

CICS Transaction Server for z/
OSバージョン 5 リリース 6

EXCI の使用



注記

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

本書は、IBM® CICS® Transaction Server for z/OS®, バージョン 5 リリース 6 (製品番号 5655-Y305655-BTA)、および新しい版で明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

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

原典：

CICS Transaction Server for z/OS
Version 5 Release 5
Using EXCI

発行：

日本アイ・ビー・エム株式会社

担当：

トランスレーション・サービス・センター

© Copyright International Business Machines Corporation 1974, 2020.

目次

この PDF について	vii
第 1 章 EXCI の概念	1
第 2 章 外部 CICS インターフェースの概要	3
EXCI プログラミング・インターフェース	4
EXEC CICS か CALL インターフェースかの選択	4
外部 CICS CALL インターフェースの図	5
EXCI EXEC CICS インターフェースの図	8
リソース・リカバリー	8
外部 CICS インターフェースによる RRMS の使用法	9
クライアント・プログラムでの同期点の使用法	11
EXCI CALL インターフェース	12
EXCI CALL インターフェース・コマンド	12
EXCI 呼び出し応答コード値	32
EXCI CALL インターフェースの戻り域	33
ヌル・パラメーターを持つ EXCI CALL の例	35
EXCI EXEC CICS インターフェース	36
EXEC CICS LINK コマンド (EXCI)	36
EXEC CICS DELETE CHANNEL コマンド (EXCI)	43
EXEC CICS DELETE CONTAINER コマンド (EXCI)	44
EXEC CICS ENDBROWSE CONTAINER コマンド (EXCI)	45
EXEC CICS GET CONTAINER コマンド (EXCI)	46
EXEC CICS GETNEXT CONTAINER コマンド (EXCI)	50
EXEC CICS MOVE CONTAINER コマンド (EXCI)	51
EXEC CICS PUT CONTAINER コマンド (EXCI)	53
EXEC CICS QUERY CHANNEL コマンド (EXCI)	57
EXEC CICS STARTBROWSE CONTAINER コマンド (EXCI)	57
EXCI クライアント・プログラムのコンパイルとリンク・エディット	58
EXCI クライアント・プログラムを実行するためのジョブ制御言語	59
EXCI プログラミングの考慮事項	61
第 3 章 EXCI の構成	63
EXCI の静的ルーティングのセットアップ	63
EXCI の動的ルーティングのセットアップ	63
CICS への接続の定義	64
EXCI の CONNECTION リソース定義	64
EXCI 接続の SESSIONS リソース定義	65
EXCI 接続の状態の照会	67
EXCI ユーザー置換可能モジュール	68
EXCI オプション・テーブル DFHXCOPT の使用	70
第 4 章 EXCI のセキュリティ	77
MRO ログオン・セキュリティおよびバインド時のセキュリティの使用	77
EXCI 領域の DFHAPPL FACILITY クラス・プロファイルの定義	77
リンク・セキュリティ	78
ユーザー・セキュリティ	78
代理ユーザー検査	79
第 5 章 EXCI のトラブルシューティング	81

トレース	81
GTF トレースのフォーマット設定	81
システム・ダンプの使用	81
システム・ダンプのフォーマット設定	82
SYSMDUMP のキャプチャー	82
コンソールでのダンプ用の MVS DUMP コマンドの使用	82
EXCI サービス・トラップ、DFHXCTRA	82
RRMS を使用した問題判別	83
EXCI トレースの入り口点	83

第 6 章 EXCI 呼び出しで返される応答と理由コード 97

応答 OK の理由コード	97
0: NORMAL	97
応答 WARNING の理由コード	97
1: PIPE_ALREADY_OPEN	97
2: PIPE_ALREADY_CLOSED	97
3: VERIFY_BLOCK_FM_ERROR	97
4: WS_FREEMAIN_ERROR	98
5: XCPIPE_FREEMAIN_ERROR	98
6: IRP_IOAREA_FM_FAILURE	98
7: SERVER_TERMINATED	99
8: XFRASSTG1_FM_FAILURE	99
応答 RETRYABLE の理由コード	99
201: NO_CICS_IRC_STARTED	99
202: NO_PIPE	100
203 (Open_Pipe 呼び出し時): NO_CICS	100
204: WRONG_MVS_FOR_RRMS	100
205: RRMS_NOT_AVAILABLE	101
応答 USER_ERROR の理由コード	101
401: INVALID_CALL_TYPE	101
402: INVALID_VERSION_NUMBER	102
403: INVALID_APPL_NAME	102
404: INVALID_USER_TOKEN	102
405: PIPE_NOT_CLOSED	102
406: PIPE_NOT_OPEN	103
407: INVALID_USERID	103
408: INVALID_UOWID	103
409: INVALID_TRANSID	104
410: DFHMEBM_LOAD_FAILED	104
411: DFHMET4E_LOAD_FAILED	104
412: DFHXCURM_LOAD_FAILED	105
413: DFHXCTRA_LOAD_FAILED	105
414: IRP_ABORT_RECEIVED	105
415: INVALID_CONNECTION_DEFN	106
416: INVALID_CICS_RELEASE	106
417: PIPE_MUST_CLOSE	106
418: INVALID_PIPE_TOKEN	106
419: CICS_AFCB_PRESENT	107
420: DFHXCLOPT_LOAD_FAILED	107
421: RUNNING_UNDER_AN_IRB	107
422: SERVER_ABENDED	108
423: SURROGATE_CHECK_FAILED	108
424: RRMS_NOT_SUPPORTED	108
425: UOWID_NOT_ALLOWED	109
426: INVALID_TRANSID2	109
427: INVALID_CCSID	109
428: INVALID_ENDIAN	109

429: DFHXCEIX_LOAD_FAILED.....	110
430: DFHXCPRX_LOAD_FAILED.....	110
431: COMMAREA_LEN_NOT_ALLOWED.....	110
432: DATA_LEN_NOT_ALLOWED.....	111
433: CCSID_NOT_ALLOWED.....	111
434: ENDIAN_NOT_ALLOWED.....	111
応答 SYSTEM_ERROR の理由コード.....	111
601: WS_GETMAIN_ERROR.....	112
602: XCGLOBAL_GETMAIN_ERROR.....	112
603: XCUSER_GETMAIN_ERROR.....	112
604: XCPipe_GETMAIN_ERROR.....	112
605: VERIFY_BLOCK_GM_ERROR.....	113
606: SSI_VERIFY_FAILED.....	113
607: CICS_SVC_CALL_FAILURE.....	113
608: IRC_LOGON_FAILURE.....	114
609: IRC_CONNECT_FAILURE.....	114
610: IRC_DISCONNECT_FAILURE.....	114
611: IRC_LOGOFF_FAILURE.....	115
612: TRANSFORM_1_ERROR.....	115
613: TRANSFORM_4_ERROR.....	115
614: IRP_NULL_DATA_RECEIVED.....	115
615: IRP_NEGATIVE_RESPONSE.....	116
616: IRP_SWITCH_PULL_FAILURE.....	116
617: IRP_IOAREA_GM_FAILURE.....	116
619: IRP_BAD_IOAREA.....	117
620: IRP_PROTOCOL_ERROR.....	117
621: PIPE_RECOVERY_FAILURE.....	117
622: ESTAE_SETUP_FAILURE.....	118
623: ESTAE_INVOKED.....	118
624: SERVER_TIMEDOUT.....	118
625: STIMER_SETUP_FAILURE.....	119
626: STIMER_CANCEL_FAILURE.....	119
627: INCORRECT_SVC_LEVEL.....	119
628: IRP_LEVEL_CHECK_FAILURE.....	120
629: SERVER_PROTOCOL_ERROR.....	120
630: RRMS_ERROR.....	120
631: RRMS_SEVERE_ERROR.....	121
632: XCGUR_GETMAIN_ERROR	121
633: INQUIRE_CHANNEL_FAILED.....	121
第 7 章 EXCI サンプル: チャネルおよびコンテナのサンプル・アプリケーション....	123
EXCI チャネルおよびコンテナ・サンプル・アプリケーションについて.....	123
EXCI チャネルおよびコンテナのサンプル・プログラムのセットアップ.....	124
EXCI チャネルおよびコンテナのサンプル・アプリケーションの実行.....	125
特記事項.....	127
索引.....	133

この PDF について

この PDF では、外部 CICS インターフェース (EXCI) を使用して CICS Transaction Server for z/OS のサービスを外部プログラムで使用可能にする方法について説明されています。CICS TS 5.4 より前は、この情報は、「外部インターフェース・ガイド」に収録されていました。

本書で使用する用語および表記の詳細については、IBM Knowledge Center の [CICS 資料で使用されている表記規則および用語](#)を参照してください。

この PDF の作成日

この PDF は、2020 年 5 月 28 日に作成されました。

第 1 章 EXCI の概念

外部 CICS インターフェースにより、非 CICS 環境から CICS アプリケーションへのアクセスが、より簡単になります。

MVS™ でプログラムを実行すると、EXEC CICS LINK PROGRAM コマンドを実行でき、CICS 領域で実行されている CICS アプリケーション・プログラムを呼び出すことができます。また、CALL インターフェースを使用した方が適切な場合には、MVS プログラムは CALL インターフェースを使用できます。

このプログラミング・インターフェースの提供により、例えば MVS プログラムで以下のことが可能になります。

- CICS がリソースにアクセスする間に、保全性を保ってリソースを更新します。
- CICS リソースを MVS ジョブの開始時にオフラインにして、終了時にオンラインにします。例えば、以下のことを行うことができます。
 - CICS ファイルのオープンおよびクローズを実行します。
 - CICS でトランザクションを有効化および無効化します (これにより、システム・バックアップおよびリカバリー手順の実行中にマスター端末オペレーターが必要なくなります)。

外部 CICS インターフェースでは、新しい方法でクライアント/サーバー・アプリケーションを実装できます。この方法では、非 CICS 環境のクライアント・プログラムが、CICS アドレス・スペースで実行されるサーバー・プログラムを呼び出します。外部 CICS インターフェースによって、TSO およびバッチ・アプリケーションを活用できるだけでなく、オープン・クライアント/サーバー環境での CICS アプリケーション・プログラムの使用を拡張できます。

第 2 章 外部 CICS インターフェースの概要

外部 CICS インターフェースは、アプリケーション・プログラミング・インターフェースです。このインターフェースにより、MVS で実行される非 CICS プログラム (クライアント・プログラム) は、CICS 領域で実行されるプログラム (サーバー・プログラム) を呼び出し、通信域を使用したり、チャンネルとコンテナのセットを使用したりして、データを受け渡すことができます。

CICS アプリケーション・プログラムは、別の CICS アプリケーション・プログラムからリンクされているかのように開始されます。

外部 CICS インターフェースを使用すると、セッションの割り振りとオープン、または CICS 領域へのパイプ (送信側プロセスと受信側プロセス間の単方向通信パス) の割り振りとオープン、およびそれらにおける分散プログラム・リンク (DPL) 要求の受け渡しが可能です。CICS 領域間通信 (IRC) 機能の複数領域操作 (MRO) 機能は、これらの要求をサポートし、各パイプは 1 つの MRO セッションにマップされます。ここでクライアント・プログラムは送信側プロセスを表し、CICS サーバー領域は受信側プロセスを表します。EXCI アドレス・スペースごとに 100 個のパイプというデフォルトの制限があります。この制限は、MVS が IPL されるときに変更できます。

この制限により、EXCI クライアントによる MRO リソースの独占が妨げられ、CICS 領域での MRO の使用が妨げられることもあります。この制限は、MRO とシステム間 MRO (XCF/MRO) 環境の両方で適用されます。**allocate_pipe** 要求が出されると、結果として **MRO LOGON** 要求が出されます。すべてのアドレス・スペースで許可される **MRO LOGON** 要求の総数には制限があります。これは、XCF/MRO を使用するとき重要です。XCF グループ内のメンバー数の制限により、**MRO LOGON** 要求の総数も制限されます。

外部 CICS インターフェースは、CICS 領域を識別し、**APPLID** システム初期設定パラメーターに定義されている CICS 領域 **APPLID** によって通信します。**APPLID** を指定するには、EXCI API 呼び出しまたは DFHXCURM ユーザー置き換え可能プログラムを使用します。また、DFHXCURM を使用すると、**allocate_pipe** 要求で使用する XCFGROUP の値を変更できます。DFHXCURM について詳しくは、[EXCI ユーザー置換可能モジュール](#)を参照してください。

注: 汎用 **APPLID** という語と汎用リソース名という語を混同しないでください。汎用リソース名は、EXCI でサポートされない z/OS Communications Server 汎用リソース・グループのみに適用されます。

クライアント・プログラムと CICS サーバー領域 (サーバー・プログラムが実行または定義されている領域) は、以下の場合を除き、同じ MVS イメージ内にある必要があります。

- CICS 領域が、システム間 MRO をサポートするシスプレックスで実行されている。
- クライアント・プログラムから発行されるすべての DPL 要求で、SYNCONRETURN オプションが指定されている。

また、MVS イメージにローカル CICS 領域がない場合は、外部 CICS インターフェースが使用する SVC パラメーターを指定するため、DFHXCLOPT テーブルに CICS SVC パラメーターをコーディングする必要があります。外部 CICS インターフェースはクロスメモリー・アクセス方式をサポートしませんが、CICS XCF/MRO 機能から提供される XCF アクセス方式を使用できます。XCF/MRO については、[EXCI オプション・テーブル DFHXCLOPT の使用](#)を参照してください。

外部 CICS インターフェースを使用するクライアント・プログラムは、同一の MVS アドレス・スペースに同時に存在する異なるユーザー (同一または別々の TCB) の複数セッションを、ユーザーが互いに認識していない場合でも、または互いに影響を与えることなく操作することができます。

クライアント・プログラムが別のクライアント・プログラムに接続されているときには、接続されたプログラムが独自の TCB 下で実行されます。

EXCI プログラミング・インターフェース

外部 CICS インターフェース (EXCI) には、EXCI CALL インターフェースと EXEC CICS インターフェースの 2 つの形式のプログラミング・インターフェースがあります。

EXCI CALL インターフェース

EXCI CALL インターフェースは、以下のアクションに使用できる 6 つのコマンドで構成されます。

- MVS で実行されている非 CICS プログラムから CICS システムへセッションを割り振り、オープンする
- これらのセッションで非 CICS プログラムから DPL 要求を発行する
- DPL 要求の完了時にセッションをクローズし、割り振り解除する

以下に、6 つの EXCI コマンドのリストを示します。

- Initialize-User
- Allocate_Pipe
- Open_Pipe
- DPL 呼び出し
- Close_Pipe
- Deallocate_Pipe

これらのコマンドと EXCI CALL の例について詳しくは、[12 ページの『EXCI CALL インターフェース』](#)を参照してください。

EXEC CICS インターフェース

EXEC CICS インターフェースには、いくつかのコマンドが用意されています。

例えば、このインターフェースには、1 回の呼び出しで EXCI CALL インターフェースの 6 つのコマンドをすべて実行する、単一の複合コマンド **EXEC CICS LINK PROGRAM** が用意されています。クライアント・アプリケーション・プログラムで **EXEC CICS LINK PROGRAM** コマンドを発行するたびに、外部 CICS インターフェースによって自動的に 6 個の EXCI 呼び出しそれぞれが起動されます。

EXEC CICS LINK PROGRAM コマンドは、CICS コマンド・レベル・アプリケーション・プログラミング・インターフェースの分散プログラム・リンク・コマンドと似ていますが、同じではありません。

EXCI には、チャンネル・コマンドとコンテナ・コマンドを使用してデータを処理する機能も用意されています。チャンネルは、そのコンテナのセットとともに、プログラム間でデータや情報を転送する際の通信域の使用に代わる方法として、**EXEC CICS LINK PROGRAM** コマンドまたは呼び出し **API DPL_REQUEST** で渡されます。

EXEC CICS インターフェースで使用するコマンドについて詳しくは、[36 ページの『EXCI EXEC CICS インターフェース』](#)を参照してください。

サーバー・プログラムの API 制約事項: 外部 CICS インターフェース 要求から呼び出される CICS サーバー・プログラムは、CICS アプリケーション・プログラミング・インターフェースの DPL サブセットに制限されます。API コマンドのこのサブセット (DPL サブセット) は、CICS 間のサーバー・プログラムの場合と同じです。サーバー・プログラムの DPL サブセットの詳細については、『[分散プログラム・リンク \(DPL\)](#)』を参照してください。

EXEC CICS か CALL インターフェースかの選択

CALL インターフェース (6 つのコマンドすべて) と EXEC CICS LINK コマンドを同じプログラムで併用して、別々の要求を実行できます。普通は、実動プログラムでこのようにすることはほとんどありません。

[EXCI サンプル・プログラム](#)には、さまざまな言語バージョンの、CICS 提供のサンプル・クライアント・プログラムが図示されています。

外部 CICS インターフェースには、形式ごとに特定の利点があります。

- 単一か頻度の低い DPL 要求の場合は、EXEC CICS LINK コマンドを使用することをお勧めします。

コーディングが簡単になるので、プログラミング・エラーの可能性が少なくなります。

EXEC CICS LINK コマンドを呼び出すたびに、外部 CICS インターフェースが CALL インターフェースのすべての関数を実行するので、結果として不要なオーバーヘッドが発生することに注意してください。

EXEC CICS LINK コマンドを使用して、別の LPAR 内の CICS サーバー領域と通信する場合には、このオーバーヘッドは非常に多くなることにも注意してください。この場合、EXEC CICS LINK コマンドを呼び出すたびに、必要な IRP のログオン、接続、切断、ログオフが行われるために多数の XCF のアクティビティが生成されます。同じ LPAR 内の CICS サーバー領域に対して複数の EXEC CICS LINK コマンドを発行している間の経過時間と比べた場合、別の LPAR 内の CICS サーバー領域に対してそれらの同じコマンドを発行する場合は、速度が低下して経過時間が大幅に長くなる可能性があります。

- 同一のクライアント・プログラムからの複数かつ頻度の高い DPL 要求の場合は、EXCI CALL インターフェースを使用することをお勧めします。

プログラムの開始時かその直後に Initialize_User コマンドと Allocate_Pipe コマンドを 1 回実行し、すべての DPL アクティビティの完了時に Deallocate_Pipe を 1 回実行するだけで済むので、こちらの選択肢の方が効率的です。これらの関数を実行している間に、必要に応じてパイプを開いたり閉じたりすることができ、パイプを開いている間に、必要な回数だけ DPL 呼び出しを発行できます。

外部 CICS CALL インターフェースの図

以下の 4 つの図は、EXCI CALL インターフェースを使用した外部 CICS インターフェースを示しています。

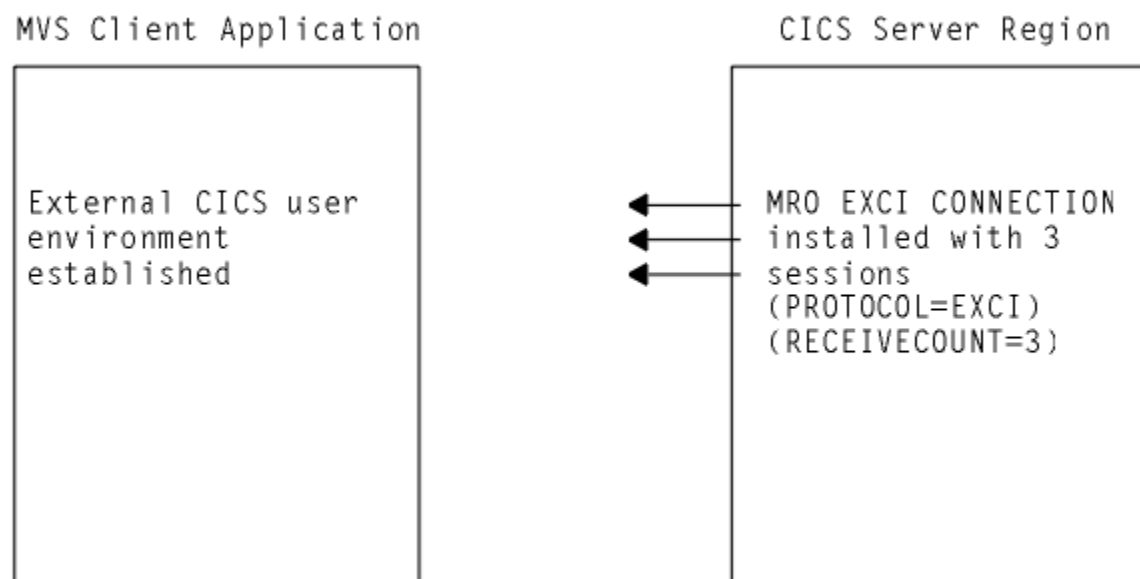


図 1. 第 1 段階: INITIALIZE_USER 呼び出し後の状況

注:

1. 5 ページの図 1 では、クライアント・アプリケーション・プログラムが INITIALIZE_USER 呼び出しを発行した時点で、ターゲットの CICS 領域は、IRC が開いており、3 つのセッションがある 1 つの EXCI 接続がインストールされている状態で稼働しています。
2. クライアント・アプリケーション・プログラムのアドレス・スペースは、EXCI ユーザー環境で初期化されています。この段階では MRO アクティビティはなく、パイプも存在しません。

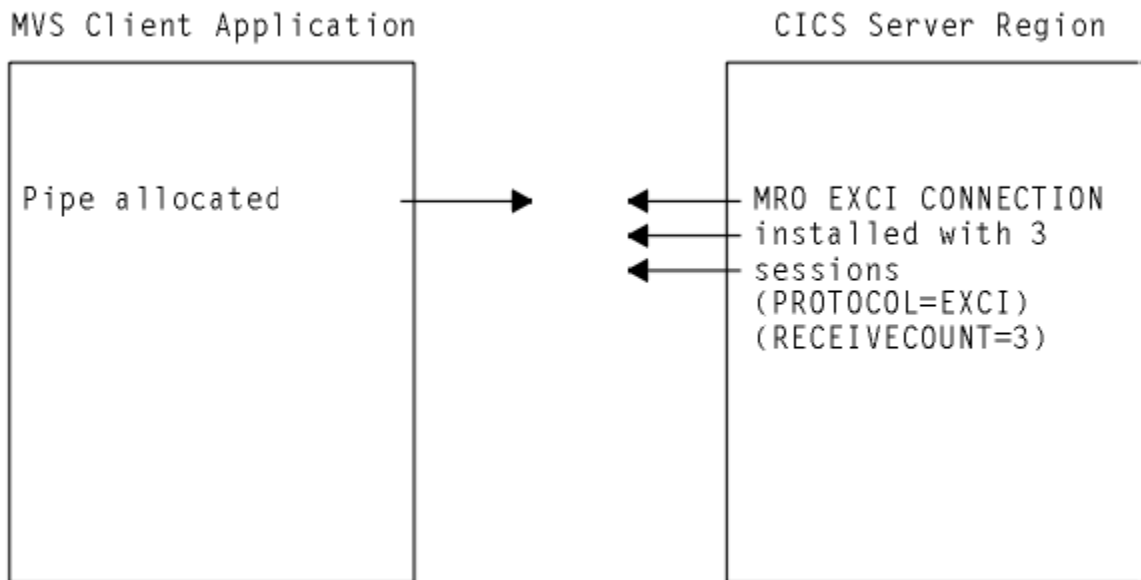


図 2. 第 2 段階: 最初の `ALLOCATE_PIPE` 呼び出し後の状況

注記: 6 ページの図 2 では、外部 CICS インターフェースは、ターゲット CICS サーバー領域を識別して、MRO にログインします。

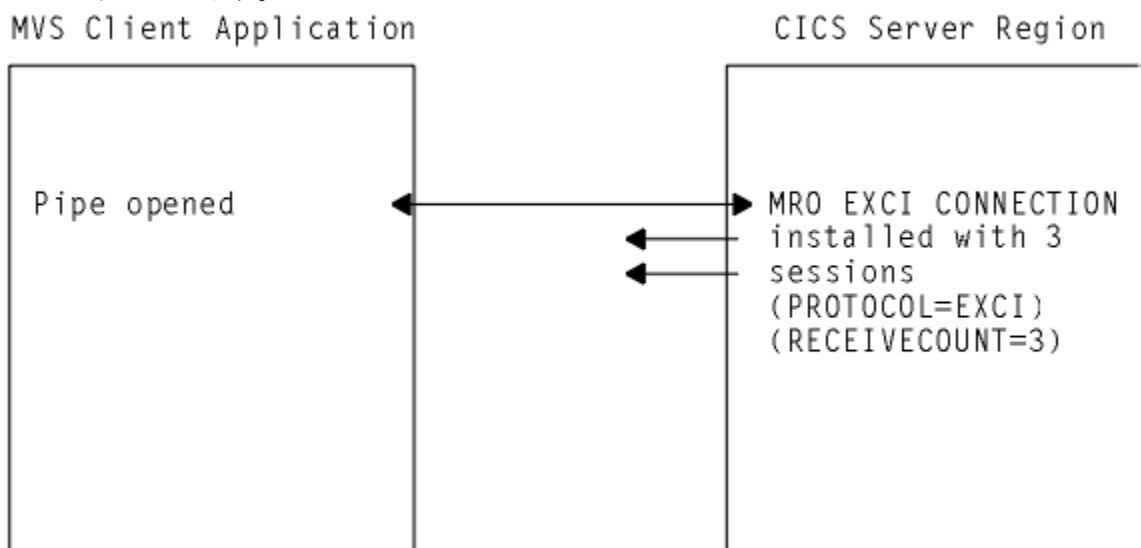


図 3. 第 3 段階: `OPEN_PIPE` 呼び出し後の状況

注:

1. 6 ページの図 3 では、外部 CICS インターフェースが CICS サーバー領域に接続し、パイプはこの時点で使用可能になっています。
2. 残りの 2 つの EXCI セッションは空いているので、同じクライアント・アプリケーション・プログラムや別のクライアント・アプリケーション・プログラム (接続が汎用の場合) からの別のパイプ・オープン要求で使用できます。

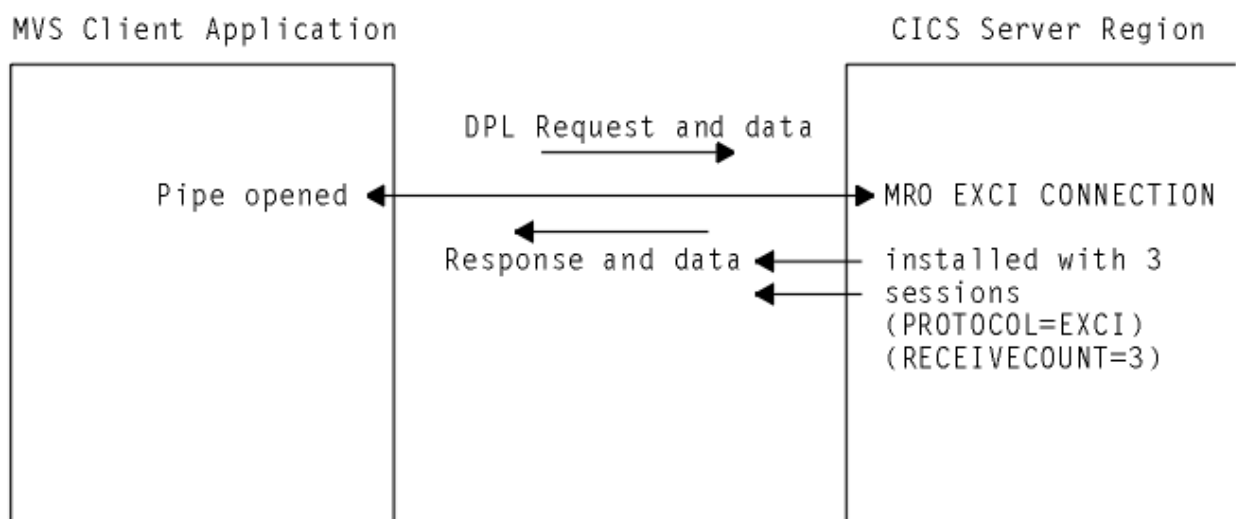


図 4. 第 4 段階: 1 つのパイプが開いており、DPL 呼び出しを処理している状況

注: 7 ページの図 4 では、外部 CICS インターフェースは、開いているパイプを介して DPL 要求と関連データを渡します。CICS サーバー領域は、開いているパイプを介して応答とデータを戻します。

パイプのクローズ: クライアント・アプリケーション・プログラムがパイプを閉じても、そのパイプは同じユーザーによる使用に備えて割り振られたままになるので、6 ページの図 2 で示されている状況になります。この段階で MRO セッションは、同じクライアント・アプリケーション・プログラムや別のクライアント・アプリケーション・プログラム (接続が汎用の場合) からの別のパイプ・オープン要求で使用できます。

パイプの割り振り解除: クライアント・アプリケーション・プログラムがパイプを割り振り解除すると、MRO からログオフされ、セッションに関連付けられているストレージがすべて解放されます。5 ページの図 1 に示されている状況になります。

EXCI EXEC CICS インターフェースの図

以下の図は、EXEC CICS インターフェースと、それが分割されて 6 つの EXCI CALL になる方法を示しています。

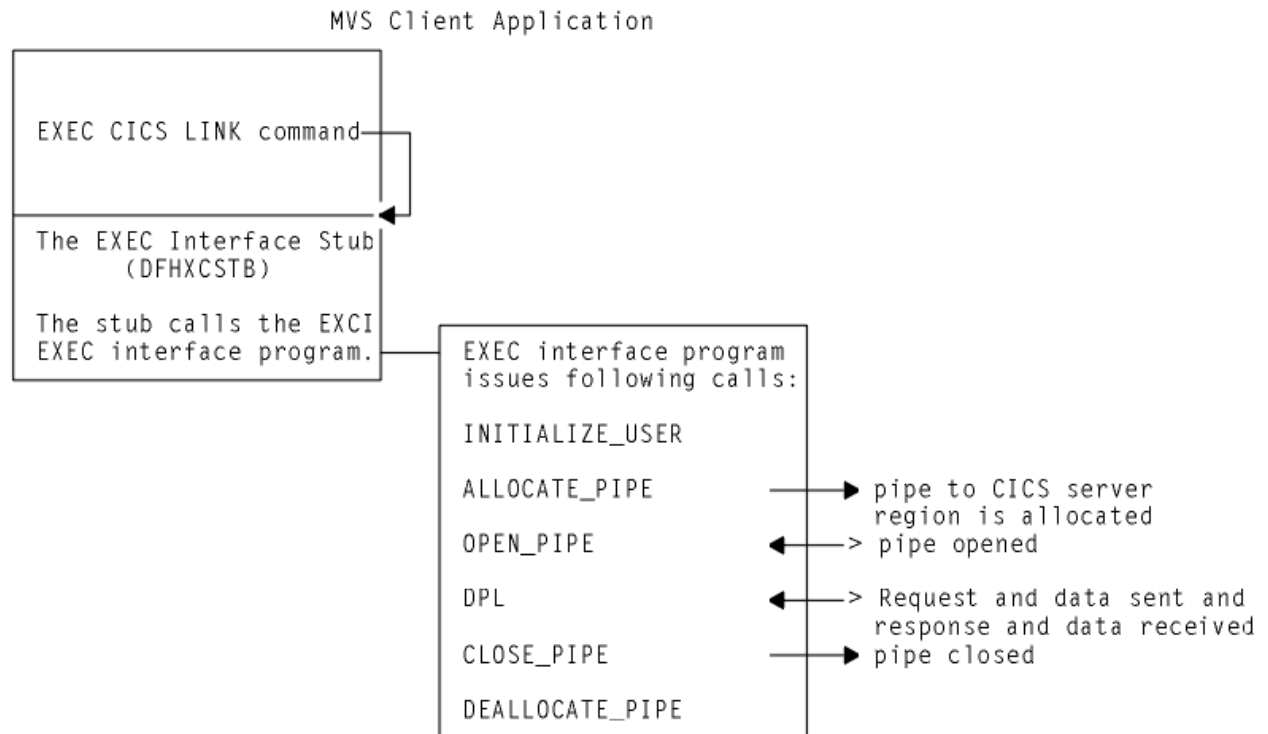


図 5. EXEC CICS コマンドを使用した外部 CICS インターフェースの図

リソース・リカバリ

リソース・リカバリは、プロトコルおよびプログラム・インターフェースで構成され、アプリケーション・プログラムが複数の保護リソースに一貫性のある変更を加えることを可能にします。外部 CICS インターフェースは、リソース・リカバリをサポートします。

外部 CICS インターフェース 要求によって呼び出される CICS サーバー・プログラムは、リカバリ可能リソースを更新できます。CICS サーバー領域のミラー・トランザクションが同期点を取るときに、変更がコミットされます。クライアント・プログラムは、同期点がいつ発生すればよいかを判別できます。以下の 2 つのオプションがあります。

- CICS サーバー領域で管理されるリソース・リカバリ。この場合は、リカバリ可能リソースに対する変更が、各 DPL 要求のコンパイル時にコミットされます。これは、クライアント・プログラムからは独立しています。また、サーバー・プログラムが制御を CICS に返すときに取られた同期点 (SYNCONRETURN オプション) に加えて、サーバー・プログラムは実行中に明示的な同期点を取ることができます。
- リカバリ可能リソース管理サービス (RRMS) のサポートを使用して EXCI クライアント・プログラムから管理されるリソース・リカバリ。クライアント・プログラムがこれを要求すると、後続の DPL 要求でサーバー・プログラムから実行された更新がまとめてコミットされます。

このオプションをサポートするには、CICS と外部 CICS インターフェースの両方が、リソース・リカバリ・サービス (RRS)、z/OS 同期点マネージャーを使用します。これは、リカバリ可能リソース・リカバリ・サービス (RRMS) の MVS コンポーネントです。¹ RRMS のコンテキストでは、CICS はリソース・マネージャーです。クライアント・プログラムは、他のリソース・マネージャーに要求を発行することができ、同じリカバリ単位 (UR) で、それらのリソース・マネージャー所有のリソースをコミットさせることができます。²

¹ RRMS は、登録サービス、コンテキスト・サービス、およびリソース・リカバリ・サービス (RRS) の 3 つの z/OS コンポーネントで構成されます。

これらのオプションは、以下のように制御されます。

- DPL_request の DPL_opts パラメーターによって。
- SYNCONRETURN オプションによって。EXEC CICS LINK PROGRAM コマンドで指定または省略されます。

SYNCONRETURN を指定すると、各 DPL 要求の完了時に同期点が取られます。SYNCONRETURN が省略されると、『11 ページの『クライアント・プログラムでの同期点の使用法』で説明しているインターフェースを使用して、クライアント・プログラムから要求されたときに、同期点が取られます。

外部 CICS インターフェースによる RRMS の使用法

z/OS リカバリー可能リソース管理サービス (RRMS) を使用して、分散プログラム・リンク (DPL) 要求を調整できます。

RRMS を使用して DPL 要求を調整するには、以下の条件を満たしていることを確認してください。

- EXCI クライアントと、DPL 要求の送信先の CICS 領域が、同じ MVS イメージ内で稼働している。これは、RRMS の制限なので、SYNCONRETURN を指定した DPL 要求には適用されません。
- DPL 要求を受け取る CICS 領域が、システム初期設定パラメーターとして **RRMS=YES** が指定されて開始されている (デフォルトは **RRMS=NO**)。
- リソース・リカバリー・サービス (RRS) が、CICS とクライアント・プログラムの実行場所の MVS イメージ内で実行されている。z/OS MVS プログラミング: リソース・リカバリーを参照してください。

次の図は、外部 CICS インターフェースと CICS が RRMS を使用方法を示しています。外部 CICS インターフェースおよび EXCI クライアント・プログラムが含まれる MVS バッチ領域と、CICS ミラーおよび CICS アプリケーション・プログラムが含まれる CICS サーバー領域間のフローが示されています。図中の番号は、図の後にリストされている、リカバリー単位 (UR) の基本ステップを指しています。

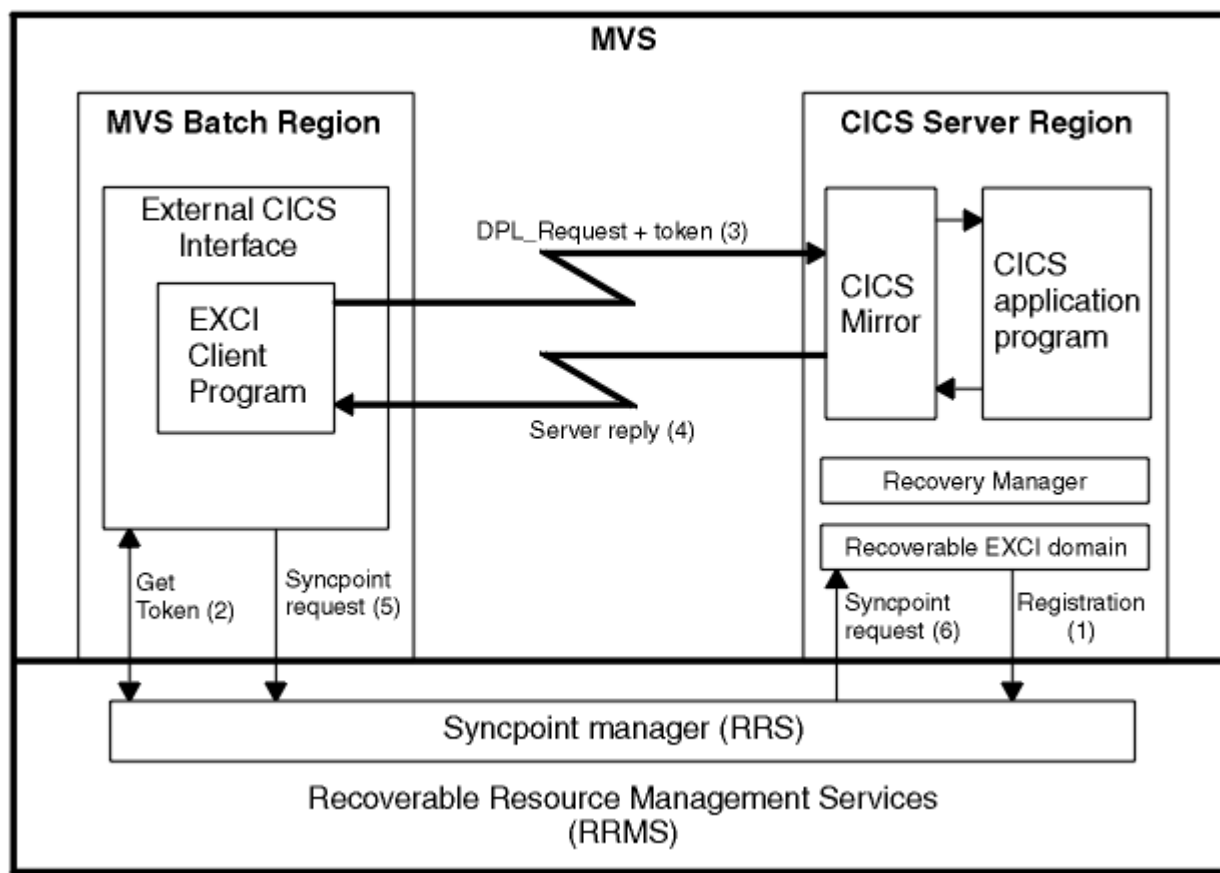


図 6. RRMS を使用する EXCI クライアントと CICS サーバー領域の概念ビュー

² リカバリー単位は、CICS の作業単位に類似しています。

1. CICS システム 初期設定パラメーター **RRMS=YES** が指定されている場合、CICS は、RRMS を持つリソース・マネージャーとして登録されます。この登録は、CICS の初期設定時に実行されます。
2. EXCI クライアント・プログラムが 2 フェーズ・コミット・モードで DPL_Request 呼び出し (SYNCONRETURN オプションを省略した呼び出し) を発行すると、RRMS から以下のものを受け取ります。
 - リカバリー単位 ID (URID)
 - コンテキスト・トークン
 - パス・トークン
3. EXCI がクライアント・プログラムのために取得した URID とトークンが、CICS サーバー領域に渡される DPL 要求に組み込まれます。この DPL 要求が UR 内の最初の要求である場合、CICS は RRS を呼び出して UR とのセッションを開始し、新しいミラー・トランザクションを付加し、トークンを妥当性検査します。要求が有効な場合は、ミラー・プログラムが、指定されたサーバー・アプリケーション・プログラムにリンクします。サーバー・プログラムは作業を完了しますが、これらの作業はすべてこの UR 内で実行されました。この作業には、ローカル・サーバー領域内でのリカバリー可能リソースの更新や、他の CICS 領域に対するデ이지ー・チェーンが含まれている場合があります。
4. サーバー・プログラムが完了すると、通信域 (COMMAREA) またはチャンネルとコンテナーを戻りコードとともにクライアント・プログラムに返します。

注: 同じ UR でステップ 3 と 4 を何回も繰り返す場合もあります。

5. EXCI クライアント・プログラムが変更のコミットやバックアウトを実行できる状態であるときには、プログラムは RRS を呼び出して 2 フェーズ・コミット・プロトコルを開始します。
6. RRS はコーディネーターの働きをして、以下のいずれかのアクションを完了します。
 - RRS からリソース・マネージャーに、UR 内のすべての更新をコミットする準備をするかどうかを尋ねます。CICS サーバー領域以外のリソース・マネージャーが UR とのセッションを開始する可能性もあります。すべて YES の場合、RRS は、作業を進めて変更をコミットするよう指示します。いずれかが NO と断定された場合は、RRS は UR 内で加えられたすべての変更をバックアウトすることをすべてのリソース・マネージャーに知らせます。
 - RRS は、UR とのセッションを開始するすべてのリソース・マネージャーに対して、UR 内で加えられたすべての変更をバックアウトするよう通知します。

この時点で、UR が完了し、CICS がミラー・タスクを切り離します。この時点の後で EXCI クライアントが新しい DPL 要求を送信する場合は、EXCI は新しい UR を開始し、CICS は新しいミラー・トランザクションを付加します。

SYNCONRETURN オプションを指定した各 DPL 要求は、ターゲットの CICS 領域内に新しいミラー・タスクを付加します。SYNCONRETURN を指定していない最初の DPL 要求も新しいミラー・タスクを付加しますが、後続の要求はこの同じミラー・タスクに送信されます。同期点が発生すると、ミラー・タスクが終了し、SYNCONRETURN を指定していない次の要求が新しいミラーを付加します。[11 ページの図 7](#) を参照してください。この図では、MVS クライアント・アプリケーションが、SYNCONRETURN を指定した DPL 要求と、SYNCONRETURN を指定しない DPL 要求を発行します。図中の番号は、図の後にリストされている、基本的な流れを指しています。

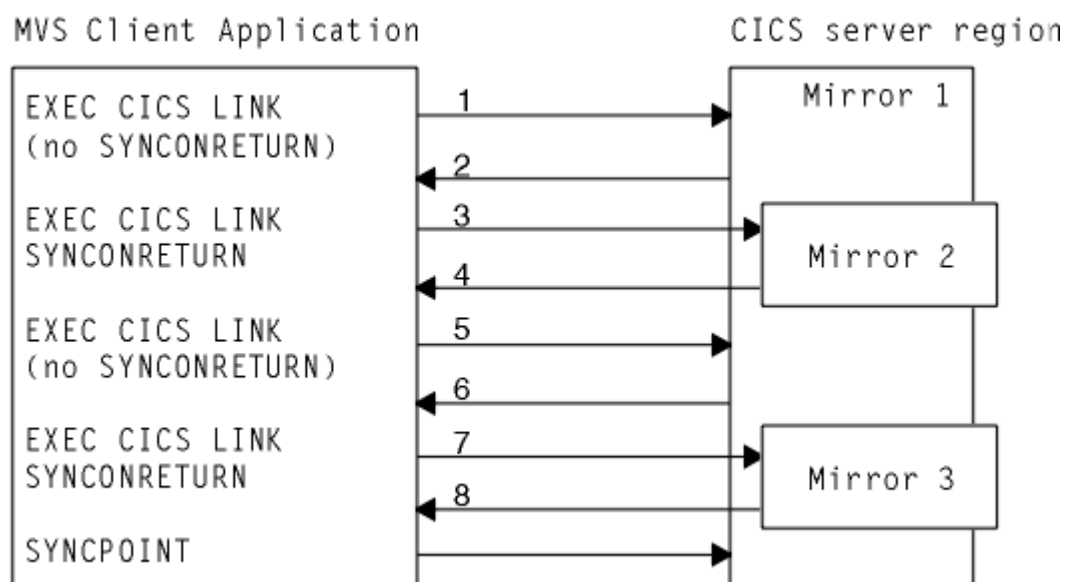


図 7. SYNCONRETURN オプションを指定した DPL 要求と SYNCONRETURN オプションを指定しない DPL 要求が混在する場合の影響

1. クライアントが、SYNCONRETURN オプションを指定しない DPL 要求を発行します。
ミラー・トランザクションが稼働していないので、新しいミラー (ミラー 1) が付加されます。
2. DPL 要求は完了し、SYNCONRETURN オプションを指定せずに発行したので、ミラー・トランザクションが別の要求を待機します。
3. クライアントが、SYNCONRETURN オプションを指定した DPL 要求を発行します。
新しいミラー・トランザクション (ミラー 2) が付加されます。
4. DPL 要求が完了すると、ミラー・トランザクションで更新されたリソースはコミットされ、ミラー・トランザクションは終了します。
5. クライアントが、SYNCONRETURN オプションを指定しない別の DPL 要求を発行します。ミラー 1 がこの DPL 要求を受け取って実行します。
6. DPL 要求が完了し、再度ミラー・トランザクションが別の要求を待機します。
7. クライアントが、SYNCONRETURN オプションを指定した DPL 要求を発行します。
新しいミラー・トランザクション (ミラー 3) が付加されます。
8. DPL 要求が完了すると、ミラー・トランザクションで更新されたリソースはコミットされ、ミラー・トランザクションは終了します。
9. クライアント・プログラムが同期点を要求します。ミラー 1 で更新されたリソースはコミットされ、トランザクションは終了します。

クライアント・プログラムでの同期点の使用法

クライアント・プログラムは、MVS 呼び出し可能サービスを使用して変更をコミットするかバックアウトするという方法で、同期点を取ることを要求できます。

クライアント・プログラムによって加えられた変更をコミットするには、以下のいずれかの MVS 呼び出し可能サービスを使用します。

Application_Commit_UR (SRRCMIT)

Application_Commit_UR の説明については、[z/OS MVS プログラミング: 高水準言語向け呼び出し可能サービス](#)を参照してください。

Commit_UR (ATRCMIT)

Commit_UR の説明については、[z/OS MVS プログラミング: リソース・リカバリー](#)を参照してください。

クライアント・プログラムによる変更をバックアウトするには、以下のいずれかの MVS 呼び出し可能サービスを使用します。

Application_Backout_UR (SRRBACK)

Application_Backout_UR の説明については、[z/OS MVS プログラミング: 高水準言語向け呼び出し可能サービス](#)を参照してください。

Backout_UR (ATRBK)

Backout_UR の説明については、[z/OS MVS プログラミング: リソース・リカバリー](#)を参照してください。

これらのインターフェースのいずれも 使用しない場合、変更は、クライアント・プログラムが正常終了するか異常終了するときに、明示的にコミットされるかバックアウトされます。以下の理由で、暗黙のコミットやバックアウトを使用することは勧められていません。

- 更新がコミットされたかバックアウトされたかを、クライアント・プログラムが知らせることができない。プログラムが正常に終了した場合でも、リソース・マネージャーが変更をバックアウトする可能性があります。
- 高水準言語のランタイム環境では、インターセプトしないとオペレーティング・システムの異常終了になるエラーをインターセプトする可能性がある。この種のエラーがインターセプトされ、クライアント・プログラムが明示的なアクションを取らないと、プログラムが正常に終了して更新がコミットされる可能性があります。このような状態でリソースが正しくコミットされたりバックアウトされたりするように、クライアント・プログラムをコーディングしてください。例えば、PL/I プログラムでは、エラーが検出されると **SRRBACK** コマンドを発行する ON 単位を組み込むことができます。同様に、COBOL プログラムでは、エラーが発生する可能性があるステートメントで ON 句を使用できます。

EXCI CALL インターフェース

EXCI CALL インターフェースは 6 つのコマンドから構成されます。これらのコマンドを使用すると、MVS で実行されている非 CICS プログラムから CICS システムにセッションを割り振って開き、これらのセッションに非 CICS プログラムから分散プログラム・リンク (DPL) 要求を発行し、DPL 要求の完了時にセッションを閉じて割り振りを解除することができます。このセクションでは、各 EXCI CALL コマンドについて説明し、EXCI CALL の例を示します。

アプリケーション・プログラム・スタブ DFHXCSTB

EXCI コマンドは、CICS 提供の DFHXCSTB と呼ばれるアプリケーション・プログラミング・スタブを介して外部 CICS インターフェースを呼び出します。非 CICS プログラムをリンク・エディットするときに、このスタブを組み込む必要があります。

EXCI CALL インターフェース・コマンド

以下の各コマンドの説明で、構文の記載箇所にはアセンブラ形式のコマンドが図示されています。コマンドごとの構文に VL,MF=(E,(1)) がありますが、これは実行形式の CALL マクロを示しており、パラメーター・リストのストレージ域がレジスター 1 でアドレス指定されていることを示しています。

コマンドは C、COBOL、および PL/I プログラミング言語でもサポートされており、これらの言語の該当する CALL 規則が使用されます。

CICS が提供するサンプル・クライアント・プログラムに、サポートされているすべての言語のこれらの CALL の例があります。詳しくは、[EXCI サンプル・プログラム](#)を参照してください。

Initialize_User

ユーザー環境を初期化します。これには、IRC 機能を使用するための権限の取得が含まれます。この環境は、TCB の存続期間に対して作成されるため、このコマンドは、各 TCB のユーザーごとに一度だけ発行する必要があります。このユーザーがこれ以降に発行するコマンドは、同じ TCB 下で発行する必要があります。

構文

```
CALL DFHXCIS,(version_number,return_area,user_token,call_type,  
              user_name),VL,MF=(E,(1))
```

パラメーター

version_number

使用される外部 CICS インターフェース・パラメーター・リストのバージョンを示すフルワード・パイナリー入力域。クライアント・プログラムでは 1 に設定されている必要があります。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこのパラメーターの等価値は、VERSION_1 です。コピーブックの詳細については、[33 ページの『戻り域と関数呼び出し EQUATE のコピーブック』](#)を参照してください。

return_area

応答コードと理由コード、およびメッセージ・ポインター・フィールドを受け取る 5 ワードの出力域。詳しくは、[33 ページの『EXCI CALL インターフェースの戻り域』](#)を参照してください。

user_token

CICS 外部インターフェースから提供される 32 ビット・トークンを格納する 1 ワードの出力域。クライアント・プログラムを表します。

ユーザー・トークンは、user-name パラメーターに対応します。クライアント・プログラムは、user_name パラメーターで定義されたユーザーについてこれ以降に発行されるすべての外部 CICS インターフェース・コマンドで、このトークンを渡す必要があります。

call_type

コマンドの機能を示す 1 ワードの入力域。クライアント・プログラムではこれを 1 に設定する必要があります。1 は、これが Initialize_User コマンドであることを示します。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこの呼び出しの等価値は、INIT_USER です。コピーブックの詳細については、[33 ページの『戻り域と関数呼び出し EQUATE のコピーブック』](#)を参照してください。

user_name

外部 CICS インターフェースのユーザーを識別する名前を保持する入力域。一般的には、これはクライアント・プログラムです。このユーザーが特定のパイプを使用する場合は、user_name の値が、特定のパイプに定義されている CONNECTION 定義の NETNAME 属性の値と一致する必要があります。

応答および理由コード

すべての応答コードがゼロ以外の場合は、固有の理由コードの値が応答の理由を示します。

注：応答コードと理由コードの数値はすべて 10 進数です。

以下は、外部 CICS インターフェースが Initialize_User 呼び出しで返すことのできる応答コードおよび理由コードの要約です。

応答 OK

コマンドは正常に実行されました (RC 0)。理由コード：

0

通常応答

応答 WARNING

コマンドは正常に実行されましたが、エラー (RC 4) が発生しました。理由コード：

3

VERIFY_BLOCK_FM_ERROR

Initialize_User 処理には、SSI 検査呼び出しのパラメーター・リストを作成するために、16MB 境界より下のストレージが必要です。この領域の FREEMAIN 中に、エラーが発生しました。

4

WS_FREEMAIN_ERROR

作業用ストレージを FREEMAIN しようとしたますが、結果として MVS FREEMAIN エラーが発生しました。

応答 RETRYABLE

セットアップ・エラーが原因でコマンドが失敗しましたが、再発行できます (RC 8)。理由コード:

201

NO_CICS_IRC_STARTED

前回の IPL 以降に IRC アクティビティがなかった MVS イメージに対して Initialize_User コマンドが発行されました。外部 CICS インターフェースは CICS SVC 番号を判別できません。

応答 USER_ERROR

クライアントかサーバーのいずれかのエラーによりコマンドが失敗しました (RC 12)。理由コード:

401

INVALID_CALL_TYPE

この EXCI 要求に、無効な呼び出しタイプ・パラメーター値が指定されています。

402

INVALID_VERSION_NUMBER

version_number パラメーターに、値 1 または 2 が指定されていません。

403

INVALID_USER_NAME

user_name パラメーターはすべてブランク文字 (X'40') で構成されています。

410

DFHMEBM_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、外部 CICS インターフェース・メッセージの発行準備においてメイン・メッセージ・モジュールをロードしようとしたますが、このモジュールのロードに失敗しました。

411

DFHMET4E_LOAD_FAILED

メッセージ・モジュール DFHMET4E のロードに失敗しました。Initialize_User 処理中に、外部 CICS インターフェースが、メッセージの発行準備においてメッセージ・テーブルをロードしようとした。このモジュールのロードに失敗しました。

412

DFHXCURM_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、ユーザー置き換え可能モジュール DFHXCURM をロードしようとした。このモジュールのロードに失敗しました。

413

DFHXCTRA_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、トラップ・モジュール (DFHXCTRA) をロードしようとした。このモジュールのロードに失敗しました。

419

CICS_AFCB_PRESENT

CICS または CICS バッチ共有データベースによって既に使用されている TCB に対して、Initialize_User 要求が出されました。外部 CICS インターフェースは CICS と TCB を共用できません。これにより、CICS アプリケーション・プログラムは EXCI 要求を出すことができません。

420

DFHXCOPT_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、オプション・モジュール DFHXCOPT をロードしようとした。このモジュールのロードに失敗しました。

421

RUNNING_UNDER_AN_IRB

EXCI 呼び出しが MVS IRB 下で発行されましたが、これは許可されません。

422

SERVER_ABENDED

DPL 要求の処理中に、CICS サーバー・アプリケーション・プログラムが、エラーを処理せずに異常終了しました。

423

SURROGATE_CHECK_FAILED

USERID パラメーターが指定された DPL 要求が出されました。

424

RRMS_NOT_SUPPORTED

SYNCONRETURN オプションが省略された DPL 要求が、OS/390® バージョン 2 リリース 5 以降で稼働していないシステムで出されました。

425

UOWID_NOT_ALLOWED

DPL 要求で SYNCONRETURN オプションが省略されましたが、UOWID の値が指定されました。このパラメーターの組み合わせは、DPL 要求では許可されません。

426

INVALID_TRANSID2

TRANSID2 パラメーター付きの DPL 要求が出されましたが、このパラメーターはすべてブランクです。

427

INVALID_CCSDID

CCSID パラメーター付きの DPL 要求が出されましたが、このパラメーターは無効値を指定しています。

428

INVALID_ENDIAN

エンディアン・パラメーター付きの DPL 要求が出されましたが、このパラメーターは無効値を指定しています。

429

DFHXCEIX_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、モジュール (DFHXCEIX) をロードしようとした。このモジュールのロードに失敗しました。

430

DFHXCPRX_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、モジュール (DFHXCPRX) をロードしようとした。このモジュールのロードに失敗しました。

応答 **SYSTEM_ERROR**

コマンドが失敗しました (RC 16)。理由コード:

601

WS_GETMAIN_ERROR

Initialize_User 処理中に、作業用ストレージの GETMAIN が失敗しました。

602

XCGLOBAL_GETMAIN_ERROR

Initialize_User 処理中に、重要な制御ブロック (XCGLOBAL) の GETMAIN が失敗しました。

603

XCUSER_GETMAIN_ERROR

Initialize_User 処理中に、ユーザー制御ブロック (XCUSER) の GETMAIN 要求が失敗しました。

605

VERIFY_BLOCK_GM_ERROR

Initialize_User 処理中に、EXCI 内部制御ブロックの GETMAIN が失敗しました。

606

SSI_VERIFY_FAILED

現在の CICS SVC 番号を取得するための MVS サブシステム・インターフェース (SSI) に対する VERIFY 呼び出しが失敗しました。

607

CICS_SVC_CALL_FAILURE

Initialize_User 処理中に、現在インストールされている CICS SVC に対する呼び出しが失敗しました。

622

ESTAE_SETUP_FAILURE

実行される可能性のあるプログラム・チェックから保護するために、外部 CICS インターフェースが MVS ESTAE を確立します。このケースでは、MVS ESTAE マクロが失敗しました。

623

ESTAE_INVOKED

呼び出し処理中にプログラム・チェックが実行され、ESTAE が呼び出されます。

627

INCORRECT_SVC_LEVEL

CICS SVC (DFHCSVC) のリリース・レベルが、外部 CICS インターフェースのリリース・レベルと同じ (またはそれ以上) ではありません。

応答コードについて詳しくは、32 ページの『EXCI 呼び出し応答コード値』を参照してください。

理由コードの詳細については、EXCI 呼び出しで返される応答と理由コードを参照してください。

Allocate_Pipe

単一セッションまたはパイプを CICS 領域に割り振ります。このコマンドでは、クライアント・プログラムは CICS 領域に接続されません。それは、**Open_Pipe** コマンドで行なわれます。

EXCI アドレス・スペースで最大 250 個のパイプを割り振ることができます。デフォルトの制限はパイプ 100 個です。ただし、CICS を MVS サブシステムとして定義する場合は、**LOGONLIM** パラメーターを使用してこれを増やすことができます。『インストール』の『EXCI パイプ割り振り』を参照してください。

この制限の設定により、EXCI クライアントで MRO リソースが独占されなくなり、CICS システムで MRO の使用が妨げられることがあります。この制限は、MRO 環境と、システム間 MRO (XCF / MRO) 環境の両方に適用されます。

ALLOCATE_PIPE 要求が出されると、結果として MRO LOGON 要求が出されます。すべてのアドレス・スペースで許可される MRO LOGON 要求の総数には制限があります。これは、XCF/MRO を使用するとき特に重要です。XCF グループ内のメンバー数の制限により、MRO LOGON 要求の総数も制限されます。

構文

```
CALL DFHXCIS,(version_number,return_area,user_token,call_type,  
              pipe_token,CICS applid,allocate_opts),VL,MF=(E,(1))
```

パラメーター

version_number

使用される外部 CICS インターフェース・パラメーター・リストのバージョンを示すフルワード・バイナリー入力域。クライアント・プログラムでは 1 に設定されている必要があります。

CICS 提供のコピーブック DFHXCPLx(ここで x は言語を示す)におけるこのパラメーターの等価値は、VERSION_1 です。コピーブックの詳細については、[34 ページの表 3](#) を参照してください。

return_area

応答コードと理由コード、およびメッセージ・ポインター・フィールドを受け取る 5 ワードの出力域。詳細については、[33 ページの『EXCI CALL インターフェースの戻り域』](#)を参照してください。

user_token

Initialize_User コマンドで返される 1 ワード・トークン。

call_type

コマンドの機能を示す 1 ワードの入力域。クライアント・プログラムでは、これを 2 に設定する必要があります。2 は、これが **Allocate_Pipe** コマンドであることを示します。

CICS 提供のコピーブック DFHXCPLx(ここで x は言語を示す)におけるこの呼び出しの等価値は、ALLOCATE_PIPE です。コピーブックの詳細については、[34 ページの表 3](#) を参照してください。

pipe_token

1 ワードの出力域。CICS はこの領域に、割り振られたセッションを表す 32 ビット・トークンを返します。このトークンは、このセッションを使用する後続のコマンドで使用する必要があります。

CICS_applid (または null_ptr)

割り振られたセッションの接続先となる CICS システムのアプリケーション ID を格納する 8 バイトの入力域。

Allocate_Pipe 関数を完了するためにはアプリケーション ID が必要ですが、**Allocate_Pipe** 呼び出しではこのパラメーターはオプションです。アプリケーション ID は、**Allocate_Pipe** 呼び出しに対するこのパラメーターで指定することも、ユーザー置換可能モジュール DFHXCURM (DFHXCURM は必ず **Allocate_Pipe** 処理中に呼び出されます) で **URMCICS** パラメーターを使用して指定することもできます。また、DFHXCURM で **URMCICS** パラメーターを使用すると、**Allocate_Pipe** 呼び出しで指定されたアプリケーション ID を指定変更することもできます。**URMCICS** パラメーターについて詳しくは、『[EXCI ユーザー置換可能モジュール](#)』を参照してください。

呼び出しでアプリケーション ID を省略する場合は、CALL パラメーター・リストに **CICS_applid** のヌル・アドレスが含まれていることを確認する必要があります。この処理をどのように行うかは、ユーザーが CICS 以外のクライアント・プログラムで使用する言語によって異なります。オプション・パラメーターを省略する呼び出しの例については、[35 ページの『ヌル・パラメーターを持つ EXCI CALL の例』](#)を参照してください。

allocate_opts

このコマンドで指定されたオプションを表す 1 バイトの入力域。オプションでは、使用するセッションのタイプ (特定または汎用) を指定します。X'00' 特定のセッションを表します。X'80' は、汎用セッションを表します。

CICS 提供のコピーブック DFHXCPLx(ここで x は言語を示す)におけるこの呼び出しの等価値は、SPECIFIC_PIPE および GENERIC_PIPE です。コピーブックの詳細については、[34 ページの表 3](#) を参照してください。

応答および理由コード

すべての応答コードがゼロ以外の場合は、固有の理由コードの値が応答の理由を示します。

注: 応答コードと理由コードの数値はすべて 10 進数です。

以下は、Allocate_Pipe 呼び出しで外部 CICS インターフェースから返される可能性のある応答コードおよび理由コードの要約です。

応答 OK

コマンドは正常に実行されました (RC 0)。理由コード:

0

通常応答

応答 **USER_ERROR**

クライアントかサーバーのいずれかのエラーによりコマンドが失敗しました (RC 12)。理由コード:

401

INVALID_CALL_TYPE

402

INVALID_VERSION_NUMBER

404

INVALID_USER_TOKEN

421

RUNNING_UNDER_AN_IRB

応答 **SYSTEM_ERROR**

コマンドが失敗しました (RC 16)。理由コード:

604

XCPIPE_GETMAIN_ERROR

608

IRC_LOGON_FAILURE

622

ESTAE_SETUP_FAILURE

623

ESTAE_INVOKED

628

IRP_LEVEL_CHECK_FAILURE

応答コードについて詳しくは、[32 ページの『EXCI 呼び出し応答コード値』](#)を参照してください。

理由コードの詳細については、[EXCI 呼び出しで返される応答と理由コード](#)を参照してください。

Open_Pipe

IRC によって、割り振られたパイプが受信セッションに接続されます。

Open_Pipe により、IRC は、割り振られたパイプを、Allocate_Pipe コマンドまたは DFHXCURM で指定された CICS 領域に定義されている適切な接続の受信セッションに接続します。適切な接続は以下のいずれかです。

- Initialize_User コマンドの *user_name* パラメーターと等しい NETNAME 値を持つ EXCI 接続 (つまり、このクライアント・プログラム専用の特定接続を使用している)
- 汎用として定義されている EXCI 接続

XCF 環境では、Open_Pipe コマンドにより、領域間通信プログラム DFHIRP が、要求を受け取る LPAR に接続します。この要求は非同期であるため、Open_Pipe コマンドが正常な戻りコードを受け取っても、それ以降の DPL_Request 呼び出しは失敗する可能性があります。

提供されているシャットダウン支援ランザクション (CESD) またはそれと同等の機能のサポートなしで CICS をシャットダウンし、セッションが開いたままになっている場合、CICS はその IRC 機能を順序正しく閉じることができない可能性があります。EXCI セッションが閉じていない場合、シャットダウン支援ランザクションのサポートなしの CICS の通常シャットダウンは待機します。CICS は、以下の情報を示すメッセージ DFHIR2321 を発行します。

- セッションが特定接続上にある場合、そのネット名
- オープン・セッションが汎用接続上にある場合、GENERIC という語

用意されているシャットダウン支援ランザクション CESD を使用する場合は、CESD が IRC の即時クローズを発行するため、開いたままのセッションから、通常シャットダウンに対する問題は提示されません。このセッションで 1 つ以上の DPL_Request 呼び出しが発行された場合は、メッセージ DFHIR2321 に、以下の情報がさらに示されます。すなわち、セッションを使用しているクライアント・ジョブのジョブ名、ステップ名、プロシージャ名、およびクライアント・プログラムが実行されている MVS イメージの MVS ID です。

構文

```
CALL DFHXCIS,(version_number,return_area,user_token,call_type,  
             pipe_token),VL,MF=(E,(1))
```

パラメーター

version_number

使用される外部 CICS インターフェース・パラメーター・リストのバージョンを示すフルワード・バイナリー入力域。クライアント・プログラムでは 1 に設定されている必要があります。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこのパラメーターの等価値は、VERSION_1 です。コピーブックの詳細については、トピック『[34 ページの表 3](#)』を参照してください。

return_area

応答コードと理由コード、およびメッセージ・ポインター・フィールドを受け取る 5 ワードの出力域。詳細については、[33 ページの『EXCI CALL インターフェースの戻り域』](#)を参照してください。

user_token

Initialize_User コマンドで返される 1 ワード・トークン。

call_type

コマンドの機能を示す 1 ワードの入力域。クライアント・プログラムでは、これを 3 に設定する必要があります。3 は、これが Open_pipe コマンドであることを示します。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこの呼び出しの等価値は、OPEN_PIPE です。コピーブックの詳細については、トピック『[34 ページの表 3](#)』を参照してください。

pipe_token

Allocate_Pipe コマンドで CICS によって渡されたトークンを含む 1 ワードの出力域。このコマンドで開かれているパイプを表します。

応答および理由コード

すべての応答コードがゼロ以外の場合は、固有の理由コードの値が応答の理由を示します。

注：応答コードと理由コードの数値はすべて 10 進数です。

以下は、Open_Pipe 呼び出しで外部 CICS インターフェースから返される可能性のある応答コードおよび理由コードの要約です。

応答 OK

コマンドは正常に実行されました (RC 0)。理由コード：

0
NORMAL

応答 WARNING

コマンドは正常に実行されましたが、エラー (RC 4) が発生しました。理由コード：

1
PIPE_ALREADY_OPEN

応答 RETRYABLE

セットアップ・エラーが原因でコマンドが失敗しましたが、再発行できます (RC 8)。理由コード：

202
NO_PIPE

203
NO_CICS

応答 USER_ERROR

クライアントかサーバーのいずれかのエラーによりコマンドが失敗しました (RC 12)。理由コード：

401
 INVALID_CALL_TYPE
402
 INVALID_VERSION_NUMBER
404
 INVALID_USER_TOKEN
418
 INVALID_PIPE_TOKEN
421
 RUNNING_UNDER_AN_IRB

応答 **SYSTEM_ERROR**

コマンドが失敗しました (RC 16)。理由コード:

609
 IRC_CONNECT_FAILURE
621
 PIPE_RECOVERY_FAILURE
622
 ESTAE_SETUP_FAILURE
623
 ESTAE_INVOKED

応答コードについて詳しくは、[32 ページの『EXCI 呼び出し応答コード値』](#)を参照してください。

理由コードの詳細については、[EXCI 呼び出しで返される応答と理由コード](#)を参照してください。

DPL_Request

サーバー (またはターゲット) アプリケーション・プログラムが存在する CICS システムに接続されているオープン・パイプを介して、分散プログラム・リンク要求を出します。

このコマンドは同期であり、TCB は CICS からの応答を待機します。パイプを開いた後、パイプを閉じるまでに、任意の数の DPL 要求を出すことができます。サーバー・プログラムでは、リンク要求が別の CICS 領域からの標準の **EXEC CICS LINK** 要求と同じように認識され、EXCI を使用して非 CICS クライアント・プログラムから送られたものであることは認識されません。

呼び出しの構文は、*version_number* が **VERSION_1** に設定されている場合に使用できるパラメーター、*version_number* が **VERSION_2** に設定されている場合に使用できるパラメーター、および *version_number* が **VERSION_3** に設定されている場合に使用できるパラメーターの 3 つの形式で示されます。

構文

VERSION_1

```
CALL DFHXCIS, (version_number, return_area, user_token, call_type,
               pipe_token, pgmname, COMMAREA, COMMAREA_len, data_len,
               transid, uowid, userid, dpl_retarea, DPL_opts), VL, MF=(E, (1))
```

VERSION_2

```
CALL DFHXCIS, (version_number, return_area, user_token, call_type,
               pipe_token, pgmname, COMMAREA, COMMAREA_len, data_len,
               transid, uowid, userid, dpl_retarea, DPL_opts,
               transid2, ccsid, endian), VL, MF=(E, (1))
```

VERSION_3

```
CALL DFHXCIS, (version_number, return_area, user_token, call_type,
```

```
pipe_token,pgmname,CHANNEL,0,0,  
transid,uowid,userid,dpl_retarea,DPL_opts,  
transid2,0,0),VL,MF=(E,(1))
```

パラメーター

version_number

使用される外部 CICS インターフェース・パラメーター・リストのバージョンを示すフルワード・バイナリー入力域。クライアント・プログラムでは、これを 1、2、または 3 に設定できます。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこのパラメーターの等価値は、VERSION_1、VERSION_2、または VERSION_3 のいずれかです。コピーブックの詳細については、[33 ページの『戻り域と関数呼び出し EQUATE のコピーブック』](#)を参照してください。

return_area

応答コードと理由コード、およびメッセージ・ポインター・フィールドを受け取る 5 ワードの出力域。詳細については、[33 ページの『EXCI CALL インターフェースの戻り域』](#)を参照してください。

user_token

Initialize_User コマンドでクライアント・プログラムに返されるユーザー・トークンを指定する 1 ワードの入力域。

call_type

コマンドの機能を示す 1 ワードの入力域。クライアント・プログラムでは、これを 6 に設定する必要があります。6 は、パイプが現在 DPL_Request 呼び出しに使用されていることを示します。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこの呼び出しの等価値は、DPL_REQUEST です。コピーブックの詳細については、[33 ページの『戻り域と関数呼び出し EQUATE のコピーブック』](#)を参照してください。

pipe_token

Allocate_Pipe コマンドで EXCI から返されるトークンを指定する 1 ワードの入力域。

DPL_Request 呼び出しに使用されているパイプを表します。

pgmname

サーバー・プログラムとして呼び出される CICS アプリケーション・プログラムの 8 文字の名前。

これは、CICS サーバー領域にインストールされている事前定義 PROGRAM リソース定義に指定されている名前、またはプログラムが自動インストールされる場合はユーザー作成の自動インストール・プログラムに認識される名前です。プログラムは、CICS サーバー領域でローカル・プログラムとして定義することも、リモートとして定義することもできます。リモートとして定義されたプログラムでは、デージー・チェーンが有効になり、EXCI-CICS DPL 呼び出しが EXCI-CICS-CICS DPL 呼び出しになります。

COMMAREA (または null_ptr)

クライアントとサーバー・プログラム間の通信域 (COMMAREA) の可変長入力域。長さは COMMAREA_len で定義します。

これは、CICS アプリケーション・プログラムに送られるデータを含むストレージ域です。この領域は、更新された COMMAREA を CICS アプリケーション・プログラム (サーバー・プログラム) から受け取る目的でも使用されます。

このパラメーターはオプションです。これが不要でない場合は、CALL パラメーター・リストにこのパラメーターのヌル・アドレスが含まれていることを確認する必要があります。この処理をどのように行うかは、ユーザーが CICS 以外のクライアント・プログラムで使用する言語によって異なります。オプション・パラメーターを省略する呼び出しの例については、[35 ページの『ヌル・パラメーターを持つ EXCI CALL の例』](#)を参照してください。

COMMAREA_len

フルワード・バイナリー入力域。このパラメーターは、COMMAREA の長さを指定します。また、サーバー・プログラムの COMMAREA (EIBCALEN) の長さでもあります。

COMMAREA を指定する場合は、長さを定義するために、このパラメーターも指定する必要があります。

COMMAREA を 2 つの CICS サーバーの間で受け渡す場合 (製品/バージョン/リリースがどんな組み合わせであっても)、24 KB を超える値を指定することはできません。それ以外の場合は、COMMAREA がそれ以降の LINK 要求で渡されないことが確実であるときに、最大 32 KB の長さの COMMAREA を使用できます。

COMMAREA を指定しなかった場合は、このパラメーターは無視されます。

data_len

フルワード・バイナリー入力域。このパラメーターは、サーバー・プログラムに渡されるストレージの連続域 (COMMAREA の先頭以降) の長さを指定します。

このパラメーターは、サーバー・プログラムに送られるデータ量を制限します。例えば、COMMAREA は大きいものの、渡されるデータ量が少ない場合に、パフォーマンスを最適化する目的で使用します。

サーバー・プログラムから戻るときに、EXCI データ変換プログラムは、非 CICS クライアント・プログラム内の COMMAREA がサーバー・プログラムの COMMAREA と同じになるようにします。これは以下の条件に対処します。

- 返されるデータが、元の COMMAREA で渡されたデータより多い。
- 返されるデータが、元の COMMAREA で渡されたデータより少ない。
- 未変更であるため、返されるデータがない。
- サーバーがヌル・データを返す。

data_len の値は、COMMAREA_len の値を超えてはなりません。値ゼロも有効で、ゼロに設定するとデータがサーバー・プログラムに送られなくなります。

COMMAREA を指定しなかった場合は、このパラメーターは無視されます。

CHANNEL

呼び出し先プログラムで使用可能にするチャンネルの名前が含まれる 16 文字の入力域。許容文字は、A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - および _ です。空白文字を先頭にしたり、埋め込んだりすることはできません。指定された名前が 16 文字よりも短い場合は、16 文字になるまで末尾に空白が埋め込まれます。チャンネルが存在していない場合は、そのチャンネルが作成されます。EXCI クライアントには、LINK レベルが 1 つしかないため、このチャンネルは有効範囲に入ったままになります。チャンネルの有効範囲について詳しくは、[チャンネルの有効範囲](#)を参照してください。

チャンネル名は常に EBCDIC です。前述の、チャンネル名に使用できる文字セットには、EBCDIC コード・ページによっては異なる表示となる文字も一部含まれています。したがって、領域間でチャンネルをシッップする場合は、名前に使用する文字を A-Z a-z 0-9 & : = , ; < > . に制限することを推奨します。- および _ です。

チャンネル名 DFHTRANSACTION を指定してトランザクション・チャンネルを使用できます。CICS では、リンク・レベルが変更されても、トランザクション・チャンネルは有効範囲から外れず、そのトランザクションで常にアクセス可能です。詳しくは、[チャンネルおよびコンテナ](#)を参照してください。

DPL_REQUEST コマンドを発行するプログラムでは、コマンドでチャンネルの名前を指定できます。指定されたチャンネルは、既に存在している場合がありますが、プログラムで 1 つ以上の **PUT CONTAINER** コマンドを使用して作成されます。あるいは、プログラムで、現在存在していないチャンネルの名前を指定することもできます。この場合は、新しい空のチャンネルが作成されます。

注: CHANNEL に続く 2 つのパラメーターは、ヌルにする必要があります。さらに、**ccsid** パラメーターと **endian** パラメーターをヌルにする必要があります。

transid (または null_ptr)

サーバー・プログラムが実行される CICS ミラー・トランザクションの ID が入る 4 文字の入力域。このトランザクションは、CICS サーバー領域に対して定義する必要があり、その定義では以下を指定する必要があります。

PROGRAM(DFHMIRS)

初期プログラムは、CICS 提供のミラー・プログラム DFHMIRS である必要があります。

DFHMIRS を初期プログラムとして指定できなかった場合は、クライアント・アプリケーション・プログラムから渡される COMMAREA が、CICS サーバー・プログラムに渡されません。また、DPL

要求が失敗し、クライアント・アプリケーション・プログラムは応答として SYSTEM_ERROR、理由として SERVER_PROTOCOL_ERROR を受け取ります。

PROFILE(DFHICCSA)

DFHCICSA プロファイルは、INBFMH パラメーターに正しい値を指定します。このパラメーターは、ミラー・トランザクションに対する INBFMH(ALL) として指定する必要があります。

CICS サーバー領域が DPL 要求を受け取ったら、ミラー・トランザクションを接続し、DFHMIRS を呼び出します。その後、ミラー・プログラムが、要求されたサーバー・プログラムに制御を渡し、クライアント・プログラムから提供される COMMAREA を渡します。サーバー・プログラムに渡される COMMAREA にはデータのみが取り込まれ、COMMAREA の残りの部分はヌルに設定されます。

transid パラメーターの目的は、サーバー・プログラムの異なる呼び出しを区別することです。これにより、異なる属性を指定するトランザクションのもとで、さまざまな方法で呼び出されたサーバー・プログラムを実行できます。例えば、トランザクションの優先順位やセキュリティ要件をさまざまな設定できます。

transid はオプションです。デフォルトでは、CICS サーバー領域は、CICS 提供のミラー・トランザクション CSMI を使用します。transid を指定しない場合は、CALL パラメーター・リストに、このパラメーターのヌル・アドレスが含まれていることを確認する必要があります。この処理をどのように行うかは、ユーザーが CICS 以外のクライアント・プログラムで使用する言語によって異なります。オプション・パラメーターを省略する呼び出しの例については、[35 ページの『ヌル・パラメーターを持つ EXCI CALL の例』](#)を参照してください。

同じ MVS リカバリー単位で複数の要求を発行する場合は、すべての要求で同じ transid を使用する必要があります。

uowid (または null_ptr)

APPC 体系形式を使用する作業単位 ID (相関目的で DPL_Request で渡される) が入る入力域。

CICS プログラムが MVS アプリケーションに制御を返すときにコミットされる DPL 要求の場合は、このパラメーターはオプションです。

RRMS リカバリー単位の一部である DPL 要求の場合は、null_ptr を指定する必要があります。RRMS リカバリー単位に既に関連付けられている作業単位 ID がある場合は、それが使用されます。そのような ID がない場合は、CICS (または別の RRMS リソース・マネージャー) によって固有のリカバリー単位 ID が生成され、RRMS リカバリー単位に関連付けられます。

uowid パラメーターを指定しない場合は、CALL パラメーター・リストにこのパラメーターのヌル・アドレスが含まれていることを確認する必要があります。この処理をどのように行うかは、ユーザーが CICS 以外のクライアント・プログラムで使用する言語によって異なります。オプション・パラメーターを省略する呼び出しの例については、[35 ページの『ヌル・パラメーターを持つ EXCI CALL の例』](#)を参照してください。

uowid パラメーターが CICS サーバー領域に渡され、CICS サーバー・プログラムによって実行される最初の作業単位の UOWID として使用されます。サーバー・プログラムがクライアント・プログラムに戻る前に中間同期点を発行する場合、CICS は、提供された uowid を後続の作業単位に使用しますが、新しい論理作業単位ごとに増分される 2 バイトのシーケンス番号が付されます。CICS サーバー・プログラムがリモート・リソースを更新する場合は、クライアント提供の UOWID が、リソースを所有するリモート・システムに配布されます。

uowid パラメーターは、相関目的のみのために EXCI CALL インターフェースで提供され、これにより特定のクライアント・プログラムから発生した作業単位を CICS で識別できます。CICS とクライアント・プログラム間で、uowid がリカバリー目的で提供されることはありません。

uowid の長さは最長で 27 バイトで、以下の形式をとります。

- UOWID 全体の長さ (このフィールドを除く) が入る 1 バイトの長さフィールド。
- 論理装置名の長さ (このフィールドを除く) が入る 1 バイトの長さフィールド。
- 最大 17 バイトの可変長の論理装置名フィールド。

APPC アーキテクチャー規則に準拠させるには、LUNAME が AAAAAAAAA.BBBBBBBB という形式である必要があります。ここで AAAAAAAAA はオプションで、さらに以下の規則があります。

- AAAAAAAAA および BBBBBBBB が、ピリオドで区切った 8 バイト名である
 - AAAAAAAAA を省略する場合は、ピリオドも省略する必要がある
 - AAAAAAAAA および BBBBBBBB が、タイプ 1134 のシンボル・ストリングである必要がある (つまり、文字ストリングが 1 文字以上の EBCDIC 大文字の A-Z および 0-9 で構成され、その先頭文字は大文字である必要がある)
 - クロック値 (8 バイトのストア・クロック (STCK) 値の中央の 6 バイト)
 - 2 バイトのシーケンス番号が付く
- 作業単位 ID を省略 (NULL ポインターを指定) し、DPL 要求が RRMS リカバリー単位の一部でない場合は、外部 CICS インターフェースによって自動的に ID が生成され、以下のように構成されます。
- X'1A' に設定される 1 バイトの長さフィールド
 - X'11' に設定される 1 バイトの LU 長フィールド
 - 以下のように構成される 17 バイトの LU 名:
 - DFHEXCIU に設定される 8 バイトの目印。
 - ピリオド (.) が入る 1 バイト・フィールド。
 - クライアントが稼働する MVS システムを表す文字の ID (SYSID) が入る 4 バイト・フィールド。
 - MVS クライアント・プログラムが実行されるアドレス・スペースの ID (ASID) が入る 4 バイト・フィールド。このフィールドには、2 バイトの 16 進アドレス・スペース ID を表す 4 文字の EBCDIC が入ります。
 - クロック値 (8 バイトのストア・クロック (STCK) 値の中央の 6 バイト)
 - X'0001' に設定される 2 バイトのシーケンス番号。

userid (または null_ptr)

CICS 領域内のユーザー・セキュリティ検査のための RACF® ユーザー ID が入る 8 文字の入力域。外部 CICS インターフェースは、サーバー・アプリケーション・プログラム内のリソースおよびコマンドのユーザー・セキュリティ検査のために、CICS サーバー領域にこのユーザー ID を渡します。

MRO 接続に ATTACHSEC(IDENTIFY) 属性が指定されている場合にのみ、ユーザー ID が必要です。この接続に ATTACHSEC(LOCAL) が指定されている場合、CICS サーバー領域はリンク・セキュリティ検査を適用します。MRO 接続のリンク・セキュリティについては、『[相互通信セキュリティ](#)』を参照してください。

外部 CICS インターフェース・セキュリティの考慮事項について詳しくは、『[EXCI セキュリティ](#)』も参照してください。

このパラメーターはオプションです。ただし、ユーザー ID を指定しなかった場合は、外部 CICS インターフェースが、クライアント・プログラムの実行時に使用されたセキュリティ・ユーザー ID を渡します。例えば、クライアント・プログラムが MVS バッチ・ジョブとして実行されている場合は、外部 CICS インターフェースが、JOB ステートメントの USER パラメーターに指定されたユーザー ID を取得して渡します。

ユーザー ID を指定したときに、EXCI オプション・テーブル DFHXCOPT に SURROGCHK=YES が指定されていた場合は、EXCI ジョブを実行しているユーザー ID が、代理ユーザー検査の対象になります。この検査は、外部 CICS インターフェースによって実行されます。これにより、クライアント・ジョブのユーザー ID は、DPL 呼び出しに指定されたユーザー ID を使用することが許可されます。代理ユーザー・セキュリティ検査の詳細については、『[EXCI セキュリティ](#)』を参照してください。

userid をデフォルトにするには、CALL パラメーター・リストにこのパラメーターのヌル・アドレスが含まれていることを確認する必要があります。この処理をどのように行うかは、ユーザーが CICS 以外のクライアント・プログラムで使用する言語によって異なります。オプション・パラメーターを省略する呼び出しの例については、35 ページの『[ヌル・パラメーターを持つ EXCI CALL の例](#)』を参照してください。

同じ MVS リカバリー単位で複数の要求を出す場合は、すべての要求で同じユーザー ID を使用する必要があります。リカバリー単位に EXEC CICS 呼び出しも含まれている場合は、すべての DPL_requests のユーザー ID が、クライアント・プログラムの実行時に使用されたセキュリティ・ユーザー ID にデフォルトで設定されるようにする必要があります。

dpl_retarea

DPL_Request プロセッサが DPL 要求に対する応答を配置する 12 バイトの出力域。一般的には、これらの応答は CICS から取得されますが、場合によっては、外部 CICS インターフェースでエラーが検出されることもあります。この場合、**EXEC CICS LINK** コマンドから返された条件と同じ例外条件が返されます。

このフィールドは、以下の状況でのみ意味があります。

- EXCI 戻り域の応答フィールドの値がゼロである
- EXCI 戻り域に、サーバー・プログラムが異常終了したことが示されている (応答=USER_ERROR、理由=SERVER_ABENDED)

この 12 バイトは、以下の情報を提供する 3 つのフィールドを形成します。

フィールド 1 (fullword value)

このフィールドは、DPL_Request 呼び出しから取得された RESP 値が入るフルワードです。

DPL_Request 呼び出しで返される可能性のある RESP 値については、『[38 ページの『エラー・コード』](#)』を参照してください。

DPL_Request 呼び出しが CICS に到達すると、このフィールドには EIBRESP 値が入ります。そうでない場合は、外部 CICS インターフェースによって設定された同等の応答が入ります。このフィールドが外部 CICS インターフェースによって設定されると、RESP が 2 番目のフィールドの RESP2 値によってさらに修飾されます。

値ゼロは通常応答であり、戻りコード・コピーブックの EXEC_NORMAL に相当します。

フィールド 2 (fullword value)

このフィールドは、リンク要求から取得された RESP2 値を入れられるフルワードで、フィールド 1 の RESP 値をさらに修飾します。

DPL_Request 呼び出しが CICS に到達する場合は、RESP2 フィールドは一般的にヌルになります (CICS は MRO リンク間で RESP2 値を返しません)。ただし、RESP フィールドが SYSIDERR (値 53) を示す場合、このフィールドは理由コードを提供します。詳細については、[34 ページの『Dpl_retarea 戻りコード』](#)を参照してください。

RESP フィールドが外部 CICS インターフェースによって設定されると、2 番目のフィールドの RESP2 値によってさらに修飾されます。例えば、*data_len* パラメーターに、*COMMAREA_len* パラメーターより大きい値が指定されている場合は、外部 CICS インターフェースが RESP 値として 22 (戻りコード・コピーブックの EXEC LENGERR に相当) を返し、RESP2 値として 13 を返します。

返される可能性のある RESP 値および RESP2 値の詳細については、『[LINK](#)』の LINK 条件を参照してください。

注: データ変換プログラムでは、RESP2 フィールドが特別な用途で使用されます。変換プログラムでエラーが発生すると、RESP2 にエラーが返されます。

フィールド 3 (fullword value)

3 番目のフィールドは 4 文字のフィールドで、以下の情報が入ります。

- サーバー・プログラムが異常終了した場合は異常終了コード
- サーバー・プログラムが異常終了しなかった場合は 4 つのブランク

サーバー・プログラムが異常終了した場合は、最後の同期点 (タスクの先頭、または中間の同期点) までバックアウトされます。SYNCONRETURN が強制されるため、サーバー・プログラムは中間同期点を発行できます。

DPL_opts (または null_ptr)

DPL_Request 呼び出しで使用されるオプションを指示する 1 バイトの入力域。

このパラメーターを省略すると、デフォルトで値 'X'00' に設定されます。DPL_opts を省略してデフォルトにする場合は、CALL パラメーター・リストに、このパラメーターのヌル・アドレスが含まれていることを確認してください。この処理をどのように行うかは、ユーザーが CICS 以外のプログラムで使用する言語によって異なります。オプション・パラメーターを省略する呼び出しの例については、[35 ページの『ヌル・パラメーターを持つ EXCI CALL の例』](#)を参照してください。

現在、DPL_opts パラメーターは、リソース・リカバリーのみに適用され、以下の値が使用されます。

X'00'

MVS RRS でサポートされる 2 フェーズ・コミット・プロトコルを使用して、クライアント・バッチ・プログラムでリソース・リカバリーを制御するために、NOSYNCONRETURN を指定したことを示します。このオプションでは、サーバー・プログラムが CICS に制御を返すときに、CICS サーバー領域で同期点が実行されません。さらに、CICS サーバー・アプリケーション・プログラムは、明示的な同期点を取ってはなりません。明示的な同期点が取られると、CICS がプログラムを異常終了させます。詳しくは、[8 ページの『リソース・リカバリー』](#)を参照してください。

X'80'

CICS サーバー領域に SYNCONRETURN が必要であることを示します。

SYNCONRETURN は、サーバー領域が、クライアント・プログラムから独立して、サーバー・プログラムの正常終了時に同期点を取るよう指定します。このオプションは、サーバー・プログラムが明示的な同期点を取ることを妨げません。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこのパラメーターの等価値は、SYNCONRETURN です。コピーブックの詳細については、[33 ページの『戻り域と関数呼び出し EQUATE のコピーブック』](#)を参照してください。

transid2 (または null_ptr) VERSION_2 以降に適用可能

CICS トランザクション ID が入る 4 文字の入力域。

サーバー・プログラムは、CICS 提供のミラー・トランザクション (CSMI または CPMI) で実行されます。ただし、EXEC インターフェース・ブロック内の EIBTRNID フィールドによってサーバー・プログラムで使用できるようになるトランザクション ID は、**transid2** パラメーターで指定された ID です。**transid2** パラメーターが指定された場合は、**transid** パラメーターは無視されます。次の表に、**transid** と **transid2** のさまざまな組み合わせの例を示します。

表 1. transid2 の使用			
transid	transid2	プログラムが実行されるトランザクション	プログラムで認識される EIBTRNID
UTRN	省略	UTRN	UTRN
UTRN	UEIB	UTRN	UTRN
省略	省略	CSMI	CSMI
省略	UEIB	CSMI	UEIB

使用する DB2ENTRY 定義を判別するために EIBTRNID が使用されるので、**transid2** パラメーターは、Db2® にアクセスするサーバー・プログラムに役立ちます。以前は、**transid** を使用して EIBTRNID を設定することのみが可能でした。そのため、ミラー・トランザクションを CICS に定義しなければなりません。 **transid2** を使用して EIBTRNID が設定されますが、ミラー・プログラムは CICS 提供の定義 CSMI で実行されます。

ccsid (または null_ptr)、VERSION_2 のみ

COMMAREA に含まれている文字データのコード化文字セット ID (CCSID) を示すフルワード・バイナリー入力域。COMMAREA がサーバー・プログラムに渡されたとき、またはサーバー・プログラムから返されたときに、文字データを変換する必要がある場合は、ccsid パラメーターを指定する必要があります。パラメーターは以下の値をとることができます。

-1 (X'FFFFFFFF')

文字データの変換が必要であること、およびサーバーにインストールされている変換テンプレートにソース CCSID が定義されていることを示します。

1 <= ccsid <= 65535

文字データの変換が必要であること、および指定された値が、サーバーにインストールされた変換テンプレートに定義されているソース CCSID を指定変更することを示します。

endian (または null_ptr)、VERSION_2 のみ

COMMAREA に含まれているバイナリー・データの形式 (ビッグ・エンディアンまたはリトル・エンディアン) を示すフルワード・バイナリー入力域。ビッグ・エンディアンは、左端のバイトに最上位数字 (例えば、System 390 アーキテクチャーで使用される数字) が含まれていることを示します。リトル・エンディアンは、右端のバイトに最上位数字 (例えば、Intel アーキテクチャーで使用される数字) が含まれていることを示します。COMMAREA がサーバー・プログラムに渡されたとき、またはサーバー・プログラムから返されたときに、バイナリー・データを変換する必要がある場合は、endian パラメーターを指定する必要があります。ccsid パラメーターは変換が必要であることを示しているものの、endian が指定されていない (デフォルトでヌルに設定される) 場合には、サーバーにインストールされている DFHCNV 変換テンプレートに指定された内容によって、バイナリー・データの変換が異なります。パラメーターは以下の値をとることができます。

16909060 (X'01020304')

バイナリー・データが、ビッグ・エンディアン形式で保持されます。

67305985 (X'04030201')

バイナリー・データが、リトル・エンディアン形式で保持されます。

応答および理由コード

すべての応答コードがゼロ以外の場合は、固有の理由コードの値が応答の理由を示します。

注: 応答コードと理由コードの数値はすべて 10 進数です。

以下は、DPL 呼び出しで外部 CICS インターフェースから返される可能性のある応答コードおよび理由コードの要約です。

応答 OK

コマンドは正常に実行されました (RC 0)。理由コード:

0

NORMAL

応答 WARNING

コマンドは正常に実行されましたが、エラー (RC 4) が発生しました。理由コード:

6

IRP_IOAREA_FM_FAILURE

7

SERVER_TERMINATED

応答 RETRYABLE

セットアップ・エラーが原因でコマンドが失敗しましたが、再発行できます (RC 8)。理由コード:

203

NO_CICS

204

WRONG_MVS_FOR_RRMS

205

RRMS_NOT_AVAILABLE

応答 USER_ERROR

クライアントかサーバーのいずれかのエラーによりコマンドが失敗しました (RC 12)。理由コード:

401

INVALID_CALL_TYPE

402

INVALID_VERSION_NUMBER

404

INVALID_USER_TOKEN

406

PIPE_NOT_OPEN

407
INVALID_USERID

408
INVALID_UOWID

409
INVALID_TRANSID

414
IRP_ABORT_RECEIVED

415
INVALID_CONNECTION_DEFN

416
INVALID_CICS_RELEASE

417
PIPE_MUST_CLOSE

418
INVALID_PIPE_TOKEN

421
RUNNING_UNDER_AN_IRB

422
SERVER_ABENDED

423
SURROGATE_CHECK_FAILED

425
UOWID_NOT_ALLOWED

426
INVALID_TRANSID2

427
INVALID_CCSID

428
INVALID_ENDIAN

431
COMMAREA_LEN_NOT_ALLOWED

432
DATA_LEN_NOT_ALLOWED

433
CCSID_NOT_ALLOWED

434
ENDIAN_NOT_ALLOWED

応答 **SYSTEM_ERROR**

コマンドが失敗しました (RC 16)。理由コード:

612
TRANSFORM_1_ERROR

613
TRANSFORM_4_ERROR

614
IRP_NULL_DATA_RECEIVED

615
IRP_NEGATIVE_RESPONSE

616
 IRP_SWITCH_PULL_FAILURE
617
 IRP_IOAREA_GM_FAILURE
619
 IRP_BAD_IOAREA
620
 IRP_PROTOCOL_ERROR
622
 ESTAE_SETUP_FAILURE
623
 ESTAE_INVOKED
624
 SERVER_TIMEDOUT
625
 STIMER_SETUP_FAILURE
626
 STIMER_CANCEL_FAILURE
629
 SERVER_PROTOCOL_ERROR
630
 RRMS_ERROR
631
 RRMS_SEVERE_ERROR
632
 XCGUR_GETMAIN_ERROR

Close_PIPE

開いているパイプを CICS から切断します。パイプは割り振られた状態のままとなり、そのトークンは同じユーザーが使用できるように有効なままとなります。閉じたパイプを再使用するには、クライアント・プログラムがまず、パイプの Allocate_Pipe コマンドで返されたパイプ・トークンを使用して、Open_Pipe コマンドを再発行する必要があります。

パイプを使用しないときには、閉じる必要があります。CICS が IRC 機能を順序正しくシャットダウンする処理が妨げられてしまうからです。そのため、すべての DPL_Request 呼び出しが完了したら、可能な限り早く Close_Pipe コマンドを実行する必要があります。

構文

```
CALL DFHXCIS,(version_number,return_area,user_token,call_type,
              pipe_token),VL,MF=(E,(1))
```

パラメーター

version_number

使用される外部 CICS インターフェース・パラメーター・リストのバージョンを示すフルワード・バイナリー入力域。クライアント・プログラムでは 1 に設定されている必要があります。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこのパラメーターの等価値は、VERSION_1 です。コピーブックの詳細については、[31 ページの『Deallocate_Pipe』](#)を参照してください。

return_area

応答コードと理由コード、およびメッセージ・ポインター・フィールドを受け取る 5 ワードの出力域。
詳細については、[33 ページの『EXCI CALL インターフェースの戻り域』](#)を参照してください。

user_token

Initialize_User コマンドで EXCI によってクライアント・プログラムに返されるトークンを指定する 1 ワードの入力域。閉じられるパイプのユーザーを表します。

call_type

コマンドの機能を示す 1 ワードの入力域。クライアント・プログラムでは、これを 4 に設定する必要があります。4 は、これが Close_Pipe コマンドであることを示します。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこの呼び出しの等価値は、CLOSE_PIPE です。コピーブックの詳細については、[33 ページの『戻り域と関数呼び出し EQUATE のコピーブック』](#)を参照してください。

pipe_token

元の Allocate_Pipe コマンドで EXCI からクライアント・プログラムに返されたトークンを示す 1 ワードの入力域。閉じられるパイプを表します。

応答および理由コード

すべての応答コードがゼロ以外の場合は、固有の理由コードの値が応答の理由を示します。

注：応答コードと理由コードの数値はすべて 10 進数です。

以下は、Close_Pipe 呼び出しで外部 CICS インターフェースから返される可能性のある応答コードおよび理由コードの要約です。

応答 OK

コマンドは正常に実行されました (RC 0)。理由コード:

0
NORMAL

応答 WARNING

コマンドは正常に実行されましたが、エラー (RC 4) が発生しました。理由コード:

2
PIPE_ALREADY_CLOSED

応答 USER_ERROR

クライアントかサーバーのいずれかのエラーによりコマンドが失敗しました (RC 12)。理由コード:

401
INVALID_CALL_TYPE
402
INVALID_VERSION_NUMBER
404
INVALID_USER_TOKEN
418
INVALID_PIPE_TOKEN
421
RUNNING_UNDER_AN_IRB

応答 SYSTEM_ERROR

コマンドが失敗しました (RC 16)。理由コード:

610
IRC_DISCONNECT_FAILURE
622
ESTAE_SETUP_FAILURE

応答コードについて詳しくは、32 ページの『EXCI 呼び出し応答コード値』を参照してください。
理由コードの詳細については、EXCI 呼び出しで返される応答と理由コードを参照してください。

Deallocate_Pipe

CICS からのパイプの割り振りを解除します。このコマンドの完了時には、パイプを使用できなくなり、それに関連付けられたトークンが無効になります。このコマンドは、必要ではなくなったパイプに対して発行する必要があります。このコマンドにより、パイプに関連付けられているストレージが解放されます。

構文

```
CALL DFHXCIS, (version_number, return_area, user_token, call_type,
               pipe_token), VL, MF= (E, (1))
```

パラメーター

version_number

使用される外部 CICS インターフェース・パラメーター・リストのバージョンを示すフルワード・バイナリー入力域。クライアント・プログラムでは 1 に設定されている必要があります。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこのパラメーターの等価値は、VERSION_1 です。コピーブックの詳細については、33 ページの『戻り域と関数呼び出し EQUATE のコピーブック』を参照してください。

return_area

応答コードと理由コード、およびメッセージ・ポインター・フィールドを受け取る 5 ワードの出力域。詳細については、33 ページの『EXCI CALL インターフェースの戻り域』を参照してください。

user_token

Initialize_User コマンドで返されるトークンが入る 1 ワードの入力域。

call_type

コマンドの機能を示す 1 ワードの入力域。クライアント・プログラムでは、これを 5 に設定する必要があります。5 は、これが Deallocate_Pipe コマンドであることを示します。

CICS 提供のコピーブック DFHXCPLx (ここで x は言語を示す) におけるこの呼び出しの等価値は、DEALLOCATE_PIPE です。コピーブックの詳細については、33 ページの『戻り域と関数呼び出し EQUATE のコピーブック』を参照してください。

pipe_token

元の Allocate_Pipe コマンドで返されたトークンが入る 1 ワードの入力域。割り振り解除しているパイプを表します。

応答および理由コード

すべての応答コードがゼロ以外の場合は、固有の理由コードの値が応答の理由を示します。

注：応答コードと理由コードの数値はすべて 10 進数です。

以下は、Deallocate_Pipe 呼び出しで外部 CICS インターフェースから返される可能性のある応答コードおよび理由コードの要約です。

応答 OK

コマンドは正常に実行されました (RC 0)。理由コード：

0

NORMAL

応答 WARNING

コマンドは正常に実行されましたが、エラー (RC 4) が発生しました。理由コード：

- 5 XCIPIPE_FREEMAIN_ERROR
- 6 IRP_IOAREA_FM_FAILURE

応答 **USER_ERROR**

クライアントかサーバーのいずれかのエラーによりコマンドが失敗しました (RC 12)。理由コード:

- 401 INVALID_CALL_TYPE
- 402 INVALID_VERSION_NUMBER
- 404 INVALID_USER_TOKEN
- 405 PIPE_NOT_CLOSED
- 418 INVALID_PIPE_TOKEN
- 421 RUNNING_UNDER_AN_IRB

応答 **SYSTEM_ERROR**

コマンドが失敗しました (RC 16)。理由コード:

- 611 IRC_LOGOFF_FAILURE
- 622 ESTAE_SETUP_FAILURE
- 623 ESTAE_INVOKED

応答コードについて詳しくは、[32 ページの『EXCI 呼び出し応答コード値』](#)を参照してください。

理由コードの詳細については、[EXCI 呼び出しで返される応答と理由コード](#)を参照してください。

EXCI 呼び出し応答コード値

次の表に、応答フィールドに返される可能性のある値を示します。すべての値が 10 進数です。

表 2. EXCI 応答コード (*return_area* の応答フィールドに返される)

コード	意味	説明
0	OK	DPL_request 以外の EXCI CALL コマンドの場合は、呼び出しが正常に実行されました。DPL_request に対して OK 応答を受け取った場合は、 <i>dpl_retarea</i> を調べて、CICS が条件コードを返さなかったことを確認することも必要です。 <i>Dpl_retarea</i> の EIBRESP フィールドがゼロである場合は、DPL 呼び出しが正常に実行されました。
4	WARNING	外部 CICS インターフェースがエラーを検出しましたが、これにより正常に完了している CALL コマンドは停止されませんでした。理由コード・フィールドは、検出されたエラーを示します。

表 2. EXCI 応答コード (*return_area* の応答フィールドに返される) (続き)

コード	意味	説明
8	RETRYABLE	EXCI CALL コマンドが失敗しました。この障害のクラスは、システム環境のセットアップにおけるエラーに関連し、外部 CICS インターフェースまたはクライアント・プログラムのエラーには関連しません。理由コードは、環境セットアップにおける特定のエラーを示します。 環境エラーを訂正したら、クライアント・プログラムを変更せずに外部 CICS インターフェース・コマンドを再発行できます。関連する環境エラーは、MVS 再 IPL を必要としないエラーです。RETRYABLE 応答の各理由コード値は、CALL を直接再発行できるか、それとも使用中のパイプをまずクローズして再オープンする必要があるかを示します。
12	USER_ERROR	EXCI CALL コマンドが失敗しました。このエラーのクラスは、クライアント・プログラム、CICS サーバー・プログラム、または CICS サーバー領域にエラーがあることを示します。CICS サーバー・システムのエラーの例として、セキュリティ検査の失敗や、CICS サーバー・プログラムの異常終了などがあります。異常終了した場合は、 <i>dpl_retarea</i> の異常終了コード・フィールドに異常終了コードが設定されます。USER_ERROR の応答の各理由コード値は、コマンドを直接再発行できるか、それとも使用中のパイプをまずクローズして再オープンする必要があるかを示します。
16	SYSTEM_ERROR	EXCI CALL コマンドが失敗しました。このクラスのエラーは、外部 CICS インターフェースがエラーを検出したことを示します。理由コード値は特定のエラーを示します。エラーを修正できる場合は、コマンドを再発行できます。SYSTEM_ERROR 応答の各理由コード値は、コマンドを直接再発行できるか、それとも使用中のパイプをまずクローズして再オープンする必要があるかを示します。

EXCI CALL インターフェースの戻り域

これは、EXCI CALL インターフェースの 5 ワードの戻り域のフォーマットです。

- 1 ワードの応答フィールド。
- 1 ワードの理由フィールド。
- 2 つの 1 ワードの理由サブフィールド (理由サブフィールド 1 および理由サブフィールド 2)。
- 1 ワードの CICS メッセージ・ポインター・フィールド。メッセージが存在しない場合、これはゼロです。メッセージが存在している場合、このフィールドには、メッセージが入っているストレージ域のアドレスが入ります。このフォーマットは次のとおりです。
 - 2 バイトの LL フィールド。LL は、メッセージ長に LLBB フィールド長を加えた長さです。
 - 2 バイトの BB フィールド。2 進ゼロに設定されます。
 - メッセージのテキストが入る可変長フィールド。

戻り域と関数呼び出し EQUATE のコピーブック

CICS は、EXCI CALL コマンドの *return_area* および *dpl_retarea* パラメーター用のストレージ域をマップする、言語固有の 4 つのコピーブックを提供します。このコピーブックは、EXCI CALL のタイプごとの EQUATE ステートメントも提供します。

34 ページの表 3 に、これらのコピーブックとその提供場所となるライブラリーを、サポートされている言語別に示します。

表 3. 戻り域およびそれに対応する名前に対して提供されているコピーブック		
コピーブック名	言語	ライブラリー
DFHXCPLD	アセンブラー	CICSTS56 .CICS.SDFHMAC
DFHXCPLH	C	CICSTS56 .CICS.SDFHC370
DFHXCPLD	COBOL	CICSTS56 .CICS.SDFHCOB
DFHXCPLL	PL/I	CICSTS56 .CICS.SDFHPL1

戻りコード

返される可能性のあるすべての戻りコードが CICS 提供のコピーブックに記載されており、外部の非 CICS プログラムのプログラム・ソースには、このコピーブックを組み込む必要があります。

サポートされる言語のコピーブックの名前と、それらが含まれているライブラリーは、[34 ページの表 4](#) に示されています。

表 4. 提供されている RESPONSE コードおよび REASON コードのコピーブック		
コピーブック名	言語	ライブラリー
DFHXCRCRCD	アセンブラー	CICSTS56 .CICS.SDFHMAC
DFHXCRCRH	C	CICSTS56 .CICS.SDFHC370
DFHXCRCO	COBOL	CICSTS56 .CICS.SDFHCOB
DFHXCRCCL	PL/I	CICSTS56 .CICS.SDFHPL1

z/OS には、『[11 ページの『クライアント・プログラムでの同期点の使用法』](#)』で説明しているインターフェースで使用するコピーブックが用意されています。これらについては、[z/OS MVS プログラミング: リソース・リカバリー](#)および [z/OS MVS プログラミング: 高水準言語向け呼び出し可能サービス](#)で説明します。

Dpl_retarea 戻りコード

これらは CICS 間の EXEC CICS DPL コマンドの戻りコードと同じですが、EXCI 呼び出しインターフェースでは以下が追加されます。

表 5. 例外条件. dpl_retarea に返される RESP 値および RESP2 値

状態	RESP2	意味
LENGERR	22	COMMAREA_LEN_TOO_BIG
LENGERR	23	COMMAREA_BUT_NO_COMMAREA_LEN

DPL_Request が CICS サーバー領域でリモート・プログラムとして定義されているプログラムを指定するときに、サーバーとリモート CICS 領域間のリンクが開いていない場合は、EXCI DPL_Request でも SYSIDERR が返されることがあります。この状態では、DPL_Retarea (コード 53) の最初のワードに SYSIDERR が返されます。SYSIDERR を修飾する理由コードは、この領域の 2 番目のワード (RESP2 値に相当) に配置されます。SYSIDERR の場合は、このフィールドに格納される情報が、CICS EIBRCODE フィールドのバイト 1 と 2 から導出されます。例えば、リモート・リンクが使用不可である場合、DPL_Retarea RESP2 フィールドのバイト 2 と 3 に保管される EIBRCODE 値は X'0800' です。RESP2 フィールドに返される可能性のある SYSIDERR 理由コードのリストについては、[アテンション ID 定数](#)、[DFHAID](#) に記載されている EIBRCODE の注記の中の SYSIDERR のセクションを参照してください。

DPL 要求がリモートとして定義されたプログラムに対するものであった場合は、リモート CICS システム上のミラーとの会話中にリカバリー不能エラーが発生すると、EXCI DPL 要求で TERMERR が返される可能性もあります。例えば、クライアント・プログラム BATCH1 が、リモートとして定義されているプログラム PROG1 の CICS に対して EXCI DPL 要求を発行し、その要求が、プログラムが存在する CICS に機能シップされるとします。CICS と CICS 間のセッションが失敗するか、またはプログラム PROG1 の実行中に CICS そのものが失敗すると、TERMERR が CICS に返され、次に BATCH1 に返されます。

TERMERR、PGMIDERR、NOTAUTH、および ROLLBACK の場合は、固有の EXCI_DPL_RESP2 値が返されません。

ヌル・パラメーターを持つ EXCI CALL の例

オプション・パラメーター (DPL_Request の *userid* など) を省略する場合は、欠落しているパラメーターに対してヌル・アドレスがパラメーター・リストに指定されていることを確認する必要があります。

次の例では、COBOL プログラムで *userid* パラメーターと *uowid* パラメーターを省略して EXCI DPL_Request を発行する方法を示します。

userid と uowid (COBOL) を省略した DPL CALL: この例では、次のように、呼び出しで使用された DPL パラメーターが WORKING-STORAGE SECTION に定義されています。

DPL パラメーター	COBOL 変数	フィールド定義
<i>version_number</i>	01 VERSION-1	PIC S9(8) COMP VALUE 1.
<i>return_area</i>	01 EXCI-RETURN-CODE.	(構造)
<i>user_token</i>	01 USER-TOKEN	PIC S9(8) COMP VALUE ZERO.
<i>call_type</i>	03 DPL-REQUEST	PIC S9(8) COMP VALUE 6.
<i>pipe_token</i>	01 PIPE-TOKEN	PIC S9(8) COMP VALUE ZERO.
<i>pgmname</i>	01 TARGET-PROGRAM	PIC X(8) VALUE "DFHœAXCS".
<i>commarea</i>	01 COMMAREA.	(構造)
<i>commarea_len</i>	01 COMM-LENGTH	PIC S9(8) COMP VALUE 98.
<i>data_len</i>	01 DATA-LENGTH	PIC S9(8) COMP VALUE 18.
<i>transid</i>	01 TARGET-TRANSID	PIC X(4) VALUE "EXCI".
<i>dpl_retarea</i>	01 EXCI-DPL-RETAREA.	(構造)
<i>dpl_opts</i>	01 SYNCONRETURN	PIC X VALUE X'80'.

ヌル・アドレスに使用されている変数は、以下のように LINKAGE SECTION で定義されています。

```
LINKAGE SECTION.  
    01 NULL-PTR USAGE IS POINTER.
```

WORKING-STORAGE SECTION で指定したデータ名、および LINKAGE SECTION で説明した NULL-PTR 名を使用すると、DPL 関数の次の呼び出しで、*uowid* パラメーターと *userid* パラメーターが省略され、パラメーター・リスト内のこれらのパラメーターが NULL-PTR 変数に置き換えられます。

```
DPL-SECTION.  
*  
* SET ADDRESS OF NULL-PTR TO NULLS.  
*  
CALL 'DFHXCIS' USING VERSION-1 EXCI-RETURN-CODE USER-TOKEN  
DPL-REQUEST PIPE-TOKEN TARGET-PROGRAM  
COMMAREA COMM-LENGTH DATA-LENGTH  
TARGET-TRANSID NULL-PTR NULL-PTR  
EXCI-DPL-RETAREA SYNCONRETURN.
```

この例は、CICS 提供のサンプル外部 CICS インターフェース・プログラム DFH0CXCC から取られたものです。このプログラムは CICSTS56.CICS.SDFHSAMP 内にあります。サポートされる他の言語で DPL 呼び出しから同じパラメーターを省略する方法の例については、以下のサンプル・プログラムを参照してください。

DFH\$AXCC

アセンブラー・サンプル

DFH\$PXCC
PL/I サンプル

DFH\$DXCC
C サンプル。

EXCI EXEC CICS インターフェース

EXCI EXEC CICS インターフェースには、いくつかのコマンドが用意されています。例えば **EXEC CICS LINK PROGRAM** コマンドは、EXCI CALL インターフェースのすべての呼び出しを行う単一の複合コマンドです。チャンネル・コマンドとコンテナー・コマンドは、ご使用のプログラムで、データの処理後にそのデータを **EXEC CICS LINK** プログラム・コマンドに渡したり、API **DPL_REQUEST** を呼び出したりするために、バッチで発行することができます。

このセクションでは、以下について説明します。

- [LINK \(EXCI\)](#)
- [40 ページの『EXEC CICS LINK コマンドの再試行』](#)
- [EXEC CICS LINK コマンドに必要な変換](#)
- [DELETE CHANNEL \(EXCI\)](#)
- [DELETE CONTAINER \(EXCI\)](#)
- [GET CONTAINER \(EXCI\)](#)
- [MOVE CONTAINER \(EXCI\)](#)
- [PUT CONTAINER \(EXCI\)](#)

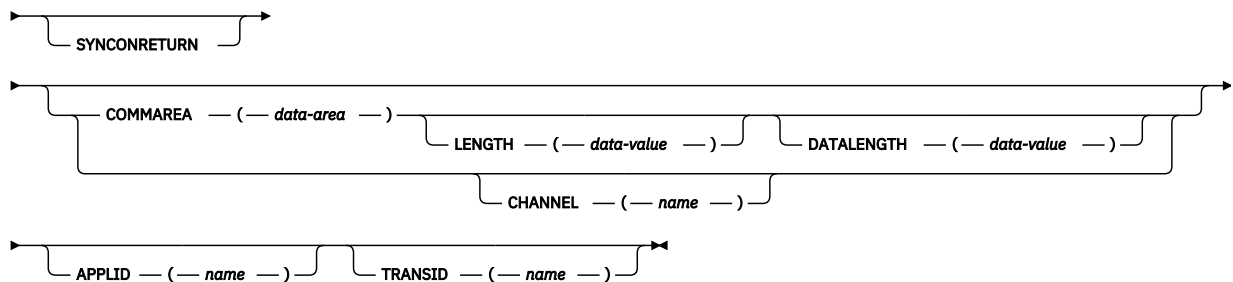
EXEC CICS LINK コマンド (EXCI)

MVS クライアント・プログラムから、サーバー CICS 領域内の指定されたサーバー・プログラムへのリンク。

フォーマット

LINK

➡ LINK — PROGRAM — (— name —) — RETCODE — (— data-area —) —➡



注:

エラー条件: CCSIDERR、CHANNELERR、CODEPAGEERR、CONTAINERERR、LENGERR、LINKERR、NOTAUTH、PGMIDERR、RESUNAVAIL、ROLLED BACK、SYSIDERR、TERMERR、WARNING

コメント

APPLID パラメーターと **RETCODE** パラメーターは除き、**EXEC CICS LINK** コマンドの外部 CICS インターフェース・パラメーターは、CICS-CICS DPL コマンドの場合と同じです。

この情報は、外部 CICS インターフェースと共に使用できるパラメーターについてのみ記述します。**EXEC CICS LINK PROGRAM** コマンドのプログラミング情報については、[LINK](#) を参照してください。

LENGTH パラメーターと **DATALENGTH** パラメーターはハーフワード・バイナリー値を指定しますが、これらのパラメーターに対応する EXCI CALL インターフェースの **COMMAREA_len** パラメーターと **data_len** パラメーターは、フルワード値を指定することに注意してください。

外部 CICS インターフェースの **EXEC CICS LINK** コマンドは、必ず汎用接続を使用します。

パラメーター

LINK コマンドの外部 CICS インターフェース 形式で利用できるパラメーターは、以下のとおりです。

APPLID(name)

ターゲット CICS サーバー領域のアプリケーション ID を指定します。

外部 CICS インターフェース・コマンドにはアプリケーション ID が必須ですが、このパラメーターは、ユーザー置換可能モジュール DFHXCURM でも指定できるため、**LINK** コマンドそのものではオプションです。**LINK** コマンドで汎用 APPLID を省略する場合は、ユーザー置換可能モジュール DFHXCURM で **URMCICS** パラメーターに指定されていることを確認する必要があります。また、DFHXCURM で **URMCICS** パラメーターを使用すると、**LINK** コマンドで指定されたアプリケーション ID を指定変更することもできます。**URMCICS** パラメーターについては、[EXCI ユーザー置換可能モジュール](#)を参照してください。

CHANNEL(name)

起動されたプログラムで使用可能にするチャンネルの名前 (1 から 16 文字) を指定します。許容文字は、A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - および _ です。ブランク文字を先頭にしたり、埋め込んだりすることはできません。指定された名前が 16 文字よりも短い場合は、16 文字になるまで末尾にブランクが埋め込まれます。チャンネルが存在していない場合は、そのチャンネルが作成されます。EXCI クライアントには、**LINK** レベルが 1 つしかないため、このチャンネルは有効範囲に入ったままになります。チャンネルの有効範囲について詳しくは、[チャンネルの有効範囲](#)を参照してください。

チャンネル名は常に EBCDIC です。前述の、チャンネル名に使用できる文字セットには、EBCDIC コード・ページによっては異なる表示となる文字も一部含まれています。したがって、領域間でチャンネルをシッップする場合は、名前に使用する文字を A-Z a-z 0-9 & : = , ; < > . に制限することを推奨します。- および _ です。

チャンネル名 DFHTRANSACTION を指定してトランザクション・チャンネルを使用できます。CICS では、リンク・レベルが変更されても、トランザクション・チャンネルは有効範囲から外れず、そのトランザクションで常にアクセス可能です。詳しくは、[チャンネルおよびコンテナ](#)を参照してください。

LINK コマンドを発行するプログラムでは、コマンドでチャンネルの名前を指定できます。指定されたチャンネルは、既に存在している場合がありますが、プログラムで 1 つ以上の **PUT CONTAINER** コマンドを使用して作成されます。あるいは、プログラムで、現在存在していないチャンネルの名前を指定することもできます。この場合は、新しい空のチャンネルが作成されます。

COMMAREA(data-area)

呼び出されたプログラムで使用可能となる通信域を指定します。このオプションでは、そのデータ域へのポインタが引き渡されます。

CICS アプリケーション・プログラムへのデータの引き渡しについて詳しくは、[他のプログラムへのデータの受け渡し](#)を参照してください。

DATALENGTH(data-value)

ストレージの連続域 (COMMAREA の先頭以降) の長さをハーフワード・バイナリー値で指定します。COMMAREA のデータの量は少ないものの、COMMAREA そのものは大きい場合は、DATALENGTH を指定してパフォーマンスを向上させます。

LENGTH(data-value)

COMMAREA のバイト単位の長さをハーフワード・バイナリー値で指定します。

COMMAREA を 2 つの CICS サーバーの間で受け渡す場合 (製品/バージョン/リリースがどんな組み合わせであっても)、24 KB を超える値を指定することはできません。それ以外の場合は、COMMAREA がそれ以降の **LINK** 要求で渡されないことが確実であるときに、最大 32763 の長さの COMMAREA を使用できます。

PROGRAM(*name*)

制御が無条件で渡される CICS サーバー・アプリケーション・プログラムのプログラム名 (1 から 8 文字) を指定します。指定する名前が CICS にプログラムとして定義されている名前であるか、または CICS サーバー領域が、指定されたプログラムの定義を自動インストールできることが必要です。

以下のような、引用符の使用法に注意してください。

```
EXEC CICS LINK PROGRAM('PROGX')
```

PROGX はプログラム名のため、引用符で囲みます。

```
EXEC CICS LINK PROGRAM(DAREA)
```

DAREA は、8 文字のプログラム名が入るデータ域の名前であるため、引用符で囲みません。

RETCODE(*data-area*)

外部 CICS インターフェースが戻りコード情報を配置する 20 バイト領域を指定します。この領域は、以下のように 5 つの 1 ワード・フィールドで形成されます。

RESP

外部 CICS インターフェースの LINK コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

ABCODE

サーバー・プログラムがサーバー領域で異常終了した場合は、有効な CICS 異常終了コードが入ります。

MSGLEN

サーバー・プログラムの実行中に CICS サーバー領域から発行されたメッセージがある場合は、その長さを示します。この長さは、メッセージ・テキストのみの実際の長さであり、この 1 ワードの長さフィールドは含まれないことに注意してください。

MSGPTR

これは、CICS サーバー領域から返されたメッセージ・テキストのアドレスです。

注：MSGLEN および MSGPTR は、RESP2 の値が 414 である LINKERR 条件のみで有効です。

SYNCONRETURN

APPLID パラメーターに指定された CICS サーバー領域が、サーバー・プログラムの正常終了時に同期点を取るように指定します。

TRANSID(*name*)

リモート領域を接続し、サーバー・プログラムを実行するミラー・トランザクションの名前を指定します。TRANSID オプションを省略すると、CICS サーバー領域が CSMI を接続します。

注：LINK コマンドで指定される TRANSID オプションは、CICS サーバー領域にインストールされているプログラム・リソース定義に指定されたあらゆる TRANSID オプションを指定変更します。

DPL 要求によって開始されるミラー・トランザクションの名前を独自に指定することができますが、トランザクションはサーバー領域で定義されている必要があり、トランザクション定義はミラー・プログラムである DFHMIRS を指定する必要があります。独自のトランザクションを定義してミラー・プログラムを呼び出すと、トランザクション・リソース定義の他のいくつかのオプションに、適切な値を自由に指定できます。

DPL_Request を使用した transid の指定に関する重要な規則について、も参照してください。 .

エラー・コード

外部 CICS インターフェースの LINK コマンドで返されるほとんどの例外条件が、CICS 間の分散プログラム・リンク・コマンドの例外条件と同じです。同一である例外条件と、それらに対応する数値は、以下のとおりです。

LENGERR
22
PGMIDERR
27
SYSIDERR
53
NOTAUTH
70
TERMERR
81
ROLLEDBACK
82
CONTAINERERR
110
RESUNAVAIL
121
CHANNELERR
122
CCSIDERR
123
CODEPAGEERR
125

これらの例外条件コードは、RESP フィールドに返されます。

RESP および **RESP2** : このセクションでは、RESP フィールドと RESP2 フィールドは、RETCODE パラメータの最初の 2 つのフィールドを指しています。

外部 CICS インターフェース 固有の例外条件は以下のとおりです。

- エラー条件 LENGERR の RESP 値は、外部 CICS インターフェース 固有の値です。
- 例外条件 WARNING および LINKERR は、外部 CICS インターフェース 固有の例外条件です。

例外 WARNING および LINKERR は、**EXEC CICS LINK** コマンドに応答して外部 CICS インターフェースから発行された個別の EXCI 呼び出しに対する応答の結果です。これらの例外 WARNING および LINKERR は、以下の説明のように EXCI 呼び出し応答に対応します。

WARNING (RESP 値 4)

これが返されるのは、**EXEC CICS LINK** 要求を処理する EXCI モジュールが、**EXEC CICS LINK** コマンドの代わりに発行された Close_Pipe または Deallocate_Pipe 要求に対する USER_ERROR または SYSTEM_ERROR 応答を受け取ったときです。CICS に対する DPL 要求は正常に完了したものの、構造の処理でエラーが発生したため、RESP 値は WARNING に設定されます。

RESP2 フィールドは EXCI 理由コードに設定されます。この理由コードはエラーの詳細情報を提供します。

LINKERR (RESP 値 88)

これが返されるのは、**EXEC CICS LINK** 要求を処理する EXCI モジュールが、**EXEC CICS LINK** コマンドの代わりに発行された EXCI 呼び出しに対する RETRYABLE、USER_ERROR、または SYSTEM_ERROR 応答を受け取ったときです。DPL 要求は失敗しました。RESP2 フィールドは EXCI 理由コードに設定されます。この理由コードはエラーの詳細情報を提供します。

EXCI 理由コードの説明については、[EXCI 呼び出しで返される応答と理由コード](#)を参照してください。

注 : 外部 CICS インターフェースは、**EXEC CICS LINK** コマンドの代わりに発行される EXCI 呼び出しへの応答で発生する WARNING 条件を無視します。EXCI 呼び出しに対する WARNING は適切な応答として扱われ、通常どおり続行されます。エラーがほかに発生していない場合は、EXEC CICS コマンドは完了し、**EXEC CICS** フィールドにゼロの応答が返されます。

EXEC CICS LINK コマンドの再試行

外部 CICS インターフェースが、**EXEC CICS LINK** コマンドの代わりに発行した EXCI 呼び出しに対する RETRYABLE 応答を受け取ると、**EXEC CICS LINK** コマンドが最大 5 回まで自動的に再試行されます (ただし、より重大なエラーが発生しない場合に限りです)。

5 回目の再試行後にさらに RETRYABLE 応答を受け取ったら、RESP フィールドが LINKERR に設定され、例外の原因である EXCI CALL 要求に返された理由が、RESP2 フィールドに返されます。

外部 CICS インターフェースは、**EXEC CICS LINK** コマンドを再試行するため、まずパイプを閉じて割り振り解除してから、6 つの EXCI CALL コマンドを再実行します。Allocate_Pipe 処理中に、EXCI CALL インターフェースはユーザー置換モジュール DFHXCURM を呼び出すので、要求が送られた CICS システムの APPLID を変更する機会があります。DFHXCURM の詳細については、[EXCI ユーザー置換可能モジュール](#)を参照してください。

40 ページの表 6 に、外部 CICS インターフェースの **EXEC CICS LINK** コマンド固有の例外条件および RESP2 値をすべてリストしてあります。

表 6. 例外条件. EXEC API から返される RESP 値および RESP2 値。

条件 (RESP)	RESP2	意味
LENGERR (22)	22	32763 バイトより大きい COMMAREA 長が指定されました
	23	COMMAREA が指定されましたが、LENGTH パラメーターが指定されませんでした
WARNING (4)	401	Close_Pipe または Deallocate_Pipe 呼び出しで無効な <i>call_type</i> パラメーター値が指定されました
	402	Close_Pipe または Deallocate_Pipe 呼び出しで無効な <i>version_number</i> パラメーターが指定されました
	404	Close_Pipe または Deallocate_Pipe 呼び出しで無効な <i>user_token</i> が指定されました
	405	まだ閉じていないパイプに対して Deallocate_Pipe 呼び出しが発行されました
	418	Close_Pipe または Deallocate_Pipe 呼び出しで無効なパイプ・トークンが発行されました
	421	Close_Pipe または Deallocate_Pipe コマンドが IRB 下で発行されました
	610	Deallocate_Pipe 呼び出しで CICS IRP ログオフに失敗しました
	611	Close_Pipe 呼び出しで CICS IRC 切断に失敗しました
	622	Close_Pipe または Deallocate_Pipe 呼び出しで MVS ESTAE セットアップに失敗しました
	623	Close_Pipe または Deallocate_Pipe 呼び出しのプログラム・チェックが原因で ESTAE が呼び出されました
LINKERR (88)	201	前回の IPL 以降に IRC アクティビティーがなかった MVS イメージに対してコマンドが発行されました
	202	使用可能なセッションがありません
	203	CICS がまだ起動していないか、(2) まだ IRC を開いていないか、(3) 汎用接続がインストールされていないか、または (4) 必要なネット名で特定の接続がインストールされていません
	204	SYNCONRETURN オプションなしの EXEC CICS LINK コマンドが発行されましたが、異なる MVS イメージ上の CICS システムが指定されています。
	205	RRS が使用不可であるときに、SYNCONRETURN オプションなしの EXEC CICS LINK コマンドが発行されました

表 6. 例外条件. EXEC API から返される RESP 値および RESP2 値。(続き)

条件 (RESP)	RESP2	意味
LINKERR (88)	401	パラメーターが無効です
	402	バージョン番号が無効です
	403	ユーザー名がすべてブランクです
	404	ユーザー・トークンのアドレスが無効です
	405	閉じていないパイプに対してコマンドが発行されました
	406	開いていないパイプに対してコマンドが発行されました
	407	すべてブランクのユーザー ID が渡されました
	408	UOWID パラメーターにエラーがあります
	409	すべてブランクまたはゼロで構成された Transid が渡されました
	410	メッセージ・モジュール DFHMEBMX のロードに失敗しました
	411	メッセージ・モジュール DFHMET4E のロードに失敗しました
	412	DFHXCURM のロードに失敗しました
	413	DFHXCTRA のロードに失敗しました
	414	CICS 間でリンクされたプログラムとして実行された場合、このサーバー・プログラムはエラーになり、適切なメッセージが端末に送られたはずです。EXCI サーバー・プログラムとしてプログラムを実行すると、RETCODE 域の MSGPTR フィールドでアドレス指定されたメッセージが返されます。
	415	ターゲット接続が、EXCI 接続ではなく MRO 接続です
	416	CICS for MVS/ESA 4.1 より前のリリースの CICS で稼働する CICS 領域に対してコマンドが発行されました
	417	MUST CLOSE 状態のパイプに対してコマンドが発行されました。それ以降の EXCI EXEC CICS LINK コマンドは予測不能な結果となるため、許可されません。
	418	Pipe_token が XCPPIPE 制御ブロックをアドレス指定していないか、または user_token と pipe_token 間に不一致があります
	419	このコマンドが使用を試行している TCB 下で、CICS が稼働中、または稼働していました。これは許可されず、コマンドは失敗します。
	420	DFHXCLOPT のロードに失敗しました
	421	コマンドが MVS IRB 下で発行されたが、これは許可されません
	422	サーバーが異常終了しました
	423	代理ユーザー検査が失敗しました
	424	RRMS をサポートしないシステムで、SYNCONRETURN オプションなしの EXEC CICS LINK コマンドが実行されました
	425	DPL 要求で SYNCONRETURN オプションが省略されましたが、UOWID の値が指定されました。
	601	作業用ストレージの GETMAIN が失敗しました。このエラーの結果、ユーザー異常終了 408 になります。
	602	GETMAIN が失敗しました。このエラーの結果、ユーザー異常終了 403 になります。

表 6. 例外条件. EXEC API から返される RESP 値および RESP2 値。(続き)

条件 (RESP)	RESP2	意味
LINKERR (88)	603	GETMAIN が失敗しました。このエラーの結果、ユーザー異常終了 410 になります。
	604	GETMAIN が失敗しました
	605	VERIFY ブロックの GETMAIN が失敗しました。このエラーの結果、ユーザー異常終了 409 になります。
	606	SSI 検査要求 (CICS SVC 命令の取得) が失敗しました。このエラーの結果、ユーザー異常終了 405 になります。
	607	SVC 呼び出しが失敗しました。このエラーの結果、ユーザー異常終了 406 になります。
	608	IRP へのログオンに失敗しました
	609	IRP への接続に失敗しました
	610	IRP からの切断に失敗しました
	611	IRP からのログオフに失敗しました
	612	transformer_1 に無効なデータが入力されました
	613	transformer_4 に無効なデータが入力されました
	614	CICS は応答しましたが、データは送信されていません
	615	CICS が要求を満たすことができません
	616	IRP_SWITCH_PULL 要求 (CICS から、より大きい入出力域に送信されたデータの読み取り) が失敗しました
	617	より大きい入出力域の GETMAIN が失敗しました
	619	IRP で、クライアント・プログラムから渡された入出力域での問題が起きました
	620	IRP が EXCI から切断されました
	621	IRP CONNECT 実行後、エラー状態で DISCONNECT コマンドが実行されました。DISCONNECT は失敗し、重大なエラーがあることが示されています。
	622	XCPRH ESTAE セットアップ・コマンドが失敗しました。このエラーの結果、ユーザー異常終了 402 になります。
	623	このコマンドの処理中に行われたプログラム・チェックが原因で、XCPRH ESTAE が呼び出されました。ESTAE はバックアウトを試行し、SYSMDUMP を取ります。パイプは MUST CLOSE 状態になっていますが、これ以降の要求は許可されます。
	624	DPL 要求が CICS に渡されましたが、DFHXCOPT に指定された時間を超えています。要求は取り消されます。
	625	MVS STIMERM マクロ呼び出しが失敗しました。
	626	MVS STIMERM CANCEL 要求が失敗しました。
	627	CICS SVC のレベルが正しくありません。このエラーの結果、ユーザー異常終了 407 になります。
	628	DFHIRP のレベルが正しくありません。
	629	DPL 要求に対する応答が CICS から返されましたが、外部 CICS インターフェースが応答を把握できません。

表 6. 例外条件. EXEC API から返される RESP 値および RESP2 値。(続き)

条件 (RESP)	RESP2	意味
	630	SYNCONRETURN オプションなしの EXEC CICS LINK コマンドの処理中に、RRMS から予期しない戻りコードを受け取りました。
	631	SYNCONRETURN オプションなしの EXEC CICS LINK コマンドの処理中に、予期しないエラーが検出されました。
	632	SYNCONRETURN オプションなしの EXEC CICS LINK コマンドの処理中に、DFHXCGR の作業用ストレージの GETMAIN が失敗しました。
	903	XCEIP ESTAE セットアップ・コマンドが失敗しました。
	904	XCEIP ESTAE が呼び出されました。

すべての応答コードおよび理由コード (同等の値を含む) について詳しく説明した各種コピーブックの詳細については、『[戻りコード](#)』を参照してください。

注: 数値の応答コードおよび理由コードはすべて、10 進数で示してあります。

EXEC CICS LINK コマンドに必要な変換

外部 CICS インターフェース・コマンドの **EXEC CICS LINK** 形式を使用するアプリケーション・プログラムは、そのプログラムを変換してからでないと、アセンブリまたはコンパイルを実行できません。これには、変換プログラム・オプション EXCI を指定して、ご使用のクライアント・プログラムの言語に適した CICS 変換プログラムのバージョンを使用します。

変換プログラム・オプション EXCI は、CICS オプションおよび DLI オプションと相互に排他的です。

EXEC CICS コマンドを含む変換プログラムについて詳しくは、[CICS 提供の変換プログラム](#)を参照してください。

外部 CICS インターフェース・クライアント・プログラムのコンパイルおよびリンク・エディットについては、『[EXCI クライアント・プログラムのコンパイルとリンク・エディット](#)』を参照してください。

EXEC CICS DELETE CHANNEL コマンド (EXCI)

指定されたチャンネルとその中のすべてのコンテナを削除します。

DELETE CHANNEL (EXCI)

➡ DELETE CHANNEL(*data-value*) — RETCODE(*data-area*)->⬅

条件: CHANNELERR

説明

DELETE CHANNEL (EXCI) は、指定されたチャンネルとその中のすべてのコンテナを削除します。チャンネルとそのコンテナを削除するときは、以下ようになります。

- コンテナ内にあるデータはすべて廃棄されます。
- チャンネルとそのコンテナに関連したすべての記憶域は解放されます。

DELETE CHANNEL コマンドを発行するアプリケーション・プログラムは、そのチャンネルを所有するプログラムでなければなりません。チャンネルを所有するプログラムは、以下のいずれかのコマンドでそのチャンネルを指定して作成したプログラムです。

- **LINK PROGRAM CHANNEL**
- **MOVE CONTAINER CHANNEL TOCHANNEL**
- **PUT CONTAINER CHANNEL**

アプリケーション・プログラムは、以下のチャンネルは削除できません。

- そのアプリケーション・プログラムが作成しなかったチャンネル

- 読み取り専用チャンネル
- トランザクション・チャンネル DFHTRANSACTION

オプション

CHANNEL(data-value)

削除するチャンネルの名前を 1 から 16 文字で指定します。チャンネルが所有しているすべてのコンテナーは削除され、チャンネル自体が削除されます。

RETCODE(data-area)

外部 CICS インターフェースが戻り情報を配置する 20 バイト領域を指定します。この領域は、5 つの 1 ワード・フィールドにフォーマット設定されます。このコマンドは、最初の 2 ワードに情報を返します。これらのワードには、RESP コードと RESP2 コードが含まれます。他の 3 ワードは、ヌルに設定されます。

RESP

外部 CICS インターフェースの **DELETE CHANNEL** コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

条件

122 CHANNELERR

RESP2 値:

2

CHANNEL オプションで指定されたチャンネルが見つかりませんでした。

3

CHANNEL オプションで指定されたチャンネルは、読み取り専用チャンネルです。

5

CHANNEL オプションで指定されたチャンネルは、トランザクション・チャンネルです。

6

CHANNEL オプションで指定されたチャンネルは、呼び出し側プログラムが所有していません。

904

XCEIP ESTAE が呼び出されました。

EXEC CICS DELETE CONTAINER コマンド (EXCI)

指定されたチャンネル・コンテナーを削除します。

DELETE CONTAINER (EXCI)

➡ DELETE — CONTAINER(*data-value*) — CHANNEL(*data-value*) — RETCODE(*data-area*) ➡

状態: CHANNELERR、CONTAINERERR、INVREQ

説明

DELETE CONTAINER (EXCI) は、コンテナーをチャンネルから削除し、そのコンテナーに含まれているデータをすべて破棄します。

コンテナーは、名前、およびコンテナーとなっているチャンネル、つまりコンテナーを「所有」しているチャンネルによって識別されます。

オプション

CHANNEL(data-value)

コンテナーを所有するチャンネルの名前 (1 から 16 文字) を指定します。チャンネル名 DFHTRANSACTION を指定してトランザクション・チャンネルを使用できます。

CONTAINER(data-value)

削除するコンテナの名前 (1 から 16 文字) を指定します。

RETCODE(data-area)

外部 CICS インターフェースが戻り情報を配置する 20 バイト領域を指定します。この領域は、5 つの 1 ワード・フィールドにフォーマット設定されます。このコマンドは、最初の 2 ワードに情報を返します。これらのワードには、RESP コードと RESP2 コードが含まれます。他の 3 ワードは、ヌルに設定されます。

RESP

外部 CICS インターフェースの **DELETE CONTAINER** コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

条件

122 CHANNELERR

RESP2 値:

2

CHANNEL オプションで指定されたチャンネルが見つかりませんでした。

3

CHANNEL オプションで指定されたチャンネルは、読み取り専用です。

904

XCEIP ESTAE が呼び出されました。

110 CONTAINERERR

RESP2 値:

10

CONTAINER オプションで指定されたコンテナが見つかりませんでした。

16 INVREQ

RESP2 値:

4

コマンドが、現在アクティブになっているチャンネルの有効範囲外で発行されました。

30

CICS で定義された読み取り専用コンテナは削除できません。

EXEC CICS ENDBROWSE CONTAINER コマンド (EXCI)

チャンネルに関連したコンテナのブラウズを終了します。

ENDBROWSE CONTAINER (EXCI)

➡ ENDBROWSE — CONTAINER — BROWSETOKEN(*data-value*) — RETCODE(*data-area*) →

状態: TOKENERR

説明

ENDBROWSE CONTAINER (EXCI) は、チャンネルに関連付けられているコンテナのブラウズを終了し、ブラウズ・トークンを無効にします。

オプション

BROWSETOKEN(data-value)

削除するブラウズ・トークンをフルワード・バイナリー値で指定します。

RETCODE(data-area)

外部 CICS インターフェースが戻り情報を配置する 20 バイト領域を指定します。この領域は、5 つの 1 ワード・フィールドにフォーマット設定されます。このコマンドは、最初の 2 ワードに情報を返します。これらのワードには、RESP コードと RESP2 コードが含まれます。他の 3 ワードは、ヌルに設定されます。

RESP

外部 CICS インターフェースの **ENDBROWSE CONTAINER** コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

条件

112 TOKENERR

RESP2 値:

3

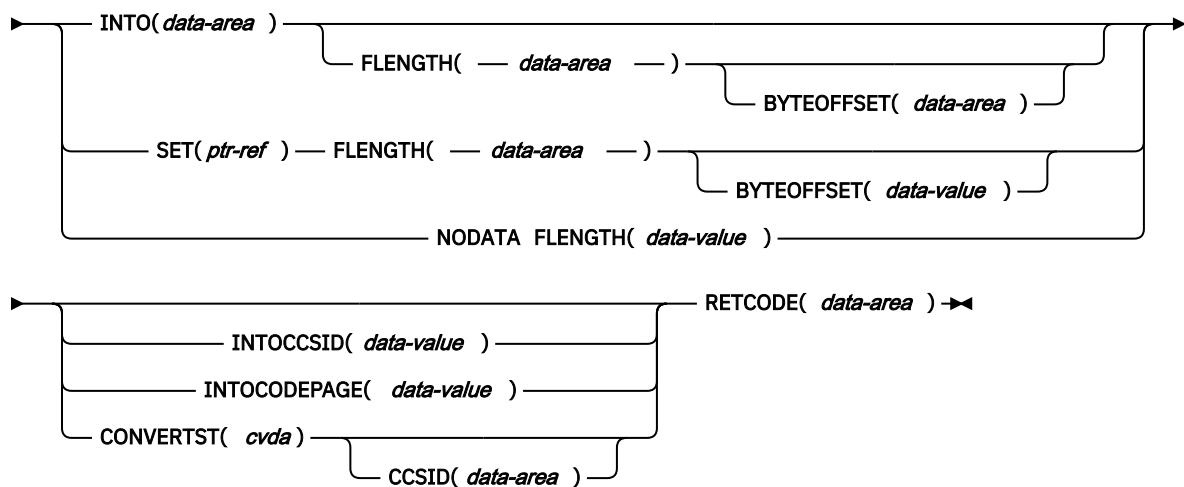
ブラウズ・トークンが無効です。

EXEC CICS GET CONTAINER コマンド (EXCI)

指定されたチャンネル・コンテナからデータを取り出します。

GET CONTAINER (EXCI)

➡ GET — CONTAINER(*data-value*) — CHANNEL(*data-value*) →



状態: CCSIDERR、CHANNELERR、CODEPAGEERR、CONTAINERERR、INVREQ、LENGERR

説明

GET CONTAINER (EXCI) は、指定されたチャンネル・コンテナに関連付けられているデータを読み取ります。

データを保持するコンテナは、名前、およびコンテナとなっているチャンネル、つまりコンテナを「所有」しているチャンネルによって識別されます。

オプション

BYTEOFFSET(data-value)

データを返す際の開始オフセット (バイト数) を指定します。このオフセットからのデータが返されます。CHAR コンテナの場合、要求されたコード・ページのデータへのオフセットとして、**BYTEOFFSET** 値が使用されます。マルチバイト文字を使用するコード・ページを使用している場合、指定する **BYTEOFFSET** 値によっては、返されるデータの先頭、末尾、またはその両方が部分的な文字になる可

能性があります。この場合、アプリケーション・プログラムに、返されたデータを処理して解釈する機能が必要です。指定された値がゼロより小さい場合は、ゼロと見なされます。

CCSID(data-area)

CONVERTST(NOCONVERT) オプションによって返されるデータのコード化文字セット ID (CCSID) を含むフルワードを返します。このオプションを使用すると、DATATYPE に CHAR が指定されたコンテナを、データを変換することなく取得できます。コンテナの DATATYPE に BIT が指定されている場合、この値はゼロになります。

CHANNEL(data-value)

コンテナを所有するチャンネルの名前 (1 から 16 文字) を指定します。チャンネル名 DFHTRANSACTION を指定してトランザクション・チャンネルを使用できます。

CONTAINER(data-value)

取り出されるデータを保持するコンテナの名前 (1 から 16 文字) を指定します。

CONVERTST(cvda)

必要なデータ変換状況を指定します。

NOCONVERT

コンテナ・データが変換されずに取り出されます。

FLENGTH(data-area)

入力フィールドとしての **FLENGTH** は、読み取られるデータの長さをフルワード・バイナリー値で指定します。出力フィールドとしての **FLENGTH** は、コンテナ内のデータの長さを返します。**FLENGTH** は、**BYTEOFFSET**、**INTO**、**SET**、または **NODATA** のどのオプションを指定するかによって、入力フィールドまたは出力フィールドになります。

BYTEOFFSET オプションが指定されている場合

FLENGTH は入力フィールドにも出力フィールドにもなります。

入力フィールドの場合、**FLENGTH** にはプログラムが受け入れるデータの最大長を指定します。

BYTEOFFSET 値で指定されたオフセットを開始点としてデータが返されます。指定された値がゼロより小さい場合は、ゼロと見なされます。

出力フィールド (すなわち、取得操作の完了時) には、CICS は、データ域を返されたデータの長さに設定します。返されるデータの最大長は、コンテナ内のデータの長さから **BYTEOFFSET** 値を引いた長さと同じです。

INTO オプションが指定されている場合

FLENGTH は入力フィールドにも出力フィールドにもなります。

入力フィールドの場合、**FLENGTH** にはプログラムが受け入れるデータの最大長を指定します。指定された値がゼロより小さい場合は、ゼロと見なされます。データの長さが指定値を超えると、データはその値に切り捨てられて、LENGERR 条件が起こります。データの長さが指定された値を下回る場合、データはコピーされますが、埋め込みは実行されません。

INTO 変数からコンパイラによって長さを生成できる場合は、**FLENGTH** を指定する必要はありません。**INTO** および **FLENGTH** の両方が指定された場合、**FLENGTH** はプログラムが受け入れるデータの最大長を指定します。

出力フィールドの場合 (すなわち、取り出し操作の完了時に)、データ域が指定されている場合は、CICS はそのデータ域をコンテナ内のデータの実際の長さに設定します。ある CCSID から別の CCSID に変換された文字データをコンテナが保持している場合、これは変換後のデータ長になります。

SET または NODATA オプションが指定されている場合

FLENGTH は出力専用フィールドになります。このフィールドが存在し、データ域として指定する必要があります。

取り出し操作が完了すると、データ域はコンテナ内のデータの実際の長さに設定されます。ある CCSID から別の CCSID に変換された文字データをコンテナが保持している場合、これは変換後のデータ長になります。

INTO(data-area)

取得したデータを配置するデータ域を指定します。

INTOCCSID(data-value)

コンテナ内の文字データの変換先のコード化文字セット ID (CCSID) をフルワード・バイナリー数値として指定します。コード・ページに IANA 名を指定する場合、または CCSID を英数字で指定する場合は、代わりに **INTOCODEPAGE** オプションを使用します。

CICS Transaction Server for z/OS アプリケーションの場合、CCSID は通常、EBCDIC CCSID です。ただし、例えば、ASCII データを EBCDIC に自動変換せずに取り出す場合は、ASCII CCSID を指定することができます。

INTOCCSID と **INTOCODEPAGE** が指定されていない場合、変換の値は、デフォルトである EXCI ジョブの CCSID になります。EXCI ジョブのデフォルトの CCSID は DFHXCOPT の **LOCALCCSID** パラメーターに指定されています。

変換できるのは文字データのみであり、さらにデータをコンテナに配置するために使用される **PUT CONTAINER** コマンドで DATATYPE として CHAR が指定された場合だけです。**PUT CONTAINER** コマンドに **FROMCCSID** または **FROMCODEPAGE** が指定されている場合は、DATATYPE として CHAR が暗黙指定されます。

チャンネルでのデータ変換の詳細については、[チャンネルを使用したデータ変換](#)を参照してください。

CCSID の詳細については、[チャンネルを使用したコード・ページ変換の準備](#)を参照してください。

INTOCODEPAGE(data-value)

適切な句読点を含む、最大 40 文字の英数字を使用して、IANA に登録されている英数字文字セット名、またはコンテナ内の文字データの変換先のコード・ページのコード化文字セット ID (CCSID) を指定します。HTTP 要求のコンテンツ・タイプ・ヘッダーで指定されている IANA 登録文字セット名を使用する場合は、CCSID オプションの代わりにこのオプションを指定します。CICS は IANA 名を CCSID に変換します。後続のデータ変換プロセスも同様です。また、フルワード・バイナリー値としてではなく、英数字で CCSID を指定する場合もこのオプションを使用します。

コード・ページの IANA 名が存在して、CICS がその使用をサポートする場合、その名前が CCSID と共にリストされます。詳しくは、[チャンネルを使用したコード・ページ変換の準備](#)を参照してください。

NODATA

取得するデータがないことを指定します。このオプションを使用して、コンテナ内の (**FLENGTH** で返される) データの長さを検出します。

データ変換が実行されると、文字データの長さが変化することがあります。したがって、文字データをこの領域の CCSID 以外の CCSID に変換する場合は、**NODATA** を指定するときに **INTOCCSID** も指定する必要があります。これにより、変換されたデータの正しい長さが **FLENGTH** で返されることが保証されます。

RETCODE(data-area)

外部 CICS インターフェースが戻り情報を配置する 20 バイト領域を指定します。この領域は、5 つの 1 ワード・フィールドにフォーマット設定されます。このコマンドは、最初の 2 ワードに情報を返します。これらのワードには、RESP コードと RESP2 コードが含まれます。他の 3 ワードは、ヌルに設定されます。

RESP

外部 CICS インターフェースの **GET CONTAINER** コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

SET(ptr-ref)

取り出されたデータのアドレスが返されるデータ域を指定します。

外部 CICS インターフェースは、以下のいずれかが発生するまでデータ域を維持します。

- このストレージにアクセスできる任意のプログラムが、同じチャンネル内の同じコンテナに対して **SET** オプションを指定した後続の **GET CONTAINER** コマンドを発行した場合。

- コンテナが **DELETE CONTAINER** コマンドによって削除された。
 - コンテナが **MOVE CONTAINER** コマンドによって移動された。
 - このチャンネルと、その中にあるコンテナが **DELETE CHANNEL** コマンドによって削除されます。
- これらのコマンドを発行する可能性がある他のプログラムとのリンクには注意してください。

FREEMAIN コマンドを使用してこのストレージを解放しないでください。

ご使用のアプリケーションでデータを保持する必要がある場合は、アプリケーションの独自のストレージにデータを移動する必要があります。

条件

123 CCSIDERR

RESP2 値:

- 1 **INTOCCSID** オプションで指定された CCSID が有効な CCSID 値の範囲外です。
- 2 **INTOCCSID** オプションで指定された CCSID とコンテナの CCSID の組み合わせはサポートされていません。(コンテナの CCSID は、コンテナの作成時に **FROMCODEPAGE** または **FROMCCSID** コマンドを使用して指定された、またはデフォルト指定された値です。)
- 3 データが BIT データ型で作成されました。コード・ページ変換を行うことはできません。コード・ページ変換されずにデータが返されました。
- 4 1 つ以上の文字を変換できませんでした。変換されなかった文字は、変換されたデータ内ではブラックに置き換えられています。
- 5 コンテナのコード・ページ変換で内部エラーが発生しました。

122 CHANNELERR

RESP2 値:

- 2 **CHANNEL** オプションで指定されたチャンネルが見つかりませんでした。
- 904 XCEIP ESTAE が呼び出されました。

125 CODEPAGEERR

RESP2 値:

- 1 **INTOCODEPAGE** オプションで指定されたコード・ページはサポートされていません。
- 2 **INTOCODEPAGE** オプションで指定されたコード・ページとチャンネルのコード・ページの組み合わせはサポートされていません。
- 3 データが BIT データ型で作成されました。コード・ページ変換を行うことはできません。コード・ページ変換されずにデータが返されました。
- 4 1 つ以上の文字を変換できませんでした。変換されなかった文字は、変換されたデータ内ではブラックに置き換えられています。
- 5 コンテナのコード・ページ変換で内部エラーが発生しました。

110 CONTAINERERR

RESP2 値:

10

CONTAINER オプションで指定されたコンテナが見つかりませんでした。

16 INVREQ

RESP2 値:

2

INTOCCSID オプションが指定されましたが、**CHANNEL** オプションが指定されていないので、現行チャンネルはありません (このコマンドを発行したプログラムに **CHANNEL** オプションが渡されなかったため)。**INTOCCSID** は、チャンネルを指定する **GET CONTAINER** コマンドでのみ有効です。

4

CHANNEL オプションが指定されておらず、(このコマンドを発行したプログラムに渡されていないために) 現行チャンネルが存在しておらず、現在アクティブになっている BTS アクティビティの有効範囲外でコマンドが発行されました。

5

CONVERTST cvda 値が無効です。

22 LENGERR

RESP2 値:

11

プログラム域の長さは、コンテナ内のデータの長さよりも短くなっています。プログラム域の方が小さい場合、データはプログラム域に収まるよう切り捨てられます。

12

オフセットが、コンテナの長さ以上になっています。

EXEC CICS GETNEXT CONTAINER コマンド (EXCI)

チャンネルに関連したコンテナをブラウズします。

GETNEXT CONTAINER (EXCI)

➡ GETNEXT — CONTAINER(*data-area*) — BROWSETOKEN(*data-value*) — RETCODE(*data-area*) ➡

状態: END、TOKENERR

説明

GETNEXT CONTAINER (EXCI) は、チャンネルに関連付けられている次のコンテナの名前を返します。

注:

1. **GETNEXT CONTAINER (EXCI)** コマンドを連続して使用すると、**STARTBROWSE CONTAINER (EXCI)** コマンドの実行時に存在していたチャンネルのすべてのコンテナの名前を取得できます。ただし、**STARTBROWSE** コマンドの後で、かつ **GETNEXT** コマンドによって返される前に削除されるコンテナの名前は返されません。
2. **STARTBROWSE** コマンドの実行後にこのチャンネルまたはアクティビティで作成された (または移動された) コンテナの名前は、返される場合と返されない場合があります。
3. コンテナが返される順序は未定義であり、変更される可能性があります。ベスト・プラクティスとして、アプリケーションは、返されるコンテナの順序に依存しないようにする必要があります。

オプション

BROWSETOKEN(*data-value*)

前の **STARTBROWSE CONTAINER (EXCI)** コマンドで返されたブラウズ・トークンをフルワード・バイナリー値で指定します。

CONTAINER(*data-area*)

次の *data-container* の 16 文字の名前を返します。

RETCODE(data-area)

外部 CICS インターフェースが戻り情報を配置する 20 バイト領域を指定します。この領域は、5 つの 1 ワード・フィールドにフォーマット設定されます。このコマンドは、最初の 2 ワードに情報を返します。これらのワードには、RESP コードと RESP2 コードが含まれます。他の 3 ワードは、ヌルに設定されます。

RESP

外部 CICS インターフェースの **GETNEXT CONTAINER** コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

条件

83 END

RESP2 値:

2

これ以上のコンテナはありません。

112 TOKENERR

RESP2 値:

3

ブラウズ・トークンが無効です。

EXEC CICS MOVE CONTAINER コマンド (EXCI)

あるチャンネルから別のチャンネルへ、コンテナ (および、その内容) を移動します。

MOVE CONTAINER (EXCI)

```
➡ MOVE — CONTAINER( data-value ) — AS( data-value ) — CHANNEL( data-value ) →  
  
    ➡ TOCHANNEL( data-value ) — RETCODE( data-area ) ➡
```

状態: CHANNELERR、CONTAINERERR、INVREQ

説明

MOVE CONTAINER (EXCI) は、あるチャンネルから別のチャンネルにコンテナを移動します。移動後、ソース・コンテナは存在しなくなります。

ソース・コンテナおよびターゲット・コンテナは、名前、およびそれらのコンテナを所有するチャンネルで識別されます。ソース・コンテナを所有するチャンネルは、**CHANNEL** オプションで識別されます。ターゲット・コンテナを所有するチャンネルは、**TOCHANNEL** オプションで識別されます。

コンテナは、以下のように移動できます。

- あるチャンネルから別のチャンネルへ
- 同じチャンネル内で (この結果、コンテナの名前が変更されます。)

チャンネル間でのデータ移動をより効果的に行う方法として、**GET CONTAINER** および **PUT CONTAINER** の代わりに **MOVE CONTAINER** を使用することもできます。

注:

1. ソース・チャンネルは、**MOVE CONTAINER** コマンドを発行するプログラムの有効範囲内になければなりません。
2. **MOVE CONTAINER** コマンドを発行するプログラムの有効範囲にターゲット・チャンネルが存在していない場合は、ターゲット・チャンネルが作成されます。
3. ソース・コンテナが存在しない場合は、エラーが発生します。

4. ターゲット・コンテナがまだ存在していない場合は、ターゲット・コンテナが作成されます。ターゲット・コンテナが既に存在している場合は、そのコンテナの以前の内容が上書きされます。
5. コンテナをそれ自体で上書きしようとしても、何も起こりません。つまり、**CONTAINER** オプションと **AS** オプションに同じ値を指定し、**CHANNEL** オプションと **TOCHANNEL** オプションに同じ値を指定することにより、同じチャンネルが指定されるようにしても、ソース・コンテナは変更も削除もされません。エラー状態は発生しません。

オプション

AS(data-value)

ターゲット・コンテナの名前 (1 文字から 16 文字) を指定します。ターゲット・コンテナが既に存在している場合は、その内容が上書きされます。

許容文字は A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - および _ です。ブランク文字を先頭にしたり、埋め込んだりすることはできません。指定された名前が 16 文字よりも短い場合は、16 文字になるまで末尾にブランクが埋め込まれます。

コンテナ名は常に EBCDIC です。前述の、コンテナ名に使用できる文字セットの中には、すべての EBCDIC コード・ページで同じ表示になるとは限らない文字があります。したがって、領域間でコンテナをシップする場合は、命名で使用する文字を A-Z a-z 0-9 & : = , ; < > . - および _ に制限することを推奨します。

CHANNEL(data-value)

ソース・コンテナを所有するチャンネルの名前 (1 文字から 16 文字) を指定します。チャンネル名 DFHTRANSACTION を指定してトランザクション・チャンネルを使用できます。

CONTAINER(data-value)

移動するソース・コンテナの名前 (1 文字から 16 文字) を指定します。

RETCODE(data-area)

外部 CICS インターフェースが戻り情報を配置する 20 バイト領域を指定します。この領域は、5 つの 1 ワード・フィールドにフォーマット設定されます。このコマンドは、最初の 2 ワードに情報を返します。これらのワードには、RESP コードと RESP2 コードが含まれます。他の 3 ワードは、ヌルに設定されます。

RESP

外部 CICS インターフェースの **MOVE CONTAINER** コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

TOCHANNEL(data-value)

ターゲット・コンテナを所有するチャンネルの名前 (1 文字から 16 文字) を指定します。新規チャンネルを指定する場合、許容文字は、A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - および _ です。ブランク文字を先頭にしたり、埋め込んだりすることはできません。指定された名前が 16 文字よりも短い場合は、16 文字になるまで末尾にブランクが埋め込まれます。チャンネルが存在していない場合は、そのチャンネルが作成されます。この新しいチャンネルは、リンク・レベルが変更されるまで、有効範囲に入ったままになります。チャンネルの有効範囲について詳しくは、[チャンネルの有効範囲](#)を参照してください。

チャンネル名は常に EBCDIC です。前述の、チャンネル名に使用できる文字セットの中には、すべての EBCDIC コード・ページで同じ表示になるとは限らない文字があります。したがって、領域間でチャンネルをシップする場合は、名前に使用する文字を A-Z a-z 0-9 & : = , ; < > . に制限することを推奨します。-, および _ に制限することをお勧めします。

チャンネル名 DFHTRANSACTION を指定してトランザクション・チャンネルを使用できます。リンク・レベルが変更されても、トランザクション・チャンネルは有効範囲から外れません。これは、そのタスクで常にアクセス可能です。詳しくは、[チャンネルおよびコンテナ](#)を参照してください。

条件

122 CHANNELERR

RESP2 値:

1 **TOCHANNEL** オプションで指定された名前に、正しくない文字または文字の組み合わせが含まれています。

2 **CHANNEL** オプションで指定されたチャンネルが見つかりませんでした。

3 **CHANNEL** オプションで指定されたチャンネルは、読み取り専用です。

904 XCEIP ESTAE が呼び出されました。

110 CONTAINERERR

RESP2 値:

10 **CONTAINER** オプションで指定されたコンテナが見つかりませんでした。

18 **AS** オプションで指定された名前に、正しくない文字または文字の組み合わせが含まれています。

16 INVREQ

RESP2 値:

30 CICS で定義されている読み取り専用コンテナは移動できません。

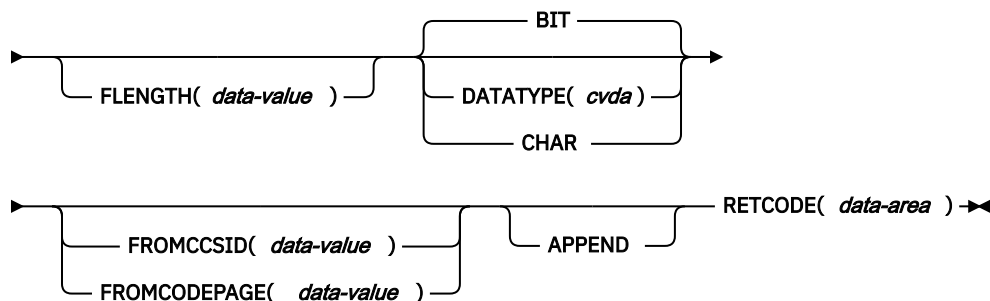
31 CICS で定義されている、既存の読み取り専用コンテナにコンテナを移動する (すなわち、上書きする) ことはできません。

EXEC CICS PUT CONTAINER コマンド (EXCI)

指定されたチャンネル・コンテナにデータを配置します。

PUT CONTAINER (EXCI)

➡ PUT — CONTAINER(*data-value*) — CHANNEL(*data-value*) — FROM(*data-area*) —➡



状態: CCSIDERR、CHANNELERR、CODEPAGEERR、CONTAINERERR、INVREQ、LENGERR

説明

PUT CONTAINER (EXCI) は、指定されたチャンネルに関連付けられているコンテナにデータを配置します。

コンテナは名前で識別されます。コンテナを所有するチャンネルは、**CHANNEL** オプションで識別されます。

指定されたコンテナが存在していない場合は、そのコンテナが作成されます。**APPEND** オプションを指定しない限り、指定されたコンテナが存在している場合は、そのコンテナの以前の内容が上書きされます。指定されたチャンネルが存在していない場合は、そのチャンネルが作成されます。

1つのチャンネルに関連付けることができるコンテナの数に制限はありません。個々のコンテナのサイズは、使用可能なストレージの量によってのみ制限されます。コンテナ・データは 64 ビット・ストレージに保持されます。EXCI ジョブで利用できる 64 ビット・ストレージの合計量は **MEMLIMIT** によって管理され、外部 CICS インターフェースによって、コンテナ・データの合計量が **MEMLIMIT** 値の 5% に制限されます。



注意: ターゲット CICS プログラムが非 LE アセンブラー amode-64 アプリケーションでない場合に、複数の大きなコンテナを作成して CICS に渡すと、コンテナ・データは、CICS アプリケーションでアクセスする際に 31 ビット・ストレージにコピーされます。したがって、コピーされたデータを入れるために十分な 31 ビット・ストレージが使用可能でなければならず、他の CICS アプリケーションで利用できるストレージの量が制限される場合があります。

オプション

APPEND

コンテナに渡されるデータを、コンテナ内の既存の内容に追加することを指定します。このオプションが指定されていない場合、コンテナに渡されたデータによって、コンテナ内の既存の内容が上書きされます。

CHANNEL(*data-value*)

コンテナを所有するチャンネルの名前 (1 から 16 文字) を指定します。許容文字は A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - および _ です。ブランク文字を先頭にしたり、埋め込んだりすることはできません。指定された名前が 16 文字よりも短い場合は、16 文字になるまで末尾にブランクが埋め込まれます。

チャンネル名は常に EBCDIC です。前述の、チャンネル名に使用できる文字セットには、EBCDIC コード・ページによっては異なる表示となる文字も一部含まれています。したがって、領域間でチャンネルをシッップする場合は、名前に使用する文字を A-Z a-z 0-9 & : = , ; < > . に制限することを推奨します。- および _ です。

チャンネルが存在していない場合は、そのチャンネルが作成されます。チャンネル名 DFHTRANSACTION を指定して、CICS でトランザクション・チャンネルを使用できます。EXCI でのトランザクション・チャンネルの使用について詳しくは、[チャンネルおよびコンテナ](#)を参照してください。

CONTAINER(*data-value*)

データが配置されるコンテナの名前 (1 から 16 文字) を指定します。

許容文字は A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - および _ です。ブランク文字を先頭にしたり、埋め込んだりすることはできません。指定された名前が 16 文字よりも短い場合は、16 文字になるまで末尾にブランクが埋め込まれます。

CICS によって要求された場合以外は、DFH で始まるコンテナ名を使用しないでください。

コンテナ名は常に EBCDIC です。前述の、コンテナ名に使用できる文字セットには、EBCDIC コード・ページによっては異なる表示となる文字も一部含まれています。したがって、領域間でコンテナをシッップする場合は、名前に使用する文字を A-Z 0-9 & : = , ; < > . に制限することを推奨します。-、および _ に制限することをお勧めします。

DATATYPE(*cvda*)

コンテナに書き込むデータのタイプを指定します。このオプションは、新規のコンテナのみに適用されます。コンテナが存在する場合、コンテナのデータ・タイプは作成時に設定されているため、変更できません。CVDA 値は、以下のとおりです。

BIT

ビット・データ。コンテナ内のデータは変換できません。**FROMCCSID** が指定されていない場合は、これがデフォルト値になります。

CHAR

文字データ。コンテナに格納するデータは、**FROMCCSID** 値または **FROMCODEPAGE** 値の設定に基づいて (必要に応じて) 変換されます。**FROMCCSID** オプションと **FROMCODEPAGE** オプションを指定しない場合は、DFHXCOPT の **LOCALCCSID** パラメーターで指定されている EXCI ジョブの CCSID でデータがエンコードされることになります。

コンテナ内のすべてのデータが、単一の文字ストリングとして変換されます。SBCS コード・ページの場合、複数の文字フィールドで構成される構造は、1 バイト文字ストリングと同等です。ただし、DBCS コード・ページの場合は、これには該当しません。DBCS コード・ページを使用する場合は、データ変換が適切に行われるためには、それぞれの文字ストリングを別個のコンテナに書き込む必要があります。

FLENGTH(data-value)

データを読み取るデータ域の長さをフルワード・バイナリー値で指定します。

FROM(data-area)

コンテナへのデータの書き込み元であるデータ域を指定します。

FROMCCSID(data-value)

コンテナに挿入する文字データの現行のコード化文字セット ID (CCSID) をフルワード・バイナリー値で指定します。コード・ページに IANA 名を指定する場合、または CCSID を英数字で指定する場合は、代わりに **FROMCODEPAGE** オプションを使用します。このオプションを使用するのは、DFHXCOPT の **LOCALCCSID** パラメーターで指定されているように、コンテナに配置するデータが EXCI ジョブの CCSID でエンコードされない場合です。

FROMCCSID オプションを指定した場合は、DATATYPE(DFHVALUE(CHAR)) が暗黙指定されます。

FROMCODEPAGE(data-value)

適切な句読点を含む、最大 40 文字の英数字を使用して、IANA に登録されている英数字文字セット名、またはコンテナに挿入する文字データの現行のコード・ページのコード化文字セット ID (CCSID) を指定します。HTTP 要求のコンテンツ・タイプ・ヘッダーで指定されている IANA 登録文字セット名を使用する場合は、CCSID オプションの代わりにこのオプションを指定します。CICS は IANA 名を CCSID に変換します。後続のデータ変換プロセスも同様です。また、フルワード・バイナリー値としてではなく、英数字で CCSID を指定する場合もこのオプションを使用します。

FROMCODEPAGE オプションを指定した場合は、DATATYPE(DFHVALUE(CHAR)) が暗黙指定されます。

RETCODE(data-area)

外部 CICS インターフェースが戻り情報を配置する 20 バイト領域を指定します。この領域は、5 つの 1 ワード・フィールドにフォーマット設定されます。このコマンドは、最初の 2 ワードに情報を返します。これらのワードには、RESP コードと RESP2 コードが含まれます。他の 3 ワードは、ヌルに設定されます。

RESP

外部 CICS インターフェースの **PUT CONTAINER** コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

条件

123 CCSIDERR

RESP2 値:

1

FROMCCSID オプションで指定された CCSID が有効な CCSID 値の範囲外です。

2

FROMCCSID オプションで指定された CCSID とコンテナの CCSID の組み合わせはサポートされていません。コンテナの CCSID は、このコンテナに対して最初の **PUT CONTAINER** コマンドで指定またはデフォルト設定された値です。無効な各組み合わせの初回使用時に、CICS はエラー・メッセージ DFHAP0802 を発行します。このメッセージには、CCSID のペアが含まれています。

4

1 つ以上の文字を変換できませんでした。変換されなかった各文字は、変換されたデータ内ではブランクに置き換えられています。

5

コンテナのコード・ページ変換で内部エラーが発生しました。このエラーは、PUT のターゲットが CICS によって作成された既存のコンテナである場合にのみ発生する可能性があります。

122 CHANNELERR

RESP2 値:

1

CHANNEL オプションで指定された名前に、正しくない文字または文字の組み合わせが含まれています。

3

CHANNEL オプションで指定されたチャンネルは、読み取り専用です。

904

XCEIP ESTAE が呼び出されました。

125 CODEPAGEERR

RESP2 値:

1

FROMCODEPAGE オプションで指定されたコード・ページはサポートされていません。

2

FROMCODEPAGE オプションで指定されたコード・ページとコンテナの CCSID の組み合わせはサポートされていません。コンテナの CCSID は、このコンテナに対する最初の **PUT CONTAINER** コマンドで、**FROMCODEPAGE** または **FROMCCSID** のいずれかを使用して指定された、またはデフォルト設定された値です。無効な各組み合わせの初回使用時に、CICS はエラー・メッセージ DFHAP0802 を発行します。このメッセージには、CCSID のペアが含まれています。

4

1 つ以上の文字を変換できませんでした。変換されなかった各文字は、変換されたデータ内ではブランクに置き換えられています。このエラーは、PUT のターゲットが既存のコンテナである場合にのみ発生する可能性があります。

5

コンテナのコード・ページ変換で内部エラーが発生しました。このエラーは、PUT のターゲットが CICS によって作成された既存のコンテナである場合にのみ発生する可能性があります。

110 CONTAINERERR

RESP2 値:

18

CONTAINER オプションで指定された名前に、正しくない文字または文字の組み合わせが含まれています。

16 INVREQ

RESP2 値:

1

DATATYPE オプションが指定されましたが、**CHANNEL** オプションが指定されていないので、現行チャンネルはありません (このコマンドを発行したプログラムに **CHANNEL** オプションが渡されなかったため)。**DATATYPE** は、(明示的にまたは暗黙的に) チャンネルを指定する **PUT CONTAINER** コマンドでのみ有効です。

2

FROMCCSID オプションが指定されましたが、**CHANNEL** オプションが指定されていないので、現行チャンネルはありません (このコマンドを発行したプログラムに **CHANNEL** オプションが渡されなかったため)。**FROMCCSID** は、(明示的にまたは暗黙的に) チャンネルを指定する **PUT CONTAINER** コマンドでのみ有効です。

30

CICS で定義されている読み取り専用コンテナに書き込みを行おうとしました。

32

DATATYPE に CHAR および BIT 以外の CVDA 値が指定されていました。

33

既存のコンテナのデータ型を変更しようとしてしました。

34

BIT のデータ型は、CCSID では無効です。

22 LENGERR

RESP2 値:

1

FLENGTH オプションに負の値が指定されていました。

EXEC CICS QUERY CHANNEL コマンド (EXCI)

チャンネル内のコンテナの数をカウントします。

QUERY CHANNEL (EXCI)

►► QUERY CHANNEL(*data-value*) — CONTAINERCNT(*data-area*) — RETCODE(*data-area*)►◄

条件: CHANNELERR

説明

QUERY CHANNEL (EXCI) は、指定チャンネル内のコンテナの数をカウントします。CHANNEL オプションを指定して、チャンネルを明示的に指定する必要があります。**QUERY CHANNEL** コマンドは、どのチャンネルでも使用できます。これには、トランザクション・チャンネル (DFHTRANSACTION) や EXCI が作成したチャンネルも含まれます。

オプション

CHANNEL(*data-value*)

チャンネルの名前を 1 から 16 文字で指定します。

CONTAINERCNT(*data-area*)

指定されたチャンネル内のコンテナ数のカウントを、フルワード・バイナリー値で返します。

RETCODE(*data-area*)

外部 CICS インターフェースが戻り情報を配置する 20 バイト領域を指定します。この領域は、5 つの 1 ワード・フィールドにフォーマット設定されます。このコマンドは、最初の 2 ワードに情報を返します。これらのワードには、RESP コードと RESP2 コードが含まれます。他の 3 ワードは、ヌルに設定されます。

RESP

外部 CICS インターフェースの **QUERY CHANNEL** コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

条件

122 CHANNELERR

RESP2 値:

2

CHANNEL オプションで指定されたチャンネルが見つかりませんでした。

EXEC CICS STARTBROWSE CONTAINER コマンド (EXCI)

チャンネルに関連したコンテナのブラウズを開始します。

STARTBROWSE CONTAINER (EXCI)

►► STARTBROWSE — CONTAINER — CHANNEL(*data-value*) — BROWSETOKEN(*data-area*)►

►— RETCODE(*data-area*)►◄

条件: CHANNELERR

説明

STARTBROWSE CONTAINER (EXCI) は、指定されたチャンネルに関連付けられた各データ・コンテナの名前の識別に使用できるブラウザ・トークンを初期化します。

注：ブラウザ・トークンは、**STARTBROWSE** コマンドを発行するプログラムでのみ使用してください。

コンテナが返される順序は未定義であり、変更される可能性があります。ベスト・プラクティスとして、アプリケーションは、返されるコンテナの順序に依存しないようにする必要があります。

オプション

BROWSETOKEN(data-area)

CICS がブラウザ・トークンを入れるフルワード・バイナリー・データ域を指定します。

CHANNEL(data-value)

コンテナをブラウザするチャンネルの名前 (1 から 16 文字) を指定します。これは、**STARTBROWSE CONTAINER** コマンドを発行するプログラムによって作成されるチャンネルの名前でなければなりません。チャンネル名 DFHTRANSACTION を指定してトランザクション・チャンネルを使用できます。

コンテナが戻される順番は定義されません。

RETCODE(data-area)

外部 CICS インターフェースが戻り情報を配置する 20 バイト領域を指定します。この領域は、5 つの 1 ワード・フィールドにフォーマット設定されます。このコマンドは、最初の 2 ワードに情報を返します。これらのワードには、RESP コードと RESP2 コードが含まれます。他の 3 ワードは、ヌルに設定されます。

RESP

外部 CICS インターフェースの **STARTBROWSE CONTAINER** コマンドの実行中に例外条件が発生したかどうかを示す 1 次応答コード。

RESP2

RESP パラメーターで発生した条件の一部を、必要に応じてさらに修飾する 2 次応答コード。

条件

122 CHANNELERR

RESP2 値:

2

CHANNEL オプションで指定されたチャンネルが見つかりませんでした。

EXCI クライアント・プログラムのコンパイルとリンク・エディット

外部 CICS インターフェースを使用して DPL 要求を CICS サーバー領域に渡すすべてのプログラムには、CICS 提供のプログラム・スタブ DFHXCSTB が組み込まれていなければなりません。

COBOL プログラムの場合は、スタブをプログラムとリンク・エディットするのではなく、DFHXCSTB を動的に呼び出すことができます。

このスタブは、EXCI CALL インターフェース・コマンドであるか、EXEC CICS インターフェース・コマンドであるかに関わらず、すべての外部 CICS インターフェース・コマンドを代行受信し、それらが該当する外部 CICS インターフェース・ルーチンに渡されて、確実に処理されるようにします。

DFHXCSTB は共通スタブで、サポートされているすべての言語で作成されたプログラムに組み込めるように設計されています。このスタブは、CICSTS56.CICS.SDFHEXCI ライブラリー内にあります。

注：CICSTS56.CICS.SDFHEXCI には、DFHXCSTB の別名である DFHXCIE と DFHXCIS に関するエントリーも含まれています。

確実にスタブを組み込むのに役立つように、CICS は複数のプロシージャール (言語ごとに 1 つずつ) を提供しているので、変換、コンパイル、リンク・エディットに使用できます。

EXCI クライアント・プログラム用に AMODE(31) を指定する必要があります。

CICS 提供の、クライアント・プログラムをコンパイルしてリンク・エディットするためのプロシージャには、リンケージ・エディター・ジョブ・ステップの PARM ステートメントに以下のパラメーターがあります。

```
LNKPARM='AMODE(31),LIST,XREF'
```

サンプル

CICS は、外部 CICS インターフェースを使用するプログラムの作成に役立つように、サンプルの MVS クライアント・プログラムとサンプルの CICS サーバー・プログラムを提供しています。これらのサンプルは、**EXCI CALL** インターフェースと **EXEC CICS LINK** コマンドの両方を使用するクライアント・アプリケーションをコーディングする方法を示しています。詳しくは、[EXCI サンプル・プログラム](#)を参照してください。

EXCI クライアント・プログラムを実行するためのジョブ制御言語

EXCI クライアント・プログラムは、例えばバッチ・ジョブとして、MVS アドレス・スペースで実行されます。クライアント・プログラムの JCL を記述する際は、以下の要件に注意してください。

要件

- CICS 提供の外部 CICS インターフェース・モジュールおよびクライアント・プログラムが含まれているライブラリーを STEPLIB 連結に含めます。外部 CICS インターフェース・モジュールは CICSTS56.CICS.SDFHEXCI で提供され、これには以下が含まれます。

```
DFH$ATCC  
DFH$AXCC  
DFH$AXNC  
DFH$DXVC  
DFHMEBMX  
DFHMET4E  
DFHXCEIX  
DFHXCIE (DFHXCSTB の別名)  
DFHXCIS (DFHXCTSB の別名)  
DFHXCOPT  
DFHXCPRX  
DFHXCSTB  
DFHXCTRA  
DFHXCURM
```

- SYSMDUMP の DD ステートメントを含めることをお勧めします。外部 CICS インターフェースは、いくつかのエラー条件に対して SYSMDUMP を使用します。
- **REGION** パラメーターは、DFHXCOPT オプション・テーブルの **TRACESZE** パラメーターに指定されている内部トレース・テーブルのサイズが収まるように、十分な大きさの領域サイズを指定する必要があります。
- EXCI クライアント・プログラムが、COMMAREA ではなく、チャンネルとコンテナを使用して CICS にデータを渡す場合は、コンテナ・データが 2 GB 境界より上の 64 ビット・ストレージに保管されるため、**MEMLIMIT** パラメーターを指定する必要があります。コンテナのストレージは、**MEMLIMIT** 値の 5% を超えることはできません。
- クライアント・プログラムからの出力用の SYSPRINT またはそれに相当する DD ステートメントを含めます。

EXCI クライアント・プログラムを開始するためのサンプル・ジョブ

60 ページの図 8 は、クライアント・プログラムを開始するために使用または変更できるサンプル・ジョブを示します。

```
//EXCI JOB (accounting_information),CLASS=A,TIME=1440,
// USER=userid,PASSWORD=pswd,REGION=100M
//*****
//* JCL to execute an external CICS interface client program *
//*****
//
EXEC PGM=pgmname,REGION=nnM,MEMLIMIT=nnG
//STEPLIB DD DSN=
CICSTS56
.CICS.EXCI.LOADLIB,DISP=SHR
// DD DSN=
CICSTS56
.CICS.SDFHEXCI,DISP=SHR
//SYSPRINT DD SYSOUT=A
//SYSMDUMP DD DSN=SYS1.SYSMDP00,VOL=SER=volid,SPACE=(CYL,(1,1)),
DISP=OLD,UNIT=3390
```

図 8. EXCI クライアント・プログラムを開始するためのサンプル・ジョブ

注:

1. **USER** パラメーターに指定されているジョブ・ユーザー ID は、RACF またはそれに相当する外部セキュリティ・マネージャー (ESM) に定義する必要があります。
2. ジョブ・ユーザー ID は、ジョブ・ステップ開始セキュリティに使用されるのに加えて、MRO ログオンおよびバインド時セキュリティ検査にも使用されます。

外部 CICS インターフェース使用時のセキュリティについては、『[EXCI セキュリティ](#)』を参照してください。

3. RACF またはそれに相当する外部セキュリティ・マネージャー (ESM) がインストールされてアクティブにされていない環境でサンプル・アプリケーション・プログラムを実行する前に、サンプル接続定義を変更する方法については、『[EXCI サンプル・プログラムのセットアップ](#)』を参照してください。

EXCI 用の CICS 提供プロシージャ

CICS では、クライアント・プログラムの変換、コンパイル、リンク・エディットに使用できるプロシージャが 9 つ提供されています。

1 つのアセンブラ・プロシージャが提供されています。高水準言語には、以下にリストされているように、スタンドアロン変換プログラムを使用するバリエーションおよび統合変換プログラムを使用するバリエーションが提供されています。

DFHEXTAL

クライアント・プログラムのアセンブラ・バージョン用アセンブラ・プロシージャで、スタンドアロン変換プログラムを使用するもの。

DFHYXTDL

Language Environment® のもとで実行され、スタンドアロン変換プログラムを使用している C バージョンのクライアント・プログラム用のプロシージャ。

DFHYXTEL

Language Environment のもとで実行され、スタンドアロン変換プログラムを使用している C++ バージョンのクライアント・プログラム用のプロシージャ。

DFHYXTPL

Language Environment のもとで実行され、スタンドアロン変換プログラムを使用している PL/I バージョンのクライアント・プログラム用のプロシージャ。

DFHYXTVL

Language Environment のもとで実行され、スタンドアロン変換プログラムを使用している COBOL バージョンのクライアント・プログラム用のプロシージャ。

DFHZXTCL

Language Environment のもとで実行され、統合変換プログラムを使用している COBOL バージョンのクライアント・プログラム用のプロシージャ。

DFHZXTDL

Language Environment のもとで実行され、統合変換プログラムを使用している C バージョンのクライアント・プログラム用のプロシージャ。

DFHZXTEL

Language Environmentのもとで実行され、統合変換プログラムを使用している C++ バージョンのクライアント・プログラム用のプロシージャ。

DFHZXTPL

Language Environmentのもとで実行され、統合変換プログラムを使用している PL/I バージョンのクライアント・プログラム用のプロシージャ。

EXCI スタブを確実にクライアント・プログラムに組み込むために、これらのプロシージャすべてに COPYLINK ステップが組み込まれています。このステップは、リンケージ・エディターにとって適切なブロック長を使用して定義された一時データ・セットにスタブをアンロードします。その後、この一時データ・セットは、LKED ステップ内の SYSLIN DD ステートメントでオブジェクト・プログラムを含む一時データ・セットと連結されます。

これらのプロシージャは、CICSTS56.CICS.SDFHPROC ライブラリー内にあります。SYS1.PROCLIB か別の適切なプロシージャ・ライブラリーにそれらのプロシージャをコピーすることをお勧めします。

EXCI プログラミングの考慮事項

いくつかの言語要件が、外部 CICS インターフェースを使用する MVS クライアント・プログラムの作成に適用されます。これらの要件は、PL/I と C で作成されるプログラムに影響します。また、すべての言語で、MVS クライアント・プログラムを終了する前に戻りコードを処理する方法を考慮してください。

PL/I に関する考慮事項

CALL 上にある、外部 CICS インターフェースに書き込まれた PL/I プログラムのパラメーターを、アセンブラー・スタイルのパラメーター・リストの形式で DFHXCIS に提供しなければなりません。

PL/I 版の EXCI コピーブック DFHXCPLL には、必要な DFHXCIS エントリー・ポイントの定義が、以下のよう形で含まれています。

```
DCL DFHXCIS ENTRY OPTIONS(INTER ASSEMBLER);
```

EXCI LINK コマンドにも同じ規則が適用されますが、この場合は CICS 変換プログラムが、正しいパラメーター・リストが構築されていることを確認します。

PL/I で作成された EXCI クライアント・プログラムの例については、サンプル・プログラムのソース DFH\$PXCC を参照してください。

C に関する考慮事項

外部 CICS インターフェースを利用するように作成される C プログラムでは、DFHXCIS の CALL 実行時にパラメーターを、アセンブラー・スタイルのパラメーター・リストの形で提供する必要があります。そのため、OS LINKAGE を使用して、DFHXCIS に対するエン트리・ポイントを宣言します。

C 版の EXCI コピーブックである DFHXCPLH には、必要な DFHXCIS エントリー・ポイント定義が、以下のよう形で含まれています。

```
#pragma linkage(dfhxcis,OS)
```

EXCI LINK コマンドにも同じ規則が適用されますが、この場合は CICS 変換プログラムが、正しいパラメーター・リストが構築されていることを確認します。

C で作成された EXCI クライアント・プログラムの例については、サンプル・プログラムのソース DFH\$DXCC を参照してください。

終了時の戻りコード (R15) の設定

外部 CICS インターフェースは、クライアント・プログラムが正常に実行されるかどうかに関係なく、終了時にレジスター 15 をクリアしません。したがって、外部 CICS インターフェースを正常に使用した後に、MVS クライアント・プログラムが正常終了した場合でも、ジョブ・ステップが未定義の戻りコードで終了する可能性があります。

終了時に意味のある戻りコードが渡されるようにするには、プログラムを終了する前に、ジョブ・ステップの戻りコードを設定します。サンプル・クライアント・プログラムには、外部 CICS インターフェースに対する最後の呼び出しからの保管された応答コードを使用して、この設定を行う方法が示されています。

例えば、COBOL サンプル DFH0CXCC プログラムは、終了する前に SAVED-RESPONSE を特殊レジスター RETURN-CODE に移動します。

第 3 章 EXCI の構成

外部 CICS インターフェース (EXCI) は、MVS で実行される CICS 以外のプログラム (クライアント・プログラム) が、CICS 領域で実行されるプログラム (サーバー・プログラム) を呼び出し、通信域を使用したりチャネルとコンテナのセットを使用したりしてデータを送受信できるようにするアプリケーション・プログラミング・インターフェースです。このセクションでは、EXCI の構成手順について説明します。

EXCI の静的ルーティングのセットアップ

EXCI を使用するアプリケーションからの要求を CICS プログラムに静的に転送することができます。

始める前に

始めに、MVS パラメーター **Maxmember** が高い値に設定されていることを確認します。このパラメーターは、DFHIRP00 リソースに対して確立できる接続の数を制御します。

手順

1. RDO グループ EXCIXXXX を CICS 領域のグループリストに追加します。EXCIXXXX がない場合は、用意されている DFH\$EXCI RDO グループからコピーを作成します。

このグループは、EXCI 機能に必要なすべての接続を含むものであり、バッチ要求用に最大 100 の接続をサポートできます。

2. アプリケーションの RDO グループを CICS 領域のグループリストに追加します。
3. DFHXCOPT を SDFHEXCI ロード・ライブラリーにアセンブルします。DFHXCOPT に SURROGATE=YES が含まれていることを確認します。
4. アプリケーション・プログラムをアセンブルし、コンパイルします。
アセンブラで作成したアプリケーション・プログラムの場合は、リンケージ・エディター・パラメーター AMODE(31) および RMODE(ANY) を使用してください。そのプログラムをアプリケーション・ロード・ライブラリーにリンクします。
5. アプリケーション・プログラムを実行するためのバッチ JCL を構成します。

- a) JCL を編集して、バッチ・プログラムが EXCI 要求を送信する CICS 領域を指定します。

```
//step0010 EXEC PGM=program,PARM='applid,userid'
```

applid は CICS 領域であり、*userid* は RACF ユーザー ID です。

- b) ロード・ライブラリーを次のように連結します。

```
//STELIB DD Disp=shr,Dsn=SYS5C.CICn.CICS730.SDFHEXCI  
//DD Disp=shr,Dsn=Your.application.loadlib
```

6. バッチ・プログラムを実行し、予期した結果になることを確認します。

EXCI の動的ルーティングのセットアップ

CICSplex® SM を使用して、EXCI を使用するアプリケーションからの要求を CICS プログラムに動的に転送することができます。

このタスクについて

手順

1. CICS 領域で、次のシステム初期設定パラメーターを指定します。
 - DSRTPGM=EYU9XLOP
 - DTRPGM=EYU9XLOP
2. RDO グループを以下のように更新します。

- a) EXCI サーバー・プログラム DFHMIRS のための RDO エントリーを追加します。

必要に応じて、EXCI の代わりに別のトランザクションを使用することもできますが、そのトランザクションではプログラム DFHMIRS を指し、プロファイル DFHCICSA を使用する必要があります。グループ DFH\$EXCI の EXCI トランザクションの後にそれをモデル化します。このトランザクションは、DFHMIRS プログラムを指します。

注：リモートとして定義されたユーザー・トランザクションは、APPC 構成内でのみ動作します。EXCI または MRO 構成内では、REMOTE 属性を指定せずに DYNAMIC(NO) 属性を指定して、ミラー・トランザクションがローカルの CICS 領域で実行されるようにする必要があります。ミラー・トランザクションを別の CICS 領域にルーティングすると、パフォーマンスに影響が及んだり、問題判別が難しくなったりする可能性があります。

- b) RDO グループを端末専有領域 (TOR) LIST に追加します。

TOR 領域のプログラムの自動インストールが無効になっていることを確認します。

3. CICSplex SM にログインし、このトランザクションに、CPSM グループ (TXNGRP、WLMDEF、WLMGRP、WLMSPEC) へのルーティング領域を定義します。
4. アプリケーション所有領域 (AOR) ごとに、サンプル定義 DFH\$EXCI と同じ方法で RDO グループを作成します。

このグループには、接続、セッション、およびアプリケーション・プログラム、または同等のトランザクションのみが含まれている必要があります。EXCI の代わりに別のトランザクションを使用した場合は、そのトランザクションをこのグループに指定する必要があります。

タスクの結果

アプリケーションが分散プログラム・リンクを発行した場合、CICSplex SM は着信トランザクションがその制御下にあるかどうかを検査します。EXCI またはこれと同等のものがトランザクション・グループで有効になっていることを検出すると、CICSplex SM はそのトランザクションを候補 AOR の 1 つに転送して処理します。

この方法の代わりに、プログラムまたはトランザクションの RDO 定義に DYNAMIC=Yes を指定することもできます。CICSplex SM は選択した領域にその要求を転送します。

CICS への接続の定義

EXCI クライアント・プログラムと CICS 領域の間の接続には、CICS 領域での接続定義が必要です。それらを定義するには、CICS が備える CONNECTION および SESSIONS リソース定義機能を使用します。

外部 CICS インターフェースのために 特別に以下のオプションが用意されています。

- CONNECTION リソース定義の CONNTYPE
- CONNECTION および SESSIONS リソース定義の PROTOCOL 属性の EXCI

EXCI の CONNECTION リソース定義

CONNECTION リソース定義の PROTOCOL 属性には、外部 CICS インターフェースを使用する MVS プログラムのための接続であることを示す EXCI オプションが用意されています。

CONNECTION リソース定義には CONNTYPE 属性が用意されています。EXCI 接続では、これを使用して、接続が汎用接続と固有接続のどちらであることを示します。これは、外部 CICS インターフェース以外のプロトコルには使用されません。

EXCI に関連するパラメーターは以下のとおりです。

CONNTYPE({SPECIFIC|GENERIC})

外部 CICS インターフェース 接続の場合に、その接続の性質を表します。

SPECIFIC

この接続は非 CICS クライアント・プログラムから CICS 領域への通信用で、かつ、固有です。固有接続とは、クライアント・プログラムで単一のユーザーに占有される 1 つ以上のセッションが含まれる MRO リンクのことをいいます。

注: ユーザーとは、TCB ごとに固有の名前で Initialize_User 要求を発行したプログラム (または Initialize_User 要求を発行する対象になったプログラム) のことです。例:

- MVS の下で実行される単純なクライアント・プログラムが、外部 CICS インターフェースの単一ユーザーになることが可能です。
- MVS の下で実行される 1 つのクライアント・プログラムで複数のパイプを開き、それらを使用して、さまざまなベンダー・パッケージのために外部 CICS インターフェース呼び出しを順次発行することができます。その場合、クライアント・プログラムの観点から見ると、各パッケージが、固有のユーザー名で識別されるユーザーに相当します。つまり、1 つのクライアント・プログラムが複数のユーザーの代理として機能できます。
- MVS の下で実行される 1 つのプログラムが複数の TCB を接続し、各 TCB でベンダー・パッケージが自身のために外部 CICS インターフェース呼び出しを発行することができます。各パッケージは、独立したクライアント・プログラムであり、独自の TCB で実行されます。各パッケージは、固有のユーザー名を持つユーザーでもあります。

固有接続では、NETNAME が必須です。

GENERIC

接続は非 CICS クライアント・プログラムから CICS システムへの通信用で、汎用です。汎用接続とは、複数の EXCI ユーザーが共用する複数のセッションが含まれる MRO リンクのことをいいます。汎用接続では NETNAME 属性を指定できません。

注: 1 つの CICS 領域にインストールする汎用 EXCI 接続は 1 つのみにする必要があります。

NETNAME

外部 CICS インターフェース 接続の場合、NETNAME は、INITIALISE_USER 呼び出しの *user_name* パラメーターで指定された、固有のパイプのユーザーの名前に対応します。

外部 CICS インターフェースの固有パイプの場合、NETNAME を指定する必要があります。

外部 CICS インターフェースの汎用パイプの場合、NETNAME をブランクのままにする必要があります。

PROTOCOL({APPC|LU61|EXCI|blank})

リンクで使用するプロトコルのタイプ。

ブランク

CICS 領域間で MRO を使用する場合。MRO の場合は PROTOCOL をブランクのままにしておく必要があります。また、SESSIONS 定義では、PROTOCOL として LU6.1 を指定する必要があります。

APPC (LUTYPE6.2 プロトコル)

拡張プログラム間通信 (APPC) プロトコル。これは、ACCESSMETHOD(VTAM®) のデフォルト値です。CICS-CICS ISC にはこれを指定します。

注: VTAM は現在 z/OS Communications Server です。

LU61

LUTYPE6.1 プロトコル。これは、CICS-CICS ISC または CICS-IMS ISC について指定しますが、MRO には指定しません。

EXCI

外部 CICS インターフェース。これを指定して、この接続が外部 CICS インターフェースを使用する非 CICS クライアント・プログラム用であることを示します。

PROTOCOL(EXCI) を指定する場合は、ACCESSMETHOD(IRC) も指定しなければなりません。EXCI は、CICS 領域間通信プログラム DFHIRP を使用して MRO で実装されるため、MVS 仮想記憶間サービス (XM) を使用する MRO リンクは使用できません。EXCI は、同じく DFHIRP を介して動作する XCF MRO リンクも使用することができます。

EXCI 接続の SESSIONS リソース定義

SESSIONS リソース定義の PROTOCOL 属性で、MRO 接続に割り振られたセッションが外部の CICS インターフェースで使用されるかどうかを示します。

SESSIONS リソース定義の詳細については、[SESSIONS リソース](#)を参照してください。EXCI に関連するパラメーターは以下のとおりです。

PROTOCOL({APPC|LU61|EXCI})

相互通信リンク (ISC または MRO) で使用するプロトコルのタイプを 指定します。

APPC (LUTYPE6.2)

拡張プログラム間通信機能 (APPC) プロトコル。CICS-CICS ISC にはこれを指定します。

LU61

LUTYPE6.1 プロトコル。CICS-CICS ISC、CICS-IMS、または MRO の場合に 指定します。

EXCI

外部 CICS インターフェース。これを指定すると、セッションが外部 CICS インターフェースを使用する非 CICS クライアント・プログラムによって使用されることが示されます。EXCI を指定する場合、SEND COUNT はブランクのままにしなければなりません。

RECEIVECOUNT({blank|number})

通常、送信の前に受信する MRO、LUTYPE6.1、または EXCI セッションの数。

MRO の場合、受信セッションで受信できるのは送信前のみです。

blank

これらのセッションは送信専用です。受信セッションはありません。

number

CONNECTION 定義のプロトコル・パラメーターでブランク、LU61、または EXCI を指定した接続での受信セッションの数を指定します。CICS は、その数を使用してセッション名の最後の 2 文字または 3 文字を生成します (詳しくは、RECEIVEPFX を参照してください)。

デフォルトの受信接頭部 (<) または独自の 1 文字の接頭部を使用する場合は、1 から 999 までの範囲の数を指定してください。

2 文字の接頭部を指定する場合、数値の範囲は 1 から 99 までに制限されます。

外部 CICS インターフェース (EXCI) 接続を除き、このシステムの RECEIVECOUNT は、他のシステムの SEND COUNT と同じでなければなりません。

RECEIVEPFX(<|prefix)

CICS が受信セッション名 (そのセッションの端末管理テーブルの端末入力 (TCTTE) の名前) の最初の 1 文字または 2 文字に使用する、1 文字または 2 文字の接頭部を指定します。

既存の接続名や端末名と競合するような接頭部は 指定できません。

< (MRO および EXCI セッション)

MRO セッションでは、独自の受信接頭部を指定しないと、CICS はデフォルトの接頭部 (より小記号 (<)) を使用します。この記号に受信数を付加したものが、受信セッション名になります。

CICS は、セッション名の最後の 3 文字を、英数字 (A から Z、1 から 9) から作成します。この 3 文字の ID は文字 AAA から始まり、セッション項目数が RECEIVECOUNT 値で設定した限界に達するまで、昇順で続きます。受信セッション名が生成されるのは送信セッションの後であり、送信セッション名に続いて 同じ順序で生成されます。

例えば、送信セッションで最後に生成されるセッション名が、デフォルトの接頭部 (<) を使用した <AAJ だとすると、CICS は受信セッション名を <AAK、<AAL、<AAM のように生成します (セッション ID のこの生成方式は、最初の接頭部記号を除いては、APPC セッションでも同じです)。

注: 独自の接頭部を指定すると、CICS は以前のリリースの LUTYPE6.1 セッションの場合と同じ方法でセッション名を生成します。

prefix (LUTYPE6.1 セッション)

セッションが LUTYPE6.1 ISC 接続上にある場合は、1 文字または 2 文字の接頭部を指定する必要があります。デフォルトの < 記号は LUTYPE6.1 セッションには使用できません。

LUTYPE6.1 セッションでは (独自の 1 文字または 2 文字の接頭部を指定した場合は MRO でも)、CICS はその接頭部に番号 (1 から 99、または 1 から 999) を付加してセッション名を生成します。番号は 1 から始まり、指定した RECEIVECOUNT に達するまで 1 ずつ増えていきます。

SEND COUNT(blank|number)

通常、受信する前に送信する MRO または LUTYPE6.1 セッションの数。

MRO の場合、送信セッションを送信しないと受信できるようになりません。

blank

これらのセッションは受信専用です。送信セッションはありません。

セッションが外部の CICS インターフェース (EXCI) 接続に存在する場合は、このフィールドをブランクにしておく必要があります。

number

CONNECTION 定義のプロトコル・パラメーターでブランクまたは LU61 を指定した接続での送信セッションの数を指定します。CICS は、その数を使用してセッション名の最後の 2 文字または 3 文字を生成します (詳しくは、SENDPFX を参照してください)。

デフォルトの送信接頭部 (>) または独自の 1 文字の接頭部を使用する場合は、1 から 999 までの範囲の数を指定してください。

2 文字の接頭部を指定する場合、数値の範囲は 1 から 99 までに制限されます。

外部 CICS インターフェース (EXCI) 接続を除き、送信システムの SENDCOUNT は、受信システムの RECEIVEDCOUNT と同じでなければなりません。

SENDPFX(>|prefix)

CICS が送信セッション名 (そのセッションの端末管理テーブルの端末入力 (TCTTE) の名前) の最初の 1 文字または 2 文字に使用する、1 文字または 2 文字の接頭部を指定します。

既存の接続名や端末名と競合するような接頭部は指定できません。

> (MRO セッション)

MRO セッションでは、独自の送信接頭部を指定しないと、CICS はデフォルトの接頭部 (より大記号 (>)) を使用します。この記号に送信数を付加したものが、送信セッション名になります。

CICS は、セッション名の最後の 3 文字を、英数字 (A から Z、1 から 9) から作成します。この 3 文字の ID は文字 AAA から始まり、セッション項目数が SENDCOUNT 値で設定した限界に達するまで、昇順で続きます。

例えば、デフォルト接頭部 (>) を使用した場合、CICS が生成するセッション名は >AAA、>AAB、>AAC のようになります。(セッション ID のこの生成方式は、最初の記号を除いては、APPC セッションの場合と同じです。)

注: 独自の接頭部を指定すると、CICS は以前のリリースの LUTYPE6.1 セッションの場合と同じ方法でセッション名を生成します。

prefix (LUTYPE6.1 セッションの場合)

セッションが LUTYPE6.1 ISC 接続上にある場合は、1 文字または 2 文字の接頭部を指定する必要があります。デフォルトの > 記号は LUTYPE6.1 セッションには使用できません。

LUTYPE6.1 セッションでは (独自の 1 文字または 2 文字の接頭部を指定した場合は MRO でも)、CICS はその接頭部に番号 (1 から 99、または 1 から 999) を付加してセッション名を生成します。番号は 1 から始まり、指定した SENDCOUNT に達するまで 1 ずつ増えていきます。

USERID(userid)

リンクのセキュリティ検査に使用する事前設定のユーザー ID。

リンク・セキュリティの事前設定ユーザー ID を指定しない場合、CICS は、リモート・ユーザーから渡されたユーザー ID をリンク・セキュリティのユーザー ID として使用します。外部 CICS インターフェース・リンクの場合、これはクライアントのユーザー ID です。

EXCI 接続の状態の照会

CICS 端末から CICS サーバー領域にアクセスできる場合は、クライアント・アプリケーション・プログラムを実行しているバッチ・ジョブについて照会し、CICS 内のサーバー・プログラムにリンクするために外部 CICS インターフェースを使用しているのがどのバッチ・ジョブかを照会できます。

このタスクについて

MRO を介して CICS にリンクされているバッチ・ジョブに関するこの情報を取得するには、CEMT INQUIRE EXCI コマンドを使用します。このコマンドを使用すると、領域間通信 (IRC) 機能を介して現在 CICS に接続されている外部 CICS インターフェース・バッチ・ジョブの名前を調べることができます。

CICS は、ジョブ ID を次の形式で返します。

```
jobname.stepname.procname - mvsid
```

stepname、または procname、あるいはその両方が存在しない (ピリオド (.) が連続していることでわかります) 場合があります。

mvsid は、ジョブが実行されている MVS システムを指します。XCF/MRO を使用する場合、ジョブは CICS が実行されているイメージとは別の MVS イメージに存在できます。

外部 CICS インターフェースを使用するジョブに関する情報は、ジョブが DPL 要求を 1 つ以上実行した場合にのみ取得可能です。ゼロ以外のタスク番号は、DPL 要求が現在アクティブであることを示しています。ゼロのタスク番号は、現在アクティブな DPL 要求が存在しないにもかかわらず、外部 CICS インターフェース・セッションがまだそのジョブに対してオープンされている (接続されている) ことを示しています。

CEMT コマンドについて詳しくは、[CEMT - マスター端末](#)を参照してください。

EXCI ユーザー置換可能モジュール

外部 CICS インターフェースには、ユーザー置換可能モジュール DFHXCURM が用意されています。

ロード・モジュールは CICSTS56.CICS.SDFHEXCI 内にあり、ソースは CICSTS56.CICS.SDFHSAMP 内にあります。ユーザー置換可能プログラムのアセンブルおよびリンク・エディットについては、[ユーザー置換可能プログラムのアセンブルとリンク・エディット](#)を参照してください。

DFHXCURM は、**allocate_pipe** コマンドの処理中と、再試行可能エラーが発生した後に、非 CICS 領域で外部 CICS インターフェースによって開始されます。

再試行可能応答は以下のとおりです。

- ターゲット CICS 領域が使用不可である
- ターゲット CICS 領域に使用可能なパイプがない
- MVS IPL 以降に IRC アクティビティーがなかった

再試行可能エラー発生後に再試行するには、EXCI 呼び出しを再発行します。

サンプル・ロジックをバイパスし、外部 CICS インターフェース呼び出し元に制御を返すブランチ命令があるため、提供されたままの DFHXCURM は実際のところダミー・プログラムです。サンプル・ロジックを使用するには、ブランチ命令を削除し、モジュールをアセンブルしてリンク・エディットしてください。以下のアクションを実行するように DFHXCURM をカスタマイズすることができます。

- **allocate_pipe** の処理中に、指定された CICS APPLID を変更して、要求を別の CICS システムに送ることができます。
- **allocate_pipe** 処理中に、要求を別の XCF グループへ送ることができます。
- 再試行可能なエラーの後に DFHXCURM が開始される場合、CICS の可用性に関する情報を保管できます。その後、**allocate_pipe** 処理のために DFHXCURM を次に呼び出す時に、どの CICS システムに要求を送付するかを決定するのにこの情報を使用できます。

DFHXCURM は、標準の MVS レジスター規則 (レジスター 1 にパラメーター・リストのアドレス、レジスター 14 に呼び出し元の戻りアドレスを格納する) を使用して呼び出されます。レジスター 1 でアドレス指定されたパラメーターは、DFHXCPLD コピーブック内にある EXCI_URM_PARMS DSECT でマップされます。DFHXCURM に渡されるパラメーターは、以下のとおりです。

URMINV

DFHXCURM の呼び出しの理由が格納されるフルワードのアドレス。以下のように定義されます。

URM_ALLOCATE	EQU 1	This invocation is for an Allocate_Pipe
URM_NO_CICS	EQU 2	The target CICS region is not available

```
URM_NO_PIPE      EQU 3  There are no pipes available
URM_NO_CICS_IRC   EQU 4  There has been no IRC activity since the MVS IPL
```

URMCICS

ターゲット CICS システムの APPLID を格納する 8 バイト領域のアドレス。この ID は、**Allocate_Pipe** コマンドの **CICS_applid** パラメーター、または **EXEC CICS LINK** コマンドの **APPLID** パラメーターで指定されます。

これらのコマンドの 1 つで指定された場合、APPLID は別のターゲット CICS 領域のものに変更できます。また、これらのコマンドの 1 つで指定された APPLID が有効な固有のアプリ ID でない場合は、その APPLID を有効な固有のアプリ ID に変更する必要があります。

allocate_pipe 要求で **CICS_applid** パラメーターが省略された場合、または **EXEC CICS LINK** コマンドで APPLID が省略された場合、このパラメーターでアドレス指定されるフィールドには 8 個のリンクが格納されます。この場合は、呼び出し元に制御を返す前に、DFHXCURM に APPLID を指定する必要があります。

URMAPPL

Initialize_User コマンドの *my_name* パラメーターで指定されたクライアント・プログラムのユーザー名を格納する 8 バイト領域のアドレス。EXEC CICS LINK コマンドで DFHXCURM が開始される場合、この名前は必ず DFHXCEIP に設定されます。

URMPROG

ターゲット・プログラムが使用可能な場合に、その名前を格納する 8 バイト領域のアドレス。この名前が使用可能なのは、DFHXCURM が EXEC CICS LINK コマンドで開始された場合のみです。外部 CICS インターフェースの allocate_pipe コマンドの場合は、DPL 呼び出しが発行されるまで、プログラム名は分かりません。

URMOPTS

allocate_opts パラメーターに指定されている割り振りオプション (X'00' または X'80') が格納される 1 バイト領域のアドレス。このアドレスが有効なのは、Allocate_Pipe 要求の場合のみです。

URMANCH

DFHXCURM 専用として提供される 4 バイト領域のアドレス。これは一般的に、複数回の呼び出しに渡って DFHXCURM の情報を保管するために使用される領域のグローバル・アンカー・アドレスを保管する目的で使用されます。例えば、必要なストレージを GETMAIN で取得し、このパラメーターでアドレス指定される 4 バイト領域にそのアドレスを保管できます。4 バイト領域の初期値はゼロに設定されます。

EXCI を使用するアドレス・スペースでは、TCB ごとに URMANCH パラメーターが 1 つずつあります。

URMXCFG

DFHXCOPT テーブルの **XCFGROUP** パラメーターに指定された XCF グループ名を格納する 8 バイト領域のアドレス。EXCI Allocate_Pipe コマンドの処理中に DFHXCURM が呼び出されたときには、このパラメーターを使用して XCF グループ名を変更します。DFHXCURM の呼び出しに失敗したが、再試行できる場合は、パイプを以前割り振ったときに使用された値がこの領域に格納されます。値の変更による影響はありません。

グループ名の長さは 8 文字でなければならず、必要に応じて右側にブランクが埋め込まれます。有効な文字は A-Z 0-9 \$ # @ です。文字 A、B、C、E、F、G、H、I、"SYS" でグループ名を始めないください。これらの名前は、IBM がその XCF グループに使用しています。また、「UNDESIG」という名前も使用しないでください。この名前は、インストール済み環境でのシステム・プログラマーによる使用のために予約されています。

文字 "DFHIR" で始まるグループ名を使用することをお勧めします。

EXCI オプション・テーブル DFHXCOPT の使用

DFHXCOPT マクロによって生成される EXCI オプション・テーブルを使用すると、外部 CICS インターフェースに必要な複数のパラメーターを指定できます。

DFHXCOPT オプション・テーブル: 新しい形式と古い形式

DFHXCOPT オプション・テーブルは、最初に導入されて以降変更が加えられ、将来の拡張に備えて、より柔軟性を持たせるため、テーブルにバージョン番号が含まれるようになりました。例えば、CICS の以前のリリースから、カスタマイズ済みの DFHXCOPT テーブルをマイグレーションすることを計画している場合、この変更に注意する必要があります。

新旧の形式を区別するために、新しい形式のテーブルは、DFHXCOPE という別名によってリンク・エディットされています。以下の手順を使用してオプション・テーブルがロードされます。

1. CICS は、DFHXCOPE という別名を使用して、DFHXCOPT テーブルをロードしようとします。CICS は、DFHXCOPE という名前のロード・モジュールを検出し、そのロードに成功した場合、そのテーブルを新しい形式のものとして想定します。
2. CICS が DFHXCOPE という名前のロード・モジュールを検出しない(または、検出されたものの、ロードに失敗した) 場合、DFHXCOPT というベース名を使用してテーブルのロードを試行します。この場合、CICS はテーブルが古い形式であると想定します。

CICS にはデフォルトの DFHXCOPT テーブルが用意されていて、そのデフォルト・テーブルのソース・コードも CICSTS56.CICS.SDFHSAMP ライブラリーにあります。デフォルトの DFHXCOPT テーブルのロード・モジュールと、その別名 DFHXCOPE は、CICSTS56.CICS.SDFHEXCI ライブラリーに用意されています。

カスタマイズされた DFHXCOPT テーブルの作成

CICS 提供のデフォルトの DFHXCOPT テーブルのソース・コードを、ユーザーの要件に合わせて調整できます。

カスタマイズした DFHXCOPT テーブルをアセンブルし、MVS クライアント・プログラムを実行するジョブの STEPLIB 連結内の適切なライブラリーにリンク・エディットする必要があります。

重要: 独自にカスタマイズした DFHXCOPT テーブルを作成する場合は、必ず DFHXCOPE 別名を使用してリンク・エディットするようにしてください。 標準 DFHAUPLE プロシージャを使用して、そのようになっていることを確認します。別名を使用せずにテーブルを再アセンブルし、リンク・エディットすると、CICS は、カスタマイズされたテーブルではなく、デフォルトのテーブル (DFHXCOPE 別名を使用して検出) をロードします。

このために、独自のバージョンの CICS DFHAUPLE プロシージャを使用することができます。DFHAUPLE プロシージャは、CICSTS56.CICS.SDFHINST に用意されています。

DFHXCOPT マクロ: 形式およびパラメーター

CICS 領域に指定するテーブルとは違って、DFHXCOPT テーブルには接尾部を付けることはできません。

次の表に、DFHXCOPT マクロとそのパラメーターの形式を示します。

DFHXCO	TYPE={ CSECT DSECT} [,ABENDBKOUT={ NO YES}] [,CICSSVC={ 216 number}] [,CONFDATA={ HIDE SHOW}] [,DURETRY={ 30 number-of-seconds}] [,GTF={ OFF ON}] [,LOCALCCSID={ 037 CCSID}] [,MSGCASE={ MIXED UPPER}]
--------	---

	除去 [,SURROGCHK={ YES NO}] [,TIMEOUT={ 0 number}] [,TRACE={ OFF 1 2 3}] [,TRACESIZE={ 16 number-of-kilobytes}] [,TRAP={ OFF ON}] [,XCFGROUP={ DFHIR000 name}] 次の END ステートメントを使用してパラメーターを終了する必要があります。 DFHXCOPT
END	

TYPE={CSECT|DSECT}

生成されるテーブルのタイプを示します。

CSECT

通常使用される正規の制御セクション。

DSECT

ダミー制御セクション。

ABENDBKOUT={NO|YES}

タスクが CICS サーバー内で異常終了した場合にグローバル作業単位の自動ロールバックを起動するかどうかを指定します。グローバル作業単位は、EXCI クライアント・プログラムが MVS RRS を通じてリソース・リカバリーを制御している場合 (つまり、DPL 要求上で SYNCONRETURN が指定されていない場合) に存在します。この場合、CICS サーバー・プログラムが異常終了する場合にロールバックするよう、グローバル作業単位にマークを付けると良いかもしれません。

注：DPL 要求上で SYNCONRETURN が指定された場合、ABENDBKOUT は無効です。

NO

グローバル作業単位はロールバックされるようにマークを付けられません。

YES

サーバー・プログラムの異常終了を処理しているとき、CICS ミラー・プログラムは、バックアウトされるようにグローバル作業単位にマークを付けます。

どちらの場合でも、EXCI クライアント・プログラムは、EXCI DPL 要求上で、戻りコード 422、SERVER_ABENDED を受け取ります。

CICSSVC={216|number}

MRO 通信用に使用されている CICS タイプ 3 SVC の番号を指定します。デフォルトは 216 です。

外部 CICS インターフェースは、クライアント・プログラムが稼働中の MVS イメージ内にある CICS MRO 領域で使用されているのと同じ SVC 番号を使用する必要があります。

0

外部 CICS インターフェースが、by means of an MVS VERIFY コマンドを使用して MVS から CICS SVC 番号を取得することを示すには、0 を指定します。

0 を指定するのは、少なくとも 1 つの CICS 領域が、MVS IPL の存続期間中に DFHIRP にログオンしたことが確かな場合だけにしてください。

number

CICS 領域間通信用に使用中の CICS SVC 番号を 200 から 255 の範囲で指定します。これは、クライアント・プログラムが稼働中の MVS イメージ (ローカル MVS) にインストールされた SVC 番号である必要があります。

IPL の存続時間中にローカル MVS の DFHIRP にログオンした MRO CICS 領域がない場合、ゼロ以外の SVC 番号を指定する必要があります。ゼロを指定して、外部 CICS インターフェース 要求が MVS から SVC を要求した場合、DFHIRP にログオンした CICS 領域がなければ、この要求は失敗します。

CICS 領域をまったく実行しておらず、クライアント・プログラムが、別の MVS に常駐するサーバー CICS 領域に対して DPL 要求を発行する MVS イメージでは、ゼロ以外の値が必要です。このような環境では、クライアント・プログラムは、ローカルに定義された SVC を使用してローカル DFHIRP にログオンし、XCF/MRO を使用してリモート CICS 領域と通信します。

注：同一の MVS イメージ内で MRO を使用するすべての CICS 領域は、DFHIRP と CICS SVC、DFHCSVC の両方とも最高レベルのものを使用する必要があります。MRO CICSplex がさまざまなリリース・レベルの CICS 領域で構成されている場合、LPA にインストールされた DFHIRP と DFHCSVC は、CICSplex 内の最新のリリース・レベルの CICS からのものである必要があります。

CONFDATA={HIDE|SHOW}

このパラメーターは、外部 CICS インターフェースが、GTF への EXCI トレース項目出力や EXCI ダンプに通常なら表示されるユーザー・データを抑止する (非表示にする) かどうかを指示するためにコーディングします。このオプションは、EXCI クライアント・プログラムと CICS サーバー・プログラムとの間でフローする COMMAREA データまたは CONTAINER データのトレースに適用されます。

HIDE

EXCI はトレース項目からユーザーの COMMAREA データまたは CONTAINER データを「非表示」にします。代わりに、トレース項目には、データが抑止されたことを示す文字ストリングが設定されます。

SHOW

データ抑止は有効ではありません。ユーザー・データはトレースされます。

DURETRY={30|number-of-seconds|0}

外部 CICS インターフェースが、SDUMP マクロを使用して MVS システム・ダンプを取得しようと試行を続ける合計時間を秒単位で指定します。

DURETRY は、外部 CICS インターフェースが SDUMP 要求を出す際に、同じ MVS システムの別のアドレス・スペースが既に SDUMP を取っている場合に、外部 CICS インターフェースが SDUMP を再発行するかどうか、および再発行する場合はその期間の長さを制御できます。

SDUMP が失敗した場合は、外部 CICS インターフェースは、次のように対応します。

- MVS が既に別のアドレス・スペースの SDUMP を取っており、DURETRY パラメーターがゼロでない場合、外部 CICS インターフェースは、SDUMP マクロを再試行する前に MVS STIMERM マクロを発行して、5 秒間待機します。外部 CICS インターフェースは、DURETRY 制限時間まで 5 秒ごとに SDUMP を再試行することを示すメッセージを発行します。
- SDUMP が次のような理由で失敗した場合、外部 CICS インターフェースは、SDUMP が失敗したことを通知するメッセージを理由とともに出力します。
 - 使用可能な SYS1.DUMP データ・セットがない。
 - ダンプの完了を妨げる入出力エラーがある。
 - SDUMP の再試行中に DURETRY の制限時間が満了した。

30

30 秒を指定すると、外部 CICS インターフェースは、最大 6 回まで (5 秒ごとに 1 回) 再試行できます。

number-of-seconds

外部 CICS インターフェースに SDUMP マクロの再試行を継続させる秒数の合計 (最大 32767 秒) をコーディングします。外部 CICS インターフェースは、成功するか、DURETRY 値以上の期間が経過するまで、5 秒ごとに 1 回 SDUMP を再試行します。

0

CICS に SDUMP を再試行させたくない場合には、ゼロの値をコーディングします。

GTF={OFF|ON}

外部 CICS インターフェース・トレース・テーブルに通常書き込まれるすべてのトレース項目が、(GTF トレースがアクティブの場合に) MVS 汎用トレース機能 (GTF) データ・セットにも書き込まれるかどうかを指定します。

OFF

トレース項目を GTF に書き込まない場合にコーディングします。

ON

トレース項目を GTF に書き込む場合にコーディングします。

LOCALCCSID={037|CCSID}

EXCI ジョブのデフォルト CCSID を指定します。CCSID は 8 文字までの値です。CCSID 値を指定しない場合は、デフォルトの LOCALCCSID が 037 に設定されます。有効な CCSID のリストについて詳しくは、以下の情報を参照してください。

- [CICS がサポートされている変換](#)
- [z/OS Unicode Services ユーザーズ・ガイドおよび解説書の関連付録](#)

037

LOCALCCSID のデフォルト値。

CCSID

その他の任意の有効な EBCDIC CCSID 値を表します。

MSGCASE={MIXED|UPPER}

DFHEXxxxx メッセージを大/小文字混合または大文字のどちらで 発行するかを指定します。

MIXED

メッセージを大/小文字混合で出力する場合にコーディングします。

UPPER

メッセージを大文字で出力する場合にコーディングします。

SURROGCHK={YES|NO}

除去: SURROGCHK パラメーターが除去されました。代理検査は常に行われます。SURROGCHK=NO というオプションを希望する場合は、IBM サポートに usermod を要求する必要があります。

クライアント・ジョブ・ユーザー ID (EXCI ジョブを実行中のユーザー ID) に対して、外部 CICS インターフェースが、代理ユーザー検査を実行するかどうかを指定します。

YES

外部 CICS インターフェースは、代理ユーザー検査を実行して、EXCI クライアント・ジョブを実行中のユーザー ID が、DPL 呼び出しで指定したユーザー ID に対して、代理として認可されているかどうかを検証します。この検査は、クライアント・ユーザー ID が、DPL 呼び出しで指定したユーザー ID と違う場合のみ実行されます。

クライアント・ユーザー ID が、SURROGAT 一般リソース・クラスの適切なプロファイルに対して認可されている必要があります。そのためには、クライアント・ユーザー ID に、SURROGAT 一般リソース・クラスの userid.DFHEXCI という名前のプロファイルに対する READ 権限を与えます。userid は、DPL 呼び出し上で指定されたユーザー ID です。

バッチ領域のユーザー ID を DPL ユーザー ID の代理ユーザーとして認可する方法の例については、[代理ユーザー検査](#)を参照してください。

NO

代理ユーザー検査は実行されません。

TIMEOUT={0|number}

外部 CICS インターフェースが、DPL コマンドの完了を待機する時間間隔を 100 分の 1 秒単位で指定します。

DPL コマンドは、1 つのチャンネルとコンテナ・セットを CICS に渡すことができます。この情報を渡すコマンドには、コンテナ・データを構成するために、CICS までの複数のデータ・フローを含めることができます。タイムアウト間隔は、データの最後のフローが CICS に送信された時に開始します。このときにデータは完了し、CICS サーバー・プログラムの呼び出しが開始します。

0

時間制限を適用せず、外部 CICS インターフェースが、DPL コマンドの完了を無期限に待機することを指定します。

number

外部 CICS インターフェースが、DPL コマンドの完了を待機する時間間隔を 100 分の 1 秒単位で指定します。number は、100 分の 1 秒単位の時間を 1 から最大 2 147 483 647 までの数で表現します。以下に例を示します。

6000

1 分のタイムアウト値を表します。

30000

5 分のタイムアウト値を表します。

60000

10 分のタイムアウト値を表します。

TRACE={OFF|1|2|3}

外部 CICS インターフェースの内部トレースを行うかどうか、およびどのレベルにするかを指定します。

OFF

外部 CICS インターフェースの内部トレースは不要です。しかし、通常のトレースをオフに切り替えても、例外トレース項目は常に CICS 領域内の外部 CICS インターフェース・トレース・テーブルに書き込まれます。

1

例外、およびレベル 1 のトレース項目が外部 CICS インターフェース・トレース・テーブルに書き込まれます。

2

例外、レベル 1、およびレベル 2 のトレース項目が外部 CICS インターフェース・トレース・テーブルに書き込まれます。

3

例外、レベル 1、レベル 2、レベル 3 のトレース項目が外部 CICS インターフェース・トレース・テーブルに書き込まれます。

TRACESIZE={16|number-of-kilobytes}

外部 CICS インターフェースによって使用されるトレース・テーブルのサイズをキロバイト単位で指定します。このトレース・テーブルは、CICS 領域内の 31 ビットのストレージ (16 MB 境界より上) に割り振られます。

16

トレース・テーブルのデフォルト・サイズであり、同時に最小サイズでもあります。

number-of-kilobytes

トレース・テーブル用に割り振られるストレージの、16 KB から 1 048 576 KB までの範囲のキロバイト数。トレース・テーブル・ストレージ用にサブプール 1 が使用されます。これは、ジョブ・ステップ TCB の期間だけ存在します。テーブルは、ページ単位で位置合わせされ、整数のページを占めます。指定された値がページ・サイズ (4 KB) の倍数でない場合は、4 KB の次の倍数に切り上げられます。

TRAP={OFF|ON}

サービス・トラップ・モジュール DFHXCTRA を使用するかどうかを指定します。DFHXCTRA は、ユーザー置き換え可能モジュールとして提供されます。ここには、IBM サービス技術員が、エラーをトラップするコードを追加できます。

OFF

DFHXCTRA を使用しない場合にコーディングします。

ON

DFHXCTRA が必要な場合にコーディングします。

XCFGROUP={DFHIR000|name}

システム間カップリング・ファシリティ (XCF) グループの名前をこのクライアント・プログラムによって結合するように指定します。

注: XCF グループを使用すれば、同じシスプレックス内の異なる MVS イメージの CICS 領域が、複数領域操作 (MRO) 接続を通じて相互に通信できます。XCF/MRO の概要と、XCF グループのセットアップ方法については、[システム間複数領域操作 \(XCF/MRO\)](#)を参照してください。

各クライアント・プログラムは、最大で 1 つの XCF グループを結合できます。

DFHIR000

デフォルトの XCF グループ名。

name

グループ名の長さは 8 文字でなければならず、必要に応じて右側にブランクが埋め込まれます。有効な文字は、A-Z、0-9、および国別文字、\$、# および @ です。IBM が自社の XCF グループ用に使用する名前の使用を回避するために、A から C、E から I の文字、または文字ストリング "SYS" で始まるグループ名を使用しないでください。また、「UNDESIG」という名前も使用しないでください。この名前は、インストール済み環境でシステム・プログラマーが使用するために予約されています。

「DFHIR」という文字で始まるグループ名を使用することをお勧めします。

第 4 章 EXCI のセキュリティ

CICS は、MVS クライアント・プログラムから受け取った要求に対して、複数の方法でセキュリティ検査を適用します。

MRO ログオン・セキュリティおよびバインド時のセキュリティの使用

CICS 領域間通信プログラムである DFHIRP は、IRP にログオンしようとする (特定の接続のみ) ユーザーまたは CICS 領域に接続しようとする (バインド時のセキュリティとも呼ばれる) ユーザーに対して 2 つのセキュリティ検査を実行します。

このタスクについて

汎用 EXCI 接続: このセクションのログオン・セキュリティ検査に関する説明は、SPECIFIC として定義された EXCI 接続にのみ適用されます。MRO ログオン・セキュリティ検査は、汎用接続に対して実行されません。

MRO ログオン・セキュリティ検査および接続 (バインド時の) セキュリティ検査に関する限り、MVS クライアント・プログラムは別の CICS 領域とまったく同じように処理されます。これは、クライアント・プログラムが領域間通信プログラムにログオンすると、IRP はクライアント・プログラムの実行ユーザー ID に対してログオン・セキュリティ検査とバインド時のセキュリティ検査を実行することを意味します。このセクションの残りの部分では、このユーザー ID はバッチ領域のユーザー ID と呼ばれます。

クライアント・プログラムが IRP に正常にログオンし、ターゲット・サーバー領域に接続できるようにするには、最初にユーザー・プロファイルのバッチ領域のユーザー ID を RACF に対して定義します。バッチ領域のユーザー ID を RACF に対して定義すると、バッチ領域に適切なログオン許可とバインド時の許可を与えることができます。

手順

- ログオン許可

バッチ領域のユーザー ID を DFHAPPL.user_name RACF FACILITY クラス・プロファイルに UPDATE 権限付きで許可します。プロファイル名の user_name の部分は、INITIALIZE_USER コマンドで定義されたユーザー名です。

バッチ領域のユーザー ID を、IRP にログオンしている特定のユーザー ID の DFHAPPL プロファイルに許可できないと、Allocate_Pipe 処理は失敗し、RESPONSE(SYSTEM_ERROR) REASON(IRC_LOGON_FAILURE) が出されます。ログオン・セキュリティ検査が失敗すると、subreason フィールド 1 は 10 進数 204 を返します。

EXCI クライアント・プログラムの FACILITY クラス・プロファイルについては、[77 ページの『EXCI 領域の DFHAPPL FACILITY クラス・プロファイルの定義』](#)を参照してください。

- バインド時の許可

バッチ領域のユーザー ID をターゲット CICS サーバー領域の DFHAPPL.applid RACF FACILITY クラス・プロファイルに READ 権限付きで許可します。

バッチ領域のユーザー ID を CICS サーバー領域の DFHAPPL.applid プロファイルに許可できないと、Open_Pipe 処理は失敗し、RESPONSE(SYSTEM_ERROR) REASON(IRC_CONNECT_FAILURE) が出されます。バインド時のセキュリティ検査が失敗すると、subreason フィールド 1 は 10 進数 176 を返します。

MRO ログオン・セキュリティ検査およびバインド時のセキュリティ検査と、RACF DFHAPPL プロファイルの定義方法の例については、[バインド・セキュリティ](#)を参照してください。

EXCI 領域の DFHAPPL FACILITY クラス・プロファイルの定義

DFHAPPL プロファイル名の user_name の部分を定義するには、次のようにします。

- EXCI CALL インターフェースの場合、*user_name* は、INITIALIZE_USER コマンドの *user_name* パラメーターに指定する名前であればなりません。

クライアント・プログラムに複数のユーザー名に対する INITIALIZE_USER コマンドがある場合、そのプログラムで指定されたユーザー名ごとに、適切な許可を持つ FACILITY クラス・プロファイルを定義します。

例えば、INITIALIZE_USER コマンドで定義されている *user_name* が DCEUSER1 の場合、次のように FACILITY クラスで DFHAPPL プロファイルを定義します。

```
RDEFINE FACILITY (DFHAPPL.DCEUSER1) UACC(NONE)
```

バッチ領域のユーザー ID が CLIENTA の場合、次のようにしてバッチ領域が IRP にログオンするのを許可します。

```
PERMIT DFHAPPL.DCEUSER1 CLASS(FACILITY) ID(CLIENTA)
ACCESS(UPDATE)
```

- EXEC CICS LINK コマンドの場合、*user_name* は外部 CICS インターフェースによって DFHXCEIP として事前設定されています。これには IRP ログオンの許可は不要です。EXEC CICS LINK インターフェースはログオン・セキュリティチェックが適用されない汎用接続を使用するからです。

リンク・セキュリティ

ターゲット CICS サーバー領域は、クライアント・プログラムからの要求に対してリンク・セキュリティチェックを実行します。

これらのセキュリティチェックは、トランザクション接続セキュリティ (ミラー・トランザクションを接続している場合)、およびサーバー・アプリケーション・プログラム内のリソースおよびコマンドのセキュリティチェックを対象としています。CICS がこれらのセキュリティチェックに使用するリンク・ユーザー ID は、バッチ領域のユーザー ID です。

これらのリンク・セキュリティチェックが原因でセキュリティ障害が発生することがないようにするには、必要に応じて、リンク・ユーザー ID が以下のリソース・プロファイルに対して許可されていることを確認する必要があります。

- ミラー・トランザクションのプロファイル。CSMI (デフォルトの場合) または *transid* パラメーターで指定されたミラー・トランザクションのいずれか。これはトランザクション接続セキュリティチェックでは必須です。
- CICS サーバー・アプリケーション・プログラムがアクセスするすべてのリソースのプロファイル。ファイル、キュー (一時データ・キューおよび一時ストレージ・キュー)、プログラムなど。これはリソース・セキュリティチェックでは必須です。
- CICS サーバー・アプリケーション・プログラムによって発行される SPI コマンドの CICS コマンド・プロファイル。INQUIRE、SET、DISCARD など。これはコマンド・セキュリティチェックでは必須です。

MRO リンク・セキュリティチェックについては、[バインド・セキュリティ](#)を参照してください。

ユーザー・セキュリティ

ターゲット CICS サーバー領域は、DPL_Request 呼び出しで渡されるユーザー ID に対してユーザー・セキュリティチェックを実行します。ユーザー・セキュリティチェックは、接続で ATTACHSEC(IDENTIFY) を指定した場合のみ実行されます。

ユーザー・セキュリティは、リンク・セキュリティに加えて実行されます。

ユーザー・セキュリティの場合、リンク・セキュリティに対して行う許可に加えて、DPL_Request 呼び出しで指定されたユーザー ID も許可する必要があります。

EXEC CICS LINK コマンドでユーザー ID を指定する方法はないことに注意してください。この場合、外部 CICS インターフェースがバッチ領域のユーザー ID を渡します。したがって、接続定義で ATTACHSEC(IDENTIFY) が指定されている場合、ユーザー・セキュリティチェックは、バッチ領域のユーザー ID に対して実行されます。

注：外部 CICS インターフェースの 接続リソース定義で ATTACHSEC(IDENTIFY) が指定される場合、RACF または同等の外部セキュリティ・マネージャー (ESM) がインストールされておらずアクティブになっていない環境でサーバー・プログラムを実行すると、サーバー・プログラムは失敗し、ATCY 異常終了が出されます。

セキュリティがアクティブになっていない外部 CICS インターフェースのサーバー・プログラムを実行する場合、ATTACHSEC(LOCAL) を指定する必要があります。

代理ユーザー検査

代理ユーザー検査は、バッチ領域のユーザー ID が別のユーザーの DPL 呼び出しを発行することを許可されている (つまり、DPL_Request 呼び出しで指定されたユーザー ID の代理として許可されている) ことを確認します。

SURROGCHK=YES (デフォルト) が EXCI オプション・テーブル DFHXCOPT で指定されている場合、EXCI クライアント・ジョブは代理ユーザー検査の対象となります。SURROGCHK=YES を指定する場合 (またはデフォルトにすることを許可する場合)、バッチ領域のユーザー ID を、すべての DPL_Request 呼び出しで指定されたユーザー ID の代理として許可します。これは、バッチ領域のユーザー ID には、SURROGAT 一般リソース・クラスの *userid.DFHEXCI* というプロファイルに対する READ 権限が必要であることを意味します (ここで、*userid* は DPL 呼び出しで指定されたユーザー ID です)。例えば、次のコマンドは DPL ユーザー ID の代理プロファイルを定義し、EXCI バッチ領域に READ 権限を付与します。

```
RDEFINE  SURROGAT dpl_userid.DFHEXCI UACC(NONE) OWNER(DPL_userid)
PERMIT   userid.DFHEXCI CLASS(SURROGAT) ID(batch_region_userid)
          ACCESS(READ)
```

代理ユーザー検査が有効である (SURROGCHK=YES) が、ユーザー ID が DPL_Request 呼び出しで指定されていない場合、代理ユーザー検査は実行されません。DPL_Request 呼び出しのユーザー ID はデフォルトでバッチ領域のユーザー ID になるからです。この代理ユーザー検査のバイパスを正常に実行するには、DPL_Request 呼び出しでユーザー ID を正しく省略したことを確認します。EXCI 呼び出しパラメーターを省略するときに NULL ポインターを正しく指定する方法については、[EXCI CALL インターフェース](#) でヌル・パラメーターを持つ EXCI CALL の例を参照してください。

代理ユーザー・セキュリティ検査を行わない場合は、DFHXCOPT オプション・テーブルで SURROGCHK=NO を指定します (SURROGCHK=YES がデフォルトであることに注意してください)。

代理ユーザー検査は、バッチ領域のユーザー ID が CICS サーバー領域ユーザー ID と同じ場合に有効です。この場合、リンク・セキュリティ検査 (78 ページの『リンク・セキュリティ』を参照) がバイパスされるからです。この場合、DPL_Request 呼び出しで指定されるユーザー ID は認証されたユーザー ID ではないため (パスワードは渡されません)、代理ユーザー検査が推奨されます。

バッチ領域のユーザー ID と CICS 領域ユーザー ID が異なる場合、リンク・セキュリティ検査が実行されます。リンク・セキュリティ検査を使用する場合、DPL_Request 呼び出しで渡される非認証ユーザー ID は、リンク・セキュリティ検査によって許可されているものより多くの権限を獲得できません。それは、リンク・セキュリティ検査で許可されているのと同じ (またはそれより低い) 権限しか獲得できません。

CICS セキュリティの詳細については、[CICS TS セキュリティ](#) を参照してください。

第 5 章 EXCI のトラブルシューティング

重要: この情報には、診断、変更、またはチューニングのための情報が含まれます。

外部 CICS インターフェースは、問題判別に役立つ診断情報を提供します。

CICS は、以下の診断情報を提供します。

- トレース
- System dumps (システム・ダンプ)
- 外部 CICS インターフェースの MVS 04xx 異常終了
- EXCI サービス・トラップ、DFHXCTRA
- EXCI トレースの入り口点

外部 CICS インターフェースのメッセージおよび 異常終了コードの詳細については、[DFHEX メッセージおよび 04xx \(外部 CICS インターフェース\) 異常終了コード](#)を参照してください。

トレース

外部 CICS インターフェースは、トレース・データを 2 つの宛先 (内部トレース・テーブルと外部 MVS GTF データ・セット) に書き込みます。内部トレース・テーブルは、非 CICS MVS アドレス・スペースにあります。トレース・データは、フォーマット設定され、外部 CICS インターフェースによって生成されたダンプに組み込まれます。

トレース項目は、外部 CICS インターフェースによって、内部トレース・テーブル、MVS GTF データ・セット、またはその両方の宛先に発行されます。これらのメッセージは、[83 ページの『EXCI トレースの入り口点』](#)にリストされています。

外部 CICS インターフェースのトレースに GTF を使用するには、GTF ユーザー・トレースがアクティブであり、MVS イメージで GTF が開始されており、DFHXC OPT オプション・テーブルに GTF=ON を指定する必要があります。

GTF トレースを CICS サーバー領域と外部 CICS インターフェース 領域の両方で使用すると、トレース項目がインターリーブされ、CICS-EXCI 環境での問題判別に役立つことがあります。

注: 外部 CICS インターフェースは、外部 CICS インターフェース・アプリケーション・プログラム内に、ユーザー TCB ごとに個別のトレース・テーブルを維持します。

外部 CICS インターフェースは、どのような形式の補助トレースもサポートしません。

GTF トレースのフォーマット設定

GTF に書き込まれた外部 CICS インターフェースのトレース 項目をフォーマット設定するには、標準 CICS DFHTR730 トレース・フォーマット設定ルーチンを使用できます。

外部 CICS インターフェースのトレース 項目のフォーマットには、CICS と同じ FID および ID (つまり、FID=X'EF' および ID=X'F6C') を使用します。

システム・ダンプの使用

外部 CICS インターフェースは、一部のエラー条件では MVS SYSMDUMP を生成し、より重大な他の条件では MVS SDUMP を生成します。これらのダンプには、すべての外部 CICS インターフェース 制御ブロック、およびトレース項目が含まれています。

システム・ダンプのフォーマット設定

CICS IPCS verb 出口の DFHPD730 を使用して、システム・ダンプを形式設定することができます。

DFHPD730 を使用して外部 CICS インターフェースのダンプをフォーマット 設定する場合、以下のキーワードを使用できます。

KE

PSW とレジスター、およびすべての外部 CICS インターフェース 制御ブロックをフォーマット 設定します。

LD

外部 CICS インターフェース・モジュールがアドレス・スペースにロードされた場所のロード・マップをフォーマット 設定し、それぞれの PTF レベルを示します。

MRO

外部 CICS インターフェース・アドレス・スペースの MRO 制御ブロック (MVS 共通サービス域 (CSA) にある共通制御ブロックを含む) をフォーマット 設定します。このオプションは、CICS に接続されたパイプの CICS アドレス・スペース内にある一部の MRO ブロックもフォーマット 設定します。

PG

チャンネルおよびコンテナの PG 制御ブロックをフォーマット 設定します。

TR

外部 CICS インターフェースのトレース・テーブルをフォーマット 設定します。トレース・テーブルは簡略形式と完全形式でフォーマット 設定できます (TR=1 は、簡略トレースを生成します)。

SU

ダンプ要約を生成します。

複数の TCB

外部 CICS インターフェースは、複数の TCB が使用中のときにシステム・ダンプを取る場合、ダンプを要求した TCB の制御ブロックおよびトレース・テーブルのみをダンプします。

コンソール・コマンドを使用して外部 CICS アドレス・スペースのメモリー・ダンプを取る場合、CICS verb 出口ルーチン DFHPD730 は、ダンプ内で検出されたすべての TCB の制御ブロックおよびトレース・テーブルをフォーマット 設定します。

SYSMDUMP のキャプチャー

外部 CICS インターフェースによって生成された SYSMDUMP をキャプチャーするには、必ずクライアント・アプリケーション・プログラムの JCL に SYSMDUMP データ・セット用の DD ステートメントを含めてください。

コンソールでのダンプ用の MVS DUMP コマンドの使用

外部 CICS インターフェースによって自動的に取られるダンプに加えて、コンソールで **MVS DUMP** コマンドを入力することにより、クライアント・アプリケーション・プログラムを実行しているアドレス・スペースのダンプを強制的に取ることもできます。

CICS IPCS verb 出口ルーチン DFHPD730 を使用して、この方法で取られたダンプをフォーマット 設定できます。TSO、SDSF、または NetView® から DUMP コマンドを発行することもできます。

また、**DUMP** コマンドを使用して、CICS サーバー・アドレス・スペースの他にクライアント・アドレス・スペースのダンプも取ることができます。両方のアドレス・スペースを含むダンプをフォーマット 設定するには、CICS IPCS verb 出口ルーチン DFHPD730 を使用します。

EXCI サービス・トラップ、DFHXCTRA

ユーザー置き換え可能プログラム DFHXCTRA は、IBM サービス担当員の指示のもとで使用できます。これは、CICS で使用される DFHTRAP に相当します。このプログラムは、外部 CICS インターフェースがトレース項目を書き込むたびに呼び出されます。

DFHXCTRA は、以下のアクションの 1 つまたはすべてを実行できます。

1. 代わりにトレース項目を書き込むように、外部 CICS インターフェースに要求する
2. SDUMP を取るように、外部 CICS インターフェースに指示する
3. GTF への現在のトレース項目の書き込みをスキップするように、外部 CICS インターフェースに指示する
4. DFHXCTRA を使用不可にするように、外部 CICS インターフェースに指示する

CICS 提供のサンプル・バージョンの DFHXCTRA は、EXCI パイプ制御ブロックを解放しようとしているときに FREEMAIN エラーが発生したことを示すトレース項目を検出した場合、4 つの機能のすべてを実行します。

DFHXCTRA のソースは、CICSTS56.CICS.SDFHMAC で提供されています。DFHXCTRA に渡されるパラメーター・リストは、コピーブック DFHXCTRD に定義されています。これは CICSTS56.CICS.SDFHMAC で提供されています。DFHXCTRD は、DFHXCTRA によって使用されるすべての外部 CICS インターフェース・トレース・ポイントも定義しています。

RRMS を使用した問題判別

DPL 要求を調整するためにリカバリー可能リソース管理サービス (RRMS) が使用されている場合、RRMS から問題判別の追加情報を入手できます。

これを行うには、ISPF ダイアログ (リソース・リカバリー・サービス (RRS) によって提供される) を使用して、以下のことができます。

- RRS ログ・ストリームをブラウズできます。
- RRS リソース・マネージャーに関する情報を表示できます。
- RRS リカバリー単位に関する情報を表示できます。

ダイアログのインストール方法および使用法については、[z/OS MVS プログラミング: リソース・リカバリー](#)を参照してください。

EXCI トレースの入り口点

表 7. 外部 CICS インターフェースのトレース 項目				
ポイント ID	Module (モジュール)	Lvl	型	データ
EX 0001	DFHXCPRH	Exc	PIPE_ALREADY_OPEN	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン
EX 0002	DFHXCPRH	Exc	PIPE_ALREADY_CLOSED	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン
EX 0003	DFHXCPRH	Exc	VERIFY_BLOCK_FM_ERROR	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0005	DFHXCPRH	Exc	XCIPIP_FM_ERR	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン

表 7. 外部 CICS インターフェースのトレース 項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 0006	DFHXCPRH	Exc	IRP_IOAREA_FM_ERR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0008	DFHXCPRH	Exc	XFRASSTG1_FM_ERR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン
EX 0201	DFHXCPRH	Exc	NO_CICS_IRC_STARTED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0202	DFHXCPRH	Exc	NO_PIPE	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン 6. ターゲット CICS のアプリケーション ID
EX 0203	DFHXCPRH	Exc	NO_CICS_ON_OPEN	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン 6. ターゲット CICS のアプリケーション ID
EX 0204	DFHXCPRH	Exc	NO_CICS_ON_DPL_1	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン 6. ターゲット CICS のアプリケーション ID
EX 0205	DFHXCPRH	Exc	NO_CICS_ON_DPL_2	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン 6. ターゲット CICS のアプリケーション ID
EX 0206	DFHXCPRH	Exc	NO_CICS_ON_DPL_3	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン 6. ターゲット CICS のアプリケーション ID
EX 0301	DFHXCCH DFHXCCEP DFHXCCEP DFHXCCEP DFHXCCEP DFHXCCEP DFHXCCEP	EX 1	SMGF 入り口	1. SMGF Plist

表 7. 外部 CICS インターフェースのトレース 項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 0302	DFHXCCH DFHXCCP DFHXCCR DFHXCNVG DFHXCSCR DFHXC64	EX 1	SMGF 出口	1. SMGF Plist
EX 0303	DFHXCCH DFHXCCP DFHXCCR DFHXCNVG DFHXCSCR DFHXC64	Exc	SMGF 例外	1. MVS return code
EX 0403	DFHXCPRH	Exc	INVALID_APPL_NAME	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0405	DFHXCPRH	Exc	PIPE_NOT_CLOSED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン
EX 0406	DFHXCPRH	Exc	PIPE_NOT_OPEN	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン
EX 0407	DFHXCPRH	Exc	INVALID_USERID	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名
EX 0408	DFHXCPRH	Exc	INVALID_UOWID	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. UOWID
EX 0409	DFHXCPRH	Exc	INVALID_TRANSID	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名
EX 0414	DFHXCPRH	Exc	ABORT_RECEIVED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS のアプリケーション ID 5. Message to be returned
EX 0415	DFHXCPRH	Exc	INVALID_CONNECTION	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 接続名 5. ターゲット CICS のアプリケーション ID

表 7. 外部 CICS インターフェースのトレース項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 0416	DFHXCPRH	Exc	INVALID_CICS_RELEASE	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS のアプリケーション ID
EX 0417	DFHXCPRH	Exc	PIPE_MUST_CLOSE	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. パイプ・トークン
EX 0418	DFHXCPRH	Exc	INVALID_PIPE_TOKEN	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. パイプ・トークン
EX 0422	DFHXCPRH	Exc	SERVER_ABENDED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. DPL return area
EX 0423	DFHXCPRH	Exc	SURROGATE_CHECK_FAILED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ジョブ・ユーザー ID 5. 代理リソース名 6. ESM 戻りコードと理由コード
EX 0426	DFHXCPRH	Exc	INVALID_TRANSID2	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名
EX 0427	DFHXCPRH	Exc	INVALID_CCSID	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名
EX 0428	DFHXCPRH	Exc	INVALID_ENDIAN	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名
EX 0431	DFHXCPRH	Exc	COMMAREA_LEN_NOT_ALLOWED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名
EX 0432	DFHXCPRH	Exc	DATA_LEN_NOT_ALLOWED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名
EX 0433	DFHXCPRH	Exc	CCSID_NOT_ALLOWED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名
EX 0434	DFHXCPRH	Exc	ENDIAN_NOT_ALLOWED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名

表 7. 外部 CICS インターフェースのトレース 項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 0603	DFHXCPRH	Exc	XCUSER_GM_ERROR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0604	DFHXCPRH	Exc	XCPipe_GM_ERROR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0605	DFHXCPRH	Exc	VERIFY_BLOCK_GM_ERROR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0606	DFHXCPRH	Exc	SSI_VERIFY_FAILED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0607	DFHXCPRH	Exc	SVC_CALL_FAILED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0608	DFHXCPRH	Exc	IRP_LOGON_FAILURE	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. ターゲット CICS のアプリケーション ID 6. Logon name
EX 0609	DFHXCPRH	Exc	IRP_CONNECT_FAIL	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン 6. ターゲット CICS のアプリケーション ID
EX 0610	DFHXCPRH	Exc	IRP_DISC_FAIL	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. ターゲット CICS のアプリケーション ID 6. パイプ・トークン
EX 0611	DFHXCPRH	Exc	IRP_LOGOFF_FAILED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. ターゲット CICS のアプリケーション ID 6. パイプ・トークン
EX 0612	DFHXCPRH	Exc	TRANSFORM_1_ERROR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター

表 7. 外部 CICS インターフェースのトレース項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 0613	DFHXCPRH	Exc	TRANSFORM_4_ERROR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0614	DFHXCPRH	Exc	IRP_NULL_DATA	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. ターゲット CICS のアプリケーション ID
EX 0615	DFHXCPRH	Exc	IRP_NEG_RESPONSE	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. ターゲット CICS のアプリケーション ID
EX 0616	DFHXCPRH	Exc	IRP_SWITCH_PULL_ERR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. ターゲット CICS のアプリケーション ID 6. パイプ・トークン
EX 0617	DFHXCPRH	Exc	IRP_IOAREA_GM_ERR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0619	DFHXCPRH	Exc	IRP_BAD_IOAREA	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. IOAREA address
EX 0620	DFHXCPRH	Exc	IRP_PROTOCOL_ERR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS のアプリケーション ID 5. パイプ・トークン
EX 0621	DFHXCPRH	Exc	PIPE_RECOVERY_FAILURE	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS のアプリケーション ID 5. パイプ・トークン
EX 0622	DFHXCPRH	Exc	ESTAE_SETUP_FAIL	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0623	DFHXCPRH	Exc	ESTAE_INVOKED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. MVS 異常終了コード

表 7. 外部 CICS インターフェースのトレース 項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 0624	DFHXCPRH	Exc	TIMEDOUT	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. Server program name 5. ターゲット CICS のアプリケーション ID
EX 0625	DFHXCPRH	Exc	STIMER_SETUP_FAIL	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0626	DFHXCPRH	Exc	STIMER_CANCEL_FAIL	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0627	DFHXCPRH	Exc	INCORRECT_SVC_LEVEL	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. SVC instruction
EX 0628	DFHXCPRH	Exc	INCORRECT_IRP_LVL	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター
EX 0629	DFHXCPRH	Exc	SERVER_PROTOCOL_ERR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 接続名 5. CICS name
EX 0633	DFHXCPRH	Exc	INQUIRE_CHANNEL_FAILED	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名
EX 0800	DFHXCPRH	Exc	RESP shows LENGERR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. COMMAREA length 6. データ長
EX 0801	DFHXCPRH	Exc	RESP shows INVREQ	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. 指定された DPL オプション
EX 0802	DFHXCPRH	Exc	RESP shows PGMIDERR	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. プログラム名 5. ターゲット CICS のアプリケーション ID

表 7. 外部 CICS インターフェースのトレース 項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 0803	DFHXCPRH	Exc	RESP shows ROLLEDBACK	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. プログラム名 5. ターゲット CICS のアプリケーション ID
EX 0804	DFHXCPRH	Exc	RESP shows NOTAUTH	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. プログラム名 5. ターゲット CICS のアプリケーション ID
EX 0805	DFHXCPRH	Exc	RESP shows SYSIDERR	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. プログラム名 5. ターゲット CICS のアプリケーション ID 6. DPL_Retarea
EX 0806	DFHXCPRH	Exc	RESP shows TERMERR	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. プログラム名 5. ターゲット CICS のアプリケーション ID
EX 0904	DFHXCTRP	Exc	Overlength trace data field	<ol style="list-style-type: none"> 1. XCTRP parameter list
EX 0905	DFHXCTRA	Exc	DFHXCTRA trace entry	<ol style="list-style-type: none"> 1. User specified data

表 7. 外部 CICS インターフェースのトレース 項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 1000	DFHXCPRH	EX 1	Entry	<p>INIT_USER コマンドの場合:</p> <ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 呼び出し元のレジスター 14 <p>Allocate_Pipe 要求の場合:</p> <ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. CICS 名 5. 割り振りオプション 6. 呼び出し元のレジスター 14 <p>オープン、クローズ、および割り振り解除の要求の場合:</p> <ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. CICS 名 5. パイプ・トークン 6. Caller's register 14 <p>For DPL requests:</p> <ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. CICS 名 5. パイプ・トークン 6. プログラム名 7. Caller's register 14
EX 1001	DFHXCPRH	EX 1	Exit	<p>INIT_USER、OPEN、CLOSE、および DEALLOCATE 要求の場合:</p> <ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. 呼び出し元のレジスター 14 <p>割り振り要求の場合:</p> <ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. 戻りコードおよびメッセージ・ポインター 5. パイプ・トークン 6. Caller's register 14 <p>For DPL requests:</p> <ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS システム 5. パイプ・トークン

表 7. 外部 CICS インターフェースのトレース 項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 1010	DFHXCEIP	EX 1	Entry	1. プログラム名 2. ターゲット CICS のアプリケーション ID 3. トランザクション ID 4. Caller's register 14 5. COMMAREA の最初の 100 バイトまで (渡された場合) 6. COMMAREA 長さ、COMMAREA が渡された場合 7. データ長、COMMAREA が渡された場合
EX 1011	DFHXCEIP	EX 1	Exit	1. EXEC retarea 2. プログラム名 3. ターゲット CICS のアプリケーション ID 4. トランザクション ID 5. Caller's register 14 6. COMMAREA の最初の 100 バイトまで (渡された場合) 7. COMMAREA 長さ、COMMAREA が渡された場合
EX 1020	DFHXCEIP	EX 1	Entry2	1. プログラム名 2. Target CICS applid 3. トランザクション ID 4. Caller's register 14 5. チャンネル名
EX 1021	DFHXCEIP	EX 1	Exit2	1. プログラム名 2. Target CICS applid 3. トランザクション ID 4. Caller's register 14 5. チャンネル名
EX 1022	DFHXCEIP	EX 1	Entry3	1. Caller's register 14
EX 1023	DFHXCEIP	EX 1	Exit3	1. Caller's register 14
EX 2000	DFHXCPRH	EX 2	IRP_LOGON	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS のアプリケーション ID 5. IRP ユーザー ID 6. SLCB アドレス 7. 接続名
EX 2001	DFHXCPRH	EX 2	IRP_CONN	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS のアプリケーション ID 5. IRP ユーザー ID 6. IRP スレッド ID 7. SCCB アドレス
EX 2002	DFHXCPRH	EX 2	IRP_DISC	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS のアプリケーション ID 5. パイプ・トークン

表 7. 外部 CICS インターフェースのトレース 項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 2003	DFHXCPRH	EX 2	IRP_LOGOFF	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. パイプ・トークン 5. IRP ユーザー ID
EX 2004	DFHXCPRH	EX 2	IRP_SWITCH	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS のアプリケーション ID 5. IRP ユーザー ID 6. IRP ユーザー・スレッド
EX 2005	DFHXCPRH	EX 2	IRP_SWITCH_DATA	<ol style="list-style-type: none"> 1. ユーザーのアプリケーション名 2. パイプ・トークン 3. Request header 4. Bind data 5. UOWID/USERID FMH 6. CICS への変換された DPL 要求 (最大 1000 バイト) 7. Final 1000 bytes of transformed DPL request
EX 2006	DFHXCPRH	EX 2	IRP_DATA	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. ターゲット CICS のアプリケーション ID 5. Length of data returned 6. Data (first 1000 bytes) 7. Data (final 1000 bytes)
EX 2007	DFHXCPRH	EX 2	PRE_URM	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. Parameters passed to DFHXCURM 5. URMINV, reason for calling URM 6. URMICICS、ターゲット CICS のアプリケーション ID 7. URMANCH, URM anchor point address
EX 2008	DFHXCPRH	EX 2	POST_URM	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. Parameters passed to DFHXCURM 5. URMINV, reason for calling URM 6. URMICICS、ターゲット CICS のアプリケーション ID 7. URMANCH, URM anchor point address
EX 2009	DFHXCPRH	EX 2	PRE-RACROUTE	<ol style="list-style-type: none"> 1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. Userid 5. 代理リソース名 6. RACROUTE パラメーター・リスト

表 7. 外部 CICS インターフェースのトレース 項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 200A	DFHXCPRH	EX 2	POST-RACROUTE	1. 呼び出し元のパラメーター・リスト 2. 呼び出しタイプ 3. 呼び出し元のユーザー名 4. Userid 5. 代理リソース名 6. RACROUTE パラメーター・リスト
EX 3000	DFHXCEIP	Exc	ESTAE_SETUP_ERROR	1. 戻り域 (20 バイト) 2. MVS 戻りコード
EX 3001	DFHXCEIP	Exc	ESTAE_INVOKED	1. 戻り域 (20 バイト)
EX 3002	DFHXCEIP	Exc	INV_CTYPE_ON_INIT	1. 戻り域 (20 バイト) 2. 呼び出しタイプ
EX 3003	DFHXCEIP	Exc	INV_VNUM_ON_INIT	1. 戻り域 (20 バイト) 2. バージョン番号
EX 3004	DFHXCEIP	Exc	INV_APPL_NAME_ON_INIT	1. 戻り域 (20 バイト) 2. ユーザー名
EX 3005	DFHXCEIP	Exc	INV_CTYPE_ON_ALLOC	1. 戻り域 (20 バイト) 2. 呼び出しタイプ
EX 3006	DFHXCEIP	Exc	INV_VNUM_ON_ALLOC	1. 戻り域 (20 バイト) 2. バージョン番号
EX 3007	DFHXCEIP	Exc	INV_UTOKEN_ON_ALLOC	1. 戻り域 (20 バイト) 2. ユーザー・トークン
EX 3008	DFHXCEIP	Exc	INV_CTYPE_ON_OPEN	1. 戻り域 (20 バイト) 2. 呼び出しタイプ
EX 3009	DFHXCEIP	Exc	INV_VNUM_ON_OPEN	1. 戻り域 (20 バイト) 2. バージョン番号
EX 3010	DFHXCEIP	Exc	INV_UTOKEN_ON_OPEN	1. 戻り域 (20 バイト) 2. ユーザー・トークン
EX 3011	DFHXCEIP	Exc	INV_PTOKEN_ON_OPEN	1. 戻り域 (20 バイト) 2. パイプ・トークン
EX 3012	DFHXCEIP	Exc	INV_CTYPE_ON_DPL	1. 戻り域 (20 バイト) 2. 呼び出しタイプ
EX 3013	DFHXCEIP	Exc	INV_VNUM_ON_DPL	1. 戻り域 (20 バイト) 2. バージョン番号
EX 3014	DFHXCEIP	Exc	INV_UTOKEN_ON_DPL	1. 戻り域 (20 バイト) 2. ユーザー・トークン
EX 3015	DFHXCEIP	Exc	INV_PTOKEN_ON_DPL	1. 戻り域 (20 バイト) 2. パイプ・トークン
EX 3017	DFHXCEIP	Exc	INV_USERID_ON_DPL	1. 戻り域 (20 バイト)
EX 3018	DFHXCEIP	Exc	PIPE_NOT_OPEN_ON_DPL	1. 戻り域 (20 バイト) 2. パイプ・トークン
EX 3019	DFHXCEIP	Exc	PIPE_MUST_CLOSE_ON_DPL	1. 戻り域 (20 バイト) 2. パイプ・トークン
EX 3020	DFHXCEIP	Exc	INV_CTYPE_ON_CLOSE	1. 戻り域 (20 バイト) 2. 呼び出しタイプ

表 7. 外部 CICS インターフェースのトレース項目 (続き)

ポイント ID	Module (モジュール)	Lvl	型	データ
EX 3021	DFHXCEIP	Exc	INV_VNUM_ON_CLOSE	1. 戻り域 (20 バイト) 2. バージョン番号
EX 3022	DFHXCEIP	Exc	INV_UTOKEN_ON_CLOSE	1. 戻り域 (20 バイト) 2. ユーザー・トークン
EX 3023	DFHXCEIP	Exc	INV_PTOKEN_ON_CLOSE	1. 戻り域 (20 バイト) 2. パイプ・トークン
EX 3024	DFHXCEIP	Exc	INV_CTYPE_ON_DEALL	1. 戻り域 (20 バイト) 2. 呼び出しタイプ
EX 3025	DFHXCEIP	Exc	INV_VNUM_ON_DEALL	1. 戻り域 (20 バイト) 2. バージョン番号
EX 3026	DFHXCEIP	Exc	INV_UTOKEN_ON_DEALL	1. 戻り域 (20 バイト) 2. ユーザー・トークン
EX 3027	DFHXCEIP	Exc	INV_PTOKEN_ON_DEALL	1. 戻り域 (20 バイト) 2. パイプ・トークン
EX 3028	DFHXCEIP	Exc	PIPE_NOT_CLOSED_ON_DEALL	1. 戻り域 (20 バイト) 2. パイプ・トークン
EX 3029	DFHXCEIP	Exc	XCEIP_RETRYING	1. 戻り域 (20 バイト)
EX 3030	DFHXCEIP	Exc	SURROGATE_CHECK_FAILED	1. 戻り域 (20 バイト)
EX 4000	DFHXCGR	EX 1	Entry	1. DFHXCGR パラメーター・リスト
EX 4001	DFHXCGR	EX 2	Exit	1. DFHXCGR パラメーター・リスト
EX 4002	DFHXCGR	EX 1	PRE_SVC1	1. SVC パラメーター・リスト
EX 4003	DFHXCGR	EX 1	POST_SVC	1. SVC パラメーター・リスト
EX 4004	DFHXCGR	Exc	RRMS_NOT_SUPPORTED	1. なし
EX 4005	DFHXCGR	Exc	RRMS_ERROR	1. なし
EX 4006	DFHXCGR	Exc	SVC_EXCEPTION	1. SVC 戻りコード
EX 4007	DFHXCGR	Exc	GETMAIN ERROR	1. なし
EX 4201	DFHXCCR	Entry	S2GF GETMAIN	1. S2GF Plist
EX 4202	DFHXCCR	Exit	S2GF GETMAIN	1. S2GF Plist
EX 4203	DFHXCCR	Exc	GETMAIN failure	1. MVS IARCP64 作成の戻りコード
EX 4204	DFHXCCR	Exc	GETMAIN failure	1. MVS IARCP64 取得の戻りコード
EX 5000	DFHXCBA	Ex 1	Entry	1. Arg0 2. EIEI Plist
EX 5001	DFHXCBA	Ex 1	Exit	1. Arg0 2. EIEI Plist
EX 5002	DFHXCBA	Exc	無効な関数	1. EIEI Plist
EX 5003	DFHXCBA	Exc	無効なフォーマット	1. EIEI Plist
EX 5004	DFHXCBA	Exc	Invalid API function	1. Arg0
EX 5005	DFHXCBA	Exc	ESTAE_INVOKED	1. EIEI Plist

第 6 章 EXCI 呼び出しで返される応答と理由コード

このセクションでは、EXCI 呼び出しインターフェースで返される応答の理由コードについて詳述します。

注：応答と理由コードの数値はすべて 10 進数で示されています。

応答 OK の理由コード

0: NORMAL

呼び出しは正常に完了しました。

説明

応答 WARNING の理由コード

1: PIPE_ALREADY_OPEN

既に関いているパイプに対して Open_Pipe 要求が発行されました。

説明

システム・アクション

なし。パイプは開いたまま残ります。

ユーザー応答

この応答が予期しないものである場合、Open_Pipe 呼び出しで誤ったパイプ・トークンが使用されていないかを調査してください。

2: PIPE_ALREADY_CLOSED

既に閉じているパイプに対して Close_Pipe 要求が発行されました。

説明

システム・アクション

外部 CICS インターフェースは要求を無視し、パイプは閉じたまま残ります。

ユーザー応答

この応答が予期しないものである場合、Close_Pipe 呼び出しで正しいパイプ・トークンが指定されていることを確認してください。

3: VERIFY_BLOCK_FM_ERROR

Initialize_User 処理には、SSI 検査呼び出しのパラメーター・リストを作成するために、16MB 境界より下のストレージが必要です。この領域の FREEMAIN 中に、エラーが発生しました。

説明

システム・アクション

FREEMAIN からの戻りコードが EXCI 副理由フィールド 1 に返されます。Initialize_User 要求は影響を受けずに続行されます。

ユーザー応答

問題が続く場合は、バッチ領域のメモリー・ダンプを取得し、そのメモリー・ダンプと MVS FREEMAIN からの戻りコードを一緒に使用して、FREEMAIN が失敗した理由を判別します。

4: WS_FREEMAIN_ERROR

作業用ストレージを FREEMAIN しようとしたましたが、結果として MVS FREEMAIN エラーが発生しました。

説明

システム・アクション

FREEMAIN からの戻りコードが EXCI 副理由フィールド 1 に返されます。Initialize_User 要求は影響を受けずに続行されます。

ユーザー応答

問題が続く場合は、バッチ領域のメモリー・ダンプを取得し、そのメモリー・ダンプと MVS FREEMAIN からの戻りコードを一緒に使用して、FREEMAIN が失敗した理由を判別します。

5: XCPIPE_FREEMAIN_ERROR

パイプ・ストレージを FREEMAIN しようとしたましたが、MVS FREEMAIN エラーが発生しました。

説明

システム・アクション

FREEMAIN からの戻りコードが EXCI 副理由フィールド 1 に返されます。ただし、外部 CICS インターフェースは、Deallocate_Pipe 要求の処理を続行します。他の理由で要求が失敗した場合、この理由コードは上書きされます。

ユーザー応答

問題が続く場合は、クライアント・アプリケーション・プログラムのアドレス・スペースのメモリー・ダンプを取得し、そのメモリー・ダンプと MVS FREEMAIN からの戻りコードを使用して、FREEMAIN が失敗した理由を判別します。

6: IRP_IOAREA_FM_FAILURE

MRO 入出力域を FREEMAIN しようとしたましたが、MVS FREEMAIN エラーが発生しました。

説明

システム・アクション

FREEMAIN からの戻りコードが EXCI 副理由フィールド 1 に返されますが、DPL 要求は完了するまで続行されます。理由 IRP_IOAREA_FM_FAILURE がアプリケーションに返されるのは、DPL 要求が完了した場合のみです。それ以外の場合は、後続の応答コードおよび理由コードによって上書きされます。

ユーザー応答

問題が続く場合は、バッチ領域のメモリー・ダンプを取得し、そのメモリー・ダンプと MVS FREEMAIN からの戻りコードを使用して、FREEMAIN が失敗した理由を判別します。

7: SERVER_TERMINATED

サーバー・プログラムが実行されている CICS セッションが、CICS によって解放されました。

説明

システム・アクション

CICS アプリケーション・サーバー・プログラムは処理中のある時点で切り離されており、このエラーに関するトレース項目を書き込む外部 CICS インターフェースに制御が返されます。

ユーザー応答

このエラーの理由である可能性が最も高いのは、サーバー・プログラムが原因で (おそらく EXEC CICS PERFORM SHUTDOWN コマンドによって) CICS が終了したことです。シャットダウン中に CICS が EXCI セッションを解放するため、シャットダウンを完了することができます。

8: XFRSTG1_FM_FAILURE

伝送領域を FREEMAIN しようとしたましたが、MVS FREEMAIN エラーが発生しました。

説明

システム・アクション

FREEMAIN からの戻りコードが EXCI 副理由フィールド 1 に返されますが、DPL 要求は完了するまで続行されます。理由 XFRSTG1_FM_FAILURE がアプリケーションに返されるのは、DPL 要求が完了した場合のみです。それ以外の場合は、後続の応答コードおよび理由コードによって上書きされます。

ユーザー応答

問題が続く場合は、バッチ領域のメモリー・ダンプを取得し、そのメモリー・ダンプと MVS FREEMAIN からの戻りコードを使用して、FREEMAIN が失敗した理由を判別します。

応答 RETRYABLE の理由コード

201: NO_CICS_IRC_STARTED

前回の IPL 以降に IRC アクティビティがなかった MVS イメージに対して Initialize_User コマンドが発行されました。外部 CICS インターフェースは CICS SVC 番号を判別できません。

説明

システム・アクション

Initialize_User 呼び出しは失敗し、外部 CICS インターフェースがユーザー置き換え可能モジュール DFHXCURM を呼び出します。

ユーザー応答

MVS イメージ内の CICS 領域が IRC にログオンしている (つまり、システム初期設定パラメーター IRCSTRT=YES を指定して開始されているか、OPEN IRC コマンドを使用して IRC を動的に開始している) ことを確認してください。あるいは、MVS イメージ内にローカル CICS 領域がない場合は、DFHXCOPT テーブルで CICSSVC パラメーターをコーディングすることで、外部 CICS インターフェースが使用する SVC パラメーターを指定する必要があります。この状態は、XCF を使用して別の MVS イメージ内の CICS 領域と通信している場合に発生する可能性があります。問題が解決されたら、Initialize_User 要求を再発行してください。

202: NO_PIPE

パイプを開こうとしましたが、パイプに関連付けられているターゲット CICS システムに空き受信セッションがありません。

説明

システム・アクション

Open_pipe 呼び出しは失敗し、外部 CICS インターフェースがユーザー 置き換え可能モジュール DFHXCURM を呼び出します。

ユーザー応答

この状態は、クライアント・アプリケーション・プログラムが (Allocate_Pipe 呼び出しを使用して) 割り振ったパイプの数が、ターゲット接続で定義されている受信セッションの数を超えていない場合にも発生する可能性があります。これは、CICS が Close_Pipe 要求によってパイプのクリーンアップを処理中である可能性があるためです。そのため、SESSIONS リソース定義を CICS に定義する場合には、理論的に必要な数より大きい RECEIVECOUNT 値を指定することが推奨されます。アプリケーション・プログラムは、Open_Pipe 要求を再発行することができます。

203 (Open_Pipe 呼び出し時): NO_CICS

パイプを開こうとしましたが、ターゲット CICS システムが使用可能ではないか、まだ IRC を開いていないか、ターゲット接続がサービス休止状態であるか、関連する EXCI 接続定義がターゲット CICS にインストールされていません。

説明

システム・アクション

パイプを開く要求は失敗し、外部 CICS インターフェースがユーザー 置き換え可能モジュール DFHXCURM を呼び出します。

ユーザー応答

副理由フィールド 1 がゼロ以外の場合 (IRP 応答コード (R15))、副理由フィールド 2 には、IRP 理由コードが含まれます。IRP 戻りコードの説明については、[データ域](#)の領域間制御ブロックを参照してください。IRP 戻りコードは、DFHIRSPS コピーブックの IRC という見出しの下にリストされています。

問題を修正したら、クライアント・アプリケーション・プログラムは Open_Pipe 呼び出しを再発行することができます。

204: WRONG_MVS_FOR_RRMS

SYNCONRETURN オプションを省略した DPL 要求が行われ、バッチ・プログラムとは別の MVS システム上にある CICS 領域を指定しています。ターゲット・システムでリカバリー可能リソース管理サービス (RRMS) のコンテキストが認識されないため、要求は拒否されます。

説明

システム・アクション

DPL 要求は失敗し、外部 CICS インターフェースがユーザー 置き換え可能モジュール DFHXCURM を呼び出します。

ユーザー応答

DPL 要求を発行したバッチ・プログラムと、その要求が送信される CICS 領域が、同じ MVS システム上にあることを確認してください。

205: RRMS_NOT_AVAILABLE

リソース・リカバリー・サービス (RRS)が使用できないときに、SYNCONRETURN オプションを省略した DPL 要求が行われました。

説明

以下の 2 つのケースがあります。

- リソース・リカバリー・サービス (RRS) が使用できない場合。
- SYNCONRETURN オプションを省略した最後の DPL 要求以降にリソース・リカバリー・サービスが再始動され、その間に同期点がない場合。

--

注: RRS は、リカバリー可能リソース管理サービス (RRMS) の一部です。

システム・アクション

DPL 要求は失敗し、外部 CICS インターフェースがユーザー 置き換え可能モジュール DFHXCUM を呼び出します。

ユーザー応答

SYNCONRETURN オプションを省略した最後の DPL 要求以降にリソース・リカバリー・サービスが再始動され、その間に同期点がない場合は、DPL 要求を再試行してください。

応答 USER_ERROR の理由コード

401: INVALID_CALL_TYPE

説明

この EXCI 要求に、無効な呼び出しタイプ・パラメーター値が指定されています。

システム・アクション

要求はリジェクトされます。

ユーザー応答

EXCI クライアント・プログラムを調べて、以下のように、*call_type* パラメーターが EXCI 呼び出しに適切な値を指定していることを確認してください。

- 1 Initialize_User
- 2 Allocate_Pipe
- 3 Open_Pipe
- 4 Close_Pipe
- 5 Deallocate_Pipe
- 6 DPL

402: INVALID_VERSION_NUMBER

説明

version_number パラメーターが指定する値が 1、2、または 3 のいずれでもありません。

システム・アクション

要求はリジェクトされます。

ユーザー応答

クライアント・アプリケーション・プログラムを調べて、すべての EXCI 呼び出しでバージョン番号として 1、2、または 3 が指定されていることを確認してください。

403: INVALID_APPL_NAME

説明

user_name パラメーターはすべてブランク文字 (X'40') で構成されています。

システム・アクション

呼び出しは拒否されます。

ユーザー応答

有効な非ブランク・ユーザー名を指定するようにアプリケーション・プログラムを変更してください。

404: INVALID_USER_TOKEN

説明

クライアント・アプリケーション・プログラムが、外部 CICS インターフェースが認識していないユーザー・トークンを使用して EXCI 要求を発行しました。

システム・アクション

要求はリジェクトされます。

ユーザー応答

Initialize_User 呼び出しが返す 4 バイトのトークンは、そのユーザーに対するそれ以降のすべての要求で使用される必要があります。クライアント・アプリケーション・プログラムを調べて、正しいトークンが渡されるようにエラーを修正してください。

405: PIPE_NOT_CLOSED

説明

まだ閉じていないパイプに対して Deallocate_Pipe 要求が発行されました。

システム・アクション

外部 CICS インターフェースは要求を無視し、パイプは開いたまま残ります。

ユーザー応答

クライアント・アプリケーション・プログラムを調べて、Deallocate_Pipe 要求が意図したものであることを確認してください。意図したものであった場合は、Deallocate_Pipe 要求を発行する前にパイプに対して Close_Pipe 要求を発行してください。

406: PIPE_NOT_OPEN

説明

開いていないパイプに対して DPL 呼び出しが発行されました。

システム・アクション

外部 CICS インターフェースは DPL 要求を拒否します。

ユーザー応答

クライアント・アプリケーション・プログラムを調べて、DPL 要求でパイプを使用する前に Open_Pipe 要求が発行されていることを確認してください。アプリケーション・プログラムによって Open_Pipe が発行された場合は、すべての DPL 要求が行われる前に誤ってそのパイプが閉じられていないことを確認してください。

407: INVALID_USERID

すべてブランクから構成された USERID パラメーターを指定して DPL 要求が発行されました。

説明

システム・アクション

DPL 要求は拒否されます。

ユーザー応答

EXCI クライアント・プログラムを調べて、DPL 要求が有効な USERID パラメーターを渡していることを確認してください。ユーザー ID を指定しない場合は、userid にヌル・アドレスを指定した呼び出しパラメーター・リストをコーディングします。ヌル・アドレスを渡すと、外部 CICS インターフェースは、クライアント・アプリケーション・プログラムを実行しているユーザー ID (バッチ領域のユーザー ID) を渡します。

408: INVALID_UOWID

無効な長さのフィールドを持つ uowid パラメーターを指定して DPL 要求が発行されました。

説明

システム・アクション

DPL 要求は拒否されます。

ユーザー応答

クライアント・アプリケーション・プログラムを調べて、DPL 要求が有効な uowid パラメーターを渡していることを確認してください。作業単位 ID を指定しない場合は、uowid にヌル・アドレスを指定した呼び出しパラメーター・リストをコーディングします。この場合、外部 CICS インターフェースが自動的に作業単位 ID を生成します。

409: INVALID_TRANSID

すべてブランクから構成された *transid* パラメーターを指定して DPL 要求が発行されました。

説明

システム・アクション

DPL 要求は拒否されます。

ユーザー応答

クライアント・アプリケーション・プログラムを調べて、*transid* パラメーターが正しく構成されていること、あるいは何らかの方法で上書きされていないことを確認してください。独自のトランザクション ID を指定しない場合は、*transid* にヌル・アドレスを指定した呼び出しパラメーター・リストをコーディングします。この場合、外部 CICS インターフェースは、デフォルトの CICS ミラー・トランザクション CSMI を使用します。

410: DFHMEBM_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、外部 CICS インターフェース・メッセージの発行準備においてメイン・メッセージ・モジュールをロードしようとしたましたが、このモジュールのロードに失敗しました。

説明

システム・アクション

Initialize_User 呼び出しは拒否されます。MVS ロード・マクロ (R15) からの戻りコードが副理由フィールド 1 に返されます。外部 CICS インターフェースがエラーを処理し、発生した異常終了 (R0) を副理由フィールド 2 に返します。

ユーザー応答

MVS 戻りコードを使用して、ロードが失敗した理由を判別します。最も可能性が高い理由は、バッチ・ジョブの STEPLIB 内に組み込まれているどのライブラリーにもメッセージ・モジュール DFHMEBMX がないことです。STEPLIB 連結に CICSTS56.CICS.SDFHEXCI ライブラリーが含まれていることを確認し、クライアント・アプリケーション・プログラムを再度開始してください。

411: DFHMET4E_LOAD_FAILED

メッセージ・モジュール DFHMET4E のロードに失敗しました。Initialize_User 処理中に、外部 CICS インターフェースが、メッセージの発行準備においてメッセージ・テーブルをロードしようとした。このモジュールのロードに失敗しました。

説明

システム・アクション

Initialize_User 呼び出しは拒否されます。MVS ロード・マクロ (R15) からの戻りコードが副理由フィールド 1 に返されます。外部 CICS インターフェースがエラーを処理し、発生した異常終了 (R0) を副理由フィールド 2 に返します。

ユーザー応答

MVS 理由コードを使用して、ロードが失敗した理由を判別します。最も可能性が高い理由は、バッチ・ジョブの STEPLIB 内に組み込まれているどのライブラリーにもメッセージ・テーブル DFHMET4E がないことです。STEPLIB 連結に CICSTS56.CICS.SDFHEXCI ライブラリーが含まれていることを確認し、クライアント・アプリケーション・プログラムを再度開始してください。

412: DFHXCURM_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、ユーザー置き換え可能モジュール DFHXCURM をロードしようとして失敗しました。

説明

システム・アクション

Initialize_User 呼び出しは拒否されます。MVS ロード・マクロ (R15) からの戻りコードが副理由フィールド 1 に返されます。外部 CICS インターフェースがエラーを処理し、発生した異常終了 (R0) を副理由フィールド 2 に返します。

ユーザー応答

MVS 理由コードを使用して、ロードが失敗した理由を判別します。最も可能性が高い理由は、バッチ・ジョブの STEPLIB 内に組み込まれているどのライブラリーにもモジュール DFHXCURM がないことです。STEPLIB 連結にモジュールを含むライブラリーが含まれていることを確認し、クライアント・アプリケーション・プログラムを再度開始してください。

413: DFHXCTRA_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、トラップ・モジュール (DFHXCTRA) をロードしようとして失敗しました。

説明

システム・アクション

Initialize_User 呼び出しは拒否されます。MVS ロード・マクロ (R15) からの戻りコードが副理由フィールド 1 に返されます。外部 CICS インターフェースがエラーを処理し、発生した異常終了 (R0) を副理由フィールド 2 に返します。

ユーザー応答

MVS 理由コードを使用して、ロードが失敗した理由を判別します。最も可能性が高い理由は、バッチ・ジョブの STEPLIB 内に組み込まれているどのライブラリーにも DFHXCTRA がないことです。STEPLIB 連結にモジュールを含むライブラリーが含まれていることを確認し、クライアント・アプリケーション・プログラムを再度開始してください。

414: IRP_ABORT_RECEIVED

DPL 要求の処理中に CICS サーバー領域内でエラーが発生したため、異常終了 FMH7 フローが外部 CICS インターフェースに返されます。

説明

システム・アクション

クライアント・アプリケーション・プログラムにメッセージが返されます。これは、サーバー・プログラムが端末から開始されていた場合に、その端末に発行されるメッセージです。クライアント・アプリケーション・プログラムには、EXCI 戻り域のメッセージ・ポインター・フィールドにメッセージへのポインターが返されます。メッセージ・フォーマットの正確な定義については、EXCI 戻り域の説明を参照してください。パイプは、「クローズが必要」状態になります。

ユーザー応答

メッセージを使用して、エラーの原因を判別します。代表的な例としては、サーバー・トランザクションが、使用不可にされているか定義されていないために接続できなかった場合や、セキュリティの障害に

よって接続できなかった場合などがあります。問題を修正し、パイプを閉じてから再度開き、DPL 要求を再発行してください。

415: INVALID_CONNECTION_DEFN

ターゲット接続が外部 CICS クライアント・アプリケーション・プログラムで使用するよう定義されていないために、DPL 要求が CICS によって拒否されました。

説明

システム・アクション

DPL 要求は拒否され、パイプは、「クローズが必要」状態になります。

ユーザー応答

最も可能性が高い理由は、CICS サーバー領域内の接続定義が、誤って EXCI 接続ではなく CICS-to-CICS MRO 接続として定義されていることです。該当する CONNECTION および SESSIONS リソース定義で PROTOCOL(EXCI) が指定されていることを確認してください。DPL 要求を再発行する前に、パイプを閉じてから再度開く必要があります。

416: INVALID_CICS_RELEASE

ターゲット CICS サーバー領域は、DPL 要求を認識しないため、この要求を拒否しました。

説明

システム・アクション

DPL 呼び出しは拒否され、パイプは、「クローズが必要」状態になります。

ユーザー応答

この理由である可能性が最も高いのは、クライアント・アプリケーション・プログラムが、外部 CICS インターフェースをサポートしないターゲット CICS サーバー領域を指定したことです。CICS for MVS/ESA 4.1 より前の CICS 領域は、EXCI 呼び出し要求を認識しません。問題を修正し、パイプを閉じてから再度開いた後、DPL 要求を再発行してください。

417: PIPE_MUST_CLOSE

「クローズが必要」状態のパイプに対して DPL 要求が発行されました。

説明

システム・アクション

DPL 要求は拒否されます。

ユーザー応答

一部の EXCI エラーは重大であるため、それ以降の DPL 要求に使用できるポイントまでパイプを復元するためには、パイプを閉じてから再度開く必要がある場合があります。それ以外のマイナー・エラーの場合は、パイプを閉じてから再度開かなくても、その後の呼び出しを行うことができます。このパイプでの以前のエラーは、より重大な種類のものであり、現在、パイプは「クローズが必要」状態になっています。パイプを閉じてから再度開き、DPL 要求を再発行してください。

418: INVALID_PIPE_TOKEN

Open_Pipe、Close_Pipe、Deallocate_Pipe、または DPL 要求が発行されましたが、呼び出し時に渡されたパイプ・トークンが有効なパイプではないか、このユーザー用に割り振られた有効なパイプではありません(つまり、ユーザー・トークンとパイプ・トークンの間にミスマッチがあります)。

説明

システム・アクション

呼び出しは拒否されます。

ユーザー応答

パイプ・トークンが上書きされていないこと、および呼び出し時に正しく渡されていることを確認してください。また、ユーザー・トークンとパイプ・トークンの間にミスマッチがないことを確認してください。

419: CICS_AFCB_PRESENT

CICS または CICS バッチ共有データベースによって既に使用されている TCB に対して、Initialize_User 要求が出されました。外部 CICS インターフェースは、TCB を CICS と共有できません。これにより、CICS アプリケーション・プログラムは EXCI 要求を出すことができません。

説明

システム・アクション

Initialize_User 要求は拒否されます。

ユーザー応答

外部 CICS インターフェースを使用するには、新規の TCB (または daughter TCB) を作成し、その固有 TCB の下で EXCI 呼び出しを発行する必要があります。

420: DFHXCOPT_LOAD_FAILED

Initialize_User 処理中に、外部 CICS インターフェースが、オプション・モジュール DFHXCOPT をロードしようとした。このモジュールのロードに失敗しました。

説明

システム・アクション

Initialize_User 呼び出しは拒否されます。MVS ロード・マクロ (R15) からの戻りコードが副理由フィールド 1 に返されます。外部 CICS インターフェースがエラーを処理し、発生した異常終了 (R0) を副理由フィールド 2 に返します。

ユーザー応答

MVS 理由コードを使用して、ロードが失敗した理由を判別します。最も可能性が高い理由は、バッチ・ジョブの STEPLIB 内に組み込まれているどのライブラリーにも DFHXCOPT がないことです。問題を修正して、クライアント・アプリケーション・プログラムを再度開始してください。

421: RUNNING_UNDER_AN_IRB

EXCI 呼び出しが MVS IRB 下で発行されましたが、これは許可されません。

説明

システム・アクション

呼び出しは拒否されます。

ユーザー応答

呼び出しが IRB の下で発行された理由を判別し、クライアント・アプリケーション・プログラムを変更してください。

422: SERVER_ABENDED

DPL 要求の処理中に、CICS サーバー・アプリケーション・プログラムが、エラーを処理せずに異常終了しました。

説明

システム・アクション

サーバー・アプリケーション・プログラムは異常終了し、バックアウトされます。EXCI 戻り域の異常終了コード・フィールドに異常終了コードが返されます。

ユーザー応答

サーバー・プログラムが異常終了した理由を判別し、問題を修正してください。

423: SURROGATE_CHECK_FAILED

USERID パラメーターが指定された DPL 要求が出されました。

説明

EXCI オプション・テーブル DFHXCOPT で SURROGCHK=YES が指定されているため、指定されたユーザー ID は代理ユーザー・セキュリティの対象でした。代理ユーザー検査は失敗しました。代理セキュリティ検査は、EXCI バッチ領域のユーザー ID が、DPL 呼び出しで指定されているユーザー ID の代理として許可されているかどうかを検査します。

システム・アクション

DPL 呼び出しは拒否されます。MVS 外部セキュリティ・マネージャーの戻りコードおよび理由コードが、副理由フィールド 1 およびフィールド 2 に返されます。RACF の場合は、これらについて [z/OS Security Server RACROUTE マクロ 解説書](#)で説明しています。

ユーザー応答

代理ユーザー検査が必要な場合は、SURROGAT 一般リソース・クラスの EXCI バッチ領域のユーザー ID にプロファイル *userid.DFHEXCI* に対する読み取り権限があることを確認してください。ここで、*userid* は DPL 呼び出しで指定されたユーザー ID です。

代理ユーザー・セキュリティ検査が必要ない場合は、DFHXCOPT オプション・テーブルで SURROGCHK=NO を指定します。

詳細については、[代理ユーザー検査](#)を参照してください。

424: RRMS_NOT_SUPPORTED

SYNCONRETURN オプションを省略した DPL 要求が、z/OS リリース 5 またはそれ以降 (上位互換) のリリースで稼働していないシステム上で行われました。

説明

システム・アクション

呼び出しは拒否されます。

ユーザー応答

適切なレベルの z/OS が稼働しているシステム上でバッチ・プログラムが実行されていることを確認してください。

425: UOWID_NOT_ALLOWED

DPL 要求で SYNCONRETURN オプションが省略されましたが、UOWID の値が指定されました。このパラメーターの組み合わせは、DPL 要求では許可されません。

説明

システム・アクション

DPL 要求は拒否されます。

ユーザー応答

クライアント・アプリケーション・プログラムを調べて、DPL 呼び出しで正しいパラメーターの組み合わせが使用されていることを確認してください。

426: INVALID_TRANSID2

すべてブランクから構成された *transid2* パラメーターを指定して DPL 要求が発行されました。

説明

システム・アクション

DPL 要求は拒否されます。

ユーザー応答

クライアント・アプリケーション・プログラムを調べて、*transid2* パラメーターが正しく構成されていること、あるいは何らかの方法で上書きされていないことを確認してください。

427: INVALID_CCsid

説明

無効な値を指定する *ccsid* パラメーターを指定して DPL 要求が発行されました。

システム・アクション

DPL 要求は拒否されます。

ユーザー応答

クライアント・アプリケーション・プログラムを調べて、*ccsid* パラメーターが正しく構成されていること、あるいは何らかの方法で上書きされていないことを確認してください。

428: INVALID_ENDIAN

説明

無効な値を指定する *endian* パラメーターを指定して DPL 要求が発行されました。

システム・アクション

DPL 要求は拒否されます。

ユーザー応答

クライアント・アプリケーション・プログラムを調べて、*endian* パラメーターが正しく構成されていること、あるいは何らかの方法で上書きされていないことを確認してください。

429: DFHXCEIX_LOAD_FAILED

説明

EXEC CICS LINK 呼び出しの処理中に、外部 CICS インターフェースがモジュール (DFHXCEIX) をロードしようとした。このモジュールのロードに失敗しました。

システム・アクション

EXEC CICS LINK 呼び出しは拒否されます。

ユーザー応答

最も可能性が高い理由は、バッチ・ジョブの STEPLIB 内に組み込まれているどのライブラリーにも DFHXCEIX がないことです。STEPLIB 連結にモジュールを含むライブラリーが含まれていることを確認し、クライアント・アプリケーション・プログラムを再度開始してください。

430: DFHXCPRX_LOAD_FAILED

説明

Initialize_User 処理中に、外部 CICS インターフェースが、モジュール (DFHXCPRX) をロードしようとした。このモジュールのロードに失敗しました。

システム・アクション

Initialize_User 呼び出しは拒否されます。MVS ロード・マクロ (R15) からの戻りコードが副理由フィールド 1 に返されます。外部 CICS インターフェースがエラーを処理し、発生した異常終了 (R1) を副理由フィールド 2 に返します。

ユーザー応答

MVS 理由コードを使用して、ロードが失敗した理由を判別します。最も可能性が高い理由は、バッチ・ジョブの STEPLIB 内に組み込まれているどのライブラリーにも DFHXCPRX がないことです。STEPLIB 連結にモジュールを含むライブラリーが含まれていることを確認し、クライアント・アプリケーション・プログラムを再度開始してください。

431: COMMAREA_LEN_NOT_ALLOWED

説明

CHANNEL パラメーターおよび **COMMAREA_LEN** パラメーターを指定した DPL 要求が発行されました。

プログラム間でのデータの転送にチャンネルを使用している場合、**COMMAREA_LEN** パラメーターをヌルにする必要があります。

システム・アクション

要求はリジェクトされます。

ユーザー応答

EXCI プログラムを確認してください。そのプログラムにチャンネルを指定した DPL 要求が含まれている場合は、**COMMAREA_LEN** パラメーターをヌルにする必要があります。

432: DATA_LEN_NOT_ALLOWED

説明

CHANNEL パラメーターおよび **DATA_LEN** パラメーターを指定した DPL 要求が発行されました。

プログラム間でのデータの転送にチャンネルを使用している場合、**DATA_LEN** パラメーターをヌルにする必要があります。

システム・アクション

要求はリジェクトされます。

ユーザー応答

EXCI プログラムを確認してください。そのプログラムにチャンネルを指定した DPL 要求が含まれている場合は、**DATA_LEN** パラメーターをヌルにする必要があります。

433: CCSID_NOT_ALLOWED

説明

CHANNEL パラメーターおよび **CCSID** パラメーターを指定した DPL 要求が発行されました。

プログラム間でのデータの転送にチャンネルを使用している場合、**CCSID** パラメーターをヌルにする必要があります。

システム・アクション

要求はリジェクトされます。

ユーザー応答

EXCI プログラムを確認してください。そのプログラムにチャンネルを指定した DPL 要求が含まれている場合は、**CCSID** パラメーターをヌルにする必要があります。

434: ENDIAN_NOT_ALLOWED

説明

CHANNEL パラメーターおよび **ENDIAN** パラメーターを指定した DPL 要求が発行されました。

プログラム間でのデータの転送にチャンネルを使用している場合、**ENDIAN** パラメーターをヌルにする必要があります。

システム・アクション

要求はリジェクトされます。

ユーザー応答

EXCI プログラムを確認してください。そのプログラムにチャンネルを指定した DPL 要求が含まれている場合は、**ENDIAN** パラメーターをヌルにする必要があります。

応答 SYSTEM_ERROR の理由コード

601: WS_GETMAIN_ERROR

Initialize_User 処理中に、作業用ストレージの GETMAIN が失敗しました。

システム・アクション

作業用ストレージなしで処理を続行することはできないため、要求は終了します。この時点で、外部 CICS インターフェースのトレースおよびダンプ・サービスを使用して診断情報を提供することができないため、EXCI は MVS 異常終了 (U0408) を発行して強制的に SYSMDUMP を行います。MVS GETMAIN 要求からの戻りコードが戻り域に返されます。

ユーザー応答

ダンプ内で GETMAIN 戻りコードを見つけ、その戻りコードとダンプの残りの部分を使用して GETMAIN が失敗した理由を判別します。この理由として考えられるのは、ジョブ用に指定された領域サイズが小さすぎることです。その場合は、領域サイズを増やして、クライアント・アプリケーション・プログラムを再度開始してください。

602: XCGLOBAL_GETMAIN_ERROR

Initialize_User 処理中に、重要な制御ブロック (XCGLOBAL) の GETMAIN が失敗しました。

システム・アクション

この制御ブロックなしで処理を続行することはできないため、要求は終了します。この時点で、外部 CICS インターフェースのトレースおよびダンプ・サービスを使用して診断情報を提供することができないため、EXCI は MVS 異常終了 (U0403) を発行して強制的に SYSMDUMP を行います。MVS GETMAIN 要求からの戻りコードが戻り域に返されます。

ユーザー応答

ダンプ内で GETMAIN 戻りコードを見つけ、その戻りコードとダンプの残りの部分を使用して GETMAIN が失敗した理由を判別します。この理由として考えられるのは、ジョブ用に指定された領域サイズが小さすぎることです。その場合は、領域サイズを増やして、クライアント・アプリケーション・プログラムを再度開始してください。

603: XCUSER_GETMAIN_ERROR

Initialize_User 処理中に、ユーザー制御ブロック (XCUSER) の GETMAIN 要求が失敗しました。

システム・アクション

Initialize_User 処理は終了します。GETMAIN からの戻りコードが戻り域の副理由フィールド 1 に返されます。外部 CICS インターフェースがメッセージ DFHEX0003 を発行し、MVS ユーザー異常終了 (0410) を発行して強制的に SYSMDUMP を行います。

ユーザー応答

GETMAIN からの戻りコードとダンプを使用して、GETMAIN が失敗した理由を判別します。この理由として考えられるのは、ジョブの領域サイズが小さすぎることです。その場合は、領域サイズを増やして、クライアント・アプリケーション・プログラムを再度開始してください。

604: XCPIPE_GETMAIN_ERROR

Allocate_Pipe 処理中に、パイプ制御ブロック (XCPIPE) の GETMAIN 要求が失敗しました。

システム・アクション

Allocate_Pipe 処理は終了します。GETMAIN からの戻りコードが EXCI 戻り域の副理由フィールド 1 に返されます。外部 CICS インターフェースがメッセージ DFHEX0003 を発行し、システム・ダンプを取得します。

ユーザー応答

GETMAIN からの戻りコードとダンプを使用して、GETMAIN が失敗した理由を判別します。この理由として考えられるのは、ジョブの領域サイズが小さすぎることです。その場合は、領域サイズを増やして、クライアント・アプリケーション・プログラムを再度開始してください。

605: VERIFY_BLOCK_GM_ERROR

Initialize_User 処理中に、EXCI 内部制御ブロックの GETMAIN が失敗しました。

システム・アクション

Initialize_User 処理は終了します。GETMAIN からの戻りコードが EXCI 戻り域の副理由フィールド 1 に返されます。このエラーは、EXCI ダンプ・サービスの初期化前に発生するため、EXCI は、MVS 異常終了 (U0409) を発行して強制的に SYSMDUMP を行います。MVS GETMAIN 要求からの戻りコードが戻り域に返されます。

ユーザー応答

ダンプ内で GETMAIN 戻りコードを見つけ、その戻りコードとダンプの残りの部分を使用して GETMAIN が失敗した理由を判別します。この理由として考えられるのは、ジョブ用に指定された領域サイズが小さすぎることです。その場合は、領域サイズを増やして、クライアント・アプリケーション・プログラムを再度開始してください。

606: SSI_VERIFY_FAILED

現在の CICS SVC 番号を取得するための MVS サブシステム・インターフェース (SSI) に対する VERIFY 呼び出しが失敗しました。

システム・アクション

Initialize_User 要求は終了します。SSI 呼び出しからの戻りコードが戻り域の副理由フィールド 1 に返されます。このエラーは、外部 CICS インターフェースのダンプ・サービスが初期化される前に発生するため、EXCI は、MVS ユーザー異常終了 (0405) を発行して強制的に SYSMDUMP を行います。

ユーザー応答

ダンプ内で戻りコードを見つけ、その戻りコードとダンプの残りの部分および SSI 文書を使用して、VERIFY 要求が失敗した理由を判別します。問題が解決されたら、クライアント・アプリケーション・プログラムを再度開始してください。

607: CICS_SVC_CALL_FAILURE

Initialize_User 処理中に、現在インストールされている CICS SVC に対する呼び出しが失敗しました。

システム・アクション

CICS SVC からの戻りコードが EXCI 戻り域の副理由フィールド 1 に返されます。このエラーは、外部 CICS インターフェースのダンプ・サービスが初期化される前に発生するため、EXCI は、MVS ユーザー異常終了 (0406) を発行して強制的に SYSMDUMP を行います。

ユーザー応答

IBM サポート・センターに戻りコードと使用可能なダンプを提供して、支援を受けてください。

608: IRC_LOGON_FAILURE

Allocate_Pipe 処理中に、外部 CICS インターフェースによる DFHIRP への LOGON 試行が失敗しました。

システム・アクション

Allocate_Pipe 要求は失敗します。DFHIRP は、副理由フィールド 1 に R15 値を返し、副理由フィールド 2 に R0 値 (理由コード) を返します。副理由フィールド 1 の最初の 2 バイトは戻りコード修飾子で、最後の 2 バイトは戻りコード自体です。

ユーザー応答

IRP 戻りコードの説明については、[データ域](#)の領域間制御ブロックを参照してください。IRP 戻りコードは、DFHIRSPS コピーブックの IRC という見出しの下にリストされています。戻りコードを使用してログオンに失敗した理由を判別し、IBM サポート担当員に連絡して障害の詳細を提供してください。

609: IRC_CONNECT_FAILURE

Open_Pipe 処理中に、ターゲット CICS システムへの接続試行が失敗しました。

システム・アクション

Open_Pipe 要求は失敗します。DFHIRP は、副理由フィールド 1 に R15 値を返し、副理由フィールド 2 に R0 値 (理由コード) を返します。副理由フィールド 1 の最初の 2 バイトは戻りコード修飾子で、最後の 2 バイトは戻りコード自体です。

ユーザー応答

IRP 戻りコードの説明については、[データ域](#)の領域間制御ブロックを参照してください。IRP 戻りコードは、DFHIRSPS コピーブックの IRC という見出しの下にリストされています。

戻りコードを使用してログオンに失敗した理由を判別し、パイプのオープン要求を再発行してください。

注: このエラーは、ターゲット CICS が使用できない (その場合は、RETRYABLE 条件 (NO_CICS) として返されます) ことが原因ではありません。

610: IRC_DISCONNECT_FAILURE

Close_Pipe 処理中に、CICS が DFHIRP 切断呼び出しを発行し、CICS への接続を終了しました。この要求は失敗しました。

システム・アクション

呼び出しは失敗し、パイプは開いたまま残ります。DFHIRP は、副理由フィールド 1 に R15 値を返し、副理由フィールド 2 に R0 値 (理由コード) を返します。副理由フィールド 1 の最初の 2 バイトは戻りコード修飾子で、最後の 2 バイトは戻りコード自体です。外部 CICS インターフェースは、システム・ダンプを取得します。

切断は失敗しましたが、パイプが引き続き CICS に接続することは可能です。ただし、バッチ・プログラムの終了時にすべての接続が自動的に切断されます。

ユーザー応答

IRP 戻りコードの説明については、[データ域](#)の領域間制御ブロックを参照してください。IRP 戻りコードは、DFHIRSPS コピーブックの IRC という見出しの下にリストされています。戻りコードおよびダンプを使用して、エラーの原因を判別してください。

611: IRC_LOGOFF_FAILURE

Deallocate_Pipe 処理中に、CICS が DFHIRP ログオフ呼び出しを発行しました。この要求は失敗しました。

システム・アクション

Deallocate_Pipe 呼び出しは失敗し、パイプは割り振られたまま残ります。DFHIRP は、副理由フィールド 1 に R15 値を返し、副理由フィールド 2 に R0 値 (理由コード) を返します。副理由フィールド 1 の最初の 2 バイトは戻りコード修飾子で、最後の 2 バイトは戻りコード自体です。外部 CICS インターフェースは、システム・ダンプを取得します。

注: パイプは割り振られたまま残るため、その後の呼び出しに使用することができます。パイプに関連付けられたストレージは解放されません。ただし、このストレージは、クライアント・アプリケーション・プログラムの終了時に解放されます。

ユーザー応答

IRP 戻りコードの説明については、[データ域](#)の領域間制御ブロックを参照してください。IRP 戻りコードは、DFHIRSPS コピーブックの IRC という見出しの下にリストされています。戻りコードおよびダンプを使用して、エラーの原因を判別してください。

612: TRANSFORM_1_ERROR

DPL 処理中に、CICS への送信を準備中のデータを処理している際に、プログラム DFHXFQ に対する内部呼び出しによってエラーが発生しました。

システム・アクション

DPL 要求は終了します。

ユーザー応答

呼び出しからの戻りコードが EXCI 副理由フィールド 1 に返され、外部 CICS インターフェースはシステム・ダンプを取得します。

これは、外部 CICS インターフェースのエラーです。IBM サポート・センターに連絡して、戻りコードとダンプの詳細を提供してください。

613: TRANSFORM_4_ERROR

DPL 処理中に、CICS サーバー領域によって返されたデータを処理している際に、モジュール DFHXFQ に対する内部呼び出しによってエラーが発生しました。

システム・アクション

DPL 要求は終了します。サーバー・アプリケーション・プログラムは実行されているので注意してください。DFHXFQ に対する呼び出しからの戻りコードが EXCI 副理由フィールド 1 に返されます。この戻りコードは、使用可能であった EIBRCODE 情報に対応しています。外部 CICS インターフェースは、システム・ダンプを取得します。

ユーザー応答

これは、外部 CICS インターフェースのエラーです。IBM サポート・センターに連絡して、戻りコードとダンプの詳細を提供してください。

614: IRP_NULL_DATA_RECEIVED

DPL 処理中に、要求がターゲット CICS に送信され、このターゲット CICS がデータを返さずに応答しました。

システム・アクション

DPL 処理は終了し、外部 CICS インターフェースは、システム・ダンプを取得します。

ユーザー応答

これは、内部プロトコル・エラーです。IBM サポート・センターに連絡して、ダンプの詳細を提供してください。

615: IRP_NEGATIVE_RESPONSE

ターゲット CICS 領域との通信を試行中に、内部プロトコル・エラーが発生しました。

システム・アクション

DPL 要求は失敗し、パイプは「クローズが必要」状態になり、外部 CICS インターフェースはシステム・ダンプを取得します。

ユーザー応答

これは、外部 CICS インターフェースのエラーです。ダンプを保持して、IBM サポート・センターに連絡してください。

注：パイプは「クローズが必要」状態です。さらに呼び出しを試行する場合は、その前にパイプを閉じてから再度開く必要があります。

616: IRP_SWITCH_PULL_FAILURE

ターゲット CICS 領域との通信を試行中に、内部プロトコル・エラーが発生しました。

システム・アクション

DPL 要求は失敗し、パイプは「クローズが必要」状態になり、外部 CICS インターフェースはシステム・ダンプを取得します。IRP 戻りコード (R15) および理由コード (ある場合) (R0) が EXCI 副理由フィールド 1 および副理由フィールド 2 に返されます。

ユーザー応答

これは、外部 CICS インターフェースのエラーです。ダンプを保持して、IBM サポート・センターに連絡してください。

注：パイプは「クローズが必要」状態であるため、さらに DPL 呼び出しを試行する場合は、その前にパイプを閉じてから再度開く必要があります。

617: IRP_IOAREA_GM_FAILURE

DPL 処理中に、内部制御ブロックの MVS GETMAIN 要求が失敗しました。

システム・アクション

DPL 要求は終了します。GETMAIN からの戻りコードが EXCI 副理由フィールド 1 に返されます。

注：このエラーは、サーバー・アプリケーション・プログラムの実行が完了した後、CICS によって返されたデータの処理中に発生します。このエラーにより、パイプは「クローズが必要」状態になります。

ユーザー応答

戻りコードを使用して、GETMAIN が失敗した理由を判別します。この理由として考えられるのは、ジョブの領域サイズが小さすぎることです。その場合は、領域サイズを増やして、バッチ・ジョブを再度開始してください。

619: IRP_BAD_IOAREA

DPL 要求中に、使用できない入出力域が DFHIRP に提供されました。

システム・アクション

DPL 要求は終了し、パイプは強制的に「クローズが必要」状態になり、外部 CICS インターフェースはシステム・ダンプを取得します。

ユーザー応答

これは、外部 CICS インターフェースのエラーです。IBM サポート・センターに連絡して、戻りコードとダンプの詳細を提供してください。

注：このエラーの後、パイプは「クローズが必要」状態であるため、さらに呼び出しを試行する場合は、その前にパイプを閉じてから再度開く必要があります。

620: IRP_PROTOCOL_ERROR

ターゲット CICS システムとの通信を試行中に、内部プロトコル・エラーが発生しました。

システム・アクション

DPL 要求は終了し、パイプは強制的に「クローズが必要」状態になり、外部 CICS インターフェースはシステム・ダンプを取得します。

ユーザー応答

これは、外部 CICS インターフェースのエラーです。ダンプを保持して、IBM サポート・センターに連絡してください。

注：このエラーの後、パイプは「クローズが必要」状態であるため、さらに呼び出しを試行する場合は、その前にパイプを閉じてから再度開く必要があります。

621: PIPE_RECOVERY_FAILURE

パイプのオープン要求中にエラーが発生しました。外部 CICS インターフェースは、パイプを再度切断することでリカバリーを試みます。この切断中に、さらにエラーが発生しました。

システム・アクション

Open_Pipe 呼び出しは終了し、パイプは「クローズが必要」状態になります。DFHIRP からの戻りコードが EXCI 副理由フィールド 1 に返され、システム・ダンプが取得されます。

ユーザー応答

IRP 戻りコードの説明については、データ域の領域間制御ブロックを参照してください。IRP 戻りコードは、DFHIRSPS コピーブックの IRC という見出しの下にリストされています。ダンプおよび IRP 戻りコードを使用して、切断に失敗した理由を判別します。EXCI トレースを使用して、オープン・パイプ・リカバリー・ルーチンが呼び出される原因となった過去のエラーを判別することも推奨されます。

注：パイプは「クローズが必要」状態になっており、さらに呼び出しを発行する場合は、その前にパイプを閉じてから再度開く必要があります。

622: ESTAE_SETUP_FAILURE

実行される可能性のあるプログラム・チェックから保護するために、外部 CICS インターフェースが MVS ESTAE を確立します。このケースでは、MVS ESTAE マクロが失敗しました。

システム・アクション

呼び出しは終了し、MVS ESTAE コマンドからの戻りコードが EXCI 副理由フィールド 1 に返されます。このエラーは、EXCI ダンプ・サービスが初期化される前に発生する可能性があるため、EXCI は、MVS 異常終了 (U0402) を発行して強制的に SYSMDUMP を行います。

ユーザー応答

戻りコードおよびダンプを使用して、ESTAE コマンドが失敗した理由を判別します。これは、内部 EXCI エラーである可能性があるため、問題が解決しない場合は、IBM サポート・センターに連絡してください。

623: ESTAE_INVOKED

呼び出し処理中にプログラム・チェックが実行され、ESTAE が呼び出されます。

システム・アクション

EXCI ESTAE によってプログラム・チェックが処理され、それ以降の EXCI 呼び出しをサポートできる状態へのリカバリーが試行されます。MVS 異常終了コードが戻り域の EXCI 副理由フィールド 1 に返されます。さらに詳細な診断を支援するために、SYSMDUMP が取得されます。

ユーザー応答

戻りコードおよびダンプを使用して、外部 CICS インターフェースでプログラム・チェックが発生した理由を判別します。この理由である可能性が最も高いのは、クライアント・プログラムのパラメーターへのアクセス中に EXCI コードが異常終了したことです。EXCI トレースを使用して、いずれかのパラメーターがこのエラーを引き起こした可能性がないかを判別します。その可能性がない場合、これは、外部 CICS インターフェースのエラーである可能性があります。ダンプを保持して、IBM サポート・センターに連絡してください。

624: SERVER_TIMEOUT

DPL 要求が発行され、ターゲット・サーバー・プログラムが CICS サーバー領域内で実行されました。しかし、サーバー・プログラムは、DFHXCOPT テーブルで指定されたタイムアウト値より長い時間実行されています。

システム・アクション

外部 CICS インターフェースは、サーバー・プログラムが完了するまで待つのを停止します。サーバー・プログラムがタイムアウトを過ぎてから完了し、DPL 呼び出しに応答しようとする可能性があるため、パイプは強制的に「クローズが必要」状態にされます。

ユーザー応答

サーバー・アプリケーション・プログラムがタイムアウトになった理由を判別します。サーバー・プログラム自体に問題がある (例えば、ループしている可能性があります) か、あるいはタイムアウト値が小さすぎる可能性があります。

625: STIMER_SETUP_FAILURE

TIMEOUT メカニズムを提供するために、外部 CICS インターフェースは、MVS STIMERM マクロ呼び出しを発行します。この呼び出しは失敗しました。

システム・アクション

呼び出しからの戻りコードが EXCI 戻り域の副理由フィールド 1 に返されます。DPL 要求は終了し、外部 CICS インターフェースは、システム・ダンプを取得します。パイプは、「クローズが必要」状態になります。

ユーザー応答

MVS 戻りコードおよびダンプを使用して、呼び出しが失敗した理由を判別します。これは、外部 CICS インターフェースのエラーである可能性があります。IBM サポート・センターに連絡して、ダンプの詳細を提供してください。

注：このエラーの後、パイプは「クローズが必要」状態であるため、さらに呼び出しを試行する場合は、その前にパイプを閉じてから再度開く必要があります。

626: STIMER_CANCEL_FAILURE

DPL 要求が正常に完了したときに、TIMEOUT 値を検査するために発行された STIMERM 要求の取り消しがエラーで失敗しました。

システム・アクション

STIMERM CANCEL からの戻りコードが EXCI 戻り域の副理由フィールド 1 に返されます。パイプは「クローズが必要」状態になり、外部 CICS インターフェースはシステム・ダンプを取得します。

ユーザー応答

戻りコードおよびダンプを使用して、MVS STIMERM CANCEL コマンドが失敗した理由を判別します。これは、外部 CICS インターフェースのエラーである可能性があります。IBM サポート・センターに連絡して、ダンプの詳細を提供してください。

注：このエラーの後、パイプは「クローズが必要」状態であるため、さらに呼び出しを試行する場合は、その前にパイプを閉じてから再度開く必要があります。

627: INCORRECT_SVC_LEVEL

CICS SVC (DFHCSVC) のリリース・レベルが、外部 CICS インターフェースのリリース・レベルと同じ (またはそれ以上) ではありません。

システム・アクション

Initialize_User 要求は終了します。このエラーは、外部 CICS インターフェースの SDUMP 機能が初期化される前に発生するため、EXCI は、MVS 異常終了 (U0407) を発行して強制的に SYSMDUMP を行います。

ユーザー応答

使用されている CICS SVC のレベルを判別し、それが外部 CICS インターフェースと同じリリース・レベル (またはそれ以上) であることを確認します。SVC 番号をデフォルト (DFHXCOPT の CICSSVC=0) にすることを許可している場合、使用される SVC 番号は、MVS イメージ上の CICS 領域が最初に使用した SVC です。つまり、最初の CICS 領域が CICS 領域間通信 (IRC) を開くために使用した SVC です。SVC 番号が DFHXCOPT の CICSSVC で指定されている場合は、指定された SVC 番号が誤ったレベルです。詳しくは、[EXCI オプション・テーブル DFHXCOPT の使用の CICSSVC パラメーターの説明](#)を参照してください。

628: IRP_LEVEL_CHECK_FAILURE

モジュール DFHIRP のリリース・レベルが、外部 CICS インターフェースのリリース・レベル以上ではありません。

システム・アクション

Allocate_pipe 要求は終了します。IRP 戻りコード (R15) が EXCI 副理由フィールド 1 に返され、使用されている DFHIRP の機能レベルが EXCI 副理由フィールド 2 に返されます。副理由フィールド 2 は、副理由フィールド 1 がゼロである場合にのみ意味があります。外部 CICS インターフェースは、システム・ダンプを取得します。

ユーザー応答

LPA にインストールされている DFHIRP モジュールのレベルを確認します。そのレベルが、少なくとも外部 CICS インターフェースと同じでなければなりません。インストールされる DFHIRP のレベルは、CICS あるいは MVS イメージ内で使用されている外部 CICS インターフェースの最高レベルでなければなりません。DFHIRP のインストールについて詳しくは、[『インストール』の『DFHIRP および DFHCSVC モジュールへの LPA のインストール』](#)を参照してください。

629: SERVER_PROTOCOL_ERROR

DPL 要求に対する応答が CICS によって返されましたが、外部 CICS インターフェースがその応答を理解できません。

システム・アクション

DPL 要求は終了し、外部 CICS インターフェースは、システム・ダンプを取得します。

ユーザー応答

ダンプを使用して、応答がエラーになった理由を判別します。この理由である可能性が最も高いのは、CICS アプリケーション・サーバー・プログラムが、CICS ミラー・タスクの制御下で実行されていなかったことです。これは、DPL 呼び出しで transid パラメーターによって指定されたトランザクション定義名が、プログラム名として DFHMIRS を指定していない場合に発生する可能性があります。これにより、認識できない応答が CICS サーバー領域から送信されます。

630: RRMS_ERROR

DPL_Request の処理中に、リカバリー可能リソース管理サービス (RRMS) から予期しない戻りコードを受信しました。

システム・アクション

DPL_Request 処理は終了します。

戻り域の副理由フィールド 1 の値は、どの RRMS インターフェースが予期しない戻りコードを返したかを示します。

- 1
CTXRCC
- 2
ATTRURD
- 3
CTXSDTA

RRMS 要求からの戻りコードは、副理由フィールド 2 に返されます。

外部 CICS インターフェースは、メッセージ DFHEX0002 を発行し、システム・ダンプを取得します。

ユーザー応答

RRMS 要求からの戻りコードとダンプを使用して、要求が失敗した理由を判別します。これは、内部 EXCI エラーである可能性、あるいは RRMS の問題である可能性があるため、IBM サポート・センターの支援が必要になる場合があります。

631: RRMS_SEVERE_ERROR

DPL_Request の処理中に、リカバリー可能リソース管理サービス (RRMS) 対応のインターフェースを使用している際に EXCI コードで予期しないエラーが発生しました。

システム・アクション

DPL_Request 処理は終了します。

外部 CICS インターフェースは、メッセージ DFHEX0002 を発行し、システム・ダンプを取得します。

ユーザー応答

ダンプを使用して、要求が失敗した理由を判別します。これは、内部 EXCI エラーである可能性があるため、IBM サポート・センターの支援が必要になる場合があります。

632: XCGUR_GETMAIN_ERROR

DPL_Request の処理中に、モジュール DFHXCGUR の作業用ストレージに対する要求が失敗しました。

システム・アクション

DPL_Request 処理は終了します。

GETMAIN からの戻りコードが戻り域の副理由フィールド 1 に返されます。外部 CICS インターフェースがメッセージ DFHEX0003 を発行し、システム・ダンプを取得します。

ユーザー応答

GETMAIN からの戻りコードとダンプを使用して、GETMAIN が失敗した理由を判別します。理由として考えられるのは、ジョブの領域サイズが小さすぎることです。その場合は、領域サイズを増やして、クライアント・アプリケーション・プログラムを再度開始してください。

633: INQUIRE_CHANNEL_FAILED

DPL_Request 処理中に、チャンネル・トークンを取得する INQUIRE_CHANNEL 要求が失敗しました。

システム・アクション

DPL_Request 処理は終了します。

外部 CICS インターフェースがメッセージ DFHEX0002 を発行し、システム・ダンプを取得します。

ユーザー応答

これは、外部 CICS インターフェースのエラーです。IBM サポート・センターに連絡して、戻りコードとダンプの詳細を提供してください。

第 7 章 EXCI サンプル: チャネルおよびコンテナのサンプル・アプリケーション

EXCI CALL インターフェースと **EXEC CICS LINK** コマンドの両方を使用するクライアント・アプリケーションをコーディングする方法を示すために、CICS は、サンプル MVS クライアント・プログラムとサンプル CICS サーバー・プログラムを提供します。EXCI チャネルおよびコンテナ・サンプル・アプリケーションのセットは、チャネルおよびコンテナを使用した CICS とのデータの受け渡しを示します。

EXCI チャネルおよびコンテナ・サンプル・アプリケーションについて

外部 CICS インターフェース・サンプル・プログラムには、サンプル MVS クライアント・プログラムとサンプル CICS サーバー・プログラムの 2 つが含まれています。データは、チャネルおよびコンテナを使用して、クライアントとサーバー・プログラムの間で受け渡されます。

表 8. 外部 CICS インターフェースのサンプル・プログラム		
言語	名前	プログラムのタイプ
アセンブラ ¹	DFH\$AXNC	クライアント・プログラム
COBOL ²	DFH0CXNC	クライアント・プログラム

注:

1. アセンブラ言語のプログラムは、ソース形式と実行可能形式で提供されます。
2. COBOL プログラムは、ソース形式でのみ提供されます。

サンプル CICS サーバー・プログラム

サンプル CICS サーバー・プログラム DFH\$AXNS は、アセンブラでのみ提供されており、ソース形式と実行可能形式です。

サンプル MVS クライアント・プログラム

サンプル・クライアント・プログラムは、EXCI CALL インターフェースおよび **EXEC CICS LINK** コマンドを使用して、単純な MVS クライアント・アプリケーションをコーディングする方法を示します。クライアント・アセンブラと COBOL サンプル・プログラムの内部設計は両方とも同じです。

注: アセンブラ・バージョンのクライアント・プログラムでは BSAM を使用します。これには、BSAM 呼び出しの前後に AMODE(24) への切り替えが行われるため、プログラムを RMODE(24) でリンク・エディットする必要があります。アセンブラ・ソース・コードに、必要な RMODE(24) ステートメントが組み込まれています。通常、EXCI クライアント・プログラムは AMODE(31),RMODE(ANY) で実行します。したがって、アセンブラ・バージョンのクライアント・プログラムは、Language Environment の MAIN プログラムとして使用するには不適切です。

以下のように、各バージョンのクライアントは、3 つの別個のセクションに分かれています。

セクション 1

セクション 1 は、EXEC インターフェースを使用して、コンテナをサーバー (CICS) に送信します。要求は、REQUEST_TYPE というコンテナ内にあり、「LINK1」または「LINK2」のいずれかが含まれています。以下で説明するサーバー・プログラム (DFH\$AXNS) は、このコンテナを使用します。最初の LINK 要求では、EBCDIC_DATA というコンテナが、単純テキスト・ストリングを使用してチャネル FIRST_CHANNEL でセットアップされます。これは、その後、**EXEC CICS LINK** 要求によって送信されます。

この要求が正常に実行されると、**EXEC CICS QUERY CHANNEL** 要求が実行されて、チャネルのコンテナの数が検査されます。この数は 3 になるはずですが、サーバー・プログラムがチャネルに EXCI_RESPONSE コンテナを追加するはずだからです。その後、**EXEC CICS STARTBROWSE**

CONTAINER、EXEC CICS GETNEXT CONTAINER、EXEC CICS ENDBROWSE CONTAINER 要求を使用してブラウズが実行され、チャンネル上のコンテナの名前がブラウズされます。この照会とブラウズは必須ではありませんが、EXCI SPI コマンドの使用法を示すために追加されています。

セクション 1 の最終部分では、いくつかのコンテナ・コマンドが実行されて、チャンネル **SECOND_CHANNEL** でコンテナ **ASCII_DATA** がセットアップされます。これは、**EXEC CICS LINK** 要求によって再度送信されます。

セクション 2

セクション 2 は **CALL** インターフェースを使用します。**INIT_USER、ALLOCATE_PIPE、および OPEN_PIPE** が実行された後、DPL セクションは、2 回実行されるループに入ります。これは、セクション 1 で送信されたコンテナの送信を繰り返しますが、代わりに **CALL** インターフェースを使用します。

セクション 2 では、**REQUEST_TYPE** コンテナには、「**CALL1**」や「**CALL2**」ではなく、「**CALL1**」または「**CALL2**」が含まれます。その他のコンテナは、セクション 1 に送信されます。

セクション 3

セクション 3 は、使用されたチャンネルを削除します。これは必須ではありませんが、ベスト・プラクティスを示すために行われます。次に、**CLOSE_PIPE** および **DEALLOCATE_PIPE** が実行されます。

サーバー・プログラム **DFHEAXNS** (アセンブラ) は、**DFHEAXNC** と **DFHOCXNC** の両方に呼び出されます。これは、**REQUEST_TYPE** コンテナの内容を使用して、今回が **EXEC** インターフェースまたは **CALL** インターフェースのどちらかによる最初の呼び出しなのか 2 回目の呼び出しなのかを確認します。コンテナを検出できない場合、プログラムは、**REQUEST_TYPE** コンテナを検出できなかったことを示す異常終了コード **NCON** で異常終了します。これが正常に実行された場合、サーバー・プログラムは指示された要求の適切なコンテナを取得し、応答をコンテナ **EXCI_RESPONSE** に入れて戻します。これは、応答にテキスト '**OK**' が含まれていることを確認するために、クライアント・プログラムによってテストされます。プログラムは、**DFHAXNSQ** という **TS** メイン・キューにレコードを書き込みます。これにより、プログラムの実行が確認でき、すべてが適切に機能したことが示されます。

アセンブラ・バージョンのクライアント・プログラムは、実行可能形式で事前生成されて提供されます。どちらのバージョンのプログラムも、以下のように 2 つのランタイム・パラメーターを受け入れます。

TARGET_SYSTEM

サーバー領域 **APPLID** を指定します。

事前生成アセンブラ・バージョンを使用する場合、独自の **CICS** サーバー領域の **APPLID** を指定するのにプログラムを再アセンブルする必要はありません。また、サンプル・クライアント・プログラムと異なる **CICS** 領域を併用することもでき、その際に毎回プログラムを変更する必要はありません。

USERID

呼び出しインターフェース **DPL_request** で使用されるユーザー ID を指定します。

これらの定位置パラメーターは、**PARM** ステートメント上でコンマで区切って指定します。

EXCI チャンネルおよびコンテナのサンプル・プログラムのセットアップ

サンプルの外部 **CICS** インターフェース・プログラムが、**CICS Transaction Server for z/OS** 配布テープに含まれています。**EXCI** サンプル・プログラムをサポートするリソース定義は、グループ **DFH\$EXCI** の **CICS** システム定義ファイル (**CSD**) に含まれています。

このタスクについて

[123 ページの表 8](#) に示されているサンプル・プログラムは、**CICSTS56.CICS.SDFHSAMP** 内にソース形式で提供されています。サンプルのアセンブラ・サーバー・プログラムは、**CICSTS56.CICS.SDFHLOAD** 内に実行可能形式でも提供されています。アセンブラ・クライアント・プログラムは、**CICSTS56.CICS.SDFHEXCI** 内に提供されています。

注: **EXCI** サンプル・プログラムのリソース定義は、**CSD** に含まれていますが、**IBM** 定義のグループ・リスト **DFHLIST** には含まれていません。したがって、**GRPLIST=DFHLIST** を指定して **CICS** を初期設定した場合、サンプルを使用するには、その前に **EXCI** リソース定義グループをインストールする必要があります。

または、システムの初期設定時にサンプル・グループを自動的にインストールされるように、始動グループ・リストにサンプル・グループを追加することもできます。

手順

1. 以下のリソース定義グループをインストールします。

DFH\$EXCI

サンプルのサーバー・トランザクション、サーバー・プログラム、EXCI 接続、およびセッションの定義が含まれています。

DFH\$AXNS と呼ばれる、アセンブラ言語のサーバー・プログラムが 1 つだけ組み込まれています。

このサンプル・アプリケーションは、トランザクション EXCI を実行するように設計されています。この EXCI は、DFHMIRS ミラー・プログラムを呼び出すように定義されており、プロファイル DFHCICSA を参照します。EXCI 用の必要なトランザクション定義は、このグループ内に組み込まれています。

固有接続と汎用接続用のサンプルの CONNECTION 定義と SESSIONS 定義が組み込まれています。

注: サンプル・グループ DFH\$EXCI 内で提供されている汎用接続と固有接続の両方の定義で、ATTACHSEC(IDENTIFY) が指定されています。RACF または同等の外部セキュリティ・マネージャー (ESM) がインストールされてアクティブになっていない環境でサンプル・プログラムを実行すると、このセキュリティ・オプションが原因で、サーバー・プログラム DFH\$AXNS は ATCY 異常終了で失敗します。

セキュリティをアクティブにしないで外部 CICS インターフェース・サンプル・プログラムを実行する場合は、接続リソース定義を変更して、ATTACHSEC(LOCAL) を指定しなければなりません。

2. バッチ・プログラムからリンクされるトランザクションの場合、トランザクション定義に、プログラム名としてミラー・プログラム DFHMIRS を指定します。
3. 領域間通信 (IRC) がオープンであることを確認します。
CICS の初期設定中に IRC がオープンされない場合は、**CEMT SET IRC OPEN** コマンドを使用してオープンに設定します。
4. COBOL バージョンの EXCI クライアント・プログラムを使用する場合、DFHZXTCL プロシージャまたは DFHYXTVL プロシージャを使用して、プログラムを変換およびコンパイルし、適切なライブラリーにリンク・エディットします。

EXCI チャネルおよびコンテナのサンプル・アプリケーションの実行

クライアント・プログラムを実行するためのバッチ・ジョブを作成することができます。また、事前生成アセンブラ・バージョンを使用してクライアント・プログラムを実行することもできます。

始める前に

1. COBOL バージョンの EXCI クライアント・プログラムを使用するには、DFHZXTCL プロシージャまたは DFHYXTVL プロシージャを使用して、プログラムを変換およびコンパイルし、適切なライブラリーにリンク・エディットする必要があります。
2. EXCI サンプル・プログラムのリソース定義は、CSD に含まれていますが、IBM 定義のグループ・リスト DFHLIST には含まれていません。したがって、GRPLIST=DFHLIST を指定して CICS を初期設定した場合、サンプルを使用するには、その前に EXCI リソース定義グループをインストールする必要があります。または、システムの初期設定時にサンプル・グループを自動的にインストールされるように、始動グループ・リストにサンプル・グループを追加することもできます。

手順

- 以下のサンプル JCL に基づいて、クライアント・プログラムを実行するためのバッチ・ジョブを作成します。

```
//EXCI      JOB (accounting_information),CLASS=A,TIME=1440,
//          USER=userid,PASSWORD=pswd,REGION=100M
//*****
//*          JCL to execute an external CICS interface client program *
//*****
//          EXEC  PGM=pgmname,REGION=nnM,MEMLIMIT=nnG
//STEPLIB DD   DSN=CICSTS54.CICS.EXCI.LOADLIB,DISP=SHR
//          DD   DSN=CICSTS54.CICS.SDFHEXCI,DISP=SHR
//SYSPRINT DD   SYSOUT=A
//SYSMDUMP DD   DSN=SYS1.SYSMDP00,VOL=SER=volid,SPACE=(CYL,(1,1)),
//          DISP=OLD,UNIT=3390
```

図 9. EXCI クライアント・プログラムを開始するためのサンプル・ジョブ

- クライアント・プログラムの事前生成アセンブラー・バージョンを使用する場合は、クライアント・プログラム用の以下の EXEC ステートメントを発行します。

```
//*****
//ASM      EXEC  PGM=client_program_name,PARM='applid,userid',REGION=0M,MEMLIMIT=1G
```

ここで、

client_program_name

クライアント・プログラムの名前 (例えば、DFH\$AXNC) を指定します。

applid (アプリケーション ID)

ターゲット CICS サーバー領域の APPLID を指定します。

注: *applid* を省略する場合、ユーザー ID の前のコンマは残しておかなければなりません。

userid

DPL_request 呼び出し用のユーザー ID を指定します。

タスクの結果

126 ページの図 10 は、クライアント・プログラムの事前生成アセンブラー・バージョン DFH\$AXNC が正常に実行された場合の DFH\$AXNC の出力例を以下に示します。

```
***** EXCI Sample Batch Client Program *****
*
*      Parameters: APPLID=IYK2Z2G1 .....
*
* EXEC Level Processor.
*   Setting up the EXEC level call.
*   The Link Request with channel FIRST_CHANNEL has completed successfully.
*   Checking response container sent by server.
*   Response OK, continue processing.
*   Query channel command completed successfully.
*   Correct number of containers returned.
*   Browse of channel names completed successfully.
*   The Link Request with channel SECOND_CHANNEL has completed successfully.
*   Checking response container sent by server.
*   Response OK, continue processing.
*
* CALL Level Processor.
*   Initialise_User call complete.
*   Allocate_Pipe call complete.
*   Open_Pipe call complete.
*   The connection has been successful.
*   Container EBCDIC_DATA was received correctly in channel FIRST_CHANNEL.
*   Container ASCII_DATA was received correctly in channel SECOND_CHANNEL.
*   Channels have been deleted.
*   Close_Pipe call complete.
*   Deallocate_Pipe call complete.
*
***** End of EXCI Sample Batch Client Program *****
```

図 10. 正常に実行された DFH\$AXNC の出力例

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。この資料の他の言語版を IBM から入手できる場合があります。ただし、これを入手するには、本製品または当該言語版製品を所有している必要がある場合があります。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。IBM 製品、プログラムまたはサービスに代えて、IBM の知的所有権を侵害することのない機能的に同等のプログラムまたは製品を使用することができません。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒 103-8510

東京都中央区日本橋箱崎町 19 番 21 号

日本アイ・ビー・エム株式会社

法務・知的財産

知的財産権ライセンス涉外

IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様自身の責任でご使用ください。

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

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Director of Licensing

IBM Corporation

North Castle Drive, MD-NC119 Armonk,

NY 10504-1785

United States of America

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関す

る実行性、互換性、またはその他の要求については確証できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名前はすべて架空のものであり、類似する個人や企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほめかしたり、保証することはできません。これらのサンプル・プログラムは特定物として現存するままの状態を提供されるものであり、いかなる保証も提供されません。IBM は、お客様の当該サンプル・プログラムの使用から生ずるいかなる損害に対しても一切の責任を負いません。

プログラミング・インターフェース情報

CICS には、プログラミング・インターフェースと見なすことのできる資料と、プログラミング・インターフェースと見なすことのできない資料があります。

オンライン製品資料の以下のセクションには、CICS Transaction Server for z/OS, バージョン 5 リリース 6 のサービスを取得するプログラムをお客様が作成するためのプログラミング・インターフェースが含まれています。

- [アプリケーションの開発](#)
- [システム・プログラムの開発](#)
- [CICS TS セキュリティー](#)
- [外部インターフェースに向けた開発](#)
- [アプリケーション開発のリファレンス](#)
- [リファレンス: システム・プログラミング](#)
- [リファレンス: 接続](#)

オンライン製品資料の以下のセクションには、CICS Transaction Server for z/OS, バージョン 5 リリース 6 のプログラミング・インターフェースとして意図されていない (プログラミング・インターフェースと誤解される可能性のある) 情報が含まれています。

- [トラブルシューティングおよびサポート](#)
- [CICS TS 診断参照](#)

PDF 形式のマニュアルで CICS 資料にアクセスする場合は、CICS Transaction Server for z/OS, バージョン 5 リリース 6 のサービスを取得するプログラムをお客様が作成するためのプログラミング・インターフェースが以下のマニュアルに含まれています。

- [アプリケーション・プログラミング・ガイドおよびアプリケーション・プログラミング・リファレンス](#)
- [Business Transaction Services](#)
- [Customization Guide](#)
- [C++ OO Class Libraries](#)
- [Debugging Tools Interfaces Reference](#)
- [Distributed Transaction Programming Guide](#)
- [External Interfaces Guide](#)
- [Front End Programming Interface Guide](#)

- IMS Database Control Guide
- インストール・ガイド
- セキュリティー・ガイド
- Supplied Transactions
- CICSplex SM Managing Workloads
- CICSplex SM Managing Resource Usage
- CICSplex SM アプリケーション・プログラミング・ガイドおよび CICSplex SM アプリケーション・プログラミング・リファレンス
- CICS における Java™ アプリケーション

PDF 形式のマニュアルで CICS 資料にアクセスする場合は、CICS Transaction Server for z/OS, バージョン 5 リリース 6 のプログラミング・インターフェースとして意図されていない (プログラミング・インターフェースと誤解される可能性のある) 情報が以下のマニュアルに含まれています。

- Data Areas
- Diagnosis Reference
- Problem Determination Guide
- CICSplex SM Problem Determination Guide

商標

IBM、IBM ロゴおよび ibm.com® は、世界の多くの国で登録された International Business Machines Corporation の商標または登録商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。

Adobe、Adobe ロゴ、PostScript、PostScript ロゴは、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。

インテル、Intel、Intel ロゴ、Intel Inside、Intel Inside ロゴ、Intel Centrino、Intel Centrino ロゴ、Celeron、Intel Xeon、Intel SpeedStep、Itanium、および Pentium は、Intel Corporation または子会社の米国およびその他の国における商標または登録商標です。

Java およびすべての Java 関連の商標およびロゴは Oracle やその関連会社の米国およびその他の国における商標または登録商標です。

Linux® は、Linus Torvalds の米国およびその他の国における登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

UNIX は The Open Group の米国およびその他の国における登録商標です。

製品資料に関するご使用条件

これらの資料は、以下のご使用条件に同意していただける場合に限りご使用いただけます。

適用範囲

IBM Web サイトの「ご利用条件」に加えて、以下のご使用条件が適用されます。

個人使用

これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、非商業的な個人による使用目的に限り複製することができます。ただし、IBM の明示的な承諾をえずに、これらの資料またはその一部について、二次的著作物を作成したり、配布 (頒布、送信を含む) または表示 (上映を含む) することはできません。

商用使用

これらの資料は、すべての著作権表示その他の所有権表示をしていただくことを条件に、お客様の企業内に限り、複製、配布、および表示することができます。ただし、IBM の明示的な承諾をえずにこれらの資料の二次的著作物を作成したり、お客様の企業外で資料またはその一部を複製、配布、または表示することはできません。

権利

ここで明示的に許可されているもの以外に、資料や資料内に含まれる情報、データ、ソフトウェア、またはその他の知的所有権に対するいかなる許可、ライセンス、または権利を明示的にも黙示的にも付与するものではありません。

資料の使用が IBM の利益を損なうと判断された場合や、上記の条件が適切に守られていないと判断された場合、IBM はいつでも自らの判断により、ここで与えた許可を撤回できるものとさせていただきます。

お客様がこの情報をダウンロード、輸出、または再輸出する際には、米国のすべての輸出入関連法規を含む、すべての関連法規を遵守するものとします。

IBM これらの資料の内容についていかなる保証もしません。これらの資料は、特定物として現存するままの状態を提供され、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任なしで提供されます。

IBM オンラインでのプライバシー・ステートメント

サービス・ソリューションとしてのソフトウェアも含めた IBM ソフトウェア製品 (ソフトウェア・オファリング) では、製品の使用に関する情報の収集、エンド・ユーザーの使用感の向上、エンド・ユーザーとの対話またはその他の目的のために、Cookie はじめさまざまなテクノロジーを使用することがあります。多くの場合、ソフトウェア・オファリングにより個人情報が収集されることはありません。IBM の「ソフトウェア・オファリング」の一部には、個人情報を収集できる機能を持つものがあります。ご使用の「ソフトウェア・オファリング」が、これらの Cookie およびそれに類するテクノロジーを通じてお客様による個人情報の収集を可能にする場合、以下の具体的事項をご確認ください。

CICSplex SM Web ユーザー・インターフェース (メイン・インターフェース) の場合:

このソフトウェア・オファリングは、展開される構成に応じて、セッション管理、認証、お客様の利便性の向上、または利用の追跡または機能上の目的のために、それぞれのお客様のユーザー名、およびその他の個人情報を、セッションごとの Cookie および持続的な Cookie を使用して収集する場合があります。これらの Cookie を無効にすることはできません。

CICSplex SM Web ユーザー・インターフェース (データ・インターフェース) の場合:

このソフトウェア・オファリングは、展開される構成に応じて、セッション管理、認証、または利用の追跡または機能上の目的のために、それぞれのお客様のユーザー名またはその他の個人情報を、セッションごとの Cookie を使用して収集する場合があります。これらの Cookie を無効にすることはできません。

CICSplex SM Web ユーザー・インターフェース (「Hello World」ページ) の場合:

このソフトウェア・オファリングは、展開される構成に応じて、個人情報を収集しないセッションごとの Cookie を使用する場合があります。これらの Cookie を無効にすることはできません。

CICS Explorer® の場合:

このソフトウェア・オファリングは、展開される構成に応じて、セッション管理、お客様の利便性の向上、または利用の追跡または機能上の目的のために、それぞれのお客様のユーザー名、およびその他の個人情報を、セッションごとの設定および持続的な設定を使用して収集する場合があります。これらの設定を無効にすることはできませんが、ユーザー・パスワードの暗号化形式でのディスクへの保管は、サインオン中にチェック・ボックスにチェック・マークを付けることによるユーザーの明示的な操作によってのみ有効化することができます。

この「ソフトウェア・オファリング」が Cookie およびさまざまなテクノロジーを使用してエンド・ユーザーから個人を特定できる情報を収集する機能を提供する場合、お客様は、このような情報を収集するにあたって適用される法律、ガイドライン等を遵守する必要があります。これには、エンドユーザーへの通知や同意の要求も含まれますがそれらには限られません。

このような目的での Cookie を含む様々なテクノロジーの使用の詳細については、『IBM オンラインでのプライバシー・ステートメント』 (<http://www.ibm.com/privacy/details/jp/ja/>) の『クッキー、ウェブ・ビー

コン、その他のテクノロジー』および『IBM Software Products and Software-as-a-Service Privacy Statement』 (<http://www.ibm.com/software/info/product-privacy>) を参照してください。

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。
なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

アセンブラー、C 言語、COBOL、PL/I のコピーブック [33](#)
アプリケーション ID、ALLOCATE_PIPE コマンドでの指定 [17](#)
大きな COMMAREA、チャンネル [43-46](#), [50](#), [51](#), [53](#), [57](#)
オープン・システム・インターフェース (OSI) [1](#)

[カ行]

外部 CICS インターフェース (EXCI)
CALL インターフェース
構文規則 [12](#)
戻り域 [33](#)
EXEC CICS か CALL インターフェースかの選択 [4](#)
応答コード [97](#)
クライアント・プログラムで同期点を取る [11](#)
クライアント・プログラムのコンパイルとリンク・エディット [58](#)
サポートされている言語 [12](#)
サポートされているプログラミング言語 [12](#)
セキュリティ [77](#)
接続の状態の照会 [67](#)
接続の定義 [64](#)
説明 [3](#)
問題判別 [81](#)
ユーザー置換可能モジュール (DFHXCURM) [68](#)
リソースおよびリカバリー [8](#)
利点 [1](#)
理由コード [97](#)
CALL インターフェース (CALL interface)
DFHAPPL プロファイル定義 [78](#)
PL/I および C 言語に関する考慮事項 [61](#)
RRMS の使用 [9](#)
外部 CICS インターフェースの利点 [1](#)
関数呼び出し EQUATE のコピーブック [33](#)

[サ行]

サーバー・プログラムのプログラミングの制約事項 [4](#)
サービス・トラップ [82](#)
作業単位 ID、DPL_Request [23](#)
サンプル・プログラム [35](#)
システム間複数領域操作 (XCF/MRO) [3](#)
シスプレックス、システム間 MRO の使用 [3](#)
ストレージの解放 [31](#)
静的ルーティング
EXCI [63](#)
セキュリティ [77](#)

[タ行]

ダンプ
フォーマット設定 [82](#)

ダンプ (続き)
SYSMDUMP [82](#)
ダンプのキャプチャー [82](#)
チャンネル、大きな COMMAREA としての [43-46](#), [50](#), [51](#), [53](#), [57](#)
動的ルーティング
EXCI [63](#)
特定の接続
その定義 [64](#)
MRO ログオン・セキュリティ検査 [77](#)
閉じたパイプの再使用 [29](#)
トラップ、DFHXCTRA
TRAP、DFHXCPT のパラメーター [74](#)
トレース
トレース・ポイント [83](#)
DFHXCPT の TRACE パラメーター [74](#)
DFHXCPT の TRACESZ パラメーター [74](#)
トレース内のユーザー・データの抑止
CONFDATA オプション [72](#)

[ナ行]

ヌル・パラメーターを持つ EXCI CALL の例 [35](#)

[ハ行]

パイプに関連付けられているストレージの解放 [31](#)
パイプのオープン [18](#)
パイプのクローズ [29](#)
パイプの切断 [29](#)
パイプの割り振り [16](#)
パイプの割り振り解除 [31](#)
バインド時のセキュリティ (bind-time security) [77](#)
バッチ・ジョブ、状態の照会 [68](#)
汎用接続
セキュリティ検査を実行しない場合の注意事項 [77](#)
その定義 [65](#)

[マ行]

問題判別
サービス・トラップ [82](#)
ダンプ [81](#)
トレース [81](#)

[ヤ行]

ユーザー環境、初期化 [12](#)
ユーザー・セキュリティ [78](#)
ユーザー置換可能モジュール
DFHXCURM [68](#)

[ラ行]

ラージ COMMAREA、チャンネル [43-46](#), [50](#), [51](#), [53](#), [57](#)
リソース定義

リソース定義 (続き)
CONNECTION 定義 [64](#)
SESSIONS 定義 [65](#)
ログオン・セキュリティ [77](#)

[ワ行]

割り振られたパイプの接続 [18](#)

A

allocate_opts、ALLOCATE_PIPE コマンドのパラメーター [17](#)
Allocate_Pipe コマンド [16](#)
ALLOCATE_PIPE コマンド
処理中の DFHXCURM の呼び出し [68](#)
セキュリティ検査の失敗 [77](#)

B

BROWSETOKEN オプション
ENDBROWSE CONTAINER (CHANNEL) コマンド [45](#)
GETNEXT CONTAINER (EXCI) コマンド [50](#)
STARTBROWSE CONTAINER コマンド [58](#)

C

C バージョン
DFHZXTDL [60](#)
C++
CICS 提供プロシージャ、DFHZXTEL [61](#)
ccsid、DPL_Request コマンドのパラメーター [26](#)
CEMT INQUIRE EXCI コマンド [68](#)
CEMT INQUIRE コマンドの EXCI [68](#)
CICS システム 初期設定パラメーター
IRCSTRT [99](#)
CICS_applid、ALLOCATE_PIPE コマンドのパラメーター [17](#)
CICSSVC、DFHXCLOPT のパラメーター [71](#)
Close_Pipe コマンド [29](#)
COMMAREA_len、DPL_Request コマンドのパラメーター [21](#)
COMMAREA、DPL_Request コマンドのパラメーター [21](#)
CONFDATA、DFHXCLOPT のパラメーター [72](#)
CONNECTION 定義
CONNTYPE 属性 [64](#)
PROTOCOL 属性 [65](#)
CONNTYPE 属性、CONNECTION 定義 [64](#)
CONTAINERCNT オプション
QUERY CHANNEL コマンド [57](#)

D

data_len、DPL_Request コマンドのパラメーター [22](#)
Deallocate_Pipe コマンド [31](#)
DELETE CHANNEL (EXCI) コマンド [43](#)
DELETE CONTAINER (EXCI) コマンド [44](#)
DFHAXCC、アセンブラー・サンプル・プログラム [35](#)
DFHDXCC、サンプル・プログラム [35](#)
DFHPXCC、PL/I サンプル・プログラム [35](#)
DFHAPPL FACILITY クラス・プロファイル、定義 [77](#)
DFHAUPLE プロシージャ [70](#)
DFHEXTAL、アセンブラー・クライアント・プログラム用の
プロシージャ [60](#)
DFHIRP (領域間通信プログラム)
実行されるセキュリティ検査 [77](#)

DFHXCIE、DFHXCSTB スタブの別名 [58](#)
DFHXCIS、DFHXCSTB スタブの別名 [58](#)
DFHXCPLD、アセンブラーの戻り域と EQUATE のコピーブック
[34](#)
DFHXCPLH、C 言語の戻り域と EQUATE のコピーブック [34](#)
DFHXCPLL、PL/I の戻り域と EQUATE のコピーブック [34](#)
DFHXCPLD、COBOL の戻り域と EQUATE のコピーブック [34](#)
DFHXCRCO、アセンブラーの戻りコード・コピーブック [34](#)
DFHXCRCRCH、C 言語の戻りコード・コピーブック [34](#)
DFHXCRCCL、PL/I の戻りコード・コピーブック [34](#)
DFHXCRCO、COBOL の戻りコード・コピーブック [34](#)
DFHXCSTB、クライアント・プログラムのスタブ [58](#)
DFHXCSTRA、EXCI サービス・トラップ [82](#)
DFHXCSTRD、パラメーター・リスト [83](#)
DFHXCURM ユーザー置換可能モジュール [68](#)
DFHYXTDL、C クライアント・プログラム用のプロシージャ
[60](#)
DFHYXTEL、C++ クライアント・プログラム用のプロシージャ
[60](#)
DFHYXTPL、PL/I クライアント・プログラム用のプロシージャ
[60](#)
DFHYXTVL、COBOL クライアント・プログラム用のプロシージャ
[60](#)
DFHZXTCL、COBOL クライアント・プログラム用のプロシージャ
[60](#)
DFHZXTDL、C クライアント・プログラム用のプロシージャ
[60](#)
DFHZXTEL、C++ クライアント・プログラム用のプロシージャ
[61](#)
DFHZXTPL、PL/I クライアント・プログラム用のプロシージャ
[61](#)
DPL_opts、DPL_Request コマンドのパラメーター [25](#)
DPL_Request 呼び出し [20](#)
dpl_retarea、DPL_Request コマンドのパラメーター [25](#)
DURETRY、DFHXCLOPT のパラメーター [72](#)

E

END 状態
GETNEXT CONTAINER (CHANNEL) コマンド [51](#)
ENDBROWSE CONTAINER (EXCI) コマンド [45](#)
endian、DPL_Request コマンドのパラメーター [27](#)
EQUATE コピーブック [33](#)
EXEC CICS LINK コマンドの再試行 [40](#)
EXEC CICS LINK コマンドの変換 [43](#)
EXEC CICS LINK の自動再試行 [40](#)

F

FACILITY クラス・プロファイル、定義 [77](#)

G

GET CONTAINER (EXCI) コマンド [46](#)
GETNEXT CONTAINER (EXCI) コマンド [50](#)
GTF、DFHXCLOPT のパラメーター [73](#)

I

Initialize_User コマンド [12](#)
IRCSTRT システム 初期設定パラメーター [99](#)

L

LINK コマンドで返される例外条件 [38](#)

M

MOVE CONTAINER (EXCI) コマンド [51](#)
MSGCASE、DFHXCOPT のパラメーター [73](#)

O

Open_Pipe コマンド [18](#)
OSI (オープン・システム・インターフェース) [1](#)

P

PROTOCOL 属性
 CONNECTION 定義 [65](#)
 SESSIONS 定義 [66](#)
PUT CONTAINER (EXCI) コマンド [53](#)

Q

QUERY CHANNEL (EXCI) コマンド [57](#)

R

RECEIVECOUNT 属性、SESSIONS 定義 [66](#)
RECEIVEPFX 属性、SESSIONS 定義 [66](#)
RESP および RESP2 フィールド [38](#)

S

SEND COUNT 属性、SESSIONS 定義 [66](#)
SENDPFX 属性、SESSIONS 定義 [67](#)
SESSIONS 定義
 PROTOCOL 属性 [66](#)
 RECEIVECOUNT 属性 [66](#)
 RECEIVEPFX 属性 [66](#)
 SEND COUNT 属性 [66](#)
 SENDPFX 属性 [67](#)
STARTBROWSE CONTAINER (EXCI) コマンド [57](#)
SYSMDUMP、キャプチャー [82](#)

T

TIMEOUT、DFHXCOPT のパラメーター [73](#)
TOKENERR 状態
 ENDBROWSE CONTAINER (CHANNEL) コマンド [46](#)
 GETNEXT CONTAINER (CHANNEL) コマンド [51](#)
transid、DPL_Request コマンドのパラメーター [22](#)
transid2、DPL_Request コマンドのパラメーター [26](#)
TYPE、DFHXCOPT のパラメーター [71](#)

U

uowid、DPL_Request のパラメーター [23](#)
user_name、INITIALIZE_USER コマンドのパラメーター [13](#)
userid、DPL_Request コマンドのパラメーター [24](#)

X

XCFGROUP、DFHXCOPT のパラメーター [74](#)

[特殊文字]

CVDA 値
 NOCONVERT
 GET CONTAINER (EXCI) コマンド [47](#)
CSMI
 CICS サーバーによって接続される [38](#)
CSMI (CICS 提供のミラー・トランザクション)
 セキュリティ [78](#)
 デフォルト transid [23](#)
 リンク・ユーザー ID の許可 [78](#)
APPEND オプション
 PUT CONTAINER (EXCI) コマンド [54](#)
AS オプション
 MOVE CONTAINER (EXCI) コマンド [52](#)
BYTEOFFSET オプション
 GET CONTAINER (EXCI) コマンド [46](#)
call_type
 ALLOCATE_PIPE コマンドのパラメーター [17](#)
 CLOSE_PIPE コマンドのパラメーター [30](#)
 DEALLOCATE_PIPE コマンドのパラメーター [31](#)
 DPL_Request コマンドのパラメーター [21](#)
 INITIALIZE_USER コマンドのパラメーター [13](#)
 OPEN_PIPE コマンドのパラメーター [19](#)
CCSID オプション
 GET CONTAINER (EXCI) コマンド [47](#)
CCSIDERR 状態
 GET CONTAINER (EXCI) コマンド [49](#)
 PUT CONTAINER (EXCI) コマンド [55](#)
CHANNEL オプション
 DELETE CHANNEL コマンド [44](#)
 DELETE CONTAINER (EXCI) コマンド [44](#)
 GET CONTAINER (EXCI) コマンド [47](#)
 MOVE CONTAINER (EXCI) コマンド [52](#)
 PUT CONTAINER (EXCI) コマンド [53](#)
 QUERY CHANNEL コマンド [57](#)
CHANNELERR 状態
 DELETE CHANNEL (EXCI) コマンド [44](#)
 DELETE CONTAINER (EXCI) コマンド [45](#)
 GET CONTAINER (EXCI) コマンド [49](#)
 MOVE CONTAINER (EXCI) コマンド [52](#)
 PUT CONTAINER (EXCI) コマンド [56](#)
 QUERY CHANNEL コマンド [57](#)
CODEPAGEERR 状態
 GET CONTAINER (EXCI) コマンド [49](#)
 PUT CONTAINER (EXCI) コマンド [56](#)
CONTAINER オプション
 DELETE CONTAINER (EXCI) コマンド [45](#)
 GET CONTAINER (EXCI) コマンド [47](#)
 GETNEXT CONTAINER コマンド (EXCI) [50](#)
 MOVE CONTAINER (EXCI) コマンド [52](#)
 PUT CONTAINER (EXCI) コマンド [54](#)
CONTAINERERR 状態
 DELETE CONTAINER (EXCI) コマンド [45](#)
 GET CONTAINER (EXCI) コマンド [49](#)
 MOVE CONTAINER (EXCI) コマンド [53](#)
 PUT CONTAINER (EXCI) コマンド [56](#)
CONVERTST オプション
 GET CONTAINER (EXCI) コマンド [47](#)
DATATYPE オプション

DATATYPE オプション (続き)
 PUT CONTAINER (EXCI) コマンド [54](#)
 FLENGTH オプション
 GET CONTAINER (EXCI) コマンド [47](#)
 PUT CONTAINER (EXCI) コマンド [55](#)
 FROM オプション
 PUT CONTAINER (EXCI) コマンド [55](#)
 FROMCCSID オプション
 PUT CONTAINER (EXCI) コマンド [55](#)
 FROMCODEPAGE オプション
 GET CONTAINER (EXCI) コマンド [55](#)
 INTO オプション
 GET CONTAINER (EXCI) コマンド [48](#)
 INTOCCSID オプション
 GET CONTAINER (EXCI) コマンド [48](#)
 INTOCODEPAGE オプション
 GET CONTAINER (EXCI) コマンド [48](#)
 INVREQ 状態
 DELETE CONTAINER (EXCI) コマンド [45](#)
 GET CONTAINER (EXCI) コマンド [50](#)
 MOVE CONTAINER (EXCI) コマンド [53](#)
 PUT CONTAINER (EXCI) コマンド [56](#)
 LENGERR 状態
 GET CONTAINER (EXCI) コマンド [50](#)
 PUT CONTAINER (EXCI) コマンド [57](#)
 NODATA オプション
 GET CONTAINER (EXCI) コマンド [48](#)
 pgmname
 DPL_Request コマンドのパラメーター [21](#)
 pipe_token
 ALLOCATE_PIPE コマンドのパラメーター [17](#)
 CLOSE_PIPE コマンドのパラメーター [30](#)
 DEALLOCATE_PIPE コマンドのパラメーター [31](#)
 DPL_Request コマンドのパラメーター [21](#)
 OPEN_PIPE コマンドのパラメーター [19](#)
 return_area
 ALLOCATE_PIPE コマンドのパラメーター [17](#)
 CLOSE_PIPE コマンドのパラメーター [30](#)
 DEALLOCATE_PIPE コマンドのパラメーター [31](#)
 DPL_Request コマンドのパラメーター [21](#)
 INITIALIZE_USER コマンドのパラメーター [13](#)
 OPEN_PIPE コマンドのパラメーター [19](#)
 SET オプション
 GET CONTAINER (EXCI) コマンド [48](#)
 TOCHANNEL オプション
 MOVE CONTAINER (EXCI) コマンド [52](#)
 user_token
 ALLOCATE_PIPE コマンドのパラメーター [17](#)
 CLOSE_PIPE コマンドのパラメーター [30](#)
 DEALLOCATE_PIPE コマンドのパラメーター [31](#)
 DPL_Request コマンドのパラメーター [21](#)
 INITIALIZE_USER コマンドのパラメーター [13](#)
 OPEN_PIPE コマンドのパラメーター [19](#)
 version_number
 ALLOCATE_PIPE コマンドのパラメーター [16](#)
 CLOSE_PIPE コマンドのパラメーター [29](#)
 DEALLOCATE_PIPE コマンドのパラメーター [31](#)
 DPL_Request コマンドのパラメーター [21](#)
 INITIALIZE_USER コマンドのパラメーター [13](#)
 OPEN_PIPE コマンドのパラメーター [19](#)
 リソース・アクセス制御機能 (RACF)
 DPL_Request コマンドでのユーザー ID の指定 [24](#)
 > 32K COMMAREA (チャンネル)
 DELETE CHANNEL (EXCI) コマンド [43](#)

> 32K COMMAREA (チャンネル) (続き)
 DELETE CONTAINER (EXCI) コマンド [44](#)
 ENDBROWSE CONTAINER (EXCI) コマンド [45](#)
 GET CONTAINER (EXCI) コマンド [46](#)
 GETNEXT CONTAINER (EXCI) コマンド [50](#)
 MOVE CONTAINER (EXCI) コマンド [51](#)
 PUT CONTAINER (EXCI) コマンド [53](#)
 QUERY CHANNEL (EXCI) コマンド [57](#)
 STARTBROWSE CONTAINER (EXCI) コマンド [57](#)
 2 フェーズ・コミット
 DPL_Request [9](#)
 RRS によって呼び出されるプロトコル [9](#)
 32K COMMAREA より大きい (チャンネル)
 DELETE CHANNEL (EXCI) コマンド [43](#)
 DELETE CONTAINER (EXCI) コマンド [44](#)
 ENDBROWSE CONTAINER (EXCI) コマンド [45](#)
 GET CONTAINER (EXCI) コマンド [46](#)
 GETNEXT CONTAINER (EXCI) コマンド [50](#)
 MOVE CONTAINER (EXCI) コマンド [51](#)
 PUT CONTAINER (EXCI) コマンド [53](#)
 QUERY CHANNEL (EXCI) コマンド [57](#)
 STARTBROWSE CONTAINER (EXCI) コマンド [57](#)
 C 言語
 クライアント・プログラムに関する特別な考慮事項 [61](#)
 コピーブック [33](#)
 サンプル・プログラム [35](#)
 CICS 提供プロシージャ、DFHYXTDL [60](#)
 CICS 提供プロシージャ、DFHYXTL [60](#)
 EXCI CALL インターフェース [12](#)
 COBOL
 コピーブック [33](#)
 CICS 提供プロシージャ、DFHYXTVL [60](#)
 CICS 提供プロシージャ、DFHZXTCL [60](#)
 EXCI CALL インターフェース [12](#)
 EXCI DPL 呼び出しの例 [35](#)
 EXEC CICS LINK コマンド
 自動再試行 [40](#)
 セキュリティ検査 [78](#)
 変換 [43](#)
 DFHAPPL プロファイル定義 [78](#)
 EXEC CICS か CALL インターフェースかの選択 [4](#)
 LINK コマンド
 EXEC CICS か CALL インターフェースかの選択 [4](#)
 PL/I
 クライアント・プログラムに関する特別な考慮事項 [61](#)
 コピーブック [33](#)
 サンプル・プログラム [35](#)
 CICS 提供プロシージャ、DFHYXTPL [60](#)
 CICS 提供プロシージャ、DFHZXTPL [61](#)
 EXCI CALL インターフェース [12](#)
 RRMS
 外部 CICS インターフェース (EXCI) による使用 [9](#)
 SYNCONRETURN
 DPL 要求 [9](#)
 DPL_Request による省略 [9](#)
 アセンブラー
 コピーブック [33](#)
 サンプル・プログラム [35](#)
 CICS 提供プロシージャ、DFHEXTAL [60](#)
 EXCI CALL インターフェース [12](#)
 大きな COMMAREA (チャンネル)
 DELETE CHANNEL (EXCI) コマンド [43](#)
 DELETE CONTAINER (EXCI) コマンド [44](#)
 QUERY CHANNEL (EXCI) コマンド [57](#)

- クライアント・プログラム
 - アドレッシング・モード [58](#)
 - 開始するためのサンプル・ジョブ [59](#)
 - コンパイル [60](#)
 - その定義 [3](#)
 - 必要な JCL
 - EXCI クライアントの実行 [59](#)
 - SYSMDUMP のキャプチャー [82](#)
 - 複数セッションの使用 [3](#)
 - 変換 [43](#), [60](#)
 - リンク・エディット [60](#)
 - EXEC CICS LINK によるサーバーへのリンク [36](#)
 - MRO ログオン・セキュリティおよびバインド時のセキュリティ [77](#)
 - PL/I および C 言語に関する考慮事項 [61](#)
- クライアント・プログラムのスタブ
 - DFHXCIE [58](#)
 - DFHXCIS [58](#)
 - DFHXCSTB [58](#)
- コンテナー・コマンド
 - DELETE CHANNEL (EXCI) [43](#)
 - DELETE CONTAINER (EXCI) [44](#)
 - ENDBROWSE CONTAINER (EXCI) [45](#)
 - GET CONTAINER (EXCI) [46](#)
 - GETNEXT CONTAINER (EXCI) [50](#)
 - MOVE CONTAINER (EXCI) [51](#)
 - PUT CONTAINER (EXCI) [53](#)
 - QUERY CHANNEL (EXCI) [57](#)
 - STARTBROWSE CONTAINER (EXCI) [57](#)
- サーバー・プログラム
 - セキュリティの考慮事項 [77](#)
 - その定義 [3](#)
 - プログラミング上の制約事項 [4](#)
 - API 制限 [4](#)
 - DPL サブセット [4](#)
 - EXEC CICS LINK によるクライアントからのリンク [36](#)
- ジョブ制御言語 (JCL)
 - EXCI クライアント・プログラム実行用 [59](#)
 - SYSMDUMP のキャプチャー用の [82](#)
- チャネル・コマンド
 - DELETE CHANNEL (EXCI) [43](#)
 - DELETE CONTAINER (EXCI) [44](#)
 - ENDBROWSE CONTAINER (EXCI) [45](#)
 - GET CONTAINER (EXCI) [46](#)
 - GETNEXT CONTAINER (EXCI) [50](#)
 - MOVE CONTAINER (EXCI) [51](#)
 - PUT CONTAINER (EXCI) [53](#)
 - QUERY CHANNEL (EXCI) [57](#)
 - STARTBROWSE CONTAINER (EXCI) [57](#)
- パイプ
 - オープン [18](#)
 - 関連付けられているストレージの解放 [31](#)
 - クローズ [29](#)
 - 接続 [18](#)
 - 切断 [29](#)
 - その定義 [3](#)
 - 閉じたパイプの再使用 [29](#)
 - 開いたままの状態に関する制約事項 [18](#)
 - 割り振り [16](#)
 - 割り振り解除 [31](#)
 - ALLOCATE_PIPE 中の DFHXCURM の呼び出し [68](#)
- パラメーター
 - null [35](#)
- 複数領域操作 (MRO)
 - 複数領域操作 (MRO) (続き)
 - システム間 (XCF/MRO) [3](#)
 - ログオン・セキュリティおよびバインド時のセキュリティ [77](#)
 - 戻りコード
 - R15 のクリア [61](#)
 - リソース・リカバリー・サービス (RRS)
 - 2 フェーズ・コミット・プロトコル [9](#)
 - リンク・エディット
 - クライアント・プログラムの [58](#)
 - DFHAUPLD の使用 [70](#)
 - DFHXCSTB オプション・テーブル [70](#)
 - DFHXCSTB スタブの使用 [12](#)
 - EXEC CICS LINK コマンドに必要な変換 [43](#)
 - 応答コード
 - Allocate_Pipe 呼び出し [17](#)
 - Close_Pipe 呼び出し [30](#)
 - Deallocate_Pipe 呼び出し [31](#)
 - DPL 呼び出し [27](#)
 - Initialize_User 呼び出し [13](#)
 - Open_Pipe 呼び出し [19](#)
 - 理由コード
 - Allocate_Pipe 呼び出し [17](#)
 - Close_Pipe 呼び出し [30](#)
 - Deallocate_Pipe 呼び出し [31](#)
 - DPL 呼び出し [27](#)
 - Initialize_User 呼び出し [13](#)
 - Open_Pipe 呼び出し [19](#)
 - EXCI
 - 静的ルーティング [63](#)
 - 動的ルーティング [63](#)
 - アドレッシング・モード (AMODE)
 - クライアント・プログラムの要件 [58](#)
 - アプリケーション・プログラミング
 - 言語に関する考慮事項 [61](#)
 - コピーブック [33](#)
 - コマンド [12](#)
 - サーバー・プログラムの制約事項 [4](#)
 - スタブ (stub) [58](#)
 - DPL サブセット [4](#)
 - EXEC CICS LINK コマンドに必要な変換 [43](#)
 - LINK コマンドで返される例外条件 [38](#)
 - RESP および RESP2 フィールド [38](#)
 - 分散プログラム・リンク (DPL)
 - サーバー・プログラムの API サブセット [4](#)
 - プログラム呼び出しの要求 [20](#)
 - userid と uowid を省略した COBOL 呼び出し例 [35](#)

