通り、IMS に到達します。IMS アドレス・スペース内では、OTMA は制御領域に含まれます。つまりメッセージ・フローはトランザクション・パイプで終了します。IMS アプリケーション・プログラムは、従属領域で Get Unique (GU) 呼び出しを発行します。

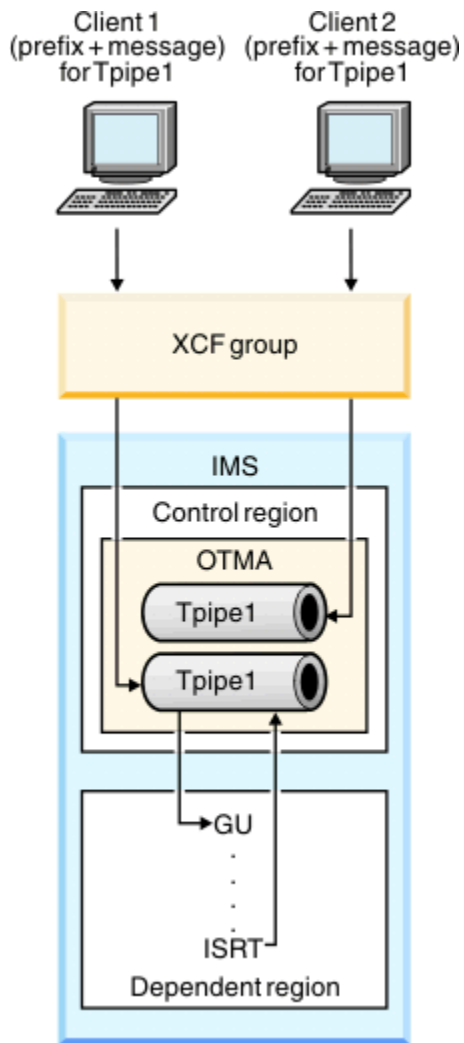


図 129. OTMA 環境での IMS メッセージ・フロー

前の図では、以下のようになっています。

1. メッセージ接頭語は、セグメント化された入力でも、常に入力トランザクションに付加されています。この接頭語は、トランザクション・パイプ名やクライアント・トークンのような重要な情報を含んでいます。

クライアント・アプリケーション・プログラムは、同じトランザクション・パイプ名を指定して、いくつかのトランザクションを送信することができます。クライアント・アプリケーション・プログラムが、受信する IMS 出力の処理方法を知ることができるように、クライアント・トークンは、常に接頭語になければなりません。

2. OTMA クライアントは、トランザクション・パイプを事前定義する必要はありません。2つの異なるクライアントが同じトランザクション・パイプ名を使用できます。多数のクライアントが同じトランザクション・パイプ名を使用できますが、各トランザクション・パイプは固有です。client 1 と client 2 は両方とも tpipe1 を使用しますが、それぞれが固有トランザクション・パイプです。

クライアントは、必要なだけのトランザクション・パイプを作成することができます。

3. トランザクション・パイプ構造は、OTMA が出力を受信するとき動的に作成され、アプリケーション出力のアンカーとして使用されます。
4. IMS アプリケーション・プログラムは、GU 呼び出しを発行するとき、OTMA メッセージ接頭語を認識しません。

IMS は、クライアント/サーバー・セッションに対して全二重メッセージ・フローをサポートします。クライアントは、代わりに半二重メッセージ・フローを要求することができますが、このフローは、クライアント自身がインプリメントし、管理する必要があります。

- メッセージ接頭語の状態データ・セクションにある相関係数トークンは、トランザクションを一意的に識別するために使うことができます。IMS は、トランザクション用のメッセージ接頭語のこのフィールドを維持します。
- クライアントは、メッセージ接頭語のメッセージ制御情報セクション中の応答要求フラグをメッセージの応答を受信するために設定することができます。
- IMS からの非送信請求出力を、クライアントが容易に識別することができるのは、メッセージ接頭語がトランザクション・パイプ名だけを指定するからです。クライアントは、IMS に出力を廃棄するか尋ねることができます。

非送信請求出力では、半二重処理を干渉しないようにします。つまり、クライアントは、ユーザー・トークンで半二重フローを維持しながら、全二重フローを準備する必要があります。競合でエラー状態にならないようにします。

## OTMA の IMS 間 TCP/IP 通信フロー

IMS 従属領域でローカルで実行されている IMS アプリケーション・プログラムは、トランザクション・メッセージを処理のために TCP/IP ネットワーク経由でリモート IMS システムに送信することができます。

TCP/IP ネットワーク経由で別の IMS システムに送信されるトランザクション・メッセージのフローを、以下のステップで説明します。

1. ローカル IMS システムで実行されている IMS アプリケーション・プログラムが、トランザクション・メッセージの宛先として指定された OTMA 宛先記述子の名前を使用して CHNG 呼び出しを発行し、次にトランザクション・メッセージを送信するために ISRT ALTPCB 呼び出しを発行します。代替方法として、OTMA 宛先記述子を使用する代わりに、ローカル IMS システムで OTMA ユーザー・データ・フォーマット設定出口ルーチン (DFSYDRU0) を使用して、ALTPCB 出力をリモート IMS システムに経路指定することもできます。
2. IMS は、OTMA 宛先記述子に指定された情報に基づいて、トランザクション・メッセージをローカル IMS Connect インスタンスに経路指定します。スーパーメンバー・グループがローカルでアクティブになっている場合、IMS は出力トランザクションを、スーパーメンバー・グループ内の最大 8 つの IMS Connect インスタンスにラウンドロビン・アルゴリズムを使用して配布します。
3. IMS Connect は、コミット後送信 (CMO) および確認応答付き送信専用プロトコルを使用して、ローカル IMS Connect インスタンスによって生成されたクライアント ID で識別された、TCP/IP 接続上のリモート IMS Connect インスタンスに、トランザクション・メッセージを送信します。
4. リモート IMS Connect インスタンスは、リモート IMS システム内の OTMA にメッセージを送信します。
5. リモート IMS システム内で、OTMA は確認応答 (ACK) を IMS Connect に返し、メッセージを IMS トランザクションとしてキューに入れます。
6. リモート IMS システムで実行されている IMS アプリケーション・プログラムが、トランザクションを処理します。
7. リモート IMS システム内の IMS アプリケーション・プログラムによって生成された出力はすべて、T パイプの保留キューに入れられます。

次の図は、このフローを示しています。

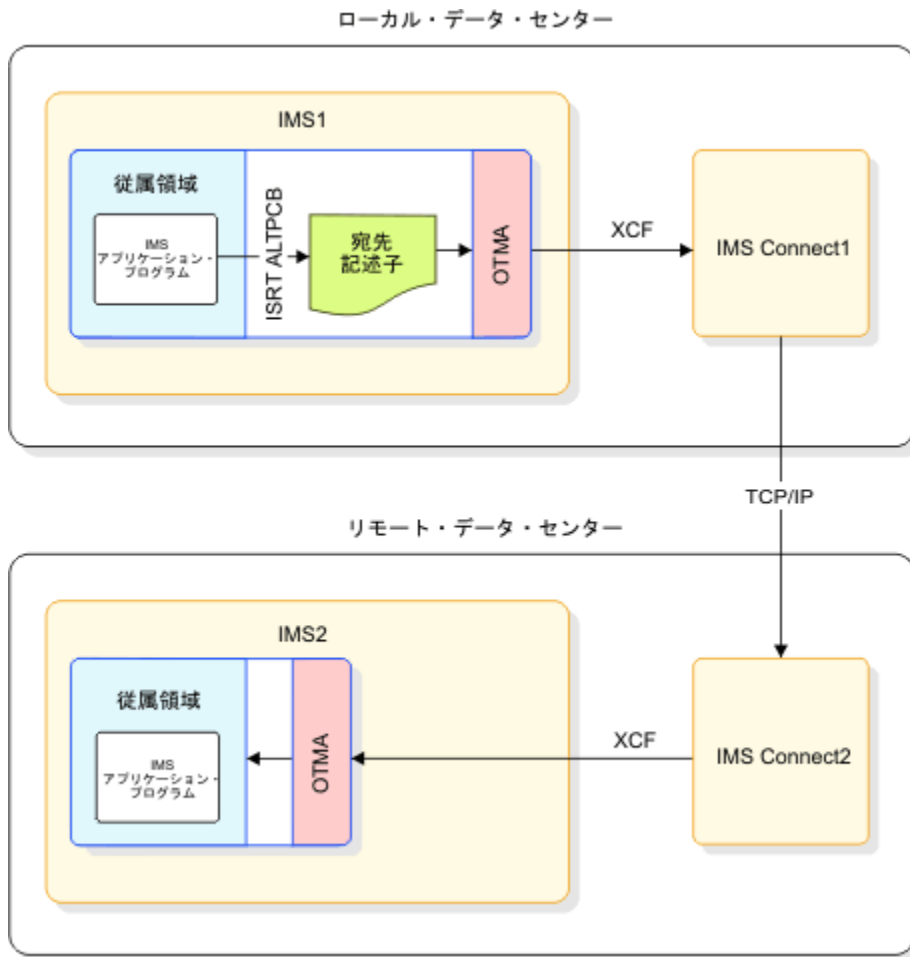


図 130. OTMA トランザクション・メッセージの IMS 間 TCP/IP 通信フロー

#### 関連概念

799 ページの『OTMA 宛先記述子の定義』

OTMA 宛先記述子を作成、削除、および更新するには、オンライン・タイプ 2 IMS コマンドを使用するか、または IMS.PROCLIB データ・セットの DFSYDTx メンバーに直接コーディングします。

154 ページの『IMS Connect による IMS 間 TCP/IP 通信のサポート』

IMS Connect は、TCP/IP ネットワークを介して相互に通信する IMS システムの TCP/IP 接続およびプロトコルを管理します。

#### 関連タスク

[IMS 間の TCP/IP 接続 \(システム定義\)](#)

## コミット後送信トランザクション処理フローのサンプル

以下の図は、非 OTMA 環境を示します。2 次論理装置タイプ 2 (SLU 2) 装置は、VTAM と IMS 装置サポート (DDM) を使用して、IMS と通信しています。トランザクションは、IMS メッセージ・キューにエンキューされます。トランザクション出力は、SLU 2 装置に戻されます。

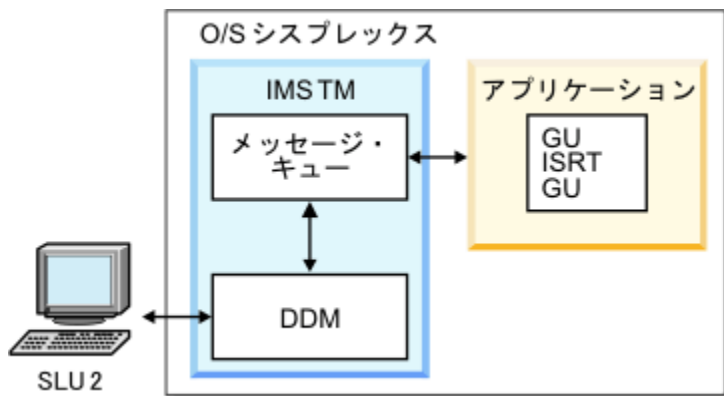


図 131. 標準 SLU2 トランザクション・フロー

以下の図は、OTMA 環境での同じトランザクション・フローを示します。トランザクションは、依然として SLU2 装置から来ていますが、装置は、VTAM ではなく、z/OS システム間カップリング・ファシリティ (XCF) グループを通して OTMA クライアントを使用して IMS と通信します。

この図は、入力フローのみ表示します。このフローは SLU2 装置から始まり、OTMA クライアントに到達し、XCF グループを通して OTMA サーバーで終了します。トランザクションはメッセージ・キュー上に置かれ、アプリケーションは get unique、insert、および get unique 呼び出しを発行します。出力は同じパスを逆に進みます。もちろん、クライアントが出力を SLU2 装置に送信する場合は、SLU2 装置は、クライアントに定義する必要があり、またクライアントは、その装置を駆動できる必要があります。

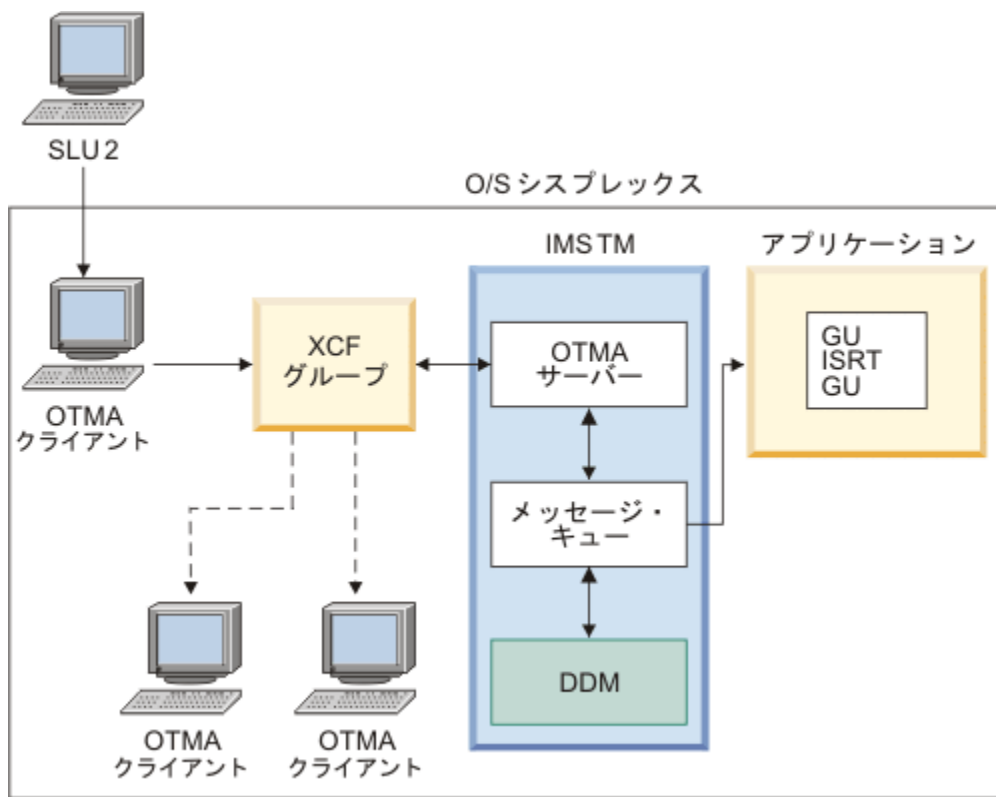


図 132. OTMA を使用した SLU2 トランザクション・フロー

OTMA フローはより複雑に見えますし、SLU2 装置についても同様です。しかし、OTMA を使用すれば、VTAM サポート装置のみでなく、どのような装置タイプでも IMS と通信できます。OTMA クライアントは、TCP/IP ネットワークのような他のネットワークに対して、ゲートウェイとしても機能することができます。



## OTMA でのトランザクション・パイプの使用

IMS トランザクションは、何らかの処理を行うための、IMS に対する要求を表します。多くのトランザクションは、IMS が処理を完了した後、応答も要求します。したがって、各トランザクションは、ソース (リクエスター) と (応答の) 宛先を持ちます。

IMS は、論理端末 (LTERM) の概念を使用して、応答が正しいリクエスターに関連付けられていることを確認します。LTERM は、トランザクション出力をリクエスターに戻す前に、それを保持しておくキューを使用します。

**定義:** トランザクション・パイプ (Tパイプ) とは、クライアントとサーバー間の論理接続です。それは、IMS 論理端末 (LTERM) と類似しています。各 LTERM に対して、IMS は、キューと出力を受信する物理ノード間の接続を維持します。OTMA は、LTERM を使用しませんが、クライアントと IMS 間の接続は依然維持します。この接続は、トランザクション・パイプ、すなわち Tパイプです。

トランザクション・パイプにより、クライアントはそのトランザクションをトランザクション・パイプ名に関連付けることができます。IMS は、トランザクション・パイプ名を使用して、入出力すべてと特定のクライアントを関連付けます。トランザクション出力とその最終宛先 (例えば、端末装置あるいはプリンターのユーザー) 間の関連付けは、IMS 内 (LTERM の場合のように) ではなく、クライアントの責任です。

トランザクション・パイプを使用することにより、IMS は、トランザクションの実ユーザーについては何も知らず、多くの場合はクライアント・アプリケーションのユーザーについても知りません。IMS は、実ユーザーについては何も知らないため、クライアントがトランザクションの出力を完全に制御します。

OTMA はトランザクション・パイプを使用して以下を提供します。

### • 柔軟性

多くのトランザクション出力は、同じトランザクション・パイプを流れることができます。

### • パフォーマンス

トランザクション・パイプを使用することにより、クライアントは、メッセージ・フロー制御と同期化に基づいてトランザクションを指定し、識別することができます。

### • クライアントと IMS 間の再同期化

トランザクション・パイプは、同期化したり、非同期化することができます。同期化トランザクション・パイプの場合、出力メッセージはすべて単一プロセスで逐次化され、シーケンス番号がメッセージに割り当てられます。これらの逐次化メッセージのロギングにより、もし障害が発生した場合、IMS とクライアントは再同期化することができます。

非同期化トランザクション・パイプでは、再同期化は必要ありません。

### • オブジェクト方向付け

トランザクションは、OTMA がトランザクション・メッセージ情報 (ユーザー・データおよびトランザクション・パイプ名など) をメッセージ内に保持するので、オブジェクトとして考えられます。

IMS は、以下の環境にある場合を除いて、トランザクション・パイプが 3 つの連続したシステム・チェックポイントでアイドル状態だった場合、その後にトランザクション・パイプを除去します。

- Tパイプまたは Tパイプ保留キューに、コミット後に送信のメッセージが入っている。
- tpipe は停止します。
- Tパイプにトレースが設定されている。
- Tパイプが同期化されている (コミット後に送信の入力トランザクション用に MQSeries® によって使用される Tパイプなど)。
- AUTO または SINGLE-WAIT のいずれかのオプションを指定した RESUME TPIPE 要求のために、Tパイプが待ち状態にある。
- Tパイプが MCP 状態であり、Tパイプが共用キュー環境で実行されていて、グローバル・キューに出力メッセージがある可能性があることを示している。

ヒント:Tパイプのキューに入れられたメッセージがなく、Tパイプについて MCP 状況が表示されていて、Tパイプを除去できない場合は、/DISPLAY TMEBER *tmembername* TPIPE *tpipename* QCNT コマンドまたは /DISPLAY TMEBER *tmembername*QCNT コマンドを発行して、MCP 状況をリセットしてください。

- Tパイプが IMS によってスキャンされている。

/DISPLAY TMEBER TPIPE コマンドを使用して、上記のリストのいずれかの状況がTパイプに当てはまるために IMS で Tパイプを除去できないかどうかを確認することができます。

以下の図は、トランザクション・パイプを OTMA クライアント/サーバー環境に適合させる方法を示します。図に示すように、トランザクション・パイプ構造は、サーバーの場合にのみ OTMA 層に常駐します。トランスポート層に常駐している z/OS システム間カップリング・ファシリティは、プロセス間通信層と考えることができます。クライアント・プロセスとサーバー・プロセス間を通信させるからです。

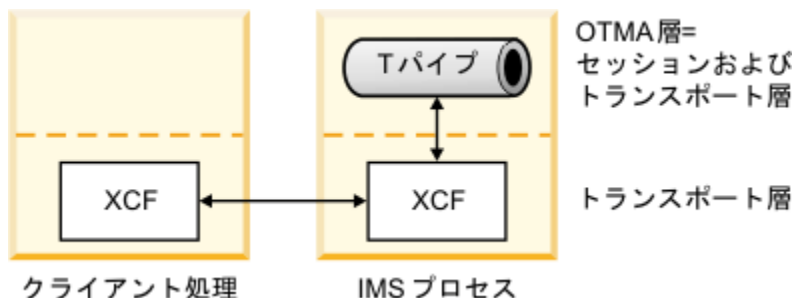


図 133. トランザクション・パイプを OTMA クライアント/サーバー環境に適合させる方法

#### 関連資料

[/DISPLAY TMEBER コマンド \(コマンド\)](#)

## トランザクション・パイプの違い

IMS LTERM および UNIX パイプは、両方ともメッセージ・トラフィックに対する片方向のフローを提供します。OTMA トランザクション・パイプは、両方向のフローを提供します。

トランザクション・パイプの概念は、どのプロトコルにも適用できます。一般的には、トランザクション・パイプは、以下の理由で IMS LTERM を置き換えます。

- 処理は全二重である。
- 複数フロー制御のメカニズムが可能である。
- 論理出力エンティティ (言い換えれば、LTERM) は、実ユーザーのノードからはずされる。
- トランザクション・パイプは、API としてではなく、クライアント/サーバー・アーキテクチャーを促進するプロトコルとしてインプリメントされる。
- トランザクション・パイプは、セッション特性とは独立したデータ制御メカニズムをセットアップするので、トランザクション特定である。

## トランザクション・パイプを使用したメッセージ・フロー

トランザクションのフロー制御は、クライアントによって操作されます。

クライアントは、サーバー中のトランザクション属性を照会して、動的にフロー制御パラメーターをバインドします。トランザクション・パイプは、通常フロー制御とは関連しません (半二重処理を使用した同期化トランザクション・パイプを除いて)。

以下の図は、z/OS システム間カップリング・ファシリティ (XCF) を使用した、クライアントとサーバー間の基本メッセージ・フローを示します。処理の順序は以下のとおりです。

1. クライアントは、入力としてトランザクションをサーバー (IMS) に送信します。
2. サーバーは、トランザクション出力メッセージをクライアントに戻します。

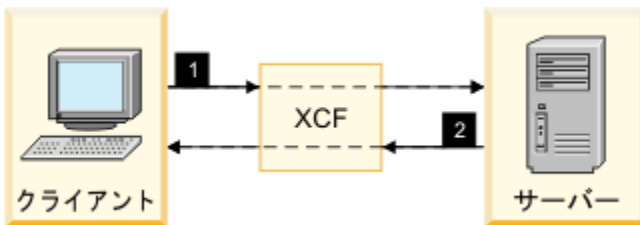


図 134. 基本的なトランザクション・パイプのメッセージ・フロー

サーバー内では、以下の図に示すように、入力トランザクションと出力メッセージは、IMS キューを使用して編成され、同期化されます。この図は、非高速機能環境の場合のコミット後送信トランザクション・フローを示します。

処理の順序は以下のとおりです。

1. クライアントはサーバーにトランザクションを送信し、サーバーはそのトランザクションをメッセージ・キュー上にエンキューします。
2. トランザクションは、処理のためにアプリケーション・プログラムにサブMITされます。
3. アプリケーション・プログラムは、トランザクションの出力を準備し、同期点処理で出力をコミットします。
4. 出力がクライアントに戻されます。

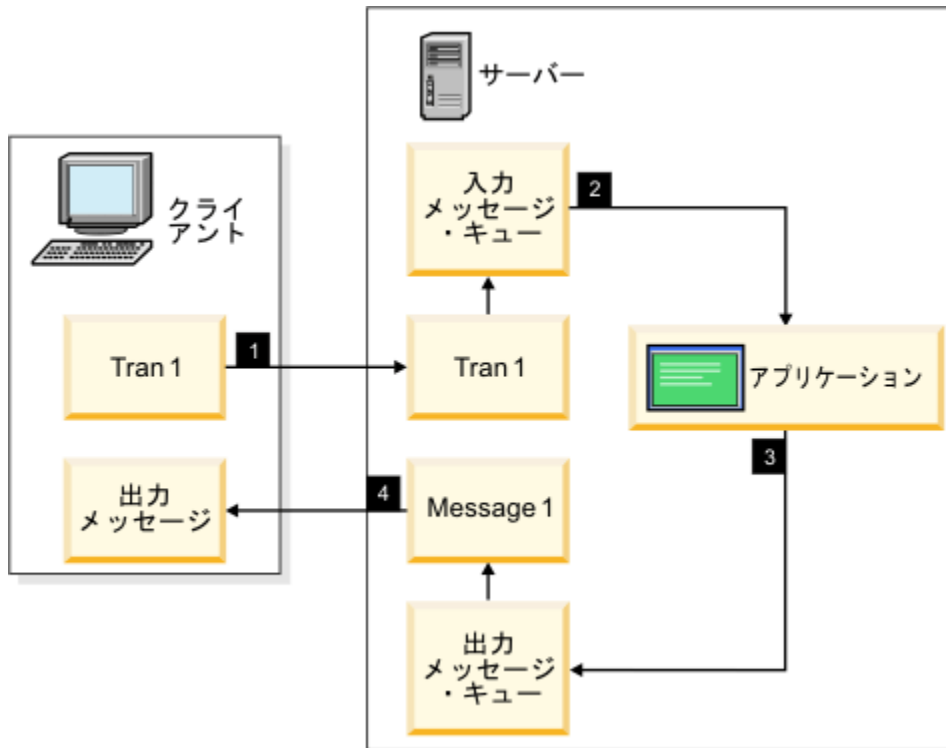


図 135. トランザクション・メッセージ・フローでのキューの使用

全二重環境では、以下の図に示すように、トランザクションおよび出力メッセージは並列で送信され、処理されます。この並列性は、トランザクションと出力メッセージごとにプロセスを作成することにより、最大化することができます。処理の順序は以下のとおりです。

1. クライアントは、トランザクションをサーバーに送信し (Tran 1)、サーバーのトランザクション・パイプはトランザクションをエンキューします。
2. トランザクション (Tran 1) は、処理のためにアプリケーション・プログラムにサブMITされます。
3. アプリケーション・プログラムは、トランザクション (Tran 1) の出力 (Message 1) をエンキューします。

4. クライアントは 2 番目のトランザクション (Tran 2 ) をサーバーに送信し、サーバーのトランザクション・パイプはそのトランザクションを エンキューします。
5. 2 番目のトランザクション (Tran 2 ) は、処理のためにアプリケーション・プログラムにサブMITされます。
6. Tran 1 の出力 (Message 1) は、クライアントに戻されます。
7. アプリケーション・プログラムは 2 番目のトランザクション (Tran 2 ) の出力 (Message 2) をエンキューします。
8. Tran 2 の出力 (Message 2) はクライアントに戻されます。

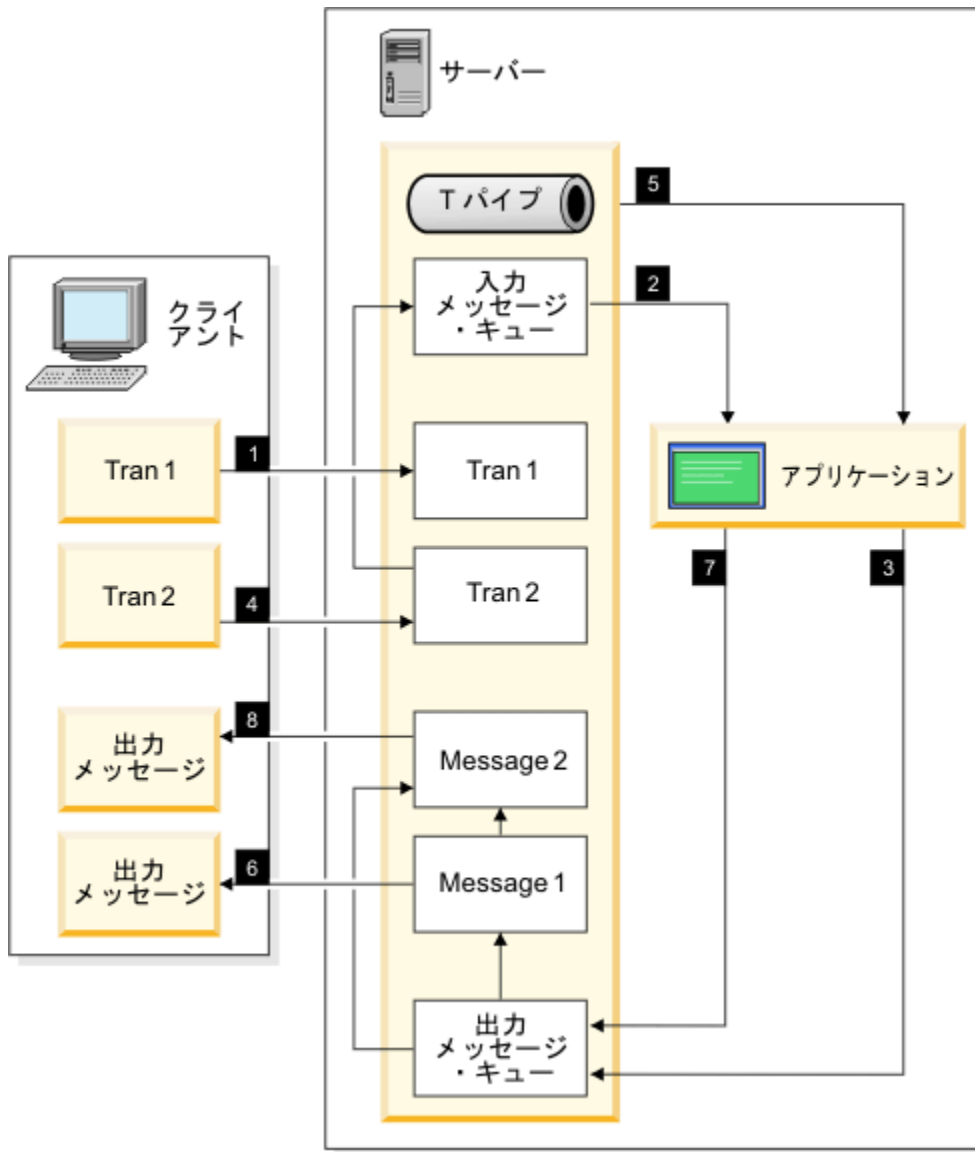


図 136. 全二重環境下でのトランザクション・パイプのフロー

**Q:** トランザクション・パイプを「同期化」として指定した場合、通信フローは半二重になりますか？

**A:** いいえ。トランザクション・パイプは常に全二重です。

実際に通信フローが半二重かどうかは、クライアントによります。同期化トランザクション・パイプの場合、IMS は、受信した順序で、すべての出力メッセージを処理します。IMS が直前のメッセージに対し ACK メッセージを受信するまでは、トランザクションの新規メッセージは送信されません。NAK メッセージにより、IMS は、そのトランザクションのすべての出力処理を停止します。

この出力処理を行っている間、クライアントはその同期化トランザクション・パイプ上で、新規入力トランザクション・メッセージを IMS に送信することができます。クライアントが IMS 出力を受け取るとともにトランザクションの送信を調整する場合は、クライアントは半二重処理を実行することができます。

#### **関連概念**

850 ページの『[OTMA コミット処理](#)』

OTMA は、IMS がトランザクションをコミットする方法を制御します。その方法には、コミット後送信あるいは送信後コミットがあります。



## 第 43 章 OTMA の使用可能化と使用

以下のトピックでは、OTMA 環境での IMS の操作方法に加えて、OTMA 環境に関連した IMS タスクについて説明します。

### このタスクについて

## OTMA の使用可能化

OTMA の使用に当たり、IMS を使用可能にするために、システム定義中に z/OS システム間カップリング・ファシリティ (XCF) グループ名および IMS OTMA メンバー名を指定します。

### このタスクについて

OTMA は、IMS TM と共にインストールされます。IMS INSTALL/IVP ダイアログは、OTMA をインストールするために使用されません。

OTMA を開始するには、IMS システム定義の際に IMS プロシージャで **OTMA=Y** 始動パラメーターを使用するか、IMS の再始動後に、タイプ 1 コマンドの **/START OTMA** を発行することができます。

IMS システム定義時に、z/OS システム間カップリング・ファシリティ (XCF) 名、始動動作、セキュリティーなど、各種の OTMA 属性を定義することができます。

#### 関連資料

[IMS PROCLIB データ・セットの DFSPBxxx メンバー \(システム定義\)](#)

[IMS プロシージャのパラメーターの説明 \(システム定義\)](#)

[IMS プロシージャ \(システム定義\)](#)

[DCC プロシージャ \(システム定義\)](#)

## OTMA 構成パラメーターの要約

以下の OTMA 構成パラメーターはすべて、IMS.PROCLIB データ・セットの DFSPBxxx メンバーおよび、IMS 始動プロシージャまたは DCC 始動プロシージャで指定することができます。

#### GRNAME=

OTMA 通信用に IMS が作成および結合する XCF グループの名前を定義します。

#### OTMA=

IMS の始動時に OTMA を開始するかどうか、および **/START OTMA** コマンドはウォーム再始動後および緊急時再始動後にリカバリー可能であるかどうかを指定します。

#### OTMAASY=

送信後コミット・メッセージの場合、プログラム間通信から発生したトランザクションの同期および非同期のスケジューリングを制御します。

#### OTMANM=

XCF グループ内の OTMA を識別する、XCF メンバー名を指定します。

#### OTMAMD=

OTMA 宛先解決 ユーザー出口 (OTMAYPRX) のパラメーター・リスト内のメンバー・オーバーライド・フィールドを使用して、OTMA クライアントから呼び出されたトランザクション用に別の XCF メンバー名を指定できるかどうかを決定します。

#### OTMASP=

IBM MQ の場合、OTMA 出力用に同期化 T パイプを使用するかどうかを指定します。

#### OTMASE=

使用したい OTMA RACF セキュリティーのタイプ (ある場合) を指定します。



## 関連タスク

829 ページの『OTMA セキュリティー』

OTMA のセキュリティーは、RACF またはそれと同様なセキュリティー製品によって適用されます。以下のトピックでは、RACF を使用した OTMA および OTMA クライアントのセキュリティーの実装について説明します。

## 関連資料

[IMS PROCLIB データ・セットの DFSPBxxx メンバー \(システム定義\)](#)

[IMS プロシーチャーのパラメーターの説明 \(システム定義\)](#)

[IMS プロシーチャー \(システム定義\)](#)

[DCC プロシーチャー \(システム定義\)](#)

## XCF グループ名の定義

OTMA 通信には XCF が必要です。IMS クライアントと OTMA クライアントの両方が、同じ XCF グループに結合する必要があります。

### このタスクについて

DFSPBxxx PROCLIB メンバー内または IMS 制御領域 JCL 内の GRNAME パラメーターで、OTMA 通信用の XCF グループ名を指定できます。有効な XCF グループ名が指定されていない場合は、OTMA を開始できません。

GRNAME パラメーターが指定されている場合、IMS は OTMA が開始された時点でのみ XCF グループに結合します。IMS の開始時に OTMA を開始するように構成されている場合、IMS は始動中に XCF グループに結合します。IMS の始動時に OTMA を開始するように構成されていない場合、IMS は、**/START OTMA** コマンドによって OTMA が開始された後にのみ XCF グループに結合します。

GRNAME= を指定して OTMA を開始した場合は、**/DISPLAY OTMA** コマンドを使用すると XCF 状況を表示することができます。XCF 情報を定義する必要はありません。

セキュリティー用の RACF を使用する場合は、IMSXCF.group.member (クライアント・メンバー名) を RACF FACILITY クラスに指定する必要があります。

## 関連資料

[IMS プロシーチャーのパラメーターの説明 \(システム定義\)](#)

## OTMA XCF メンバー名の定義

XCF では、XCF グループの各メンバーの XCF メンバー名が必要です。

### このタスクについて

OTMANM パラメーターを使用して、OTMA 用の XCF メンバー名を指定できます。OTMANM パラメーターを指定しない場合、OTMA は IMS システムの APPLID を OTMA XCF メンバー名として使用します。

拡張回復機能 (XRF) のインストール済み環境では、OTMANM パラメーターは使用されません。代わりに、IMS は USERVAR 名を XCF メンバー名として使用します。USERVAR 名は、IMS 制御領域 JCL、DFSPBxxx PROCLIB メンバー、または DFSHSBxx PROCLIB メンバーで指定します。

**推奨事項:** IMS 緊急時再始動中および正常再始動中に、グループ名および IMS メンバー名を変更しないでください。

## 関連資料

[IMS プロシーチャーのパラメーターの説明 \(システム定義\)](#)

## OTMA 開始時の定義

IMS が開始された時に OTMA を開始するように構成するか、IMS の稼働後に OTMA を開始することができます。

### 始める前に

**前提条件:** OTMA を開始する前に、最初に GRNAME 実行パラメーターを定義する必要があります。GRNAME パラメーターは、OTMA が通信用に使用する XCF グループを識別します。

### 手順

- IMS の始動中に OTMA を開始するには、DFSPBxxx PROCLIB メンバーまたは IMS 制御領域 JCL で OTMA=Y を指定します。
- IMS の稼働後に OTMA を開始するには、OTMA=N または OTMA=M を指定するか、OTMA パラメーターを完全に省略します。IMS が稼働した後で、**/START OTMA** コマンドを発行します。

**注:** **/START OTMA** コマンドは、リカバリー可能です。つまり、**/START OTMA** コマンドによって OTMA が開始された場合、OTMA=N が指定されたり、あるいは OTMA パラメーターが指定されていない場合であっても、ウォーム再始動および緊急時再始動中に IMS は OTMA=N 指定を無視して OTMA を再始動します。

- /START OTMA** コマンドが発行された後のウォーム再始動または緊急時再始動中に IMS が OTMA を再始動するのを防止するには、OTMA=M を指定します。OTMA=M が指定されている場合、IMS はウォーム再始動または緊急時再始動中に **/START OTMA** コマンドをリカバリーしません。

### 関連資料

[IMS プロシージャのパラメーターの説明 \(システム定義\)](#)

## OTMA セキュリティー検査のレベルの定義

RACF などのセキュリティー製品を使用する場合、OTMASE パラメーターを使用して、さまざまなセキュリティー検査レベルを指定できます。

### このタスクについて

OTMASE パラメーターで指定された値は、IMS の稼働中に **/SECURE OTMA** コマンドを発行することによりオーバーライドすることができます。

セキュリティー検査のレベルには、以下のものがあります。

#### CHECK (C)

IMS コマンドは、CIMS クラスに対して検査されます。IMS トランザクションは、TIMS クラスに対して検査されます。

#### FULL (F)

「検査」と同じセキュリティーのタイプですが、従属領域に対して追加検査が行われます。F は、OTMASE パラメーターのデフォルト値です。

#### JOIN (J)

OTMA クライアント・ビッド要求のみが、RACF Facility クラス IMSXCF.xcfgroup.member プロファイル (存在する場合) を使用して検査されます。IMS トランザクションおよびコマンドについて、RACF への呼び出しは行われません。

#### NONE (N)

OTMA RACF セキュリティーは、「なし」です。RACF への呼び出しは行いません。

#### PROFILE (P)

各 OTMA メッセージは、行われるセキュリティー検査のレベルを定義します。

### 関連資料

[IMS プロシージャのパラメーターの説明 \(システム定義\)](#)

[/SECURE コマンド \(コマンド\)](#)

## 複数のアクティブ RESUME TPIPE 要求の並列処理に対応する OTMA T パイプのサポート

OTMA クライアント記述子で MULTIRTP=Y を指定すると、OTMA クライアントに関連付けられた OTMA T パイプは複数のアクティブな RESUME TPIPE 要求を並列にサポートできます (MULTIRTP の指定がクライアントによってオーバーライドされない限り)。

デフォルトでは、OTMA は単一のアクティブ RESUME TPIPE 要求のみをサポートする TPIPE を作成し、アクティブ RESUME TPIPE 要求が終了するまで、その他の RESUME TPIPE 要求をキューに入れます。単一のアクティブ RESUME TPIPE 要求のみをサポートすることによって、OTMA からの出力メッセージの処理順序をより高度に制御できます。

複数のアクティブな RESUME TPIPE 要求の並列処理のサポートを有効にすると、出力メッセージに対する OTMA T パイプのスループットを大幅に高めることができ (特に同期または非同期のコールアウト要求の場合)、OTMA T パイプのフェイルオーバー保護を大幅に改善できます。OTMA は、出力メッセージを作成される順に IMS アプリケーション・プログラムから送信しますが、ネットワーク接続と IMS Connect クライアント・アプリケーション・プログラムの両方のパフォーマンスの差があるため、出力の確認と処理が行われる順序が予測不能になります。

OTMA T パイプが複数のアクティブ RESUME TPIPE 要求をサポートする場合、OTMA クライアントは、T パイプが並列で処理する複数のアクティブ RESUME TPIPE 要求を使用して、T パイプから出力メッセージをプルすることができます。いずれか 1 つの RESUME TPIPE 要求の処理で障害が起きた場合でも、OTMA は他のアクティブな RESUME TPIPE 要求を通じて T パイプ上で出力メッセージの配信を続行します。これにより、RESUME TPIPE 要求または T パイプ自体が IMS からの出力メッセージのボトルネックになることが防止されます。

複数のアクティブな RESUME TPIPE 要求に対する OTMA T パイプ・サポートは、アクティブな OTMA クライアントが異常終了してもバックアップ OTMA クライアントへの切り替えが必要でなくなるため、OTMA T パイプのフェイルオーバー保護も向上させます。OTMA T パイプが複数のクライアントからの複数のアクティブな RESUME TPIPE 要求をサポートしている場合、いずれか 1 つの OTMA クライアントが接続に失敗するか接続を失っても、他のクライアントは T パイプからの出力の処理を続行でき、処理が中断されることはありません。

複数のアクティブな RESUME TPIPE 要求に対するサポートは、複数の IMS アプリケーション・プログラムからの出力を、OTMA T パイプを通じて同じ最終的な宛先へ経路指定する複雑さやコストを減らすこともできます。RESUME TPIPE 要求の並列処理がサポートされていない場合、最適なパフォーマンスを達成し、潜在的なボトルネックを回避するためには、IMS アプリケーション・プログラムからの出力を複数の OTMA 宛先記述子または OTMA T パイプを通じて経路指定するのが一般的です。しかし、通常そのような構成を実装するには、それぞれの IMS アプリケーション・プログラム内、複数の OTMA 宛先記述子内、および OTMA クライアント内で、何らかの組み合わせの固有なコーディングが必要になります。複数のアクティブな RESUME TPIPE 要求がサポートされている場合は、複数のアプリケーション・プログラムからの出力を単一の OTMA 宛先記述子と単一の OTMA T パイプを通じて経路指定することにより、同様なパフォーマンス上の利点を実現できます。その場合、複数の OTMA クライアントは、代替クライアント ID として指定された OTMA T パイプ名を使用して同じ RESUME TPIPE 要求を発行することにより、出力を取り出すことができます。

診断目的では、複数のアクティブな RESUME TPIPE 要求のサポートが使用可能である場合、RESUME TPIPE トークンを使用して、各 RESUME TPIPE 要求を発行元のクライアントに関連付けることができます。RESUME TPIPE トークンと、配信されなかった出力が再経路指定される先の T パイプの ID は、OTMA と IMS Connect をサポートしている既存のコマンドを発行することによって表示できます。

### 関連概念

[796 ページの『OTMA クライアント記述子』](#)

OTMA クライアント記述子を使用して、特定の OTMA クライアントに関する情報を IMS に提供します。OTMA クライアント記述子項目は、DFSYDTx PROCLIB メンバー内で、記述子項目の列 1 に入っている M によって識別されます。

### 関連資料

[OTMA クライアント記述子の構文およびパラメーター \(システム定義\)](#)

## RESUME TPIPE 要求の並列処理の使用可能化

RESUME TPIPE 要求の並列処理を行うための OTMA T パイプのサポートは、数種類の方法で使用可能にすることができます。

### このタスクについて

複数の RESUME TPIPE 要求の並列処理に対するサポートは、IMS で、または RESUME TPIPE 要求を使用する OTMA クライアントによって使用可能に設定できます。

### 手順

- RESUME TPIPE 要求の並列処理に対する OTMA T パイプ・サポートの IMS システム・デフォルトを設定するには、DFSYDTx PROCLIB メンバー内で、OTMA システム・クライアント記述子 DFSOTMA に MULTIRTP=Y を指定します。
- 単一の OTMA クライアントに対するサポートを指定するには、その OTMA クライアントの OTMA クライアント記述子内で MULTIRTP=Y を指定します。特定のクライアントに対する MULTIRTP の指定は、IMS システム・デフォルトをオーバーライドします。
- OTMA クライアントは、そのクライアント・ビット要求の中で MULTIRTP サポートを使用可能にすることができます。そのためには、クライアント・ビット要求の OTMA ヘッダーのクライアント・ビット・セクションに対する状態データ内で、バイト・オフセット 64 にある TMAMHFG3 フィールドに X'80' を指定します。OTMA クライアントによるクライアント・ビット要求内での指定は、IMS の OTMA クライアント記述子による指定をオーバーライドします。

## OTMA スーパーメンバーと並列 RESUME TPIPE のサポート

IMS Connect などの OTMA クライアントによって MULTIRTP=Y を指定すると、OTMA スーパーメンバーは複数のアクティブな RESUME TPIPE 要求をサポートできます。

スーパーメンバー・グループ内で MULTIRTP サポートを使用可能にすると、通常操作時のスーパーメンバー・グループのスループットと、グループの 1 つのメンバーが何らかの理由で停止した場合のスーパーメンバー・グループのフェイルオーバー・サポートの両方を向上させることができます。

MULTIRTP サポートが使用可能の場合、スーパーメンバー T パイプ保留キュー上のメッセージは、前に送信されたメッセージに対する確認応答を待つことなく、アクティブな RESUME TPIPE 要求を持つ任意の使用可能な接続上で即時に送信されます。スーパーメンバーから受信した出力が発生順に処理されることを OTMA クライアントが必要とする場合は、MULTIRTP サポートを使用可能にしないでください。

MULTIRTP サポートが使用可能でない場合は、一度に 1 つの RESUME TPIPE だけがアクティブになり、メッセージは、前のメッセージに対する確認応答が受信されるまで送信されません。

同じ IMS システム内の同じスーパーメンバーに接続したすべての OTMA クライアントは、並列 RESUME TPIPE 要求のサポートに同じ MULTIRTP 設定を使用する必要があります。

そのスーパーメンバーに最初に接続したクライアントは、そのスーパーメンバー・グループの MULTIRTP 設定を定義します。最初のクライアントの後にそのスーパーメンバーに接続するクライアントが、異なる MULTIRTP 値を持つ場合、クライアント・ビット要求は NAK およびセンス・コード X'0037' によって拒否されます。

### 関連タスク

875 ページの『非同期コミット後送信出力の共用: OTMA スーパーメンバー 機能』

保留キュー対応 OTMA クライアント (IMS Connect など) は、OTMA のスーパーメンバー 機能を有効にすることにより、非同期コミット後送信 (CM0) 出力メッセージを共用できます。OTMA スーパーメンバー 機能は、z/OS Sysplex Distributor 環境で IMS Connect の複数のインスタンスをサポートするために、特に設計されたものです。

## OTMA スーパーメンバーの MULTIRTP サポートの使用可能化または変更

OTMA スーパーメンバー T パイプ・キューによる並列アクティブ RESUME TPIPE 要求 (MULTIRTP) のサポートは、スーパーメンバー・グループに最初に接続したクライアントのクライアント・ビッド要求の MULTIRTP 値によって決定されます。

### このタスクについて

既存のスーパーメンバー・グループの MULTIRTP 値を変更するには、すべてのクライアント接続を終了し、意図する MULTIRTP 値でクライアントを構成した後に、クライアント接続を再確立する必要があります。

### 手順

1. 既存のスーパーメンバー・グループの MULTIRTP 値を変更する場合は、そのスーパーメンバー・グループへのすべてのデータ・ストア接続を終了します。  
IMS Connect からの接続の場合は、以下のいずれかのコマンドを指定してデータ・ストア接続を停止することができます。
  - タイプ 2 **UPDATE IMSCON** TYPE(DATASTORE) NAME(*datastore\_name*) STOP(COMM)
  - WTOR STOPDS *datastore\_name*
  - z/OS UPDATE DATASTORE NAME(*datastore\_name*) STOP(COMM)
2. 必要な MULTIRTP 値を使用して、OTMA クライアントを構成します。  
IMS Connect の場合、IMS Connect システム構成の中か、個々のデータ・ストア接続の定義の中で MULTIRTP パラメーターを指定できます。
3. スーパーメンバーへの OTMA クライアント接続を開始します。  
IMS Connect からの接続の場合は、以下のいずれかのコマンドを指定してデータ・ストア接続を開始することができます。
  - タイプ 2 **UPDATE IMSCON** TYPE(DATASTORE) NAME(*datastore\_name*) START(COMM)
  - WTOR STARTDS *datastore\_name*
  - z/OS UPDATE DATASTORE NAME(*datastore\_name*) START(COMM)

### タスクの結果

データ・ストア接続が開始されると、OTMA スーパーメンバーは最初に接続するクライアントの MULTIRTP 値を採用し、OTMA クライアントからの RESUME TPIPE 要求を新しい MULTIRTP 値に従って処理します。

スーパーメンバー・グループ内で確立された MULTIRTP 値と異なる MULTIRTP 値を使用してスーパーメンバー・グループに接続を試みるすべてのクライアントは、NAK メッセージとセンス・コード X'0037' によって拒否されます。

## IBM MQ 用の同期化 T パイプの指定

IBM MQ などの OTMA クライアントでは、ALTPCB 出力用の OTMA T パイプが同期化されている必要がある場合があります。

### このタスクについて

同期化された T パイプを指定するには、2つの方法があります。1つの方法は、OTMA ユーザー・データ・フォーマット設定出口ルーチン (DFSYDRUO) で出力フラグを設定することです。もう1つの方法は、OTMASP=Y を指定することです。DFSYDRUO 出口ルーチンをコーディングする唯一の理由が同期化出力フラグを設定することである場合は、代わりに OTMASP パラメーターを使用できます。

デフォルトでは、OTMA は OTMA 出力用に非同期化 T パイプを作成します。

ユーザーの組織で、OTMA 出力用に作成される T パイプを制御するために DFSYDRUO 出口ルーチンと OTMASP パラメーターの両方を使用する場合、どのような場合に同期化 T パイプが作成されるかを以下の表に示します。



表 143. OTMASP パラメーターおよび DFYDRUO 出口の両方を使用したときに作成される T パイプ

DFSYDRUO の設定	OTMASP=Y の場合の結果	OTMASP=N の場合の結果
同期化 T パイプの作成	同期化 T パイプ	同期化 T パイプ
非同期化 T パイプの作成	同期化 T パイプ	非同期化 T パイプ

#### 関連資料

IMS プロシージャのパラメーターの説明 (システム定義)

OTMA 宛先解決ユーザー出口 (DFSYPXO およびその他の OTMAYPRX タイプの出口) (出口ルーチン)

## OTMA クライアントの OTMAYPRX メンバー名のオーバーライドの使用可能化

OTMA 宛先解決 ユーザー出口 (OTMAYPRX) のパラメーター・リストには、メンバー・オーバーライド・フィールドが含まれており、これを使用すると、非 OTMA LTERM から受け取ったトランザクションを OTMA クライアントに経路指定することができます。

### このタスクについて

デフォルトでは、OTMA クライアントから受け取ったトランザクションを経路指定するためにメンバー・オーバーライド・フィールドを使用することはできません。

### 手順

- OTMA クライアントと非 OTMA LTERM の両方から受け取ったトランザクション用に OTMAYPRX パラメーター・リスト内のメンバー・オーバーライド・フィールドを使用するには、OTMAMD=Y を指定します。

## プログラム間通信出力メッセージの非同期送達の設定

複数のプログラム間通信を開始する送信後コミット (CM1) メッセージを使用している場合、適切な出力のみが同期 CM1 モードで OTMA クライアントに返されるようにするには、OTMAASY=Y を指定します。

### このタスクについて

OTMAASY=Y を指定すると、入力トランザクションが複数のプログラム間通信をトリガーした場合、応答トランザクションのみが OTMA によって同期でスケジュールされます。

プログラム間通信から発生した非応答トランザクションは、予期された同期出力より前にクライアントに返されるかどうかに関係なく、非同期でスケジュールされます。IMS は、非同期でスケジュールされるトランザクションに対しては、DFS2082 メッセージを発行しません。

トランザクションが非同期でスケジュールされる場合、IMS は DFS2082 メッセージを発行しません。

OTMAASY=N を指定するか、OTMAASY パラメーターを省略すると、予期された同期出力より前にプログラム間通信からの出力が返されると、エラーが発生することがあります。

OTMAASY=S を指定すると、入力トランザクションが複数のプログラム間通信をトリガーした場合、ISRT を介して ALTPCB (非高速の場合) に対して実行されたプログラム間通信で開始された最初のトランザクションだけが OTMA によって同期的にスケジュールされます。

#### 関連資料

IMS プロシージャのパラメーターの説明 (システム定義)

## OTMA 記述子

OTMA は、2 種類の記述子 (OTMA クライアント記述子および OTMA 宛先記述子) を備えます。

OTMA クライアント記述子は、OTMA クライアントの特定の特性を指定します。例えば、DFSYDRUO 出口ルーチン、送信後コミット・メッセージ・タイムアウト値、メッセージあふれしきい値などです。

OTMA 宛先記述子は、以下の宛先属性を定義します。

- 別の IMS Connect インスタンスとの TCP/IP 接続を経由してリモート IMS システムに送達するために IMS Connect に送信される OTMA ALTPCB メッセージ。
- IMS Connect または非 OTMA の宛先 (例えば、SNA 端末またはプリンターなど) へ送信される OTMA ALTPCB 出力。
- IMS Connect へ送信され、以下のいずれかを最終的な宛先とする IMS アプリケーション・プログラムからのコールアウト要求。
  - Web アプリケーション・サーバー (WebSphere Application Server など) で実行される Enterprise Java Bean (EJB) またはメッセージ駆動型 Bean (MDB)
  - Web サービス
  - ユーザー作成の IMS Connect クライアント

OTMA クライアント記述子と OTMA 宛先記述子は、DFSYDTx PROCLIB メンバー内の項目として指定されます。

OTMA 宛先記述子は、以下のタイプ 2 IMS コマンドを使用して、オンラインで作成および更新することもできます。

- CREATE OTMADESC
- DELETE OTMADESC
- QUERY OTMADESC
- UPDATE OTMADESC

#### 関連資料

[IMS PROCLIB データ・セットの DFSYDTx メンバー \(システム定義\)](#)

## OTMA クライアント記述子

OTMA クライアント記述子を使用して、特定の OTMA クライアントに関する情報を IMS に提供します。OTMA クライアント記述子項目は、DFSYDTx PROCLIB メンバー内で、記述子項目の列 1 に入っている M によって識別されます。

OTMA クライアント記述子内の情報は、IMS で OTMA クライアントからのメッセージを管理するのに役立ちます。OTMA クライアント記述子に組み込まれている情報は、OTMA クライアント・ビッド要求内、または場合によっては /START OTMA コマンド内でも指定できます。これらの代替方式で提供された情報は、OTMA クライアント記述子内で提供された情報をオーバーライドする場合があります。

OTMA クライアント記述子を使用して、OTMA および IMS に対し、OTMA クライアントの以下の属性を指定できます。

- OTMA クライアントが使用する OTMA 宛先解決出口ルーチンの名前。
- synclevel=confirm または synclevel=syncpt を使用するコミット後送信メッセージおよび送信後コミット・メッセージの両方の場合に、OTMA で OTMA クライアントからの ACK または NAK 応答を待機するときに使用するタイムアウト値。
- メッセージあふれ状態がトリガーされるまでに IMS システムで処理できる OTMA クライアントからの入力メッセージの最大数。

OTMA クライアント記述子はオプションです。しかし、OTMA クライアントが実際に OTMA に接続する前に、出力を OTMA クライアントに経路指定するのに特に便利な場合があります。OTMA クライアントが最終的に接続した時点で、クライアント・ビッド要求内の指定は、OTMA クライアント記述子内で行われた指定をオーバーライドします。

以下のデフォルトは、OTMA クライアント記述子のパラメーターを指定しなかった場合に適用されます。

- IMS は、クライアント・ビッド要求が異なる出口ルーチン名を指定している場合以外は、デフォルトの OTMA ユーザー・データ・フォーマット設定出口ルーチン (DFSYDRU0) を使用します。



- 確認応答のタイムアウト値には、IMS はデフォルトの 2 分を使用します。ただし、クライアント・ビット要求または **/START TMEMBER** コマンドの TIMEOUT パラメーターで異なるタイムアウト値が指定されている場合は除きます。
- IMS は、メッセージあふれ検出にデフォルトのしきい値の 5000 メッセージを使用します。ただし、クライアント・ビット要求または **/START TMEMBER** コマンドの INPUT パラメーターで異なる値が指定されている場合は除きます。

OTMA クライアント記述子は、IMS の初期設定時に構築されます。記述子は、IMS.PROCLIB の DFSYDTx メンバーに含まれています。DFSYDTx の「x」は、IMS 中核接尾部です。同じ IMS システムに複数のクライアントがある場合は、各クライアント名を分離した行にリストします。桁 3 から 18 は、クライアントごとに違うことを確認してください。

デフォルトでは、最大 254 個のクライアント記述子を定義できます。DFSOTMA 記述子を使用しない場合は、最大 255 個のクライアント記述子を指定できます。

DFSOTMA 記述子で MDESCMAX パラメーターを指定することで、クライアント記述子の最大数を変更できます。

### 関連資料

[IMS PROCLIB データ・セットの DFSYDTx メンバー \(システム定義\)](#)

## OTMA 宛先記述子

OTMA 宛先記述子は、OTMA を介して経路指定されるメッセージの宛先を定義するために使用します。

OTMA 宛先記述子は、OTMA 宛先解決ユーザー出口 (OTMAYPRX) および OTMA ユーザー・データ・フォーマット設定出口ルーチン (DFSYDRUO) を使用するよりも簡単な宛先記述方法を提供します。

OTMA 宛先タイプには以下のものがあります。

#### TYPE=IMSCON

IMS Connect の宛先

#### TYPE=IMSTRAN

同期プログラム間通信を使用する他の IMS アプリケーション・プログラム

#### TYPE=MQSERIES

IBM MQ

#### TYPE=NONOTMA

OTMA 以外の宛先

宛先を OTMA 宛先記述子によって記述した場合、IMS は、DFSYPX0 ユーザー出口および DFSYDRUO 出口ルーチン呼び出しません。この記述子の動作は、EXIT=YES を指定することにより、TYPE=IMSCON、TYPE=NONOTMA、または TYPE=MQSERIES によってオーバーライドすることができます。その場合は、メッセージを処理するために出口が呼び出され、記述子のルーティング情報を変更する必要があるかどうか判断されます。

## IMS Connect 記述子

OTMA 宛先記述子は、IMS Connect を介した IMS アプリケーション・プログラムから外部データまたはサービス・プロバイダーへのコールアウト要求のルーティングをサポートします。IMS Connect は、以下のいずれかの IMS Connect クライアントを介して、コールアウト要求を経路指定します。

- IMS TM Resource Adapter
- IMS Enterprise Suite SOAP Gateway
- ユーザー作成の IMS Connect クライアント

以下のタイプのメッセージの場合はすべて、宛先タイプを IMS Connect (TYPE=IMSCON) として定義します。

- IMS Connect クライアントの出力
- IMS アプリケーション・プログラムからのコールアウト要求

- IMS 間 TCP/IP 通信を介して処理のために別の IMS システムに経路指定されるトランザクション・メッセージ

SNA 端末やプリンターなどの宛先にメッセージを経路指定する場合は、宛先タイプを非 OTMA (TYPE=NONOTMA) として定義します。

記述子に指定できる宛先属性は、宛先のタイプによって異なります。

IMS Connect 宛先の場合は、以下の宛先属性を指定できます。

- OTMA T メンバー名
- OTMA スーパーメンバー名
- T パイプ名
- IMS 間 TCP/IP 接続を介してリモート IMS システム宛てに送信されるトランザクション・メッセージの場合は、以下を指定できます。
  - ローカル IMS Connect インスタンスの RMTIMSCON 構成ステートメントで定義された、リモート IMS Connect への接続の名前
  - リモート IMS Connect インスタンスの DATASTORE 構成ステートメントで定義された、リモート IMS システムの IMS ID。
  - オプションで、リモート IMS システムが ALTPCB メッセージを処理するためにスケジュールするトランザクション・コード。
  - オプションで、リモート IMS システムが ALTPCB メッセージのトランザクション許可を実行するために使用するユーザー ID。
- SOAP Gateway 用の IMS Connect XML 変換サポートの場合、XML アダプター名および XML コンバーター名。
- DL/I ICAL 呼び出しを発行する IMS アプリケーション・プログラムによって行われる同期コールアウト要求のタイムアウト間隔。指定された時間内に応答を受信しないと、要求はタイムアウトになり、IMS 従属領域が解放されます。OTMA 宛先記述子で指定されたタイムアウト値が、IMS アプリケーションの DL/I ICAL 呼び出しで指定されたタイムアウト値と異なる場合、OTMA は 2 つの値のうち、小さい方の値を使用します。

IMS Connect 宛先の場合、OTMA 宛先記述子はアプリケーション・プログラムと T パイプの間で、抽象化の層を提供します。この抽象化の層は、IMS Connect 出力の経路指定の際に柔軟性を提供します。例えば、記述子内で指定する宛先名は、明示的なものでも、マスク文字を使用して汎用化されたものでもかまいません。また、記述子内で指定する T パイプ名は、個別の T パイプ名でもスーパーメンバー名でもよく、ブランクのままにしてもかまいません。ブランクの場合は、指定した宛先名が T パイプ名として使用されます。

## IBM MQ 記述子

OTMA は、TYPE=MQSERIES 記述子および以下の情報を使用して、非同期コールアウト要求を IBM MQ に経路指定することができます。

- OTMA T メンバー名
- OTMA スーパーメンバー名
- T パイプ名
- IBM MQ アプリケーションの MQMD\_REPLYTOQ 値および MQMD\_REPLYTOQMGR 値
- オプションで、OTMA 経路指定出口 (DFSYPRX0 および DFSYDRU0) を呼び出すためのオーバーライド
- オプションで、MQ メッセージ記述子構造のその他の IBM MQ 値

## IMS 同期プログラム間通信記述子

TYPE=IMSTRAN を指定した記述子を作成することにより、メッセージを直接、別の IMS アプリケーション・プログラムに経路指定することができます。オプションで、記述子には次のようなその他の属性を指定できます。

- 切り替えられたアプリケーションからの遅延応答メッセージの経路指定先である OTMA T メンバーおよび T パイプ
- 遅延応答メッセージに対して IMS ユーザー出口 (DFSCMUX0) を呼び出すためのオーバーライド
- ターゲット・アプリケーションが IOPCB に応答するかどうかを判別するために IMS が実行できる検査

## 関連資料

[IMS PROCLIB データ・セットの DFSYDTx メンバー \(システム定義\)](#)

## OTMA 宛先記述子の定義

OTMA 宛先記述子を作成、削除、および更新するには、オンライン・タイプ 2 IMS コマンドを使用するか、または IMS.PROCLIB データ・セットの DFSYDTx メンバーに直接コーディングします。

デフォルトでは、最大 510 個の宛先記述子を定義できます。DFSOTMA 記述子に DDESCMAX パラメーターを指定することにより、最大数を変更できます。

OTMA 宛先記述子を管理するためのタイプ 2 コマンドを以下に示します。

- **CREATE OTMADESC**
- **DELETE OTMADESC**
- **QUERY OTMADESC**
- **UPDATE OTMADESC**

上記のいずれかのタイプ 2 コマンドを使用して OTMA 宛先記述子を変更すると、その変更は即座に有効になり、IMS システム・ログに x'221B' ログ・レコードとして、および IMS チェックポイント・ログに x'4035' ログ・レコードとして記録されます。タイプ 2 コマンドで行われた変更は、DFSYDTx PROCLIB メンバーには保管されず、IMS のウォーム・スタート後および緊急時再始動後のみ保持されます。タイプ 2 コマンドで行われた変更は、IMS のコールド・スタート後は保持されません。

タイプ 2 コマンドを使用して OTMA 宛先記述子を作成、変更、または削除した場合、その変更は、次のコールド・スタートまで、DFSYDTx 内の同じ記述子の既存の定義をオーバーライドします。

DFSYDTx メンバー内で行われた変更を有効にするには、IMS をコールド・スタートする必要があります。IMS は、コールド・スタート中にのみ DFSYDTx メンバーを読み取るからです。

DFSYDTx メンバーには OTMA クライアント記述子も保管されますが、OTMA 宛先記述子は記述子項目の 1 桁目の「D」により識別できます。

タイプ 2 コマンドを使用して DFSYDTx メンバーにコーディングされている OTMA 宛先記述子を更新および削除できますが、その変更内容は DFSYDTx メンバーには反映されません。変更または削除した記述子はログに記録され、ログ・レコードが DFSYDTx メンバーに保管されている記述子のステートメントをオーバーライドします。

**QUERY OTMADESC** コマンドを使用すると、現在ログに記録されている OTMA 宛先記述子に関する情報が返されます。

## 関連資料

[CREATE OTMADESC コマンド \(コマンド\)](#)

[DELETE OTMADESC コマンド \(コマンド\)](#)

[QUERY OTMADESC コマンド \(コマンド\)](#)

[UPDATE OTMADESC コマンド \(コマンド\)](#)

[OTMA 宛先記述子の構文およびパラメーター \(システム定義\)](#)

## OTMA 宛先記述子の宛先名のマスキング

OTMA 宛先記述子で宛先名を指定する場合、アスタリスクをワイルドカード文字として使用するか、宛先名フィールドで最後の文字のマスクとして使用して、宛先名を汎用化することができます。

例えば、OTICON\* を宛先名として指定した場合、この記述子は OTICON01 や OTICON02 など、OTICON で始まる宛先名を指定したすべての出力に適用されます。

OTMA 宛先記述子は、DFSYDTx PROCLIB メンバーに任意の順序でリストできます。ただし、IMS が出力を経路指定するために記述子を検索する場合、IMS は最も具体的に指定された宛先名から最も汎用的な宛先名への順序で記述子を検索します。つまり、IMS は常に固有の宛先名の後で、アスタリスクが指定された宛先名を読み取ります。

記述子で、OTICON\* など、マスクした宛先名を指定し、T パイプ名を指定していない場合は、出力で実際の T パイプの完全な名前を指定する必要があります。デフォルトでは、記述子で T パイプ名が指定されていない場合、宛先名が T パイプ名として使用されます。宛先名がマスクされている場合、IMS は出力によって指定された宛先名を読み取って、使用する特定の T パイプを判別する必要があります。

マスクした宛先名を使用している記述子で T パイプ名を指定した場合、マスクした宛先名に一致する宛先へ送信されるすべての出力は、同じ T パイプへ経路指定されます。

## DFSOTMA 記述子

オプションの DFSOTMA 記述子を使用して、すべての OTMA クライアントに適用されるサポートのグローバル属性、デフォルト属性、制限値、およびタイプを定義します。

DFSOTMA 記述子は OTMA タイプ M の記述子であり、この イムス システムに接続するすべての OTMA クライアントのシステム・デフォルトおよびグローバル・パラメーターを設定します。

DFSOTMA 記述子は、OTMA クライアント記述子で指定できるすべてのパラメーターをサポートしていません。サポートされていないパラメーター (DRU= および T/O= など) は、DFSOTMA クライアント記述子で指定されている場合は無視されます。

すべての OTMA クライアントに指定できるグローバル値には、以下のものがあります。

- メッセージあふれ防止
- すべての OTMA メンバーまたはクライアントのグローバル tpipe 制限
- 複数のアクティブな RESUME TPIPE 要求のサポート
- 保管域接頭部割り振りの限界
- OTMA 記述子の最大数
- キャッシュ・アクセサー環境エレメント (ACEE) がサブプール 249 に保管されていることができる ラクトール ユーザー ID の最大数。
- 入力 LTERM オーバーライド名の イブン ワークロード・マネージャーへの引き渡し

DFSOTMA 記述子はタイプ M 記述子であるため、DFSOTMA 記述子を使用される場合、定義できる OTMA クライアント記述子の最大数は 1 つ少なくなっています。これは、OTMA クライアント記述子もタイプ M 記述子であるためです。

DFSOTMA 記述子は、イムス 再始動タイプに関係なく、常に DFSYDTx PROCLIB メンバーからロードされます。DFSOTMA 記述子のチェックポイント・ログ・レコードはありません。

### 関連タスク

818 ページの『[OTMA ACEE フラッディング制御](#)』

OTMA アクセス機能環境エレメント (ACEE) フラッディング制御機能は、IMS 制御領域の仮想記憶域が使い尽くされるのを防止します。

### 関連資料

[DFSOTMA 記述子の構文およびパラメーター \(システム定義\)](#)

## OTMA 記述子の制限の変更

デフォルトでは、IMS は定義可能な OTMA 記述子の数を、タイプ M の記述子の場合は 255 個、タイプ D の記述子の場合は 510 個に制限します。これらの制限は変更できます。

### このタスクについて

OTMA 記述子は、拡張共通ストレージ域 (ECSA) に格納されます。定義できる OTMA 記述子の数を制限すると、記述子が過大なストレージを使用しないようにすることができます。

通常、OTMA がどの時点で OTMA 記述子に使用するストレージの量も、定義された最大限度より少なくなります。OTMA は、現在定義されている OTMA 記述子に十分なだけの ECSA ストレージしか割り振らないからです。

定義できる OTMA 記述子の最大数を変更するには、以下のようになります。

## 手順

1. DFSYDTx メンバーが既に DFSOTMA 記述子を含んでいる場合は、その DFSOTMA 記述子が IMS PROCLIB データ・セットの DFSYDTx メンバー内で最初にリストされている 255 個のタイプ M 記述子の 1 つであることを確認してください。  
**推奨事項:** DFSOTMA 記述子は、見つけやすいように、また、255 番目のタイプ M 記述子より後にリストされる危険がないように、DFSYDTx メンバー内の最初の項目としてコーディングしてください。
2. DFSYDTx メンバーに DFSOTMA 記述子が含まれていない場合は、DFSOTMA 記述子を作成してください。これは、IMS PROCLIB データ・セットの DFSYDTx メンバー内で最初にリストされている 255 個のタイプ M 記述子の 1 つでなければなりません。  
**推奨事項:** DFSOTMA 記述子は、見つけやすいように、また、255 番目のタイプ M 記述子より後にリストされる危険がないように、DFSYDTx メンバー内の最初の項目としてコーディングしてください。
3. DFSOTMA 記述子内で、タイプ D またはタイプ M の記述子の新しい最大数を指定します。
  - タイプ D 記述子 (宛先記述子) の場合は、DDESCMAX パラメーターで新しい最大数を指定します。
  - タイプ M 記述子 (クライアント記述子および DFSOTMA 記述子) の場合は、MDESCMAX パラメーターで新しい最大数を指定します。
4. IMS を再始動して、新規または更新された DFSOTMA 記述子をオンラインの IMS システムに追加します。

## IMS 間通信の OTMA サポート

IMS 間 TCP/IP 通信を使用して、ローカル IMS システムからリモート IMS システムに OTMA メッセージを送信することができます。

リモート IMS システムにメッセージを送信するには、OTMA 宛先記述子または OTMA ユーザー・データ・フォーマット設定出口ルーチン (DFSYDRUO) で、リモート IMS システムを OTMA 宛先として定義します。これにより、IMS 従属領域で実行されている IMS アプリケーション・プログラムは、指定された OTMA 宛先名を持つ ALTPCB にメッセージを挿入することができます。OTMA は、メッセージを IMS Connect 用の T パイプへのキューに入れ、IMS Connect は、TCP/IP ネットワーク経由でメッセージをリモート IMS システムに送信します。

OTMA クライアント記述子を使用して、ACK タイムアウト間隔を設定できます。タイムアウト間隔は、リモート・システムに送信されるトランザクション・メッセージに対する ACK または NAK の応答を OTMA が待つ時間の長さを決定します。

リモート IMS システム宛ての OTMA メッセージは、片方向にのみ流れます。リモート IMS システムによって生成された応答をローカル IMS システムに返すには、リモート IMS システム内でリターン・パスを別途定義し、応答を別個のトランザクションとして返す必要があります。

IMS Connect のローカル・インスタンスとリモート・インスタンスが、ローカル IMS システムとリモート宛先 IMS システム間の TCP/IP 接続を管理します。2 つの IMS システム間の TCP/IP 接続は、IMS Connect 内で、他の必要な IMS Connect 構成ステートメントに加えて、RMTIMSCON 構成ステートメントを使用して定義されます。

リモート IMS システム上に過剰な数の T パイプが累積するリスクを最小限に抑えるために、OTMA の IMS 間 TCP/IP 接続には持続接続を使用してください。IMS は、アイドル状態の T パイプをクリーンアップします。ただし、それらの T パイプへの出力がキューに入れられている場合を除きます。

リモート IMS Connect インスタンスへの接続を定義する RMTIMSCON ステートメントで PERSISTENT=Y を指定することにより、ローカル IMS Connect インスタンスへの接続の持続性を指定します。

### 関連概念

154 ページの『IMS Connect による IMS 間 TCP/IP 通信のサポート』



IMS Connect は、TCP/IP ネットワークを介して相互に通信する IMS システムの TCP/IP 接続およびプロトコルを管理します。

### 関連タスク

IMS 間の TCP/IP 接続 (システム定義)

### 関連資料

OTMA 宛先記述子の構文およびパラメーター (システム定義)

## IMS 間通信のスーパーメンバー・サポート

OTMA スーパーメンバー機能を使用すると、リモート IMS システムに送信されるメッセージを、IMS Connect の複数のローカル・インスタンス間に分散させることができます。

OTMA は、ラウンドロビン分散アルゴリズムを使用して、IMS 間メッセージをスーパーメンバー・グループ内の各 IMS Connect インスタンスに分散させます。IMS Connect インスタンスがスーパーメンバー・グループに参加または離脱した場合、OTMA は、スーパーメンバーのラウンドロビン・リストを動的に更新します。

OTMA は、IMS 間 TCP/IP 通信に使用されるスーパーメンバー・グループ内で、最大 8 つの T メンバーをサポートします。OTMA は、IMS Connect の DATASTORE 構成ステートメントで定義された各接続を 1 つの T メンバーとして数えます。各 IMS Connect インスタンスが同じスーパーメンバー・グループと確立するデータ・ストア接続の数によっては、8 つ以下の IMS Connect インスタンスで、T メンバーの制限に達することがあります。

8 つを超える T メンバーがスーパーメンバー・グループを使用するように定義されている場合、OTMA は最初の 8 つのみを使用してスーパーメンバー・グループを結合します。

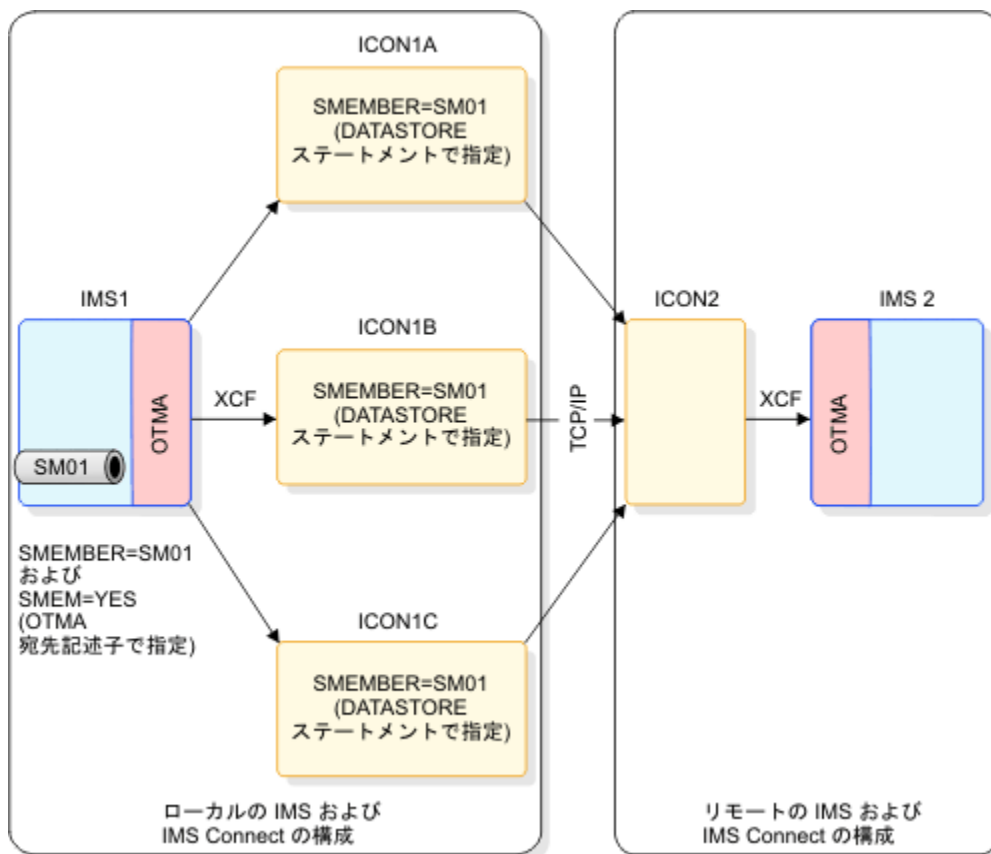


図 137. OTMA IMS 間 TCP/IP 接続でのスーパー・メンバー・グループの使用

### 関連タスク

OTMA IMS 間 TCP/IP 接続に対する IMS Connect スーパーメンバー・グループの定義 (システム定義)

## リモート・トランザクション・コードの指定

リモート IMS システムに送信される OTMA メッセージには、リモート IMS システムでメッセージを処理するトランザクションをスケジュールするためのトランザクション・コードを含めることができます。

### このタスクについて

トランザクション・コードは、送信側アプリケーション・プログラム、OTMA 宛先記述子、またはその両方で指定することができます。

トランザクション・コードがアプリケーション・プログラムで指定される場合、トランザクション・コードは出力メッセージのデータ部分の最初の 8 バイトに含められます。

トランザクション・コードが OTMA 宛先記述子で指定される場合、IMS Connect はメッセージの長さを 8 バイト増やして、トランザクション・コードをメッセージの長さフィールドとデータ・セクションの間に挿入します。トランザクション・コードが 8 バイトより少ない場合、残りの部分は空白で埋め込まれます。

OTMA 宛先記述子によるトランザクション・コードの指定は、送信側アプリケーション・プログラムによるトランザクション・コードの指定から独立しています。アプリケーション・プログラムと OTMA 宛先記述子の両方でトランザクション・コードが指定された場合、両方が出力メッセージに含められ、リモート IMS システムは両方を処理できる必要があります。

例えば、アプリケーションがトランザクション・コードを指定する場合、メッセージのフォーマットは LLZMSGDATA です。ここで、LLZZ セクションにはメッセージの長さが入り、MSGDATA セクションにはトランザクション・コードとメッセージ・データの両方が入ります。OTMA 宛先記述子もトランザクション・コードを指定する場合、メッセージの長さが 8 バイト増えて、メッセージのフォーマットは LLZZTRANCODMSGDATA になります。ここで、TRANCOD には、OTMA 宛先記述子で使用されたトランザクション・コードが入ります。

### 手順

- OTMA 宛先記述子でトランザクション・コードを指定するには、RMTTRAN パラメーターを指定します。

### 関連概念

[797 ページの『OTMA 宛先記述子』](#)

OTMA 宛先記述子は、OTMA を介して経路指定されるメッセージの宛先を定義するために使用します。

### 関連資料

[OTMA 宛先記述子の構文およびパラメーター \(システム定義\)](#)

## リモート IMS システムに送信されるメッセージのフォーマット

ローカル IMS アプリケーションで作成される OTMA リモート ALTPCB 出力メッセージは、IMS トランザクション・メッセージの標準形式に従う必要があります。

次の例は、IMS トランザクション・メッセージの標準形式を示しています。

```
LLZZ | TRANCOD | DATA
```

以下の表は、フォーマットの各部分について説明しています。

表 144. トランザクション・メッセージのフォーマット

フィールド名	フィールド長	説明
LL	2	LL フィールドと ZZ フィールドを含めた、メッセージの長さ。
ZZ	2	2 進ゼロ。



表 144. トランザクション・メッセージのフォーマット (続き)

フィールド名	フィールド長	説明
TRANCODE	8	リモート IMS システムでスケジュールされるトランザクション。OTMA 宛先記述子の RMTTRAN パラメーターでトランザクション・コードが指定されている場合、IMS Connect はトランザクション・コードをここに挿入します。
DATA	可変	リモート IMS システム上の IMS アプリケーション・プログラムによって処理されるデータ。送信側アプリケーション・プログラムがトランザクション・コードを指定している場合、トランザクション・コードはここに (DATA セクションの最初の 8 バイト) に含まれます。

#### 関連資料

[入力メッセージの形式と内容 \(アプリケーション・プログラミング\)](#)

## OTMA サポートの出口ルーチン

IMS で配信される出口ルーチンには OTMA 専用のものがあります。また、OTMA は、IMS で配信される出口ルーチンの中で、OTMA 専用ではない他の複数の出口ルーチンもサポートします。

以下の出口ルーチンは OTMA をサポートしています。

- OTMA ユーザー・データ・フォーマット出口ルーチン (DFSYDRU0)
- OTMA 入出力編集ユーザー出口 (OTMAIOED)
 

**制約事項:** OTMAIOED ユーザー出口は、DL/I ICAL 呼び出しを発行する IMS アプリケーション・プログラムから OTMA によって受信されたものを含め、同期コールアウト要求メッセージではサポートされていません。
- OTMA 宛先解決 ユーザー出口 (OTMAYPRX)
- OTMA RESUME TPIPE セキュリティー ユーザー出口 (OTMARTUX)。この出口ルーチンを使用して、非同期保留キュー上のメッセージを保護することができます。

さらに、以下の出口ルーチンが OTMA によってサポートされます。

- コマンド許可 (DFSCCMD0)
- TM/MSC メッセージ経路指定および制御ユーザー出口 (DFSMSCEO)
- キュー・スペース通知 (DFSQSPC0)
- トランザクション許可 (DFSCTRN0)

#### 関連タスク

838 ページの『非同期保留キュー内のメッセージの保護』

非同期保留キューに入っているメッセージを RESUME TPIPE 呼び出しの無許可使用から保護するためには、RACF、OTMA RESUME TPIPE セキュリティー ユーザー出口 (OTMARTUX)、またはその両方を使用する必要があります。

#### 関連資料

[OTMA ユーザー・データ・フォーマット設定出口ルーチン \(DFSYDRU0\) \(出口ルーチン\)](#)

[OTMA 入出力編集ユーザー出口 \(DFSYIOEO\) およびその他の OTMAIOED タイプの出口 \(出口ルーチン\)](#)

[OTMA 宛先解決ユーザー出口 \(DFSYPRX0\) およびその他の OTMAYPRX タイプの出口 \(出口ルーチン\)](#)

[OTMARTUX: OTMA RESUME TPIPE セキュリティー・ユーザー出口 \(DFSYRTUX\) およびその他の OTMARTUX タイプの出口 \(出口ルーチン\)](#)

## 宛先決定のための OTMAYPRX ユーザー出口および DFSYDRU0 出口ルーチンの使用

トランザクション・パイプ名は、IMS LTERM 名または APPC/IMS TP 名と同じでもかまいません。

OTMA 用の宛先かどうかをはっきりさせるために、IMS は、OTMA 出口ルーチンを提供して、IMS がどこで宛先名を解決すべきかを指定させることができます。

- OTMA 宛先解決ユーザー出口 (OTMAYPRX)
- OTMA ユーザー・データ・フォーマット出口ルーチン (DFSYDRU0)

1つの IMS サブシステム中には、多くの DFSYDRU0 出口ルーチンを持つことができますが、OTMAYPRX ユーザー出口は1つだけです。

出口ルーチンは、実際の宛先名を変更することはできません。

OTMA メッセージ用の宛先を決定するには、2つのフェーズが必要です。各フェーズ中で、OTMA 出口ルーチンは以下のように呼び出されます。

### フェーズ 1

宛先解決ユーザー出口 (OTMAYPRX) が呼び出されて、出力の初期宛先が判別されます。

このユーザー出口は、処理のためにメッセージを OTMA クライアントに送信するか、IMS TM に送信するかを決定することができます。この出口ルーチンは、最終的な宛先を決定することはできません (十分なパラメーターが渡されていないからです)。

### フェーズ 2

DFSYDRU0 出口ルーチンが呼び出されて、出力の最終宛先が判別されます。各クライアントは、別々の DFSYDRU0 出口ルーチンを指定できます。

DFSYDRU0 出口の名前は、ユーザーまたは OTMA クライアントによって決定されます。各クライアントは、それぞれ専用の DFSYDRU0 出口をもつことができます。OTMA クライアントに関連した DFSYDRU0 出口ルーチンの名前を表示するには、/DISPLAY TMEMBER コマンドを発行します。

これらの出口ルーチンは両方とも、IMS アプリケーション・プログラムが ISRT 呼び出しを代替プログラム連絡ブロック (PCB) に対して発行したとき、あるいは CHNG または PURG 呼び出しを発行したときに制御を得ます。しかし、宛先が IMS スケジューラー・メッセージ・ブロック (SMB) の場合は、DFSYDRU0 出口ルーチンは、制御を受け取りません。以下の図は、メッセージ宛先判別の2つのフェーズを示します。

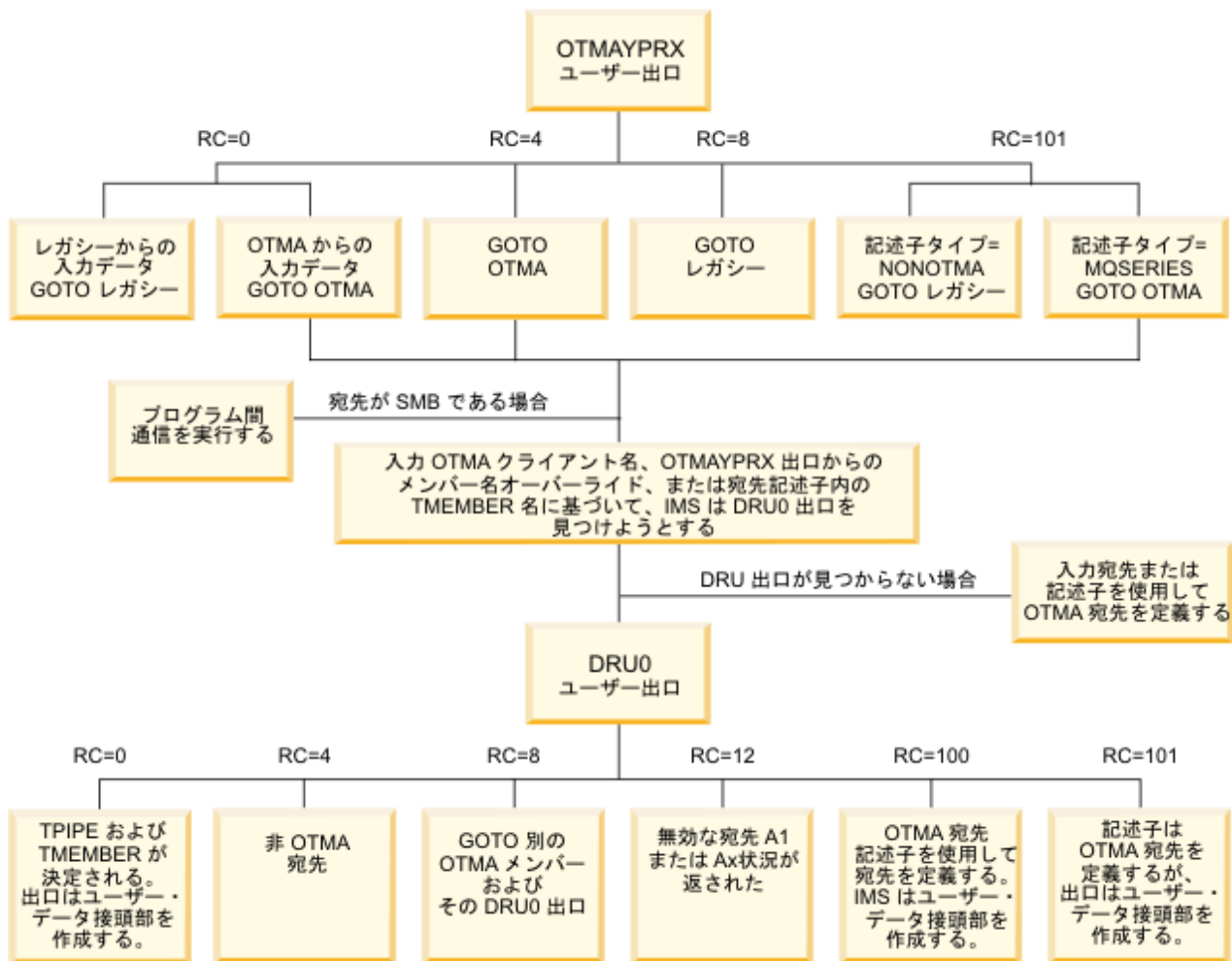


図 138. OTMAYPRX OTMA ユーザー出口および DFSYDRU0 OTMA ユーザー出口のメッセージ宛先の決定方法

#### 推奨事項:

- DFSYDRU0 出口の入力パラメーター・リストでオフセット +8 に指定された宛先名は、元の CHNG 呼び出し、OTMA 宛先記述子、または OTMAYPRX 出口ルーチンによって設定された T パイプ・オーバーライドから提供される可能性があります。DFSYDRU0 出口の入力パラメーター・リストのオフセット +27 で入力フラグ X'40' が指定された場合、元の CHNG 呼び出しからの宛先名は、オフセット +104 の DFSYDRU0 出口ルーチンの入力パラメーター・リストに含まれます。
- SMB 名には潜在的な矛盾があるため、OTMA クライアントは、トランザクション・パイプ名をトランザクション名または経路指定キーとして使用しないでください。
- OTMAYPRX および DFSYDRU0 OTMA ユーザー出口は、共用キュー・グループ内のフロントエンドあるいはバックエンドの IMS システムで同じにしてください。出口ルーチンが、1 つ以上のバックエンド IMS システムで異なる場合は、非同期出力は、どのバックエンド IMS システムが入力を処理するかによって、異なった宛先に送信される可能性があります。

出力のプロンプト・デリバリーを確かめるために、共用キュー・グループ中の各バックエンド IMS システムごとに OTMA を使用可能にします。バックエンド IMS システムに使用可能な OTMA がない場合は、代替 PCB に挿入される非同期 OTMA 出力はどれも、単純にキューされ、オペレーターが /STA OTMA コマンドを発行するまで、送達されません。

- IMS PROCLIB メンバー DFSPBxxx に OTMAMD=Y を指定すると、OTMAYPRX ユーザー出口から異なる DFSYDRU0 出口ルーチンへ転送せずに OTMA メッセージを送信できます。
- IMS PROCLIB メンバー DFSPB xxx に OTMASP=Y を指定すると、ALT-PCB 出力メッセージ用に常に同期 T パイプを作成します。

注: SCD および PST アドレスは、両方の OTMA ユーザー出口の入力パラメーターで使用可能です。出力メッセージの最初のセグメントのアドレスは、どちらのユーザー出口にも受け渡されません。

#### 関連資料

[OTMA ユーザー・データ・フォーマット設定出口ルーチン \(DFSYDRU0\) \(出口ルーチン\)](#)

[OTMA 宛先解決ユーザー出口 \(DFSYPRX0 およびその他の OTMAYPRX タイプの出口\) \(出口ルーチン\)](#)

## OTMA 用に IMS を管理する

以下のトピックでは、OTMA 用の IMS 管理の考慮事項について説明します。

### IMS の会話および OTMA

OTMA と IMS の会話は、送信後コミットであり、リカバリー不能です。

IMS は、固有の会話 ID を、メッセージの OTMA 状態データ・セクション内のサーバー・トークン・フィールドに保管されているトランザクション・パイプごとに作成します。この ID は、トランザクション・パイプを使用している会話用の後続のクライアント入力と一緒に、IMS に渡す必要があります。

IMS がトランザクションを確認するとき、サーバー状態フラグは、メッセージ接頭語の状態データ・セクション内で「会話型状態」にセットされます。このフラグは、会話中、後続のクライアント入力すべてにセットする必要があります。

IMS 会話は、/EXIT コマンドを発行することによって終了できます。別の方法として、クライアントが IMS 会話を終了したい場合には、クライアントはメッセージ接頭語のメッセージ制御情報セクションにコミット確認フラグをセットしたメッセージ (割り振り解除フローを示します) を送信できます。その後 IMS は、IMS 会話を終了させます。割り振り解除フローは、メッセージ接頭語の状態データ・セクション中にサーバー状態フラグとして「会話型状態」を指定しますが、サーバー・トークン・フィールドにもセットする必要があります。

### MSC および OTMA トランザクション

OTMA トランザクションで、IMS 複数システム結合機能 (MSC) を使用することができます。

OTMA 環境での MSC 処理は、以下のように APPC/IMS 環境での MSC 処理に似ています。

- クライアントは、IMS にトランザクションを送信します。トランザクションが、その IMS システムに対してリモートとして定義されている場合は、トランザクションは、処理のためにリモート IMS システムに送信されます。トランザクションが、ローカル・トランザクションとして定義され、他の IMS システムにメッセージ通信を行う場合は、交換メッセージは、処理のためにそのリモート IMS システムに送信されます。
- リモート・アプリケーション・プログラムからの出力は、発信元 IMS に戻されます。
- IMS は、データが OTMA データで、クライアントにデータを送信するのに、トランザクション・パイプを使用することを認識します。

リモート・アプリケーションがリモート宛先である代替 PCB へ挿入すると、データは、OTMA 宛先には経路指定されません。メッセージが接頭部を持っていても、リモート宛先は、出力メッセージを OTMA クライアントに経路指定しません。メッセージが、オリジナル・クライアントに正しく経路指定し直されると、リモート IMS は、リモート・トランザクションに挿入しなければなりません。そのトランザクション (オリジナル IMS 側での) は、その後、代替 PCB と事前経路指定出口ルーチンを使用して、OTMA クライアントにメッセージを送信する必要があります。

MSC リンクが OTMA クライアントに接続されているフロントエンド IMS サブシステムに存在している限り、共用キュー環境中で、OTMA での MSC を使用することができます。

## 高速機能および OTMA トランザクション

高速機能トランザクションは、送信後コミット・トランザクションとして稼働する必要があります。

このコミット・モードに矛盾する OTMA トランザクションでのどのパラメーターも、トランザクションをリジェクトさせます。クライアント入力の特ランザクションが正しく定義されている場合は、既存の高速機能アプリケーション・プログラムは、OTMA で稼働します。

## IMS 再始動処理および OTMA

z/OS システム間カップリング・ファシリティグループに接続されている IMS サブシステムが、再始動しなければならない場合は、IMS は、再始動時にそのグループに再接続しますが、すべてのクライアントは、IMS に新規クライアント・ビッド要求を送信する必要があります。

## XRF 処理および OTMA

XRF は OTMA をサポートします。

**推奨事項:** アクティブおよび代替の IMS サブシステムは、同じ z/OS システム間カップリング・ファシリティ (XCF) グループにあることが必要です。それらが同じ XCF グループにない場合は、XRF テークオーバーの後、クライアントは、新規 XCF グループに接続する必要があります。それらが同じ XCF グループにある場合は、XRF テークオーバーの後、クライアントは、自動的に XCF グループに再接続されます。

アクティブ IMS は、メンバー名として IMS USERVAR 名を指定して XCF グループに結合します。代替 IMS は、XRF テークオーバーの後だけそのグループに結合し、同じ USERVAR 名を、メンバー名を使用して接続します。アクティブ・サブシステムと XRF 代替サブシステムが、同じ XCF グループにある場合は、XRF テークオーバー時に、新規アクティブ IMS は、以下のように処理します。

- 古い IMS メンバーが切断されていることを確認する
- そのグループを再結合する
- OTMA クライアント宛てにキューされた出力を送達し始め、続いてそのクライアントにクライアント・ビッド処理を行う

IMS XRF は、**/START OTMA** コマンドおよび **/STOP OTMA** コマンドを追跡しません。アクティブ IMS および代替 IMS 両者の IMS プロシージャは、両方とも、OTMA=Y を指定して、XRF テークオーバー時に IMS が自動的に XCF グループを再結合することを確認めます。

XRF 環境中では、OTMA クライアントは、クラス 3 端末相当で、自動的に IMS に再接続しません。クライアントは、IMS が XCF グループを残したことを検出して、グループを再結合するために、IMS (または XRF 代替) を待ちます。その後、クライアントは、新規クライアント・ビッド要求を IMS に送信します。

XRF 複合システムの代替システムは、再同期が開始した時あるいは特定 T パイプの再同期を追跡しません。再同期中に IMS が失敗したら、クライアントは IMS の失敗を検出して、新規 XRF アクティブ・システムと再同期する必要があります。

## IMS Queue Control Facility および OTMA

IMS Queue Control Facility (QCF) は、OTMA メッセージをサポートします。QCF を使用して、サポートされているすべての IMS リリース間、あるいは共用キューおよび非共用キュー間でスイッチすることができ

ます。QCF は、ISPF インターフェースおよび BMP 環境下で制御ステートメントを使用して、オンラインでアクセスすることが可能です。

TMEMBER および TPIPE は、INCLUDE と EXCLUDE 制御ステートメント用のオペラントです。

### tmember

1 文字から 16 文字の OTMA ターゲット・メンバー (クライアント) 名です。同じ文字で始まる名前のグループは、同じ文字の後にアスタリスク (\*) を使用することにより汎用的に指定できます。最初の文字としてアスタリスクを 1 つ指定した場合、すべての OTMA トランザクションを組み込みまたは除外します。

## T パイプ

1 文字から 8 文字の OTMA トランザクション・パイプ名です。同じ文字で始まる名前のグループは、同じ文字の後にアスタリスク (\*) を使用することにより汎用的に指定できます。

選択的に OTMA メッセージ をリカバリーするには、INCLUDE および EXCLUDE 制御 ステートメントを使用してください。OTMA オペランドのある INCLUDE および EXCLUDE ステートメントのフォーマットは、以下のとおりです。:

```
INCLUDE operand(,)
```

```
EXCLUDE operand(,)
```

operand は、カラム 10 から始まり、以下のうちの 1 つでなければなりません。

- T MEMBER=T メンバー
- T PIPE=T パイプ

**例:** トランザクション・パイプ名 S4A1BV6 を使用して、OTMA メッセージをすべて選択するには、以下のよう指定します。

```
INCLUDE T MEMBER=*,  
T PIPE=S4A1BV6
```

同じ T メンバーを持つメッセージはすべて一緒にグループ化され、T パイプ名によってカウントが報告されます。

**例:**

```
**** MESSAGES INSERTED BY DESTINATION ****  
BY OTMA DESTINATION  
T MEMBERNAME  
T PIPE1                count  
T PIPE2                count
```

クライアント・ビッド要求が、現行の OTMA 宛先解決出口ルーチンの名前を変更した場合は、IMS 終了前にエンキューされ、その後メッセージ・リキュアーによって再処理されるどのトランザクションも、変更された出口ルーチン名を使用しない可能性があります。代替 PCB への挿入では、クライアント記述子内で出口ルーチン名を使用します。

QCF を使用して、メッセージ・タイプだけでなくメッセージのカテゴリーを識別することも可能です。以下の表は、カテゴリー・パラメーター、およびパラメーターに関連したサポート対象のメッセージ・タイプとキーワードを示します。

表 145. カテゴリー・タイプによるメッセージの選択

カテゴリー・パラメーター	説明	サポートされたメッセージ・タイプとキーワード
DESTTYPE	選択されたメッセージ・タイプに対するメッセージの宛先をチェックします	APPC、LTERM、MSC、OTMA、LTRAN、RTRAN、TRANS、VSP
SRCETYPE	選択されたメッセージ・タイプに対するメッセージの送信元をチェックします	APPC、MSC、OTMA、VSP
MSGTYPE	選択されたメッセージ・タイプに対するメッセージの送信元または宛先をチェックします。	APPC、LTERM、MSC、OTMA、VSP

**関連資料:** QCF のカテゴリーごとのメッセージ選択の詳細については、「IMS Queue Control Facility for z/OS ユーザー・ガイド」を参照してください。

## OTMA での共用キューの使用

このトピックでは、OTMA での IMS 共用キューの使用に関する一般情報について説明します。

代替 PCB 処理のデリバリーを確実にするために、OTMA をすべての IMS 上で使用可能にしてください。そして、共用キュー・グループ中の IMS システムごとに固有の z/OS システム間カップリング・ファシリテーターメンバー名を割り当てます。

**/DISPLAY TRANS ALL QCNT** を使用して、処理待ちの共用キュー・グループに現時点で存在するすべての OTMA トランザクションを表示してください。**/DISPLAY QCNT OTMA MSGAGE 0** を使用して、共用キューにあるすべての OTMA アウトバウンド・メッセージを表示します。

**注:** OTMA アウトバウンド・メッセージを処理していた IMS システムが正常にシャットダウンし、その後コールド・スタートした場合に、OTMA アウトバウンド・メッセージが共用キューに残っていると、それらのアウトバウンド・メッセージは共用キューで動けない状態になり、配信できなくなります。ただし通常シャットダウンの後にウォーム・スタートが続いた場合は、シャットダウン時に共用キューにあったアウトバウンド・メッセージは配信が可能です。

IMS Connect などの OTMA ホールドキュー対応クライアントの場合は、OTMA スーパーメンバー機能を使用することにより、アウトバウンド・メッセージが共用キューで動けない状態にならず、確実にクライアントに配信できるようになります。

HIOP ストレージ・プールの一時的な不足の結果として、メッセージ DFS1269E を受け取る可能性があります。このメッセージは、共用キュー・リソースの登録時の IMS の内部障害を通知します。OTMA 用に共用キュー・リソースを再登録するには、IMS コマンド **/STOP OTMA** および **/START OTMA** を発行してください。

共用キュー環境では、IMS は 3 つのシステム・チェックポイントの後に、アイドル状態で、キューに入っておらず、同期化されていないトランザクション・パイプを除去します。

OTMA ALTPCB メッセージを共用キュー環境から IMS 間 TCP/IP 接続を介してリモート IMS システムに送信する場合、トランザクション・メッセージは、TCP/IP 接続を管理している IMS Connect インスタンスに直接接続されている IMS システムからのみリモート IMS システムに送信できます。

共用キュー環境の OTMA コミット後送信メッセージの場合、OIC TCB 下で他のジョブに使用されるメインの ITASK とは別に、メッセージを処理するための新規 ITASK が新規 OID TCB 下に作成されます。

### 関連概念

805 ページの『[宛先決定のための OTMAYPRX ユーザー出口および DFSYDRUO 出口ルーチンの使用](#)』トランザクション・パイプ名は、IMS LTERM 名または APPC/IMS TP 名と同じでもかまいません。

### 関連資料

[DFS1269E メッセージ情報 \(診断\)](#)

### 関連情報

[DFS1269E \(メッセージおよびコード\)](#)

## OTMA コミット後送信メッセージ

OTMA コミット後送信 (コミット・モード 0 または CM0) メッセージは、共用キュー・グループ内のどの IMS システム上でも処理可能です。プログラム間通信も、どの IMS 上でも稼働可能です。

フロントエンド IMS と他のバックエンド IMS との間での OTMA メッセージ処理のパフォーマンスを高めるために、各共用キュー・フロントエンド IMS 用のオペレーター識別タスク制御ブロック (OID TCB) 下の新しい IMS ITASK が作成されます。

## OTMA 非送信請求メッセージ

OTMA クライアントは、非送信請求メッセージを受信するために、共用キュー・グループ内のすべての IMS システムに接続する必要があります。OTMA クライアント接続が必要なのは、非送信請求メッセージを起す可能性のあるトランザクションが、共用キュー・グループ内のどの IMS 上でも稼働できるためです。



非送信請求メッセージを処理する IMS (バックエンド・システム) が、そのメッセージを受信するのとは異なる IMS の場合は、非送信請求メッセージはバックエンド・システムによって送信されます。したがって、OTMA はバックエンド IMS でも使用可能でなければなりません。

## OTMA 送信後コミット・メッセージ

OTMA 送信後コミット・メッセージは、共用キュー・グループ内にあるすべての IMS システムで処理することが可能です。

入力同期トランザクションからプログラム間通信で作成された同期および非同期トランザクションは、常に、プログラム間通信を開始したトランザクションと同じ IMS システムで実行されます。

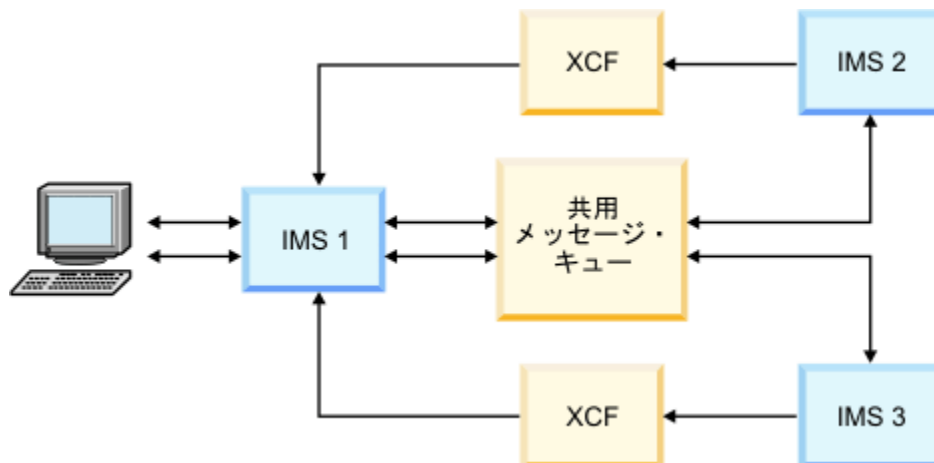


図 139. 共用キュー・グループの複数の IMS システムで処理される OTMA メッセージ

また、プログラム間通信は、保護会話 (sync レベル 2) に対しては許可されていません。

送信後コミット処理を使用する同期トランザクションは、次のコミット・レベルをサポートします。

- NONE
- CONFIRM
- SYNCPT

コミット後送信処理を使用する非同期トランザクションは、次のコミット・レベルをサポートします。

- RESYNC
- NO RESYNC

同期および非同期トランザクションのコミット・レベルを以下の図に示します。

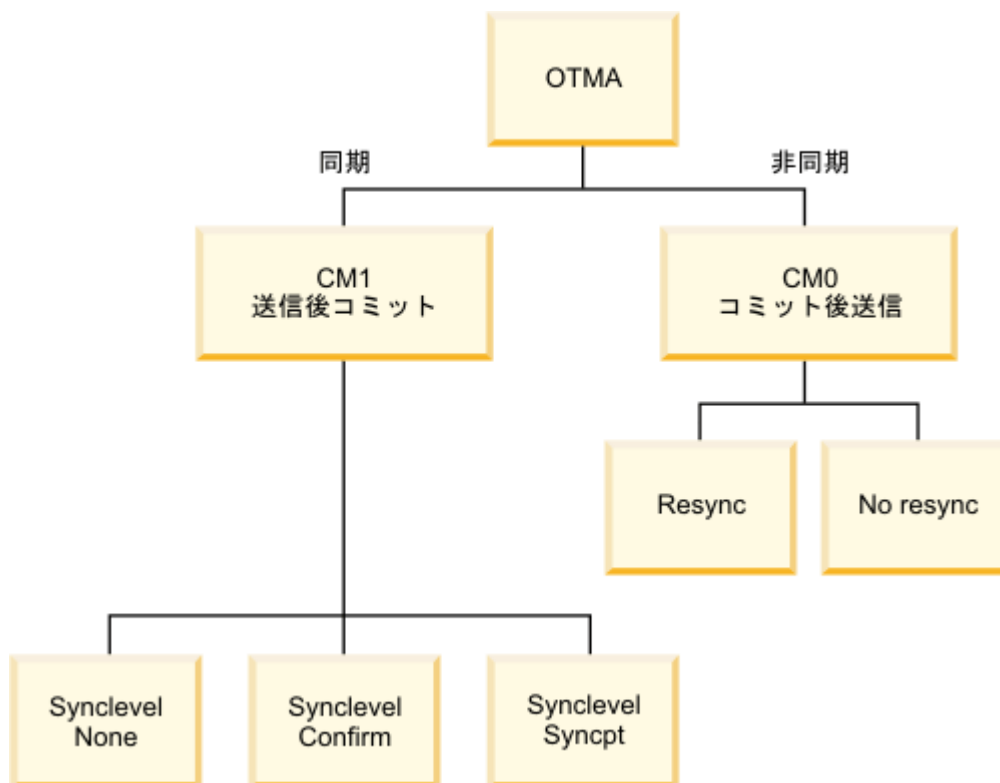


図 140. 同期および非同期トランザクションとそれぞれのコミット・レベル

/DISPLAY ACTIVE コマンドを使用して、OTMA 送信後コミットの共用キュー機能がアクティブかどうか判別してください。

## その他の IMS コマンドの使用

IMS コマンドの /DISPLAY TMEMBER *membername* TPIPE *tpipename* QCNT は、T パイプ状況と特定 IMS システムに対する共用キュー中の出力メッセージ・キューのカウントを示します。

### 関連概念

842 ページの『OTMA プログラム間通信処理』

OTMA では 2 タイプのメッセージ通信が出現します。それは、コミット後送信および送信後コミットです。

## バックエンド IMS システムからの ALTPCB 出力のリトリブ

共用キュー環境では、追加ステップを実行しない限り、バックエンド IMS システムによって共用キューに入れられる代替 PCB (ALTPCB) 出力は、バックエンド・システムへの類似性を持ちます。追加ステップを実行しない限り、この類似性が原因で、発信元のフロントエンド IMS システムは出力をリトリブすることができません。

### このタスクについて

バックエンド IMS システムによって共用キューに入れられた ALTPCB 出力をフロントエンド IMS システムがリトリブできるようにするには、以下のソリューションを使用できます。

### 手順

1. スーパーメンバー 機能を使用可能にし、フロントエンド IMS システムで RESUME TPIPE 呼び出しを発行するように OTMA クライアントをコーディングします。  
RESUME TPIPE 呼び出しのすべての取り出しオプションは、共用キュー環境でサポートされます。
2. バックエンドの類似性を使用不可にします。このオプションは、IBM MQ のように、OTMA クライアントがスーパーメンバー 機能を使用しない場合に便利です。

以下のいずれかの方法で、バックエンド類似性を使用不可にすることができます。

- フロントエンド IMS システムの DFSYDTx PROCLIB メンバー内の OTMA クライアント 記述子に、ALTPCBE=YES を指定する。
- 入力ランザクション・メッセージの状態データ接頭部の第 3 バイトに、TMAMALTB フラグ (X'01') を設定する。

この 2 つのオプションのいずれかが指定されている場合、バックエンド IMS システムで生成された ALTPCB 出力は、共用キューを介してフロントエンド IMS に送達されます。

#### 関連タスク

875 ページの『[非同期コミット後送信出力の共用: OTMA スーパーメンバー 機能](#)』

保留キュー対応 OTMA クライアント (IMS Connect など) は、OTMA のスーパーメンバー 機能を有効にすることにより、非同期コミット後送信 (CMO) 出力メッセージを共用できます。OTMA スーパーメンバー 機能は、z/OS Sysplex Distributor 環境で IMS Connect の複数のインスタンスをサポートするために、特に設計されたものです。

## IMS の終了および OTMA

IMS が終了すると、OTMA は OTMA クライアントへの通知を試行し、事情に応じて追加のアクションを実行する可能性があります。

#### 関連タスク

[IMS のシャットダウン \(オペレーションおよびオートメーション\)](#)

### IMS 終了の OTMA クライアント通知

IMS のシャットダウン後に新規要求が送信されないように、IMS 終了すると OTMA クライアントに通知が送られます。

計画されている IMS シャットダウン中、OTMA はプロトコル・コマンド TMAMMNTR (X'3C' Resource Monitor) と TMAMCSPA (X'14' Suspend TPIPE) をすべての OTMA クライアントに送信します。

計画されていない IMS シャットダウン中、OTMA は異常終了処理中に XCF グループから離脱し、XCF 出口ルーチンが OTMA クライアントに通知します。

#### 関連資料

909 ページの『[サーバー状態プロトコル・コマンド](#)』

サーバー状態プロトコル・コマンドの状態データは、DFSYMSG マクロの TMAMHDR DSECT によってマップされます。

889 ページの『[メッセージ制御情報セクション](#)』

各 OTMA メッセージごとに、OTMA メッセージ接頭語の最初のセクションにメッセージ制御情報を指定する必要があります。

### IMS の終了と IMS 間 TCP/IP メッセージ

IMS が正常終了または異常終了するとき T パイプが WAIT\_R 状態にある場合、OTMA は再始動処理中に T パイプ上のメッセージをデフォルト・タイムアウト・キューに転送します。

T パイプの WAIT\_R 状況は、OTMA がリモート IMS システムに送信されたメッセージの受信の確認応答を待機中であることを示します。

**/STOP TMEMBER** TPIPE コマンドを発行すると、T パイプの WAIT\_R 状況をクリアできます。

**/DISPLAY TMEMBER** コマンドを使用すると、どの OTMA T パイプが WAIT\_R 状況にあるかを判別できます。

#### 関連資料

[/DISPLAY TMEMBER コマンド \(コマンド\)](#)

## OTMA の制約事項および要件

---

OTMA には一般の制約事項と要件がいくつか適用されます。

### 制約事項

OTMA メッセージのすべての接頭部の最大長は、4096 バイトです。この長さには、アプリケーション・データは含みません。

SET0 呼び出しを使用する既存の IMS アプリケーション・プログラムは、期待どおりに稼働しないことがあります。SET0 呼び出しを使用する APPC/IMS アプリケーション・プログラムは、暗黙的な OTMA サポートを使用するために修正が必要になる可能性があります。

OTMA は、IMS メッセージ形式サービス (MFS) をサポートしません。しかし、MFS メッセージ出力記述子 (MOD) 名は、クライアントが、OTMA メッセージの接頭部中に指定することができます。

OTMA は、IMS フロントエンド・スイッチをサポートしません。

OTMA は、暗号化できません。

OTMA では、主記憶データベース (MSDB) へのアクセスは、読み取り専用です。OTMA からの MSDB へのアクセスでは、更新アクセスは使用可能ではありません。

OTMA は、IMS DBCTL 環境中では操作しません。

OTMA では、IMS 端末管理コマンド (/FORMAT、/HOLD、/RCL、および /SIGN コマンドなど) は許可されません。

### 要件

IMS 会話型トランザクションおよび高速機能トランザクションは、送信後コミットとして定義する必要があります。既存の高速機能アプリケーションは、OTMA で実行できます。

IMS 端末 (例えば、SLU 2 端末) からのトランザクションは、出力を直接クライアントに経路指定できませんので、OTMA 宛先解決ユーザー出口 (OTMAYPRX) を使用する必要があります。

クライアントが、メッセージ接頭語のセキュリティ・データ・セクションで、「セキュリティ検査なし」を指定しない場合は、すべてのユーザー ID は RACF で検査する必要があります。

z/OS システム間カップリング・ファシリティー (XCF) マクロを含む IMS モジュールは、IMS の新規リリースに対して再アSEMBルする必要があります。

同じ z/OS リソース・リカバリー・サービス (RRS) リカバリー単位の一部であり、共通データベース・リソースにアクセスする OTMA 保護会話メッセージは、同じ IMS システム上でスケジュール設定し、実行する必要があります。

IMS は、保護会話トランザクション間のロックとデータベース・バッファを共有しますが、このサポートは単一の IMS システム内に限定されます。共有キュー間を越えて機能することはありません。そのようなメッセージは、同じ IMS システムでスケジュールされるようにする必要があります。

## システム・リソースおよび OTMA の管理

---

OTMA は、IMS システム・リソースを使用します。OTMA がそのようなリソースをどのように使用するかに、さまざまな要因が影響します。その使用状況をモニターする方法がいくつかあります。

以下のトピックでは、このシステム・リソースの考慮事項について説明します。

## OTMA リソース・モニター

OTMA は、OTMA トランザクションの処理に使用される IMS システム・リソースをモニターし、IMS システムの OTMA トランザクション処理の状況を OTMA クライアントに通知します。

OTMA クライアントは、IMS システムが OTMA メッセージを正常に処理していないという通知を受け取ると、異なる IMS システムに OTMA トランザクション・メッセージを転送するなど、適切なアクションを実行できます。

OTMA は、通知をサーバー状態プロトコル・コマンドとして OTMA クライアントに送信します。OTMA は、以下の環境でサーバー状態プロトコル・コマンドを発行します。

- OTMA クライアントが T パイプ接続を確立している場合
- IMS の OTMA メッセージの処理機能に重大な変更が実施された場合
- 60 秒間隔のハートビート・メッセージとして

サーバー状態プロトコル・メッセージでは、以下の状態を使用して IMS システムの全体的な状態を分類します。

### 正常状態 (X'03')

IMS は使用可能であり、OTMA メッセージを正常に処理しています。

### 機能低下状態 (X'02)

IMS の OTMA メッセージの処理速度が低下しています。1 つ以上の状態が IMS の OTMA メッセージの処理速度が通常よりも低下していることを示す場合、OTMA は機能低下状態プロトコル・コマンドを発行します。

### 使用不可状態 (X'01)

IMS は、OTMA トランザクションの処理を受け入れることができなくなっています。OTMA は、使用不可状態プロトコル・コマンドを発行して、1 つ以上の重大な状態により、IMS が OTMA メッセージを処理できないというアラートを OTMA クライアントに出します。

クライアントに IMS 処理の全体的な状態を通知するだけでなく、IMS 処理が機能低下または使用不可の状態の場合に、サーバー状態プロトコル・コマンドに、機能低下または使用不可の状態に関連する特定のリソースの状態に関する追加情報を含めることもできます。

サーバー状態プロトコル・コマンドは、OTMA メッセージ・ヘッダーのメッセージ制御情報セクションのプロトコル・コマンド・タイプ・フィールド (TMAMCTYP) 内の値 X'3C' で識別されます。サーバー状態プロトコル・コマンド自体は、OTMA ヘッダーの状態データ・セクションに送信され、DFSYSMSG マクロの TMAMRSIM フィールド接頭部によりマップされます。

OTMA クライアント、OTMA T パイプ接続、および IMS で現在処理されている OTMA メッセージに関する情報を表示するには、タイプ 2 IMS コマンド **QUERY OTMATI** を発行します。

### 関連概念

[816 ページの『現行のトランザクション・ワークロードの表示』](#)

IMS で現在処理されている OTMA メッセージの数とタイプに関する情報を表示するには、タイプ 2 IMS コマンド **QUERY OTMATI** を発行します。

### 関連資料

[909 ページの『サーバー状態プロトコル・コマンド』](#)

サーバー状態プロトコル・コマンドの状態データは、DFSYSMSG マクロの TMAMHDR DSECT によってマップされます。

[889 ページの『メッセージ制御情報セクション』](#)

各 OTMA メッセージごとに、OTMA メッセージ接頭語の最初のセクションにメッセージ制御情報を指定する必要があります。

## 現行のトランザクション・ワークロードの表示

IMS で現在処理されている OTMA メッセージの数とタイプに関する情報を表示するには、タイプ 2 IMS コマンド **QUERY OTMATI** を発行します。

パラメーターを指定せずに **QUERY OTMATI** コマンドを発行すると、IMSplex 内の各 IMS システム上の OTMA メッセージの現行数が、メッセージのサブミットに使用されている OTMA クライアントおよび T パイプと共にリストされます。例えば、3 つの IMS システムから構成される IMSplex で、**QUERY OTMATI** を発行すると、以下の情報が返されます。

TMember	TPipe	MbrName	CC	MsgCnt
MQ	APPLA	IMSA	0	102
MQ	APPLB	IMSB	0	201
WAS	APPLC	IMSB	0	301

**QUERY OTMATI** コマンドにパラメーターを指定すると、返される情報をフィルターに掛けて詳細化できます。

例えば、メッセージの待機時間が長すぎて IMS で処理できないかどうかを確認する必要がある場合は、**MSGAGE(nnn)** サブパラメーターを指定すると、IMS 内の *nnn* よりも古いすべてのメッセージの数が返されます。ここで *nn* はクロック時間を表し、秒単位で指定します。以下は、**QUERY OTMATI MSGAGE(3)** を指定した例です。各 IMS システムのメッセージ数には、3 秒より古いメッセージのみが含まれます。

TMember	TPipe	MbrName	CC	MsgCnt	MsgAge
MQ	APPLA	IMSA	0	50	3
MQ	APPLB	IMSB	0	80	5
WAS	APPLC	IMSB	0	100	4

上記の例で、**MsgAge** 列に表示される数は、3 秒より古いすべてのメッセージの中で最も新しくサブミットされたメッセージの経過時間です。例えば、T パイプ **APPLB** の **MQ** によって **IMSB** にサブミットされた、3 秒より古い 80 個のメッセージの中で、メッセージの最短の処理時間は 5 秒です。

**QUERY OTMATI** コマンドには、25 個以上のパラメーターおよびサブパラメーターがあり、これらを指定すると、OTMA クライアントからのメッセージの処理に関連した各種コンポーネントおよびリソースに関する情報が返されます。メッセージの数と経過時間だけでなく、以下のような情報もリトリブできます。

- メッセージ処理のためにスケジュールに入れられたトランザクションまたはアプリケーション・プログラム
- セキュリティー情報
- 2 フェーズ・コミット処理のコンテキスト・トークン
- OTMA コミット・モード
- 同期レベル
- コールアウト処理の相関関係子トークン
- クライアントからの確認応答用のタイムアウト設定
- 会話型トランザクションに属するメッセージの会話 ID。

## ストレージ上の OTMA メッセージ TIBs の影響

OTMA クライアントから受信したメッセージごとに、イムスはトランザクション・インスタンスブロック (TIB) を作成します。

それぞれの TIB には、約 8 KB の拡張専用域ストレージが必要です。

タイプ 2 イムス コマンド **照会オットマティ** を発行することにより、現在 イムス によって処理されている TIBs および OTMA メッセージに関する情報を表示することができます。表示できる情報には、メッセージ数、メッセージの経過時間、およびその他の情報が含まれます。



Normally TIBs are freed and the storage released by イムス either after イムス enqueues the commit-then-send (CM0) input message of TIBs created for commit-then-send messages or after イムス returns a send-then-commit (CM1) output message of TIBs created for send-then-commit messages.

通常、過度の数の TIBs は、メッセージあふれ条件またはオーファン TIBs のいずれかによって引き起こされます。

通常、メッセージあふれ条件は以下の結果として発生します

- 停止されたプログラム
- メッセージ処理領域の非可用性
- イムスがメッセージを処理できる速度を超える着信メッセージの率

メッセージあふれ条件を回避するために、イムスが一度に処理できる OTMA メッセージの最大数を指定することによって、メッセージあふれ検出を有効にすることができます。イムスは、TIBs をカウントすることによって現在処理されているメッセージの数を決定します。メッセージ検出が使用可能になっている場合、イムスは、TIBs の数が最大に近づいているとメッセージを出します。

## 孤立 TIBs

イムスが TIB を解放できない状況があります。これが発生すると、TIB はオーファン TIB としてストレージ内に存続します。

孤立 TIBs は、イムスが OTMA トランザクションを処理するときに作成されますが、送信後コミット出力メッセージまたは送信後コミット出力メッセージの代わりに、送信後コミット出力メッセージまたは DFS2082 メッセージのどちらも生成しません。例えば、送信後コミット・メッセージが非応答モード・トランザクションとして定義されている非同期トランザクションに対してプログラム間通信をトリガーする場合、および OTMAASY=Y の場合、イムスが TIB を削除できない場合には、切り替え先トランザクションが IOPCB への応答を生成すると考えられます。

イムスには、メッセージあふれが有効になっている場合に、合計 TIBs の数に孤立 TIBs が含まれています。メッセージあふれ条件のために OTMA 入力停止された場合は、**/DISPLAY OTMA** または **/ T MEMBER を表示** コマンドを発行して、TIBs の数を調べることができます。OTMA 入力停止した後にまだ高い数の TIBs が存在する場合は、孤立 TIBs が存在する可能性があります。

イムス 会話型トランザクションに関連する孤児 TIBs を除去することができます。コマンド **/ 会話の表示** を発行することにより、会話型トランザクションに関連する孤立 TIBs を識別することができます。会話のために孤立した TIB が存在する場合は、コマンド **/EXIT 会話** を発行して、孤立した TIB を除去することができます。

また、IMS PROCLIB データ・セットの DFSYDTx メンバー内に DFSOTMA 記述子の **終了時間=** パラメータを指定することによって、OTMA 内の会話型トランザクションに関連する孤立 TIBs を除去することもできます。会話型トランザクションの前の反復の完了後に、会話型トランザクションが指定された時間アイドル状態のままである場合、OTMA はトランザクションを終了し、そのトランザクションに割り振られたイムス リソースを解放します。

## 関連概念

816 ページの『[現行のトランザクション・ワークロードの表示](#)』

IMS で現在処理されている OTMA メッセージの数とタイプに関する情報を表示するには、タイプ 2 IMS コマンド **QUERY OTMATI** を発行します。

818 ページの『[メッセージあふれ検出](#)』

イムスが、OTMA クライアントから受信したメッセージを迅速に処理できない場合、またはまったく処理できない場合は、イムス システム内のメッセージのビルドがメッセージあふれの状態になることがあります。メッセージあふれ条件が発生すると、イムスがメッセージを処理するために使用するメッセージとトランザクション・インスタンス・ブロック (TIBs) の両方が必要とするストレージが、境界より下のストレージを使い尽くし、潜在的に z/OS システム S40D システム 異常終了を引き起こす可能性があります。

## 関連資料

[DFSOTMA 記述子の構文およびパラメーター \(システム定義\)](#)



## OTMA ACEE フラッシング制御

OTMA アクセス機能環境エレメント (ACEE) フラッシング制御機能は、IMS 制御領域の仮想記憶域が使い尽くされるのを防止します。

### このタスクについて

OTMA セキュリティーが使用可能に設定されている場合、パフォーマンス上の目的で、OTMA は RACF ユーザー・プロファイルを RACF アクセサー環境エレメント (ACEE) としてメモリーにキャッシュします。その後、この ACEE は、後続の RACF の呼び出しで、入力メッセージ内の IMS コマンドまたは要求された IMS トランザクションに対するユーザー ID の権限を判別するために使用されます。ACEE を使用すると、RACF に対する入出力が少なくなります。OTMA ACEE フラッシング制御を有効にすると、ACEE をキャッシュに入れることができる RACF ユーザー・プロファイルの数が制限されます。これにより、OTMA によってキャッシュに入れられる ACEE の数が制限されます。

OTMA ACEE フラッシング制御機能が有効にされているかを確認し、キャッシュに入れられている OTMA ACEE の総数を確認するには、**/DISPLAY OTMA** コマンドを発行します。

### 手順

1. OTMA ACEE フラッシング制御を有効にするには、IMS PROCLIB データ・セットの DFSYDTx メンバーの DFSOTMA 記述子で、**TOACEE=YES** を指定します。
2. キャッシュに入れられた ACEE を持つことができる RACF ユーザー ID の最大数を定義するには、DFSOTMA 記述子の **ACEEUSR=** パラメーターの値を指定します。

### 関連資料

[DFSOTMA 記述子の構文およびパラメーター \(システム定義\)](#)

## メッセージあふれ検出

イムスが、OTMA クライアントから受信したメッセージを迅速に処理できない場合、またはまったく処理できない場合は、イムスシステム内のメッセージのビルドがメッセージあふれの状態になることがあります。メッセージあふれ条件が発生すると、イムスがメッセージを処理するために使用するメッセージとトランザクション・インスタンス・ブロック (TIBs) の両方が必要とするストレージが、境界より下のストレージを使い尽くし、潜在的に z/OS システム S40D システム異常終了を引き起こす可能性があります。

メッセージあふれ状態に関連する問題を回避するために、OTMA は、イムスシステムで処理されるのを待機している入力メッセージの数をモニターすることができます。モニターがアクティブな場合、OTMA は、各 OTMA クライアントからの入力メッセージの数と、結合されたすべての OTMA クライアントからの入力メッセージの合計数をモニターします。個々の OTMA クライアントからの入力メッセージの数がその定義された最大値に達すると、OTMA はそのクライアントからの新規入力メッセージを抑止します。入力メッセージの合計数がグローバル最大に達すると、OTMA は警告メッセージを発行しますが、新しい入力メッセージは受け入れられます。

個々の OTMA クライアントと、すべての OTMA クライアントの入力メッセージの最大数を結合することができます。許容されるメッセージの最大数として 200 から 9999 を指定することも、0 を指定することにより、メッセージあふれ検出を使用不可にすることも

OTMA クライアントのデフォルトの最大数は 5000 です。結合されたすべての OTMA クライアントのデフォルトの最大数は 10,000 です。

イムスシステム内のメッセージの数が、OTMA クライアントまたはすべての OTMA クライアントのいずれかに定義された最大数の 80% に達した場合、OTMA は、適切な OTMA クライアントに対してサーバー状態プロトコル・コマンドを発行し、イムスはマスター・コンソールにメッセージ DFS1988W を発行します。また、イムスは、最大数に達するまで、DFS1988W を 5% 増加するたびに発行します。

イムスシステム内の個々のクライアントのメッセージの数が最大値を超えると、OTMA は別のサーバー状態プロトコル・コマンドを発行し、イムスはメッセージ DFS1989E を発行します。次に、OTMA は、メッセージあふれ条件が解決されるまで、OTMA クライアントからのすべての入力メッセージをリジェクトします。

イムスシステム内のメッセージの数が最大値の 50 パーセントになると、OTMA は、あふれ状態が解放されたことを通知する新しいサーバー状態プロトコル・コマンドを OTMA クライアントに発行します。

グローバル・フラディング制御をアクティブにして、新しい OTMA 入力トランザクションを抑止するには、以下のいずれかのタスクを実行

- すべての OTMA メンバーについて、OTMA クライアント記述子の DFSOTMA メンバーにグローバル・フラディング限度値 **レポート=** を指定します。
- Issue the イムス command / **TMEMBER** **すべての入力を開始** *nbr*.

OTMA クライアント記述子に指定された値は、/ **開始 TMEMBER** コマンドによってオーバーライドすることができます。クライアント・ビッド要求により小さい数が指定されている場合、クライアント・ビッド要求は、OTMA クライアント記述子に指定された値をオーバーライドすることもできます。

## 関連概念

816 ページの『[ストレージ上の OTMA メッセージ TIBs の影響](#)』

OTMA クライアントから受信したメッセージごとに、イムスは トランザクション・インスタンスブロック (TIB)を作成します。

## OTMA T メンバーの管理

IMS は、OTMA ターゲット・メンバー (T メンバー) に関する情報を取得したり、これらのメンバーを動的に変更したりするためのコマンドを提供します。

### このタスクについて

**START TMEMBER** および **STOP TMEMBER** コマンドを使用して、特定の OTMA T メンバーおよび T パイプの開始およびシャットダウンを行います。コマンドでは、エンキューされた CM1 メッセージおよび CM0 メッセージ用の入力メッセージ保留キューのサイズおよびタイムアウトの長さを変更するためのパラメーターも提供されます。

/**STOP TMEMBER** コマンドは、T パイプ上でクライアントからの ACK メッセージまたは NAK メッセージを待っているすべての CM0 メッセージの待ち状態 (WAIT\_A、WAIT\_H、または WAIT\_S) もクリアします。

**DISPLAY TMEMBER** コマンドを使用すると、一部または全部の T メンバーに関する詳細情報をリトリブすることができます。 **DISPLAY TMEMBER ALL** を指定して、構成済みの OTMA T メンバーおよび関連の T パイプのすべてを対象にしたレポートをリトリブすることも、特定の T メンバーおよび T パイプの名前を指定してコマンドを限定し、検索の範囲を狭めることもできます。応答出力には、各 T パイプの処理モード、セキュリティ・レベル、状況、ワークロード統計、およびその他の情報が表示されます。

CM0 出力メッセージの場合のみ、T パイプ上のエンキュー・カウントおよびデキュー・カウントが更新されます。CM1 メッセージはリカバリー可能ではないため、同期レベルに関係なく、CM1 メッセージのカウントは更新されません。

**DISPLAY STATUS TMEMBER** コマンドを使用して、停止された T パイプを持つ OTMA メンバーのリストをリトリブすることもできます。このコマンドは、管理者の注意が必要な T メンバーを判別するために役立ちます。また、このコマンドは、各 T メンバーのスーパーメンバー名 (構成されている場合) もリトリブします。

停止したメッセージまたは失効したメッセージを OTMA T パイプから手動でデキューすることが必要な場合があります。例えば、クライアント・アプリケーションを変更した後で T パイプから不整合メッセージをパージし、代わりに新規メッセージをリトリブできるようにすることが必要な場合があります。

**DEQUEUE** コマンドを使用すると、特定の T パイプからメッセージをパージすることができます。このコマンドでは、その T パイプに最初にエンキューされたメッセージをデキューするか、エンキューされたすべてのメッセージをデキューかのオプションが提供されます。

T メンバーまたは特定の T パイプについての追加情報が必要な場合は、**TRACE TMEMBER** コマンドを使用して、OTMA クライアントのアクティビティ・トレースを使用可能にすることができます。スーパーメンバー全体に対するトレースをオンにすることもできます。特定の OTMA クライアントについて、現在トレース中の T メンバーおよび T パイプを判別するには、**DISPLAY TRACE TMEMBER** コマンドを使用します。トレース情報は、OTMA ログ・レコードで入手できます。

一時的Tパイプは、存在しないTパイプに対して **/TRACE tpipe** あるいは **/STOP tpipe** コマンドを発行したとき、作成されます。一時的Tパイプは、入力メッセージがIMSに到達したときあるいは出力メッセージが、Tパイプの出力キューにキューされたとき、永続Tパイプに変換されます。

**/DISPLAY tpipe** コマンドを一時的Tパイプに出したとき、TMP の状況が表示されます。

#### 関連資料

[/DISPLAY TMEMBER コマンド \(コマンド\)](#)

[/DISPLAY STATUS コマンド \(コマンド\)](#)

[/DEQUEUE コマンド \(コマンド\)](#)

[/START TMEM コマンド \(コマンド\)](#)

[/STOP TMEM コマンド \(コマンド\)](#)

## Tパイプ・リソースのインパクト

トランザクション・パイプ (Tパイプ) は大量のIMSリソースと処理時間を消費するため、Tメンバー当たりのTパイプの作成数を制限するようにしてください。

IMSは、以下の環境にある場合を除いて、トランザクション・パイプが3つの連続したシステム・チェックポイントでアイドル状態だった場合、その後にトランザクション・パイプを除去します。

- TパイプまたはTパイプ保留キューに、コミット後に送信のメッセージが入っている。
- tpipe は停止します。
- Tパイプにトレースが設定されている。
- Tパイプが同期化されている (コミット後に送信の入力トランザクション用にMQSeriesによって使用されるTパイプなど)。
- AUTO または SINGLE-WAIT のいずれかのオプションを指定した RESUME TPIPE 要求のために、Tパイプが待ち状態にある。
- Tパイプが MCP 状態であり、Tパイプが共用キュー環境で実行されていて、グローバル・キューに出力メッセージがある可能性があることを示している。

**ヒント:** Tパイプのキューに入れられたメッセージがなく、Tパイプについて MCP 状況が表示されていて、Tパイプを除去できない場合は、**/DISPLAY TMEMBER tmembername TPIPE tpipename QCNT** コマンドまたは **/DISPLAY TMEMBER tmembernameQCNT** コマンドを発行して、MCP 状況をリセットしてください。

- TパイプがIMSによってスキャンされている。

**/DISPLAY TMEMBER TPIPE** コマンドを使用して、上記のリストのいずれかの状況がTパイプに当てはまるためにIMSでTパイプを除去できないかどうかを確認することができます。

特定のOTMAクライアント用に作成されるTパイプの数を制御できる1つの方法は、各OTMAクライアント用とIMSシステム用のTパイプの最大許容数を設定することです。

### 個別のOTMAクライアントのTパイプの制限

OTMAクライアントのTパイプの最大数は、IMS.PROCLIB データ・セット内のDFSYDTxメンバーのOTMAクライアント記述子にMAXTPパラメーターを指定することによって指定されます。

OTMAクライアントに対してMAXTPパラメーターが指定されている場合、OTMAはそのクライアント用に作成されるTパイプの数をモニターし、この数が特定のレベルに達すると警告を出します。

OTMAクライアントに対してMAXTP値が設定されると、OTMAはそのクライアント用に作成されるTパイプの数をモニターします。Tパイプの総数が最大許容数の80%に達した場合、IMSは警告メッセージDFS4382Wをシステム・コンソールおよびマスター端末オペレーター (MTO) に出します。Tパイプの数が最大数の100%に達した場合、IMSはエラー・メッセージDFS4383Eをシステム・コンソールおよびMTOに出します。新規のTパイプを必要とする入力トランザクションは、OTMAセンス・コードX'29'付きのNAKで拒否されます。

警告メッセージまたはエラー・メッセージが出された後、そのクライアント用の T パイプの総数が最大許容数の 50% 低下すると、IMS はメッセージ DFS4384I を発行して、T パイプの数が正常に戻ったことを示します。

## IMS システムでの T パイプの制限

DFSOTMA という名前の OTMA クライアント記述子を定義し、MAXTP パラメーターを指定することにより、すべての OTMA クライアントのグローバル TPIPE 警告しきい値を定義することができます。DFSOTMA 記述子は、すべての OTMA クライアントに適用される IMS システムの値を定義する OTMA システム・クライアント記述子です。

IMS システム内の T パイプの数が DFSOTMA MAXTP 値の 80% に達すると、IMS はシステム・コンソールと MTO の両方にメッセージ DFS4515W を発行し、すべての OTMA クライアントが新規 T パイプを作成しないようにします。入力トランザクションで新規 T パイプが必要な場合、トランザクションは、OTMA センス・コード X'29' を含む NAK メッセージによってリジェクトされ、メッセージ DFS4516E がシステム・コンソールと MTO に送信されます。

DFS4515W または DFS4516E が発行され、システム内でモニターされている T パイプの総数が 50%、またはグローバル T パイプ警告しきい値の別のユーザー指定レベルまで低下すると、IMS はメッセージ DFS4517I を発行して、T パイプの数が正常に戻ったことを示します。

OTMA DFSOTMA システム・クライアント記述子で IMS システム内の T パイプ数のグローバル最大数が定義されておらず、1 つ以上の OTMA クライアントがそれらの OTMA クライアント記述子に T パイプの最大許容数を定義している場合は、それらのすべてのクライアント記述子の中で最も高い MAXTP 値が IMS システムのグローバル警告しきい値として使用されます。モニター対象のすべての OTMA クライアントで使用中の T パイプの総数がグローバル警告しきい値に達すると、IMS は警告メッセージ DFS4385W をシステム・コンソールおよび MTO に出します。

グローバル警告が出された後、T パイプの総数がグローバル警告しきい値の 80% まで低下すると、IMS はメッセージ DFS4386I を発行して、T パイプの数が正常に戻ったことを示します。

## X'3C' プロトコル・コマンドの通知

OTMA は、各種の警告、エラー、および解除のしきい値の時点で、OTMA クライアントに対して X'3C' プロトコル・コマンドも送信します。警告またはエラーに関する X'3C' プロトコル・コマンドを受け取ると、クライアント・アプリケーションは後続のトランザクションを必要に応じて別の IMS システムに転送することができます。T パイプの数が解除しきい値を下回ると、再度 X'3C' プロトコル・コマンドが、警告またはエラーのフラグをオフにして発行されます。

## T パイプ数に関する情報の表示

IMS コマンドの **/DISPLAY OTMA** および **/DISPLAY T MEMBER** を使用すると、MAXTP 値が設定されている OTMA クライアントの現行の T パイプ数を表示できます。ある OTMA クライアントが T パイプの最大許容数に達している場合、コマンドの出力には、その OTMA クライアントの USER\_STATUS として MAX TPIPE が表示されます。

複数の OTMA クライアント間の最高の MAXTP 値によって設定されるグローバル警告しきい値は、IMS サーバー用の出力フィールド TPNCT の下に表示されます。

## 単純 T パイプ機能を使用した T パイプ・ストレージの削減

DFSYDTx IMS.PROCLIB メンバーの DFSOTMA 記述子に **LITETP=YES** を指定すると、単純 T パイプ機能を有効にできます。この機能が有効な場合、フロントエンド入力トランザクションを処理するために共用キュー・バックエンド IMS システム内で T パイプが作成される際に使用されるストレージが少なくなります。**LITETP=YES** を指定すると、IMS がさらに多くの T パイプをサポートできるようになります。**LITETP=YES** が指定された場合、IMS 初期設定時にメッセージ DFS7411I が出され、この機能が有効であることを示します。

単純 T パイプでは必要なストレージが通常の T パイプよりも少ないため、T パイプフラッシング制御用の T パイプ・カウントを計算する際に、バックエンド T パイプで加重係数が使用されます。加重係数は、通常

の T パイプ・ストレージ・サイズを基準とする単純 T パイプ・ストレージ・サイズのパーセンテージです。これは、通常、28% です。DFSOTMA 記述子構文内の **LITETP=** パラメーター、および調整後の T パイプ・カウンターの計算用のパラメーターの記述を参照してください。

#### 関連資料

[/DISPLAY TMEMBER コマンド \(コマンド\)](#)

[DFSOTMA 記述子の構文およびパラメーター \(システム定義\)](#)

## アイドル状態の T パイプの自動除去

IMS は、システム・チェックポイント処理のときにトランザクション・パイプ (T パイプ) をスキャンし、削除できる T パイプがないかどうかを判別します。

IMS は、以下の環境にある場合を除いて、トランザクション・パイプが 3 つの連続したシステム・チェックポイントでアイドル状態だった場合、その後にトランザクション・パイプを除去します。

- T パイプまたは T パイプ保留キューに、コミット後に送信のメッセージが入っている。
- `tpipe` は停止します。
- T パイプにトレースが設定されている。
- T パイプが同期化されている (コミット後に送信の入力トランザクション用に MQSeries によって使用される T パイプなど)。
- AUTO または SINGLE-WAIT のいずれかのオプションを指定した RESUME TPIPE 要求のために、T パイプが待ち状態にある。
- T パイプが MCP 状態であり、T パイプが共用キュー環境で実行されていて、グローバル・キューに出力メッセージがある可能性があることを示している。

**ヒント:** T パイプのキューに入れられたメッセージがなく、T パイプについて MCP 状況が表示されていて、T パイプを除去できない場合は、`/DISPLAY TMEMBER tmembername TPIPE tpipename QCNT` コマンドまたは `/DISPLAY TMEMBER tmembernameQCNT` コマンドを発行して、MCP 状況をリセットしてください。

- T パイプが IMS によってスキャンされている。

`/DISPLAY TMEMBER TPIPE` コマンドを使用して、上記のリストのいずれかの状況が T パイプに当てはまるために IMS で T パイプを除去できないかどうかを確認することができます。

前述リストにある環境の 1 つが真であるために IMS が 3 つの連続するチェックポイントに渡ってアイドル状態になっていた T パイプを除去できない場合、IMS は、アイドル状態の T パイプを以下のシステム・チェックポイントで再度除去しようとします。

OTMA クライアント、OTMA T パイプ接続、および IMS で現在処理されている OTMA メッセージに関する情報を表示するには、タイプ 2 IMS コマンド **QUERY OTMATI** を発行します。

#### 関連資料

[/DISPLAY TMEMBER コマンド \(コマンド\)](#)

[QUERY OTMATI コマンド \(コマンド\)](#)

## OTMA での会話型トランザクションの終了

OTMA で、会話がハングした後に TPIPE キューにあるメッセージをクリーンアップするために、あるいはその他の理由で、会話型トランザクションを終了することが必要になる場合があります。

### このタスクについて

会話型トランザクションは、例えば、会話を開始した OTMA クライアントが、IMS によって終了を検出されることなく終了した場合などに、ハングすることがあります。そのような場合、ハングしたトランザクションに関連付けられているストレージ (トランザクション・インスタンス・ブロック (TIB) およびスクラッチパッド域に必要なストレージなど) は、解放されません。ハングしたトランザクション会話が終了しないと、長い間には、それらによって浪費されるストレージの量がシステムに大きな影響を及ぼすおそれがあります。



## 手順

以下のオプションのいずれかを使用して、ハングした会話型トランザクションを終了できます。

- IMS PROCLIB データ・セットの DFSYDTx メンバー内に DFSOTMA 記述子の **ENDCONV=** パラメーターを指定します。  
会話型トランザクションの事前の反復が完了した後に指定の期間、会話型トランザクションが活動停止したままである場合、OTMA がそのトランザクションを終了し、そのトランザクションに対して割り振られた IMS リソースを解放します。
- 以下のように、**QUERY OTMATI** コマンドおよび **/EXIT CONV** コマンドを使用します。
  - a) SHOW(ALL) キーワードと 1 つ以上の T メンバーか T パイプ、またはその両方の名前を指定して、IMS タイプ 2 コマンド **QUERY OTMATI** を発行します。
  - b) CONVID 列の **QUERY OTMATI** コマンド出力で、MsgAge 列のメッセージ経過時間値がハングした会話型トランザクションを示すのに十分な大きさであるトランザクション・インスタンスの会話型 ID をメモします。
  - c) トランザクション・インスタンスを識別して、その会話型 ID をメモした後、**/EXIT CONV conv\_id TMEMBER tmembername TPIPE tpipename** コマンドを発行します。

## 関連資料

[/EXIT コマンド \(コマンド\)](#)

[QUERY OTMATI コマンド \(コマンド\)](#)

[DFSOTMA 記述子の構文およびパラメーター \(システム定義\)](#)

## IMS が OTMA 入力メッセージに対して割り振る SAP 数の指定

パフォーマンスを維持して ECSA ストレージの使用量を制限するには、IMS PROCLIB データ・セットの OTMA DFSYDTx メンバーの DSAP パラメーターと DSAPMAX パラメーターを指定することで、IMS が各 OTMA クライアントのために事前に割り振る保管域接頭部 (SAP) の数と、IMS が実行時に動的に割り振ることができる SAP の最大数を調整できます。

### このタスクについて

IMS は、OTMA クライアントの OTMA 入力メッセージ (OIM) タスク制御ブロック (TCB) と一緒に SAP を使用して、OTMA 入力メッセージを処理します。IMS では、入力メッセージごとに 1 つの SAP が必要です。IMS がメッセージの処理を完了した後、IMS は別のメッセージに SAP を再利用します。

デフォルトで、IMS は、クライアントの接続時に 18 個の SAP を OTMA クライアントに事前に割り振ります。事前に割り振られた SAP は、IMS が再始動されるまで OTMA クライアントに割り振られたままです。

新しい入力メッセージが OTMA クライアントから受信されるときにすべての SAP が使用中である場合、IMS は、SAP の最大数に達するまで、必要に応じて追加の SAP を動的に割り振ります。デフォルトで、1 つの OTMA クライアントに動的に割り振ることができる SAP の最大数は 500 です。動的に割り振られた SAP がなくなったら、IMS は、事前に割り振られた SAP のみが残るまで、動的に割り振られた SAP を徐々にクリーンアップします。

IMS が追加の SAP を十分な速度で動的に割り振ることができないか、最大数の SAP がすでに割り振られた場合、IMS は選択ディスパッチングを使用して、優先順位が高い作業が処理されてから、優先順位が低い作業が処理されるように、処理に優先順位を付けます。

IMS システム・チェックポイント時に、IMS が前回のシステム・チェックポイント以降に短時間でも OTMA 入力メッセージに選択ディスパッチングを使用した場合、IMS は、メッセージ DFS0769I、OIM SELECTIVE DISPATCHING - SAPS を発行します。

大部分の場合、選択ディスパッチングは、IMS が追加の SAP を割り振るまでのわずかな時間だけしかアクティブにならないため、問題やパフォーマンス上の大きな影響は発生しません。ただし、IMS が頻繁に選択ディスパッチングを使用するか、定期的に大量の重要な作業が予想される場合、すべての着信 OTMA メッセージを十分に処理できることを確実にするために、IMS が事前に割り振る SAP 数を調整することを検討できます。また、IMS が OTMA クライアントに割り振ることができる SAP の最大数を増やすことも検討できます。

OIM TCB による SAP の使用状況の統計は、IMS レベルでは X'450F' ディスパッチャー統計ログ・レコードに書き込まれます。個々の OTMA メンバー・レベルでは個別の X'4518' TCB ディスパッチャー統計ログ・レコードに書き込まれます。これらのログ・レコード内の統計を調べると、選択ディスパッチングの発生に関する便利な情報が得られます。

IMS は、拡張共通ストレージ域 (ECSA) に SAP を保管します。割り振る SAP 数を増やすと、使用する ECSA ストレージが増えます。

## 手順

- IMS が OTMA クライアントに事前に割り振る SAP 数を指定するには、IMS PROCLIB データ・セットの DFSYDTx メンバーで、OTMA クライアント記述子に DSAP パラメーターを指定します。
- IMS が OTMA クライアントに動的に割り振ることができる SAP の最大数を指定するには、IMS PROCLIB データ・セットの DFSYDTx メンバーで、OTMA クライアント記述子に DSAPMAX パラメーターを指定します。
- 事前に割り振られる SAP 数に OTMA システム・デフォルト値を指定するには、IMS PROCLIB データ・セットの DFSYDTx メンバーで、DFSOTMA 記述子に DSAP パラメーターを指定します。
- IMS が動的に割り振ることができる SAP の最大数に OTMA システム・デフォルト値を指定するには、IMS PROCLIB データ・セットの DFSYDTx メンバーで、DFSOTMA 記述子に DSAPMAX パラメーターを指定します。

## 関連資料

[OTMA クライアント記述子の構文およびパラメーター \(システム定義\)](#)

[DFSOTMA 記述子の構文およびパラメーター \(システム定義\)](#)

## 関連情報

[DFS0769I \(メッセージおよびコード\)](#)

## IMS メッセージ・キュー・データ・セットのサイズおよび OTMA

OTMA から IMS に入力するメッセージには、OTMA メッセージ接頭語およびその他の既存 IMS メッセージ接頭語の両方を含みます。

OTMA メッセージ接頭語は、可変長です。ユーザー・データ・セクションを除いても、OTMA メッセージ接頭語は、非常に長く、長さが 200 バイトを超えることがあります。ユーザー・データ・セクションを含む OTMA メッセージ接頭語は、IMS メッセージ・キュー・データ・セットに保管されるので、そのキュー・バッファ・プールの使用が増加します。

**ヒント:**このようにキュー・バッファ・プールの使用が増加するため、メッセージ・キュー・データ・セットのサイズを増やすことを試みてください。

OTMA メッセージ接頭語のセキュリティー・データ・セクションにネットワーク・セキュリティー資格情報が含まれている場合、OTMA メッセージのサイズは最大 504 バイト増える可能性があります。そのため、SHMSG および LGMSG のメッセージ・キュー・データ・セットのサイズと、メッセージ・キュー・プールのサイズを増やすことを検討してください。

## OTMA のバッファ・プールの使用

IMS-OTMA 環境で OTMA が重いトラフィックの場合は、LUMP と HIOP でのプールの使用が顕著に増加します。

LUMP と HIOP プールは、専用ストレージから割り当てられますから、IMS 制御領域のサイズを増加させることが必要になる場合があります。また、特定の OTMA 制御ブロックは、他の限定リソースである拡張共通サービス域 (ECSA) から割り振られます。

**推奨事項:** 作業負荷に応じて ECSA サイズを増加してください。例えば、クライアントが、数秒の間に 100 T パイプを超えて、20 メッセージ以上を送信している場合は、IMS 制御領域を 200 MB 以上に増やし、ECSA サイズを 50 MB 以上に増やすようにしてください。IMS 制御領域サイズまたは ECSA サイズを増加できない場合は、作業負荷のバランスを考えて、IMS がそのバッファをより効率的に再利用できるようにしてください。



## 従属領域の占有率および OTMA

送信後コミット・トランザクションは、出力が送信されている間 (同期点が起こる前)、従属領域に残ります。 **DISPLAY ACT** コマンドを使用すると、OTMA 関連の待ち状態にある特定の従属領域に関する追加情報を入手できます。

以下の状態 (STATUS 応答フィールドから) は、OTMA 関連のアクティビティが進行中であることを示しています。

- WAIT-CALLOUT
- WAIT-RRS PC
- WAIT-SYNCPOINT
- TERM-WAIT SYNCPT
- TERM-WAIT RRS

領域が同期プログラム通信の応答を待っている場合、その領域で要求がタイムアウトになるまでの残り時間が、**DISPLAY ACT REGION** コマンドからの出力の END TIME フィールドに表示されます。

### 推奨事項:

- 多くのトランザクションが、送信後コミット (CM1) トランザクションの場合は、スループット・パフォーマンスを改善するために、従属領域の数を増やしてください。
- synclevel=confirm または synclevel=syncpt を使用する CM1 メッセージについては、OTMA クライアントが必要な ACK 応答または NAK 応答を適切な時間内に提供しない場合に従属領域を解放するためのタイムアウト間隔を指定してください。
- 可能な限り、コミット後送信 (CM0) OTMA トランザクションを多く使用してください。

### 関連タスク

825 ページの『OTMA メッセージへの確認応答のタイムアウト間隔』

OTMA が OTMA 出力メッセージに対する ACK または NAK の確認応答を待機する時間の長さを決める、ACK タイムアウト間隔を指定できます。

### 関連資料

[/DISPLAY ACT コマンド \(コマンド\)](#)

## OTMA メッセージへの確認応答のタイムアウト間隔

OTMA が OTMA 出力メッセージに対する ACK または NAK の確認応答を待機する時間の長さを決める、ACK タイムアウト間隔を指定できます。

### このタスクについて

ACK タイムアウト間隔は、次のタイプの OTMA 出力メッセージに対して指定できます。

- 処理のためにリモート IMS システムに送信されるトランザクション・メッセージ
- 一部の送信後コミット (CM1) 応答メッセージ
- コミット後送信 (CM0) 応答メッセージ

リモート IMS システムに送信されるトランザクション・メッセージの場合、ACK タイムアウト間隔が有効期限切れになると、OTMA はトランザクション・メッセージをタイムアウト・キューに転送します。トランザクション・メッセージがタイムアウトになった後に OTMA がローカル IMS Connect から ACK メッセージを受け取った場合、OTMA はローカル IMS Connect に対して X'2B' センス・コード付きの NAK を発行します。IMS Connect は NAK メッセージを破棄します。

synclevel=confirm または synclevel=syncpt を使用する送信後コミット・トランザクションでは、タイムアウト間隔の有効期限が切れる前に、OTMA が、OTMA クライアントから予定される ACK または NAK 応答を受信しなかった場合、OTMA はトランザクションを異常終了し、IMS はトランザクションをバックアウトして、OTMA クライアントに OTMA CM1 割り振り解除メッセージを発行し、z/OS システム・コンソールにメッセージ DFS0809E を発行します。

コミット後送信トランザクションでは、タイムアウト間隔の有効期限が切れる前に、予定される ACK または NAK 応答を受信しなかった場合の OTMA のアクションは、クライアントが、IMS Connect などのように保留キューに対応しているか、または IBM MQ などのように保留キューに対応していないかに応じて異なります。保留キュー対応クライアントの場合、OTMA のアクションは、出力が I/O PCB キューまたは T パイプの保留キューのいずれにあるかに応じて異なります。

保留キュー対応クライアントで、コミット後送信トランザクションの出力が I/O PCB キューにあるときに、タイムアウト間隔の有効期限が切れた場合、OTMA は以下のアクションを実行します。

1. OTMA は、CMO 出力を転送 T パイプに送信します。
2. 転送 T パイプが指定されていない場合は、OTMA は出力をタイムアウト・キューに送信します。
3. タイムアウト・キューが指定されていない場合は、OTMA は CMO 出力をデフォルトのタイムアウト・キュー DFS\$\$TOQ に送信します。
4. DFS3494E を z/OS システム・コンソールに発行します。

保留キュー対応クライアントで、コミット後送信トランザクションの出力が T パイプ保留キューにあるときに、タイムアウト間隔の有効期限が切れた場合、OTMA は以下のアクションを実行します。

1. OTMA は、出力をタイムアウト・キューに送信します。
2. タイムアウト・キューが指定されていない場合は、OTMA は CMO 出力をデフォルトのタイムアウト・キュー DFS\$\$TOQ に送信します。
3. DFS3494E を z/OS システム・コンソールに発行します。

保留キューに非対応のクライアントでは、OTMA は、CMO 出力が I/O PCB キューにあるときの保留キュー対応クライアントの場合と同じアクションを実行します。

DL/I ICAL 呼び出しを発行する IMS アプリケーション・プログラムからの同期コールアウト要求の場合、OTMA はその要求を CMO 出力メッセージとして処理します。OTMA は、状態データの接頭部に CMO フラグを設定し、同期コールアウト・メッセージの受信の確認応答を待機します。ACK のタイムアウト間隔内に確認応答が返されなかった場合、OTMA は以下のアクションを実行します。

1. 戻りコード X'100' および理由コード X'104' を IMS アプリケーション・プログラムに発行します。
2. 同期コールアウト要求メッセージを廃棄します。メッセージは転送されません。
3. DFS3494E を z/OS システム・コンソールに発行します。

OTMA クライアントのタイムアウト間隔は、OTMA クライアント記述子で指定できます。OTMA クライアント記述子内のタイムアウト間隔をオーバーライドするには、**/START TMEMBER** コマンドの TIMEOUT パラメーターに異なる間隔を指定します。また、OTMA クライアント記述子内のタイムアウト間隔は、OTMA クライアントのクライアント・ビッド要求の中で、より短いタイムアウト間隔を指定してもオーバーライドできます。

個々のトランザクションには、OTMA クライアント用に指定した値よりも短いタイムアウト間隔のみを指定できます。

送信後コミット・タイムアウト値を指定しなかった場合、OTMA はデフォルト値の 120 秒を使用します。

OTMA クライアント用に設定されている現行のタイムアウト間隔を表示するには、**/DISPLAY TMEMBER** コマンドを発行します。

## OTMA クライアント・レベルでの確認応答タイムアウト間隔の指定

OTMA クライアント・レベルでは、1 つ以上の場所で ACK タイムアウト間隔を指定できます。

### このタスクについて

タイムアウト間隔は、以下の場所で指定できます。

- OTMA クライアント記述子の T/O= パラメーター
- /START TMEMBER コマンドの TIMEOUT パラメーター
- OTMA クライアントからのクライアント・ビッド要求内で状態データのオフセット 65 (X'41') にある 1 バイトのフィールド

コミット後送信トランザクションの場合、タイムアウト・キューの名前を指定するには、以下の操作を行います。

- クライアント・ビッド要求のバイト・オフセット 45 (X'2D') に X'04' を指定します。
- バイト・オフセット 66 (X'42') にタイムアウト・キューの名前を 8 バイト文字で指定します。

タイムアウト・キュー名を指定しない場合、タイムアウトになった CM0 出力はデフォルトのタイムアウト・キュー DFS\$\$TOQ に経路指定されます。

リモート IMS システムを宛先とするトランザクション・メッセージがタイムアウトになった場合、OTMA はそれらのメッセージを DFS\$\$TOQ に転送します。

## 関連概念

796 ページの『OTMA クライアント記述子』

OTMA クライアント記述子を使用して、特定の OTMA クライアントに関する情報を IMS に提供します。OTMA クライアント記述子項目は、DFSYDTx PROCLIB メンバー内で、記述子項目の列 1 に入っている M によって識別されます。

## 関連資料

899 ページの『サーバー使用可能コマンドおよびクライアント・ビッド・コマンド』

OTMA メッセージ接頭語のサーバー使用可能コマンドおよびクライアント・ビッド・コマンド用の状態データは、DFSMSG マクロの TMAMHDR DSECT によってマップされます。

/START TMEM コマンド (コマンド)

## CM1 トランザクション・メッセージでの確認応答タイムアウトの指定

IMS が出力応答に対する確認応答 (ACK または NAK) を待つ時間の長さを制限するために、synclevel=confirm または synclevel=syncpt を使用する送信後コミット (CM1) トランザクション用の入力メッセージの OTMA ヘッダーで、タイムアウト間隔を指定することができます。

## このタスクについて

タイムアウト間隔を設定するには、以下のようになります。

- トランザクション・メッセージのメッセージ制御情報接頭部のオフセット X'1E' にある 1 バイト・フィールドに、10 進整数を使用して、分単位で間隔を指定する。
- フラグ X'08' を状態データのバイト 5 に設定する。

## OTMA のトランザクション有効期限サポート

トランザクションに有効期限を指定して、クライアントが使用できなくなったトランザクションを IMS に処理させないようにすることで、処理コストを削減できます。

トランザクションに有効期限が指定されている場合、OTMA はそのトランザクションをモニターし、有効期限が切れる前にトランザクションが処理またはエンキューされていなければ、そのトランザクションを破棄します。

IMS 従属領域内の IMS アプリケーション・プログラムが、MSC、高速機能、または会話型処理のトランザクションをリトリブすると、OTMA はトランザクションの有効期限のモニターを停止します。このいずれかのイベントが発生すると、有効期限は適用されなくなり、IMS はトランザクションを処理して、出力を OTMA と OTMA クライアントに返します。

トランザクション有効期限は、IMS に対してトランザクションを定義するとき有効にするか、トランザクションが OTMA にサブミットされる際の OTMA メッセージ・ヘッダーの中で有効にすることができます。トランザクションの定義で指定された有効期限は、そのトランザクション・タイプのデフォルトの有効期限になり、トランザクションのすべてのインスタンスに適用されます。OTMA メッセージ・ヘッダーで指定された有効期限は、OTMA ヘッダーとともにサブミットされたトランザクション・インスタンスだけに適用され、トランザクション定義で指定されている有効期限より優先されます。

OTMA メッセージ・ヘッダーでは、有効期限を時点または時間の長さとして指定できます。時点として指定される有効期限は、ストア・クロック (STCK) 形式で指定され、トランザクションの有効期限が切れる時刻を示します。時間の長さとして指定される有効期限は、秒単位で指定されます。OTMA は、z/OS システ

ム間カップリング・ファシリティ (XCF) からトランザクションを受け取った時点から有効期限を計算します。

OTMA は、トランザクションの有効期限が切れているかどうかを、次の 3 つの時点でチェックします。

1. OTMA が XCF から初めてトランザクションを受け取る時。有効期限を過ぎている場合、OTMA はトランザクションを破棄してクライアントに NAK メッセージを返します。
2. OTMA が IMS にトランザクションをエンキューするとき。OTMA がエンキューする前にトランザクションの有効期限が切れていた場合、OTMA はそのトランザクションを破棄してクライアントに NAK メッセージを返します。
3. (MSC、高速機能、または会話型以外のトランザクションの場合) IMS アプリケーション・プログラムが GU 呼び出しを発行して入力キューからトランザクションをリトリートするとき。

IMS アプリケーション・プログラムが GU 呼び出しを使用してトランザクションをリトリートする前に入力キュー上のトランザクションの有効期限が切れた場合、OTMA はそのトランザクションを破棄し、デフォルトでは、OTMA クライアントにメッセージ DFS3688I を発行します。ただし、有効期限が切れた OTMA 入力メッセージに TMAMINPT フラグが設定されている場合は、DFS3688I メッセージを返す代わりに、OTMA は元の入力トランザクション・データを OTMA クライアントに返送します。

OTMA クライアントごとに、オプションで OTMA を構成して、IMS.PROCLIB データ・セットの DFSYDTx メンバー内の OTMA クライアント記述子に TODUMP=YES を指定し、入力キュー上で有効期限が切れたトランザクションについて症状ダンプを作成し、DFS554A メッセージを発行するように設定することができます。

また、入力メッセージの OTMA 状態データ接頭部に TMAMDUMP フラグを設定することで、個別のメッセージについて症状ダンプと DFS554A メッセージを要求することもできます。個別のメッセージでの TMAMDUMP フラグの設定は、そのメッセージについてのみ TODUMP=NO の指定をオーバーライドします。

以下のトランザクション・タイプについては、OTMA はトランザクションが IMS にエンキューされるまで有効期限をモニターします。

- MSC リモート・トランザクション
- 高速機能トランザクション

OTMA は、以下のトランザクション・タイプの有効期限をモニターしません。

- IMS 会話型トランザクション
- プログラム間通信の結果としてキューに入れられたトランザクション

トランザクションの定義の中で有効期限を定義する場合は、ステージ 1 システム定義マクロ TRANSACT の EXPRTIME パラメーター、またはタイプ 2 動的リソース定義コマンド **CREATE TRAN** または **UPDATE TRAN** の EXPRTIME パラメーターを使用します。宛先作成出口ルーチン (DFSINSX0) で作成されたトランザクションに有効期限を指定することもできます。これらの方法で有効となったトランザクション有効期限は、OTMA 固有ではありません。

## STCK 形式での OTMA トランザクション有効期限の指定

OTMA メッセージ・ヘッダーでは、トランザクション有効期限をストア・クロック (STCK) 形式で指定することができます。

### このタスクについて

STCK 形式で指定される有効期限は、トランザクションの有効期限が切れる時刻です。有効期限の時刻までに、トランザクションが IMS アプリケーション・プログラムによってリトリートされなかった、または MSC や会話型処理のためにエンキューされなかった場合、そのトランザクションは有効期限が切れ、OTMA によって破棄されます。

OTMA トランザクションに対して有効期限を STCK 形式で指定する方法は以下のとおりです。

## 手順

- OTMA ヘッダーの状態データ・セクションで、TMAMHIST フィールドに X'01' を指定します。
- OTMA ヘッダーの状態データ・セクションで、TMAMOSXP フィールドに時刻仕様のバイト・オフセットを指定します。これは、OTMA ヘッダーのユーザー・データ・セクションにおける時刻仕様のオフセットです。
- OTMA ヘッダーのユーザー・データ・セクションで、トランザクションの有効期限が切れる時刻を STCK 形式で指定するように OTMA クライアントをコーディングします。

## 関連資料

904 ページの『トランザクションおよびコールアウト・メッセージ』

OTMA メッセージ接頭語のトランザクション関連情報および同期コールアウト要求情報の状態データは、DFSYMSG マクロの TMAMHDR DSECT によってマップされます。

917 ページの『ユーザー・データ・セクション』

OTMA メッセージ接頭語のユーザー・データ・セクションは可変長で、セキュリティー・データ・セクションの後に続きます。どんなデータでも含むことができます。

## 秒単位での OTMA トランザクション有効期限の指定

OTMA メッセージ・ヘッダーでは、トランザクション有効期限を秒単位の時間の長さとして指定することができます。

## このタスクについて

秒単位で指定された場合、OTMA は z/OS システム間カップリング・ファシリティ (XCF) からトランザクションを受け取った時刻に、指定された時間の長さを加算することで有効期限を計算します。

例えば、指定された時間の長さが 10 秒である場合は、OTMA が XCF からトランザクションを受け取ってから 10 秒後にそのトランザクションの有効期限が切れます。10 秒が経過するまでに、トランザクションが IMS アプリケーション・プログラムによってリトリブされなかった、または MSC や会話型処理のためにエンキューされなかった場合、そのトランザクションは有効期限が切れ、OTMA によって破棄されます。

トランザクション有効期限の時間の長さを指定するには、OTMA ヘッダーの状態データ・セクションで以下のフィールドをコーディングします。

## 手順

- TMAMHIST フィールドで X'02' を指定します。
- TMAMOSXP フィールドで、OTMA が有効期限の計算に使用する時間の長さを秒単位で指定します。OTMA は、XCF からトランザクションを受け取った時刻にこの値を加算することで、有効期限を計算します。

## 関連資料

904 ページの『トランザクションおよびコールアウト・メッセージ』

OTMA メッセージ接頭語のトランザクション関連情報および同期コールアウト要求情報の状態データは、DFSYMSG マクロの TMAMHDR DSECT によってマップされます。

## OTMA セキュリティー

OTMA のセキュリティーは、RACF またはそれと同様なセキュリティー製品によって適用されます。以下のトピックでは、RACF を使用した OTMA および OTMA クライアントのセキュリティーの実装について説明します。

## このタスクについて

## OTMA の RACF セキュリティー・レベル

OTMA では、5 つのレベルの RACF セキュリティー (CHECK、FULL、JOIN、NONE、PROFILE を使用します)。

OTMA 用にグローバルに有効にできる RACF セキュリティー・レベルは、一度に 1 つだけです。ただし、/ SECURE OTMA TMEMBER コマンドを使用すると、個々の OTMA クライアントに異なるセキュリティ・レベルを指定できます。

### NONE

システム・ワイド・セキュリティ・レベル。RACF は、OTMA を通じて受信したメッセージに対しては呼び出されません。特に、

RACF は、IMS が IBM MQ または IMS Connect から接続要求 (クライアント・ビッド) を受信するときは呼び出されません。

RACF は、着信メッセージのユーザー ID が有効なユーザー ID (RACF に定義済みのもの) かどうかを検査するためには、呼び出されません。

RACF は、着信メッセージのユーザー ID が、IMS コマンド、またはメッセージで要求された IMS トランザクションに対して許可されているかどうかを確認するためには、呼び出されません。

ユーザー ID キャッシング・スキームは使用されません。

### PROFILE

メッセージ毎セキュリティ・レベル。つまり、OTMA を通じて入力されたそれぞれの着信メッセージがチェックされ、RACF を呼び出すかどうかを決定します。IMS は、各着信メッセージを個別にチェックして、セキュリティ値が NONE、CHECK、または FULL に設定されているかどうかを確認します。特に、

IMS Connect から入力されたメッセージは、1 バイトのセキュリティ・フラグ・フィールドを含みます。このフィールドの値が、RACF を呼び出すかどうかを決定します。

IBM MQ-IMS Bridge アプリケーションから入力されたメッセージは、MQIHL 構造内に SecurityScope フィールドを含んでいます。このフィールドの値が、RACF を呼び出すかどうかを決定します。

**ヒント:** アプリケーション開発者が、各着信メッセージに RACF セキュリティー・レベルを N (NONE)、C (CHECK)、あるいは F (FULL) として設定している場合は、PROFILE セキュリティー・レベルの使用を検討してください。このケースでは、各メッセージに設定されたセキュリティ・レベルが、そのメッセージに関するセキュリティ検査のために IMS が RACF を呼び出すかどうかを決定します。ユーザーは、IMS コマンドおよび IMS トランザクションのセキュリティについてアプリケーション・プログラマーが決めることを希望しない場合があるかもしれません。RACF は、IMS が IBM MQ または IMS Connect から接続要求 (クライアント・ビッド) を受信したときに呼び出されます。

### JOIN

システム全体のセキュリティ・レベル。すなわち、IMS Connect または MQ シリーズなどの OTMA クライアントからの初期クライアント・ビッド要求でユーザー ID を許可するためにのみ RACF が呼び出されます。

接続が許可された後、接続で受け取ったメッセージで実行される追加のトランザクションまたはコマンドのセキュリティ検査はありません。

### CHECK

システム・ワイド・セキュリティ・レベル。OTMA を通じて受信したメッセージに対して RACF が呼び出されることを意味します。特に、以下の場合に RACF が呼び出されます。

- ・ クライアント・ビッド接続要求の場合。クライアント・ビッドが成功すると、キャッシュ、またはハッシュ・テーブルが各 OTMA クライアントに対してビルドされます。

ユーザー ID キャッシング・スキームは、IMS/OTMA 環境下で使用されます。キャッシング・スキームは、許可検査の効率を向上させます。

キャッシュまたはハッシュ・テーブルは、すでに検査済みのユーザー ID を保管するために使用されます。それぞれの OTMA クライアント (IMS Connect、IBM MQ for z/OS など) は、クライアント・ビ

ッドが正常に行われた後に IMS 制御領域内に作成されたハッシュ・テーブルを持っています。ハッシュ・テーブルを使用することで、ユーザー ID を VERIFY するための RACF への呼び出し回数が最小化されます。このようにすることで、同じユーザー ID が IMS/OTMA 宛ての多重メッセージを入力する場合、IMS がユーザー ID の有効な入力に対するハッシュ・テーブルをチェックすることが可能となり、RACF への VERIFY 呼び出しを回避することが可能になります。ハッシュ・テーブルのユーザー ID の入力は、ユーザー ID のアクセサー環境エレメント (ACEE) に対するポインターを含んでいます。ポイントされた ACEE は、RACF へのリソース (コマンドおよびトランザクション) FASTAUTH 呼び出しに対して使用することが可能です。

- 着信メッセージのユーザー ID が有効なユーザー ID (RACF に定義済みのもの) かどうかを検査するため。

OTMA クライアント (IMS Connect または IBM MQ for z/OS) が着信メッセージ内に UTKOKEN を提供した場合、IMS は、RACF への VERIFY 呼び出し上に UTKOKEN のアドレスを提供します。VERIFY 処理中の UTKOKEN を使用することで、パフォーマンスが向上します。RACF は、検査済みユーザー ID に対して ACEE セキュリティー管理ブロックを IMS に戻します。

- 着信メッセージのユーザー ID が、IMS コマンド、またはメッセージで要求された IMS トランザクションに対して許可されているかどうかを検査するため。検査許可処理の間に RACF によって以前に作成された ACEE のアドレスは、RACF への FASTAUTH 呼び出し上で IMS によって提供されます。
- 着信メッセージのユーザー ID が、DL/I CHNG または AUTH 呼び出しの宛先として設定された IMS トランザクション・コードに対して許可されていることを検査するため。キャッシュされた ACEE は、これらの呼び出しに使用され、多数の CHNG または AUTH 呼び出しを発行するアプリケーション・プログラムのパフォーマンス上の問題を解消します。

## FULL

システム・ワイド・セキュリティー・レベル。OTMA を通じて受信したメッセージに対して RACF が呼び出される。

FULL は、以下の例外を除いて、CHECK と同じ特性を持っています。

- 検査処理の間に RACF が再び呼び出され、追加の ACEE セキュリティー管理ブロックが従属領域に作成されます。
- 従属領域は、要求側のユーザー ID の下で実行されます。従属領域からアクセスされるリソース、例えばファイルなどは、そのユーザー ID に対して許可されている必要があります。

## OTMA セキュリティーの指定

OTMA および OTMA クライアントに対するセキュリティーの指定の大部分は、トランザクション・メッセージの接頭部および **/SECURE OTMA** コマンドを使用して、システム定義中に行うことができます。

### このタスクについて

OTMA セキュリティーを指定するすべての方式で、影響の有効範囲が同じわけではありません。OTMA z/OS システム間カップリング・ファシリティ (XCF) グループの全メンバーに対してセキュリティーを指定できる方式もあれば、1つの OTMA クライアントまたは1つの OTMA トランザクションまたはコマンド・メッセージに対してだけセキュリティーを指定できる方式もあります。

### システム定義時の OTMA セキュリティーの指定

IMS および DCC 実行プロシージャの OTMASE= 実行パラメーターを使用して、OTMA z/OS システム間カップリング・ファシリティグループ内のすべての OTMA クライアントに対して RACF セキュリティー・レベルをグローバルに設定できます。OTMA で IMS Connect を使用した場合、IMS TM への OTMA クライアント接続が認証されたときに RACF 統計が記録されるようにすることもできます。

### 手順

- OTMASE= パラメーターを使用して指定できる RACF セキュリティーのレベルは、以下のとおりです。
  - CHECK
  - FULL



- JOIN
- NONE
- PROFILE

OTMASE= 実行パラメーターを使用して行った指定は、**/SECURE OTMA** コマンドを発行することによってオーバーライドできます。

- OTMA で IMS Connect を使用しており、IMS TM への OTMA クライアント接続が認証されたときに RACF 統計をシステム定義の段階で記録されるようにしたい場合は、IMS PROCLIB データ・セットの HWSCFGxx メンバーの HWS ステートメントに TMRACFST=Y を指定します。

### 関連タスク

791 ページの『[OTMA セキュリティー検査のレベルの定義](#)』

RACF などのセキュリティ製品を使用する場合、OTMASE パラメーターを使用して、さまざまなセキュリティ検査レベルを指定できます。

183 ページの『[IMS Connect に対する RACF セキュリティー統計の使用可能化](#)』

IMS Connect が RACF を呼び出すように構成されている場合、IMS Connect が RACF 呼び出し **RACROUTE REQUEST=VERIFY** を発行するときに、RACF セキュリティー統計が記録され、更新されるようにすることができます。IMS DB への ODBM クライアント接続および IMS TM への OTMA クライアント接続の RACF 統計が記録され、更新されるようにすることができます。RACF 統計を使用可能にした後は、統計は 1 日 1 回まで更新されます。

### 関連資料

830 ページの『[OTMA の RACF セキュリティー・レベル](#)』

OTMA では、5 つのレベルの RACF セキュリティー (CHECK、FULL、JOIN、NONE、PROFILE) を使用します。

[IMS 環境で使用されるプロシージャ \(システム定義\)](#)

[ODACCESS ステートメント \(システム定義\)](#)

[QUERY IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

## OTMA セキュリティーのオンラインでの変更

オンライン・コマンドを使用すると、すべての OTMA クライアントについてグローバルに、および各 OTMA クライアントについて個別にセキュリティを変更できます。IMS TM への OTMA クライアント接続を認証するために IMS Connect が RACF 呼び出し **RACROUTE REQUEST=VERIFY** を出したときに RACF 統計が記録されるかどうかをオンラインでも指定できます。

### すべての OTMA クライアントについてグローバルに、また、各 OTMA クライアントについて個別にセキュリティを変更

オンライン・タイプ 1 コマンド **/SECURE OTMA** を使用して、すべての OTMA クライアントについてグローバルに、または個々の OTMA クライアントについて、セキュリティを変更できます。**/SECURE OTMA** コマンドを使用すると、IMS または DCC 実行プロシージャで OTMASE= パラメーターを使用して行ったセキュリティ指定を、オーバーライドすることができます。

**/SECURE OTMA** コマンドでは、以下のパラメーターを使用することにより、OTMA z/OS システム間カップリング・ファシリティー (XCF) グループに対してグローバルに RACF セキュリティー・レベルを指定するか、OTMA XCF グループについて RACF セキュリティーを使用不可にすることができます。

- **/SECURE OTMA CHECK**
- **/SECURE OTMA FULL**
- **/SECURE OTMA JOIN**
- **/SECURE OTMA PROFILE**
- **/SECURE OTMA NONE**

以下のコマンドを発行することにより、個々の OTMA クライアントに対して RACF セキュリティー・レベルを指定できます。

- **/SECURE OTMA CHECK TMEMBER** *tmember\_name*
- **/SECURE OTMA FULL TMEMBER** *tmember\_name*
- **/SECURE OTMA JOIN TMEMBER** *tmember\_name*
- **/SECURE OTMA PROFILE TMEMBER** *tmember\_name*
- **/SECURE OTMA NONE TMEMBER** *tmember\_name*

個々の OTMA クライアントに対するセキュリティーの指定は、OTMA XCF グループの残りのメンバーに対するグローバルなセキュリティー設定をオーバーライドします。

**/SECURE OTMA NONE** を指定する場合、IMS は、クライアント・ビッド要求あるいはトランザクションに対して、どんなセキュリティーをクラスで指定しようとして、セキュリティー検査に RACF は使用しません。

OTMA に対して RACF セキュリティー検査が使用不可に設定されている場合は、OTMA を通じて以下のデフォルトの IMS コマンドだけを発行できます。

- **/BROADCAST**
- **/LOCK**
- **/LOG**
- **/RDISPLAY**
- **/UNLOCK**

これらのコマンドについての完全な説明は、「IMS V15 コマンド 第 1 巻: IMS コマンド A-M」および「IMS V15 コマンド 第 2 巻: IMS コマンド N-V」に記載されています。

## IMS TM への OTMA クライアント接続に対する RACF 統計の使用可能化

IMS TM への OTMA クライアント接続を認証するために IMS Connect が **RACROUTE REQUEST=VERIFY** 呼び出しを出したときに記録される RACF 統計をオンラインで使用可能にするには、**UPDATE IMSCON TYPE(CONFIG)** コマンドで **TMRACFST(ON)** キーワードを使用します。

RACF 統計を使用可能にすると、IMS Connect は **RACROUTE REQUEST=VERIFY** 呼び出しで **STAT=ASIS** パラメーターを使用します。STAT=ASIS を使用した場合、RACF メッセージおよび統計は、インストール済み環境における RACF **SETROPTS** コマンド上の現行のオプションによって制御されます。

RACF 統計を使用可能にした後、統計は、RACF によってシステム管理機能 (SMF) データ・セットまたはログ・ストリームに 1 日 1 回まで記録されます。RACF 統計の記録に使用される SMF データ・セットまたはログ・ストリームは、RACF 構成で指定されます。

### 関連タスク

183 ページの『IMS Connect に対する RACF セキュリティー統計の使用可能化』

IMS Connect が RACF を呼び出すように構成されている場合、IMS Connect が RACF 呼び出し **RACROUTE REQUEST=VERIFY** を発行するときに、RACF セキュリティー統計が記録され、更新されるようにすることができます。IMS DB への ODBM クライアント接続および IMS TM への OTMA クライアント接続の RACF 統計が記録され、更新されるようにすることができます。RACF 統計を使用可能にした後は、統計は 1 日 1 回まで更新されます。

### 関連資料

[ODACCESS ステートメント \(システム定義\)](#)

[QUERY IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

[UPDATE IMSCON TYPE\(CONFIG\) コマンド \(コマンド\)](#)

## OTMA 用の RACF ACEE のリフレッシュ

OTMA セキュリティーが使用可能に設定されている場合、パフォーマンス上の目的で、OTMA は RACF ユーザー・プロファイルを RACF アクセサー環境エレメント (ACEE) としてメモリーにキャッシュします。

### このタスクについて

その後、この ACEE は、後続の RACF の呼び出しで、入力メッセージ内の IMS コマンドまたは要求された IMS トランザクションに対するユーザー ID の権限を判別するために使用されます。ACEE を使用すると、RACF に対する入出力が少なくなります。

DASD 上の RACF データベース内のユーザー・プロファイルが変更されても、IMS が RACF イベント通知機能 (ENF) に登録できない場合、その変更はキャッシュに入れられた ACEE では反映されません。また、オンライン・メモリー内では、ACEE がリフレッシュされるまでユーザーの古いアクセス特権が変更されない場合があります。

### 手順

OTMA によってキャッシュに入れられた RACF ACEE は、以下のいずれかの方法でリフレッシュすることができます。

- z/OS ENF が IMS に変更を通知するときに OTMA によって自動的にリフレッシュします。

DASD 上の RACF データベース内のユーザー・プロファイルが変更されると、ENF は ENF イベント・コード 71 で IMS に変更を通知します。OTMA は、通知を受信すると、変更されたユーザー ID の ACEE をリフレッシュします。

IMS は、イベント・コード 71 通知を受け取るために、始動時に ENF に自動的に登録します。何らかの理由で ENF 登録が失敗した場合、IMS はメッセージ DFS3525E を発行し、OTMA は、RACF で変更が行われたときに ACEE を自動的にリフレッシュできません。

IMS が ENF に登録できない場合でも、確実に、キャッシュされた ACEE が最新の RACF セキュリティー定義を反映するように、OTMA クライアントは、キャッシュされた ACEE の自動リフレッシュの間隔の時間の長さを定義するエージング値を指定することができます。

IMS の始動時に ENF 登録が失敗し、お客様の RACF セキュリティー管理者が RACF での OTMA に対するユーザーのアクセス権を変更した場合は、**/SECURE OTMA REFRESH** コマンドを発行して、そのユーザー ID の ACEE のリフレッシュが必要な場合があります。リフレッシュしない場合は、ユーザーの古いアクセス権は、ACEE エージング限度に達するまでオンライン・メモリー内に未変更のまま残ります。

- OTMA ACEE のエージング値を指定することができます。この OTMA ACEE エージング値に到達すると、OTMA は、OTMA クライアントから受け取る次の入力メッセージを処理する前に ACEE をリフレッシュします。

IMS Connect が OTMA に渡す ACEE エージング値を定義するには、IMS Connect PROCLIB メンバー HWSCFGxx の DATASTORE 構成ステートメント内の OAAV= キーワードを使用します。

IBM MQ の ACEE エージング値を定義するには、CSQ6SYSYP マクロで OTMACON= パラメーターを使用します。

すべての OTMA クライアントのグローバル ACEE エージング値を定義するには、**/SECURE OTMA ACEEAGE** コマンドを発行します。**/SECURE OTMA ACEEAGE** コマンドを発行して OTMA クライアントのグローバル ACEE エージング値を定義する場合、指定したエージング値によって、すべての OTMA クライアントの既存の ACEE エージング値がオーバーライドされます。**/SECURE OTMA ACEEAGE** コマンドで **TMEMBER** パラメーターを指定する場合、定義した ACEE エージング値によって、ユーザーが指定した OTMA クライアントの既存の値 (ある場合) がオーバーライドされます。

キャッシュに入れられた各 OTMA ACEE には、独自のエージング値があります。この値は、その ACEE にアクセスする、最低のエージング値を持つ OTMA クライアント (TMEMBER) に基づきます。例えば、IBM MQ がその ACEE のエージング値を 5 日に設定し、IMS Connect がその ACEE のエージング値を 1 日に設定しているとします。IBM MQ のみが使用する ACEE のエージング値は 5 日です。IMS Connect のみが使用する ACEE のエージング値は 1 日です。IBM MQ と IMS Connect の両方が ACEE を使用する場合、そのエージング値は 1 日です。これは、5 日と 1 日の間の最低の値です。

ACEE 有効期限値は、クライアント・ビッド時間で指定されます。

- **/SECURE OTMA REFRESH** コマンドを発行して、手動で OTMA ACEE をリフレッシュすることができます。

すべての OTMA ACEE をグローバルにリフレッシュするには、**/SECURE OTMA REFRESH** コマンドを発行します。個々の OTMA クライアントの ACEE をリフレッシュするには、**/SECURE OTMA REFRESH TMEMBER tmember\_name** コマンドを発行します。特定のユーザー ID の ACEE だけをリフレッシュするには、**/SECURE OTMA REFRESH USER user\_id** コマンドを発行します。

- **/STOP** コマンドおよび **/START OTMA** コマンドを発行して、ACEE テーブルを再作成することもできます。

### 関連タスク

[818 ページの『OTMA ACEE フラッシュ制御』](#)

OTMA アクセス機能環境エレメント (ACEE) フラッシュ制御機能は、IMS 制御領域の仮想記憶域が使い尽くされるのを防止します。

### 関連資料

[/SECURE コマンド \(コマンド\)](#)

[DFSOTMA 記述子の構文およびパラメーター \(システム定義\)](#)

[IMS PROCLIB データ・セットの HWSCFGxx メンバー \(システム定義\)](#)

## OTMA メッセージ接頭語でのセキュリティの指定

OTMA メッセージ接頭語でのセキュリティの指定は、セキュリティ・データ・セクションに入っています。セキュリティ・データ・セクションは、各トランザクションあるいはコマンドで必須で、OTMA プロトコル・コマンドではオプションです。

### このタスクについて

OTMA メッセージ接頭語で、セキュリティについて以下の指定ができます。

- このメッセージに使用する、以下の RACF セキュリティ・レベル:

- FULL
- CHECK
- NONE

- ユーザー・トークン
- ユーザー ID
- SAF プロファイル
- ネットワーク・ユーザー ID
- ネットワーク・セッション ID

クライアント・ビッド要求の状態データの中で、アクセサー環境エレメント (ACEE) のエージング間隔を指定できます。ACEE エージング間隔は、ACEE の、リフレッシュされるまでに使用できる時間量を表しています。入力メッセージが OTMA によって受信されたときにエージング間隔が満了している、OTMA はメッセージの妥当性を検査する前に、ACEE をリフレッシュします。

### 関連資料

[889 ページの『OTMA メッセージ接頭語』](#)

OTMA メッセージには、IMS.ADFSMAC データ・セットの DFSYMSG マクロによってマップされるフォーマットに準拠した接頭部があります。

## OTMA で使用される RACF セキュリティー・クラス

OTMA を通じて IMS にサブミットされるトランザクションまたはコマンドを RACF で保護するためには、RACF に対し、それらのトランザクションやコマンドに適用するセキュリティを定義する必要があります。

トランザクション・セキュリティ定義は、RACF TIMS クラスに保管されます。コマンド・セキュリティ定義は、CIMS クラスに保管されます。トランザクションが TIMS クラスに入っていないか、コマンドが CIMS クラスに入っていない場合でも、指定するオプションに関係なく、**/SECURE OTMA** コマンドを使用することにより、そのトランザクションまたはコマンドを使用できます。

OTMA コマンドを入力すると、OTMA は、コマンドを妥当性検査するために RACHECK を発行します。OTMA は、検査のために DFSCCMD0 にコマンド verb だけを渡し、CVB 制御ブロック全体は渡しません。

非同期保留キューを RESUME TPIPE 呼び出しの無許可ユーザーから保護するために RACF を使用している場合は、デフォルト RIMS リソースまたは Rxxxxxxx リソース・クラスを定義する必要があります。ここで、xxxxxxx は、DFSDCxxx PROCLIB メンバー内の RCLASS パラメーターの値です。定義する RACF リソース・クラスには、保護されている非同期保留キューの名前と、各キューへのアクセスを許可されているユーザー ID が含まれている必要があります。

## 分散ネットワーク・セキュリティ資格情報のサポートおよび OTMA

OTMA は、分散環境でユーザーによって入力されたセキュリティ資格情報をサポートします。分散ネットワーク・セキュリティ資格情報が OTMA に渡された後、その資格情報は、メッセージ接頭部に関する情報が入った IMS ログ・レコードに入れられます。

分散ネットワーク・セキュリティ資格情報には、ネットワーク・ユーザー ID とネットワーク・セッション ID を含めることができます。

### ネットワーク・ユーザー ID

ユーザーの分散 ID。ネットワーク・ユーザー ID の最大長は 246 バイトです。IMS TM リソース・アダプターのユーザーの場合、ネットワーク・ユーザー ID は X.500 シリーズの標準の識別名 (DN) です。

### ネットワーク・セッション ID

分散ユーザーのセッション ID。ネットワーク・セッション ID の最大長は 254 バイトです。IMS TM リソース・アダプターのユーザーの場合、ネットワーク・セッション ID はドメイン・ネーム、レルム、またはレジストリー名です。

ネットワーク・セキュリティ資格情報は、以下のいずれかのユーザー・メッセージ出口を使用する分散環境内のアプリケーションから IMS に渡すことができます。

- HWSSMPL0
- HWSSMPL1
- HWSJAVA0

**制約事項:** DataPower、IMS Connect API、および SOAP Gateway クライアントからの分散ネットワーク・セキュリティ資格情報は、IMS Connect によってサポートされません。

ユーザー・メッセージ出口 HWSSMPL0 または HWSSMPL1 を使用するユーザー作成のアプリケーションからネットワーク・セキュリティ資格情報を渡せるようにするには、IRM 拡張を定義する必要があります。この IRM 拡張を使用して、ユーザーが入力したネットワーク・セキュリティ資格情報を IMS Connect に渡します。ID \*NETUID\* を持つ IRM 拡張はネットワーク・ユーザー ID を渡すために使用され、ID \*NETSID\* を持つ IRM 拡張はネットワーク・セッション ID を渡すために使用されます。ネットワーク・セキュリティ資格情報が IMS Connect に渡された後、ユーザー・メッセージ出口 HWSSMPL0 および HWSSMPL1 ルーチンは、IRM 拡張内のネットワーク・セキュリティ資格情報を使用して、OTMA メッセージ接頭部を作成します。セキュリティ資格情報は、OTMA メッセージ接頭部のセキュリティ・データ・セクションに含まれます。

HWSJAVA0 ユーザー・メッセージ出口を使用するアプリケーションに対してユーザーがネットワーク・セキュリティ資格情報を入力すると、その資格情報はアプリケーションから IMS TM リソース・アダプター

に渡されます。その後、IMS TM リソース・アダプター は OTMA メッセージ接頭部のセキュリティー・データ・セクションに資格情報を組み込みます。

分散ネットワーク・セキュリティー資格情報は、OTMA メッセージ接頭語のセキュリティー・データ・セクションに入れられて IMS に渡されるため、メッセージ接頭語に関する情報が入っているすべての IMS ログ・レコード (ログ・レコード X'01' および X'03' など) に分散ネットワーク・セキュリティー資格情報が入っています。

高速機能メッセージにネットワーク・セキュリティー資格情報が含まれていて、そのメッセージがローカル IMS システム上で高速機能の急送メッセージ・ハンドラー (EMH) によって処理される場合、その資格情報は X'5901' ログ・レコードに記録されます。

ネットワーク・セキュリティー資格情報が含まれる高速機能メッセージがフロントエンド IMS システムで共用キュー環境の EMH キュー (EMHQ) を使用して処理される場合、その資格情報は X'5911' ログ・レコードに入れられます。処理を実行する IMS システムであるバックエンド IMS システムでは、資格情報は X'5901' ログ・レコードに入れられます。

分散ネットワーク・セキュリティー資格情報が OTMA に渡された後、IMS がその資格情報を RACF SMF レコードに記録できるようにすることができます。分散ネットワーク・セキュリティー資格情報を RACF SMF レコードに記録できるようにするには、IMS PROCLIB データ・セットの DFSYDTx メンバーの OTMA クライアント記述子に LOGSTR=YES を指定します。LOGSTR=YES を指定した後、OTMA に送信される分散ネットワーク・セキュリティー資格情報の最初の 255 バイトが RACF SMF レコードに記録されます。

IMS OTMA 呼び出し可能インターフェース (OTMA C/I) の `otma_send_receivev` API および `otma_send_asyncx` API を使用して、ネットワーク・ユーザー ID およびネットワーク・セッション ID を IMS に渡すことができます。各 API で、ネットワーク・ユーザー ID の最大 100 バイト、およびネットワーク・セッション ID の最大 100 バイトを IMS に渡すことができます。

また、トランザクション許可出口ルーチン (DFSCTRNO) を使用して、OTMA メッセージ接頭部内のネットワーク・セキュリティー資格情報のアドレスを渡すこともできます。

以下の OTMA ユーザー出口ルーチン (OTMA メッセージ接頭部のセキュリティー・データ・セクションのアドレスが含まれます) を使用すると、OTMA 入力メッセージ内のネットワーク・セキュリティー資格情報が IMS に渡された場合に、その資格情報にアクセスすることができます。

- DFSYIOE0
- DFSYPRX0
- DFSYDRU0

#### 関連資料

[otma\\_send\\_asyncx API \(アプリケーション・プログラミング API\)](#)

[otma\\_send\\_receivev API \(アプリケーション・プログラミング API\)](#)

[914 ページの『セキュリティー・データ・セクション』](#)

セキュリティー・データ・セクションは、各トランザクションあるいはコマンドで必須で、OTMA プロトコル・コマンドではオプションです。

[916 ページの『OTMA セキュリティー・データ・フィールドの説明』](#)

以下に、メッセージ接頭語のセキュリティー・データ・セクションの内容について、さらに詳しい情報を提供します。

[OTMA 入出力編集ユーザー出口 \(DFSYIOE0 およびその他の OTMAIOED タイプの出口\) \(出口ルーチン\)](#)

[OTMA 宛先解決ユーザー出口 \(DFSYPRX0 およびその他の OTMAYPRX タイプの出口\) \(出口ルーチン\)](#)

[OTMA ユーザー・データ・フォーマット設定出口ルーチン \(DFSYDRU0\) \(出口ルーチン\)](#)

[トランザクション許可出口ルーチン \(DFSCTRNO\) \(出口ルーチン\)](#)

[OTMA クライアント記述子の構文およびパラメーター \(システム定義\)](#)

## 非同期保留キュー内のメッセージの保護

非同期保留キューに入っているメッセージを RESUME TPIPE 呼び出しの無許可使用から保護するためには、RACF、OTMA RESUME TPIPE セキュリティー ユーザー出口 (OTMARTUX)、またはその両方を使用する必要があります。

### このタスクについて

T パイプ保留キューに対してセキュリティが使用可能である場合、RESUME TPIPE 呼び出しを発行するユーザー ID は、OTMA クライアントにメッセージを送信するためには、RESUME TPIPE 呼び出しメッセージに含まれている TPIPE 名へのアクセスを事前に許可されている必要があります。

RACF が行うセキュリティ検査および OTMARTUX ユーザー出口が行うセキュリティ検査は任意です。これらは、組み合わせて使用することも、単独で使用することもできます。RACF と OTMARTUX の両方を使用する場合は、OTMARTUX ユーザー出口に制御が与えられる前に、最初に RACF が呼び出されます。この場合、OTMARTUX ユーザー出口は、必要に応じて RACF をオーバーライドすることができます。

### RACF を使用した非同期保留キューの保護

非同期保留キューに対して RACF セキュリティー検査が有効にされている場合、許可ロジックはセキュリティ・ヘッダーを検証して、TPIPE 名の下のユーザー ID を許可します。

### このタスクについて

RESUME TPIPE 呼び出しの受信時に、RESUME TPIPE リソース・クラス (RIMS または Rxxxxxxx) が存在し、呼び出しで指定された T パイプ名がそのリソース・クラスに定義されている場合にのみ、RACF セキュリティー検査が実行されます。

RACF セキュリティーが有効かどうかに関係なく、OTMA RESUME TPIPE セキュリティー ユーザー出口 (OTMARTUX) を使用できます。OTMARTUX ユーザー出口と RACF を一緒に使用した場合は、常に RACF セキュリティーが最初に呼び出されます。その場合、OTMARTUX ユーザー出口は RACF プロシージャの結果をオーバーライドできます。

RESUME TPIPE セキュリティーを有効にするには、以下の手順を実行します。

### 手順

1. RESUME TPIPE リソース・クラス (RIMS または Rxxxxxxx) がまだ存在しない場合、SECURITY システム定義マクロで RCLASS キーワードを使用して定義します。  
IMS の始動時に、RESUME TPIPE リソース・クラスが IMS に対して定義されていない場合、IMS はメッセージ DFS3187I を発行します。IMS の稼働後は、RESUME TPIPE リソース・クラスが存在しないことを警報するそれ以上の警告は出されません。
2. RESUME TPIPE リソース・クラスに、保護対象の非同期保留キューの T パイプ名と、キューへのアクセスが許可されるユーザー ID を指定します。
3. OTMASE 始動パラメーターまたは **/SECURE OTMA** コマンドを使用して、OTMA 用の適切なレベルの RACF セキュリティーを指定します。  
指定できる OTMA セキュリティーの適切なレベルは、FULL、CHECK、または PROFILE です。PROFILE のセキュリティ・レベルが指定されている場合、RESUME TPIPE 要求メッセージは FULL または CHECK を指定する必要があります。
4. RACF 保護の非同期保留キューにアクセスする、RESUME TPIPE 要求メッセージをコーディングします。RESUME TPIPE 要求メッセージには、以下を含める必要があります。
  - OTMA 接頭部の制御データ・セクション内の T パイプ名
  - OTMA 接頭部のセキュリティ・セクション内のユーザー ID
  - OTMA セキュリティー・レベルが PROFILE の場合、FULL または CHECK のセキュリティ・フラグ設定



## 関連情報

[DFS3187I \(メッセージおよびコード\)](#)

## OTMA RESUME TPIPE セキュリティー・ユーザー出口 (OTMARTUX) を使用した非同期保留キューの保護

OTMA RESUME TPIPE セキュリティー・ユーザー出口 (OTMARTUX) は、OTMA 非同期保留キューに入っているメッセージを取り出すために発行される RESUME TPIPE 呼び出しに対して、2 つの可能なセキュリティー層のうちの 1 つを提供します。

### このタスクについて

RESUME TPIPE 呼び出しのセキュリティーが使用可能に設定されている場合、OTMARTUX ユーザー出口は、RESUME TPIPE 呼び出しが開始されたとき、保留キューからメッセージを取り出す前に呼び出し側の権限を検査します。

OTMARTUX ユーザー出口は、RESUME TPIPE 呼び出しに対する RACF セキュリティー検査と一緒に使用することも、単独で使用することもできます。OTMARTUX ユーザー出口と RACF を一緒に使用した場合は、常に RACF セキュリティーが最初に呼び出されます。その場合、OTMARTUX ユーザー出口は RACF プロシージャの結果をオーバーライドできます。

RESUME TPIPE 呼び出しのセキュリティーが RACF と OTMARTUX ユーザー出口の両方によって適用される場合、OTMARTUX ユーザー出口は、RACF セキュリティー・プロシージャの成功または失敗に関係なく起動されます。OTMARTUX ユーザー出口は、RACF セキュリティー検査の結果を受け入れるか、結果をオーバーライドするか、より制限の強いセキュリティー規則を適用することができます。より制限の強い規則の例としては、その日の特定の時間内だけ出力メッセージにアクセスできる許可をユーザーに与えることなどが考えられます。

許可が成功した場合は、保留キュー内の出力メッセージが IMS Connect に返信されます。

許可が失敗した場合は、RESUME TPIPE 呼び出しの拒否メッセージ (NAK) がクライアントへ送信されます。

OTMARTUX ユーザー出口をバイパスするには、RESLIB に DFSYRTUX が含まれていないことを確認し、xxx メンバーの USER\_EXITS セクションに OTMARTUX ユーザー出口タイプに対する EXITDEF ステートメントを定義しないでください。

### 関連資料

[OTMARTUX: OTMA RESUME TPIPE セキュリティー・ユーザー出口 \(DFSYRTUX およびその他の OTMARTUX タイプの出口\) \(出口ルーチン\)](#)

## OTMA IMS 間 TCP/IP 接続のセキュリティー

IMS 間 TCP/IP 接続上のリモート IMS システムに送信される OTMA ALTPCB メッセージの場合、トランザクション許可はリモート IMS システムによって実行されます。

ローカル IMS システムで実行されているアプリケーション・プログラムが ISRT ALTPCB 呼び出しを発行してリモート・システムにメッセージを送信する場合、メッセージの接頭部にアプリケーション・プログラムのユーザー ID が組み込まれます。

IMS.PROCLIB データ・セットの OTMA DFSYDTx メンバー内の OTMA 宛先記述子で、ユーザー ID を指定することもできます。ユーザー ID が OTMA 宛先記述子で指定されている場合、リモート IMS システムは、ISRT 呼び出しを発行したアプリケーション・プログラムのユーザー ID の代わりに、OTMA 宛先記述子内のユーザー ID を使用します。

IMS Connect に RACF パスチケット・ユーザー認証を実装することで、TCP/IP 接続をセキュアにすることができます。IMS Connect は、接続が最初に確立されたときに、RACF パスチケットを認証します。持続ソケットが使用されている場合、初期認証が実行された後は、その接続で受信するメッセージはすべてトラステッド・ユーザーから来たものとして扱われ、接続が持続している限り、それ以上の認証は行われません。

### 関連タスク

188 ページの『IMS 間 TCP/IP 接続の保護』

IMS 間 TCP/IP 接続を保護するために、IMS Connect は RACF パスケットを使用して、IMS Connect の 1 つのインスタンスを IMS Connect のもう 1 つ別のインスタンスのトラステッド・ユーザーとして設定します。

IMS 間の TCP/IP 接続 (システム定義)

**関連資料**

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

## 一般的な OTMA のセキュリティ考慮事項

OTMA に関するいくつかの一般的なセキュリティ考慮事項があり、それらを把握しておく必要があります。

- セキュリティーのために RACF (または同等の製品) を使用する場合は、FACILITY クラスに IMSXCF.group.client\_member\_name を定義してください。

FACILITY クラスに IMSXCF.group.client\_member\_name を定義した場合、そして IMS セキュリティーが NONE に設定されていない場合は、クライアント・ビッド要求のユーザー・トークンは有効でなければならず、ユーザーは、FACILITY クラスに READ アクセスを持つ必要があります。

クライアント・ビッド要求のユーザー・トークンが RACF 検査に失敗すると、クライアントは、サーバーから NAK メッセージを受信します。

- z/OS の z/OS システム間カップリング・ファシリティ クライアントを許可してください。
- 完全セキュリティが定義されている OTMA アプリケーションの場合、セキュリティ環境は、アプリケーションが終了するまで保持されます。
- IMS が、OTMA からメッセージを受信した後、OTMA セキュリティーがアクティブであれば、IMS は RACF を呼び出し、着信メッセージのユーザー ID が有効な RACF ユーザー ID であるかどうかを検査します。IMS にはユーザー ID のパスワードが渡されないため、RACF に対する呼び出しはユーザー ID だけを検査します。パスワードの妥当性検査が必要な場合は、メッセージを IMS へ送信する前に、パスワードの妥当性を検査する必要があります。
- IMS は、RACF に対する呼び出しの入力メッセージの UTOKEN を使用することで、ユーザー ID を検査するだけでなく、検査済みユーザー ID のそれぞれを表示する IMS 制御領域内のセキュリティ管理ブロックも作成します。検査済み RACF ユーザー ID を表す、IMS 制御領域に作成されたセキュリティ管理ブロックは、アクセサー環境エレメントまたは ACEE と呼ばれます。
- DL/I ICAL 呼び出しには、OTMA セキュリティー構成に関連した特殊なセキュリティ要件があります。同期プログラム間通信要求処理に使用される DFSYICAL T メンバーは、OTMA が使用可能になっていない場合でも OTMA セキュリティー構成の設定値を使用します。
- 分散ネットワーク・セキュリティ資格情報 (ネットワーク・ユーザー ID およびネットワーク・セッション ID を含む) が OTMA メッセージ接頭部のセキュリティ・データ・セクションで IMS に渡されると、その資格情報は、IMS ログ・レコード (X'01' および X'03' など) に組み込まれます。分散ネットワーク・セキュリティ資格情報を RACF SMF レコードに記録できるようにするには、IMS PROCLIB データ・セットの DFSYDTx メンバーの OTMA クライアント記述子に LOGSTR=YES を指定します。LOGSTR=YES を指定した後、OTMA に送信される分散ネットワーク・セキュリティ資格情報の最初の 255 バイトが RACF SMF レコードに記録されます。

**関連概念**

836 ページの『[分散ネットワーク・セキュリティ資格情報のサポートおよび OTMA](#)』

OTMA は、分散環境でユーザーによって入力されたセキュリティ資格情報をサポートします。分散ネットワーク・セキュリティ資格情報が OTMA に渡された後、その資格情報は、メッセージ接頭部に関する情報が入った IMS ログ・レコードに入れられます。

IMS セキュリティー (システム管理)

**関連資料**

ICAL 呼び出し (アプリケーション・プログラミング API)

## OTMA 環境での DL/I 呼び出しの使用

一部の DL/I 呼び出しには、OTMA で使用する場合に特別な考慮事項があります。

### CHNG

CHNG 呼び出しが、OTMA がサブミットしたトランザクションから出される場合は、宛先は同じ OTMA クライアント (T パイプ名が CHNG 呼び出しによってセットされる) であると想定されます。この行動は、OTMA 事前経路指定および宛先解決出口ルーチンで変更できます。

代替 PCB に対する CHNG 呼び出しを発行する IMS アプリケーション・プログラム (オプション・リストを指定) は、宛先を判別するために、IMS に OTMA 事前経路指定および宛先解決出口ルーチンを呼び出させることはありません。しかし、代替 PCB (APPC 記述子を指定する) への CHNG 呼び出しを発行する IMS アプリケーション・プログラムは、IMS に OTMA 出口ルーチンを呼び出して、宛先を決定させます。

このアプリケーション・プログラムは、入出力 PCB に対する ISRT 呼び出しを発行して OTMA の宛先にデータを送信することもできます。

OTMA アプリケーション・プログラムは、APPC の宛先に対する CHNG および ISRT 呼び出しを使用することができます。

### INQY (ヌル)

OTMA 宛先に発行した INQY 呼び出しは、以下の情報を戻します。トランザクション・パイプ名、クライアント z/OS システム間カップリング・ファシリティメンバー名、ユーザー ID、グループ名、および同期レベル。

### ICAL

IMS アプリケーション・プログラムは、ICAL 呼び出しを発行して、同期コールアウト要求を IMS インストール済み環境の外部にあるデータ・プロバイダーまたはサービス・プロバイダーへ送信します。OTMA は、ICAL 呼び出しによって開始されたコールアウト要求を、保留キューに対応した OTMA クライアントに経路指定し、その応答を、待機中の IMS アプリケーション・プログラムに逆に経路指定します。OTMA は、同期コールアウト要求の OTMA 宛先記述子の中にタイムアウト値を指定する機能を備えています。このタイムアウト値は、ICAL 呼び出しの中で指定するタイムアウト値によってオーバーライドすることができます。

ICAL 呼び出しを使用して、同期プログラム間通信要求を出すこともできます。OTMA 宛先記述子が TYPE=IMSTRAN で構成されている場合、OTMA は制御を別の IMS アプリケーションに切り替え、待機中の IMS アプリケーション・プログラムに送信するための応答を待ちます。

OTMA クライアントからのトランザクションのメッセージ接頭部に分散ネットワーク・セキュリティー資格情報が含まれている場合、セキュリティー資格情報は、ICAL 呼び出しによって開始された同期コールアウト要求で IMS に渡すことができます。分散ネットワーク・セキュリティー資格情報は、IMS 要求メッセージ (IRM) 接頭部で以下のフィールドを指定して RESUME TPIPE 呼び出しが定義されている場合にのみ、ICAL 呼び出しによって IMS から渡されます。以下のフィールド指定が定義されていない場合、IMS は、同期コールアウト要求における OTMA メッセージ接頭部のセキュリティー・データ・セクションから分散ネットワーク・セキュリティー資格情報を除去します。

#### IRM\_ARCH

X'05' (IRM\_ARCH5)

#### IRM\_F6

X'80' (IRM\_F6\_NWSE)

### PURG

IMS アプリケーション・プログラムが PURG 呼び出しを発行すると、IMS は、OTMA 事前経路指定および宛先解決出口ルーチンを呼び出して宛先を判別します。

### SETO

SETO 呼び出しを発行する IMS アプリケーション・プログラムでは、IMS が、宛先の判別のために、OTMA 事前経路指定および宛先解決出口ルーチンを呼び出すことはありません。

SETO 呼び出しを発行する既存の IMS アプリケーション・プログラムは、要求どおりに実行されないことがあります。これは、プログラムが OTMA によって発信されたトランザクションを処理する場合、

プログラムに戻りコードが返されるためです。SETO 呼び出しを発行する APPC/IMS アプリケーション・プログラムは、暗黙の OTMA サポートが必要な場合、修正を必要とする場合があります。

これらのアプリケーション・プログラムが働くようにするには、INQY 呼び出しを使用してから、SETO 呼び出しを発行します。アプリケーション・プログラムが、INQY 呼び出しの出力を使用してトランザクションが OTMA クライアントから発信されたかどうかを決定できる場合は、SETO 呼び出しは発行しません。

IMS が OTMA 出口ルーチン呼び出すようにする、DL/I 呼び出しの場合、IMS は、宛先が (例えば、他の DL/I 呼び出しによって) セットされていないならば、出口ルーチン呼び出すだけです。

保護会話 (z/OS リソース・リカバリー・サービス 環境内の 1 つのリカバリー単位で複数のリソース・マネージャーのリソースにアクセスするなど) を開始するには、クライアント・アダプター・コード (OTMA ユーザー) は、専用コンテキストを獲得して所有し、メッセージ接頭部の状態データ・セクションにコンテキスト ID を提供する必要があります。

**定義:** コンテキストとは、リソース・マネージャーが仕事をする z/OS エンティティーで、この環境では専用コンテキストが要求されます。

IMS とクライアント間のメッセージ・トラフィック時に、メッセージ・ヘッダーのコンテキスト ID フィールドがゼロでない場合、保護会話処理が発生します。

### 関連タスク

[RESUME TPIPE による同期コールアウト要求のリトリブ \(コミュニケーションおよびコネクション\)](#)

### 関連資料

[トランザクション管理 \(アプリケーション・プログラミング API\)](#)

## OTMA プログラム間通信処理

OTMA では 2 タイプのメッセージ通信が出現します。それは、コミット後送信および送信後コミットです。

OTMA コミット後送信入力メッセージの場合 (非同期メッセージまたはコミット・モード 0 メッセージとも呼ぶ)、プログラム間通信は常に別のコミット・モード 0 (CM0) メッセージとなります。OTMA 送信後コミット入力メッセージの場合 (同期メッセージまたはコミット・モード 1 メッセージとも呼ぶ)、プログラム間通信は以下の条件によって結果が異なります。

- I/O PCB への ISRT 呼び出しがある
- 高速 PCB が交換のために使用される
- 複数プログラムへの交換がある
- IMS 始動パラメーター OTMAASY=Y または OTMAASY=S が指定されている
- トランザクションが保護されている

したがって、コミット・モード 1 (CM1) 入力メッセージの P2P 交換では、考えられるのは別の CM1 メッセージ、1 つの DFS2082 メッセージ、または 1 つの CM0 メッセージです。また、いくつかの OTMA クライアント (例えば、IBM MQ) は、CM1 入力メッセージに対して CM0 出力メッセージを受け入れることができます。しかし、別の OTMA クライアントは、そうでない場合があります。

以下のトピックでは、さまざまな送信後コミット・メッセージ通信の使用シナリオを示します。

## OTMA の単一ストリームのプログラム間通信

単一ストリームのプログラム間通信を以下の図に示します。

プログラム A はプログラム B に切り替わり、プログラム B はプログラム C に切り替わります。そして、I/O PCB へ挿入します。このプログラム・フローのモデルは、正常に送信後コミット出力メッセージを送信します。単一ストリームは、プログラム間通信が次々に発生することを意味しています。P2P メッセージ通信では、高速 PCB は使用されません。



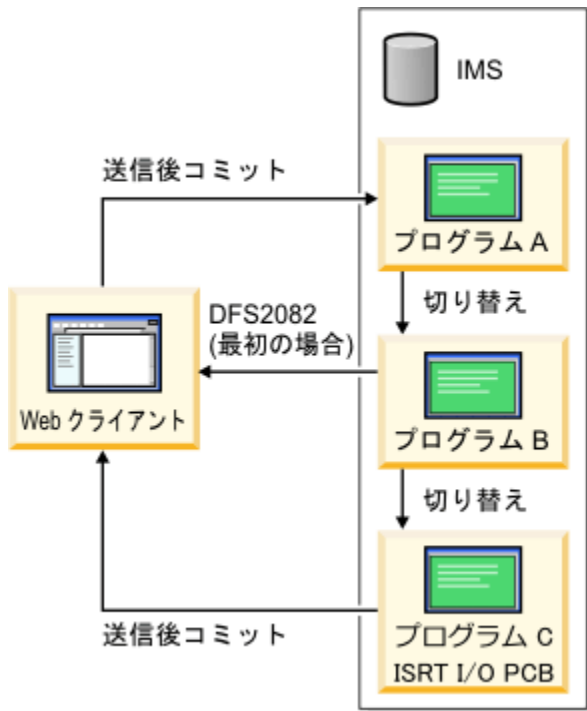


図 141. 単一ストリームのプログラム間通信

## I/O PCB への ISRT なしの OTMA プログラム間通信

複数の切り替えが順番に発生し、I/O PCB に何も戻して挿入することがない場合は、メッセージ DFS2082 が OTMA クライアントに送り返されます。

最後のプログラム間通信先、つまりプログラム C は、I/O PCB へ挿入して戻すことはしません。このため、IMS は OTMA クライアント用にメッセージ DFS2082 を生成します。



**重要:** プログラム C が MSC を経由したリモート IMS で実行され、I/O PCB へ戻って挿入されることがない場合は、リモート IMS はメッセージ DFS2082 を発行しません。ただし、この場合、OTMA クライアントのプログラムがハングして、フロントエンドの IMS 制御領域がその制御ブロックを構築することになる可能性があります。この種の構築は、ストレージ関連のシステム障害を招く可能性があります。IMS を再始動すると、制御ブロックが解放されます。

## 高速 PCB を使用する OTMA プログラム間通信

高速 PCB を使用する P2P メッセージ通信は、コミット後送信出力メッセージを 発行できます。

プログラム A は、非高速 PCB ではなく、高速 PCB を使用して P2P メッセージ通信を実行します。プログラム B からの出力はコミット後送信です。高速 PCB を使用すると、プログラム B は非同期的に処理されるからです。プログラムが非同期的に処理され、I/O PCB に挿入した場合は、出力メッセージはコミット後送信メッセージとして送信されます。ただし、プログラム A も非高速 PCB を使用してプログラム C に切り替わった場合は、プログラム C は I/O PCB へ挿入します。C からの出力は送信後コミット・メッセージとなります。

## 複数プログラムへの OTMA プログラム間通信

プログラムが I/O PCB に戻って挿入されると、プログラム間で他のメッセージ通信がある場合は、非同期的に処理されます。

例えば、プログラム A はプログラム B に切り替わり、プログラム B は I/O PCB に戻って挿入されます。プログラム B からの出力は送信後コミット・メッセージとなります。その後、プログラム B はプログラム C に切り替わり、非同期的に処理されます。

「レース」条件は、プログラム間通信が複数プログラムに切り替わる時に発生します。プログラム A は、非高速 PCB を使用して、複数プログラムに切り替わります。最初にスケジュールされた切り替えプログラ

ムだけが、同期的に処理されます。他の切り替えプログラムは非同期的に処理されます。同期的に処理されたプログラムが I/O PCB に挿入した場合、出力メッセージは送信後コミット・メッセージです。

複数プログラムの 1 つが、リモート・プログラムになる可能性もあります。このプログラム・フローを以下の図に示します。プログラム A は、MSC を介してリモート・プログラム B へ切り替わります。最初に、プログラム B は、ローカル IMS で新しいプログラムであるプログラム C を起動します。次に、ローカル IMS を経由して OTMA クライアントに応答を挿入します。ローカル IMS で最初に発生すること (プログラム C のスケジューリングまたは OTMA クライアントに対する応答の処理) に応じて、不要な DFS2082 メッセージがクライアントに送信される場合があります。これも、レース条件です。プログラム C がローカル IMS で最初に処理される場合は、DFS2082 メッセージが送信されます。応答が最初に処理される場合は、プログラム B からの予期出力は、送信後コミットを使用して同期的に送信されます。

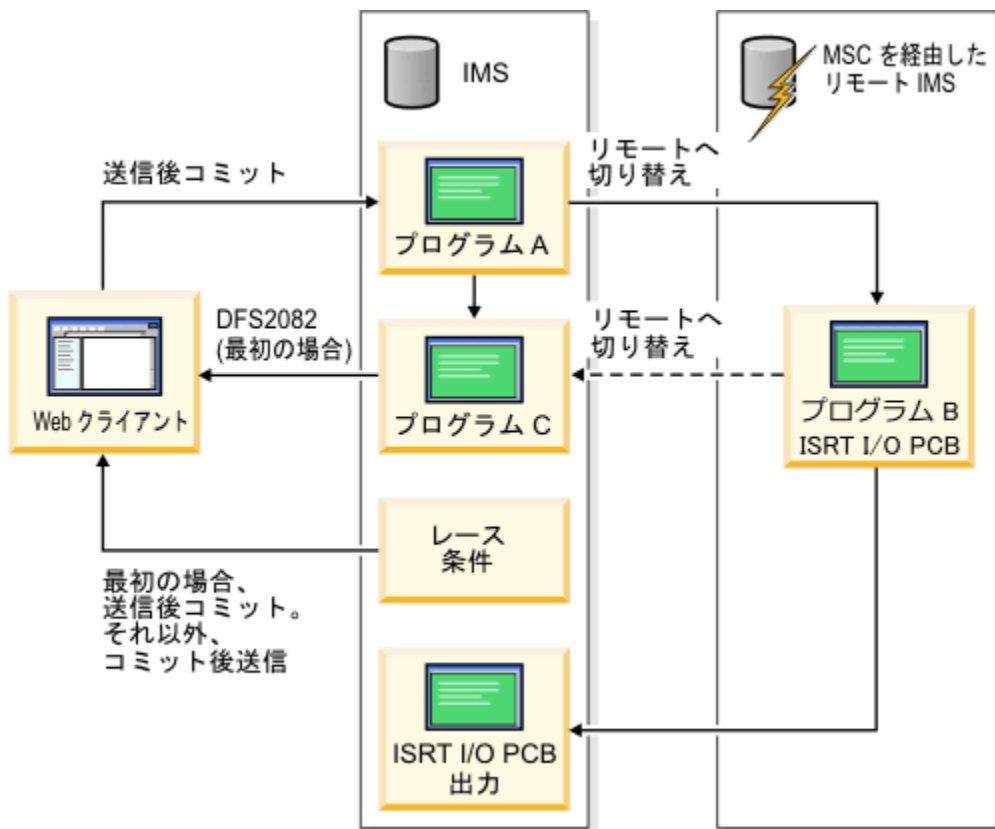


図 142. 複数プログラムへのプログラム間通信の結果として生じるレース条件

これらの環境でレース条件を作成することを回避するには、以下のうちいずれかを実行してください。

- プログラムを変更して、送信後コミット (CM1) に対する同一トランザクション内での複数のプログラム間通信を回避する。
- 同じトランザクション内で複数のプログラム間通信を実行しているときに、コミット後送信 (CM0) 入力を使用する。
- IMS 始動パラメーター OTMAASY を使用して、P2P メッセージ通信処理をシリアライズする。

レース条件を回避するために OTMAASY パラメーターを使用すると、単一ストリーム・モデルに類似したプログラム間通信モデルを作成することができます。例えば、前の図では、応答モード・トランザクションが同期的に処理されるプログラム B は、送信後コミット (同期) 出力メッセージを送信できます。非応答モード・トランザクションで実行しているプログラム C は、メッセージを非同期的に処理します。

#### 関連タスク

795 ページの『プログラム間通信出力メッセージの非同期送達の設定』

複数のプログラム間通信を開始する送信後コミット (CM1) メッセージを使用している場合、適切な出力のみが同期 CM1 モードで OTMA クライアントに返されるようにするには、OTMAASY=Y を指定します。

#### 関連資料

[IMS プロシージャのパラメーターの説明 \(システム定義\)](#)

## 保護トランザクションの OTMA プログラム間通信

保護トランザクションに対して P2P メッセージ通信を実行する 非会話型プログラムの場合、ABENDU0711 が理由コード 1D で戻されます。会話型プログラムの場合、プログラムは X6 状況コードを受信します。

## その他の OTMA プログラム間通信の考慮事項

以下の考慮事項は、プログラム間通信にも適用されます。

- P2P メッセージ通信は、OTMA 保護メッセージ (synclevel = SyncPt の送信後コミット入力) に対してサポートされません。
- 非会話型プログラムが、共用キュー環境のプログラムに対してプログラム間メッセージ通信を実行する場合、SQ (共用キュー) 環境のプログラムは、最初のプログラムがスケジュールされたときと同じ IMS 上で稼働している必要があります。ただし、同期 APPC/OTMA に対するサポートがアクティブ (AOS=Y) で、かつ IMS 始動パラメーター OTMAASY=S が指定されている場合は除きます。
- 入力会話型トランザクションが発生した場合、会話の継続へのメッセージ通信のみが同期的にスケジュールされます。それ以外のすべてのトランザクションは非同期にスケジュールされます。
- 同期 APPC/OTMA サポート (DFSDCxxx PROCLIB メンバーに対する AOS=Y) と z/OS リソース・リカバリ・サービス (RRS) サポート (始動プロシージャに対する RRS=Y) の両方が使用可能な共用キュー環境では、他の IMS システムとのアウトバウンド APPC 保護会話を開始するバックエンド IMS システム上で実行されるアプリケーション・プログラムは、同じ宛先トランザクションへのプログラム間通信が 1 回 のみに制限されます。

アプリケーション・プログラムで、APPC アウトバウンド保護会話を別の IMS システム上に割り振った後、複数のプログラム間通信を実行した場合、結果は予測不能となり、MPP 従属領域内で WAIT-RRS/PC 状態が発生する可能性があります。





## 第 44 章 OTMA クライアント

OTMA 環境には、サーバーと 1 つ以上のクライアントが含まれます。

この章では、IMS トランザクションを処理するクライアントとサーバーの相互作用の方法について説明します。

### OTMA クライアントとは？

OTMA クライアントは、トランザクションを IMS サーバーに送信して出力を受信する z/OS アプリケーション・プログラムです。アプリケーション・プログラムは、z/OS システム間カップリング・ファシリティ (XCF) グループのメンバーで、OTMA プロトコルを使用しなければなりません。

異機種 (非 z/OS) ネットワークは、あらゆる方法で z/OS と接続することが可能です。以下の図は、XCF を使用可能なアプリケーションの一部を示します。以下のものが含まれます。

- IBM MQ アプリケーション
- OEM アプリケーション
- IMS 接続アプリケーション
- DCE/RPC アプリケーション
- その他の IBM アプリケーション

これらのどれも、IMS と通信する OTMA クライアントに接続できます。

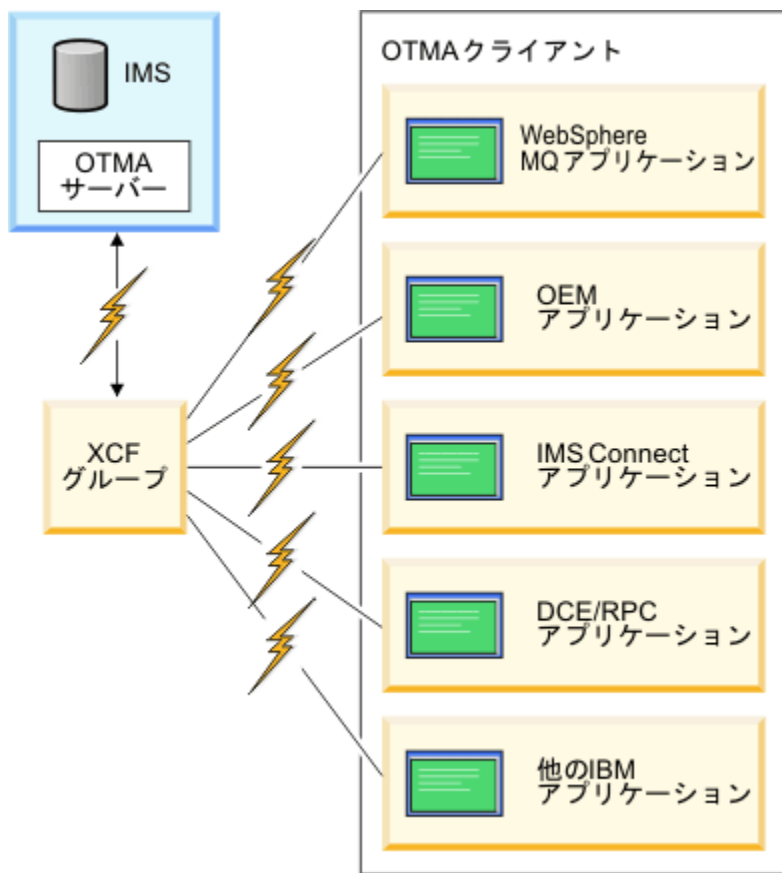


図 143. XCF を使用して z/OS 上の IMS に接続するアプリケーション

OTMA クライアントは、IMS の外部からトランザクションが IMS に入ることができる ゲートウェイです。

OTMA 処理には、以下が含まれます。

1. クライアントは、トランザクションまたはコマンドをサーバー (IMS) に送信します。
2. サーバーがクライアントに出力を戻します。

## OTMA の命名規則

OTMA クライアントおよびトランザクション・パイプの名前は、IMS で定義された規則に従う必要があります。

OTMA クライアントおよびトランザクション・パイプの名前は、以下の規則に準拠する必要があります。

- 文字タイプ A (A から Z、0 から 9、@、\$) である
- 非空白文字で始まる
- 最大長 (クライアント名の場合は 16、トランザクション・パイプ名の場合は 8) より短い場合は、空白を埋め込む
- 組み込み空白は含まない
- 予約語 (例えば、"TO" または "SECURITY") を使用しない
- "DFS" または "DBCDM" で始まらない
- IMS キーワード (例えば、"LINE" または "NODE") を使用しない

さらに、トランザクション・パイプ名については、以下のとおりです。

- IMS トランザクション名と重複しない。
- z/OS システム・コンソール (例えば、"WTOR")、IMS MTO、あるいは 2 次 MTO と同じ名前にしない。

IMS は、英大文字変換は行いません。小文字を使用する場合、クライアントは、サーバーから否定応答 (NAK) を受信します。

## OTMA クライアントから送信されるメッセージ

OTMA クライアントは、メッセージを送信して、IMS と通信します。

まず、ユーザーは、クライアントに接続している装置あるいはプログラムを使用して、アプリケーション・データを入力します。次に、クライアントは何らかの情報 (メッセージ接頭語) を追加して、メッセージを IMS に送信します。IMS からの出力は、メッセージとしてクライアントに送信され、クライアントは、メッセージ接頭語を使用して、データを正しい装置またはプログラムに経路指定します。

### OTMA メッセージ接頭語の構成部分

OTMA メッセージ接頭語は、以下のセクションを持ちます。

- メッセージ制御情報

このセクションには、トランザクション・パイプ名、メッセージ・タイプ、シーケンス番号 (ある場合)、送信後コミット・メッセージのタイムアウト値、および各種のフラグと標識が含まれています。

- 状態データ

このセクションには、宛先オーバーライド (ある場合)、マップ名、同期レベル、コミット・モード、トークン、およびサーバー状態が含まれます。

- セキュリティー・データ

このセクションには、ユーザー ID、ユーザー・トークン、およびセキュリティー・フラグが含まれています。

- ユーザー・データ

このセクションには、クライアントに必要な特殊情報が含まれています。

以下のメッセージ接頭語は、メッセージのアプリケーション・データ・セクションです。このセクションには、処理のために IMS に送信されるデータか、IMS 応答を含みます。

## OTMA メッセージ接頭語の規則

メッセージは、単一セグメントまたは複数セグメントを持っているので、以下の規則が OTMA メッセージ接頭語に適用されます。

- 単一セグメント・メッセージは、完全接頭部 (メッセージ制御情報、状態データ、セキュリティー・データ、およびユーザー・データ) を持ちます。
- 複数セグメント・メッセージでは、最初のセグメントだけが、完全接頭部を持ちます。後続のセグメントでは、メッセージ制御情報とアプリケーション・データ・セクションだけが送信されます。
- IMS が送信する確認通知 (ACK あるいは NAK) メッセージは、最初の入力バッファだけを戻します。このメッセージは、完全接頭部とアプリケーション・データ・セクションを運びます (クライアント要求に含まれている場合)。

## OTMA で使用するシーケンス番号

OTMA は、メッセージのシーケンス番号に次の 2 つのタイプを使用します。送信シーケンス番号とリカバリ可能シーケンス番号。OTMA では、送信シーケンス番号とリカバリ可能シーケンス番号は、異なって使用されます。

### 送信シーケンス番号の使用

送信シーケンス番号は、入力および出力メッセージに使用されます。送信シーケンス番号は、入力メッセージごとにクライアントが増加させる必要があります。IMS がクライアントに出力を送信する場合、出力メッセージ内の送信シーケンス番号も増加します。送信シーケンス番号は、すべての OTMA 入出力メッセージに使用されます。入力メッセージ中の送信シーケンス番号は、複数セグメントを識別するためにも使用されます。

例えば、2 セグメントの OTMA 入力メッセージがあるとします。最初のセグメント・メッセージは、送信シーケンス番号 =XXX およびセグメント番号 =1 を持ちます。2 番目のセグメント・メッセージは、同じ送信シーケンス番号 =XXX およびセグメント番号 =2 を持ちます。OTMA は、同じ送信シーケンス番号を持っているので、2 セグメント・メッセージをチェーニングします。

OTMA は、以下の方法で送信シーケンス番号を使用します。

- IMS からのすべての ACK と NAK メッセージは、入力でクライアントがサブミットした送信シーケンス番号を使用します。
- IMS がクライアントに送信する OTMA コマンドはすべて、送信シーケンス番号 0 (ゼロ) を持ちます。そして、再同期フロー以外は、これらの OTMA コマンドは、すべて単一セグメントです。
- IMS エラー・メッセージと IMS トランザクション出力用の送信シーケンス番号は、各トランザクション・パイプごとにセットされます。トランザクション・パイプの送信シーケンス番号は、メッセージごとに 1 ずつ増加し、決して 0 (ゼロ) にはなりません。シーケンス番号が 4,294,967,295 (32 ビットの最大値) を超えると、1 にリセットされます。

### リカバリ可能シーケンス番号の使用

リカバリ可能シーケンス番号は再同期の制御にのみ使用されます。クライアントが再同期をサポートしない場合は、リカバリ可能シーケンス番号 =0 (ゼロ) です。再同期は、同期化された T パイプおよびコミット後送信入出力にのみ有効です。リカバリ可能シーケンス番号も、入出力メッセージごとに増加します。再同期サポートには、リカバリ可能シーケンス番号が正しく増加したかどうかを検査するロジックが追加されています。シーケンス番号が正しく増加されなかった場合は、NAK が送信されます。再同期は、リカバリ可能シーケンス番号に従属しているので、再同期は各入出力ごとに訂正されなければなりません。リカバリ可能シーケンス番号は、再同期を制御するために、それらを使用するトランザクション・パイプに適用されます。

### 関連概念

[863 ページの『OTMA でのクライアント/サーバー再同期』](#)

クライアント・トランザクションが一度だけ処理されることを保証するために、OTMA は、同期化トランザクションの protocols を提供します。

#### 関連資料

889 ページの『OTMA メッセージ接頭語』

OTMA メッセージには、IMS.ADFSMAC データ・セットの DFSYMSG マクロによってマップされるフォーマットに準拠した接頭部があります。

## OTMA クライアントからのタイプ 1 コマンドの送信

タイプ 1 の IMS コマンドは、OTMA クライアント・アプリケーションからサブミットできます。これにはいくつかの制約事項と推奨事項が適用されます。

### このタスクについて

大部分のコマンドは、送信後コミット (CM1) プロトコルを使用してサブミットする必要があります。ただし、以下のコマンドは、コミット後送信 (CM0) プロトコルを必要とします。

- /DBDUMP DATABASE
- /DBRECOVERY AREA|DATABASE
- /START AREA|DATABASE
- /START REGION
- /STOP AREA|DATABASE
- /STOP REGION

**DISPLAY TRANSACTION** コマンドは、OTMA クライアントから発行された場合、異なる動作をします。このコマンドは、その出力を IMS マスター端末ではなく、クライアントに直接送信します。メッセージ接頭部のメッセージ制御情報セクション内の拡張応答要求フラグの設定に応じて、出力は、設計済みフォーマット (このコマンドでのみサポート) または標準 IMS フォーマットになります。

**推奨事項:** クライアントは、コミット後送信 (CM0) フローを使用しなければならないので、これらのコマンドからの出力は、入力コマンドと結び付けることはできません。OTMA 接頭部は、複製されません (入力と出力の両者に共通のただ 1 つのフィールドは、トランザクション・パイプ名です)。そのため、クライアントが IMS 処理用に予約しているトランザクション・パイプを使用して IMS コマンドをサブミットするように、クライアントを構成してください。

#### 制約事項:

OTMA クライアントは、以下のタイプのサブシステムから IMS コマンドをサブミットすることはできません。

- IMS 拡張回復機能 (XRF) 代替サブシステム
- CICS-IMS DBCTL サブシステム

#### 関連資料

[/DISPLAY TRAN コマンド \(コマンド\)](#)

## OTMA コミット処理

OTMA は、IMS がトランザクションをコミットする方法を制御します。その方法には、コミット後送信あるいは送信後コミットがあります。

#### 定義:

- コミット後送信 トランザクション (IMS 標準フロー) の場合、IMS はトランザクションを処理し、OTMA クライアントに 応答を送信する前に、データをコミットします。
- 送信後コミット・トランザクションの場合、IMS は、トランザクションを処理し、データをコミットする前に、OTMA クライアントに 応答を送信します。

**Q:** コミット後送信の処理オプションと 送信後コミットの処理オプションの主な違いは何ですか？

**A:** コミット後送信の処理オプションでは、同期点処理の一部としてトランザクション出力をコミットし、その後、クライアントに出力を送達します。

送信後コミットの処理オプションでは、まずトランザクション出力を送達し、クライアントから確認応答を受信してから、同期点処理を完了させます。

**Q:** コミット・モード 0 およびコミット・モード 1 の処理オプションでは何が起こりますか？

**A:** コミット・モード 0 は、現在「コミット後送信」と呼ばれ、コミット・モード 1 は、「送信後コミット」と呼ばれます。用語の「コミット後送信」および「送信後コミット」の方が、これらの処理オプションを参照する場合、より直接的なので、「コミット・モード 0」と「コミット・モード 1」は使用していません。

OTMA トランザクションの場合、クライアントは、IMS から以下の 1 つを受信します。

- 入力 ACK メッセージと、それに続く出力メッセージ。

さらに、送信後コミット・トランザクションは、ACK メッセージとそれに続く「割り振り解除」フロー（メッセージ接頭語のメッセージ制御情報セクション中のコミット確認フラグが「コミット済み」あるいは「アポート」に設定されるとき示される）も受信します。

- センス・コード付き NAK メッセージ。
- プロセス・フラグ付き NAK メッセージは、メッセージ接頭語のメッセージ制御情報セクション中に、「Error Message Follows」をセットします。後続のメッセージは、NAK メッセージと同じメッセージ接頭語を持ち、IMS エラー・メッセージをメッセージ接頭語のアプリケーション・データ・セクションに持ちます。

## OTMA コミット処理の要約

以下の表は、コミット後送信と送信後コミット処理間の違いを要約しています。

最初の列にいくつか変数がリストされていますが、処理オプション間の差は次の 2 つの列で説明されます。以下の表は注意しておかなければいけない使用上の注意です。

変数 es	コミット後送信	送信後コミット
会話型	クライアントは、NAK メッセージを受信する。	サポートされる。
高速機能 (Fast Path)	クライアントは、NAK メッセージを受信する。	サポートされる。
非会話型 および非高速機能のトランザクション	IMS は、出力をクライアントにエンキューした後、コミットする。出力は後で送達される。	IMS は、クライアントに出力を送信した後、コミットする。
入力をエンキューするか？	はい。	はい。
出力をエンキューするか？	はい。	いいえ。
同期化トランザクション・パイプを指定したか？	サポートされる。	クライアントは、NAK メッセージを受信する。
タイムアウト間隔を適用したか？	いいえ。	タイムアウト値を synclevel=confirm または synclevel=synchpt で指定した場合は、はい。

注：

- IMS 会話は、「コミット後送信」コミット・モードを使用できない。
- 入出力の送信後コミットは、リカバリー不能。

- リカバリー不能出力 (送信後コミット) の場合、同期レベルが「確認」に設定されているなら、IMS は確認応答を要求する。
- リカバリー可能トランザクションの場合、IMS は、出力メッセージに対する確認応答を、常に要求しません。
- コミット後送信トランザクションの場合、IMS は常に確認応答を要求します。
- 同期化トランザクション・パイプは、コミット後送信トランザクションに対してのみ使用できます。
- 送信後コミット (CM1) 入力メッセージがトランザクションに送られると、OTMA はそのトランザクションを、たとえそれが NONRESPONSE として定義されていても、RESPONSE モードとして扱います。アプリケーションが IOPCB に応答する場合、出力は送信後コミットです。アプリケーションが IOPCB に応答せず、プログラム間メッセージ通信を完了しない場合、OTMA は DFS2082 RESPONSE MODE TRANSACTION TERMINATED WITHOUT REPLY メッセージで応答します。
- コミット後送信 (CM0) 入力メッセージがトランザクションに送信され、OTMA 状態データ接頭部に TMAMHRSP フラグが設定されている場合、OTMA は、そのトランザクションが NONRESPONSE として定義されていても、そのトランザクションを RESPONSE モードとして処理します。アプリケーションが IOPCB に応答せず、プログラム間メッセージ通信を完了しない場合、OTMA は DFS2082 メッセージで応答します。

**制約事項:** コミット後送信トランザクションのこの DFS2082 メッセージは、元の入力トランザクションに対してのみ出され、プログラム間通信はサポートしません。

#### 関連情報

[DFS2082 \(メッセージおよびコード\)](#)

## OTMA コミット処理フローのサンプル

2つのコミット・モードの違いを説明するために、以下にリストされたトピックでは、各コミット・モードごとに IMS とクライアント間のデータのサンプル・フローを示します。

#### 関連資料

857 ページの『[OTMA メッセージ・フローのサンプル](#)』

以下のトピックでは、いくつかのサンプル・メッセージ・フローを示し、メッセージ接頭語中の各種フィールドの設定方法について記述します。

### コミット後送信フロー

IMS 標準フローでもあるコミット後送信フローは、IMS 出力をクライアントに送信する前にエンキューします。このフローは状態標準トランザクション処理に使用します。

標準フローを使用するには、メッセージ接頭語の状態データ・セクションでコミット後送信 (コミット・モード 0) を指定します。このサンプル・フローでは、以下のことを想定しています。

- トランザクション・パイプは同期化される。IMS は、トランザクション・パイプ用に リカバリー可能入出力のシーケンス番号を維持します。
- 確認通知は、常に要求される (IMS およびクライアントの両者によって)。

NAK 応答を IMS が受信すると、その後出力はキューに戻され、後で送達されます。

そのフローを以下の図に示します。図の後に、フローについての詳細を提供する順次リストがあります。



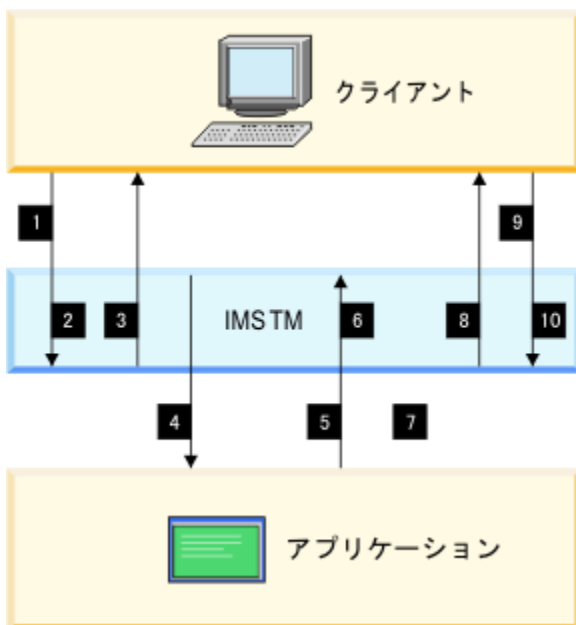


図 144. コミット後送信 (IMS 標準) フロー

図に示されたフローの順序は以下のとおりです。

1. トランザクションが開始される (応答が T パイプを要求/同期する)。
2. トランザクションが SMB に挿入される。
3. ACK
4. GU 呼び出し後、IOPCB への ISRT。
5. 同期点
6. 出力が T パイプにエンキューされ、DB がコミットされる。
7. トランザクションが完了する。
8. 出力が、応答要求とともに送信される。
9. ACK
10. 出力がデキューされる。

単一のコミット後送信トランザクション・パイプに対するメッセージ・アクティビティのフローの例を、以下の図に示します。図の後に、フローについての詳細を提供する順次リストがあります。

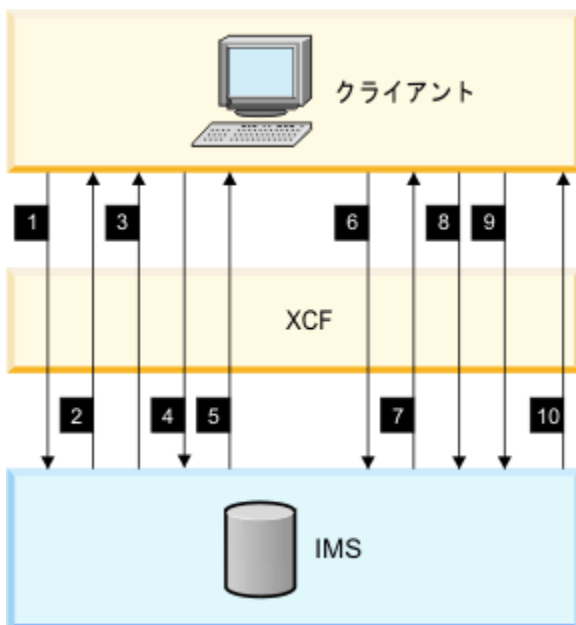


図 145. コミット後送信フローに対するメッセージ・フローのサンプル

図に示されたフローの順序は以下のとおりです。

1. Tran1
2. Tran1 ACK
3. Tran2
4. Tran1 の出力
5. Tran2 への ACK
6. Tran3
7. Tran3 への ACK
8. Tran4
9. Tran1 の出力への ACK
10. Tran4 への ACK

以下の条件が満たされている場合、トランザクションの応答モードに関係なく、OTMA は DFS2082 メッセージをクライアントに送信します。

- 状態データ接頭部に TMAMHRSP オプション・フラグが設定されている。
- 元の入力トランザクションを処理する IMS アプリケーションが IOPCB に応答しない。
- 元の入力トランザクションを処理する IMS アプリケーションがプログラム間メッセージ通信を完了しない。

#### 関連資料

904 ページの『トランザクションおよびコールアウト・メッセージ』

OTMA メッセージ接頭語のトランザクション関連情報および同期コールアウト要求情報の状態データは、DFSYMSG マクロの TMAMHDR DSECT によってマップされます。

#### 関連情報

[DFS2082 \(メッセージおよびコード\)](#)

## 送信後コミット・フロー

送信後コミット (コミット・モード 1 または CM1) フローでは、IMS が同期点処理を完了する前に IMS 出力が送信されます。

送信後コミット・フローを使用するには、メッセージ接頭語の状態データ・セクションでコミット・モードを 1 に指定します。このサンプル・フローでは、以下のことを想定しています。

- トランザクション・パイプは同期化されない。
- 同期レベルは、状態データ・セクション中で「なし」として指定されている。したがって、IMS は、出力の送信時に応答 (ACK) を要求しません。

そのフローを以下の図に示します。図の後に、フローについての詳細を提供する順次リストがあります。

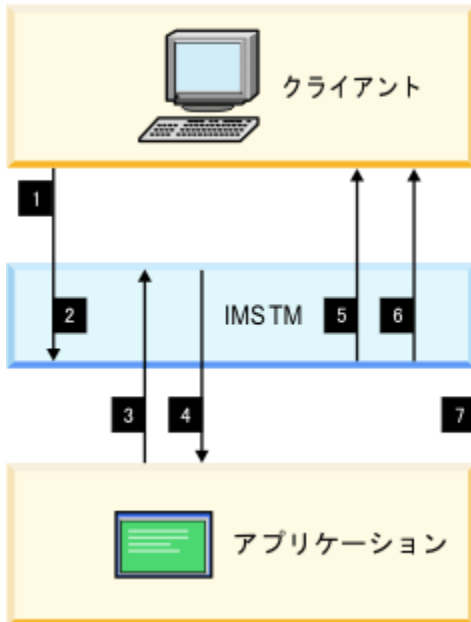


図 146. 送信後コミット・フロー

上の図に示されたフローの順序は以下のとおりです。

1. トランザクションが開始される。
2. トランザクションが SMB に挿入される。
3. GU 呼び出し後、IOPCB への ISRT。
4. 同期点が開始される。
5. 出力が送信される。応答は要求されていない。sync=confirm が指定されているときのみ、応答が要求される。
6. コミットが確認される。IMS が同期点を完了する。
7. トランザクションが完了する。

単一トランザクション・パイプに対するメッセージ・アクティビティのフローの例を、以下の図に示します。図の後に、フローについての詳細を提供する順次リストがあります。

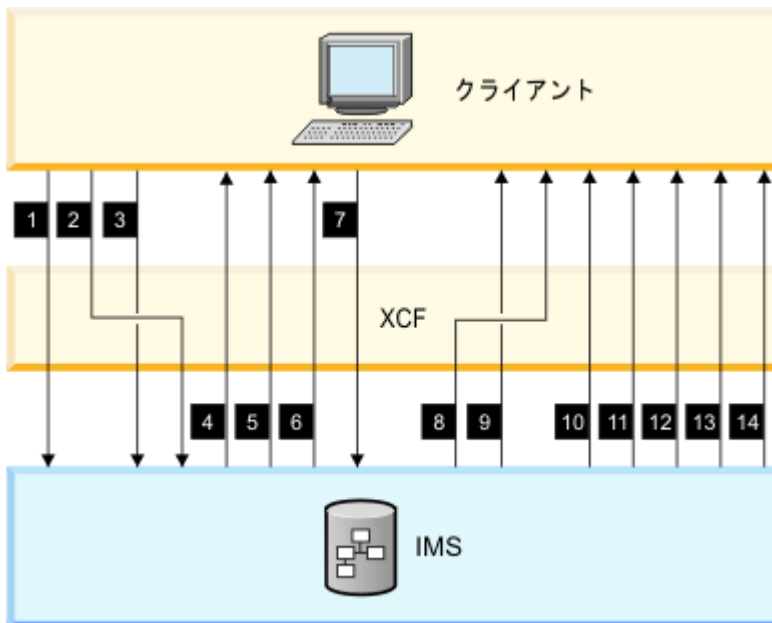


図 147. 送信後コミット・フローに対するサンプル・メッセージ・フロー

上の図に示されたフローの順序は以下のとおりです。

1. Tran1
2. Tran2 要求/応答
3. Tran3 要求/応答
4. Tran3 への ACK
5. Tran1 の出力
6. Tran2 への ACK
7. Tran4
8. Tran4 の出力
9. Tran4 の確認
10. Tran2 の出力
11. Tran3 の出力
12. Tran1 の確認
13. Tran3 の確認
14. Tran2 の確認

前の図に示すように、クライアントは、実際の出力を受け取る前に出力の確認を受け取ることができます。z/OS システム間カップリング・ファシリティ (XCF) は、すべてのメッセージが順次に送信されることを保証するわけではないためです。クライアントは、メッセージ受信処理中に、または、XCF メッセージ出口ルーチンを使用して、この状態を操作できなければなりません。

送信後コミット・トランザクションが正常に実行されない場合、OTMA はメッセージ DFS2082 をクライアントに送信し、IOPCB への挿入を実行しません。

## 確認付き送信後コミット・フロー

送信後コミット・フローは、IMS で処理されるように、トランザクションに対する同期がないことを想定しています。

このトピックでは、トランザクションの受信時に、トランザクションをすべて確認する (各メッセージが応答を要求する) フローを示します。下の図に示されたサンプルは、以下のことを想定しています。

- メッセージ接頭語の状態データ・セクションに、コミット・モード 1 が指定されている。

- トランザクション・パイプは同期化されない。
- 同期レベルは、状態データ・セクションで CONFIRM (確認) と 指定されている。

NAK を IMS が受信した後、アプリケーションでユーザー 119ABEND が起こり、IMS は、DFS554 メッセージをクライアントに出します。

下の図の後に、フローの詳細を示す順次リストがあります。

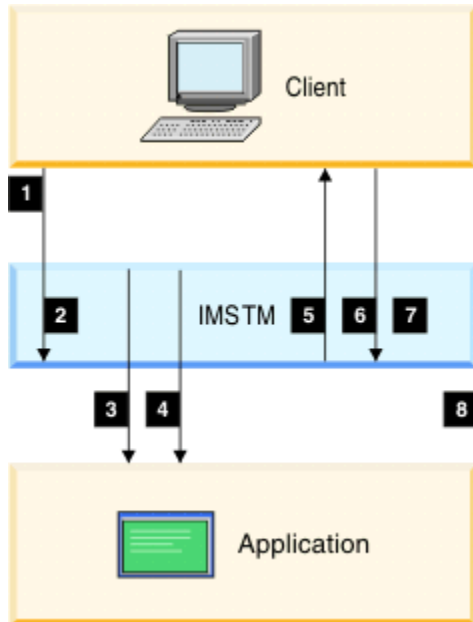


図 148. 確認付きの送信後コミット・フロー

上の図に示されたフローの順序は以下のとおりです。

1. トランザクションが開始される。
2. トランザクションが SMB に挿入される。
3. GU 呼び出し後、IOPCB への ISRT。
4. 同期点が開始される。
5. 出力が送信され、応答が要求される。
6. ACK
7. DB がコミットされ、コミットが確認され、IMS が同期点を完了する。
8. トランザクションが完了する。

## OTMA メッセージ・フローのサンプル

以下のトピックでは、いくつかのサンプル・メッセージ・フローを示し、メッセージ接頭語中の各種フィールドの設定方法について記述します。

図中では、以下の省略形をメッセージ接頭語の部分に使用します。

### MC

メッセージ制御情報セクション

### SD

状態データ・セクション

### SE

セキュリティー・データ・セクション

### US

ユーザー・データ・セクション

## AP

### アプリケーション・データ・セクション

サンプル・フロー・チャートでは、接頭部のどの部分が、指定のメッセージに必須であるか、どの部分が適用できないかを示します。オプションのフィールドと接頭部セクションは、括弧で囲みます。

クライアントがサブミットするトランザクションについては、以下の原理が適用されます。

- IMS が ACK メッセージをクライアントに送信した後、IMS は、(トランザクションが正常にコミットされたか、打ち切られたかを示す) コミット確認を送信します。
- コミット確認により、クライアント・トランザクションが終了します。

### 関連資料

919 ページの『サンプル OTMA メッセージ』

以下の 3 つのサンプル OTMA メッセージは、OTMA メッセージが、メッセージ接頭語の部分を含み、完全に構成された場合にどのようになるかを示します。各例は、必ずしも相互に関連しているとは限りません。

## クライアント・ビッドのメッセージ・フロー

以下の図は、クライアント・ビッド・フローで、クライアントがサーバーに接続しようとするところを示します。

このフローは、クライアントがすでに z/OS システム間カップリング・ファシリティグループに結合されていて、サーバーがそのグループに結合されたことを通知するとき起こります。クライアント・ビッド・フローは以下のとおりです。

1. クライアント・ビッド: MC, SD, SE
2. ACK: MC, SD, SE

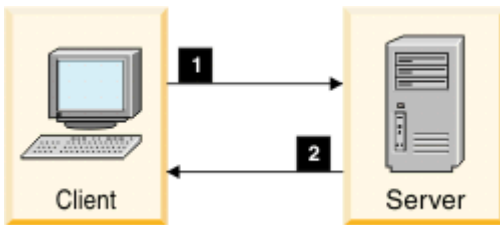


図 149. クライアント・ビッド・フロー

以下の表は、クライアント・ビッド交換の最初のフローであるクライアント・ビッド要求での OTMA メッセージ接頭語の内容を示しています。この表は、OTMA メッセージ接頭語のメッセージ制御データ、状態データ、およびセキュリティー・データの各セクションの内容を示しています。

表 147. クライアント・ビッド交換の最初のクライアント・ビッド・フローでの OTMA メッセージ接頭語の内容

メッセージ接頭語セクション	接頭部セクションの内容
メッセージ制御データ	アーキテクチャー・レベル=1 メッセージ・タイプ=コマンド 応答フラグ=応答要求 コマンド・タイプ=クライアント・ビッド 接頭部フラグ=状態データ+セキュリティー・データ
状態データ	コマンド・メッセージの状態データ・フォーマットは、長さ、メンバー名、発信元のトークン、宛先トークン (DRU 出口名) 最大ブロック・サイズ、エージング値、ハッシュ・テーブル・サイズのフィールドに適用される
セキュリティー・データ (UTOKEN)	

以下の表は、クライアント・ビッド交換の 2 番目のフローである確認応答での OTMA メッセージ接頭語の内容を示しています。この表は、OTMA メッセージ接頭語のメッセージ制御データ、状態データ、およびセキュリティー・データの各セクションの内容を示しています。

表 148. クライアント・ビッド交換の 2 番目のフローである確認応答での OTMA メッセージ接頭語の内容

メッセージ接頭語セクション	接頭部セクションの内容
メッセージ制御データ	アーキテクチャー・レベル=1 メッセージ・タイプ=コマンドと応答 応答フラグ=ACK コマンド・タイプ=クライアント・ビッド 接頭部フラグ=状態データ+セキュリティー・データ
状態データ	コマンド・メッセージの状態データ・フォーマットは、長さ、メンバー名、発信元のトークン、宛先トークン (DRU 出口名) 最大ブロック・サイズ、エージング値、ハッシュ・テーブル・サイズのフィールドに適用される
セキュリティー・データ	UTOKEN

## サーバー使用可能フロー

以下の図は、サーバー使用可能交換のメッセージ・フローで、サーバーがクライアントに接続しようとしているところを示しています。

このフローは、サーバーがすでに z/OS システム間カップリング・ファシリティ (XCF) グループに結合され、クライアントがそのグループに結合することを認識している場合にのみ起こります。クライアントは、サーバーが XCF グループに結合したことの認識のために待つ必要なく、クライアントは、グループに結合するとすぐ、クライアント・ビッド・メッセージを送信する必要があります。

クライアントは、そのクライアント・ビッド要求が正常に完了し、XCF グループに接続した後は、サーバー使用可能メッセージを無視する必要があります。表示されたフローは以下のとおりです。

1. サーバー使用可能: MC、SD、SE
2. クライアント・ビッド: MC、SD、SE
3. ACK: MC、SD、SE

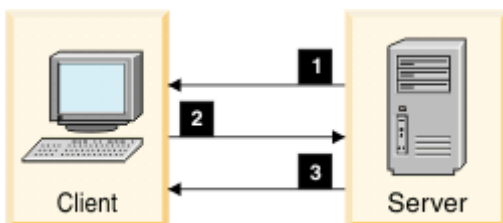


図 150. サーバー使用可能フロー

以下の表には、メッセージ接頭語の内容を示します。メッセージ・フロー・タイプ、メッセージ接頭語セクション、および接頭部 MC、SD、SE に対するメッセージ接頭語セクションの関連した内容とともに、フロー・ステップが表示されます。前の図と次の表の両方に示されている、順序を表示するために使用されている番号は、実際のメッセージ接頭語の一部ではありません。

表 149. サーバー使用可能フローのメッセージ接頭語の内容

フローのステップ	メッセージ・フロー	メッセージ接頭語セクション	接頭部セクションの内容
1	サーバー使用可能	MC	アーキテクチャー・レベル=1 メッセージ・タイプ=コマンド 応答フラグなし コマンド・タイプ=サーバー使用可能
		SD	コマンド・メッセージの状態データ・フォーマットは、長さ、メンバー名、発信元のトークン、宛先トークンに適用される



表 149. サーバー使用可能フローのメッセージ接頭語の内容 (続き)

フローのステップ	メッセージ・フロー	メッセージ接頭語セクション	接頭部セクションの内容
2	クライアント・ビッド	MC	アーキテクチャー・レベル=1 メッセージ・タイプ=コマンド 応答フラグ=応答要求 コマンド・タイプ=クライアント・ビッド 接頭部フラグ=状態データ+セキュリティー・データ
		SD	コマンド・メッセージの状態データ・フォーマットは、長さ、メンバー名、発信元のトークン、宛先トークン (DRU 出口名) 最大ブロック・サイズ、エージング値、ハッシュ・テーブル・サイズのフィールドに適用される
		SE	(UTOKEN)
3	ACK	MC	アーキテクチャー・レベル=1 メッセージ・タイプ=コマンドと応答 応答フラグ=ACK コマンド・タイプ=クライアント・ビッド 接頭部フラグ=状態データ+セキュリティー・データ
		SD	コマンド・メッセージの状態データ・フォーマットは、長さ、メンバー名、発信元のトークン、宛先トークン (DRU 出口名) 最大ブロック・サイズ、エージング値、ハッシュ・テーブル・サイズのフィールドに適用される
		SE	UTOKEN

## コミット後送信トランザクション・フロー

以下の図は、コミット後送信トランザクションで、クライアントが処理のためにトランザクションをサーバーにサブミットするフローを示します。

フローは以下のとおりです。

1. トランザクション ABC: MD、SD、SE、(US)、AP
2. ACK: MC、SD、SE、(US)
3. トランザクション出力: MC、SD、(US)、AP
4. ACK: MC、SD

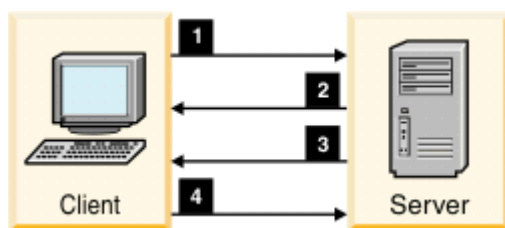


図 151. コミット後送信トランザクション・フロー

以下の表には、メッセージ接頭語の内容を示します。メッセージ・フロー・タイプ、メッセージ接頭語セクション、および接頭部 MC、SD、SE、US、AP に対するメッセージ接頭語セクションの関連した内容とともに、フロー・ステップが表示されます。前の図と次の表の両方に示されている、順序を表示するために使用されている番号は、実際のメッセージ接頭語の一部ではありません。

表 150. コミット後送信トランザクション・フローのメッセージ接頭語の内容

フローの ステップ	メッセージ・フ ロー	メッセージ接 頭語セクショ ン	接頭部セクションの内容	
1	トランザクション 'ABC'	MC	アーキテクチャー・レベル = 1 メッセージ・タイプ = トランザクション 応答フラグ = 応答要求 トランザクション・パイプ名 接頭部フラグ = SD/SE/(US)/AP 送信シーケンス番号	
		SD	長さ 同期フラグ = コミット・モード 0 同期レベル = 確認またはなし (マップ名) (相関関係子) サーバー・ユーザー・データの長さ = 0	
		SE	長さ (セキュリティ・フラグ) フィールドの長さ ユーザー ID の長さ (ユーザー ID タイプ = 02) (ユーザー ID) プロファイルの長さ (プロファイル・タイプ = 03) (RACF グループ) Utoken の長さ (Utoken タイプ = 00) (Utoken)	
		(US)	このオプションのセクションは、トランザクション出力 (長さのユーザー・データ) と共に返される	
		AP	長さ ZZ アプリケーション・データ (例の「ABC」)	
		ACK	MC	アーキテクチャー・レベル = 1 メッセージ・タイプ = トランザクションと応答 応答フラグ = ACK トランザクション・パイプ名 接頭部フラグ = SD/SE 送信シーケンス番号
2	ACK	SD	長さ 同期フラグ = コミット・モード 0 同期レベル = 確認またはなし (マップ名) (相関関係子) サーバー・ユーザー・データの長さ = 0	
		SE	長さ (セキュリティ・フラグ) フィールドの長さ ユーザー ID の長さ (ユーザー ID タイプ = 02) (ユーザー ID) プロファイルの長さ (プロファイル・タイプ = 03) (RACF グループ) Utoken の長さ (Utoken タイプ = 00) (Utoken)	
		(US)	このオプションのセクションは、トランザクション出力 (長さのユーザー・データ) と共に返される	
		トランザクション 出力	MC	アーキテクチャー・レベル = 1 メッセージ・タイプ = データ 応答フラグ = 応答要求 トランザクション・パイプ名 接頭部フラグ = SD/(US)/AP 送信シーケンス番号 サーバー・トークン
		SD	長さ 同期フラグ = コミット・モード 0 同期レベル = 確認またはなし (マップ名) サーバー・トークン (相関関係子) サーバー・ユーザー・データの長さ (サーバー・ユーザー・データ)	
		(US)	このオプションのセクションは、トランザクション出力、長さ (ユーザー・データ) と共に返される	
3	ACK	AP	長さ ZZ トランザクション出力データ	
		トランザクション 出力	MC	アーキテクチャー・レベル = 1 メッセージ・タイプ = データと応答 応答フラグ = ACK トランザクション・パイプ名 接頭部フラグ = SD 送信シーケンス番号
		SD	長さ 同期フラグ = コミット・モード 0 同期レベル = 確認またはなし (マップ名) サーバー・トークン (相関関係子) サーバー・ユーザー・データの長さ (サーバー・ユーザー・データ)	
4	ACK	ACK	MC	アーキテクチャー・レベル = 1 メッセージ・タイプ = データと応答 応答フラグ = ACK トランザクション・パイプ名 接頭部フラグ = SD 送信シーケンス番号
		SD	長さ 同期フラグ = コミット・モード 0 同期レベル = 確認またはなし (マップ名) サーバー・トークン (相関関係子) サーバー・ユーザー・データの長さ (サーバー・ユーザー・データ)	

## OTMA でのトランザクション保護

z/OS 環境では、トランザクションの保護およびリカバリーは、z/OS リカバリー・リソース管理サービス (RRMS) の一部である z/OS リソース・リカバリー・サービス (RRS) によって管理されます。RRS は、複数の主幹業務のリソースにわたる、調整された変更に適用できます。

### このタスクについて

OTMA は、IMS が保護トランザクションをサポートできるようにする 2 つのコンポーネントの 1 つです。2 番目のコンポーネントは APPC/IMS です。

保護トランザクションを処理するには、制御領域 JCL で、RRS=Y を指定します。すると、IMS は、CRGGRM サービスを使用してリソース・マネージャー (RM) として RRMS に登録します。また、CRGSEIF サービスを使用して、コンテキスト・サービスおよび RRS 出口マネージャーで出口も設定します。IMS は、以下の RRS 出口ルーチンをサポートします。

- PREPARE
- COMMIT
- BACKOUT
- EXIT\_FAILED
- ONLY\_AGENT
- SUBORDINATE\_FAILED

初期設定中に、IMS は、メッセージ DFS0653I を発行して、RRS に正常に接続したことで、保護トランザクションを処理できるようになったことを示します。

## OTMA クライアントからの保護トランザクションの開始

以下は、IMS Connect や Db2 for z/OS ストアード・プロシージャなどの OTMA クライアントから、保護トランザクションを開始する手順です。

### 手順

1. OTMA のメッセージ接頭語に Synclevel=2 (Syncpt) を指定します。
2. コンテキスト・トークンを獲得するかまたは再利用します。RSS コンテキスト・トークンを獲得するには、CTXBEGC サービスを使用します。
3. OTMA メッセージ接頭語にコンテキスト・トークンを設定します。
4. ATREINT を使用して、リカバリー単位 (UR) のインタレストを表明します。
5. OTMA にメッセージを送信します。
6. IMS トランザクションからの出力を待ちます。
7. 出力を受信したら、確認応答 (ACK) を IMS に送信します。
8. z/OS リソース・リカバリー・サービス (RRS) コミット (ATRACMT) またはバックアウト (ATRABCK) を開始します。

### タスクの結果

OTMA クライアントは、サーバー分散同期点 RM (SDSRM) の役割を想定します。これは、OTMA クライアントがコンテキストを所有し、OTMA クライアントのみが RRS コミットを開始したり呼び出したりすることを許されている RM であることを意味します。このフローは、以下の例外はありますが、APPC/IMS 保護されたトランザクションのフローに似ています。

- OTMA クライアントは、RRS Commit\_Agent\_UR サービス (ATRACMT) を使用してコミットを開始する。
- その後 RRS は、IMS にコミットするように直接知らせる。直前の ATRACMT 呼び出しの結果として、RRS は利害関係のあるすべての RM のコミット出口を駆動します。

## IMS における保護トランザクションの処理

IMS は、保護トランザクションを受け取り、OTMA メッセージ・ヘッダーからコンテキスト・トークンを取り出します。IMS は、保護トランザクションを IMS メッセージ・キューに置く前に、自分の制御ブロックにコンテキスト・トークンに保管します。

また、保護トランザクションをメッセージ・キューに置く前に、IMS は Express\_Context\_Interest サービス (CTXEINT) を使用して、コンテキストへのインタレストを表します。そして、IMS は、保護トランザクションがメッセージ・キューに入れられている間にコンテキストに何か起こったかどうかを知らされます。

IMS が保護トランザクションを従属領域でスケジュールに入れる場合、IMS は Switch\_Context サービス (CTXSWCH) を使用して、コンテキスト・トークンを従属領域 TCB に交換します。ATREINT を使用して、UR への保護インタレストも表します。その後 IMS は、保護トランザクションをその特定のトランザクションを処理するアプリケーション・プログラムに渡します。そのアプリケーションは、ビジネス・ロジック (例えば、データベースのアップデート、メッセージ送信、その他) を含みます。アプリケーションがその作業を完了すると、アプリケーションは、コミット・ポイントに到達します。そして、IMS は、アプリケーション出力を OTMA クライアントに送り返し、コミットあるいは z/OS リソース・リカバリー・サービスからのバックアウト・イベントを待ちます。

## OTMA でのクライアント/サーバー再同期

クライアント・トランザクションが一度だけ処理されることを保証するために、OTMA は、同期化トランザクションのプロトコルを提供します。

同期化されたコミット後送信 (コミット・モード 0) トランザクション・パイプを使用することにより、クライアントと IMS は、クライアントまたは IMS の障害イベントで、メッセージ・フローを再獲得します。再同期は、IMS またはクライアントが正常あるいは異常に終了するとき起こります。

トランザクション再同期では、以下のことを行います。

- データが再処理されるのを防ぐ
- データが受信されないで、クライアントがデータを再送信しようとするのを検出する
- 再同期が可能でないことを検出する
- クライアントに、再同期するために取るべきアクションを決めさせる

OTMA 再同期は対称的ではないため、システムの動作は、クライアントあるいは IMS の役割に依存します。再同期はまた、送信あるいは受信のシーケンス番号に対する対称を維持しません。例えば、IMS フローの入力側と出力側の違いは以下のとおりです。

### 入力

IMS は、トランザクションがエンキューされる時、クライアント・シーケンス番号のログを取りますが、それ以降、クライアントは、そのトランザクションをデキューして制御することはしません。

### 出力

アプリケーション出力は、同期化されたトランザクション・パイプにエンキューされますが、そのとき、出力シーケンス番号のログに記録されません。出力を送信し、クライアントから確認応答を受信した後だけ、IMS は最終的にメッセージをデキューし、増分されたシーケンス番号のログに記録します。

同期化トランザクション・パイプを使用する出力はすべて、順序付けされます。2 番目の出力メッセージは、クライアントからの ACK メッセージを最初の出力メッセージに対して受け取るまで送信されません。

**Q:** なぜ非同期化 T パイプではなく、同期化 T パイプを使用するのですか？

**A:** 同期化 T パイプを使用すると、クライアントまたは IMS 障害の場合に、クライアント・トランザクションが 1 回だけ処理されます。そのため、同期化 T パイプでは、より良いトランザクション・リカバリー可能性を確認できます。しかし、トランザクション・リカバリーを保証するためには、同期化 T パイプとともに再同期論理をインプリメントする必要があります。

トランザクションのリカバリー可能性があまり重要でない場合は、非同期化 T パイプを使用してください。非同期化 T パイプの場合、クライアントは、再同期論理を必要としません。

## OTMA 再同期の前提事項

OTMA 再同期プロセスは、いくつかの前提事項に基づいています。

前提事項を以下に示します。

- クライアントも IMS も、トランザクション・メッセージのログを取った後に ACK メッセージを送信します。
- クライアントは、IMS が取るべき再同期アクションを決めます。
- クライアントおよび IMS は両方とも、トランザクションおよび出力メッセージがリカバリー可能かどうかを決めることができます。クライアントは、**/DISPLAY TRANSACTION** コマンドの設計済みフォームを使用して、トランザクションのリカバリー可能性を決定することができます。
- リカバリー可能 OTMA メッセージは、メッセージ接頭語のメッセージ制御情報セクション中に、リカバリー可能シーケンス番号の値を組み込みます。この値は、T パイプを使用してリカバリー可能メッセージが送信されるたびに、1 つずつ増加します ([865 ページの『IMS トランザクションとコマンドの結果の要約』](#)を参照)。
- 0 (ゼロ) は、有効なリカバリー可能シーケンス番号ではありません。
- リカバリー可能な送信と受信のシーケンス番号は、トランザクション・パイプ単位ベースで維持されます。
- IMS は、すべての IMS コマンド入力に対して再同期をサポートしません。クライアントが、同期化トランザクション・パイプを使用して IMS コマンドをサブミットする必要がある場合、リカバリー可能シーケンス番号は、0 (ゼロ) にセットする必要があります。リカバリー可能シーケンス番号が 0 (ゼロ) でない場合、IMS は、コマンド入力をセンス・コード X'0023' でリジェクトします。

## リカバリー可能 OTMA トランザクション

OTMA 開始のトランザクションとコマンドのリカバリー可能性は、複数の要因によって決まります。

要因には、次のものがあります。

- リカバリー可能トランザクションかリカバリー不能トランザクションか？
- リカバリー可能コマンドかリカバリー不能コマンドか？
- リカバリー可能シーケンス番号が 0 (ゼロ) かどうか？
- 同期化トランザクション・パイプか非同期化トランザクションか？
- コミット・モード 0 (コミット後送信) かコミット・モード 1 (送信後コミット) か？

送信後コミット・トランザクションを使用してサブミットされたリカバリー可能 IMS トランザクション・フローは、リジェクトされません。しかし、送信後コミット・トランザクションは、IMS 再始動中に廃棄されます (それらはリカバリー不能です)。

同期化トランザクション・パイプを使用するトランザクションはリカバリー可能です。入力メッセージは、送信後コミット・トランザクションに対してはリカバリー可能でなく、ACK メッセージを要求することは、トランザクションがリカバリー可能かどうかに影響しません。同期化トランザクション・パイプの適切な同期に対して ACK メッセージを要求する必要があります。

同様に、トランザクションが IMS に到達したとき、そのリカバリー可能性は、トランザクションを IMS にどのように定義しているかによります。

## リカバリー不能 OTMA トランザクション

リカバリー不能トランザクションの場合、クライアントは、トランザクションがリカバリー不能であることを認識し、それを処理し、その後でそれについて忘れる必要があります。

送信後コミット・トランザクションの場合、出力はリカバリー不能であり、再同期されません。また、送信後コミット・トランザクションは、非同期化トランザクション・パイプと関連付ける必要があります。

## IMS トランザクションとコマンドの結果の要約

以下の図は、クライアントが同期化 T パイプを使用して、色々な処理条件のもとでサブミットする IMS トランザクションの結果を要約したものです。

この要約は、ゼロおよび非ゼロのリカバリー可能シーケンス番号を区別し、ゼロおよび非ゼロのシーケンス両方に対するコミット・モード 0 および 1 のリカバリー可能およびリカバリー不能トランザクションの差を示します。

これらの表で、CM0 は、コミット後送信 (コミット・モード 0) トランザクションを示し、CM1 は送信後コミット (コミット・モード 1) トランザクションを示します。

表 151. 同期化された T パイプを使用した IMS トランザクションの結果

リカバリー可能シーケンス番号	CM0 リカバリー可能トランザクション	CM0 リカバリー不能トランザクション	CM1 リカバリー可能トランザクション	CM1 リカバリー不能トランザクション
<b>0 (ゼロ)</b>	クライアントは、ACK メッセージを受信する。出力はリカバリー可能で、更新される入出力リカバリー可能シーケンスはない。	クライアントは、ACK メッセージを受信する。出力は、リカバリー可能ではなく、入出力リカバリー可能シーケンス番号は、更新されない。	クライアントは、センス・コード X'001C' で NAK メッセージを受信する。	クライアントは、センス・コード X'001C' で NAK メッセージを受信する。
<b>0 (ゼロ) ではない</b>	リカバリー可能シーケンス番号が有効ならば、クライアントは ACK メッセージを受信する。無効ならば、クライアントは、センス・コード X'001F' で NAK メッセージを受信する。トランザクションと出力は、リカバリー可能。	クライアントは、センス・コード X'0023' で NAK メッセージを受信する。クライアントは、リカバリー可能シーケンス番号 0 (ゼロ) をセットする必要がある。	クライアントは、センス・コード X'001C' で NAK メッセージを受信する。	クライアントは、センス・コード X'001C' で NAK メッセージを受信する。

以下の表は、クライアントが非同期化 T パイプを使用して、さまざまな処理条件のもとでサブミットする IMS トランザクションの結果を要約したものです。この要約は、ゼロおよび非ゼロのリカバリー可能シーケンス番号を区別し、ゼロおよび非ゼロのシーケンス両方に対するコミット・モード 0 および 1 のリカバリー可能およびリカバリー不能トランザクションの差を示します。

表 152. 非同期化された T パイプを使用した IMS トランザクションの結果

リカバリー可能シーケンス番号	CM0 リカバリー可能トランザクション	CM0 リカバリー不能トランザクション	CM1 リカバリー可能トランザクション	CM1 リカバリー不能トランザクション
<b>0 (ゼロ)</b>	クライアントは、ACK メッセージを受信する。トランザクションと出力は、リカバリー可能。	クライアントは、ACK メッセージを受信する。トランザクションと出力は、リカバリー可能ではない。	クライアントは、ACK メッセージを受信する。トランザクションと出力は、リカバリー可能ではない。	クライアントは、ACK メッセージを受信する。トランザクションと出力は、リカバリー可能ではない。



表 152. 非同期化された T パイプを使用した IMS トランザクションの結果 (続き)

リカバリー可能シーケンス番号	CM0 リカバリー可能 トランザクション	CM0 リカバリー不能 トランザクション	CM1 リカバリー可能 トランザクション	CM1 リカバリー不能 トランザクション
0 (ゼロ) ではない	クライアントは、セ ンス・コード X'0023' で NAK メッセージを 受信する。クライア ントは、リカバリー 可能シーケンス番号 0 (ゼロ) をセットす る必要がある。	クライアントは、セ ンス・コード X'0023' で NAK メッセージを 受信する。クライア ントは、リカバリー 可能シーケンス番号 0 (ゼロ) をセットす る必要がある。	クライアントは、セ ンス・コード X'0023' で NAK メッセージを 受信する。クライア ントは、リカバリー 可能シーケンス番号 0 (ゼロ) をセットす る必要がある。	クライアントは、セ ンス・コード X'0023' で NAK メッセージを 受信する。クライア ントは、リカバリー 可能シーケンス番号 0 (ゼロ) をセットす る必要がある。

以下の表は、クライアントが同期化 T パイプまたは非同期化 T パイプを使用して、色々な処理条件のもとで発行するコマンドの結果を要約したものです。この要約は、ゼロおよび非ゼロのリカバリー可能シーケンス番号を区別し、同期化および非同期化 T パイプ両方に対する コミット・モード 0 および 1 の間の差を表示します。

表 153. クライアントが発行するコマンドの結果

リカバリー可能シーケンス番号	CM0 での同期化 T パイプ	CM1 での同期化 T パイプ	CM0 での非同期化 T パイプ	CM1 での非同期化 T パイプ
0 (ゼロ)	クライアントは、ACK メッセージを受信す る。コマンド出力 は、リカバリー可能 で、出力リカバリー 可能シーケンス番号 は、更新される。	クライアントは、セ ンス・コード X'001C' で NAK メッセージを 受信する。	クライアントは、ACK メッセージを受信す る。出力は、リカバ リー可能ではない。	クライアントは、ACK メッセージを受信す る。出力は、リカバ リー可能ではない。
0 (ゼロ) ではない	クライアントは、セ ンス・コード X'0023' で NAK メッセージを 受信する。クライア ントは、リカバリー 可能シーケンス番号 0 (ゼロ) をセットす る必要がある。	クライアントは、セ ンス・コード X'001C' で NAK メッセージを 受信する。	クライアントは、セ ンス・コード X'0023' で NAK メッセージを 受信する。クライア ントは、リカバリー 可能シーケンス番号 0 (ゼロ) をセットす る必要がある。	クライアントは、セ ンス・コード X'0023' で NAK メッセージを 受信する。クライア ントは、リカバリー 可能シーケンス番号 0 (ゼロ) をセットす る必要がある。

#### 関連概念

[保全性の表 \(アプリケーション・プログラミング\)](#)

#### 関連タスク

432 ページの『[IMS トランザクション・タイプおよびトランザクション状態](#)』

トランザクションは、論理装置から IMS へ送られる最も一般的な データ・タイプです。

## OTMA 再同期プロトコル

OTMA 再同期は、各クライアントでの一連のコマンド交換に基づきます。

クライアントでの OTMA 再同期のコマンド交換は以下のとおりです。

#### CBresynch (Client\_Bid resynch)

CBresynch は、クライアントによって送信され、クライアントと IMS の両方が正常に z/OS システム間カップリング・ファシリティ (XCF) グループに結合された後、IMS との再同期を要求します。



### SRVresynch (Server resynch)

SRVresynch は、クライアントが正常に XCF グループと結合し、CBresynch を発行した後、IMS からクライアントに開始される必要があります。SRVresynch には、IMS が認知している同期化された T パイプをすべて含みます。

### REQresynch (Request resynch)

REQresynch は、同期化された T パイプごとに IMS からクライアントに発行する必要があります。REQresynch には、T パイプ名、T パイプ用の IMS リカバリー可能送信シーケンス番号、および T パイプ用の IMS リカバリー可能受信シーケンス番号を含みます。

### REPresynch (Reply resynch)

REPresynch は、クライアントから T パイプごとに発行されます。REPresynch は、IMS から REQresynch 要求への応答です。

### TBresynch (tpipe\_Bid resynch)

Tpipe\_Bid resynch は、クライアントによって発行され、特定の T パイプに対して IMS で再同期を開始します。

IMS は、T パイプ構造中の送信および受信番号を追跡します。送信および受信番号は、入力および出力メッセージごとに更新されます。同期化が発生すると、クライアントと IMS の両者は、それらの送信および受信番号を共用して、両サイドが同期していることを検査します。IMS からの REQresynch コマンドは、IMS からの送信および受信番号をリリースします。クライアントは、その番号を受け入れ、クライアントの送信および受信番号との比較を行います。送信および受信番号が同じでない場合は、クライアントは、REPresynch により、IMS にアクションを指定します。両サイドが、同じ送信および受信番号の場合は、同期化は正常に完了します。同期化が失敗した場合は、失敗した T パイプを識別し、使用しないようにします。

再同期のコマンド・メッセージ通信の後には、OTMA 再同期プロトコルが続きます。通常、再同期中に起こるイベントのシーケンスは、以下のとおりです。

1. クライアントは、クライアントが IMS と再同期化しようとするとき、CBresynch を発行します。
2. IMS は、CBresynch の受信の確認に ACK を送信します。この時点から、IMS は、すべての同期化 T パイプに対して、入力または出力の非同期化タイプを静止します。同期化トランザクション・パイプ (T パイプ) に対する再同期が進行中に、IMS が入力を受信した場合は、入力はセンス・コード X'0025' でリジェクトされます。
3. IMS は SRVresynch コマンドを構築し、それをクライアントに送信します。SRVresynch コマンドは、IMS がそのクライアントに対して認知する同期化された T パイプ名をすべてリストします。
4. クライアントは、SRVresynch コマンドを受信し、ACK または NAK メッセージを IMS に出します。
5. IMS が ACK メッセージを受信すると、IMS は T パイプごとに再同期処理を開始します。IMS は、T パイプ名、T パイプ用の IMS リカバリー可能送信シーケンス番号、および T パイプ用の IMS リカバリー可能受信シーケンス番号を含む REQresynch コマンドを送信します。

IMS が、SRVresynch コマンドから NAK メッセージを受信すると、IMS は、DFS2393 メッセージを MTO に送信し、クライアントからクライアント・ビッド要求または CBresynch コマンドを待ちます。

6. クライアントは REQresynch 要求を受信します。REQresynch 要求からの情報と自身の T パイプの情報を比較することにより、クライアントは、REPresynch 応答を IMS に送信し、T パイプについて IMS に通知します。
7. IMS は REPresynch 応答を受信し、クライアントからの情報に基づいて、T パイプ上でアクションを取ります。IMS は、クライアントが示したアクションを取った後、ACK メッセージをクライアントに送信します。IMS は、T パイプが入出力を操作できるようにします。IMS がクライアントの要求を実行できない場合は、IMS は T パイプを停止し、NAK メッセージをクライアントに送信します。
8. 複数の T パイプが存在する場合は、再同期処理が並列して T パイプごとに繰り返されます。クライアントが SRVresynch コマンドを受信した後、SRVresynch 要求に組み込まれていない他の T パイプは、いつでも、いずれの方向にも出力を送信できます。

以下の図は、遅延なし再同期のフローを示します。図の後に、ハイレベル・フローの説明を提供する順次リストがあります。

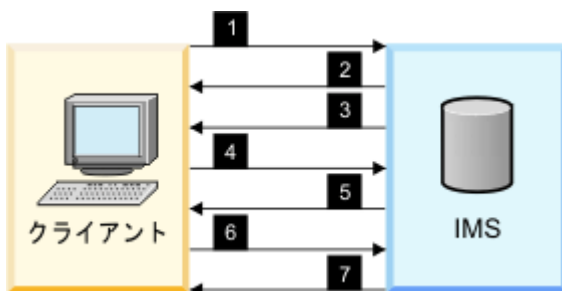


図 152. 再同期のフロー (遅延なし)

1. 再同期を伴うクライアント・ビッド要求
2. ACK メッセージ
3. SRVresynch コマンド
4. ACK メッセージ
5. REQresynch コマンド
6. REPresynch コマンド
7. ACK または NAK メッセージ

クライアントが、特定の T パイプに対して再同期の据え置きを決めた場合は、T パイプのイベント・シーケンスは少し違ってきます。

REPresynch コマンドで、クライアントは、「再同期の停止と待ち」の標識をセットすることができ、IMS に、クライアントからの TBresynch コマンドを待つ間、入力あるいは出力を据え置くように、要求することができます。最初の 4 ステップが完了したと想定すると、後続のイベントは以下のようになります。

1. IMS は、T パイプ名、IMS リカバリー可能送信シーケンス番号、および IMS リカバリー可能受信シーケンス番号を含む REQresynch コマンドを送信します。
2. クライアントは REQresynch 要求を受信します。ただし、製品固有の理由により、クライアントは、「TBresynch の停止と待ち」の標識をオンにして REPresynch コマンドを送信することによって、この T パイプに対する再同期を据え置きます。
3. IMS は、REPresynch コマンドの受け取りを確認するために、ACK メッセージを送信し、TBresynch を待ちます。しばらくして、IMS は、T パイプの入出力を静止します。IMS が TBresynch を待っている間に入力を受信する場合は、IMS は、NAK メッセージをセンス・コード X'0025' でクライアントに送信します。
4. クライアントは、TBresynch コマンドを送信して、IMS にこの T パイプに対する再同期を再開するよう要求します。
5. IMS は、T パイプ名、IMS リカバリー可能送信シーケンス番号、および IMS リカバリー可能受信シーケンス番号を含む REQresynch コマンドを送信します。関連する T パイプが、クライアントの TBresynch コマンドを使用して位置指定できない場合は、NAK メッセージをセンス・コード X'0025' で受信します。
6. クライアントは REQresynch 要求を受信します。REQresynch 要求からの情報と自身の T パイプの情報を比較することにより、クライアントは、REPresynch 応答を IMS に送信し、T パイプについて IMS に通知します。
7. IMS は REPresynch 応答を受信し、クライアントが要求した情報に基づいて、T パイプにアクションを取ります。IMS は、クライアントが示したアクションを取った場合は、ACK メッセージをクライアントに送信します。それ以外は、IMS は、NAK メッセージをセンス・コード X'0025' または X'0026' でクライアントに送信します。

以下の図は、据え置かれた再同期のフローを示します。図の後に、ハイレベル・フローの説明を提供する順次リストがあります。

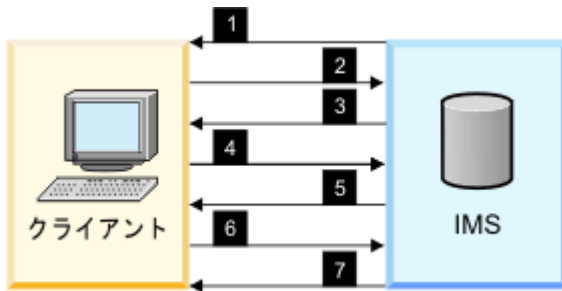


図 153. 再同期のフロー (据え置き)

1. REQresynch コマンド
2. TBresynch に対する STOP AND WAIT を伴う REPresynch コマンド
3. ACK メッセージ
4. TBresynch コマンド
5. REQresynch コマンド
6. REPresynch コマンド
7. ACK または NAK メッセージ

#### 関連資料

899 ページの『状態データ・セクション』

状態データは、すべての OTMA メッセージで必須です。これは、メッセージ接頭語のメッセージ制御情報セクションのすぐ後にあります。トランザクション関連情報が入っています。

## OTMA 再同期メッセージ・フローのサンプル

以下の図は、同期化されたトランザクション・パイプを介したメッセージ・フローを示します。

IMS およびクライアント各サイドの受信および送信のシーケンス番号は、文字 **R** および **S** で表し、メッセージ接頭語のメッセージ制御情報セクションにセットされます。これらの番号はメッセージ全体に適用されます (複数セグメント・メッセージを含む)。**R** および **S** は、必ずしも関連しません。

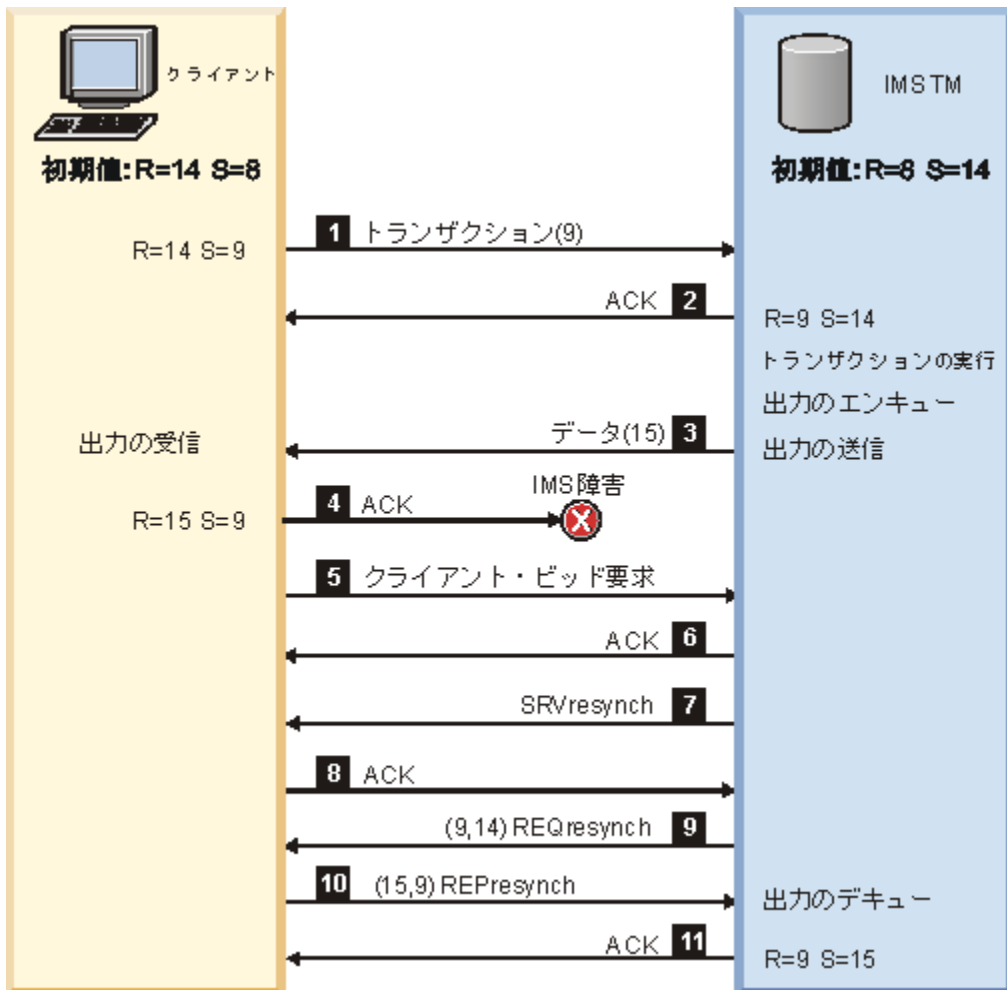


図 154. OTMA 再同期メッセージ・フローのサンプル

- **1** クライアントがトランザクションをサブミットした後、IMS は、トランザクションをエンキューし、トランザクションが実行される。受信シーケンス番号が、1 増加する。
- **2** IMS は、クライアントに ACK メッセージを送信し、受信およびエンキューしているトランザクションを確認する。
- **3** IMS は、出力をエンキューし、データをクライアントに送信する。
- **4** クライアントは、ACK メッセージを IMS に送信して、出力を受信していることを確認する。しかし、IMS 障害のため、IMS は、メッセージ 15 に対するこの ACK を受信することはない。  
再同期は以下のように進行します。
- **5** クライアントは、クライアント・ビッド要求を IMS に送信して、再同期を開始する。
- **6** IMS は、クライアントに ACK メッセージを送信して、再同期を開始しようとする。
- **7** IMS は、SRVresynch コマンドをクライアントに送信して、再同期を開始する。
- **8** クライアントは、ACK メッセージを IMS に送信して、SRVresynch コマンドを受信していることを確認する。
- **9** IMS は、REQresynch コマンドをクライアントに送信して、受信および送信のシーケンス番号を (9,14) に更新する。
- **10** クライアントは、REPresynch コマンドを IMS に送信して、受信および送信のシーケンス番号を (15,9) に更新し、IMS に最後の出力メッセージをデキューするよう指示する。IMS は、メッセージ 15 をデキューし、S を 15 に更新する。
- **11** IMS は、ACK メッセージをクライアントに送信する。

## OTMA 再同期メッセージのサンプル

このトピックでは、OTMA 再同期メッセージのサンプルを示します。

### 再同期メッセージを伴うクライアント・ビッド要求

以下の例には、再同期メッセージを使用した OTMA クライアント・ビッド要求を示します。

MESSAGE CONTROL INFORMATION:

```
01102000 0C004040 40404040 4040A0C0 | ..... .{ |
00000000 00000000 00000000 00000000 | ..... |
```

STATE DATA:

```
0036D4D8 E7C3C6F6 40404040 40404040 | ..MQXCF6 |
40400100 00010002 00010100 00020002 | ..... |
0002C4C6 E2E8C4D9 E4F00800 00007FFF | ..DFSYDRU0...". |
FFFF0000 00650056 C3525100 5001A051 | .....C...&... |
```

### CBresynch の受け取りを確認する ACK メッセージ

以下の例には、CBresynch の受け取りを確認するための ACK メッセージを示します。

MESSAGE CONTROL INFORMATION:

```
01308000 0C004040 40404040 4040A0C0 | ..... .{ |
00000000 00000000 00000000 00000000 | ..... |
```

STATE DATA:

```
0036D4D8 E7C3C6F6 40404040 40404040 | ..MQXCF6 |
40400100 00010002 00010100 00020002 | ..... |
0002C4C6 E2E8C4D9 E4F00800 00007FFF | ..DFSYDRU0...". |
FFFF0000 00650056 C3525100 5001A051 | .....C...&... |
```

### SRVresynch コマンド・メッセージ

以下の例は、SRVresynch コマンド・メッセージを示します。

MESSAGE CONTROL INFORMATION:

```
01102000 2C000000 00000000 0000A080 | ..... |
00000000 00000000 00000000 00010000 | ..... |
```

STATE DATA:

```
000AD1C2 D1F0F0F0 F0C50000 00000000 | ..JBj0000E..... |
```

### SRVresynch の受け取りを確認する ACK メッセージ

以下の例には、SRVresynch 受け取りを確認するため、クライアントが送信した ACK メッセージを示します。

MESSAGE CONTROL INFORMATION:

```
0130A000 2C000000 00000000 0000A080 | ..... |
00000000 00000000 00000000 00010000 | ..... |
```

STATE DATA:

```
000AD1C2 D1F0F0F0 F0C50000 00000000 | ..JBj0000E..... |
```

## REQresynch コマンド・メッセージ

以下の例には、REQresynch コマンド・メッセージを示します。

MESSAGE CONTROL INFORMATION:

```
01100000 3000D1C2 D1F0F0F0 F0C5A080 |.....JBJ0000E..|
00000000 00000000 00000000 00000000 |.....|
```

STATE DATA:

```
001AD1C2 D1F0F0F0 F0C50000 00020000 |..JBJ0000E.....|
00020000 00000000 00000000 00000000 |.....|
```

## REPresynch コマンド・メッセージ

以下の例には、REPresynch コマンド・メッセージを示します。

MESSAGE CONTROL INFORMATION:

```
01100000 3400D1C2 D1F0F0F0 F0C5A080 |.....JBJ0000E..|
00000003 00000000 00000000 00000000 |.....|
```

STATE DATA:

```
001AD1C2 D1F0F0F0 F0C50000 00020000 |..JBJ0000E.....|
00020000 00000000 00000000 00000000 |.....|
```

## 正常な再同期の ACK メッセージ

以下の例には、T パイプ上の再同期が正常に完了したことをクライアントに通知するために、IMS が送信する ACK メッセージを示します。

MESSAGE CONTROL INFORMATION:

```
01308000 3400D1C2 D1F0F0F0 F0C5A080 |.....JBJ0000E..|
00000003 00000000 00000000 00000000 |.....|
```

STATE DATA:

```
001AD1C2 D1F0F0F0 F0C50000 00020000 |..JBJ0000E.....|
00020000 00000000 00000000 00000000 |.....|
```

## 関連資料

889 ページの『OTMA メッセージ接頭語』

OTMA メッセージには、IMS.ADFSMAC データ・セットの DFSYMSG マクロによってマップされるフォーマットに準拠した接頭部があります。

## コミット後送信出力の管理

OTMA は、保留キュー対応 OTMA クライアントに即時に返信されないコミット後送信 (CMO) 出力を管理するために、さまざまなオプションを備えています。出力をページするか、出力を転送するか、あるいは OTMA スーパーメンバーを使用して複数の OTMA クライアント間で出力を共有することができます。

## コミット後送信非同期出力のページ

I/O PCB 出力を OTMA クライアントに送信できないとき、非同期のコミット後送信メッセージ出力をサポートする OTMA クライアントは、T パイプ・キューからの OTMA ページ出力を持つことができます。

### このタスクについて

ページ機能を指定するには、OTMA クライアントは、元の入力メッセージか、あるいは出力が取り出せなかったときの NAK 応答のどちらかの OTMA 状態データ接頭部に、ページ・フラグを設定する必要があります。

### NAK 応答出力のページ

保留キュー対応の OTMA クライアントでは、OTMA は、クライアントから NAK 応答を受け取ったときに、コミット後送信 (CM0) I/O PCB 出力をページできます。

### このタスクについて

NAK メッセージが OTMA 状態データ接頭部内のページ要求フラグをオンにした場合、OTMA はコミット後送信 I/O PCB 出力をデキューします。

NAK メッセージに、ページ要求も転送要求も指定されていない場合、OTMA は I/O PCB 出力を再び入力 T パイプの非同期保留キューに入れます。T パイプの非同期保留キューに格納されたメッセージは、あとから RESUME TPIPE 呼び出しによって取り出すことができます。

NAK メッセージに、ページ要求と転送要求の両方が指定されている場合、OTMA は出力を再び入力 T パイプの非同期保留キューに入れ、システム・コンソールにエラー・メッセージを送ります。

初期入力メッセージにページ要求が指定されている場合に、OTMA が、z/OS システム間カップリング・ファシリティ 障害あるいはクライアント 障害のため保留キュー対応クライアントにコミット後送信 I/O PCB 出力を送信できなければ、OTMA は出力をページします。

OTMA は保留キューからの非同期出力のページはサポートしません。

## コミット後送信非同期出力の転送

出力が OTMA クライアントに送信されないときに、OTMA 保留キューをサポートする OTMA クライアントは、OTMA に対して、I/O PCB 出力を代替保留キューに転送させることができます。

### このタスクについて

OTMA クライアントは、OTMA に入力された次のメッセージ・タイプの転送機能を指定することができます。

- コミット後送信トランザクションの入力メッセージ
- Send-Only トランザクションの入力メッセージ
- コミット後送信出力に対する NAK 応答

OTMA クライアント (T メンバー) に対して T パイプの最大数が定義されており、その T メンバーに関連付けられた T パイプの数が最大数に達した場合、メッセージを転送するために新規の T パイプを作成することができません。この場合、T パイプの最大数に達したためにメッセージを転送できないと、OTMA はそのメッセージを削除します。

T メンバー用の T パイプの最大数は、DFSYDTx PROCLIB メンバー内の OTMA クライアント記述子の MAXTP パラメーターで指定します。



## 送信専用トランザクションの非同期出力の転送

保留キュー対応の OTMA クライアントからの Send-Only トランザクションでは、OTMA は I/O PCB 出力キューのコミット後送信 (CM0) メッセージを転送することができます。

### このタスクについて

送信専用トランザクション・メッセージ内に非同期コミット後送信出力の転送要求を指定するには、OTMA メッセージ接頭語のユーザー・データ・セクションで、出力の転送先にしたい代替の非同期保留キューの T パイプ名を指定します。IMS Connect ユーザーについては、285 ページの『IMS Connect で使用される OTMA ユーザー・データ・フィールド』で説明されている OMUSR\_REROUT\_NM を参照してください。

転送が指定されていると、OTMA は Send-Only メッセージの入力 T パイプに代わり転送 T パイプに出力を転送します。

また、OTMA は送信専用トランザクションで生成された IMS DFS メッセージを代替 T パイプに転送します。

しかし、Send-Only トランザクションが入力メッセージに対する ACK または NAK を要求すると、OTMA は入力トランザクションがキューに入ってから ACK を返します。入力された送信専用トランザクションをキューに入れることができない場合、OTMA は DFS メッセージなしで NAK を返します。

### NAK 応答出力の転送

保留キュー対応の OTMA クライアントでは、OTMA は、クライアントから NAK 応答を受け取ったときに、I/O PCB T パイプ・キューか保留キューのどちらかにあるコミット後送信 (CM0) メッセージを転送することができます。

### このタスクについて

NAK メッセージが転送フラグをオンにして、状態データ接頭部の転送 TPIPE フィールドに転送 T パイプ名を指定すると、OTMA は、それを転送 TPIPE フィールドで指定した非同期出力保留キューに再度キューすることにより、I/O PCB 出力キューにあるコミット後送信メッセージを転送します。

NAK メッセージに、ページ要求と転送要求の両方が指定されている場合、OTMA はシステム・コンソールにエラー・メッセージを送り、出力を再び元の入力メッセージで使った T パイプの非同期保留キューへ入れます。

転送要求が初期入力メッセージに指定されている場合に、OTMA が z/OS システム間カップリング・ファシリティー 障害あるいはクライアント障害のため保留キュー対応クライアントにコミット後送信 I/O PCB 出力を送信できなければ、OTMA は出力を再び入力 T パイプの非同期保留キューに入れます。

NAK メッセージの中で OTMA に非同期コミット後送信出力を転送するよう要求するには、OTMA ヘッダーのトランザクションに関連した状態データのバイト 5 に 'X'20' を設定し、出力の転送先にする代替の非同期保留キューの T パイプ名を状態データのバイト 62 に指定します。

## コミット後送信出力の確認応答のタイムアウト

OTMA クライアントからの確認応答のタイムアウト間隔を指定できます。そのタイムアウトの経過後、OTMA は、T パイプからコミット後送信メッセージを除去し、それをデフォルトのタイムアウト・キューまたは指定されたキューに入れます。

指定したタイムアウト値は、コミット後送信出力および送信後コミット出力の両方の確認応答に適用されます。ただし、タイムアウト間隔の有効期限が切れると、OTMA は、送信後コミット出力の場合のように出力を廃棄する代わりに、コミット後送信出力を転送します。

コミット後送信出力の確認応答の有効期限が切れると、OTMA は指定された転送 T パイプ保留キュー、指定されたタイムアウト T パイプ保留キュー、または OTMA デフォルト・タイムアウト T パイプ保留キュー DFS\$STOQ に出力を転送します。

### 関連タスク

825 ページの『OTMA メッセージへの確認応答のタイムアウト間隔』

OTMA が OTMA 出力メッセージに対する ACK または NAK の確認応答を待機する時間の長さを決める、ACK タイムアウト間隔を指定できます。

## 非同期コミット後送信出力の共用: OTMA スーパーメンバー機能

保留キュー対応 OTMA クライアント (IMS Connect など) は、OTMA のスーパーメンバー 機能を有効にすることにより、非同期コミット後送信 (CMO) 出力メッセージを共用できます。OTMA スーパーメンバー 機能は、z/OS Sysplex Distributor 環境で IMS Connect の複数のインスタンスをサポートするために、特に設計されたものです。

### このタスクについて

OTMA スーパーメンバー 機能は、共通の出力キューを使用することにより、参加しているすべての OTMA クライアントのすべての非同期 CMO 出力を管理します。参加しているすべての保留キュー対応クライアントは、独自の RESUME TPIPE 呼び出しを発行することにより、スーパーメンバー 出力キューに入っている CMO メッセージを取り出すことができ、その際、どのクライアントがその CMO 出力の当初の宛先だったかは問題になりません。

スーパーメンバー・キューにあるメッセージは、どの OTMA クライアント・インスタンスが出力を発信したか、またはどの OTMA クライアント・インスタンスが RESUME TPIPE 呼び出しを発行したかを一切考慮せずに、先入れ先出し法によって取り出されます。

参加している OTMA クライアントからの RESUME TPIPE 呼び出しでは、スーパーメンバーに関することを指定する必要はありません。スーパーメンバー 機能は RESUME TPIPE 呼び出しが、参加している OTMA クライアントから発行されていることを認識し、共通キューに出力を返します。

IMS Connect インスタンスのスーパーメンバー 機能への参加を登録するには、IMS Connect HWS 構成ステートメントの SMEMBER パラメーターで、スーパーメンバー 名を指定します。

複数の IMS システムを含む構成内で OTMA スーパーメンバーを使用するには、すべての IMS システムで共用キューを使用可能にしておく必要があります。共用キュー・バックエンド IMS がスーパーメンバー 対応環境で ALT-PCB 出力を作成した場合は、スーパーメンバー・セット内の OTMA クライアントにより、その出力を任意のフロントエンド IMS からリトリブできます。共用キュー環境では、RESUME TPIPE 呼び出しのすべてのリトリブ・オプションがサポートされます。

複数の OTMA クライアントを持つ IMS システムが 1 つしかない場合、共用キューのサポートは必要ありません。

スーパーメンバー 機能をアクティブにするには、IMS Connect 構成メンバー (HWSCFGxx) で、HWS 構成ステートメントの SMEMBER パラメーターに 1 文字から 4 文字までのスーパーメンバー 名を指定します。スーパーメンバー 名は固有でなければならず、既存の OTMA メンバー名と同じ名前にすることはできません。

スーパーメンバー 名を指定する最初の IMS Connect インスタンスは、スーパーメンバー 機能をアクティブにすると同時に、IMS Connect の他の全インスタンスに対してそのスーパーメンバー 名を定義します。その後アクティブにされ、同じスーパーメンバー・キューを使用する IMS Connect インスタンスは、最初の IMS Connect インスタンスが指定したのと同じスーパーメンバー 名を指定する必要があります。

以下の IMS Connect コマンドのいずれかを発行して、アクティブにされているスーパーメンバーの 名前を表示できます。

- IMS タイプ 2 フォーマット・コマンド **QUERY IMSCON TYPE(CONFIG)**
- WTOR フォーマット・コマンド **VIEWHWS**
- z/OS MODIFY フォーマット・コマンド **QUERY MEMBER TYPE(IMSCON)**

スーパーメンバーに関する追加情報を表示するために、IMS コマンドの **/DISPLAY OTMA** および **/DISPLAY TMEMBER TPIPE** を発行することもできます。

正規の OTMA メンバーを指定して、OTMA コマンドの **/START TMEMBER TPIPE**、**/STOP TMEMBER TPIPE**、または **/TRACE TMEMBER TPIPE** を発行すると、OTMA は関連するスーパーメンバーがあれば、それらが含まれるよう、コマンドの範囲を拡張します。

OTMA コマンドを、既存のスーパーメンバーに対して直接発行することもできます。例えば、スーパーメンバーから非同期メッセージをデキューするには、**/DEQUEUE TMEMBER supermember\_name TPIPE tpipe\_name PURGE** コマンドを発行します。

ALT-PCB 処理によって作成された非同期出力メッセージは、スーパーメンバー 内に直接格納されます。入力メッセージが、スーパーメンバーをサポートする IMS Connect を通じてサブミットされた場合、OTMAYPRX ユーザー出口および DFSYDRU0 出口ルーチンの入力パラメーター・リストには、そのメッセージがスーパーメンバーをサポートするクライアントからのものであることを示すフラグが含まれています。また、入力パラメーター・リストが指す OTMA 状態データ接頭部に、スーパーメンバー名 (ある場合) が含まれています。

## 非同期保留キューへの出力の表示

OUTPUT キーワードを指定して **/DISPLAY TMEMBER** コマンドを発行することにより、非同期保留キューにメッセージ数を表示することができます。

### このタスクについて

OTMA が保留キュー対応の場合 (例えば、IMS Connect) は、**/DISPLAY TMEMBER TPIPE OUTPUT** コマンドを使用して、1 次キューと保留キューの両方の出力カウントを表示することができます。

## 第 45 章 コールアウト要求の OTMA サポート

IMS 従属領域で実行されている IMS アプリケーション・プログラムは、IMS の Open Transaction Manager Access (OTMA) コンポーネントを介して、データまたはサービスを要求する目的で、IMS インストールの外部にあるサーバーに対してコールアウトすることができます。同期コールアウト・インターフェースを使用して、同期プログラム間通信により別の IMS アプリケーションからのサービスを要求することもできます。

このようなコールアウト要求をサポートする IMS 構成では、複数のコンポーネントが必要です。必要なコンポーネントは、コールアウト要求が同期で処理されるか非同期で処理されるかによって異なります。

同期プログラム間通信要求の場合は、OTMA は必要ありません。同期プログラム間通信要求は OTMA トランザクションとしてキューに入れられますが、IMS は OTMA 送信後コミット (CM1) プロトコルの内部実装を使用して、要求を指定されたトランザクションに透過的に経路指定します。IMS.PROCLIB データ・セットの DFSPBxxx メンバーで OTMA=Y を指定するか、START OTMA コマンドを発行して、OTMA クライアントとの XCF 接続を開始する必要はありません。

### IMS アプリケーション・プログラムからのコールアウト要求

OTMA は、保留キュー対応可能な OTMA クライアント (IMS Connect など) とともに、IMS 従属領域の中で実行する IMS アプリケーション・プログラムから、IMS インストールの外部にあるデータまたはサービスのプロバイダーへのコールアウト要求をサポートします。

外部プロバイダーには、以下のようなものを含めることができます。

- Web アプリケーション・サーバーで実行している Enterprise JavaBeans (EJB) アプリケーションまたはメッセージ駆動型 Bean (MDB) アプリケーション (WebSphere Application Server)
- Web サービス
- ユーザー作成 IMS Connect クライアントを介してアクセスされるデータ・ソースまたはサービス

サービスまたはデータに対する要求は、コールアウト要求と呼ばれます。IMS アプリケーション・プログラムがコールアウト要求を行うとき、IMS はクライアント/サーバー関係の中のクライアントとして表示されます。この中では、サーバーは、IMS がコールアウト要求を行う相手の外部アプリケーションです。

OTMA を介して経路指定されるコールアウト要求は、同期または非同期で処理できます。

同期コールアウト要求の場合、IMS 従属領域で実行している IMS アプリケーション・プログラムは DL/I ICAL 呼び出しを発行して、従属領域の中で、応答も処理するために待機します。ICAL 呼び出しが発行されると、IMS は同期コールアウト要求に対して相関トークンを生成し、要求を OTMA 宛先に経路指定します。相関トークンはコールアウト要求とともに含められ、要求側の IMS アプリケーション・プログラムにそれ自体の応答を戻すよう経路指定するために、応答とともに IMS に戻されなければなりません。

非同期コールアウト要求の場合、IMS 従属領域で実行している IMS アプリケーション・プログラムは、コールアウト・メッセージを ALTPCB キューに挿入してから従属領域を解放するために終了します。IMS は非同期コールアウト要求には相関トークンを生成しません。このため、コールアウト要求の応答が必要な場合は、元のコールアウト要求への応答の相関を IMS アプリケーション・プログラムで管理する必要があります。IMS は、非同期コールアウト要求への応答を受信すると、その応答を新しいトランザクションとして処理します。

同期コールアウト要求に関する構成要件は、非同期コールアウト要求に関する構成要件とは幾分異なりますが、基本的なコンポーネントは同じで、以下のとおりです。

- IMS の外部にあるソースに対してデータまたはサービスをコールアウトする IMS アプリケーション。同期コールアウト要求では、この同じアプリケーションが応答を処理します。非同期コールアウト要求では、応答が必要な場合、応答は同じアプリケーション・プログラムの別のインスタンスで処理されるか、別のアプリケーション・プログラムでまとめて処理されます。
- OTMA。以下のいずれかの方法を使用して定義された宛先に基づいて、適切なクライアント T パイプ・キューにコールアウト要求を経路指定します。

– OTMA 宛先記述子

- タイプ 2 コマンド
- 非同期コールアウト要求の場合、OTMA 宛先解決ユーザー出口 (OTMAYPRX) および OTMA ユーザー・データ・フォーマット設定出口ルーチン (DFSYDRU0)
- 保留キュー対応 OTMA クライアント。IMS Connect など。IMS Connect は、IMS への TCP/IP ゲートウェイと、OTMA とのインターフェースの両方の役割を果たします。
- IMS Connect を使用している場合、IMS Connect TCP/IP クライアント (IMS TM Resource Adapter など)、IMS Enterprise Suite SOAP Gateway、またはユーザー作成 IMS Connect クライアント。
- データまたはサービスのプロバイダー (EJB アプリケーションまたは Java プラットフォームの MDB アプリケーション、Enterprise Edition (Java EE) 環境、または Web サービスなど)。

同期コールアウト構成でも非同期コールアウト構成でも、セキュリティは RACF、OTMA RESUME TPIPE セキュリティ・ユーザー出口 (OTMARTUX)、またはその両方などのセキュリティ製品によりインプリメントできます。

### 関連概念

797 ページの『OTMA 宛先記述子』

OTMA 宛先記述子は、OTMA を介して経路指定されるメッセージの宛先を定義するために使用します。

## 同期コールアウト要求

同期コールアウト要求はリアルタイムで処理され、IMS 従属領域で実行している IMS アプリケーション・プログラムから外部のデータまたはサービスのプロバイダーに送信され、IMS アプリケーション・プログラムが IMS 従属領域の中でスケジュールされているうちに IMS アプリケーション・プログラムに戻されません。

ご使用の環境を同期コールアウト用に構成する方法は、非同期コールアウト用に構成するよりも簡潔です。これは、要求を行った IMS アプリケーションへのコールアウト要求の相関を IMS が管理するためです。

### 制約事項:

- OTMA は、フロントエンドの IMS システムとバックエンドの IMS システムを持つように構成された共用キュー環境での同期コールアウト要求をサポートしません。
- OTMA 入出力編集 ユーザー出口 (OTMAIOED) は、DL/I ICAL 呼び出しを発行した IMS アプリケーション・プログラムから OTMA が受信した同期コールアウト要求メッセージについても、同期コールアウト応答メッセージについてもサポートされません。

OTMA は、同期コールアウト要求メッセージをリカバリー不能なコミット後送信 (CM0) 出力メッセージとして処理します。ただし、IMS は、同期プログラム間通信要求を送信後コミット・モード (CM1) で処理します。同期プログラム間通信要求は、IMS Connect などの外部クライアントには経路指定されず、それらを使用するために OTMA を使用可能にする必要はありません。

出力コールアウト・メッセージのスループットを向上させるには、複数のアクティブ RESUME TPIPE 要求に対する OTMA サポートを使用可能にすることを検討してください。これにより、単一の OTMA T パイプが複数のクライアントに並列で出力を送信できるようになります。複数のアクティブ RESUME TPIPE 要求に対するサポートは、OTMA クライアント記述子で使用可能にすることも、IMS Connect システムまたはデータ・ストアの定義で使用可能にすることもできます。

以下の概略的なステップでは、IMS Connect を OTMA 保留キュー対応クライアントとして使用するとき、同期コールアウト環境を構成する方法の概要を示します。

1. OTMA 宛先記述子を、コールアウト要求を、IMS Connect クライアントの T パイプ (コールアウト要求を取り出すように構成されている) に経路指定するようにコーディングします。
2. IMS を再始動してください。
3. IMS Connect ユーザー・メッセージ出口ルーチンを変更する必要がある場合は、IMS Connect ユーザー・メッセージ 出口ルーチンを再アセンブルしてバインドします。
4. IMS Connect ユーザー・メッセージ出口ルーチンを変更した場合は、IMS Connect を再始動します。
5. コールアウト要求を取得するように IMS Connect TCP/IP クライアントを構成し、この要求をデータまたはサービスのプロバイダーに受け渡して、その応答を IMS Connect に戻します。IMS Connect TCP/IP クライアントの構成方法について詳しくは、以下のリストの中から該当する文書を参照してください。



- [IMS TM Resource Adapter の概要](#)
- [コールアウト・プログラミング・モデル \(TM Resource Adapter\)](#)
- [IMS Enterprise Suite SOAP ゲートウェイの概要](#)
- [同期コールアウト要求のためのユーザー作成 IMS Connect クライアントの構成](#)

6. DL/I ICAL 呼び出しを発行してコールアウト要求を開始し、その応答を処理するように、IMS アプリケーション・プログラムをコーディングします。

コールアウト・サービス・プロバイダーが IMS Connect などの OTMA クライアントを介して IMS に接続する場合、同期コールアウト応答は、送信専用プロトコルを使用して OTMA および IMS に返送されます。

オプションとして、コールアウト・サービス・プロバイダーは、ACK 付き送信専用プロトコルを使用できます。この場合、OTMA は、コールアウト・サービス・プロバイダーからの各応答に対して ACK 応答または NAK 応答を発行する必要があります。また、ACK 付き送信専用プロトコルでは、コールアウト・サービス・プロバイダーは、ACK メッセージまたは NAK メッセージをリトリブするために追加の受信を実行することも必要です。

デフォルトでは、ACK 付き送信専用プロトコルが使用されている場合、OTMA は応答メッセージからのデータを各 ACK メッセージに含めます。各 ACK から応答データを省略することも可能です。これを行うには、OTMA メッセージ接頭語の状態データ・セクションのバイト 3 にある「同期フラグ」または「コールアウト・フラグ」フィールドに 'X'10' を指定します。

IMS には、コールアウト・サポートをテストするためのさまざまなサンプル・アプリケーション・プログラムが用意されています。詳しくは、[コールアウト機能のサンプル \(インストール\)](#) を参照してください。

コールアウト要求のための構成の初期セットアップを完了したら、DL/I テスト・プログラム (DFSDDLTO) を使用して、ご使用の IMS アプリケーション・プログラムとは無関係に、ICAL DL/I 呼び出しを検査およびデバッグすることができます。

### 関連概念

[879 ページの『同期プログラム間通信要求』](#)

要求を別の IMS トランザクションに送信し、同じ作業単位で応答を受信するには、DL/I ICAL 要求を使用します。このような ICAL インターフェースの使用法は、同期プログラム間通信要求と呼ばれます。Java 従属領域リソース・アダプターを使用する Java アプリケーション・プログラムも、JMS API によって同期プログラム間通信要求を発行できます。

[308 ページの『同期コールアウト応答のための送信専用プロトコル』](#)

IMS Connect クライアントは、送信専用プロトコルを使用して、IMS アプリケーション・プログラムからの同期コールアウト要求に対する応答を返します。

### 関連タスク

[209 ページの『同期コールアウト要求のためのユーザー作成 IMS Connect クライアントの構成』](#)

同期コールアウト要求をサポートするには、IMS から新しいコールアウト要求をリトリブし、コールアウト要求の受信を確認して (ACK または NAK)、IMS Connect を介して同期コールアウト応答を IMS に返すように、ユーザー作成 IMS Connect クライアントを構成する必要があります。

[216 ページの『IMS に対するコールアウト応答のリターン』](#)

ユーザー作成の IMS Connect クライアントは、同期コールアウト応答用の送信専用プロトコルを使用して、コールアウト応答メッセージを IMS に返します。

## 同期プログラム間通信要求

要求を別の IMS トランザクションに送信し、同じ作業単位で応答を受信するには、DL/I ICAL 要求を使用します。このような ICAL インターフェースの使用法は、同期プログラム間通信要求と呼ばれます。Java 従属領域リソース・アダプターを使用する Java アプリケーション・プログラムも、JMS API によって同期プログラム間通信要求を発行できます。

ターゲット・トランザクションは別の作業単位で実行されるため、2 フェーズ・コミットまたは z/OS リソース・リカバリー・サービスには適格ではありません。

ターゲット・トランザクションは、同じ IMS システム内、複数システム結合機能 (MSC) を使用するリモート IMS システム内、または共用キューによってアクセス可能な IMS システム内にある場合があります。

IMS は、ICAL 呼び出しのトランザクションを OTMA トランザクションとしてスケジュールに入れます。同期プログラム間通信要求を行うために、OTMA を使用可能にする必要はありません。

同期プログラム間通信機能を使用するには、まず、宛先アプリケーションに TYPE=IMSTRAN を指定して OTMA 宛先記述子を構成する必要があります。宛先記述子は、IMS.PROCLIB データ・セットの DFSYDTx メンバーに設定するか、**CREATE OTMADESC** コマンドを使用して作成することができます。

TMEMBER、TPIPE、および SMEM の各パラメーターは、IMSTRAN 記述子タイプに対して、他の記述子タイプとは異なる動作をします。これらのパラメーターは、ターゲット・アプリケーション・プログラムからの遅延応答メッセージのデフォルト宛先をオプションで指定するために使用されます。

以下の図は、5 つの手順から成る同期プログラム間通信処理モデルを示しています。

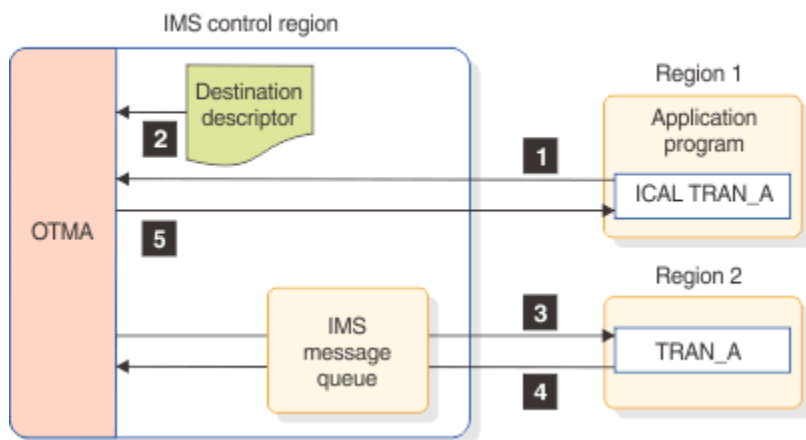


図 155. IMS 同期プログラム間通信処理

1. 領域 1 で実行されているアプリケーション・プログラムは、TRAN\_A に対して ICAL を発行します。ICAL 要求は、IMS 制御領域に経路指定されます。
2. IMS 制御領域は、要求に指定されている宛先記述子を読み取ります。
3. IMS メッセージ・キューを使用して、送信後コミット (CM1) プロトコルを使用する、領域 2 内の TRAN\_A トランザクションがスケジュールに入れられます。OTMA ルーティング機能は、IMS システムで OTMA が使用可能になっていない場合でも使用されます。
4. ターゲット・アプリケーションは処理を終了すると、IMS 制御領域に応答メッセージを返します。
5. IMS 制御領域は、領域 2 からの応答メッセージによって、領域 1 内の発信元の DL/I ICAL 呼び出しに応答します。

注：従属領域は、MPP、JMP、IFP、BMP、または JBP のいずれかです。

IMS は、同期プログラム間通信メッセージに OTMA メッセージ・ヘッダー・フォーマットを使用します。状態データ・ヘッダーで、コミット・モードは 1 に設定されており、同期レベルは CONFIRM に設定されています。TMAMHCOR フィールドは、ICAL 呼び出しが発行された IMS 領域からの LCRETOKN 値に設定されています。ただし、同期プログラム間通信が OTMA トランザクションから開始された場合、IMS は、元のトランザクションからの情報を挿入します。すなわち、TMAMHCOR は元の相関関係子の値に設定され、TMAMHCID は元のクライアント・メンバー名に設定され、TMAMRTOD は元の T パイプ名に設定されます。

PSTUSID からのユーザー ID 名と PSTGRPNM からのグループ名は、セキュリティー・データ・ヘッダーに含まれます。

元のトランザクションが OTMA トランザクションでなく、メッセージの OTMA 宛先記述子に遅延メッセージ・ルーティング情報が含まれていない場合、ユーザー・データ・ヘッダーは必要ありません。ICAL 呼び出しが、ユーザー・データ・ヘッダー情報を含む OTMA トランザクションからのものである場合は、同じユーザー・データ・ヘッダー情報が、同期プログラム間通信メッセージに使用されます。

制約事項：



- OTMA 入出力編集出口ルーチン (DFSYIOE0) は、同期プログラム間通信要求または応答メッセージに対して呼び出されません。
- TM および MSC メッセージ・ルーティングおよび制御出口ルーチン (DFSMSCE0) は、同期プログラム間通信要求に対して呼び出されません。
- ターゲット・トランザクションは、開始アプリケーション・プログラムの RRS コミット有効範囲の一部ではありません。
- BMP および JBP アプリケーションは、DBCTL 環境では同期プログラム間通信要求を作成することができません。
- ターゲット・トランザクションには、高速機能 MSDB への読み取り専用アクセス権があります。
- ターゲット・トランザクションは、IMS 会話型トランザクションであってはなりません。
- 共用キュー環境で同期プログラム間通信を使用するには、参加するすべての IMS システムがバージョン 13 以降である必要があります。

## Java アプリケーションからの同期プログラム間通信要求の実行

同期プログラム間通信要求は、IMS Java 従属領域 (JDR) で実行されている Java アプリケーションからも発行できます。IMS Java 従属領域リソース・アダプターは、同期プログラム間通信要求および他のコールアウト要求のフロントエンドとして標準 JMS API を使用します。

### 関連タスク

[同期コールアウト機能の実装 \(アプリケーション・プログラミング\)](#)

[Java 従属領域からの同期プログラム間通信要求の実行 \(アプリケーション・プログラミング\)](#)

### 関連資料

[OTMA 宛先記述子の構文およびパラメーター \(システム定義\)](#)

[ICAL 呼び出し \(アプリケーション・プログラミング API\)](#)

## 非同期コールアウト要求

IMS 従属領域で実行されている IMS アプリケーション・プログラムは、アウトバウンド・メッセージを送信して、WebSphere Application Server EJB アプリケーションまたは Web サービスにサービスまたはデータを要求できます。サービスまたはデータに対する要求は、**コールアウト要求** と呼ばれます。

非同期コールアウト要求では、要求を出す IMS アプリケーション・プログラムが、従属領域で応答を待つ必要はありません。非同期コールアウト要求を発行すると、アプリケーション・プログラムはただちに終了して、従属領域を解放することができます。IMS に返されるコールアウト要求への応答があると、それは新しい着信トランザクションとして扱われ、IMS はそれを処理するための新しいアプリケーション・プログラム・インスタンスをスケジュールに入れます。

しかし、非同期コールアウト要求が応答を生成する場合、従属領域を解放して得られる利点は、その応答を管理する複雑さが加わることで相殺されることがあります。非同期コールアウト応答では、ご使用のシステムが、応答を元の要求に相関させる方法を作成する責任を負います。同期コールアウト要求の場合は、IMS が相関を管理します。

IMS には、コールアウト・サポートをテストするためのさまざまなサンプル・アプリケーション・プログラムが用意されています。詳しくは、[コールアウト機能のサンプル \(インストール\)](#)を参照してください。

## 非同期コールアウト機能の実装

非同期コールアウト機能をサポートする IMS 構成を実装するには、相関、アプリケーション・プログラミング、および OTMA 経路指定の計画を作成する必要があります。

## 非同期コールアウト応答の相関の計画

コールアウト要求によって応答が生成される場合、その応答は、コールアウト要求を生成した元のトランザクションから独立して実行する新しいトランザクションとして扱われます。このため、この応答は、適切な IMS トランザクションに相関させる必要があります。

### このタスクについて

相関を計画するには、相関にどのデータを使用するか、およびコールアウト応答が戻るまで IMS データベースに相関データを格納しておくかどうかを決める必要があります。

また、ISRT *alternate\_pcb* 呼び出しの中の IMS アプリケーションで使用される T パイプ名、および EJB または RESUME TPIPE 呼び出し用の IMS Enterprise Suite SOAP Gateway 接続バンドルで使用される T パイプ名の両方が、同じ OTMA 非同期保留キューを指すことを確認する必要があります。これは、それぞれに同じ名前を使用するか、または OTMA 宛先ルーティング出口ルーチンに適切なロジックをコーディングする方法で行います。T パイプ名は、IMS Connect を介して外部アプリケーションにより提供される代替クライアント ID に基づいて決まります。コールアウト要求を発行するアプリケーション・プログラムは、ISRT *alternate\_pcb* 呼び出しの中で非同期保留キューの T パイプ名を指定する必要があります。

## 非同期コールアウト要求用のアプリケーション・プログラミング

非同期コールアウト要求を発行するアプリケーション・プログラムのコールアウト要求メッセージには、呼び出し対象の外部アプリケーションが必要とするデータを含める必要があります。

外部アプリケーションからの応答の相関が必要な場合、呼び出し側のアプリケーション・プログラムも、相関に必要なとされるデータを収集する必要があります。このアプリケーション・プログラムには、コールアウト要求機能用の専用 OTMA 非同期保留キューの T パイプ名も含まれている必要があります。

コールアウト要求メッセージに何らかの相関データが含まれる場合は、外部アプリケーション・プログラムがそのデータを許容して、応答とともに返送することを確認する必要があります。

オプションで、このアプリケーション・プログラムは、相関のため、または何らかの IMS 処理の結果 (コールアウト応答とともに返送する必要があるもの) として、データベースにデータを保管することもあります。

コールアウト要求に対する応答を処理するアプリケーション・プログラムは、外部アプリケーション・プログラムから戻される相関データがあればそれを処理し、コールアウト・データベースに相関データまたは IMS データが保管されていればそれをリトリブし、適切なキュー、端末、またはユーザーに、適宜最終データを戻します。

IMS TM Resource Adapter を使用している場合、外部でコールアウト要求を処理する Java アプリケーションが、適切な待機時間でループする RESUME TPIPE 呼び出しを発行することにより、OTMA 非同期保留キューを「listen する」ように、このアプリケーションを構成することができます。Java アプリケーション・プログラムも、相関データを保存して、コールアウト応答とともに送信して戻すように設計されていることがあります。

### 関連概念

884 ページの『IMS アプリケーション・プログラムおよび非同期コールアウト機能』

IMS がコールアウト要求への応答を処理する必要があったり、端末がトランザクションのサブミット後に確認応答を必要としたり、応答モードのトランザクションがプログラム間通信を開始したりする場合には、必要に応じて IMS アプリケーション・プログラムを修正または作成して、コールアウト機能をサポートします。

## 非同期コールアウト要求の OTMA 経路指定

非同期コールアウト要求の OTMA 経路指定は、ISRT *alternate\_pcb* 呼び出しを適切な T パイプに経路指定するために必要です。

OTMA 経路指定は、OTMA 宛先記述子または OTMA 宛先解決 ユーザー出口 (OTMAYPRX) および OTMA ユーザー・データ・フォーマット設定出口ルーチン (DFSYDRUO) により実行されます。

記述子に EXIT=YES を設定することにより、定義された宛先記述子を出口ルーチンがオーバーライドすることを許可できます。それ以外の場合は、宛先記述子が使用され、出口ルーチンは呼び出されません。

コールアウト応答の OTMA ルーティングがどのように処理されるかは、ご使用のシステムによって異なります。コールアウト応答は、ユーザーか端末、または OTMA 非同期保留キューに逆経路指定されます。

## 関連概念

797 ページの『[OTMA 宛先記述子](#)』

OTMA 宛先記述子は、OTMA を介して経路指定されるメッセージの宛先を定義するために使用します。

## 関連資料

[OTMA 宛先解決ユーザー出口 \(DFSYPXO およびその他の OTMAYPRX タイプの出口\) \(出口ルーチン\)](#)

[OTMA ユーザー・データ・フォーマット設定出口ルーチン \(DFSYDRUO\) \(出口ルーチン\)](#)

## IMS Connect TCP/IP クライアントからの非同期コールアウト要求の開始

非同期コールアウト要求を IMS Connect TCP/IP クライアントから開始するには、TCP/IP クライアントが送信専用要求をサブミットする必要があります。この要求は、コールアウト要求を発行する IMS アプリケーション・プログラムを実行します。

## このタスクについて

コールアウト要求への応答を待ってこれを取り出すには、TCP/IP クライアントが RESUME TPIPE 呼び出しを発行する必要があります。

## IMS TM Resource Adapter と非同期コールアウト要求

非同期コールアウト機能を IMS TM Resource Adapter と一緒に使用している場合、WebSphere Application Server 管理者は、EJB Bean またはメッセージ駆動型 Bean が IMS Connect を介して IMS からコールアウト要求をリトリブするために使用する共用可能接続ファクトリーを構成する必要があります。

この Bean は、タイムアウト値が長い、ループする RESUME TPIPE 呼び出しを発行することにより、IMS TM Resource Adapter を使用して、OTMA 非同期保留キューを「listen」します。RESUME TPIPE 呼び出しが発行されたときにキューにメッセージがない場合、IMS TM Resource Adapter は待ち状態のままになります。コールアウト要求が非同期保留キューに到達する前に RESUME TPIPE 呼び出しがタイムアウトした場合、制御は Bean に戻されます。すると Bean は、別の RESUME TPIPE 呼び出しを発行して、listen し続けるか、ループを抜け出してエラー・チェックを実行します。

また、Java アプリケーション・プログラマーは、コールアウト要求をキューに入れる OTMA T パイプを識別する代替クライアント ID を指定するように Bean を変更する必要があります。

オプションで、Bean がコールアウト要求自体を処理したり、コールアウト要求を Web サービスに受け渡すようにしたりすることができます。

IMS Connect の複数のインスタンスを介して IMS に入力を配布する z/OS Sysplex Distributor とともに IMS TM Resource Adapter を使用している場合、そのコールアウト要求は、OTMA スーパーメンバー 関数が有効になっている TMEMBER を使用して行なわれる必要があります。

IMS TM Resource Adapter および EJB アプリケーションのコーディングに関する情報 (サンプル・アプリケーション・コードを含む) については、[IMS TM Resource Adapter の概要およびコールアウト・プログラミング・モデル \(TM Resource Adapter\)](#)を参照してください。

## SOAP Gateway と非同期コールアウト要求

IMS Enterprise Suite SOAP Gateway は、非同期コールアウト機能をサポートしています。この機能を使用すると、IMS アプリケーション・プログラムが SOAP Gateway を介して Web サービスに非同期コールアウト要求を行うことができます。

SOAP Gateway は、非同期コールアウト要求に対して有効にされると、IMS Connect に対して RESUME TPIPE 呼び出しを継続的に発行することによって、新しいコールアウト要求を listen します。コールアウト要求メッセージが T パイプ・キューにある場合、OTMA はコールアウト要求を IMS Connect に送信します。IMS Connect はこのメッセージ (必要に応じて XML に変換する) を処理して、このメッセージを SOAP Gateway に送信します。

IMS Connect の複数のインスタンスを介して IMS に入力を配布する z/OS Sysplex Distributor とともに SOAP Gateway を使用している場合、そのコールアウト要求は、OTMA スーパーメンバー 関数が有効になっている TMEMBER を使用して行なわれる必要があります。

コールアウト要求メッセージが XML への変換を必要とする場合、IMS Connect で XML 変換サポートを有効にする必要があります。コールアウト要求メッセージは、それ自体の OTMA ヘッダーの中に XML アダプターとコンバーターを指定する必要があります。コールアウト要求メッセージが XML アダプターとコンバーターを指定していない場合、IMS Connect はメッセージを XML に変換しません。

コールアウト要求メッセージには、OTMA 宛先記述子を使用して XML アダプターとコンバーターを指定することができます。以下に例を示します。

```
D DESCNAME TYPE=IMSCON TMEMBER=HWS1 TPIPE=PIPENAME
D DESCNAME ADAPTER=HWSXMLA0 CONVRTR=CNVNMED
```

この例では、IMS アプリケーション・プログラムが記述子名 DESCNAME を使用してメッセージを ALT IOPCB に挿入する場合、OTMA はアダプター名とコンバーター名をコールアウト要求メッセージの OTMA ヘッダーに追加して、メッセージを T パイプ PIPENAME のキューに入れます。

非同期コールアウト機能を使用した XML 変換の IMS Connect によるサポートでは、IMS アプリケーション・プログラムに呼び出しを行わせようとする対象の各 Web サービスに対するコールアウト XML コンバーターが必要です。コールアウト XML コンバーターは、Rational® Developer for System z バージョン 7.1.1 以降を使用して生成します。

SOAP Gateway について詳しくは、IBM 資料内にある SOAP Gateway の資料を参照してください。

#### 関連概念

[159 ページの『IMS Connect の XML 変換サポートの概要』](#)

一部の IMS Connect クライアントでは、IMS Connect は入力メッセージに収められた XML データを、COBOL または PL/I で作成された IMS アプリケーション・プログラムが使用するデータ構造に変換できます。これに対応する出力メッセージのデータも IMS アプリケーション・プログラムのプログラミング言語から IMS Connect クライアントが想定する XML データに変換されます。

[797 ページの『OTMA 宛先記述子』](#)

OTMA 宛先記述子は、OTMA を介して経路指定されるメッセージの宛先を定義するために使用します。

## IBM MQ と非同期コールアウト要求

IBM MQ は、ALT IOPCB インターフェースからの非同期コールアウト要求をサポートします。

OTMA は、IBM MQ アプリケーションの宛先記述子を使用して構成できます。IBM MQ によって予期される MQ メッセージ記述子 (MQMD) データ構造には、MQ メッセージ処理を制御するフィールドが含まれています。OTMA 記述子には、MQMD 構造を構成するためのパラメーターが含まれています。これは、IMS.PROCLIB データ・セットの DFSYDTx メンバー、またはタイプ 2 コマンドを使用して設定できます。

宛先に一致する記述子は、デフォルトのルーティングおよびメッセージ処理のパラメーターを設定します。ただし、OTMA ルーティング出口 (DFSPRXO および DFSYDRU0) を使用すると、宛先記述子のルーティング情報をオーバーライドすることができます (その記述子について EXIT=YES パラメーターが指定されて出口が使用可能になっている場合)。

#### 関連概念

[797 ページの『OTMA 宛先記述子』](#)

OTMA 宛先記述子は、OTMA を介して経路指定されるメッセージの宛先を定義するために使用します。

#### 関連資料

OTMA 宛先記述子の構文およびパラメーター (システム定義)

## IMS アプリケーション・プログラムおよび非同期コールアウト機能

IMS がコールアウト要求への応答を処理する必要があったり、端末がトランザクションのサブミット後に確認応答を必要としたり、応答モードのトランザクションがプログラム間通信を開始したりする場合には、



必要に応じて IMS アプリケーション・プログラムを修正または作成して、コールアウト機能をサポートします。

コールアウト要求への応答が不要な場合、アプリケーション・プログラムは、代替 PCB に対する挿入呼び出しの中に適切な OTMA 宛先名を指定するだけで済みます。アプリケーション・プログラムの修正は必要ありません。

ただし、コールアウト要求に対する応答を必要とする場合は、IMS アプリケーション・プログラムを修正して、この応答を元の入力トランザクションに相関させる必要があります。相関の方式は、最終的には開発者と、ご使用のシステムによって決まりますが、アプリケーション・プログラムの中に、発信しているコールアウト要求メッセージで相関のために使用するある種のデータを含めてから、その同じデータを応答メッセージの中で探すことが必要になる可能性はかなりあります。

また、コールアウト要求を発行する IMS アプリケーション・プログラムが、コールアウト応答が戻されたときに IMS データベースの中に相関データまたはその他のデータを (取り出すために) 格納することもできます。共用キュー環境では、このようなデータベースは、応答トランザクションが、別の IMS システムでスケジュールされる可能性がある場合にも便利です。応答を LTERM または IMS Connect TCP/IP クライアントに送信して戻さなければならない場合は、そのデータベースに LTERM 名または TMEMBER と TPIPE 名も格納する必要があります。

次のサンプル・コードは、コールアウト機能とコールアウト応答に使用できる COBOL コピーブックのデータ構造体の例を示しています。

コールアウト要求データ構造体には、相関データを EJB に送信するための相関関係子フィールド CORRID が含まれます。

```
01 CALLREQ
   02 LL          PIC S9(4) COMP.
   02 ZZ          PIC S9(4) COMP.
   02 CORRID     PIC (8).
   02 ACCTNUM    PIC (20).
```

応答が戻されると、COBOL コピーブック・データ構造体の CORRID フィールドには同じ相関データが含まれます。IMS アプリケーションはこれを使用して、必要に応じて最初の要求への応答を関連付けることができます。

```
01 CALLRESP
   02 LL          PIC S9(4) COMP.
   02 ZZ          PIC S9(4) COMP.
   02 CORRID     PIC (8).
   02 ACCTBAL    PIC (20).
```

## 非同期コールアウト要求でアプリケーション・プログラムおよび端末の停止を回避する

一般に、端末にはタイムアウト・メカニズムはインプリメントされておらず、応答が端末に戻されない場合、端末は停止します。

コールアウト要求は、少なくとも次のような 2 つの場合に、端末が中断状態になることがあります。コールアウト要求を発行するアプリケーション・プログラムが、端末に回答せずに終了した場合。または、応答モードのトランザクションが、コールアウト要求または入力側端末への応答を適切に相関させずにプログラム間通信を実行した場合。

前者のような場合は、例えば端末が、コールアウト要求を開始するトランザクションの送信後に応答を待って待機し、そのトランザクションを処理する IMS アプリケーション・プログラムがコールアウト要求を ALTPCB に挿入した後で終了した場合、その端末は応答を待って停止します。これを回避するには、コールアウト要求を発行する IMS アプリケーション・プログラムを、入力側端末に対して確認応答を送信するように修正する必要があります。

後者の場合では、端末は、以下のような状況で停止することがあります。アプリケーション・プログラム PGMA が、アプリケーション・プログラム PGMB へのプログラム間通信を開始した後で応答を待って待機する応答モードのトランザクションを処理し、次にアプリケーション・プログラム PGMB が、IMS への応答を生成するコールアウト要求を生成する。次に、3 番目のアプリケーション・プログラム PGMC がこの応答を処理するが PGMA に通知しない場合、PGMA は応答を待って停止します。これを回避するには、

PGMB と PGMC を経由した応答の相関をサポートするように PGMA を修正し、PGMC がこの応答を PGMA に受け渡し、最初のクライアント 端末に戻るようする必要があります。コールアウト要求によって応答が生成されない場合、応答モードのトランザクションを PGMA などのアプリケーション・プログラムとともに使用しないでください。

## 非同期コールアウト要求への応答の相関

コールアウト要求の結果、外部アプリケーションまたは Web サービスが応答メッセージを IMS に戻した場合、その要求と応答を処理する IMS アプリケーション・プログラムは、応答メッセージを適正に処理するために、その応答メッセージをコールアウト要求に相関させる必要があります。

### このタスクについて

応答メッセージは、コールアウト要求を発信した端末または TCP/IP クライアント・アプリケーションにも相関させる必要があります。

## 非同期コールアウト要求への応答の相関

応答メッセージを最初の非同期コールアウト要求に相関させるのは、IMS アプリケーション・プログラムが行います。IMS アプリケーションは、コールアウト要求にデータ・エレメント (メッセージ ID や固有のコールアウト要求 ID など) を定義できます。これを使用して、応答を最初の入力メッセージに相関させることができます。

### このタスクについて

応答が、その最初のコールアウト要求に相関した後は、その相当を元の 入力側端末または IMS Connect TCP/IP クライアントに相関させることができます。

## 入力端末への非同期コールアウト要求応答の経路指定

応答メッセージを端末に経路指定するために、非同期コールアウト要求を発行する IMS アプリケーションは、その端末の LTERM 名を収集できます。

### このタスクについて

応答メッセージを受け取った後、その応答を処理する IMS アプリケーションは、LTERM 名によって発信元の端末を識別することができます。LTERM 名は、コールアウト要求と応答メッセージのデータの一部として含まれることも、応答を処理する IMS アプリケーションがアクセス可能な一時データベースに保管することもできます。

一時データベースを使用している場合、応答をコールアウト要求に相関させるために使用されたデータは、データベース内の LTERM 名とともに含める必要があります。応答が戻された後、IMS アプリケーションは、開始側端末にこの応答を戻すことができます。これは、最初のコールアウト要求を発行した IMS アプリケーション・プログラムが収集した LTERM 名に対する応答を、CHNG 呼び出しおよび ISRT することにより行います。

## IMS Connect TCP/IP クライアントへの非同期コールアウト応答の経路指定

応答を TCP/IP クライアントに経路指定するには、非同期コールアウト要求メッセージを発行する IMS アプリケーション・プログラムが IMS Connect TMEMBER および OTMA TPIPE 名 (開始側の TCP/IP クライアントが使用するもの) を収集する必要があります。

### このタスクについて

そのコールアウト要求への応答を処理する IMS アプリケーション・プログラムは、その応答を受信すると、ただちに TMEMBER と TPIPE 名を使用して、応答を正しい TCP/IP クライアントに戻します。応答を正しい TMEMBER に問題なく経路指定するには、OTMA 宛先記述子か、OTMAYPRX ユーザー出口と DFSYDRUO 出口ルーチンを使用する必要があります。

この応答を処理する IMS アプリケーション・プログラムが TMEMBER および TPIPE 名を使用できるようにするために、コールアウト要求および応答メッセージのデータの一部としてこれらの名前を含めることも、

応答を処理する IMS アプリケーション・プログラムからアクセス可能な一時データベースにこれらの名前を保管することもできます。

一時データベースを使用する場合、その応答を処理する IMS アプリケーション・プログラムは、応答トランザクションが戻されたときに、さらに相関を使用して、正しい TMEMBER および TPIPE 名を取得する必要があります。相関が完成すると、IMS アプリケーション・プログラムは TMEMBER および TPIPE 名を使用して CHNG 呼び出しと ISRT を行い、応答を元の IMS Connect TCP/IP クライアントに送信して戻します。

## RESUME TPIPE 要求のコールアウトおよび OTMA 並列処理

RESUME TPIPE 要求の OTMA 並列処理を使用可能にすることにより、発信コールアウト要求メッセージのスループットを向上させることができます。

デフォルトでは、OTMA T パイプは一度に 1 つのアクティブ RESUME TPIPE 要求のみをサポートします。さらに多くの RESUME TPIPE 要求を受信すると、OTMA は T パイプへのキューに入れ、それらの要求はアクティブの RESUME TPIPE 要求が終了するまでアクティブのままです。コールアウト・メッセージの発信量が多い場合、OTMA T パイプがボトルネックになる可能性があります。

OTMA T パイプが複数のアクティブ RESUME TPIPE 要求をサポートし、複数のクライアントが T パイプに RESUME TPIPE 要求を発行する場合、OTMA はコールアウト・メッセージを複数のクライアントに並列で送信できます。

RESUME TPIPE 要求の並列処理に対するサポートは、OTMA クライアント記述子に MULTIRTP=Y を指定することで使用可能にできます。DFSOTMA システム・クライアント記述子を使用することにより、MULTIRTP=Y をすべての OTMA クライアントに対するデフォルトとして定義できます。

アクティブ RESUME TPIPE 要求の最大数は、OTMA クライアント記述子に LIMITRTP パラメーターを指定することで設定できます。

RESUME TPIPE 要求の並列処理に対するサポートは、IMS Connect でも使用可能に設定できます。この場合は、OTMA へのデータ・ストア接続の定義で MULTIRTP=Y を指定します。また、IMS Connect インスタンスのシステム構成で、IMS Connect から MULTIRTP=Y をすべての接続に対するデフォルトとして指定することもできます。

### 関連概念

792 ページの『[複数のアクティブ RESUME TPIPE 要求の並列処理に対応する OTMA T パイプのサポート](#)』  
OTMA クライアント記述子で MULTIRTP=Y を指定すると、OTMA クライアントに関連付けられた OTMA T パイプは複数のアクティブな RESUME TPIPE 要求を並列にサポートできます (MULTIRTP の指定がクライアントによってオーバーライドされない限り)。

335 ページの『[並列 RESUME TPIPE 要求による出力のリトリブ](#)』

OTMA 出力メッセージ (特に、コールアウト要求メッセージ) のスループットを向上させるには、HWS 構成ステートメントまたは DATASTORE IMS Connect 構成ステートメントで MULTIRTP=Y を指定して、OTMA T パイプが複数のアクティブ RESUME TPIPE 要求を並列でサポートすることを使用可能に設定できます。





## 第 46 章 OTMA メッセージ接頭語

OTMA メッセージには、IMS.ADFSMAC データ・セットの DFSYMSG マクロによってマップされるフォーマットに準拠した接頭語があります。

メッセージ接頭語の最大長は、4096 バイトで、この長さにはアプリケーション・データは含みません。

OTMA メッセージ接頭語には、以下のセクションが含まれます。

1. メッセージ制御情報セクション
2. 状態データ・セクション
3. セキュリティー・データ・セクション
4. ユーザー・データ・セクション
5. アプリケーション・データ・セクション

以下の表は、OTMA メッセージ接頭語のセクションを表示し、各セクション内の一部のキー・フィールドをリストしています。この表では、用語"サーバー"は IMS を参照します。

表 154. OTMA メッセージ接頭語セグメントおよびそれらのキー・フィールド

メッセージ制御情報	状態データ	セキュリティー・データ	ユーザー・データ	アプリケーション・データ
T パイプ名	宛先オーバーライド	ユーザー ID		
メッセージ・タイプ	マップ名	UTOKEN		
シーケンス番号	同期フラグ	セキュリティー・フラグ		
プロセス・フラグ	Sync_level			
応答標識	コミット・モード・トークン			
チェーン標識	サーバー状態			

### メッセージ制御情報セクション

各 OTMA メッセージごとに、OTMA メッセージ接頭語の最初のセクションにメッセージ制御情報を指定する必要があります。

メッセージ制御情報セクションは、DFSYMSG マクロの TMAMCTL DSECT によってマップされます。

以下の表は、メッセージ接頭語 (889 ページの『第 46 章 OTMA メッセージ接頭語』の表の 1 列目) のメッセージ制御情報セクションの内容を要約しています。この要約には、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 155. OTMA メッセージ接頭語のメッセージ制御情報セクションのフォーマット

バイト	長さ	内容	値	意味
0	1	アーキテクチャー・レベル	X'01'	OTMA リリース・レベル。すべてのメッセージに必須です。

表 155. OTMA メッセージ接頭語のメッセージ制御情報セクションのフォーマット (続き)

バイト	長さ	内容	値	意味
1	1	メッセージ・タイプ: このフィールドの指定は、すべてのメッセージについて必須です。	X'80'	データ: サーバー出力データ (IMS アプリケーション・プログラムからの出力)。このデータは、トランザクションではありません。
			X'40'	トランザクション: サーバーへのトランザクションまたは IMS コマンドの入力。実際のトランザクション名は、接頭部のアプリケーション・データ・セクションで指定します。
			X'20'	応答: 応答メッセージ。
			X'10'	プロトコル・コマンド: OTMA プロトコル・コマンド (IMS コマンドではない)。下記のバイト・オフセット 4 の『プロトコル・コマンド・タイプ』を参照してください。
			X'08'	コミット確認: コミット完了。同期点完了を通知するためにサーバーが使用します。送信後コミット・トランザクションにのみ使用されます。下にある"コミット確認フラグ"を参照してください。
			X'04'	同期プログラム間通信: 同期プログラム間通信要求。
2	1	応答フラグ: この応答フラグ・フィールド内の値は、相互に排他的です。	X'80'	ACK: 肯定応答。
			X'40'	NAK: 否定応答。NAK は、追加のセンス・コードを伴うことがあります。
			X'20'	応答要求: 応答がこのメッセージに要求された。
			X'10'	拡張応答要求: 設計済みトランザクションあるいはコマンドの属性をクライアントに戻すことを要求する。
			X'08'	同期コールアウト要求に対する応答: このメッセージは、IMS アプリケーション・プログラムによって発行された同期コールアウト・メッセージに対する応答です。
			X'04'	GU 呼び出しの前に有効期限が切れた入力トランザクション: サーバーによって設定されるこのフラグは、IMS アプリケーションが IMS 入力キュー上の入力トランザクションをリトリブする GU 呼び出しを発行する前に、その入力トランザクションがタイムアウトになったことを示します。
			X'02'	遅延 ACK または NAK 応答のサポート: OTMA クライアントはこのフラグを設定して、OTMA がクライアントから遅延または無効 ACK/NAK を受け取った場合に NAK をクライアントに送信するように OTMA に指示します。
			X'01'	トランザクションの有効期限切れ時に入力メッセージを返す: トランザクションが IMS によって処理される前にタイムアウトになった場合、OTMA はメッセージ DFS3588I の代わりに元の入力メッセージをクライアントに返すことを要求するために、クライアントによって設定されます。
3	1	コミット確認フラグ	X'80'	コミット済み: サーバーは正常にコミットしました。
			X'40'	打ち切り: サーバーはコミットを打ち切った。
			X'08'	OTMA タイムアウト条件のために異常終了: タイムアウトの限度に到達するまでに、OTMA クライアントからの ACK または NAK 応答が受信されなかったことを示します。
4	1	プロトコル・コマンド・タイプ	X'04'	クライアント・ビッド: クライアントがサーバーに送信する。応答要求フラグおよび適切な状態データ・フィールド (例えば、メンバー名) を同様にセットする必要があります。
			X'08'	サーバー使用可能: サーバーがクライアントに送信する。適切な状態データ・フィールド (例えば、メンバー名) も同様にセットする必要があります。
			X'0C'	CBresynch: クライアントがサーバーに再同期を要求するために送信する。これに続く再同期のクライアント・ビッド要求はオプションで、その要求によって、サーバーがクライアントに SRVresynch コマンドを送信します。
			X'14'	すべての T パイプに対する延期処理: サーバーは、クライアントのメッセージ・アクティビティのすべてを延期させるためにこのコマンドを送信する。
			X'18'	すべての T パイプに対する再開処理: サーバーは、クライアントのメッセージ処理を再開させるためにこのコマンドを送信する。
			X'1C'	T パイプに対する入力延期: サーバーは過負荷になるときこのコマンドを送信する。
			X'20'	T パイプに対する入力再開: サーバーはクライアント入力が準備できたとき、このコマンドを送信する (T パイプに対する入力延期コマンドに続いて)。
			X'24'	T パイプに対する出力の再開: クライアントが、キューされた T パイプ出力を再送するためにサーバーに送信する。
			X'26'	すべての T パイプからの出力再開: OTMA がクライアントに関連付けられたすべての T パイプからの出力メッセージの送信を再開するように要求するために、クライアントによって送信されます。T パイプの保留キューには適用されません。
			X'28'	T パイプ用の保留キューからの出力の再開: クライアントからサーバーへ、T パイプ用の保留キューに入っているメッセージを要求するために送信されます。
			X'29'	RESUME TPIPE 要求の取り消し: クライアントが、保留中の RESUME TPIPE 要求を取り消すために、サーバーに送信します。RESUME TPIPE 要求は、OTMA 状態データのバイト 4 に RESUME TPIPE トークンを指定することによって識別する必要があります。
			X'2A'	T パイプの保留キュー上にメッセージがない: サーバーが、RESUME TPIPE 要求を受け取り、T パイプの保留キューが空になったときに、クライアントに送信します。
			X'2C'	SRVresynch: サーバーが、CBresynch を送信したクライアントに送信する。このコマンドは、サーバー内のすべての同期化 T パイプを識別する。
			X'30'	REQresynch: サーバーが、同期化 T パイプの状態を指定するためにクライアントに送信する。
			X'34'	REPresynch: クライアントが、サーバーが行う再同期のタイプを指示するためにサーバーに送信する。
X'38'	TBresynch: クライアントが、特定 T パイプに対する再同期を開始するためにサーバーに送信する。			
X'3C'	サーバー状態: サーバー処理の現行状態を使用してサーバーからクライアントに送信される。クライアントは、サーバーの処理速度が低下した場合に、この情報を使用してトランザクションを異なるサーバーにリダイレクトします。			

表 156. OTMA メッセージ接頭語のメッセージ制御情報セクションのフォーマット

バイト	長さ	内容	値	意味
5	1	プロセス・フラグ	X'80'	RESUME TPIPE トークン: RESUME TPIPE 要求の場合のみ、RESUME TPIPE 要求に、その要求を一意的に識別するトークンが含まれていることを示します。OTMAはこのトークンを使用して、複数の RESUME TPIPE 要求を順番にキューに入れて処理します。 RESUME TPIPE トークンは、クライアントが RESUME TPIPE 要求を取り消す場合にも、クライアントによって指定されます。 出力メッセージの場合、このフラグは、すべての T パイプ (TMAMCSPA) における中断状態の処理は IMS シャットダウンが原因であることを示します。これは単なる出力フラグです。
			X'40'	同期化された T パイプ: 入出力シーケンス番号がトランザクション・パイプに対して維持される。
			X'20'	非同期出力: サーバーが、非送信請求のキューされたデータ・メッセージを送信中。
			X'10'	エラー・データ・メッセージが続く: エラー・データ・メッセージが続く。サーバーが NAK を送信するとき送信します。
			X'08'	保留キュー内のメッセージ: T パイプ用の保留キューに入っている 1 つ以上のメッセージ。共有キューでは使用不可。
			X'02'	追加の info set. RESUME TPIPE 要求の場合、RESUME TPIPE ID 情報の OTMA メッセージ接頭語のメッセージ制御情報のバイト 24 を参照してください。コールアウト・メッセージの場合には、RESUME TPIPE ID 情報の PTMA メッセージ接頭語の状態データ・セクション接頭部のトランザクションおよびコールアウト・メッセージ内のバイト 22 を参照してください。
			X'01'	エラー・メッセージの送信: CM1 応答としてエラー・メッセージが送信された。CM1 トランザクションが、唯一の応答としてエラー・メッセージを生成しました。

表 157. OTMA メッセージ接頭語のメッセージ制御情報セクションのフォーマット

バイト	長さ	内容	値	意味
6	8	T パイプ名		OTMA 識別とプロセス制御トークン。この名前は、IMS アプリケーション・プログラムに対して I/O PCB 上の LTERM 名をオーバーライドするために使用されます。
14	1	チェーン・フラグ: このフラグは、複数セグメント・メッセージに必須です。	X'80'	先頭チェーン: 複数セグメントの最初のセグメント。1 セグメントだけのメッセージは、チェーンの先頭フラグとチェーンの最後フラグの設定で指示されます。
			X'40'	中間チェーン: 複数セグメントの部分。
			X'20'	チェーン内での最後: 複数セグメントの最後のセグメント。
			X'10'	チェーンの廃棄: メッセージ・セグメントの現行チェーンを廃棄する。
15	1	接頭部フラグ: このフィールドの値は、メッセージ接頭語のどのセクションがこのメッセージに添付されているかを示します。	X'80'	状態データ: 状態データ・セクションはメッセージに組み込まれている。状態データ・セクションはメッセージごとに必須です。
			X'40'	セキュリティー: セキュリティー・セクションはメッセージに組み込まれている。
			X'20'	ユーザー・データ: ユーザー・データ・セクションはメッセージに組み込まれている。このデータは OTMA クライアントが指定します。
			X'10'	アプリケーション・データ: アプリケーション・データ・セクションはメッセージに組み込まれている。
16	4	送信シーケンス番号		トランザクション・パイプに対するシーケンス番号。トランザクション・パイプごとの各送信で増えます。
20	4	センス・コード		バイト・オフセット 20 に編成 (TMAMCSNS)
20	2	センス・コード		NAK メッセージを伴う。
22	2	理由コード		NAK メッセージを伴う。 バイト・オフセット 20 に編成 (TMAMCSNS)
20	4	エージング値		アクセサ環境エレメント (ACEE) のエージング値を秒単位で指定します。IMS は、現行 ACEE の経過時間がこの値を超えた場合、ACEE を作成します。 エージング値は、どれくらいの頻度でユーザー ID ACEE が更新されるべきかを指定します。また、エージング値フラグ X'40' を状態データのバイト 5 に設定する必要があります。この値は、ACK または NAK メッセージには適用されません。 キャッシング・サポートの最小値は 300 秒 (5 分) です。指定したエージング値が最小値を下回る場合、IMS は常に、キャッシュされない ACEE を作成します。
24	4	リカバリー可能シーケンス番号		トランザクション・パイプに対するリカバリー可能シーケンス番号。同期化トランザクション・パイプを使用しているリカバリー可能メッセージの各送信で増えます。再同期にのみ必要です。
28	2	セグメント・シーケンス番号		複数セグメント OTMA メッセージのセグメントのシーケンス番号。
30	1	CM1 ACK タイムアウト	1 から 255	このメッセージのメッセージ・レベルのタイムアウト値を秒単位で指定します。
31	1	予約済み		synclevel=confirm または synclevel=syncpt のときに、送信後コミット・メッセージにのみ適用されます。 このトランザクションについてのみ、メッセージ・レベルのタイムアウト値を設定します。値は、OTMA で設定されたタイムアウト値を超えることはできず、超えた場合は無視されます。 タイムアウト値は、OTMA で /START TMEMBER コマンドによって、または OTMA クライアント記述子によって設定されます。
		制御データ・セグメントの数	1 から 255	状態データに TMAMHCTD が設定されている場合、制御データ・セグメントの数を指定します。このフィールドは、IMS が制御データを伴う ICAL コールアウト・メッセージを送信する場合にのみ使用されます。

表 157. OTMA メッセージ接頭語のメッセージ制御情報セクションのフォーマット (続き)

バイト	長さ	内容	値	意味
バイト・オフセット 24 に編成 (TMAMCRSQ)				
24	8	RESUME TPIPE リクエスト		ター ID

## 関連資料

892 ページの『OTMA メッセージ制御情報フィールドの説明』

OTMA メッセージ接頭語のメッセージ制御情報部分にあるフィールドを使用すると、タイプ、内容、処理要件などの OTMA メッセージの特性を指定できます。

## OTMA メッセージ制御情報フィールドの説明

OTMA メッセージ接頭語のメッセージ制御情報部分にあるフィールドを使用すると、タイプ、内容、処理要件などの OTMA メッセージの特性を指定できます。

このトピックでは、メッセージ接頭語のメッセージ制御情報セクションのフィールドについて説明します。

### アーキテクチャー・レベル (TMAMCALV)

OTMA アーキテクチャー・レベルを指定します。クライアントは、アーキテクチャー・レベルを指定し、サーバーは、どのアーキテクチャー・レベルを使用しているかを、応答メッセージに指示します。クライアントとサーバーが使用するアーキテクチャー・レベルは、一致している必要があります。

すべてのメッセージに必須です。

### メッセージ・タイプ (TMAMCMGT)

メッセージ・タイプを指定します。各 OTMA メッセージは、メッセージ・タイプの値を指定する必要があります。値は相互に排他的ではありません。例えば、サーバーが、ACK メッセージをクライアントがサブミットしたトランザクションに送信する場合、トランザクション・フラグと応答フラグの両方がセットされます。

### データ (TMAMCDTA - X'80')

クライアントに送信されたサーバー出力データを指定します。クライアントが、メッセージ接頭語の状態データ・セクションで同期レベルに「確認」を指定すると、サーバーも応答フラグに対して「要求される応答」をセットします。クライアントが同期レベルを指定していない場合は、サーバーは、デフォルトの「確認」を使用します。

### トランザクション (TMAMCTXN - X'40')

サーバーへのクライアント入力データを指定します。

サーバーが、ACK メッセージあるいは、NAK メッセージのどちらで応答するかは、「応答要求」が応答フラグに対してもセットされているかによります。

### 応答 (TMAMCRSP - X'20')

メッセージ・タイプを応答メッセージとして指定してください。応答は、メッセージ応答フラグが、「応答要求」を指定するときセットされます。

このフラグがセットされている場合は、応答フラグは ACK あるいは NAK を指定します。

送信シーケンス番号は、元のデータ・メッセージおよび応答メッセージに対してマッチすることが必要です。サーバーにチェーニングされたトランザクション入力メッセージは、(特定のトランザクション・パイプに対する) 次のトランザクションが送信される前に、常に応答を要求する必要があります。

### コマンド (TMAMCCMD - X'10')

OTMA プロトコル・コマンドを指定します。OTMA コマンドは、常に応答フラグに対する応答要求を指定する必要があります。

### コミット確認 (TMAMCCMT - X'08')

コミット完了を指定します。これは、同期点を完了したときサーバーが送信し、送信後コミット・トランザクションにのみ適用されます。このフラグも、IMS 会話型トランザクションを終了することを OTMA サーバーに知らせるために OTMA クライアントによって設定することができます。

### 応答フラグ (TMAMCRSI)

メッセージが応答メッセージであるか、または応答が要求されているかを指定します。

トランザクションへの確認通知は、トランザクションが拡張応答要求を指定している場合のみ、メッセージ接頭語のアプリケーション・データ・セクションに、そのトランザクションの属性を含んでいます。

### ACK (TMAMCAACK - X'80')

肯定応答を指定します。

### NAK (TMAMCNAK - X'40')

否定応答を指定します。

NAK の理由の詳細については、センス・コード・フィールドを参照してください。

### 応答要求 (TMAMCRRQ - X'20')

このメッセージに応答が要求されていることを指定します。これには、メッセージ・タイプとして、データ、トランザクション、コマンドがセットできます。

送信後コミット IMS コマンド出力を送信しているとき、IMS は、同期レベルが何であれ、ACK を要求しません。

### 拡張応答要求 (TMAMCERQ - X'10')

拡張応答がこのメッセージに要求されていることを指定します。クライアントは、トランザクション(あるいは、トランザクション・コードの代わりに IMS コマンドを指定したトランザクション)にのみ、セットできます。

このフラグがトランザクションに対してセットされる場合は、IMS は、ACK メッセージのアプリケーション・データ・セクションにそのトランザクションの設計済み属性を戻します。

このフラグがコマンドに対してセットされる場合は、IMS は、ACK メッセージのアプリケーション・データ・セクションに設計済み属性を戻します。このフラグは、IMS コマンド **/DISPLAY TRANSACTION** および **/DISPLAY TRANSACTION ALL** にセットできます。

### 同期コールアウト要求に対する応答 (TMAMSYRP - X'08')

このメッセージが、IMS 従属領域内で実行されている IMS アプリケーション・プログラムによって発行された同期コールアウト・メッセージに対する応答であることを指定します。

同期コールアウト応答のフラグが設定されている場合、ほとんどの保留キュー対応のクライアント (IMS Connect など) では、メッセージ接頭語の状態データ・セクションのクライアント・フラグ・フィールド (TMAMHCFL) に送信専用メッセージ・フラグ (TMAMHSOM - X'80') も設定されます。

応答フラグ・フィールドに、応答要求フラグ (TMAMCRRQ - X'20') と TMAMSYRP フラグの両方が設定されている場合、OTMA は、その応答が IMS ICAL アプリケーションに正常に送達されたかどうかを示す標識 (ACK または NAK) をクライアントに送信します。

### 遅延 ACK または NAK 応答のサポート (TMAMDACK - X'02')

クライアントは、OTMA がクライアントから遅延したまたは無効な確認応答を受け取った場合にクライアントに NAK 応答を返すよう要求することを示します。例えば、ACK タイムアウト間隔が有効期限切れになった後で、あるいは T パイプの WAIT 状況をクリアするために **/STO TMEBER TPIPE** コマンドが発行された後で、OTMA が確認応答を受け取った場合、OTMA はクライアントに NAK を返します。OTMA がクライアントに返す NAK には、X'2B' センス・コードが組み込まれます。

### コミット確認フラグ (TMAMCCCI)

コミット要求が成功したことを指定します。サーバーがクライアントにコミット確認メッセージを送信します。これらのメッセージは、送信後コミット・トランザクションに対してのみ適用され、メッセージ接頭語の状態データ・セクションにある同期レベル・フラグには影響されません。

### コミット済み (TMAMCCTD - X'80')

サーバーが正常にコミットしたことを指定します。

### 打ち切り (TMAMCABT -X'40')

サーバーがコミットを打ち切ったことを指定します。

### コミット済み (TMAMCRTC - X'20')

サーバーが RRS を介してクライアントからコミット通知を受信した後、サーバーが IOPCB 内の出力をコミットする準備ができていることを示します。

### OTMA タイムアウト条件のために異常終了 (TMAMCTMO -X'08')

タイムアウトの限度に達するまでに、OTMA クライアントから ACK または NAK の応答を受信しなかったことを示します。

### コマンド・タイプ (TMAMCTYP)

OTMA プロトコルのコマンド・タイプを指定します。

IMS MTO コマンドを、メッセージのアプリケーション・データ・セクションに指定します。

### クライアント・ビッド (TMAMCBID - X'04')

クライアントが OTMA サーバーに送信する最初のメッセージを指定します。このコマンドには、メッセージ接頭語のメッセージ制御情報セクションに、応答要求フラグおよびセキュリティ・フラグもセットする必要があります。適切な状態データ・フィールド (例えば、メンバー名など) も同様にセットする必要があります。

セキュリティ・データ接頭部では、U トークン・フィールドを指定する必要があり、これにより、OTMA サーバーは、OTMA クライアントとして行動するためのクライアント権限の妥当性検査をすることができます。

サーバーは、クライアント・ビッド要求に応答できるため、このメッセージは、クライアントがデータ・メッセージを受け入れて、開始可能になるまで、送信してはいけません。

### サーバー使用可能 (TMAMCAVL - X'08')

サーバーがクライアントに送信する最初のメッセージを指定します。これは、クライアントが接続される前、サーバーが z/OS システム間カップリング・ファシリティ (XCF) グループに接続したときに送信されます。クライアントは、クライアント・ビッド要求を持つ「サーバー使用可能」メッセージをサーバーに応答します。適切な状態データ・フィールド (例えば、メンバー名) も同様にセットする必要があります。

クライアントが最初に接続する場合は、いつサーバーが接続するのが XCF によって知らされ、クライアント・ビッド要求の処理を開始します。

### CBresynch (TMAMCRSN - X'0C')

再同期に対するクライアント要求とともにクライアント・ビッド・メッセージを指定します。このコマンドはオプションで、サーバーにより、SRVresynch メッセージがクライアントに送信されます。CBresynch コマンドは、クライアントが IMS と再同期しようとして、既存の同期化 T パイプがクライアントに対して存在しているとき、クライアントが OTMA サーバーに送信する最初のメッセージです。メッセージ接頭語の CBresynch メッセージ標識以外は、メッセージ接頭語に必要な情報は、クライアント・ビッド・コマンドと同じにする必要があります。

IMS がクライアントからクライアント・ビッド要求を受信し、IMS が既存の同期化 T パイプを認識している場合、IMS は、通知メッセージ DFS2394I を MTO に発行します。IMS は、すべての同期化 T パイプに対して、リカバリー可能な送信あるいは受信のシーケンス番号を 0 (ゼロ) にリセットします。

### すべての T パイプに対する処理延期 (TMAMCSPA - X'14')

サーバーが、クライアントについてのすべてのメッセージ・アクティビティを延期していることを指定します。後続のデータ入力のすべては、NAK メッセージをサーバーから受信します。同様に、クライアントは、後続のサーバー・メッセージに対して、NAK メッセージを送信する必要があります。

クライアントは、/STOP TPIPE コマンドを OTMA メッセージとしてサブミットすることにより、特定のトランザクション・パイプの処理を延期することができます。

### すべての T パイプに対する処理再開 (TMAMCRSA - X'18')

サーバーがクライアントについてのメッセージ・アクティビティを再開していることを指定します。



クライアントは、**/START TPIPE** コマンドを OTMA メッセージとしてサブミットすることにより、停止されていた特定のトランザクション・パイプの処理を再開することができます。

#### **T パイプに対する入力延期 (TMAMCSPN - X'1C')**

サーバーが過負荷で、トランザクション・パイプに対する入力を一時的に延期していることを指定します。後続のすべてのクライアント入力は、メッセージ接頭語のメッセージ制御情報セクションに指定されているトランザクション・パイプに対する NAK メッセージを受信します。このコマンドに対する応答は要求されていません。

このコマンドは、マスター端末オペレーターが、**/STOP TPIPE** コマンドを入力したときも、IMS により送信されます。

#### **T パイプに対する入力再開 (TMAMCRSM - X'20')**

サーバーが、以前の T パイプ・コマンドに対する入力延期に続いて、クライアント入力を再開する準備ができていることを指定します。このコマンドに対する応答は要求されていません。

このコマンドは、IMS マスター端末オペレーターが、**/START TPIPE** コマンドを出すときにも IMS に送信されます。

#### **T パイプからの出力再開 (TMAMCRTP - X'24')**

OTMA サーバーに 1 つ以上の T パイプ名を指定します。T パイプ上のキューされた出力のすべては、もう一度再送されます。

#### **すべての T パイプからの出力再開 (TMAMCRAT - X'26')**

OTMA がクライアントに関連付けられたすべての T パイプからの出力メッセージの送信を再開するよう要求するために、クライアントによって送信されます。T パイプの保留キューには適用されません。

#### **T パイプ用の保留キューからの出力再開 (TMAMCRHQ - X'28')**

クライアントが、T パイプに対する保留キューからのメッセージの検索を要求していることを指定します。メッセージをリトリブするためのコマンド・オプションがあります。

#### **RESUME TPIPE 要求の取り消し (TMAMCDRH - X'29')**

クライアントが、保留中の RESUME TPIPE 要求を取り消すためにサーバーに送信します。RESUME TPIPE 要求は、OTMA 状態データのバイト 4 に RESUME TPIPE トークンを指定することによって識別する必要があります。

#### **T パイプの保留キュー上にメッセージがない (TMAMCMMSG - X'2A')**

サーバーが、RESUME TPIPE 要求を受け取り、T パイプの保留キューが空になったときに、クライアントに送信します。

#### **SRVresynch (TMAMCSRS - X'2C')**

クライアントの CBresynch コマンドへのサーバーの応答を指定します。このコマンドは、そのサーバー内の同期化トランザクション・パイプの状態を指定します (送信と受信のシーケンス番号)。

このコマンドは、1 つまたは複数のセグメントを持つ単一メッセージとして送信され、ACK が要求されます。

#### **REQresynch (TMAMCRQS - X'30')**

特定の T パイプに対する送信シーケンス番号および受信シーケンス番号を指定します。REQresynch は IMS からクライアントに送信されます。

#### **REPresynch (TMAMCRPS - X'34')**

T パイプに対するクライアントの本来あるべき状態情報を指定します。クライアントは、IMS から受信した REQresynch コマンドに応答して、REPresynch コマンドを IMS に送信します。

#### **TBresynch (TMAMCTBR - X'38')**

クライアントが、IMS から REQresynch コマンドを受信する準備ができていることを指定します。

#### **リソースの状態 (TMAMMNTN - X'3C')**

クライアントに OTMA リソースの状態を通知するために、サーバーが設定します。クライアントは、サーバーの処理速度が低下した場合に、この情報を使用してトランザクションを異なるサーバーにリダイレクトします。

#### **処理フラグ (TMAMCPFG)**

クライアントまたはサーバーが、メッセージ処理を制御できるオプションを指定します。

### RESUME TPIPE トークン (TMAMQ RTP - X'80')

RESUME TPIPE 要求の場合のみ、RESUME TPIPE 要求に、その要求を一意的に識別するトークンが含まれていることを示します。OTMAはこのトークンを使用して、複数のRESUME TPIPE 要求を順番にキューに入れて処理します。

RESUME TPIPE トークンは、クライアントがRESUME TPIPE 要求を取り消す場合にも、クライアントによって指定されます。

### 同期化 T パイプ (TMAMCSYP - X'40')

トランザクション・パイプが同期化されることを指定します。失敗した場合、トランザクション・パイプを再同期することをクライアントに許可します。コミット後送信トランザクションにのみ有効です。

このフラグにより、入出力シーケンス番号がトランザクション・パイプに対して維持されます。トランザクション・パイプを介して経路指定されるトランザクションはすべて、このフラグを首尾一貫して(オン/オフのいずれかで)指定する必要があります。

### 非同期出力 (TMAMCASY - X'20')

サーバーが、非送信請求のキューされた出力を送信中であることを指定します。これは、IMS が代替 PCB にメッセージを挿入するとき起こります。

IMS コマンドによっては、コミット後送信としてサブミットされる時、IMS にこのフラグ・セットを持つクライアントに出力を送信させます。この場合、OTMA 接頭部には、クライアントが出力を発信元のコマンド・メッセージと関連させるのに使用することができる、識別情報を含んでいません。これらのコマンドの出力データ・メッセージは、単純にトランザクション・パイプ名を識別します。IMS は、トランザクション・パイプ名だけで、何らかの非送信請求エラー・メッセージを送信することもできます。

### エラー・メッセージが続く (TMAMCERR - X'10')

エラー・メッセージがこのメッセージに続くことを指定します。このフラグは、サーバーからの NAK メッセージに対してセットされます。その後、追加のエラー・メッセージがクライアントに送信されます。

非同期出力フラグは、出力がIMS アプリケーションによって生成されていないので、エラー・データ・メッセージ中にはセットされません。

### 保留キュー中のメッセージ (TMAMCQUE - X'08')

1つ以上のメッセージが、送達される T パイプに対する保留キュー中に存在することを指定します。このフラグは、T パイプの保留キューから送信されたIMS 出力メッセージに対して常にオンです。したがって、このフラグは、IMS 出力メッセージの保留キュー中にメッセージがあって、それが T パイプの正規のキューから送信されたものかどうかを決めるのに使用することができます。

IMS 出力メッセージが、正規のキューから送信されたものか、保留キューから送信されたものかを決めるためには、状態データのサーバー状態中の「保留キューから」のフラグを検査してください。

保留キューから1つ以上のメッセージを取り出すには、「T パイプ用の保留キューからの出力の再開」プロトコル・コマンドを発行します。

### 追加の info set (TMAMXINF - X'02')

RESUME TPIPE 要求の場合、これは、その要求を出したクライアントのユーザー ID がメッセージ制御情報セクションの最後の8バイトに含まれていることを示します。以下のRESUME TPIPE リクエスト ID (TMAMRTID) を参照してください。

コールアウト・メッセージの場合、これは、要求を出したクライアントのユーザー ID がオフセット 22 の状態データ・セクションに含まれていることを示します。トランザクションおよびコールアウト・メッセージ (コミュニケーションおよびコネクション)にあるRESUME TPIPE ユーザー ID (TMAMRTOD) を参照してください。

### エラー・メッセージの送信 (TMAMCER3 - X'01')

CM1 トランザクションの処理中にエラーが検出され、アプリケーションはIOPCBにISRTを発行しませんでした。したがって、DFS メッセージは、このCM1 トランザクションに対する単なる応答にすぎません。OTMA クライアント、例えば、IMS Connect または IBM MQ は、OTMA がコミット

確認メッセージ内に割り振り解除異常終了フラグをセットした場合でも、クライアント・アプリケーションに DFS メッセージを配信する必要があります。

#### **IMS シャットダウンが進行中 (OMCTSHDN - X'80')**

このフラグは、すべての T パイプに対する中断状態の処理 (TMAMCSPA) は IMS シャットダウンに因るものであることを示しています。

これは、出力フラグのみです。

#### **T パイプ名 (TMAMCTNM)**

トランザクション・パイプ名を指定する 8 バイトの文字フィールド。IMS に対して、この名前は、I/O PCB 上の LTERM 名をオーバーライドするのに使用します。このフィールドは、すべてのトランザクション、データ、コミット確認のメッセージ・タイプに適用できます。また、ある種の応答およびコマンドのメッセージ・タイプにも適用できます。

#### **チェーン・フラグ (TMAMCCHN)**

そのメッセージに含むセグメントの数を指定します。このフラグは、トランザクションおよびデータのメッセージ・タイプに適用可能で、複数セグメント・メッセージの場合必須です。

#### **チェーンの先頭 (TMAMCFIC - X'80')**

複数セグメントを構成するセグメント・チェーン中の最初のセグメントを指定します。メッセージの後続のセグメントには、メッセージ接頭語のメッセージ制御情報セクションのみが必要です。他の適用可能な接頭語セグメント (例えば、トランザクション・メッセージ上でクライアントが指定したものは、最初のセグメント (チェーンの先頭フラグをセット) だけが送信されます。

OTMA メッセージのセグメントが 1 つだけの場合は、チェーンの最後のフラグもセットされなければなりません。

#### **中間チェーン (TMAMCMIC - X'40')**

複数セグメントを構成するセグメント・チェーン内で、最初でも最後でもないセグメントを指定します。これらのセグメントは、メッセージ接頭語のメッセージ制御情報セクションのみを必要とします。

**制約事項:** クライアントおよびサーバーのトークンは、メッセージ接頭語の状態データ・セクションにあるので、セグメント化されたメッセージを相関させたり、結合させたりするには使用できません。この目的のためには、トランザクション・パイプ名と送信シーケンス番号が使用できます。それらは、セグメントごとにメッセージ接頭語のメッセージ制御情報セクションにあります。

#### **チェーンの最後 (TMAMCLIC - X'20')**

複数セグメントの最後のセグメントを指定します。

#### **チェーンの廃棄 (TMAMCCAN - X'10')**

複数セグメント・メッセージのチェーン全体を廃棄しようとしていることを指定します。チェーンの最後のフラグもセットする必要があります。

#### **接頭部フラグ (TMAMCPFL)**

OTMA メッセージに付加されるメッセージ接頭語のセクションを指定します。すべてのメッセージは、メッセージ制御情報および状態データのセクションを持つ必要がありますが、他のセクションの任意の組み合わせも OTMA メッセージで送信できます。

#### **状態データ (TMAMCSTD - X'80')**

メッセージが、メッセージ接頭語の状態データ・セクションを含むことを指定します。

#### **セキュリティー・データ (TMAMCSEC - X'40')**

メッセージが、メッセージ接頭語のセキュリティー・データ・セクションを含むことを指定します。

#### **ユーザー・データ (TMAMCUSR - X'20')**

メッセージが、メッセージ接頭語のユーザー・データ・セクションを含むことを指定します。

#### **アプリケーション・データ (TMAMCAPP - X'10')**

メッセージが、メッセージ接頭語のアプリケーション・データ・セクションを含むことを指定します。

#### **送信シーケンス番号 (TMAMCSSN)**

トランザクション・パイプのシーケンス番号を指定します。このシーケンス番号は、メッセージあるいはトランザクションを送信するとき、クライアントおよびサーバーによって更新されます。

**推奨事項:** その番号を、トランザクション・パイプごとに別々に増やしてください。

この番号は、特定のメッセージを確認する ACK または NAK メッセージと突き合わせるためにも 使用されます。

#### **センス・コード (TMAMCSNS)**

NAK メッセージに付随する 4 バイトのセンス・コードを指定します。TMAMCSNS は 2 つの部分で構成されます。すなわち、2 バイトのセンス・コード (TMAMCSNC) と 2 バイトの理由コード (TMAMCRSC) です。このフィールドに返されるコードの説明については、「IMS V15 メッセージおよびコード 第 2 巻: DFS 以外メッセージ」を参照してください。

#### **センス・コード (TMAMCSNC)**

NAK メッセージに付随する 2 バイトのセンス・コードを指定します。このフィールドに返されるコードの説明については、「IMS V15 メッセージおよびコード 第 2 巻: DFS 以外メッセージ」を参照してください。

#### **理由コード (TMAMCRSC)**

NAK メッセージに付随する 2 バイトの理由コードを指定します。このコードは、センス・コードをさらに修飾します。

#### **ユーザー ID エージング値 (TMAMAGNG)**

入力ユーザー ID の 4 バイトのエージング値を秒数で指定します。このフィールドは、OTMA 接続用の OTMA クライアント・ビッド・コマンドに指定されたエージング値とは異なります。クライアント・ビッド・コマンドのエージング値は、すべての OTMA ユーザー ID に対するデフォルトのエージング値を設定します。ただし、ユーザー ID エージング値は、特定のユーザー ID に対してデフォルトのエージング値をオーバーライドします。ユーザー ID エージング値が 300 秒 (5 分) より少ない場合、IMS は常に、キャッシュされないアクセサー環境エレメント (ACEE) を作成します。

#### **リカバリー可能シーケンス番号 (TMAMCRSQ)**

トランザクション・パイプのリカバリー可能シーケンス番号を指定します。同期化トランザクション・パイプを使用しているリカバリー可能メッセージの各送信で増えます。クライアントおよびサーバーは両方とも、それらのリカバリー可能シーケンス番号を増分し、送信シーケンス番号とは別に維持します。再同期にのみ必要です。

#### **セグメント・シーケンス番号 (TMAMCSEQ)**

複数セグメント・メッセージのセグメントのシーケンス番号を指定します。この番号は、メッセージが必ずしも XCF で順次に送達されるとは限らないので、セグメントごとに更新する必要があります。

メッセージのセグメントが 1 つだけの場合は、この番号は、値 1 でなければなりません。

#### **RESUME TPIPE リクエスター ID (TMAMRTID)**

RESUME TPIPE 要求を発行しているクライアントのユーザー ID。

#### **関連概念**

##### 863 ページの『OTMA でのクライアント/サーバー再同期』

クライアント・トランザクションが一度だけ処理されることを保証するために、OTMA は、同期化トランザクションのプロトコルを提供します。

#### **関連資料**

##### 866 ページの『OTMA 再同期プロトコル』

OTMA 再同期は、各クライアントでの一連のコマンド交換に基づきます。

##### 899 ページの『状態データ・セクション』

状態データは、すべての OTMA メッセージで必須です。これは、メッセージ接頭語のメッセージ制御情報セクションのすぐ後にあります。トランザクション関連情報が入っています。

##### 914 ページの『セキュリティー・データ・セクション』

セキュリティー・データ・セクションは、各トランザクションあるいはコマンドで必須で、OTMA プロトコル・コマンドではオプションです。

##### 917 ページの『ユーザー・データ・セクション』

OTMA メッセージ接頭語のユーザー・データ・セクションは可変長で、セキュリティー・データ・セクションの後に続きます。どんなデータでも含むことができます。

##### 918 ページの『アプリケーション・データ・セクション』

OTMA メッセージのアプリケーション・データ・セクションは可変長で、メッセージ接頭語のユーザー・データ・セクションの後に続きます。

#### 904 ページの『トランザクションおよびコールアウト・メッセージ』

OTMA メッセージ接頭語のトランザクション関連情報および同期コールアウト要求情報の状態データは、DFSYSMSG マクロの TMAMHDR DSECT によってマップされます。

#### 912 ページの『T パイプ用の保留キューからの出力再開』

OTMA メッセージ接頭語の T パイプに対する保留キューの出力再開部分の状態データは、DFSYSMSG マクロの TMAMHDR DSECT によってマップされます。

## 状態データ・セクション

状態データは、すべての OTMA メッセージで必須です。これは、メッセージ接頭語のメッセージ制御情報セクションのすぐ後にあります。トランザクション関連情報が入っています。

状態データ・セクションは、トランザクション関連情報の場合と OTMA プロトコル・コマンドの場合ではフォーマットが異なります。状態データ・セクションの後にセキュリティー・データ・セクションが続くことがあります。

## サーバー使用可能コマンドおよびクライアント・ビッド・コマンド

OTMA メッセージ接頭語のサーバー使用可能コマンドおよびクライアント・ビッド・コマンド用の状態データは、DFSYSMSG マクロの TMAMHDR DSECT によってマップされます。

以下の表は、コマンド・メッセージの状態データのフォーマットをバイト・オフセット、長さ、内容、16 進値、意味、および使用法のコメントで要約したものです。

表 158. サーバー使用可能化およびクライアント・ビッド・コマンドのフォーマット

バイト	長さ	内容	値	意味
0	2	長さ		長さフィールドを含む、メッセージ接頭語の状態データ・セクションの長さの合計を指定します。
2	16	メンバー名		発信元サーバーの z/OS システム間カップリング・ファシリティ (XCF) メンバー名を指定します。
18	8	発信元トークン		メッセージの発信元 (クライアントかサーバー) の XCF メンバー・トークンを指定します。
26	8	宛先トークン		メッセージの宛先 (クライアントかサーバー) の XCF メンバー・トークンを指定します。
注: 以下のフィールドは、クライアント・ビッド・コマンドにのみ現れます。				
34	8	DRU 出口名		OTMA 宛先解決出口ルーチンの名前を指定します。
42	2	最大ブロック・サイズ		サーバーとクライアント間の XCF 会話用の最大ブロック・サイズを指定します。このフィールドはオプションです。

表 158. サーバー使用可能化およびクライアント・ビッド・コマンドのフォーマット (続き)

バイト	長さ	内容	値	意味
44	1	クライアント・ビッド・フラグ	X'80'	<p>保留キュー: Tパイプに対する保留キューが必要であることを指定します。</p> <p>保留キューは、NAK 応答を受信するコミット後送信出力と、OTMA クライアント用の代替 PCB 出力を保持できます。キュー中の出力メッセージは、クライアントがメッセージを送達することを要求するまで、送達されません。保留キューの使用はオプションです。保留キューがない場合、Tパイプに対する正規キューが、すべてのコミット後送信出力メッセージを保持し、送達するために使用されます。デフォルトでは、Tパイプに対する保留キューはありません。</p> <p>OTMA 再同期プロトコルは現在保留キューをサポートしていません。</p>
			X'40'	OTMA 呼び出し可能インターフェース用に予約済み。
			X'20'	IMS Connect 用に予約済み。
			X'10'	IBM MQ 用に予約済み。
			X'08'	同期プログラム間通信用に予約済み。
45	1	追加クライアント・ビッド・フラグ	X'80'	<p>メッセージあふれ制御: メッセージあふれ制御用にクライアント・ビッド・コマンドによってバイト 62 に指定された入力メッセージの最大数を、OTMA で使用することを指定します。</p>
			X'20'	<p>メンバー・タイムアウト: クライアント・ビッド・コマンドによってバイト 65 で指定された送信後コミット・メッセージのメンバー・タイムアウト値を、OTMA で使用することを指定します。</p>
			X'10'	<p>カスケード・トランザクション・サポート: 1つの LPAR 上の OTMA クライアントから別の LPAR 上の OTMA へのグローバル RRS トランザクション (synchlevel=2、つまり同期点) のサポートを指定します。</p>
			X'08'	<p>スーパーメンバー・フラグ: このメンバーまたはクライアントについて、スーパーメンバー 機能をアクティブにします。</p> <p>このオプションのスーパーメンバー 機能を使用するには、メンバーまたはクライアントはこのフラグを設定し、スーパーメンバー 名を指定する必要があります。下記のバイト 54 のスーパーメンバー 名を参照してください。</p>
			X'04'	<p>同期コールアウト・サポート: IMS アプリケーションによって発行された同期コールアウト・メッセージのサポートを指定します。</p>
			X'02'	<p>CMO タイムアウト・キュー名: CMO タイムアウト・キュー名を設定します。</p>
			X'01'	<p>IMS 間 TCP/IP 通信: IMS Connect が IMS 間 TCP/IP 通信をサポートすることを示すために、IMS Connect によって設定されます。</p>

表 158. サーバー使用可能化およびクライアント・ビッド・コマンドのフォーマット (続き)

バイト	長さ	内容	値	意味
46	4	エージング値		<p>アクセサー環境エレメント (ACEE) のエージング値を秒単位で指定します。</p> <p>ユーザー ID の既存の ACEE の経過時間がこの値より大きい場合、IMS は新しい ACEE を作成します。</p> <p>指定したエージング値がサポートされる最小値より小さい場合、IMS はキャッシュされない ACEE を作成しません。</p> <p>キャッシング・サポートの最小値は 300 秒 (5 分) です。</p>
50	4	ハッシュ・テーブル・サイズ		<p>複数セグメントのメッセージ入力を処理するための IMS OTMA ハッシュ・テーブルのサイズを定義します。</p> <p>ハッシュ・テーブルは、すべての入力セグメントを正しくチェーニングするために使用されます。推奨値は、X'00000065' です。</p>
54	4	スーパーメンバー名		<p>1 バイトから 4 バイトのスーパーメンバー名を指定します。</p> <p>スーパーメンバー機能がアクティブにされた場合、保留キュー対応のメンバーまたはクライアント (例えば、IMS Connect) に対するすべての非同期出力は、同じスーパーメンバー名を指定する一連のクライアントによって共有されます。</p>
58	2	コールアウト・トークンのオフセット		<p>このクライアントが OTMA によって配置される同期コールアウト・トークンを予期する、ユーザー・データ・セクション内のオフセットを指定します。</p>
60	2	リモート宛先定義へのオフセット		<p>リモート IMS システムにメッセージを送達するために必要な定義を検出できる、ユーザー・データ・セクション内のオフセットを指定します。</p>
62	2	メッセージあふれしきい値	0-9999	<p>このメンバーまたはクライアントからの並行入力メッセージの最大数を指定します。</p> <p>0 を指定した場合、OTMA のデフォルトは 5000 です。1 から 200 までの値は、200 として扱われます。</p>
64	1	複数のアクティブな Resume TPIPE のサポート	X'80'	<p>この T パイプに MULTIRTP=Y が設定されます。T パイプは、複数のアクティブな RESUME TPIPE 要求をサポートします。</p>
			X'40'	<p>この T パイプに MULTIRTP=N が設定されます。T パイプは、アクティブな RESUME TPIPE 要求を一度に 1 つだけサポートします。</p>



表 158. サーバー使用可能化およびクライアント・ビッド・コマンドのフォーマット (続き)

バイト	長さ	内容	値	意味
65	1	クライアントからの確認応答用のタイムアウト値		<p>コミット後送信 (CM0) および送信後コミット (CM1) の両方の OTMA メッセージに対するメンバー・レベルのタイムアウト値 (秒) を指定します。</p> <p>CM1 メッセージの場合、このタイムアウト値は、synclevel=confirm または synclevel=syncpt の場合にのみ適用されます。</p> <p>有効なタイムアウト間隔は、1 秒から 255 秒までです。タイムアウト値を指定しなかった場合、OTMA のデフォルトは 120 秒です。</p> <p>このフィールドに指定したタイムアウト値は、OTMA によって設定されたタイムアウト値を超えることはできず、超えた場合は無視されます。</p> <p>タイムアウト値は、OTMA で /START TMEMBER コマンドによって、または OTMA クライアント記述子によって設定されます。</p>
66	8	CM0 タイムアウト・キュー名		<p>コミット後送信 (CM0) 出力用の T パイプ・タイムアウト・キューの 8 バイトの名前を指定します。</p>

#### 関連資料

267 ページの『IMS Connect で使用される OTMA 状態データ・フィールド』

このトピックの表では、OTMA 状態データ・ヘッダーのフィールドおよびそれらのフィールドの順序を説明します。

## SRVresynch コマンド

OTMA メッセージ接頭語の SRVresynch コマンド部分の状態データは、DFSYMSG マクロの TMAMHDR DSECT によってマップされます。

SRVresynch コマンドは、IMS によって送信され、既知の同期化 T パイプ名をすべてクライアントに渡します。コマンド・データが、単一バッファに収まらない場合は、チェーニングされた複数セグメント・バッファが代わりに使用されます。以下の表は、SRVresynch コマンドの状態データのフォーマットを要約したものです。この要約には、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 159. SRVresynch コマンドのフォーマット

バイト	長さ	内容	説明
0	2	長さ	長さフィールドを含む、状態データ・セクションの長さ。
2	8	T パイプ名	<p>トランザクション・パイプ名。</p> <p>必要なだけ T パイプ名を繰り返すことができます。</p>

## REQresynch コマンド

OTMA メッセージ接頭語の REQresynch コマンド部分の状態データは、DFSYMSG マクロの TMAMHDR DSECT によってマップされます。

REQresynch コマンドは IMS によって使用され、送信シーケンス番号および特定 T パイプに対する受信シーケンスをクライアントに渡します。以下の表は、REQresynch コマンドの状態データのフォーマットを要約したものです。この要約には、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 160. REQresynch コマンドのフォーマット

バイト	長さ	内容	説明
0	2	長さ	長さフィールドを含む、状態データ・セクションの長さ。
2	8	Tパイプ名	トランザクション・パイプ名。
10	4	送信シーケンス番号	トランザクション・パイプの IMS リカバリー可能送信シーケンス番号。
14	4	受信シーケンス番号	トランザクション・パイプの IMS リカバリー可能受信シーケンス番号。
18	1	Tパイプ・フラグ 1	将来の利用のために予約済み。
19	1	Tパイプ・フラグ 2	将来の利用のために予約済み。
20	6	RESERVED	

## REPresynch コマンド

OTMA メッセージ接頭語の REPresynch コマンド部分の状態データは、DFSMSG マクロの TMAMHDR DSECT によってマップされます。

REPresynch は、IMS からの REQresynch 要求に応答して、クライアントが送信します。これは Tパイプに対する要求状態情報を含みます。以下の表は、REPresynch コマンドの状態データのフォーマットを要約したものです。この要約には、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 161. REPresynch コマンドのフォーマット

バイト	長さ	内容	値	説明
0	2	長さ		長さフィールドを含む、状態データ・セクションの長さ。
2	8	Tパイプ名		トランザクション・パイプ名。
10	4	送信シーケンス番号		トランザクション・パイプに対するクライアント・リカバリー可能送信シーケンス番号。
14	4	受信シーケンス番号		トランザクション・パイプに対するクライアント・リカバリー可能受信シーケンス番号。
18	1	Tパイプ・フラグ 1	X'00'	続行: 最後のメッセージの受信が済み、IMS はこの同期トランザクション・パイプの処理を続行します。
			X'04'	最後の出力のデキュー: IMS は、最後の出力メッセージをデキューできます。リカバリー可能送信シーケンス番号が更新される。
			X'08'	シーケンス番号のリセット: IMS は、このコマンドで渡されたとおりに、リカバリー可能送信シーケンス番号およびリカバリー可能受信シーケンス番号をリセットします。
			X'0C'	Tパイプの停止: IMS は、この同期トランザクション・パイプを停止します。
			X'10'	Tパイプの停止および TBresynch の待機: IMS は、この同期トランザクション・パイプを停止し、クライアントからの TBresynch を待ちます。
19	1	Tパイプ・フラグ 2		予約済み。

表 161. REPresynch コマンドのフォーマット (続き)

バイト	長さ	内容	値	説明
20	6	予約済み		

## TBresynch コマンド

OTMA メッセージ接頭語の TBresynch コマンド部分の状態データは、DFSYSMSG マクロの TMAMHDR DSECT によってマップされます。

TBresynch コマンドは、クライアントが、IMS からの REQresynch を受信する準備ができている場合、クライアントから IMS に送信されます。TBresynch コマンドは、以下の 2 つの状態で発行することができます。

- クライアントは、REPresynch を"TBresynch を停止し、待機する"を指定して、IMS に送信した後、ACK メッセージを受信した。
- クライアントは、初期に遅れのない再同期が、この T パイプに対し完了した後の任意の時点で、IMS を使用して TBresynch を要求する。

以下の表は、TBresynch コマンドの状態データのフォーマットを要約したものです。この要約には、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 162. TBresynch コマンドのフォーマット

バイト	長さ	内容	説明
0	2	長さ	長さフィールドを含む、状態データ・セクションの長さ。
2	8	T パイプ名	トランザクション・パイプ名。

## トランザクションおよびコールアウト・メッセージ

OTMA メッセージ接頭語のトランザクション関連情報および同期コールアウト要求情報の状態データは、DFSYSMSG マクロの TMAMHDR DSECT によってマップされます。

以下の表は、トランザクション関連情報および同期コールアウト要求情報の状態データのフォーマットを要約したものです。この要約では、バイト、長さ、内容、16 進値、および各フィールドの使用説明を示しています。

表 163. トランザクション関連情報の状態データのフォーマット

バイト	長さ	内容	値	説明
0	2	長さ		長さフィールドを含む、メッセージ接頭語の状態データ・セクションの長さの合計を指定します。

表 163. トランザクション関連情報の状態データのフォーマット (続き)

バイト	長さ	内容	値	説明
2	1	サーバー状態: トランザクションが稼働しているモードを指定します。	X'80'	<p>会話型状態: 会話型トランザクション用。</p> <p>サーバーおよびクライアントは、どちらも会話データを送信するときに、このフラグをセットします。</p>
			X'40'	<p>応答モード: 応答モード・トランザクション用。</p> <p>出力を送信するときサーバーがセットします。</p> <p>この状態は、OTMA がセッションまたは端末を使用しないので、OTMA サーバーでは、あまり意味を持ちません。</p> <p>コミット後送信 (CM0) トランザクションに対して、クライアントによって設定されます。CM0 トランザクションに対してこのフラグが設定されているときに、IMS アプリケーションが IOPCB に応答しないか、別のトランザクションへのメッセージ通信を完了しない場合、トランザクションの応答モードに関係なく、OTMA はクライアントに対して DFS2082 メッセージを発行します。</p>
			X'20'	<p>保留キューから: 出力メッセージが T パイプ用の IMS 保留キューから送信されたことを指定します。</p> <p>サーバーは最初、コミット後送信出力メッセージを送信しているとき、このフラグをセットします。クライアントは、後続の ACK あるいは NAK を IMS に送信しているときも、このフラグをセットする必要があります。</p>
			X'10'	<p>保留キュー対応: 非同期出力を保留キューへ送信できます。</p> <p>サーバーによってセットされます。</p>
			X'08'	<p>転送メッセージ: 出力は転送されました。</p> <p>転送された出力が処理されるときにサーバーによってセットされます。</p>
			X'04'	<p>CM1 タイムアウト: この CM1 出力について、タイムアウトが発生します。</p> <p>サーバーによってセットされます。</p>
			X'02'	<p>この値は、メッセージ・タイプによって異なる意味を持ちます。</p> <p>トランザクションの有効期限: 時間の長さ: トランザクション・メッセージでは、X'02' を指定すると、トランザクションの有効期限機能が有効になり、指定された長さの時間が経過すると、サブミットされたトランザクションの有効期限が切れます。時間の長さは TMAMOSXP フィールドに指定します。</p> <p>ネットワーク・セキュリティ情報が含まれる: RESUME TPIPE 呼び出しを発行したクライアントへのコールアウト・メッセージでは、X'02' は、メッセージのセキュリティ・データ・セクションにネットワーク・セキュリティ情報が含まれることを指定します。</p>
			X'01'	<p>この値は、メッセージ・タイプによって異なる意味を持ちます。</p> <p>トランザクション有効期限: 時点: クライアントからのトランザクション・メッセージでは、X'01' は、トランザクションの有効期限機能を有効にし、サブミットされたトランザクションは指定された時点で有効期限が切れることを示します。時点は、ユーザー・データ接頭部に STCK フォーマットで指定します。トランザクション・メッセージの状態データのバイト 22 (TMAMOSXP フィールド) で時刻指定のバイト・オフセットを指定します。</p> <p>RESUME TPIPE トークンが含まれる: T パイプの保留キューからサーバーによって送信されたメッセージの場合、X'01' は、その出力メッセージには OTMA 状態データ接頭部のバイト 48 (TMAMRTOK) に RESUME TPIPE トークンが含まれていることを示します。</p>

表 163. トランザクション関連情報の状態データのフォーマット (続き)

バイト	長さ	内容	値	説明
31		同期またはコールドアウト・フラグ: トランザクションのコミット・モードを指定するか、メッセージが同期コールドアウト要求であることを指定します。	X'80'	<i>TMAMHCTD</i> : このコールドアウト・メッセージには、メッセージのアプリケーション・データ・セクションに制御データがあります。
			X'40'	コミット後送信: コミット後送信 (CM0) トランザクション。 サーバーは、出力をクライアントへ送信する前にコミットします。
			X'20'	送信後コミット: 送信後コミット (CM1) トランザクション。 サーバーは、出力をコミットする前にクライアントに送信します。
			X'10'	メッセージ応答なしの送信専用 ACK: ACK 付き送信専用プロトコルを使用し OTMA に送信される、同期コールドアウト要求への応答メッセージの場合、この値は、応答メッセージへの ACK には応答データも含まれないことを示します。
			X'08'	同期コールドアウト・メッセージ: IMS 従属領域内で実行されている IMS アプリケーションから発信された同期コールドアウト・メッセージ。
			X'04'	IMS 間 TCP/IP 通信メッセージ: IMS Connect が IMS 間 TCP/IP 通信をサポートすることを示すために、IMS Connect によって設定されます。
			X'02'	トランザクション有効期限切れに対するダンプが必要: アプリケーションが IMS 入力キュー上のこのトランザクションをリトリブするための GU 呼び出しを発行する前にこのトランザクションの有効期限が切れた場合、症状ダンプと DFS554A メッセージを要求するために、クライアントによって設定されます。
			X'01'	FE IMS への SQ BE ALTPCB メッセージの送信: 共用キュー環境のバックエンド IMS システムで生成された出力をフロントエンド IMS システムを介してクライアントに返すことを要求するために、クライアントによって設定されます。
4	1	同期レベル: トランザクション同期レベル、つまりクライアントと IMS アプリケーション・プログラムがプログラム出力メッセージについて対話する方法を指定します。  デフォルトは「確認」で、IMS はコミット後送信出力をクライアントに送信するときに、常に応答を要求します。	X'00'	なし: z/OS リソース・リカバリー・サービス リカバリー・プラットフォームの下で会話時に更新されたリソースに対する調整コミット処理に、各プログラムが参加することを指定します。  サーバー・アプリケーション・プログラムは、出力をクライアントに送信するとき、ACK メッセージを要求しません。  「なし」だけが、送信後コミット・トランザクションに有効です。
			X'01'	確認: 同期が要求されることを指定します。  サーバーは、メッセージ接頭語のメッセージ制御情報セクション中に、「応答要求」をセットした応答フラグとともにトランザクション出力を送信します。  「確認」は、コミット後送信あるいは送信後コミット・トランザクションのいずれにも使用できます。
			X'02'	<i>Syncpt</i> : RRS リカバリー・プラットフォームの下で会話時に更新されたリソースに対する協調コミット処理に、各プログラムが参加することを指定します。  このレベルでの会話は保護会話とも呼ばれ、この会話の下で更新されたリソースは、2 フェーズ・コミット・プロトコルを使用します。

表 163. トランザクション関連情報の状態データのフォーマット (続き)

バイト	長さ	内容	値	説明
5	1	クライアント・フラグ: クライアントに要求されたオプションの処理を指定します。	X'80'	送信専用メッセージ: これは送信専用メッセージです。応答は保留キューに置かれます。  このフラグは、クライアントが、オープン中の保留キューを要求した場合のみ、有効です。
			X'40'	コールアウトのエージング値または NAK メッセージ  エージング値に対して指定された場合、このフラグは、メッセージ制御情報のバイト 20 で指定されたアクセサ環境エレメント (ACEE) のエージング値を受け入れるように IMS に指示します。  同期コールアウト・メッセージに対する NAK 応答について指定された場合、OTMA はコールアウト・メッセージを、別の RESUME TPIPE 呼び出したりはコールアウト・メッセージ・タイムアウトによって取り出されるまで、キューの中に保持します。
			X'20'	転送要求: 転送が要求されました。  このフラグをセットすると、CMO 出力を <b>Destination Override</b> フィールドに指定された宛先に転送します。
			X'10'	CMO IOPCB 出力のバージ: CMO IOPCB 出力をクライアントへ配信できない場合、その出力を削除します。  クライアントによって、入力メッセージの中でセットされます。
			X'08'	メッセージ・レベルのタイムアウトの使用可能化: メッセージ制御情報セクションで指定されたタイムアウト値を使用可能にします。  このタイムアウト値は、この特定の CM1 メッセージにのみ適用されます。メッセージ・レベルのタイムアウトが指定されていない場合は、メンバー・レベルのタイムアウト値が使用されます。
			X'04'	廃止。EWLM は、IMS によってサポートされなくなりました。このフラグが指定されている場合、OTMA によって無視されます。
			X'02'	PURG 呼び出しの無視: TP PCB 内に複数の PURG 呼び出しを生成するコミット後送信 (CMO) メッセージの場合、このオプションのフラグを設定すると、IMS は PURG 呼び出しを無視します。これにより、OTMA クライアント・アプリケーションは、複数の出力セグメントで 1 つの応答 CMO メッセージを受信します。  このフラグがセットされていない場合、1 つの CMO メッセージの TP PCB に複数の ISRT および PURG 呼び出しが存在すると、OTMA クライアント・アプリケーションは複数の応答メッセージを受信します。  このフラグは、CMO 入力メッセージにのみ適用されます。このフラグを CM1 入力メッセージで指定した場合、IMS は残っているすべてのプログラム間通信および ALT-PCB 処理について、フラグをリセットし、無視します。
X'01'	送信専用 (順次): プロトコルを順次配信付き送信専用として指定します。			
6	8	マップ名		入力または出力データ・ストリーム (例えば、3270 データ・ストリーム) をマップするために、サーバーが使用するフォーマット設定マップを指定します。  OTMA が、MFS サポートを提供しなくても、出力データ・ストリームを定義するためにマップ名を使用することができます。名前は、I/O PCB に置かれる 8 バイトの MOD 名です。IMS は、メッセージが挿入されるとき、接頭部中のこのフィールドを、I/O PCB 中のマップ名で置き換えます。  マップ名はオプションです。
14	16	サーバー・トークン		CM1 メッセージの場合、サーバーは、関連の目的でサーバー・トークンを設定します。クライアントは、サーバーから CM1 出力メッセージを受け取った後、後続の確認応答 (ACK または NAK) あるいは会話型反復で、サーバー・トークンを返す必要があります。
サーバー・トークンが指定されないか該当しない場合のバイト 14 から 29 の代替マッピング				

表 163. トランザクション関連情報の状態データのフォーマット (続き)

バイト	長さ	内容	値	説明
14	4	スーパーメンバー名		ALTPCB からの出力に、OTMAYPRX ユーザー出口および DFSYDRUO 出口ルーチンのスーパーメンバー名を指定します。
18	2	トランザクションの有効期限の値またはオフセット		サーバー状態フラグが X'02' (TMAMTXP2) に設定された場合、このフィールドは、入力トランザクションの有効期限が切れるまでの時間の長さを含みます。 サーバー状態フラグが X'01' (TMAMTXP1) に設定された場合、このフィールドは、ユーザー・データ接頭部に STCK フォーマットで指定された時点のオフセットを含みます。 このフィールドは、CM1 会話型反復には使用できません。
20	2	同期コールアウト・メッセージの関連トークンへのオフセット		クライアントが同期コールアウト・メッセージ内で関連トークンを検出できる、ユーザー・データ・セクションのオフセットを指定します。 この値は、クライアント・ビッド・メッセージ内の状態データの TMAMOSYN フィールドでクライアントが指定したオフセットに一致する必要があります。
22	8	RESUME TPIPE ユーザー ID		RESUME TPIPE 要求を出すクライアントのユーザー ID を指定します。
バイト 14 から 29 の代替マッピングの終わり				
30	16	関連トークン		入力を出力と関連させるクライアント・トークン。 このトークンはオプションで、サーバーでは使用しません。 <b>推奨事項:</b> このトークンを使用すると、クライアントのトランザクション管理に役立ちます。
46	16	CM1 コンテキスト ID		クライアントによって送信される入力トランザクション・メッセージで、SYNCLVL=02 および保護会話で使用される RRS トークンを指定します。
CM1 コンテキスト ID が使用されない場合のバイト 46 から 61 の代替マッピング				
46	8	RESUME TPIPE トークン		Tパイプの保留キューからサーバーによって送信される出力メッセージで、このフィールドには RESUME TPIPE 要求トークンが入ります。
54	8	XCF から受信した時刻		トランザクションの有効期限が切れた後にクライアントに返されるメッセージ内で、このフィールドには、有効期限が切れたトランザクションを IMS が XCF から受け取った時刻が入ります。
バイト 46 から 61 の代替マッピングの終わり				
54	8	コールアウトのプログラム名		IMS 従属領域から送信される IMS 同期コールアウト・メッセージ内で、このフィールドは、DL/I ICAL 呼び出しを行うプログラムを示します。
62	8	宛先オーバーライド		IMS アプリケーション・プログラムの I/O PCB 中の LTERM 名をオーバーライドするのに使用する LTERM 名を指定します。 このオーバーライドは、トランザクション・パイプ名で I/O PCB 中の LTERM 名をオーバーライドしたくない場合に使用されます。 このオプションのオーバーライドは、ブランクで始まる場合は使用されません。
70	2	サーバー・ユーザー・データ長		存在する場合は、サーバー・ユーザー・データの長さを指定します。 サーバー・ユーザー・データの最大長は、256 バイトです。サーバー・ユーザー・データの長さは、長さの計算には含まれません。
72	*	サーバー・ユーザー・データ		サーバーが必要とするデータを指定します。 それが、クライアントによるトランザクション・メッセージに含まれている場合は、サーバーが出力データ・メッセージ中に戻ります。可変長です。オプションです。



## サーバー状態プロトコル・コマンド

サーバー状態プロトコル・コマンドの状態データは、DFSYMSG マクロの TMAMHDR DSECT によってマップされます。

サーバー状態プロトコル・コマンドは、OTMA によって使用されます。その目的は、IMS 処理の現行状態または状態の変更を OTMA クライアントに通知することと、60 秒間隔で発行されるハートビート・メッセージとして使用することです。

サーバー状態プロトコル・コマンドは、OTMA ヘッダーのメッセージ制御情報セクションのコマンド・タイプ・フィールド (TMAMCTYP) 内の値 X'3C' で識別されます。

以下の表は、サーバー状態プロトコル・コマンドの状態データのフォーマットを要約したものです。この要約には、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 164. サーバー状態プロトコル・コマンドのフォーマット

10 進オ フセ ット	16 進オ フセ ット	長さ	フィールド名	値と説明
0	X'00 	2	TMAMHLEN	長さフィールド自体を含む、状態データ・セクションの長さ。
2	X'02 	2	TMAMRSIM_STATUS	サーバーの OTMA リソースの状態。 <b>X'03' - 正常状態</b> IMS は使用可能であり、OTMA メッセージを正常に処理しています。 <b>X'02' - 機能低下状態</b> IMS の OTMA メッセージの処理速度が低下しています。1つ以上の状態が IMS の OTMA メッセージの処理速度が通常よりも低下していることを示す場合、OTMA は機能低下状態プロトコル・コマンドを発行します。 <b>X'01' - 使用不可状態</b> IMS は、OTMA トランザクションの処理を受け入れることができなくなっています。OTMA は、使用不可状態プロトコル・コマンドを発行して、1つ以上の重大な状態により IMS が OTMA メッセージを処理できないというアラートを OTMA クライアントに出します。
4	X'04 	1	TMAMRSIM_SVRFLG1	使用不可リソースの 1 番目のグループのリソース用フラグ。 <b>X'80' - TMAMRSIM_S1SHTDN</b> IMS サーバーはシャットダウン中であり、使用できなくなっています。
5	X'05 	1	TMAMRSIM_SVRFLG2	予約済み
6	X'06 	1	TMAMRSIM_SVRFLG3	予約済み
7	X'07 	1	TMAMRSIM_SVRFLG4	使用不可リソースの 4 番目のグループのリソース用フラグ。 <b>X'01' - TMAMRSIM_S4FLOOD</b> サーバーは、処理を待っている OTMA メッセージであふれ、使用できなくなっている。

表 164. サーバー状態プロトコル・コマンドのフォーマット (続き)

10 進オ フセ ット	16 進オ フセ ット	長さ	フィールド名	値と説明
8	X'08 ,	1	TMAMRSIM_WRNFLG1	機能低下リソースの 1 番目のグループのリソース用フラグ。 <b>X'80' - TMAMRSIM_W1FLOOD</b> すべての OTMA クライアントに関する全体的なあふれ警告 <b>X'40' - TMAMRSIM_W1MTP</b> T パイプの最大許容数を指定する、OTMA クライアントに対する警告。T パイプの総数がグローバル警告しきい値に達しました。グローバル警告しきい値は、いずれかの OTMA クライアント記述子の MAXTP パラメーターで指定された最高値によって定義されます。 <b>X'20' - TMAMRSIM_W1MTPF</b> DFSOTMA クライアント記述子項目によって定義された MAXTP 制限に到達しました。この後の、すべての OTMA クライアントからの新規 T パイプの要求は、リジェクトされます。共用キュー環境で処理されているアプリケーション GU に関して、MAXTPBE=NO がバックエンド IMS システムに定義されている場合は例外になります。 <b>X'10' - TMAMRSIM_W1MTP80</b> IMS システム内の T パイプの数が、DFSOTMA システム・クライアント記述子の MAXTP パラメーターによって定義されているグローバル T パイプ制限の 80% に達しました。
9	X'09 ,	1	TMAMRSIM_WRNFLG2	予約済み
10	X'0 A'	1	TMAMRSIM_WRNFLG3	予約済み
11	X'0 B'	1	TMAMRSIM_WRNFLG4	機能低下リソースの 4 番目のグループのリソース用フラグ。 <b>X'08' - TMAMRSIM_W5MTP</b> この OTMA クライアント用の T パイプの数が、OTMA クライアント記述子の MAXTP パラメーターでこのクライアントに設定された T パイプの最大許容数に達しました。T パイプの数が減少するまでは、この OTMA クライアントに対して新規の T パイプは作成できません。 <b>X'04' - TMAMRSIM_W4MTP</b> この OTMA クライアント用の T パイプの数が、OTMA クライアント記述子の MAXTP パラメーターでこのクライアントに対して設定された T パイプの最大許容数の 80% に達しました。 <b>X'02' - TMAMRSIM_AWE</b> メッセージ AWE が 80% のあふれに達しています。 <b>X'01' - TMAMRSIM_W4FLOOD</b> このクライアントのみに対するあふれ警告。サーバーで処理を待機中の OTMA メッセージの数が、サーバーに定義された最大許容数の 80% に達しています。

表 164. サーバー状態プロトコル・コマンドのフォーマット (続き)

10 進オット	16 進オット	長さ	フィールド名	値と説明
12	X'0C '	1	TMAMRSIM_NRSFLGS	非リソース関連標識に関するその他のフラグ。  <b>X'80' - TMAMRSIM_HB60S</b> このメッセージをハートビート・メッセージとして識別します。サーバーは使用可能であり、リソースの使用量は通常の制限内です。ハートビート・メッセージは、60 秒ごとに送信される。
13	X'0 D'	3	予約済み	
16	X'10 '	16	TMAMRSIM_SRVNAME	OTMA サーバーの 16 文字の z/OS システム間カップリング・ファシリティ (XCF) メンバー名。
32	X'20 '	16	TMAMRSIM_CLTNAME	OTMA クライアントの 16 文字の XCF メンバー名。
48	X'30 '	20	予約済み	
68	X'44 '	12	TMAMRSIM_UTC	このメッセージの UTC 時間

#### 関連概念

815 ページの『OTMA リソース・モニター』

OTMA は、OTMA トランザクションの処理に使用される IMS システム・リソースをモニターし、IMS システムの OTMA トランザクション処理の状況を OTMA クライアントに通知します。

#### 関連資料

889 ページの『メッセージ制御情報セクション』

各 OTMA メッセージごとに、OTMA メッセージ接頭語の最初のセクションにメッセージ制御情報を指定する必要があります。

## T パイプに対する出力の再開

OTMA メッセージ接頭語の T パイプに対する出力再開部分の状態データは、DFSYSMSG マクロの TMAMHDR DSECT によってマップされます。

このコマンドは、任意のキューされた出力を強制的に再送するために、クライアントが送信します。コマンドでは、T パイプの数と T パイプ名が必要です。

その T パイプに対する保留キューが存在し、メッセージが保持されている場合は、それらのメッセージもクライアントに送信されます。以下の表は、T パイプに対する出力再開の状態データのフォーマットを要約したものです。この要約には、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 165. T パイプ・コマンドに対する出力再開のフォーマット

バイト	長さ	内容	説明
0	2	長さ	状態データ・セクションの長さ。
2	2	T パイプ・カウント	コマンド中の T パイプ名の数。
4	8	T パイプ名	トランザクション T パイプ名。  必要に応じて、異なった T パイプ名を追加できます。

## 「すべての T パイプからの出力再開」プロトコル・コマンドのフォーマット

「すべての T パイプからの出力再開」プロトコル・コマンドのフォーマットは、DFSYMSG マクロの TMAMHDR DSECT によってマップされます。

「すべての T パイプからの出力再開」プロトコル・コマンドは、OTMA がクライアントに関連付けられたすべての T パイプからキューに入れられた出力メッセージの送信を再開するよう要求するために、OTMA クライアントによって送信されます。このプロトコル・コマンドは、T パイプの保留キューからの出力は再開しません。

IMS 間 TCP/IP 通信の場合、IMS Connect は、リモート IMS Connect インスタンスへの接続が復元した後、出力を再開するためにこのプロトコル・コマンドを発行することができます。この場合、IMS Connect は、リモート IMS Connect インスタンスの名前をこのコマンドと一緒に含めることができます。OTMA は、メッセージにもそのリモート IMS Connect インスタンスの名前が含まれていれば、T パイプ上の最も古いメッセージを送信します。

以下の表は、OTMA メッセージ接頭語の状態データ・セクション内の「すべての T パイプからの出力再開」プロトコル・コマンドのフォーマットを示しています。

表 166. 「すべての T パイプからの出力再開」コマンドのフォーマット

バイト	長さ	内容	説明
0	2	長さ	状態データ・セクションの長さ。
2	2	予約済み	予約フィールド。
4	8	リモート IMS Connect インスタンスの名前	IMS Connect は、このフィールドを使用して、特定のリモート IMS Connect インスタンスを宛先とする、T パイプ・キュー上のメッセージをリトリブします。

## T パイプ用の保留キューからの出力再開

OTMA メッセージ接頭語の T パイプに対する保留キューの出力再開部分の状態データは、DFSYMSG マクロの TMAMHDR DSECT によってマップされます。

OTMA クライアントは、コマンドを送信して、T パイプに対する保留キュー上の 1 つあるいはすべてのキューされたメッセージを送達することを IMS に通知します。このコマンドを出さない場合は、メッセージは、保留キュー中に保持されることがあります。しかし、コマンドの中で指定したオプションを使用して、IMS にメッセージを保持し、送信する方法を要求することができます。以下の表の 4 つのオプションのうちの 1 つを状態データに指定できます。クライアントあるいは z/OS システム間カップリング・ファシリテーターが IMS に NAK メッセージを戻す場合、現在のオプションは No-Auto にリセットされ、それがデフォルトになります。

以下の表は、T パイプ保留キューの出力再開の状態データのフォーマットを要約したものです。この要約には、必要に応じて、バイト、長さ、内容、16 進値、および説明が含まれています。

表 167. T パイプ・コマンドに対する保留キュー用の出力再開のフォーマット

バイト	長さ	内容	値	説明
0	2	長さ		状態データ・セクションの長さ。

表 167. T パイプ・コマンドに対する保留キュー用の出力再開のフォーマット (続き)

バイト	長さ	内容	値	説明
2	1	配信オプション	X'00'	No-Auto: コマンドが出されたときのみ、キュー中のすべてのメッセージを処理しきる。これはデフォルトです。
			X'01'	1つのみ: コマンドが発行された場合、キュー内の1つのメッセージを配信します。
			X'02'	自動: キュー中のすべてのメッセージを処理しきる。その後、メッセージがキューされると、メッセージを自動的に送達する。
			X'04'	自動的に1つ: キュー内にメッセージがある場合、自動的に1つのメッセージを配信します。メッセージは、既にキュー内にある場合も、後で送信される場合もあります。メッセージを送信すると、このオプションはNo-Autoに再設定されます。
3	1	コールアウト・モード	X'80'	RESUME TPIPE 呼び出しは、同期コールアウト・メッセージだけを取り出します。
			X'40'	RESUME TPIPE 呼び出しは、同期コールアウト・メッセージと非同期メッセージの両方を取り出します。
			X'20'	RESUME TPIPE 呼び出しは、制御データをサポートします。
			X'10'	RESUME TPIPE 呼び出しは、ネットワーク・セキュリティ資格情報をサポートします。
4	8	RESUME TPIPE トークン		<p>OTMA クライアント (IMS Connect など) は、RESUME TPIPE 要求を一意的に識別するために、RESUME TPIPE トークンを生成します。</p> <p>T パイプが IMS Connect からの複数のアクティブ RESUME TPIPE 要求をサポートする場合 (MULTIRTP=Y)、<b>/DISPLAY TMEMBER(tmename)</b> TPIPE(tpipename) コマンドを発行することにより、RESUME TPIPE トークンとそれに関連付けられている代替クライアント ID を表示できます。</p> <p>このトークンは、クライアントが RESUME TPIPE 要求を取り消すときにも使用されます。</p>

## T パイプの保留キューからの出力再開要求の取り消し

OTMA の「T パイプの保留キューからの出力再開要求取り消し」プロトコル・コマンドは、前にクライアントがサブミットした RESUME TPIPE 要求を取り消すために、クライアントによって送信されます。

OTMA は、RESUME TPIPE を取り消すための要求を受け取ると、RESUME TPIPE トークンを使用して、取り消す要求を見つけます。見つかった場合、RESUME TPIPE 要求は破棄されます。

表 168. 「T パイプの保留キューからの出力再開要求の取り消し」プロトコル・コマンドのフォーマット

バイト	長さ	内容	説明
0	2	長さ	状態データ・セクションの長さ。
2	2	予約済み	予約フィールド

表 168. 「Tパイプの保留キューからの出力再開要求の取り消し」プロトコル・コマンドのフォーマット (続き)

バイト	長さ	内容	説明
4	8	トークン	取り消す RESUME TPIPE 要求の RESUME TPIPE トークン。

## Tパイプの保留キュー上にメッセージがない

OTMA サーバーは、現行の RESUME TPIPE 要求用のメッセージまたは応答が T パイプの保留キューに入っていないことを OTMA クライアントに通知するために、「Tパイプの保留キュー上にメッセージがない」プロトコル・コマンドを送信します。

OTMA は、以下のいずれかのイベントが発生した場合に、このプロトコル・コマンドを発行します。

- 「Resume Output for the Special Queue for Tpipe」のオプションが TMAMCRHQ\_ONE であり、クライアント用の IMS メッセージがない場合。
- 「Resume Output for the Special Queue for Tpipe」のオプションが TMAMCRHQ\_NOAUTO であり、クライアント用の IMS メッセージがない場合。
- 「Resume Output for the Special Queue for Tpipe」のオプションが TMAMCRHQ\_NOAUTO であり、OTMA が OTMA キュー内の既存のメッセージをすべてフラッシュした場合。

表 169. 「Tパイプの保留キュー上にメッセージがない」プロトコル・コマンドのフォーマット

バイト	長さ	内容	説明
0	2	長さ	状態データ・セクションの長さ。
2	2	予約済み	予約フィールド
4	8	トークン	メッセージがない RESUME TPIPE 要求の RESUME TPIPE トークン。

## セキュリティ・データ・セクション

セキュリティ・データ・セクションは、各トランザクションあるいはコマンドで必須で、OTMA プロトコル・コマンドではオプションです。

OTMA メッセージ接頭語のセキュリティ・データ部分は、DFSMSG マクロの TMAMSEC DSECT によってマップされます。

以下の表は、メッセージ接頭語のセキュリティ・データ・セクションの内容を要約したものです。この要約には、必要に応じて、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 170. セキュリティ・データ・フィールドの内容

バイト	長さ	内容	値	説明
0	2	長さ		長さフィールドを含む、セキュリティ・データ・セクションの長さ。

表 170. セキュリティー・データ・フィールドの内容 (続き)

バイト	長さ	内容	値	説明
2	1	セキュリティ・フラグ	N	セキュリティなし: RACF 検査を行わない。 ユーザー ID とパスワードは、すでに検査済みであることを想定しています。
			C	検査: RACF は、トランザクションとコマンドを検査する。 トランザクションとコマンドの許可 RACCHECK が行われます (TCLASS および CCLASS)。
			F	フル: RACF は、トランザクション、コマンド、および領域を検査します。 トランザクション、IMS コマンド、および MPP 領域の許可 RACCHECK が行われます。
3	1	予約済み		
	1	Utoken の長さ		UTOKEN の長さ と UTOKEN タイプの長さを加えたもの。 長さには、長さフィールド自身を含みません。
	1	UTOKEN タイプ	X'00'	データ・タイプが続く。
	*	UTOKEN		ユーザー・トークン。 可変長で、1 から 80 バイトです。
	1	ユーザー ID の長さ		ユーザー ID の長さ と ユーザー ID タイプの長さを加えたもの。 長さには、長さフィールド自身を含みません。
	1	ユーザー ID のタイプ	X'02'	データ・タイプが続く。
	*	ユーザー ID		ユーザー ID。 可変長で、1 バイトから 8 バイトです。
	u	プロファイルの長さ		プロファイルの長さ と プロファイル・タイプの長さを加えたもの。 長さには、長さフィールド自身を含みません。
	1	プロファイル・タイプ	X'03'	データ・タイプが続く。
	*	プロファイル		SAF プロファイル。 可変長で、1 バイトから 8 バイトです。
	1	ネットワーク・ユーザー ID の長さ		ネットワーク・ユーザー ID の長さ に ネットワーク・ユーザー ID タイプの長さである 1 バイトを加えたもの。 長さには、この長さフィールド自身を含みません。
	1	ネットワーク・ユーザー ID タイプ	X'04'	データ・タイプが続く。



表 170. セキュリティー・データ・フィールドの内容 (続き)

バイト	長さ	内容	値	説明
	*	ネットワーク・ユーザー ID		最大 246 バイトとすることができる分散ユーザー ID。IMS TM リソース・アダプターを使用するお客様の場合、これは X.500 シリーズの標準の識別名 (DN) です。
	1	ネットワーク・セッション ID の長さ		ネットワーク・セッション ID の長さにネットワーク・セッション ID タイプの長さである 1 バイトを加えたもの。  長さには、この長さフィールド自身を含みません。
	1	ネットワーク・セッション ID タイプ	X'05'	データ・タイプが続く。
	*	ネットワーク・セッション ID		分散ユーザーのネットワーク・セッション ID。可変長で、1 から 254 バイトです。IMS TM リソース・アダプターを使用するお客様の場合、これは、ドメイン・ネーム、レルム、レジストリー名のいずれかです。

#### 関連資料

916 ページの『OTMA セキュリティー・データ・フィールドの説明』

以下に、メッセージ接頭語のセキュリティ・データ・セクションの内容について、さらに詳しい情報を提供します。

## OTMA セキュリティー・データ・フィールドの説明

以下に、メッセージ接頭語のセキュリティ・データ・セクションの内容について、さらに詳しい情報を提供します。

### 長さ

長さフィールドを含む、メッセージ接頭語のセキュリティ・データ・セクションの長さを指定します。

### セキュリティ・フラグ

行われるセキュリティ検査のタイプを指定します。ユーザー ID とパスワードは、すでに検査済みであることを想定しています。

### セキュリティなし

セキュリティ検査は行われなことを指定します。

### 検査

トランザクションとコマンドのセキュリティ検査が行われることを指定します。

### フル

トランザクション、コマンド、および MPP 領域のセキュリティ検査が行われることを指定します。

### 予約済み

予約フィールドの後は、以下の 3 つのフィールドが任意の順序で現れたり、省略することができます。各フィールドは以下の構造を持っています。

- 長さフィールド
- フィールド・タイプ
- データ・フィールド

長さフィールドは、長さの計算では計算されません。ユーザー ID あるいはプロファイルの実際の長さは、各フィールドの長さに指定した値より小さくはいけません。

### UTOKEN の長さ

ユーザー・トークンの長さとしてユーザー・トークン・タイプの長さを加えたものを指定します。

## UTOKEN タイプ

このフィールドがユーザー・トークンを含むことを指定します。

## UTOKEN

ユーザー・トークンを指定します。ユーザー ID とプロファイルが、ユーザー・トークンを作成するために使用されます。ユーザー・トークンは、IMS 従属領域において渡されます。

クライアントが既に RACF を呼び出していた場合は、フィールド・タイプ X'00' で UTOKEN を渡す必要があります (RACF は再度呼び出すことができないため)。

## ユーザー ID の長さ

ユーザー ID の長さとユーザー ID タイプの長さを加えたものを指定します。

## ユーザー ID タイプ

このフィールドがユーザー ID を含むことを指定します。

## ユーザー ID

実際のユーザー ID を指定します。

## プロファイルの長さ

プロファイルの長さとプロファイル・タイプの長さを加えたものを指定します。

## プロファイル・タイプ

このフィールドがプロファイルを含むことを指定します。

## プロファイル

システム許可機能 (SAF) プロファイルを指定します。RACF の場合、これはグループ名です。

## ネットワーク・ユーザー ID の長さ

ネットワーク・ユーザー ID の長さにネットワーク・ユーザー ID タイプの長さである 1 バイトを加えたものを指定します。長さには、この長さフィールド自身を含みません。

## ネットワーク・ユーザー ID タイプ

直後のデータがネットワーク・ユーザー ID であることを示すには、X'04' を指定します。

## ネットワーク・ユーザー ID

最大 246 バイトとすることができる分散ユーザー ID を指定します。IMS TM リソース・アダプターを使用するお客様の場合、これは X.500 シリーズの標準の識別名 (DN) です。

## ネットワーク・セッション ID の長さ

ネットワーク・セッション ID の長さにネットワーク・セッション ID タイプの長さである 1 バイトを加えたものを指定します。長さには、この長さフィールド自身を含みません。

## ネットワーク・セッション ID タイプ

直後のデータがネットワーク・セッション ID であることを示すには、X'05' を指定します。

## ネットワーク・セッション ID

分散ユーザーのネットワーク・セッション ID を指定します。これは、最大 254 バイトとすることができます。IMS TM リソース・アダプターを使用するお客様の場合、これは、ドメイン・ネーム、レルム、レジストリー名のいずれかです。

## ユーザー・データ・セクション

OTMA メッセージ接頭語のユーザー・データ・セクションは可変長で、セキュリティ・データ・セクションの後に続きます。どんなデータでも含むことができます。

OTMA メッセージ接頭語のユーザー・データ部分は、DFSMSG マクロの TMAMUSR DSECT によってマップされます。

以下の表は、メッセージ接頭語のユーザー・データ・セクションの内容を要約したものです。この要約には、必要に応じて、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 171. ユーザー・データ・フィールドの内容

バイト	長さ	内容	説明
0	2	長さ	長さフィールドを含む、ユーザー・データ・セクションの長さ。

表 171. ユーザー・データ・フィールドの内容 (続き)

バイト	長さ	内容	説明
2	*	ユーザー・データ	ユーザー・データ。 オプションで可変長です。

#### 関連資料

918 ページの『OTMA ユーザー・データ・フィールドの説明』

以下に、メッセージ接頭語のユーザー・データ・セクションの内容について、さらに詳しい情報を提供します。

## OTMA ユーザー・データ・フィールドの説明

以下に、メッセージ接頭語のユーザー・データ・セクションの内容について、さらに詳しい情報を提供します。

#### 長さ

長さフィールドを含む、メッセージ接頭語のユーザー・データ・セクションの長さを指定します。ユーザー・データの最大長は、1024 バイトです。

#### ユーザー・データ

オプションのユーザー・データを指定します。このデータは、クライアントが管理し、DFSYDRU0 出力ルーチンを使用して、作成および更新できます。サーバーは、任意の出力メッセージの最初のセグメントとして、クライアントに未変更でこのセクションを戻します。

#### 転送 T パイプ

OTMA 保留キュー対応クライアント (IMS Connect など) から送信される入力トランザクション用の転送 T パイプ名を指定します。このオプション・フィールドは、ユーザー・データ接頭部の先頭からオフセット X'5C' で開始し、次の 2 つのシナリオで使用されます。

- OTMA が CMO IOPCB 出力メッセージをクライアントに送信できないとき、転送 T パイプ名が指定されていれば、それを使用して OTMA は出力を転送します。
- Send-Only の入力トランザクションが転送 T パイプ名を指定していると、Send-Only トランザクションの結果できる IOPCB 出力メッセージは、転送 T パイプのキューに入れられます。

## アプリケーション・データ・セクション

OTMA メッセージのアプリケーション・データ・セクションは可変長で、メッセージ接頭語のユーザー・データ・セクションの後に続きます。

アプリケーション・データ・セクションには、IMS コマンドとトランザクションを組み込みます。このセクション内のデータは、受信側 (サーバーあるいはクライアント) によっては変更されず、直接サーバー・アプリケーション・プログラムあるいはクライアント・アプリケーション・プログラムに伝送されます。

OTMA メッセージ接頭語のアプリケーション・データ部分は、DFSYMSG マクロの TMAMAPP DSECT によってマップされます。

以下の表は、メッセージ接頭語のアプリケーション・データ・セクションの内容を要約したものです。この要約には、必要に応じて、バイト、長さ、内容、16 進値、および意味が含まれ、使用法のコメントも含まれています。

表 172. アプリケーション・データ

バイト	長さ	内容	説明
0	2	長さ	アプリケーション・データ・セクションの長さ。 長さには、長さフィールド自身を含みます。最大長は 32 KB (32767 バイト) です。
2	2	ZZ	アプリケーション・データの IMS ZZ フィールド。

表 172. アプリケーション・データ (続き)

バイト	長さ	内容	説明
4	*	アプリケーション・データ	<p>オプションのアプリケーション・データ。</p> <p>複数の送信要求がサーバー出力セグメントに対して要求されることがあります。クライアント・トランザクションの場合、LLZZ に続いて、データ域の最初の 8 バイトにトランザクション・コードを指定します。MULTSEG を指定したトランザクションの場合、標準の IMS LLZZ フォーマットがセグメントごとに要求されます。トランザクション・コードは、最初のセグメントにのみ必要です。</p> <p>可変長です。アプリケーション・データの最大長は 32 KB-4 です。</p>

## サンプル OTMA メッセージ

以下の 3 つのサンプル OTMA メッセージは、OTMA メッセージが、メッセージ接頭語の部分を含み、完全に構成された場合にどのようなようになるかを示します。各例は、必ずしも相互に関連しているとは限りません。

### OTMA クライアント・ビッド・メッセージ

以下の図は、OTMA クライアント・ビッド・メッセージを示します。メッセージ接頭語の、状態データ・セクションとセキュリティー・データ・セクションを加えた全長は、X'8C' バイトです。

MESSAGE CONTROL INFORMATION:

```
01102000 04004040 40404040 4040A0C0 |..... ff{|
00000000 00000000 00000000 00000400 |.....|
```

STATE DATA + SECURITY DATA:

```
0036C3D3 C9C5D5E3 F1404040 40404040 |..CLIENT1 |
40400100 00010003 00020100 00010003 | .....|
0001C4C6 E2E8C4D9 E4F02000 00007FFF |..DFSYDRU0...".|
FFFF0000 00650056 C3525100 50018059 |.....C...&...|
15569555 55555555 55555555 B7B686B0 |..n.....%&f[|
81A61515 1B1B1B1B 1B1B1B1B B7B686B0 |aw.....%&f[|
81A61515 55555555 55555555 8C918CA4 |aw....._]_u|
15151515 55555555 55555555 09151515 |.....|
15151515 15151515 15151515 |.....|
```

### OTMA トランザクション・メッセージ

以下の図は、OTMA トランザクション・メッセージです。メッセージ接頭語の、状態データ、セキュリティー・データ、およびアプリケーション・データの各セクションの全長は、X'D6' バイトです。

MESSAGE CONTROL INFORMATION:

```
01402000 0000E3D7 C9D7C5F1 4040A0D0 |. ....TPIPE1 ff{|
00000001 00000000 00000001 00010000 |.....|
```

STATE DATA + SECURITY DATA + APPLICATION DATA:

```
00480020 0100E3C5 E2E3D4C1 D7400000 |.....TESTMAP ..|
00000000 00000000 00000000 0000C9D4 |.....IM|
E2F0F0F0 F0F10000 00000000 00004040 |S00001.....|
40404040 40404040 40404040 40404040 |.....|
40404040 40400000 0056C652 51005001 | .....F...&.|
80465551 95555555 55555555 55555555 |.....n.....|
55555555 55555555 55555555 55555555 |.....|
55555555 55555555 55555555 555586A3 |.....ft|
A781B0B7 A4155555 55555555 5555B1B7 |xa[%u.....&%|
8CB6A5A5 A415B7BD B7A41515 15150038 |%&vvu.%"%u.....|
0000C1D7 D6D3F1F8 4040E2C1 E840C8C5 |..APOL18 SAY HE|
D3D3D640 40404040 40404040 40404040 |LL0|
40404040 40404040 40404040 40404040 |.....|
```

**OTMA 応答メッセージ**

以下の図は、OTMA 応答メッセージを示しています。メッセージ接頭語の、状態データ、セキュリティ・データ、およびアプリケーション・データの各セクションの全長は、X'EE' バイトです。

## MESSAGE CONTROL INFORMATION:

```
01A08000 0000E3D7 C9D7C5F1 404080D0 |.ff....TPIPE1 .}|
00000001 00000000 00000001 00010000 |.....|
```

## STATE DATA + SECURITY DATA + APPLICATION DATA:

```
00480020 0100E3C5 E2E3D4C1 D740AB7F |.....TESTMAP %"|
28EB9FAD 9A024040 40404040 4040C9D4 |.. Y.. IM|
E2F0F0F0 F0F10000 00000000 00004040 |S00001.....|
40404040 40404040 40404040 40404040 |
40404040 40400000 0056C652 51005001 |.....F...&|
80465551 95555555 55555555 55555555 |....n.....|
55555555 55555555 55555555 55555555 |.....|
55555555 55555555 55555555 555586A3 |.....ft|
A781B0B7 A4155555 55555555 5555B1B7 |xa[%u.....&%|
8CB6A5A5 A415B7BD B7A41515 15150050 |%&vvu.%"u....&|
0300D6E4 E3E2C5C7 40D5D67E F0F0F0F0 |..OUTSEG NO=0000|
F140E2D7 C5C3C9C6 C9C5C440 E2C5C7E2 |1 SPECIFIED SEGS|
C9E9C57E F0F0F0F8 F06B40E2 C5C7D5D6 |IZE=00080, SEGN0|
7EF0F0F0 F0F34040 40404040 40F67EC3 |=00003 6=C|
D6D340F6 F04040F7 7EC3D6D3 40F7 |OL 60 7=COL 7..|
```

---

## 第 47 章 OTMA 呼び出し可能インターフェース

IMS OTMA 呼び出し可能インターフェース (C/I) が提供する高水準インターフェースにより、他の z/OS サブシステム上のアプリケーション・プログラムから、OTMA を介して IMS アプリケーションにアクセスすることができます。

OTMA C/I API は、C/C++ プログラムで使用可能な API 呼び出しで構成されています。API 呼び出しは、IMS/OTMA z/OS システム間カップリング・ファシリティ (XCF) グループへの結合、IMS への接続、通信セッションの割り当て、IMS トランザクション/コマンドの送信、IMS からの出力の受信、通信セッションのクローズ、および XCF グループからの離脱のために使用されます。

OTMA C/I API 呼び出しおよびサンプルの OTMA C/I アプリケーション・プログラムについては、「IMS V15 システム・プログラミング API」に説明があります。OTMA C/I が返すコードについては、「IMS V15 メッセージおよびコード 第 4 巻: IMS コンポーネント・コード」で説明されています。

以下の図は、OTMA C/I の概要を表します。サンプル z/OS 環境として、サンプル C および C++ API 呼び出し (OTMA\_OPEN など) が左から右に示されています。API 呼び出しはオブジェクト・スタブ、SVC インターフェース・ルーチン、API (例えば、DFSYOPEN) をパススルーし、最終的には、XCF グループを通り IMS OTMA に渡ります。

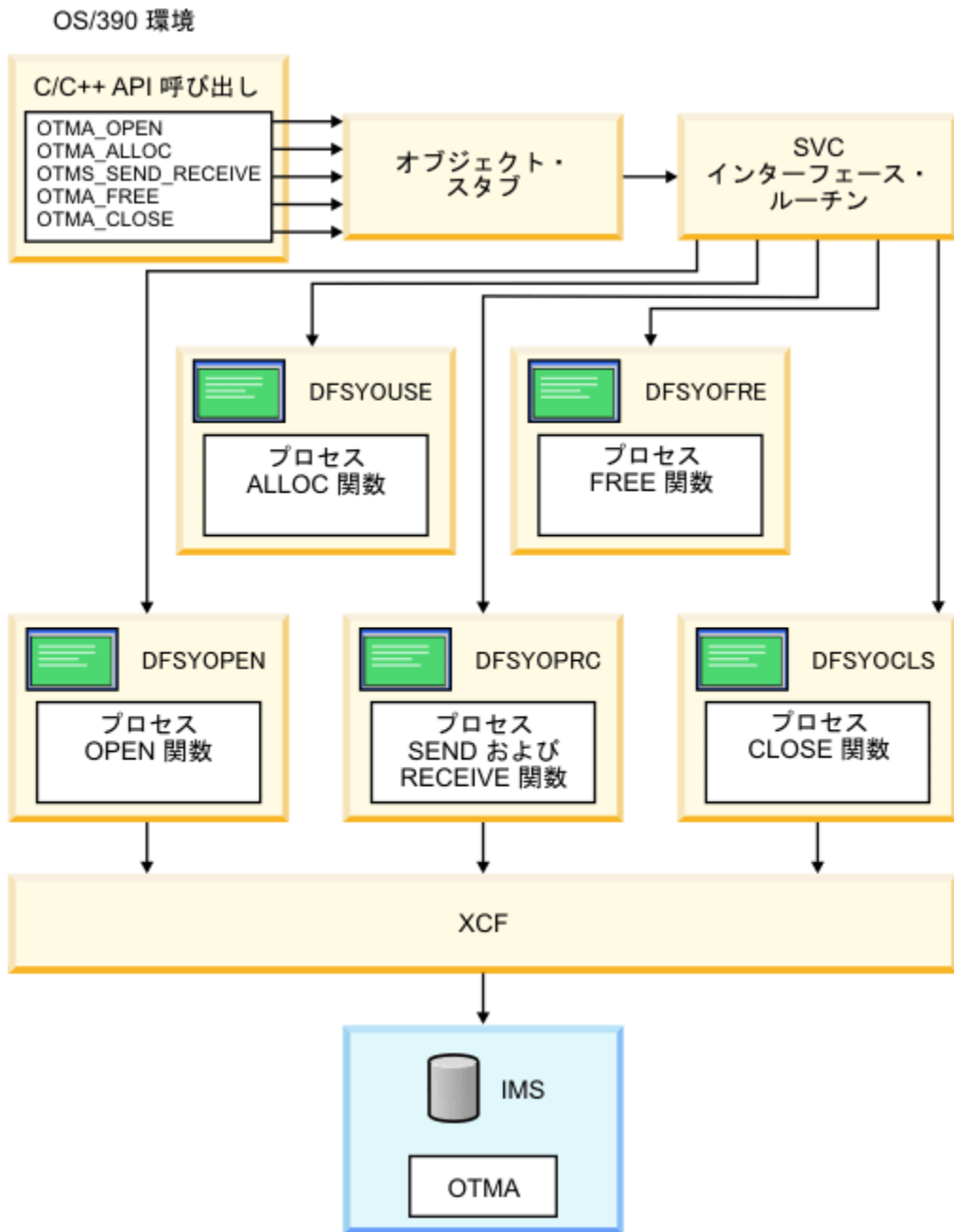


図 156. OTMA 呼び出し可能インターフェースの概要

API 呼び出しを起動するプログラムは、問題プログラム状態あるいは監視プログラム状態の許可ライブラリまたは無許可ライブラリから稼働することができます。C ヘッダー・ファイルである DFSYCO が、API 呼び出しを定義するために提供されます。ロード・モジュールの DFSYCRET は、各 API 呼び出しに対するエントリー・ポイントを含み、アプリケーション・プログラムにリンクされます。OTMA C/I は、API 呼び出しを処理するために BPE SVC サービスを使用します。

**利点:** OTMA C/I の主要な利点は、使いやすさです。

OTMA C/I を使用するその他の理由として、以下の理由があります。

- OTMA および XCF の詳細を抽出する
- IMS トランザクションおよびコマンドをサブミットする
- 他の z/OS サブシステムで実行しているプログラムを複数の IMS システムに接続することを可能にする
- 許可あるいは無許可のライブラリから API を呼び出す
- すべての IMS OTMA リリースへ接続する



## OTMA C/I の初期設定

OTMA C/I は、独立型プログラムである DFSYSVIO を提供します。これは、z/OS の IPL の実行後または OTMA C/I への保守の適用後に OTMA C/I を初期設定する場合に実行する必要があります。

### このタスクについて

注: いずれかの OTMA C/I クライアントがアクティブな場合は、DFSYSVIO プログラムを実行しないでください。

DFSYSVIO が開始された時にいずれかの OTMA C/I クライアントがアクティブな場合、DFSYSVIO は DFS3948E と DFS3950A を発行します。OTMA C/I クライアントがまだアクティブな時に SVC の初期設定を実行すると、XCF エラーが起こることがあります。

アクティブ・クライアントは、z/OS コマンド **D GRS,RES=(DFSOTMA,\*)** を発行することによって識別できます。

DFSYSVIO プログラムを実行した時にいずれかのクライアントがアクティブな場合は、それらのクライアントを終了し、DFS3950A に「RETRY」と応答して、再び SVC の初期設定を試行してください。

クライアントを終了できない場合は、DFS3905A に「CANCEL」と応答して、SVC を初期設定せずにユーティリティを停止してください。

クライアントが孤立しており、再度使用されない場合は、DFS3905A に「BYPASS」と応答して、SVC の初期設定を続行してください。このオプションの使用には、十分な注意が必要です。それらの既存クライアントが再度 OTMA C/I を使用しようとした場合、XCF エラーが発生する可能性があるためです。

DFSYSVIO は、OTMA C/I モジュールの 1 つである DFSYSVCO を実行します。DFSYSVCO は、SVC サービスをロードおよび登録しますが、これは、アプリケーション・プログラムがアクセスするのと同じ z/OS イメージ上で稼働する許可アドレス・スペースで行われます。OTMA C/I を保守した後に OTMA C/I を初期設定すると、OTMA C/I クライアントによって OTMA C/I 呼び出しが実行されたとき、その呼び出しの処理に必要な DFSYSVCO ロード・モジュール内のすべての OTMA C/I モジュールが、DFSYSVCO と同じレベルにあることが保証されます。

OTMA C/I は 1 つだけ存在し、しかも OTMA C/I は IMS 内でなく SVC を通じてクライアント・アドレス・スペース内で実行されるため、DFSYSVIO はクライアントの接続先の IMS システムに関係なく、システム内のすべての OTMA C/I クライアントに対して OTMA C/I をリフレッシュします。

読者は、OTMA 呼び出し可能インターフェース 初期設定プログラム用の z/OS プログラム特性テーブル (PPT) にエントリーを追加する必要があります。これを行うステップは、以下のとおりです。

### 手順

1. SYS1.PARMLIB データ・セットの SCHEDxx メンバーを編集します。
2. 以下のエントリーを SCHEDxx メンバーに追加します。

```
PPT PGMNAME(DFSYSVIO) /* PROGRAM NAME = DFSYSVIO */
CANCEL /* PROGRAM CAN BE CANCELED */
KEY(7) /* PROTECT KEY ASSIGNED IS 7 */
SWAP /* PROGRAM IS SWAPPABLE */
NOPRIV /* PROGRAM IS NOT PRIVILEGED */
DSI /* REQUIRES DATA SET INTEGRITY */
PASS /* CANNOT BYPASS PASSWORD PROTECTION */
SYST /* PROGRAM IS A SYSTEM TASK */
AFF(NONE) /* NO CPU AFFINITY */
```

3. 以下のアクションを取って、SCHEDxx の変更を有効にします。

- z/OS システムを再 IPL します。
- または
- MVS SET SCH= コマンドを発行します。

**関連資料:** プログラム・プロパティ・テーブルの更新に関する追加資料については、「z/OS MVS 初期設定およびチューニング解説書」を参照してください。

## タスクの結果

以下に、DFSYSVIO を稼働するための JCL プロシージャのサンプルを挙げます。

```
//OTMAINIT PROC RGN=3000K,SOUT=A
//*
//IEFPROC EXEC PGM=DFSYSVIO,
// REGION=&RGN
//*
//STEPLIB DD DISP=SHR,UNIT=SYSDA,
// DSN=IMSVS.SDFSRESL
//*
//SYSPRINT DD SYSOUT=&SOUT
//SYSUDUMP DD SYSOUT=&SOUT
```

OTMA C/I をインストールし、初期設定した後、これを使用して任意の IMS OTMA リリースに接続できます。

## OTMA C/I のセキュリティ

z/OS システム間カップリング・ファシリティグループを無許可の呼び出し元から保護するには、IMSXCF.OTMACI を使用します。これは RACF リソースで、OTMA C/I 用に RACF ファシリティ・クラスに定義されています。

### このタスクについて

RACF リソースを定義すると、OTMA C/I が XCF JOIN を実行する前に、RACF RACHECK が実行されます。このメソッドは、XCF、XCF グループ、およびそのメンバーへのアクセスを保護します。この RACF 検査は、無許可呼び出し元が、OTMA C/I を使用しているときのみ行われます。

## OTMA C/I の制約事項

OTMA 呼び出し可能インターフェース (C/I) には、特定の制約事項が適用されます。

制約事項を以下に示します。

- C/I は、呼び出し可能になる前に、z/OS 環境にインストールしておく必要があります。C/I が未インストールで呼び出された場合、otma\_create または otma\_open が発行されたときに、F92 異常終了が発生します。C/I が正常にインストールされていない場合は、DFS3911E エラー・メッセージが発生します。
- C および C++ 以外のアプリケーション・プログラム言語は、現在、OTMA C/I でサポートされていません。
- すべての OTMA 呼び出しは、otma\_open 呼び出しと同じ状態 (PSW キー、監視プログラム状態か問題プログラム状態か、許可か無許可か) にする必要があります。例えば、otma\_open 呼び出しを行ったときに許可されていた場合は、すべての後続呼び出しに対して許可されている必要があります。
- IMS OTMA の再同期フィーチャーはサポートされません。
- IMS コマンドの /SECURE OTMA PROFILE は現在サポートされていません。

## CM1 トランザクションの otma\_send\_receive API 呼び出しの後に OTMA C/I セッションがタイムアウトになる

OTMA C/I には、otma\_send\_receive API 呼び出し用のタイムアウト機能がなく、OTMA メッセージ・レベルのトランザクション有効期限機能をサポートしませんが、タイムアウト機能を実装することができます。

### このタスクについて

OTMA C/I クライアント・セッション用のタイムアウト機能を実装するには、以下のいずれかの方法を使用します。

## 手順

- OTMA C/I クライアントが実行する IMS トランザクションを定義するときに、TRANSACT システム定義マクロに EXPRTIME パラメーターをコーディングします。
- OTMA C/I クライアントにタイムアウト機能を含めます。
- EXPRTIME パラメーターをコーディングし、OTMA C/I クライアントにタイムアウト機能を含めます。

## タスクの結果

**推奨事項:** OTMA C/I クライアントにタイムアウト機能を含める代わりに、EXPRTIME パラメーターを使用して、OTMA C/I を介してサブミットされるトランザクションのタイムアウト機能を実装してください。OTMA C/I クライアントにタイムアウト機能を実装した場合、OTMA が IMS から応答を受信した後であるが待機 ECB を通知する前に、OTMA C/I クライアントがセッション・ストレージを解放すると、稀にはあるが、z/OS X'0C4' 異常終了が発生する可能性があります。

OTMA C/I クライアントにタイムアウト機能を含めた場合は、以下の操作によって、z/OS X'0C4' 異常終了の可能性を低くすることができます。

- OTMA C/I クライアントのタイムアウト機能をコーディングするときに、OTMA C/I クライアントが otma\_free 呼び出した後で、かつセッション・ストレージを解放する前に、処理の遅延を含めます。
- OTMA C/I クライアントが実行する IMS トランザクションの TRANSACT システム定義マクロの EXPRTIME パラメーターに指定するタイムアウト値を小さくします。

## 関連資料

[TRANSACT マクロ \(システム定義\)](#)



## 第 48 章 OTMA 設計済みトランザクション属性

IMS /DISPLAY TRANSACTION コマンドを OTMA から発行すると、出力は、OTMA メッセージの形式で、メッセージ接頭語のアプリケーション・データ・セクションに入れられて、クライアントに戻されます。

以下の表に、ある特定のトランザクションの属性セグメントを示し、設計済みコマンド出力の構文を説明します。ここでの説明には、バイト、長さ、内容、16 進値、および意味が含まれ、必要に応じて使用法のコメントも含まれています。

表 173. トランザクション属性セグメント

バイト	長さ	内容	値	意味	コメント
0	2	長さ		トランザクション属性セグメントの長さ (LL)。	この長さには、長さフィールド自身を含みます。
2	2	<b>ZZ</b>			
4	8	トランザクション・コード		8 バイトの IMS トランザクション・コード。	
12	1	トランザクション・タイプ・フラグ <b>1</b>			フラグ 1 の値は相互に排他的です。
		有効	X'00'	有効な OTMA トランザクション。	
		CPIC	X'04'	CPI-C (APPC) トランザクション。	
		FPX	X'08'	高速機能専用トランザクション。	
		FPP	X'0C'	高速機能利用可能トランザクション。	
		MSC	X'10'	MSC リモート・トランザクション。	
		無効な構文	X'FE'	構文エラー。	データ域にはエラー・テキストを含みます。
		無効	X'FF'	トランザクションが見つからないか、無効である。	
13	1	トランザクション・タイプ・フラグ <b>2</b>			フラグ 2 の値は、相互に排他的ではありません。
		応答、	X'80'	IMS 応答モード・トランザクション。	
		会話	X'40'	IMS 会話型トランザクション。	
		更新	X'20'	トランザクションは更新可能性を持つ。	
		リカバリー不能	X'10'	トランザクションをリカバリー不能として定義する。	
		複数セグメント	X'08'	トランザクションは複数セグメントを持つ。	
		英大文字	X'04'	英大文字変換要求。	
14	1	トランザクションの状況			指示した値は、相互に排他的ではありません。
		STOP	X'80'	トランザクション入力のキューイングが停止した。	以下の IMS コマンドの 1 つがトランザクション用に出されました。 • /STOP • /PURGE
		OLC	X'40'	トランザクション入力のキューイングが停止した。	以下の IMS コマンドの 1 つがトランザクション用に出されました。 • /MOD PREPARE • /MOD COMMIT
		NOSCH	X'20'	IMS スケジューリングが停止した。	以下の IMS コマンドの 1 つがトランザクション用に出されました。 • /PSTOP • /LOCK
15	1	予約済み			
16	2	長さ		存在する場合は、エラー・テキストの長さ (LL)。	エラー・テキストがある場合は、メッセージの後続セクションのすべてを置換します。

表 173. トランザクション属性セグメント (続き)

バイト	長さ	内容	値	意味	コメント
18	*	エラー・テキスト		エラー・メッセージのテキスト。	可変長です。エラー・テスト・セクションは、トランザクション・タイプ・フラグ1が無効な構文(X'FE')にセットされている場合のみ適用可能です。
16	8	PSB 名		IMS PSB 名。	エラー・テキストがない場合のみ現れます。
24	1	クラス		IMS スケジューリング用の SMB メッセージ・クラス。	
25	1	現行優先順位		現行 SMB 優先順位。	
26	1	通常優先順位		通常の SMB 優先順位。	
27	1	限界優先順位		SMB 限界優先順位。	
28	2	エンキュー・カウント		エンキューされたメッセージの数。	
30	2	デキュー・カウント		デキューされたメッセージの数。	
32	2	エンキュー限界		エンキュー限界カウント。	
34	2	処理限界カウント		処理限界カウント。	
36	2	最大出力セグメント長		最大出力セグメントの長さ。	
38	2	メッセージ・セグメントの出力限界		メッセージ・セグメントの出力限界。	
40	2	並列処理の限界		TRANSACTION ステートメントからの PARLIM 値。	
42	1	領域カウント		トランザクションが現在スケジュールされている領域の数。	

---

## 第 11 部 SLU P および金融機関通信

これらのトピックでは、SLU P と金融機関通信の概要、SLU P および金融機関通信に使用される IMS 機能、SLU P および金融機関通信でのネットワーク操作、および SLU P ネットワーク・プロトコルなどを含めて、SLU P および金融機関通信について詳しく説明します。





## 第 49 章 SLU P と金融機関通信の概要

以下のトピックでは、SLU P と金融機関通信システムの管理上の概要、および IMS がこのアーキテクチャをどのようにしてインプリメントするかを説明します。

IBM 金融機関通信システム用の ユーザー作成アプリケーション・プログラムは、IMS の下で 2 つの異なる方法で操作するように定義することができます。その方法とは、IBM 3600/4700 金融機関通信コントローラーとして操作するかまたは、2 次論理装置タイプ P (SLU P) として操作する方法です。このシステムは、それぞれ TYPE マクロ・ステートメントで、UNITYPE=FINANCE と UNITYPE=SLUTYPEP で表します。

### 定義:

- 以下のトピックでは、両方のシステム・タイプに適用されるサポートを説明する場合に *SLU P* という用語を使用しています。
- これらのシステムを区別する必要がある場合は、金融機関 および *SLUTYPEP* という用語を使用しています。

SLU P と金融機関システムの主な相違点は、プログラマブル制御装置のワークステーションの MFS サポートのレベルの違いです。

- 金融機関システムでは、ワークステーションは、ディスプレイ、プリンター、通帳記帳装置、および 4730 システムなどの ATM (自動預金支払機) として識別されます。MFS は、画面送り、プリンター・ページのフォーマット設定、ATM などについて 詳細サポートを提供できます。
- SLU P では、コントローラーの各 LU はプログラム名を持ち、その固有の装置特性については、IMS は関知しません。プログラマブル・コントローラーは、装置制御とフォーマット設定をします。MFS 管理担当者とリモート・コントローラーのプログラム管理担当者は、交換メッセージのデータ・フィールド構造を設定する必要があります。MFS は、入力構造を受け入れ、それを IMS アプリケーション・プログラムに合うよう再調整します。MFS は出力時に、アプリケーション・プログラムが出したデータを受け取り、そのデータをコントローラーへ伝送するための正しい形式に変換します。この MFS の機能を分散表示管理 (DPM) といいます。

### 関連タスク

111 ページの『ETO、3600/金融機関、および SLU P』

静的にシステム定義された 3600/金融機関と SLU P 端末には、**/SIGN** コマンドを使用するかログオン・ユーザー・データを使用することでサインオンできます。

## IMS-SLU P ネットワーク

IMS-SLU P ネットワークの 3 つの主要な要素は、リモート・プログラムを実行するコントローラーと端末、通信リンク、およびホスト IMS システムを実行する中央処理装置です。

この各エレメントには、システムが実行する全データ処理の一部を実行する プログラミングが含まれています。IMS はホスト・プロセッサに常駐していて、SLU P システムのコントローラーの アプリケーション・プログラムと通信リンクを介して通信します。コントローラーのアプリケーション・プログラムは、コントローラーに接続されている端末をモニターします。

IMS は、IBM 4701/4702 金融機関通信制御装置に接続されている 4700 端末をサポートします。IMS は、IBM 4730 自動取引装置 (PBM) を、SLU P として 37x5 に直接接続される場合と、SLU P システムの一部として 4701/4702 に接続される場合の両方をサポートします。

**関連資料:** 4700 端末のリストについては、「*IBM 4700 Finance Communication System: System Summary*」を参照してください。

IMS は、金融機関システムの一部として、IBM 3602 または 4701/4702 のどちらかに接続されている 3600 端末もサポートします。(IMS は 3600 シリーズ端末と 4700 シリーズ端末の区別はしません。)

SLU P プロトコルで IMS と接続する他の IBM 製品の例としては、シリーズ /1、IBM 3650、IBM 8100 などがあります。

この端末やワークステーションはすべて、TERMINAL マクロ・ステートメントの COMPT または ICOMPT のキーワードを使用すると、IMS に対して SLU P システムのコンポーネントとして定義することができます。

## システム構成

SLU P システムを構成する前に、計画しているシステムに必要な金融操作を特定する必要があります。この作業を行った後、論理装置やコンポーネントの構成を IMS に定義することができます。

### このタスクについて

IMS から見ると SLU P システムは、1 つのサブシステムであり、このシステムは 1 つ以上の論理装置から構成されています。

**定義:** システムを構成する場合、1 つ以上の論理装置を定義することができます。これらの論理装置は、各種装置、ストレージ、およびアプリケーション・プログラムで構成できます。金融機関通信システムの資料では、これらの論理装置が論理ワークステーションと呼ばれます。

## SLU P および金融機関ワークステーション

ワークステーションは、1 つ以上の端末装置で構成されます。各ワークステーションは、テラー操作 (預金、引き出しなど)、報告書の印刷または現金の支払いなどの特定のタイプの金融操作を実行します。

IMS に定義されたワークステーションは、必ずしも 4701/4702 でそのワークステーションに接続している実際の装置を反映する必要はありませんが、IMS システムから見たワークステーションを反映している必要があります。

構成例は、IMS ライブラリー (IMS.ADFSMAC; メンバー名=DFSCP360) にあります。

## システム・コントローラー・アプリケーション・プログラム

ワークステーションの操作は、SLU P システムのコントローラーに常駐するユーザー作成のアプリケーション・プログラムで制御されます。アプリケーション・プログラムは 1 つ以上のワークステーションを制御するように設計することができ、そのワークステーションが必要とする機能だけを実行する必要があります。

アプリケーション・プログラムで使用可能な機能には次のものがあります。

- ワークステーションに関連する端末の読み取りと書き込み
- 端末から受信したデータの編集と検証
- コントローラー・ディスクットの読み取りと書き込み
- ホスト・プロセッサの読み取りと書き込み
- ホスト・プロセッサから受信したデータの編集と検証
- ホスト・プロセッサまたは IMS が使用不能時のオフライン操作

## MFS および XRF を使用するコントローラー・アプリケーション・プログラムの作成

システムが MFS および XRF を使用する場合は、コントローラー・アプリケーション・プログラムを作成する際に、特定の処置を行う必要があります。

### このタスクについて

MFS を使用するシステム用のコントローラー・アプリケーション・プログラムを作成する場合は、以下のことを行います。

**注:** 4700 制御装置の TERMINAL マクロ命令に WL (警告行) オプションを指定しないでください。WL オプションを指定すると、LWRITE 命令で予期しない金融機関システム・データ例外エラーが起こることがあります。

## 手順

- 4700 制御装置の TERMINAL マクロに PS (ページ・サイズ) オプションを指定できるのは、IMS MFS DEV ステートメントに EJECT を指定した場合だけです。

PS を指定しない場合には、IMS からの EJECT は改行機能になります。ワークステーションで「ページ終了」条件が発生すると、予期しない金融機関システム・データ例外エラーが起こることがあります。IMS からの EJECT は行カウントをリセットし、その結果「ページ終了」条件になります。"ページ終了"条件になります。

4700 制御装置の TERMINAL マクロ命令に CFOLD オプションを指定する場合には、MSF 用に定義された形式が通帳の中央折り返しのためのスペーシングを起こさないようにする必要があります。SLU P システムは、この機能を自動的に行います。

- IMS MFS DFLD ステートメントが ATTR=YES を指定しているかどうか検査してください。指定している場合、下線 (X'6D') を印刷しようとする、次のいずれかが起こります。
  - 3610/3612 上でデータ・チェック  
OUTRTBL ステートメントが 3610/3612 装置に 128 文字セット以外を指定している場合は、EBCDIC 値 (X'6D') の印刷位置を指定してください。
  - 3618 装置上にブランクの印刷  
OUTRTBL ステートメントが 3618 装置に 48 文字セットを指定している場合は、EBCDIC 値 (X'6D') の印刷位置を指定してください。
- 1 つの IMS 出力伝送 (最後の文字以外) 内で ページ替え文字 (X'0C') を送信するため、MFS DEV ステートメント・オプション EJECT、BGNMXG、BGNPP、または ENDPP が使用されているかどうかを検査してください。  
使用されている場合は、ページ替え後に通帳記帳装置で「要介入」条件が発生します。この場合には、コントローラーのアプリケーション・プログラムが、複数の LWRITE 命令を通帳記帳装置に出して IMS 出力伝送を印刷できるようにしなければなりません。各 LWRITE に、ページ替え文字まで (ページ替え文字を含む) または伝送終了までのデータを入れなければなりません。

## 次のタスク

**関連資料:** コーディングおよびコントローラーのマクロの詳細については、「IBM 4700 Finance Communication System: System Summary」を参照してください。

## XRF システム用のコントローラー・アプリケーション・プログラムに関する考慮事項

SLU P システムが XRF 複合システムで稼働する場合には、コントローラー・アプリケーション・プログラムは、XRF のテークオーバー時に失われたメッセージを処理できるようにしなければなりません。

代替システムが制御を受け取った時に「経過中」であった多くのメッセージは、アプリケーション・プログラムまたは端末オペレーターからの処置がなくても、回復されます。この場合、テークオーバーは透過的でありこの事態を気にする必要はありません。

テークオーバーが透過的でない時には、メッセージ DFS3861I が出され、このメッセージは次の組み合わせのいずれかとして出されます。

- メッセージ DFS3861I のみー出力メッセージがリカバリー可能である場合、この未確定メッセージが出されます。リカバリー不能である場合、このメッセージは出されません。受信側が IMS から出力を受信中であり、受信側がメッセージを受信したかどうかを新しいアクティブ IMS が判別できない場合には、IMS は例外応答を要求しますが、受信側プログラムではそれを受信しませんでした。
- CANCEL コマンドの後にメッセージ DFS3861I が続くー複数セグメント出力が進行中 (チェーン内の最後はまだ送信されていません)。フライト中 (未了) メッセージがリカバリー可能である場合には、そのフライト中 (未了) メッセージを再度受信する必要があります。フライト中メッセージがリカバリー不能である場合には、次のフライト中メッセージを受信します (それがキューにある場合)。
- 例外応答の後にメッセージ DFS3861I が続くーユーザーの入力メッセージは失われました。最後の入力メッセージを再送信します。

# 金融機関から SLU P へのコントローラー・アプリケーション・プログラムの変換

IMS に金融機関として定義された論理装置が現在使用できるすべての機能 (SCAN/NOSCAN オプションおよび金融機関 MFS を除く) は、SLU P として定義された論理装置の機能と類似しています。

## このタスクについて

IMS は、2 次論理装置タイプ P に出力を送信する際に、出力で制御文字シーケンスをスキャンしません。したがって、NOSCAN オプションが暗黙指定されます。アプリケーション・プログラムが MFS を使用する場合には、分散表示管理 (DPM) オプションを使用できるように MFS を変換しなければなりません。このオプションは、MFS と論理装置内に常駐するユーザー・プログラム間におけるメッセージ・フォーマット設定作業を分散します。DPM オプションを使用すれば、物理端末特性は MFS に対して定義されません。MFS は、データをフォーマット設定して、そのデータを SLUTYPEP のユーザー・プログラム・コンポーネントに渡します。必要な場合には、フォーマット設定をユーザー・プログラムが完了して、データを物理装置に渡さなければなりません。

IMS に SLU P として定義された場合に正しく実行するためには、金融端末コントローラー内で実行される既存のリモート・ユーザー作成プログラムを次のように変換しなければなりません。

- 入力メッセージ・ヘッダーを SLU P メッセージ・ヘッダーに変換しなければなりません。
- 開始ブラケットのみを示す入力メッセージが方向変換も示さなければなりません。
- TERMINAL マクロの SCAN オプションが指定されている場合には、SCAN オプションによって提供された同じオプションを検出して処理できるように、金融機関コントローラー・アプリケーション・プログラムを変換しなければなりません。
- コントローラー・アプリケーション・プログラムが MFS を使用する場合には、装置独立の分散表示管理オプションを使用できるように、リモート・アプリケーションを変換しなければなりません。
- 金融機関固有の MFS から DPM への変換時には、IMS MFS 言語ユーティリティを使用して、MFS 形式記述を再定義および再コンパイルしなければなりません。

## 使用する VTAM 機能

IMS SLU P システム間におけるデータの物理伝送は、VTAM によって制御されます。

**関連資料:** IMS (VTAM アプリケーション・プログラム) とコントローラーの間のデータ伝送を管理する通信機能の詳細については、「*Network Program Products General Information*」および「*z/OS Communications Server: SNA Programming*」を参照してください。

IMS が特に SLU P システム用に使用する VTAM 機能には次のものがあります。

- 接続、切断およびログオン・モードの設定
- メッセージと応答
- 確定応答 1 および確定応答 2 (IMS とワークステーションの間で伝送されたデータはすべて、確定応答 1、確定応答 2、確定応答 1 と 2、または例外確定応答 1 ないし 2 を要求しなければなりません。VTAM コマンドは、確定応答 1 を要求しなければなりません。)
- 順序付けとチェーン
- 静止
- ブラケットおよび方向変換標識の使用を含む、順序正しい通信を確認するための機能。SNA プロトコルでは、ブラケットは 1 つ以上の要求単位 (RU) とその応答のチェーンであり、トランザクションを表す 2 つの LU から LU へのハーフセッション間で交換されます。VTAM では、方向変換標識は、送信側が送信を終了し、受信の準備ができたことを意味します。
- シーケンス番号のリカバリー
- 入力の受信、メッセージの送信
- 無条件ブラケット終了
- 基本エラー・リカバリー・プロシージャ (ERP)

ERP は、IMS 出力エラー・メッセージがワークステーションに送られると終了します。

IMS では、サービス・クラス (COS) とセッション故障停止通知 (SON) 機能もサポートしています。

### 関連概念

430 ページの『IMS における SON/COS サポートの使用』

セッション停止通知 (SON) とサービス・クラス (COS) は、IMS がセッション停止を認識できるようにする VTAM と SNA の機能です。

### 関連資料

982 ページの『エラー処理』

以下のトピックでは、伝送またはプロトコル・エラーから生じる障害を処理するために IMS によって使用され、コントローラーまたはコントローラー・アプリケーション・プログラムに必要なプロシージャについて説明します。

## SLU P で使用される VTAM コマンドと標識

IMS アプリケーション・プログラムと他の VTAM 論理装置の間のデータ伝送には、VTAM コマンドおよび標識 (通信制御情報) が必要です。

以下の表に、IMS が SLU P コントローラーとの間で送受信する VTAM コマンドおよび標識を示します。サポートされないコマンドまたは標識を使用した場合は、その結果を予測することはできません。

表の後に説明されているリカバリー要求コマンドと方向変換標識を除くコマンドおよび標識の動作は、「z/OS Communications Server: SNA Programming」で説明されています。

**推奨事項:** SLU P セッションでは、すべての入力メッセージに方向変換 (CD) またはブラケット終了 (EB) のいずれかが必要です。そうでないと、セッションは終了します。

表 174. IMS によって送受信される VTAM コマンドおよび標識

VTAM コマンド/標識	IMS がコントローラーへ送信	IMS がコントローラーから受信
INDEPENDENT OF SESSION		
セッション開始		X
プロシージャ・エラー	X	
セッション終了		X
SESSION CONTROL		
バインド	X	
クリア	X	
リカバリー要求 (RQR)		X
シーケンス番号のセットとテスト (STSN)	X	
開始データ通信 (SDT)	X	
アンバインド	X	
NORMAL (同期) FLOW		
ブラケット開始 (BB)	X	X
BID	X	
CANCEL	X	X
CHASE		X
ブラケット終了 (EB)	X	X



表 174. IMS によって送受信される VTAM コマンドおよび標識 (続き)

VTAM コマンド/標識	IMS がコントローラーへ送信	IMS がコントローラーから受信
論理装置状況 (LUS)		X
静止完了 (QC)	X	
受信準備 (RTR)		X
方向変換 (CD)		X
EXPEDITED (非同期) FLOW		X
チェーン終了静止 (QEC)		X
解放静止 (RELQ)		X
シャットダウン要求 (RSHUTD)		X
シャットダウン (SHUTD)	X	
シャットダウン完了 (SHUTC)		X
信号		X

## リカバリー要求コマンド

VTAM リカバリー要求 (RQR) コマンドを受信すると、IMS は CLEAR を出します。SLU P アプリケーション・プログラマーは、リカバリー可能 IMS 出力メッセージが進行中にこのコマンドを送信する場合は、注意が必要です。

RQR コマンドを IMS に送信するためには、アプリケーション・プログラムはまず、進行中の出力操作に回答しなければなりません。アプリケーション・プログラムが回答を送信せずに RQR コマンドだけを送信すると、コントローラーは自動的に LEXIT または次の LREAD ステートメントで未解決の DR2 応答を送信します。IMS が DR2 応答より前に RQR コマンドを受信すると、IMS が生成する CLEAR によって DR2 応答が失われることがあります。その結果、現行出力メッセージの状況は予測不能になります。

**制約事項:** ノードが応答モードになっており、応答メッセージがまだ出力可能でないとき、すなわち、入力応答モード・トランザクションがまだキューに入っているか実行中であるときには、SNA リカバリー要求 (RQR) コマンドによるセッション開始および再同期は許可されません。アプリケーション同期点より前では、応答モード・トランザクションはリカバリー可能でも再始動可能でもないので、入力処理が完了するまでセッション入力確認は行われません。

### 関連概念

952 ページの『メッセージ再同期』



メッセージ再同期の目的は、セッション全般を通じてメッセージの完全性を保証することにあります。

## 方向変換標識

IMS では、入力メッセージまたは VTAM コマンドの最後のチェーン・エレメントでの方向変換標識の使用がサポートされています。これにより、装置サポートは VTAM 操作とさらに互換性を保つことができます。IMS の操作は、この標識の影響を受けません。

## 接続の確立とログオン・モードの指定

IMS は、VTAM OPNDST マクロ命令を使用して接続を確立する時、IMS との通信時に論理装置が従わなければならない規則を定義するセッション・パラメーターを指定します。

### このタスクについて

IMS は、ユーザーが指定した、または VTAM デフォルトのセッション・パラメーターを調べ、IMS が依存するパラメーターのみを上書きします。残りのバイトは変更されません。ユーザー・データを BIND パラメーターに含めることができます。

BIND に組み込まれるユーザー・データは IMS によってサインオン・プロセスへの入力として使用され、機能的には IMS /SIGN コマンドと類似しています。これは ETO 金融機関端末および SLU P 端末には必須ですが、静的端末の場合には任意指定です。

IMS システム定義時に TERMINAL マクロ・ステートメントで、あるいはログオン記述子でモード名を指定しなければ、次のどの方法を使用してもモード名を指定することができます。

- VTAM ネットワーク・オペレーター・コマンド **VARY** で指定
- IMS **OPNDST**
- 他の論理装置によるシステム初期設定の要求で指定

IMS TERMINAL マクロ・ステートメントまたはログオン記述子でモード名を IMS /OPNDST コマンドまたは VTAM **VARY** コマンドでオーバーライドしない限り、そのモード名が使用されます。

モード名が、IMS システム定義時にログオン記述子を使用して指定されず、IMS コマンドでも、または CINIT を使用しても指定されない場合には、VTAM はデフォルトのモード名を想定します。

**関連資料:** ログオン・モード・テーブルの設定と、ログオン・モード・テーブル項目の定義について詳しくは、「z/OS Communications Server: SNA リソース定義解説書」を参照してください。

次の 2 つの方法のいずれかを使用して、デフォルトのログオン・モード・テーブル項目の使用をオーバーライドすることができます。

- システム定義時に、TERMINAL マクロで MODETBL= キーワードを使用してログオン・モード項目を指定することができます。
- /OPNDST コマンドの MODETBL キーワードにログオン・モード・テーブルを指定して、システム定義のテーブル項目を置き換えることができます。

IMS は機能管理、伝送サービス、および 1 次と 2 次のネットワーク・プロトコル用のセッション・パラメーターを上書きします。

### 関連資料

[SLU P と LU 6.1 のバインド・パラメーター \(システム・プログラミング API\)](#)

[TERMINAL マクロ \(システム定義\)](#)

[/OPNDST コマンド \(コマンド\)](#)

## XRF 複合システムとの接続の確立

XRF 複合システムにおいて IMS とのセッションを確立するには、ユーザーのログオン要求に、USERVAR テーブルで定義した USERVAR 名または XRF 複合システム内の両方の IMS システムに共用される MNPS ACB 名のいずれかを含める必要があります。

### このタスクについて

USERVAR を使用する XRF の場合、VTAM は、USERVAR ログオン・メッセージを現在アクティブな IMS アプリケーションと突き合わせます。MNPS を使用する XRF の場合、VTAM は、アクティブ IMS の MNPS ACB に直接リンクします。

USERVAR を使う XRF を使用する場合、SLU P 端末プログラムが、BIND の PLUNAME フィールドでアプリケーション ID 名を使用して IMS とのセッションを再確立することはできません。代わりに、SLU P 端末プログラムは BIND のユーザー・データ・フィールドで USERVAR を使用する必要があります。この制約事項は、MNPS を使用する XRF には適用されません。これは、このタイプの XRF システムは USERVAR を使用しないためです。

SLU P システムは、通常、クラス 1 端末として定義されます。XRF 複合システムでは、代替 IMS がアクティブ IMS から処理をテークオーバーする時、ほとんどの場合は、そのテークオーバーはクラス 1 端末セッションを失うことなく完了されます。端末は、テークオーバーに気付く場合と気付かない場合があります。

4730 自動取引装置を SLU P 装置として XRF IMS に接続する場合には、次の装置データ構成オプションを使用してください。

- 無効メッセージ: 10 または 11
- 複数コマンド: 10 または 11

1 桁目 (1) は、4730 がある種のネットワーク・メッセージ (XRF のテークオーバー後に送信される DFS3861 メッセージなど) に対して例外応答を送信しないようにするものです。これらのメッセージに対する例外応答は、4730 と IMS 間の SLU P セッションを終了させます。2 桁目 (0 または 1) は、エラーをログに記録するかどうかを示すものです。

**関連資料:** 4730 の詳細については、「*IBM 4730 Personal Banking Machine Operations Support Manual*」を参照してください。

### 関連概念

933 ページの『[XRF システム用のコントローラー・アプリケーション・プログラムに関する考慮事項](#)』SLU P システムが XRF 複合システムで稼働する場合には、コントローラー・アプリケーション・プログラムは、XRF のテークオーバー時に失われたメッセージを処理できるようにしなければなりません。

[XRF のバリエーション: MNPS と USERVAR \(システム管理\)](#)

[XRF の端末セッションを管理するための MNPS の指定 \(オペレーションおよびオートメーション\)](#)

### 関連資料

[金融機関通信システム・バインド・パラメーター \(システム・プログラミング API\)](#)

[XRF のための IMS.PROCLIB メンバーの定義 \(システム管理\)](#)

[IMS PROCLIB データ・セットの DFSHSBxx メンバー \(システム定義\)](#)

## ブラケットと送信 / 受信の管理

方向変換およびブラケット標識は、SLU P システムと IMS 間における同期伝送の開始、終了、および方向制御に使用されます。

これらの標識に対する IMS のサポートは、以下の表に要約されています (X = サポートされる)。

表 175. VTAM ブラケットおよび方向変換標識の使用

同期伝送	BB	EB	BB/EB	BB/CD
IMS がデータとともに受信した	X		X	X
IMS がデータとともに送信した		X	X	

表 175. VTAM ブラケットおよび方向変換標識の使用 (続き)

同期伝送	BB	EB	BB/EB	BB/CD
IMS がデータ・フロー同期コマンド (通常フロー) とともに受信した		X		

IMS は、ブラケット不在標識やインバウンド同期データでのブラケット終了のみの使用をサポートしません。IMS は、開始ブラケット (BB) のみを伴う出力同期データ、方向変換 (CD) のみを伴う出力同期データ、または BB/CD のみを伴う出力同期データは、送信しません。IMS は常に入力同期データでブラケットを終了するので (無条件ブラケット終了)、出力同期コマンドで EB または CD を送信する必要はありません。入力同期コマンドは EB または CD を指定できますが、現行ブラケットおよび送信 / 受信状態と矛盾しないようにする必要があります。そうでないと、セッションは終了します。IMS は、VTAM 標識をデータ・フロー同期コマンド (通常フロー) とともに送信しません。



# 第 50 章 SLU P および金融機関システムに使用される IMS 機能

以下のトピックでは、システム・ネットワーク体系 (SNA) 環境をサポートする IMS 機能について詳細に説明します。

## コンポーネントの定義

IMS は、ワークステーションを 1 つの物理端末と見なします。ワークステーションを複数の装置で構成する場合は、おのおのの物理装置を、そのワークステーションのコンポーネントとして定義する必要があります。

各コンポーネントは、そのワークステーションと関連する IMS 論理端末 (LTERM) を 1 つもつことができます。1 つの LTERM は各コンポーネントと関連付けられています。ユーザー作成の MPP は、この LTERM を通して、ワークステーションの特定のコンポーネントをアドレッシングすることができます。

IMS は、すべての入力をコンポーネント・リストにある最初の LTERM から行い、必要な操作とセキュリティ・チェックを行うものと見なします。メッセージ通信、特定の論理端末のブロードキャスト・メッセージおよびトランザクションの応答データは指定された出力 LTERM と関連するコンポーネントに命令します。

## LTERM 命名

CHNG 呼び出しを使用しやすくするために、LTERM の命名規則を定義してください。1 つの方法としては、LTERM 名をワークステーションとコンポーネント ID の組み合わせにすることです。

**例:** IMS は、4704 キーボードと表示システム・コンポーネントを 1 つのコンポーネントと見なし、4706 磁気ストライプ読取装置を別のコンポーネントと見なし、また 4710 レシート・プリンターもさらに別のコンポーネントと見なします。したがって、ワークステーション 100 は 3 つのコンポーネント、すなわち WS100DS (4704)、WS100MS (4706)、および WS100RP (4710) を持つことができます。このように命名規則を標準化しておくと、MPP は入出力 PCB (LTERM 名フィールド) を問い合わせるワークステーションを識別し、CHNG 呼び出しを使用して適切な出力の代替 PCB を指定することができます。

SLU P などの複数コンポーネント端末の場合は、IMS が端末をサービスするために十分なキューを作成するようする必要があります。IMS が必要な数のキューを作成するためには、その固有論理装置に対して出口ルーチンまたは特定のユーザー記述子を使用することができます。

## 出力コンポーネント選択

IMS システム定義により、1 台のワークステーションに最大 4 つの出力コンポーネントをもたせることができます。

1 つのワークステーションに 5 つ以上の出力コンポーネントが必要な場合や、複数の LTERM が望ましくない場合には、ユーザー定義の MPP - ワークステーション間プロトコルおよびデータ形式を提供する必要があります。

ワークステーション・コンポーネントには、X'01'、X'02'、X'03'、および X'04' の識別番号が割り当てられます。金融機関端末の場合には、端末が定義される順序に基づいて識別番号が決まります。SLU P 端末の場合には、識別番号はデフォルトの Comp<sup>1</sup>、(プログラム 1、基本) になります。IMS 非キューのシステム・メッセージには特定の宛先がないので、最初のコンポーネントに対して出されます。

IMS トランザクション入力編集ルーチンを使用すると、入力メッセージにステーションのノード名を追加することができます。MPP はノード名を参照して出力に使用する LTERM 名を判別することができます。拡張端末オプション (ETO) 機能を用いて作成された端末の場合には、ユーザーをサインオフできるので、入力メッセージにユーザー名を付加してください。

端末応答モードで稼働中のワークステーション、あるいは会話型トランザクションに応答メッセージを返すには、MPP で入出力 PCB に応答を挿入する必要があります。ただし、ステーションによっては、その応

答メッセージは、入出力 PCB に指定された LTERM と関連付けられたコンポーネント以外に対して出されたものである可能性があります。入出力 PCB の宛先は、CHNG 呼び出しで変更することができないので、代替 PCB タイプが必要になります。代替 PCB タイプは、これらの応答要件を満たす必要があり、変更可能でもなければなりません。応答代替 PCB と呼ばれるこの PCB タイプは、PSBGEN を使用して定義できます。会話型トランザクションや端末応答モードのステーションに対して応答メッセージを返すには、入出力 PCB の代わりに、この PCB を使うことができます。これは、修正可能と定義することもできます。CHNG 呼び出しと一緒に使うと、応答の宛先を選択することができます。CHNG 呼び出しを使用する場合、呼び出しで指定する LTERM は、入出力 PCB で指定する LTERM と同じ物理端末を割り当てる必要があります。もしこの割り当てがされない場合は、状況コードが ISRT 呼び出しに戻ります。

## 関連タスク

71 ページの『[拡張端末オプションの管理](#)』

IMS 拡張端末オプション (ETO) を使用すると、VTAM 端末とユーザーをシステム定義時に事前定義しなくても、IMS に動的に追加できるようになります。

## 入力コンポーネントの判別

入力コンポーネントと出力コンポーネント間の適切な関係は、IMS システム定義中の NAME マクロを使用して、または ETO ユーザー記述子とサインオン出口ルーチンを使用して設定することができます。

この関係により、端末はその入力コンポーネントを指定することができます。出力を IMS システム定義中に定義されたコンポーネントに返すことができます。入力コンポーネントを定義し、使用することにより、LTERM 命名規則、MPP 変更呼び出しおよび代替 PCB の挿入を必要以上に使用しなくて済みます。

ユーザーは、オペレーター、装置、またはリモート・アプリケーション・プログラムが制御する処理要件のそれぞれに対してコンポーネントを定義することにより、SLU P を使用するための接続を設定することができます。

IMS は、リモート論理装置に入力コンポーネント選択を実行し、入力はすべて必要な操作とセキュリティー・チェックを経たコンポーネント・リストの最初の LTERM から行われるものと見なします。SLU P からの入力の場合、入力コンポーネント選択は、オプションの入力メッセージ・ヘッダーが示すコンポーネントに基づいて実行されます。機能管理ヘッダーを受信しないと、IMS は入力がコンポーネント 1 の LTERM に関連付けられていると想定します。メッセージ通信、特定の LTERM に対するブロードキャスト・メッセージ、およびトランザクションへのデータ応答は、指定された出力 LTERM に関連付けられたコンポーネントに送信されます。

## 端末応答モード

端末応答モードは、IMS に接続されたトランザクション、ユーザー、および端末に定義できる操作のモードです。

SLU P ステーションが端末応答モードで作動している時には、IMS がトランザクションを受信してから、ワークステーションが応答メッセージを受信したという通知を IMS が受信するまで、ワークステーションと IMS 間のすべての操作は停止されます。通常の IMSIMS 出力メッセージの受信通知は、リカバリー可能トランザクションの出力を要求する DR2 応答の受信でされます。高速機能出力メッセージの受信通知は、次の入力メッセージの受信または RTR コマンドの受信です。MFS ページ出力メッセージの場合、この受信通知は、次の入力メッセージの受信、MFS NEXTMSG または NEXTMSGP 制御コマンドの受信、あるいはメッセージの最後のページで要求された DR2 応答 (高速機能出力でない場合) の受信です。リカバリー不能トランザクションによる出力は、例外 DR2 応答を要求しますが、ワークステーションからの応答は必要ありません。端末応答モードを使用すると、コントローラー・アプリケーション・プログラムに必要な処理を減らすことができます。

端末応答モードは、IMS システム定義中に、または ETO ユーザー記述子で選択することができます。システムは強制、否定、トランザクション依存として定義することができます。

- 強制と定義した場合は、入力トランザクションはすべて端末応答モードになります。
- 否定と定義した場合は、端末応答モードになる入力トランザクションはありません。
- トランザクション依存と定義した場合は、端末応答モードはトランザクションごとに決定します。端末応答モードをトランザクション依存で使用すると、コントローラーのアプリケーション・プログラムは



より複雑になります。その理由は、アプリケーション・プログラムが、強制と否定の両方の 端末応答モード環境からもたらされる通信プロトコルを取り扱わなければならないからです。

端末応答モードを呼び出せるのは有効なトランザクションだけであり、メッセージ通信や、IMS コマンド、VTAM コマンドおよび標識、あるいは MFS 制御要求などにより呼び出すことはできません。受け付けられないトランザクション (例えば、セキュリティー違反や、トランザクション・コードが無効であるという理由で) は、端末応答モードを呼び出すことはできません。

ワークステーションが端末応答モードで稼働している場合、ワークステーションが IMS とのセッションを設定した後で以下の処理が行われます。

1. ワークステーションが入力トランザクションを送信する。
2. IMS はワークステーションを端末応答モードにする。
3. IMS はトランザクションをメッセージ処理プログラム (MPP) に渡す。
4. MPP がトランザクションを処理する。
5. MPP が入出力 PCB または応答代替 PCB を使用して応答を返す。
6. IMS は DRx 応答が要求されていればこれを返す。
7. IMS は応答をワークステーションに送信する。
8. ワークステーションは DR2 応答が要求されていればこれを返す。
9. IMS はワークステーションを端末応答モードから外す。

ワークステーションが端末応答モードにある間は、IMS はワークステーションからの入力を受け取らず、MPP からの応答以外は出力をワークステーションに送信しません。入力トランザクションの処理中に MPP が異常終了した場合、IMS は、例外応答および関連した IMS エラー・メッセージを送る場合があります。端末応答モードを開始したトランザクションに対する応答ではない出力はすべて、IMS の出力キューに保留されます。IMS は、ワークステーションを端末応答モードから外した後、その非送信請求出力を送信します。

高速機能が使用されている場合、以下の処理が行われます。

1. ワークステーションが入力トランザクションを送信する。
2. IMS はワークステーションを端末応答モードにする。
3. IMS はトランザクションを高速機能メッセージ処理プログラム (IFP) に渡す。
4. IFP がトランザクションを処理する。
5. IFP が入出力 PCB または応答代替 PCB を使用して応答を返す。
6. IMS は DRx が要求されていればこれを返す。
7. IMS は応答をワークステーションに送信する。
8. 出力には、例外 DR2 応答が必要であるため、ワークステーションは次の入力メッセージか RTR コマンドを送る。IMS で、メッセージ・キューにワークステーションへの出力がある場合は、確定応答が必要です。ワークステーションは DR2 を送信します。
9. IMS はワークステーションを端末応答モードから外す。

マスター端末オペレーターは、マスター端末から **/STOP NODE** コマンドと **/START NODE** コマンドを順に発行することにより、高速機能端末応答モードで応答が返される前に、ワークステーションをリセットできます。

## 端末応答モードのためのワークステーション定義

端末応答モードで操作するワークステーションを定義する前に、端末応答モードの操作シーケンスを理解する必要があります。

端末応答モードで操作するステーションの場合、プロセッサ時間がより短くてすみ、このようなワークステーションを制御するコントローラー・アプリケーション・プログラムの複雑さを減らすことができます。

次の事項も考慮してください。



- 端末応答モードで操作するステーションから、典型的なデータ収集アプリケーションを実行することはできません。この応答モードでは、IMS が次のトランザクションを受信する前に、各トランザクションは MPP からの応答を必要とします。この応答待ちのため、データ入力プロセスの時間が長くなります。
- IMS は、出力応答が利用可能になるまで入力メッセージに応答を送りません。したがって応答が返された時には、入力が処理され、アプリケーションが同期点に達したことを示します。
- ワークステーションが端末応答モードである間、IMS は、そのステーションからそれ以上の入力を受け入れようとはしません。エラーで応答の作成や伝送ができない場合は、マスター端末オペレーターの介入が必要になります。ある条件下では、ワークステーションに応答が送れない場合があり、その条件として次のようなものがあります。
  - LTERM が停止した
  - IMS が MPP をスケジュールすることができない (データベース停止、MPP 停止)
  - MPP 論理エラーのために、ゼロ長応答を生成する EMH (急送メッセージ・ハンドラー) を除き、応答が返されなかった。

上記のいずれかの条件が発生した場合、ワークステーションは一時的に操作不能になります。マスター端末オペレーターがエラーを診断して訂正するまで、ワークステーションは使用可能になりません。

- IMS の出力キューに残っている応答メッセージ、またはセッション終了後にユーザー MPP で挿入された応答メッセージは、次のセッションが開始すると再び送られます。BID オプションを指定している場合は VTAM BID コマンドがこのメッセージより先行します。開始ブラケットとブラケット終了の両標識が応答メッセージで送られます。

システム分析者は、自分のシステムのアプリケーション・プログラムと操作環境に照らしてこれらの考慮事項を評価する必要があります。

## 関連概念

942 ページの『端末応答モード』

端末応答モードは、IMS に接続されたトランザクション、ユーザー、および端末に定義できる操作のモードです。

## ブラケット間状態中に送信された出力メッセージ

ワークステーションが出力から保護されておらずブラケット間状態にある時に、IMS 出力メッセージが送信されると、IMS は次のいずれかを送信します。

IMS は次のいずれかを送信します。

- ワークステーションが NOBID オプションで定義されている場合、ブラケット開始とブラケット終了の両方 (つまり、データに対するブラケットの要求) を使用してメッセージを送信。
- ワークステーションが BID オプションで定義されている場合は、ブラケットを開始する許可を要求する BID コマンドを送信。

ワークステーションは、要求された DR1 または DR2 を持つブラケットを受け入れることができます。送信要求に DR1 を指定すると、IMS は出力メッセージを開始、終了の両ブラケット付きで送信します。

ワークステーションは、該当する例外 DR1 または DR2 を含むブラケットを拒否することができます。

ブラケット拒否後、ワークステーションが出力可能になった場合、ワークステーションは、READY-TO-RECEIVE (RTR) コマンドを出して、IMS の出力を要求します。

次のステップは、セッションがブラケット間にある時、BID オプションで定義されたワークステーションが端末応答モードにある時に、IMS がワークステーションへの出力をどのように取り扱うかを示しています。

1. IMS は、BID オプションで定義されたワークステーションへのメッセージ通信を受信すると、メッセージを出力キューに入れます。
2. ワークステーションが端末応答モードでなくなってから、IMS は BID コマンドをワークステーションに送り、出力がブラケット間で保留状態にあることを知らせます。IMS は、この BID コマンドに対し DR1 応答を要求します。
3. ワークステーションは出力準備ができると、DR1 応答を返します。

4. IMS は DR1 応答を受け取ると、出力を送信します。
5. 要求があれば、ワークステーションは DR2 応答を返します。
6. ワークステーションが出力を受信する準備ができていない場合は、例外応答 DR1 を返し、「送信要求拒否」を示します。出力がリカバリー可能な場合は、IMS は、メッセージをキューに返し、ワークステーションが出力を受け取り可能であることを指示するまで待ちます。例外応答が "続く RTR はない" こと示し、出力がリカバリー不能である場合は、IMS はその例外応答を廃棄します。例外応答が「RTR が後に続く」ことを示している場合、IMS はメッセージをキューに返し (回復の可能性にかかわらず) VTAM の RTR (受信可) コマンドを待ちます。
7. ワークステーションが出力を受信する準備ができたなら、ワークステーションは VTAM RTR コマンドを送り、DR1 応答を要求します。
8. 送信要求を出した出力メッセージがまだ使用可能ならば、IMS は DR1 応答を返し、その後出力メッセージが続きます。
9. ワークステーションは DR2 応答 (12) を返します。

IMS がワークステーションから RTR コマンドを待つ間、ワークステーションはどんなタイプの処理でも実行することができます。この間、IMS はトランザクションの受信や応答はしますが、RTR コマンドを受け取るまで非送信請求出力を伝送することはできません。

ワークステーションが RTR を送信した時に IMS にはもう送信する出力がない場合 (/ASSIGN か /DEQUEUE コマンドが介入した場合など)、またはデータがリカバリー不能のため IMS がそのデータを廃棄した場合、IMS は例外 DR1 応答を返し、出力がないことを示すエラー・メッセージが出されます。

#### 関連概念

948 ページの『金融機関ステーションの表示画面保護』

金融機関ステーションが NOBID オプションで定義されている場合、IMS は IBM 3270 情報表示システムの画面保護機能と類似のサポートを提供します。

#### 関連資料

983 ページの『コントローラーまたはステーション検出エラー』

IMS からのメッセージでエラーを検出したとき、あるいは単にその時点でメッセージを受け入れることができない場合に、コントローラーは以下の 4 つのバイトのセンス・データを含む例外応答 DR2 を必ず戻します。

## ブラケット間状態中に送信される出力メッセージのための設計

IMS はブラケット間にある時、出力メッセージを実際に伝送する前に、BID コマンドをそのワークステーションに送って送信要求をします。

その送信要求が受信拒否され、例外センス・データに RTR がすぐ続いて来ることを示している場合はリカバリー可能出力メッセージはキューに残ります。それ以外では、メッセージは廃棄されます。

IMS がリカバリー不能出力メッセージ (照会のみトランザクション) に対する例外応答を受信すると、メッセージはなくなります。というのは、IMS はメッセージを VTAM に送るとすぐに、メッセージをキューから除去するからです。

#### 関連概念

944 ページの『ブラケット間状態中に送信された出力メッセージ』

ワークステーションが出力から保護されておらずブラケット間状態にある時に、IMS 出力メッセージが送信されると、IMS は次のいずれかを送信します。

#### 関連資料

983 ページの『コントローラーまたはステーション検出エラー』

IMS からのメッセージでエラーを検出したとき、あるいは単にその時点でメッセージを受け入れることができない場合に、コントローラーは以下の4つのバイトのセンス・データを含む例外応答 DR2 を必ず戻します。

## IMS メッセージ形式サービス

以下のトピックでは、SLU P システムに特別に適用されるいくつかの IMS MFS 機能について説明します。

### 関連概念

[IMS メッセージ形式サービス \(アプリケーション・プログラミング\)](#)

[MFS メッセージ形式 \(アプリケーション・プログラミング API\)](#)

## ワークステーション環境のための MFS の設計

入出力データは MFS によって処理できます。入力トランザクション編集ルーチンの出口で、追加編集ができます。

ワークステーションで MFS を使用できるかどうかは、IMS システム 定義時に、システム 管理者によって個別に定義されます。MFS を使用可能と定義した場合に、各入出力メッセージはオプションとして MFS を使用して処理することができます。MFS が処理するメッセージの形式は、IMS 提供の MFS 言語ユーティリティーを使用して定義されます。

MFS を使用した場合の操作と、IMS 基本編集機能を使用した場合の操作とは、まったく異なります。装置形式とオペレーターのプロシージャは、注意深く設計し、使いやすく、オペレーターの生産性を高めることを目的にする必要があります。

MFS には、メッセージ・フォーマット設定のレベルが2つあります。1つは、装置レベルのメッセージ・フォーマット設定で、もう1つは、分散表示管理 (DPM) です。装置レベルのメッセージ・フォーマット設定は金融機関論理装置用です。DPM サポートは SLU P 論理装置用です。

空画面を防止するため、ユーザー指定の MFS 形式記述の結果として MFS によって生成される空レコード (データ長 = 0) を、IMS は送信しません。

## MID/MOD チェーン

ワークステーション用の MFS 入力フォーマット設定が行われるのは、入力メッセージにメッセージ入力記述子 (MID) 名が与えられた時です。

### このタスクについて

MFS メッセージ出力記述子 (MOD) は、次の入力メッセージのフォーマット設定に使用される MID 名を提供できます。入力メッセージを送信する時にこの MID 名を提供するのは、ワークステーションの責任です。これにより、MID 名が提供されることになります。

SLU P システム・アプリケーション・プログラムの設計時に次のプロシージャに従うことにより、ワークステーション・オペレーターはほとんど介入せずに、MID/MOD チェーンを完成させることができます。

### 手順

1. 受信した出力メッセージのヘッダーから MID 名を除去し、その名前を次の入力メッセージで使用するために保管する。
2. 出力メッセージを表示または印刷する。
3. 次のオペレーター入力を受け取る。
4. ステップ 1 で保管された MID 名をトランザクションに追加する。
5. そのトランザクションを IMS に送信する。

### タスクの結果

#### 関連タスク

[971 ページの『金融機関ワークステーションに対する MFS 入力フォーマット設定の活動化』](#)

MFS を使用する場合、入力メッセージはメッセージ記述子とフォーマット設定記述子により処理することができます。

946 ページの『MID/MOD チェーン』

ワークステーション用の MFS 入力フォーマット設定が行われるのは、入力メッセージにメッセージ入力記述子 (MID) 名が与えられた時です。

## SLU P システムのための MFS 出力フォーマット設定

MFS を使用して、ページのフォーマット設定、標準ヘッダーとフッタの生成、および用紙の制御を行うことができます。MFS がない場合には、メッセージ処理プログラム (MPP) またはコントローラー・アプリケーション・プログラムがこれらの機能を提供しなければなりません。

また、MFS を使用すると、MPP は装置からの独立性が得られ、いくつかある端末タイプのどれからも呼び出せる少量タイプのアプリケーションでも SLU P コンポーネントが使用できるようになります。この装置からの独立性により、コントローラー・アプリケーション・プログラムは多量タイプのアプリケーションに専念することができ、複雑さを減らし、保守作業も減らせます。

ワークステーションに対する MFS の可用性は、IMS システムの定義時または ETO ログオン記述子の定義時に、システム管理者によって、ステーション別に定義されます。

## MFS メッセージ・リカバリー

MFS によって処理される入力メッセージは、キューイングされてログ記録がとられる前に編集されます。出力メッセージは伝送の直前に編集されます。したがってリカバリー目的のために、入力メッセージは MFS によってフォーマット設定され、出力メッセージは IMS アプリケーション・プログラムによってフォーマット設定されます。

## MFS 制御機能 (金融機関)

MFS を使用する装置のオペレーターは、複数の制御機能を使用できます。

MFS を用いた場合に装置のオペレーターが使用できる制御機能は次のとおりです。

### ページ先送り (NEXTPP)

現行メッセージに次の物理ページがあれば、それを送信します。

### 論理ページ先送り (NEXTLP)

現行メッセージの次の論理ページの最初の物理ページ (1 ページしかない場合もあります) を送信します。

### 論理ページ要求

nn を所要の論理ページの番号としたとき PAGEREQ=nn。要求が有効であれば、現行メッセージの指定された論理ページに定義されているデータ・フィールドを送信します。

### メッセージ先送り保護 (NEXTMSGP)

現行出力メッセージの印刷または表示を一時中断して、出力キュー内の次のメッセージの送信を開始します。次のメッセージが存在しなければ、例外 DR2 を戻してオペレーターに通知します。

### メッセージ先送り (NEXTMSG)

次のメッセージがあれば、現行出力メッセージの印刷または表示を一時中断して、そのメッセージの送信を開始します。

これらの制御機能を MFS に指示するには、その機能を入力機能管理ヘッダーに組み込むか、あるいは入力データ内のオペレーター制御フィールドを MFS に対して定義します。

## MFS 制御機能 (SLU P)

SLU P として定義された論理装置には、次の装置レベル MFS 制御機能が使用できます。

SLU P 端末は、MFS 装置形式でページング・オプションを指定した、DPM フォーマット設定出力メッセージを処理します。ページング・オプションは、OPTIONS=DPAGE または OPTIONS=PPAGE にできます。



### ページ先送り (NEXTPP)

現行メッセージに OPTIONS=PPAGE が定義されている場合には、次の表示ページに定義されているデータ・フィールドを送信します。現行メッセージに OPTIONS=DPAGE が定義されている場合には、次の論理ページに定義されているデータ・フィールドを送信します。

### 論理ページ先送り (NEXTLP)

現行メッセージがあり、OPTIONS=PPAGE または OPTIONS=DPAGE のいずれかが指定されている場合には、現行メッセージの次の論理ページに定義されているデータ・フィールドを送信します。

### 論理ページ要求

PAGEREQ=nn、ここで nn は所要の論理ページの番号。要求が有効であれば、現行メッセージの指定された論理ページに定義されているデータ・フィールドを送信します。

### メッセージ先送り保護およびメッセージ先送り (NEXTMSGP および NEXTMSG)

現行出力メッセージの転送を一時中断して、出力キュー内の次のメッセージの送信を開始します。

制御機能 NEXTPP、NEXTLP、NEXTMSGP、および NEXTMSG の指定は、入力機能管理ヘッダーで行うか、あるいは入力データ内のオペレーター制御フィールドを MFS に対して定義することによって行うことができます。論理ページ要求は、入力データ内か、オペレーター制御フィールドを使用してのみ入力することができます。

## MFS ページングおよび BID オプション

BID オプションは出力保護を提供しますが、パフォーマンス・コストが高く つきます (各メッセージごとに追加のライン・フローが必要)。

BID オプションが指定され、ブラケット間で出力が行われる時には、IMS は、出力を送信する前に送信要求 (BID) するために肯定応答を待機します。

MFS ページ出力の送信中に BID が送信されないようにするため、各入力ページング要求は開始ブラケットと方向変換を指示する必要があります。その結果、ページはイン・ブラケット状態で送信されるので、各出力ページにはブラケット終了のみが含まれ、送信要求の結果は含まれません。

## 金融機関ステーションの表示画面保護

金融機関ステーションが NOBID オプションで定義されている場合、IMS は IBM 3270 情報表示システムの画面保護機能と類似のサポートを提供します。

IMS は、コンポーネントからの入力メッセージを介在させずに、2つの出力メッセージを続けて送信してコンポーネントを表示することはしません。したがって、ユーザーは1つのメッセージを見て応答してから、次の出力メッセージを見ることができます。

IMS が出力メッセージを表示コンポーネントに伝送する場合、そのコンポーネントに保護のマークを付け、出力不可にします。IMS はそのコンポーネントから入力メッセージを受け取るまで、そこには次の出力メッセージを送信しません。この入力メッセージとは、IMS トランザクション、メッセージ通信、IMS コマンド、VTAM RTR コマンド、または MFS 制御要求です。これらのいずれかを受信すると、IMS はその表示コンポーネントの状況は無保護に変更します。

ワークステーションが BID オプションで定義されると、メッセージは連続してステーションに送信されなくなります。1回の送信の後で、次のメッセージを送信する前に、ワークステーションからの入力または、IMS からの BID コマンドへの肯定応答を受信しなければなりません。

ワークステーションが MFS を使用している場合には、画面は物理ページ単位で保護されます。追加のデータ画面を要求するためには、MFS 制御要求または入力データが使用されます。

ワークステーションが MFS を使用していない場合は、画面はメッセージ単位で保護されます。IMS トランザクションの入力、メッセージ通信、またはコマンドによって、画面は無保護状態になります。VTAM 受信可 (RTR) コマンドを使用して、次の出力メッセージを要求することができます。使用可能なメッセージがない場合や、ノードが出力を送信できない状態である (出力が停止または静止されている) 場合には、IMS は例外 DR1 を戻し、その後使用可能な出力がないことを示すエラー・メッセージが続きます。

MFS ページ化出力の伝送時に RTR コマンドが受信されると、IMS は、実行する MFS 制御機能を判別できないので、例外 DR1 応答を戻し、その後無効なページング要求を示すエラー・メッセージを表示します。

現行 IMS 出力メッセージが表示コンポーネントに送信された場合に、IMS が例外応答を受信した後、画面は無保護状態になります。IMS が例外応答を送信すると、定義された表示コンポーネントはすべて自動的に保護状態になり、出力に使用できなくなります。

## 関連資料

947 ページの『MFS 制御機能 (金融機関)』

MFS を使用する装置のオペレーターは、複数の制御機能を使用できます。

## 拡張出力コンポーネントの保護 (SLU P)

SLU P が NOBID オプションで定義されている場合、IMS は IBM 3270 情報表示システムの画面保護機能や金融機関コンポーネントの表示画面保護と類似した機能をサポートします。

LU から入力メッセージが適切に介入することなく、IMS は、2 つ続けて出力メッセージを出力保護のあるコンポーネントに伝送しません。この結果、次の出力メッセージが同じコンポーネントに送信される前に、LU はユーザー定義のプロシージャに従ってその出力メッセージを処理できます。SLU P サポートでは、すべてのコンポーネントの出力を保護することができます。

次の条件のもとで IMS は、コンポーネントは保護された状態で出力には使用不能であるとします。

- IMS が、IMS TERMINAL マクロで PROGRAM2 として定義されたコンポーネントにメッセージを送信した場合
- IMS が、ページング・オプションを定義された MFS によってフォーマット設定されたメッセージ (またはメッセージのページ) を送信した場合

コンポーネントが保護されている間は、論理装置からの入力メッセージによってコンポーネントが無保護にリセットされるまで、IMS は別の出力メッセージをコンポーネントに送信しません。コンポーネントの状況が無保護にリセットできる入力メッセージは次のとおりです。

- FM ヘッダーを含む IMS トランザクション
- FM ヘッダーを含むメッセージ通信
- FM ヘッダーを含む IMS
- 特定のコンポーネントが無保護であることを示す FM ヘッダーを含む MFS 制御要求
- 上記のいずれかで、FM ヘッダーを含まないか、あるいはゼロのコンポーネントを示す FM ヘッダー (これは端末のすべてのコンポーネントの保護をリセットする) を含むもの
- RTR コマンド

コンポーネントの状況が無保護にリセットされると、IMS は使用可能な出力を送信します。コンポーネントが PROGRAM2 として定義され、ページング・オプション OPTIONS=DPAGE または OPTIONS=PPAGE をもつ装置形式によって記述されていない場合には、そのコンポーネントへの出力はメッセージ単位で保護されます。入力メッセージは、出力コンポーネント保護をリセットするために使用されます。

SLU P コンポーネントが MFS DPM を使用するよう定義されており、使用される装置形式がページング・オプション OPTIONS=DPAGE または OPTIONS=PPAGE で定義されている場合には、論理ページまたは表示ページがそのコンポーネントに送信されると、出力コンポーネントは保護にセットされます。出力コンポーネントの保護をリセットし、MFS から追加の論理ページまたは表示ページを要求するには、MFS 制御要求または入力データが使用されます。

MFS ページ出力の伝送時に可能な入力はこのものだけです。

- メッセージ・ヘッダーを含まない入力 (すべてのコンポーネントの保護をリセットします)。
- MFS が現在ページングしている特定のコンポーネントを示すヘッダーを含む入力。この入力以外の場合には、IMS は端末の複数のコンポーネントに同時に出力を送信できないので、セッションは終了します。

SLU P 端末が BID オプションで定義されている場合には、すべてのメッセージの後で出力コンポーネントは保護されます。この操作モードでは、メッセージは連続して端末に送信されません。端末からの入力は受信しなければなりません。受信しないと、追加の出力がキューに入れられた場合、1 回の送信の後で、次のメッセージが送信される前に、BID コマンドが送信されます。

端末が MFS を使用していない場合には、画面の保護はメッセージ単位で実行されます。IMS トランザクションの入力、メッセージ通信、またはコマンドによって、画面は無保護状態になります。

受信可 (RTR) コマンドが受信され、使用可能なメッセージがない場合や、端末が出力を送信できない (出力が停止または静止されている) 状態にある場合には、IMS は例外 DR1 を戻し、その後使用可能な出力がないことを示すエラー・メッセージが続きます。MFS ページ化出力の伝送時に RTR コマンドが受信されると、IMS は、実行する MFS 制御機能を判別できないので、例外 DR1 応答を戻し、その後無効なページング要求を示すエラー・メッセージを表示します。

IMS が例外応答を送信すると、論理装置に定義された表示コンポーネントはすべて自動的に保護状態になり、出力に使用できなくなります。

PROGRAM2 として定義されたコンポーネントへの現行 IMS 出力メッセージ、または MFS DPM ページング出力メッセージについて例外応答が受信されると、出力が送信されたコンポーネントは無保護状態のままになります。

IMS が例外応答を送信すると、PROGRAM2 として定義された表示コンポーネントはすべて自動的に保護状態になり、出力に使用できなくなります。

#### 関連資料

947 ページの『MFS 制御機能 (SLU P)』

SLU P として定義された論理装置には、次の装置レベル MFS 制御機能が使用できます。

## 入出力編集オプション (SLU P)

SLU P として定義されたコンポーネントの入力または出力について、システム定義 TERMINAL マクロの COMPTn パラメーターにある編集 サブパラメーターのタイプ、および拡張端末オプション (ETO) ログオン記述子を、コンポーネント単位レベルで指示することができます。

提供される 4 つのタイプの編集は次のとおりです。

#### BASIC

IMS への入力時にブロック解除が実行されない要求。MFS は入力または出力に使用できません。

#### BASIC-SCS1

SNA 文字ストリング (SCS) 定義の改行 (NL-X'15') 文字または用紙送り (FF-X'0C') 文字が IMS に送信された時に非ブロック化することを要求。MFS は入力または出力に使用されません。

#### MFS-SCS1

SCS に定義された改行 (NL-X'15') 文字 または用紙送り (FF-X'0C') 制御文字が IMS に送信された時に非ブロック化することを要求。入力と出力の両方で MFS-SCS1 形式を使用することができます。

#### DPM-An

MFS 分散表示管理 (DPM) を使用してメッセージをフォーマット設定することができます。IMS への入力時に非ブロック化は行われません。「n」は 1 から 15 の数字です。

DPM の場合、装置タイプ記号名は DPM-Xn として MFS に指定されます (X は A または B です)。メッセージ・フォーマット設定は、DPM-An 形式の装置タイプ記号名を使用することによって、SLU P として定義された論理装置に対する TERMINAL マクロで指定されます。DPM-Bn 装置タイプ記号名は、システム間連絡 (ISC) に対して定義された論理装置を参照します。

BASIC または SCS1 が指定された場合には、1 つ以上の関連した伝送として IMS が受信または送信した各メッセージが、VTAM チェーンを形成します。

- BASIC の場合、入力 FM ヘッダーが提供されなくなると、IMS への各入力伝送は、MFS に表示される IMS セグメントとして、あるいは IMS メッセージ処理プログラムに直接表示される IMS セグメントとして扱われます。
- SCS1 の場合には、SCS 定義の各改行または用紙送り制御文字で、IMS 入力 セグメントが作成されます。IMS セグメントは、伝送の一部または全部から、または複数にまたがる伝送から作成することができます。

BASIC または SCS1 の場合には、各出力セグメントは IMS によって 1 つの伝送で送信されます。ただし、出力セグメントと IMS が付加した FM ヘッダーが、システム定義 TERMINAL マクロの OUTBUF パラメーターで、または ETO ログオン記述子で IMS に指定された論理装置のバッファー・サイズより大きい場合には、1 つの伝送では送信されません。この場合には、IMS は必要なだけ多くの回数でセグメントを送信します。各伝送は、セグメントの最終伝送を除き、定義された最大出力バッファー・サイズです。



DPM または MFS-SCS1 が指定されている場合には、入力メッセージには MFS DPM、出力メッセージには MFS-SCS1 を使用して、メッセージ単位でフォーマット設定することができます。フォーマット設定されないメッセージは、前に BASIC または SCS1 について説明したように編集されます。

#### 関連概念

[MFS メッセージ・フォーマット設定機能 \(アプリケーション・プログラミング API\)](#)

[IMS メッセージ形式サービス \(アプリケーション・プログラミング\)](#)

#### 関連タスク

[71 ページの『拡張端末オプションの管理』](#)

IMS 拡張端末オプション (ETO) を使用すると、VTAM 端末とユーザーをシステム定義時に事前定義しなくても、IMS に動的に追加できるようになります。

## 応答またはブラケットによるリカバリー可能入力の確認

メッセージ再同期を可能にするために、IMS は、オプションとして確定応答を要求する入力更新またはリカバリー可能照会トランザクションを許します。

IMS は、ユーザーがワークステーションに OPTIONS=OPTACK を指定すれば、確定応答に対するこの要求を許可します。

このオペランドを指定すれば、IMS は、入出力ブラケット標識を使用して、次の出力で入力を確認することができます。

**制約事項:** OPTIONS=ACK は、ETO 機能を使用して作成された SLU P 端末ではサポートされません。

OPTACK オプションが定義されている場合、ワークステーションが IMS へのリカバリー可能入力で開始ブラケット、方向変換、および例外 DR1 または例外 DR2 (DR1 または DR2 ではなく) を要求すると、パフォーマンスを向上させることができます。IMS からの次の出力は、ブラケット終了のみを指示することによって、入力を確認します。IMS から送信される出力メッセージのタイプは、リカバリー可能入力のタイプ、定義された応答モード、および出力の可用性によって決まります。

セッションがブラケット間状態にある時には、競合が起こる可能性があります。この場合 IMS は、ワークステーションが開始ブラケットと方向変換を示す入力メッセージを送信している同じ時に、BB と EB の両方を示す非送信請求出力メッセージまたは BID コマンドを送信します。この出力メッセージは入力メッセージを確認しないので、ワークステーションによって受諾または拒否される可能性があります。IMS リカバリー不能出力メッセージを拒否すると、メッセージが失われることとなります。IMS がリカバリー可能 IMS 出力メッセージ (DR2 応答を要求する出力) を送信している時にコンテンションが起こった場合には、ワークステーションが例外または確定応答を送信しなければ、IMS は入力メッセージを受信することができません。

#### 関連概念

[952 ページの『メッセージ再同期』](#)

メッセージ再同期の目的は、セッション全般を通じてメッセージの完全性を保証することにあります。

[944 ページの『ブラケット間状態中に送信された出力メッセージ』](#)

ワークステーションが出力から保護されておらずブラケット間状態にある時に、IMS 出力メッセージが送信されると、IMS は次のいずれかを送信します。

#### 関連タスク

[71 ページの『拡張端末オプションの管理』](#)

IMS 拡張端末オプション (ETO) を使用すると、VTAM 端末とユーザーをシステム定義時に事前定義しなくても、IMS に動的に追加できるようになります。

#### 関連資料

[953 ページの『金融機関および SLU P での高速機能メッセージ』](#)

高速機能を使用する際には、TERMINAL システム定義マクロで特定のオプションを指定する必要があります。

[967 ページの『SLU P メッセージ・プロトコル』](#)

VTAM コマンドおよび標識と MFS 制御要求の伝送は、1 回の伝送で行わなければなりません。IMS トランザクション、コマンド、およびメッセージ通信の伝送は、1 回または複数回の伝送で行うことができます。

[947 ページの『MFS 制御機能 \(金融機関\)』](#)

MFS を使用する装置のオペレーターは、複数の制御機能を使用できます。

495 ページの『TERMINAL マクロ』

TERMINAL マクロのいくつかのシステム定義キーワード・パラメーターは、ISC セッションを定義するために重要です。

## メッセージ・リカバリー

Tメッセージ・リカバリーのプロセスは、IMS 内で、IMS システム・ログおよび、チェックポイントと再始動のルーチンを使用して行われます。

IMS は通信リンクを直接制御するために、リカバリーが可能です。しかし、SLU P システムでは、この制御をコントローラーおよびコントローラー・アプリケーション・プログラムと共用しています。コントローラー・アプリケーション・プログラムが IMS からメッセージを受信している時に、プロセッサ障害または IMS 異常終了などのネットワーク障害が起こった場合には、従来の IMS リカバリー方法では、失われたメッセージを常に検出できるとはかぎりません。IMS リカバリー方法は、SLU P システムがメッセージ再同期用にコントローラー・アプリケーション・プログラムを組み込むように、拡張されています。

コントローラー・アプリケーション・プログラムは、メッセージと再同期化して、失われたメッセージ状況を検知および訂正できます。メッセージ再同期中にコントローラー・アプリケーション・プログラムに障害が発生すると、システムのメッセージの健全性が失われます。

ワークステーションとのセッションが、リカバリー可能メッセージを送信した時点とそのメッセージへの応答を受信する時点の間に終了した場合には、ワークステーションにはメッセージ再同期が必要になります。再同期のタイプはメッセージがどのように定義されているかにより異なります。

応答モード、応答会話型モード、および高速機能を除くすべてのメッセージの場合には、入力は、正常にエンキューされてスケジューリングできるようになった時に、リカバリー可能となります。メッセージ再同期時に、IMS は正常に受信した最後のリカバリー可能メッセージを示します。

応答モード、応答会話型モード、および高速機能の場合には、入力は、アプリケーションがその最初の同期点に達するまでリカバリー可能および再始動可能になりません。この同期点の前で IMS システムに障害が起こった場合には、メッセージは再始動不能と見なされます。メッセージ再同期中に、IMS はメッセージが同期点に到達したかどうか、または再発行が必要かどうかを示します。

IMS がワークステーションにメッセージを送信した後に (ただし、IMS が応答を受信する前に)、ワークステーションとのセッションが終了した場合は、このワークステーションにメッセージ再同期が必要になります。応答を受信しなかった出力メッセージは、メッセージ再同期によってワークステーションがメッセージを受信したかどうかを判別されるまで、このワークステーションと関連付けられたままでなければなりません。メッセージを別のワークステーションに移動するために /ASSIGN コマンドが使用された場合には、メッセージ再同期は不可能になります。

## メッセージ再同期

メッセージ再同期の目的は、セッション全般を通じてメッセージの完全性を保証することにあります。

IMS がコールド・スタートされなければ、メッセージ再同期はセッション開始時に起こります。金融機関および SLU P セッションは、IMS が制御ブロックを作成した後で記述子を変更または削除されたとしても、元の記述子から作成された制御ブロックを使用してウォーム・スタートします。コールド・スタートだけが、作成された制御ブロックが新しい記述子または更新された記述子を表すことを保証します。

ネットワーク障害のためメッセージ再同期が必要な場合、IMS が通常データ伝送を許可するためには、メッセージ再同期が正常に完了していなければなりません。メッセージ再同期を開始するには、IMS から VTAM に STSN (シーケンス番号の設定およびテスト) コマンドを送ります。LU はこのコマンドに対して応答する必要があります。正しく応答するためには、次のもののコピーが保守されている必要があります。

- LU によって送信された最後のインバウンド同期点メッセージの最後の要求単位 (RU) のシーケンス番号。インバウンド同期点メッセージは次のいずれかです。
  - この LU に対して IMS に ACK オプションが定義されている場合は、最後の正常なリカバリー可能入力メッセージ (DR1 または DR2 を要求したメッセージ)
  - この LU に OPTACK オプションが定義されている場合は、最後の入力メッセージ

- この LU によって受信された最後のアウトバウンド同期点メッセージの最後の要求単位 (RU) のシーケンス番号。アウトバウンド同期点メッセージは、正常な最後のリカバリー可能出力メッセージです。リカバリー可能出力メッセージは、DR2 応答を要求しているメッセージです。

高速機能が使用されている場合、各出力メッセージはリカバリー可能であり、例外 DR2 を要求します。これらの高速機能リカバリー可能メッセージは、明示的に要求された DR2 応答によってではなく、出力メッセージ・ヘッダー内のフラグによって識別されます。

必要な場合、端末によって送信された最終インバウンドリカバリー可能メッセージのコピーを LU が保守することができます。この保存が行われると、SLU P システムは、再同期に続いて、IMS が受信しなかったメッセージを再送信することができます。

**制約事項:** ノードが応答モードになっており、応答メッセージがまだ出力可能でないとき、すなわち、入力応答モード・トランザクションがまだキューに入っているか実行中であるときには、SNA リカバリー要求 (RQR) コマンドによるセッション開始および再同期は許可されません。アプリケーション同期点以前では、応答モード・トランザクションはリカバリー可能でも再始動可能でもないので、入力処理が完了するまでセッション入力確認は行われません。

応答モード・トランザクションがこの「未確定」状態にある時にセッション開始または再同期が行われると、セッションは終了し、IMS は、関連した入力同期点要求がコミットまたはバックアウトされたことを示すことができません。この一時エラー条件は、メッセージ DFS2081I を使用してマスター端末オペレーターに示されます。IMS /DISPLAY コマンドを使用して、いつ応答モード応答メッセージが使用可能になり、セッション開始を再試行できるかを判別することができます。表示状況 RESP-INP は、入力がまだ進行中であることを示します。RESP は、入力処理が完了して出力の伝送が可能であることを示します。

## XRF 複合システムの金融機関および SLU P

XRF 複合システムでの金融機関セッションおよび SLU P セッションは、クラス 1、2、または 3 の端末として扱われます。

クラス 1 は XRF バックアップ・システムの待機セッションを使用するとアクティブ IMS セッションをトラッキングして最高レベルの透過性とパフォーマンスを提供します。クラス 2 と 3 は、IMS システムに障害が起きた場合に、アクティブ・セッションを終了させる働きをします。テークオーバー時に、代替 IMS システムは自動的にセッションを再初期設定し、必要に応じて自動的に元のアクティブなシステム・ユーザーのサインオンをします。クラス 3 のサポートは、XRF テークオーバー後は手動でセッションを再始動しサインオンする必要があります。

z/OS の視点からの XRF については、「z/OS Communications Server: SNA ネットワーク・インプリメンテーション・ガイド」を参照してください。

### 関連概念

[拡張回復機能の概要 \(システム管理\)](#)

### 関連タスク

938 ページの『XRF 複合システムとの接続の確立』

XRF 複合システムにおいて IMS とのセッションを確立するには、ユーザーのログオン要求に、USERVAR テーブルで定義した USERVAR 名または XRF 複合システム内の両方の IMS システムに共用される MNPS ACB 名のいずれかを含める必要があります。

## 金融機関および SLU P での高速機能メッセージ

高速機能を使用する際には、TERMINAL システム定義マクロで特定のオプションを指定する必要があります。

TERMINAL マクロで次のオプションを指定する必要があります。

### OPTACK

開始ブラケット標識と方向変換標識が入っている入力メッセージを次の出力メッセージのブラケット終了標識によって確認することができます。

### FORCERESP または TRANSRESP

応答モード・メッセージを IMS に入力することができます。

最善のパフォーマンスを得るためには、高速機能入力メッセージが確定応答を要求するのではなく、開始ブラケット、方向変換、および例外 DR2 応答でこのメッセージをコーディングします。

### **FPACK/NFPACK**

特殊な高速機能出力プロトコルを使用するかどうかを指定します。

高速機能トランザクションは単一セグメント・トランザクションで、リカバリー可能として定義されます。高速機能トランザクションは応答モードとして定義しなければなりません。

## **高速機能出力メッセージ (金融機関)**

高速機能出力メッセージは、システム定義 TERMINAL マクロのオプション FPACK または ETO ログオン記述子で定義されると、例外 DR2 応答を要求して送信されます。

高速機能出力は、以下の場合に DR2 応答を要求するよう送信されます。

- キューにある出力がディスプレイ以外のコンポーネントに送信可能になった場合
- システム定義 TERMINAL マクロのオプション BID または ETO ログオン記述子オプション BID が定義されており、キューにある出力メッセージがディスプレイのコンポーネントに送信可能である場合
- システム定義 TERMINAL マクロのオプション NOBID または ETO ログオン記述子オプション NOBID が定義されて、高速機能出力応答メッセージがディスプレイ以外のコンポーネントに向けられた時にキューにある出力がディスプレイ・コンポーネントに送信可能となった場合。高速機能出力によって、ディスプレイ・コンポーネントは画面保護されません。

非高速機能出力とは異なり、出力メッセージは未処理のままにされ (キューから除去されない)、ワークステーションがデータ、受信可 (RTR) コマンド、または DR2 応答を送信してメッセージをキューから除去するまで、ワークステーションは応答モードのままです。異常に長い間入力が生成されない場合には、RTR コマンドを考慮する必要があります。例えば、端末オペレーターが端末を離れる予定がある時などです。ワークステーションからの入力メッセージ、RTR コマンド、または前の出力が受信されたという DR2 応答確認は、リカバリー可能です。したがって IMS は、出力メッセージをキューから除去したり、入力メッセージを処理したり、使用可能な出力を送信したりすることができます。

システム定義 TERMINAL マクロのオプション NFPACK または ETO ログオン記述子は、高速機能出力例外 DR2 および次の入力確認プロトコルを使用してはならないことを示します。この場合、高速機能出力メッセージは、標準リカバリー可能出力メッセージ応答プロトコルを要求するように常に送信されます (DR2 応答)。

DR2 応答が受信されると、IMS は出力メッセージをキューから除去し、端末を応答モードから外し、キューにある使用可能な出力を送信します。IMS 画面保護サポートのため、高速機能出力応答メッセージが同じコンポーネントに送信された場合には、キューにある出力はディスプレイのコンポーネントには送信されません。

### **関連概念**

948 ページの『[金融機関ステーションの表示画面保護](#)』

金融機関ステーションが NOBID オプションで定義されている場合、IMS は IBM 3270 情報表示システムの画面保護機能と類似のサポートを提供します。

## **高速機能出力メッセージ (SLU P)**

高速機能出力メッセージは、システム定義 TERMINAL マクロのオプション・ステートメント FPACK または ETO ログオン記述子で定義されると、特定のイベントが発生した場合に、例外 DR2 応答を要求して送信されます。

高速機能出力メッセージは、次のいずれかの場合に例外 DR2 応答を要求します。

- キューにある出力メッセージが、PROGRAM1 としてワークステーションに定義されたコンポーネントに送信可能である。
- システム定義 TERMINAL マクロのオプション BID、または ETO ログオン記述子オプション BID が定義され、キューにある出力が PROGRAM2 としてワークステーションに定義されたコンポーネントに送信可能である。



- システム定義 TERMINAL マクロのオプション NOBID、または ETO ログオン記述子オプション NOBID が定義され、キューにある出力メッセージがプログラムに送信可能である。このコンポーネントは直前の出力またはこの高速機能メッセージにより、出力保護されません。(つまり、複数のコンポーネントは、本章で定義されているように、選択的に保護あるいはリセットされることがあります。)

非高速機能出力とは異なり、出力メッセージは未処理のままにされ(デキューされない)、端末は、データ、受信可(RTR) コマンド、または DR2 応答を送信してメッセージをデキューするまで、応答モードのままです。異常に長い間入力生成されない場合には、RTR コマンドを考慮する必要があります。例えば、端末オペレーターが端末を離れる予定がある時などです。端末からの入力メッセージ、RTR コマンド、または DR2 応答は、先行する出力が受信され、リカバリー可能であることを確認します。したがって、IMS は出力メッセージをデキューしたり、入力メッセージを処理したり、使用可能な出力を送信したりすることができます。

システム定義 TERMINAL マクロのオプション NFPACK または ETO ログオン記述子は、高速機能出力例外 DR2 および次の入力確認プロトコルを使用してはならないことを示します。この場合、高速機能出力メッセージは、標準リカバリー可能出力メッセージ応答プロトコルを要求するように常に送信されます(DR2 応答)。

DR2 応答が受信されると IMS は、出力メッセージをキューから除去し、端末を応答モードから外し、ワークステーションに定義された PROGRAM1 コンポーネントまたは非出力保護 PROGRAM2 コンポーネントのキューにあるすべての使用可能な出力を送信します。出力保護は、PROGRAM2 コンポーネントへの出力時にコンポーネント・レベルでセットされ、後続の入力に基づいて1つ以上のコンポーネントについて選択的にリセットすることができます。

すべての 4730 端末に対して NFPACK オプションも使用する必要があります。これにより、高速機能出力応答の確認の直後に、非同期出力メッセージを送信することができます。

#### 関連概念

949 ページの『[拡張出力コンポーネントの保護\(SLU P\)](#)』

SLU P が NOBID オプションで定義されている場合、IMS は IBM 3270 情報表示システムの画面保護機能や金融機関コンポーネントの表示画面保護と類似した機能をサポートします。

## 高速機能メッセージ再同期

高速機能が定義されている場合、シーケンス番号管理とメッセージ再同期の変更が必要になる場合があります。

#### 関連概念

952 ページの『[メッセージ再同期](#)』

メッセージ再同期の目的は、セッション全般を通じてメッセージの完全性を保証することにあります。



## 第 51 章 SLU P および金融機関でのネットワーク操作

本章トピックでは、IMS ネットワークの開始方法、セッションの開始方法、各種トランザクション・タイプ、メッセージ通信、および IMS コマンドについて説明します。

### IMS ネットワークの開始

ワークステーションで IMS とのセッションを設定するには、事前に VTAM、NCP、コントローラー、および論理装置をアクティブにする必要があります。

#### このタスクについて

特に、以下のコンポーネントをアクティブにする必要があります。

- VTAM と NCP がアクティブになっていること。
- VTAM では自動的に活動化されないコントローラーや論理装置が、VTAM ネットワーク・オペレーターにより **VARY** コマンドで活動化されていること。
- コントローラーが、電源オンされており、かつ VTAM と NCP 用に適切に構成され、初期設定され、活動化されていること。

### IMS を作動可能にする

IMS は、VTAM ログオン (セッション初期設定) 要求を受信準備済みにしなければなりません。IMS を作動可能にするには、DC キーワードを指定した **IMS /START** コマンドを発行します。

#### このタスクについて

**/START DC** コマンドは、キューにある VTAM ログオン要求 (および IMS に属することを VTAM が認識しているワークステーションのログオン要求) をすべて IMS に渡すように VTAM に伝えます。

**/START DC** コマンドは以下のプロセスを活動化します。

- IMS Transaction Manager 処理を開始する。
- VTAM アクセス方式制御ブロックをオープンする。
- IMS VTAM ログオン出口を使用可能にする。

IMS **/START DC** コマンド発行の前で、かつ IMS VTAM アクセス方式制御ブロック (ACB) のオープン後に、VTAM が受信したすべてのログオン要求は、**/START DC** コマンド処理が完了するまで VTAM のキューに入れられます。IMS の初期設定時に VTAM がアクティブであり、DFSDCxxx PROCLIB メンバーのキーワード VACBOPN=INIT が指定されていると、IMS VTAM ACB がオープンされます。DFSDCxxx PROCLIB メンバーのキーワード VACBOPN=DELAY が指定された場合は、IMS VTAM ACB のオープンは、**/START DC** コマンドが処理されるまで遅延されます。

### セッション開始 (ワークステーション開始)

セッションとは、IMS またはシステム・ユーティリティーなどの VTAM アプリケーション・プログラムに対するワークステーションの論理接続のことです。データをワークステーションと IMS の間で伝送するには、その前にワークステーションと IMS 間のセッションを設定する必要があります。IMS をコールド・スタートする場合を除き、メッセージ再同期はセッション開始時に実行されます。

#### このタスクについて

セッション開始は、以下の方法のいずれか 1 つで要求することができます。



- ワークステーションが **INITIATE-SELF** コマンドを送ってセッション開始を要求する。VTAM がコマンドを検証し、IMS にその要求を渡します。端末が IMS XRF 複合システムとのセッションを要求している場合は、アプリケーション名には IMS USERVAR 名または MNPS ACB を使用する必要があります。
- z/OS VTAM ネットワーク・オペレーターは、LOGON オプションを指定した VTAM **VARY** コマンドを使用して、ワークステーションに代わってセッション開始を要求する。VTAM はその要求を処理し IMS に渡します。
- VTAM が、IMS に属すると VTAM に定義されている各ワークステーションごとに、ログオン要求を IMS に渡す。
- IMS のマスター端末オペレーターが、IMS **/OPNDST** コマンドを入力してワークステーションにセッション開始を要求する。

セッション開始がどのように要求されたかにかかわらず、IMS が要求を受信すれば同一の処理が始まります。

## セッション開始伝送順序

次のリストは、ワークステーションでセッション開始を要求した時の伝送順序を示しています。

数字は主要なイベントの順序を示します。

- コントローラーが **INITIATE-SELF** コマンドを送信します。VTAM は、コントローラーまたは NCP のいずれかによって引き起こされた可能性のある構文エラーを探し、コマンドの妥当性を検査します。
- コマンドが無効の場合、あるいは VTAM が開始要求を検査した時に有効なパラメーターが見つからなかった場合、VTAM は EXC/DR1 応答を戻します。EXC/DR1 は開始要求を終了します。SLU P システムでは開始要求を再試行するか、あるいは正しい処置を開始することをオペレーターに通知します。

コマンドが有効な場合、VTAM は DR1 応答を戻します。その後、VTAM は **INITIATE-SELF** コマンドのリソース・フィールドの内容を検査します。このフィールドには、ワークステーションがセッションを開始する VTAM アプリケーション・プログラムの名前が入っていなければなりません。VTAM は、リソース・フィールドの内容を、そのアクティブ・アプリケーション・プログラム (オープン VTAM ACB を持つプログラム) のリストと比較します。

- 一致したプログラムが見つかった場合、VTAM は ログオン要求を指定のアプリケーション・プログラム (IMS) に渡します。

XRF 環境では、VTAM は現在アクティブになっているシステムに要求を経路指定します。

要求を受信すると、IMS は、VTAM が提供したワークステーション・ノード名を IMS システム定義時に定義されたノード名と比較します。

- IMS がワークステーションを認識しない場合、IMS は、VTAM にプロシージャー・エラー・コマンドを送信させる VTAM **CLSDST** コマンドを発行します。プロシージャー・エラーはセッション開始要求を終了します。システムは、セッションが拒否されたことをワークステーション・オペレーターに通知します。

IMS がワークステーションを認識した場合、IMS は、VTAM に **BIND** コマンドを送信させる VTAM **OPNDST** コマンドを発行します。**BIND** コマンドには、**OPNDST** コマンドを発行したアプリケーション・プログラム (IMS) の名前が含まれています。

MNPS の代わりに USERVAR を使用する XRF 複合システムでは、VTAM からの **BIND** コマンドのユーザー・データ・フィールドに USERVAR セグメントが、PLUNAME フィールドにアクティブなアプリケーション ID が入っています。

IMS は、従うべき通信規則とプロトコルを定義する一組の **BIND** パラメーターを提供します。IMS は、ワークステーション・オペレーターまたはネットワーク・オペレーターによって提供されたモード・テーブル項目のパラメーターを無視します。

ワークステーションは **BIND** コマンドに応答しなければなりません。

**関連資料:** **BIND** データの内容について詳しくは、「IMS V15 システム・プログラミング API」の『SLU P および LU 6.1 のバインド・パラメーター』を参照してください。

- ワークステーションがセッションを開始することができない場合は、ワークステーションは EXC/DR1 を戻します。これは、VTAM または IMS のいずれかがセッション開始を要求しているが、ワークステーションは現在通信不能である (例えば、オフライン処理に参与している) 場合に起こる可能性があります。

ワークステーションがセッションを開始することができる場合には、DR1 が戻されます。IMS は DR1 を受信した時に、必要であれば、メッセージ再同期を実行します。

- メッセージ再同期が不要な場合、または完了した場合、IMS は VTAM 開始データ通信 (SDT) コマンドを送信します。

ワークステーションはセッション中で、IMS に データを伝送することができます。

## 関連資料

[SLU P と LU 6.1 のバインド・パラメーター \(システム・プログラミング API\)](#)

## メッセージ再同期でのコントローラー・アプリケーション・プログラム関与

コントローラー・アプリケーション・プログラムは、メッセージ再同期に関与しなければならず、そのためには、IMS に 正常に送信された最終リカバリー可能メッセージとワークステーションが正常に受信した最終リカバリー可能メッセージのシーケンス番号のコピーを保持していなければなりません。

## 設計の考慮事項

システム・アプリケーション分析者は、シーケンス番号の設定およびテスト (STSN) 処理に必要なシーケンス番号を保持する場所を判別する必要があります。

使用可能なオプションは次の 3 つです。

- コントローラー・ディスクまたはディスクット

コントローラーのディスクまたはディスクットは、最も信頼できる方式で、メッセージの再伝送が必要な場合には、ワークステーション・オペレーターは関与する必要がありません。ディスクまたはディスクットの永続ファイルまたは一時ファイルいずれかを使用することができます。ディスクまたはディスクット・ファイルへの書き込み時には、REPLACE 命令を使用する必要があります。REPLACE によって、データはディスクまたはディスクットへ物理的に書き込まれ、コントローラーの制御記憶装置にはバッファーされません。シーケンス番号は、コントローラーの制御記憶装置の内容を破壊する可能性がある電源障害またはコントローラー障害が発生しても、その間ディスクあるいはディスクットのストレージに確実に保存されています。ただし、ディスクまたはディスクットのアクセス時間およびデータ転送時間のために、リカバリー可能トランザクションごとのオーバーヘッドがさらに増大する可能性があります。

- コントローラーの制御記憶装置

シーケンス番号の保存にコントローラーの制御記憶装置を使用している場合は、停電またはコントローラー障害によってコントローラーの保管内容が破壊されると、メッセージ再同期を行うことができなくなります。メッセージ再同期が実行されないため、リカバリー可能入力メッセージおよびリカバリー可能出力メッセージが消失または重複する可能性があります。

- ワークステーション出力装置

メッセージ・シーケンス番号は、ディスプレイまたはジャーナル・プリンターなどのワークステーション出力コンポーネントに保管することができます。しかし、この方式では、STSN 処理時に番号をリトリブするためにワークステーション・オペレーターが介入する必要があります。

## シーケンス番号管理

通常のメッセージ伝送時にシーケンス番号がどのように処理されるかを理解することは、SLU P システムが、このトピックで述べた機能を実行する方法と理由を理解するための前提条件です。

シーケンス番号の管理は、コントローラー、VTAM、および VTAM アプリケーション・プログラム (IMS) の間で共用されます。コントローラーは、シーケンス番号をそのワークステーションが起点となるメッセージに割り当てます。VTAM は、VTAM シーケンス番号を IMS が起点となるメッセージに割り当てます。リカバリー可能性を確実にするために、IMS は、送信するメッセージと関連したシーケンス番号の個別コピーを保持します。

シーケンス番号は各ワークステーションに割り当てられます。コントローラーは、各ワークステーションのそれぞれについて、自分が割り当てた最終シーケンス番号(最終割り当て値と呼ばれます)および受信した最終シーケンス番号(最終受信値と呼ばれます)を覚えています。同様に、IMSは、VTAMが割り当てた最終シーケンス番号(最終割り当て値と呼ばれます)とIMSが受信した最終シーケンス番号(最終受信値と呼ばれます)を追跡します。

コントローラーはデータを伝送する要求を発行する時に、要求ワークステーションの最終割り当て値を更新し、新しい番号をデータに付加し、メッセージを送信します。VTAMは、メッセージを受信すると、単にそのメッセージをIMSに渡します。次にIMSは、そのワークステーションの最終受信値に1を追加し、この値を受信したばかりのメッセージのシーケンス番号と比較します。2つの数値が一致する場合、IMSは、メッセージを必要に応じて処理します。2つの数値が一致しない場合、IMSは、VTAM CLSDST マクロ命令を発行してセッションを終了します。

IMSに伝送するメッセージがある場合、IMSはVTAMの最終割り当て値のコピーを更新し、VTAMにメッセージを送信します。VTAMはその最終割り当て値を更新し、新しい番号をメッセージに付加し、メッセージを送信します。コントローラーは付加したシーケンス番号を除去し、その最終受信値を更新し、この値をメッセージに付いているシーケンス番号と比較します。2つの番号が一致する場合には、コントローラーはそのメッセージをアプリケーション・プログラムに送信します。2つの番号が一致しない場合には、コントローラーはシーケンス・エラーを示すために例外DR2を戻します。シーケンス・エラー指示を受信すると、IMSは、VTAM CLSDST マクロ命令を発行してセッションを終了します。

## シーケンス番号の設定およびテスト (STSN)

IMSがワークステーションにシーケンス番号の設定およびテスト(STSN)コマンドを送信すると、メッセージ再同期が開始されます。

STSNコマンドには、次の5バイトのデータ・フィールドが含まれています。

### バイト0

アクション・コード

### バイト1、2

IMSがワークステーションから受信した最終インバウンド同期点メッセージのコントローラー・シーケンス番号

### バイト3、4

IMSがワークステーションに送信した最終アウトバウンド同期点メッセージのVTAMシーケンス番号

IMSはアクション・コードを使用して、コントローラー・シーケンス番号およびVTAMシーケンス番号を検査するようにSLUPシステムに要求します。アクション・コード・バイトの各ビットは次のとおりです。

### ビット0、1

コントローラー・シーケンス番号フィールドを参照します。

### ビット2、3

VTAMシーケンス番号フィールドを参照します。

### ビット4、5、6、7

予約済み

アクション・コードのビット0、1、2および3に受け入れ可能な値は次のとおりです。

### 00 IGNORE

IMSでは使用されません。

### 01 SET

シーケンス番号をシーケンス番号フィールドで指示されている値に設定します。

### 10 INVALID

IMSでは使用されません。SLUPシステムは、シーケンス番号のそのバージョンをコマンドの応答で戻す必要があります。

### 11 SET AND TEST

シーケンス番号をシーケンス番号フィールドで指示されている値に設定します。SLUPシステムは、当該のシーケンス番号値が受け入れ可能かどうかをコマンドへの応答で指示する必要があります。

IMS は、コントローラー・シーケンス番号に SET オプションを使用します。VTAM シーケンス番号には、IMS は SET または SET AND TEST のいずれかを使用します。SET を使用するの、直前のセッションからのリカバリー可能出力メッセージへの確認応答が未解決になっていない場合です。また、SET は、最終リカバリー可能メッセージのキューからの除去をもたらす、 /DEQUEUE コマンドまたは /ASSIGN コマンドを IMS マスター端末オペレーターが入力する場合にも使用されます。SET AND TEST を使用するの、IMS がワークステーションに リカバリー可能メッセージを送信したが、セッションの終了前に必要な確認応答を受信しなかった場合です。SLU P システム が STSN コマンドに送信する応答は、ワークステーションがメッセージを受信したかどうかを示します。

SLU P システムは、STSN コマンドを受信した場合に、次のことが可能でなければなりません。

- コントローラー・シーケンス番号を検査し、必要な場合には、IMS へのメッセージの再送を行うこと。
- VTAM シーケンス番号を検査し、番号が受け入れ可能かどうかを IMS に通知すること。
- IMS に DR1 を戻し、必要な場合には、STSN コマンドに対して 5 バイトのデータ応答を戻すこと。

コントローラー・シーケンス番号を検査するために、SLU P システムは IMS が提供した番号を直前のセッションから保持されていた番号と比較します。比較した番号が等しい場合は、IMS がワークステーションの送信したすべてのメッセージを受信したことを示します。IMS 提供の番号の方が小さい場合、IMS は、ワークステーションが送信した最終インバウンド同期点メッセージを受信していません。これら 2 つの番号の差が 1 より大きくなる可能性があるのは、ワークステーションが ACK オプションを使用して定義されている時に、介入するリカバリー不能メッセージまたはチェーン・メッセージをワークステーションが送信した場合です。IMS が受信しなかったメッセージは、メッセージ再同期が完了した後で再送する必要があります。メッセージのコピーを保守している場合には、そのコピーを送信する場合があります。メッセージのコピーを保持していない場合には、SLU P システムは、IMS が最終インバウンドリカバリー可能メッセージを受信しなかったことをワークステーション・オペレーターに通知することができます。

VTAM シーケンス番号を検査するために、SLU P システムは IMS が提供した番号を直前のセッションから保持していた番号と比較します。番号が等しい場合は、ワークステーションがすべてのメッセージを受信したことを示しています。IMS 提供の番号と等しくない場合には、ワークステーションは最終アウトバウンドリカバリー可能メッセージを受信していません。

STSN コマンドへの応答は、IMS への同期テストの結果を示しています。SLU P システムは DR1 を戻す必要があります。5 バイトのデータは任意で戻します。コントローラーと VTAM の両方のシーケンス番号が受け入れ可能な場合は、DR1 のみを必要とします。一方のシーケンス番号または両方のシーケンス番号が受け入れ可能でない場合には、DR1 および 5 バイトのデータの応答が必要です。次のように、データ応答の形式は STSN コマンドと同じ形式です。

#### バイト 0

アクション・コード

#### バイト 1、2

ワークステーションが IMS に送信した最終インバウンドリカバリー可能メッセージのコントローラー・シーケンス番号

#### バイト 2、3

ワークステーションが IMS から受信した最終アウトバウンドリカバリー可能メッセージの VTAM シーケンス番号

SLU P システムはアクション・コードを使用して、テスト結果を指示します。アクション・コード・バイトの各ビットは次のとおりです。

#### ビット 0、1

コントローラー・シーケンス番号フィールドを参照します。

#### ビット 2、3

VTAM シーケンス番号フィールドを参照します。

#### ビット 4、5、6、7

予約済み

アクション・コードのビット 0、1、2 および 3 に受け入れ可能な値は次のとおりです。

#### 00 RESET

コマンドのシーケンス番号は受け入れ不能で、直前の値で再設定する必要があります。このコードを IMS に戻してはなりません。

## 01 TEST POSITIVE

コマンドのシーケンス番号は受け入れ可能です。このコードは SET オプションへの応答で戻す必要があります。SET AND TEST オプションが指定されており、SLU P システムが当該のシーケンス番号は受け入れ可能であると判定した場合は、TEST POSITIVE が戻されなければなりません。

## 10 INVALID

SLU P システムがシーケンス番号でエラーを検出しました。

## 11 TEST NEGATIVE

SLU P システムが、SET AND TEST オプションによって提供されたシーケンス番号を受け入れません。

TEST POSITIVE は SET オプションに対する唯一の受け入れ可能な応答です。TEST POSITIVE および TEST NEGATIVE は SET AND TEST オプションに対する受け入れ可能な応答です。シーケンス番号の比較時に、SLU P システムが「should-not-occur」条件を検出した場合には、INVALID を使用する必要があります。この条件の例は、アプリケーション・プログラムのシーケンス番号のコピーより大きいコントローラー・シーケンス番号がコマンドで指定されている場合です。

IMS は、SET オプションに対する TEST NEGATIVE 応答、RESET 応答、または INVALID 応答を受信すると、そのセッションを終了します。

TEST POSITIVE 以外の任意のアクション・コードを戻す時、SLU P システムは、STSN 応答を使用して、当システムに該当するシーケンス番号を戻すことになっています。問題判別時およびデバッグ時には、このシーケンス番号が役立つことがあります。IMS は戻り値を使用しないので、どの値でも戻すことができます。シーケンス番号が受け入れ可能ではないことを示すためには TEST NEGATIVE 応答だけで十分です。

IMS は、STSN 応答を受信すると、ワークステーションにデータ通信開始 (SDT) コマンドを送信します。SDT コマンドによって、通常メッセージ伝送を開始することができます。IMS は SET AND TEST オプションへの TEST NEGATIVE 応答を受信した場合、SDT コマンドを送信し、最終リカバリー可能メッセージとそれに続く他のすべての使用可能出力を再送します。IMS は SET AND TEST オプションへの TEST POSITIVE 応答を受信した時に、SDT コマンドを送信し、最終リカバリー可能メッセージをデキューし、他のすべての使用可能出力を送信します。

IMS から受信した STSN コマンドの内容と SLU P システムおよび VTAM の結果のアクションは、以下の表に要約してあります。

表 176. シーケンス番号の設定およびテスト (STSN) の要約

リモート・システム番号のアクション・コード	VTAM 番号のアクション・コード	コントローラー・シーケンス番号 963 ページの『1』	VTAM シーケンス番号 963 ページの『1』	STSN 受信時のリモート・システム・アクション	STSN 受信時の VTAM アクション
01 設定		IMS が受信した最終インバウンド同期点メッセージ。		コントローラー・シーケンス番号フィールドを STSN のコントローラー・フィールドで指示された値に設定する。	VTAM の値を STSN のコントローラー・フィールドで指示された値に設定する。
	01 設定		IMS が送信し、ワークステーションが応答する、最終アウトバウンド同期点メッセージ。	ホスト・シーケンス番号フィールドを STSN の VTAM フィールドで指示された値に設定する。	VTAM の最終割り当て値を STSN の VTAM フィールド内の値に設定する。
	11 設定およびテスト		IMS が送信する最終リカバリー可能メッセージ。ワークステーションからの応答は受信されない。	ホスト・シーケンス番号フィールドを STSN の VTAM フィールドで指示された値に設定する。	VTAM の最終割り当て値を STSN の VTAM フィールド内の値に設定する。

#### 注記:

1. コントローラーおよび VTAM のシーケンス番号フィールドには、任意の値を入れることができます。STSIN コマンドに対する SLUP システムの有効な応答と、その結果として実行される IMS アクションを、以下の表に要約します。

表 177. アクション・コードがリモート・システム番号の場合の STSN 応答の要約

リモート・システムのアクション・コード値	必要なアクション・コード <a href="#">963</a> ページの『1』:	応答受信時の IMS アクション
01 設定	01 テスト肯定	ステーションに SDT を送信する。

#### 注記:

1. 応答が上記以外のものである場合は IMS はセッションを終了します。

表 178. アクション・コードが VTAM 番号の場合の STSN 応答の要約

VTAM 番号のアクション・コード値	必要なアクション・コード <a href="#">963</a> ページの『1』:	応答受信時の IMS アクション
01 設定	01 テスト肯定	ステーションに SDT を送信する。
11 設定およびテスト	01 テスト肯定	最終リカバリー可能メッセージをデキューし、ステーションに SDT を送信する。
11 設定およびテスト	11 テスト否定	ステーションに SDT を送信し、最終リカバリー可能メッセージを再送する。

#### 注記:

1. 応答が上記以外のものである場合は IMS はセッションを終了します。

## IMS からの出力の延期

コントローラー・アプリケーション・プログラムで IMS からそれ以上のどんな出力も受信したくない場合、あるいは受信できない場合に、プログラムは IMS に、VTAM チェーン終了時静止 (QEC) コマンドを送信することができます。IMS は DR1 および VTAM 静止完了 (QC) コマンドを戻します。IMS は、VTAM 解放静止 (RELQ) コマンドを受信するまでワークステーションに出力をそれ以上送信しません。

### このタスクについて

また、ワークステーションは、IMS に VTAM **LUSTATUS** コマンド または **SIGNAL** コマンドを送信して、IMS からの出力を延期することができます。

## セッション終了

セッション終了によって、ワークステーションは VTAM アプリケーション・プログラムとの現在の論理接続から解放されます。これで、ワークステーションが他の VTAM アプリケーションとのセッションに使用可能になります。あるいは通信を全面的に終了することができます。

セッション終了には、順序正しい終了および即時終了の 2 タイプがあります。

#### 定義:

- 順序正しい終了では、ワークステーションは、セッションが終了する前に、通常処理を完了することができます。
- 即時終了では、ワークステーションはセッションを 無条件に終了することになります。

以下に、セッション終了を呼び出すことができるものを示します。

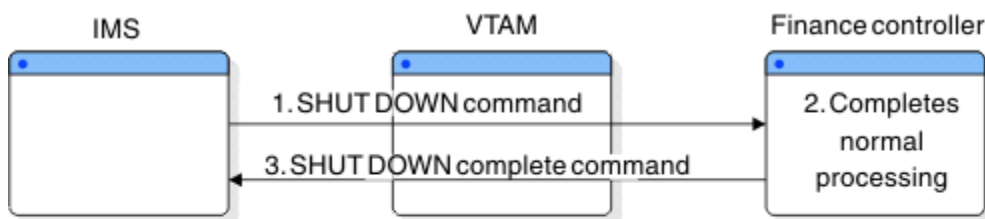
IMS マスター端末オペレーター



## VTAM ネットワーク・オペレーター ワークステーション

以下の図は、この2つのタイプのセッション終了処理を要約したものです。

### Orderly termination



### Immediate termination

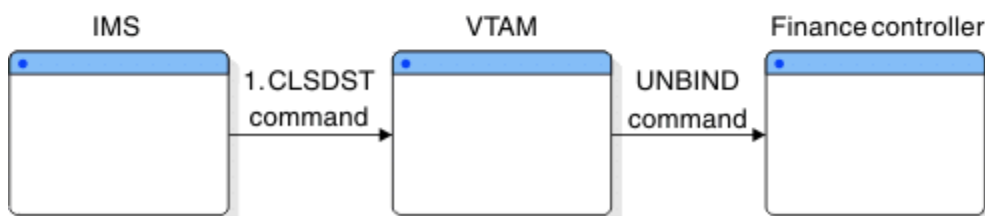


図 157. 終了処理

セッション終了のための特定のプロシージャーは、各インストール・システムごとに判別する必要があります。プロシージャーの開発にあたっては、セッション終了処理の要件を承知しておいてください。

## 順序正しい終了

ネットワークの順序正しい終了は、IMS マスター端末オペレーターが、**QUIESCE** パラメーターを指定した IMS コマンド **/CHECKPOINT FREEZE**、**/CHECKPOINT PURGE**、または **/CHECKPOINT DUMPQ** を使用して呼び出します。

### このタスクについて

**QUIESCE** コマンドはネットワークのシャットダウン処理を開始する **/CHECKPOINT** パラメーターです。**QUIESCE** を指定した場合、IMS は、すべてのワークステーションに VTAM シャットダウン (SHUTD) コマンドを送信し、すべてのワークステーションが正常処理を完了し VTAM シャットダウン完了 (SHUTC) コマンドを戻すまで待機します。

すべてのワークステーションがシャットダウンの完了を示した場合、IMS は、チェックポイント処理を実行してから VTAM CLSDST マクロ命令を発行します。CLSDST により、VTAM はすべてのワークステーションに **UNBIND** コマンドを送信します。このコマンドは IMS とのセッションからワークステーションを解放します。コントローラーはワークステーションからのそれ以降のデータ伝送を禁止します。

シャットダウン処理中 (VTAM シャットダウン完了コマンドから **CLEAR** コマンドまでの間)、システムのワークステーションは、シャットダウン処理の結果生じるすべての IMS 出力に対して準備済みである必要があります。

順序正しい終了の処理中に、IMS マスター端末オペレーターは、順序正しい終了処理の完了を待たずにネットワークを無条件に終了することができます。即時終了を呼び出すことによって、ネットワークを終了することができます。



## 即時終了

IMS マスター端末オペレーター、VTAM ネットワーク・オペレーター、またはワークステーション・ユーザーは、SLU P ワークステーションの即時終了を呼び出すことができます。

### このタスクについて

IMS マスター端末オペレーターは、**FREEZE** パラメーター、**PURGE** パラメーター、または **DUMPQ** パラメーターを指定した (ただし、**QUIESCE** パラメーターは指定しない) **/CHECKPOINT** コマンドを使用して、ネットワークの即時終了を呼び出します。IMS は VTAM CLSDST マクロ命令を発行します。CLSDST により、VTAM はすべてのワークステーションに **UNBIND** コマンドを発行します。このコマンドは IMS とのセッションからワークステーションを解放します。コントローラーは、ワークステーションからのそれ以降のデータ伝送をすべて禁止します。

ワークステーションまたはネットワークの一部を選択的に終了するために、IMS マスター端末オペレーターは、IMS **/CLSDST** コマンドまたは **/STOP** コマンドを使用することができます。**/CLSDST** コマンドおよび **/STOP** コマンドにより、IMS は指定のワークステーションに VTAM CLSDST マクロ命令を発行します。また、**/STOP** コマンドは、ワークステーションに **/START** コマンドが発行されるまで、それ以降のセッションが設定されないようにします。

z/OS VTAM ネットワーク・オペレーターは、ワークステーションを即時に終了するために **VARY** コマンドを使用します。ワークステーション・エラーが入出力操作の完了を妨げているセッションを終了するために、z/OS VTAM ネットワーク・オペレーターによる **VARY** コマンドを使用した介入が必要になる場合があります。

ワークステーションは、IMS とのセッションを終了するために VTAM セッション制御コマンドを使用します。通常の処理環境において、ワークステーション側から IMS とのセッションを終了する場合は、そのワークステーションから VTAM シャットダウン要求コマンドを送信する必要があります。IMS は、そのワークステーションに対して現在進行中のすべての入力または出力を完了し、その後 VTAM CLSDST マクロ命令を発行します。

ワークステーションがリカバリー不能なエラー状態を検出したためにセッション終了が望ましい場合は、そのワークステーションから VTAM セッション終了コマンドを送信することができます。VTAM は、そのワークステーションをセッションから解放し、IMS に対して適宜に通知します。

## IMS ネットワークのシャットダウン (SLU P)

IMS ネットワークのシャットダウンには、IMS をシャットダウンする場合とシャットダウンしない場合があります。

### このタスクについて

IMS **/CHECKPOINT** コマンドは、ネットワークの終了および IMS のシャットダウンを呼び出すために使用します。使用する **/CHECKPOINT** の形式によって、ネットワーク終了を即時に行うか、あるいはワークステーション処理が完了するまで待つかが決まります。

#### **/CHECKPOINT FREEZE|DUMPQ|PURGE QUIESCE**

IMS のシャットダウン前に、すべてのワークステーションは正常処理を完了することができます。

#### **/CHECKPOINT FREEZE|DUMPQ|PURGE**

すべてのワークステーションに対して即時セッション終了が行われます。

### 関連タスク

[965 ページの『即時終了』](#)

IMS マスター端末オペレーター、VTAM ネットワーク・オペレーター、またはワークステーション・ユーザーは、SLU P ワークステーションの即時終了を呼び出すことができます。

## SLU P メッセージ

SLU P システムに対する IMS サポートの主な機能は、ワークステーションのデータの受信および伝送を行うこと、およびデータが確実に正しく処理されるようにすることです。

入出力メッセージは次のデータ・タイプから構成することができます。

- IMS トランザクション
- IMS メッセージ通信
- IMS コマンド
- VTAM コマンドと標識
- メッセージ形式サービス (MFS) 制御要求

IMS は、上記メッセージのいずれの送受信時にも 非図形文字を処理することができます。

### 関連概念

447 ページの『IMS 非グラフィック・メッセージ・データに対するセンシティブィティ』

以下のトピックでは、ユーザーが IMS メッセージの中で非グラフィック・データを送受信しようとする場合の、特定の文字に対する IMS センシティブィティについて説明します。

## 送信 / 受信とブラケット・プロトコル

VTAM で定義される方向変換標識 およびブラケット・プロトコルは、同期伝送の開始と終了を指定し、さらにそのような伝送のフローを制御するために使用します。

キューに入れられた入出力処理のために、IMS は、複数の関連入力または複数の関連出力をサポートしません。イン・ブラケット状態において発生する可能性があるものを以下に示します。

- 1つの入力メッセージ (コード化 BB/EB)
- 1つの出力メッセージ (コード化 BB/EB)
- 1つの入力メッセージと 1つの出力応答 (入力コード化 BB/CD および出力コード化 EB)

入力メッセージおよび出力応答は次の場合にだけ関連します。

- 高速機能トランザクション
- 端末応答モード・トランザクション
- 会話型トランザクション

他のトランザクション・タイプの場合には、入力メッセージと出力応答との関連付けがされている場合とされていない場合があります。

### 関連概念

970 ページの『入力ブラケット付けプロトコル』

IMS とセッション中のワークステーションは、チェーン・メッセージの最初の伝送 (複数回伝送の場合) および非チェーン・メッセージの各伝送の両方で、開始ブラケット (BB) または開始ブラケットとブラケット終了 (EB) の両方を指示する必要があります。

977 ページの『出力ブラケット付けプロトコル』

出力メッセージの場合、IMS は、ブラケット開始とブラケット終了を、単一セグメント出力メッセージと複数セグメント・メッセージの最初の伝送で指定します。

## 第 52 章 SLU P メッセージ・プロトコル

VTAM コマンドおよび標識と MFS 制御要求の伝送は、1 回の伝送で行わなければなりません。IMS トランザクション、コマンド、およびメッセージ通信の伝送は、1 回または複数回の伝送で行うことができます。

ワークステーションの伝送バッファが送信するすべてのデータを保持するのに十分な大きさでない場合には、1 つのメッセージにつき複数回の伝送が必要になります。複数回伝送は、ワークステーション・ユーザーがデータを分割したい場合にも使用することができます。複数回伝送は、チェーン・メッセージの概念を通じて論理的に関係づけられます。複数回伝送メッセージの各伝送は、チェーン内の位置 (つまり、チェーンの先頭、中間チェーン、チェーンの最後) によって識別されます (971 ページの『出力メッセージ』の図を参照)。

チェーン入力メッセージを使用している SLU P システムにおいては、チェーン標識を使用して、伝送のチェーン位置を指定します。この標識は非チェーン・メッセージに設定してはなりません。チェーン標識は、先頭チェーン伝送でオンにし、各中間チェーン伝送でオフにし、そして最終チェーン伝送で再びオンにしなければなりません。

MFS が定義されていれば、入力メッセージを MFS によって処理することができます。入力フォーマット設定の場合には、MFS は可能な入力コンポーネントを区別しませんが、4701/4702 アプリケーション・プログラムからすべての入力がされることを想定しています。

### 入力機能管理ヘッダーの一般形式 (金融機関)

IMS トランザクション、メッセージ通信、およびコマンドは、標準 IMS 形式 (トランザクション・コード、論理端末名、またはコマンド verb の後にブランクを挟んでテキストが続く形式) になっています。ただし、MFS または物理端末入力編集ルーチンを使用して、メッセージを編集して標準形式にした場合は、例外です。

IMS トランザクションの最初の伝送 (1 回だけの伝送の場合もある) に、可変長の FM ヘッダーがあります。ワークステーションに MFS が定義されていない場合には、ヘッダーは無視されます。MFS が定義されている場合には、ヘッダーに MFS 制御要求または MID 名のいずれかを入れることができます。MFS 制御要求の場合には、FM ヘッダーの最初の 2 バイトだけが必要となります。

#### バイト 0

長さバイトを含むヘッダー長 (2 進)。バイト 0 および 1 が MFS 制御要求です。

#### バイト 1

メッセージ記述 (2 進)。

#### バイト 2

MID 名長 (2 進)。

#### バイト 3-10

MID 名 (1-8 バイト) (EBCDIC)。

IMS はヘッダーを使用しますが、そのヘッダーはメッセージを MPP または MFS に送信する前に除去します。IMS がヘッダーを含むメッセージを受信すると、IMS はそのメッセージの長さを調べます。メッセージ長が 2 バイトであれば、IMS はそのメッセージが MFS 制御要求であると仮定します。MFS 制御要求は、メッセージ記述 (バイト 1) で指示されます。

**関連情報:** PAGEREQ 機能の使用の詳細については、「IMS V15 アプリケーション・プログラミング API」を参照してください。

### 入力メッセージ記述子バイト (金融機関)

メッセージ長が 2 バイトを超えていれば、IMS はメッセージ記述を調べて、MFS MID 名が存在するかどうかを判別します。この MID 名によって指定された形式記述がメッセージのフォーマット設定に使用されます。

メッセージ記述子バイトの形式は次のとおりです。

**ビット 0**

予約済み

**ビット 1**

ページ先送り要求 (NEXTTPP)

**ビット 2**

メッセージ先送り要求 (NEXTMSG)

**ビット 3**

メッセージ先送り保護要求 (NEXTMSGP)

**ビット 4**

次の論理ページ要求 (NEXTLP)

**ビット 5**

予約済み

**ビット 6**

この入力メッセージのフォーマット設定には、MID 名長フィールドと MID 名フィールドで指示されている形式記述を使用する必要があります。

**ビット 7**

予約済み

ビット 1、2、3、および 4 (MFS 制御要求) は相互に排他的であり、メッセージ長が 2 バイトの場合にだけ検査されます。MFS MID 名によって入力メッセージをフォーマット設定する場合には、ビット 6 をオンにしなければなりません。ビット 6 が MFS MID 名が存在することを示している場合には、ビット 0-5 はゼロでなければなりません。

## 入力機能管理ヘッダーの一般形式 (SLU P)

IMS トランザクション、メッセージ通信、およびコマンドは、標準 IMS 形式 (トランザクション・コード、論理端末名、またはコマンド verb の後にブランク を挟んでテキストが続く形式) になっています。

メッセージが標準外形式である場合には、MFS または物理端末入力編集ルーチンを使用して、メッセージを編集して標準形式にすることができます。IMS トランザクションの最初の伝送 (1 回だけの伝送の場合もある) に、可変長の FM ヘッダーがあります。MFS が端末に定義されていない場合には、ヘッダーは無視されます。MFS が定義されている場合には、ヘッダーに MFS 制御要求または MFS フォーマット設定の呼び出しに使用されるオプション MFS フィールドを入れることができます。これらの場合に IMS はヘッダーを使用しますが、そのヘッダーはメッセージ MPP または MFS に渡す前に除去します。ヘッダーの指定や形式が誤っていると、セッションが終了する可能性があります。SLU P が使用するヘッダー・タイプは X'42' です。

入力 FM ヘッダーの形式は次のとおりです。

**バイト 0**

(2 進の) 長さバイトを含むヘッダー長 (先頭バイト)

**バイト 1**

ヘッダー・タイプ (X'42' でなければならない)

**バイト 2**

メッセージ記述子 1 (フラグ・バイトは 2 進)

**バイト 3**

メッセージ記述子 2 (フラグ・バイトは 2 進)

**バイト 4**

入力コンポーネント識別 (2 進)

オプション MFS フィールド

**DPM の場合****バイト 5 および 6**

バージョン ID (2 進) - バイト 3 のビット 0 がオンの場合。

## バイト 7 から 15

長さバイト (1 バイト) を含む MID 名長とその後に続く MID 名 (1 バイトから 8 バイト)。バイト 3 のビット 0 がオフである (FM ヘッダー にバージョン ID が入っていない) 場合は、MID 名長と MID 名はバイト 5 から 13 に入ります。

## SCS の場合

### バイト 3-13

長さバイト (1 バイト) を含む MID 名長 (2 進) と その後に続く MID 名 (1-8 バイト)。

IMS がヘッダーを含むメッセージを受信すると、IMS はそのメッセージの長さを調べます。メッセージ長が 5 バイトであれば、IMS はそのメッセージが MFS 制御要求であると仮定します。特定の MFS 制御要求は、メッセージ記述子 1 (バイト 2) に示されます。メッセージ長が 5 バイトを超えていれば、IMS はメッセージ記述バイトを調べて、MFS バージョン ID と MID 名が存在するかどうかを判別します。両方が存在する場合には、バージョン ID が MID 名長の直前になければなりません。(SCS1 の場合には、バージョン ID は存在しません。) MID 名によって指定された形式記述はこれに伴う入力メッセージのフォーマット設定に使用され、バージョン ID は形式記述レベルの妥当性検査に使用されます。FM ヘッダーがないか、バージョン ID がないか、またはバージョン ID がゼロの場合には、MFS は形式記述レベルに関する妥当性検査を迂回します。

## 関連概念

[DPM 形式のバージョン識別機能 \(アプリケーション・プログラミング API\)](#)

## 入力メッセージ記述子バイト (SLUP)

メッセージ記述子 1 (FM ヘッダーのバイト 2) の形式は次のとおりです。

### ビット 0

予約済み

### ビット 1

ページ先送り要求 (NEXTPP)

### ビット 2

メッセージ先送り要求 (NEXTMSG)

### ビット 3

メッセージ先送り保護要求 (NEXTMSGP)

### ビット 4

次の論理ページ要求 (NEXTLP)

### ビット 5

予約済み

### ビット 6

MID 名長フィールドと MID 名フィールドによって指示された形式記述を使用して、この入力メッセージをフォーマット設定する必要があります

### ビット 7

予約済み

ビット 1、2、3、および 4 (MFS 制御要求) は相互に排他的です。指定された MFS MID 名によって入力メッセージをフォーマット設定する場合には、ビット 6 をオンにしなければなりません。ビット 6 が MID 名が存在することを示している場合には、ビット 0-5 はゼロでなければなりません。

メッセージ記述子 2 (FM ヘッダーのバイト 3) の形式は次のとおりです。

### ビット 0

MFS DPM のバージョン ID。FM ヘッダーのバイト 5 および 6 にバージョン ID が入っている場合には、ビット 0 をオンにしなければなりません。

### ビット 1-7

予約済み



## 入力コンポーネント識別 (SLU P)

入力コンポーネント識別は、選択された入力コンポーネントに対して使用されます。コンポーネント識別は、0 から 4 までの値とすることができます。この値は、IMS システム定義時に NAME マクロで指定した ICOMPT、あるいは ETO ユーザー記述子で指定した ICOMPT と一致させます。

コンポーネント識別がゼロの場合、または FM ヘッダーが送信されない場合には、入力はコンポーネント 1 と関連付けられます。さらに、IMS は入力コンポーネント識別を使用して、コンポーネント出力保護をリセットします。値 0 (ゼロ) が送信された場合には、端末のすべてのコンポーネントについて出力保護がリセットされます。値 1-4 が送信された場合には、指定されたコンポーネントだけについて出力保護がリセットされます。

### 関連概念

941 ページの『コンポーネントの定義』

IMS は、ワークステーションを 1 つの物理端末と見なします。ワークステーションを複数の装置で構成する場合は、おのおのの物理装置を、そのワークステーションのコンポーネントとして定義する必要があります。

### 関連タスク

71 ページの『拡張端末オプションの管理』

IMS 拡張端末オプション (ETO) を使用すると、VTAM 端末とユーザーをシステム定義時に事前定義しなくても、IMS に動的に追加できるようになります。

## 入力ブラケット付けプロトコル

IMS とセッション中のワークステーションは、チェーン・メッセージの最初の伝送 (複数回伝送の場合) および非チェーン・メッセージの各伝送の両方で、開始ブラケット (BB) または開始ブラケットとブラケット終了 (EB) の両方を指示する必要があります。

ブラケット終了標識は、IMS とワークステーションの両方をブラケット間でコンテンツン状態にします。ブラケット開始標識だけが送信された場合には、ワークステーションは次の各時点のそれぞれにおいて、方向変換 (CD) 標識を送信する必要があります。

- チェーン・メッセージの最終伝送 (複数回伝送の場合)
- 非チェーン・メッセージの各伝送

IMS は複数の入力メッセージを相互に関連付けることができないので、CD 標識が必要になります。CD 標識は、指定されていない場合は上記のとおり黙指定されます。

BB と CD が指示された場合には、IMS はワークステーションに対する次の出力で EB を指示します。BB だけが指定されているチェーン入力の受信中にエラーを検出した場合は、IMS は EB と IMS エラー・メッセージを戻します。

**制約事項:** SLU P 端末の場合には、各入力チェーンに CD または EB 標識が伴わなければなりません。そうでないと、セッションは終了します。

### 関連概念

971 ページの『出力メッセージ』

IMS からの出力メッセージは、いくつかの異なるタイプのいずれかとなります。

### 関連資料

982 ページの『エラー処理』

以下のトピックでは、伝送またはプロトコル・エラーから生じる障害を処理するために IMS によって使用され、コントローラーまたはコントローラー・アプリケーション・プログラムに必要なプロシージャーについて説明します。

## 金融機関ワークステーションに対する MFS 入力フォーマット設定の活動化

MFS を使用する場合、入力メッセージはメッセージ記述子とフォーマット設定記述子により処理することができます。

### このタスクについて

IMS が入力メッセージを受信すると、そのメッセージに MID 名が伴っていなければ、IMS 基本編集が実行されます。MID 名の指定は、MID 名を入力機能管理ヘッダーまたは必要な MFS エスケープ文字 (//) を含むメッセージ・テキストの始めに組み込むことによって行うことができます。MID 名がある時には、指定された MID および関連した装置入力形式 (DIF) を使用して、MFS がメッセージを編集します。MID 名は、オペレーターまたはリモート・アプリケーション・プログラムのいずれかによって提供されます。

## 出力メッセージ

IMS からの出力メッセージは、いくつかの異なるタイプのいずれかとなります。

IMS からのさまざまなタイプの出力メッセージには、以下のものがあります。

- リカバリー可能またはリカバリー不能入力トランザクションに対するデータ応答。
- IMS コマンドに対するデータ応答。
- メッセージ通信
- VTAM 標識
- IMS システム・メッセージ
- ブロードキャスト・メッセージ
- ワークステーションをブラケット間およびコンテンション状態に戻すためのブラケット終了を含むヌル (長さ = 0) ・メッセージ

IMS からワークステーションへのすべてのメッセージは、1 回の伝送で送信されます。ただし、次の場合にはその限りではありません。

- MPP、メッセージ通信、コマンド、または MFS が複数セグメントの出力メッセージを提供する場合。
- ワークステーションの読み取りバッファが小さすぎて MPP または MFS の提供する単一セグメント出力メッセージを保持できない場合。
- ブロードキャスト・メッセージが複数セグメントである場合。

メッセージの各セグメントは、可能な時、すなわち単一チェーンが指示された時には常に、1 回の伝送で送信されます。ただし、メッセージ・セグメントが受信側ワークステーションの読み取りバッファのサイズを超えている場合には、完全なセグメントが送信されるまで、セグメントは必要な伝送回数に分割されます。複数セグメント出力メッセージは、チェーン入力メッセージと同様に処理されます。すなわち、各セグメントは、チェーンの先頭、中間チェーン、またはチェーンの最後として正しく識別されます。

IMS メッセージ・キュー・データ・セットおよびワークステーション出力バッファのサイズを定義する際には、注意が必要です。不適切な指定をすると、IMS から複数セグメント出力チェーンが送られてくる可能性があります。出力バッファ・サイズが小さすぎると、大きい IMS セグメントは複数回伝送になる可能性があります。メッセージ・キュー・データ・セット・サイズが小さすぎると、大きいメッセージ処理プログラム (MPP) 挿入に対して複数の IMS セグメントが作成される可能性があります。

**関連情報:** MSGQUEUE および TERMINAL マクロの詳細については、「IMS V15 システム定義」を参照してください。

**定義:** MFS ページング出力 とは、ディスプレイ装置宛てのメッセージの各物理ページのことです。これは、あたかも完全なメッセージであるかのように送信されます。



以下の図は、IMS に送信される伝送、MPP によって生成されるセグメント、および IMS によって伝送されるセグメントの間の関係を示しています。

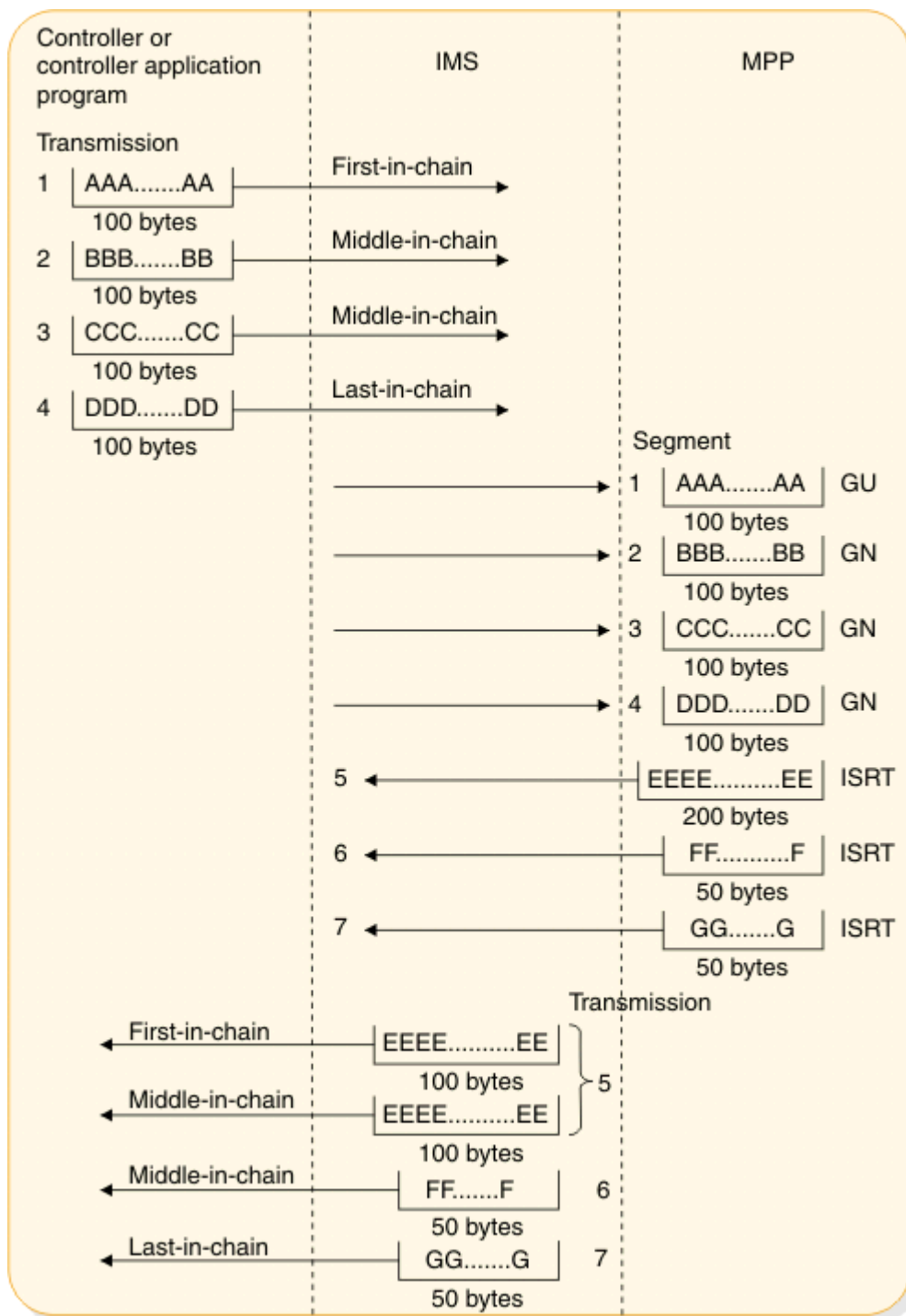


図 158. 金融機関通信システムと IMS MPP 間のチェーン・メッセージ対話

メッセージを正常に受信した後、SLU P システムは、タイプ読み取りフィールド (SMSCRT)、フラグ読み取りフィールド (SMSCRF)、およびオプションのフラグ読み取りフィールド拡張 (SMSCRE) を調べて、受信したメッセージのタイプと特性を判別しなければなりません。

**関連資料:** これらのフィールドの詳細については、「IBM 4700 Finance Communication System: Controller Programming Library, Volume 3: Communication Program」を参照してください。

非 MFS フォーマット設定出力セグメントが複数回伝送で送信され、そのワークステーションに SCAN オプションが定義されている場合には、IMS は 4701/4702 装置の選択制御列、位置制御列、または書き込み

制御列が伝送の前後で分割されないようにします。透過書き込み制御列が2回の伝送に分割されると、最初の伝送のデータ長が変更され、適切な書き込み制御文字が2回目の伝送の始めに挿入されます。

IMS システムの定義時または ETO ログオン記述子に、NOSCAN オプションが指定された場合、非 MFS フォーマット設定出力で装置制御列の編集は行われません。出力セグメントが MFS によってフォーマット設定された場合には、装置制御列 (選択および位置) 編集が常に行われます。さらに、セグメントの各伝送 (最後の伝送を除く) が 3 文字からなる空水平位置列で終了し、物理 SLU P システム出力装置の自動改行機能を抑制します。

## MFS 分散表示管理出力 (SLU P)

IMS は、出力メッセージ全体を 1 つ以上の関連伝送から成る単一チェーンとして送信します。ページング・オプション OPTIONS=DPAGE または OPTIONS=PPAGE で定義された装置形式を使用している MFS DPM 出力の場合には、IMS は、各論理または表示ページを、あたかも完全なメッセージであるかのように、1 つ以上の関連伝送からなる単一チェーンで送信します。

ページング・オプション OPTIONS=DPAGE または OPTIONS=PPAGE で定義された装置形式を使用している MFS DPM 出力の場合には、IMS は、出力機能管理ヘッダーを、メッセージの各チェーン (論理または表示ページ) の最初の伝送 (1 回の伝送の場合もある) の一部として送信します。残りの伝送には、データ・フィールド (DFLD) が定義されたレコード長まで入っています。この機能管理ヘッダーには、DPM バージョン ID、DPAGE または PPAGE 名、および MID 名 (指定された場合) が入っています。

ページング・オプション OPTIONS=DPAGE または OPTIONS=PPAGE の DPM 形式仕様に用紙リテラルが定義されると (DEV ステートメントの FORMS= パラメーター)、IMS は、用紙リテラルのみを含む機能管理ヘッダーからなる単独チェーン伝送を使用して、最初の論理または表示ページを先に送信します。バージョン ID、DPAGE 名、および PPAGE 名はいずれも提供されません。出力コンポーネントは各出力チェーンの終わり方で保護されます。これにより、IMS は、適切な入力メッセージまたは制御要求が受信されるまで、後続の出力、すなわち 論理または表示ページ (チェーン) を送信しません。

ページング OPTIONS=MSG で定義された装置形式を使用している MFS DPM 出力の場合には、IMS はメッセージ全体を 1 つ以上の関連伝送から成る単一チェーンとして送信します。機能管理ヘッダーは最初の伝送 (1 回の伝送の場合もある) として送信され、このヘッダーには、DPM バージョン ID、形式名、および MFS 形式仕様で定義された任意の MID 名または用紙リテラルがあります。この場合には、非 DPM 出力と同様、コンポーネント保護は、宛先コンポーネントが PROGRAM2 として定義されている場合にだけ提供されます。

DPM の場合には、MFS に対して装置タイプ記号名が DPM-Xn として指定されます。この X には A または B を設定できます。DPM-An 形式の装置タイプ記号名を使用することにより、SLU P として定義された論理装置に対する TERMINAL マクロ、または ETO ログオン記述子でメッセージ・フォーマット設定が指定されます。DPM-Bn 装置タイプ記号名は、システム間連絡機能に対して定義された論理装置を参照します。

## 出力機能管理ヘッダーの一般形式 (金融機関)

IMS は、送信されるすべてのメッセージ (IMS 空メッセージと VTAM コマンドおよび標識を除く) について、チェーンの先頭または単独チェーンの伝送にヘッダーを付加します。

このヘッダーは後に続くメッセージのタイプを記述し、場合によっては、そのメッセージを受信する出力コンポーネントも示します。このヘッダーにはオプションの MFS 情報を入れることができます。

出力機能管理ヘッダーの長さは、3-29 バイトです。フォーマットは次のとおりです。

### バイト 0

長さバイトを含むヘッダー長 (2 進)

### バイト 1

メッセージ記述 (2 進)

### バイト 2

出力コンポーネント識別 (2 進)

### バイト 3-28

MFS データ (長さバイト: 2 進; 名前: EBCDIC)

## 出力メッセージ記述子バイト (金融機関)

メッセージ記述子バイト (バイト 1) の各ビットは、オンの場合、それぞれ次の意味をもちます。

### ビット 0

このメッセージは、IMS システム・メッセージまたはブロードキャスト・メッセージです。

### ビット 1

このメッセージは MFS によってフォーマット設定されています。

### ビット 2

MFS システム制御域 (SCA) がこのメッセージについて装置アラームを要求しています (コントローラー・アプリケーション・プログラムは、端末のライトのオンなど適切な処置を取ることができます。)

### ビット 3

このメッセージは、キュー解除されたメッセージです (特定の宛先がない)。ビット 0 も設定されます。

### ビット 4

予約済み。

### ビット 5

このメッセージは、高速機能リカバリー可能出力です。

### ビット 6

MFS データ・フィールドに MID 名があります。

### ビット 7

MFS データ・フィールドに FORMS 名があります。

## 出力コンポーネント ID バイト (金融機関)

出力コンポーネント識別 (バイト 2) には X'01' から X'04' までの値が入っています。

この値は、このメッセージの宛先である端末コンポーネントを識別します。出力コンポーネント ID は、IMS システム定義時に、ETO ユーザー記述子で、またはサインオン出口ルーチン (DFSSGNX0) および出力作成出口ルーチン (DFSINSX0) を使用して、割り当てられます。ID は、メッセージの送信側 (IMS またはメッセージ処理プログラムのいずれか) によって指定されます。

### 関連概念

941 ページの『出力コンポーネント選択』

IMS システム定義により、1 台のワークステーションに最大 4 つの出力コンポーネントをもたせることができます。

## MFS データ・バイト (金融機関)

バイト 3 からヘッダーの終わりまでのバイトには、MFS データを入れることができる上に、1 つまたは 2 つのフィールドを組み込むことができます。メッセージ記述子バイトのビット 6 および 7 は、これらのフィールドが組み込まれているかどうかを示します。

最初のフィールド (存在する場合) は 2-9 バイトの長さで、このフィールドには長さバイトを含む 1 バイトの長さ標識と、1-8 バイトの MFS MID 名が入ります。この MID 名フィールドは、次の入力メッセージに使用する MID の名前がメッセージ出力記述に入っている場合に存在し、コントローラー・プログラムによって IMS に戻されます。第 2 のフィールド (存在する場合) は 2-17 バイトの長さで、このフィールドには長さバイトを含む 1 バイトの長さ標識と、1-16 バイトの FORMS 名が入ります。FORMS 名は、このメッセージに必要な特殊用紙を識別します。メッセージを印刷する前に、アプリケーション・プログラムは、用紙が適所にあること、およびページ・サイズと用紙位置決めが設定されていることを確認する必要があります。

## 出力機能管理ヘッダーの一般形式 (SLU P)

IMS は、送信されるすべてのメッセージ (IMS 空メッセージと VTAM コマンドおよび標識を除く) について、チェーンの先頭または単独チェーンの伝送にヘッダーを付加します。

このヘッダーは後に続くメッセージのタイプを記述し、場合によっては、そのメッセージを受信する出力コンポーネントも示します。また、このヘッダーには、オプションの MFS 情報を入れることもできます。SLU P が使用するヘッダー・タイプは X'42' です。

出力機能管理ヘッダーの長さは、5-42 バイトです。形式は次のとおりです。

### バイト 0

長さバイトを含むメッセージ・ヘッダー長 (2 進)

### バイト 1

ヘッダー・タイプ (X'42')

### バイト 2

メッセージ記述子 1 (フラグ・バイトは 2 進)

### バイト 3

メッセージ記述子 2 (フラグ・バイトは 2 進)

### バイト 4

出力コンポーネント識別 (2 進)

### バイト 5-41

MFS データ (長さバイト: 2 進; 名前: EBCDIC)

## 出力メッセージ記述子バイト (SLU P)

メッセージ記述子 1 (バイト 2) の各ビットは、オンの場合に次の意味をもちます。

### ビット 0

このメッセージは、IMS システム・メッセージまたはブロードキャスト・メッセージです。

### ビット 1

このメッセージは MFS によってフォーマット設定されています。

### ビット 2

予約済み。

### ビット 3

このメッセージは、キュー解除されたメッセージです (特定の宛先がない)。ビット 0 も設定されます。

### ビット 4

予約済み。

### ビット 5

高速機能リカバリー可能出力

### ビット 6

MFS データ・フィールドに MID 名があります。

### ビット 7

MFS データ・フィールドに FORMS 名があります。

メッセージ記述子 2 (バイト 3) の各ビットは、オンの時に次の意味をもちます。

### ビット 0

この出力は、MFS DPM によってフォーマット設定されています。バージョン ID は存在する場合とそうでない場合があり、次の 1 つが MFS データ・フィールドに存在します。

- DPM 形式名
- 論理ページ名 (DPAGE ステートメント・ラベル)
- 表示名 (PPAGE ステートメント・ラベル)

### ビット 1

予約済み。

## ビット 2

このチェーンの終わりで、出力機能管理ヘッダーのバイト 4 で指示されている コンポーネントが保護されます。

## ビット 3-7

予約済み。

メッセージ記述子 2 のビット 2 がオンに設定されている場合には、IMS は、該当する入力が発生するまで、別の出力チェーンの送信を行いません。この該当する入力とは、入力メッセージ、MFS 制御要求、または READY-TO-RECEIVE (RTR) コマンドのいずれかです。

## 関連概念

949 ページの『拡張出力コンポーネントの保護 (SLU P)』

SLU P が NOBID オプションで定義されている場合、IMS は IBM 3270 情報表示システムの画面保護機能や金融機関コンポーネントの表示画面保護と類似した機能をサポートします。

## MFS データ・バイト (SLU P)

バイト 5 からヘッダーの終わりまでには、最大 4 つの MFS フィールドを入れることができます。フィールドの数は、メッセージ記述子 1 のビット 6 および 7 と、メッセージ記述子 2 のビット 0 の設定によって決まります。

ビット 0 がオンの場合には、最初のフィールド (存在する場合) は 2 バイトの MFS DPM バージョン識別です。このフィールドは、すべての MFS DPM フォーマット設定出力に存在します。このフィールドには、バージョン ID または値 0 (ゼロ) を入れることができます。どちらの場合においても、以下のフィールド (第 2、第 3、または第 4 のフィールド) のいずれか 1 つが存在します。

第 2 のフィールド (存在する場合) は 2 バイトから 9 バイトの長さで、このフィールドには 長さバイトを含む 1 バイトの長さ標識と、その後続く 1 バイトから 8 バイトの MFS MID 名が入ります。この MID 名フィールドは、次の入力メッセージに使用する MID の名前がメッセージ出力記述に入っている場合に存在し、コントローラー・プログラムによって IMS に戻されます。

第 3 のフィールド (存在する場合) は、MFS 装置形式のページング・オプションで定義される 2-9 バイトのデータ名です。

第 4 のフィールド (存在する場合) は 2 バイトから 17 バイトの長さで、このフィールドには 長さバイトを含む 1 バイトの長さ標識と、その後続く 1 バイトから 16 バイトのユーザー指定 MFS 用紙リテラルが入ります。用紙リテラルは、このメッセージに必要な特殊セットアップまたは用紙を識別します。用紙リテラルの受信時に端末で従うプロシージャは、ユーザーが定義する必要があります。

メッセージ形式名 (FMT ステートメント・ラベル) は、OPTIONS=MSG が指定されている場合に存在します。論理ページ名 (DPAGE ステートメント・ラベル) は、OPTIONS=DPAGE が指定されている場合に存在します。表示ページ名 (PPAGE ステートメント・ラベル) は、OPTIONS=PPAGE が指定されている場合に存在します。このフィールドは、DPM からのすべての出力 (メッセージ記述子 2 のビット 0) に現れます。

HDRCTL=FIXED または HDRCTL=VARIABLE オプションで MFS 装置形式を定義すると、DPM フォーマット設定出力の機能管理ヘッダーの内容と形式に影響が及ぶことがあります。HDRCTL=VARIABLE が指定されると、各 MFS ヘッダー・フィールドのサイズは可変となります。HDRCTL=FIXED が指定されると、次の MFS ヘッダー・フィールドが送信されて、最大サイズまで埋め込まれます。

- 出力ヘッダー内の MID 名フィールドは、最大長の 8 バイトになるまで後書きブランクで埋め込まれる。次の入力をユーザー提供のメッセージ出力記述によってフォーマット設定するための MID 名が指定されていない場合には、8 個のブランクが送信されます。
- メッセージ形式名は、最大長の 6 バイトになるまで埋め込まれる。論理ページ名または表示ページ名は、最大長の 8 バイトになるまで埋め込まれます。



## 出力ブラケット付けプロトコル

出力メッセージの場合、IMS は、ブラケット開始とブラケット終了を、単一セグメント出力メッセージと複数セグメント・メッセージの最初の伝送で指定します。

入力メッセージで開始ブラケットと方向変換が指定された場合は、IMS が送信する次の出力ではブラケット終了のみを指定します。開始ブラケットだけが指定されているチェーン入力を受信時に IMS がエラーを検出した場合は、IMS はブラケット終了と IMS エラー・メッセージを戻します。

高速機能、会話型、または応答モードのトランザクションの場合には、出力メッセージは直前の入力メッセージに対する応答です。その他の場合には、出力メッセージは直前の入力メッセージと関係することもあるかもしれませんが、そうでないこともあります。

IMS は単一チェーン出力メッセージ (FM ヘッダーなし、データ長=0) を送信します。このメッセージは、例外 DR2 を要求し、ただちに使用可能な他の出力がない時に FM ヘッダー・ブラケット終了を指示します。これは、次の時に行われます。

- "否認された端末応答モード"として IMS に定義されたワークステーションからの入力メッセージが、開始ブラケットと方向変換を指定している。
- "強制された端末応答モード"として IMS に定義されたワークステーションからの入力メッセージが、開始ブラケットと方向変換を指定していて、IMS トランザクション以外である。
- "トランザクション依存端末応答モード"として IMS に定義されたワークステーションからの入力メッセージが、開始ブラケットと方向変換を指定していて、応答モード・トランザクション以外である。

IMS は、例外応答センス・コード 0811 および 0812 の受信後にも、単一チェーン出力メッセージ (FM ヘッダーなし、データ長=0) を送信します。このメッセージは、例外応答 DR2 を要求し、即時に使用可能な出力がほかに 1 つもない場合は FM ヘッダー開始ブラケット / ブラケット終了を指示します。

### 関連資料

982 ページの『エラー処理』

以下のトピックでは、伝送またはプロトコル・エラーから生じる障害を処理するために IMS によって使用され、コントローラーまたはコントローラー・アプリケーション・プログラムに必要なプロシージャーについて説明します。

## SLU P に対する MFS 出力フォーマット設定の活動化

MFS 出力フォーマット設定は、出力メッセージに関連したメッセージ出力記述子 (MOD) がある時に行われます。

### このタスクについて

出力メッセージに MOD が関連付けられるのは、次のいずれかの場合です。

- MPP が、出力メッセージとともに、MOD の名前 (MOD 名) を提供する。
- その定義が MOD 名を出力フォーマット設定に指定したメッセージ入力記述子 (MID) によって入力メッセージが処理された。
- 出力メッセージが、MFS 編集を使用している装置からのメッセージ通信である。

出力メッセージに関連した MOD が存在しない場合、標準 IMS 出力編集が行われます。

ワークステーションに対する入力フォーマット設定が行われるのは、MID 名とメッセージそのものが IMS に送信された場合だけです。しかし、次の MID 名をメッセージ出力記述に指定することは可能です。IMS は入力の受信を直接制御することはできないので、適切な MID の使用を保証するためには、コントローラーの関与が必要です。したがって、ワークステーションは、IMS が出力メッセージをフォーマット設定している間に入力メッセージを送信することができます。IMS が出力を処理している時に送信された入力は、IMS が出力処理を完了するまで、VTAM によってキューに入れられます。次の MID 名が指定されている場合、VTAM によってキューに入れられた入力は、あたかも現行出力の結果であるかのように見えます。内部 MID 名を使用してメッセージをフォーマット設定すると、予測不能な結果が生じます。IMS は MID 名を入力メッセージの一部として送信することによって、これを回避します。

IMS は、次の MID の名前が指定されていることをコントローラーに通知し、要求された MID 名と出力メッセージをワークステーションへ送信します。そうすると、コントローラーはその MID 名を保管し、出力メッセージを表示し、次のオペレーター入力を読み取り、保管した MID 名をトランザクションに追加して、そのトランザクションを IMS に送信します。

## 応答要求 (金融機関)

IMS からの出力メッセージは、特定の応答を要求します。IMS コマンド、ブロードキャスト出力、メッセージ通信、および更新照会またはリカバリー可能照会として IMS に定義された非高速機能トランザクションは、非オペレーター論理ページ MFS 出力の最終物理ページまたは最終論理ページに応答する場合を除き、応答 DR2 を必要とします。

リカバリー不能照会またはオペレーター論理ページング MFS 出力に対する応答として IMS に定義されたトランザクションは、例外応答 DR2 を必要とします。MOD が PAGE=YES を指定していないメッセージの最終物理ページを除き、MFS ページング出力は、トランザクションがどのように定義されていても、例外応答 DR2 を要求します。なぜなら、各ページの後は、出力の別のページが許可される前に、明示 MFS 制御またはデータ入力が続かなければならないからです。そうでない場合は、そのメッセージはデキューされます。

高速機能出力メッセージは、トランザクション・タイプにかかわらず、送信されると例外応答 DR2 を要求します。ワークステーションからの次の入力メッセージまたは RTR コマンドにより、前の出力が受信されたこと、および IMS はそのメッセージをデキューしても構わないことが確認されます。

### 関連資料

978 ページの『入力応答要件』

以下の表は、ワークステーションと IMS 間におけるすべてのタイプのデータ伝送について、応答の要求に関する入力要件を要約したものです。

## 応答要求 (SLU P)

非オペレーター論理ページ MFS 出力の最終物理ページまたは最終論理ページに 応答する場合を除き、IMS からの出力メッセージは特定の応答を必要とします。IMS コマンド、ブロードキャスト出力、メッセージ通信、および更新照会またはリカバリー可能照会として IMS に定義された非高速機能トランザクションは、応答 DR2 を必要とします。

出力の別のページが認められたりメッセージがデキューされる前に、各ページの後に明示 MFS 制御またはデータ入力が続かなければならないので、リカバリー不能照会またはオペレーター論理ページング MFS 出力に対する応答として IMS に定義されたトランザクションは、例外 DR2 を必要とします。MOD が PAGE=YES を指定していないメッセージの最終物理ページを除き、MFS ページング出力は、トランザクションがどのように定義されていても、例外応答 DR2 を要求します。

### 関連資料

978 ページの『入力応答要件』

以下の表は、ワークステーションと IMS 間におけるすべてのタイプのデータ伝送について、応答の要求に関する入力要件を要約したものです。

## 入力応答要件

以下の表は、ワークステーションと IMS 間におけるすべてのタイプのデータ伝送について、応答の要求に関する入力要件を要約したものです。

表 179. メッセージ・タイプ別の入力応答要件

データ・タイプ	受け入れられる応答 (ACK オプション)	受け入れられる応答 (OPTACK オプション)
更新トランザクション	DR2 <sup>979</sup> ページの『1』 例外 DR1 または例外 DR2 <sup>979</sup> ページの『3』	DR1 または DR2 例外 DR1 または例外 DR2 <sup>979</sup> ページの『1』、 <sup>979</sup> ページの『2』



表 179. メッセージ・タイプ別の入力応答要件 (続き)

データ・タイプ	受け入れられる応答 (ACK オプション)	受け入れられる応答 (OPTACK オプション)
リカバリー可能照会トランザクション	DR2 <sup>979</sup> ページの『1』 例外 DR1 または例外 DR2 <sup>979</sup> ページの『3』	DR1 または DR2 例外 DR1 または例外 DR2 <sup>979</sup> ページの『1』、979 ページの『2』
リカバリー不能照会トランザクション	DR1 または DR2 例外 DR1 または DR2 <sup>979</sup> ページの『1』	DR1 または DR2 例外 DR1 または例外 DR2 <sup>979</sup> ページの『1』、979 ページの『2』
高速機能トランザクション <sup>979</sup> ページの『4』	N/A	DR1 または DR2 例外 DR1 または例外 DR2 <sup>979</sup> ページの『1』、979 ページの『2』
IMS メッセージ通信	DR2 <sup>979</sup> ページの『1』	DR1 または DR2 例外 DR1 または例外 DR2 <sup>979</sup> ページの『1』、979 ページの『2』
IMS コマンド	DR1 または DR2 例外 DR1 または DR2 <sup>979</sup> ページの『1』	DR1 または DR2 例外 DR1 または例外 DR2 <sup>979</sup> ページの『1』、979 ページの『5』
VTAM コマンド / 標識	DR1 <sup>979</sup> ページの『1』	DR1 <sup>979</sup> ページの『1』
SLU P システム MFS 制御要求	DR1 または DR2 例外 DR1 または DR2 <sup>979</sup> ページの『1』	DR1 または DR2 例外 DR1 または DR2 <sup>979</sup> ページの『1』

注:

1. 推奨または必須応答。
2. 高速機能ユーザーは常に、例外応答 DRx と一緒に OPTACK オプション を使用する必要があります。
3. コマンド /DIS、/RDIS、および /FOR で CD がある時のみ。
4. ETO ユーザーは常に OPTACK オプションを使用する必要があります。
5. 方向変換 (CD) がある時のみ。

## 出力応答要件

以下の表は、すべてのタイプの出力メッセージについて、IMS が要求する応答を要約したものです。

表 180. 出力応答、メッセージ・タイプ別に要求される

出力データ・タイプ	応答要求
更新、リカバリー	DR2
照会、リカバリー	DR2
照会、リカバリー不能	例外 DR1 および DR1 <sup>979</sup> ページの『3』
高速機能 (リカバリー)	例外 DR2 <sup>980</sup> ページの『1』 および DR2 <sup>980</sup> ページの『2』
最終 MFS ページ (MOD で PAGE=YES が指定されない時)	応答要求については上記の出力データ・タイプを参照
非最終 MFS ページ (MOD で PAGE=YES が指定された時はすべてのページ)	例外 DR2 および DR2 <sup>980</sup> ページの『2』

表 180. 出力応答、メッセージ・タイプ別に要求される (続き)

出力データ・タイプ	応答要求
IMS コマンド応答: /FORMAT、/DISPLAY、/RDISPLAY、その他のコマンド	DR2 例外 DR2 および DR2 <sup>980</sup> ページの『2』
テスト・モード出力	例外 DR1 および DR1 <sup>979</sup> ページの『3』
ブロードキャストまたはメッセージ通信出力	DR2

注:

- 例外応答 DR2 が要求された場合は、次の入力メッセージまたは RTR によって確認されます。
- 応答 DR2 は、以下の場合のいずれか 1 つが生じたときに要求されます。
  - BB を伴うメッセージが送信された (データを伴うビッド)。
  - IMS メッセージ・キュー上のメッセージがワークステーションへの送信待ち状態。
  - NFPACK オプションがシステム定義 TERMINAL マクロまたは ETO ログオン記述子で定義されている。
- DR1 は、BB を伴うメッセージが送信された (データを伴うビッドが出された) 場合に要求されます。

## IMS トランザクション・タイプ

トランザクションは、ワークステーションから IMS へ送る最も一般的なデータ・タイプです。IMS は、2 種類のトランザクション (更新と照会) をサポートします。

更新トランザクションでは、データベースを変更することができます。照会トランザクションでは、データベースのデータを見ることはできますが、変更または更新はできません。トランザクションは IMS システム定義で更新あるいは照会のいずれかに定義されます。

照会トランザクションについては、リカバリー可能かリカバリー不能かが追加属性として定義されます。リカバリー可能と定義した照会トランザクションはどのネットワークのエレメントで失敗してもいつでもリカバリー可能です。リカバリー不能照会トランザクションは、入出力エラー状態または IMS システム再始動の後では回復されません。

更新トランザクションは、すべてリカバリー可能です。

高速機能トランザクションは、すべてリカバリー可能と定義する必要がありますが、照会か更新かということについては、どちらでも構いません。

1 つのワークステーションに、1 つあるいは両方のタイプのトランザクションの取り扱いができるよう定義できます。リカバリー可能トランザクションとリカバリー不能トランザクションの両方を取り扱う場合は、コントローラーやコントローラー・アプリケーション・プログラムについて、他の決定や処理が必要になることがあります。トランザクションをリカバリー可能か、リカバリー不能かに定義するには、それぞれのトランザクション・タイプの操作環境に対する利点と欠点とをよく検討する必要があります。

以下のトピックでは、照会メッセージ・タイプとそれぞれのタイプが必要とする応答について説明します。

### リカバリー可能な照会トランザクション

リカバリー可能トランザクションが確実に回復できるように、ワークステーションは、特定の必要な機能を実行する必要があります。

リカバリー可能トランザクションが確実に回復できるよう、ワークステーションは、以下のことを行う必要があります。

- IMS への入力時に応答 DR1 または DR2 を要求する。OPTACK オプションが定義されている場合は、IMS への入力時にオプションで例外応答 DR1 または例外応答 DR2 を要求し、開始ブラケットと方向変換を指定してください。

- 応答 DR1 または DR2 が戻るまで、あるいは、OPTACK オプションが指定されている場合はブラケット終了を含む応答メッセージが戻るまで、入力メッセージのシーケンス番号と、入力メッセージのコピー (オプション) を保持しておく。
- IMS が要求する応答 DR2 を戻す。高速機能の場合は、別の入力メッセージまたは RTR コマンドの送信は、データのロギングまたは書き出しを行うことによって IMS 出力メッセージに対して責任をもてるようになってから行うようにしてください。

リカバリー可能トランザクションが確実に回復できるよう IMS は以下のことをします。

- 応答 DR2 を要求する。あるいは、送信待ちの高速機能出力はあるが、送信待ちの出力メッセージがない場合は、例外応答 DR2 を要求し、出力の肯定応答にもなる 後続のデータまたは RTR を待つ。
- 応答 DR2、入力データ、または RTR (高速機能の場合) が戻るまで、メッセージのシーケンス番号とそのメッセージのコピーを保持する。
- データのロギングまたは書き出しを行うことによって当該のメッセージに対して責任をもてるようになった後、要求された応答 DR1 または DR2 を戻す。

リカバリー可能メッセージの送信と応答メッセージの肯定応答の間で障害が起こった場合、送信側は、メッセージが宛先に届いたかどうかを判断することができません。再始動プロシージャの実行中に、IMS は VTAM コマンドを使用して IMS が受信した最終インバウンド同期点メッセージのシーケンス番号と IMS が送信した最終アウトバウンド同期点メッセージのシーケンス番号をコントローラーに知らせます。これにより、受信されなかったメッセージがあった場合は、それらをすべて再送することができます。

複数のリカバリー可能メッセージが送られた場合は、そのうちの 1 つだけが一時点での処理対象になります。これは、送信側がメッセージを送信した後、応答を受け取るまで、次のメッセージを送れないということです。IMS は 1 つのメッセージを読み取り、そのメッセージを入力キューに入れ、応答 DR1 または DR2 や応答メッセージを戻した後、次のメッセージを受け取ります。

## リカバリー不能照会トランザクション

IMS は、リカバリー可能性を実現するために必要な処理を行わないことを除いて、リカバリー不能トランザクションをリカバリー可能トランザクションと同じように取り扱います。この結果、リカバリー不能トランザクションの処理時間は少なく済みますが、ネットワークに障害 (例えば、回線障害、プロセッサ障害、キュー障害など) が起こった場合は、そのトランザクションがなくなる可能性があります。

リカバリー不能トランザクションの場合は、応答 DR1 と DR2 のどちらも要求する必要がありません。これは回復が保証されていないので、メッセージの受信側で肯定応答をする必要がないからです。ただし、リカバリー不能トランザクションでは、例外応答 DR1 または DR2 を要求する必要があります。例外応答 DR1 または DR2 の場合は、通常的环境下では応答を戻さないため、応答 DR1 または DR2 の場合より回線伝送が少なく済みます。

## リカバリー不能メッセージの IMS 受信の検証

SLU P システムは、IMS に対してリカバリー不能メッセージの確認応答を要求しません。しかし、SLU P システムが、開始ブラケットとブラケット終了の両方を指示するリカバリー不能メッセージを複数伝送しており、それらの受信の検証をしたい場合は、VTAM CHASE コマンドを送信することができます。

VTAM 標識は、応答 DR1 を要求する必要があります。SLU P システムが通常の確定応答 1 を受信した場合は、CHASE コマンドの前に送信したメッセージはすべて IMS によって受信済みです。

## IMS メッセージ通信

IMS は送信または受信したメッセージ通信をリカバリー可能トランザクションと同じように扱います。

このため、メッセージ通信には以下のことが当てはまります。

- メッセージ通信では、ワークステーションに ACK オプションが定義されている場合は、応答 DR1 または DR2 を要求する必要があります。OPTACK オプションが指定されている場合は、メッセージ通信では例外応答 DR1 または DR2 をオプションで要求し、さらに IMS への入力時に開始ブラケットと方向変換を指定する必要があります。
- メッセージ通信は IMS が回復します。

- メッセージ通信のシーケンス番号はメッセージ再同期で使用します。

## IMS コマンド

ネットワーク・システム・アナリストはワークステーションに IMS コマンド を入力できるかどうかを判別し、入力できる場合、どのコマンドが可能かを判別します。

IMS コマンドを送る場合、IMS はコントローラーに特定の応答要求をしません。応答 DR1 と DR2、および例外応答 DR1 と DR2 が使用可能ですが、例外応答 DR1 または DR2 を使用することをお勧めします。

IMS のコマンド処理では、IMS はコマンドを処理し終えたときにメッセージを戻します。したがって、応答 DR1 または DR2 が要求されている場合は、IMS はまず 応答 DR1 または DR2 を戻し、その直後にコマンドの完了を確認するメッセージを戻します。

コントローラーが例外応答 DR1 または DR2 を要求している場合は、応答 DR1 または DR2 の送信に必要な追加の回線伝送とコマンド完了メッセージを 除去することができます。

例外応答 DR1 または DR2 が要求されている場合は、コマンド完了メッセージが コマンドの肯定応答をし、コマンド処理が完了したことを示します。ACK オプションがワークステーションに定義されている場合は、コマンドの VTAM シーケンス番号はメッセージ再同期に関与しません。VTAM シーケンス番号が関与するのは、 応答 DR1 または DR2 だけが要求されている場合です。OPTACK オプションが定義されている場合は、すべての入力がメッセージ再同期に関与します。

## VTAM コマンドと標識

VTAM コマンドが IMS に送信される時には、DR1 を要求しなければなりません。

IMS が VTAM コマンドを送信する時には、DR1 を要求します。BIND または BID コマンドが送信される場合には、応答は DR1 または 例外 DR1 のいずれかでなければなりません。STSN コマンドが送信される場合には、DR1 が必要です。

コントローラーは、ワークステーションに対する他のすべての VTAM コマンドおよび標識に 応答します。

## MFS 制御要求

メッセージ形式サービス (MFS) を使用している時には、MFS 制御要求を用いて、ページ化メッセージを表示したり、表示コンポーネント画面を 制御したりすることができます。

制御要求は、入力機能管理ヘッダーで指定されます。

応答 DR1、DR2、または例外応答 DR1 および DR2 を要求することができます。例外応答 DR1 または DR2 の使用をお勧めします。なぜなら、コントローラーが例外応答 DR1 または DR2 を 要求している場合は、応答 DR1 または DR2 の送信に必要な追加の回線伝送と コマンド完了メッセージを除去することができるからです。

## エラー処理

以下のトピックでは、伝送またはプロトコル・エラーから生じる障害を処理するために IMS によって使用され、コントローラーまたはコントローラー・アプリケーション・プログラムに必要なプロシージャーについて説明します。

## IMS 検出エラー

IMS が VTAM、NCP、またはコントローラーから異常な送信または受信状態を検出すると、IMS は常にセッションを終了します。受信したメッセージからエラーが検出されると、IMS は表示コンポーネントを保護し、次に示す 4 つのバイトのセンス情報を含む例外応答 DR1 または DR2 を戻します。

### バイト 0、1

システム・センス・フィールド

### バイト 2、3

ユーザー・センス・フィールド

## システム・センス・フィールド (SSENSE)

バイト 0 および 1 に、次の値の 1 つが入ります。

### X'0800'

ユーザー・センス・フィールドに、ユーザー・メッセージ・テーブルからのユーザー・メッセージが入ります。

### X'0819'

ユーザー・センス・フィールドに、IMS メッセージ番号 290 (2 進) が入ります。使用可能な出力はありません。この値は、前に受信した RTR コマンドに対する非 MFS 応答の場合にだけ設定されます。

### X'0826'

ユーザー・センス・フィールドに、IMS メッセージ番号が入ります。

## ユーザー・センス・フィールド (USENSE)

IMS はユーザー・センス・フィールドを使用して、該当するエラー・メッセージの番号 (2 進に変換される) を渡します。

例えば、無効なトランザクション・コードの場合は、次のメッセージが生成されます。

```
DFS064 DESTINATION CANNOT BE FOUND OR CREATED
```

IMS は、無効なトランザクション・コードを受信すると、ユーザー・センス・フィールドに X'0040' を含む例外応答 DR1 または DR2 を戻します。

XRF 複合システムでは、XRF セッション・テークオーバーの前後で入力トランザクションが失われると、代替システムが SSENSE=X'0826'、USENSE X'0F15'、および「DFS3861I SYSTEM TAKEOVER OCCURRED (DFS3861I システム・テークオーバー発生)」というメッセージを送信します。そうすると、最後の入力レコードを再送信しなければなりません。

IMS は、各セッションを無条件ブラケット終了とバインドします。IMS が入力に対する例外応答 DR1 または DR2 を送信する場合には、IMS は (現行ブラケット状態にかかわらず) 送信状態のままであればならず、ワークステーションは受信状態にならなければなりません。エラーのある入力メッセージがブラケット終了を指定している場合には、IMS とワークステーションの両方がブラケット間状態にならなければなりません。IMS は、IMS エラー・メッセージ全体を当該ブラケット標識を含む単一チェーンとして送信することによって、ワークステーションをブラケット間およびコンテンション状態にします。

IMS エラー・メッセージは複数のセグメントをもつことができ、最大メッセージ長は 32,000 バイトです。ただし、ワークステーションに送信される現行 IMS エラー・メッセージは、すべて単一セグメントであり、その長さは 132 バイト未満です。これらのメッセージは、FM ヘッダーと、すべての出力メッセージに適用される 伝送規則に従います。

ユーザー・センス・データの生成は、標準 IMS エラー・メッセージ形式に依存します。ユーザー出力編集ルーチンがこの形式を変更すると、セッションが終了する可能性があります。

### 関連概念

[933 ページの『XRF システム用のコントローラー・アプリケーション・プログラムに関する考慮事項』](#)

SLUP システムが XRF 複合システムで稼働する場合には、コントローラー・アプリケーション・プログラムは、XRF のテークオーバー時に失われたメッセージを処理できるようにしなければなりません。

## コントローラーまたはステーション検出エラー

IMS からのメッセージでエラーを検出したとき、あるいは単にその時点でメッセージを受け入れることができない場合に、コントローラーは以下の 4 つのバイトのセンス・データを含む例外応答 DR2 を必ず戻します。

### バイト 0、1

システム・センス・フィールド

### バイト 2、3

ユーザー・センス・フィールド



## システム・センス・フィールド

バイト 0 および 1 には、X'0802' (RECOVERABLE ERROR) または X'0811' (BREAK)が入っている必要があります。

### X'0802'

リカバリー可能エラーメッセージがリカバリー可能であれば、IMS はそのメッセージを出力キューに戻し、当該ワークステーションからの次の入力の後か、または当該宛先への追加出力がキュー状態になったときに、そのメッセージを再送します。チェーン・メッセージの再伝送は、先頭チェーン伝送から始まります。例外 DR2 を受信するとオープン・ブラケットがクローズされるので、再送信されたメッセージには、ブラケット終了のみまたは開始ブラケット標識とブラケット終了標識の両方が入ります(どちらが入るかは、入力が起こったかどうか、またはブラケット標識が指定されたかどうかで決まります)。リカバリー不能メッセージはすでにデキューされているため、それを回復して再送することはできません。

### X'0811'

中断: 出力を取り消します。出力メッセージは終了します。このメッセージは IMS からデキューされ、再送されません。リカバリーは無理です。キューに入っている、当該ワークステーション宛ての残りの出力メッセージは、すべて送信されます。このセンス・コードを受信した後で使用可能な追加出力がなければ、IMS はヌルの出力メッセージを送信します。

### X'0812'

一時エラー: リソースが使用不能です。IMS のアクションは X'0802' の場合と同じです。

### X'0813'

「TEMPORARY ERROR, BID OR BRACKET REJECTED—NOT READY TO RECEIVE (一時エラー、BID またはブラケットが拒否された: 受信は不可)」が後に続きます。IMS のアクションは X'0802' の場合と同じです。

### X'0814'

「TEMPORARY ERROR, BID OR BRACKET REJECTED—READY TO RECEIVE (一時エラー、BID またはブラケットが拒否された: 受信は可)」が後に続きます。IMS のアクションは X'0802' の場合と同じです。

### X'081C'

ステーションコンポーネント・ダウン-バイト 3 が有効なコンポーネント (X'01' から X'04') を示している場合には、IMS は指示されたコンポーネントに動作不能のマークを付けます。バイト 3 が X'00' または無効なコンポーネント (X'01' - X'04' 以外) を示している場合には、IMS はこれをセッション終了要求と解釈して、セッションを終了します。進行中のメッセージがリカバリー可能であれば、IMS はそれを出力キューに戻し、ワークステーションまたはコンポーネントが再び動作可能になった時点でそのメッセージを再送します。リカバリー不能メッセージを回復して再送することはできません。

バイト 0 および 1 に上記の値以外の値が入っていると、それらはすべてセッション終了要求として処理されます。

BID オプションが定義されている場合には、コントローラー・アプリケーション・プログラムは、BID コマンドに対する例外応答 DR1 で X'0813' または X'0814' を使用しなければなりません。そうしないと、無効応答のためにセッションが終了します。

X'0813' センス・コードでは、メッセージは、リカバリー可能な場合には IMS 出力キューに戻され、リカバリー不能な場合にはデキューされます。リカバリー可能データがキューに戻されると、ワークステーションからの次の入力の後か、または当該宛先に対する追加出力がキューに入っている状態になった時に、BID が再び送信されます。

X'0814' センス・コードでは、リカバリー可能メッセージは IMS 出力キューに戻され、リカバリー不能メッセージはデキューされます。IMS はその後、RTR コマンドを待ちます。この時点で、ワークステーションは入力を IMS に送信することができます。IMS がブラケット間状態にあるときは、RTR が受信されるまで、IMS が出力を開始することはありません。IMS がイン・ブラケット送信状態のまましていると(最後の入力が BB/CD を運んだ)、IMS は EB を含む出力を送信して、現行ブラケットを終了します。

### 関連概念

971 ページの『出力メッセージ』

IMS からの出力メッセージは、いくつかの異なるタイプのいずれかとなります。

## ユーザー・センス・フィールド

このフィールドは、システム・センス・フィールドが X'081C' に設定された時に使用されます。バイト 2 は IMS によって検査されないため、どのような値でも含むことができます。

バイト 3 には、出力メッセージの送信先となっているコンポーネントの 16 進識別 (X'nn') が入ります。IMS はそのコンポーネントに動作不能のマークを付けて、ワークステーションの他の動作可能コンポーネントに出力を送信し続けます。X'00' または無効なコンポーネントが指定されると、セッションは終了します。

IMS は、各セッションを無条件ブラケット終了とバインドします。これは、X'0813' および X'0814' を除くすべてのサポートされているセンス・コードについて IMS とワークステーションの両方がブラケット間およびコンテンション状態になることを意味します。この 2 つのセンス・コードの場合、IMS はブラケット間および受信状態になり、ワークステーションはブラケット間および送信状態になります。

## VTAM 論理装置状況 (LUSTATUS) コマンド

SLU P システムがコンポーネント・ダウン条件またはワークステーション・ダウン条件を検出した場合、そのとき IMS からのメッセージを受信中でなければ、SLU P システム は VTAM **LUSTATUS** コマンドを送信して IMS にその条件を通知することができます。

この LUSTATUS には、**ステーション / コンポーネント・ダウン** 用の例外応答 DR1 または DR2 と形式が等しい 4 バイトのデータを添付する必要があります。IMS は、**LUSTATUS** を受信すると、指示されたコンポーネントに動作不能のマークを付けて、セッションを終了します。

ワークステーションはまた、VTAM **LUSTATUS** を送信することによって、コンポーネントが動作可能になった時に IMS に通知することができます。システム・センス・フィールドは X'0001' に等しくなければならず、ユーザー・センス・フィールドのバイト 3 は X'01' から X'04' の値に等しくなければなりません。IMS は、指示されたコンポーネントの動作不能状態をリセットし、そのコンポーネントに対してキューに入っている出力を送信します。

**LUSTATUS** に対するその他のすべての値は、セッション終了要求と見なされます。

## VTAM 受信可 (RTR) コマンド

以下のリストは、RTR コマンドの受信時に実行される、定義済みの IMS 機能を要約したものです。

- 未処理の高速機能出力メッセージはデキューされ、IMS はワークステーションを端末応答モードから除外します。
- 出力保護コンポーネントは、無保護のマークが付けられ、出力に使用できるようになります。
- 直前の例外応答で X'0814' センス・コードのために延期されていた出力が許可されます。
- IMS 出力が送信可能になると、IMS は DR1 応答を戻し、その直後に出力メッセージが続きます。
- 使用可能な IMS 出力がない場合、またはワークステーションが出力を送信できない状態 (出力が停止または静止された) にある場合には、IMS は例外 DR1 を戻し、その後に、使用可能な出力がないことを示すエラー・メッセージが続きます。
- MFS ページング出力の伝送時に RTR が受信されると、IMS は例外 DR1 を戻し、その後に無効なページング要求を示すエラー・メッセージが続きます。
- IMS が例外応答を戻すと、論理装置に定義された表示コンポーネントは いずれも保護状態になり、出力に使用できなくなります。

## VTAM CANCEL コマンド

完了していない進行中のチェーン・メッセージを終了させるために、IMS は VTAM **CANCEL** コマンドを使用しますが、SLU P システムも このコマンドを使用する必要があります。

IMS は、次の場合に **CANCEL** コマンドをワークステーションに送信します。

- IMS マスター端末オペレーターが IMS /**DEQUEUE** コマンドを出し、IMS がメッセージのチェーンの最後の伝送をまだ送信していない。



- IMS は、メッセージの伝送に対する例外応答 DR1 または DR2 を受信したが、まだチェーンの最後の伝送を送信していない。

コントローラーまたはコントローラー・アプリケーション・プログラムは、次の場合に **CANCEL** コマンドを使用する必要があります。

- メッセージのチェーンの最後の伝送を送信する前に、IMS から例外応答 DR1 または DR2 を受信した。
- メッセージのチェーンの最後の伝送を送信する前に IMS への複数 RU チェーン入力を取り消す。この場合、取り消すメッセージに開始ブラケット標識しか入っていないければ、VTAM **CANCEL** コマンドに、ブラケット終了標識、方向変換標識、または追加標識を組み込むことができます。

## VTAM リカバリー要求コマンド

ワークステーションが、通常操作時に、IMS とのセッションを終了しないが、IMS と再同期するようなエラー状態(例えば、コントローラーでのディスク障害やプログラム・チェックなど)を検出した場合には、アプリケーション・プログラムはメッセージ再同期を開始することができます。

メッセージ再同期の開始は、VTAM リカバリー要求 (RQR) コマンドを送信することによって行われます。IMS は、**CLEAR** コマンドとその後続くシーケンス番号の設定およびテスト (STSN) コマンドで応答します。その後メッセージ再同期が続きますが、これは [952 ページの『メッセージ・リカバリー』](#) で説明したのと同じです。

**制約事項:** ノードが応答モードになっており、応答メッセージがまだ出力可能でないとき、すなわち、入力応答モード・トランザクションがまだキューに入っているか実行中であるときには、SNA リカバリー要求 (RQR) コマンドによるセッション開始および再同期は許可されません。アプリケーション同期点より前では、応答モード・トランザクションはリカバリー可能でも再始動可能でもないので、入力処理が完了するまでセッション入力確認は行われません。

IMS に送信される SIGNAL コマンドに 4 バイトのデータを付けることができます。最初の 2 バイトはシステム・シグナル・データです。最後の 2 バイトは、IMS が無視するユーザー・データです。最初の 2 バイトは次のように処理されます。これ以外の値はすべて無視されます。

### X'0000'

特定の処置は取られません。しかし、IMS に対して送信された直前の例外応答 DR1 または DR2 のためにワークステーションがアイドル状態になっている場合には、IMS が、このデータを受信したことによって、動作中の無保護コンポーネントに使用可能な出力をどうしても送信してしまう可能性があります。この入力、出力コンポーネント保護または表示保護をリセットしません。

### X'0001'

このデータは、IMS に送信を停止させ、現行出力メッセージの終了時点で入力を待機させるアテンション・シグナルとして扱われます。このタイプのシグナルは、入力、必要な応答 DR1 または DR2、あるいは例外応答 DR1 または DR2 のいずれかを IMS へ送信する前に使用することができます。

## 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。本書の他言語版を IBM から入手できる場合があります。ただし、ご利用にはその言語版の製品もしくは製品のコピーを所有していることが必要な場合があります。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒 103-8510

東京都中央区日本橋箱崎町 19 番 21 号

日本アイ・ビー・エム株式会社

法務・知的財産

知的財産権ライセンス 渉外

IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

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

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

*IBM Director of Licensing*

*IBM Corporation*

*North Castle Drive, MD-NC119*

*Armonk, NY 10504-1785*

*US*

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

記載されている性能データとお客様事例は、例として示す目的でのみ提供されています。実際の結果は特定の構成や稼働条件によって異なります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名前はすべて架空のものであり、類似する個人や企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、さまざまなオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。これらのサンプル・プログラムは特定物として現存するままの状態を提供されるものであり、いかなる保証も提供されません。IBM は、お客様の当該サンプル・プログラムの使用から生ずるいかなる損害に対しても一切の責任を負いません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、

次のように、著作権表示を入れていただく必要があります。

©(お客様の会社名)(年).

このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。

© Copyright IBM Corp. \_年を入れる\_.

## プログラミング・インターフェース情報

この情報では、IMS が提供するプロダクト・センシティブ・プログラミング・インターフェースとそれに関連する情報と同時に、IMS が提供する診断、修正、またはチューニング情報についても記述しています。

プロダクト・センシティブ・プログラミング・インターフェースにより、お客様のインストール済み環境で、このソフトウェア製品の診断、修正、モニター、修復、調整、またはチューニングなどの作業を実行することができます。これらのインターフェースを使用すると、IBM のソフトウェア製品の詳細設計や実装に対する依存関係が生じます。このためプロダクト・センシティブ・プログラミング・インターフェースは上記の特別な目的にだけ使用してください。詳細設計やその実現方法に依存しているので、このようなインターフェースに合わせて作成したプログラムは、新しい製品のリリース、バージョンで実行するとき、または保守サービスの結果として、変更が必要になることがあります。プロダクト・センシティブ・プログラミング・インターフェースとそれに関連する情報は、セクションやトピックの単位の場合はその冒頭で識別され、それ以外の場合は「プロダクト・センシティブ・プログラミング・インターフェース」というマーキングで識別されます。IBM では、上記の冒頭部での識別の記述、およびその記述を参照する本書内のすべての記述を、そのような記述によって示される全体コピーまたは部分コピーに含めるよう求めています。

診断、修正、チューニングの情報は、IMS の診断、変更、またはチューニングをお客さまが行う手助けをするために提供されます。診断、修正、またはチューニング情報は、プログラミング・インターフェースとしては使用しないでください。

診断、修正、またはチューニング情報は、節またはトピックの場合はその冒頭で識別され、それ以外の場合は次のようにマーク付けされています。診断、変更、またはチューニング情報。

## 商標

---

IBM、IBM ロゴおよび [ibm.com](http://www.ibm.com)<sup>®</sup> は、世界の多くの国で登録された International Business Machines Corporation の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。

Adobe、Adobe ロゴ、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは Oracle やその関連会社の米国およびその他の国における商標または登録商標です。

UNIX は The Open Group の米国およびその他の国における登録商標です。

## 製品資料に関するご使用条件

---

これらの資料は、以下のご使用条件に同意していただける場合に限りご使用いただけます。

### 適用される条件

このご使用条件は、IBM Web サイトのすべてのご利用条件に追加して適用されます。

### 個人使用

これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布 (頒布、送信を含む) または表示 (上映を含む) することはできません。

### 商業的使用

これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

### 権利

ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。

お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入 関連法規を含む、すべての関連法規を遵守するものとします。

IBM は、これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態を提供され、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任なしで提供されます。

## IBM オンライン・プライバシー・ステートメント

---

サービス・ソリューションとしてのソフトウェアも含めた IBM ソフトウェア製品（「ソフトウェア・オファリング」）では、製品の使用に関する情報の収集、エンド・ユーザーの使用感の向上、エンド・ユーザーとの対話またはその他の目的のために、Cookie はじめさまざまなテクノロジーを使用することがあります。多くの場合、ソフトウェア・オファリングにより個人情報が収集されることはありません。IBM の「ソフトウェア・オファリング」の一部には、個人情報を収集できる機能を持つものがあります。ご使用の「ソフトウェア・オファリング」が、これらの Cookie およびそれに類するテクノロジーを通じてお客様による個人情報の収集を可能にする場合、以下の具体的事項をご確認ください。

この「ソフトウェア・オファリング」は、Cookie もしくはその他のテクノロジーを使用して個人情報を収集することはありません。

この「ソフトウェア・オファリング」が Cookie およびさまざまなテクノロジーを使用してエンド・ユーザーから個人を特定できる情報を収集する機能を提供する場合、お客様は、このような情報を収集するにあたって適用される法律、ガイドライン等を遵守する必要があります。これには、エンドユーザーへの通知や同意の要求も含まれますがそれらには限られません。

このような目的での Cookie を含むさまざまなテクノロジーの使用の詳細については、『IBM プライバシー・ステートメント』（<https://www.ibm.com/jp-ja/privacy>）および『IBM オンライン・プライバシー・ステートメント』（<https://www.ibm.com/jp-ja/privacy/details>）の『クッキー、ウェブ・ビーコン、その他のテクノロジー』および『IBM Software Products and Software-as-a-Service Privacy Statement』（<http://www.ibm.com/software/info/product-privacy>）というタイトルのセクションを参照してください。



## 参考文献

この参考文献のリストには、IMS 15 ライブラリーのすべての資料が記載されています。

表題	頭字語	資料番号
IMS V15 アプリケーション・プログラミング	APG	SC43-4281
IMS V15 アプリケーション・プログラミング API	APR	SC43-4279
IMS V15 コマンド 第1巻: IMS コマンド A-M	CR1	SC43-4284
IMS V15 コマンド 第2巻: IMS コマンド N-V	CR2	SC43-4285
IMS V15 コマンド 第3巻: IMS コンポーネントおよび z/OS コマンド	CR3	SC43-4286
IMS V15 コミュニケーションおよびコネクション	CCG	SC43-4277
IMS V15 データベース管理	DAG	SC43-4276
IMS V15 データベース・ユーティリティー	DUR	SC43-4280
IMS Version 15 Diagnosis	DGR	GC27-6786
IMS V15 出口ルーチン	ERR	SC43-4279 SA88-7180
IMS V15 インストール	INS	SC27-6788
IMS Version 15 Licensed Program Specifications	LPS	GC27-6799
IMS V15 メッセージおよびコード 第1巻: DFS メッセージ	MC1	GC43-4282
IMS V15 メッセージおよびコード 第2巻: DFS 以外メッセージ	MC2	GC43-4283
IMS V15 メッセージおよびコード 第3巻: IMS 異常終了コード	MC3	GC27-6791
IMS V15 メッセージおよびコード 第4巻: IMS コンポーネント・コード	MC4	GC27-6792
IMS V15 オペレーションおよびオートメーション	OAG	SC43-4275
IMS V15 リリース計画	RPG	GC43-4272
IMS V15 システム管理	SAG	SC43-4271
IMS V15 システム定義	SDG	GC43-4272
IMS V15 システム・プログラミング API	SPR	SC43-4269
IMS V15 システム・ユーティリティー	SUR	SC43-4270





# 索引

日本語, 数字, 英字, 特殊文字の順に配列されています。  
なお, 濁音と半濁音は清音と同等に扱われています。

## [ア行]

- アクセシビリティ
  - キーボード・ショートカット [xxv](#)
  - 機能 [xxv](#)
- アクセス制御環境エレメント
  - キャッシュ・ユーザー ID のエージング値 [891](#)
- アクセス不可能な ETO ユーザー制御ブロック [103](#)
- 宛先 (destination)
  - 判別メッセージ [805](#)
- 宛先解決出口ルーチン (DFSYPRX0)
  - 基本メッセージ・フロー [778](#)
  - OTMA の使用制限 [814](#)
  - OTMA 用に IMS をカスタマイズ [804](#)
- 宛先記述子, OTMA
  - 限度 [800](#)
- 宛先キュー名, ATTACH FM ヘッダー [578](#)
- 宛先コードの定義 [391](#)
- 宛先システム (MSC) [692](#)
- 宛先プロセス名 (DPN)
  - メッセージ経路指定, ISC [479](#)
  - ATTACH FM ヘッダー [575](#)
  - SCHEDULER FM ヘッダー [575](#)
- アプリケーション・スレッド [123](#)
- アプリケーション・プログラミング
  - OTMA
    - DL/I 呼び出し [841](#)
- アプリケーション・プログラム
  - 既存のプログラム, ISC を伴う [478](#)
  - 金融機関から SLU P への変換 [934](#)
  - コールアウト機能の概要 [884](#)
  - 定義 [376](#)
  - 非同期コールアウト要求への応答の相関 [886](#)
  - リモート [377](#)
  - CICS, IMS で [638](#)
- IMS
  - OTMA [774](#)
  - IMS が提供する機能 [378](#)
  - ISC サンプル [673](#)
  - SLU P コントローラー [934](#)
  - XRF 考慮事項, SLU P [933](#)
- アプリケーション・プログラム, z/OS の
  - ODBA (Open Database Access) インターフェースの使用 [767](#)
  - Open Database Access (ODBA) インターフェースの使用 [767](#)
- アプリケーション・プログラム・インターフェース [467](#)
- アプリケーション呼び出し処理 [124](#)
- 暗号化
  - OTMA の制約事項 [814](#)
- 異常終了 (abend)
  - CICS [657](#)
  - MSC 会話 [705](#)
- 印刷出力
  - 共用プリンター [451](#)
  - 印刷出力 (続き)
    - 候補プリンター [451](#)
    - スプール出力制御 [450](#)
    - 制御 [449](#), [451](#)
    - 操作上の考慮事項 [451](#)
    - プリンター・コンポーネント [450](#)
    - 3270 プリンター・コンポーネント [450](#)
    - 3270R と ETO [451](#)
    - ASSIGN (/ASSIGN) コマンド [449](#)
    - VTAMLIST 定義 [450](#)
    - /ASSIGN コマンド [449](#)
  - エージング値 [891](#)
  - エラー
    - ISC セッション終了, 原因となる [549](#)
  - エラー処理
    - エラー・メッセージ [563](#)
    - コントローラーが検出したエラー
      - システム・センス・フィールド [983](#)
      - 使用 [983](#)
      - ユーザー・センス・フィールド [985](#)
    - 中断コード X'0811'
      - 出力メッセージ [984](#)
    - メッセージのキューイング [557](#)
    - メッセージの長さ [557](#)
  - CANCEL コマンド
    - いつ IMS によって送信されるか [985](#)
    - 使用 [985](#)
  - FM ヘッダー [557](#)
  - IMS 検出エラー [983](#)
  - ISC
    - 選択受信側 ERP [545](#)
    - 送信側 ERP [550](#)
    - 送信側 ERP センス・コード [550](#)
    - ページング・エラー, データ・フロー・リセット状態時の [549](#)
    - ページング・エラー, 非ページ化メッセージの [552](#)
    - ページング・エラー, ページ化メッセージでの [544](#)
    - CICS-IMS セッション [652](#)
  - LUSTATUS コマンド [542](#), [985](#)
  - MFS 検出エラー [563](#)
  - RQR コマンド [986](#)
  - SIGNAL コマンド [986](#)
  - SLU P
    - IMS 検出エラー [982](#)
  - VTAM 論理装置状況コマンド [642](#)
  - XRF 複合システムにおいて IMS が出すエラー・メッセージ [983](#)
- エラー抽出サービス
  - APPC/IMS [45](#)
- エラー・メッセージ
  - 原因となる状態
    - 出力で BB のみを指定 [977](#)
    - 入力で BB のみを指定 [970](#)
  - 長さ [557](#), [983](#)
  - MFS 検出エラー [563](#)
- エラー・リカバリー・プロシージャ (ERP)
  - 拡張 [652](#)

エラー・リカバリー・プロシージャ (ERP) (続き)

機能管理ヘッダー [652](#)

選択受信側

センス・コード [545](#)

CICS-IMS セッション [652](#)

FM ヘッダー

形式 [580](#)

IMS が実行する [934](#)

応答時間 [442](#), [731](#)

応答モード

端末

終了方法 [944](#)

図 [943](#)

ステーション別 [943](#)

メッセージ・キュー・データ・セットのサイズ [971](#)

ワークステーション出力バッファのサイズ [971](#)

動的確立 [473](#)

応答モード (response mode)

エラー

ISC [518](#)

端末

概要 [405](#)

活動時 [942](#)

ステーション別 [943](#)

制約事項 [943](#)

設計の考慮事項 [603](#), [604](#), [943](#)

定義 [942](#)

トランザクション依存を指定した影響 [942](#)

ISC

入力 [644](#)

応答要求、出力メッセージ [979](#)

応答要件

出力 ISC メッセージ [521](#)

照会トランザクション

回復可能 [520](#), [980](#)

リカバリー不能 [520](#), [981](#)

リカバリー不能照会トランザクション [432](#)

IMS コマンドと標識 [963](#), [978](#)

IMS 受信検証 [521](#), [981](#)

ISC メッセージ [522](#)

LU 6.2 アプリケーション・プログラム [432](#)

MFS 制御要求 [982](#)

VTAM 標識とコマンド [466](#), [963](#)

オープン・システム間相互接続 [773](#)

オープン・データベース

IMS Connect

タイムアウト指定 [319](#)

オープン・データベース・アクセス (ODBA)

インターフェース [767](#)

概要 [767](#)

セキュリティ

APSB [770](#)

RAS [769](#)

セキュリティの定義 [768](#)

接続

構成 [761](#)

セットアップ [767](#)

モジュール

JOBLIB にどちらを置くか [768](#)

APSB セキュリティ [770](#)

IMS データベースへのアクセス [767](#)

RAS セキュリティ [769](#)

オトマ

メッセージあふれ検出 [818](#)

オトマ (続き)

リソース管理

トランザクション・インスタンス・ブロック (TIB) [816](#)

オプション、CICS

システム定義 (system definition) [629](#)

テーブルの準備 [629](#)

オペレーター論理ページング

同期点 [522](#)

ページ化メッセージ [567](#)

有効な時に [527](#)

オペレーター論理ページング (operator logical paging)

ページング・エラー [563](#)

QXFR FM ヘッダー [585](#)

SLU P、MFS オプション [947](#)

VTAM 標識 [532](#)

オンライン・パフォーマンス [441](#)

オンライン変更 (online change)

オンライン変更ユーティリティ [442](#)

ライブラリー [443](#)

DISPLAY MODIFY コマンド [433](#)

## [カ行]

開始

セッション [431](#)

外部エントリー・ベクトル・テーブル (EEVT) [136](#)

回復可能

トランザクション [864](#)

リソース [862](#)

外部サブシステム

コールアウト要求 [881](#)

接続機能 [115](#)

接続先 [115](#)

外部サブシステム・コマンド・サポート [125](#)

外部サブシステム 接続機能

アプリケーション・スレッド [123](#)

アプリケーション呼び出し処理 [124](#)

外部サブシステムが用意する機能 [120](#)

外部サブシステム・コマンド・サポート [125](#)

外部サブシステム・モジュールのロード [136](#)

概要 [119](#)

言語インターフェース・モジュール [140](#)

コマンド

/START SUBSYS [142](#)

/STOP SUBSYS [142](#)

固有の言語インターフェース・エントリー・ポイント [140](#)

サインオン処理 [122](#)

作成、ESAP の作業域の [137](#)

作成、ESMT の [127](#)

サブシステム終了 [142](#)

サブシステム接続 [122](#)

識別プロセス [122](#)

従属領域

接続の確立 [138](#)

従属領域接続 [143](#)

終了 ECB [123](#), [142](#)

終了要求、外部サブシステムからの [142](#)

処理 [135](#)

スレッド [122](#)

制御領域識別の延期 [138](#)

接続

従属領域 [138](#)

外部サブシステム 接続機能 (続き)

接続処理  
概要 [122](#)  
接続の開始 [137](#)  
接続の確立 [122](#)  
接続の終了 [123](#)  
通知メッセージ [122](#), [139](#)  
停止状態とは [143](#)  
出口ルーチン・インターフェース [120](#)  
トークン [119](#)  
複数の外部サブシステムへのアクセス [141](#)  
ユーザー許可処理 [122](#)  
リカバリー・コーディネーター [119](#)  
リカバリー・トークン [124](#), [141](#)  
リソースの調整 [124](#)  
ロード、ESAP の [135](#)  
2 フェーズ・コミット処理 [124](#)  
CHANGE コマンド [141](#)  
CRC [125](#)  
DFSEMODL マクロ [127](#)  
DFSEWAL マクロ [130](#)  
DISPLAY SUBSYS コマンド [141](#)  
EEVPEEA [135](#)  
EEVPEWA [137](#)  
EEVT [136](#)  
EEVT のマッピング [136](#)  
ESMT  
作成 [127](#)  
説明 [120](#)  
EWAL [130](#)  
IMS サービス、ESAP で使用できる [126](#)  
IMS への外部サブシステムの指定方法 [120](#)  
INQ パラメーター [124](#)  
OASN [141](#)  
PDSE [127](#)  
RTT [120](#)  
SSR コマンド [142](#)  
外部サブシステム 接続パッケージ (ESAP) [120](#)  
外部サブシステムの定義 [380](#)  
外部サブシステム・モジュール・テーブル  
PDSE [127](#)  
外部サブシステム・モジュールのロード [136](#)  
外部実行モード、ISC [471](#)  
会話  
保護 [842](#)  
OTMA および IMS [807](#)  
会話型異常終了出口ルーチン [700](#)  
会話型処理 (conversational processing) [399](#)  
会話型トランザクション  
OTMA での終了 [822](#)  
OTMA、終了 [822](#)  
会話型プログラム  
IMS Connect サポート [295](#)  
会話の終了、MSC 会話 [705](#)  
会話モード  
エラー  
ISC [518](#)  
正常終了、ISC 拡張 [520](#)  
制約事項 [490](#)  
説明 [405](#)  
IMS-CICS [642](#)  
返されるクライアント ID  
形式 [253](#)  
拡張回復機能 [808](#)

拡張回復機能 (XRF)

クラス 1 端末 [420](#)  
クラス 2 端末 [419](#)  
クラス 3 端末 [419](#)  
サービス・クラス [420](#)  
端末 [418](#)  
端末サポート [395](#), [420](#)  
端末のクラス [418](#)  
通信の確立  
金融機関通信システム [938](#)  
システムのテークオーバーに関する考慮事項 [938](#)  
ISC [501](#)  
SLU P [938](#)  
定義 [416](#)  
テークオーバーに関する考慮事項  
金融機関通信システム [938](#)  
SLU P [938](#)  
マスター端末 [404](#)  
リカバリー・モード [410](#)  
MNPS [417](#)  
SLU P アプリケーション・プログラム [933](#)  
拡張出力コンポーネントの保護 [949](#)  
拡張端末オプション (ETO)  
アルゴリズム  
ログオン記述子 (logon descriptor) [86](#)  
LTERM の割り振り [98](#)  
印刷関連付けの技法 [100](#)  
会話 [400](#)  
画面サイズ制御バイト [78](#)  
画面定義の例 (非 SNA 3270) [80](#)  
記述子  
ログオン (logon) [85](#)  
ログオン、定義 [65](#)  
MFS [90](#)  
VTAM TERMINAL マクロ [85](#)  
記述子作成 [84](#)  
記述子の格納 [85](#)  
記述子のコーディング  
概要 [84](#)  
キュー (動的ユーザー・メッセージ)、定義 [63](#)  
計画  
ユーザー ID [82](#)  
ユーザー・キュー名 [83](#)  
構造、作成と削除 [67](#)  
構造  
端末、定義 [64](#)  
ユーザー、定義 [64](#)  
サインオフの定義 [109](#)  
サインオン  
サインオン・データの提供 [96](#)  
サインオン出口ルーチン (DFSSGNX0) [101](#)  
サインオンの定義 [95](#)  
システム定義 (system definition) [84](#)  
システム定義時間の削減 [380](#)  
出力  
不注意による出力データ・ストリーム [109](#)  
状況を保存するコマンド [99](#)  
静的端末、定義 [63](#)  
選択の指針  
ログオン記述子 [86](#)  
LOGOND パラメーター [86](#)  
装置タイプ、定義 [76](#)  
装置特性テーブル [90](#)  
定義

拡張端末オプション (ETO) (続き)  
定義 (続き)  
自動サインオフおよび自動ログオフ・タイマー [106](#)  
出口ルーチン  
サインオン出口ルーチン (DFSSGNX0) [101](#)  
出口ルーチン、コーディング [92](#)  
出口ルーチンのリスト [92](#)  
デフォルト CINIT/BIND ユーザー・データ形式 [95](#)  
動的端末 [379](#)  
動的ユーザー、定義 [64](#)  
特殊処理モード [99](#)  
ノード・ユーザー記述子 [89](#)  
非 SNA 3270 装置  
プリンターおよびディスプレイ [76](#)  
非同期出力 [107](#)  
非発信元端末への出力メッセージ送達 [108](#)  
複数のサインオン [96](#)  
プリンター  
直接印刷 [100](#)  
プリンター・ノード名 [100](#)  
ユーザー記述子  
インストール時作成 [88](#)  
ノード・ユーザー記述子 [88](#)  
3275 装置 [77](#)  
DFSUSER 記述子 [90](#)  
ETO STSN 装置の /SIGN コマンド [112](#)  
ETO ログオン記述子での MODETBL [428](#)  
LU 2  
装置 [77](#)  
LU 6.1 (ISC) 端末 [110](#)  
MFS 装置特性テーブル [90](#)  
NTO 装置 [78](#)  
RACF [379](#)  
STSN 端末  
/SIGN コマンドのサポート [112](#)  
VTAM 考慮事項  
CINIT セッション制御ブロックのログオン [76](#)  
VTAM PSERVIC パラメーター [76](#)  
拡張端末オプション (ETO) (Extended Terminal Option (ETO))  
印刷の共用 [102](#)  
応答モード (response mode) [112](#)  
概念  
要約 [69](#)  
概要 [63](#)  
会話 [112](#)  
カスタマイズ [68, 72](#)  
記述子  
更新 [93](#)  
削除 [93](#)  
使用 [68](#)  
追加 [93](#)  
定義 [65](#)  
MSC (複数システム 結合機能) [92](#)  
user [88](#)  
共通ログオン記述子 [85](#)  
計画  
運用 [83](#)  
LU2 [78](#)  
自動サインオフ [104](#)  
自動ログオフ [105](#)  
自動ログオン (autologon) [106](#)  
出力  
割り当て [107](#)

拡張端末オプション (ETO) (Extended Terminal Option (ETO)) (続き)  
状況をリセットして制御ブロックを解放するコマンド  
[99](#)  
使用の利点 [63](#)  
初期設定  
記述子妥当性検査 [93](#)  
DFSINTX0 [93](#)  
推奨事項 [90](#)  
制御ブロックの削除  
サインオフ後 [109](#)  
ログオフ後 [109](#)  
静的端末  
動的端末、併用 [75](#)  
制約事項 [72](#)  
セキュリティー  
サインオン [81](#)  
送達不能キュー [107, 108](#)  
装置特性テーブル [78](#)  
端末  
静的端末と動的端末の併用 [75](#)  
定義 [63](#)  
端末 - LTERM 関連 [98](#)  
定義  
パラメーター [103](#)  
出口ルーチン  
使用 [68](#)  
動的端末  
静的端末、併用 [75](#)  
動的端末 (dynamic terminal)  
定義 [63](#)  
特定の宛先を持つ LTERM [98](#)  
配布不能データ、送達不能キュー [108](#)  
非 SNA 3270 装置  
画面サイズとモデル情報 [78](#)  
非同期出力  
宛先、有効 [107](#)  
プリンター  
概要 [100](#)  
定義 [101](#)  
プリンターの共用 [102](#)  
ユーザー記述子  
インストール先作成 [89](#)  
システム定義中の作成 [88](#)  
選択の基準 [88](#)  
ノード・ユーザー記述子 [89](#)  
DFSUSER [90](#)  
要件 [72](#)  
用語 [63](#)  
利点  
可用性 [71](#)  
LTERM [71](#)  
ログオフ [109](#)  
ログオン (logon)  
動的、特定の端末タイプへの制限 [94](#)  
ログオン記述子  
システム定義中の作成 [85](#)  
NTO 端末、3600/ 金融機関端末 [87](#)  
3600/ 金融機関 [111](#)  
ABENDU0015 [93](#)  
DFS2085 [96](#)  
DFS3641W [93](#)  
DFS3645 [96](#)  
DFS3649A [97](#)  
DFS3650I [98](#)

拡張端末オプション (ETO) (Extended Terminal Option (ETO)) (続記述子 (続き))  
 DFS3672 [96](#)  
 DFSINSX0 [90](#)  
 DFSSGNX0 [90](#)  
 DLQT [108](#)  
 ETO 開始 [93](#)  
 ISC TCP/IP [63](#)  
 LTERM  
   制御ブロックの作成および再利用 [94](#)  
 LU 2  
   画面サイズとモデル情報 [78](#)  
 MFS [82](#)  
 MFS 装置記述子  
   MFS DCT ユーティリティ (DFSUTB00) [91](#)  
 MFS 装置特性テーブル  
   画面サイズとモデル情報 [78](#)  
 MFS DCT ユーティリティ [78](#)  
 MSC (複数システム結合機能)  
   記述子 (descriptor) [83](#)  
   サポート [83](#)  
   MSNAME マクロ [83](#)  
 SLU P [111](#)  
 SNA コマンド [94](#)  
 STSN 端末  
   概要 [111](#)  
 VTAM CINIT LUNAME [86](#)  
 拡張プログラム間通信機能 (APPC)/IMS  
   エラー抽出サービス [45](#)  
 確定応答  
   応答要件 (図) [978](#)  
   金融機関通信システム [935](#)  
   SLU P [935](#)  
 確認応答  
   OTMA ACK タイムアウト間隔 [825](#)  
 確立、外部サブシステムへの接続の [122](#)  
 仮想記憶通信アクセス方式 (VTAM) [376](#)  
 カップリング・ファシリティー (coupling facility)  
   定義 [383](#)  
 過負荷の回避、MSC [702](#)  
 画面  
   保護 [407](#)  
   無保護画面オプション [446](#)  
 画面保護、金融機関通信システム  
   BID オプション [948](#)  
 空キューの指示 [542](#)  
 空出力メッセージ  
   送信されない時 [946](#)  
   送信される時 [977](#)  
 管理  
   キュー制御機能 [808](#)  
   高速機能 (Fast Path) [808](#)  
   ETO [71](#)  
   MFS [440](#)  
   MSC [807](#)  
   MSC (複数システム結合機能) [701](#)  
   XRF [808](#)  
 キーボード・ショートカット [xxv](#)  
 疑似異常終了 U0830  
   回避 [744](#)  
 記述子  
   ログオン (logon)  
     ETO [65](#)  
   DFSOTMA 記述子 [800](#)  
   OTMA  
     OTMA (続き)  
       限度 [800](#)  
       最大 [800](#)  
   OTMA 宛先記述子  
     宛先名のマスキング [799](#)  
     定義 [799](#)  
   OTMA クライアント記述子 [796](#)  
   OTMA、指定 [795](#)  
 機能  
   金融機関通信システム [941](#)  
   コンポーネントの定義 [941](#), [942](#)  
   端末応答モード [942](#)  
   表示画面保護 [948](#), [949](#)  
   メッセージ形式サービス [946](#)  
   メッセージ・リカバリー [947](#)  
   IMS-CICS セッションで使用可能 [603](#), [604](#), [639](#)  
   ISC [469](#)  
   ISC 対 MSC [456](#)  
   ISC と CICS [463](#), [603](#), [604](#)  
   ISC に対するテスト・モード [470](#)  
   SLU P [941](#)  
 機能打ち切り  
   ループの検出、ISC [551](#)  
   LUSTATUS コマンドで指示された [543](#)  
 機能管理 (FM) ヘッダー  
   エラー・リカバリー・プロシージャ (ERP)  
     形式 [580](#)  
   金融機関通信システム [967](#), [973](#)  
   出力プロセス名 [577](#)  
   出力メッセージ (output message) [471](#)  
   セキュリティ [459](#)  
   タイプ X'42' (SLU P) [970](#)  
   入力 FM ヘッダーの長さ  
     ATTACH FM ヘッダー [557](#)  
   入力プロセス名 [577](#)  
   入力メッセージ [469](#)  
   メッセージ記述子バイト形式  
     出力 [973](#), [975](#)  
     入力 [967](#), [969](#)  
   要求される処理モード [471](#)  
   CICS で使用するためのコーディング [645](#)  
   IMS によって挿入される [447](#)  
   IMS、データ記述子 [651](#)  
 ISC  
   エラー・リカバリー・プロシージャ [557](#), [652](#)  
   応答 QMODEL FM ヘッダー [568](#)  
   概要 [555](#)  
   形式の参照 [571](#)  
   システム・メッセージ [559](#), [652](#)  
   データ記述子 [565](#), [579](#)  
   同期処理 [557](#)  
   入力 QMODEL FM ヘッダー [567](#)  
   プロセスの開始 [557](#)  
   ヘッダーの形式 [571](#)  
   ATTACH [557](#)  
   ATTACH, MFS [564](#)  
   DPM メッセージ、MFS [562](#)  
   IMS によってサポートされる、要約 [555](#)  
   ISC 編集の呼び出し [556](#)  
   MFS [561](#)  
   QMODEL [566](#), [581](#)  
   QMODEL, CICS [651](#)  
   RAP [558](#)



機能管理 (FM) ヘッダー (続き)

ISC (続き)

RAP、MFS [569](#)  
SCHEDULER [558](#), [649](#)  
SCHEDULER, MFS [564](#)  
SYSMSG [559](#)

MFS [651](#)

SLU P [968](#), [975](#), [976](#)

機能交換 (CAPEX)

CICS からの ISC TCP/IP セッションの開始 [618](#)  
CICS との ISC TCP/IP セッションの開始 [617](#)

基本エラー・リカバリー・プロシージャ [934](#)

基本編集

出力メッセージ・セグメント [447](#)  
ISC に伴う編集オプション [445](#)

基本編集 (basic edit)

出力メッセージ [445](#)  
入力メッセージ [444](#)  
入力メッセージ・セグメント [448](#)  
バイパス [446](#)  
非 MFS プログラム [479](#)  
IMS 機能 [444](#)  
IMS システム [435](#)  
ISC メッセージ [469](#)  
SLU 1 透過データ [445](#)

キャッシング・スキーム、ユーザー ID [830](#)

キュー

共用 [383](#)  
バランシング・グループ [390](#)  
論理端末 [402](#)

キュー・サブシステム [639](#)

キュー制御機能

共用キューのサポート [808](#)  
非共用キューのサポート [808](#)  
メッセージ・カテゴリーの識別 [809](#)

急送メッセージ・ハンドラー (EMH) (expedited message handler) (EMH))

キュー・オプション  
概要 [390](#)

キューの回転 [533](#)

競争、BIND [503](#)

共存

IMSplex および MSC [731](#)  
MSC および IMSplex [731](#)

共用キュー

環境

での操作の概説 [383](#)  
に必要なコンポーネント [384](#)  
のコンポーネントの図 [385](#)

疑似異常終了 U0830

回避 [744](#)

コミット後送信メッセージ [810](#)

出力メッセージのキュー・カウント [810](#)

送信後コミット・メッセージ [811](#)

定義 [383](#)

非送信請求メッセージ [810](#)

プログラム間メッセージ通信 [845](#)

プログラム間通信 [811](#)

メッセージ経路指定

MSC-IMSplex 構成内 [732](#)

利点 [384](#)

ALTPCB 出力と IMS Connect [179](#)

APPC および OTMA メッセージ

MSC リモート・トランザクションの処理 [738](#)

共用キュー (続き)

EMH キュー・オプション

概要 [390](#)

IMS Connect と ALTPCB 出力 [179](#)

IMSplex 内のリンク定義

削除 [737](#)

MSC (複数システム結合機能)

共存 [731](#)

MSNAME 定義と SYSID の共用 [734](#)

MSC MSNAME 定義

重複 [736](#)

MSNAME ステートメントと SYSID の共用 [734](#)

MSNAME 定義

削除 [737](#)

OTMA

ALTPCB 出力、リトリブ [812](#)

OTMA の使用可能化 [806](#)

SYSID

IMSplex と MSC ネットワークが共存する時の [738](#)

IMSplex 内の MSC SYSID の管理 [737](#)

IMSplex 内の MSC SYSID 複製 [738](#)

MSNAME 定義と SYSID の共用 [734](#)

T パイプ状況 [812](#)

z/OS システム・ログの役割 [385](#)

金融機関通信 [929](#)

金融機関通信システム

アプリケーション・プログラム

金融機関から SLU P への変換 [934](#)

使用可能な機能 [932](#)

MFS 考慮事項 [932](#)

主なパーツ [931](#)

機能 [941](#)

構成例 [932](#)

サポートされる端末 [931](#)

システムのテークオーバーに関する考慮事項 [938](#)

ブラケットと送信 / 受信のプロトコル [938](#), [966](#)

メッセージ再同期 (message resynchronization)

コントローラー・アプリケーション・プログラム  
[959](#)

ワークステーション [932](#)

VTAM

機能 [934](#)

コマンドと標識 [935](#)

XRF 複合システム、接続の確立 [938](#)

クライアント

発行されたコマンド [865](#)

IMS へのハイパフォーマンス・アクセス [776](#)

OTMA に接続することのできる数 [776](#)

OTMA、XRF 環境 [808](#)

クライアント記述子、OTMA

限度 [800](#)

クライアント通信

および IBM WebSphere [148](#)

Local オプション [148](#)

クライアント・ビッド要求

メッセージ接頭語の内容 [858](#)

メッセージ・フロー [858](#)

クラス 1 端末

概要 [418](#)

状況リカバリー・モード [410](#)

XRF サポート [418](#)

クラス 2 端末

概要 [419](#)

状況リカバリー・モード [411](#)

クラス 2 端末 (続き)  
XRF サポート [418](#)  
クラス 3 端末  
概要 [419](#)  
状況リカバリー・モード [411](#)  
XRF サポート [418](#)  
経路指定  
メッセージ  
MSC-IMSpIex 構成内 [732](#)  
経路指定出口ルーチン  
端末経路指定出口ルーチン [751](#)  
プログラム経路指定 [752](#)  
リンク経路指定 [751](#)  
MSC (複数システム結合機能) 会話 [704](#)  
言語インターフェース [140](#)  
言語インターフェース・エントリー・ポイント [140](#)  
言語インターフェース・モジュール [140](#)  
検査  
システム間のトランザクション定義 [720](#)  
リモート宛先 [700](#)  
IMS 端末サポート [394](#)  
交換単位コード、ATTACH FM ヘッダー [573](#)  
交渉可能バインド [502](#)  
構成  
金融機関通信システム [932](#)  
ISC [460](#)  
SLU P システム [932](#)  
VTAM [429](#)  
構造  
作成と削除 [67](#)  
対、定義の [384](#)  
リスト  
オーバーフロー、定義 [385](#)  
基本、定義 [385](#)  
定義 [384](#)  
構造リカバリー・データ・セット (SRDS) (structure recovery data set (SRDS))  
概要 [385](#)  
高速機能  
概要 [379](#)  
出力メッセージ [942](#)  
入力メッセージ [942](#)  
メッセージ [379](#)  
リカバリー可能な照会トランザクション [432](#)  
MFS ページ出力メッセージ [942](#)  
高速機能 (Fast Path)  
宛先コード [390](#)  
アプリケーションの制限 [391](#)  
端末 [415](#)  
端末応答モード [942](#)  
入力編集/経路指定出口ルーチン (DBFHAGU0) [391](#)  
パラメーター  
FORCERESP [953](#)  
FPACK/NFPACK [954](#)  
OPTACK [953](#)  
リカバリー [410](#)  
リカバリー可能な照会トランザクション [980](#)  
EMH バッファ・サイズ [415](#)  
MFS NEXTMSG または NEXTMSGP 制御コマンド [942](#)  
RTR コマンド [954](#), [955](#)  
TERMINAL マクロ  
オプション [953](#)  
必須パラメーター [953](#)  
TERMINAL マクロの NFPACK オプション [954](#), [955](#)

高速機能 EMH  
共用キュー環境  
概要 [390](#)  
高速機能可能システムによるメッセージ・バッファ [391](#)  
高速機能の管理 [808](#)  
高速ネットワーク再接続  
および IMS シャットダウン [423](#)  
サインオン・セキュリティ [424](#)  
サポート・レベルの指定 [421](#)  
サポート・レベルの変更 [422](#)  
持続セッション・トラッキング [422](#)  
端末再接続プロトコル [423](#)  
VTAM に対する RNR サポート・レベルの定義 [434](#)  
VTAM に対する持続サポート・レベルの定義 [434](#)  
VTAM の定義 [433](#)  
XRF または VGR との併用 [423](#)  
構文、メッセージ接頭語の [889](#)  
構文図  
読み方 [xxiii](#)  
候補プリンター [451](#)  
コールアウト  
並列 RESUME TPIPE 要求に対する OTMA T パイプのサポート [792](#)  
T パイプ、並列 RESUME TPIPE 要求に対するサポート [792](#)  
コールアウト機能  
アプリケーション・プログラムの概要 [884](#)  
非同期  
コールアウト要求への応答の関連 [886](#)  
IMS Connect TCP/IP クライアント [883](#)  
IMS TM Resource Adapter  
サポートの概要 [883](#)  
z/OS Sysplex Distributor [883](#)  
コールアウト制御データ  
形式 [257](#)  
コールアウトのサポート  
非同期  
IMS Connect クライアントのプログラミング [882](#)  
コールアウト・メッセージ  
IMS Connect  
RESUME TPIPE 要求 [332](#)  
コールアウト要求  
同期  
MULTIRTP [887](#)  
OTMA 構成の概要 [877](#)  
OTMA サポート [878](#)  
RESUME TPIPE 要求の並列処理 [887](#)  
OTMA  
非同期コールアウト・サポートの実装 [881](#)  
OTMA サポート [877](#)  
コールド・スタート、リカバリー ISC セッション [506](#)  
コマンド  
応答要件 [982](#)  
クライアントにより発行された [865](#)  
システムの開始に使用される [957](#)  
トランザクション応答 [621](#)  
非同期応答 [606](#)  
CBresynch [866](#)  
CICS EXEC  
使用可能な機能 [603](#), [604](#)  
非同期 API [603](#), [604](#)  
プログラム・フロー [621](#)  
DFC プロトコルの作成 [603](#), [604](#)  
CICS-IMS セッション [643](#)



## コマンド (続き)

CRC (コマンド認識文字) [125](#)  
DEQUEUE TMEMBER [819](#)  
DISPLAY STATUS [819](#)  
DISPLAY TMEMBER [819](#)  
DISPLAY TRACE [819](#)  
IMS  
    システム開始 [431](#)  
    並列 ISC セッション [470](#)  
ISC セッションからの発行 [470](#)  
ISC セッション障害のリカバリー [508](#)  
LUSTATUS プロトコル、ISC [543](#)  
PRN, IMS コマンド上で指定されない [577](#)  
RACF による保護 [414](#)  
REPresynch [866](#)  
REQresynch [866](#)  
SRVresynch [866](#)  
START TMEMBER [819](#)  
STOP TMEMBER [819](#)  
TBresynch [866](#)  
TRACE TMEMBER [819](#)  
VTAM  
    コマンドと標識 [466, 935](#)  
    受信準備済み (RTR) [545](#)  
    リカバリー要求 [986](#)  
    BID [944](#)  
    BIS [554, 638](#)  
    CANCEL [536, 985](#)  
    CHASE [538](#)  
    CICS による LUSTATUS [606, 642](#)  
    LUSTATUS コマンド [542, 985](#)  
    RSHUT [545](#)  
    RTR [985](#)  
    SIGNAL [553, 986](#)  
/CHANGE [141](#)  
/DBDUMP DATABASE [850](#)  
/DBRECOVERY AREA [850](#)  
/DISPLAY SUBSYS [141](#)  
/DISPLAY TMEMBER [812](#)  
/DISPLAY TRANSACTION [774](#)  
/SECURE OTMA NONE [833](#)  
/SSR [125](#)  
/START SUBSYS [123](#)  
/START AREA [850](#)  
/START REGION [850](#)  
/STOP AREA [850](#)  
/STOP REGION [850](#)  
/STOP SUBSYS [123](#)

コマンド認識文字 (CRC) [125](#)

コマンド・レベル API、CICS [603, 604](#)

コミット (commit)

    作業単位 [504, 652](#)  
    サンプル・フロー [852](#)  
    処理 [850](#)  
    モード [850](#)  
    LUSTATUS コマンド [543](#)

コミット後送信

    転送機能 [303](#)  
    ページ機能 [301](#)  
    OTMA、出力の管理 [872](#)  
    OTMA、出力の転送 [873](#)  
    OTMA、出力のページ [873](#)  
    OTMA、スーパーメンバー [875](#)

コミット後送信交換

コミット後送信交換 (続き)

    OTMA メッセージ接頭語の内容 [860](#)

    OTMA メッセージ・フロー [860](#)

コミット後送信フロー [852, 860](#)

コミット・モード

    コミット・モード 0 [293](#)

    コミット・モード 1 [293](#)

    CM1 タイムアウト値 [293](#)

コミット・モード 0

    転送機能 [303](#)

    ページ機能 [301](#)

コンテキスト [842](#)

コンテンション、ブラケット [503, 951](#)

コンポーネントの定義

    選択

        出力コンポーネント [477, 941](#)

        入力コンポーネント [474, 941](#)

        SLU P 入力コンポーネント [942](#)

    パラメーターの定義 [532](#)

    LTERM 命名 [473, 941](#)

    SLU P システム [941](#)

コンポーネント保護

    拡張出力 [949](#)

    状態 [407](#)

## [サ行]

サーバー

    OTMA 定義 [774](#)

サーバー resynch [866](#)

サーバー使用可能交換

    OTMA メッセージ接頭語の内容 [859](#)

    OTMA メッセージ・フロー [859](#)

サーバー状態プロトコル・コマンド

    概要 [815](#)

サービス

    ATRABCK [862](#)

    ATRACMT [862](#)

    ATREINT [862](#)

    CRGGRM [862](#)

    CRGSEIF [862](#)

    CTXBEGC [862](#)

    CTXEINT [863](#)

    CTXSWCH [863](#)

サービス、ESAP で使用できる [126](#)

再始動

    利用可能な出力、ISC [507](#)

    CICS アプリケーションのコーディング [657](#)

再始動処理

    OTMA

        タイムアウト・キューに転送されるメッセージ [813](#)

再始動およびリカバリー [652](#)

再始動トランザクション、CICS [627](#)

再同期

    サンプル・メッセージ [871](#)

    据え置き [868](#)

    フロー [867](#)

    OTMA プロトコル [866](#)

    OTMA、概要 [863](#)

再同期、メッセージ [504](#)

サインオフ [382](#)

サインオフの定義 [382](#)

サインオン処理、外部サブシステム接続のための [122](#)

サインオン出口ルーチン (DFSSGNX0)

サインオン出口ルーチン (DFSSGNX0) (続き)

ETO

印刷の関連付け [101](#)

サインオンと静的端末 [427](#)

サインオンの定義 [382](#)

作業域の作成、ESAP の [137](#)

作業単位

コミット [504, 652](#)

セッション開始の状況 [511](#)

定義 [504](#)

バックアウト [504, 652](#)

保留状態、一方的な決定 [506](#)

例 [504](#)

CICS [652](#)

作業負荷の分散、MSC [702](#)

作成、ESAP の作業域の [137](#)

作成、ESMT の [127](#)

サブシステム

外部、定義 [380](#)

キュー [639](#)

直接制御 [639](#)

サブシステム始動サービス

使用 [138](#)

サブシステム終了 [142](#)

サブシステム接続 [122](#)

サブプール・ユーザー

動的割り振り [473](#)

VTAM [473](#)

サポート環境、IMS の [774](#)

サンプル

メッセージ (message)

応答 (response) [919](#)

クライアント・ビッド [919](#)

トランザクション (transaction) [919](#)

OTMA メッセージ [919](#)

サンプル・プログラム

IMS と CICS との間での ISC の使用 [673](#)

シーケンス番号

回復可能 [849](#)

管理 [516, 959](#)

使用 [509, 959, 960](#)

ストレージ [516](#)

説明 [508](#)

送信シーケンス番号 [849](#)

定義 [848](#)

保守 [505](#)

時間設定

IMS Connect

概要 [319](#)

IMS DB クライアント [319](#)

IMS TM クライアント [319](#)

IMS 間 TCP/IP 接続 [329](#)

IMS Connect タイムアウト間隔 [320](#)

識別プロセス [122](#)

システム

定義

IMS ISC サンプル [491, 496](#)

メッセージ、長さ [560](#)

システム ID (SYSID) [693](#)

システム間連絡 (ISC)

SYSERROR [589](#)

SYSSTAT [589](#)

アプリケーション・プログラム

アクセス [478](#)

システム間連絡 (ISC) (続き)

アプリケーション・プログラム (続き)

経路指定の例 [481, 483-488](#)

MFS 不使用 [479](#)

エラー処理

送信側 ERP [550](#)

通知済み MTO [545](#)

エラー・リカバリー・プロシージャ

例、受信側検出エラー [670](#)

例、送信側検出エラー [669](#)

例 [669](#)

概要 [455](#)

機能 [469](#)

経路指定の例 [481](#)

経路指定パラメーター [480](#)

交換単位コード、ATTACH FM ヘッダー [573](#)

再同期プロシージャ [505](#)

サポートされるトランザクション・タイプ、CICS セッ

ション [603, 604, 642](#)

システム定義

マクロ・パラメーター [494](#)

実行モード

外部指定 [471](#)

出力プロトコル [477](#)

出力編集オプション [469](#)

静的定義 [491](#)

センス・コード

受信された [552](#)

選択受信側 ERP

X'0864xxxx': 機能打ち切り [546](#)

X'0865xxxx': 機能打ち切り [547](#)

X'0866xxxx': 機能打ち切り [548](#)

データ記述子 FM ヘッダー

入力で使用 [565](#)

データ・フロー制御

例 [659, 660, 664, 665, 667](#)

テスト・モード [470](#)

同期点標識 [524](#)

ネットワーク、開始 [499](#)

ネットワーク、開始 [499](#)

ネットワーク操作 [499, 500](#)

ノード定義

マクロ使用 [491](#)

ハーフセッション、1 次側と 2 次側 [502](#)

複数チェーン入力メッセージの交換制限 [573](#)

プロトコル

定義 [499](#)

編集オプション [445](#)

編集機能

概要 [469](#)

メッセージ再同期 (message resynchronization)

セッション [504](#)

メッセージ通信

ATTACH FM ヘッダー [571](#)

メッセージの保水性、CICS [653](#)

ATTACH FM ヘッダー

形式 [571](#)

ビット内容 [571](#)

CICS

アウトバウンド宛先プロセス名フィールド

(ATTPDN) [646](#)

アウトバウンド宛先プロセス名フィールド

(SCDDPN) [649](#)

異常終了、トランザクション [656](#)

システム間連絡 (ISC) (続き)

CICS (続き)

交換単位コード (ATTIU) [647](#)  
サンプル・プログラム DFSISCOO [674](#)  
サンプル・プログラム、ACB 生成 [680](#)  
サンプル・プログラム、MFS 形式 [678](#)  
サンプル・プログラム、PSB 生成 [679](#)  
サンプル・プログラム、インストール [673](#)  
サンプル・プログラム、コンパイルとバインドのための JCL [676](#)  
サンプル・プログラム、システム定義ステートメント [677](#)  
セッション再始動、TCP/IP [620](#)  
セッション終了、TCP/IP [619](#)  
セッション終了、TCP/IP、無条件 [620](#)  
セッション終了、異常、TCP/IP [620](#)  
セッション終了、正規手順での、TCP/IP [620](#)  
セッションの開始 [637](#)  
セッションの数 [631](#)  
セッション・パラメーター [631](#)  
セッション名 [631](#)  
端末の装置依存データ [463](#)  
データ・ストリーム・プロファイル (ATTDSP) [647](#)  
トランザクションの異常終了 [656](#)  
ネットワーク名 [630](#)  
非同期トランザクション、定義 [636](#)  
非ブロック化アルゴリズム (ATTDBA) [648](#)  
未確定処理 [636](#)  
戻り1次リソース名 (ATTRPRN) [647](#)  
戻り1次リソース名 (SCDRPRN) [650](#)  
戻り宛先プロセス名 (ATTRDPN) [647](#)  
戻り宛先プロセス名 (SCDRDPN) [649](#)  
1次リソース名フィールド (ATTPRN) [646](#)  
1次リソース名フィールド (SCDPN) [649](#)  
ATTACC [648](#)  
ATTDBA [648](#)  
ATTDSP [647](#)  
ATTDPN [646](#)  
ATTDQN [647](#)  
ATTDSP [647](#)  
ATTIU [647](#)  
ATTPRN [646](#)  
ATTRDPN [647](#)  
ATTRPRN [647](#)  
SCDDP [650](#)  
SCDDPN [649](#)  
SCDDQN [650](#)  
SCDPN [649](#)  
SCDRDPN [649](#)  
SCDRPRN [650](#)  
TCP/IP セッション再始動 [620](#)  
TCP/IP セッション終了 [619](#)  
TCP/IP セッション終了、異常 [620](#)  
TCP/IP セッション終了、正規手順での [620](#)  
TCP/IP セッション終了、無条件 [620](#)  
TCP/IP セッションの再始動 [620](#)  
TCP/IP セッションの終了 [619](#)  
TCP/IP セッションの終了、正規手順での [620](#)  
TCP/IP セッションの終了、無条件で [620](#)

CICS-IMS [601](#)

DFC 同期点

出力で [521](#)

リカバリー可能メッセージ [522](#)

リカバリー不能メッセージ [524](#)

システム間連絡 (ISC) (続き)

DFC 同期点 (続き)

例外 [521](#)

MFS メッセージ [522](#)

DFC プロトコル [528](#)

ETO [63](#)

FM ヘッダー

SYSERROR [589](#)

SYSSTAT [589](#)

概要 [555](#)

メッセージの経路指定 [479](#)

IMS によってサポートされる [555](#)

SCHEDULER [588](#)

IMS 間セッション

MSC と ISC の共存 [489](#)

IMS コマンド、発行 [470](#)

IMS サポート、CICS [601](#)

ISC セッション

サポートされないコマンド [628](#)

ISC セッションのバインド [502](#)

ISC ノード

定義 [629](#)

ISC プロトコル

通信の設計 [477](#)

ISC 編集 ATTACH パラメーターの例 [591](#)

MFS における ATTACH および SCHEDULER パラメーターの例 [595](#)

MSC の共存 [489](#)

SC プロトコル

セッションの開始 [500](#)

SCHEDULER [588](#)

TCP/IP

セッション再始動 [620](#)

セッション終了 [619](#)

セッション終了、異常 [620](#)

セッション終了、正規手順での [620](#)

セッション終了、無条件 [620](#)

セッションの再始動 [620](#)

セッションの終了 [619](#)

セッションの終了、正規手順での [620](#)

セッションの終了、無条件で [620](#)

ETO [63](#)

IMS Connect、サポートの概要 [153](#)

TCP/IP サポート

概要 [608](#)

静的端末定義 [611](#), [612](#)

制約事項 [610](#)

セキュリティー [610](#)

セッション、開始 [617](#)

セッションの開始、CICS から [618](#)

端末定義 [611](#)

動的端末定義 [611](#)

要件 [609](#)

リンクの定義、CICS への [615](#)

リンクの定義、IMS Connect への [614](#)

CICS フロントエンド・トランザクション・タイプ [621](#)

TCP/IP から VTAM への切り替え [613](#)

VTAM へのフォールバック [613](#)

VTAM 機能 [465](#)

XRF 複合システム (XRF complex) [501](#)

システム間連絡機能 (ISC)

アプリケーション・プログラム

例 [673](#)

システム間連絡機能 (ISC) (続き)

- エラー・リカバリー、例外応答後のページ [538](#)
- 拡張会話方式終了 [520](#)
- 制御ブロック・ストレージ、並列セッション [471](#)
- 対 CICS [642](#)
- プロトコル
  - IMS 間セッションの制約事項 [491](#)
- 編集機能
  - FM ヘッダーの呼び出し [556](#)
- メッセージ経路指定
  - FM ヘッダー [479](#)
- ATTDBA [574](#)
- ATTDSP [574](#)
- CICS-IMS
  - アプリケーション関連の概念 [639](#)
- FM ヘッダー
  - 形式の参照 [571](#)
  - ATTDBA [574](#)
  - ATTDSP [574](#)
  - CICS セッション [645](#)
  - MFS [561](#)
  - QGET [581](#)
  - QGETN [582](#)
  - QPURGE [583](#)
  - QSTATUS [584](#)
  - QXFR [585](#)
  - RAP [588](#)
- IMS 間セッション
  - 設計の考慮事項 [489](#)
- IMS コマンドを発行する並列セッション [470](#)
- ISC 編集 ATTACH パラメーターの例 [591](#)
- QGET [581](#)
- QGETN [582](#)
- QPURGE [583](#)
- QSTATUS [584](#)
- QXFR [585](#)
- RAP [588](#)
- SC (セッション制御) プロトコル [500](#)
- SYMSMSG における ATTACH パラメーターの例 [593](#)
- XRF
  - USERVAR、セッション開始 [501](#), [502](#)

システム制御域 (SCA) (system control area (SCA)) [478](#)

システム定義

- MSC (複数システム結合機能)
  - マクロ [710](#)
- システム定義 (system definition)
  - マクロ
    - MSC マクロ [705](#)
- MSC
  - 一般的考慮事項 [705](#)
  - 含意 [719](#)
  - 検査 [720](#)
  - パートナー [711](#)
- MSC (複数システム結合機能)
  - 出口ルーチン [718](#)
  - リンク優先順位の設定 [716](#)
  - ローカル [710](#)

システム・リソース [814](#)

システム・ログ

- MSC [729](#)

システム・ログ、z/OS

- 共用キューおよび [385](#)

シスプレックス環境

- 定義 [383](#)

シスプレックス環境 (続き)

- での共用キュー [383](#)
- 持続セッション・トラッキング
  - 終了 [422](#)
- 持続ソケット接続 [309](#)
- 実行モード
  - 指定、ISC [471](#), [472](#)
  - CICS 通信 [603](#), [604](#)
  - ISC [603](#), [604](#)
  - ISC セッション障害のリカバリー [508](#)
  - ISC でサポートされる、リスト [458](#)
- 自動化操作プログラム [391](#)
- 自動ページ化出力、同期 [606](#)
- シャットダウン
  - OTMA
    - 概要 [813](#)
    - クライアント通知 [813](#)
- 修飾、LU 名 [53](#)
- 修正済みのアプリケーション・プログラム
  - リモート実行、MSC [38](#)
  - LU 6.2 記述子 [45](#)
- 従属領域
  - コミット後送信トランザクション [825](#)
  - 送信後コミット・トランザクション [825](#)
  - OTMA の使用 [825](#)
- 従属領域接続 [143](#)
- 終了 ECB [123](#), [142](#)
- 終了、外部サブシステムとの接続の [123](#)
- 終了、セッション [513](#)
- 終了
  - OTMA
    - 概要 [813](#)
    - クライアント通知 [813](#)
- 終了ブラケット標識
  - 使用 (図) [938](#)
  - 定義 [477](#)
  - ATTACH と一緒に使用 [472](#)
  - LUSTATUS コマンド [542](#)
- 終了要求、外部サブシステムからの [142](#)
- 受信準備済み (RTR) コマンド [545](#)
- 受信用バッファ [427](#)
- 出力
  - 制御 [449](#)
  - ISC 再始動で利用可能 [507](#)
  - MFS DPM [973](#)
- 出力アルゴリズム
  - RU チェーン [575](#)
  - VLVB [575](#)
- 出力エラー、MFS オンライン検出 [563](#)
- 出力応答、メッセージ・タイプ別に要求される [979](#)
- 出力機能管理ヘッダー、ISC [645](#)
- 出力コンポーネント
  - 選択
    - 識別番号 [474](#), [941](#)
    - システム・メッセージ [559](#), [941](#)
    - 変更 [473](#), [941](#)
    - 定義 [477](#)
    - 入力コンポーネントとの関係 [472](#)
  - 出力コンポーネント ID バイト、出力メッセージ [974](#)
  - 出力コンポーネント保護、拡張 [949](#)
  - 出力制御 [449](#)
  - 出力装置、タイプによる制御文字 [448](#)
  - 出力ブラケット付け [977](#)
  - 出力プロトコル、判別 [477](#)

出力編集オプション、SLU P [950](#)  
出力メッセージ (output message)  
一時停止 [963](#)  
金融機関通信システム  
タイプ読み取りフィールド (SMSCRT) [971](#)  
複数回伝送 [971](#)  
フラグ読み取りフィールド (SMSCRF/SMSCRE) [971](#)  
高速機能 (Fast Path)  
金融機関システム [954](#)  
SLU P システム [954](#)  
コミット時 [521](#)  
出力 FM ヘッダー  
応答要求 [978](#)  
金融機関形式 [973](#)  
メッセージ記述子バイト (SLU P) [975](#)  
メッセージ記述子バイト (金融機関) [974](#)  
ID バイト [974](#)  
MFS データ・バイト (SLU P) [976](#)  
MFS データ・バイト (金融機関) [974](#)  
SLU P 形式 [975](#)  
出力ブラケット付け [977](#)  
説明 [971](#)  
タイプ [971](#)  
フォーマット設定、活動化 MFS [561](#), [977](#)  
ブラケット間  
処理方法 [944](#)  
図 [944](#)  
設計の考慮事項 [945](#)  
メッセージ通信 [981](#)  
分割 [971](#)  
要求された同期点、ISC [521](#)  
ISC のブラケット化 [532](#)  
MFS DPM [562](#)  
NULL  
送信される時 [977](#)  
目的 [971](#)  
SLU P システム  
タイプ読み取りフィールド (SMSCRT) [971](#)  
複数回伝送 [971](#)  
出力メッセージ構造 [370](#)  
順次バッファリング  
CICS [764](#)  
障害  
イン・ブラケットでの [505](#)  
システム、CICS-IMS セッション時 [653](#)  
CICS-IMS セッション [653](#)  
ISC、イン・ブラケットからの回復 [506](#)  
照会フラグの処理 [124](#)  
状況  
回復可能 [408](#)  
有効な  
エンド・ユーザー [408](#)  
コマンド [408](#)  
リカバリー不能 [408](#)  
リソース  
種別 [408](#)  
状況リカバリー・モード (status recovery mode)  
リソース・タイプ [409](#)  
GLOBAL [408](#)  
LOCAL [409](#)  
NONE [409](#)  
RCVYCONV= [409](#)  
RCVYFP= [409](#)  
RCVYRESP= [409](#)  
状況リカバリー・モード (status recovery mode) (続き)  
RCVYSTSN= [409](#)  
状態データ  
形式  
クライアント・ビッド・コマンド [899](#)  
使用可能サーバー・コマンド [899](#)  
トランザクション関連情報 [904](#)  
REPresynch コマンド [903](#)  
SRVresynch コマンド [902](#)  
T パイプの保留キューの出力再開のため [912](#)  
TBresynch コマンド [904](#)  
メッセージ接頭語のセクション [899](#)  
商標 [987](#), [989](#)  
情報管理システム (IMS) (Information Management System (IMS)) [378](#)  
処理、類似性の  
MSC-IMSplex 構成内 [733](#)  
垂直区分化 [683](#)  
スーパーメンバー  
管理、CMO 出力 [875](#)  
並列 RESUME TPIPE 要求  
使用可能化 [794](#)  
並列処理 [794](#)  
IMS Connect [158](#)  
IMS Connect サポート [346](#)  
IMS 間 TCP/IP 通信 [158](#), [802](#)  
MULTIRTP [793](#), [794](#)  
OTMA [875](#)  
RESUME TPIPE 要求  
複数のアクティブな、サポート [793](#)  
並列処理 [793](#)  
据え置きプログラム間通信 [400](#)  
スクラッチパッド域 (SPA) (scratchpad area (SPA)) [399](#)  
スケジューラー・メッセージ・ブロック (SMB) (Scheduler Message Block (SMB))  
OTMA [778](#)  
スケジューリング  
アルゴリズム [398](#)  
高速機能メッセージ [390](#)  
スプール出力制御 [450](#)  
スレーブ MSC とマスター MSC (複数システム 結合機能) の関係 [683](#)  
スレッド、外部サブシステム [122](#)  
制御機能、MFS [435](#)  
制御ブロック  
ISC [471](#)  
MFS [442](#)  
制御ブロックのマッピング  
EEVT [129](#)  
制御文字 [448](#)  
正常なセッション終了  
CICS [638](#)  
ISC [513](#)  
VTAM [963](#)  
制約の使用不可化  
リソース・タイプ整合性 [390](#)  
リソース名の一意性 [389](#)  
セキュリティ  
中間 IMS  
MSC [722](#)  
トラステッド・ユーザー  
IMS Connect [206](#)  
ネットワーク・セキュリティ資格情報の伝搬 [186](#)  
パスチケット



セキュリティー (続き)  
   パスチケット (続き)  
     IMS Connect [201](#)  
   パスワード  
     変更、IMS Connect [198](#)  
     IMS Connect 大/小文字混合パスワード [200](#)  
   パスワード・フレーズ  
     変更、IMS Connect [199](#)  
   分散セキュリティ資格情報の伝搬 [186](#)  
   メッセージ接頭語の OTMA セキュリティー・データ・セクション [914](#)  
   APPC トランザクション [58](#)  
   DRDA クライアント [202](#)  
   IMS Connect  
     概要 [161](#)  
     サンプル出口からの RACROUTE 呼び出しでのエラー [193](#)  
     大/小文字混合パスワード [200](#)  
     出口ルーチン [190](#)  
     デフォルトの RACF ユーザー ID [184](#)  
     トラステッド・ユーザー [206](#)  
     パスチケット [201](#)  
     パスワードの変更 [198](#)  
     パスワード・フレーズの変更 [199](#)  
     OTMA ACEE エージング値の指定 [207](#)  
     RACF、直接サポートの使用可能化 [183](#)  
     RACF、統計の使用可能化 [183](#)  
     RACF、汎用戻りコードまたはメッセージ [182](#)  
   IMS Connect 間の接続 [188](#)  
   IMS 間の TCP/IP 接続 [188](#)  
   ISC  
     ISC TCP/IP 接続 [610](#)  
   MSC  
     中間 IMS [722](#)  
   MSC (複数システム結合機能) [722](#)  
   OTMA  
     個別のメンバー・セキュリティの指定 [832](#)  
   OTMA ACEE エージング値  
     IMS Connect [207](#)  
   OTMA RACF セキュリティー・レベル [830](#)  
   OTMA クライアント、概要 [774](#)  
   RACF パスチケット [202](#)  
 セキュリティー・オプション  
   コマンド許可  
     ユーザー ID [413](#)  
     DFSCCMD0 コマンド許可出口ルーチン [414](#)  
   サインオン検査 [412](#)  
   セキュリティ・プロファイル  
     ユーザー ID [413](#)  
     DFSCTRNO トランザクション許可出口ルーチン [413](#)  
   トランザクション許可  
     RACF [413](#)  
   トランザクション・コマンド [414](#)  
   トランザクション・セキュリティ  
     RACF [58](#)  
     UACC (NONE) [58](#)  
   APPC/IMS、RACF [414](#)  
   APPC/IMS、SAF [414](#)  
   ETO  
     ユーザー ID [415](#)  
   password  
     ユーザー ID [414](#)  
   RACF [412](#)  
 セキュリティー・サポート  
   セキュリティ・サポート (続き)  
     パスチケット・リプレイ保護 [205](#)  
   セキュリティの考慮事項  
     CICS [644](#)  
   セキュリティ・レベル  
     CHECK [830](#)  
     FULL [831](#)  
     NONE [830](#)  
     PROFILE [830](#)  
   設計済みメッセージ [850](#)  
   セッション  
     確立 [382](#)  
     単一 [503](#)  
     定義 [377](#)  
     バインディング [502](#)  
     並列 [503](#)  
     リモート制御 [490](#)  
     ISC VTAM、設定 [500](#)  
     ISC のバインド要件 [502](#)  
   セッション開始  
     要求される  
       マスター端末オペレーター、IMS [501](#)  
   セッション終了  
     異常 [513](#), [638](#)  
     条件付き対無条件 [468](#)  
     正規手順での  
       開始済みの [554](#), [964](#)  
       順序 [554](#), [964](#)  
       定義 [964](#)  
       CICS によって要求された [638](#)  
     正常 [513](#), [638](#)  
     即時  
       ステーション [513](#)  
       定義 [638](#), [963](#)  
       ネットワーク・オペレーター、z/OS VTAM [513](#), [964](#)  
       マスター端末オペレーター、IMS [513](#), [963](#)  
       ワークステーション [964](#)  
     定義 [513](#), [963](#)  
     要約 (図) [963](#)  
     BID オプションの指定、影響 [984](#)  
     CICS [638](#)  
     ISC の対称 (SBI/BIS) [554](#)  
   セッションの開始  
     可能なセッション状態 [511](#)  
     段階的な説明 [958](#)  
     伝送順序 [958](#)  
     バインド・パラメーター  
       金融機関通信システム [958](#)  
     要求された  
       システム定義 (system definition) [500](#), [957](#)  
       ネットワーク・オペレーター、z/OS VTAM [500](#), [957](#)  
       マスター端末オペレーター、IMS [957](#)  
       ワークステーション [957](#), [970](#)  
     CICS [636](#)  
     要求の方法 [431](#)  
     IMS-CICS [636](#)  
     ISC VTAM [500](#)  
   セッションの割り振り、CICS [636](#)  
   セッション・パラメーター  
     使用 [501](#), [937](#)  
     接続の設定 [501](#), [937](#)  
   セッション・ローカル・フラグ、ATTACH FM ヘッダー [578](#)  
   接続  
     イムス

## 接続 (続き)

イムス (続き)

前提知識 [xxiii](#)

説明 [467, 937](#)

IMS セッション・パラメーター

更新に対する制約事項 [937](#)

ISC [500, 502](#)

XRF 複合システム、接続の確立 [938](#)

接続の設定 [467](#)

接続ファクトリー

IMS Universal Database リソース・アダプター [6](#)

設定、照会パラメーターの [124](#)

設定およびテスト・シーケンス番号 (STSN) コマンド [514](#)

接頭部

規則 [849](#)

構文 [889](#)

OTMA メッセージの [848](#)

センス・コード

エラー [552](#)

選択受信側 ERP [545](#)

送信側 ERP [550, 551](#)

定義 [552](#)

ISC 時に受信された [552](#)

ISC 中に送信された [553](#)

選択受信側 ERP

説明 [545](#)

センス・コード [545](#)

全二重メッセージ・フロー [780](#)

専用コンテキスト [842](#)

関連

非同期コールアウト要求への応答の関連 [886](#)

関係係数トークン [780](#)

送信 / 受信とブラケット・プロトコル [966](#)

送信 / 受信のプロトコル

ブラケット付け

出力 [977](#)

入力 [970](#)

IMS [966](#)

送信側 ERP、センス・コード [550](#)

送信後コミット

確認フロー付き [856](#)

フロー [855](#)

ACK/NAK タイムアウト値 [293](#)

送信後コミット・トランザクション (CM1)

ACK タイムアウト [827](#)

送信後コミットのリカバリー [851](#)

送信されるリカバリー可能メッセージ、シーケンス番号 [959](#)

送信専用トランザクション

出力の転送 [303](#)

送信専用プロトコル

確認応答付き [307](#)

順次配信付き [308](#)

装置クラス制御 [448](#)

装置出力形式 (DOF) [442](#)

装置入力形式 (DIF) (device input format (DIF)) [442](#)

即時セッション終了 [513](#)

即時プログラム間通信 [400](#)

ソケット

警告しきい値 [316](#)

警告用のパーセンテージの設定 [316](#)

予約 [315](#)

リセットしきい値 [316](#)

IMS Connect

トランザクションの処理 [312](#)

ソケット (続き)

IMS Connect の最大と UNIX システム・サービス [314](#)

IMS Connect 用の最大数 [313](#)

UNIX システム・サービスのソケット制限 [314](#)

ソケット接続

持続 [309, 350](#)

タイプ [309](#)

トランザクション (transaction) [310](#)

非持続 [310](#)

IMS Connect

IMS TM クライアントのソケット・タイプの設定 [311](#)

IMS 間 TCP/IP 通信 [312](#)

IMS 間 TCP/IP 通信

クリーンアップ [312](#)

持続性 [312](#)

終了シナリオ [312](#)

## [タ行]

大/小文字混合パスワード

IMS Connect [200](#)

帯域幅

MSC リンク [702](#)

代替 PCB、2 次トランザクション [399](#)

代替応答 PCB

使用 [942](#)

変更 [942](#)

代替機能、CICS [642](#)

タイプ 2 接続

WebSphere Application Server Liberty 構成の概要 [17](#)

WebSphere Application Server Liberty 構成例 [19](#)

タイプ 2 接続、IMS Universal Database リソース・アダプター

概要 [17](#)

タイプ 4 接続

構成例

WebSphere Application Server Liberty [15](#)

WebSphere Application Server Liberty 構成の概要 [13](#)

タイプ 4 接続 IMS Universal Database リソース・アダプター

概要 [13](#)

タイプ読み取りフィールド、出力メッセージ [971](#)

タイマー設定 [344](#)

タイムアウト

IMS Connect

概要 [319](#)

入力メッセージ [320](#)

IMS DB クライアント [319](#)

IMS TM クライアント [319](#)

IMS 間 TCP/IP 接続 [329](#)

OTMA ACK タイムアウト [825](#)

単一ストリームのプログラム間通信 [842](#)

単一セッション、ISC バインド要件 (IMS 間) [489](#)

端末

XRF 付きクラス [1 395](#)

応答モード (response mode) [405](#)

会話モード [405](#)

画面保護状態 [407](#)

コンポーネント保護状態 [407](#)

シスプレックス、内

リカバリー状況 [411](#)

状態 [406](#)

小バッファ装置 [449](#)

接続 [395](#)



端末 (続き)

操作モード [405](#)  
装置クラス制御 [448](#)  
定義 [375](#)  
停止状態 [406](#)  
テスト・モード (test mode) [406](#)  
入出力装置の分離 [402](#)  
排他モード [406](#)  
非交換通信ネットワーク [403](#)  
物理、定義 [73](#)  
プロファイル [425](#), [427](#)  
ページ保護状態 [407](#)  
モードと状態 [405](#), [407](#)  
要件の文書化 [394](#)  
ロック・モード [406](#)  
論理  
    キュー [402](#)  
    チェーン [401](#)  
    物理端末との関係 [401](#)  
    マスター端末 (master terminal) [404](#)  
COMPINOP 状態 [407](#)  
ETO と排他モード [406](#)  
IMS サポート [394](#)  
INOP 状態 [407](#)  
LU 6.2 端末と高速機能 [415](#)  
NTO [404](#)  
QERROR 状態 [406](#)  
QLOCK 状態 [406](#)  
SNA QUIESCE [407](#)  
VTAM で接続 [394](#)  
XRF 付きクラス 2 [395](#)

端末、ユーザー  
    装置選択 [931](#)  
    論理装置としての定義 [931](#)

端末管理コマンド  
    サポートされていない OTMA [814](#)

端末経路指定出口ルーチン [751](#)

端末再接続プロトコル [423](#)

端末モード  
    応答モード (response mode) [405](#)  
    会話モード [405](#)  
    排他的 [406](#)  
    ロック・モード [406](#)  
    SNA QUIESCE [407](#)

チェーン、論理端末 [401](#)  
チェーンの最終での静止コマンド、VTAM [963](#)  
チェーン・メッセージ通信シーケンス [971](#)

逐次トランザクション  
    MSC ネットワーク内 [718](#)

チャンネル間 (CTC)  
    MSC (複数システム結合機能) 物理リンク・タイプ [683](#)  
    MSC 物理リンク  
        定義 [713](#)

中間 IMS  
    MSC  
        セキュリティ検査 [722](#)

中間システム (MSC) [692](#)

中断コード X'0811'  
    取られる処置 [984](#)

チューニング  
    バッファ [729](#)  
    MSC (複数システム結合機能) 環境 [701](#)  
    MSC のチューニングとモニター [729](#)

直接経路指定

直接経路指定 (続き)

    リンクを通して渡されないパスワード [722](#)

直接経路指定 (directed routing)  
    MSC [699](#)

直接制御サブシステム [639](#)

通信  
    イムス  
        前提知識 [xxiii](#)

通信、設定 [382](#)

通信コントローラー [377](#), [378](#)

通信ネットワーク  
    金融機関通信システム [932](#)  
    コンポーネント  
        責任 [376](#), [390](#)  
    SLU P システム [932](#)

通知メッセージ [122](#), [139](#)

定義  
    IMS [491](#)  
    ISC ノード [491](#)

停止  
    ワークステーション [513](#)

停止状態 [143](#), [406](#), [433](#)

停止状態とは [143](#)

停止ブラケット開始 (SBI) コマンド  
    IMS-CICS セッション [638](#)

停止ブラケット開始コマンド [554](#)

データ記述子 FM ヘッダー  
    出力形式 [579](#)  
    出力で使用 [566](#)  
    入力形式 [579](#)  
    入力で使用 [565](#)  
    IMS の使用 [651](#)

データ・ストリーム・プロファイル、ATTACH FM ヘッダー  
[574](#)

データ・セット満杯条件 [450](#)

データ・タイプ、IMS 拡張機能ネットワーク [966](#)

データ通信開始 (SDT) コマンド [512](#)

データ伝送  
    エラー処理 [982](#)  
    エラー・リカバリー・プロシージャ [652](#)

応答要件  
    要約 [978](#)  
    リカバリー可能な照会トランザクション [524](#), [980](#)  
    リカバリー不能照会トランザクション [524](#), [981](#)  
    IMS コマンドと標識 [466](#), [935](#)  
    IMS メッセージ通信 [524](#), [981](#)  
    LU 6.2 アプリケーション・プログラム [432](#)  
    MFS 制御要求 [982](#)  
    VTAM コマンドと標識 [466](#), [935](#)

確定応答、ISC [465](#)

メッセージ  
    出力 [971](#)  
    入力 [966](#)

例外応答、プロトコル  
    金融機関通信システム [935](#)  
    CICS-IMS [643](#), [652](#)  
    ISC [465](#)  
    SLU P、メッセージ・データ・タイプ [966](#)

データ透過性 (data transparency) [469](#)

データの区分化、MSC (複数システム結合機能) [683](#)

データ・フロー  
    端末からプログラムへ [398](#)  
    プログラムから出力端末へ [399](#)  
    プログラム間 [399](#)

データ・フロー (続き)  
メッセージ通信 [401](#)  
MSC (複数システム結合機能) ネットワーク内の [688](#)  
データ・フロー制御 (DFC) プロトコル  
エラー処理 [519](#)  
応答要件 [520](#)  
コマンド  
    [BID 528](#)  
    センス・コード [552](#)  
    対称セッション・シャットダウン [554](#)  
    チェーン [537](#)  
    同期点および応答  
        出力 [526](#)  
        入力 [524](#)  
        CICS-IMS [644](#)  
    入力メッセージ、バックアウト [521](#)  
    ハーフセッションの同期化 [520](#)  
    ブラケットおよび半二重 [528](#)  
    ページ化メッセージ・エラー [544](#)  
    メッセージのブラケット化  
        入力 [529](#)  
    例 [659](#)  
    例外応答 [519](#)  
    例外応答後の PURGE [538](#)  
    ERP PURGE [538](#)  
    LWA によって補助される回復性 [520](#)  
データ・フロー・リセット、エラーの処理 [549](#)  
データベース区分 [703](#)  
データベースの分割、MSC (複数システム結合機能) [703](#)  
データベース・リカバリー・アダプター (DRA)  
    セットアップ [767](#)  
データベース・リカバリー・アダプター (DRA) 始動テーブル  
    ODBA (Open Database Access) での使用 [767](#)  
    Open Database Access (ODBA) での使用 [767](#)  
データベース・リソース・アダプター (DRA)  
    接続  
        構成 [761](#)  
データベースを重複 [702](#)  
出口インターフェース・ブロック  
    IMS Connect  
        XIBDS [169](#)  
出口インターフェース・ブロック (XIB) [168](#)  
出口インターフェース・ブロック (XIB1) [172](#)  
出口インターフェース・ブロック ODBM データ・ストア項目  
[173](#)  
出口ルーチン  
    経路指定 [750](#)  
    類似性経路指定 [752](#)  
    ETO の [92](#)  
    IMS Connect  
        概要、機能に固有の出口ルーチン [166](#)  
        概要、ユーザー・メッセージ出口ルーチン [164](#)  
        概要 [163](#)  
    IMSplex メッセージ経路指定 [750](#)  
    IMSplex 類似性経路指定 [752](#)  
    MSC メッセージ経路指定 [750](#)  
    MSC 用 [718](#)  
    TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン [750](#)  
出口ルーチン、OTMA での [804](#)  
テスト・モード [406](#)  
テスト・モード (test mode) [470](#)  
デッド・レター・キュー HWS\$DLQ [352](#)  
転送機能、IMS Connect [303](#)

透過オプション [446](#)  
同期、ISC ハーフセッション [520](#)  
同期コールアウトのための COR トークン [256](#)  
同期コールアウト要求  
    MULTIRTP [887](#)  
    OTMA 構成の概要 [877](#)  
    OTMA サポート [878](#)  
    OTMA サポートの概要 [877](#)  
    RESUME TPIPE 要求の並列処理 [887](#)  
同期処理  
    定義 [471](#)  
    ATTACH FM ヘッダー、ISC [557](#)  
    CICS [621](#)  
同期点  
    定義 [398](#)  
    入力メッセージ、ISC [521](#)  
ISC  
    出力 [526](#)  
    入力 [524](#)  
    要件 [522](#)  
    CICS [644](#)  
同期プログラム間通信 [879](#)  
同期レベル  
    CONFIRM [294](#)  
    NONE [293](#)  
    SYNCH [294](#)  
統計  
    帯域幅モード、容量 [725](#)  
    論理リンク、MSC [723](#)  
    論理リンク、高位値の統計 [728](#)  
    論理リンク、統計のリセット [724](#)  
    論理リンク、バッファ・サイズ [724](#)  
    論理リンク、ベンチマーク [723](#)  
    論理リンク、リンク容量 [725](#)  
    MSC リンク  
        帯域幅モードのバッファ・サイズとフォーマット  
        [725](#)  
        バッファ、最適なサイズの判別 [725](#)  
        非帯域幅モードのバッファ・サイズとフォーマット  
        [727](#)  
    MSC リンク・タイプ、最適 [724](#)  
    MSC 論理リンク [723](#)  
    MSC 論理リンク、高位値の統計 [728](#)  
    MSC 論理リンク、統計のリセット [724](#)  
    MSC 論理リンク、バッファ・サイズ [724](#)  
    MSC 論理リンク、ベンチマーク [723](#)  
    MSC 論理リンク、リンク容量 [725](#)  
統計分析ユーティリティと MSC [729](#)  
動的端末 [379](#)  
動的割り振り VTAM サブプール [473](#)  
特記事項  
    商標 [987](#), [989](#)  
    特記事項 [987](#)  
トラステッド・ユーザー  
    IMS Connect [206](#)  
トランザクション  
    異常終了 [656](#)  
    回復可能 [864](#)  
    会話および OTMA [814](#)  
    会話型  
        OTMA での終了 [822](#)  
    高速機能および OTMA [814](#)  
    コミット後送信 [850](#), [852](#)  
    終了

トランザクション (続き)  
終了 (続き)  
OTMA 会話型トランザクション [822](#)  
送信後コミット [850](#), [855](#)  
標準のフロー [852](#)  
保護 [862](#)  
有効期限  
IMS Connect サポート [329](#)  
OTMA サポートの概要 [827](#)  
OTMA、STCK 形式での指定 [828](#)  
OTMA、秒単位での指定 [829](#)  
リカバリー不能 [864](#)  
IMS 間 TCP/IP 通信  
トランザクション・コード、指定 [803](#)  
IMS、同期化 T パイプを使用する [865](#)  
IMS、非同期化 T パイプを使用する [865](#)  
OTMA  
会話型トランザクションの終了 [822](#)  
トランザクション有効期限の概要 [827](#)  
秒単位での有効期限の指定 [829](#)  
STCK 形式での有効期限の指定 [828](#)  
OTMA グループ化 [777](#)  
トランザクション (transaction)  
コード、固有な [397](#)  
コードの定義 [391](#)  
状態 [432](#)  
複数システム [683](#)  
MSC 統計 [730](#)  
トランザクション・コード (リモート宛先) [696](#)  
トランザクション・ソケット 接続 [310](#)  
トランザクション・タイプ  
更新 [980](#)  
コマンド [982](#)  
照会  
定義 [978](#)  
リカバリー可能またはリカバリー不能 [980](#)  
定義 [432](#), [980](#)  
テスト・モード、ISC における [470](#)  
メッセージ通信  
IMS [981](#)  
ISC [573](#)  
ISC 例 [481](#), [485](#)  
ISC セッション、中の [470](#)  
ISC でサポートされる、リスト [458](#)  
トランザクション同期点の関係 [521](#), [644](#)  
トランザクション・パイプ  
アイドル、自動除去 [822](#)  
アイドル状態の T パイプの除去 [822](#)  
アイドル状態のトランザクション・パイプの除去 [783](#)  
およびメッセージ・フロー [784](#)  
キューおよびメッセージ・フローの使用 [784](#)  
クライアントが作成できる数 [779](#)  
使用 [783](#)  
全二重環境下でのフロー [785](#)  
定義 [777](#), [783](#)  
同期化 [783](#)  
非同期化 [783](#)  
命名規則 [848](#)  
リソースのインパクト [820](#)  
LTERM との違い [784](#)  
OTMA クライアント/サーバー環境下で [784](#)  
UNIX パイプとの違い [784](#)  
トランザクション分析ユーティリティ [729](#)  
トランザクション・マネージャー

トランザクション・マネージャー (続き)  
概要 [375](#)  
トランザクション・マネージャー・サービス [379](#)

## [ナ行]

内部実行モード、ISC [472](#)  
名前の一意性、リソース  
制約の使用不可化 [389](#)  
入出力装置の分離 [402](#)  
入出力編集出口ルーチン (DFSYIOE0) [804](#)  
入力エラー、IMS の ISC 検出 [548](#)  
入力コンポーネント  
出力コンポーネントとの関係 [472](#)  
対出力コンポーネント [941](#)  
ヘッダー・タイプ 'X'42' の識別 [970](#)  
入力システム (MSC) [692](#)  
入力ブラケット付け  
応答 [970](#)  
入力編集オプション  
ISC [469](#)  
SLU P [950](#)  
入力メッセージ  
チェーン・メッセージ [971](#)  
同期点  
ISC [521](#)  
入力メッセージ (input message)  
応答要件 [978](#)  
金融機関通信システム・チェーン標識 [967](#)  
高速機能 [954](#)  
チェーン・メッセージ [537](#)  
同期点  
可用性 [521](#)  
ブラケット付け [967](#), [970](#)  
ヘッダー  
オプション MFS (メッセージ形式サービス) フィールド [968](#)  
金融機関形式 [967](#)  
使用 [967](#)  
DPM [968](#)  
MID 名の位置 [967](#), [969](#)  
SCS [969](#)  
SLU P 形式 [968](#)  
メッセージ記述子バイト  
ヘッダー・タイプ 'X'42' [969](#)  
MID 名標識ビット [968](#), [969](#)  
要件 [967](#)  
1 つのメッセージを複数回に分けて伝送 [971](#)  
ISC のバックアウト [521](#)  
ISC のブラケット化 [529](#)  
MFS (メッセージ形式サービス) 入力フォーマット設定、活動化 [971](#)  
MFS (メッセージ形式サービス) 目的の場合の発信元 [971](#)  
MFS 入力フォーマット設定、活動化 [561](#)  
SLU P  
制約事項 [970](#)  
チェーン標識 [967](#)  
入力メッセージ、応答要件 [524](#)  
入力メッセージ構造  
クライアント [241](#)  
ヌル出力メッセージ (null output message)  
目的 [971](#)  
ネットワーク [701](#)  
ネットワーク

## ネットワーク (続き)

- オプションのコンポーネント [375](#)
- 開始
  - 金融機関通信システム [431, 957](#)
- 管理
  - アクティビティ [393](#)
- 計画 [393](#)
- シャットダウン [500, 964](#)
- 終了
  - 金融機関通信システム [963, 965](#)
- 設計の考慮事項 [393](#)
- 操作
  - 金融機関通信システム [931, 957](#)
  - 通信の確立 [382](#)
  - IMS-CICS [636, 638](#)
  - ISC [499, 500](#)
- 操作の準備 [425](#)
- 通信 [375](#)
- 定義 [425, 426, 719](#)
- 非交換通信ネットワーク [403](#)
- 複数システムの影響 [719](#)
- 要件の文書化 [394](#)
- 論理端末ネットワーク設計 [401](#)
- APPC/IMS による操作 [382](#)
- VTAM と NCP パラメーター [425](#)

ネットワーク、通信 [375](#)

ネットワーク修飾 LU 名 [53](#)

ネットワーク・セキュリティ資格情報の伝搬  
IMS Connect [186](#)

ネットワーク・セキュリティ・セグメント  
形式 [258](#)

ネットワーク体系モデル [773](#)

ネットワークの役割 [376, 378](#)

ノード定義、ISC [491, 497](#)

ノード名 [395](#)

ノード・ユーザー記述子 [89](#)

## [ハ行]

ページ機能、IMS Connect [301](#)

パートナー・システム、MSC (複数システム 結合機能) リンク  
内の [685](#)

ハーフセッション

- 定義 [499, 503](#)
- 同期化、ISC [520](#)
- ペア [496, 503](#)

排他モード [406](#)

バインド

- 拒否された、ISC [549](#)
- 交渉可能 [502](#)
- 非同期処理の要求 [558](#)
- IMS-CICS セッション [654](#)
- ISC 照合順序の先頭に名前セッション
  - 単一 [503](#)
  - 並列 [503](#)

パスチケット

- リプレイ 保護 [205](#)
- IMS Connect [201](#)

パスワード

- IMS Connect 大/小文字混合パスワード [200](#)
- IMS Connect、RACF パスワードの変更 [198](#)
- IMS Connect、RACF パスワード・フレーズの変更 [199](#)

バックアウト作業単位、メッセージ再同期 [504](#)

バックエンド・サブシステム

## バックエンド・サブシステム (続き)

経路指定トランザクション [490](#)

定義 [457](#)

CICS [624](#)

IMS [621, 623](#)

バッファ

小バッファ装置 [449](#)

BUFSIZE パラメーター [712](#)

MSC (複数システム 結合機能) の考慮事項 [701](#)

MSC (複数システム 結合機能) リンク [683](#)

MSC VTAM リンク [712](#)

MSC 非 VTAM リンク [712](#)

MSC リンク

- 帯域幅モードのバッファ・サイズとフォーマット [725](#)
- バッファ・サイズの判別 [725](#)
- 非帯域幅モードのバッファ・サイズとフォーマット [727](#)
- リンク・バッファのフォーマット [725](#)

MSC リンクおよび帯域幅 [702](#)

MSC リンクのサイズ [712](#)

バッファ・サイズ

- BUFSIZE パラメーター [712](#)
- IMS 間セッション [490](#)
- ISC [495](#)
- MSC 物理リンク [712](#)

パフォーマンス [814](#)

パフォーマンスのモニター [814](#)

パラメーター

- OTMA 構成パラメーター、要約 [789](#)
- OTMAMD [789](#)
- OTMANM [789](#)
- OTMASE [789](#)
- OTMASP [789](#)

バランシング・グループ

- 宛先コードおよび [390](#)
- 定義 [390](#)

半二重プロトコル、IMS 使用 [528](#)

半二重メッセージ・フロー [780](#)

汎用リソース

- MSC TCP/IP [745](#)
- TCP/IP
  - 類似性、IMS Connect 内のクリア [748](#)
  - 類似性、MSC のクリア [747](#)
  - IMS Connect サポート [157](#)
  - MSC および XRF [749](#)
  - MSC の類似性管理 [746](#)
  - MSC の類似性の持続性 [747](#)
  - MSC 類似性、IMS Connect 内のクリア [748](#)
  - MSC 類似性、クリア [747](#)
  - MSC 類似性管理 [746](#)
  - MSC 類似性の持続性 [747](#)
  - XRF および MSC [749](#)

汎用リソース・グループ

- 概要 [387](#)
- 定義 [388](#)
- 利点 [387](#)

汎用リソース名 (generic resource name)

- 定義 [388](#)

汎用リソース・メンバーの定義 [388](#)

非会話型 プログラム [845](#)

非グラフィック・メッセージ・データ [447](#)

非交換通信ネットワーク [403](#)

非持続ソケット接続接続 [310](#)

非同期コールアウト要求  
OTMA 構成の概要 [877](#)  
OTMA サポートの概要 [877](#)  
非同期出力  
インプリメント [334](#)  
管理 [334](#)  
出力メッセージの管理と制御 [339](#)  
代替クライアント ID [345](#)  
別のクライアントの出力のリトリーブ [345](#)  
メッセージ・フロー [347](#)  
ETO  
有効な宛先 [107](#)  
OTMA、出力の管理 [872](#)  
OTMA、出力のページ [873](#)  
RESUME TPIPE、代替クライアント ID [345](#)  
非同期出力処理  
コミット・モード [344](#)  
ソケット・タイプ [344](#)  
タイマー設定 [344](#)  
同期レベル [344](#)  
非同期処理  
ATTACH EB [472](#)  
IMS-CICS セッション [626](#)  
ISC 実行モード [471](#)  
SCHEDULER FM ヘッダー [558](#)  
非同期メッセージ  
OTMA、出力の転送 [873](#)  
OTMA、スーパーメンバー [875](#)  
非ブロック化アルゴリズム、ATTACH FM ヘッダー [574](#)  
表示画面保護  
金融機関通信システム 端末 [948](#)  
定義 [948](#), [949](#)  
BID オプション [948](#), [949](#)  
MFS [948](#), [949](#)  
NOBID オプション [948](#), [949](#)  
ファースト・スピーカー  
2 次論理装置 [468](#)  
ISC 競合勝者 [468](#)  
プール・マネージャ  
MFS の説明 [443](#)  
MFSTEST [443](#)  
フォーマット設定、メッセージ  
透過オプション [446](#)  
メッセージ接頭語の制御文字 [448](#)  
SDF II による作成 [442](#)  
複数 IMS システムの接続 [701](#)  
複数システム [701](#)  
複数システム結合機能 (MSC)  
帯域幅モード、容量 [725](#)  
ネットワークおよび経路指定の概要 [688](#)  
汎用リソース、VTAM [750](#)  
論理リンク  
統計およびリンク容量 [725](#)  
ベンチマーク [723](#)  
ベンチマークの統計 [723](#)  
容量および統計 [725](#)  
論理リンク統計およびリンク容量 [725](#)  
論理リンク・パフォーマンスのベンチマーク [723](#)  
VTAM 汎用リソース [750](#)  
複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC))  
最適なリンク・タイプ [724](#)  
システム定義 [705](#)  
システム定義検査 [721](#)

複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC)) (続き)  
使用可能化  
システム定義時 [709](#)  
使用不可化 [710](#)  
セキュリティの考慮事項 [722](#)  
帯域幅、リンク [702](#)  
統計  
最適なリンク・バッファ・サイズの評価 [725](#)  
統計、論理リンクの [723](#)  
トランザクション・コードの定義 [705](#)  
バッファ  
サイズ、リンク統計を使用した最適化 [725](#)  
帯域幅モードのサイズ [725](#)  
帯域幅モードのフォーマット [725](#)  
非帯域幅モードのサイズ [727](#)  
非帯域幅モードのフォーマット [727](#)  
バッファ・サイズ [712](#)  
バッファ・サイズ、リンク [702](#)  
パフォーマンス、論理リンク [723](#)  
汎用リソース  
類似性、IMS Connect 内のクリア [748](#)  
類似性、TCP/IP リンクのクリア [747](#)  
類似性管理、TCP/IP リンク [746](#)  
類似性の持続性、TCP/IP リンク [747](#)  
TCP/IP [745](#)  
TCP/IP および XRF [749](#)  
XRF および TCP/IP [749](#)  
物理リンク  
タイプ [711](#)  
定義 [711](#), [712](#)  
CTC、定義 [713](#)  
MTM、定義 [713](#)  
TCP/IP、定義 [714](#)  
VTAM、定義 [715](#)  
リンク帯域幅 [702](#)  
リンク・タイプ、最適 [724](#)  
リンク統計  
最適なリンク・バッファ・サイズの評価 [725](#)  
リンクのチューニング [723](#)  
リンク・バッファ・サイズ [702](#)  
類似性、クリア  
IMS Connect [748](#)  
TCP/IP 汎用リソース [747](#)  
類似性管理  
TCP/IP 汎用リソース [746](#)  
類似性の持続性  
TCP/IP 汎用リソース [747](#)  
論理リンク  
高位値の統計 [728](#)  
統計、高位値 [728](#)  
統計、リセット [724](#)  
統計およびバッファ・サイズ [724](#)  
統計のリセット [724](#)  
バッファ・サイズおよび統計 [724](#)  
論理リンク統計 [723](#)  
論理リンク統計、高位値 [728](#)  
論理リンク統計およびバッファ・サイズ [724](#)  
論理リンク統計のリセット [724](#)  
DRD に対する使用可能化 [706](#)  
IMS Connect  
汎用リソース [157](#)  
IMS 間 TCP/IP 通信  
IMS Connect の MSC のサポート [156](#)  
IMSplex



複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC)) (続き)

- ISC、判別方法 [474](#)
- SLU P
  - 方向標識 [938](#)
- ブラケット・プロトコル
  - 出力ブラケット付け [977](#)
  - 出力メッセージ、ISC [532](#)
  - 入力ブラケット付け [970](#)
  - 入力メッセージ、ISC [529](#)
  - IMS [528](#), [966](#)
- フラッディング制御
  - ACEE [818](#)
  - OTMA [818](#)
- プリンター
  - 共用 [451](#)
  - 候補 [451](#)
- プリンターの共用
  - システム間 [451](#)
- フロー
  - 確認付き送信後コミット [856](#)
  - コミット後送信 [852](#), [860](#)
  - 再同期の [867](#)
  - 送信後コミット [855](#)
  - データの
    - MSC (複数システム結合機能) ネットワーク内の [688](#)
- プログラム間 (P2P) [842](#)
- プログラム間通信
  - 宛先名 [696](#)
  - 会話型トランザクション [700](#)
  - 使用のシナリオ
    - 高速 PCB を使用する [843](#)
    - 単一ストリーム [842](#)
    - 複数プログラムへの [843](#)
    - 保護トランザクションの場合の [845](#)
    - I/O PCB への ISRT なし [843](#)
    - OTMAASY オプションを使用する [843](#)
- 同期 [879](#)
- 非会話型トランザクション [700](#)
- リモート宛先検査 [704](#)
- レース条件 [843](#)
- CM0 メッセージ [842](#)
- CM1 メッセージ [842](#)
- プログラム仕様ブロック (PSB) (program specification block (PSB))
  - CICS DRA
    - コーディングのガイドライン [763](#)
- プログラム連絡ブロック (PCB) [747](#)
- プロセッサの使用率、MSC [702](#)
- プロトコル
  - 出力 [477](#)
  - IMS Connect 会話型
    - 送信後コミット、同期レベル = CONFIRM (ACK 応答) [299](#)
    - 送信後コミット、同期レベル = CONFIRM (NAK 応答) [300](#)
    - 送信後コミット、同期レベル = NONE、クライアントが終了したトランザクション [298](#)
    - 送信後コミット、同期レベル = NONE、プログラムが終了したトランザクション [297](#)
  - IMS Connect 送信専用 [306](#)
  - IMS 間セッションの制約事項 [491](#)
  - OTMA
    - サーバー状態プロトコル・コマンド [815](#)
- プロトコル・レベル [294](#)

複数システム結合機能 (MSC) (Multiple Systems Coupling (MSC)) (続き)

- IMSplex (続き)
  - リモート APPC および OTMA トランザクションのバックエンド処理の有効化 [743](#)
  - リモート APPC または OTMA トランザクションのバックエンド処理 [738](#)
  - APPC トランザクション、IMSplex 外部の MSC システムに送信 [744](#)
  - APPC および OTMA リモート・トランザクション [738](#)
  - OTMA トランザクション、IMSplex 外部の MSC システムに送信 [744](#)
- IMSRSC リポジトリに対する使用可能化 [707](#)
- SPA [704](#)
- SYSID の定義 [709](#)
- TCP/IP
  - 汎用リソース [745](#)
- TCP/IP 汎用リソース
  - 類似性、IMS Connect 内のクリア [748](#)
  - 類似性、クリア [747](#)
  - 類似性管理 [746](#)
  - 類似性の持続性 [747](#)
  - XRF [749](#)
  - /MSVERIFY コマンド [721](#)
- 複数システム検査ユーティリティ (DFSUMSV0) [699](#)
- 複数システム検査ユーティリティ、使用 [720](#)
- 複数システム・トランザクション統計の抽出 [730](#)
- 複数チェーン入力メッセージの交換制限 [573](#)
- 複数の外部サブシステム [141](#)
- 複数のサインオン
  - シスプレックス環境で [96](#)
  - 説明 [96](#)
  - 命名規則 [96](#)
- 物理端末
  - 装置クラス制御 [448](#)
  - 定義 [73](#)
  - 入出力装置の分離 [402](#)
- 物理リンク
  - タイプ [711](#)
  - 定義
    - バッファ・サイズ [712](#)
    - CTC [713](#)
    - MTM [713](#)
    - TCP/IP [714](#)
    - VTAM [715](#)
  - MSC (複数システム結合機能) [683](#)
  - フラグ読み取りフィールド、出力メッセージ [971](#)
  - ブラケット開始の停止 (BIS) コマンド
    - セッション・シャットダウン [554](#)
  - IMS-CICS セッション [638](#)
- ブラケット開始標識
  - 使用 (図) [938](#)
  - LUSTATUS コマンド [542](#)
- ブラケット拒否
  - 金融機関通信システム [958](#), [984](#)
  - ISC [550](#)
  - SLU P [958](#), [984](#)
- ブラケット・コンテンション
  - 解決 [503](#), [951](#)
  - 無効なページング [951](#)
- ブラケットと送信 / 受信の管理
  - 金融機関通信システム
    - プロトコル [938](#)
    - 方向標識 [938](#)

フロントエンド切り替え出口ルーチン (DFSFEBJ0)  
   特殊サポート [457](#)  
 フロントエンド・サブシステムの機能 [457](#)  
   CICS  
     SEND LAST コマンド [623](#)  
     SEND/RECEIVE コマンド [621](#)  
     START/RETRIEVE コマンド [626](#)  
   IMS [624](#), [628](#)  
 フロントエンド・スイッチ  
   OTMA、サポートされていない [814](#)  
 分散 2 フェーズ・コミット [356](#)  
 分散セキュリティー資格情報の伝搬  
   IMS Connect [186](#)  
 分散トランザクション処理、ISC [457](#), [479](#)  
 分散表示管理 (DPM)  
   オプション [447](#)  
   出力 [973](#)  
   FM ヘッダー [562](#)  
   OPTIONS=DPAGE または PPAGE [562](#), [973](#)  
   SLU P サポート [946](#)  
 分散プラットフォーム用 WebSphere Application Server  
   IMS Universal ドライバーの構成 [3](#)  
 分散リレーショナル・データベース体系 (DRDA)  
   IMS Connect サポート [151](#)  
 文書  
   端末プロファイル [425](#)  
   出口ルーチン [718](#)  
 並列セッション  
   許可済みユーザー [473](#)  
   バインド要件 [503](#)  
   ISC、IMS 間セッションの定義 [489](#)  
 ページ先送り機能 (NEXTTPP)、MFS [948](#)  
 ページ削除機能、MFS [562](#)  
 ページ保護状態 [407](#)  
 ページング・エラー  
   オンライン検出、MFS [563](#)  
   ISC [544](#)  
 ヘッダー  
   出力メッセージ (output message) [447](#)  
   タイプ X'42'  
     コンポーネント識別 [970](#)  
     データ・バイト [976](#)  
     メッセージ記述子バイト [969](#), [975](#)  
   入力メッセージ (input message) [447](#)  
   ISC [447](#)  
 変換  
   COBOL、XML の例 [222](#)  
   XML から COBOL への例 [222](#)  
 編集  
   オプション  
     データ通信出口ルーチン [469](#)  
     入出力 [469](#)  
     ISC 入出力、リスト [458](#)  
   ISC  
     デフォルト・エディター [469](#)  
   ISC 中の基本編集 [469](#)  
   MFS  
     CICS [607](#)  
     ISC 中 [469](#)  
 編集オプション  
   タイプ  
     DPM-An [950](#), [973](#)  
     MFS-SCS1 [950](#)  
   編集オプション (続き)  
     タイプ (続き)  
       SCS1 [950](#)  
       SLU P、TERMINAL マクロの COMPTn パラメーター [950](#)  
       TERMINAL マクロの COMPTn パラメーター [495](#)  
       TERMINAL マクロの OUTBUF パラメーター [495](#), [950](#)  
   編集機能  
     FM ヘッダーの呼び出し [556](#)  
     ISC 概説 [469](#)  
   返信 resynch [866](#)  
   方向変換 (CD) 標識  
     金融機関通信システム [937](#)  
     送信 / 受信のプロトコル [938](#), [966](#)  
     SLU P [937](#)  
   報告書  
     IMS モニター [730](#)  
     MSC [730](#)  
   保管域接頭部 (SAP)  
     OTMA 入力メッセージ [823](#)  
   保護  
     会話 [842](#)  
     拡張出力コンポーネント [949](#)  
     トランザクション [862](#)  
     表示画面 [948](#)  
   保護リソース、定義 [28](#)  
   保守上の変更点 [441](#)  
   保全性  
     メッセージ (message)  
       IMS-CICS セッション [654](#)  
       ISC での [459](#)  
       NOCHECK PROTECT [626](#)  
   保留キュー、非同期  
     代替クライアント ID の非同期出力のリトリブ [345](#)  
  
 [マ行]  
  
 マイグレーション  
   MSC から IMSplex [734](#)  
 マクロ  
   IMS Connect [167](#)  
   XCF、および OTMA [814](#)  
 マクロ・キーワード  
   BUFPOOLS マクロの COMM [427](#)  
   COMM マクロの APPLID [426](#)  
   COMM マクロの PASSWD [426](#)  
   COMM マクロの RECANY [427](#)  
   ETO ログオン記述子での MODETBL [428](#)  
   TERMINAL マクロでの MODETBL [428](#)  
 マスター MSC とスレーブ MSC (複数システム結合機能) の関係 [683](#)  
 マスター端末  
   MSC 経路指定 [752](#)  
   XRF 複合システム (XRF complex) [404](#)  
 マスター端末 (master terminal)  
   装置選択 [404](#)  
   ログオン要件 [428](#)  
   LTERM の予約 [395](#)  
 マスター端末オペレーター (MTO) [428](#)  
 マルチノード持続セッション (MNPS)  
   XRF [417](#)  
   ミラー・トランザクション、CICS [626](#)  
   無条件ブラケット終了、IMS エラー処理 [982](#), [984](#)  
   無保護画面オプション [446](#)  
   命名規則 [440](#)



## メッセージ

### 経路指定

- IMS 間 TCP/IP 通信フロー [780](#)
- MSC-IMSplex 構成内 [732](#)
- OTMA 宛先記述子 [797](#)
- OTMA 宛先記述子、宛先名のマスキング [799](#)
- OTMA 宛先記述子、定義 [799](#)
- TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン [750](#)

### 順序どおり [856](#)

### 設計済みフォーム [850](#)

### 端末/入力経路指定 [750](#)

### フロー

- トランザクション・パイプの使用 [784](#)

- XCF を使用して [784](#)

### 保護会話、OTMA 制約事項 [814](#)

### 類似性経路指定 [752](#)

### IMSplex [750](#)

### IMSplex 類似性経路指定 [752](#)

### MSC [750](#)

### OTMA

- 送信 [848](#)

### TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン [750](#)

## メッセージ (message)

### 宛先決定 [805](#)

### 拡張 [776](#)

### 経過中

- 装置 LU [430](#)

- プログラム LU [430](#)

### 交換

- 共用 キュー環境での [845](#)

### 再同期

- サンプル [871](#)

### サンプル

- 再同期の成功 [872](#)

- 再同期を伴うクライアント・ビッド要求 [871](#)

- CBresynch の受信の確認 [871](#)

- REPresynch コマンド [872](#)

- REQresynch コマンド [872](#)

- SRVresynch コマンド [871](#)

- SRVresynch の受信の確認 [871](#)

### サンプル OTMA メッセージ [919](#)

### シーケンス番号 [849](#)

### 出力セグメントの編集 [447](#)

### スケジューリング

- 高速機能 (Fast Path) [390](#)

- 定義 [397](#)

### 制御文字 [448](#)

### 接頭部

- 構文 [889](#)

- 状態データ・セクション [899](#)

- 内容 [860](#)

- OTMA アプリケーション・データ・セクション [918](#)

- OTMA セキュリティー・データ [914](#)

- OTMA セキュリティー・データ・セクション [914](#)

- OTMA ユーザー・データ・セクション [917](#)

### 選択方法の例 [809](#)

### 選択リカバリ [809](#)

### 全二重フロー [780](#)

### 半二重フロー [780](#)

### 非グラフィック・メッセージ・データに対するセンシティブィティ [447](#)

### フォーマット設定と編集 [442](#)

## メッセージ (message) (続き)

### フロー

- 確認付き送信後コミット [856](#)

- コミット後送信 [852](#), [860](#)

- 再同期 [869](#)

- 全二重環境下 [785](#)

- 送信後コミット [855](#)

- 定義 [777](#)

- 割り振り解除 [850](#)

- T パイプ内のキューの使用 [784](#)

### 変換 [369](#)

- リキュアー [808](#)

- IMS の処理 [383](#)

- IMS の編集 [444](#)

- MFS 編集のバイパス [446](#)

- OTMA 環境でのフロー [778](#)

- Z2 フィールド [448](#)

### メッセージ記述子バイト

- 出力メッセージ (output message) [973](#), [975](#)

- 入力 FM ヘッダー [967](#), [969](#)

### メッセージ・キューの定義 [391](#)

### メッセージ形式サービス

- OTMA、サポートされていない [814](#)

### メッセージ形式サービス (MFS)

- アプリケーション・プログラム、ISC でのアクセス [478](#)

- エスケープ文字、ISC でサポートされない [561](#)

- オンライン・エラー検出 [563](#)

- オンライン・パフォーマンス [441](#)

### 概要 [435](#)

### 管理 [440](#)

- 言語ユーティリティー [436](#), [443](#)

- コンポーネント [436](#)

- コンポーネント、概説 [442](#)

- サービス・ユーティリティー [443](#)

### 削除文字 [447](#)

### 出力フォーマット設定

- 典型的なアプリケーション・プログラム・プロシージャ [977](#)

- MOD 名 [977](#)

- 出力フォーマット設定、ISC [561](#)

### 出力メッセージ

- 編集セグメント [447](#)

- MFS 定義の方法 [435](#)

### 使用可能な機能 [946](#)

### 制御機能

- 金融機関通信システム [947](#)

- NEXTLP [948](#)

- NEXTPP [948](#)

- PAGE=nn [948](#)

### 制御要求、応答要件 [982](#)

### 説明 [380](#)

- 端末キーボードのロックとアンロック [447](#)

### 定義 [946](#)

- データ・バイト出力メッセージ [976](#)

- デフォルト形式 [589](#)

- 同期点、ISC メッセージ [522](#)

- 入力セグメント [447](#)

- 入力フォーマット設定 [561](#)

- 入力メッセージ [435](#)

- プール・マネージャ [443](#)

### フォーマット

- 入力の活動化 [958](#)

- ページ削除機能、ISC でサポートされない [562](#)

- ページング、CICS [604](#), [641](#)

メッセージ形式サービス (MFS) (続き)

編集 [446](#)

無効なページング要求 [949, 951](#)

メッセージ・エディター [443](#)

メッセージ・リカバリ [947](#)

ライブラリー、オンライン変更 [442](#)

利点 [441](#)

Bid オプション [948](#)

DPM [973](#)

DPM オプション [447](#)

FM ヘッダー

出力フォーマット設定活動化 [561](#)

使用可能な機能 [561](#)

タイプ [561](#)

入力フォーマット設定の活動化 [561](#)

編集 [561](#)

ISC アプリケーション・プログラムのバイパス [478](#)

ISC メッセージの編集 [469](#)

MFS のバイパス [446](#)

MID/MOD チェーン [946](#)

MSC (複数システム結合機能) [705](#)

SDF II [443](#)

SLU P

ステーション別、可用性 [947](#)

利点 [947](#)

メッセージ経路指定

例 [481](#)

ISC での

経路指定パラメーターの使用 [480](#)

MSC リンクを超えた [489](#)

MSC (複数システム結合機能) ネットワーク内の [689](#)

メッセージ構造

ユーザー作成の IMS Connect クライアント・アプリケーション・プログラム用 [243](#)

IMS TM Resource Adapter 用 [243](#)

メッセージ・コンテンツ [504, 951](#)

メッセージ再同期

設計の考慮事項 [505](#)

メッセージ再同期 (message resynchronization)

開始の方法と時期 [504, 952](#)

関連システム定義オプション [505](#)

高速機能 (Fast Path) [955](#)

最終インバウンド / アウトバウンド [952](#)

シーケンス番号 [505, 959](#)

実行 [654](#)

設計の考慮事項 [959](#)

説明 [505, 654](#)

送信 / 受信とブラケット・プロトコル [966](#)

通常データ伝送に対する影響 [952, 959](#)

ハーフセッション・ペアの極性 [505](#)

方向およびブラケット標識 [966](#)

メッセージ・シーケンス番号のオプション [959](#)

目的 [504](#)

要件 [504, 952](#)

CICS-IMS セッション [654](#)

STSN

形式 [516, 960](#)

の使用 [509](#)

フロー、1 次から 2 次 [514](#)

フロー、2 次から 1 次 [515](#)

メッセージ先送り機能 (NEXTMSG) [948](#)

メッセージ先送り保護機能 (NEXTMSGP) [948](#)

メッセージ出力記述子 (MOD) [977](#)

メッセージ出力記述子 (MOD) (message output descriptor (MOD)) [442](#)

メッセージ処理 [383](#)

メッセージ制御/エラー出口ルーチン (DFSCMUX0) [719](#)

メッセージ接頭語の Z2 フィールド [448](#)

メッセージ通信 (message switch)

例 [481](#)

ATTACH FM ヘッダー [571](#)

ISC [479, 480](#)

メッセージ通信、DFSAPPC [55](#)

メッセージ通信、ISC [469](#)

メッセージ通信、応答要件 [524](#)

メッセージ入力記述子 (MID) [442](#)

メッセージの経路指定

宛先名 [696](#)

パラメーター [480](#)

ISC 例 [481](#)

SYSID [696](#)

メッセージのデキュー、暗示 [506](#)

メッセージの編集

基本編集および非グラフィック・メッセージ [448](#)

基本編集または MFS 編集のバイパス [446](#)

出力セグメント [447](#)

透過オプション [446](#)

IMS の編集 [444](#)

メッセージの保全性 [459](#)

メッセージ・フォーマット

OTMA

IMS 間 TCP/IP 通信 [803](#)

メッセージ・ヘッダー [447](#)

メッセージ・リカバリ

金融機関通信システム [952](#)

制約事項 [953](#)

メッセージ再同期 (message resynchronization) [526, 952](#)

MFS [947](#)

SLU P システム [952](#)

メモリー間 (MTM)

MSC (複数システム結合機能) 物理リンク・タイプ [683](#)

MSC 物理リンク

定義 [713](#)

モード、端末

応答 (response) [405](#)

会話 (conversation) [405](#)

テスト・モード (test mode) [406](#)

排他的 [406](#)

ロック [406](#)

ETO と排他モード [406](#)

LU 6.2 [405](#)

SNA QUIESCE [407](#)

モジュール

DFSAERA0 [768](#)

DFSAERGO [768](#)

DFSAERM0 [768](#)

DFSCDLI0 [768](#)

戻り 1 次リソース名 [577](#)

戻り宛先プロセス名 [577](#)

モニターとチューニング、MSC [729](#)

[ヤ行]

ユーザー

操作モード [405](#)

ユーザー ID

ユーザー ID (続き)  
IMSplex、内  
リカバリー状況 [411](#)  
ユーザー ID キャッシング・スキーム [830](#)  
ユーザー異常終了 [119ABEND 856](#)  
ユーザー記述子 (user descriptor) [90](#)  
ユーザー許可処理 [122](#)  
ユーザー作成クライアント・アプリケーション  
IMS Connect メッセージ構造 [243](#)  
ユーザー・データ・フォーマット設定出口ルーチン  
(DFSYDRU0) [804](#)  
ユーザー・メッセージ出口  
サポート [223](#)  
説明と構造 [241](#)  
ユーザー・ワークステーション  
ブラケット・プロトコル [529](#)  
API (アプリケーション・プログラム・インターフェース)  
CICS 同期 [639](#)  
CICS 非同期 [639](#)  
ユニコード  
IMS Connect [369](#)  
要求 resynch [866](#)  
要求時ページ化メッセージ  
制御  
QMODEL FM ヘッダー [566](#)  
用紙 [450](#)  
呼び出し可能インターフェース (C/I)  
の紹介 [921](#)

**[ラ行]**

ライブラリー、オンライン変更 [442](#)  
リカバリー  
高速機能 (Fast Path) [410](#)  
リソース 状況 [407](#)  
リカバリー可能照会トランザクション、応答要件  
IMS [978, 980](#)  
ISC [522, 524](#)  
リカバリー可能トランザクションとリカバリー不能トランザクション [47](#)  
リカバリー可能入力、確認 [951](#)  
リカバリー可能メッセージ、ISC 同期点 [522](#)  
リカバリー可能メッセージ対リカバリー不能メッセージ [978, 980](#)  
リカバリー環境  
分散リソース、定義 [29](#)  
ローカル・リソース、定義 [29](#)  
リカバリー・コーディネーター [119](#)  
リカバリー単位  
未確定の定義 [29](#)  
リカバリー単位 (unit of recovery)  
経過中、定義 [29](#)  
定義 [29](#)  
リカバリー・トークン [124, 141](#)  
リカバリー不能出力 [852](#)  
リカバリー不能照会トランザクション  
応答要件 [432, 981](#)  
LU 6.2 アプリケーション・プログラム [432](#)  
リカバリー不能トランザクション [490, 864](#)  
リカバリー不能メッセージ、ISC 同期点 [524](#)  
リカバリー・プロシージャー [419](#)  
リスト構造  
オーバーフロー、定義 [385](#)

リスト構造 (list structure)  
基本、定義 [385](#)  
定義 [384](#)  
リセット付加プロセス FM ヘッダー [558](#)  
リソース  
状況種別 [408](#)  
状況リカバリー [407](#)  
状況リカバリー・モード [408](#)  
MSC (複数システム結合機能) の考慮事項 [701](#)  
TM  
共用 [389](#)  
リソース・アクセス管理機能 (RACF) [722](#)  
リソース構造 (resource structure)  
利点 [389](#)  
IMS TM リソース  
管理 [388](#)  
リソース・タイプ整合性  
制約の使用不可化 [390](#)  
リソースの調整 [124](#)  
リソース変換テーブル [120](#)  
リソース・マネージャー  
IMS [30](#)  
IMS TM リソース、RM の利点 [389](#)  
リソース・マネージャー、定義 [28](#)  
リソース名の一意性  
制約の使用不可化 [389](#)  
リソース・リカバリー・サービス  
保護会話メッセージ、OTMA 制約事項 [814](#)  
OTMA の制約事項および要件 [814](#)  
リモート LTERM [697](#)  
リモート宛先検査  
システム保全性 [700](#)  
MSC (複数システム結合機能) 会話 [704](#)  
リモート宛先名 [696](#)  
リモート・システム (remote system)  
MSC (複数システム結合機能)  
定義 [688](#)  
リモート端末オペレーター (RTO) [428](#)  
リンク  
IMSplex 内の MSC リンク定義の削除 [737](#)  
MSC  
最適なリンク・タイプ [724](#)  
MSC (複数システム結合機能)  
物理 [683](#)  
論理 [685](#)  
IMSplex 内の定義 [734](#)  
MSC 帯域幅モード、容量 [725](#)  
MSC リンクの帯域幅 [702](#)  
MSC 論理リンク統計 [723](#)  
MSC 論理リンク統計、リセット [724](#)  
MSC 論理リンク統計およびバッファー・サイズ [724](#)  
MSC 論理リンクの高位値の統計 [728](#)  
MSC 論理リンク・ベンチマーク [723](#)  
MSC 論理リンク容量 [725](#)  
MSC、帯域幅の制御 [702](#)  
リンク (link)  
優先順位、設定 [716](#)  
MSC (複数システム結合機能) [683](#)  
リンク・パス  
MSC (複数システム結合機能) [686](#)  
類似性  
定義 [388](#)  
MSC-IMSplex 構成内 [733](#)  
類似性経路指定 [752](#)

ルーティング・パス (routing path)  
MSC (複数システム結合機能) [689](#)  
例外応答  
プロトコル  
  応答要件 (図) [978](#)  
  VTAM 機能 [465](#)  
  DFC コマンド、除去 [518](#)  
  ISC 応答モード・エラー [519](#)  
レース条件  
  回避 [843](#)  
  定義 [843](#)  
ローカル・システム  
  MSC (複数システム結合機能)  
  定義 [688](#)  
ロード、ESAP の [135](#)  
ロギング  
  MSC [729](#)  
ログオフ [382](#)  
ログオフの定義 [382](#)  
ログオン記述子 (logon descriptor)  
  ETO  
  定義 [65](#)  
ログオンの定義 [382](#)  
ログオン・モード  
  セッションの開始 [937](#)  
  デフォルト・ログオン・モード・テーブル [467](#)  
ログ先行書き込み (LWA) [459](#)  
ログ・トランザクション分析ユーティリティー、MSC 統計  
[730](#)  
ログ分析報告書  
  MSC 記入項目用の ID 列 [731](#)  
  MSC トランザクション [731](#)  
ログ・マージ・ユーティリティー  
  ログ出力の制御 [730](#)  
  MSC ログ [730](#)  
ロック・モード [406](#)  
論理宛先  
  MSC (複数システム結合機能) [690](#)  
論理作業単位 [652](#)  
論理装置  
  定義 [375](#)  
  プログラマブル、定義 [377](#)  
論理装置 (LU) (logical unit (LU))  
  再割り当て [54](#)  
  複数、管理 [54](#)  
  LU 名の修飾 [53](#)  
論理装置状況コマンド [542](#)  
論理端末 (LTERM)  
  入出力装置の分離 [402](#)  
  命名規則  
  必要性の除去 [942](#)  
  リモート [697](#)  
  ETO [379](#)  
  LTERM サブプール、ユーザー、およびコンポーネント  
  [473](#)  
  MSC および [752](#)  
論理端末 (LTERM) (logical terminal (LTERM))  
  キュー [402](#)  
  コンポーネントの定義 [574](#)  
  チェーン [401](#)  
  定義 [395](#)  
  ネットワーク設計 [401](#)  
  物理端末との関係 [401](#)  
  マスター端末 (master terminal) [404](#)

論理端末 (LTERM) (logical terminal (LTERM)) (続き)  
  命名方法 [941](#)  
論理ネットワーク設計 [401](#)  
論理ページ先送り機能 (NEXTLP)、MFS [948](#)  
論理ページ要求機能 (PAGE=nn)、MFS [948](#)  
論理リンク  
  定義 [715](#)  
  MSC (複数システム結合機能) [685](#)  
論理リンク・パス  
  MSC (複数システム結合機能) [686](#)  
論理リンク・パスに対するシステム識別 [686](#)

## [ワ行]

ワークステーション  
  金融機関通信システム [932](#)  
  SLU P [932](#)  
ワークステーション、ユーザー  
  ブラケット・プロトコル  
  出力メッセージ、ISC [532](#)  
  入力メッセージ、ISC [529](#)  
用語 [473](#)  
論理装置として定義 [473](#)  
論理装置の定義 [377](#)  
API  
  CICS 同期 [603](#), [604](#)  
  CICS 非同期 [603](#), [604](#)  
ワークステーション開始 [431](#)

## [数字]

1 次リソース名 (PRN) [577](#)  
1 フェーズ・コミット  
  IMS Connect  
  IMS Universal ドライバー [359](#)  
  IMS TM Resource Adapter [367](#)  
  IMS Universal Database リソース・アダプター [359](#)  
  IMS Universal ドライバー [359](#)  
119ABEND コード [856](#)  
2 次論理装置  
  ISC のファースト・スピーカー [468](#)  
2 次論理装置 (secondary logical unit)  
  設計の考慮事項 [468](#)  
2 次論理装置タイプ P [942](#)  
2 つの IMS システム間の通信 [473](#)  
2 フェーズ・コミット (two-phase commit)  
  アプリケーション・コンポーネント [356](#)  
  アプリケーション・サーバー [356](#)  
  エンタープライズ情報システム [356](#)  
  概要 [355](#)  
  グローバル・トランザクション [363](#)  
  コミット・フェーズ [363](#)  
  コンテキスト・トークン [363](#)  
  サーバー分散同期点マネージャー (SDSRM) [356](#)  
  準備フェーズ [363](#)  
  通信リソース・マネージャー (CRM) [356](#)  
  トランザクション・マネージャー [356](#)  
  分散 2 フェーズ・コミット [356](#)  
  分散クライアント・フロー [363](#), [365](#)  
  リソース・アダプター [356](#)  
  リソース・マネージャー [356](#)  
  IMS Connector for Java [356](#)  
  IMS Connect

2 フェーズ・コミット (two-phase commit) (続き)  
IMS Connect (続き)  
IMS TM トランザクションの LPAR 間サポート [366](#)  
IMS Universal ドライバー の 1 フェーズ・コミット [359](#)  
IMS Connect サポート [355](#)  
IMS Connect の IMS TM Resource Adapter サポート [363](#)  
IMS Connect の IMS Universal ドライバー サポート [356](#)  
IMS TM Resource Adapter  
1 フェーズ・コミット最適化 [367](#)  
IMS Connect サポート [363](#)  
IMS Universal Database リソース・アダプター  
1 フェーズ・コミット最適化 [359](#)  
IMS Universal ドライバー  
グローバル・トランザクション [356](#)  
コミット・フェーズ [356](#)  
コンテキスト・トークン [356](#)  
準備フェーズ [356](#)  
分散クライアント・フロー [356](#)  
1 フェーズ・コミット最適化 [359](#)  
IMS Connect サポート [356](#)  
X/Open XA プロトコル [356](#)  
z/OS リソース・リカバリー・サービス (RRS) [356](#)  
2 フェーズ・コミット処理 [124](#)  
2 フェーズ・コミット・プロセス、定義 [29](#)  
3270 コピー・コマンド [451](#)  
3600 端末  
金融機関または SLU P としてサポート [931](#)  
3650 システム、SLU P としてサポートされる [931](#)  
4700 端末  
金融機関または SLU P としてサポート [931](#)  
4730 自動取引装置  
XRF 環境での接続 [938](#)  
8100 システム、SLU P としてサポートされる [931](#)

## A

ACEE  
フラッシュ制御 [818](#)  
IMS Connect  
OTMA ACEE エージング値の指定 [207](#)  
OTMA ACEE のリフレッシュ [834](#)  
OTMA、ACEE のリフレッシュ [834](#)  
ACK  
送信専用プロトコル [307](#)  
ACK オプション  
金融機関通信システム [981](#), [982](#)  
SLU P [981](#), [982](#)  
ACK タイムアウト  
送信後コミット・トランザクション (CM1) [827](#)  
ALTPCB  
共用キューと IMS Connect [179](#)  
IMS Connect と共用キュー [179](#)  
OTMA  
共用キュー、出力のリトリブ [812](#)  
OTMA 宛先記述子  
宛先名のマスキング [799](#)  
定義 [799](#)  
IMS 間通信フローでの役割 [780](#)  
ALTPCBE [812](#)  
API (アプリケーション・プログラム・インターフェース)  
暗黙的 [379](#)  
明示的 [379](#)

API (アプリケーション・プログラム・インターフェース)、  
ISC  
VTAM の IMS 使用 [467](#)  
API (アプリケーション・プログラム・インターフェース)、  
ISC (システム間連絡)  
同期、CICS [603](#), [604](#), [639](#)  
非同期、CICS [603](#), [604](#), [639](#)  
APPC  
アウトバウンド LU  
指定 [43](#)  
APPC (拡張プログラム間通信機能)  
宛先構造 [397](#)  
宛先コード [397](#)  
キューイング、SERIAL=YES オプション [396](#)  
交換 [401](#)  
構造 [397](#)  
セグメント、単一または複数 [397](#)  
タイプ [396](#)  
としてのコミュニケーション・マネージャー [31](#)  
入力セグメント [397](#)  
編集とフォーマット設定 [435](#)  
APPC/IMS [396](#)  
DFSAPPC [396](#)  
IMS メッセージ [396](#)  
LU 6.2 メッセージ [396](#)  
SYNCLVL=SYNCPT [31](#)  
APPC (拡張プログラム間通信機能)/IMS  
アプリケーション・プログラム  
標準 IMS [35](#), [36](#)  
変更 IMS [35](#)  
リモート標準 IMS [37](#)  
CPI-C ドリブン [35](#)  
エラー抽出サービス [45](#)  
概要 [33](#)  
サイド情報  
mode\_name [43](#), [395](#)  
partner\_LU\_name [43](#), [395](#)  
TP\_name [43](#), [395](#)  
制約事項 [36](#)  
セキュリティ [58](#)  
タイムアウト・サービス [45](#)  
デフォルト会話特性 [57](#)  
トランザクション再試行特性  
高速機能再試行条件 [53](#)  
デッドロック [53](#)  
ロック拒否 [53](#)  
トランザクション・セキュリティ  
ACCESS (EXECUTE) [58](#)  
UACC (NONE) [58](#)  
ネットワーク修飾 LU 名 [53](#)  
パートナー LU 名 [43](#)  
廃棄可能メッセージ [35](#)  
廃棄不能メッセージ [35](#)  
非同期出力送達 [57](#)  
フラッシュ制御 [34](#)  
変更 IMS アプリケーション・プログラム [37](#)  
モード名 [43](#)  
リカバリー可能性 [432](#)  
リカバリー可能トランザクションとリカバリー不能トランザクション [47](#)  
リモート APPC トランザクションに対するリモート・サービス [38](#)  
リモート APPC トランザクションに対するローカル・サービス [36](#), [38](#)



## APPC (拡張プログラム間通信機能)/IMS (続き)

### API

暗黙的 [35, 379](#)

明示的 [35, 379](#)

### APPC/IMS の設定

システム定義 [40](#)

APPC=Y あるいは N [40](#)

### APPC/MVS 管理ユーティリティ (ATBSDFMU) の例 [41](#)

### CPI 通信アプリケーション・プログラム [25](#)

### CPI-C 初期設定 [395](#)

### DFSAPPC

オプション・キーワード [55](#)

メッセージ通信 [55](#)

### LTERM インターフェース [39](#)

### LU 6.2

装置 [379](#)

端末サポート [395](#)

APPC/IMS との関係 [33](#)

### LU 名の修飾 [53](#)

### LU、再割り当て [54](#)

### LU、複数の管理 [54](#)

### LUADD オプション・キーワード [44](#)

### MOD 名 [39](#)

### MSC

アプリケーション・プログラムの障害およびトランザクションの回復 [50](#)

中間 IMS 障害およびトランザクションの回復 [50](#)

トランザクションの障害点 [48](#)

標準 IMS アプリケーション [36](#)

リモート IMS 障害およびトランザクションの回復 [50](#)

ローカル IMS 障害およびトランザクションの回復 [49](#)

ローカル・トランザクションの廃棄可能性 [48](#)

APPC におけるトランザクションの回復 [47](#)

LU 6.2 セッションおよびトランザクションの回復 [48](#)

LU 6.2 リカバリー可能性フロー [51](#)

MSC リンクおよびトランザクションの回復 [49](#)

### MSC (複数システム結合機能)

IMSplex 内のリモート・トランザクション処理 [738](#)

### MSC および変更 IMS アプリケーション [38](#)

### PARMLIB [44](#)

### RACF [58](#)

### SIADD [43](#)

sym\_dest\_name [43](#)

Sync\_Level オプション (NONE、CONFIRM、SYNCPT) [36, 37](#)

### SYS1.APPCSI [43](#)

### SYS1.APPCTP [40](#)

### SYS1.PARMLIB(APPCPMxx) の例 [44](#)

### TP プロファイル

ダイアログ例 [40](#)

定義 [38](#)

ATBSDFMU ユーティリティ [40](#)

DFSTPPE0 [42](#)

DFSTPROF [41](#)

TSO ICQASRM0 [41](#)

VSAM データ・セット [40](#)

### TP 名 [43](#)

TPN (トランザクション・プログラム名) [42, 55, 56](#)

### XRF [52](#)

### APPC におけるトランザクションの回復 [47](#)

### APPC/IMS

## APPC/IMS (続き)

および保護トランザクション [862](#)

### APPC/MVS 管理ユーティリティ (ATBSDFMU) の例 [41](#)

APPLID= パラメーター、ISC セッション定義 [494](#)

ATCCONxx メンバー (VTAM ノード) [429](#)

ATCSTRyy メンバー (VTAM 開始リスト) [429](#)

ATRABCK サービス [862](#)

ATRACMT サービス [862](#)

ATREINT サービス [862](#)

ATTACH FM ヘッダー

形式 [571, 579](#)

パラメーターの説明 [573, 579](#)

ビット内容 [571, 573](#)

プロセス開始 [557](#)

CICS で [646](#)

EB 標識 [472](#)

MFS [564](#)

ATTIU パラメーター、FM ヘッダーの長さ [557](#)

## B

### BB (ブラケット開始) 標識

使用 (図) [938](#)

定義 [477](#)

LUSTATUS コマンド [542](#)

### BID オプション

効果

出力メッセージ [945](#)

表示画面保護 [948](#)

MFS ページング [948](#)

設計の考慮事項 [944, 952](#)

セッション終了が原因 [984](#)

BID コマンド [528, 944, 952](#)

### BIND 競争 [503](#)

BINPDSB1= パラメーター BINTRNDS オプション [445](#)

### BIS (ブラケット開始の停止) コマンド

セッション・シャットダウン [554](#)

IMS-CICS セッション [638](#)

### BPE (基本プリミティブ環境)

ヘッダー・データ [248](#)

ヘッダーの形式 [240](#)

## C

### callout

#### 端末

中断状態の端末の回避 [885](#)

#### 同期

エラー応答、IMS Connect へのリターン [217](#)

応答、IMS Connect に返す [308](#)

確認応答メッセージ、IMS Connect

IRM\_F3\_REROUT [215](#)

確認応答メッセージ、IMS Connect NAK [215](#)

確認応答メッセージ、IMS Connect SYNCNAK [216](#)

コールアウト応答のリターン [216](#)

コールアウト要求のリトリブ [212](#)

送信専用プロトコル [216](#)

タイムアウト、OTMA ACK [825](#)

プログラム間通信 [879](#)

ユーザー作成の IMS Connect クライアントのコーディング [209](#)

IMS Connect 関連トークンの構造 [249, 256](#)

callout (続き)  
同期 (続き)  
IMS Connect に対する確認応答メッセージ (ACK および NAK) [214](#)  
IMS Connect メッセージ・フォーマット [211](#)  
IMS Connect、応答を返す [308](#)  
RESUME TPIPE のエラー・シナリオ [214](#)  
RESUME TPIPE 呼び出し [212](#)  
非同期  
応答の相関 [886](#)  
入力端末への応答の経路指定 [886](#)  
IMS Connect クライアントへの応答の経路指定 [886](#)  
IBM MQ [884](#)  
IMS Connect サポート [209](#)  
OTMA  
経路指定 [882](#)  
SOAP Gateway [883](#)  
CANCEL コマンド  
プロトコル [536](#)  
ページング・エラー [544](#), [552](#)  
SLU P セッション [985](#)  
CANCEL 要求  
送信側 ERP [550](#)  
CAPEX (機能交換)  
CICS からの ISC TCP/IP セッションの開始 [618](#)  
CICS との ISC TCP/IP セッションの開始 [617](#)  
CASCADE=  
IMS TM トランザクションの IMS Connect LPAR 間サポートの使用可能化 [366](#)  
CBresynch [866](#)  
CBresynch コマンド [894](#)  
CCTL  
CICS の接続 [765](#)  
CD (方向変換) 標識  
金融機関通信システム [937](#)  
送信 / 受信のプロトコル [938](#), [966](#)  
定義 [477](#)  
要求 [524](#)  
LUSTATUS コマンド [542](#)  
SLU P [937](#)  
CHANGE (/CHANGE) コマンド  
ISC セッションの開始 [506](#)  
CHASE コマンド [538](#)  
CHECK セキュリティー・レベル [830](#)  
CICS  
アプリケーション  
実行、IMS Universal ドライバー を使用 [22](#)  
構成  
DRA のタスク [766](#)  
順次バッファリング [764](#)  
タイプ 2 IMS Universal ドライバー のインストール [20](#)  
CCTL 接続 [765](#)  
DFST2CIC DLL [20](#)  
DL/I サポート [764](#)  
DRA  
PSB のコーディング [763](#)  
DRA 構成 [766](#)  
IMS Universal JDBC ドライバー、タイプ 2  
構成 [20](#)  
IMS Universal ドライバー  
アプリケーションの実行 [22](#)  
IMS Universal ドライバー、タイプ 2  
構成 [20](#)  
IMS データベースへのアクセス [763](#)

CICS (続き)  
ISC  
アプリケーション関連の概念 [639](#)  
サポートされる機能 [601](#)  
ISC (システム間連絡)  
セッション再始動、TCP/IP [620](#)  
セッション終了、TCP/IP [619](#)  
セッション終了、TCP/IP、無条件 [620](#)  
セッション終了、異常、TCP/IP [620](#)  
セッション終了、正規手順での、TCP/IP [620](#)  
TCP/IP セッション再始動 [620](#)  
TCP/IP セッション終了 [619](#)  
TCP/IP セッション終了、異常 [620](#)  
TCP/IP セッション終了、正規手順での [620](#)  
TCP/IP セッション終了、無条件 [620](#)  
TCP/IP セッションの再始動 [620](#)  
TCP/IP セッションの終了 [619](#)  
TCP/IP セッションの終了、正規手順での [620](#)  
TCP/IP セッションの終了、無条件で [620](#)  
ISC TCP/IP  
CICS フロントエンド・トランザクション・タイプ [621](#)  
ISC TCP/IP サポート  
機能交換 (CAPEX) [617](#), [618](#)  
セッション、開始 [617](#)  
セッションの開始、CICS から [618](#)  
ISC サポート  
ISC TCP/IP リンクの定義、CICS への [615](#)  
ISC ノード  
定義 [629](#)  
Make ファイル [20](#)  
TCP/IP  
セッション再始動、ISC [620](#)  
セッション終了、ISC [619](#)  
セッション終了、ISC、無条件 [620](#)  
セッション終了、異常、ISC [620](#)  
ISC セッション再始動 [620](#)  
ISC セッション終了 [619](#)  
ISC セッション終了、異常 [620](#)  
ISC セッション終了、正規手順での [620](#)  
ISC セッション終了、無条件 [620](#)  
ISC セッションの再始動 [620](#)  
ISC セッションの終了 [619](#)  
ISC セッションの終了、正規手順での [620](#)  
ISC セッションの終了、無条件で [620](#)  
\_CXX\_LSYSLIB 環境変数 [20](#)  
CICS からの IMS コマンドの送信 [643](#)  
CICS のコーディング  
システム定義マクロ [629](#)  
表 [629](#)  
ISC のアプリケーション [638](#)  
CICS リソース定義 [629](#)  
CICS を動作可能にする [636](#)  
CICS-IMS 間の通信  
会話モード [642](#)  
機能  
概要 [463](#)  
基本 [642](#)  
説明 [603](#), [604](#)  
代替 [642](#)  
機能管理ヘッダーのコーディング [645](#)  
基本機能 [642](#)  
構成 [460](#)  
コマンド・レベル API [603](#), [604](#)



## CICS-IMS 間の通信 (続き)

システム定義オプションのコーディング [629](#)

処理の流れ

RECEIVE [624](#)  
RETRIEVE [626](#)  
SEND INVITE [621](#)  
SEND LAST [623](#)  
SEND/RECEIVE [621](#)  
START/RETRIEVE [626](#)

セッション

開始 [636](#)  
再確立 [654](#)  
再同期化 [654](#)  
終了 [638](#)  
同期点 [644](#)  
バインディング [654](#)  
保全性 [654](#)  
未解決のトラフィックの処理 [655](#)

セッションの開始 [636](#)

セッションの保全性 [654](#)

装置マッピング機能 [460](#)

代替機能 [642](#)

同期点 [644](#)

トランザクション

異常終了 [640](#), [656](#)  
サポートされる属性 [603](#), [604](#), [640](#)  
定義 [636](#)

のアプリケーション・コーディング [638](#)

非同期処理のフロー [626](#)

ミラー・トランザクション [626](#)

リカバリーおよび再始動 [652](#)

ATTACH パラメーター [646](#)

CICS テーブルの作成 [629](#)

CICS トランザクション [636](#)

CICS トランザクションの定義 [636](#)

IMS コマンド [643](#)

ISC を使用してデータを IMS に受け渡す [464](#)

LU 6.1 リンク

オンライン・リソース定義 [629](#)  
互換性のあるノード [630](#)  
説明 [629](#)  
複数のリンク [634](#)  
マクロ・レベルのリソース定義 [629](#)

MFS サポート [651](#)

MSC リンク [461](#)

SCHEDULER パラメーター [649](#)

CICS-ISC インストール・オプション [629](#)

CIMS クラス [414](#)

Client\_Bid resynch [866](#)

CLSDST (/CLSDST) コマンド

ISC TCP/IP セッション [620](#)

CM0

転送機能 [303](#)

ページ機能 [301](#)

OTMA、出力の管理 [872](#)

OTMA、出力の転送 [873](#)

OTMA、出力のページ [873](#)

OTMA、スーパーメンバー [875](#)

CM1 (送信後コミット) トランザクション

ACK タイムアウト [827](#)

COBOL

XML 変換サポート

概要 [159](#)

COMM マクロ・ステートメント

## COMM マクロ・ステートメント (続き)

APPLID = キーワード [426](#)

RECANY = キーワード [427](#)

Common Queue Server (CQS)

概要 [386](#)

共用キュー環境、内 [384](#)

クライアント

    出口ルーチン [386](#)

クライアントの定義 [383](#)

チェックポイント・データ・セット (checkpoint data set)  
[385](#)

定義 [383](#)

COMPINOP 状態 [407](#)

CPI 通信 [23](#)

CPI 通信アプリケーション・プログラム

異常終了 [26](#)

仮名ファイル [27](#)

経過中のリカバリー単位、定義 [29](#)

コミット処理 [25](#)

コミット呼び出し [25](#)

システム再始動 [27](#)

正常終了 [25](#)

セッションの失敗 [26](#)

バックアウト処理 [25](#)

バックアウト呼び出し [26](#)

プログラミング要件 [27](#)

未確定解決処理 [27](#)

未確定リカバリー単位の定義 [29](#)

戻りコード [26](#)

リカバリー [27](#)

2 フェーズ・コミット・プロセスおよび [29](#)

APSB 呼び出し [25](#)

ATBCMTP verb [26](#)

Db2 for z/OS 計画名の使用 [25](#)

ESS 接続機能 [25](#)

RTT [25](#)

SQL 呼び出し [25](#)

SRRCMIT [25](#)

CPI-C (会話用共通プログラミング・インターフェース)

サイド情報 [395](#)

初期設定 [395](#)

トランザクション [398](#)

APPC/IMS [395](#)

CQS (Common Queue Server)

概要 [386](#)

共用キュー環境、内 [384](#)

クライアント

    出口ルーチン [386](#)

クライアントの定義 [383](#)

チェックポイント・データ・セット (checkpoint data set)  
[385](#)

定義 [383](#)

CRC (コマンド認識文字) [125](#)

CREATE MSNAME コマンド

    論理リンク・パス [686](#)

CREATE MSPLINK コマンド

    CTC 物理リンクの定義 [713](#)

    MTM 物理リンクの定義 [713](#)

    TCP/IP 物理リンクの定義 [714](#)

    VTAM 物理リンクの定義 [715](#)

CRGGRM サービス [862](#)

CRGSEIF サービス [862](#)

CSM (完了状況メッセージ)

形式 [254](#)

CTC (チャンネル間)  
MSC (複数システム結合機能) 物理リンク・タイプ [683](#)  
MSC 物理リンク  
定義 [713](#)  
CTXBEGC サービス [862](#)  
CTXEINT サービス [863](#)  
CTXSWCH サービス [863](#)

## D

D 記述子、OTMA  
限度 [800](#)  
最大 [800](#)  
Db2 for z/OS  
CPI-C  
計画名の使用 [25](#)  
Java 従属領域  
Db2 for z/OS のアクセス [117](#)  
DB2 接続機能  
システムの準備 [117](#)  
DBCTL  
サポートされていない OTMA [814](#)  
初期設定 [765](#)  
DBFHAGUO (入力編集/経路指定出口ルーチン) [391](#)  
DCCTL (データ通信制御) [381](#)  
DCCTL (データ通信制御機能)  
生成 [381](#)  
プロシージャ [381](#)  
IMS BTS [381](#)  
TM バッチ [381](#)  
DEADQ STATUS [103](#)  
DEQUEUE (/DEQUEUE) コマンド  
ISC 出力メッセージのコミット [521](#)  
DEQUEUE コマンド  
FORCESS 対 SYNCSESS [506](#)  
IMS 間 ISC セッション [506](#)  
DFC (データ・フロー制御) プロトコル  
エラー処理  
応答モード [519](#)  
会話モード [519](#)  
選択受信側 ERP [545](#)  
応答要件  
リカバリー可能な照会トランザクション [520](#)  
リカバリー不能照会トランザクション [520](#)  
コマンド  
BID [528](#)  
BIS [554](#)  
CANCEL [536](#)  
CHASE [538](#)  
LUSTATUS [542](#)  
RSHUT [545](#)  
RTR [545](#)  
SBI [554](#)  
SIGNAL [553](#)  
センス・コード [552](#)  
対称セッション・シャットダウン [554](#)  
チェーン [537](#)  
同期点および応答  
可用性 [521](#)  
出力 [526](#)  
出力で要求された [521](#)  
同期入力の例外 [521](#)  
入力 [524](#)  
入力で要求された [521](#)

DFC (データ・フロー制御) プロトコル (続き)  
同期点および応答 (続き)  
リカバリー可能メッセージ [522](#)  
リカバリー不能メッセージ [524](#)  
CICS-IMS [644](#)  
MFS 出力メッセージ、ISC [522](#)  
入力メッセージ、バックアウト [521](#)  
ハーフセッションの同期化 [520](#)  
ブラケットおよび半二重 [528](#)  
ページ化メッセージ・エラー [544](#)  
メッセージのブラケット化  
出力 [532](#)  
入力 [529](#)  
例 [659](#)  
例外応答 [519](#)  
例外応答後の PURGE [538](#)  
ERP PURGE [538](#)  
LWA によって補助される回復性 [520](#)  
DFS3650I (セッション状況メッセージ) [427](#)  
DFS AERA0 モジュール [768](#)  
DFS AERGO モジュール [768](#)  
DFS AERMO モジュール [768](#)  
DFSAPPC メッセージ通信 [55](#)  
DFSCCMD0 [804](#)  
DFSCDLIO モジュール [768](#)  
DFSCMUX0 出口ルーチン [719](#)  
DFSCTRNO [804](#)  
DFSEMODL マクロ [127](#)  
DFSEWAL マクロ [130](#)  
DFSFEBJ0 (フロントエンド切り替え出口ルーチン)  
特殊サポート [457](#)  
DFSMSCEO [804](#)  
DFSOTMA 記述子 [800](#)  
DFSPBxxx  
OTMA パラメーター  
GRNAME [790](#)  
OTMA= [791](#)  
OTMAASY [795](#)  
OTMAMD [795](#)  
OTMANM [790](#)  
OTMASE [791](#)  
OTMASP [794](#)  
DFSQSP0 [804](#)  
DFSSIML0 出口ルーチン [452](#)  
DFS YDRU0 出口ルーチン [804](#)  
DFS YIOE0 出口ルーチン [804](#)  
DFS YMSG DSECT [889](#)  
DFS YPRX0 出口ルーチン [804](#)  
DFS YRTUX 出口ルーチン [804](#)  
DIF (装置入力形式) [442](#)  
DISPLAY (/DISPLAY) コマンド  
QSTOP 状態 [433](#)  
DL/I  
OTMA  
使用される呼び出し [841](#)  
DOF (装置出力形式) [442](#)  
DPM (分散表示管理)  
オプション [447](#)  
出力 [973](#)  
FM ヘッダー [562](#)  
OPTIONS=DPAGE または PPAGE [562](#), [973](#)  
SLU P サポート [946](#)  
DPN (宛先プロセス名)  
メッセージ経路指定、ISC [479](#)

## DPN (宛先プロセス名) (続き)

ATTACH FM ヘッダー [575](#)

SCHEDULER FM ヘッダー [575](#)

## DR2 応答

### 要件

通常 IMS 出力メッセージ [942](#)

リカバリー可能照会トランザクション [980](#)

MFS ページ出力メッセージ、非高速機能 [942](#)

### 例外

高速機能 [955](#)

高速機能 (Fast Path) [954](#)

リカバリー不能出力 [942](#)

リカバリー不能照会トランザクション [979](#)

MFS 出力 [978](#), [979](#)

MOD で PAGE=YES を指定しない場合 [963](#), [980](#)

## DRA

### CICS

DRA 構成 [766](#)

## DRA (データベース・リカバリー・アダプター)

セットアップ [767](#)

## DRA (データベース・リカバリー・アダプター) 始動テーブル

ODBA (Open Database Access) での使用 [767](#)

Open Database Access (ODBA) での使用 [767](#)

## DRA (データベース・リソース・アダプター)

### 接続

構成 [761](#)

## DRD

MSC に対する使用可能化 [706](#), [707](#)

## DRDA (分散リレーショナル・データベース体系)

IMS Connect サポート [151](#)

## DSCA オペランド、DEV ステートメント [451](#)

## E

## EB (ブラケット終了) 標識

使用 (図) [938](#)

定義 [477](#)

ATTACH と一緒に使用 [472](#)

LUSTATUS コマンド [542](#)

## ECSA

OTMA の使用 [824](#)

## EEVPEEA [135](#)

## EEVPEWA [137](#)

## EEVT (外部エントリー・ベクトル・テーブル) [136](#)

## EEVT のマッピング [136](#)

## EMH (急送メッセージ・ハンドラー)

キュー・オプション

概要 [390](#)

## EMH バッファ 415

## EMHL 制御領域初期設定パラメーター [415](#)

## ERP (エラー・リカバリー・プロシージャ)

拡張 [652](#)

機能管理ヘッダー [652](#)

選択受信側

センス・コード [545](#)

CICS-IMS セッション [652](#)

FM ヘッダー

形式 [580](#)

IMS が実行する [934](#)

## ERP PURGE、ISC 例外応答後 [538](#)

## ESMT (外部サブシステム・モジュール・テーブル)

外部サブシステム・モジュールのロード [136](#)

作業域定義 [130](#)

作成 [127](#)

## ETO

装置特性テーブル

構築 [91](#)

## 端末

成長に備えた計画 [76](#)

IDCO トレース機能 [110](#)

XRF による ETO 端末のリカバリー [88](#)

## ETO (拡張端末オプション)

アルゴリズム

ログオン記述子 [86](#)

LTERM の割り振り [98](#)

印刷関連付けの技法 [100](#)

印刷の共用 [102](#)

応答モード (response mode) [112](#)

概念

要約 [69](#)

概要 [63](#)

会話 [112](#), [400](#)

カスタマイズ [68](#), [72](#)

画面サイズ制御バイト [78](#)

画面定義の例 (非 SNA 3270)

ディスプレイ (画面サイズ指定) [80](#), [81](#)

ディスプレイ (モデル指定) [80](#), [81](#)

モデル 2 プリンター [80](#)

LUO ビデオ [80](#)

## 記述子

更新 [93](#)

コーディングの概要 [84](#)

削除 [93](#)

システム定義中の作成 [84](#)

使用 [68](#)

追加 [93](#)

定義 [65](#)

ログオン (logon) [85](#)

ログオン、定義 [65](#)

MFS [90](#)

MSC (複数システム結合機能) [92](#)

user [88](#)

VTAM TERMINAL マクロ [85](#)

記述子の格納 [85](#)

キュー (動的ユーザー・メッセージ)、定義 [63](#)

共通ログオン記述子 [85](#)

## 計画

運用 [83](#)

ユーザー ID [82](#)

ユーザー・キュー名 [83](#)

LU2 [78](#)

構造、作成と削除 [67](#)

## 構造

端末、定義 [64](#)

ユーザー、定義 [64](#)

サインオフの定義 [109](#)

サインオン

サインオン・データの提供 [96](#)

LTERM の割り振り [95](#)

サインオン出口ルーチン (DFSSGNX0) [101](#)

サインオンの定義 [95](#)

システム定義 [84](#)

自動サインオフ [104](#)

自動ログオフ [105](#)

自動ログオン (autologon) [106](#)

## 出力

不注意による出力データ・ストリーム [109](#)

割り当て [107](#)

ETO (拡張端末オプション) (続き)  
状況を保存するコマンド [99](#)  
状況をリセットして制御ブロックを解放するコマンド [99](#)  
使用の利点 [63](#)  
初期設定  
記述子妥当性検査 [93](#)  
DFSINTX0 [93](#)  
推奨事項 [90](#)  
制御ブロックの削除  
サインオフ後 [109](#)  
ログオフ後 [109](#)  
静的端末  
動的端末、併用 [75](#)  
静的端末、定義 [63](#)  
制約事項 [72](#)  
セキュリティー  
静的端末と動的端末 [81](#)  
選択の指針  
ログオン記述子 [86](#)  
LOGOND パラメーター [86](#)  
送達不能キュー [107](#), [108](#)  
装置タイプ、定義 [76](#)  
装置特性テーブル [78](#), [90](#)  
端末  
静的端末と動的端末の併用 [75](#)  
定義 [63](#)  
端末 - LTERM 関連 [98](#)  
定義  
自動サインオフおよび自動ログオフ・タイマー [106](#)  
パラメーター [103](#)  
出口ルーチン  
サインオン出口ルーチン (DFSSGNX0) [101](#)  
使用 [68](#)  
出口ルーチン、コーディング [92](#)  
出口ルーチンのリスト [92](#)  
デフォルト CINIT/BIND ユーザー・データ形式 [95](#)  
動的端末  
静的端末、併用 [75](#)  
動的端末 (dynamic terminal)  
定義 [63](#)  
動的ユーザー、定義 [64](#)  
特殊処理モード [99](#)  
特定の宛先を持つ LTERM [98](#)  
ノード・ユーザー記述子 [89](#)  
配布不能データ、送達不能キュー  
/DISPLAY STATUS USER コマンド [108](#)  
/DISPLAY USER DEADQ コマンド [108](#)  
非 SNA 3270 装置  
画面サイズとモデル情報 [78](#)  
プリンターおよびディスプレイ [76](#)  
非同期出力  
宛先、有効 [107](#)  
非発信元端末への出力メッセージ送達 [108](#)  
複数のサインオン [96](#)  
プリンター  
概要 [100](#)  
直接印刷 [100](#)  
定義 [101](#)  
プリンター・ノード名 [100](#)  
プリンターの共用 [102](#)  
ユーザー記述子  
インストール先作成 [89](#)  
インストール時作成 [88](#)

ETO (拡張端末オプション) (続き)  
ユーザー記述子 (続き)  
システム定義中の作成 [88](#)  
選択の基準 [88](#)  
ノード・ユーザー記述子 [88](#), [89](#)  
DFSUSER [90](#)  
要件 [72](#)  
用語 [63](#)  
利点  
可用性 [71](#)  
LTERM [71](#)  
ログオフ [109](#)  
ログオン (logon)  
サインオン・データ [94](#)  
動的、特定の端末タイプへの制限 [94](#)  
INITOTHER [94](#)  
INITSELF [94](#)  
USS LOGON [94](#)  
/OPNDST コマンド [94](#)  
ログオン記述子  
システム定義中の作成 [85](#)  
NTO 端末、3600/ 金融機関端末 [87](#)  
3275 装置 [77](#)  
3600/ 金融機関 [111](#)  
ABENDU0015 [93](#)  
DFS2085 [96](#)  
DFS3641W [93](#)  
DFS3645 [96](#)  
DFS3649A [97](#)  
DFS3650I [98](#)  
DFS3672 [96](#)  
DFSINSX0 [90](#)  
DFSSGNX0 [90](#)  
DFSUSER 記述子 [90](#)  
DLQT [108](#)  
ETO STSN 装置の /SIGN コマンド [112](#)  
ETO 開始 [93](#)  
ETO ログオン記述子での MODETABL [428](#)  
ISC TCP/IP [63](#)  
LTERM  
制御ブロックの作成および再利用 [94](#)  
LU 2  
画面サイズとモデル情報 [78](#)  
装置 [77](#)  
LU 6.1 (ISC) 端末 [110](#)  
MFS [82](#)  
MFS 装置記述子  
MFSDCT ユーティリティー (DFSUTB00) [91](#)  
MFS 装置特性テーブル [78](#), [90](#)  
MFSDCT ユーティリティー [78](#)  
MSC (複数システム結合機能)  
記述子 (descriptor) [83](#)  
サポート [83](#)  
MSNAME マクロ [83](#)  
NTO 装置 [78](#)  
RACF [379](#)  
SLU P [111](#)  
SNA コマンド [94](#)  
STSN 端末  
概要 [111](#)  
/SIGN コマンドのサポート [112](#)  
VTAM CINIT LUNAME [86](#)  
VTAM 考慮事項  
CINIT セッション制御ブロックのログオン [76](#)

ETO (拡張端末オプション) (続き)

VTAM 考慮事項 (続き)

VTAM PSERVIC パラメーター [76](#)

ETO STSN 装置の /SIGN コマンド [112](#)

ETO 装置に対する STSN サポート [112](#)

EWAL [130](#)

EXEC のパラメーター、SSM [120](#)

EXIT (/EXIT) コマンド

会話の異常終了 [405](#)

ISC 会話モード・エラー [519](#)

express\_context\_interest サービス [863](#)

## F

FM (機能管理) ヘッダー

エラー・リカバリー・プロシージャ (ERP)

形式 [580](#)

金融機関通信システム [967](#), [973](#)

出力 FM ヘッダー [973](#)

出力プロセス名 [577](#)

出力メッセージ (output message) [471](#)

セキュリティー [459](#)

タイプ X'42' (SLU P) [970](#)

入力 FM ヘッダーの長さ

ATTACH FM ヘッダー [557](#)

入力プロセス名 [577](#)

入力メッセージ [469](#)

メッセージ記述子バイト形式

出力 [973](#), [975](#)

入力 [967](#), [969](#)

要求される処理モード [471](#)

CICS で使用するためのコーディング [645](#)

IMS によって挿入される [447](#)

IMS、データ記述子 [651](#)

ISC

エラー・リカバリー・プロシージャ [557](#), [652](#)

応答 QMODEL FM ヘッダー [568](#)

概要 [555](#)

システム・メッセージ [559](#), [652](#)

データ記述子 [565](#), [579](#)

同期処理 [557](#)

入力 QMODEL FM ヘッダー [567](#)

プロセスの開始 [557](#)

ATTACH [557](#)

ATTACH, MFS [564](#)

DPM メッセージ、MFS [562](#)

ISC 編集の呼び出し [556](#)

MFS [561](#)

QMODEL [566](#), [581](#)

QMODEL, CICS [651](#)

RAP [558](#)

RAP, MFS [569](#)

SCHEDULER [558](#), [649](#)

SCHEDULER, MFS [564](#)

SYMSMSG [559](#)

MFS [651](#)

SLU P [968](#), [975](#), [976](#)

FM ヘッダー [447](#)

FMH [447](#)

FPACK/NFPACK オプション、高速機能 [954](#)

FULL セキュリティー・レベル [831](#)

## G

GRAPHIC= パラメーター

オプションなし [447](#)

SEG ステートメント [447](#)

GRNAME パラメーター [790](#)

## H

HANDLE CONDITION、CICS [637](#)

HIOP (高入出力プール)

VTAM 出力バッファ

RECASZ 実行パラメーター [427](#)

HIOP ストレージ・プール

一時的な不足、OTMA の再登録 [810](#)

HWS\$DLQ [352](#)

HWSIMSCB マクロ [223](#)

HWSJAVA0

Local オプション・クライアント通信 [148](#)

HWSOMCTL DSECT [261](#)

HWSOMUSR DSECT [285](#)

HWSSMPL0

および IRM [229](#)

単純なフローでのメッセージ構造 [259](#)

HWSSMPL1

および IRM [229](#)

単純なフローでのメッセージ構造 [259](#)

HWSXIB マクロ [168](#)

HWSXIB1 マクロ [172](#)

HWSXIBDS マクロ [169](#)

HWSXIBOD マクロ [173](#)

## I

IBM MQ

OTMA

同期化された T パイプ、定義 [794](#)

ICAL

同期プログラム間通信 [879](#)

IMS

会話 (conversation)

およびコミット後送信モード [851](#)

サポートされる z/OS リソース・リカバリー・サービス 出口 [862](#)

スケジューラー・メッセージ・ブロック (SMB)  
(Scheduler Message Block (SMB))

OTMA [778](#)

制御領域サイズおよび OTMA [824](#)

トランザクション

同期化 T パイプを使用する [865](#)

非同期化 T パイプを使用して [865](#)

ハイパフォーマンス・アクセス [776](#)

標準フロー [852](#)

保護トランザクションの処理 [863](#)

リカバリー可能トランザクション [305](#)

DFSnnnnn メッセージ [351](#)

IMS.ADFSMAC [889](#)

OTMA での装置サポート [776](#)

OTMAASY 始動パラメーター [843](#)

IMS Connect

概要 [147](#)

会話型プログラム・サポート [295](#)

会話型プロトコル



## IMS Connect (続き)

### 会話型プロトコル (続き)

- 送信後コミット、同期レベル = CONFIRM (ACK 応答) [299](#)
- 送信後コミット、同期レベル = CONFIRM (NAK 応答) [300](#)
- 送信後コミット、同期レベル = NONE、クライアントが終了したトランザクション [298](#)
- 送信後コミット、同期レベル = NONE、プログラムが終了したトランザクション [297](#)

### 可用性、検査 [352](#)

### 共用キューと ALTPCB 出力 [179](#)

### クライアント ID

#### 重複の回避 [316](#)

#### 取り消し [316](#)

### クライアント通信、Local の制約事項 [148](#)

### クライアント通信、TCP/IP [148](#)

### クライアント呼び出しのフロー [348](#)

### コールアウト・サポート、同期

#### エラー応答のリターン [217](#)

#### 確認応答メッセージ (ACK および NAK) [214](#)

#### 確認応答メッセージ、IRM\_F3\_REROUT [215](#)

#### 確認応答メッセージ、NAK [215](#)

#### 確認応答メッセージ、SYNCAK [216](#)

### コールアウトのサポート [209](#)

### コールアウト要求

#### OTMA からのリトリブ [331](#)

### コミット後送信

#### DL/I PURG 呼び出しを無視する [242](#)

### コミット後送信、転送機能 [303](#)

### コミット後送信、ページ機能 [301](#)

### コミット後送信出力の転送

#### 宛先の指定 [304](#)

#### 出力が経路指定される状況 [304](#)

#### 複数メッセージ出力のページ [305](#)

### コミット・モード 0、転送機能 [303](#)

### コミット・モード 0、ページ機能 [301](#)

### 実行

#### 概要 [162](#)

### 自動的に再接続

#### IMS 間 TCP/IP 通信 [159](#)

### 出力の転送 [303](#)

### 出力のページ [301](#)

### 出力メッセージ構造

#### クライアントに対する [242](#)

### スーパーメンバー

#### IMS 間 TCP/IP 通信 [158](#)

### セキュリティ

#### 概要 [161](#)

#### サンプル出口 RACROUTE 呼び出しへのエラー応答 [193](#)

#### トラステッド・ユーザー [206](#)

#### パスチケット、RACF [201](#)

#### 非同期保留キュー [190](#)

#### ユーザー・メッセージ出口ルーチン [165](#)

#### IMS Connect インスタンス間の接続 [188](#)

#### IMS Universal ドライバー [185](#)

#### OTMA ACEE エージング値の指定 [207](#)

#### OTMA RESUME TPIPE セキュリティー・ユーザー出口 (OTMARTUX) [190](#)

#### RACF パスチケット [202](#)

#### RESUME TPIPE [190](#)

### セキュリティ、デフォルトの RACF ユーザー ID [184](#)

### セキュリティ出口ルーチン [190](#)

## IMS Connect (続き)

### 送信後コミット・タイムアウト値 [293](#)

### 送信専用プロトコル

#### 確認応答付き [307](#)

#### 順次配信付き [308](#)

#### 同期コールアウト、コーディング [216](#)

### 送信専用プロトコルおよびコールアウト要求 [308](#)

### ソケット

#### 警告しきい値 [316](#)

#### 警告用のパーセンテージの設定 [316](#)

#### 最大数 [313](#)

#### トランザクションの処理 [312](#)

#### 予約 [315](#)

#### リセットしきい値 [316](#)

#### UNIX システム・サービスの最大数 [314](#)

### ソケット接続

#### 持続 [309](#)

#### トランザクション (transaction) [310](#)

#### 非持続 [310](#)

### 大/小文字混合パスワード [200](#)

### 代替クライアント ID [345](#)

### タイマー

#### 取り消し [327](#)

### タイマー取り消し [327](#)

### タイムアウト値、CM1 ACK/NAK [293](#)

### タイムアウト間隔、設定 [320](#)

### タイムアウト指定

#### IMS DB クライアント [319](#)

#### IMS TM クライアント [319](#)

#### IMS 間 TCP/IP 接続 [329](#)

### 定義

#### 概要 [162](#)

### データ・ストア

#### 状況 [169](#)

### データ・ストア、IMS DB

#### 状況 [173](#)

### 出口インターフェース・ブロック

#### フォーマット、IMS DB 接続 [172](#)

#### XIB1 フォーマット [172](#)

#### XIBDS [169](#)

#### XIBOD のフォーマット [174](#)

### 出口インターフェース・ブロック (XIB)

#### エンTRIES のフォーマット [168](#)

### 出口インターフェース・ブロック (XIB1) [172](#)

### 出口インターフェース・ブロック ODBM データ・ストア (XIBOD) [173](#)

### 出口インターフェース・ブロック・データ・ストア・エン

#### トリー

#### エンTRIES のフォーマット [170](#)

### 出口ルーチン

#### 概要、ユーザー・メッセージ出口ルーチン [164](#)

#### 概要 [163](#)

#### 出口インターフェース・ブロック (XIB) [168](#)

#### 出口インターフェース・ブロック (XIB1) [172](#)

#### 出口から戻るときの入力メッセージ・フォーマット [245](#)

#### マクロ・サポート [167](#)

### 出口ルーチン、機能に固有

#### 概要 [166](#)

### 出口ルーチン、セキュリティ [190](#)

### デッド・レター・キュー [352](#)

### 転送機能 [303](#)

### トランザクション

#### 制約事項 [293](#)

## IMS Connect (続き)

- トランザクション・プロトコル [293](#)
- トランザクション有効期限
  - IRM での期限の設定 [330](#)
- ネットワーク・セキュリティー資格情報の引き渡し [186](#)
- ページ機能 [301](#)
- 配信不能出力のページ
  - 出力がページされる状況 [302](#)
  - HWSSMPL0 および HWSSMPL1 [302](#)
- パスワード管理 [198](#)
- 非同期 OTMA 出力
  - リトリート [331](#)
- 非同期コールアウトのサポート
  - プログラミング [882](#)
- 非同期出力
  - エンド・ユーザー・アプリケーションからの要求 [335](#)
  - 共用 [346](#)
  - グループ化 [346](#)
  - 自動メッセージ制御 [342](#)
  - 待機付き単一 [340](#)
  - 単一メッセージ制御 [339](#)
  - 非オプション・メッセージ制御 [341](#)
  - 非自動メッセージ制御 [341](#)
  - RESUME TPIPE 要求 [332](#)
- 非同期出力サポート [345](#)
- 複数メッセージ出力のページ [302](#)
- プロトコル
  - コールアウト応答のための送信専用 [308](#)
  - 送信専用 [306](#)
  - 並列 RESUME TPIPE 要求、問題の診断 [338](#)
  - 並列 RESUME TPIPE 要求の実装 [337](#)
  - RESUME TPIPE 要求 [332](#)
  - RESUME TPIPE、並列処理 [335](#)
  - RESUME TPIPE、並列処理、使用可能化 [336](#), [337](#)
- プロトコル・レベル [294](#)
- 分散セキュリティー資格情報の引き渡し [186](#)
- 分散リレーショナル・データベース体系 (DRDA) [151](#)
- ヘッダーの形式 [239](#)
- マクロ [167](#)
- メッセージ構造
  - 例 [259](#)
- メッセージ・タイマー
  - 取り消し [327](#)
- メッセージ・タイマーの間隔 [321](#)
- メッセージ・タイムアウト間隔 [320](#)
- メッセージ・フォーマット
  - メッセージ出口への出力 [248](#)
  - IRM 拡張 [237](#)
  - IRM 固定部分 [224](#)
- ユーザー ID、RACF のデフォルト [184](#)
- ユーザー作成クライアント・アプリケーション
  - メッセージ構造 [243](#)
- ユーザー・メッセージ出口
  - 出力メッセージ構造 [239](#)
- ユーザー・メッセージ出口ルーチン
  - 概要 [164](#)
  - セキュリティー [165](#)
- ユニコード [369](#)
- 類似性
  - クリア、MSC TCP/IP リンク [748](#)
- 2 フェーズ・コミット (two-phase commit)
  - IMS TM Resource Adapter サポート [363](#)
  - IMS TM トランザクションの LPAR 間サポート [366](#)

## IMS Connect (続き)

- 2 フェーズ・コミット (two-phase commit) (続き)
  - IMS Universal ドライバー サポート [356](#)
- ALTPCB 出力と共用キュー [179](#)
- BPE ヘッダー・フォーマット [240](#)
- callout
  - 送信専用プロトコル [216](#)
  - 同期、メッセージ・フォーマット [211](#)
  - 同期コールアウト応答のリターン [216](#)
  - 同期コールアウトのための RESUME TPIPE 呼び出し [212](#)
  - 同期コールアウトのためのユーザー作成クライアントのコーディング [209](#)
  - 同期コールアウト要求のリトリート [212](#)
  - RESUME TPIPE のエラー・シナリオ [214](#)
- CM0
  - DL/I PURG 呼び出しを無視する [242](#)
  - CM0 確認応答の OTMA タイムアウト [328](#)
  - CM0 複数セグメント出力の PURG 呼び出しを無視する [242](#)
  - CM0、転送機能 [303](#)
  - CM0、ページ機能 [301](#)
  - CM1 ACK/NAK タイムアウト値 [293](#)
- DFS メッセージ
  - クライアント応答 [350](#)
- HWS\$DLQ [352](#)
- HWSJAVA0
  - ユーザー定義メッセージ [166](#)
- HWSOMCTL DSECT [261](#)
- HWSSMPL0
  - 単純なフローでのメッセージ構造 [259](#)
- HWSSMPL1
  - 単純なフローでのメッセージ構造 [259](#)
- HWSSOAP1 [159](#)
- HWSXIB マクロ [168](#)
- HWSXIB1 マクロ [172](#)
- HWSXIBDS マクロ [169](#)
- HWSXIBOD マクロ [173](#)
- IBM MQ [884](#)
- IMS TM Resource Adapter
  - コールアウトのサポート [883](#)
  - サポートの概要 [152](#)
  - 生成されたクライアント ID [152](#)
  - 重複クライアント ID [152](#)
  - メッセージ構造 [243](#)
  - 2 フェーズ・コミット・サポート [363](#)
- IMS Universal ドライバー
  - 接続経路指定 [152](#)
  - 別名 [152](#)
  - 1 フェーズ・コミット [359](#)
  - 2 フェーズ・コミット・サポート [356](#)
- IMS からの 外部サービスの呼び出し [881](#)
- IMS 間 TCP/IP 通信
  - 概要 [154](#)
  - 自動的に再接続 [159](#)
  - 終了シナリオ [312](#)
  - スーパーメンバー・サポート [158](#)
  - ソケット接続 [312](#)
  - MSC サポート [156](#)
  - OTMA サポート [158](#)
  - TCP/IP 汎用リソース・サポート [157](#)
- IMSplex サポート [177](#)
- IRM
  - 固定部分 [224](#)



## IMS Connect (続き)

- IRM\_F3\_IPURG [242](#)
- ISC
  - サポートの概要 [153](#)
- ISC サポート
  - 定義、ISC TCP/IP リンクの [614](#)
- Local オプション [147](#)
- MAXSOC パラメーター
  - 使用法 [313](#)
  - RESVSOC の影響 [315](#)
- MSC
  - 類似性、クリア [748](#)
- ODBM
  - 状況 [173](#)
- OMHDRIPG [242](#)
- OTMA 宛先記述子 [797](#)
- OTMA 会話型プロトコル
  - 送信後コミット、同期レベル = NONE [295](#), [296](#)
- OTMA の IMS 間 TCP/IP メッセージ・フロー [780](#)
- OTMA メッセージ・ヘッダー
  - 状態データ・フィールド [267](#)
  - セキュリティ・データ・フィールド [281](#)
  - 注記 [292](#)
  - メッセージ制御フィールド [261](#)
  - ユーザー・データ・セクションのフォーマット [285](#)
  - HWSOMCTL DSECT [261](#)
- PING 応答 [255](#)
- ping 機能 [352](#)
- RACF サポート [181](#)
- RACF パスワードの変更 [198](#)
- RACF パスワード・フレーズの変更 [199](#)
- RACF ユーザー ID キャッシュ [185](#)
- RACF、直接サポートの使用可能化 [183](#)
- RACF、デフォルトのユーザー ID [184](#)
- RACF、統計の使用可能化 [183](#)
- RACF、汎用戻りコードまたはメッセージ [182](#)
- RESUME TPIPE 要求
  - 並列処理 [335](#)
  - 並列処理、実装 [337](#)
  - 並列処理、使用可能化 [336](#), [337](#)
  - 並列処理、問題の診断 [338](#)
- RESUME TPIPE
  - 同期コールアウト、コーディング [212](#)
- RESUME TPIPE プロトコル
  - フローの例 [333](#)
- RESUME TPIPE 呼び出し
  - およびタイムアウト [343](#)
- RESVSOC
  - MAXSOC に対する影響 [315](#)
- SOA 複合ビジネス・アプリケーション・サポート [295](#)
- SOAP Gateway [883](#)
- Sysplex Distributor [160](#)
- TCP/IP
  - 障害 [318](#)
  - KeepAlive インターバル [318](#)
- TCP/IP クライアントからの非同期コールアウト要求 [883](#)
- TCP/IP 設定 [371](#)
- TCP/IP 通信 [145](#)
- UNIX システム・サービス
  - ソケット制限 [314](#)
- XIB (出口インターフェース・ブロック)
  - エントリーのフォーマット [168](#)
  - フォーマット、IMS DB 接続 [172](#)

## IMS Connect (続き)

- XIB (出口インターフェース・ブロック) (続き)
  - XIB1 フォーマット [172](#)
- XIB1
  - 形式 [172](#)
- XIBDS
  - エントリーのフォーマット [170](#)
- XIBOD
  - エントリーのフォーマット [174](#)
- XML
  - メッセージ構造 [220](#)
  - COBOL への変換 [219](#)
  - COBOL 変換の例 [222](#)
  - XML アダプター [159](#)
  - XML コンバーター [159](#), [219](#)
  - XML 変換サポート、概要 [159](#)
  - z/OS Sysplex Distributor [160](#)
- IMS TM
  - IMS Connect
    - タイムアウト指定 [319](#)
- IMS TM Resource Adapter
  - 1 フェーズ・コミット [367](#)
  - 2 フェーズ・コミット
    - IMS Connect サポート [363](#)
- IMS Connect
  - サポートの概要 [152](#)
  - 生成されたクライアント ID [152](#)
  - 重複クライアント ID [152](#)
  - 2 フェーズ・コミット・サポート [363](#)
  - IMS Connect コールアウトのサポート [883](#)
  - IMS Connect メッセージ構造 [243](#)
  - IMS からの 外部サービスの呼び出し [881](#)
- IMS TM リソース
  - リソース・タイプ整合性
    - 使用不可化、制約の [390](#)
- IMS Universal Database リソース・アダプター
  - タイプ 2
    - WebSphere Application Server for z/OS へのインストール [8](#)
  - タイプ 2 接続
    - WebSphere Application Server for z/OS 構成の概要 [7](#)
  - タイプ 4
    - EAR ファイル、WebSphere Application Server でのインストール [7](#)
  - 1 フェーズ・コミット [359](#)
  - 2 フェーズ・コミット [356](#)
  - WebSphere Application Server for z/OS
    - アプリケーション、インストール [12](#)
    - クラスパス、設定 [10](#)
    - データ・ソース、インストール [10](#)
- IMS Universal Database リソース・アダプター、インストール [5](#)
- IMS Universal ドライバー
  - セキュリティ
    - IMS Connect [185](#)
  - 接続の構成 [3](#)
  - 1 フェーズ・コミット [359](#)
  - 2 フェーズ・コミット
    - IMS Connect サポート [356](#)
- CICS
  - CICS でのインストール [20](#)
- IMS Connect セキュリティー [185](#)
- IMS Connect

- IMS Universal ドライバー (続き)
  - IMS Connect (続き)
    - タイムアウト指定 [319](#)
    - 2 フェーズ・コミット・サポート [356](#)
  - IMS Connect サポート
    - 接続経路指定 [152](#)
  - WebSphere Application Server
    - 構成 [4](#)
- IMS アプリケーション・プログラムからのコールアウト
  - 並列 RESUME TPIPE 要求に対する OTMA T パイプのサポート [792](#)
  - T パイプ、並列 RESUME TPIPE 要求に対するサポート [792](#)
- IMS からの出力の延期 [963](#)
- IMS 間 TCP/IP 通信
  - ソケット接続
    - クリーンアップ [312](#)
    - 持続性 [312](#)
    - 終了シナリオ [312](#)
  - IMS Connect
    - タイムアウト指定 [329](#)
  - OTMA
    - 出力メッセージのフォーマット [803](#)
    - メッセージ、出力時のフォーマット [803](#)
    - IMS 間 TCP/IP 通信 [803](#)
    - OTMA サポートの概要 [801](#)
    - OTMA スーパーメンバー・サポートの概要 [802](#)
- IMS 間セッション、ISC プロトコル制約事項 [491](#)
- IMS 間通信、LU 6.1 プロトコル [489](#), [491](#)
- IMS 間の TCP/IP 接続
  - ソケット、予約 [315](#)
  - RESVSOC パラメーター
    - 使用法 [315](#)
- IMS サービス、ESAP で使用できる [126](#)
- IMS ネットワークの開始
  - 前提条件 [431](#), [957](#)
  - IMS を作動可能にする
    - 結果 [499](#), [957](#)
    - /START コマンド [499](#), [957](#)
- IMS ネットワークのシャットダウン [500](#), [965](#)
- IMS のカスタマイズ [804](#)
- IMS のリモート制御 [490](#)
- IMS への外部サブシステムの指定方法 [120](#)
- IMS メッセージ通信、応答要件 [981](#)
- IMS モニター
  - 報告書印刷プログラム [730](#)
  - MSC の考慮事項 [729](#)
- IMS を作動可能にする [499](#), [957](#)
- IMS-CICS 間の通信
  - 機能
    - 代替 [642](#)
  - 基本機能 [642](#)
  - コーディング
    - 機能管理ヘッダー [645](#)
    - システム定義オプション [629](#)
  - 処理の流れ
    - RECEIVE [624](#)
    - RETRIEVE [628](#)
    - SEND INVITE [621](#)
    - SEND LAST [623](#)
    - SEND/RECEIVE [621](#)
    - START/RETRIEVE [626](#)
  - セッション
    - 開始 [636](#)
- IMS-CICS 間の通信 (続き)
  - セッション (続き)
    - 再確立 [654](#)
    - 再同期化 [654](#)
    - 終了 [638](#)
    - 同期点 [644](#)
    - バインディング [654](#)
    - 未解決のトラフィックの処理 [655](#)
  - セッションの開始 [636](#)
  - セッションの終了 [638](#)
  - セッションの保全性 [654](#)
  - 代替機能 [642](#)
  - 同期点 [644](#)
  - トランザクション
    - サポートされる属性 [603](#), [604](#), [642](#)
    - サポートされるタイプ [603](#), [604](#), [642](#)
  - のアプリケーション・コーディング [638](#)
  - 非同期処理のフロー [626](#)
  - リカバリーおよび再始動 [652](#)
  - ATTACH パラメーター [646](#)
  - CICS テーブルの作成 [629](#)
  - CICS トランザクション、定義 [636](#)
  - IMS コマンド [643](#)
  - IMS-CICS ISC [636](#)
  - LU 6.1 リンク
    - オンライン・リソース定義 [629](#)
    - 互換性のあるノード [630](#)
    - 説明 [629](#)
    - 複数のリンク [634](#)
    - マクロ・レベルのリソース定義 [629](#)
  - MFS サポート [651](#)
  - SCHEDULER パラメーター [649](#)
- IMS.FORMAT、MFS からの出力 [436](#)
- IMS.PROCLIB [419](#)
- IMSID
  - IMSplex と MSC ネットワークが共存する時の [738](#)
- IMSplex
  - インストール [178](#)
  - 環境要件 [177](#)
  - 疑似異常終了 U0830
    - 回避 [744](#)
  - 端末管理 [388](#)
  - 定義 [383](#)
  - メッセージ経路指定
    - MSC-IMSplex 構成内 [732](#)
  - 類似性
    - MSC-IMSplex 構成内 [733](#)
  - APPC および OTMA メッセージ
    - MSC リモート・トランザクションの処理 [738](#)
  - IBM Management Console for IMS and Db2 for z/OS [177](#)
  - IMS Connect 構成ファイル [178](#)
  - IMS Connect サポート [178](#)
- IMSID
  - IMSplex と MSC ネットワークが共存する時の [738](#)
- IMSplex 内のリンク定義
  - 削除 [737](#)
- MSC
  - リモート APPC または OTMA トランザクションのバックエンド処理 [738](#)
  - APPC および OTMA リモート・トランザクション [738](#)
  - MSC (複数システム結合機能)
    - 共存 [731](#)
    - マイグレーション [734](#)

IMSplex (続き)  
MSC (複数システム結合機能) (続き)  
  MSNAME 定義と SYSID の共用 [734](#)  
MSC MSNAME 定義  
  重複 [736](#)  
  MSNAME 定義と SYSID の共用 [734](#)  
MSNAME 定義  
  削除 [737](#)  
OM アクセス [177](#)  
SCI (Structured Call Interface) [177](#)  
SYSID  
  IMSplex 内の MSC SYSID の管理 [737](#)  
  IMSplex 内の MSC SYSID 複製 [738](#)  
  MSNAME 定義と SYSID の共用 [734](#)  
IMSplex サポート [177](#)  
IMSplex 類似性経路指定 [752](#)  
IMSRSC リポジトリ  
  MSC での使用 [753](#), [754](#), [757](#), [758](#)  
INOP 状態 [407](#)  
IOPCB [778](#)  
IRM  
  IMS 要求メッセージ [237](#)  
  IRM 拡張 [237](#)  
IRM (IMS 要求メッセージ)  
  固定部分フォーマット [224](#)  
IRM\_TIMER  
  値 [320](#)  
  使用法 [320](#)  
ISC  
  サポートされる機能  
    CICS セッション [601](#)  
  CICS  
    サポートされる機能 [601](#)  
    同期処理、概要 [601](#)  
    非同期処理、概要 [601](#)  
  TCP/IP  
    サポートされる機能 [601](#)  
    同期処理、概要 [601](#)  
    非同期処理、概要 [601](#)  
ISC (システム間連絡)  
  SYSERROR [589](#)  
  SYSSTAT [589](#)  
  アプリケーション・プログラム  
    アクセス [478](#)  
    経路指定の例 [481](#), [483-488](#)  
    例 [673](#)  
    MFS (メッセージ形式サービス) の使用 [478](#)  
    MFS 不使用 [479](#)  
  エラー処理  
    送信側 ERP [550](#)  
    通知済み MTO [545](#)  
  エラー・リカバリー、例外応答後のページ [538](#)  
  エラー・リカバリー・プロシージャ  
    例、受信側検出エラー [670](#)  
    例、送信側検出エラー [669](#)  
    例 [669](#)  
  概要 [455](#)  
  拡張会話方式終了 [520](#)  
  機能 [469](#)  
  基本機能 [455](#)  
  クラス 2 端末 [459](#)  
  経路指定 [455](#)  
  経路指定の例 [481](#)  
  経路指定パラメーター [480](#)

ISC (システム間連絡) (続き)  
  交換単位コード、ATTACH FM ヘッダー [573](#)  
  構成 [460](#)  
  再同期プロシージャ  
    イン・ブラケット障害からの回復 [506](#)  
    コールド・スタート、回復セッションへの [506](#)  
    コマンド、障害のリカバリー [508](#)  
    再始動で利用可能な出力 [507](#)  
    再始動プロセス [508](#)  
    作業単位のバックアウトの制御 [506](#)  
    シーケンス番号の保守 [505](#)  
    実行モード、障害のリカバリー [508](#)  
    セッション障害、IMS 障害 [508](#)  
    セッション障害、IMS は実行中である [507](#)  
  サポートされるトランザクション・タイプ、CICS セッ  
  ション [603](#), [604](#), [642](#)  
  システム定義 (system definition)  
    マクロ・パラメーター [494](#)  
  実行モード  
    外部指定 [471](#)  
    処理モード [472](#)  
    同期対非同期 [472](#)  
    内部、CICS で [603](#), [604](#)  
    内部定義 [472](#)  
  出力プロトコル [477](#)  
  出力編集オプション [469](#)  
  制御ブロック・ストレージ、並列セッション [471](#)  
  静的定義 [491](#)  
  セキュリティ [459](#)  
  セッション制御 [455](#)  
  センス・コード  
    受信された [552](#)  
    送信された [553](#)  
  選択受信側 ERP  
    X'0864xxxx': 機能打ち切り [546](#)  
    X'0865xxxx': 機能打ち切り [547](#)  
    X'0866xxxx': 機能打ち切り [548](#)  
  対 CICS [642](#)  
  データ記述子 FM ヘッダー  
    入力で使用 [565](#)  
  データ・フロー制御  
    例 [659](#), [660](#), [664](#), [665](#), [667](#)  
  テスト・モード [470](#)  
  同期点標識 [524](#)  
  ネットワーク、開始 [499](#)  
  ネットワーク、開始 [499](#)  
  ネットワーク操作 [499](#), [500](#)  
  ノード定義  
    マクロ使用 [491](#)  
    要約 [496](#)  
    例 [492](#)  
    COMM マクロ [494](#)  
    NAME マクロ [494](#)  
    SUBPOOL マクロ [494](#)  
    TERMINAL マクロ [495](#)  
  ハーフセッション、1 次側と 2 次側 [502](#)  
  複数チェーン入力メッセージの交換制限 [573](#)  
  プロトコル  
    定義 [499](#)  
    要約 [455](#)  
    IMS 間セッションの制約事項 [491](#)  
  分散サービス、に対するサポート [458](#)  
  分散処理 [457](#)  
  編集オプション [445](#)

## ISC (システム間連絡) (続き)

## 編集機能

概要 [469](#)FM ヘッダーの呼び出し [556](#)

## メッセージ経路指定

パラメーター [479](#), [480](#)FM ヘッダー [479](#)MSC リンクを通しての [489](#)

## メッセージ再同期 (message resynchronization)

セッション [504](#)要約 [526](#)

## メッセージ通信

ATTACH FM ヘッダー [571](#)メッセージの保全性 [459](#)メッセージの保全性、CICS [653](#)

## ATTACH FM ヘッダー

形式 [571](#)ビット内容 [571](#)ATTDDBA [574](#)ATTDSP [574](#)

## CICS

アウトバウンド宛先プロセス名フィールド (ATTDPN) [646](#)アウトバウンド宛先プロセス名フィールド (SCDDPN) [649](#)異常終了、トランザクション [656](#)交換単位コード (ATTIU) [647](#)サンプル・プログラム DFSISCO0 [674](#)サンプル・プログラム、ACB 生成 [680](#)サンプル・プログラム、MFS 形式 [678](#)サンプル・プログラム、PSB 生成 [679](#)サンプル・プログラム、インストール [673](#)サンプル・プログラム、コンパイルとバインドのための JCL [676](#)サンプル・プログラム、システム定義ステートメント [677](#)セッション再始動、TCP/IP [620](#)セッション終了、TCP/IP [619](#)セッション終了、TCP/IP、無条件 [620](#)セッション終了、異常、TCP/IP [620](#)セッション終了、正規手順での、TCP/IP [620](#)セッションの開始 [637](#)セッションの数 [631](#)セッション・パラメーター [631](#)セッション名 [631](#)端末の装置依存データ [463](#)データ・ストリーム・プロファイル (ATTDSP) [647](#)トランザクションの異常終了 [656](#)ネットワーク名 [630](#)非同期トランザクション、定義 [636](#)非ブロック化アルゴリズム (ATTDDBA) [648](#)未確定処理 [636](#)戻り 1 次リソース名 (ATTRPRN) [647](#)戻り 1 次リソース名 (SCDRPRN) [650](#)戻り宛先プロセス名 (ATTRDPN) [647](#)戻り宛先プロセス名 (SCDRDPN) [649](#)1 次リソース名フィールド (ATTIPRN) [646](#)1 次リソース名フィールド (SCDPN) [649](#)ATTACC [648](#)ATTDDBA [648](#)ATTDSP [647](#)ATTDPN [646](#)ATTDQN [647](#)ATTDSP [647](#)

## ISC (システム間連絡) (続き)

## CICS (続き)

ATTIU [647](#)ATTPRN [646](#)ATTRDPN [647](#)ATTRPRN [647](#)SCDDP [650](#)SCDDPN [649](#)SCDDQN [650](#)SCDPN [649](#)SCDRDPN [649](#)SCDRPRN [650](#)TCP/IP セッション再始動 [620](#)TCP/IP セッション終了 [619](#)TCP/IP セッション終了、異常 [620](#)TCP/IP セッション終了、正規手順での [620](#)TCP/IP セッション終了、無条件 [620](#)TCP/IP セッションの再始動 [620](#)TCP/IP セッションの終了 [619](#)TCP/IP セッションの終了、正規手順での [620](#)TCP/IP セッションの終了、無条件で [620](#)CICS と IMS の機能 [463](#)

## CICS-IMS

アプリケーション関連の概念 [639](#)

## DFC 同期点

出力で [521](#)リカバリー可能メッセージ [522](#)リカバリー不能メッセージ [524](#)例外 [521](#)MFS メッセージ [522](#)

## DFC プロトコル

応答モード・エラー [519](#)応答要件 [520](#)会話モード・エラー [519](#)可用性 [521](#)出力メッセージのブラケット化 [532](#)選択受信側 ERP [545](#)チェーン [537](#)同期点、入力メッセージ [521](#)入力メッセージのバックアウト [521](#)入力メッセージのブラケット化 [529](#)ハーフセッションの同期化 [520](#)ブラケットおよび半二重 [528](#)ページ化メッセージ・エラー [544](#)例外応答 [518](#)BID コマンド [528](#)CANCEL コマンド [536](#)CHASE コマンド [538](#)LUSTATUS [542](#)LWA によって補足される回復性 [520](#)RSHUT コマンド [545](#)RTR コマンド [545](#)ETO [63](#)

## FM ヘッダー

SYSERROR [589](#)SYSSTAT [589](#)エラー・リカバリー・プロシージャ [557](#)応答 QMODEL [568](#)概要 [555](#)形式の参照 [571](#)データ記述子 [565](#)同期処理 [557](#)プロセスの開始 [557](#)メッセージの経路指定 [479](#)

## ISC (システム間連絡) (続き)

## FM ヘッダー (続き)

ATTACH [557](#)  
 ATTACH, MFS [564](#)  
 ATTDDBA [574](#)  
 ATTDSP [574](#)  
 CICS セッション [645](#)  
 DPM メッセージ、MFS [562](#)  
 IMS によってサポートされる [555](#)  
 ISC 編集の呼び出し [556](#)  
 MFS [561](#)  
 QGET [581](#)  
 QGETN [582](#)  
 QMODEL [566](#)  
 QPURGE [583](#)  
 QSTATUS [584](#)  
 QXFR [585](#)  
 RAP [588](#)  
 RAP (リセット付加プロセス) [558](#)  
 RAP (リセット付加プロセス)、MFS [569](#)  
 SCHEDULER [558](#), [588](#)  
 SCHEDULER, MFS [564](#)  
 SYSMSG [559](#)

IMS 間構成 [461](#)

## IMS 間セッション

会話方式機構の制約事項 [490](#)  
 設計の考慮事項 [489](#), [491](#)  
 単一セッションの定義 [489](#)  
 バックエンド IMS への経路指定 [490](#)  
 バッファ・サイズ [490](#)  
 プロトコル制約事項 [491](#)  
 並列セッションの定義 [489](#)  
 メッセージのデキュー [506](#)  
 リモート制御 [490](#)  
 MSC と ISC の共存 [489](#)

IMS 機能 [457](#)IMS コマンド、発行 [470](#)IMS コマンドを発行する並列セッション [470](#)IMS サポート [457](#)IMS サポート、CICS [601](#)IMS への CICS データの受け渡し [464](#)IMS/CICS 間構成 [460](#)

## ISC セッション

サポートされないコマンド [628](#)

ISC セッションのバインド [502](#)

## ISC ノード

定義 [629](#)

## ISC プロトコル

通信の設計 [477](#)

ISC 編集 ATTACH パラメーターの例 [591](#)

MFS における ATTACH および SCHEDULER パラメーターの例 [595](#)

MSC の共存 [489](#)QGET [581](#)QGETN [582](#)QPURGE [583](#)QSTATUS [584](#)QXFR [585](#)RAP [588](#)SC (セッション制御) プロトコル [500](#)

## SC プロトコル

イン・ブラケット障害からの回復 [506](#)  
 コールド・スタート・リカバリー [506](#)  
 コマンド、障害のリカバリー [508](#)

## ISC (システム間連絡) (続き)

## SC プロトコル (続き)

再始動で利用可能な出力 [507](#)  
 再始動プロシージャの設計 [505](#)  
 再始動プロセス [508](#)  
 シーケンス番号の保守 [505](#)  
 実行モード、障害のリカバリー [508](#)  
 セッション終了 [513](#), [549](#)  
 セッション障害、IMS 障害 [508](#)  
 セッション障害、IMS は実行中である [507](#)  
 セッション状態 [511](#)  
 セッションの開始 [500](#)  
 セッションの実行 [512](#)  
 セッションのバインド [502](#)  
 バインド競争の解決 [503](#)  
 保留状態の作業単位の制御 [506](#)  
 XRF、セッション開始 [501](#)

SCHEDULER [588](#)SYMSMSG における ATTACH パラメーターの例 [593](#)

## TCP/IP

正規手順でのセッション終了 [620](#)  
 セッション再始動 [620](#)  
 セッション終了 [619](#)  
 セッション終了、異常 [620](#)  
 セッション終了、正規手順での [620](#)  
 セッション終了、無条件 [620](#)  
 セッションの再始動 [620](#)  
 セッションの終了 [619](#)  
 セッションの終了、無条件で [620](#)  
 ETO [63](#)

IMS Connect、サポートの概要 [153](#)

## TCP/IP サポート

概要 [608](#)

静的端末定義 [611](#), [612](#)

制約事項 [610](#)

セキュリティ [610](#)

セッション、開始 [617](#)

セッションの開始、CICS から [618](#)

端末定義 [611](#)

動的端末定義 [611](#)

フォールバック [613](#)

要件 [609](#)

リンクの定義、CICS への [615](#)

リンクの定義、IMS Connect への [614](#)

CICS フロントエンド・トランザクション・タイプ [621](#)

TCP/IP から VTAM への切り替え [613](#)

## VTAM API

CICS [603](#), [604](#)

IMS [467](#)

VTAM 機能 [465](#)

## XRF

MNPS、セッション開始 [501](#)

USERVAR、セッション開始 [502](#)

XRF 複合システム (XRF complex) [459](#), [501](#)

ISC 拡張会話の終了 [520](#)

## ISC データ・フロー制御

例 [659](#)ISC での RU (要求単位) のチェーン [537](#)

## ISC 編集

デフォルト・エディター [469](#)

非 MFS プログラム [479](#)

ISC-CICS インストール・オプション [629](#)



## J

### Java

#### 従属領域

Db2 for z/OS のアクセス [117](#)

#### Db2 for z/OS

Java 従属領域からのアクセス [117](#)

IMS から外部 Java 環境への接続 [1](#)

IMS からの Db2 for z/OS データへのアクセス [117](#)

### Java アプリケーション・プログラム

コールアウト・サポートの概要 [883](#)

IMS からの 外部サービスの呼び出し [881](#)

### JBPs

IMS からの Db2 for z/OS データへのアクセス [117](#)

### JCL

トランザクションを保護するための [862](#)

### JDBC

#### タイプ 2

IMS Universal JDBC ドライバーおよび CICS [20](#)

IMS Universal ドライバー および CICS [20](#)

### JMPs

IMS からの Db2 for z/OS データへのアクセス [117](#)

### JOBLIB

TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン [750](#)

## L

### LINKLIST

TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン [750](#)

### Local オプション・クライアント 通信 [148](#)

### LOCK 状態 [432](#)

### LOGON MODE テーブル [428](#)

### LTERM (論理端末)

キュー [402](#)

コンポーネントの定義 [574](#)

チェーン [401](#)

定義 [395](#)

入出力装置の分離 [402](#)

ネットワーク設計 [401](#)

物理端末との関係 [401](#)

マスター端末 (master terminal) [404](#)

命名規則

必要性の除去 [942](#)

命名方法 [941](#)

リモート [697](#), [752](#)

ETO [379](#)

LTERM サブプール、ユーザー、およびコンポーネント [473](#)

### LU

#### アウトバウンド

指定 [43](#)

APPC アウトバウンド LU 指定 [43](#)

### LU (論理装置)

再割り当て [54](#)

複数、管理 [54](#)

LU 名の修飾 [53](#)

### LU 6.2

記述子 [395](#)

### LUSTATUS コマンド

応答モード・エラー [519](#)

会話モード

正常終了の要求 [520](#)

### LUSTATUS コマンド (続き)

会話モード (続き)

送信エラー [519](#)

空キュー [542](#)

機能打ち切り [543](#)

コミット (commit) [543](#)

テスト・モードの終了 [470](#)

プロトコル [542](#)

ページング・エラー [544](#)

CICS [606](#), [642](#)

NO-OP [543](#)

SLU P システム [985](#)

LUSTATUS コマンドで指示された NO-OP [543](#)

LWA (ログ先行書き込み) [520](#)

## M

### M 記述子、OTMA

限度 [800](#)

最大 [800](#)

MAXFILEPROC パラメーター、UNIX システム・サービス [314](#)

### MFS (メッセージ形式サービス)

アプリケーション・プログラム、ISC でのアクセス [478](#)

エスケープ文字、ISC でサポートされない [561](#)

オンライン・エラー検出 [563](#)

オンライン・パフォーマンス [441](#)

概要 [435](#)

管理 [440](#)

言語ユーティリティー [436](#), [443](#)

コンポーネント [436](#)

コンポーネント、概説 [442](#)

サービス・ユーティリティー [443](#)

削除文字 [447](#)

出力フォーマット設定

典型的なアプリケーション・プログラム・プロシージャ [977](#)

MOD 名 [977](#)

出力フォーマット設定、ISC [561](#)

出力メッセージ

編集セグメント [447](#)

MFS 定義の方法 [435](#)

使用可能な機能 [946](#)

制御機能

金融機関通信システム [947](#)

NEXTLP [948](#)

NEXTMSG [948](#)

NEXTMSGP [948](#)

NEXTPP [948](#)

PAGE=nn [948](#)

SLU P [947](#)

制御要求、応答要件 [982](#)

説明 [380](#)

端末キーボードのロックとアンロック [447](#)

定義 [946](#)

データ・バイト出力メッセージ [976](#)

デフォルト形式 [589](#)

同期点、ISC メッセージ [522](#)

入力セグメント [447](#)

入力フォーマット設定 [561](#)

入力メッセージ [435](#)

プール・マネージャ [443](#)

フォーマット

出力の活動化 [971](#)

MFS (メッセージ形式サービス) (続き)  
 フォーマット (続き)  
 入力の活動化 [958](#)  
 ページ削除機能、ISC でサポートされない [562](#)  
 ページング、CICS [604](#), [641](#)  
 編集 [446](#)  
 無効なページング要求 [949](#), [951](#)  
 メッセージ・エディター [443](#)  
 メッセージ・リカバリー [947](#)  
 ライブラリー、オンライン変更 [442](#)  
 利点 [441](#)  
 Bid オプション [948](#)  
 DPM [973](#)  
 DPM オプション [447](#)  
 FM ヘッダー  
 応答 QMODEL [568](#)  
 出力フォーマット設定活動化 [561](#)  
 使用可能な機能 [561](#)  
 タイプ [561](#)  
 データ記述子 [565](#)  
 入力フォーマット設定の活動化 [561](#)  
 編集 [561](#)  
 ATTACH [564](#)  
 DPM メッセージ [562](#)  
 MID 名の指定 [561](#)  
 MOD 名 [561](#)  
 QGET [568](#)  
 QGETN [567](#)  
 QMODEL [566](#), [651](#)  
 QPURGE [568](#)  
 QSTATUS [569](#)  
 QXFR [568](#)  
 RAP (リセット付加プロセス)、MFS [569](#)  
 SCHEDULER [564](#)  
 ISC アプリケーション・プログラムのバイパス [478](#)  
 ISC メッセージの編集 [469](#)  
 MFS のバイパス [446](#)  
 MID/MOD チェーン [946](#)  
 MSC (複数システム結合機能) [705](#)  
 SDF II [443](#)  
 SLU P  
 ステーション別、可用性 [947](#)  
 利点 [947](#)  
 MFS サービス・ユーティリティー [439](#)  
 MFS 装置特性テーブル  
 項目 [393](#)  
 使用方法 [78](#)  
 非 SNA 3270 装置での使用 [79](#)  
 MFS バイパス・オプションの効果 [436](#)  
 MFS 分散表示管理 (DPM) [469](#)  
 MFSTEST プロシージャ [443](#)  
 MID (メッセージ入力記述子) [442](#)  
 MID/MOD チェーン [946](#)  
 MNPS (マルチノード持続セッション)  
 XRF [417](#)  
 MOD (メッセージ出力記述子)  
 名前の指定 [561](#), [977](#)  
 目的 [442](#)  
 MODETBL  
 デフォルト LOGON MODE ID の指定 [428](#)  
 MODETBL= キーワード  
 デフォルトのオーバーライド [467](#)  
 ISC ログオン中の使用 [467](#)  
 MSASSIGN (/MSASSIGN) コマンド

MSASSIGN (/MSASSIGN) コマンド (続き)  
 物理リンクから論理リンクへの割り当て [685](#)

MSC  
 メッセージ経路指定 [732](#)

MSC (複数システム結合機能)  
 宛先システム (destination system) [692](#)  
 概念 [683](#)  
 概要 [380](#), [683](#)  
 管理  
 APPC [701](#)  
 疑似異常終了 U0830  
 回避 [744](#)  
 キューイング要約報告書 [730](#)  
 共用キュー  
 MSNAME 重複 [736](#)  
 MSNAME 定義と SYSID の共用 [734](#)  
 共用キューを使用する IMSplex  
 共存 [731](#)  
 共用キューを使用する IMSplex 内での MSNAME 重複 [736](#)  
 最適なリンク・タイプ [724](#)  
 システム ID (SYSID) [693](#)  
 システム定義 [705](#)  
 システム定義検査 [721](#)  
 使用可能化  
 システム定義時 [709](#)  
 使用不可化 [710](#)  
 セキュリティー検査  
 中間 IMS [722](#)  
 セキュリティーの考慮事項 [722](#)  
 設計の考慮事項 [701](#)  
 操作プロシージャ [722](#)  
 帯域幅、リンク [702](#)  
 帯域幅モード、容量 [725](#)  
 逐次トランザクション処理 [718](#)  
 中間システム (intermediate system) [692](#)  
 直接経路指定、プログラム間通信 [700](#)  
 定義 [683](#)  
 データの区分化 [683](#)  
 データ・フロー [688](#)  
 統計  
 最適なリンク・バッファ・サイズの評価 [725](#)  
 統計、論理リンクの [723](#)  
 トランザクション・コードの定義 [705](#)  
 名前検査のためのユーティリティー [720](#)  
 入力システム [692](#)  
 ネットワークおよび経路指定の概要 [688](#)  
 バッファ  
 サイズ、リンク統計を使用した最適化 [725](#)  
 帯域幅モードのサイズ [725](#)  
 帯域幅モードのフォーマット [725](#)  
 非帯域幅モードのサイズ [727](#)  
 非帯域幅モードのフォーマット [727](#)  
 バッファ・サイズ [712](#)  
 バッファ・サイズ、リンク [702](#)  
 パフォーマンス [723](#), [729](#)  
 汎用リソース  
 類似性、IMS Connect 内のクリア [748](#)  
 類似性、TCP/IP リンクのクリア [747](#)  
 類似性管理、TCP/IP リンク [746](#)  
 類似性の持続性、TCP/IP リンク [747](#)  
 TCP/IP [745](#)  
 汎用リソース、VTAM [750](#)  
 標準アプリケーション・プログラム [36](#)



## MSC (複数システム結合機能) (続き)

- 物理リンク
  - タイプ [711](#)
  - 定義 [711](#), [712](#)
  - CTC、定義 [713](#)
  - IMSplex からの削除 [737](#)
  - MTM、定義 [713](#)
  - TCP/IP、定義 [714](#)
  - VTAM、定義 [715](#)
- メッセージ経路指定
  - MSC-IMSplex 構成内 [732](#)
- モニターとチューニング [729](#)
- 優先順位の定義 [716](#)
- リカバリー可能トランザクションとリカバリー不能トランザクション [47](#)
- リソース消費の最小化 [701](#)
- リモート・システム
  - 定義 [688](#)
- リンク
  - 物理 [683](#)
  - 論理 [685](#)
- リンク帯域幅 [702](#)
- リンク・タイプ、最適 [724](#)
- リンク定義
  - IMSplex 内 [734](#)
- リンク統計
  - 最適なリンク・バッファ・サイズの評価 [725](#)
- リンクのチューニング [723](#)
- リンク・バッファ・サイズ [702](#)
- 類似性
  - MSC-IMSplex 構成内 [733](#)
- 類似性、クリア
  - IMS Connect [748](#)
  - TCP/IP 汎用リソース [747](#)
- 類似性管理
  - TCP/IP 汎用リソース [746](#)
- 類似性の持続性
  - TCP/IP 汎用リソース [747](#)
- ルーティング・パス (routing path) [689](#)
- ローカル・システム
  - 定義 [688](#)
- 論理宛先 [690](#)
- 論理リンク
  - 高位値の統計 [728](#)
  - 統計、高位値 [728](#)
  - 統計、リセット [724](#)
  - 統計およびバッファ・サイズ [724](#)
  - 統計およびリンク容量 [725](#)
  - 統計のリセット [724](#)
  - バッファ・サイズおよび統計 [724](#)
  - ベンチマーク [723](#)
  - ベンチマークの統計 [723](#)
  - 容量および統計 [725](#)
  - IMSplex からの削除 [737](#)
- 論理リンク、SDLC リンク [685](#)
- 論理リンク統計 [723](#)
- 論理リンク統計、高位値 [728](#)
- 論理リンク統計およびバッファ・サイズ [724](#)
- 論理リンク統計のリセット [724](#)
- 論理リンク・パス [686](#)
- 論理リンク・パフォーマンスのベンチマーク [723](#)
- 論理リンク容量 [725](#)
- APPC および OTMA メッセージ
  - リモート処理 [738](#)

## MSC (複数システム結合機能) (続き)

- APPC におけるトランザクションの回復 [47](#)
- APPC/IMS
  - アプリケーション・プログラムの障害およびトランザクションの回復 [50](#)
  - 中間 IMS 障害およびトランザクションの回復 [50](#)
  - トランザクションの障害点 [48](#)
  - リモート IMS 障害およびトランザクションの回復 [50](#)
  - ローカル IMS 障害およびトランザクションの回復 [49](#)
  - ローカル・トランザクションの廃棄可能性 [48](#)
  - LU 6.2 セッションおよびトランザクションの回復 [48](#)
  - LU 6.2 リカバリー可能性フロー [51](#)
  - MSC リンクおよびトランザクションの回復 [49](#)
- APPC/IMS に対するサポート [46](#)
- DRD に対する使用可能化 [706](#)
- IMS Connect
  - 汎用リソース [157](#)
- IMS 間 TCP/IP 通信
  - IMS Connect の MSC のサポート [156](#)
- IMSID
  - IMSplex 内 [738](#)
- IMSplex
  - リモート APPC および OTMA トランザクションのバックエンド処理の有効化 [743](#)
  - リモート APPC または OTMA トランザクションのバックエンド処理 [738](#)
  - APPC トランザクション、IMSplex 外部の MSC システムに送信 [744](#)
  - APPC および OTMA リモート・トランザクション [738](#)
  - IMSplex と MSC が共存する場合のメッセージ経路指定 [733](#)
  - IMSplex 内のメッセージ経路指定 [733](#)
  - MSNAME 定義と SYSID の共用 [734](#)
  - OTMA トランザクション、IMSplex 外部の MSC システムに送信 [744](#)
- IMSplex 内の MSNAME 定義
  - 削除 [737](#)
- IMSplex 内のリンク定義
  - 削除 [737](#)
- IMSplex へのマイグレーション [734](#)
- IMSRSC リポジトリに対する使用可能化 [707](#)
- IMSRSC リポジトリの使用 [753](#), [754](#), [757](#), [758](#)
- ISC 機能 [456](#)
- ISC と共存 [489](#)
- ISC と比較 [456](#)
- ISC と比較する機能 [456](#)
- LU 6.2 アプリケーション・トランザクション・バッファ・サイズ [712](#)
- MSC 会話障害 [705](#)
- MSNAME 定義
  - IMSplex 内での共用 [734](#)
- SPA [704](#)
- SYSID
  - IMSplex 内での共用 [734](#)
  - IMSplex 内の管理 [737](#)
  - IMSplex 内の複製 [738](#)
- SYSID (システム ID) [693](#)
- SYSID テーブル
  - IMSplex 内の MSNAME 定義の削除 [737](#)
- SYSID の定義 [709](#)

MSC (複数システム結合機能) (続き)  
TCP/IP  
汎用リソース [745](#)  
TCP/IP 汎用リソース  
類似性、IMS Connect 内のクリア [748](#)  
類似性、クリア [747](#)  
類似性管理 [746](#)  
類似性の持続性 [747](#)  
VTAM 汎用リソース [750](#)  
/MSVERIFY コマンド [721](#)  
MSC (複数システム結合機能) 内の無効な宛先 [700](#)  
MSC (複数システム結合機能) におけるリソース要求の平衡化 [702](#)  
MSC (複数システム結合機能) の会話型処理  
異常終了 (abend) [705](#)  
の設計 [703](#)  
リモート宛先検査 [704](#)  
MSC (複数システム結合機能) の水平区分化 [683](#)  
MSC LU 6.2 の BUFSIZE パラメーター [712](#)  
MSC におけるリンク受信経路指定出口ルーチン [751](#)  
MSC プログラム経路指定出口ルーチン [752](#)  
MSC 用パフォーマンス情報の調整 [729](#)  
MSC=  
MSC の使用可能化 [708](#)  
MSC の使用不可化 [710](#)  
MSDB  
OTMA、読み取り専用アクセス [814](#)  
MSLINK 定義  
IMSplex からの削除 [737](#)  
MSLINK マクロ  
定義 [715](#)  
MSNAME 定義  
共用キュー・グループ内の  
重複 [736](#)  
IMSplex からの削除 [737](#)  
IMSplex 内のシステム間での共用 [734](#)  
MSNAME マクロ  
報告書  
MSC 用 [730](#)  
論理パス名 [716](#)  
論理リンク・パス [686](#)  
MSC (複数システム結合機能) 直接経路指定 [699](#)  
SYSID キーワード [716](#)  
MSPLINK 定義  
IMSplex からの削除 [737](#)  
MSPLINK マクロ  
物理リンクの定義  
バッファ・サイズ [712](#)  
CTC 物理リンクの定義 [713](#)  
MTM 物理リンクの定義 [713](#)  
TCP/IP 物理リンクの定義 [714](#)  
VTAM 物理リンクの定義 [715](#)  
MTM (メモリー間)  
MSC (複数システム結合機能) 物理リンク・タイプ [683](#)  
MSC 物理リンク  
定義 [713](#)  
MTO (マスター端末オペレーター)  
セッションの拒否の通知 [512](#)  
端末再始動 [446](#)  
論理リンクの割り当て [683](#)  
ISC エラー [546](#)  
/OPNDST コマンド [428](#)  
MULT1 パラメーター [477](#), [533](#)  
MULT2 パラメーター [478](#), [533](#)

MULTIRTP  
コールアウト・メッセージ [887](#)  
MULTIRTP パラメーター  
スーパーメンバー [793](#), [794](#)  
**N**  
NAME マクロ  
コンポーネントを関連付けるのに使用される [474](#)  
リモート論理端末の [686](#)  
ISC セッションとしての定義 [494](#)  
MSNAME マクロ [719](#)  
NCP (ネットワーク制御プログラム) [377](#), [378](#)  
NONE セキュリティー・レベル [830](#)  
NOSCAN オプション [973](#)  
NRESTART コマンド [450](#)  
NTO 端末 [404](#)  
**O**  
OASN (起点アプリケーション・シーケンス番号) [141](#)  
ODBA (Open Database Access)  
アプリケーション・プログラム、バインディング [768](#)  
モジュール  
JOBLIB にどちらを置くか [768](#)  
ODBA (オープン・データベース・アクセス)  
インターフェース [767](#)  
概要 [767](#)  
セキュリティ  
APSB [770](#)  
RAS [769](#)  
セキュリティの定義 [768](#)  
接続  
構成 [761](#)  
セットアップ [767](#)  
モジュール  
STEPLIB にどちらを置くか [768](#)  
APSB セキュリティー [770](#)  
IMS データベースへのアクセス [767](#)  
RAS セキュリティー [769](#)  
Open Database Access (ODBA)  
アプリケーション・プログラム、バインディング [768](#)  
モジュール  
STEPLIB にどちらを置くか [768](#)  
Open Transaction Manager Access  
サンプル・メッセージ [919](#)  
ヘッダー [261](#)  
呼び出し可能インターフェース (C/I)  
制約事項 [924](#)  
セキュリティ [924](#)  
の紹介 [921](#)  
Open Transaction Manager Access (IMS Open Transaction  
Manager Access)  
呼び出し可能インターフェース (C/I)  
初期設定 [923](#)  
Open Transaction Manager Access (OTMA)  
宛先記述子  
限度 [800](#)  
インストール [789](#)  
応答フラグ  
遅延確認応答のサポート [892](#)  
開始  
OTMA の開始時機の定義 [791](#)

Open Transaction Manager Access (OTMA) (続き)

概要 [773](#)  
会話型トランザクション  
終了 [822](#)  
カップリング・ファシリティ  
システム間 [773](#)  
カップリング・ファシリティ (coupling facility)  
XCF [773](#)  
管理 [789](#)  
記述子  
限度 [800](#)  
最大 [800](#)  
記述子、指定 [795](#)  
機能 [774](#)  
クライアント  
保護トランザクションの開始 [862](#)  
クライアント記述子  
限度 [800](#)  
DFSOTMA [800](#)  
構成 [789](#)  
構成パラメーター [789](#)  
コールアウト、非同期  
応答の相関 [882](#)  
コールアウト構成の概要 [877](#)  
コールアウト要求 [877](#)  
コミット後送信出力  
確認応答のタイムアウト [874](#)  
サーバー定義 [774](#)  
サポートされる IMS 環境 [774](#)  
システム定義  
セキュリティ [831](#)  
始動プロシージャ [789](#)  
使用可能化 [789](#)  
スーパーメンバー  
並列 RESUME TPIPE サポート、変更 [794](#)  
並列 RESUME TPIPE 要求のサポート [793](#)  
MULTIRTP [793](#)  
MULTIRTP 設定、変更 [794](#)  
RESUME TPIPE 要求、複数のアクティブな [793](#)  
制約事項および要件 [814](#)  
セキュリティ  
一般的考慮事項 [840](#)  
検査レベル、定義 [791](#)  
システム定義 [831](#)  
非同期保留キュー [838](#)  
OTMA クライアント・セキュリティの指定 [832](#)  
OTMA メンバー・セキュリティの指定 [832](#)  
セキュリティ、RACF [830](#)  
デフォルトのクライアント値、設定 [800](#)  
同期コールアウトのサポート  
MULTIRTP [887](#)  
RESUME TPIPE 要求の並列処理 [887](#)  
トランザクション有効期限  
概要 [827](#)  
秒単位 [829](#)  
STCK 形式 [828](#)  
ネットワーク・セキュリティ資格情報の伝搬 [836](#)  
パラメーター  
GRNAME [790](#)  
OTMA= [791](#)  
OTMAASY [795](#)  
OTMAMD [795](#)  
OTMANM [790](#)  
OTMASE [791](#)

Open Transaction Manager Access (OTMA) (続き)

パラメーター (続き)  
OTMASP [794](#)  
非同期保留キュー  
セキュリティ [838](#)  
プログラム間通信  
非同期出力、指定 [795](#)  
プロトコル  
概要 [777](#)  
分散セキュリティ資格情報の伝搬 [836](#)  
保管域接頭部 (SAP)  
制限 [823](#)  
割り振り [823](#)  
メッセージ (message)  
接頭部の長さ [814](#)  
メッセージ接頭語  
セキュリティ・データ・フィールド、説明 [916](#)  
メッセージ制御フィールド、説明 [892](#)  
ユーザー・データ・フィールド、説明 [918](#)  
メンバー記述子  
限度 [800](#)  
DFSOTMA [800](#)  
メンバー名のオーバーライド、使用可能化 [795](#)  
利点 [775](#)  
ACEE、リフレッシュ [834](#)  
callout  
非同期コールアウト・サポートの実装 [881](#)  
CM0 確認応答のタイムアウト [874](#)  
CM0 出力  
確認応答のタイムアウト [874](#)  
Cross-System Coupling Facility [773](#)  
D 記述子  
限度 [800](#)  
DFSOTMA 記述子 [800](#)  
GRNAME [790](#)  
IBM MQ  
同期化された T パイプ、定義 [794](#)  
IMS 間 TCP/IP 通信のメッセージ・フロー [780](#)  
IMS 間 TCP/IP 通信フロー [780](#)  
IMS 管理 [807](#)  
M 記述子  
限度 [800](#)  
DFSOTMA [800](#)  
MSC (複数システム結合機能)  
IMSplex 内のリモート・トランザクション処理 [738](#)  
MULTIRTP  
使用可能化 [793](#)  
スーパーメンバー [793](#)  
スーパーメンバー・サポート、変更 [794](#)  
OTMA C/I  
OTMA C/I セッションのタイムアウト [924](#)  
OTMA 宛先記述子  
宛先名のマスキング [799](#)  
定義 [799](#)  
OTMA クライアント、概要 [847](#)  
OTMA クライアント記述子 [796](#)  
OTMA= [791](#)  
OTMAASY [795](#)  
OTMAMD [795](#)  
OTMANM [790](#)  
OTMASE [791](#)  
OTMASP [794](#)  
PROCLIB メンバー DFSPBxxx [789](#)  
T パイプ

## Open Transaction Manager Access (OTMA) (続き)

### T パイプ (続き)

- 並列 RESUME TPIPE 要求、サポート [792](#)
- MULTIRTP [792](#)
- MULTIRTP、使用可能化 [793](#)
- RESUME TPIPE 要求、複数のアクティブ [792](#)
- RESUME TPIPE 要求の並列処理、サポートの使用可能化 [793](#)

### XCF

- グループ名、定義 [790](#)
- メンバー名、定義 [790](#)

### OPNDST (/OPNDST) コマンド

- ISC TCP/IP セッション [617](#)
- MODETBL キーワードとの関連 [428](#)

### OPNDST コマンド

- 使用の結果 [102](#)
- ログオン [94](#)

### OPTACK オプション

- 金融機関通信システム [981](#)
- 高速機能 (Fast Path) [953](#)
- SLU P [951](#), [979](#)

### OPTIONS= DPAGE または PPAGE

- 出力メッセージ (output message) [585](#)
- MFS DPM [562](#), [973](#)

## OTMA

### 会話、IMS の管理 [807](#)

#### 共用キュー

- ALTPCB 出力、リトリブ [812](#)

#### クライアント

- 経路指定 [777](#)
- 定義 [847](#)

#### クライアント・ビッドのメッセージ・フロー [858](#)

#### 高速 PCB およびプログラム間通信 [843](#)

#### コールアウト要求

- IMS Connect [331](#)

#### コマンド

- 設計済み出力 [777](#)
- IMS [850](#)
- OTMA の用語 [850](#)

#### コミット (commit)

- 処理の要約 [851](#)

#### コミット後送信出力、管理 [872](#)

#### コミット後送信出力、スーパーメンバー [875](#)

#### コミット後送信出力、転送 [873](#)

#### コミット後送信出力、ページ [873](#)

#### コミット後送信メッセージ・フロー [860](#)

#### サーバー使用可能メッセージ・フロー [859](#)

#### サーバー状態プロトコル・コマンド [909](#)

#### 再始動 処理 [808](#)

#### 再同期

- 概要 [863](#)
- 前提事項 [864](#)

#### サポートされていない DBCTL [814](#)

#### サポートされていない端末管理コマンド [814](#)

#### サンプル・メッセージ [919](#)

#### シャットダウン

- 概要 [813](#)
- クライアント通知 [813](#)

#### 終了

- 概要 [813](#)
- クライアント通知 [813](#)

#### 出力の転送

- 送信専用出力 [874](#)
- NAK 応答 [874](#)

## OTMA (続き)

### 出力のページ

- NAK 応答 [873](#)

### 状態データ

- すべての T パイプからの出力再開のフォーマット [912](#)

- T パイプに対する出力再開のフォーマット [911](#)

### 状態データ (state data)

- REQresynch コマンドのフォーマット [902](#)

### スーパーメンバー

- IMS 間 TCP/IP 通信 [158](#), [802](#)

### セキュリティ

- 指定 [831](#)
- 非同期保留キュー [838](#)
- メッセージ接頭語でのセキュリティの指定 [835](#)
- DFSVRTUX [839](#)
- IMS 間 TCP/IP 接続 [839](#)
- OTMA RESUME TPIPE セキュリティ 出口ルーチン [839](#)

### 設計済みトランザクション属性 [927](#)

#### 接頭部

- 内容 [858-860](#)

#### 送信専用

- 出力の転送 [874](#)

#### タイムアウト

- CM1 トランザクション [827](#)

#### タイムアウト、ACK [825](#)

#### タイムアウト間隔

- OTMA クライアント用の指定 [826](#)

#### 同期プログラム間通信 [879](#)

#### トランザクション・コード、指定

- IMS 間 TCP/IP 通信 [803](#)

#### トランザクション属性セグメント [927](#)

#### バッファ・プール (buffer pool)

- HIOP [824](#)

- LUMP [824](#)

#### パフォーマンス

- リソース・モニター [815](#)

#### 非同期出力

- IMS Connect [331](#)

#### 非同期保留キュー [876](#)

#### プログラム間通信

- 使用のシナリオ [842](#)

#### プロトコル・コマンド

- 「T パイプの保留キュー上にメッセージがない」のフォーマット [914](#)

- 概要 [815](#)

- サーバー状態 [909](#)

- RESUME TPIPE 要求取り消しのフォーマット [913](#)

#### フロントエンド・スイッチ、サポートされていない [814](#)

#### 保護会話メッセージ、制約事項 [814](#)

#### メッセージ

- クライアント・ビッド・フロー [858](#)

- 現行ワークロードの表示 [816](#)

- コミット後送信フロー [860](#)

- サーバー使用可能フロー [859](#)

- サンプル・フロー [857](#)

- 接頭部、内容 [858-860](#)

- フロー、クライアント・ビッド・メッセージ [858](#)

- フロー、コミット後送信メッセージ [860](#)

- フロー、サーバー使用可能メッセージ [859](#)

#### メッセージ (message)

- 回復可能 [864](#)

- キュー・データ・セット・サイズ [824](#)

## OTMA (続き)

- メッセージ (message) (続き)
  - IMS 再同期サポート [864](#)
- メッセージ暗号化、制約事項 [814](#)
- メッセージ形式サービス、サポートされていない [814](#)
- メッセージ接頭語
  - 応答フラグ (TMAMCRSI) [893](#)
  - コマンド・タイプ (TMAMCTYP) [894](#)
  - コミット確認フラグ (TMAMCCCI) [893](#)
  - 処理フラグ (TMAMCPFG) [895](#)
  - 接頭部フラグ (TMAMCPFL) [897](#)
  - センス・コード (TMAMCSNC) [898](#)
  - センス・コード (TMAMCSNS) [898](#)
  - 送信シーケンス番号 (TMAMCSSN) [897](#)
  - チェーン・フラグ (TMAMCCHN) [897](#)
  - メッセージ・タイプ (TMAMCMGT) [892](#)
  - ユーザー ID エージング値 (TMAMAGNG) [898](#)
  - 理由コード (TMAMCRSC) [898](#)
  - T パイプ名 (TMAMCTNM) [897](#)
  - TMAMAGNG (ユーザー ID エージング値) [898](#)
  - TMAMCCCI (コミット確認フラグ) [893](#)
  - TMAMCCHN (チェーン・フラグ) [897](#)
  - TMAMCMGT (メッセージ・タイプ) [892](#)
  - TMAMCPFG (処理フラグ) [895](#)
  - TMAMCPFL (接頭部フラグ) [897](#)
  - TMAMCRSC (理由コード) [898](#)
  - TMAMCRSI (応答フラグ) [893](#)
  - TMAMCSNC (センス・コード) [898](#)
  - TMAMCSNS (センス・コード) [898](#)
  - TMAMCSSN (送信シーケンス番号) [897](#)
  - TMAMCTNM (T パイプ名) [897](#)
  - TMAMCTYP (コマンド・タイプ) [894](#)
- メッセージ接頭語 (message prefix)
  - 「T パイプの保留キュー上にメッセージがない」のフォーマット [914](#)
  - 概要 [848](#)
  - RESUME TPIPE 要求取り消しのフォーマット [913](#)
- メッセージ接頭語、アプリケーション・データ・セクション [918](#)
- メッセージ接頭語、セキュリティ・データ・セクション [914](#)
- メッセージ接頭語、ユーザー・データ・セクション [917](#)
- メッセージ接頭語のアプリケーション・データ・セクション [918](#)
- メッセージ接頭語のセキュリティ・データ・セクション [914](#)
- メッセージ接頭語のユーザー・データ・セクション [917](#)
- メッセージ・フォーマット
  - IMS 間 TCP/IP 通信 [803](#)
- 呼び出し可能インターフェース (C/I) [921](#)
- リカバリー可能メッセージ [864](#)
- リソース
  - モニター [815](#)
  - リソース・モニター [815](#)
  - リソース・リカバリー・サービス [814](#)
- ACEE フラッドニング制御 [818](#)
- callout
  - IBM MQ [884](#)
  - SOAP Gateway [883](#)
- CM0 出力、管理 [872](#)
- CM0 出力、スーパーメンバー [875](#)
- CM0 出力、ページ [873](#)
- CM1 ACK タイムアウト [827](#)
- CM1 トランザクション

## OTMA (続き)

- CM1 トランザクション (続き)
  - タイムアウト [827](#)
- DL/I 呼び出し
  - CHNG [841](#)
  - INQY [841](#)
  - PURG [841](#)
  - SETO [841](#)
- I/O PCB およびプログラム間通信 [843](#)
- IMS Connect
  - タイムアウト指定 [319](#)
  - OTMA ACEE エージング値の指定 [207](#)
- IMS 間 TCP/IP 通信
  - 自動的に再接続 [159](#)
  - 出力メッセージのフォーマット [803](#)
  - スーパーメンバー [158](#)
  - スーパーメンバー・サポートの概要 [802](#)
  - トランザクション・コード、指定 [803](#)
  - メッセージ、出力時のフォーマット [803](#)
  - IMS Connect の OTMA のサポート [158](#)
  - OTMA サポートの概要 [801](#)
- IMS 再始動処理
  - IMS 間 TCP/IP メッセージ [813](#)
  - T パイプ WAIT\_R 状況 [813](#)
- IMS の会話 [807](#)
- LTERM [783](#)
- MSDB、読み取り専用アクセス [814](#)
- RACF セキュリティー・クラス [836](#)
- SLU 2 トランザクション・フロー標準 [781](#)
- TMAMAGNG [898](#)
- TMAMCCCI [893](#)
- TMAMCCHN [897](#)
- TMAMCMGT [892](#)
- TMAMCPFG [895](#)
- TMAMCPFL [897](#)
- TMAMCRSC [898](#)
- TMAMCRSI [893](#)
- TMAMCRSQ [898](#)
- TMAMCSEQ [898](#)
- TMAMCSNC [898](#)
- TMAMCSNS [898](#)
- TMAMCSSN [897](#)
- TMAMCTNM [897](#)
- TMAMCTYP [894](#)
- TMAMRTID [898](#)
- XCF マクロ、再アセンブル [814](#)
- /DISPLAY TRANSACTION
  - 出力形式 [927](#)
- OTMA C/I
  - OTMA C/I セッションのタイムアウト [924](#)
- OTMA (Open Transaction Manager Access)
  - 宛先解決出口ルーチン (DFSYPX0) [778](#)
  - 宛先記述子
    - 限度 [800](#)
  - インストール [789](#)
  - 応答フラグ
    - 遅延確認応答のサポート [892](#)
  - 開始
    - OTMA の開始時機の定義 [791](#)
  - 概要 [773](#)
  - 会話型トランザクション
    - 終了 [822](#)
  - カップリング・ファシリティー



## OTMA (Open Transaction Manager Access) (続き)

- カップリング・ファシリティ (続き)
  - システム間 [773](#)
- カップリング・ファシリティ (coupling facility)
  - XCF [773](#)
- 管理 [789](#)
- 記述子
  - 限度 [800](#)
  - 最大 [800](#)
- 記述子、指定 [795](#)
- 機能 [774](#)
- クライアント
  - 保護トランザクションの開始 [862](#)
  - XRF 環境 [808](#)
- クライアント記述子
  - 限度 [800](#)
  - DFSOTMA [800](#)
- 構成 [789](#)
- 構成パラメーター [789](#)
- コールアウト、非同期
  - 応答の相関 [882](#)
- コールアウト構成の概要 [877](#)
- コールアウト要求 [877](#)
- コミット後送信出力
  - 確認応答のタイムアウト [874](#)
- サーバー定義 [774](#)
- 再同期プロトコル [866](#)
- サポートされる IMS 環境 [774](#)
- システム定義
  - セキュリティ [831](#)
- 始動プロシージャ [789](#)
- 使用可能化 [789](#)
- スーパーメンバー
  - 並列 RESUME TPIPE サポート、変更 [794](#)
  - 並列 RESUME TPIPE 要求のサポート [793](#)
  - MULTIRTP [793](#)
  - MULTIRTP 設定、変更 [794](#)
  - RESUME TPIPE 要求、複数のアクティブな [793](#)
- 制約事項および要件 [814](#)
- セキュリティ
  - 一般的考慮事項 [840](#)
  - 検査レベル、定義 [791](#)
  - OTMA クライアント・セキュリティの指定 [832](#)
  - OTMA メンバー・セキュリティの指定 [832](#)
- セキュリティ、RACF [830](#)
- デフォルトのクライアント値、設定 [800](#)
- 同期コールアウト要求
  - MULTIRTP [887](#)
  - RESUME TPIPE 要求の並列処理 [887](#)
- トランザクション有効期限
  - 概要 [827](#)
  - 秒単位 [829](#)
  - STCK 形式 [828](#)
- ネットワーク・セキュリティ資格情報の伝搬 [836](#)
- パラメーター
  - GRNAME [790](#)
  - OTMA= [791](#)
  - OTMAASY [795](#)
  - OTMAMD [795](#)
  - OTMANM [790](#)
  - OTMASE [791](#)
  - OTMASP [794](#)
- 非同期保留キュー
  - セキュリティ [838](#)

## OTMA (Open Transaction Manager Access) (続き)

- プログラム間通信
  - 非同期出力、指定 [795](#)
- プロトコル
  - 概要 [777](#)
- 分散セキュリティ資格情報の伝搬 [836](#)
- 保管域接頭部 (SAP)
  - 制限 [823](#)
  - 割り振り [823](#)
- 保護メッセージ [845](#)
- メッセージ (message)
  - 接頭部の長さ [814](#)
- メッセージ接頭語
  - セキュリティ・データ・フィールド、説明 [916](#)
  - メッセージ制御フィールド、説明 [892](#)
  - ユーザー・データ・フィールド、説明 [918](#)
- メッセージ接頭語のメッセージ制御情報セクション [889](#)
- メッセージ通信 [842](#)
- メンバー記述子
  - 限度 [800](#)
  - DFSOTMA [800](#)
- メンバー名のオーバーライド、使用可能化 [795](#)
- 利点 [775](#)
- ACEE、リフレッシュ [834](#)
- callout
  - 非同期コールアウト・サポートの実装 [881](#)
- CMO 確認応答のタイムアウト [874](#)
- CMO 出力
  - 確認応答のタイムアウト [874](#)
- Cross-System Coupling Facility [773](#)
- D 記述子
  - 限度 [800](#)
- DFSOTMA 記述子 [800](#)
- GRNAME [790](#)
- IBM MQ
  - 同期化された T パイプ、定義 [794](#)
- IMS アプリケーション・プログラム [774](#)
- IMS 間 TCP/IP 通信のメッセージ・フロー [780](#)
- IMS 間 TCP/IP 通信フロー [780](#)
- IMS 管理 [807](#)
- IOPCB [778](#)
- M 記述子
  - 限度 [800](#)
  - DFSOTMA [800](#)
- MSC (複数システム結合機能)
  - IMSplex 内のリモート・トランザクション処理 [738](#)
- MULTIRTP
  - 使用可能化 [793](#)
  - スーパーメンバー [793](#)
  - スーパーメンバー・サポート、変更 [794](#)
- OIM TCB [823](#)
- OTMA C/I
  - OTMA C/I セッションのタイムアウト [924](#)
- OTMA 宛先記述子
  - 宛先名のマスキング [799](#)
  - 定義 [799](#)
- OTMA クライアント、概要 [847](#)
- OTMA クライアント記述子 [796](#)
- OTMA= [791](#)
- OTMAASY [795](#)
- OTMAMD [795](#)
- OTMANM [790](#)
- OTMASE [791](#)
- OTMASP [794](#)

OTMA (Open Transaction Manager Access) (続き)  
PROCLIB メンバー DFSPBxxx [789](#)  
T パイプ  
並列 RESUME TPIPE 要求、サポート [792](#)  
MULTIRTP [792](#)  
MULTIRTP、使用可能化 [793](#)  
RESUME TPIPE 要求、複数のアクティブ [792](#)  
RESUME TPIPE 要求の並列処理、サポートの使用可能化 [793](#)  
XCF  
グループ名、定義 [790](#)  
メンバー名、定義 [790](#)  
/START OTMA [791](#)  
OTMA RESUME TPIPE セキュリティー 出口ルーチン (DFSRTUX) [804](#)  
OTMA 宛先解決出口ルーチン (DFSYPXO)  
複数の IMS システム [738](#)  
IMSplex [738](#)  
MSC [738](#)  
OTMA 宛先記述子 [797](#)  
OTMA 管理  
ターゲット・メンバー [819](#)  
T メンバー [819](#)  
OTMA 記述子  
限度 [800](#)  
最大 [800](#)  
OTMA クライアント  
命名規則 [848](#)  
OTMA クライアント 記述子 [796](#)  
OTMA メッセージ接頭語のアプリケーション・データ・セクション [918](#)  
OTMA メッセージ接頭語のセキュリティー・データ・セクション [914](#)  
OTMA メッセージ接頭語のユーザー・データ・セクション [917](#)  
OTMA メッセージのリカバリー、選択 [809](#)  
OTMA= パラメーター [791](#)  
OTMAASY オプション [843](#)  
OTMAASY パラメーター [795](#)  
OTMAMD パラメーター [789](#), [795](#)  
OTMANM パラメーター [789](#), [790](#)  
OTMASE パラメーター [789](#), [791](#)  
OTMASP パラメーター [789](#), [794](#)

## P

PCB (プログラム連絡ブロック)  
代替 PCB [474](#)  
入出力 PCB がない LTERM [941](#)  
I/O PCB [474](#)  
PDSE  
外部サブシステム・モジュール・テーブル [127](#)  
ping  
IMS Connect の可用性 [352](#)  
PING 応答  
形式 [255](#)  
PL/I  
XML 変換サポート  
概要 [159](#)  
PRN (1 次リソース名)  
メッセージ経路指定、ISC [480](#)  
ATTACH FM ヘッダー [577](#)  
SCHEDULER FM ヘッダー [577](#)  
PROFILE セキュリティー・レベル [830](#)

PSB (プログラム仕様ブロック)  
CICS DRA  
コーディングのガイドライン [763](#)  
PSTOP 状態 [433](#)  
PURGE  
エラー・リカバリー手順  
開始 [538](#)  
終了 [538](#)  
DFC サポート層 [538](#)  
エラー・リカバリー・プロシージャ  
選択受信側ページ後の状態 [541](#)  
送信側ページ後の状態 [539](#)  
単一 [538](#)  
チェーンのスパン [539](#)  
複数チェーン [538](#)  
DFC プロトコル [538](#)  
送信側 ERP [550](#)  
PURGE 状態 [433](#)

## Q

QCF [809](#)  
QEC コマンド [963](#)  
QERROR 状態 [406](#)  
QGET FM ヘッダー  
形式 [581](#)  
説明 [568](#)  
QGETN FM ヘッダー  
形式 [582](#)  
説明 [567](#)  
QLOCK 状態 [406](#)  
QMODEL FM ヘッダー  
応答または出力 [568](#)  
形式 [581](#)  
要求または入力 [567](#)  
CICS [651](#)  
IMS によってサポートされる [566](#), [581](#)  
QGET [568](#)  
QGETN [567](#)  
QPURGE [568](#)  
QSTATUS [569](#)  
QXFR [568](#)  
QPURGE FM ヘッダー  
形式 [583](#)  
IMS によって受信された [568](#)  
QSTATUS FM ヘッダー  
形式 [584](#)  
IMS によって送信された [569](#)  
QSTOP 状態 [433](#)  
QUIESCE (/QUIESCE) コマンド  
ISC TCP/IP セッション [620](#)  
QXFR FM ヘッダー  
形式 [585](#)  
出力として [568](#)

## R

RACF  
パスチケットと IMS Connect [201](#)  
コマンド・セキュリティー [414](#)  
大/小文字混合パスワード、IMS Connect [200](#)  
パスワード、IMS Connect の変更 [198](#)  
パスワード・フレーズ、IMS Connect の変更 [199](#)



RACF (続き)  
 ACEE  
 OTMA ACEE のリフレッシュ [834](#)  
 IMS Connect サポート [181](#)  
 IMS Connect 大/小文字混合パスワード [200](#)  
 IMS Connect のパスチケットのサポート [201](#)  
 IMS Connect ユーザー ID キャッシュ [185](#)  
 IMS Connect、直接サポートの使用可能化 [183](#)  
 IMS Connect、デフォルトのユーザー ID [184](#)  
 IMS Connect、統計の使用可能化 [183](#)  
 IMS Connect、パスワードの変更 [198](#)  
 IMS Connect、パスワード・フレーズの変更 [199](#)  
 IMS Connect、汎用戻りコードまたはメッセージ [182](#)  
 OTMA セキュリティー・クラス [836](#)

RACF (リソース・アクセス管理機能)  
 無許可端末使用 [412](#)  
 APPC トランザクション・セキュリティ [58](#)  
 ETO セキュリティー [379](#)  
 FACILITY クラス定義 [790](#)  
 MSC (複数システム結合機能) のセキュリティ [722](#)  
 OTMA セキュリティー [829](#)  
 OTMA の制約事項 [814](#)

RACF パスチケット  
 IMS Connect  
 DRDA クライアント [202](#)

RACROUTE 呼び出し  
 IMS Connect  
 サンプル 出口からの呼び出しへのエラー応答 [193](#)

RAP FM ヘッダー  
 形式 [588](#)  
 説明 [558](#)  
 例 [558](#)  
 MFS [569](#)

Rapid Network Reconnect (RNR)  
 計画 [421](#)

RCVYCONV= [409](#)  
 RCVYFP= [409](#)  
 RCVYRESP= [409](#)  
 RCVYSTSN= [409](#)

RDPN (戻り宛先プロセス名)  
 メッセージ経路指定、ISC [480](#)  
 ATTACH FM ヘッダー [577](#)  
 SCHEDULER FM ヘッダー [577](#)

RELQ コマンド [963](#)  
 REPresynch [866](#)  
 REPresynch コマンド [903](#)  
 REQresynch [866](#)  
 REQresynch コマンド [902](#)

Resource Manager (RM)  
 IMS TM リソース  
 管理 [388](#)

RESUME TPIPE 要求  
 代替クライアント ID [337](#)  
 非同期出力のリトリブ  
 IMS Connect [331](#)  
 並列処理  
 インプリメント [337](#)  
 コールアウト・メッセージ [887](#)  
 代替クライアント ID [337](#)  
 問題の診断 [338](#)  
 IMS Connect [335](#)  
 IMS Connect からの使用可能化 [336](#), [337](#)  
 IMS Connect  
 並列処理 [335](#)

RESUME TPIPE 要求 (続き)  
 IMS Connect (続き)  
 並列処理、実装 [337](#)  
 並列処理、使用可能化 [336](#), [337](#)  
 並列処理、問題の診断 [338](#)

RESUME TPIPE  
 代替クライアント ID [345](#)  
 別のクライアントの出力のリトリブ [345](#)  
 IRM\_RT\_ALTCID [345](#)

RESUME TPIPE プロトコル  
 フローの例 [333](#)

RM 類似性 [409](#)

RMM (MOD 要求メッセージ)  
 形式 [253](#)

RNR [722](#)  
 RNR (高速ネットワーク再接続)  
 計画 [421](#)

RPRN (戻り 1 次リソース名)  
 メッセージ経路指定、ISC [480](#)  
 ATTACH FM ヘッダー [577](#)  
 SCHEDULER FM ヘッダー [577](#)

RQ\* メッセージ、LUSTATUS コマンド [542](#)  
 RQD\* [465](#), [522](#), [526](#)  
 RQE\* [465](#), [522](#), [526](#)  
 RQR コマンド、SLU P [936](#)  
 RR\_BACKED\_OUT [26](#)  
 RR\_OK [26](#)  
 RR\_PROGRAM\_STATE\_CHECK [26](#)  
 RRS [862](#)  
 RRS (z/OS リソース・リカバリー・サービス)  
 説明 [28](#)  
 リソース・リカバリー [28](#)

RRSAF [117](#)  
 RSHUT コマンド [545](#)  
 RSM (要求状況メッセージ)  
 形式 [254](#)

RTO (リモート 端末オペレーター) [429](#)

RTR (受信可) コマンド  
 高速機能 [985](#)  
 コンポーネント 状況を無保護にリセット [949](#)  
 プロトコル [545](#)  
 要約 [545](#), [985](#)  
 IMS 機能 [985](#)

RTT (リソース変換テーブル) [120](#)

## S

SBI (停止ブラケット開始) コマンド  
 セッション・シャットダウン [554](#)  
 CICS [638](#)  
 IMS-CICS セッション [638](#)

SC (セッション制御) プロトコル  
 セッション終了  
 異常 [513](#)  
 正常 [513](#)  
 セッション状態 [511](#)  
 セッションの開始  
 完了 [512](#)  
 ISC VTAM [500](#)  
 セッションのバインド  
 競争の解決 [503](#)  
 交渉可能 BIND 対交渉不能 BIND [502](#)  
 単一セッション [503](#)  
 同期化セッション [503](#)

SC (セッション制御) プロトコル (続き)  
 セッションのバインド (続き)  
 並列セッション [503](#)  
 バインド・パラメーター [509](#)  
 メッセージ再同期  
 プロシージャラーの設計 [505](#)  
 メッセージ再同期 (message resynchronization)  
 使用されるコマンド [504](#)  
 必要時 [504](#)  
 MNPS を使用する XRF、接続設定 [501](#)  
 STSN 形式 [516](#)  
 STSN フロー  
 1 次から 2 次へのハーフセッション [514](#)  
 2 次から 1 次へのハーフセッション [515](#)  
 USERVAR を使用する XRF、接続設定 [502](#)  
 XRF 複合システム、ISC セッションの確立 [501](#)  
 XRF 複合システム、接続設定 [501](#)

SCA (システム制御域)、ISC サポート [478](#)

SCHEDULER FM ヘッダー  
 概要 [558](#)  
 形式 [588](#)  
 チェーン・メッセージ・サポート [559](#)  
 パラメーターの説明 [575](#), [577](#)  
 非同期実行の要求 [472](#)  
 例 [591](#)  
 ATTACH [558](#)  
 IMS-CICS セッション [649](#)  
 MFS [564](#)

SDF II  
 MFS 形式 [443](#)

SDT (データ通信開始) コマンド  
 セッション開始の完了 [512](#)  
 IMS [962](#)

SEND INVITE EXEC コマンド、CICS [621](#)  
 SEND LAST EXEC コマンド、CICS [623](#)  
 SEND/RECEIVE EXEC コマンド、CICS [621](#)  
 SET IPCONN CICS コマンド  
 ISC TCP/IP セッション [618](#)

SIGNAL コマンド  
 プロトコル [553](#)  
 VTAM [986](#)

SINGLE1 パラメーター [477](#), [532](#)  
 SINGLE2 パラメーター [477](#), [533](#)

SLU 2 トランザクション・フロー  
 OTMA を使用して [781](#)

SLU P  
 アプリケーション・プログラム  
 金融機関からの変換 [934](#)  
 使用可能な機能 [932](#)  
 MFS 考慮事項 [932](#)  
 XRF 考慮事項 [933](#)  
 機能 [941](#)  
 構成例 [932](#)  
 コントローラーが検出したエラー [983](#)  
 サポートされる端末 [931](#)  
 セッション終了  
 即時 [965](#)  
 入力コンポーネントの定義 [942](#)  
 ネットワーク  
 主なパーツ [931](#)  
 ブラケットと送信 / 受信のプロトコル [938](#), [966](#)  
 メッセージ再同期 (message resynchronization)  
 コントローラー・アプリケーション・プログラム  
[959](#)

SLU P (続き)  
 ワークステーション [932](#)  
 IMS 検出エラー [982](#)  
 MFS DPM オプション [973](#)  
 SCAN/NOSCAN [934](#)  
 VTAM  
 機能 [934](#)  
 コマンド [982](#)  
 コマンドと標識 [935](#)  
 標識 [982](#)  
 XRF [953](#)  
 XRF 複合システム、接続の確立 [938](#)

SLU P ネットワークの開始 [431](#)

SNA QUIESCE [407](#)  
 SNA コマンド [466](#)  
 SPA (スクラッチパッド域)  
 サイズ [700](#), [703](#)  
 特性 [399](#)  
 トランザクション・コード・フィールド [400](#), [405](#)  
 MSC (複数システム結合機能) 会話 [703](#)

SPOOL コマンド  
 出力制御 [450](#)

SRVresynch [866](#)  
 SRVresynch コマンド [902](#)

START (/START) コマンド  
 SLU P ネットワークの開始 [957](#)

START/RETRIEVE EXEC コマンド、CICS [626](#)

STEPLIB  
 TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン [750](#)

STOP (/STOP) コマンド [406](#)

STSN 機能、ETO の要件 [111](#)

STSN コマンド  
 アクション・コード [960](#)  
 アクション・コードの形式 [517](#), [960](#)  
 応答 (response)  
 応答要件 [960](#)  
 要約 [962](#)  
 応答  
 要件 [962](#)  
 金融機関通信システム [960](#)  
 コントローラー・シーケンス番号、検査 [960](#)  
 同期点 [508](#)  
 必要なアクション [961](#)  
 フロー [514](#)  
 メッセージ再同期 (message resynchronization) [960](#),  
[962](#)  
 VTAM シーケンス番号、検査 [960](#)

SUBPOOL マクロ [473](#), [494](#)

switch\_context サービス [863](#)

SYS1.VTAMLST  
 アプリケーション・ノードとして IMS を定義 [426](#)  
 ATCCONxx メンバー [429](#)  
 ATCSTRyy メンバー [429](#)  
 VTAM ノード [426](#)

SYSERROR FM ヘッダー  
 形式 [589](#)

SYSID  
 IMSplex 内の管理 [737](#)  
 IMSplex 内のシステム間での共用 [734](#)  
 IMSplex 内の複製 [738](#)

SYSID (システム ID) [693](#)

SYSID キーワード  
 論理リンク・パス [686](#)

SYSID テーブル  
IMSplex 内の MSNAME 定義の削除 [737](#)  
SYSMSG FM ヘッダー  
形式 [589](#)  
送信 [559](#)  
タイプ [560](#)  
ATTACH FM ヘッダー [559](#)  
Sysplex Distributor  
IMS Connect [875](#)  
IMS Connect サポート [160](#)  
OTMA サポート [875](#)  
SYSSTAT FM ヘッダー  
形式 [589](#)  
SYSSTAT FM ヘッダー、フォーマット設定 [589](#)

## T

T パイプ  
アイドル、自動除去 [822](#)  
アイドル状態の T パイプの除去 [822](#)  
出力のリトリブ  
IMS Connect [331](#)  
代替クライアント ID の非同期出力のリトリブ [345](#)  
リソースのインパクト [820](#)  
MULTIRTP  
使用可能化 [793](#)  
RESUME TPIPE 要求  
複数のアクティブ、サポート [792](#)  
複数のアクティブな、サポートの使用可能化 [793](#)  
並列処理 [792](#)  
並列処理、使用可能化 [793](#)  
WAIT\_R 状況 [813](#)  
TBresynch [866](#)  
TBresynch コマンド [904](#)  
TCP/IP  
出力メッセージの転送機能 [303](#)  
出力メッセージのページ機能 [301](#)  
セキュリティ  
IMS Connect のインスタンス間の接続 [188](#)  
IMS システム間の接続 [188](#)  
OTMA ACEE エージング値の指定 [207](#)  
ソケット接続  
持続 [309](#)  
トランザクション (transaction) [310](#)  
非持続 [310](#)  
汎用リソース  
類似性、IMS Connect 内のクリア [748](#)  
類似性、MSC リンクのクリア [747](#)  
IMS Connect の MSC のサポート [157](#)  
MSC および XRF [749](#)  
MSC の類似性管理 [746](#)  
MSC の類似性の持続性 [747](#)  
MSC 類似性、IMS Connect 内のクリア [748](#)  
MSC 類似性、クリア [747](#)  
MSC 類似性管理 [746](#)  
MSC 類似性の持続性 [747](#)  
XRF および MSC [749](#)  
複数システム結合機能 (MSC)  
汎用リソース [745](#)  
汎用リソースおよび XRF [749](#)  
汎用リソース・グループ内の類似性の持続性 [747](#)  
汎用リソースでの類似性管理 [746](#)  
類似性、IMS Connect 内のクリア [748](#)  
類似性、クリア [747](#)

TCP/IP (続き)  
複数システム結合機能 (MSC) (続き)  
XRF および汎用リソース [749](#)  
メッセージ・フォーマット [223](#)  
CICS  
サポートされる機能、ISC [603](#)  
ISC セッション、サポートされる機能 [601](#), [603](#)  
IMS Connect [145](#)  
IMS Connect 関連の TCP/IP 設定 [371](#)  
IMS Connect トラストド・ユーザー [206](#)  
IMS Connect の KeepAlive インターバル [318](#)  
IMS Connect の RACF パスチケットのサポート [201](#)  
IMS Connect メッセージ構造 [223](#)  
IMS Connect、クライアント通信 [148](#)  
IMS からの 外部サービスの呼び出し [881](#)  
IMS 間 TCP/IP 通信  
自動的に再接続 [159](#)  
スーパーメンバー・サポート [158](#)  
IMS Connect サポートの概要 [154](#)  
IMS Connect の MSC のサポート [156](#)  
IMS Connect の OTMA のサポート [158](#)  
ISC  
サポートされる機能、CICS の [603](#)  
CICS セッション、サポートされる機能 [601](#), [603](#)  
CICS フロントエンド・トランザクション・タイプ [621](#)  
CICS、サポートされる機能 [603](#)  
IMS Connect、サポートの概要 [153](#)  
ISC サポート  
静的端末定義 [611](#), [612](#)  
セッション、開始 [617](#)  
セッションの開始、CICS から [618](#)  
端末定義 [611](#)  
動的端末定義 [611](#)  
リンクの定義、CICS への [615](#)  
リンクの定義、IMS Connect への [614](#)  
ISC セッション再始動 [620](#)  
ISC セッション終了  
異常 [620](#)  
正規手順での [620](#)  
無条件 [620](#)  
ISC セッションの再始動 [620](#)  
ISC セッションの終了  
正規手順での [620](#)  
無条件で [620](#)  
ISC と ETO [63](#)  
ISC、サポート  
要件 [609](#)  
ISC、に対するサポート  
概要 [608](#)  
制約事項 [610](#)  
セキュリティ [610](#)  
TCP/IP から VTAM への切り替え [613](#)  
VTAM へのフォールバック [613](#)  
MSC  
汎用リソース [745](#)  
汎用リソースおよび XRF [749](#)  
汎用リソース・グループ内の類似性の持続性 [747](#)  
汎用リソースでの類似性管理 [746](#)  
類似性、IMS Connect 内のクリア [748](#)  
類似性、クリア [747](#)  
XRF および汎用リソース [749](#)  
MSC 物理リンク  
定義 [714](#)

## TERMINAL マクロ

ISC セッションとしての定義 [495](#)  
MODETBL= キーワード [428](#)

## TM リソース

共用 [389](#)  
共用、使用不可化 [389](#)  
リソース名の一意性  
使用不可化、制約の [389](#)

## TM/MSC メッセージ経路指定および制御ユーザー出口ルーチン

メッセージ経路指定 [750](#)  
類似性経路指定 [752](#)  
IMSpIex 類似性経路指定 [752](#)  
JOBLIB [750](#)  
LINKLIST [750](#)  
STEPLIB [750](#)

## TMAMAGNG [898](#)

## TMAMALTB [812](#)

## TMAMCCCI [893](#)

## TMAMCCHN [897](#)

## TMAMCMGT [892](#)

## TMAMCPFG [895](#)

## TMAMCPFL [897](#)

## TMAMCRSC [898](#)

## TMAMCRSI [893](#)

## TMAMCRSQ [898](#)

## TMAMCSEQ [898](#)

## TMAMCSNC [898](#)

## TMAMCSNS [898](#)

## TMAMCSSN [897](#)

## TMAMCTNM [897](#)

## TMAMCTYP [894](#)

## TMAMRTID [898](#)

## TMEMBER オペランド [808](#)

## TPIPE オペランド [809](#)

## tpipe\_Bid resynch [866](#)

## TRANSACT マクロ

大文字への変換 [444](#)  
EDIT=ULC [447](#)  
PRTY= キーワード [716](#)

## TYPE マクロ

端末の定義 [394](#)

## TYPE マクロの UNITYPE キーワード [418](#)

## U

## UNBIND コマンド、セッション開始の停止 [512](#)

## Universal ドライバー

タイプ 2 接続  
CICS、構成 [20](#)

## UNIX システム・サービス

ソケット制限 [314](#)  
MAXFILEPROC パラメーター [314](#)

## UNIX パイプ [784](#)

## user

シスプレックス、内  
リカバリー状況 [411](#)

## USER [470](#)

## USTOPPED 状態 [433](#)

## V

## VLVB レコード [575](#)

## VTAM

ネットワークの定義 [425](#)

MSC (複数システム結合機能) 物理リンク・タイプ [683](#)

MSC および VTAM 汎用リソース [750](#)

MSC 物理リンク

定義 [715](#)

RNR (高速ネットワーク再接続)

計画 [421](#)

## SLU P

コマンド [982](#)

標識 [982](#)

## VTAM (仮想記憶通信アクセス方式)

### 機能

金融機関通信システム [934](#)

コマンドと標識 [466](#), [935](#)

データ伝送 [466](#)

IMS [935](#)

ISC [466](#)

ISC によって使用された [604](#), [640](#)

SLU P [934](#)

コマンドと標識 [466](#), [935](#)

出力バッファ [427](#)

### 生成

ストレージ要件 [427](#)

ホスト・サブシステムとしての IMS [426](#)

LOGON MODE ID [428](#)

NCP バッファ・プール値 [427](#)

VTAM 構成 [429](#)

VTAM ノード [426](#)

VTAM バッファ・プール値 [427](#)

### セッションの確立

金融機関通信システム [937](#)

SLU P [937](#)

接続されている端末 [394](#)

選択したノード名 [395](#)

同期データ・リンク制御 (SDLC) [394](#)

ネットワークの役割 [376](#)

半二重プロトコル、ISC [528](#)

ブラケット・プロトコル

金融機関通信システム [938](#)

CD (方向変換) [938](#)

SLU P [938](#)

並列セッション [715](#)

### マクロ

SEND [468](#)

TERMSESS [468](#)

2 進データ同期通信 (BSC) [394](#)

ACBNAME パラメーター [715](#)

BB 標識 [477](#), [542](#)

BID コマンド [528](#), [944](#)

BIS コマンド

CICS での使用 [638](#)

LU 6.1 ハーフセッション [554](#)

CANCEL コマンド [536](#), [985](#)

CD 標識 [477](#), [937](#)

CHASE コマンド [538](#)

COMM マクロ [715](#)

EB 標識 [477](#), [542](#)

IMS に対する関係 [378](#)

MFS サポートを持つ装置 [394](#)

MSC (複数システム結合機能) リンク [683](#)

RQR コマンド [936](#), [986](#)

SBI コマンド

CICS での使用 [638](#)

## VTAM (仮想記憶通信アクセス方式) (続き)

SBI コマンド (続き)

LU 6.1 ハーフセッション [554](#)

SDT コマンド [512](#)

SESSION パラメーター [715](#)

SIGNAL コマンド [553, 986](#)

UNBIND コマンド [512](#)

VTAM 開始リスト [429](#)

VTAM サポート [776](#)

VTAM で定義されている USSTAB オプション [429](#)

VTAM ネットワーク管理 [373](#)

VTAM 汎用リソース

複数システム結合機能 (MSC) [750](#)

## W

WAIT\_R 状況

IMS 再始動 [813](#)

WebSphere Application Server

構成

IMS Universal Database リソース・アダプター、インストール [5](#)

接続ファクトリー

定義 [6](#)

プロパティ [6](#)

IMS Universal Database リソース・アダプター

アプリケーション、インストール [7](#)

IMS Universal ドライバー

構成 [4](#)

WebSphere Application Server for z/OS

構成

タイプ 2 IMS Universal Database リソース・アダプター、インストール [8](#)

タイプ 2 IMS Universal ドライバー

構成 [3](#)

タイプ 2 接続、IMS Universal Database リソース・アダプター

概要 [7](#)

IMS Universal Database リソース・アダプター

アプリケーション、インストール [12](#)

クラスパス、設定 [10](#)

データ・ソース、インストール [10](#)

IMS Universal ドライバーの構成 [3](#)

WebSphere Application Server Liberty

タイプ 2 接続

構成ファイル [19](#)

タイプ 4 接続

構成ファイル [15](#)

## X

XCF (z/OS システム間カップリング・ファシリティ)

基本的な OTMA メッセージ・フロー [784](#)

XML

COBOL への変換 [222](#)

IMS Connect

メッセージ構造 [220](#)

XML の COBOL への変換 [219](#)

XML 変換の例 [222](#)

IMS Connect XML 変換サポート

概要 [159](#)

XRF

SLU P [953](#)

XRF (拡張回復機能)

クラス 1 端末 [420](#)

クラス 2 端末 [419](#)

クラス 3 端末 [419](#)

サービス・クラス [420](#)

端末 [418](#)

端末サポート [395, 420](#)

端末のクラス [418](#)

通信の確立

金融機関通信システム [938](#)

システムのテークオーバーに関する考慮事項 [938](#)

ISC [501](#)

SLU P [938](#)

定義 [416](#)

テークオーバーに関する考慮事項

金融機関通信システム [938](#)

SLU P [938](#)

マスター端末 [404](#)

リカバリー・モード [410](#)

APPC/IMS [45, 52](#)

MNPS [417](#)

OTMA での処理 [808](#)

SLU P アプリケーション・プログラム [933](#)

## Z

z/OS

システム・ログ [385](#)

IMS Connect の Sysplex Distributor のサポート [160](#)

Sysplex Distributor、IMS Connect サポート [160](#)

z/OS Sysplex Distributor

IMS Connect サポート [160](#)

z/OS アプリケーション・プログラム

ODBA (Open Database Access) を使用した IMS へのアクセス [767](#)

Open Database Access (ODBA) を使用した IMS データベースへのアクセス [767](#)

z/OS システム間カップリング・ファシリティ (XCF)

マクロおよび OTMA [814](#)

z/OS プログラム [774](#)

z/OS リソース・リカバリー・サービス

および保護トランザクション [862](#)

IMS にサポートされる出口 [862](#)

z/OS リソース・リカバリー・サービス (RRS) [28](#)

## [特殊文字]

/ASSIGN コマンド

ETO 制限 [103](#)

ISC セッションのコールド・スタート [506](#)

/CHANGE コマンド

ISC セッションの開始 [506](#)

/CLSDST コマンド

ISC TCP/IP セッション [620](#)

/DEQUEUE コマンド

ISC 出力メッセージのコミット [521](#)

/DISPLAY

TMEMBER コマンド [812](#)

TRANSACTION コマンド [774](#)

/DISPLAY コマンド

保留状態の ISC 出力 [506](#)

QSTOP 状態 [433](#)

/EXIT コマンド

/EXIT コマンド (続き)  
 会話の異常終了 [405](#)  
 ISC 会話モード・エラー [519](#)  
/MSASSIGN コマンド  
 物理リンクから論理リンクへの割り当て [685](#)  
/MSVERIFY コマンド  
 エラー応答 [721](#)  
 動的妥当性検査 [721](#)  
/OPNDST コマンド  
 ISC TCP/IP セッション [617](#)  
 MODETBL キーワードとの関連 [428](#)  
/QUIESCE コマンド  
 ISC TCP/IP セッション [620](#)  
/SECURE OTMA NONE コマンド [833](#)  
/SSR コマンド [125](#)  
/START OTMA コマンド [808](#)  
/START コマンド  
 IMS を作動可能にする [431](#)  
 SLU P ネットワークの開始 [957](#)  
/STOP OTMA コマンド [808](#)  
/STOP コマンド [406](#)









プログラム番号: 5635-A06  
5655-DS5  
5655-TM4