

照会報告書作成プログラム



QMF (OS/390 版) インストールおよび 管理の手引き

バージョン 7

照会報告書作成プログラム



QMF (OS/390 版) インストールおよび 管理の手引き

バージョン 7

お願い

本書、ならびに本書が記述する製品をご使用になる前に、603ページの『付録G. 特記事項』を必ずお読みください。

本書は、DB2 ユニバーサル・データベース・サーバー (OS/390 版)(DB2 UDB for OS/390) バージョン 7 (プログラム番号 5675-DB2) リリース 1 のフィーチャーである照会報告書作成プログラムに適用されます。また改訂版などで特に断りのない限り、これ以降のすべてのリリースにも適用されます。

本書は、GD88-7234-00 の改訂版です。

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

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

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

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

(URL は、変更になる場合があります)

原典： GC27-0719-00
Query Management Facility™
Installing and Managing QMF
on OS/390
Version 7

発行： 日本アイ・ビー・エム株式会社

担当： ナショナル・ランゲージ・サポート

第1刷 2000.9

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

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

© Copyright International Business Machines Corporation 1983, 2000. All rights reserved.

Translation: © Copyright IBM Japan 2000

目次

QMF ライブラリー	xiii
本書について	xv
本書の対象読者	xv
前提知識	xv
本書の使用法	xvi
本書で使用される規則および用語	xvii
前提条件および関連情報	xvii
各国語機能情報の表し方	xviii
QMF の資料の発注方法	xix

第1部 QMF (OS/390 版) のインストール 1

第1章 QMF 紹介およびインストール・プロセス 5

QMF 紹介	5
QMF が他のデータベースのデータにアクセスする方法	6
リモート作業単位	7
DB2 UDB (OS/390 版) 間分散作業単位	8
データベース・インストール・プロセスの概要	9
QMF のための DB2 UDB (OS/390 版) の要件	9
DB2 UDB (OS/390 版) に関する前提知識	9
QMF のインストールによって作成された DB2 UDB (OS/390 版) オブジェクト	11
データベース許可 ID Q	11
QMF インストール・プロセスのロードマップ	11
QMF をリモート作業単位用に設定する	14
DB2 UDB (OS/390 版) 間分散作業単位用の QMF の設定	15
例	15

第2章 QMF の計画 17

ハードウェア要件	17
前提条件のソフトウェア	17
リモート作業単位をサポートするために必要な製品	23
記憶域要件の計画	24

パフォーマンスを向上させるためのモジュールの移動	24
CICS の場合	26
SMPE 記憶域の見積もり	26
配布ライブラリー用のスペースの見積もり	27
ターゲット・ライブラリーのサイズの見積もり	27
複数のリリースのための計画	28
ユーザー・データ・セット用のスペースの見積もり	28
Program Directory の読み取りおよびサービスの適用	29
CICS のもとでの QMF の計画	29
QMF のための CICS の調整	29
QMF のための GDDM の調整	30
AIX [®] 用の DB2 UDB (OS/390 版) のための QMF の計画	30
ワークシートへの記入	31

第3章 入力パラメーターの指定 35

ステップ 1 -- QMF インストール・パラメーターの指定	35
開始の前に	35
インストール・パネルの開始	36
ローカル DB2 UDB (OS/390 版) パラメーターの指定	38
データベース・インストールの効力範囲の指定	40
リモート・サーバー・ロケーションの指定	41
DB2 UDB (OS/390 版) および QMF のパラメーターの指定	42
リモート・サーバー・パラメーターの指定	44
QMF 表スペースに関するスペース・パラメーターの指定	46
QMF 索引スペースのパラメーターの指定	47
ジョブ・カードの指定	47
ステップ 2 -- ジョブの調整	48
ステップ 3 -- フォアグラウンドでの QMF のインストール	49

第4章 QMF バッチ・インストール・ジョブの 実行依頼	51
ステップ 4 -- QMF パネルのインストール	51
ステップ 5 -- QMF/GDDM マップ・グループ のインストール	51
ステップ 6 -- QMF/GDDM サンプル図表書式 のインストール	52
ステップ 7 -- REXX EXEC および CLIST レ コードの変換	52
REXX EXEC レコードの変換	52
CLIST レコードの変換	53
DB2 ユニバーサル・データベース (OS/390 版) アプリケーションとしての QMF の準備	55
ステップ 8 -- DB2 UDB (OS/390 版) への QMF インストール・プログラムのバインド	56
ステップ 9 -- QMF 制御表の作成	56
タイプ 2 への QMF 制御表索引の変換	56
リモート作業単位についてのヒント	57
QMF バージョン 3 リリース	
3.0、2.0、1.1、1.0 からの移行	58
QMF バージョン 2.4 からの移行	58
タイプ 2 に変換した索引の回復	61
QMF の旧リリースがインストールされて いない場合の制御表の作成	61
ステップ 10 -- QMF IVP 用の表スペースの作 成	64
QMF サンプル表の設定	65
ステップ 11 -- 旧サンプル表の削除	65
ステップ 12 -- QMF サンプル表の作成	66
ステップ 13 -- QMF パッケージのバインド	67
ステップ 14 -- DB2 UDB (OS/390 版) へのコ ミュニケーション・パッケージのバインド	68
ステップ 15 -- QMF アプリケーション・プラ ンの DB2 UDB (OS/390 版) へのバインド	68
第5章 QMF の TSO 用調整	71
ステップ 16 -- TSO ログオン・プロシージャ ーの作成	71
TSO での QMF の開始	71
TSO ログオン・プロシージャの準備	72
データ抽出プログラム (DXT) [™] に関する考 慮事項	75
ステップ 17 -- QMF の開始	76
ISPF を使用した QMF の開始	76
TSO での QMF の開始	78

ステップ 18 -- バッチ IVP を実行するための
QMF バッチ・ジョブの設定 (オプション) . 79

第6章 QMF の CICS 用調整	81
ステップ 19 -- CICS での DB2 UDB (OS/390 版) に対する QMF の記述	81
ステップ 20 -- QMF と DFHEAI および DFHEAI0 とのリンク・エディット	82
QMF と CICS コマンド・インターフェ ース・モジュールとのリンク・エディット	82
QMF 提供の管理プログラムの変換、アセ ンブル、およびリンク・エディット	82
ステップ 21 -- QMF/GDDM データ・セット の定義およびロード	83
GDDM ADMF データ・セットへの QMF/GDDM マップ・セットのロード	83
QMF/GDDM 図表および QMF トレース・ データ・セットの作成	83
ステップ 22 -- CICS 管理テーブルの更新	84
CICS 管理テーブルの更新 (CICS バージョ ン 2)	85
CICS 管理テーブルの更新 (CICS バージョ ン 3 またはそれ以降のリリース)	87
ステップ 23 -- QMF プロファイルの調整	89
ステップ 24 -- CICS 始動ジョブ・ストリーム の更新	89

第7章 QMF の ワークステーション・デー タ・サーバー用調整	91
ステップ 25 -- DB2 DRDA AS への QMF イン ストール・プログラムのバインド	92
ステップ 26 -- DB2 DRDA AS での QMF 制 御表の作成	93
ステップ 27 -- DB2 DRDA AS への QMF ア プリケーション・プログラムのバインド	93
ステップ 28 -- DB2 DRDA AS での QMF サ ンプル表の作成	94
DB2 DRDA AS からの QMF の削除	95
QMF の削除	95
DB2 DRDA AS からの QMF サンプル表 の削除	95
DB2 DRDA AS に対する QMF の開始	96

第8章 QMF の DB2 (AS/400[®]) サーバー 用調整	97
ステップ 29 -- DB2 (AS/400 版) への QMF インストール・プログラムのバインド	97

ステップ 30 -- DB2 (AS/400 版) サーバーでの QMF 制御表の作成	98	ステップ 4 ~ 8 -- ジョブの手動による実行依頼	138
ステップ 31 -- DB2 (AS/400 版) サーバーへの QMF アプリケーション・プログラムのバインド	98	ステップ 4 -- QMF パネルのインストール	138
ステップ 32 -- DB2 (AS/400 版) サーバーでの QMF サンプル表の作成	99	ステップ 5 -- NLF/GDDM マップ・グループのインストール	138
DB2 (AS/400 版) サーバーに対する QMF の始動	100	ステップ 6 -- REXX EXEC または CLIST レコードの変換	139
第9章 QMF インストールのテスト	101	ステップ 7A -- QMF 制御表の更新	142
ステップ 33 (TSO の場合) -- IVP の実行	101	ステップ 7B と 7C -- QMF NLF サンプル表の設定	147
ステップ 33 (CICS の場合) -- IVP の実行	103	ステップ 7B -- 以前の QMF NLF サンプル表の削除	147
QMF を開始する前に	103	ステップ 7C -- NLF サンプル表の作成	148
QMF の開始およびテスト	104	ステップ 8 -- NLF/QMF の TSO 用調整	150
ステップ 34 -- QMF アプリケーション照会およびアプリケーション・オブジェクトのインストール (TSO).	106	ステップ 9 -- NLF/QMF の CICS 用調整	151
ステップ 35 -- バッチ・モード IVP の実行 (オプション).	108	ステップ 10 -- QMF NLF の Workstation Database Server 用調整 (オプション)	155
ステップ 36 -- インストール後の終結処置	109	ステップ 11 -- QMF NLF の DB2 (AS/400 版) サーバー用調整 (オプション).	158
旧アプリケーション・プランの解放	110	ステップ 12 -- バッチ IVP 実行のための NLF バッチ・ジョブのセットアップ (オプション).	159
QMF バージョン 7.1 と前のリリースが、異なる DB2 UDB (OS/390 版) サブシステム内にある場合	111	ステップ 13 -- QMF 対話モードでの IVP の実行	159
ステップ 37 -- 永続ライブラリーの受け入れ	112	ステップ 14 -- 各国語サンプル照会およびプロシージャのインストール	160
ステップ 38 -- セキュリティーの終結処置	113	ステップ 15 -- バッチ・モード IVP の実行 (オプション).	162
第10章 QMF NLF の計画およびインストール	115	ステップ 16 -- ポストインストールの終結処置	162
プロファイル表と NLF	115	ステップ 17 -- 永続ライブラリーの受け入れ	163
QMF NLF の計画	116	ステップ 18 -- CDS 間環境の作成	163
ハードウェアとプログラム製品の要件	116	第11章 リモート・サーバーでの QMF 7.1 パッケージのバインド	165
SMP/E 要件	116	第2部 QMF (OS/390 版) の管理 167	
QMF NLF ユーザー・データ・セット	118	第12章 必要な記憶域	177
IBM ソフトウェア配布 (ISD) テープ	118	QMF 記憶域要件	177
FMID	118	OS/390 の記憶域	177
インストール・プロセス	119	CICS/MVS 領域	178
予備 : Program Directory を読み NLF ワークシートに記入する	125	CICS/ESA 領域	178
ステップ 1 -- QMF NLF インストール・パラメーターの指定	128		
ステップ 2 -- ジョブの調整	136		
ステップ 3 -- フォアグラウンドでの QMF NLF のインストール	137		

第13章 QMF の開始	179	DSQSIROW (表示用に検索される報告書の 行数の制御).	214
QMF を開始する前に	179	セッション開始時点での QMF アクティビ ティーのトレース	217
環境の確立	179	DSQSDEBUG (トレース詳細レベルの設定)	217
クイック・スタート	180	DSQSDBQT (CICS のトレース・データ用 記憶域のタイプの指定).	218
QMF を ISPF のもとで実行するように設定	181	DSQSDBQN (CICS のトレース・データ用 記憶域の名前の指定).	219
メニュー・オプションからの QMF の開 始	182	セッション中の初期アクティビティーの制御	220
ISPSTART コマンドによる QMF の開始	185	DSQSDBNM (QMF 開始時の接続先のロケ ーションの指定)	220
ISPF でバッチ・モードによる QMF の開 始	185	DSQSMODE (対話式または非対話式 QMF セッションの指定)	221
ISPF のもとでの QMF の開始の例 . . .	186	DSQSSUBS (QMF で使用される DB2 サ ブシステムの命名)	221
QMF を TSO のもとで実行するように設定	187	DSQSRUN (QMF の開始時に実行するプ ロシージャーの命名).	222
TSO ID の定義	187	DSQSPLAN (QMF アプリケーション・プ ランの命名).	229
ID のタイプの選択	188	DSQSPRID (TSO プロファイル・キーの指 定).	230
DSQQMFE モジュールにより QMF を直接開 始	189	DSQSDBCS (2 バイト文字セット・データの 印刷の設定).	230
TSO CALL コマンドを使用した QMF の開 始	189	REXX プログラム DSQSCMD を使用したデ フォルトの開始値の設定	231
TSO のバッチ環境での QMF の開始 . . .	190		
TSO のもとでの QMF の開始の例 . . .	190		
TSO EXEC の作成	191		
プログラム・ロード・ライブラリーの検査	191		
TSO に関する考慮事項	191		
QMF データ・セットの検査	192		
GDDM データ・セットの検査	192		
QMF をバッチ・ジョブとしてネイティブ OS/390 で実行するように設定	193		
QMF を CICS のもとで実行するように設定	194		
CICS/DB2 接続機能の使用	195		
CICS のもとでの QMF の開始の例 . . .	196		
QMF を SRPI からサーバーとして実行する ように設定	196		
第14章 開始手順のカスタマイズ	199	第15章 QMF セッション制御機能	235
プログラム・パラメーターの要約	199	Q.SYSTEM_INI のインストールまたは除去	235
クイック・スタート	200	デフォルトのシステム初期化プロシー ジャーのインポート	235
報告書記憶域のカスタマイズおよび報告書パ フォーマンス	203	Q.SYSTEM_INI プロシージャーを実行する時 期	236
DSQSBSTG (報告書データ用の記憶域の調 整).	203	Q.SYSTEM_INI の使用法	236
DSQSRSTG (アプリケーションに使用され る予約記憶域の調整).	205	QMF と共に出荷される例	237
DSQSPILL (エクストラ記憶域の獲得) . .	207	ユーザー・セッション・プロシー ジャーの例	237
DSQSSPQN (CICS 予備記憶域の名前の指 定).	214	オブジェクトのリストを表示するプロシー ジャー	238
		セキュリティおよびセッション・プロシー ジャーの共用	239
		診断の考慮事項	239
		第16章 エンド・ユーザーのための QMF サ ポートの確立	241
		QMF コード・ページに関する考慮事項 . .	241

Q AUTHID の役割	241	ユーザーが図表をサポートできるようにする	282
クイック・スタート	241	TSO および ISPF での図表サポート	283
QMF へのユーザー・アクセスを可能にする		CICS での図表のサポート	284
ユーザー・プロファイルの作成	242	制御表を使用した QMF オブジェクトの保守	284
インストール先システムのためのプロフ		Q.OBJECT_DIRECTORY 表の読み取り	285
イル構造の確立	243	Q.OBJECT_DATA 表の読み取り	286
新規のユーザー・プロファイルの		Q.OBJECT_REMARKS 表の読み取り	288
Q.PROFILES 表への追加	243	QMF 照会、書式、およびプロシージャー	
固有のプロファイルを持たないユーザーに		のリスト	288
よる QMF の使用の防止	244	QMF 照会、書式、およびプロシージャー	
Q.PROFILES 制御表の読み取り	245	の表示	289
正しいプロファイルをユーザーの操作環境		照会、書式、およびプロシージャーの所有	
に提供	251	権の転送	289
ユーザー・プロファイルの更新	252	古くなった 照会、書式、およびプロシ	
Q.PROFILES 制御表からのプロファイルの		ジャーの削除	290
削除	254	OS/390 データ・セット内の照会、書式、	
QMF アプリケーション・プランおよびパッ		およびプロシージャーのインポート	291
ッケージに対するアクセスの制御	255	QMF オブジェクト制御表用の表スペース	
アプリケーション・プランおよびパッケ		の拡大	291
ージに対するアクセスの付与	255	DB2 サブシステムの保守	294
QMF アプリケーション・プランおよびパ		データ・セットの管理	294
ッケージに対するユーザー・アクセスの取		制御表の保守	295
り消し	256	索引使用の判別	297
QMF オブジェクトおよびデータベース・オ		バッファー・プールの切り替え	297
ブジェクトに対するアクセスの制御	256	DB2 カタログ表を使用する表および視点の保	
オブジェクトをアクセスするのに必要な		守	298
DB2 特権	257	表と視点のリスト	298
DB2 特権の付与と取り消し	258	表または視点の所有権の転送	298
オブジェクトにアクセスするのに必要な		表または視点のデータベースからの削除	299
SQL 特権	265	ローカルで定義された日付 / 時刻フォーマッ	
SQL 特権の付与と取り消し	268	トのサポート	299
他のユーザーとの QMF オブジェクトの		フォーマットの指定	299
共用	270	編集ルーチンを使用可能に	299
非コミット読み取りの許可	271	DXT エンド・ユーザー・ダイアログへのア	
オブジェクトの作成に関する標準の設定	271	クセス (ISPF のみ)	300
ユーザーのデータベース・オブジェクト・リ		EXTRACT コマンドのサポート	301
ストのカスタマイズ	272	リソースの割り振り	301
デフォルト・オブジェクト・リストの使用	272	DXT データ・セットの割り振り	301
デフォルト・リストの変更	275	CLIST の使用によるリソースの割り振り	
オブジェクト・リストの記憶域要件	277	および割り振り解除	302
ユーザーがデータベース内で表を作成できる		割り振り CLIST の準備	302
ようにする	277	割り振り解除 CLIST の準備	306
ユーザー用の表スペースの選択および割り		ユーザー用文書編集インターフェースのカス	
当て	279	タマイズ	311
ユーザーへの DB2 CREATETAB 権限の		アプリケーションの変更	312
付与	282	文書インターフェース・マクロの名前変更	312

Q.DSQAED1S プロシーチャーをデータベ ースに入れる	313
Q への所有権の転送	313
データ・コンポーネントの変更	314
CLIST およびマクロの変更	316
QMF EDIT コマンドのカスタマイズ	319
NLF 環境での英語サポートのイネープリング	321
グローバル変数を使用した通貨記号の定義	322
第17章 ユーザーによるオブジェクトの印刷 323	
クイック・スタート	323
オブジェクトの印刷	324
印刷に QMF または GDDM のいずれのサー ビスを使用するか決定	326
GDDM サービスを使用して印刷を処理する	327
プリンターのための GDDM ニックネーム の選択	327
ニックネーム指定の作成	328
GDDM デフォルト・モジュールをニック ネームで更新	333
外部デフォルトのファイル内でニックネ ーム定義をテスト (TSO およびネイティブ OS/390 バッチのみ)	334
TSO およびネイティブ OS/390 バッチに ニックネーム・ファイルを割り振る	334
CICS でニックネームを使用	334
QMF が GDDM ニックネームとインター フェースする方法	336
TSO およびネイティブ OS/390 バッチの印刷 で QMF サービスを使用	337
CICS で印刷を行うために QMF サービスを 使用	337
一時記憶域キューと一時データ・キューの 選択	337
PRINT コマンドを使用して出力をキュー に経路指定	338
グローバル変数を使用して印刷のためのク ューを定義	339
CICS 一時記憶域キューからの印刷	339
CICS 一時記憶域キューからの報告書を見 る	339
TSO およびネイティブ OS/390 バッチの印刷 機能キーの同義語の定義	339
CICS の印刷機能キーの同義語の定義	340
ユーザー・プロファイルを更新して GDDM 印刷を可能にする	341

第18章 QMF コマンドのカスタマイズ	343
クイック・スタート	343
QMF が提供するデフォルトの同義語の使用	344
TSO における印刷報告書の表示 (DPRE)	345
コマンド同義語表の作成	347
コマンド同義語定義を表へ入力	349
動詞の選択	350
目的語名の選択	351
同義語定義の選択	352
同義語のアクティブ化	356
コマンド同義語表の保守の最小化	357
1 つの同義語表を全ユーザーに割り当てる	357
同義語表の視点を個々のユーザーに割り当 てる	358

第19章 QMF 機能キーのカスタマイズ	361
クイック・スタート	361
カスタマイズしたいキーの選択	361
フルスクリーン・パネルのデフォルト・キ ー	362
ウィンドウ・パネルのデフォルト・キー	363
機能キー表の作成	364
ユーザーの機能キー定義を表へ入力	366
コマンドと機能キーの結び付け	366
機能キーにラベルを付けて画面上に配置 キー定義の例	369
カスタマイズしたいパネルの識別	371
フルスクリーン・パネル ID	371
ウィンドウ・パネル ID	372
新規機能キー定義のアクティブ化	374
機能キー表の検査と問題診断	375

第20章 QMF 書式のためのユーザー独自の 編集コードの作成	377
クイック・スタート	377
編集コードを選択する	378
DATE、TIME、および TIMESTAMP 情報の 処理	380
データをフォーマット設定するためのユーザ ー出力ルーチンの呼び出し	381
出力ルーチンとの情報の受け渡し	383
インターフェース制御ブロックのフィール ド	384
入力域の特徴を示すフィールド	386
出力域の特徴を示すフィールド	387
QMF 終了時に制御を出力ルーチンに渡す	387

編集ルーチンを HLASM またはアセンブラーで作成	388	管理プログラム出口ルーチンのプログラム・コンポーネント	446
TSO、SRPI、APPC およびネイティブ OS/390 用の編集ルーチンをアセンブラーで作成	388	TSO、SRPI、APPC、およびネイティブ OS/390 と管理プログラム出口ルーチンとの対話方法	447
編集ルーチンを CICS 用にアセンブラーで作成	390	CICS の管理プログラム出口ルーチンとの対話方法	448
アセンブラー編集ルーチンと QMF との対話方法	394	QMF が管理プログラム出口ルーチンを呼び出す方法と時期	450
編集ルーチンを PL/I で作成	397	リソース制御情報を管理プログラム出口に渡す	458
言語環境プログラム (LE) を使用せずに TSO、SRPI、APPC またはネイティブ OS/390 用に PL/I で編集ルーチンを作成	397	QMF セッション間のリソース制御情報の保管	474
言語環境プログラム (LE) を使用して TSO、SRPI、APPC、またはネイティブ OS/390 での編集ルーチンを PL/I で作成	400	ユーザー・アクティビティーの取り消し	475
編集ルーチンを CICS 用に PL/I で作成	403	取り消されたアクティビティーに対するメッセージ	476
PL/I 編集ルーチンと QMF との対話方法	407	CICS でのユーザー管理プログラム出口ルーチンの変換、アセンブル、リンク・エディット	478
編集ルーチンを COBOL で作成	412	CICS の管理プログラム出口ルーチンの変換	478
言語環境プログラム (LE) [®] を使用せずに TSO、SRPI、APPC、またはネイティブ OS/390 で編集ルーチンを COBOL で作成	412	管理プログラム出口ルーチンのアセンブル	478
言語環境プログラム (LE) を使用して TSO、SRPI、APPC、またはネイティブ OS/390 で編集ルーチンを COBOL で作成	416	管理プログラム出口ルーチンのリンク・エディット	479
編集ルーチンを CICS 用に COBOL で作成	419	管理プログラム出口ルーチンの、TSO およびネイティブ OS/390 バッチでのアセンブルとリンク・エディット	480
COBOL 編集ルーチンと QMF との対話方法	423	管理プログラム出口ルーチンのアセンブル	480
2 バイト文字セット・データの処理	428	管理プログラム出口ルーチンのリンク・エディット	480
DBCS データの編集コード	429	DB2 管理プログラムの使用	481
編集ルーチンが受け取るもの	429	リソースのモニター	481
編集ルーチンは必ず正しい結果を戻すこと	430	管理プログラム間の相違点	482
		最大プロセッサ時間の超過	482
		DB2 管理プログラムの QMF への適用	483
第21章 管理プログラム出口ルーチンを使用した QMF リソースの制御	433	第22章 リモート・データベース接続のカスタマイズ	485
クイック・スタート	433	クイック・スタート	485
IBM 提供の管理プログラム出口ルーチンの使用	434	必要なりモート・データベース接続の判別	486
デフォルトの限界のアクティブ化	436	リモート作業単位との接続	488
管理プログラム出口ルーチンのリソースの制御方法	437	DB2 間分散作業単位による接続 (DB2 (OS/390 版) のみ)	488
ユーザー独自のリソース限界の定義	441	リモート作業単位に必要な接続の検査	489
ユーザー独自のリソース制御表の作成	443	DB2 接続の検査	489
IBM 提供の管理プログラム出口ルーチンの変更またはユーザー独自のルーチン作成	445	DB2 (VM 版) 接続の検査	490

DB2 (OS/390 版) 以外のロケーションへの	アプリケーション使用権限の割り当て . . .	514
QMF OS/390 ユーザーによるアクセスの準備	アプリケーションの使用	515
コマンド同義語表の作成	アプリケーションの開始	515
DPRE コマンドをサポートする QMF の	プロンプト・パネルへの入力	516
準備	バッチ・アプリケーションの変更	520
その他のコマンドをサポートする QMF	ネイティブ OS/390 での QMF バッチの実行	525
の準備	ユーザーが CICS でバッチ・モードを使用で	
機能キー表の作成	きるようにする	527
QMF 管理プログラム制御表の更新	端末からのバッチの実行	527
各国語機能の QMF サーバーへのインス	端末なしでのバッチの実行	528
トール	プロシージャのデバッグ	528
コード・ページ・サポート	終了リターン・コード	529
ユーザーがリモート・データベースにアクセ		
スできるようにする		
ユーザー・プロファイルの更新		
現行 SQL 許可 ID へのアクセスの指定		
ローカル・データベースへの接続		
リモート・データベースへの接続		
CICS におけるリモート・データベースで		
の実行		
ロケーション名の指定		
ユーザー・アクセスのためにデータのある		
べき位置		
SQL エラーの防止		
ユーザー ID の変換		
各リモート QMF ロケーションからの		
QMF ユーザーの削除		
自分のロケーションに管理者がアクセスでき		
るようにする		
第23章 バッチ・プログラムとして実行する	第24章 障害追及と問題診断	531
ための QMF のカスタマイズ	クイック・スタート	531
クイック・スタート	一般的な問題の障害追及	532
ユーザーが TSO でバッチ・モードを使用で	初期化エラーの処理	532
きるようにする	警告メッセージの処理	533
バッチ・モードでの操作権限	印刷時の GDDM エラーの処理	534
RACF セキュリティーの考慮事項	印刷時の QMF エラーの処理	536
TSO SUBMIT コマンドによる OS/390 へ	表示エラーの処理	537
のジョブの送信	パフォーマンスの問題の解決	538
TSO で QMF バッチ・ジョブ実行のため	診断援助機能を使用した問題の判別	541
の JCL	症状に応じた適切な診断援助機能の選択	541
TSO または ISPF 使用によるフォアグラ	QMF メッセージ・サポートを使用した問	
ウンドでの QMF バッチの実行	題の診断	542
プロシージャのデバッグ	QMF トレース機能の使用	544
QMF バッチ照会 / プロシージャ・アプリ	異常終了の診断	552
ケーション (BATCH) の ISPF での使用	TSO での QMF 割り込み機能の使用	555
	Q.ERROR_LOG 表からのエラー・ログ報	
	告書の使用	557
	IBM への問題報告	558
	ServiceLink を使用して既に報告されてい	
	る問題の検索	558
	IBM サポート・センターと共に作業する	561
	付録A. 問題が発生した場合	563
	表示される可能性のあるエラー・メッセージ	563
	ABENDASRA	563
	AEY9 ABEND	564
	AZTS ABEND	564
	DSNT302I	564
	DSQ10297	564
	DSQ10493	565
	DSQ36805	565
	DSQI004I	565

DSQI0026	565	呼び出し可能インターフェースに関する考 慮事項	586
G050 ABEND	565	書式アプリケーション移行援助プログラム	587
IDC3012I.	565	OS/390 の ISPF のもとでの QMF の実行	588
IDC3009I.	565	移行に関するその他の考慮事項	588
IDC0551I.	565	31 桁 10 進数のサポート	589
IEW0342	566	管理プログラム	589
IEW0461	566	CICS でのユーザー編集ルーチン	590
DSQ22843	566	TSO およびネイティブ OS/390 バッチ で のユーザー編集ルーチン	590
警告メッセージ	566	CICS での呼び出し可能インターフェース	590
エラー・メッセージが表示されなかった場 合	567	CICS での印刷	590
QMF トレース・データ・セット		OS/390 上の CICS でのエクスポート / イ ンポート・サポート	590
DSQDEBUG へのアクセス	567	バージョン 2 からの移行	591
付録B. DB2 に常駐している QMF オブジェ クト	569	フォールバック	592
汎用 QMF データベース・オブジェクト . . .	569	前のプロファイルの再確立	593
VSAM クラスタ	570	前のリリースでの QMF バージョン 7 の オブジェクトの使用	593
QMF 制御表	570	前のリリースでの バージョン 7 QMF コ マンドの使用	594
デフォルトのリスト視点	571	31 桁 10 進数のサポート	595
QMF プラン	571	付録E. QMF および GDDM プログラムの CICS への定義のしかた	597
QMF パッケージ	571	QMF プログラムの CICS/MVS への定義のし かた	597
NLF パーツ	572	常駐の QMF プログラム	597
QMF サンプル表	572	非常駐プログラムがパフォーマンスに与え る影響	598
付録C. フォールバック	573	QMF 初期化中の GDDM 定義のロードのし かた	598
前のプロファイルの再確立	573	非常駐 GDDM プログラムが QMF に与 える影響	598
前のリリースでの QMF バージョン 7 のオ ブジェクトの使用	573	QMF インストール後に図表機能を追加す る	599
前のリリースでの バージョン 7 QMF コマ ンドの使用	575	リソースの使用を制御するためのトランザク ション経路指定の使用	599
31 桁 10 進数のサポート	575	付録F. QMF 制御表および QMF で使用さ れる表スペース	601
付録D. QMF OS/390 リリース間の移行	577	付録G. 特記事項	603
移行とは何か	577	商標	605
フォールバックとは何か	578	用語集	607
QMF バージョン 7 アプリケーション・プラ ンおよびパッケージへのアクセス権の付与	578		
DB2 サブシステムと移行	579		
同じ DB2 サブシステム上にある QMF の 移行	579		
異なる DB2 サブシステム間での QMF の 移行	582		
QMF オブジェクトの移行	585		
照会および書式	586		
プロシージャ	586		
アプリケーションの移行	586		

参考文献	625	HLASM の資料	628
APPC の資料	625	ISPF/PDF の資料	628
CICS の資料	625	OS/390 の資料	628
COBOL の資料	626	PL/I の資料	629
DATABASE 2 の資料	626	REXX の資料	629
DCF の資料	627	ServiceLink の資料	629
DRDA の資料	627	VM の資料	629
DXT の資料	627	VSE の資料	629
図形データ表示管理プログラム (GDDM) の 資料	627		

QMF ライブラリー

資料のご注文は、IBM 担当員にお申し付けください。

評価



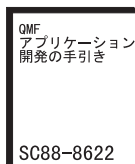
インストール、
プランニング、
管理、
および診断



使用



アプリケーション・
プログラミング



オンライン・
ライブラリー



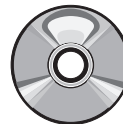
SK2T-0730
OS/390、VM、
および VSE



SK2T-6700
OS/390 のみ



SK2T-2067
VM のみ



SK2T-0060
VSE のみ

QMF ライブラリー

本書について

本書は、データベース管理者およびシステム・プログラマーが照会報告書作成プログラム (QMF) 製品を OS/390 (OS/390)[®] のもとでインストールおよび管理するために使用するものです。

本書の対象読者

本書は、DATABASE 2[™] (OS/390 版)(DB2 UDB (OS/390 版)) リレーショナル・データベースを併用して QMF のインストールおよび管理を担当する OS/390 システム・プログラマーを対象としています。また、本書は、ネットワーク・アプリケーションのインストールおよび管理を担当するネットワーク管理者も対象としています。

前提知識

ユーザーは、特定の操作環境を構成する各コンポーネントについて習熟しておく必要があります。コンポーネントには次のようなものがあります。

- OS/390。
- 多重仮想記憶 / エンタープライズ・システム体系 (MVS/ESA)[™] オペレーティング・システム。
- QMF とその関連製品をサポートする環境であるタイム・シェアリング・オプション (TSO)。この環境で TSO は、多くの管理用タスクを行う場合に役立ちます。管理用タスクには、DB2 ユーティリティの実行、QMF IMPORT コマンドの使用による QMF プロシージャおよび照会の作成、QMF コマンド・インターフェースを介して CLIST から行う QMF コマンドの実行などがあります。
- QMF 用のダイアログ管理機能である対話式システム生産性向上機能 (ISPF)。
- 汎用のデータ通信およびオンライン・トランザクション処理システムである顧客情報管理システム (CICS)[®]。CICS/MVS は、QMF と MVS/ESA[™] の間のインターフェースになります。
- QMF でユーザーの画面にパネルを表示し、図表を作成できるようにする図形データ表示管理プログラム (GDDM)[®]。
- QMF 用のデータベース・マネージャーである DATABASE 2 (DB2)[™]。

DB2 では、バッチ方式により、または DB2I (DB2 対話機能) によって実行できる多くのユーティリティーも用意されています。

- DB2 ロード・ユーティリティーにデータを提供することができる機能であるデータ抽出プログラム (DXT)[™]。
- 拡張システム修正変更プログラム (SMP/E)。
- 対話式 SQL 照会の構成および実行を可能にする機能である SPUFI。DB2I を用いて SPUFI を実行することによって、QMF の制御表の変更、システム・カタログの使用、および DB2 オブジェクトの構造の除去、作成、または変更を行うことができます。

重要:

1. SPUFI 照会を QMF のもとで実行すると、QMF のパフォーマンスに影響を生じる可能性があります。
- アセンブラー・プログラム言語。IBM[®] 提供の管理プログラム出口ルーチンの変更、またはユーザー独自の管理プログラム出口ルーチンの作成を計画する場合は、これが必要になります。QMF 書式のためにアセンブラーで編集コードを作成する場合は、HLASM またはアセンブラーを使用することもできます。
 - PL/I。これは、QMF 書式のために PL/I でユーザー独自の編集コードを作成する場合に使用することができます。
 - VS COBOL II および COBOL/370[™]。ユーザー独自の編集コードを QMF 書式用の COBOL で作成することを計画している場合は、これを使用する場合があります。

これらの製品についての資料は、625ページの『参考文献』に一覧が示されています。

また、QMF が提供するエンド・ユーザー機能についても、いくつかは習熟しておくと便利です。QMF エンド・ユーザー機能については、*QMF 使用の手引き* で説明しています。この資料および他の QMF 資料の資料番号は、本書の xiii ページにリストしてあります。

本書の使用法

本書で説明する管理とカスタマイズの作業は、本書で説明しているインストーラ・プロシージャに従って QMF がすでにインストールされていることを前提にしています。インストール先システムのデフォルトの局面のいずれかをカスタマイズする必要がある場合には、597ページの『付録E. QMF および GDDM プログラムの CICS への定義のしかた』を参照してください。

本書に示してある管理とカスタマイズのほとんどの作業は、QMF 製品そのものを使用して行われます。したがって、本書で説明している作業を開始する前に、インストール検査プロシージャ (IVP) が実行されたかどうかを確認してください。まだ実行されていない場合は、ご自分で IVP を実行し、QMF が正しくインストールされており、かつ現場の要件を満たすように構成されていることを確認してください。IVP は、QMF インストール・プロセスの最終ステップです。

このようなタスクのほとんどでは、DB2 データベース管理者 (DBA) 権限が必要です。プログラムのインストール担当者が本書に示されているデフォルト手順を実行した場合は、QMF インストール時にユーザー ID Q が定義されています。このユーザー ID には DBA 権限があります。

本書の多くの章には、「クイック・スタート」と呼ばれる節が含まれています。これらの節を使用すれば、特定のタスクごとにその実施方法の概要を把握することができます。各章ごとにクイック・スタートの節に目を通した上で、該当のタスクに伴うすべてのステップの実施方法についてさらに理解する必要がある場合は、そこに示されているページを参照してください。

本書で使用される規則および用語

インストール作業をできるだけ簡単なものにするために、IBM 製品のフルネームの多くは短縮されています。各製品は、固有の名前ではなく総称名で示されます。たとえば、DB2 (OS/390 版) は DB2 となります。

前提条件および関連情報

インストール時には、本書のほかに下記の資料を参照できるように手元においてください。

- *QMF Program Directory*
- *QMF 予防サービス計画 (PSP)* バケット

QMF Program Directory では、本書の発行後にインストール・プロセスに加えられた変更について説明しています。この資料は、インストール・テープと一緒に出荷用カートンに同梱されています。

QMF 予防サービス計画 (PSP) もお読みください。この資料は IBM ソフトウェア配布 (ISD) テープと一緒に出荷され、これにはインストールに関する追加情報が記載されています。PSP 資料については、29ページの『*Program Directory* の読み取りおよびサービスの適用』の節で詳しく説明しています。167ページの『第2部 QMF (OS/390 版) の管理』では、QMF の前のバージョンおよびリリースからのオブジェクトの移行方法について説明しています。

QMF 資料のリストについては、xiiiページの『QMF ライブラリー』を参照してください。他の IBM 製品ファミリーについての資料は、625ページの『参考文献』に一覧が示されています。

各国語機能情報の表し方

QMF では、数種類の異なる言語を使用することができます。それぞれの言語は、各国語機能 (NLF) で提供されます。

NLF を使用すると、ユーザーは、英語以外の言語で、QMF コマンドを入力したり、ヘルプやその他の情報を表示したり、QMF 作業を実行したりすることができます。NLF は、QMF の独立した機能としてインストールされます。NLF のインストールに関する詳細については、115ページの『第10章 QMF NLF の計画およびインストール』を参照してください。

本書で説明しているタスクはすべて、基本 QMF 製品 (英語) および NLF のいずれでも行うことができます。手順は、大部分が基本セッションと NLF セッションの両方で同じですが、NLF のユーザーにとっての特殊な考慮事項については、その前に必ず **NLF を使用する場合** という語句を付けてその旨を示してあります。

本書で使用しているプログラムおよびフェーズの名前には、*n* というシンボルが含まれているものがありますが、このシンボルは該当の名前を変えてもかまわないことを示しています。NLF をご使用の場合は、インストールした NLF に一致する 1 文字の各国語 ID (NLID) を表1から選んで、本書に現れる *n* というシンボルをすべて置換してください。下記の表には、QMF がそれぞれの国語を識別する名前も示してあります。

表 1. QMF 基本 (英語) および各国語機能 (NLF) を表す NLID

NLF	NLID	この NLF に QMF が使用する名前
ブラジル・ポルトガル語	P	PORTUGUES
デンマーク語	Q	DANSK
英語	E	ENGLISH
フランス語	F	FRANCAIS
ドイツ語	D	DEUTSCH
イタリア語	I	ITALIANO
日本語	K	NIHONGO
韓国語	H	HANGEUL
中国語 (簡体字)	R	S-CHINESE

表 1. QMF 基本 (英語) および各国語機能 (NLF) を表す NLID (続き)

NLF	NLID	この NLF に QMF が使用する名前
スペイン語	S	ESPANOL
スウェーデン語	V	SVENSKA
スイス・フランス語	Y	FRANCAIS (SUISSE)
スイス・ドイツ語	Z	DEUTSCH (SCHWEIZ)
英大文字	U	UPPERCASE

大文字機能 (UCF) では、英語を使用しますが、テキストはすべて英大文字に変換されます。カタカナ端末で作業しているユーザーは大文字を使用することによって、基本 QMF 製品を使用し英語のオンライン・ヘルプとメッセージを表示することができます。カタカナ・サポートが備えられた端末には、IBM 5550 マルチステーションのほかに、IBM 3277、3278、および 3279 端末があります。

QMF の資料の発注方法

QMF の資料の注文は、IBM 担当員にお申し付けください。

QMF の資料の一覧表は、xiiiページの『QMF ライブラリー』に記載してあります。

第1部 QMF (OS/390 版) のインストール

第1章 QMF 紹介およびインストール・プロセス	5
QMF 紹介	5
QMF が他のデータベースのデータにアクセスする方法	6
リモート作業単位	7
DB2 UDB (OS/390 版) 間分散作業単位	8
データベース・インストール・プロセスの概要	9
QMF のための DB2 UDB (OS/390 版) の要件	9
DB2 UDB (OS/390 版) に関する前提知識	9
分散データにアクセスするための QMF のインストール	10
QMF のインストールによって作成された DB2 UDB (OS/390 版) オブジェクト	11
データベース許可 ID Q	11
QMF インストール・プロセスのロードマップ	11
QMF をリモート作業単位用に設定する	14
リモート作業単位を用いたデータのアクセス	14
DB2 UDB (OS/390 版) 間分散作業単位用の QMF の設定	15
例	15
システム構成例および要件	15
第2章 QMF の計画	17
ハードウェア要件	17
前提条件のソフトウェア	17
リモート作業単位をサポートするために必要な製品	23
記憶域要件の計画	24
パフォーマンスを向上させるためのモジュールの移動	24
CICS の場合	26
SMP/E 記憶域の見積もり	26
配布ライブラリー用のスペースの見積もり	27
ターゲット・ライブラリーのサイズの見積もり	27
複数のリリースのための計画	28
ユーザー・データ・セット用のスペースの見積もり	28

Program Directory の読み取りおよびサービスの適用	29
CICS のもとでの QMF の計画	29
QMF のための CICS の調整	29
QMF のための GDDM の調整	30
GDDM 2.3 デフォルト・パラメーターの変更	30
GDDM のためのインストール検査プロシージャ (IVP) の実行	30
AIX [®] 用の DB2 UDB (OS/390 版) のための QMF の計画	30
ワークシートへの記入	31
第3章 入力パラメーターの指定	35
ステップ 1 -- QMF インストール・パラメーターの指定	35
開始の前に	35
インストール・パネルの開始	36
ローカル DB2 UDB (OS/390 版) パラメーターの指定	38
データベース・インストールの効力範囲の指定	40
ローカル DB2 UDB (OS/390 版) サブシステム用の QMF パラメーターの指定	40
リモート・サーバー・ロケーションの指定	41
DB2 UDB (OS/390 版) および QMF のパラメーターの指定	42
リモート・サーバー・パラメーターの指定	44
QMF 表スペースに関するスペース・パラメーターの指定	46
QMF 索引スペースのパラメーターの指定	47
ジョブ・カードの指定	47
ステップ 2 -- ジョブの調整	48
ステップ 3 -- フォアグラウンドでの QMF のインストール	49
第4章 QMF バッチ・インストール・ジョブの実行依頼	51
ステップ 4 -- QMF パネルのインストール	51
ステップ 5 -- QMF/GDDM マップ・グループのインストール	51

ステップ 6 -- QMF/GDDM サンプル図表書式のインストール	52	ISPF 選択メニューのカスタマイズ	77
ステップ 7 -- REXX EXEC および CLIST レコードの変換	52	TSO での QMF の開始	78
REXX EXEC レコードの変換	52	ステップ 18 -- バッチ IVP を実行するための QMF バッチ・ジョブの設定 (オプション)	79
CLIST レコードの変換	53	第6章 QMF の CICS 用調整	81
DB2 ユニバーサル・データベース (OS/390 版) アプリケーションとしての QMF の準備	55	ステップ 19 -- CICS での DB2 UDB (OS/390 版) に対する QMF の記述	81
ステップ 8 -- DB2 UDB (OS/390 版) への QMF インストール・プログラムのバインド	56	ステップ 20 -- QMF と DFHEAI および DFHEAI0 とのリンク・エディット	82
ステップ 9 -- QMF 制御表の作成	56	QMF と CICS コマンド・インターフェース・モジュールとのリンク・エディット	82
タイプ 2 への QMF 制御表索引の変換	56	QMF 提供の管理プログラムの変換、サンプル、およびリンク・エディット	82
リモート作業単位についてのヒント	57	ステップ 21 -- QMF/GDDM データ・セットの定義およびロード	83
QMF バージョン 3 リリース		GDDM ADMF データ・セットへの QMF/GDDM マップ・セットのロード	83
3.0、2.0、1.1、1.0 からの移行	58	QMF/GDDM 図表および QMF トレース・データ・セットの作成	83
QMF バージョン 2.4 からの移行	58	ステップ 22 -- CICS 管理テーブルの更新	84
索引スペースの停止 (DSQ1TBD1)	58	CICS 管理テーブルの更新 (CICS バージョン 2)	85
VSAM ファイルの割り振り (DSQ1TBA1)	59	DCT (宛先管理テーブル)	85
制御表の再作成	60	FCT (ファイル管理テーブル)	86
タイプ 2 に変換した索引の回復	61	PCT (プログラム管理テーブル)	86
QMF の旧リリースがインストールされていない場合の制御表の作成	61	PPT (処理プログラム・テーブル)	87
別名および VSAM ファイルの割り振り	62	CICS 管理テーブルの更新 (CICS バージョン 3 またはそれ以降のリリース)	87
QMF 制御表およびカタログ視点の作成およびロード	63	DCT (宛先管理テーブル)	87
ステップ 10 -- QMF IVP 用の表スペースの作成	64	CSD の更新	88
QMF サンプル表の設定	65	ステップ 23 -- QMF プロファイルの調整	89
ステップ 11 -- 旧サンプル表の削除	65	ステップ 24 -- CICS 始動ジョブ・ストリーム の更新	89
ステップ 12 -- QMF サンプル表の作成	66	第7章 QMF の ワークステーション・データ・サーバー用調整	91
ステップ 13 -- QMF パッケージのバインド	67	ステップ 25 -- DB2 DRDA AS への QMF インストール・プログラムのバインド	92
ステップ 14 -- DB2 UDB (OS/390 版) へのコミュニケーション・パッケージのバインド	68	ステップ 26 -- DB2 DRDA AS での QMF 制御表の作成	93
ステップ 15 -- QMF アプリケーション・プランの DB2 UDB (OS/390 版) へのバインド	68	ステップ 27 -- DB2 DRDA AS への QMF アプリケーション・プログラムのバインド	93
第5章 QMF の TSO 用調整	71	ステップ 28 -- DB2 DRDA AS での QMF サンプル表の作成	94
ステップ 16 -- TSO ログオン・プロシージャの作成	71	DB2 DRDA AS からの QMF の削除	95
TSO での QMF の開始	71		
TSO ログオン・プロシージャの準備	72		
データ抽出プログラム (DXT) TM に関する考慮事項	75		
ステップ 17 -- QMF の開始	76		
ISPF を使用した QMF の開始	76		

QMF の削除	95
DB2 DRDA AS からの QMF サンプル表 の削除	95
DB2 DRDA AS に対する QMF の開始	96

第8章 QMF の DB2 (AS/400® 版) サーバー 用調整	97
ステップ 29 -- DB2 (AS/400 版) への QMF インストール・プログラムのバインド	97
ステップ 30 -- DB2 (AS/400 版) サーバーで の QMF 制御表の作成	98
ステップ 31 -- DB2 (AS/400 版) サーバーへ の QMF アプリケーション・プログラムのバ インド	98
ステップ 32 -- DB2 (AS/400 版) サーバーで の QMF サンプル表の作成	99
DB2 (AS/400 版) サーバーに対する QMF の 始動	100

第9章 QMF インストールのテスト	101
ステップ 33 (TSO の場合) -- IVP の実行	101
ステップ 33 (CICS の場合) -- IVP の実行	103
QMF を開始する前に	103
QMF の開始およびテスト	104
ステップ 34 -- QMF アプリケーション照会 およびアプリケーション・オブジェクトのイン ストール (TSO).	106
ステップ 35 -- バッチ・モード IVP の実行 (オプション).	108
ステップ 36 -- インストール後の終結処置	109
旧アプリケーション・プランの解放	110
QMF バージョン 7.1 と前のリリースが、 異なる DB2 UDB (OS/390 版) サブシス テム内にある場合	111
ステップ 37 -- 永続ライブラリーの受け入れ	112
ステップ 38 -- セキュリティーの終結処置	113

第10章 QMF NLF の計画およびインストー ル	115
プロファイル表と NLF	115
QMF NLF の計画	116
ハードウェアとプログラム製品の要件	116
SMP/E 要件	116
QMF NLF 用の SMP/E データ・セッ ト	116
QMF NLF の配布ライブラリー	117

QMF NLF のターゲット・ライブラリ ー	117
QMF NLF ユーザー・データ・セット	118
IBM ソフトウェア配布 (ISD) テープ	118
FMID	118
インストール・プロセス	119
予備 : Program Directory を読み NLF ワ ークシートに記入する	125
ステップ 1 -- QMF NLF インストール・ パラメーターの指定	128
準備	128
実行	128
ステップ 2 -- ジョブの調整	136
QMF NLF のインストール	137
ステップ 3 -- フォアグラウンドでの QMF NLF のインストール	137
DB2 権限	137
ステップ 4 ~ 8 -- ジョブの手動による 実行依頼	138
ステップ 4 -- QMF パネルのインストー ル	138
準備	138
実行	138
ジョブの再実行	138
ステップ 5 -- NLF/GDDM マップ・グル ープのインストール	138
準備	139
実行	139
ジョブの再実行	139
ステップ 6 -- REXX EXEC または CLIST レコードの変換	139
ステップ 6A -- QMF REXX EXEC レ コードの変換: 固定長から可変長に	139
ステップ 6B -- QMF CLIST レコード の変換: 固定長から可変長に	141
ステップ 7A -- QMF 制御表の更新	142
サブステップ 7Aa -- 前の QMF NLF リリースがインストールされていない 場合	142
サブステップ 7Ab -- QMF NLF 2.2 ま たは 2.3 がインストールされている場 合	144
サブステップ 7Ac -- QMF NLF 2.4 が インストールされている場合	145
サブステップ 7Ad -- QMF NLF 3.1 が インストールされている場合	146

ステップ 7B と 7C -- QMF NLF サンプル表の設定	147	ステップ 11 -- QMF NLF の DB2 (AS/400 版) サーバー用調整 (オプション).	158
ステップ 7B -- 以前の QMF NLF サンプル表の削除	147	DB2 (AS/400 版) サーバーでの QMF NLF 制御表の更新の作成	158
準備	147	DB2 (AS/400 版) サーバーでの QMF NLF サンプル表の作成	159
DB2 許可	148	ステップ 12 -- バッチ IVP 実行のための NLF バッチ・ジョブのセットアップ (オプション).	159
実行	148	ステップ 13 -- QMF 対話モードでの IVP の実行	159
ジョブの再実行	148	ステップ 14 -- 各国語サンプル照会およびプロシージャのインストール	160
ステップ 7C -- NLF サンプル表の作成	148	ステップ 14A -- 既存のサンプル照会およびプロシージャの削除	160
準備	148	ステップ 14B -- 各国語サンプル照会およびプロシージャのインストール	161
DB2 権限	149	ステップ 15 -- バッチ・モード IVP の実行 (オプション).	162
実行	149	ステップ 16 -- ポストインストールの最終処置	162
ジョブの再実行	149	ステップ 17 -- 永続ライブラリーの受け入れ	163
ステップ 8 -- NLF/QMF の TSO 用調整	150	ステップ 18 -- CDS 間環境の作成	163
ステップ 9 -- NLF/QMF の CICS 用調整	151		
ステップ 9A -- DB2 RCT への NLF/QMF トランザクション ID の追加	151		
ステップ 9B -- DFHEAI および DFHEAI0 を使用したリンク・エディット	152		
サブステップ 9Ba -- CICS コマンド・インターフェース・モジュールを使用した QMF のリンク・エディット	152		
サブステップ 9Bb -- QMF 提供の管理プログラムの変換、アセンブル、およびリンク・エディット	153		
ステップ 9C -- ADMF データ・セットへの QMF/GDDM マップ・セットのロード	153		
ステップ 9Da -- CICS 管理テーブルの更新 (CICS V2 のみ)	154		
ステップ 9Db -- CICS 管理テーブルの更新 (CICS ESA のみ)	154		
ステップ 9E -- CICS 領域のジョブ・ストリームの更新	155		
ステップ 9F -- IVP の実行	155		
ステップ 10 -- QMF NLF の Workstation Database Server 用調整 (オプション)	155		
ステップ 10A -- Workstation Database Server での QMF NLF 制御表の作成	156		
ステップ 10B -- Workstation Database Server での QMF NLF サンプル表の作成	156		
Workstation Database Server からの QMF NLF の削除	157		
		第11章 リモート・サーバーでの QMF 7.1 パッケージのバインド	165

第1章 QMF 紹介およびインストール・プロセス

この章では、QMF ホスト製品を紹介します。この章では、また、QMF を DB2 ユニバーサル・データベース (OS/390 版)(DB2 UDB (OS/390 版)、 DB2 (VM または VSE 版)(DB2 (VSE または VM 版))、および DB2 (AS/400 版) の各データベースに接続する方法、ならびに QMF のインストール方法の概要を示します。

QMF 紹介

QMF は、ユーザーがデータベースにアクセスできるようにし、データベースに含まれているデータに基づいて報告書または図表を作成できるようにする、照会報告書作成プログラムです。

QMF は MVS/ エンタープライズ・システム体系 (MVS/ESA) のもとで実行され、主に DB2 UDB (OS/390 版) を介してデータにアクセスします。QMF は、顧客情報管理システム (CICS) の制御下でタイム・シェアリング・オプション拡張機能 (TSO/E) とオンライン・トランザクション・マネージャーの両方を使用して実行されます。CICS のユーザーは、CICS 内から QMF を開始し、CICS/DB2 接続機構を介してデータにアクセスすることができます。

ホスト環境では、QMF は、IBM 図形データ表示管理プログラム (GDDM) を使用してパネルを表示します。ただし、ユーザーは、対話式システム生産性向上機能 (ISPF) を使用してアプリケーション・パネルを表示することができます。図1 は、ホスト指定構成においてこれらの製品がどのように QMF に関係づけられるかを示したものです。

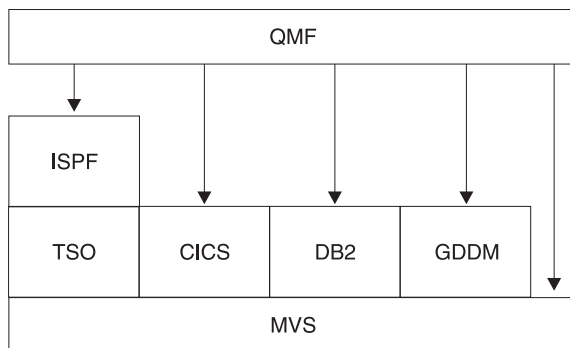


図1. ホスト指定構成における QMF

QMF は、次のオブジェクトを処理します。

データ 表に含まれ、報告書でフォーマット設定された英数字で表現される情報。

照会 ユーザーが必要とするデータおよびユーザーが実行したいアクションを指定します。

書式 取り出したデータを報告書または図表に合わせて調整する方法を記述します。

プロシージャ

1 つのグループとして実行できる 1 つまたは複数の QMF コマンドを含んでいます。

プロファイル

該当のユーザーのセッションを処理する方法に関する情報を含んでいます。

QMF が他のデータベースのデータにアクセスする方法

QMF を使用して、QMF 初期化時にまたは QMF セッション内から、分散ネットワーク内の DB2 UDB (OS/390 版)、DB2 (VSE または VM 版)、DB2 (AS/400 版)、または DB2 ユニバーサル・データベースのいずれかのデータベースに接続することができます。1 つのロケーションへの接続が成功すると、ローカルにデータやオブジェクトにアクセスする場合と同じように、そのデータベースのデータおよび QMF オブジェクトにアクセスすることができます。SQL CONNECT コマンドの詳細については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

QMF は、次の 2 つのデータ・アクセス方式をサポートしています。

- 分散リレーショナル・データベース体系 (DRDA)[™] リモート作業単位
- DB2 UDB (OS/390 版) 間分散作業単位

DRDA は、分散テクノロジーに対する IBM の取り組み方です。DRDA 内には、リモート作業単位、分散作業単位、および分散要求などのさまざまなタイプのサポート機能があります。DRDA 環境では、QMF はリモート作業単位のみをサポートしています。

DB2 UDB (OS/390 版) 間分散作業単位により、DB2 UDB (OS/390 版) に特定の通信方式を使用して他の DB2 UDB (OS/390 版) サブシステムにアクセスすることができます。DB2 UDB (OS/390 版) では、このタイプの接続を システム管理アクセス と呼びます。

この 2 つのタイプのアクセス方式は、作業単位 の定義に基づいています。この作業単位は 1 つの論理トランザクションです。論理トランザクション は一連の SQL ステートメントから構成されており、これにより、操作のすべてが正常に行われるかまたはそのシーケンス全体が正常なものではないと見なされます。

リモート作業単位

このタイプの分散アクセスにより、ユーザー・プログラムまたはアプリケーション・プログラムは、作業単位ごとに 1 つのリモート・ロケーションでデータの読み取りまたは更新を行うことができます。

あるロケーションへ接続されると、ローカル・データベースにアクセスするのと同じように、そのロケーションにあるデータおよび QMF オブジェクトにアクセスすることができます。

DB2 UDB (OS/390 版) 分散データ機能 (DDF) は DB2 UDB (OS/390 版) バージョン 2 リリース 3 以降、DRDA[®] のデータ構造を採用しており、DB2 (VSE または VM 版) はバージョン 7 リリース 1 では DRDA のデータ構造を採用しています。リモート作業単位を使用すると、DB2 UDB (OS/390 版) は、DRDA をインプリメントしている任意のリモート・データベース管理システムのためのサーバーまたはリクエスター (パートナー・システムからのサポートのレベルに応じて決まる) としての働きをすることができます。

始動プログラム・パラメーター DSQSDBNM または QMF CONNECT コマンドを使用して接続先のリモート・ロケーションを指定する場合には、そのデータベースにアクセスするすべての後続の QMF コマンドはそのリモート・ロケーションに送られます。(DDF がインストールされている場合は、CONNECT TO メッセージが QMF ホーム・パネルに表示されます。)

図2 は、リモート作業単位を用いた QMF を図示したものです。

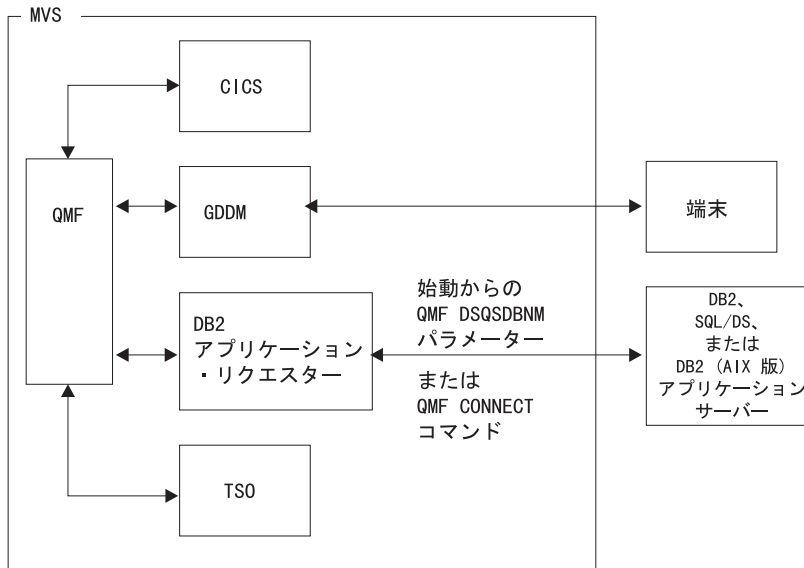


図2. リモート作業単位を用いた QMF

DB2 UDB (OS/390 版) 間分散作業単位

これは分散作業単位の初期の版であり、DB2 UDB (OS/390 版) バージョン 2 リリース 2 からサポートされるようになりました。これにより、DB2 UDB (OS/390 版) 専用の通信方式を使用して他の DB2 UDB (OS/390 版) サブシステムにアクセスすることができます。この方式により、作業単位ごとに 1 つのロケーションに接続し 1 つの照会を実行することができます。DB2 UDB (OS/390 版) 間分散作業単位では、別名または 3 部分名を使用してサブシステムのロケーションを判別し、そのサブシステムに接続します。ただし、QMF には、このタイプのデータ・アクセスをサポートするために少なくとも DB2 UDB (OS/390 版) バージョン 2 リリース 3 またはそれ以降のリリースが必要です。9ページの図3 は、DB2 UDB (OS/390 版) 間アクセス接続を示したものです。

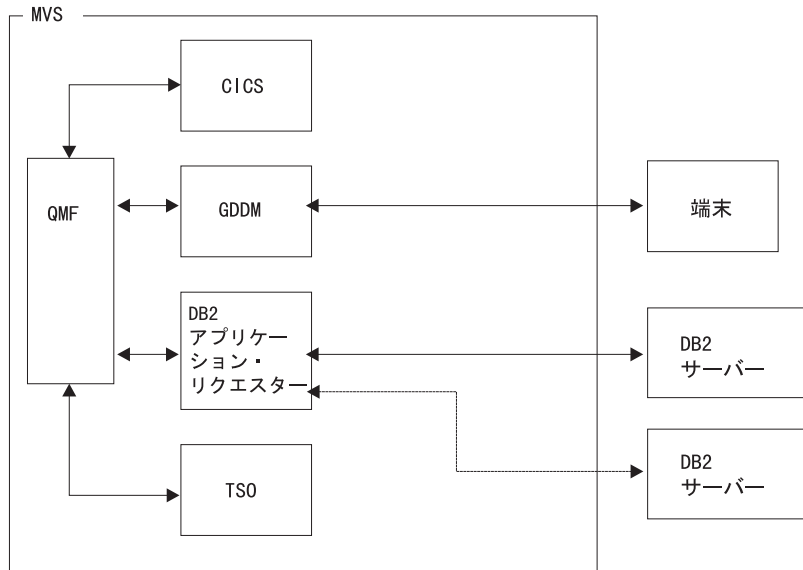


図3. DB2 UDB (OS/390 版) 間接続

データベース・インストール・プロセスの概要

OS/390 での QMF のインストールには、次のオブジェクト・グループが含まれます。

- QMF ターゲット・ライブラリーおよび配布ライブラリー
- QMF アプリケーション・プランおよびパッケージ
- QMF 制御表、カタログ視点、およびサンプル表

QMF のための DB2 UDB (OS/390 版) の要件

QMF は、データベースに対して標準インターフェースを使用する DB2 UDB (OS/390 版) アプリケーション・プログラムです。QMF を使用するためには、QMF を少なくとも 1 つの DB2 UDB (OS/390 版) サブシステムにインストールする必要があります。ユーザーのデータ・ネットワークの設計によっては、QMF を複数の DB2 UDB (OS/390 版) サブシステムにインストールする必要があります。

DB2 UDB (OS/390 版) に関する前提知識

QMF は DB2 UDB (OS/390 版) のアプリケーションであるため、QMF を効率よくインストールするには、DB2 UDB (OS/390 版) のインストールを実行する場合と同じ概念を理解しておく必要があります。たとえば、次のことを理解しておく必要があります。

- CREATE、INSERT、および GRANT の各 SQL ステートメント
QMF のインストール時に、これらのステートメントを使用します。これらのステートメントについては、*DB2 UDB (OS/390 版) SQL 解説書* で詳細に説明されています。
- アプリケーション・プラン、DBRM、パッケージ、および バインド の各用語
これらの用語は、*DB2 UDB (OS390 版) アプリケーション・プログラミングおよび SQL の手引き* で説明されています。
- データベース、表スペース、表、および視点
これらの用語の間での基本的な関係を理解しておく必要があります。これらの用語に関する概念情報については、*DB2 UDB (OS390 版) 管理の手引き 第 2 巻*を参照してください。
- DB2 UDB (OS/390 版) セキュリティー機構
SYSADM および DBADM 権限がどのようなものであるかならびに権限の付与および取り消しの方法を理解しておく必要があります。権限を PUBLIC に付与することの意味も理解しておく必要があります。これらのトピックについては、*DB2 UDB (OS390 版) 管理の手引き 第 2 巻*にある説明を参照してください。
- QMF のインストールを計画している DB2 サブシステムの ID
これらの用語に関する概念情報については、*DB2 UDB (OS390 版) 管理の手引き 第 2 巻*を参照してください。

分散データにアクセスするための QMF のインストール

ユーザーは、下記の用語について習熟しておく必要があります。

- アプリケーション・リクエスター
- アプリケーション・サーバー
- 現行ロケーション (現行サーバー)
- 分散作業単位
- ローカル DB2 UDB (OS/390 版)
- ロケーション名
- リモート作業単位

これらの用語の定義および詳細については、*DB2 UDB (OS390 版) SQL 解説書*を参照してください。

QMF のインストールによって作成された DB2 UDB (OS/390 版) オブジェクト

QMF によってアクセスされる DB2 UDB (OS/390 版) システムには、インストール時に QMF 用に作成されるいくつかの DB2 UDB (OS/390 版) オブジェクト・タイプが含まれます。

QMF を分散データ環境にインストールする計画がない場合、または QMF を DB2 UDB (OS/390 版) 間分散作業単位環境にインストールする計画がある場合には、次のすべてのオブジェクトを、QMF によってアクセスされる各サブシステムにインストールする必要があります。

- QMF インストール計画およびパッケージ
- QMF 制御表
- QMF カタログ視点
- QMF SAVE DATA および IMPORT TABLE コマンド用の表スペース
- QMF サンプル表
- QMF パッケージ
- QMF アプリケーション・プラン

これらのオブジェクト・タイプの詳細については、569ページの『付録B. DB2 に常駐している QMF オブジェクト』を参照してください。

データベース許可 ID Q

Q という DB2 UDB (OS/390 版) 許可 ID は QMF ですべての制御表、サンプル表、およびカタログ視点を所有しますが、QMF のインストールにはこの許可 ID は必要ありません。ただし、この許可 ID がいない場合は、SYSADM 権限が必要です。

ユーザーの権限 (インストーラーとしての) が取り消されると、特権が他の権限によって付与されない限り、インストール・プロセス時に付与された権限も取り消されます。したがって、必要な権限を保持する ID を使用して QMF をインストールする必要があります。

QMF インストール・プロセスのロードマップ

この節では、QMF インストール・オプション・タイプおよび各インストール・オプションのもとで作成される DB2 UDB (OS/390 版) オブジェクトを列挙しています。これらのオブジェクトの詳細については、569ページの『付録 B. DB2 に常駐している QMF オブジェクト』を参照してください。

- 初期インストールまたは移行
 - 予備 : Program Directory を読み、ワークシートに記入する。

- Program Directory の説明に従って SMPE インストールを完了する。
- 全データベース・インストールを開始する。
- 全データベース・インストール
 - ステップ 1 とステップ 2 を実行する。
 - バッチ・モードでインストールするには、ステップ 3 ~ 16 に進む。
 - フォアグラウンド・モードでインストールするには、ステップ 3 を実行する。

このタイプのインストールにより、ローカル DB2 UDB (OS/390 版) には次のものが作成されます。ローカル DB2 UDB (OS/390 版) は、QMF アプリケーション・プランが結合されるサブシステムです。

- QMF ターゲット・ライブラリーおよび配布ライブラリー
- 2 つのインストール・パッケージ
- 1 つの QMF インストール・プラン
- QMF 制御表
- QMF カタログ視点
- QMF SAVE DATA および IMPORT TABLE コマンド用の表スペース
- QMF サンプル表
- QMF アプリケーション・パッケージ
- 1 つの QMF アプリケーション・プラン

次の場合は、全インストールを実行する必要があります。

- QMF の初期インストールである場合。
- QMF の唯一のインストールである場合。
- QMF から複数のローカル DB2 UDB (OS/390 版) サブシステムにアクセスする必要がある場合。追加のローカル DB2 UDB (OS/390 版) インストールごとにこのタイプのインストールを行ってください。
- サーバー・データベース・インストール
 - ステップ 1 とステップ 2 を実行する。
 - バッチ・モードでインストールするには、ステップ 8 ~ 16 に進む。
 - フォアグラウンド・モードでインストールするには、ステップ 3 を実行する。

このタイプのインストールにより、次のものが作成されます。

- DB2 UDB (OS/390 版) アプリケーション・サーバーでの 2 つのインストール・パッケージ
- ローカル DB2 UDB (OS/390 版) サブシステムでの 1 つの QMF インストール計画 (CURRENTSERVER ロケーションとしてアプリケーション・サーバー名を使用して)

- QMF 制御表
- QMF カタログ視点
- QMF SAVE DATA および IMPORT TABLE コマンド用の表スペース
- QMF サンプル表
- QMF アプリケーション・パッケージ

別の DB2 UDB (OS/390 版) データベースに定義されているデータにアクセスする計画がある場合には、サーバー・データベース・インストールを行う必要があります。ユーザーのローカル DB2 UDB (OS/390 版) サブシステムからアクセスできる任意の DB2 UDB (OS/390 版) サブシステムで、このタイプのインストールを行うことができます。該当のローカル DB2 UDB (OS/390 版) サブシステムからインストールを行ってください。

ワークステーション・データ・サーバー用の QMF のインストールについては、91ページの『第7章 QMF の ワークステーション・データ・サーバー用調整』を参照してください。

- リクエスター・データベース・インストール
 - ステップ 1 とステップ 2 を実行する。
 - バッチ・モードでインストールするには、ステップ 8、ステップ 15、およびステップ 16 を実行する。
 - フォアグラウンド・モードでインストールするには、ステップ 3 を実行する。

このタイプのインストールにより、次のものが作成されます。

- ローカル DB2 UDB (OS/390 版) サブシステムでの 2 つのインストール・パッケージ
 - 1 つの QMF インストール・プラン
 - 1 つの QMF パッケージ (DSQIRDBR)
 - 1 つの QMF アプリケーション・プラン
 - QMF ランタイム・ライブラリー

リモート作業単位を使用して他のデータベースにアクセスする必要がある場合で、QMF の実行時にこの DB2 UDB (OS/390 版) サブシステムをローカル DB2 UDB (OS/390 版) サブシステムとして使用する計画がある場合には、リクエスター・データベース・インストールを行う必要があります。QMF ランタイム・ライブラリーがインストールされている同じ OS/390 システムで定義されている DB2 UDB (OS/390 版) サブシステムで、QMF アプリケーション・リクエスターを設定することができます。

QMF をリモート作業単位用に設定する

QMF からリモート作業単位を使用するために DB2 UDB (OS/390 版) サブシステムを設定する最も簡単な方法は、まず全 QMF インストールを行ってから、同じ OS/390 システム上の追加の DB2 UDB (OS/390 版) サブシステムごとに全データベース・インストールを行うことです。DB2 サブシステム (リモート作業単位をサポートする) が全データベース・インストールを受け入れた後で、ユーザーは、そのサブシステムを QMF 用のアプリケーション・リクエスターまたはアプリケーション・サーバーとして使用することができます。ただし、特定の DB2 UDB (OS/390 版) サブシステムのみをアプリケーション・リクエスターまたはアプリケーション・サーバーとして使用したい場合には、必要なオブジェクトのみをインストールすることができます。

考慮事項: QMF CONNECT コマンドは、接続中の QMF のインスタンスが同じリリースである場合にのみ機能します。

リモート作業単位を用いたデータのアクセス

QMF からリモート・ロケーションに接続するために DSQSDBNM 始動プログラム・パラメーターまたは QMF CONNECT コマンド (この両方はリモート作業単位アクセスを暗黙指定する) を使用したい場合には、まずどの DB2 UDB for OS/390 サブシステムが QMF 用のアプリケーション・リクエスターおよびアプリケーション・サーバーとして機能するかを判別する必要があります。

- QMF 用のアプリケーション・リクエスターとしてのみ機能するサブシステムには、そのサブシステムに結合される QMF プラン、QMF パッケージ (DSQIRDBR) の 1 つ、および QMF インストール・プログラムの 1 つが必要です。これらのオブジェクトは、リクエスター・データベース・インストール・オプションまたは全データベース・インストール・オプションによって作成されます。
- QMF 用のアプリケーション・サーバーとして機能するサブシステムには、QMF パッケージ、インストール・プログラム、制御表、カタログ視点、SAVE DATA 用の表スペース、およびサンプル表が必要です。全データベース・インストール・オプションまたはサーバー・データベース・インストール・オプションを使用して、これらのオブジェクトを作成してください。
- アプリケーション・リクエスターとアプリケーション・サーバーの両方として機能するサブシステムには、アプリケーション・サーバーのみの場合と同じオブジェクトが必要です。全データベース・インストール・オプションを使用して、これらのオブジェクトを作成してください。

DB2 UDB (OS/390 版) 間分散作業単位用の QMF の設定

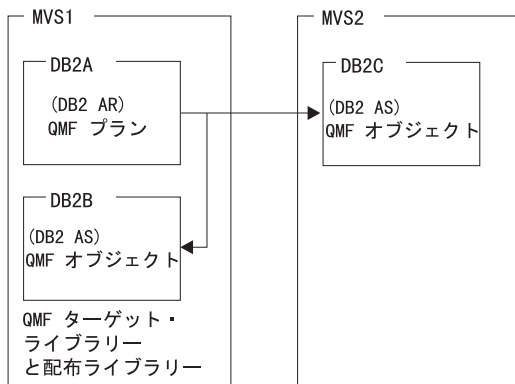
リモート・データへの DB2 UDB (OS/390 版) 間分散作業単位アクセスは通常、QMF に対して透過的に行われます。したがって、ユーザーが選択するインストール・プロセスは、リモート作業単位を併用するかどうかに応じて決まります。リモート作業単位を使用したい場合には、QMF を適切なサブシステムにインストールしてください。リモート作業単位と DB2 UDB (OS/390 版) 間分散作業単位の両方を使用する場合、3 部分名を使用してアクセスできるロケーションは、現行サーバーがアクセスできるロケーション (現行サーバーが DB2 のロケーションにある場合) です。

例

次の例は、リクエスター・データベース・インストール・オプションとサーバー・データベース・インストール・オプションを使用してリモート作業単位環境で QMF をインストールできる 1 つの方法を示したものです。

システム構成例および要件

- OS/390 オペレーティング・システム MVS1 には、2 つの DB2 UDB (OS/390 版) (バージョン 2 リリース 3) サブシステム (DB2A と DB2B) があります。このシステムは TSO システムであり、DB2A はアプリケーション・リクエスターであり、DB2B はアプリケーション・サーバーです。
- OS/390 オペレーティング・システム MVS2 には、1 つの DB2 UDB (OS/390 版) (バージョン 2 リリース 3) サブシステム (DB2C) があります。このシステムは BATCH であり、DB2C はアプリケーション・サーバーです。DB2C は、MVS1 上の各 TSO ユーザーにアクセスすることができます。
- QMF は、アプリケーション・リクエスターとしての DB2A に、かつアプリケーション・サーバーとしての DB2B および DB2C にインストールする必要があります。DB2A の許可ユーザーは、別の OS/390 オペレーティング・システムにログオンすることなしに、DB2B と DB2C に保管されているデータにアクセスすることができます。



QMF オブジェクトは、制御表、サンプル表、視点、およびアプリケーション・パッケージです。

サンプル構成のインストール・シーケンス:

1. MVS1 で、QMF のターゲット・ライブラリーと配布ライブラリーをインストールする。
2. MVS1 で、リクエスター・データベース・インストール・オプションを使用して QMF を DB2A にインストールし、QMF ランタイム・ライブラリーをカスタマイズする。
3. MVS1 で、サーバー・データベース・インストール・オプションを使用して QMF を DB2B にインストールする。DB2A をローカル DB2 として、DB2B をアプリケーション・サーバーとして使用します。
4. MVS1 で、サーバー・データベース・インストール・オプションを使用して QMF を DB2C にインストールする。DB2A をローカル DB2 UDB (OS/390 版) として、DB2B をアプリケーション・サーバーとして使用します。リモート・インストールが MVS1 で実行されるので、MVS2 にログオンする必要はありません。

第2章 QMF の計画

この章では、QMF のインストールおよび実行に必要なハードウェア、プログラム製品、および直接アクセス記憶装置 (DASD) について説明します。この章には、インストール時の参照が簡単に行えるようにするための計画ワークシートが記載されています。

ハードウェア要件

QMF は、当該オペレーティング・システムによってサポートされるプロセッサ上で稼働します。QMF は、OS/390 および DB2 UDB (OS/390 版) によってサポートされるすべての DASD 装置および図形データ表示管理プログラム (GDDM) によってサポートされるすべての端末にアクセスすることができます。

各国語文字セットを使用したい場合には、各国語文字をサポートするワークステーションが必要になります。

前提条件のソフトウェア

以下の表は、QMF (OS/390 版) バージョン 7 のサポートに必要な最低限度のリリース・レベルのプログラム製品をリストしています。QMF バージョン 7 の発表時に使用可能でなかった後のリリースは、特に断りのない限りサポートされません。

表 2. QMF (OS/390 版) バージョン 7 の前提条件ソフトウェア

QMF (OS/390 版) の前提条件ソフトウェア

必須ソフトウェア	バージョンおよびリリース	番号
MVS/ESA SP JES2 または	バージョン 4 リリース 2	5695-047
MVS/ESA SP JES3	バージョン 4 リリース 2	5695-048
MVS/ データ機能プロダクト (DFP)	バージョン 3 リリース 1	5665-XA3
DATABASE 2(DB2)	バージョン 3 リリース 1	5685-DB2
GDDM-MVS	バージョン 2 リリース 3	5665-356

インストールのみ :

表 2. QMF (OS/390 版) バージョン 7 の前提条件ソフトウェア (続き)

QMF (OS/390 版) の前提条件ソフトウェア

必須ソフトウェア	バージョンおよびリリース	番号
対話式システム生産性向上機能 (ISPF)(MVS 版)	バージョン 3 リリース 5	5685-504
拡張システム修正変更プログラム (SMP/E)	バージョン 1 リリース 8	5668-949
TSO 環境 :		
TSO/ 拡張機能 (TSO/E)	バージョン 2 リリース 4	5685-025
CICS 環境 :		
CICS/ESA または	バージョン 4 リリース 1.1	5655-018
CICS/ESA または	バージョン 3 リリース 1	5683-083
CICS/MVS	バージョン 2 リリース 1.1	5665-356

表 2. QMF (OS/390 版) バージョン 7 の前提条件ソフトウェア

QMF (OS/390 版) の前提条件ソフトウェア

必須ソフトウェア	バージョンおよびリリース	番号
MVS/ESA SP JES2 または	バージョン 5 リリース 2	5645-001
MVS/ESA SP JES3	バージョン 5 リリース 2.1	5645-001
DFSMSdfp	バージョン 1 リリース 3	5645-001
DATABASE2 (DB2)	バージョン 3 リリース 1	5685-DB2
GDDM/MVS	バージョン 3 リリース 1.1	5645-001
インストールのみ :		
対話式システム生産性向上機能 (ISPF)(MVS 版)	バージョン 4 リリース 2.0	5645-001
拡張システム修正変更プログラム (SMP/E)	バージョン 1 リリース 8.1	5645-001
TSO 環境 :		
TSO/ 拡張機能 (TSO/E)	バージョン 2 リリース 5	5645-001
CICS 環境 :		
CICS/ESA または	バージョン 3 リリース 3	5683-083
CICS/ESA	バージョン 4 リリース 1.1	5655-018

表 2. QMF (OS/390 版) バージョン 7 の前提条件ソフトウェア

QMF (MVS/XA 版) の前提条件ソフトウェア

必須ソフトウェア	バージョンおよびリリース	番号
MVS/SP-JES2 または	バージョン 2 リリース 2	5740-XC6

表2. QMF (OS/390 版) バージョン 7 の前提条件ソフトウェア (続き)

QMF (MVS/XA 版) の前提条件ソフトウェア

必須ソフトウェア	バージョンおよびリリース	番号
MVS/SP-JES3	バージョン 2 リリース 2.1	5665-291
MVS/XA データ機能プロダクト (DFP)	バージョン 2 リリース 1	5665-XA2
GDDM/MVS	バージョン 2 リリース 1	5665-356
インストールのみ :		
対話式システム生産性向上機能 (ISPF)(MVS 版)	バージョン 2 リリース 3	5665-319
拡張システム修正変更プログラム (SMP/E)	バージョン 1 リリース 8	5668-949
TSO 環境 :		
TSO/ 拡張機能 (TSO/E)	バージョン 2 リリース 1	5685-025
CICS 環境 :		
CICS/MVS	バージョン 2 リリース 1	5665-356

次の表は、QMF (OS/390 版) バージョン 7 のオプション機能のサポートに必要な最小限度のリリース・レベルのプログラム製品をリストしています。QMF バージョン 7 の発表時に使用可能でなかった後のリリースは、特に断りのない限りサポートされません。

表3. QMF (OS/390 版) バージョン 7 用のオプション機能の前提条件ソフトウェア

製品	バージョンおよびリリース	番号
ISPF 関連機能 (QMF 文書インターフェース、QMF EDIT コマンド用のデフォルト編集プログラム、印刷報告書表示アプリケーション (DPRE)、ISPF コマンド、および DXT/ エンド・ユーザー・ダイアログのブリッジ・サポート)		
ISPF (MVS 版)	バージョン 3 リリース 5	5685-054
プロシージャー内のロジック、報告書の計算および条件、書式列定義、および IBM 提供のコマンド同義語 (DPRE、ISPF、BATCH、LAYOUT) の使用		
TSO/ 拡張機能 (TSO/E)	バージョン 2 リリース 4	5685-025
図表 (対話式図表ユーティリティー)		
GDDM 図形表示機能 (PGF)	バージョン 2 リリース 1.1	5668-812
QMF 文書インターフェース。以下の編集プログラムが必要です。		
パーソナル・サービス /TSO (PS/TSO)	リリース 1	5665-346

QMF の計画

表3. QMF (OS/390 版) バージョン 7 用のオプション機能の前提条件ソフトウェア (続き)

製品	バージョンおよびリリース	番号
データ抽出プログラム (DXT)、エンド・ユーザー・ダイアログおよび QMF EXTRACT コマンド		
データ抽出プログラム (DXT)	バージョン 2 リリース 5	5668-788
QMF 高性能オプション (HPO) :		
ISPF (MVS 版)	バージョン 3	5685-054
QMF (Windows 版) :		
Microsoft Windows** または	バージョン 3 リリース 1	
Microsoft Windows** (ワークグループ用) または	バージョン 3 リリース 1 または リリース 1.1	
Microsoft Windows 95 または		
Microsoft Windows NT		
IBM APPC ネットワーク・サービス (Windows 版) または	バージョン 1	
Microsoft SNA サーバーまたは	バージョン 2	
Novell Netware (SAA 用) または	バージョン 2	
Attachmate EXTRA! APPC クライアント	バージョン 3 リリース 11	
呼び出し可能インターフェースに書き込む呼び出し可能インターフェース・プログラムは、以下で書くことができます。		
IBM C/370 コンパイラーおよび	バージョン 2	5688-187
C/370 ライブラリー	バージョン 2	5688-188
IBM HLASM	バージョン 1 リリース 1 または リリース 2	5696-234
VS COBOL II コンパイラーおよびライブラリー	バージョン 1 リリース 4	5688-023
VS COBOL II コンパイラー、ライブラリーおよびデバッグ機能	バージョン 1 リリース 4	5668-958
AD/Cycle COBOL/370	バージョン 1 リリース 1	5688-197
IBM COBOL (MVS および VM 版)	バージョン 1 リリース 2	5688-197
AD/Cycle C/370 コンパイラー	バージョン 1 リリース 1	5688-216

表 3. QMF (OS/390 版) バージョン 7 用のオプション機能の前提条件ソフトウェア (続き)

製品	バージョンおよびリリース	番号
VS FORTRAN (REXX および FORTRAN 用の SAA 呼び出し可能インターフェースは、QMF/CICS 環境ではサポートされません。)	バージョン 2 リリース 5	5668-806
OS PL/I	バージョン 2 リリース 3	5668-909
IBM PL/I (MVS および VM 版)	バージョン 1 リリース 1.1	5688-265
REXX : TSO 拡張機能 (TSO/E) (REXX および FORTRAN 用の SAA 呼び出し可能インターフェースは、QMF/CICS 環境ではサポートされません。)	バージョン 2 リリース 1	5685-025
REXX (REXX および FORTRAN 用の SAA 呼び出し可能インターフェースは、QMF/CICS 環境ではサポートされません。)	VM/ESA では :	
アセンブラー H	バージョン 2 リリース 1	5668-962
IBM C/C++ (MVS/ESA 版) (言語環境プログラム (MVS および VM 版) (MVS フィーチャー) に関連します。)	バージョン 3	5655-121
ユーザー編集ルーチンは、次の言語で作成することができます。		
IBM HLASM	バージョン 1	5696-234
VS COBOL II コンパイラーおよびライブラリー	バージョン 1 リリース 4	5688-023
COBOL/370 コンパイラーおよびライブラリー	バージョン 1 リリース 1	5688-197
IBM COBOL (MVS および VM 版)	バージョン 1 リリース 2	5688-197
VS COBOL II コンパイラーおよびライブラリー	バージョン 1 リリース 3.1	5688-023
VS COBOL II コンパイラー、ライブラリーおよびデバッグ機能	バージョン 1 リリース 3.1	5668-958
OS PL/I	バージョン 2 リリース 3	5668-909

表3. QMF (OS/390 版) バージョン 7 用のオプション機能の前提条件ソフトウェア (続き)

製品	バージョンおよびリリース	番号
IBM PL/I (MVS および VM 版)	バージョン 1 リリース 1.1	5688-265
アセンブラー H または標準アセンブラー	バージョン 2 リリース 1	5668-962
管理プログラム出口ルーチン		
IBM HLASM	バージョン 1	5696-234
アセンブラー H または標準アセンブラー	バージョン 2 リリース 1	5668-962
リモート作業単位 (OS/390)		
リモート DB2 (OS/390 版) DRDA アプリケーション・サーバーへの接続 :		
ローカル DB2 (OS/390 版) ロケーションでは		
DB2 (MVS 版)	バージョン 3 リリース 1	5685-DB2
QMF (OS/390 版)	バージョン 7	5675-DB2
リモート DB2 データベースでは :		
DB2 (MVS 版)	バージョン 3 リリース 1	5685-DB2
QMF (OS/390 版)	バージョン 7	5675-DB2
リモート DB2 (VM 版) DRDA アプリケーション・サーバーへの接続 :		
MVS/ESA ロケーションのローカル DB2 では :		
DB2 (MVS 版)	バージョン 3 リリース 1	5685-DB2
QMF (OS/390 版)	バージョン 7	5675-DB2
リモート DB2 (VM/ESA または VSE/ESA 版) データベースでは :		
SQL/DS (VM 版)	バージョン 3 リリース 5	5688-103
SQL/DS (VM 版)	バージョン 3 リリース 3	5706-255
リモート DB2 (VSE 版) DRDA アプリケーション・サーバーへの接続 :		
ローカル DB2 (OS/390 版) ロケーションで		
DB2 (MVS 版)	バージョン 3 リリース 1	5685-DB2
QMF (OS/390 版)	バージョン 7	5675-DB2

表 3. QMF (OS/390 版) バージョン 7 用のオプション機能の前提条件ソフトウェア (続き)

製品	バージョンおよびリリース	番号
リモート DB2 (VM または VSE 版) データベースでは		
SQL/DS	バージョン 3 リリース 5	5688-103
DB2 (VSE 版)	バージョン 6	5648-061
DB2 PE、DataJoiner、Common Server への接続 :		
ローカル DB2 (OS/390 版) ロケーションで		
DB2 (MVS 版)	バージョン 3 リリース 1 (PTF UP75959 および PTF UN54601 が必要)	5685-DB2
QMF (OS/390 版)	バージョン 7	5675-DB2
APPC 通信用に構成されたリモート・データベースで :		
DB2 Parallel Edition (AIX 版) または	バージョン 1 リリース 2	5765-328
DataJoiner (AIX 版) または	バージョン 1 リリース 2	84H1212
DB2 (Windows NT 版)	バージョン 2 リリース 1	53H7474
DB2 (OS/2 版) または	バージョン 2 リリース 1	41H2114
DB2 (AIX 版) または	バージョン 2 リリース 1	41H2128
DB2 (HP-UX 版) または	バージョン 2 リリース 1	10H2366
DB2 (Solaris 版) または	バージョン 2 リリース 1	
DB2 (SCO OpenServer 用) または	バージョン 2 リリース 1	79H5359
DB2 (SINIX 版)	バージョン 2 リリース 1	79H4133

リモート作業単位をサポートするために必要な製品

リモート作業単位 (RUW) サポートは、QMF が稼働するすべての環境で使用できるわけではありません。たとえば、VSE/ESA で QMF を実行しているとき、ユーザーは他のロケーションへ接続することはできません。ただし、VSE DB2 データベースに保管されている QMF オブジェクトへは、分散リレーショナル・データベース体系 (DRDA) ネットワーク内の他の QMF リクエスターがアクセスすることができます。RUW がユーザーの操作環境でサポートされているかいないかを調べるためには、使用しているデータベースのドキュメンテーションを参照してください。

記憶域要件の計画

QMF の記憶域要件は、次のとおりです。

- 31 ビットのアドレッシング・モードで実行できる QMF モジュールには、2.8 MB が必要です。
- 24 ビットのアドレッシング・モードで実行する必要がある QMF モジュールには、52 KB が必要です。
- QMF の照会の実行および QMF 報告書データの保持にユーザーが必要とする最小限記憶域は、0.5 から 1.0 MB の範囲です。ユーザー独自の記憶域所要量は、報告書のサイズおよび使用する報告書フォーマット設定オプションに応じて大きくなる可能性があります。

たとえば、標準 TSO 環境で ISPF および GDDM を使用して実行する場合、記憶域は約 6.0 MB 必要です。

ISPF および GDDM をページ可能リンク・パック域 (PLPA) に入れることにより、領域のサイズを小さくすることもできます。これに応じて、共通域が増大します。

パフォーマンスを向上させるためのモジュールの移動

インストール後には、ライブラリー QMF710.SDSQLOAD には QMF プログラムに関するロード・モジュールが含まれます。25ページの表4 は、パフォーマンスを向上させるためにリンク・パック域ライブラリーに移すことができるモジュールを示したものです。

表4. PLPA または EPLPA に置くことができるモジュール

モジュール	説明
DSQQMFE DSQQMF DSQCSUB DSQCTOPX DSQCCI DSQCCISW DSQCBST DSQCELTT DSQCEBLT DSQCIX	QMF は、ユーザーが QMF を呼び出すときにこのセットのモジュールを使用します。DSQCTOPX と DSQCCI は PLPA にのみ入れることができます。
DSQUEDIT DSQUXIA DSQUXIC DSQUXILE DSQUXIP	これらのモジュールはユーザー編集ルーチンに関係づけられます。頻繁に使用する予定がない限り、これらのモジュールをリンク・パック域に移してはなりません。
DSQCIB COBOL DSQCICX C/370 DSQCIA アセンブラー DSQCIFE FORTRAN DSQCIF FORTRAN DSQCIPX PL/1 DSQCIPL PL/1 DSQCIR RPG DSQCIX REXX	QMF 呼び出し可能インターフェースは、このセットのモジュールを使用します。これらのモジュールは再入可能であり、EPLPA に入れることができます。ただし、呼び出し可能インターフェース・モジュールは小さいものであり、通常はユーザーのアプリケーション・モジュールとリンク・エディットされます。
DSQUEGV3	これは管理プログラム・モジュールです。

26ページの表5 は、PLPA または EPLPA に入れることができないモジュールを示したものです。

QMF の計画

表 5. PLPA または EPLPA に置くことができない TSO モジュール

モジュール	説明
DSQCI	QMF は、QMF の呼び出し時にこのモジュールを使用します。
DSQUEGV1	このモジュールは管理プログラム・ルーチンです。
DSQMAPB DSQ0BINS DSQ0BSQL DSQCTO80 DSQCFR80	これらは、QMF インストールおよびサービス更新モジュールです。

CICS の場合

QMF は、同じ CICS アドレス・スペース内に QMF の複数のユーザーが存在する CICS において会話型トランザクションとして稼働します。QMF トランザクションを実行する各ユーザーには、CICS 領域 からの 1.0 MB 以上の記憶域が必要です。24 KB を除く全部の記憶域を 16 MB 境界より上に割り振ることができます。QMF モジュール (最大 2.7 MB) の 1 つのコピーを EPLPA にまたは 16 MB を超える CICS 領域内に入れることができます。また、52 KB を PLPA にまたは 16 MB 以下の CICS 領域内に入れることができます。

SMP/E 記憶域の見積もり

拡張システム修正変更プログラム (SMP/E) は、QMF のインストールに使用する基本ツールです。SMP/E を使用して、2 つのタイプのライブラリーへのインストールを行います。

- ターゲット・ライブラリー。実行システムを作成する実行可能コードを含みます。
- 配布ライブラリー。すべてのシステム・エレメントのマスター・コピーを含みます。

SMP/E データ・セット用の推定 DASD スペース (シリンダー数) は、表6 に示されています。

表 6. SMP/E データ・セット用の DASD スペース

DDname	3380	3390	9345
SMPCDS	1	1	1
SMPCSI	8	8	8
SMPLOG	1	1	1
SMPMTS	1	1	1

表6. SMP/E データ・セット用の DASD スペース (続き)

DDname	3380	3390	9345
SMPPTS	1	1	1
SMPSTS	1	1	1

配布ライブラリー用のスペースの見積もり

QMF 配布ライブラリーおよびその見積もり DASD スペース (トラック単位で) は、表7 に示されています。

表7. QMF 配布ライブラリー用の DASD スペース

DSNAME	内容	3380	3390	9345
QMF710.ADSQOBJ	QMF オブジェクト・モジュール	11	11	9
QMF710.ADSQACE	QMF インストール・プロシージャ	15	15	13
QMF710.ADSQDBMD	データベース要求モジュール	1	1	1
QMF710.ADSQPMSE	QMF ISPF パネル	1	1	1

ターゲット・ライブラリーのサイズの見積もり

表8 は、ターゲット・ライブラリー用の必須 DASD スペース (シリンダー数) の見積もりを示したものです。

表8. QMF ターゲット・ライブラリー用の DASD スペース

DSNAME	内容	3380	3390	9345
QMF710.SDSQLOAD	QMF ロード・モジュール	8	8	7
QMF710.SDSQSAPE	IVP、サンプル照会	17	17	15
QMF710.SDSQDBRM	QMF DBRM	1	1	1
QMF710.SDSQPLBE	QMF 用の ISPF パネル	1	1	1
QMF710.SDSQCLTE	QMF CLIST サンプル	2	2	2
QMF710.SDSQSLBE	サンプル ISPF スケルトン	1	1	1
QMF710.SDSQMLBE	サンプル ISPF メッセージ	1	1	1
QMF710.SDSQEXCE	TSO/E REXX プロシージャ	1	1	1
QMF710.SDSQUSRE	サンプル・ユーザー出口ルーチン	1	1	1

スペースを各ライブラリーに割り振った後で、SMP/E を使用して QMF をインストールすることができます。

複数のリリースのための計画

QMF は SMP/E を使用して製品をインストールしますが、SMP/E では、ある製品について複数のリリースを同じ SMP/E データ・セット内で保持することはできません。そのため、次のいずれかを実行する必要があります。

- QMF バージョン 7 リリース 1 用にもう一つ SMP/E データ・セットを作成する。
- QMF のバックアップ・コピーを作成する。バックアップ・コピーには次のものが含まれている必要があります。
 - QMF 配布ライブラリーおよびターゲット・ライブラリー
 - SMP/E データ・セット

QMF のバックアップ・コピーを作成せずに同じターゲット・ライブラリー、配布ライブラリー、および SMP/E ライブラリーへのインストールを試みると、SMP/E は、インストール時に前の QMF リリース情報を SMP/E から削除します。移行時に、前の QMF リリースに戻るかまたは前の QMF リリースを実行するかのいずれかの問題が生じる可能性があります。

ユーザー・データ・セット用のスペースの見積もり

QMF ユーザー・ライブラリーに必要な推定 DASD スペース (シリンダー数) は、表9 に示されています。

表9. QMF ユーザー・データ・セット用の DASD スペース

DSNAME	内容	3380	3390	9345
QMF710.DSQMAPE	GDDM マップ・グループ・ファイル	1	1	1
QMF710.DSQCHART	GDDM サンプル図表ファイル	1	1	1
QMF710.DSQUCFRM	GDDM/CICS サンプル図表書式ファイル (VSAM フォーマットで拡張された)	1	1	1
QMF710.DSQPVARE	QMF メッセージ・ヘルプ・パネル (順次フォーマットで拡張された)	なし	6	6
QMF710.DSQPNLE	QMF メッセージ・ヘルプ・パネル (VSAM フォーマットで拡張された)	なし	10	9
QMF710.GDDM.ADMFGDDM/CICS	データ・セット (VSAM フォーマットでの)	1	1	1

Program Directory の読み取りおよびサービスの適用

インストール・プロセスを開始する前に、補足データを得るために *QMF Program Directory* を参照してください。この資料は、QMF テープと一緒に出荷用カートンに同梱されています。*Program Directory* は QMF の各リリース間で更新されるので、この資料には、本書に対する変更だけでなく有用な情報 (プログラム一時修正 (PTF) およびプログラム診断依頼書 (APAR) の記述を含む) も含まれています。

使用システムのサービス・レベルが現行のものであるか確かめてください。QMF の最新の PTF およびその前提となる製品を要求するには、IBM ソフトウェア・サービス・サポート (Software Service Support) に連絡するか、あるいは米国では IBMLink™ (ServiceLink)、ヨーロッパでは EMEA DIAL を使用してください。さらに、QMF の予防保守計画 (PSP) バケツ (SUBSET: UPGRADE QMF710 のもとでの QMFMVS) を要求してください。PSP バケツには、一般ヒント、HIPER APAR、および文書変更が含まれています。Information/Access または ServiceLink にアクセスする加入者は、情報を直接ダウンロードすることができます。

CICS のもとでの QMF の計画

QMF をインストールする前に、CICS および GDDM のインストール、調整、およびテストを済ませておく必要があります。

QMF のための CICS の調整

QMF は大規模会話型トランザクションであるので、QMF 処理には、平均的な CICS トランザクションよりも時間がかかります。したがって、QMF トランザクション専用の CICS 領域での QMF トランザクション処理を分離することができます。

16 MB 以下の使用可能な記憶域の量に応じて、同じ CICS 領域で QMF を実行できるユーザーの数に上限があります。追加の QMF ユーザーをサポートするには、複数の CICS 領域および複数領域オプションを使用してください。

ある CICS システム (たとえば、端末所有領域) から QMF トランザクション (たとえば、アプリケーション所有領域) を処理するように指定された CICS システムに、QMF トランザクションを経路指定することができます。この経路指定を行う場合は、複数のトランザクション ID または動的トランザクション経路指定を使用してください。両方とも、*CICS/OS390 相互通信の手引き* の資料に説明があります。

QMF のための GDDM の調整

QMF のインストール時には、QMF は GDDM の ADMF ファイルを変更しません。さらに、ユーザーは、CICS に対して GDDM リソース (たとえば、プログラムやトランザクションなど) を定義する必要があります。GDDM をインストールおよび調整する方法の詳細については、バージョン 2.3 の場合は、*GDDM 導入およびシステム管理* を、バージョン 3.1 の場合は *GDDM 計画*、*テストおよび保守* を参照してください。

GDDM 2.3 デフォルト・パラメーターの変更

GDDM バージョン 2.3 を使用する場合は、ADMADFC 外部デフォルト・モジュールの IOSYNCH パラメーターが YES に設定されているか確認してください。

GDDM のためのインストール検査プロシージャ (IVP) の実行

GDDM のために IVP を実行してください。IVP により、QMF のインストールの問題が最小限に抑えられ、QMF をクリーン・システムにインストールできるようになります。

AIX[®] 用の DB2 UDB (OS/390 版) のための QMF の計画

AIX サーバー用の DB2 UDB (OS/390 版) と一緒に作動させるために QMF をカスタマイズするには、ホストとサーバーの両方で若干の変更が必要になります。

OS/390 では、QMF は、DB2 UDB (OS/390 版) の分散データ機能 (DDF) を使用して、AIX データベース用の DB2 UDB (OS/390 版) に常駐する分散データにアクセスします。DB2 UDB (OS/390 版) の DDF は、分散リレーショナル・データベース体系 (DRDA) をサポートする他のデータベース管理システムまたはアプリケーションと交信するための LU 6.2 通信プロトコルを使用する VTAM[®] アプリケーションです。OS/390 の QMF からのデータにアクセスするための分散データベース・システムの接続については、*DB2 UDB (OS/390 版) 管理の手引き*、第 1 巻を参照してください。

DB2 UDB (OS/390 版) では、通信データベース (CDB) 表が、リモート・データベース管理システム間のアクセス制御に使用されます。DB2 UDB (OS/390 版) をサーバーとしてのみ使用したい場合には、CDB を移植する必要はありません。この場合、デフォルト値が使用されます。ただし、リモート・データベースからデータを要求したい場合は、CDB 表を更新する必要があります。これらのトピックについては、*DB2 UDB (OS/390 版) 管理の手引き*、第 2 巻および *DB2 UDB (OS/390 版) SQL 解説書* に説明されています。

AIX サーバー用の DB2 UDB (OS/390 版) では、QMF を該当のデータベースにインストールする前に、CREATE DATABASE コマンドを発行する必要があります。OS/390 DRDA アプリケーション・リクエスター用の DB2 UDB (OS/390 版) と AIX DRDA アプリケーション・サーバー用の DB2 UDB (OS/390 版) との間で APPC 通信が定義され、操作可能であるか検査してください。AIX サーバー用の DB2 UDB (OS/390 版) のインストールの詳細については、AIX サーバー用の DB2 UDB (OS/390 版) に関するインストール指示を参照してください。

OS/390 からのインストール、AIX 用の DB2 UDB (OS/390 版) への QMF オブジェクトのインストール、およびインストールの前提条件の詳細については、91ページの『第7章 QMF の ワークステーション・データ・サーバー用調整』を参照してください。

ワークシートへの記入

表10、32ページの表11、および33ページの表12 は、QMF のインストール時に値を指定する必要があるパラメーターを示したものです。これらの表をワークシートとして使用してください。

表 10. QMF インストール・パラメーター (バージョン 7 ワークシート - その 1)

パラメーター	値
ロケーション名	
ターゲット・ライブラリーの接頭部 (デフォルト = QMF710)	
配布ライブラリーの接頭部 (デフォルト = QMF710)	
ターゲット・ライブラリーのボリューム (デフォルト = xxxxxx)	
配布ライブラリーのボリューム (デフォルト = xxxxxx)	
SMP/E データ・セットの接頭部 (デフォルト = IMSVS)	
ローカル DB2 UDB (OS/390 版) サブシステム ID (デフォルト = DSN)	
ローカル DB2 UDB (OS/390 版) リリース・レベル (デフォルト = V3R1)	
ローカル DB2 UDB (OS/390 版) 出口ライブラリー (デフォルト =DSN710.SDSNEXIT)	

QMF の計画

表 10. QMF インストール・パラメーター (バージョン 7 ワークシート - その 1) (続き)

パラメーター	値
ローカル DB2 UDB (OS/390 版) ロード・ライブラリー (デフォルト =DSN710.SDSNLOAD)	
通信データベースが ローカル DB2 UDB (OS/390 版) 通信データベースに インストールされているか否か	YES または NO
通信データベースがローカル DB2 UDB (OS/390 版) サブシステムにインストールされている場合は、次の情報を収集してください。	
インストールの効力範囲	F (全データベース)、 S (サーバー・データベース)、 または R (リクエスター・ データベース)
データベース・インストールの効力範囲が“S” (サーバー・データベース) でない場合は、次の情報を収集してください。	
QMF ランタイム・ライブラリーをカスタマイズする	YES または NO
QMF アプリケーション・プラン ID (デフォルト = QMF710)	

表 11. QMF インストール・パラメーター (バージョン 7 ワークシート - その 2)

パラメーター	値
データベース・インストールの効力範囲が“S” (サーバー・データベース) の場合は、次の情報を収集してください。	
DB2 UDB (OS/390 版) サーバー・ロケーションがリモート DB2 UDB (OS/390 版) サブシステム内にあるか否か (デフォルト = NO)	YES または NO
データベース・インストールの効力範囲が“F” (全データベース) である場合、またはデータベース・インストールの効力範囲が“S” (サーバー・データベース) であり、そのサーバー・データベースがローカル・サブシステムと同じである場合には、次の情報を収集してください。	
DB2 UDB (OS/390 版) ユーザー・カタログ (ICF) (デフォルト = DSNC7101.USER.CATALOG)	
DB2 UDB (OS/390 版) ユーザー・カタログ・パスワード	
QMF 表スペース・カタログ別名 (デフォルト = QMFDSN)	
QMF 表スペース・カタログ・パスワード (QMF 制御表用)	
QMF 表ボリューム	
DB2 UDB (OS/390 版) デフォルト句読法	, (コンマ) または . (ピリオド)

表 11. QMF インストール・パラメーター (バージョン 7 ワークシート - その 2) (続き)

パラメーター	値
前の QMF レベル (移行インストールのみ)	V2R4、V3R1、V3R1M1、 V3R2、V3R3、または NONE
データベース・インストールの効力範囲が “S” (サーバー・データベース) であり、そのサーバー・データベースがローカル DB2 UDB (OS/390 版) サブシステムと異なる場合には、次の情報を収集してください。	
DB2 UDB (OS/390 版) サーバーのロケーション名	
DB2 UDB (OS/390 版) サーバーが別のオペレーティング・システム内にあるか否か	YES または NO
サーバー用の DB2 UDB (OS/390 版) ユーザー・カタログ (ICF) (デフォルト = DSNC7101.USER.CATALOG)	
DB2 UDB (OS/390 版) ユーザー・カタログ・パスワード	
サーバーでの QMF 表スペース・カタログ別名 (デフォルト = QMFDSN)	
QMF 表スペース・カタログ・パスワード (QMF 制御表用)	
サーバーでの QMF 表ボリューム	
サーバーでの DB2 UDB (OS/390 版) デフォルト句読法	, (コンマ) または . (ピリオド)
サーバーでの前の QMF レベル (移行インストールのみ)	V2R4、V3R1、V3R1M1、 V3R2、V3R3、または NONE

表 12. QMF インストール・パラメーター (バージョン 7 ワークシート - その 3)

パラメーター	1 次	2 次
前の QMF レベルが NONE ではなく、データベース・インストールの効力範囲が “R” ではない場合は、次の情報を収集してください。		

QMF の計画

表 12. QMF インストール・パラメーター (バージョン 7 ワークシート - その 3) (続き)

パラメーター	1 次	2 次
QMF 制御表の表スペース サイズ: (1K 単位) 表スペース名 デフォルトのサイズ (1 次、2 次)		
- Q.OBJECT_DIRECTORY (注を参照)	(200,20)	
- Q.OBJECT_REMARKS	" (200,20)	
- Q.OBJECT_DATA	" (5000,200)	
- Q.PROFILES	" (100,20)	
- Q.ERROR_LOG	" (100,20)	
- Q.COMMAND_SYNONYMS	" (100,20)	
- Q.RESOURCE_TABLE	" (100,20)	
- Q.DSQ_RESERVED	(100,20)	100 20
- SAVE DATA (オプション)	" (100,20)	
表索引 サイズ: (1K 単位) 表索引名 デフォルトのサイズ (1 次、2 次)		
- Q.OBJECT_DIRECTORYX (注を参照)	(100,20)	
- Q.OBJECT_REMARKSX	" (100,20)	
- Q.OBJECT_OBJDATAX	" (100,20)	
- Q.PROFILEX	" (100,20)	
- Q.COMMAND_SYNONYMSX	" (100,20)	
以下の項目を決定してください (該当する場合)。		
QMF ジョブをフォアグラウンドでインストールするか、または JCL ファイルを自身で調整して各ジョブをバッチで実行するか	フォアグラウンドまたはバッチ	
CLIST ライブラリーを固定長にするかまたは可変長にするか	可変長または固定長	
EXEC ライブラリーを固定長にするかまたは可変長にするか	可変長または固定長	
SAVE DATA 表スペースを作成したいか否か	YES または NO	

注: 制御表および索引は、QMF の初期インストールの際にのみ提供されます。

第3章 入力パラメーターの指定

この章では、インストール先システムに特有の入力パラメーターを使用して CLIST をカスタマイズします。次に、ユーザーのパラメーター情報を用いて、各メンバーを更新するジョブを実行します。

この章の以下のステップを実行する前に、まず、*QMF Program Directory* の記載事項に従って、SMP/E を使用して OS/390 の環境に QMF をインストールする必要があります。

ステップ 1 -- QMF インストール・パラメーターの指定

このステップでは、一連の QMF インストール・パネルを介してループします。各パネルでは、31ページの表10 から 33ページの表12 までのワークシートに記入した QMF および DB2® (OS/390 版) の情報を入力するようにプロンプト指示されます。

開始の前に

このステップを開始する前に、次の要件を考慮に入れてください。

1. このステップを実行するためには、ISPF セッションがアクティブになっている必要があります。
2. ISPF セッションのコマンド行から、パネル ID をオンにするために (PANELID) を入力します。
3. デフォルトの QMF ターゲット名 (当初は DSQ1EJAL で指定されている) を変更した場合には、DSQ1EINS、DSQ1EIN1、および DSQ1EIN2 の各 CLIST を変更するか、または 第4章 QMF バッチ・インストール・ジョブ の実行依頼に進む必要があります。

DSQ1EINS の調整部分により、SDSQSAPE および SDSQEXCE データ・セットの各メンバーが変更されます。DSQ1EINS を呼び出す前に、SDSQSAPE および SDSQEXCE のバックアップ・コピーを作成することをお勧めします。バックアップ・コピーは、QMF のインストールが完了した時点で削除することができます。

4. データベース専用のインストール (全データベース・インストール、サーバー・データベース・インストール、またはリクエスター・データベース・インストール) のいずれかを実行する場合は、そのインストールのために使用

入力パラメーターの指定

している QMF ターゲット・ライブラリーを、インストール中に他のデータベースのユーザーが アクセスできない ようにしてください。

5. これらのプロシーチャーを介してループバックする場合には、53ページの『CLIST レコードの変換』インストール・ステップにより SDSQCLTE が FB から VB に変更されることに注意してください。DSQ1EINS を実行するためには、手動で VB を FB に変更してください。

インストール・パネルの開始

1. 次のように入力してください。

```
TSO EXEC 'prefix.SDSQCLTE(DSQ1EINS)' 'QMFPRE(prefix)'
```

ここで、*prefix* はワークシートに記入されている QMF ターゲット・ライブラリーの接頭部です。

この処理により、次のいずれかが生成されます。

- 「QMF インストール -- メインメニュー」。これは 図4 に示されているもので、インストール・パラメーターの指定を完了して保管が済んだ場合に表示されます。
- 「QMF インストール -- ローカル DB2 UDB (OS/390 版) パラメーター」パネル。これは 39ページの図5 に示されているもので、インストール・パラメーターのレコードがない場合に表示されます。

```
DXYEIN00          INSTALL QMF -- MAIN MENU
ISPF Command ==>

Currently working on installation into DB2 UDB for OS/390 subsystem DSN

You can now re-specify the install parameters, tailor the installation
files, install QMF with the tailored files in foreground, quit and run
the tailored install files in batch, or quit and return here later.

ENTER CHOICE HERE      ==>          ("P" - INPUT PARAMETERS,
                                     "T" - TAILOR INSTALL FILES,
                                     "I" - INSTALL IN FOREGROUND,
                                     "X" - EXIT INSTALL DIALOGS)

PRESS:  ENTER to continue   PF01 for help   PF03 to end
```

図4. インストール・メインメニュー

入力パラメーターを 1 回以上正常に入力した後では、ループ方式でインストール・メインメニューに戻ります。メインメニューには、次の 4 つのオプションが用意されます。

P インストール・パラメーター

T インストール・ファイルの調整

QMF に必要なすべてのインストール・データ・セットを調整します。このオプションにより、次のことを行うためにジョブを編集することができます。

- QMF GDDM マップおよびパネル・ファイルのフォーマット設定
- DB2 UDB (OS/390 版) への QMF アプリケーション・プランのバインド
- SAVE DATA 表スペースの作成 (オプション)
- サンプル表の削除 (移行インストールのみ)
- QMF サンプル表のインストール
- QMF プラン ID および DB2 UDB (OS/390 版) サブシステム名の QMF 呼び出し可能インターフェース (REXX EXEC DSQSCMDE) 用調整
- インストール検査プロシージャ (IVP) の設定

SDSQSAPE および SDSQEXCE のファイルを以前に調整しており、その調整済みのファイルを保持したい場合は、**T** を選択する前にそのファイルのバックアップをとってください。これは、入力パラメーターのプロシージャによりその情報が上書きされるためです。このステップについては、48ページの『ステップ 2 -- ジョブの調整』の節で詳しく説明しています。

I フォアグラウンドでのインストール (オプション)

このオプションにより、ジョブをオンライン環境で実行依頼することができます。51ページの『第4章 QMF バッチ・インストール・ジョブの実行依頼』の説明のように、ジョブを手動で実行依頼することもできます。

X インストール・ダイアログの終了 (一連のパネルを終了させるため)

このパネルには、最後に使用された DB2 UDB (OS/390 版) サブシステム名も表示されます。P オプションを選択した場合は、この DB2 UDB (OS/390 版) 名を無視することができます。これは、DB2 UDB (OS/390 版) 名および他の QMF インストール・パラメーターが後続のパネルで上書きされる可能性があるためです。同様に、パネル上の DB2 UDB (OS/390 版) サブシステム名を無視し **P** を入力して次のパネルに進むことにより、追加

入力パラメーターの指定

の DB2 UDB (OS/390 版) サブシステムに関する QMF インストール・パラメーターをカスタマイズすることができます。

2. 最初のパラメーター入力パネルを出すには、P オプションを選択する。

各パネルで情報を入力すると、QMF は、ユーザーが選択したデータベース名のもとで QMF710.SDSQCLTE ライブラリーに入力データを保管します。

最後の入力パラメーター・パネルへの入力を完了する前にこのステップを終了すると、入力データは保管されません。最後のパネルでは、インストールの調整に使用されるジョブ・カード情報が求められます。バッチではなくフォアグラウンドでインストールを行いたい場合は、ジョブ・カード情報を入力する必要はなく、パネル上の指示されたスポットに x を入力するだけで済みます。

最後のインストール・パラメーターを入力した後で、メインメニューに戻ります。パラメーターの検討または変更を行いたい場合は、P を入力し、再び各入力パネルに戻ります。インストール・パラメーターの入力を適切に行えた場合は、次のステップに進みます。(必要があれば、この時点でインストール・プロセスを終え、あとで再開することができます。この場合、インストール・パラメーターは保管されます。)

ローカル DB2 UDB (OS/390 版) パラメーターの指定

39ページの図5 に示されているパネルは、まだインストール・パラメーターをいずれも保管していない場合に表示されます。このパネルは、メインメニューから P オプションを選択した場合にも表示されます。

31ページの表10 と32ページの表11 のワークシートの情報を使用して、パネルへの入力を済ませてください。


```

DXYEIN10          INSTALL QMF -- LOCAL DB2 PARAMETERS
ISPF Command ==>

LOCAL DB2 SUBSYSTEM ID   ==> DSN
LOCAL DB2 RELEASE LEVEL  ==> ("31" FOR V3R1, ETC)
LOCAL DB2 EXIT LIBRARY   ==>
LOCAL DB2 LOAD LIBRARY   ==>

COMMUNICATIONS DATABASE(CDB) INSTALLED AT LOCAL DB2 ==> ("Y","N")

PRESS:  ENTER to continue   PF01 for help   PF03 to end
    
```

図5. ローカル DB2 UDB (OS/390 版) パラメーター

このパネルで使用可能なオプションは、次のとおりです。

Local DB2 UDB for OS/390 subsystem ID (ローカル DB2 UDB (OS/390 版) サブシステム ID)

QMF アプリケーション・プランが結合される DB2 UDB (OS/390 版) サブシステム ID を指定します (必須。デフォルトは DSN です)。

Local DB2 UDB for OS/390 release level (ローカル DB2 UDB (OS/390 版) リリース・レベル)

ローカル・サブシステムの DB2 UDB (OS/390 版) リリース・レベルを指定します (必須。デフォルトなし)。

Local DB2 UDB for OS/390 exit library (ローカル DB2 UDB (OS/390 版) 出口ライブラリー)

ローカル DB2 UDB (OS/390 版) サブシステムの出口ライブラリーを指定します (必須。デフォルトなし)。

Local DB2 UDB for OS/390 load library (ローカル DB2 UDB (OS/390 版) ロード・ライブラリー)

ローカル・サブシステムの DB2 UDB (OS/390 版) ロード・ライブラリーを指定します (必須。デフォルトなし)。

Communications database (CDB) installed at local DB2 UDB for OS/390 (ローカル DB2 UDB (OS/390 版) にインストールされている通信デ

入力パラメーターの指定

データベース (CDB))

DB2 UDB (OS/390 版) 通信データベースがローカル DB2 UDB (OS/390 版) サブシステムにインストールされているかどうかを指定します (必須。デフォルトなし)。

データベース・インストールの効力範囲の指定

図6 に示されているパネルは、通信データベースがローカル DB2 UDB (OS/390 版) サブシステムにインストールされていることを、ユーザーが前のパネルで指定した場合に表示されます。

```
DXYEIN12          INSTALL QMF -- SCOPE OF DATABASE INSTALL
ISPF Command ==>

SCOPE OF DATABASE INSTALL      ==>      ("F" - full database,
                                           "R" - requester database only,
                                           "S" - server database only)

PRESS:  ENTER to continue   PF01 for help   PF03 to end
```

図6. データベース・インストールの効力範囲

データベース・インストールの効力範囲を指定します。上記のオプションの詳細については、11ページの『QMF インストール・プロセスのロードマップ』を参照してください。

QMF 6 を初めてインストールする場合は、全データベース・インストール・オプションを選択してください。

ローカル DB2 UDB (OS/390 版) サブシステム用の QMF パラメーターの指定

41ページの図7 に示されているパネルは、全データベース・インストールおよびリクエスター・データベース・インストールの場合に表示されます。

```

DXYEIN11          INSTALL QMF -- QMF PARAMETERS AT LOCAL DB2
ISPF Command ==>

CUSTOMIZE QMF RUNTIME LIBRARIES          ==> Y          ("Y" or "N")

- Install QMF panels
- Install QMF/GDDM map groups
- Install QMF/GDDM sample charts forms
- Make QMF REXX EXECs available
- Make QMF CLISTS available

QMF APPLICATION PLAN ID AT LOCAL DB2    ==> QMF710

PRESS:  ENTER to continue   PF01 for help   PF03 to end
    
```

図7. ローカル DB2 UDB (OS/390 版) での QMF パラメーター

このパネルで使用可能なオプションは、次のとおりです。

Customize QMF runtime libraries (QMF ランタイム・ライブラリーをカスタマイズする)

QMF ランタイム・ライブラリーのカスタマイズが必要な場合は YES を指定します。オペレーティング・システムごとに 1 回これらのライブラリーをカスタマイズするだけで済みます (必須。デフォルトなし)。

QMF application plan ID at local DB2 UDB for OS/390 (ローカル DB2 UDB (OS/390 版) での QMF アプリケーション・プラン ID)

ローカル DB2 UDB (OS/390 版) サブシステムで結合する必要がある QMF アプリケーション・プラン名を指定します (必須。デフォルトなし)。

リモート・サーバー・ロケーションの指定

42ページの図8 に示されているパネルは、『Scope of database install』 (データベース・インストールの効力範囲) パネルで、サーバー・データベースの S を指定した場合に表示されます。

入力パラメーターの指定

```
DXYEIN14          INSTALL QMF -- DB2 SERVER SYSTEM
ISPFL Command ==>

DB2 SERVER LOCATION IN REMOTE DB2 SUBSYSTEM  ==> N      ("Y" OR "N")

(If the DB2 server location is different from
 the requester location, the DB2 server is remote.)
```

図8. DB2 UDB (OS/390 版) リモート・サーバー・パネル

このパネルで使用可能なオプションは、次のとおりです。

DB2 UDB for OS/390 server location in remote DB2 UDB for OS/390 system (DB2 UDB (OS/390 版) サーバー・ロケーションがリモート DB2

UDB (OS/390 版) システム内にあるか否か)

サーバー・データベースがローカル DB2 UDB (OS/390 版) サブシステムと異なっているかどうかを指定します (必須。デフォルトなし)。

DB2 UDB (OS/390 版) および QMF のパラメーターの指定

データベース・インストールの効力範囲が F (全データベース) である場合、またはデータベース・インストールの効力範囲が S (サーバー・データベース) であり、そのサーバー・データベースがローカル DB2 UDB for OS/390 サブシステムと同じである場合には、43ページの図9 に示されているパネルが表示されます。

```

DXYEIN16          INSTALL QMF -- DB2 AND QMF PARAMETERS
ISPF Command ==>

DB2 USER CATALOG          ==>

DB2 USER CATALOG PASSWORD ==>

QMF TABLESPACES CATALOG ALIAS ==> QMFDSN
QMF TABLESPACES CATALOG PASSWORD ==>

QMF TABLESPACES VOLUME   ==>      ( VOLUME SERIAL NUMBER
                                     OR "AST",
                                     AST stands for *)

PREVIOUS QMF LEVEL       ==>      ("V2R4","V3R1","V3R1M1",
                                     "V3R2","V3R3","V6R1","NONE")

PRESS:  ENTER to continue  PF01 for help  PF03 to end
    
```

図9. DB2 UDB (OS/390 版) および QMF パラメーター

以下のパラメーターへの入力を済ませてください。

DB2 UDB for OS/390 user catalog (DB2 UDB (OS/390 版) ユーザー・カタログ)

QMF インストールが QMF カタログ別名 (VCAT 名) の作成に使用する ICF カタログを指定します (必須。デフォルトなし)。

DB2 UDB for OS/390 user catalog password (DB2 UDB (OS/390 版) ユーザー・カタログ・パスワード)

DB2 UDB (OS/390 版) ユーザー・カタログにアクセスするためのパスワードを指定します。このユーザー・カタログにより、QMF インストールがこのユーザー・カタログ内に QMF カタログ別名を作成することができます (オプション)。

QMF tablespaces catalog alias (QMF 表スペース・カタログ別名)

すべての QMF 表スペースの VCAT 名を指定します。これらの QMF 表スペースに関連づけられる VSAM データ・セットは、この別名値の高水準修飾子をもっています。QMF の前のレベルからの移行を行う場合には、前のリリースと同じ別名値を使用してください (必須。デフォルトなし)。

QMF tablespaces catalog password (QMF 表スペース・カタログ・パスワード)

インストール先システムによって作成されるすべての QMF 制御表スペースおよび索引スペースのパスワードを指定します (オプション)。

入力パラメーターの指定

QMF tablespaces volume (QMF 表スペース・ボリューム)

QMF 表スペースが常駐するボリューム通し番号を指定します (必須。デフォルトなし)。

Default punctuation (デフォルト句読法)

DB2 UDB (OS/390 版) での小数点の記号を指定します (必須。デフォルトなし)。

Previous QMF level (直前の QMF レベル)

移行を行う QMF の直前のリリース・レベルを指定します (必須。データベース内に直前のリリース・レベルがない場合は、NONE を入力します)。

リモート・サーバー・パラメーターの指定

図10 に示されているパネルは、サーバーがローカル DB2 UDB (OS/390 版) システムと異なっている場合にのみ表示されます。

```
DXYEIN15          INSTALL QMF -- REMOTE SERVER PARAMETERS
ISPF Command ==>

DB2 SERVER LOCATION NAME          ==>
DB2 SERVER ON A REMOTE OS/390 SYSTEM ==>          ("Y" OR "N")
DB2 USER CATALOG FOR SERVER      ==>
DB2 USER CATALOG PASSWORD        ==>

QMF TABLESPACES CATALOG ALIAS AT SERVER ==> QMFDSN
QMF TABLESPACES CATALOG PASSWORD      ==>
QMF TABLESPACES VOLUME            ==>          ( VOLUME SERIAL NUMBER
                                                OR "AST",
                                                AST stands for *)

DEFAULT PUNCTUATION AT SERVER      ==> .          ("," OR ".")
PREVIOUS QMF LEVEL INSTALLED AT SERVER ==>          (V2R4,V3R1,V3R3,
                                                V3R1M1,V3R2,NONE)

ROUTE XEQ JCL STATEMENT TO SERVER SYSTEM (REQUIRED IF SYSTEM IS REMOTE)
FOR JES2, USE THE FORMAT: /*ROUTE XEQ <NODEID>.<USERID>
FOR JES3, USE THE FORMAT: // *ROUTE XEQ <NODEID>.<USERID>
==>

PRESS: ENTER to continue  PF01 for help  PF03 to end
```

図10. リモート・サーバー・パラメーター

以下のパラメーターへの入力を済ませてください。

DB2 UDB for OS/390 server location name (DB2 UDB (OS/390 版) サーバーのロケーション名)

リモート・サーバー・データベースの DB2 UDB (OS/390 版) ロケーション名を指定します (必須。デフォルトなし)。

DB2 UDB for OS/390 server in another operating system (DB2 UDB

(OS/390 版) サーバーが別のオペレーティング・システム内にあるか否か)

リモート・サーバー・データベースがリクエスター・データベース・システムとは異なるオペレーティング・システム内にあるかどうかを指定します (必須。デフォルトなし)。

DB2 UDB for OS/390 user catalog for server (サーバー用の DB2 UDB

(OS/390 版) ユーザー・カタログ)

QMF インストールが QMF カタログ別名 (QMF VCAT 名) の作成に使用する ICF カタログを指定します (必須。デフォルトなし)。

DB2 UDB for OS/390 user catalog password (DB2 UDB (OS/390 版) ユーザー・カタログ・パスワード)

DB2 UDB (OS/390 版) ユーザー・カタログにアクセスするためのパスワードを指定します。このユーザー・カタログにより、QMF インストールがこのユーザー・カタログ内に QMF カタログ別名を作成することができます (オプション)。

QMF tablespaces catalog alias at server (サーバーでの QMF 表スペース・カタログ別名)

すべての QMF 表スペースの VCAT 名を指定します。これらの QMF 表スペースに関連づけられる VSAM データ・セットは、この別名値の高水準修飾子をもっています。QMF の前のレベルからの移行を行う場合には、前のリリースと同じ別名値を使用してください (必須。デフォルトなし)。

QMF tablespaces catalog password (QMF 表スペース・カタログ・パスワード)

ード)

インストール先システムによって作成されるすべての QMF 制御表スペースおよび索引スペースのパスワードを指定します (オプション)。

QMF tablespaces volume for server (サーバー用の QMF 表スペース・ポ

リューム)

QMF 表スペースが常駐するボリューム通し番号を指定します (必須。デフォルトなし)。

Default punctuation at server (サーバーでのデフォルト句読法)

小数点の記号を指定します (必須。デフォルトなし)。

入力パラメーターの指定

Previous QMF level installed at server (サーバーにインストールされている QMF の前のレベル)

移行を行う QMF の前のリリース・レベルを指定します (必須。データベース内に前のリリース・レベルがない場合は、“NONE” を入力します)。

ROUTE XEQ JCL statement to server system (サーバー・システムに対する ROUTE XEQ JCL ステートメント)

特定のインストール・ジョブを実行用にリモート・システムに送るために ROUTE JCL ステートメントを指定します (サーバー・システムがリクエスト・システムと異なることをユーザーが指定した場合は、必須)。

QMF 表スペースに関するスペース・パラメーターの指定

データベース・インストールの効力範囲が F (全データベース) である場合、またはデータベース・インストールの効力範囲が S (サーバー・データベース) であり、そのデータベース内に前の QMF リリース・レベルがない場合には、図11 に示されているパネルが表示されます。

```
DXYEIN17      INSTALL QMF -- QMF TABLESPACES SPACE PARAMETERS
ISPF Command ==>

Specify the sizes (in 1K units) for the following tablespaces

TABLESPACE FOR QMF
CONTROL TABLE:      PRIMARY          SECONDARY
-----
Q.OBJECT_DIRECTORY  ==> 200          ==> 20
Q.OBJECT_REMARKS    ==> 200          ==> 20
Q.OBJECT_DATA       ==> 5000         ==> 200
Q.PROFILES          ==> 100          ==> 20
Q.ERROR_LOG         ==> 100          ==> 20
Q.COMMAND_SYNONYMS  ==> 100          ==> 20
Q.RESOURCE_TABLE    ==> 100          ==> 20
"SAVE DATA" TABLESPACE ==> 100          ==> 20

PRESS:  ENTER to continue   PF01 for help   PF03 to end
```

図 11. QMF 表スペースのスペース・パラメーター

QMF 制御表スペースの 1 次および 2 次割り振りを指定します。QMF は、これらの値を使用して、すべての VSAM ファイルをこれらの表スペースに割り振ります。インストール先システムのサイズに応じて、拡張用にフリー・スペースを使用できるようにするためにデフォルトのサイズを大きくしたり小さくしたりする必要が生じる場合があります。図11 は、デフォルトのサイズ (1K 単位) を示したものです。

QMF 索引スペースのパラメーターの指定

データベース・インストールの効力範囲が F (全データベース) または S (サーバー・データベース) インストールであり、そのデータベース内に前の QMF リリース・レベルがない場合に、図12 に示されているパネルが表示されます。

```

DXYEIN18      INSTALL QMF -- QMF INDEXSPACES SPACE PARAMETERS
ISPF Command ==>

Specify the sizes (in 1K units) for the following table indexes

TABLE INDEX          PRIMARY          SECONDARY
-----
Q.OBJECT_DIRECTORYX ==> 100          ==> 20
Q.OBJECT_REMARKSX   ==> 100          ==> 20
Q.OBJECT_OBJDATAAX ==> 100          ==> 20
Q.PROFILEX          ==> 100          ==> 20
Q.COMMAND_SYNONYMX  ==> 100          ==> 20

PRESS:  ENTER to continue   PF01 for help   PF03 to end
    
```

図12. QMF 索引スペースのスペース・パラメーター

デフォルトのサイズ (1K 単位) が 図12 にリストされています。

QMF 索引スペースの 1 次および 2 次割り振りを指定します。QMF は、これらの値を使用して、すべての VSAM ファイルをこれらの表スペースに割り振ります。インストール先システムのサイズに応じて、拡張用にフリー・スペースを使用できるようにするためにデフォルトのサイズを大きくしたり小さくしたりする必要が生じる場合があります。

ジョブ・カードの指定

48ページの図13 に示されているパネルは、P オプション (インストール・パラメーター) に関する最後のパネルです。

入力パラメーターの指定

```
DXYEIN19                INSTALL QMF -- JOBCARD
ISPF Command ==>

Modify the Job cards below to represent your installation requirements.
The "USER" and "PASSWORD" parameters must be specified in systems using
RACF.  Since part of this install involves creating objects in DB2, you
will need DB2 SYSADM authority.  Please see the "QMF Installation Guide
for OS/390" for more detail.

If you will be performing the installation in foreground rather than
batch, and you DO NOT want the batch (JCL) files tailored, enter an
'X' here: ==>

JOB CARD INFORMATION (used for batch (JCL) tailoring)
==> //QMFINSTL JOB (ACCT),NAME,
==> //          CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),
==> //          USER=Q,PASSWORD=Q
==> // *

PRESS:  ENTER to continue   PF01 for help   PF03 to end
```

図 13. ジョブ・カード

QMF は、このジョブ・カード情報を使用して、インストール先システムに関する残りのすべてのインストール・ジョブを実行依頼します。このパネルへの入力が完了すると、「*QMF* インストール -- メインメニュー」に戻ります。このパネルで、**P** を選択して選択項目を検討するか、またはジョブの調整に進むことができます。

ステップ 2 -- ジョブの調整

ジョブを調整するためには、メインメニューで **T** を選択します。このステップでは、35ページの『ステップ 1 -- QMF インストール・パラメーターの指定』で入力したインストール・パラメーター設定値を使用して既存の SDSQSAPE および SDSQEXCE メンバーを更新します。

このステップでは、

- 35ページの『ステップ 1 -- QMF インストール・パラメーターの指定』でユーザーが選択したインストール・パスに合わせてシステムが JCL およびコピー・ファイルを調整しているという旨のメッセージが出されます。
- QMF 呼び出し可能インターフェース REXX EXEC QMF710.SDSQEXCE(DSQSCMDE) が、各パラメーター (QMF プラン ID および DB2 UDB (OS/390 版) サブシステム名) のデフォルト値を更新するために全データベースおよびリクエスター・インストールに合わせて変更されます。

このステップの終わりでは、「*QMF* インストール -- メインメニュー」に戻ります。そこで、*QMF* のインストールを続けることができます。

考慮事項: *QMF710.SDSQSAPE* ライブラリーのインストール JCL または制御ファイルのいずれも手動で編集してはなりません

(編集指示がない限り)。これらのファイルを調整する CLIST は、これらのファイルの行の順序およびフォーマットによって異なります。

ジョブをバッチで実行依頼することを選択した場合は、調整処理が正しく行われたことを検査するように求められます。ジョブがユーザーの意図に合致しない場合は、調整のために「*QMF* インストール -- メインメニュー」に常に戻ってください。

次に実行するステップは、*QMF* のインストールをフォアグラウンドで行うかまたはバッチで手動で行うかを選択することです。フォアグラウンド・インストールを選択するためには、「*QMF* インストール -- メインメニュー」で **I** を選択し、『ステップ 3 -- フォアグラウンドでの *QMF* のインストール』の指示に従ってください。手動インストールを選択するためには、「*QMF* インストール -- メインメニュー」で **X** を選択し、51ページの『第4章 *QMF* バッチ・インストール・ジョブの実行依頼』の指示に従ってください。

I を選択すると、インストール・オプションを要求するパネルが表示されます。情報の入力後に、インストールが進行中であることを示すメッセージが出されます。

ステップ 3 -- フォアグラウンドでの *QMF* のインストール

フォアグラウンドでジョブを実行依頼するために「*QMF* インストール -- メインメニュー」で **I** を選択すると、ジョブはユーザーの現行 LOGON ID のもとでインストールされます。ユーザーの LOGON ID に 11ページの『データベース許可 ID Q』の説明のように SYSADM 権限が付与されているか確認してください。

QMF をリモート DB2 UDB for OS/390 サーバーにインストールする場合は、フォアグラウンド・インストールを実行する前に、次のことを行ってください。

- ***QMF* を初めてインストールする場合 :**

全データベース・インストール・オプションを使用して、*QMF* をローカル DB2 UDB (OS/390 版) サブシステムにインストールする。

入力パラメーターの指定

QMF をローカル DB2 UDB (OS/390 版) サブシステム内にはないサーバーにインストールする場合には、フォアグラウンドでこの CLIST を呼び出す前に、VSAM データ・セットの割り振りに必要なジョブを実行依頼してください。

初期インストールの場合は、ジョブ QMF710.SDSQSAPE(DSQ1TBAJ) を実行依頼してください。

- **QMF バージョン 2 から移行する場合 :**

QMF バージョン 2 リリース 2、3、または 4 から移行する場合は、全データベース・インストール・オプションを使用して QMF をローカル DB2 UDB (OS/390 版) に移行します。

1. サーバーでの索引スペースを停止するために
QMF710.SDSQSAPE(DSQ1SPDB) で -STOP コマンドを発行してください。
2. QMF710.SDSQSAPE(DSQ1TBA1) を実行依頼する。
3. サーバーでの索引スペースを開始するために
QMF710.SDSQSAPE(DSQ1STDB) で -START コマンドを発行する。

- **QMF バージョン 3 から移行する場合 :**

アクションをとる必要はありません。

これで、フォアグラウンド・インストールは完了しました。次のいずれかに進んで、インストールを続けます。

- 71ページの『第5章 QMF の TSO 用調整』
- 81ページの『第6章 QMF の CICS 用調整』
- 91ページの『第7章 QMF の ワークステーション・データ・サーバー用調整』

第4章 QMF バッチ・インストール・ジョブの実行依頼

この章では、バッチ環境で QMF をインストールする方法について説明します。

ステップ 4 -- QMF パネルのインストール

DSQ1EPNL では、DSQ1EJAL の実行時に 17ページの『第2章 QMF の計画』で作成された 2 つのデータ・セット (DSQPVARE および DSQPNLE) を使用します。

DSQ1EPNL では、QMF パネルの拡張バージョンをパネル・ファイル QMF710.DSQPNLE にコピーします。

1. QMF710.SDSQSAPE(DSQ1EPNL) を編集する。
2. ジョブのインストリーム・プロシージャー内の次の値を検査または変更する。

```
//DSQ1PNL PROC RGN='2048K',           Job-step region size
//              QMFTPRE='QMF710'      Prefix for QMF target libraries
//              LKEY='E'               Language identifier
```

3. ジョブ QMF710.SDSQSAPE(DSQ1EPNL) を実行依頼する。
4. 0 というリターン・コードを検査する。

ステップ 5 -- QMF/GDDM マップ・グループのインストール

DSQ1EMAP では、特定の GDDM マップ・グループの拡張バージョンを QMF710.DSQMAPE という名前のマップ・グループ・ライブラリーにコピーします。マップ・グループは、ソース・ライブラリー QMF710.SDSQSAPE からコピーされます。

1. DSQ1EMAP を編集する。
2. ジョブのインストリーム・プロシージャー内の次のパラメーターを検査および変更する (必要な場合)。

```
//DSQ1MAP PROC RGN='2048K',           Job-step region size
//              QMFTPRE='QMF710',      Prefix for QMF target libraries
//              MAPID=
```

3. ジョブ QMF710.SDSQSAPE(DSQ1EMAP) を実行依頼する。
4. 0 というリターン・コードを検査する。

ステップ 6 -- QMF/GDDM サンプル図表書式のインストール

DSQ1CHRT では、GDDM 図表書式ファイルの拡張バージョンをライブラリー QMF710.SDSQSAPE からライブラリー QMF710.DSQCHART にコピーします。

1. DSQ1CHRT を編集する。
2. ジョブのインストリーム・プロシーチャーの入力パラメーターのデフォルト値を検査または変更する。

```
//DSQCHRT PROC RGN='2048K',           Job-step region size
//              QMFTPRES='QMF710',     Prefix for QMF target libraries
//              CHART=
```

3. ジョブ QMF710.SDSQSAPE(DSQ1CHRT) を実行依頼する。
4. 0 というリターン・コードを検査する。

ステップ 7 -- REXX EXEC および CLIST レコードの変換

このステップでは、REXX EXEC および CLIST レコードを固定長から可変長に変換します。次の 2 つのジョブを変換のために使用します。

- DSQ1EJVE は、QMF REXX EXEC レコードを固定長から可変長に変換します。
- DSQ1EJVC は、QMF CLIST レコードを固定長から可変長に変換します。

REXX EXEC レコードの変換

QMF EXEC ライブラリーには固定長レコードが入っています。このライブラリーは、固定長レコードが入っている他の EXEC ライブラリーだけに連結することができます。ライブラリーに可変長レコードが入っている場合は、可変長レコードを用いて QMF ライブラリーのコピーを作成する必要があります。

1. 次のことを検査してください。
 - 他の EXEC ライブラリーに固定長レコードが入っている場合は、このステップをとばして先をお読みください。
 - 他の EXEC ライブラリーに可変長レコードが入っている場合は、このステップを続けてお読みください。

このインストールのあとの方で、QMF EXEC ライブラリー (QMF710.SDSQEXCE.VB) を SYSEXEC データ・セットとして割り振ります。そのために、このライブラリーは通常、他の EXEC ライブラリーに連結されます。

例

次の JCL (DSQ1EINV からのもの) では、ライブラリー QMF710.SDSQEXCE は、SYS2.EXEC という名前の EXEC ライブラリーに連結されます。

```
//SYSEXEC DD DSN=SYS2.EXEC,DISP=SHR
//          DD DSN=QMF710.SDSQEXCE,DISP=SHR
```

詳細については、*QMF* アプリケーション開発の手引き を参照してください。

2. QMF710.SDSQSAPE(DSQ1EJVE) を編集する。

ライブラリーのコピーのボリューム通し番号を変更する。

```
//DSQTEVB.SYSUT2 DD DISP=(NEW,CATLG),UNIT=SYSDA,
// SPACE=(8800,(400,50,25)),VOL=SER=XXXXXX,
// DCB=(RECFM=VB,LRECL=84,BLKSIZE=8800)
```

3. インストール先システムに適合するようにジョブ・ステートメントを変更する。

```
//DSQ1EJVE JOB (ACCT),NAME,
// CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),
// USER=Q,PASSWORD=Q
```

4. ジョブのインストリーム・プロシージャ内の QMFTPRE パラメーターの値を検査および変更する (必要な場合)。

```
//DSQTEVB PROC RGN='2048K',
// QMFTPRE='QMF710',          Prefix for QMF libraries
// CLIST=                      Leave blank;
//*                             used when the procedure is called
```

5. ジョブ QMF710.SDSQSAPE(DSQ1EJVE) を実行依頼する。

6. 0 というリターン・コードを検査する。

ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

CLIST レコードの変換

QMF CLIST ライブラリーには固定長レコードが入っています。このライブラリーは、固定長レコードが入っている他の CLIST ライブラリーだけに連結することができます。ライブラリーに可変長レコードが入っている場合は、可変長レコードを用いて QMF ライブラリーのコピーを作成する必要があります。

1. 次のことを検査してください。

QMF バッチ・インストール・ジョブの実行依頼

- 他の CLIST ライブラリーに固定長レコードが入っている場合は、このステップをとばして先をお読みください。
- インストール先システムで可変長レコードの CLIST ライブラリーを使用する場合は、このステップを続けてお読みください。このステップでは、QMF710.SDSQCLTE.VB という名前の可変長レコードを含む CLIST ライブラリーを作成します。SYSPROC 連結のためにこの新しい CLIST を使用します。(QMF インストール・プロセスと SMP/E の両方では更新処理用に固定ブロック CLIST が必要になるので、QMF710.SDSQCLTE を固定ブロックのままにしておく必要があります。)

このインストールのあとで、QMF CLIST ライブラリー (QMF710.SDSQCLTE) を SYSPROC データ・セットとして割り振る必要があります。そのために、このライブラリーは通常、他の CLIST ライブラリーに連結されます。

例

次の JCL (DSQ1EINV からのもの) では、ライブラリー QMF710.SDSQCLTE は、SYS2.CLIST という名前の CLIST ライブラリーに連結されます。

```
//SYSPROC DD DSN=SYS2.CLIST,DISP=SHR
//          DD DSN=QMF710.SDSQCLTE,DISP=SHR
```

2. QMF710.SDSQSAPE(DSQ1EJVC) を編集する。
3. インストール先システムに適合するようにジョブ・ステートメントを検査または変更する。

```
//DSQ1EJVC JOB (ACCT),NAME,
//          CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),
//          USER=Q,PASSWORD=Q
```

4. ジョブのインストリーム・プロシーチャーの QMFTPRE パラメーターの値を検査または変更する。

```
//DSQTIVB PROC RGN='2048K',
//          QMFTPRE='QMF710',          Prefix for QMF libraries
//          EXEC=                      Leave blank;
//*                                     used when the procedure is called
```

5. ライブラリーのコピーのボリューム通し番号を変更する。

```
//DSQTIVB.SYSUT2 DD DISP=(NEW,CATLG),UNIT=SYSDA,
//          SPACE=(8800,(400,50,25)),VOL=SER=XXXXXX,
//          DCB=(RECFM=VB,LRECL=84,BLKSIZE=8800)
```

6. ジョブ QMF710.SDSQSAPE(DSQ1EJVC) を実行依頼する。

7. 0 というリターン・コードを検査する。

DB2 ユニバーサル・データベース (OS/390 版) アプリケーションとしての QMF の準備

この一連のステップで、次のことを行います。

- DB2 UDB (OS/390 版) リソースを作成する。
- 作成したリソースを DB2 UDB (OS/390 版) で使用できるようにする。
- QMF を DB2 UDB (OS/390 版) にバインドする。

QMF の旧リリースが DB2 UDB (OS/390 版) サブシステムにインストールされている場合には、これらのリソースの一部はすでに使用可能になっています。

QMF を DB2 (OS/390 版) バージョン 7 またはそれ以降のデータベースにインストールする場合には、プランおよびパッケージのバインドに用いるデータベース・アプリケーション・エンコードのインストール・パラメーターが EBCDIC または EBCDIC ccsid に設定されていることを確認してください。

56ページの『ステップ 8 -- DB2 UDB (OS/390 版) への QMF インストール・プログラムのバインド』では、編集を行ない、2 つの DB2 UDB (OS/390 版) アプリケーション・プログラムをバインドします。その他のステップでは、TSO バッチ・ジョブ (プログラム IKJEFT01 を使用して) を実行し、56ページの『ステップ 8 -- DB2 UDB (OS/390 版) への QMF インストール・プログラムのバインド』の出力を使用して DB2 UDB (OS/390 版) ステートメントを実行します。これらのステップを構成する項目の大部分は、ライブラリー QMF710.SDSQSAPE または QMF710.SDSQLOAD のメンバーです (これらのメンバーは SMP/E によってこれらのライブラリーに入れられました)。TSO バッチを実行するこれらすべてのステップでは、システム・メッセージ内のステップ完了コードを検査します。完了メッセージは、指示されるように、SYSTSPRT または SYSTEM の出力の中にあります。SYSPRINT には、IBM サポート用に診断情報が追加されています。

各サブステップは再始動することができます。これは、DB2 UDB (OS/390 版) データベースに対するすべての変更がジョブの終了までコミットされないためです。

ステップ 8 -- DB2 UDB (OS/390 版) への QMF インストール・プログラムのバインド

このステップでは、プログラム DSQCBSQL および DSQCBINS を DB2 UDB (OS/390 版) にバインドします。このバインドによる並列アプリケーション・プランには DSQIN710 という名前が付けられます。

1. QMF710.SDSQSAPE(DSQ1BSQL) を編集する。
2. ジョブのインストリーム・プロシーチャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQ1BSQL PROC RGN='2048K',           Job-step region size
//  QMFTPRE='QMF710',                 Prefix for QMF target libraries
//  DB2EXIT='DSN710.SDSNEXIT',       Exit DB2 UDB for OS/390 library name
//  DB2LOAD='DSN710.SDSNLOAD'       DB2 UDB for OS/390 program library name
```

このジョブでは、一連の DB2 UDB (OS/390 版) ステートメントを使用して作業を行います。これらのステートメントはライブラリー QMF710.SDSQSAPE のメンバーです。

3. ジョブ QMF710.SDSQSAPE(DSQ1BSQL) を実行依頼する。
4. 0 というリターン・コードが表示されているか検査する。

リターン・コードが 0 以外の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

リクエスター・データベース・インストールを行っている場合は、68ページの『ステップ 15 -- QMF アプリケーション・プランの DB2 UDB (OS/390 版) へのバインド』に進みます。それ以外の場合は、次のステップに進みません。

ステップ 9 -- QMF 制御表の作成

このステップでは、8 つの QMF 制御表および 6 つのカタログ視点を作成します。これらの表および視点の詳細については、569ページの『付録B. DB2 に常駐している QMF オブジェクト』を参照してください。

タイプ 2 への QMF 制御表索引の変換

DB2 (MVS/ESA 版) バージョン 4 またはそれ以降のリリースを実行している場合は、QMF 制御表索引が TYPE 2 索引として移行または作成されます。QMF 索引のタイプを決定するには、次の照会を実行してください。

```
SELECT NAME,CREATOR,TBNAME,TBCREATOR,INDEXTYPE FROM SYSIBM.SYSINDEXES
WHERE CREATOR = 'Q'
```

INDEXTYPE が ' '(ブランク) であれば、移行ジョブは索引を変更して、TYPE 2 にします。TYPE 2 索引への移行後に、索引は回復保留状態にされます。索引の回復、ロード、または再編成が行われるまでは、索引の変更は行われません。DB2 のユーティリティの REBUILD INDEX (ただし、DB2 バージョン 5 またはそれ以前のバージョンを使用している場合には RECOVER INDEX) を実行して、QMF 索引の変換を完了します。

ユーザーが DB2 R310 へフォールバックしたい場合は、フォールバックする前に索引を元の TYPE 1 へ変換しなおす必要があります。ユーザーは、「ALTER INDEX」SQL ステートメントを使用する場合があります。構文については、DB2 UDB (OS/390 版) SQL 解説書を参照してください。QMF の制御表索引は、570ページの表69 にリストされています。

リモート作業単位についてのヒント

DB2 UDB (OS/390 版) リモート・ロケーションの表および視点にアクセスしたい場合は、各リモート・ロケーションでの QMF カタログ視点を作成するためのインストール・ジョブを実行する必要があります。

どのジョブを実行するかは、QMF の旧版から移行を行っているか否かによって決まります。また、QMF VnRn; 用にユーザーが選択した DB2 UDB (OS/390 版) サブシステムに QMF のどのバージョンおよびリリースがインストールされているかによっても異なります。

DB2 UDB (OS/390 版) サブシステムの QMF レベル

以下の手順に従ってください

QMF 7

64ページの『ステップ 10 -- QMF IVP 用の表スペースの作成』。いずれの制御表も変更する必要はありません。そのため、このステップをとばして次のステップに進むことができます。

QMF 3.x

58ページの『QMF バージョン 3 リリース 3.0、2.0、1.1、1.0 からの移行』

QMF 2.4

58ページの『QMF バージョン 2.4 からの移行』

QMF は新規

61ページの『QMF の旧リリースがインストールされていない場合の制御表の作成』

QMF バッチ・インストール・ジョブの実行依頼

QMF バージョン 3 リリース 3.0、2.0、1.1、1.0 からの移行

QMF バージョン 7 用にユーザーが選択した DB2 UDB (OS/390 版) サブシステムに現在 QMF バージョン 3 のいずれかのリリースがインストールされている場合には、このステップを実行してください。

DSQ1TBJ0 では、QMF 3.x 制御表を QMF バージョン 7 レベルに移行します。

バージョン 3 からの移行を行っていない場合は、このステップをとばして先に進んでください。

1. QMF710.SDSQSAPE(DSQ1TBJ0) を編集する。
2. ジョブのインストリーム・プロシージャ内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48 ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1TBJ0 PROC RGN='2048K',           Job-step region size
//   QMFTPRES='QMF710',               Prefix for QMF target libraries
//   DB2EXIT='DSN710.SDSNEXIT',       Exit DB2 UDB for OS/390 library name
//   DB2LOAD='DSN710.SDSNLOAD'       DB2 UDB for OS/390 program library name
```

3. ジョブ QMF710.SDSQSAPE(DSQ1TBJ0) を実行依頼する。
4. リターン・コードの 0 または 4 の受取を検査する。完了メッセージについては SYSTEMR を検討する。

リターン・コードが **0** または **4** 以外の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

QMF バージョン 2.4 からの移行

QMF バージョン 7 用にユーザーが選択した DB2 UDB (OS/390 版) サブシステムに現在 QMF バージョン 2 リリース 4 がインストールされている場合には、このステップを実行してください。

このインストール・ステップには、次の 3 つのジョブが含まれています。

- DSQ1TBD1 では、索引スペースを停止する。
- DSQ1TBA1 では、VSAM ファイルを割り振る。
- DSQ1TBJ0 では、制御表を再作成する。

索引スペースの停止 (DSQ1TBD1)

このジョブでは、QMF プロファイル索引スペースを停止し、この索引スペースが次のステップで再定義できるようにします。

1. サーバーがローカル DB2 UDB (OS/390 版) 内にあるかどうかを判別する。サーバーがローカル DB2 UDB (OS/390 版) 内に**存在する**場合のみ、このジョブを実行してください。
サーバーでの索引スペースを停止するために QMF710.SDSQSAPE(DSQ1TBD1) で -STOP コマンドを発行してください。
2. QMF710.SDSQSAPE(DSQ1TBD1) を編集する。
3. ジョブのインストリーム・プロシージャー内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48 ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1TBD1 PROC RGN='2048K',           Job-step region size
//  QMFTPRE='QMF710',                 Prefix for QMF target libraries
//  DB2EXIT='DSN710.SDSNEXIT',        Exit DB2 UDB for OS/390 library name
//  DB2LOAD='DSN710.SDSNLOAD'        DB2 UDB for OS/390 program library name
```

4. ジョブ QMF710.SDSQSAPE(DSQ1TBD1) を実行依頼する。
5. リターン・コードの 0 の受取を検査する。完了メッセージについては SYSTEMM を検討する。

リターン・コードが 0 以外の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

VSAM ファイルの割り振り (DSQ1TBA1)

このジョブでは、VSAM ファイルを Q.PROFILEX 用の QMF 索引スペースに割り振ります。このジョブでは、一連の IDCAMS ステートメントを使用して作業を行います。

1. 分散データ用に DSQ1TBA1 を調整する。
DB2 UDB (OS/390 版) がバージョン 2 リリース 3 であり、サーバー・システムがリモートである (すなわち、ローカル・システム内に存在しない) 場合には、ジョブ・カードの後に /*ROUTE XEQ JCL ステートメント (JES2 の場合) または /*ROUTE XEQ JCL ステートメント (JES3 の場合) を挿入する必要があります。これらのステートメントは必須です。これは、リモート・サーバー・システムで ROUTE カードを使用してこのジョブを実行することにより、そのリモート・サーバー・システムで別名および VSAM データ・セットを割り振る必要があるためです。
2. QMF710.SDSQSAPE(DSQ1TBA1) を編集する。

QMF バッチ・インストール・ジョブの実行依頼

3. ジョブのインストリーム・プロシージャ内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1TBA1 PROC RGN='2048K',      Job-step region size
//          QMFTPRE='QMF710' Prefix for the target libraries
```

4. QMF710.SDSQSAPE(DSQ1TBA1) を実行依頼する。
5. 0 というリターン・コードを検査する。

エラー・メッセージについて SYSPRINT を検査します。クラスターの DELETE および PURGE に関して、次のエラー・メッセージが表示される可能性があります。このメッセージは無視することができます。

```
IDG3012I Entry QMFDSN.DSNDBC.DSQDBCTL.PROFILEX I001,A001.
```

制御表の再作成

DSQ1TBJ0 では、制御表の変更、制御表索引の除去および再作成、Q.DSQ_RESERVED 表スペースおよび制御表の作成、ならびに QMF カタログ視点の作成を行います。このジョブでは、一連の DB2 UDB (OS/390 版) ステートメントを使用して作業を行い、いくつかのメンバーを保持します。

1. QMF710.SDSQSAPE(DSQ1TBJ0) を編集する。
2. ジョブのインストリーム・プロシージャ内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1TBD1 PROC RGN='2048K',      Job-step region size
//          QMFTPRE='QMF710',      Prefix for QMF target libraries
//          DB2EXIT='DSN710.SDSNEXIT', Exit DB2 UDB for OS/390 library name
//          DB2LOAD='DSN710.SDSNLOAD' DB2 UDB for OS/390 program library name
```

3. ユーザーのユーザー・カタログがパスワード保護されているかどうかを判別する。パスワード保護されている場合は、パスワード文節をメンバー DSQ1VSTB 内の STOGROUP ステートメントに追加します。

```
CREATE STOGROUP PASSWORD(password)
```

4. QMF710.SDSQSAPE(DSQ1TBJ0) を実行依頼する。
5. リターン・コードの 0 または 4 を受け取ったかどうかを検査する。完了メッセージについては SYSTERM を検討する。

リターン・コードが **0** または **4** 以外の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

タイプ 2 に変換した索引の回復

DSQ1TBJ0 を実行した結果、索引が TYPE 2 に変更された場合には、その索引は回復、ロード、または再編成をする必要があります。

索引の再作成する必要があるのかがわからない場合は、次のステップを行ってください。

1. DSQ1TBJ0 ジョブの出力を見直す。最終リターン・コードが 4 の場合、ジョブ出力のなかで ALTER INDEX のストリングを探す。ALTER INDEX が見つかった場合、次のステップを行う必要がある。
2. 次の DB2 コマンドを実行する。

```
-DISPLAY DATABASE (DSQDBCTL) SPACENAM (*)
```

STATUS が RECP または RW、RECP の索引 (TYPE=IX) がある場合、その索引は再作成する必要があります。

DB2 (OS/390 版) のリリースに応じて、次に示す DB2 ユーティリティ・ジョブのストリームを実行する必要があります。

バージョン 5 およびそれ以前のバージョンの場合

```
RECOVER INDEX(ALL) TABLESPACE DSQDBCTL.DSQTSTCT1
RECOVER INDEX(ALL) TABLESPACE DSQDBCTL.DSQTSTCT2
RECOVER INDEX(ALL) TABLESPACE DSQDBCTL.DSQTSTCT3
RECOVER INDEX(ALL) TABLESPACE DSQDBCTL.DSQTSPRO
RECOVER INDEX(ALL) TABLESPACE DSQDBCTL.DSQTSSYN
```

バージョン 6 およびそれ以降のバージョンの場合

```
REBUILD INDEX (Q.OBJECT_DIRECTORYX)
REBUILD INDEX (Q.OBJECT_REMARKSX)
REBUILD INDEX (Q.OBJECT_OBJDATAAX)
REBUILD INDEX (Q.PROFILEX)
REBUILD INDEX (Q.COMMAND_SYNONYMSX)
```

QMF の旧リリースがインストールされていない場合の制御表の作成

QMF 用の DB2 UDB (OS/390 版) サブシステムに QMF の以前のリリースがインストールされていない場合には、このステップを実行してください。

このステップには、次の 2 つのインストール・ジョブが含まれています。

- DSQ1TBAJ では、別名を割り振ります。(使用システムですでに別名が定義されている場合は、このステップをとばして先に進んでください。)
- DSQ1TBLJ では、QMF 制御表およびカタログ視点を作成およびロードします。

QMF バッチ・インストール・ジョブの実行依頼

別名および VSAM ファイルの割り振り

DSQ1TBAJ では、別名を QMF 制御表および視点に割り振ります。このジョブでは、一連の IDCAMS ステートメントを使用して作業を行い、1 つのステップが組み込まれます。

1. QMF710.SDSQSAPE(DSQ1TBAJ) を編集する。
2. ジョブおよびジョブ・ステップのインストリーム・プロシーチャー内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1TBAJ PROC RGN='2048K',      Job-step region size
//          QMFTPRES='QMF710',    Prefix for QMF target libraries
```

3. 分散データ用にジョブを調整する (分散データが存在する場合)。
DB2 UDB (OS/390 版) がバージョン 2 リリース 3 であり、サーバー・システムがリモートである (すなわち、ローカル・システム内に存在しない) 場合には、ジョブ・カードの後に /*ROUTE XEQ JCL ステートメント (JES2 の場合) または /*ROUTE XEQ JCL ステートメント (JES3 の場合) を挿入する必要があります。これらのステートメントは必須です。これは、リモート・サーバー・システムで ROUTE カードを使用してこのジョブを実行することにより、そのリモート・サーバー・システムで別名および VSAM データ・セットを割り振る必要があるためです。

DSQ1TBLR

このジョブ・ステップでは、DB2 UDB (OS/390 版) VSAM カタログ内で QMF 用の別名を定義します。このジョブ・ステップには次のステートメントが含まれます。

```
DEFINE ALIAS -
  (NAME('QMFDNS') RELATE('DSNC7101.USER.CATALOG'))
```

4. このステートメントがユーザーの調整指定と一致しているか検査する。
5. QMF710.SDSQSAPE(DSQ1TBAJ) を実行依頼する。
6. 0 というリターン・コードを検査する。

最初のステップが失敗した場合は、エラー・メッセージについて SYSPRINT を検査し、エラーを訂正してからジョブを再実行します。2 番目のステップが失敗した場合は、DSQ1VSTA を再始動します。該当のロケーションに関して SQLCODE 203 が表示された場合には、563ページの『付録A. 問題が発生した場合』を参照してください。

QMF 制御表およびカタログ視点の作成およびロード

DSQ1TBLJ では、一連の DB2 UDB (OS/390 版) ステートメントを使用して作業を行うことにより、QMF 制御表およびカタログ視点を作成し、またロードします。

1. QMF710.SDSQSAPE(DSQ1TBLJ) を編集する。
2. ジョブおよびジョブ・ステップのインストリーム・プロシージャ内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1TBLJ PROC RGN='2048K',           Job-step region size
//      QMFTPRE='QMF710',             Prefix for QMF target libraries
//      DB2EXIT='DSN710.SDSNEXIT',    Exit DB2 UDB for OS/390 library name
//      DB2LOAD='DSN710.SDSNLOAD'    DB2 UDB for OS/390 program library name
```

3. ユーザーのユーザー・カタログがパスワード保護されているかどうかを判別する。パスワード保護されている場合は、パスワード文節をメンバー DSQ1VSTB 内の STOGROUP ステートメントに追加します。

```
CREATE STOGROUP PASSWORD(password)
```

4. QMF カatalog別名 (デフォルト **QMFDSN**) のインストール・パラメーターが下記のメンバーでのユーザーの調整指定と一致しているかを検査する。

```
DSQ1VSTD
DSQ1TBLB
DSQ1TBLI
DSQ1TBLU
DSQ1TBLE
DSQ1TBLN
DSQ1TBLG
```

5. QMF710.SDSQSAPE(DSQ1TBLJ) を実行依頼する。
6. リターン・コードの 0 または 4 を受け取ったかどうかを検査する。完了メッセージについては SYSTEMM を検討する。

リターン・コードが 0 または 4 以外の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

ステップ 10 -- QMF IVP 用の表スペースの作成

DSQ1STGJ では、QMF インストール検査プロシージャ (IVP) 用に表スペース (DSQDBDEF.DSQTSDEF と名付けられる) を作成します。表スペースを作成する前に、このジョブでは、この表スペース用に記憶域グループ (DSQSGDEF) およびデータベース (DSQDBDEF) を作成します。インストール後に、ユーザーは、各ユーザーが作成する表のためにこの表スペースを使用することができます。

QMF バージョン 7 用にユーザーが選択した DB2 UDB (OS/390 版) サブシステムに表スペースがすでに存在している場合には、このステップをとばして先に進んでください。2 番目の表スペースを作成しようとすると、オブジェクトがすでに存在していることを示すエラー・メッセージが表示されます。

1. QMF710.SDSQSAPE(DSQ1STGJ) を編集する。
2. ジョブのインストリーム・プロシージャ内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48 ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1STGJ PROC RGN='2048K',           Job-step region size
//   QMFTPRE='QMF710',                 Prefix for QMF target libraries
//   DB2EXIT='DSN710.SDSNEXIT',       Exit DB2 UDB for OS/390 library name
//   DB2LOAD='DSN710.SDSNLOAD'       DB2 UDB for OS/390 program library name
```

3. メンバー DSQ1STGC を編集する。
4. ユーザーのユーザー・カタログがパスワード保護されているかどうかを判別する。パスワード保護されている場合は、パスワード文節をメンバー DSQ1STGC 内の STOGROUP ステートメントに追加します。

```
CREATE STOGROUP PASSWORD(password)
```

5. ジョブのインストリーム・プロシージャ内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48 ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
CREATE STOGROUP DSQSGDEF
  VOLUMES (DSNVOL)           QMF tables volume
  VCAT QMFDSN;              QMF catalog alias
```

このメンバー内の GRANT ステートメントにより、すべてのユーザーが、IVP 表スペースに表を作成することができます。この権限を特定のユーザーに合わせて制限することができます。インストーラーが IVP を実行する場合には、そのインストーラーを組み込む必要があります。プログラムがイ

ンストーラーの許可 ID (前提事項)のもとで実行される場合は、そのインストーラーには自動的に権限が付与されます。

6. QMF710.SDSQSAPE(DSQ1STGJ) を実行依頼する。
7. リターン・コードの 0 の受取を検査する。完了メッセージについては SYSTEMR を検討する。

リターン・コードが 0 以外の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

QMF サンプル表の設定

次の 2 つのステップでは、QMF サンプル表を設定します。

『ステップ 11 -- 旧サンプル表の削除』では、QMF の旧リリース用に作成されたサンプル表のコピーを除去します。

66ページの『ステップ 12 -- QMF サンプル表の作成』では、QMF サンプル表を作成します。

QMF を旧リリースから移行する場合に、これらのステップを実行します。

QMF をデータベースに初めてインストールする場合には、66ページの『ステップ 12 -- QMF サンプル表の作成』のみを実行します。

ステップ 11 -- 旧サンプル表の削除

このステップでは、QMF の旧バージョンからの既存の QMF サンプル表を削除します。このステップでは、QMF バージョン 2 で作成された 6 つの DB2 UDB (OS/390 版) 視点は除去しません。

1. QMF710.SDSQSAPE(DSQ1EDSJ) を編集する。
2. ジョブのインストリーム・プロシージャー内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1EDSJ PROC RGN='2048K',           Job-step region size
//      QMFTPRE='QMF710',             Prefix for QMF target libraries
//      DB2EXIT='DSN710.SDSNEXIT',    Exit DB2 UDB for OS/390 library name
//      DB2LOAD='DSN710.SDSNLOAD'     DB2 UDB for OS/390 program library name
```

3. QMF710.SDSQSAPE(DSQ1EDSJ) を実行依頼する。
4. リターン・コードの 0 の受取を検査する。完了メッセージについては SYSTEMR を検討する。

QMF バッチ・インストール・ジョブの実行依頼

リターン・コードが 0 以外の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

ステップ 12 -- QMF サンプル表の作成

DSQ1EIVS では、QMF サンプル表を作成します。これらの表の詳細については、569ページの『付録B. DB2 に常駐している QMF オブジェクト』を参照してください。

分散データベースに関するヒント

サンプル表は、各ユーザーが 3 部分名を使用して別の DB2 UDB (OS/390 版) サブシステムでサンプル表を参照できるように、PUBLIC AT ALL LOCATIONS (すべてのロケーションでの公用) が許可されています。

ネットワーク内の各ロケーションの QMF ユーザーは、QMF のインストール先のロケーションで作成されたすべてのサンプル表の使用を許可されています。

1. QMF710.SDSQSAPE(DSQ1EIVS) を編集する。
2. ジョブのインストリーム・プロシージャー内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1EIVS PROC RGN='2048K',           Job-step region size
//   QMFTPRE='QMF710',                 Prefix for QMF target libraries
//   DB2EXIT='DSN710.SDSNEXIT',        Exit DB2 UDB for OS/390 library name
//   DB2LOAD='DSN710.SDSNLOAD'        DB2 UDB for OS/390 program library name
//   CDS='2',                           punctuation for decimal point
//   CDP='4'                             default is period (comma is CDS 6
//*                                       CDP 7)
```

3. QMF の別のレベルから移行する場合は、ジョブ・ステップ 1 をコメント化する。

```
//*STEP1 EXEC PGM=IKJEFT01,REGION=&RGN
```

4. メンバー QMF710.SDSQSAPE(DSQ1VSTC) を編集する。

このファイルにより、サンプル表のための記憶域グループ、データベース、および表スペースを作成します。

5. 下記のパラメーターが正しいか検査する。

```
CREATE STOGROUP DSQ1STBG
  VOLUMES (DSNVOL)      QMF tables volume
  VCAT QMFDSN;         QMF Catalog Alias in VCAT
```

6. ユーザーのユーザー・カタログがパスワード保護されているかどうかを判別する。パスワード保護されている場合は、パスワード文節をメンバー DSQ1VSTC 内の STOGROUP ステートメントに追加します。

```
CREATE STOGROUP PASSWORD(password)
```

7. ジョブ QMF710.SDSQSAPE(DSQ1EIVS) を実行依頼する。
8. リターン・コードの 0 の受取を検査する。完了メッセージについては SYSTEMM を検討する。

リターン・コードが 0 以外の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

ステップ 13 -- QMF パッケージのバインド

DSQ1BINJ は、QMF パッケージを DB2 UDB (OS/390 版) にバインドします。

1. QMF710.SDSQSAPE(DSQ1BINJ) を編集する。
2. ジョブのインストリーム・プロシージャ内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48 ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1BINJ PROC RGN='2048K',           Job-step region size
//      QMFTPRE='QMF710',             Prefix for QMF target libraries
//      DB2EXIT='DSN710.SDSNEXIT',    Exit DB2 UDB for OS/390 library name
//      DB2LOAD='DSN710.SDSNLOAD'     DB2 UDB for OS/390 program library name
```

3. QMF710.SDSQSAPE(DSQ1BINJ) を実行依頼する。
4. 4 以下のリターン・コードを検査する。

リターン・コードが 5 以上の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、また前のステップを再実行する必要がある場合は、まず QMF プランとパッケージを空にしてから、ステップ 8 から再始動します。

ステップ 14 -- DB2 UDB (OS/390 版) へのコミュニケーション・パッケージのバインド

DB2 UDB (OS/390 版) コミュニケーション・パッケージを使用しない場合は、このステップをとばして先に進んでください。

DSQ1BICD は、DB2 UDB (OS/390 版) コミュニケーション・パッケージを QMF にバインドします。

1. QMF710.SDSQSAPE(DSQ1BICD) を編集する。
2. ジョブのインストリーム・プロシージャー内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48 ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```
//DSQ1BICD PROC RGN='2048K',      Job-step region size
//  QMFTPRE='QMF710',           Prefix for QMF target libraries
//  DB2EXIT='DSN710.SDSNEXIT',   Exit DB2 UDB for OS/390 library name
//  DB2LOAD='DSN710.SDSNLOAD'    DB2 UDB for OS/390 program library name
```

3. ジョブ QMF710.SDSQSAPE(DSQ1BICD) を実行依頼する。
4. 4 以下のリターン・コードを検査する。

リターン・コードが 5 以上の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

ステップ 15 -- QMF アプリケーション・プランの DB2 UDB (OS/390 版) へのバインド

QMF を開始する前に、QMF を DB2 UDB (OS/390 版) にバインドする必要があります。QMF を DB2 UDB (OS/390 版) にバインドする前に、QMF に必要なすべての DB2 UDB (OS/390 版) リソースが、QMF をバインドするとき使用する許可 ID に対して使用可能になっている必要があります。インストール・プロセスのこの時点で、必須の DB2 UDB (OS/390 版) リソースのすべてを使用可能にしておく必要があります。

DSQ1BINR は、バインドを行うジョブです。このジョブでは、ローカル DB2 UDB (OS/390 版) サブシステムの DB2 UDB (OS/390 版) に QMF アプリケーションをバインドします。

1. QMF710.SDSQSAPE(DSQ1BINR) を編集する。

2. ジョブのインストリーム・プロシージャー内のインストール・パラメーターがユーザーの調整指定と一致するかを検査する。一致していない場合は、48ページの『ステップ 2 -- ジョブの調整』に戻り、インストール・パラメーターを訂正します。

```

//DSQ1BINR PROC RGN='2048K',           Job-step region size
//  QMFTPRE='QMF710',                 Prefix for QMF target libraries
//  DB2EXIT='DSN710.SDSNEXIT',        Exit DB2 UDB for OS/390 library name
//  DB2LOAD='DSN710.SDSNLOAD'         DB2 UDB for OS/390 program library name
    
```

3. ジョブ QMF710.SDSQSAPE(DSQ1BINR) を実行依頼する。
4. 4 以下のリターン・コードを検査する。

リターン・コードが 5 以上の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

これで、QMF を TSO または CICS 用に調整する準備が整いました。

- TSO の場合は、71ページの『第5章 QMF の TSO 用調整』をお読みください。
- CICS の場合は、81ページの『第6章 QMF の CICS 用調整』をお読みください。

第5章 QMF の TSO 用調整

この章では、QMF の TSO 用調整について説明します。この章には、次のステップが収められています。

- 『ステップ 16 -- TSO ログオン・プロシーチャーの作成』
- 76ページの『ステップ 17 -- QMF の開始』
- 79ページの『ステップ 18 -- バッチ IVP を実行するための QMF バッチ・ジョブの設定 (オプション)』

ステップ 16 -- TSO ログオン・プロシーチャーの作成

DSQ1EINV は、IBM 提供のサンプル TSO プロシーチャーです。

TSO での QMF の開始

TSO ユーザーは、2 つの方法のいずれかを使用して QMF を開始することができます。ISPF ユーザーは、ISPF SELECT サービスおよび ISPSTART コマンドを使用して QMF を開始することができます。ISPF を使用しない場合は、各ユーザーは DSQQMFE モジュールを使用することができます。ISPF ダイアログの詳細については、対話式システム生産向上機能 対話管理サービスの解説を参照してください。

QMF インストーラーとして、ユーザーは TSO ログオン・プロシーチャーを保持する必要があります。インストーラーとして TSO にログオンして端末モニター・プログラム (TMP) を開始すると、TSO ログオン・プロシーチャーが呼び出されます。

TMP は、ユーザーが TSO セッションを行う時のユーザーと端末の間の主要インターフェースです。インストール先システムは、それ独自の TMP または IBM 提供の標準のものを使用していることがあります。TMP が標準のものでない場合は、以下の説明が当てはまらない場合もあります。

TMP を呼び出すことに加えて、ログオン・プロシーチャーは、TSO セッションの開始時にリソースをそのユーザーに割り振ります。QMF ユーザーは、すべての TSO ユーザーに必要な最小のセットより多くのリソースを必要とします。ログオン・プロシーチャーを使用して、適切な TSO 環境を確立するためのこれらの追加リソースを確実に提供するようにします。

QMF の TSO 用調整

TSO ログオン・プロシーチャーは、ユーザーが TSO にログオンした時点で開始されます。このプロシーチャーが実行されると、ログオン CLIST も実行することができます。

サンプル・ログオン・プロシーチャーは、単に QMF にアクセスするための手段として TSO を使用するユーザーに、リソースを割り振ります。TSO セッションを使用してより多くのことを行いたいユーザーの場合は、より多くのリソースが必要になる可能性があります。

ログオン・プロシーチャーで割り振られるリソースの一部は、QMF を呼び出す CLIST でも割り振ることができます。

TSO ログオン・プロシーチャーの準備

1. QMF710.SDSQSAPE(DSQ1EINV) を編集する。
2. 領域パラメーターを探し出し、それが 24ページの『記憶域要件の計画』の説明のように最小記憶域要件を満たしているか確認する。

```
//DSQ1EINV EXEC PGM=IKJEFT01,TIME=1440,DYNAMNBR=30,REGION=4096K
```

3. プログラム・ロード・ライブラリーを検討する。
 - a. STEPLIB ステートメントまたは CLIST のいずれを用いてプログラム・モジュールを割り振りたいかを定める。

サンプルには、ISPF、ISPF-PDF、QMF、DB2 UDB (OS/390 版)、および GDDM 用のロード・ライブラリーが組み込まれています。これらのライブラリーのすべてが、STEPLIB ステートメント内にある必要はありません。いくつかのライブラリーは、あとで CLIST を使用して割り振ることができます。QMF を開始する前に、CLIST を使用して、ISPF および QMF ライブラリーを ISPLLIB データ・セットとして割り振ることができます。

- b. 適宜、ISPF 用に調整する。

ISPF を使用してインストールを行っている場合は、ISPF ISPLLIB DD ステートメントを使用して STEPLIB 割り振りを行うことができます。
- c. QMF の複数の並行版を同じ DB2 UDB (OS/390 版) サブシステムで実行したいかどうかを決める。

異なるプラン ID を含んだ QMF の複数の並行バージョンを同じ DB2 UDB (OS/390 版) データベースで実行したい場合は、同じ QMF ロード・ライブラリーを同じプロシーチャーで使用することはできません。次のリストは、QMF の複数のバージョンに関するロード・モジュール・ライブラリー名を示したものです。

QMF バージョン

ロード・モジュール・ライブラリー名

バージョン 7

QMF710.SDSQLOAD

バージョン 7

QMF610.SDSQLOAD

バージョン 3 リリース 3.0

QMF330.DSQLOAD

バージョン 3 リリース 2.0

QMF320.DSQLOAD

バージョン 3 リリース 1.1

QMF311.DSQLOAD

バージョン 3 リリース 1.0

QMF310.DSQLOAD

バージョン 2 リリース 2.4

QMF240.DSQLOAD

バージョン 2 リリース 3.0

QMF230.DSQLOAD

バージョン 2 リリース 2.0

QMF220.DSQLOAD

```
//*****
//*          PROGRAM LOAD LIBRARIES          *
//*****
//STEPLIB DD DSN=QMF710.SDSQLOAD,DISP=SHR      * QMF MODULES *
//          DD DSN=ISR.V4R1M0.ISRLOAD,DISP=SHR * PDF MODULES * Opt. for non-ISPF users
//          DD DSN=ISP.V4R1M0.ISPLOAD,DISP=SHR * ISPF MODULES * Opt. for non-ISPF users
//          DD DSN=DSN710.SDSNEXIT,DISP=SHR    * DB2 MODULES *
//          DD DSN=DSN710.SDSNLOAD,DISP=SHR    * DB2 MODULES *
//          DD DSN=GDDM230.SADMMOD,DISP=SHR    * GDDM MODULES *
```

4. SDSQEXCE を SYSEXEC または SYSPROC に割り振る。

インストール先によって設定された DD 名を、EXEC の TSO 検索順序に使用します。この検索順序は、TSO デフォルト・モジュールの IRXTSPRM と IRXISPRM、TSO EXECUTIL コマンド、および TSO ALTLIB コマンドでの設定値に影響を受けます。インストール先システムでの REXX EXEC に関する検索順序がわからない場合は、SDSQEXCE を SYSEXEC と SYSPROC の両方に割り振ります。

```
//*****
//*          DATASETS USED BY TSO          *
//*****
//SYSPROC DD DSN=SYS2.CLIST,DISP=SHR           * CLIST Library
//          DD DSN=QMF710.SDSQCLTE,DISP=SHR
//SYSEXEC DD DSN=SYS2.EXEC,DISP=SHR
```

```
//          DD DSN=QMF710.SDSQEXCE,DISP=SHR
//SYSHELP  DD DSN=SYS1.HELP,DISP=SHR
//EDT      DD DSN=&EDIT,UNIT=SYSDA,SPACE=(1688,(40,12))
```

5. 適宜、ISPF ライブラリーを調整する。

ISPF ライブラリーはオプションです。ISPF 関連の機能を使用する場合は、これらのライブラリーを割り振ります。

```
//*****
//*          DATASETS USED BY ISPF          *
//*****
//ISPLIB  DD DSN=QMF710.SDSQPLBE,DISP=SHR   * Panel libraries
//          DD DSN=ISR.V4R1M0.ISRPLIB,DISP=SHR
//          DD DSN=ISP.V4R1M0.ISPPLIB,DISP=SHR
//ISPMLIB DD DSN=QMF710.SDSQMLBE,DISP=SHR   * Message Libraries
//          DD DSN=ISR.V4R1M0.ISRMLIB,DISP=SHR
//          DD DSN=ISP.V4R1M0.ISPMLIB,DISP=SHR
//ISPSLIB DD DSN=QMF710.SDSQSLBE,DISP=SHR   * ISPF Skeleton Libraries
//          DD DSN=ISR.V4R1M0.ISRSLIB,DISP=SHR
//          DD DSN=ISP.V4R1M0.ISPSLIB,DISP=SHR
//ISPTLIB DD DSN=ISR.V4R1M0.ISRTLIB,DISP=SHR * Table Input Libraries
//          DD DSN=ISP.V4R1M0.ISPTLIB,DISP=SHR
//ISPPROF DD UNIT=SYSDA,SPACE=(TRK,(9,1,4)), * User's ISPF Profile Library
//          DCB=(LRECL=80,BLKSIZE=8800,RECFM=FB,DSORG=PO)
```

6. GDDM データ・セットを検査する。

これらは、ADM で始まる DD 名に割り振られています。

a. ADMGGMAP が適切に割り振られているか確認する。

ADMGGMAP ライブラリーは必ず割り振られている必要があります。

b.

独自の図表書式を保管したいユーザーすべてに、個別のライブラリーを割り振る。次のような DD ステートメントによって新規のライブラリーを作成する。

```
//DSQUCFRM DD DSN=aaaaaaaa,DISP=(NEW,CATLG),
//          UNIT=xxxx,VOL=SER=yyyy,
//          SPACE=(400,(200,50,25)),
//          DCB=(LRECL=400,BLKSIZE=400,RECFM=F)
```

DSN、UNIT、VOL、および SPACE パラメーターは指定しますが、DCB パラメーターは変更しないでください。

- 1) DSQ1EINV で DSQUCFRM の項目を見つけます。
- 2) aaaaaaa をユーザーのライブラリー名で置き換えます。
- 3) ユーザー・ライブラリーごとにこの項目を複製およびカスタマイズします。

c. ADMCDATA、ADMGDF、および ADMSYMBL 用の DD ステートメントの xxxx を、GDDM インストール時に作成されたデータ・セットの

名前で置き換える。これらのデータ・セットが存在しない場合は、次のステートメントを使用してデータ・セットを定義します。

```
//ADMCDATA DD DSN=xxxx,DISP=(NEW,CATLG),
// UNIT=xxxx,SPACE=(TRK,(5,1,10)),
// DCB=(RECFM=F,LRECL=400,BLKSIZE=400,DSORG=PO)
```

```
//*****
//*          QMF/GDDM DATA SETS          *
//*****
//ADMGGMAP DD DSN=QMF710.DSQMAPE,DISP=SHR * GDDM Map Group
//ADMCFORM DD DSN=QMF710.DSQCHART,DISP=SHR * QMF-Supplied Chart Forms
//DSQCFRM DD DSN=aaaaaaa,DISP=SHR * Saves User-Defined ICUFORMS
//ADMCDATA DD DSN=xxxx,DISP=SHR
//ADMGDF DD DSN=xxxx,DISP=SHR
//ADMSYMBL DD DSN=xxxx,DISP=SHR
```

7. QMF 設定の変更のために調整する。

DSQDEBUG、DSQDUMP、および SYSUDUMP の各データ・セットはすべて現在、プリンターにデフォルト指定されます。代替りの情報をデータ・セットに送るためにこの定義を調整することができます。

DSQDUMP、DSQDEBUG、および DSQPRINT にはすべて、DCB パラメーターが必要です。DSQPRINT の場合は、印刷制御文字用の LRECL に 1 を追加します。

```
//*****
//*          DATASETS USED BY QMF          *
//*****
//DSQPNLE DD DSN=QMF710.DSQPNLE,DISP=SHR * Panel Definition File
//DSQPRINT DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330) * Print Output
//DSQDEBUG DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210) * Trace Output
//DSQEDIT DD UNIT=SYSVIO,DCB=(RECFM=FBA,LRECL=79,BLKSIZE=4029), * Edit Transfer File
// DISP=NEW,SPACE=(CYL,(1,1))
//DSQDUMP DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=125,BLKSIZE=1632) * Snap Dump Output
//SYSUDUMP DD SYSOUT=A
//DSQPILL DD DSN=&&SPILL,DISP=(NEW,DELETE), * User's Spill File
// UNIT=SYSVIO,SPACE=(CYL,(1,1),RLSE),
// DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)
```

データ抽出プログラム (DXT)TM に関する考慮事項

管理ヘルプを使用して、別のユーザーが DXT ダイアログを開始することができます。DXT ダイアログを開始する方法の 1 つは、JCL を各ユーザーの TSO ログオン・プロシーチャーに追加することです。お勧めする方法は、IBM が QMF と一緒に提供する 2 つの CLIST を変更することです。

ステップ 17 -- QMF の開始

ログオン・プロシージャを使用して TSO にログオンすると、TSO READY モードになります。このモードから、ISPF を使用するかまたは使用せずに、QMF を開始することができます。

ISPF を使用した QMF の開始

1. 呼び出し可能インターフェースを使用してアプリケーション・プログラムから QMF を開始するか、あるいはパラメーターを指定するかまたは指定せずに ISPSTART コマンドを発行する。以下の例では、ISPSTART コマンドを使用してデータベース・サブシステム名 (DSN) およびプラン ID (QMF710) のデフォルト値を上書きする方法を示します。
 - パラメーターを指定した場合 -- ユーザーのインストール・タイプに応じて適切なコマンドを選択します。

QMF を別の DB2 UDB (OS/390 版) サブシステムにインストールする場合は、ssid の値をユーザーのサブシステム ID 値に変更する必要があります。

– 全インストール :

```
ISPSTART PGM(DSQMFE) NEWAPPL(DSQE) PARM(DSQSSUBS=ssid,DSQSPLAN=planid, ...)
```

– サーバー・インストール :

```
ISPSTART PGM(DSQMFE) NEWAPPL(DSQE)
```

– リクエスター・インストール :

```
ISPSTART PGM(DSQMFE) NEWAPPL(DSQE) PARM(DSQSSUBS=ssid,DSQSPLAN=planid,  
DSQSDBNM=<location>,...)
```

QMF ホーム・パネルが表示されます。QMF セッションが終了すると、TSO READY モードに戻ります。

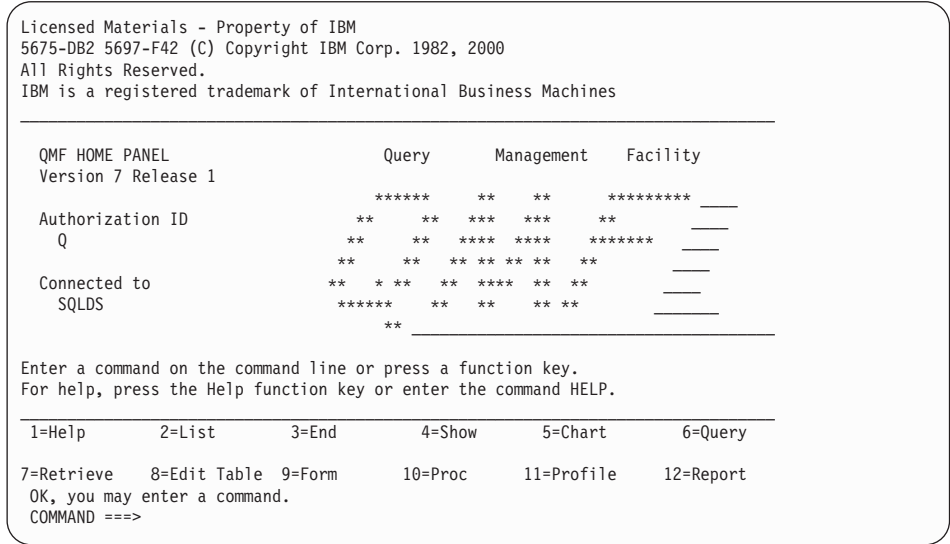


図 14. QMF ホーム・パネル

- パラメーターを指定しない場合：

```
ISPSTART
```

この場合、ISPF マスター・アプリケーション・メニューが表示されます。このメニューから QMF を選択することができます。QMF セッションが終了すると、ISPF マスター・アプリケーション・メニューに戻ります。次の節では、QMF を組み込むために ISPF 選択メニューをカスタマイズする方法について説明します。

ISPF 選択メニューのカスタマイズ

ISPF は、そのインストール・プロセスの一部としてマスター・アプリケーション・メニューを提供します。QMF の呼び出しは、ISPF マスター・アプリケーション・メニューから、またはユーザーが使用したい他の任意の選択メニューから行うことができます。78ページの図15 は、QMF を組み込むために ISPF マスター・アプリケーション・メニューをコード化する方法の 1 例を示しています。QMF の行はオプション 2 です。

QMF 呼び出し可能インターフェース REXX プロシージャ

QMF710.SDSQEXCE(DSQSCMDE) を使用して、ユーザーが TSO から QMF に渡すプログラム・パラメーターを変更することができます。プログラム・パラメーターを渡すもう 1 つの方法は、QMF が使用する ISPF サービス呼び出しを使用して行う方法です。

- リクエスター・インストールの場合 :

```
CALL 'QMF710.SDSQLOAD(DSQMFE)' 'DSQSSUBS=dbname,DSQSPLAN=planid,
DSQSDBNM=<location>...'
```

ステップ 18 -- バッチ IVP を実行するための QMF バッチ・ジョブの設定 (オプション)

このステップでは、バッチ・モードの IVP のためのバッチ・ジョブを設定します。このテストを実行しない場合は、このステップをとばしてもかまいません。このテストを実行したい場合は、108ページの『ステップ 35 -- バッチ・モード IVP の実行 (オプション)』まで待つ必要があります。このテストを早い段階で実行しようとする、プロシージャ Q.DSQ1EBAT がまだ使用可能になっていないため、このテストは失敗します。

バッチ・ジョブを作成する場合 :

1. サンプル・ログオン・プロシージャ (DSQ1EINV) のコピーを作成する。
2. JOB ステートメントを追加する。

RACF[®] 環境で作動している場合は、USER パラメーターの値をインストーラーのログオン ID にします。たとえば、インストーラーが JONES の場合は、JOB ステートメントは次のようになります。

```
//BATCH JOB USER=JONES,PASSWORD=password
```

ここで、*password* は JONES のパスワードです。

3. SYSTERM および SYSIN の各 DD ステートメントを削除する。
4. 次のステートメントをログオン・プロシージャの末尾に追加する。

```
//SYSTSPRT DD SYSOUT=A
//SYSTSIN DD *
        PROFILE PREFIX(JONES)
        ISPSTART PGM(DSQMFE) NEWAPPL(DSQE) PARM(M=B,I=Q.DSQ1EBAT,S=ssid)
/*
```

2 番目の JCL ステートメント内の最初の制御カードはオプションです。インストール先システムに RACF がインストールされていない場合は、この制御カードを使用します。JONES を、ステップを実行しているユーザーのログオン ID で置き換えます。

2 番目の JCL ステートメント内の 2 番目の制御カードにより、バッチ・モード (DSQSMODE=B) で QMF を呼び出します。ssid を、QMF のインストール先のデータベース・サブシステムのサブシステム ID で置き換えます。サブシステム ID を指定しない場合、デフォルト DSN が使用されます。このようにして呼び出されると、QMF はプロシージャ

QMF の TSO 用調整

Q.DSQ1EBAT を呼び出します。このプロシージャが呼び出されると、制御が TSO に戻ります。これにより、SYSTSIN では TSO ステートメントがこれ以上検出されないため、ジョブが終了します。

101ページの『第9章 QMF インストールのテスト』に進みます。

第6章 QMF の CICS 用調整

この章では、QMF の CICS 用調整に必要な各ステップについて説明します。

CICS での QMF の調整処理を実行する前に、CICS 用の DB2 UDB (OS/390 版) および GDDM のインストールおよび調整を完全に済ませておく必要があります。詳細については、*GDDM 導入およびシステム管理* および *DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

ステップ 19 -- CICS での DB2 UDB (OS/390 版) に対する QMF の記述

このステップでは、CICS 用に QMF を調整する前に、前提となるすべてのステップを完了していることを確認します。

1. DB2 UDB (OS/390 版) と CICS との接続機能および CICS 用の DB2 UDB (OS/390 版) 接続機能がインストールされていることを確認する。

QMF は CICS/DB2 接続機能を使用して、CICS 環境で DB2 UDB (OS/390 版) のデータにアクセスします。これらの製品に関する QMF 固有の情報については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

2. QMF トランザクション ID のプラン ID および許可 ID がリソース管理テーブル (RCT) にあるかどうかを検査する。

QMF のトランザクションを呼び出すユーザーは、関連 RCT 項目の許可のもとで操作します。(サンプル RCT については、QMF サンプル・ライブラリー QMF710.SDSQSAPE のメンバー DSQ1ERCT を参照してください。)

ユーザーの RCT に RACF 情報が含まれている場合は、その許可 ID は有効な RACF ID でなければなりません。

3. ユーザーの RCT を *DB2 UDB (OS/390 版) 管理の手引き* に説明されているように再生成してください。

リソース管理テーブル (RCT) の完全な説明については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

すべての QMF プログラムは、インストール時にバインドされます。したがって、CICS 用に個別にバインドする必要はありません。

ステップ 20 -- QMF と DFHEAI および DFHEAIO とのリンク・エディット

このステップでは、2 つのジョブ (DSQ1ELNK および DSQ1EGLK) を使用して、QMF と CICS インターフェース・モジュール (DFHEAI および DFHEAIO) とをリンク・エディットします。QMF は CICS のもとで作動するとき CICS コマンド・レベル・アプリケーション・プログラミング・インターフェースを使用するので、QMF プログラムを実行するためには、まずリンク・エディットを行う必要があります。

QMF と CICS コマンド・インターフェース・モジュールとのリンク・エディット

DSQ1ELNK では、QMF と、CICS によって生成された LOADLIB データ・セット内にある CICS コマンド・インターフェース・モジュール (DFHEAI および DFHEAIO) とをリンク・エディットします。

重要: CICS インターフェース・モジュール DFHEAI および DFHEAIO を組み込むためには、QMF サービスを適用するたびにこのステップを実行する必要があります。

1. QMF710.QMFSAMPE(DSQ1ELNK) を編集する。
2. ジョブおよびジョブ・ステップのインストリーム・プロシーチャー内のインストール・パラメーターがユーザーの調整指定と一致しているかを確認する。

```
//DSQ1ELNK PROC REG=4096K,           Job Step Region
//      QMFTPRE='QMF710',           DSN Prefix for QMF product
//      CLOAD='CICS.LOADLIB',       Name of CICS LOADLIB
//      OUTC='*'                     Print SYSOUT class
```

3. QMF710.QMFSAMPE(DSQ1ELNK) を実行依頼する。
4. 0 というリターン・コードを確認する。リターン・コードが 0 でない場合は、問題を訂正してから DSQ1ELNK を再実行します。

QMF 提供の管理プログラムの変換、アセンブル、およびリンク・エディット

DSQ1EGLK では、QMF 提供の管理プログラムに関する変換、アセンブル、およびリンク・エディットを行います。

1. QMF710.QMFSAMPE(DSQ1EGLK) を編集する。
2. ジョブおよびジョブ・ステップのインストリーム・プロシーチャー内のインストール・パラメーターがユーザーの調整指定と一致しているかを確認する。

```
//DSQ1EGLK PROC SUFFIX=1$,          CICS ASM Translator suffix
//      QMFTPRE='QMF710',           DSN Prefix for QMF product
//      CMACS='CICS.MACLIB',        Name of CICS MACLIB
```

```
//      CLOAD='CICS.LOADLIB',      Name of CICS LOADLIB
//      A=,                          A=A for CICS Aligned MAP
//      ASMBLR=IEV90,                Assembler Program Name
//      REG=4096K,                    Job step region
//      OUTC='*',                      Print SYSOUT class
//      WORK='SYSDA'                  Work unit
```

3. QMF710.QMFSAMPE(DSQ1EGLK) を実行依頼する。
4. LINKPROG (このジョブにはリターン・コード 4 が戻される可能性がある) を除くすべてのジョブについて、リターン・コード 0 を検査する。リターン・コードが 0 または 4 でない場合は、問題を訂正してからこのジョブを再実行します。

ステップ 21 -- QMF/GDDM データ・セットの定義およびロード

このステップでは、いくつかのデータ・セットを定義およびロードします。

- DSQ1EADM では、QMF/GDDM マップ・セットを GDDM ADMF データ・セットにロードします。
- DSQ1BFRM では、QMF/GDDM 図表および QMF トレース・データ・セットを作成します。

GDDM ADMF データ・セットへの QMF/GDDM マップ・セットのロード

重要: このジョブでは、既存の QMF マップがすべて置き換えられます。既存の QMF マップをすべて保持したい場合は、ADMF のバックアップ・コピーを必ず作成してください。

1. QMF710.SDSQSAPE(DSQ1EADM) を編集する。
2. ジョブおよびジョブ・ステップのインストリーム・プロシーチャー内のインストール・パラメーターがユーザーの調整指定と一致しているかを検査する。

```
//DSQ1EADM PROC RGN='2048K',        Job-step region size
//      QMFTPRE='QMF710',          QMF prefix name for target libraries
//      GDDMADM='GDDM.ADMF'        GDDM ADMF data set name
```

3. QMF710.SDSQSAPE(DSQ1EADM) を実行依頼する。
4. 0 というリターン・コードを検査する。リターン・コードが 0 でない場合は、問題を訂正してから DSQ1EADM を再実行します。

QMF/GDDM 図表および QMF トレース・データ・セットの作成

QMF の旧リリースから QMF バージョン 7 に移行する場合は、このステップをとばして先に進んでください。

QMF の CICS 用調整

DSQ1BFRM では、QMF/GDDM 図表および QMF トレース・データ・セットを作成します。

1. QMF710.SDSQSAPE(DSQ1BFRM) を編集する。
2. ジョブのインストリーム・プロシージャ内のインストール・パラメーターを探し出し、それらがユーザーの指定と一致しているかを確認する。

```
//DSQ1BFRM PROC QMFTPRE='QMF710',      DSN Prefix for QMF Product
//          GDDMADM='GDDM.ADMF',      GDDM ADMF Data Set Name
//          CHRTVOL='QMFVOL',         QMF/GDDM Charts Volume
//          TRCVOL='QMFVOL'           Trace Data Set Volume
```

3. DSQ1CFRM COPY を編集する。これは DSQ1BFRM の SYSIN で参照されます。
4. インストール先システムに合うように VSAM 制御ステートメントを調整する。

```
DEFINE CLUSTER (NAME(QMF710.DSQCFRM) -
                VOLUMES(QMFVOL) - QMF/GDDM Charts volume
                UNIQUE -
                RECSZ(400 400) -
                CONTROLINTERVALSIZE(2048) -
                KEYS(20 0)) -
                DATA -
                (RECORDS(1000 300)) -
                CATALOG(VSAMUSERCAT) VSAM user catalog
```

5. QMF710.SDSQSAPE(DSQ1BFRM) を実行依頼する。
6. 0 というリターン・コードを検査する。リターン・コードが 0 でない場合は、次のいずれのステップが正しく実行されたかを判別します。
 - DSQ1CFRM の一部が実行された場合は、DSQ1CFRM を編集し、正常に実行されたステップを除去します。それ以外の場合は、オブジェクトがすでにそこに存在していることを示すエラー・メッセージが表示されます。
 - DSQ1CFRM の全部が実行され、トレース・ファイルが割り振られた場合は、DSQ1BFRM を編集し、QMF トレース・データ・セット DSQDEBUG を作成するための最後のジョブ・ステップを除去します。

ステップ 22 -- CICS 管理テーブルの更新

CICS の現行バージョンに合致するプロシージャを選択します。

- 85ページの『CICS 管理テーブルの更新 (CICS バージョン 2)』
- 87ページの『CICS 管理テーブルの更新 (CICS バージョン 3 またはそれ以降のリリース)』

CICS 管理テーブルの更新 (CICS バージョン 2)

CICS バージョン 2 のもとで QMF を実行するためには、まず、下記の CICS 管理テーブルを変更して、QMF のプログラムおよびトランザクションを CICS に対して定義する必要があります。

- 宛先管理テーブル (DCT)
- ファイル管理テーブル (FCT)
- プログラム管理テーブル (PCT)
- 処理プログラム・テーブル (PPT)

QMF は、QMF サンプル・ライブラリー QMF710.SDSQSAPE のメンバーとして CICS 管理テーブル・ステートメントを指定することにより、このタスクを単純化します。QMF を CICS 環境で正常に実行するためには、QMF を実行するすべての CICS システムにこれらのメンバーを組み込む必要があります。

CICS 文書は、CICS の各テーブルを設定する方法に関する情報の正式なソースです。詳細については、*CICS/OS390 資源定義 (マクロ)* および *CICS/OS390 資源定義 (オンライン)* を参照してください。CICS 情報の一部は、各テーブルを使用せずにオンラインで指定することができます。

DCT (宛先管理テーブル)

QMF710.SDSQSAPE の DSQ1CDCS および DSQ1CDCT では、QMF トレース・データ・セットを CICS に対して記述します。

1. ユーザーの CICS ソースを DFHDCT 用に編集する。
2. TYPE=SDSCI のローカル項目を検出し、次の例に示すように DSQ1CSCS のコピー・ステートメントを追加する。

```

-----
*   LOCAL ENTRIES FOR TYPE=SDSCI SHOULD BE PLACED BELOW THIS BOX
-----
COPY DSQ1CDCS

```

3. QMF トレース機能をインストールする。
ローカル項目が指定されている個所を検出し、次の例に示すように TYPE=EXTRA のコピー・ステートメント (DSQ1CDCT) を追加する。

```

-----
*   OTHER LOCAL ENTRIES SHOULD BE PLACED BELOW THIS BOX
-----
COPY DSQ1CDCT

```

4. 新規の DFHDCT モジュールを作成するためにメンバーをアSEMBルおよびリンク・エディットする。

QMF の CICS 用調整

リターン・コード 0 が表示されてジョブが完了したことを確認してください。0 以外のリターン・コードが表示された場合には、リスト出力を検査し、エラーを訂正してください。

FCT (ファイル管理テーブル)

QMF710.SDSQSAPE(DSQ1EFCT) では、QMF パネル・データ・セットおよび QMF 提供の ICU 書式データ・セットを記述します。

1. ユーザーの CICS ソースを DFHFCT 用に編集する。
2. DFHFCT TYPE=SHRCTL マクロを探し出し、32K という VSAM CI サイズまたは LSRPOOL=NONE を指定するようにする。

QMF パネル・データ・セットには、32K の VSAM CI サイズが必要です。QMF は、LSRPOOL を明示的に定義しません。その代わりに、QMF により、LSRPOOL は 1 という CICS デフォルトをとることができます。インストール先システムの LSR プールが 32K 以下の場合には、32K の VSAM CI サイズをサポートする LSRPOOL を指定するか、または LSRPOOL=NONE を指定します。

3. QMF パネル・ファイルのローカル項目に、次の例に示すように DSQ1EFCT のコピー・ステートメントを追加する。

```
*-----  
* LOCAL ENTRIES SHOULD BE PLACED BELOW THIS BOX  
*-----  
COPY DSQ1EFCT
```

4. 新規の DFHFCT モジュールを作成するためにメンバーをアセンブルおよびリンク・エディットする。

リターン・コード 0 が表示されてジョブが完了したことを確認してください。0 以外のリターン・コードが表示された場合には、リスト出力を検査し、エラーを訂正してください。

PCT (プログラム管理テーブル)

QMF710.SDSQSAPE(DSQ1EPCT) では、QMF のトランザクションを CICS に対して記述します。

1. ユーザーの CICS ソースを DFHPCT 用に編集する。
2. QMF トランザクションのローカル項目に、次の例に示すように DSQ1EPCT のコピー・ステートメントを追加する。

```
*-----  
* LOCAL ENTRIES SHOULD BE PLACED BELOW THIS BOX  
*-----  
COPY DSQ1EPCT
```


3. 新規の DFHPCT モジュールを作成するためにメンバーをアSEMBルおよびリンク・エディットする。

リターン・コード 0 が表示されてジョブが完了したことを確認してください。0 以外のリターン・コードが表示された場合には、リスト出力を検査し、エラーを訂正してください。

PPT (処理プログラム・テーブル)

QMF710.SDSQSAPE(DSQ1EPPT) では、QMF のプログラムを CICS に対して記述します。

1. ユーザーの CICS ソースを DFHPPT 用に編集する。
2. QMF トランザクションのローカル項目に、次の例に示すように DSQ1EPPT のコピー・ステートメントを追加する。

```

*-----
*   LOCAL ENTRIES SHOULD BE PLACED BELOW THIS BOX
*-----
          COPY DSQ1EPPT

```

3. 新規の DFHPPT モジュールを作成するためにメンバーをアSEMBルおよびリンク・エディットする。

リターン・コード 0 が表示されてジョブが完了したことを確認してください。0 以外のリターン・コードが表示された場合には、リスト出力を検査し、エラーを訂正してください。

CICS 管理テーブルの更新 (CICS バージョン 3 またはそれ以降のリリース)

QMF を CICS/ESA® のもとで実行するためには、まず QMF を CICS に対して記述する必要があります。そのためには、管理テーブル・ステートメントと、CICS システム定義 (CSD) を更新するジョブの両方を変更する必要があります。

CICS 文書は、CICS の各テーブルを設定する方法に関する情報の正式なソースです。詳細については、*CICS/OS390 資源定義* (マクロ) および *CICS/OS390 資源定義* (オンライン) を参照してください。

DCT (宛先管理テーブル)

QMF710.SDSQSAPE の DSQ1CDCS および DSQ1CDCT では、QMF トレース・データ・セットを CICS に対して記述します。

1. ユーザーの CICS ソースを DFHDCT 用に編集する。
2. TYPE=SDSCI のローカル項目を検出し、次の例に示すように DSQ1CSCS のコピー・ステートメントを追加する。

```

*-----*
* LOCAL ENTRIES FOR TYPE=SDSCI SHOULD BE PLACED BELOW THIS BOX
*-----*

```

COPY DSQ1CDCS

3. QMF トレース機能をインストールする。
ローカル項目が指定されている個所を検出し、次の例に示すように
TYPE=EXTRA のコピー・ステートメント (DSQ1CDCT) を追加する。

```

*-----*
* OTHER LOCAL ENTRIES SHOULD BE PLACED BELOW THIS BOX
*-----*

```

COPY DSQ1CDCT

4. 新規の DFHDCT モジュールを作成するためにメンバーをアセンブルおよび
リンク・エディットする。

リターン・コード 0 が表示されてジョブが完了したことを確認してください。
0 以外のリターン・コードが表示された場合には、リスト出力を検査し、エラー
を訂正してください。

CSD の更新

DSQ1ECS D では、QMF と呼ばれる新規の LIST を作成します (これは CSD
で定義されます)。CICS は、バッチ・ジョブで CSD を更新するためのユーテ
ィリティー・プログラム (DFHCSDUP) を提供します。DFHCSDUP を使用し
て、RCT および DCT を除くすべての QMF/CICS 管理テーブルを更新してく
ださい。RCT に関するその他の考慮事項については、81ページの『ステップ
19 -- CICS での DB2 UDB (OS/390 版) に対する QMF の記述』を参照して
ください。

1. RDO VIEW Lsrpool(name) コマンドを使用して、LSRPOOL の現行定義を検
査する。
QMF パネル・データ・セットには、32K の VSAM CI サイズが必要です。
QMF は、LSRPOOL 項目を明示的に定義しません。その代わりに、
QMF は 1 という CICS デフォルトをとります。インストール先システム
の LSRPOOL が 32K 以下の場合には、DFHCSDUP を介して 32K の
VSAM CI サイズをサポートする LSRPOOL を指定してください。
2. QMF710.SDSQSAPE(DSQ1ECS D) を編集する。
3. ユーザーの調整指定と一致するように、ジョブのインストリーム・プロシー
ジャー内のインストール・パラメーターを検査または変更する。

```

//DSQ1ECS D PROC REG=2048K,           Job Step Region
//      QMFTPRE='QMF710',             DSN Prefix for QMF
//      CLOAD='CICS.LOADLIB',         Name of CICS Program Lib
//      CCSD='CICS.DFHCS D',          Name of CICS CSD file
//      OUTC='*'                       Print sysout class

```

4. ジョブを実行依頼し、リターン・コード 0 が表示されてジョブが実行されたことを検査する。0 以外のリターン・コードが表示された場合には、リスト出力を検査し、エラーを訂正してください。

注: CICS V4 およびそれ以降のリリースの場合は、DEFINE FILE(DSQPNLE) ステートメントまたは DEFINE FILE(DSQUCFRM) ステートメントからの下記のエラーを無視することができます。

```
E 'RESSECNUM' is not valid and is ignored
```

他のエラーはすべて訂正する必要があります。

ステップ 23 -- QMF プロファイルの調整

Q.PROFILE 表の ENVIRONMENT 列により、単一の AUTHID が環境 (TSO または CICS) に応じて異なるプロファイルを保持することができます。TSO のもとでインストールされると、QMF は最初は、値 NULL を ENVIRONMENT 列のすべてのものに割り当てます。次に、新規の行が、SYSTEM の AUTHID および CICS の ENVIRONMENT 項目と共に追加されます。

CICS および TSO で同じ AUTHID を使用し、かつ TSO コマンドを含むコマンド同義語を使用する場合は、下記のように、すべての NULL 項目を TSO 項目に変更します。

```
UPDATE Q.PROFILES SET ENVIRONMENT='TSO' WHERE ENVIRONMENT = NULL
```

このステートメントを入力すると、QMF は CICS 環境用に SYSTEM 行を使用します。

ステップ 24 -- CICS 始動ジョブ・ストリームの更新

このステップでは、CICS 始動ジョブ・ストリームの中になければならない DD ステートメントを更新します。

1. リンク・エディットされた RCT を含むライブラリーが通常のライブラリー検索順序 (STEPLIB、JOBLIB、リンク・ライブラリー) を使用して OS/390 にアクセス可能であるかを確認する。

```
//STEPLIB DD DSN=CICS.SDFHAUTH,DISP=SHR
//          DD DSN=DSN710.SDSNEXIT,DISP=SHR
//          DD DSN=DSN710.SDSNLOAD,DISP=SHR
```

この例では、DFHSIP (CICS.LOADLIB1 からロードする) が、許可状態で制御を受け取る必要があります。各連結済みライブラリーを個別に APF 許可する必要があります。

QMF の CICS 用調整

DSN.SDSNLOAD はリンク・エディットされた RCT を入れるライブラリーであり、これも許可を与える必要があります。

ユーザーの CICS リリースが 4.1

またはそれ以降のリリースの場合: DB2 では、DFHRPL DD ステートメント内に DB2 プログラム・ライブラリーを必要としません。ただし、QMF では初期化時に DSNHDECP 用の EXEC CICS LOAD を必要とします。したがって、QMF では、SDSNEXIT または SDSNLOAD は (カスタマイズされた DSNHDECP モジュールがどこに置かれていても) DFHRPL DD 連結内にある必要があります。これらの DB2 ライブラリーを CICS プログラム・ライブラリーの後に必ず入れてください。

2. QMF、GDDM、および DB2 UDB (OS/390 版) のモジュールを含むロード・ライブラリーを、CICS モジュール・ロード・ライブラリー・リスト DFHRPL に入れる。

```
//DFHRPL DD ...  
//      DD DSN=QMF710.SDSLOAD,DISP=SHR  
//      DD DSN=GDDM.SADMMOD,DISP=SHR  
//      DD DSN=DSN.SDSNEXIT,DISP=SHR  
//      DD DSN=DSN.SDSNLOAD,DISP=SHR
```

CICS から接続する場合は正しい DB2 UDB (OS/390 版) リリース・レベルを必ず使用してください (QMF は DSNHDECP および DSNCLI をロードします)。

3. GDDM および QMF で必要とされる下記のデータ・セットに必ずアクセスできるようにする。

```
//*      GDDM DATA SETS  
//ADMF  DD DSN=GDDM.ADMF,DISP=SHR      QMF Map Group  
//ADML  DD SYSOUT=A  
//ADMS  DD SYSOUT=A  
//ADMT  DD SYSOUT=A  
//*      QMF DATA SETS  
//DSQPNLE DD DSN=QMF710.DSQPNLE,DISP=SHR  QMF Panel File  
//DSQDEBUG DD DSN=QMF710.DSQDEBUG,DISP=SHR  Trace and Error Messages  
//DSQUCFRM DD DSN=QMF710.DSQUCFRM,DISP=SHR  User-Defined ICU Forms
```

4. 変更を CICS のテーブルおよび CICS 始動ジョブに組み込むために、CICS をシャットダウンしてから再始動する。続いて、101ページの『第9章 QMF インストールのテスト』に進みます。

第7章 QMF の ワークステーション・データ・サーバー用調整

この章では、下記のすべての DB2 製品を集合的に DB2 DRDA AS と呼びます。必要に応じて、下記のいずれかの製品に対してより明細な解説を加えます。

- DB2 ユニバーサル・データベース™ V5 (UDB (AIX 版)、UDB (OS/2 版)、DB2 (Windows NT 版)、...)
- DB2 Common Server V2.1 (DB2 (AIX 版)、DB2 (OS/2 版)、DB2 (Windows NT 版)、...)
- DB2 パラレル・エディション (Parallel Edition) V1.2
- DataJoiner® V1.2

DB2 DRDA AS に関する QMF サポートはオプションです。上記の DB2 DRDA アプリケーション・サーバーのいずれかに QMF を接続しようとする場合にのみ、この章で説明している各ステップを実行する必要があります。

QMF を OS/390 から DB2 DRDA AS にインストールする前に、DB2 Common Server、DB2 パラレル・エディション (Parallel Edition)、または DataJoiner 用に以下の準備を行う必要があります。

- DB2 DRDA AS のプラットフォームに関するインストール先システム ID を作成し、それを SYSADM GROUP のメンバーにする。
- 下記のコマンドを使用して DB2 DRDA AS のプラットフォームに関するデータベースを作成する。

```
"db2 create database" <database-name>
```

注: 通常は、認証 SERVER (これがデフォルト) を保持するためのデータベースを作成することになります。ただし、IBM コミュニケーション・マネージャー (OS/2 用) および Microsoft® SNA Server (Windows NT® 用) で制約事項を処理するパスワードのため、認証 CLIENT を保持するようにデータベースを変更する必要があります。データベース認証を設定するために使用する特定のシステム・コマンドについては、該当の DB2 コマンド解説書を参照してください。

- 下記のコマンドを使用して、DB2 DRDA AS のプラットフォームで、インストール先システム ID にローカルに接続し、その権限レベルが SYSCRTL または SYSADM であることを検査する。

QMF の ワークステーション・データ・サーバー用調整

```
"db2 connect to" <database-name>  
"user" <sysadm-id> "using" <password>
```

```
"db2 get authorizations"
```

- (オプションで) 必要に応じて、追加の管理権限をグループ、ユーザー、または PUBLIC に付与する。QMF を OS/390 から DB2 (VM または VSE 版) サーバーにインストールする場合は、共通および私用の DB スペースを作成する必要があります。QMF では、表、照会、プロシージャ、書式、およびデータのために共通 DB スペースの一部を必要とします。

以下のステップが QMF の移行と、初めて QMF をインストールする場合に適用されます。

これらのステップでは、システム・メッセージにあるステップの完了コードを検査します。完了メッセージは、指示されるように、SYSTSPRT または SYSTEMM の出力の中にあります。SYSPRINT には、IBM サポート用に診断情報が追加されています。

ステップ 25 -- DB2 DRDA AS への QMF インストール・プログラムのバインド

このステップでは、プログラム DSQCBSQL および DSQCBINS を DB2 DRDA AS にバインドします。これらのパッケージに関連付けられているアプリケーション・プランは DSQSI610 です。

1. QMF710.SDSQSAPE(DSQ1BDJ1) を編集する。
2. ジョブのインストリーム・プロシージャのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQBIND PROC RGN='2048K',      Job-step region size  
// QMFTPRE='QMF710',           Prefix for QMF target libs  
// DB2EXIT='DSN710.SDSNEXIT',   Exit DB2 library name  
// DB2LOAD='DSN710.SDSNLOAD'    DB2 program library name
```

3. <ssid> をユーザーの DB2 (OS/390 版) サブシステム ID に変更する。
4. <location> を、DB2 (OS/390 版) 通信データベースで定義された DB2 DRDA AS データベース・アプリケーション・サーバーのロケーション名に変更する。
5. (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
6. ジョブ QMF710.SDSQSAPE(DSQ1BDJ1) を実行依頼する。

- リターン・コードが 0 の場合、Procstep BIND を検査する。エラー・メッセージについて SYSTSPRT を検査する。リターン・コードが 0 以外の場合は、先に進んではなりません。必要に応じて訂正アクションを行ってから、この JOB を再実行します。

ステップ 26 -- DB2 DRDA AS での QMF 制御表の作成

このステップでは、DB2 DRDA AS での QMF 制御表を作成します。

- QMF710.SDSQSAPE(DSQ1EDJ2) を編集する。
- ジョブのインストリーム・プロシージャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQEXSQL PROC RGN='2048K',      Job-step region size
// QMFTPRES='QMF710',             Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',     Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'      DB2 program library name
```

- <ssid> をユーザーの DB2 (OS/390 版) サブシステム ID に変更する。
- (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
- ジョブ QMF710.SDSQSAPE(DSQ1EDJ2) を実行依頼する。
- リターン・コードが 0 または 4 の場合、Stepname DSQCTBL を検査する。完了メッセージについては SYSTERM を検討する。

リターン・コードが 0 または 4 以外の場合は、先に進んではなりません。エラー・メッセージについて SYSTSPRT または SYSPRINT を検査する。訂正アクションを行い、その後でこのジョブを再実行します。

ステップ 27 -- DB2 DRDA AS への QMF アプリケーション・プログラムのバインド

このステップでは、QMF アプリケーション・プログラムを DB2 DRDA AS にバインドします。このステップの正常終了後に、QMF バージョン7.1は DB2 DRDA AS に接続することができます。

- QMF710.SDSQSAPE(DSQ1BPKG) を編集する。
- ジョブのインストリーム・プロシージャーのパラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQBIND PROC RGN='2048K',      Job-step region size
// QMFTPRES='QMF710',             Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',     Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'      DB2 program library name
```

QMF の ワークステーション・データ・サーバー用調整

3. <ssid> をユーザーの DB2 UDB (OS/390 版) サブシステム ID に変更する。
4. ユーザーの DB2 UDB (OS/390 版) アプリケーション・リクエストのローカル・サブシステム ID に変更する。
5. <location> を、DB2 UDB (OS/390 版) 通信データベースで定義された DB2 DRDA AS データベース・アプリケーション・サーバーのロケーション名に変更する。
6. (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
7. ジョブ QMF710.SDSQSAPE(DSQ1BPKG) を実行依頼する。
8. リターン・コードの 0 または 4 は、上記のジョブが正常に実行されたことを示している。リターン・コードが 4 より大きい場合、エラーの原因について、SYSTSPRT、SYSTEMR および SYSPRINT の出力を調べる。訂正アクションを行ったのち、上記ジョブを再実行する。

ステップ 28 -- DB2 DRDA AS での QMF サンプル表の作成

このステップでは、DB2 DRDA AS での QMF サンプル表を作成します。

1. QMF710.SDSQSAPE(DSQ1EDJ4) を編集する。
2. ジョブのインストリーム・プロシーチャーの両方のインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQEXSQL PROC RGN='2048K',      Job-step region size
// QMFTPRE='QMF710',              Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',     Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'      DB2 program library name
```

```
//DSQINSQL PROC RGN='2048K',      Job-step region size
// QMFTPRE='QMF710',              Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',     Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'      DB2 program library name
```

3. <ssid> をユーザーの DB2 UDB (OS/390 版) サブシステム ID に変更する。
4. (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
5. ジョブ QMF710.SDSQSAPE(DSQ1EDJ4) を実行依頼する。
6. リターン・コードが 0 または 4 の場合、Stepname DSQSINS を検査する。完了メッセージについては SYSTEMR を検討する。

リターン・コードが 0 または 4 以外の場合は、エラー・メッセージについては SYSTSPRT と SYSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

DB2 DRDA AS からの QMF の削除

この節では、DB2 DRDA AS から QMF を削除する方法について説明します。

QMF の削除

このステップは、QMF をすでに含んでいる DB2 DRDA AS アプリケーション・サーバーに QMF を再インストールする場合にのみ実行する必要があります。

考慮事項: このステップでは、DB2 DRDA AS からすべての QMF 制御表およびパッケージを削除します。このステップの実行後には、QMF 6 は DB2 DRDA AS に接続することができなくなります。

1. QMF710.SDSQSAPE(DSQ1EDX1) を編集する。
2. ジョブのインストリーム・プロシーチャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQEXSQL PROC RGN='2048K',      Job-step region size
// QMFTPRES='QMF710',           Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',   Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'    DB2 program library name
```

3. <ssid> をユーザーの DB2 (OS/390 版) サブシステム ID に変更する。
4. (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
5. ジョブ QMF710.SDSQSAPE(DSQ1EDX1) を実行依頼する。
6. リターン・コードが 0 または 4 の場合、Stepname DSQCDDROP を検査する。完了メッセージについては SYSTERM を検討する。

リターン・コードが 0 または 4 以外の場合は、エラー・メッセージについては SYSTSPRT と SYSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

DB2 DRDA AS からの QMF サンプル表の削除

このステップは、QMF をすでに含んでいる DB2 DRDA AS アプリケーション・サーバーに QMF を再インストールする場合にのみ実行する必要があります。

QMF の ワークステーション・データ・サーバー用調整

このステップでは、DB2 DRDA AS からすべての QMF サンプル表を除去します。

1. QMF710.SDSQSAPE(DSQ1EDX2) を編集する。
2. ジョブのインストリーム・プロシーチャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQEXSQL PROC RGN='2048K',      Job-step region size
// QMFTPRE='QMF710',             Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',    Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'     DB2 program library name
```

3. <ssid> をユーザーの DB2 UDB (OS/390 版) サブシステム ID に変更する。
4. (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
5. ジョブ QMF710.SDSQSAPE(DSQ1EDX2) を実行依頼する。
6. リターン・コードが 0 または 4 の場合、Stepname DSQCDROP を検査する。完了メッセージについては SYSTERM を検討する。

リターン・コードが 0 または 4 以外の場合は、エラー・メッセージについては SYSTSPRT と SYSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。 .

DB2 DRDA AS に対する QMF の開始

QMF を TSO または CICS のもとで開始したと仮定すれば、QMF を DB2 DRDA AS のもとで開始したい場合には、ユーザーの START コマンドの QMF パラメーターを変更する必要があります。次のように指定してください。

```
(DSQSSUBS=<ssid>,DSQSDBNM=<location>
```

ここで、<ssid> はユーザーの DB2 UDB (OS/390 版) サブシステム ID であり、<location> はユーザーの DB2 DRDA AS ロケーション名です。

これで、101ページの『第9章 QMF インストールのテスト』に進む準備が整いました。

第8章 QMF の DB2 (AS/400® 版) サーバー用調整

バージョン 7 リリース 1 以降、QMF は、QMF アプリケーション・リクエスターから DB2 (AS/400 版) バージョン 4 リリース 4 (または それ以降のバージョン) サーバーへの接続をサポートします。このサポートはオプションです。DB2 (AS/400 版) バージョン 4 リリース 4 (または、それ以降のバージョン) サーバーへ QMF を接続しようとする場合にのみ、この章で説明している各ステップを実行する必要があります。QMF を DB2 (AS/400 版) にインストールする前に、以下の準備を行う必要があります。

- DB2 (AS/400 版) の照会マネージャーから管理権限のあるユーザー ID を使用して、SQL CREATE COLLECTION Q をサーバーで実行する。このユーザー ID は、セキュリティ担当者、または *ALLOBJ 権限を持つユーザー ID である必要がある。
- QMF ユーザーが、Q *LIB 用の *USE 権限を持っていることを確認する。

次にあげるステップは QMF をはじめてインストールする際の手順ですが、エラーを訂正するために、必要に応じて再実行することもできます。

ステップ 29 -- DB2 (AS/400 版) への QMF インストール・プログラムのバインド

このステップでは、プログラム DSQCBSQL および DSQCBINS を DB2 (AS/400 版) にバインドします。これらのパッケージに関連付けられているアプリケーション・プランは DSQSI710 です。

1. QMF710.SDSQSAPE(DSQ1BAS1) を編集する。
2. ジョブのインストリーム・プロシージャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQBIND PROC RGN='2048K',      Job-step region size
// QMFTPRE='QMF710',           Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',  Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'   DB2 program library name
```

3. ユーザーの DB2 (OS/390 版) サブシステム ID に変更する。
4. DB2 (OS/390 版) 通信データベースで定義された DB2 (AS/400 版) データベース・サーバーのロケーション名に変更する。
5. (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
6. ジョブ QMF710.SDSQSAPE(DSQ1BAS1) を実行依頼する。

QMF の DB2 (AS/400 版) サーバー用調整

7. 上記ジョブのリターン・コードを検査する。エラー・メッセージについて SYSTSPRT を検査する。リターン・コードが 0 以外の場合は、先に進んではなりません。必要な場合には訂正アクションを行い、それから上記ジョブを再実行する。

ステップ 30 -- DB2 (AS/400 版) サーバーでの QMF 制御表の作成

このステップでは、DB2 (AS/400 版) サーバーでの QMF 制御表を作成します。

1. QMF710.SDSQSAPE(DSQ1EAS2) を編集する。
2. ジョブのインストリーム・プロシーチャーのインストール・パラメーターのデフォルト値を検査および (必要な場合) 変更する。

```
//DSQEXSQL PROC RGN='2048K',      Job-step region size
// QMFTPRE='QMF710',             Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',    Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'     DB2 program library name
```

3. ユーザーの DB2 (OS/390 版) サブシステム ID に変更する。
4. (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
5. ジョブ QMF710.SDSQSAPE(DSQ1EAS2) を実行依頼する。
6. ステップ名 DSQCTBL についてリターン・コード 0 または 4 を検査する。エラー・メッセージがないかジョブ出力全体を見直す。

リターン・コードが 0 または 4 以外の場合は、先に進んではなりません。ジョブ出力全体を見直したならば、訂正アクションを行い、ジョブを再実行する。

ステップ 31 -- DB2 (AS/400 版) サーバーへの QMF アプリケーション・プログラムのバインド

このステップでは、QMF アプリケーション・プログラムを DB2 (AS/400 版) にバインドします。このステップが正常に完了したならば、QMF バージョン 7.1 は DB2 (AS/400 版) に接続することができます。

1. QMF710.SDSQSAPE(DSQ1BPKG) を編集する。
2. ジョブのインストリーム・プロシーチャーのパラメーターのデフォルト値を検査および変更 (必要な場合) する。

```
//DSQBIND PROC RGN='2048K',      Job-step region size
// QMFTPRE='QMF710',             Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',    Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'     DB2 program library name
```

3. ユーザーの DB2 UDB (OS/390 版) サブシステム ID に変更する。
4. ユーザーの DB2 UDB (OS/390 版) アプリケーション・リクエスターのローカル・サブシステム ID に変更する。
5. <location> を、DB2 UDB (OS/390 版) 通信データベースで定義された DB2 (AS/400 版) データベース・アプリケーション・サーバーのロケーション名に変更する。
6. (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
7. ジョブ QMF710.SDSQSAPE(DSQ1BPKG) を実行依頼する。
8. リターン・コードの 0 または 4 は、上記のジョブが正常に実行されたことを示している。リターン・コードが 4 より大きい場合、エラーの原因について、SYSTSPRT、SYSTEMR および SYSPRINT の出力を調べる。訂正アクションを行ったのち、上記ジョブを再実行する。

ステップ 32 -- DB2 (AS/400 版) サーバーでの QMF サンプル表の作成

このステップでは、DB2 (AS/400 版) での QMF サンプル表を作成します。

1. QMF710.SDSQSAPE(DSQ1EAS4) を編集する。
2. ジョブのインストリーム・プロシーチャーの両方のインストール・パラメーターのデフォルト値を検査および変更 (必要な場合) する。

```
//DSQEXSQL PROC RGN='2048K',      Job-step region size
// QMFTPRES='QMF710',             Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',     Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'      DB2 program library name
```

```
//DSQINSQL PROC RGN='2048K',      Job-step region size
// QMFTPRES='QMF710',             Prefix for QMF target libs
// DB2EXIT='DSN710.SDSNEXIT',     Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD'      DB2 program library name
```

3. ユーザーの DB2 UDB (OS/390 版) サブシステム ID に変更する。
4. (オプションで) これ以後に調整を行う場合に備えて JOB 内のコメントを検討する。
5. ジョブ QMF710.SDSQSAPE(DSQ1EAS4) を実行依頼する。
6. リターン・コードが 0 または 4 の場合、Stepname DSQSINS を検査する。完了メッセージについては SYSTEMR を検討する。

リターン・コードが 0 または 4 以外の場合は、エラー・メッセージがないか SYSTSPRT と SYSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

DB2 (AS/400 版) サーバーに対する QMF の始動

QMF を TSO または CICS のもとで始動する場合には、QMF を DB2 (AS/400 版) のもとで始動したいのであれば、ユーザーの START コマンドの QMF パラメーターを変更する必要があります。次のように指定してください。

```
(DSQSSUBS=<ssid>,DSQSDBNM=<location>
```

ここで、<ssid> はユーザーの DB2 UDB (OS/390 版) サブシステム ID であり、<location> はユーザーの DB2 (AS/400 版) ロケーション名です。

これで、101ページの『第9章 QMF インストールのテスト』に進むことができます。

第9章 QMF インストールのテスト

この章では、インストール・プロセスの最終ステップについて説明します。

この章には、次の各ステップが含まれています。

- 『ステップ 33 (TSO の場合) -- IVP の実行』
- 103ページの『ステップ 33 (CICS の場合) -- IVP の実行』
- ステップ 34 -- QMF アプリケーション照会およびアプリケーション・オブジェクトのインストール (TSO)
- ステップ 35 -- バッチ・モード IVP の実行 (オプション)
- ステップ 36 -- インストール後の終結処置
- ステップ 37 -- 永続ライブラリーの受け入れ
- ステップ 38 -- セキュリティーの終結処置

ステップ 33 (TSO の場合) -- IVP の実行

このステップは、QMF の最終テスト (インストール検査プロシージャー (IVP) と呼ばれる) を済ませるためのものです。OS/390 用の QMF インストールをテストするためには、QMF を開始していくつかの QMF コマンドを発行する必要があります。QMF 製品インストールの大部分は、単に QMF を開始することによってテストされます。QMF をバッチ・モードで実行したい場合は、独立した IVP があります。この IVP は対話式 IVP のあとに続きます。

1. 本書で述べられている概要に従って、基本製品に関するすべてのインストールおよび調整を完了する。
2. 適切な権限を持っているか確認する。

Q という許可 ID を使用して QMF トランザクションを開始する場合は、ユーザーにはすでに必要な DB2 UDB (OS/390 版) 権限が付与されています。許可 ID Q を使用しない場合は、少なくとも、次の SQL ステートメントによって付与される権限が必要になります。

```
GRANT SELECT ON Q.PROFILES TO authid
GRANT SELECT ON Q.ERROR_LOG TO authid
GRANT ALL ON Q.OBJECT_DIRECTORY TO authid
GRANT ALL ON Q.OBJECT_DATA TO authid
GRANT ALL ON Q.OBJECT_REMARKS TO authid
```

ここで、*authid* は 1 次許可 ID です。

QMF インストールのテスト

IVP の SAVE DATA コマンドを実行するのに十分な DB2 UDB (OS/390 版) 権限も持っている必要があります。受信側のデータベースおよび表スペースを作成した場合には、この権限をすでに持っていることになります。そうでない場合は、少なくとも、次の SQL ステートメントによって付与される権限が必要になります。

```
GRANT CREATETAB ON DATABASE dbname TO authid  
GRANT USE OF TABLESPACE dbname.table space TO authid
```

ここで、*dbname* はデータベース名、*table space* は表スペース名、および *authid* は 1 次許可 ID です。

64ページの『ステップ 10 -- QMF IVP 用の表スペースの作成』で表スペースおよびデータベースを作成したときにデフォルト値を選択した場合には、そのデータベースには DSQDBDEF、表スペースには DSQTSDEF という名前が付けられます。デフォルト値を選択しなかった場合には、これらの名前は QMF の旧版の IVP から得られたものになる可能性があります。

3. QMF を開始する。

76ページの『ステップ 17 -- QMF の開始』の説明のように、ログオン・プロシージャまたは CLIST を使用して QMF を呼び出します。

QMF ホーム・パネルが表示されます。

```
Licensed Materials - Property of IBM  
5675-DB2 5697-F42 (C) Copyright IBM Corp. 1982, 2000  
All Rights Reserved.  
IBM is a registered trademark of International Business Machines
```

```
QMF HOME PANEL  
Version 7 Release 1
```

	Query	Management	Facility
Authorization ID	*****	** **	*****
Q	** **	** **	**
Connected to	** **	** **	**
SQLDS	*****	** **	** **
	**		

```
Enter a command on the command line or press a function key.  
For help, press the Help function key or enter the command HELP.
```

1=Help	2=List	3=End	4=Show	5=Chart	6=Query
7=Retrieve	8=Edit Table	9=Form	10=Proc	11=Profile	12=Report

```
OK, you may enter a command.  
COMMAND ==>
```

図 16. QMF ホーム・パネル

ロケーション名がデータベース用に定義されていない場合には、*Connected to <location_name>* は、QMF ホーム・パネル上に表示されません。

QMF をインストールしたばかりの Workstation Database Server または DB2 (OS/390 版) データベースにユーザーが接続されていることを確認してください。必要に応じて、QMF CONNECT コマンドを使用して正しいロケーションに接続することができます。

QMF が正しく開始されない場合には、エラー・メッセージが表示されることがあります。通常のエラー状態および訂正アクションの説明については、563ページの『付録A. 問題が発生した場合』を参照してください。問題を訂正し、再び IVP を開始してください。

4. ホーム・パネルでヘルプ機能キーを押して、ヘルプ・パネルの存在の妥当性を検査する。
5. F3 または F12 を押して、ヘルプ・パネルを終了する。
6. QMF 提供のサンプル表のリストを入手する。

コマンド行に QMF コマンド LIST TABLES (OWNER=Q) を入力し、Enter キーを押します。

F8 を押すと、追加のパネルが表示されます。取り消し機能キーを押して、QMF ホーム・パネルに戻ります。F12 を押して、QMF セッションを終了します。

これで、対話モードのインストール検査が完了しました。

ステップ 33 (CICS の場合) -- IVP の実行

このステップは、QMF の最終テスト (インストール検査プロシージャ (IVP) と呼ばれる) を済ませるためのものです。MVS/CICS 用に QMF が適切にインストールされたことをテストするためには、QMF を開始していくつかの QMF コマンドを発行する必要があります。QMF 製品インストールのほとんどのエレメントは、単に QMF を開始することによってテストされます。

QMF を開始する前に

1. 本書で述べられている概要に従って、インストールおよびカスタマイズのすべてのステップを完了する。
2. データベース接続を開始する (まだ開始されていない場合)。
3. 一時データ・キュー (DSQD) を検査して、QMF トレース機能がインストールされていることを検査する。CICS クリア画面から、次のコマンドを入力します。

QMF インストールのテスト

CEMT INQUIRE QUEUE(DSQD)

次のような画面が表示されるはずですが。

```
STATUS:  RESULTS  - OVERTYPE to MODIFY
Que(DSQD)      Ext  Ena  Ope
```

Ena Ope は、キューがオープンされていて使用可能になっていることを示します。DSQD がオープンされていて使用可能になっていることが表示されていない場合には、CICS DCT に加えた変更を検討する必要があります。QMF トレース・ファイルが正しくインストールされているか検査します。詳細については、84ページの『ステップ 22 -- CICS 管理テーブルの更新』を参照してください。

QMF の開始およびテスト

このプロシージャでは、MVS/CICS 製品用に QMF を開始し、その製品が適切にインストールされているかテストします。このプロシージャのいずれかの段階でエラー・メッセージが表示された場合、そのエラー・メッセージは、QMF が適切に開始されなかったことを示しています。このような状況下では、563ページの『付録A. 問題が発生した場合』で説明しているような、より一般的な問題の調査を始めてください。

1. QMF に接続されている CICS システムにサインオンする。
2. Esc (エスケープ) 機能キーを押して、ネイティブ CICS セッションを開始する。
3. CICS トランザクション QMFE を発行して、QMF を開始する。警告メッセージをオンラインで表示できるように一時記憶域キュー (DSQSDBQT) の使用も指定します。一時記憶域キュー名 (DSQD) を使用して QMF を開始するために、次のように指定します。

```
QMFE DSQSDBQT=TS,DSQSDBQN=DSQD
```

QMF ホーム・パネルが表示されます。

```

Licensed Materials - Property of IBM
5675-DB2 5697-F42 (C) Copyright IBM Corp. 1982, 2000
All Rights Reserved.
IBM is a registered trademark of International Business Machines

-----
QMF HOME PANEL                      Query      Management  Facility
Version 7 Release 1

Authorization ID                      *****
Q                                     ** ** *****
** ** ** ** ** ** ** ** ** ** ** **
** ** ** ** ** ** ** ** ** ** ** **
Connected to                          ** * ** ** ** ** ** ** **
SQLDS                                 ***** ** ** ** ** **
**
-----

Enter a command on the command line or press a function key.
For help, press the Help function key or enter the command HELP.

-----
1=Help      2=List      3=End      4=Show      5=Chart      6=Query
7=Retrieve  8=Edit Table  9=Form     10=Proc     11=Profile   12=Report
OK, you may enter a command.
COMMAND ==>
    
```

4. QMF オンライン・ヘルプの存在を検査する。
ヘルプ機能キーを押します。次のヘルプ・パネルが表示されます。

```

-----
Licensed Materials - Property of IBM
5645-DB2 5648-A70 (C) Copyright IBM Corp. 1982, 1998
All Rights Reserved.
IBM is a registered trademark of International Business Machines
-----+-----
                                Help: Query Management Facility
-----+-----
Select a topic.                                                         1 to 7 of 14
1. What's new in Version 7
2. Profile
3. QMF commands
4. Prompted Query
5. SQL (Structured Query Language)
6. Table Editor
7. Forms
-----+-----
| F1=Help  F3=Exit  F7=Backward  F8=Forward  F9=Keys  F12=Cancel |
-----+-----

OK, HELP performed. Please proceed.
    
```

- PF3 または PF12 を押して、ヘルプ・パネルを終了します。
5. QMF 提供のサンプル表のリストを入手する。

QMF インストールのテスト

コマンド行に QMF コマンド LIST TABLES (OWNER=Q) を入力し、Enter キーを押します。QMF を以前にインストールしたか否かによって、所有者 Q を含んだ表が、次の画面とは異なることがあります。

Table List		
Action	Name	Owner
	APPLICANT	Q
	COMMAND_SYNONYMS	Q
	DSQ_RESERVED	Q
	DSQEC_ALIASES	Q
	DSQEC_COLS_LDB2	Q
	DSQEC_COLS_RDB2	Q
	DSQEC_QMFobjs	Q
	DSQEC_TABS_LDB2	Q
	DSQEC_TABS_RDB2	Q
	INTERVIEW	Q
	ORG	Q
	PARTS	Q

1 to 7 of 36

F1=Help F4=Command F5=Describe F6=Refresh F7=Backward F8=Forward
F9=Clear F10=Comments F11=Sort F12=Cancel

OK, your database object list is displayed.

PF8 を押すと、追加のパネルが表示されます。取り消し機能キーを押して、QMF ホーム・パネルに戻ります。PF12 を押して、QMF セッションを終了します。

これで、インストール検査は完了しました。CICS トランザクションを使用して QMF 警告メッセージがあるかどうかを判別するために、一時記憶域キューをブラウズすることができます。

CEBR DSQD

IVP がエラーなしで実行された場合には、一時記憶域キュー DSQD は空です。

ステップ 34 -- QMF アプリケーション照会およびアプリケーション・オブジェクトのインストール (TSO)

このステップでは、QMF アプリケーションのサンプル照会およびサンプル・プロシーチャーを更新します。これらのアプリケーションには、印刷報告書の表示 (DPRE)、レイアウト、および文書インターフェースが含まれています。オプションのバッチ IVP では、このテストの一部としてこれらのサンプル照会およびサンプル・プロシーチャーを使用します。

QMF が正常にインストールおよびテストされた後で、QMF を使用して、QMF 提供のサンプル照会、サンプル・プロシージャ、およびコマンド同義語を作成することができます。

1 つまたは 2 つの QMF プロシージャを実行することにより、このステップを完了することができます。

プロシージャ

説明

DSQ1ESQD QMF の旧版からサンプル照会およびサンプル・プロシージャを削除します。

DSQ1ESQI 新規のサンプル照会およびサンプル・プロシージャを QMF データベースに追加する。

1. 現行のサンプル照会およびサンプル・プロシージャを削除する。

既存の QMF リリースがシステム上にない場合、または旧版が別の DB2 UDB (OS/390 版) サブシステム内にある場合は、このステップをとばして先に進んでください。(削除すべきものは何もありません。)

- a. QMF セッションを開始する。
- b. QMF をインストールしたばかりの Workstation Database Server または DB2 (OS/390 版) サーバーに接続する。
- c. QMF 内から次のコマンドを入力する。

```
IMPORT PROC FROM 'QMF710.SDSQSAPE(DSQ1ESQD)'
```

ここで、*QMF710* は QMF データ・セットの接頭部です。別の接頭部を使用した場合には、それに応じて名前を変更してください。

- d. プロシージャを実行する。

2. サンプル照会およびサンプル・プロシージャを QMF データベースに追加する。

QMF セッションで次のコマンドを入力します。

```
IMPORT PROC FROM 'QMF710.SDSQSAPE(DSQ1ESQI)'
```

ここで、*QMF710* は QMF データ・セットの接頭部です。別の接頭部を使用した場合には、それに応じて名前を変更してください。

3. 各オブジェクトが正しくインストールされたことを示すメッセージが表示されているか検査する。

障害が生じた場合は、部分的に作成されたオブジェクトを削除するために最初の実行ステップを再実行します。続いて、2 番目のステップを実行します。

ステップ 35 -- バッチ・モード IVP の実行 (オプション)

インストール先システムでバッチ・モードの QMF を使用しない場合は、このステップをとばして先に進んでください。

このステップでは、79ページの『ステップ 18 -- バッチ IVP を実行するための QMF バッチ・ジョブの設定 (オプション)』で作成したバッチ・モード・ジョブを実行することにより、バッチ・モード IVP をテストします。このジョブでは、QMF がプロシージャ Q.DSQ1EBAT を実行するバックグラウンド TSO セッションを開始します。このプロシージャでは、バッチ・モード IVP を処理し、次のバッチ・モード操作をテストします。

- QMF のアクセスおよび開始
- 照会のインポート、保管、実行、および削除
- 新規の表の保管、検索、および削除
- 照会の印刷
- 照会のエクスポート、および QMF TSO コマンドを使用した照会の削除

IVP がエラーなしで実行され、次の照会が印刷されると、IVP は正常に実行されたこととなります。

```
DELETE FROM &NAME  
WHERE OWNER = USER AND NAME = 'QMF_IVPQUERY'
```

1. JCL を検査する。

QMF がバッチ・モードで必要とするリソースと対話式に必要とするリソースは、基本的に同じものです。サンプル TSO ログオン・プロシージャからバッチ・ジョブを作成することができます。ユーザーのバッチ・ジョブが必ず DSQPRINT を割り振るようになっています。QMF PRINT コマンドからの出力は、このファイルに入れられます。

2. QMF プロシージャ Q.DSQ1EBAT を検査する。

Q.DSQ1EBAT は、サンプル照会およびサンプル・プロシージャを使用して作成されています。このプロシージャは SHARE=YES を指定して保管されました。したがって、このプロシージャを画面上で検査および編集することができます。QMF データ・セットの接頭部として QMF710 を使用しない場合には、このプロシージャの IMPORT コマンドを変更する必要があります。この IMPORT コマンドを使用して、QMF サンプル・ライブラリーから照会を取り出します。

このプロシージャを変更する場合は、ユーザー自身のログオン ID のもとでこのプロシージャを保管する必要があります。プロシージャを変更する場合は、必ず SHARE=YES を指定してください。次に、QMF を ISPF ダイアログとして開始した場合には、このプロシージャの新規の所有権を反映するようにバッチ IVP JCL の ISPSTART ステートメントを変更する

必要があります。たとえば、ユーザーのログオン ID が JONES の場合は、変更されたステートメントは次のようになります。

```
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSMODE=B,DSQSRUN=JONES.DSQ1EBAT)
```

3. ジョブを実行する。
4. 印刷された出力、表 Q.ERROR_LOG、および DSQDEBUG データ・セットにエラーがないかどうか検査する。Q.ERROR_LOG または DSQDEBUG でエラーが記録されている場合は、ヘルプ・コマンドを使用して、該当するメッセージ・ヘルプ・パネルを参照することができます。

ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

ステップ 36 -- インストール後の終結処置

QMF の旧リリースがインストールされていない場合には、このステップをとばして先に進んでください。

考慮事項: このステップでは、QMF の旧リリースを除去します。旧バージョンが必要でなくなったことを確認するまで、このステップを実行してはなりません。

次のいずれかのプロシージャを選択してください。

- 旧アプリケーション・プランの解放
このステップでは、QMF バージョン 7 と旧リリースが同じ DB2 UDB (OS/390 版) サブシステム内にある場合にその旧リリースを除去します。
- QMF バージョン 7.1 と前のリリースが、異なる DB2 UDB (OS/390 版) サブシステム内にある場合
このステップでは、QMF バージョン 7 と旧リリースが異なる DB2 UDB (OS/390 版) サブシステム内にある場合にその旧リリースを除去します。

この 2 つのサブステップのいずれかを実行した後で、QMF の旧リリースの各ライブラリーを削除することができます。110ページの表13 は、デフォルトの接頭部を含んだライブラリーを列挙したものです。インストール先システムでの名前は、この図に示されている名前とは異なっている可能性があります。

QMF インストールのテスト

考慮事項: バージョン 6 のデータ・セットを削除することのないよう接頭部には特に注意を払ってください。

表 13. QMF の旧リリースから削除されるライブラリー

V2R4 データ・セット	V3RxMy データ・セット
QMF240.DSQOBJ	QMF3xy.ADMFE
QMF240.DSQMACE	QMF3xy.CICS.DFHTEMP
QMF240.DSQPMSE	QMF3xy.DSQPMSE
QMF240.DSQDBRMD	QMF3xy.DSQDBRMD
QMF240.DSQSAMPE	QMF3xy.DSQSAMPE
QMF240.DSQMAPE	QMF3xy.DSQMAPE
QMF240.DSQCLSTE	QMF3xy.DSQCLSTE
QMF240.DSQEXECE	QMF3xy.DSQEXECE
QMF240.DSQUSERE	QMF3xy.DSQUSERE
QMF240.DSQPLIBE	QMF3xy.DSQPLIBE
QMF240.DSQSLIBE	QMF3xy.DSQSLIBE
QMF240.DSQMLIBE	QMF3xy.DSQMLIBE
QMF240.DSQLOAD	QMF3xy.DSQLOAD
QMF240.DSQDBRM	QMF3xy.DSQDBRM
QMF240.DSQTLIBE	QMF3xy.DSQTLIBE
QMF240.DSQCHART	QMF3xy.DSQCHART
	QMF3xy.DSQMACE
	QMF3xy.DSQOBJ
	QMF3xy.DSQPNLE
	QMF3xy.DSQPVARE
	QMF3xy.DSQUCFRM

旧アプリケーション・プランの解放

QMF バージョン 7 と旧リリースが同じ DB2 UDB (OS/390 版) サブシステム内にある場合のみ、このステップを実行します。

1. QMF710.SDSQSAPE(DSQ1JFPL) を編集する。
2. ユーザー・サイトの規則に適合するようにジョブ・ステートメントを変更する。
3. ジョブのインストリーム・プロシージャ内のパラメーターの値を検査または変更する (必要な場合)。

```
//DSQ1JFPL PROC RGN='2048K',           Job-step region size
//                QMFTPRES='QMF710',     QMF prefix
//                DB2EXIT='DSN710.SDSNEXIT', DB2 UDB for OS/390 exit library
//                DB2LOAD='DSN710.SDSNLOAD' DB2 UDB for OS/390 program library
```

4. QMF710.SDSQSAPE(DSQ1DEL1) を編集する。
5. DSN を DB2 UDB (OS/390 版) サブシステムの名前で置き換え、QMF330 を旧リリースのアプリケーション・プランの名前で置き換える。

DSN SYSTEM(DSN)
FREE PLAN(QMF330)

表 14. QMF の各リリースのデフォルト

旧リリース	デフォルト
QMF 3.3 バージョン 3.3	QMF330
QMF バージョン 3.2	QMF320
QMF バージョン 3.1.1	QMF311
QMF バージョン 3.1	QMF310
QMF バージョン 2.4	QMF240

6. ジョブ QMF710.SDSQSAPE(DSQ1JFPL) を実行依頼する。

ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

QMF バージョン 7.1 と前のリリースが、異なる DB2 UDB (OS/390 版) サブシステム内にある場合

QMF バージョン 7 と旧リリースが、異なる DB2 UDB (OS/390 版) サブシステム内にある場合のみ、このステップを実行します。このステップでは、旧アプリケーション・プランを解放し、QMF の旧リリースに属するさまざまな DB2 UDB (OS/390 版) エンティティーを除去します。

考慮事項: このジョブでは、DB2 UDB (OS/390 版) サブシステムから QMF のすべてのトレースを除去します。また、このジョブは、QMF の現行リリースが DB2 UDB (OS/390 版) サブシステム内に存在しない場合のみ実行する必要があります。

1. QMF710.SDSQSAPE(DSQ1DELA) を編集する。
2. ユーザー・サイトの規則に適合するようにジョブ・ステートメントを変更する。
3. ジョブのインストリーム・プロシージャー内のパラメーターの値を検査または変更する (必要な場合)。

```
//DSQ1DELA PROC RGN='2048K',           Job-step region size
//          QMFTPRE='QMF710',         QMF prefix
//          DB2EXIT='DSN710.SDSNEXIT', DB2 UDB for OS/390 exit library
//          DB2LOAD='DSN710.SDSNLOAD'  DB2 UDB for OS/390 program library
```

4. メンバー QMF710.SDSQSAPE(DSQ1DEL1) を編集する。
5. DSN を DB2 UDB (OS/390 版) サブシステムの名前で置き換え、QMF311 を旧リリースのアプリケーション・プランの名前で置き換える。

DSN SYSTEM(DSN)
FREE PLAN(QMF320)

QMF インストールのテスト

表 15. QMF の各リリースのデフォルト

旧リリース	デフォルト
QMF 3.3 バージョン 3.3	QMF330
QMF バージョン 3.2	QMF320
QMF バージョン 3.1.1	QMF311
QMF バージョン 3.1	QMF310
QMF バージョン 2.4	QMF240
QMF バージョン 2.3	QMF230
QMF バージョン 2.2	QMF220

6. メンバー QMF710.SDSQSAPE(DSQ1DEL2) を編集する。

このメンバーには、視点、表スペース、データベース、および記憶域グループを除去するための SQL ステートメントが含まれています。

QMF の旧リリースに、該当のユーザーの SAVE DATA コマンドおよび IVP (64 ページのステップ 17) に関する受信側の表スペースが含まれていない場合は、次のステートメントを削除してください。

```
DROP STOGROUP DSQSGDEF
```

```
QMF710
```

7. メンバー QMF710.SDSQSAPE(DSQ1DEL13) を編集する。

このメンバーには、QMF 制御表に関するユーザー管理のデータ・セットを削除するためのステートメントが含まれています。このデータ・セットが DB2 管理のものである場合は、このステップを実行する必要はありません。

8. ジョブ QMF710.SDSQSAPE(DSQ1DELA) を実行依頼する。

ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

リクエスター・データベース・インストールまたはサーバー・データベース・インストールを行っている場合は、113 ページの『ステップ 38 -- セキュリティーの終結処置』に進んでください。

ステップ 37 -- 永続ライブラリーの受け入れ

DSQ1EJAC では、SMP/E ACCEPT ジョブを実行します。このジョブは、各ライブラリーを永続的なものにします。

考慮事項: QMF バージョン 7 が旧リリースとして同じ DB2 UDB (OS/390 版) サブシステム内にインストールされている場合には、ローカル受け入れ検査が完了するまで永続ライブラリーを受け入れてはなりません。IVP が正常に実行された後の任意の時点で各ライブラリーを受け入れることができます。

ステップ 38 -- セキュリティーの終結処置

JCL には現在、有効なユーザー ID およびパスワードが含まれています。これにより、機密漏れが生じます。この機密漏れをできるだけ早く訂正してください。考えられる解決法の 1 つは、JCL を編集し、パスワード値を空白にすることです。

インストール制御ファイルには、DB2 UDB (OS/390 版) カタログ用のパスワードならびにすべての QMF 制御表スペース用のパスワードが含まれています。これらのパスワードを削除するか、またはこれらのパスワードへのアクセスを制限してください。インストール制御ファイルは QMF710.SDSQCLTE(XXXXINST) に入っています。ここで、『XXXX』は DB2 UDB (OS/390 版) サブシステム ID です。

第10章 QMF NLF の計画およびインストール

QMF NLF は、特定言語の要件に合うように設定された QMF 環境を提供するソフトウェアです。

例

ドイツ語の環境で QMF を操作するときには、QMF のコマンド、キーワード、ウィンドウ、およびメッセージはドイツ語で表示されます。

一般に、基本英語セッションで使用できる QMF 機能は NLF セッションで実行することができ、この逆方向の実行も可能です。

この章では、基本 QMF 製品に必要なインストール手順を同じようなものとして示します。プロシージャに著しい相違点がある場合は、NLF のインストール・プロシージャについて説明を加えます。ジョブ、ライブラリー、またはプログラム名が異なる場合は、適宜、名前を示しますが、後に続く手順は QMF Program Directory の中で説明します。

モジュール、ライブラリー、またはジョブ名には n が含まれていますが、これは各国語 ID です。製品の出荷前に、このシンボル n は実際の NLF ID に置き換えられます。したがって、ユーザーは、このシンボルを置き換える必要はありません。(それぞれの NLF の FMID 値のリストについては、118ページの表20 を参照してください。)

プロファイル表と NLF

NLF のインストール時には、NLF が使用できるように QMF プロファイル表 (Q.PROFILES) に 3 行が追加されます。TSO、CICS、および CMS の環境では、これらの行はユーザー ID の SYSTEM と共に挿入されます。インストールするそれぞれの NLF ごとに固有の行が追加されます。

NLF は NLF が使用されるそれぞれの DB2 サブシステムの中にインストールする必要があります。NLF の JCL と制御ステートメントは、該当機能の IBM ソフトウェア配布 (ISD) テープに入れて出荷されます。

QMF NLF の計画およびインストール

注:

1. QMF NLF をインストールする 直前 に QMF 7 基本製品を受け取っていることが必要です。
2. このインストールでは、QMF が DB2 SMP/E データ・セットを共用するものと見なしています。

QMF NLF の計画

この節では、ハードウェアとプログラム製品の要件、SMP/E 要件、配布ライブラリー、ターゲット・ライブラリー、および NLF 用のユーザー・データ・セットについて説明します。

ハードウェアとプログラム製品の要件

GDDM と ISPF 環境がコントローラー、端末、キーボードと同じように、インストールしようとしている各国語機能の国別文字を表示するように設定されていることを確認してください。

SMP/E 要件

SMP/E データ・セット、配布ライブラリー、ターゲット・ライブラリー、およびユーザー・データ・セットには、追加の DASD スペースが必要です。QMF NLF の配布ライブラリー、ターゲット・ライブラリー、ユーザー・ライブラリー用の、以下に示す DASD スペースは、基本 QMF 製品のインストールに必要なものに追加されるものです。基本 QMF をインストールするための SMP/E 要件については、26ページの『SMP/E 記憶域の見積もり』を参照してください。QMF およびその機能は SMP/E データ・セットに追加されます。

QMF NLF 用の SMP/E データ・セット

SMP/E データ・セットに必要な、追加の推定 DASD スペース (シリンダー数) は表16 に示されています。

表 16. SMP/E データ・セット用の追加 DASD スペース (シリンダー数)

DDNAME	3380	3390	9345
SMPCDS	1	1	1
SMPLOG	1	1	1
SMPMTS	1	1	1
SMPPTS	1	1	1
SMPSTS	1	1	1
SMPCSI	8	8	8

QMF NLF の配布ライブラリー

NLF の QMF 7 配布ライブラリーは次のとおりです。

- QMF710.ADSQMACn。QMF NLF インストール・プロシージャ、IVP、サンプル照会、および QMF プロシージャが入っています。
- QMF710.ADSQPMSn。QMF NLF の ISPF パネルが入っています。

QMF NLF 配布ライブラリーと必要な追加の推定 DASD スペース (シリンダー数) は 表17 に示されています。

表 17. QMF NLF 配布ライブラリー用の追加 DASD スペース (シリンダー数)

DSNAME	内容	3380	3390	9345
QMF710.ADSQMACn	QMF NLF インストール・プロシージャ	15	13	15
QMF710.ADSQPMSn	QMF NLF ISPF パネル	1	1	1

QMF NLF のターゲット・ライブラリー

QMF NLF ターゲット・ライブラリーに必要な、追加の推定 DASD スペース (シリンダー数) は 表18 に示されています。

表 18. QMF NLF ターゲット・ライブラリー用の追加 DASD スペース (シリンダー数)

DSNAME	3380	3390	9345
QMF710.SDSQSApN	17	15	17
QMF710.SDSQPLBn	1	1	1
QMF710.SDSQCLTn	2	1	2
QMF710.SDSQMLBn	1	1	1
QMF710.SDSQEXCn	1	1	1
QMF710.SDSQUSRn			

QMF NLF の計画およびインストール

QMF NLF ユーザー・データ・セット

QMF NLF ユーザー・データ・セットに必要な推定 DASD スペース (シリンダー数) は 表19 に示されています。

表 19. QMF NLF ユーザー・データ・セット用の DASD スペース (シリンダー数)

DSNAME	内容	3380	3390	9345
QMF710.DSQMAPn	GDDM マップ・グループ・ファイル	1	1	1
QMF710.DSQPVARn	QMF メッセージ・ヘルプ・パネル (順次フォーマットで拡張)	6	6	なし
QMF710.DSQPNLn	QMF メッセージ・ヘルプ・パネル (VSAM フォーマットで拡張)	10	9	なし

IBM ソフトウェア配布 (ISD) テープ

QMF NLF をインストールするためには、まず IBM ISD テープの内容をお読みください。このテープの内容は次のとおりです。

- SMP/E 制御ステートメント
- QMF 7 NLF の JCLIN
- インストール検査プロシージャの JCL
- ロード・モジュール・フォーマットのプログラム
- QMF 7 NLF が使用するパネルおよびその他の項目

ISD テープは SMP/E (RELFILE) フォーマットです。このフォーマットは、OS/390 SMP/E 解説書 に詳しく説明されています。

FMID

機能変更 ID (FMID) は、QMF NLF を SMP/E に識別させるために使用します。それぞれの NLF の言語 ID と FMID は、表20 に示されています。

表 20. 言語 ID と FMID

各国語機能	言語 ID	QMF 7 FMID
U/C 英語	U	JSQ7751
中国語 (簡体字)	R	JSQ7753
デンマーク語	Q	JSQ7755
フランス語	F	JSQ7756
ドイツ語	D	JSQ7757

表 20. 言語 ID と FMID (続き)

各国語機能	言語 ID	QMF 7 FMID
イタリア語	I	JSQ7758
日本語漢字	K	JSQ7759
韓国ハングル語	H	JSQ775A
ブラジル・ポルトガル語	P	JSQ775B
スペイン語	S	JSQ775C
スウェーデン語	V	JSQ775D
スイス・フランス語	Y	JSQ775E
スイス・ドイツ語	Z	JSQ775F
カナダ・フランス語	C	JSQ775G

SMP/E は、プログラムのすべての変更をそのプログラムのシステム・リリース・レベル (SREL) に結び付けます。QMF のシステム・リリース・レベルは P115 です。

テープ上のファイルは、最初のを除き、IEBCOPY アンロード区分データ・セットであり、NLF 配布ライブラリーに対応しています。最初のデータ・セットには、NLF に対する SMP/E 制御ステートメントが含まれています。このテープには、インストールに必要なプロシージャとデータがすべて入っています。

インストール・プロセス

インストール手順の概要は次ページに示されています。インストール手順を完全に理解するためには、本書の第 2 章の説明をお読みください。NLF の Program Directory も参照してください。

NLF JCL および制御ステートメントは ISD テープに入れて出荷されます。次に示す図は、以前の QMF NLF リリースの有る無しとは無関係に、QMF 7 を DB2 サブシステムにインストールするために必要な手順を示したものです。

NLF では、QMF 7 サンプル・ライブラリー QMF710.SDSQSAPE およびロード・モジュール・ライブラリー QMF710.SDSQLOAD を使用する必要があります。

QMF NLF の計画およびインストール

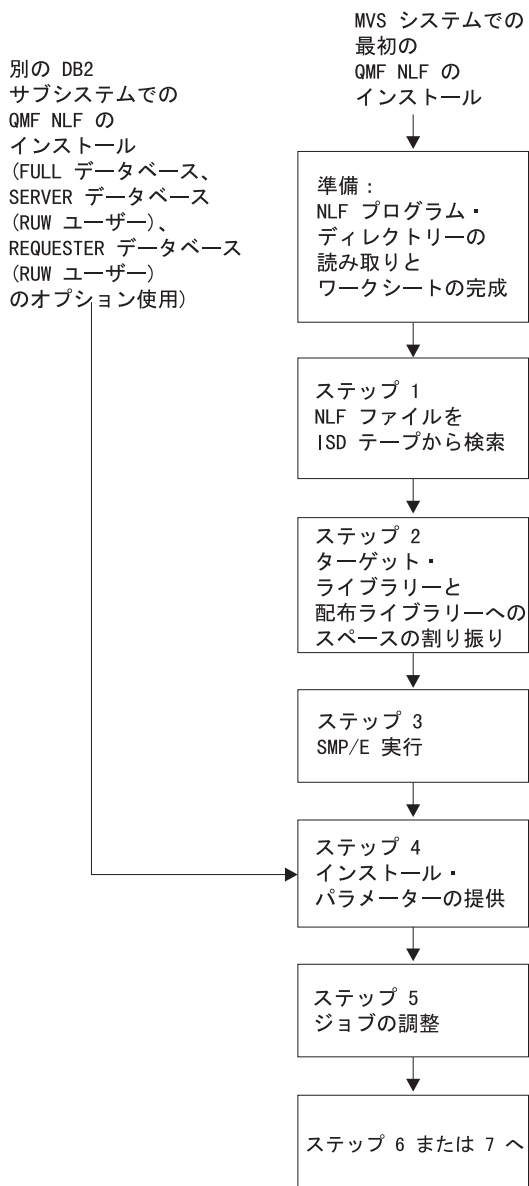


図 17. QMF NLF のインストール手順 -- その 1

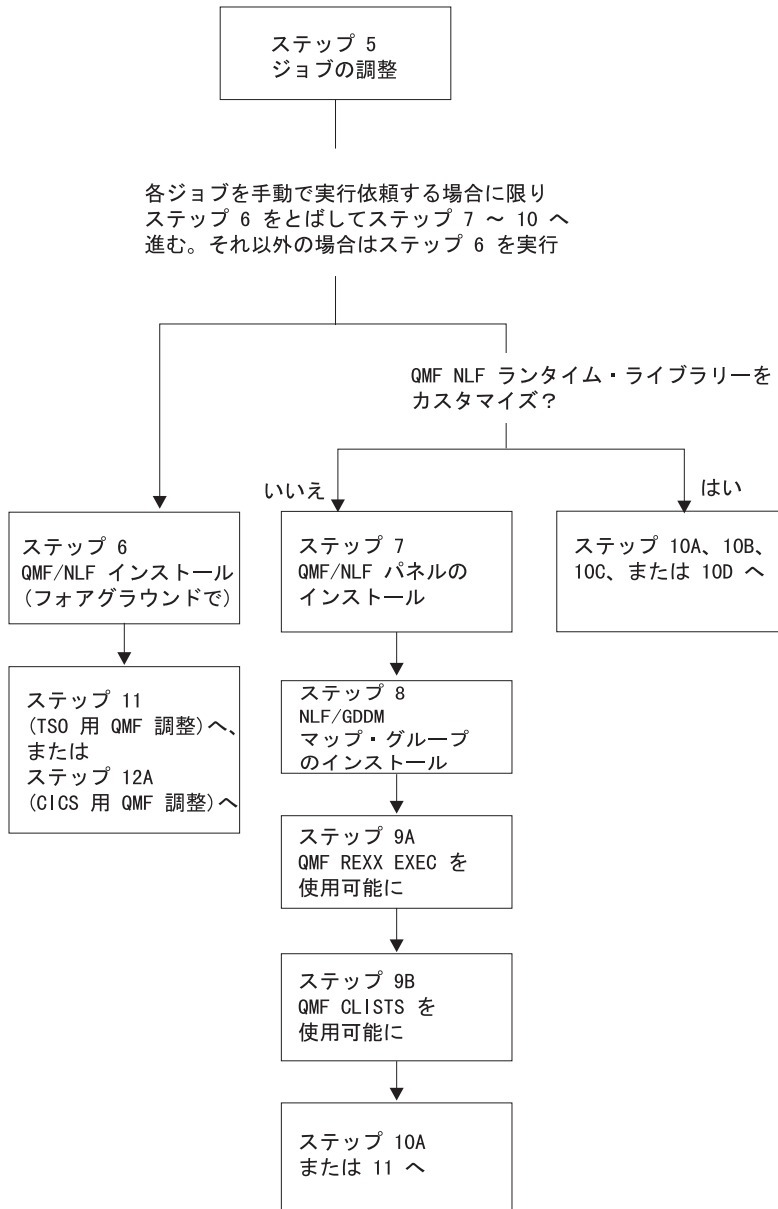


図 18. QMF NLF のインストール手順 -- その 2

QMF NLF の計画およびインストール

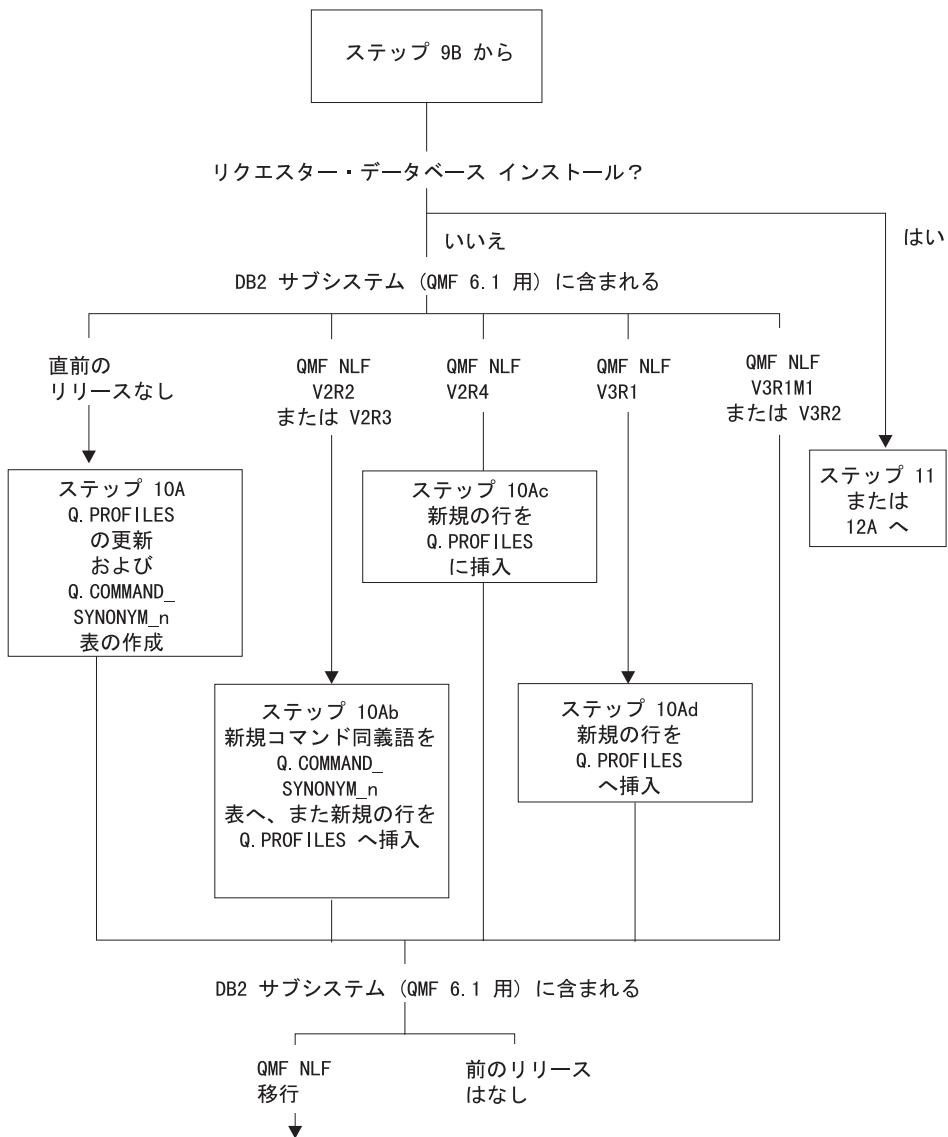


図 19. QMF NLF のインストール手順 -- その 3

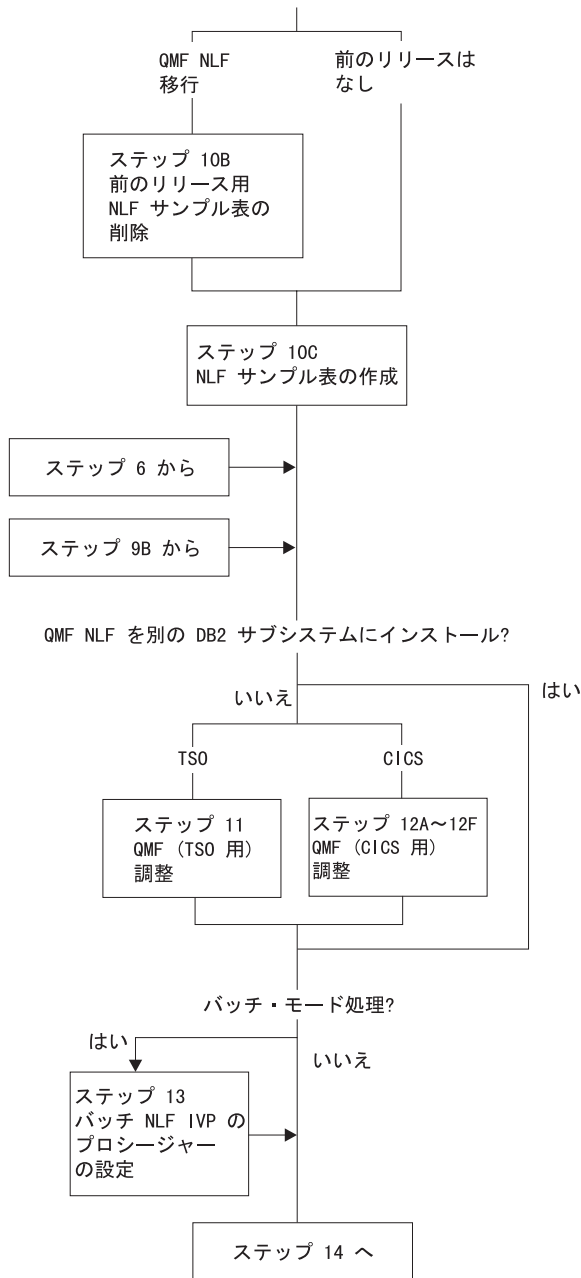


図 20. QMF NLF のインストール手順 -- その 4

QMF NLF の計画およびインストール

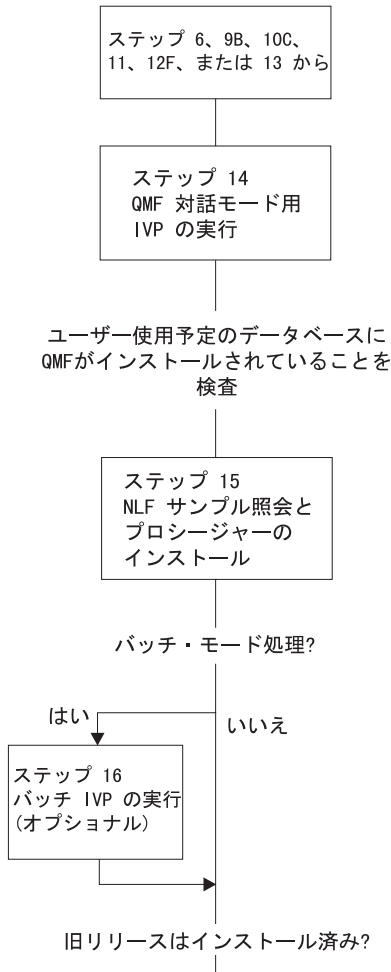


図 21. QMF NLF のインストール手順 -- その 5

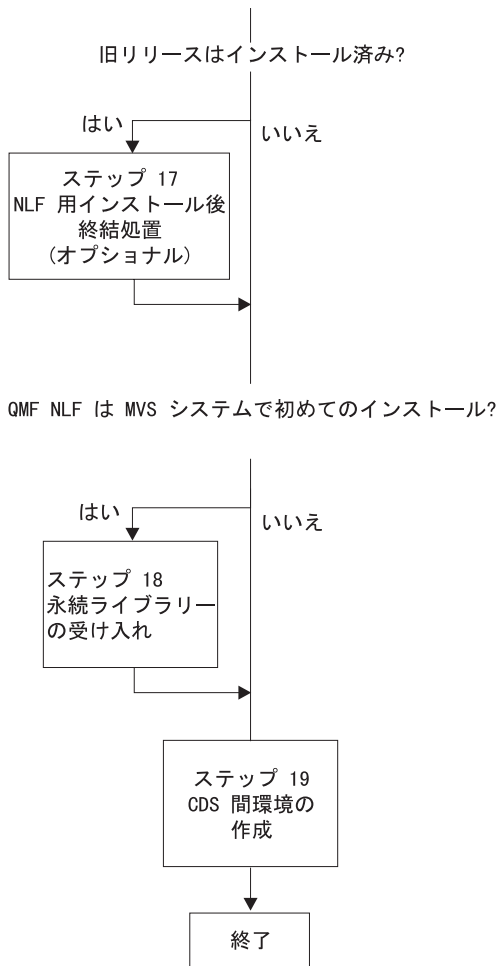


図 22. QMF NLF のインストール手順 -- その 6

予備 : Program Directory を読み NLF ワークシートに記入する

インストール・プロセスを開始する前に、NLF の Program Directory の補足データをお読みください。Program Directory は QMF NLF のリリースが変われば更新されることを考慮して、その中には PTF と APAR の説明と併せ、本書の発行日以後になされた本書の変更を含む、有用な情報が収められています。126ページの表21 は、QMF NLF のインストール時に用意しなくてはならない情報を示したものです。この表をワークシートとして使用することができます。

QMF NLF の計画およびインストール

表 21. QMF NLF インストール・パラメーター (バージョン 7 ワークシート - その 1)

パラメーター	値
ターゲット・ライブラリー接頭部 (デフォルト = QMF710)	
配布ライブラリー接頭部 (デフォルト = QMF710)	
ターゲット・ライブラリー・ボリューム (デフォルト = xxxxxx)	
配布ライブラリー・ボリューム (デフォルト = yyyyyy)	
SMP/E データ・セット接頭部 (デフォルト = IMSVS)	
ローカル DB2 サブシステム ID (デフォルト=DSN)	
ローカル DB2 リリース・レベル (デフォルト =V3R1)	
ローカル DB2 出口ライブラリー (デフォルト =DSN710.SDSNEXIT)	
ローカル DB2 ロード・ライブラリー (デフォルト =DSN710.SDSNLOAD)	
ローカル DB2 にインストールされる通信データベース	YES または NO
通信データベースがローカル DB2 にインストールされる場合は、次の情報を収集してください。	
インストールの効力範囲	F (全データベース)、S (サーバー・データベース)、または R (リクエスター・データベース)
データベース・インストールの効力範囲が“S”でない場合は、次の情報を収集してください。	
QMF ランタイム・ライブラリーをカスタマイズする	YES または NO
QMF アプリケーション・プラン ID (デフォルト =QMF710)	
データベース・インストールの効力範囲が“F”または“S”の場合は、次の情報を収集してください。	
QMF 表スペース・カタログ別名 (デフォルト =QMFDSN)	
QMF 表スペース・カタログ・パスワード (QMF 制御表用)	

表 21. QMF NLF インストール・パラメーター (バージョン 7 ワークシート - その 1) (続き)

パラメーター	値
QMF 表スペース・ボリューム	
DB2 デフォルト句読法	, (コンマ) または . (ピリオド)
前の QMF NLF レベル (移行インストールのみ)	2.2、2.3、2.4、3.1、3.1.1、3.2、3.3、または NONE

表 22. QMF NLF インストール・パラメーター (バージョン 7 ワークシート - その 2)

パラメーター	1 次	2 次
データベース・インストールの効力範囲が“R”でない場合は、次の情報を収集してください。		
QMF 制御表の表スペース サイズ: (1K 単位) 表スペース名 デフォルトのサイズ (1 次、2 次) - Q.COMMAND_SYNONYMS_n (100,20) (注を参照)	_____	_____
サイズ: (1K 単位) 表索引名 デフォルトのサイズ (1 次、2 次) - Q.COMMAND_SYNONYMSX_n (100,20) (注を参照)	_____	_____
以下のことを決定してください (該当する場合)。		
QMF NLF ジョブをフォアグラウンドでインストールするか、または JCL ファイルを自身で調整して各ジョブをバッチで実行するか	フォアグラウンドまたはバッチ	
CLIST ライブラリーを固定長にするかまたは可変長にするか	可変長または固定長	
EXEC ライブラリーを固定長にするかまたは可変長にするか	可変長または固定長	
SAVE DATA 表スペースを作成したいか否か	YES または NO	

以下のステップを実行する前に、まず QMF NLF の Program Directory の説明に従って SMP/E を使用して OS/390 環境に QMF NLF をインストールする必要があります。

ステップ 1 -- QMF NLF インストール・パラメーターの指定

このステップで、QMF および DB2 環境を記述する情報を入力します。必要なデータを『埋め込む』ことができる一連の QMF インストール・ダイアログ・パネルが表示されます。126ページの表21 のワークシートにユーザーが記入した情報を使用してください。

準備

このステップを開始する前に、次の要件を考慮に入れてください。

1. このステップを実行するためには、ISPF セッションがアクティブになっている必要があります。
2. 当初は DSQ1nJAL で指定した QMF ターゲット・ライブラリー名を使用していない (すなわち、ユーザーによる変更が QMFTPRE のような許可変数以外にない) 場合には、DSQ1nINS、DSQ1nIN1、および DSQ1nIN2 CLISTS を変更するか、または 138ページの『ステップ 4 ~ 8 -- ジョブの手動による実行依頼』に進む必要があります。
3. QMF を別のデータベースにインストールする場合には、インストール用にユーザーが使用する QMF ターゲット・ライブラリーを、インストール時に他のデータベースのユーザーがアクセスできないようにしてください。
4. 固定ブロック SDSQCLTn および SDSQEXCn は、ステップ 1 ~ 3 で必要になります。

実行

次のように入力してください。

```
TSO EXEC 'prefix.SDSQCLTn(DSQ1nINS)' 'QMFPRE(prefix)'
```

ここで、*prefix* は、ユーザーがワークシートに記入した QMF ターゲット・ライブラリー接頭部 です。

次のいずれかの方法で、パラメーターの入力パネルを出します。

- インストール・パラメーターを『初めて』入力する場合は、最初のパラメーター入力パネルが自動的に表示されます。
- このステップが少なくとも 1 回は正常に完了している場合は、次の 4 つのオプションを含んだ初期パネルが表示されます。

- P** インストール・パラメーター
- T** インストール・ファイルの調整
- I** フォアグラウンドでのインストール
- X** インストール・ダイアログの終了

最初のパラメーター入力パネルを出すには、**P** オプションを選択してください。

各パネルに情報を入力すると、入力データは、ユーザーが選択したデータベース名のもとで QMF710.SDSQCLTn ライブラリーに保管されます。

最後のパネルへの入力を完了する前にこのステップを終了すると、入力データは保管されません。最後のパネルでは、インストールの調整に使用されるジョブ・カード情報が求められます。フォアグラウンドでインストールを行いたい場合は、ジョブ・カード情報を入力する必要はなく、パネル上の指示されたスポットに x を入力するだけで済みます。

最後のインストール・パラメーターを指定した時点で、メインメニューが表示されます。各パラメーターの検討または変更を行いたい場合は、**P** を入力してから、再び各入力パネルに戻ります。

インストール・パラメーターの指定を適切に行えた場合は、次のステップに進みます。(必要があれば、この時点でインストール・プロセスを終え、あとで再開することができます。この場合、インストール・パラメーターは保管されません。) ワークシートの情報を使用して、各パネルへの入力を完了してください。

メインメニュー: メインメニューは、インストール・パラメーターをすべて保管した場合に表示されます。それ以外の場合は、インストールを起動するときに表示される最初のパネルになります。

このメニューにより、インストール・パラメーターの指定、ジョブの調整、フォアグラウンドでの QMF NLF のインストール、またはインストール・ダイアログの終了を行うことができます。

QMF NLF の計画およびインストール

```
INSTALL QMF NLF -- MAIN MENU

ISPF COMMAND ==>

CURRENTLY WORKING ON INSTALLATION INTO DB2 SUBSYSTEM DSN

YOU CAN NOW RE-SPECIFY THE INSTALL PARAMETERS, TAILOR THE INSTALLATION
FILES, INSTALL QMF WITH THE TAILORED FILES IN FOREGROUND, QUIT AND RUN
THE TAILORED INSTALL FILES IN BATCH, OR QUIT AND RETURN HERE LATER.

ENTER CHOICE HERE      ==>      ("P" - INPUT PARAMETERS,
                                "T" - TAILOR INSTALL FILES,
                                "I" - INSTALL IN FOREGROUND,
                                "X" - EXIT INSTALL DIALOGS)

PRESS:  ENTER TO CONTINUE   PF01 FOR HELP   PF03 TO END
```

図 23. ダイアログ・メインメニュー

注: 最後に使用した DB2 サブシステム名がこのパネルに表示されます。

このパネルで使用可能なオプションは、次のとおりです。

- P** QMF NLF インストール・パラメーターをカスタマイズします。これらのパラメーターについては、後続のパネルに記述されます。
別の DB2 サブシステム に合わせて QMF NLF インストール・パラメーターをカスタマイズしたい場合は、このパネルに表示される DB2 サブシステム名を無視し、P を入力して次のパネルに進みます。
- T** QMF 7 に必要なすべての NLF インストール・データ・セットを調整します。
これらのパネルは、本書には示されていません。
- I** QMF NLF をフォアグラウンドでインストールします。このオプションにより、ジョブ (ステップ 4 ~ 8) をオンライン環境で実行依頼することができます。これらのパネルは、本書には示されていません。
- X** インストール・ダイアログを終了します。

ローカル DB2 パラメーター: インストール・パラメーターをいずれも保管していない場合には、このパネルが最初に表示されます。それ以外の場合は、このパネルは、“P” オプションを選択した場合にのみ表示されます。

```

                                INSTALL QMF NLF -- LOCAL DB2 PARAMETERS
ISPF COMMAND ==>

LOCAL DB2 SUBSYSTEM ID   ==>
LOCAL DB2 RELEASE LEVEL ==>      ("31" FOR V3R1)
LOCAL DB2 EXIT LIBRARY   ==>
LOCAL DB2 LOAD LIBRARY   ==>

COMMUNICATIONS DATABASE(CDB) INSTALLED AT LOCAL DB2 ==>      ("Y","N")

PRESS:  ENTER TO CONTINUE   PF01 FOR HELP   PF03 TO END

```

図 24. ローカル DB2 パラメーター

このパネルで使用可能なオプションは、次のとおりです。

LOCAL DB2 SUBSYSTEM ID

QMF アプリケーション・プランが結合された DB2 サブシステム ID を指定します (必須。デフォルトは DSN です)。

LOCAL DB2 RELEASE LEVEL

ローカル DB2 サブシステムの DB2 リリース・レベルを指定します (必須。デフォルトなし)。

LOCAL DB2 EXIT LIBRARY

ローカル DB2 サブシステムの DB2 出口ライブラリーを指定します (必須。デフォルトなし)。

LOCAL DB2 LOAD LIBRARY

ローカル DB2 サブシステムの DB2 ロード・ライブラリーを指定します (必須。デフォルトなし)。

COMMUNICATIONS DATABASE(CDB) INSTALLED AT LOCAL DB2

DB2 通信データベースがローカル DB2 サブシステムにインストールされているか否かを指定します (必須。デフォルトなし)。

データベース・インストールの効力範囲: このパネルが表示されるのは、DB2 リリース・レベルが“23”であり、かつ通信データベースがローカル DB2 と共にインストールされている場合です。

QMF NLF の計画およびインストール

```
INSTALL QMF NLF -- SCOPE OF DATABASE INSTALL
ISPF COMMAND ==>

SCOPE OF DATABASE INSTALL      ==>      ("F" - FULL DATABASE,
                                           "R" - REQUESTOR DATABASE ONLY,
                                           "S" - SERVER DATABASE ONLY)

PRESS:  ENTER TO CONTINUE   PF01 FOR HELP   PF03 TO END
```

図 25. データベース・インストールの効力範囲

このパネルで使用可能なオプションは、次のとおりです。

SCOPE OF DATABASE INSTALL

データベース・インストールの効力範囲を指定します。使用可能なオプションは、FULL データベース、REQUESTER データベース、および SERVER データベースです。これらのオプションの詳細については、11ページの『QMF インストール・プロセスのロードマップ』を参照してください。

QMF 7 NLF を初めてインストールする場合には、FULL データベース・インストール・オプションを選択してください。

ローカル DB2 での QMF パラメーター: このパネルは、これが SERVER データベース・インストールではない場合に表示されます。

```

                INSTALL QMF NLF -- QMF PARAMETERS AT LOCAL DB2
ISPF COMMAND ==>

CUSTOMIZE QMF RUNTIME LIBRARIES      ==>          ("Y" OR "N")

- INSTALL QMF PANELS
- INSTALL QMF/GDDM MAP GROUPS
- INSTALL QMF/GDDM SAMPLE CHARTS FORMS
- MAKE QMF REXX EXECs AVAILABLE
- MAKE QMF CLISTS AVAILABLE

QMF APPLICATION PLAN ID AT LOCAL DB2  ==>

PRESS:  ENTER TO CONTINUE   PF01 FOR HELP   PF03 TO END
    
```

図 26. ローカル DB2 での QMF パラメーター

このパネルで使用可能なオプションは、次のとおりです。

CUSTOMIZE QMF RUNTIME LIBRARIES

QMF NLF ランタイム・ライブラリーにカスタマイズが必要か否かを指定します。オペレーティング・システムごとに 1 回これらのライブラリーをカスタマイズするだけで済みます (必須。デフォルトなし)。

QMF APPLICATION PLAN ID AT LOCAL DB2

ローカル DB2 サブシステムで結合された QMF アプリケーション・プラン名を指定します (必須。デフォルトなし)。

DB2 および QMF のパラメーター: このパネルが次に表示されます。

QMF NLF の計画およびインストール

```
INSTALL QMF NLF -- DB2 AND QMF PARAMETERS
ISPF COMMAND ==>

QMF TABLESPACES CATALOG ALIAS    ==>
QMF TABLESPACES CATALOG PASSWORD ==>
QMF TABLESPACES VOLUME           ==> ("SYSxxx" OR "AST",
                                       x is from 0 to 9,
                                       AST stands for *)
DEFAULT PUNCTUATION                ==> ("," OR ".")
PREVIOUS QMF LEVEL                 ==> ("V2R2", "V2R3", "V2R4",
                                       "V3R1", "V3R1M1",
                                       "V3R2", "V3R3", "V6R1"
                                       "NONE")

PRESS:  ENTER TO CONTINUE   PF01 FOR HELP   PF03 TO END
```

図27. DB2 および QMF のパラメーター

このパネルで使用可能なオプションは、次のとおりです。

QMF TABLESPACES CATALOG ALIAS

すべての QMF NLF 表スペースの VCAT 名を指定します。これらの QMF NLF 表スペースに関連づけられる VSAM データ・セットは、この別名値の高水準修飾子をもっています。前のレベルの QMF からの移行を行う場合には、前のリリースと同じ別名値を使用する必要があります (必須。デフォルトなし)。

QMF TABLESPACES CATALOG PASSWORD

インストール先システムによって作成されるすべての QMF NLF 制御表スペースおよび索引スペースのパスワードを指定します (オプション)。

QMF TABLESPACES VOLUME

QMF NLF 表スペースが常駐するボリューム通し番号を指定します (必須。デフォルトなし)。

DEFAULT PUNCTUATION

QMF NLF の DB2 での小数点の記号を指定します (必須。デフォルトなし)。

PREVIOUS QMF LEVEL

移行を行う QMF NLF の前のリリース・レベルを指定します (必須。データベース内に前のリリース・レベルがない場合は、NONE を入力します)。

QMF 索引スペースのスペース・パラメーター: このパネルは、データベース内に前の QMF NLF リリース・レベルがインストールされていない場合に表示されます。

```

INSTALL QMF NLF -- QMF INDEXSPACES SPACE PARAMETERS
ISPF COMMAND ==>

SPECIFY THE SIZES (IN 1K UNITS) FOR THE FOLLOWING TABLE INDEXES

TABLE INDEX          PRIMARY          SECONDARY
-----
Q.COMMAND_SYNONYMX_n  ==>              ==>

PRESS:  ENTER TO CONTINUE    PF01 FOR HELP    PF03 TO END
    
```

図 28. QMF 索引スペースのスペース・パラメーター

注: デフォルトのサイズ (1 KB 単位) が上にリストされます。

QMF 索引スペースの 1 次および 2 次割り振りを指定します。これらの値は、QMF がすべての VSAM ファイルをこれらの表スペースに割り振る場合に使用されます。インストール先システムのサイズに応じて、拡張用にフリー・スペースを使用できるようにするためにデフォルトのサイズを大きくしたり小さくしたりする必要が生じる場合があります。

ジョブ・カード: このパネルは常に、“P” オプションの最後のパネルとして表示されます。

```
INSTALL QMF NLF -- JOBCARD

ISPF COMMAND ==>

MODIFY THE JOB CARDS BELOW TO REPRESENT YOUR INSTALLATION REQUIREMENTS
THE "USER" AND "PASSWORD" PARAMETERS MUST BE SPECIFIED IN SYSTEMS USING
RACF. USE A USERID WITH THE APPROPRIATE AUTHORITY FOR THE DATABASE. SEE
THE "QUERY MANAGEMENT FACILITY: INSTALLATION GUIDE FOR MVS" PUBLICATION
FOR MORE DETAIL.

IF YOU WILL BE PERFORMING THE INSTALLATION IN FOREGROUND AND DO NOT
THE JCL FILES TO BE TAILORED, ENTER AN 'X' HERE. ==>

JOB STATEMENT INFORMATION:
==> //QMFINSTL JOB (ACCT),NAME,
==> //          CLASS=A,MSGCLASS=A,MSGLEVEL=(1,1),
==> //          USER=Q,PASSWORD=Q
==> // *

PRESS:  ENTER TO CONTINUE    PF01 FOR HELP    PF03 TO END
```

図 29. ジョブ・カード

このジョブ・カードは、インストール先システムでのすべての QMF インストール・ジョブの実行依頼に使用されます。

ステップ 2 -- ジョブの調整

インストール・パラメーターの指定を適切に行えた場合は、**P T I** パネルで **T** オプションを選択します。次のことが起こります。

- 128ページの『ステップ 1 -- QMF NLF インストール・パラメーターの指定』でユーザーが選択したインストール・パスに合わせてシステムが JCL を調整しているという旨のメッセージが出されます。
- QMF710.SDSQEXCn(DSQSCMDn) (QMF 呼び出し可能インターフェース REXX EXEC) が、サーバー・データベース・インストールを行っていない限り、パラメーター planid および DB2 subsystem name のデフォルトを更新するために変更されます。

このステップの終わりには、**P T I** パネルが表示されます。そこで、QMF NLF のインストールを続けることができます。

オンラインで調整を行いたい場合は、JCL および制御ファイルのインストール順序を変更してはなりません。これは、この CLIST では JCL および制御ファイルを特定の順序にしておく必要があるからです。JCL または制御ファイルの順序を変更する場合は、この CLIST を変更する必要があります。

QMF NLF のインストール

次のいずれかの方法で調整済みのジョブをインストールすることができます。

- フォアグラウンドでジョブをインストールする (『ステップ 3 -- フォアグラウンドでの QMF NLF のインストール』)。
- 各ジョブを手動で実行依頼する (138ページの『ステップ 4 -- QMF パネルのインストール』から始める)。このオプションにより、インストール環境の要件を満たすことができます。このオプションを使用すると、デフォルトにより用意される特定のインストール・パラメーター値を変更することができます。

ステップ 3 -- フォアグラウンドでの QMF NLF のインストール

フォアグラウンドで QMF NLF をインストールしたい場合には、**P T I** パネルで **I** オプションを選択します。ここで、インストール・オプションを求めるダイアログ・パネルが表示されます。情報の入力後に、インストールが進行中であることを示すメッセージが出されます。

DB2 権限

フォアグラウンドでジョブを実行依頼すると、ジョブは現行の LOGON ID のもとでインストールされます。LOGON ID にその許可 ID として Q が含まれている場合は、少なくとも、次の SQL ステートメントによって付与される DB2 権限が必要になります。

```
GRANT USE OF BUFFERPOOL BP0 TO Q
GRANT CREATESG TO Q
GRANT SELECT ON SYSIBM.SYSTABLES TO Q WITH GRANT OPTION
GRANT SELECT ON SYSIBM.SYSTABAUTH TO Q WITH GRANT OPTION
GRANT SELECT ON SYSIBM.SYSCOLUMNS TO Q WITH GRANT OPTION
```

LOGON ID に Q 以外の許可がある場合には、次の SQL ステートメントによって付与される権限が必要になります。

```
GRANT SYSADM TO authid
```

ここで、*authid* は 1 次許可 ID です。これで、ステップ 3 を終了します。次のいずれかのステップからインストールを続けてください。

- 150ページの『ステップ 8 -- NLF/QMF の TSO 用調整』
- 151ページの『ステップ 9 -- NLF/QMF の CICS 用調整』

QMF NLF を別の DB2 サブシステムにインストールする場合は、次のようにしてください。

- 全データベースまたはサーバー・データベースを使用する場合は、142ページの『ステップ 7A -- QMF 制御表の更新』に進む。

QMF NLF の計画およびインストール

- リクエスター・データベースを使用する場合は、150ページの『ステップ 8 -- NLF/QMF の TSO 用調整』または 151ページの『ステップ 9A -- DB2 RCT への NLF/QMF トランザクション ID の追加』に進む。

ステップ 4 ~ 8 -- ジョブの手動による実行依頼

以下のステップでは、バッチ環境で QMF をインストールする方法について説明します。

ステップ 4 -- QMF パネルのインストール

このステップでは、QMF パネルの拡張バージョンをパネル・ファイル QMF710.DSQPNLn にコピーします。

準備

このステップに関するジョブは QMF710.SDSQSAPn(DSQ1nPNL) です。137ページの『ステップ 3 -- フォアグラウンドでの QMF NLF のインストール』で行った調整が十分でなかった場合は、インストール先システムに適合するように JOB ステートメントを変更してください。また、ジョブのインストリーム・プロシージャ内の 2 つのパラメーターの値を変更する必要がある場合があります。

パラメーター名

値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部 (**QMF710**)

RGN ジョブ・ステップの領域サイズ (**2048K**)

実行

ライブラリー QMF710.SDSQSAPn のジョブ DSQ1nPNL を実行する。

ジョブの再実行

ジョブを再実行する前に、次のことを行います。

- ターゲット・ライブラリーに追加されたメンバーがある場合は、それを削除する。
- ライブラリーを圧縮して使用スペースを回復する。
- 障害を引き起こしたエラーを訂正する。

ステップ 5 -- NLF/GDDM マップ・グループのインストール

QMF は GDDM 画面マッピング機能を使用します。このステップでは、サンプル・ライブラリー (デフォルトの名前は QMF710.SDSQSAPn です) 内にある

NLF/GDDM マップ・グループ・ファイル、ターゲット・マップ・グループ・ライブラリー (デフォルトの名前は QMF710.DSQMAPn です) に合わせて拡張します。

準備

このステップに関するジョブは QMF710.SDSQSAPn(DSQ1nMAP) です。137ページの『ステップ 3 -- フォアグラウンドでの QMF NLF のインストール』で行った調整が十分でなかった場合は、インストール要件に適合するように JOB ステートメントを変更してください。また、ジョブのインストリーム・プロシージャ内の 2 つのパラメーターの値を変更する必要がある場合があります。

パラメーター名

値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部 (**QMF710**)

RGN ジョブ・ステップの領域サイズ (**2048K**)

実行

ライブラリー QMF710.SDSQSAPn 内のジョブ DSQ1nMAP を実行します。

ジョブの再実行

ジョブを再実行する前に、次のことを行います。

- ターゲット・ライブラリーに追加されたメンバーがある場合は、それを削除する。
- ライブラリーを圧縮して使用スペースを回復する。
- 障害を引き起こしたエラーを訂正する。

ステップ 6 -- REXX EXEC または CLIST レコードの変換

このステップでは、QMF REXX EXEC または QMF CLIST レコードを固定長から可変長に変換します。

ステップ 6A -- QMF REXX EXEC レコードの変換: 固定長から可変長に

注: OS/390 環境で REXX EXEC を使用するためには、TSO/E バージョン 2 (またはそれ以降のリリース) がインストールされている必要があります。

この手順のあとで、QMF EXEC ライブラリー (QMF710.SDSQEXCn) を SYSEXEC データ・セットとして割り振る必要があります。そのためには、このライブラリーを他の EXEC ライブラリーに連結する必要があります。

例

次の JCL では、ライブラリー QMF710.SDSQEXCn は、SYS2.EXEC という名前の EXEC ライブラリーに連結されます。

```
//SYSEXEC DD DSN=SYS2.EXEC,DISP=SHR
//          DD DSN=QMF710.SDSQEXCn,DISP=SHR
```

QMF EXEC ライブラリーには固定長レコードが入っています。このライブラリーは、固定長レコードが入っている他の EXEC ライブラリーだけに連結することができます。ライブラリーに可変長レコードが入っている場合は、可変長レコードを用いて QMF ライブラリーのコピーを作成する必要があります。次のことを検査してください。

- 他の EXEC ライブラリーに固定長レコードが入っている場合は、このステップをとばして先をお読みください。
- 他の EXEC ライブラリーに可変長レコードが入っている場合は、このステップを続けてお読みください。(あとで、QMF EXEC ライブラリーの作成済みコピー (QMF710.NEW.SDSQEXCn) を SYSEXEC 連結で使用します。)

詳細については、*QMF* アプリケーション開発の手引き を参照してください。

準備: このステップに関するジョブは QMF710.SDSQSAPn(DSQ1nJVE) です。DSQTEVB.SYSUT2 ステートメントの 'xxxxxx' をライブラリーのコピーのボリューム通し番号に変更してください。136ページの『ステップ 2 -- ジョブの調整』で行った調整が十分でなかった場合は、次のことを行うことができます。

- インストール先システムに適合するようにジョブ・ステートメントを変更する。
- ジョブのインストリーム・プロシージャの QMFTPRE パラメーターの値を変更する (必要な場合)。この値は QMF ライブラリーの接頭部です (デフォルトは QMF710 です)。
- EXEC プロシージャ・パラメーターをブランクにしておく。このパラメーターは、EXEC プロシージャが呼び出されるときにジョブによって使用されます。
- このライブラリーの名前を変更する (必要な場合)。

実行: ジョブ DSQ1nJVE を実行します。

ジョブの再実行: ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

ステップ 6B -- QMF CLIST レコードの変換: 固定長から可変長に

この手順のあとで、QMF CLIST ライブラリー (QMF710.SDSQCLTn) を SYSPROC データ・セットとして割り振る必要があります。そのためには、このライブラリーを他の CLIST ライブラリーに連結する必要があります。

例

次の JCL では、ライブラリー QMF710.SDSQCLTn は、SYS2.CLIST という名前の CLIST ライブラリーに連結されます。

```
//SYSPROC DD DSN=SYS2.CLIST,DISP=SHR
//          DD DSN=QMF710.SDSQCLTn,DISP=SHR
```

QMF CLIST ライブラリーには固定長レコードが入っています。このライブラリーは、固定長レコードが入っている他の CLIST ライブラリーだけに連結することができます。ライブラリーに可変長レコードが入っている場合は、可変長レコードを用いて QMF ライブラリーのコピーを作成する必要があります。次のことを検査してください。

- 他の CLIST ライブラリーに固定長レコードが入っている場合は、このステップをとばして先をお読みください。
- 他の CLIST ライブラリーに可変長レコードが入っている場合は、このステップを続けてお読みください。(あとで、QMF CLIST ライブラリーの作成済みコピー (QMF710.NEW.SDSQCLTn) を SYSPROC 連結で使用します。)

準備: このステップに関するジョブは QMF710.SDSQSAPn(DSQ1nJVC) です。DSQTIVB.SYSUT2 ステートメントの 'xxxxxx' をライブラリーのコピーのボリューム通し番号に変更してください。136ページの『ステップ 2 -- ジョブの調整』で行った調整が十分でなかった場合は、次のことを行うことができます。

- インストール先システムに適合するようにジョブ・ステートメントを変更してください。
- ジョブのインストリーム・プロシージャの QMFTPRE パラメーターの値を変更する (必要な場合)。この値は QMF ライブラリーの接頭部です (デフォルトは QMF710 です)。

QMF NLF の計画およびインストール

- 'CLIST' プロシージャ・パラメーターをブランクにしておく。このパラメーターは、このプロシージャが呼び出されるときにジョブによって使用されます。
- このライブラリーの名前を変更する (必要な場合)。

実行: ジョブ DSQ1nJVC を実行します。

ジョブの再実行: ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

リクエスター・データベース・インストールを行う場合は、150ページの『ステップ 8 -- NLF/QMF の TSO 用調整』または 151ページの『ステップ 9A -- DB2 RCT への NLF/QMF トランザクション ID の追加』に進んでください。

ステップ 7A -- QMF 制御表の更新

ユーザーが実行するサブステップ (7Aa, 7Ab, 7Ac, 7Ad) は、実行中の移行インストールのタイプに応じて決まります。

- 前の QMF NLF リリースがインストールされていない場合は、『サブステップ 7Aa -- 前の QMF NLF リリースがインストールされていない場合』を実行する。
- QMF NLF 2.2 または 2.3 の移行を行っている場合は、144ページの『サブステップ 7Ab -- QMF NLF 2.2 または 2.3 がインストールされている場合』を実行する。
- QMF NLF 2.4 の移行を行っている場合は、145ページの『サブステップ 7Ac -- QMF NLF 2.4 がインストールされている場合』を実行する。
- QMF NLF 3.1 の移行を行っている場合は、146ページの『サブステップ 7Ad -- QMF NLF 3.1 がインストールされている場合』を実行する。
- QMF NLF 3.1.1, QMF NLF 3.2, または QMF NLF 3.3 がこの DB2 サブシステム内にインストールされている場合は、147ページの『ステップ 7B -- 以前の QMF NLF サンプル表の削除』に進む。

これらのステップでは、TSO バッチを実行し、システム・メッセージにあるステップの完了コードを検査します。完了メッセージは、指示されるように、SYSTSPRT または SYSTEMM の出力の中にあります。SYSPRINT には、IBM サポート用に診断情報が追加されています。

サブステップ 7Aa -- 前の QMF NLF リリースがインストールされていない場合

前の QMF NLF がインストールされていない場合は、このステップを実行してください。

このステップでは、次のことを行います。

- NLF 項目を Q.PROFILES 表に追加する。このジョブは QMF710.SDSQSAPn(DSQ1nUPO)
- Q.COMMAND_SYNONYM_n という名前のコマンド同義語表を NLF 環境用に作成する。このジョブは QMF710.SDSQSAPn(DSQ1nCCS)

準備: DSQ1nUPO および DSQ1nCCS の JOB ステートメントを、インストール先システムに適合するように変更する。JOB ステートメントの USER パラメーターの値は現在、QMF 表の所有者に合わせた Q です。ユーザーの許可 ID が Q でない場合は、この値をユーザーの 1 次許可 ID に変更してください。

また、ジョブのインストリーム・プロシージャ内の次のパラメーター値にも必要な変更を加えてください。

パラメーター名
値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部 (**QMF710**)

DB2EXIT

DB2 出口ライブラリーの名前 (**DSN710.SDSNEXIT**)

DB2LOAD

DB2 プログラム・ライブラリーの名前 (**DSN710.SDSNLOAD**)

RGN ジョブ・ステップの領域サイズ (**2048K**)

DB2 権限: ユーザーが Q である場合には、次の照会を実行して、ジョブを実行するのに十分な権限を得ます。

```
GRANT CREATETAB ON DATABASE DSQDBCTL TO Q
```

データベース DSQDBCTL がユーザー Q によって作成されなかった場合は、ユーザーが照会を必要とすることがあります。

ユーザーが Q でない場合には、次の照会を実行して、ジョブを実行するのに十分な権限を得ます。

```
GRANT INSERT, UPDATE ON TABLE Q.PROFILES TO authid
GRANT CREATETAB ON DATABASE DSQDBCTL TO authid
```

ここで、*authid* は 1 次許可 ID です。

実行: 次に示す適切なジョブを実行する。

- DSQ1nUPO (行を Q.PROFILES に追加する場合)

QMF NLF の計画およびインストール

- DSQInCCS (必須 SQL ステートメントを実行する場合)

完了メッセージについて SYSTERM を検討します。エラーが発生した場合は、エラー・メッセージについて SYSTSPRT または SYSPRINT を検査します。

ジョブの再実行: ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

サブステップ 7Ab -- QMF NLF 2.2 または 2.3 がインストールされている場合

QMF NLF 2.2 または 2.3 からの移行を行う場合にのみ、このステップを実行してください。

このジョブは、IRM および LAYOUT コマンド同義語の NLF 同義語を Q.COMMAND_SYNONYM_n 表に追加し、Q.PROFILES 制御表を更新します。

準備: このステップで使用されるジョブは QMF710.SDSQSAPn(DSQInICS) です。インストール先システムに適合するようにジョブ・ステートメントを変更してください。また、ジョブのインストリーム・プロシージャー内の次のパラメーターにも必要な変更を加えてください。

パラメーター名
値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部名 (QMF710)

DB2EXIT

DB2 出口ライブラリーの名前 (DSN710.SDSNEXIT)

DB2LOAD

DB2 プログラム・ライブラリーの名前 (DSN710.SDSNLOAD)

RGN ジョブ・ステップの領域サイズ (2048K)

DB2 権限: ユーザーが Q である場合には、ジョブを実行するのに必要な権限を持ちます。

ユーザーが Q でない場合には、次の照会を実行して必要な権限を得ます。

```
GRANT INSERT ON TABLE Q.COMMAND_SYNONYM_n TO authid
```

ここで、*authid* は 1 次許可 ID です。

実行: ジョブ QMF710.SDSQSAPn(DSQInICS) を実行します。

完了メッセージについて **SYSTEM** を検討します。エラーが発生した場合は、エラー・メッセージについて **SYSTSPRT** または **SYSPRINT** を検査します。

ジョブの再実行: ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

サブステップ 7Ac -- QMF NLF 2.4 がインストールされている場合

QMF NLF V2R4 からの移行を行う場合にのみ、このステップを実行してください。

このジョブでは、**Q.PROFILES** 制御表を更新します。

準備: このステップで使用されるジョブは **QMF710.SDSQSAPn(DSQ1nUP1)** です。インストール先システムに適合するようにジョブ・ステートメントを変更してください。必要な場合、ジョブのインストリーム・プロシーチャーのインストール・パラメーター値を変更します。

パラメーター名

値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部名 (**QMF710**)

DB2EXIT

DB2 出口ライブラリーの名前 (**DSN710.SDSNEXIT**)

DB2LOAD

DB2 プログラム・ライブラリーの名前 (**DSN710.SDSNLOAD**)

RGN ジョブ・ステップの領域サイズ (**2048 KB**)

DB2 権限: ユーザーが **Q** である場合には、ジョブを実行するのに必要な権限を持ちます。

ユーザーが **Q** でない場合には、次の照会を実行して必要な権限を得ます。

```
GRANT INSERT ON TABLE Q.PROFILES TO authid
```

ここで、*authid* は 1 次許可 ID です。

実行: ジョブ **QMF710.SDSQSAPn(DSQ1nUP1)** を実行します。

完了メッセージについて **SYSTEM** を検討します。エラーが発生した場合は、エラー・メッセージについて **SYSTSPRT** または **SYSPRINT** を検査します。

QMF NLF の計画およびインストール

ジョブの再実行: ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

サブステップ 7Ad -- QMF NLF 3.1 がインストールされている場合

QMF NLF V3R1 からの移行を行う場合にのみ、このステップを実行してください。

このジョブでは、Q.PROFILES 制御表を更新します。

準備: このステップで使用されるジョブは QMF710.SDSQSAPn(DSQ1nUP2) です。インストール要件に適合するようにジョブ・ステートメントを変更します。必要な場合、ジョブのインストリーム・プロシーチャーのインストール・パラメーター値を変更します。

パラメーター名
値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部名 (QMF710)

DB2EXIT

DB2 出口ライブラリーの名前 (DSN710.SDSNEXIT)

DB2LOAD

DB2 プログラム・ライブラリーの名前 (DSN710.SDSNLOAD)

RGN ジョブ・ステップの領域サイズ (2048K)

DB2 権限: ユーザーが Q である場合には、ジョブを実行するのに必要な権限を持ちます。

ユーザーが Q でない場合は、次の照会を実行して必要な権限を得ます。

```
GRANT INSERT ON TABLE Q.PROFILES TO authid
```

ここで、*authid* は 1 次許可 ID です。

実行: ジョブ QMF710.SDSQSAPn(DSQ1nUP2) を実行します。

完了メッセージについて SYSTERM を検討します。エラーが発生した場合は、エラー・メッセージについて SYSTSPRT または SYSPRINT を検査します。

ジョブの再実行: ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

ステップ 7B と 7C -- QMF NLF サンプル表の設定

次のいずれかが適用される場合は、ステップ 7B とステップ 7C の両方をとばして先をお読みください。

- NLF が大文字機能 (UCF) である。
- QMF NLF の前のバージョン 2 (全リリース) からサンプル表をすでにインストールしている。

この 2 つのステップでは、QMF NLF サンプル表を設定します。最初のステップでは以前に作成された表を除去し、2 番目のステップでは新規の表をインストールします。障害が起こった場合は、この 2 つのステップを再始動することができます。これは、そのステップによって実行されたジョブが終了するまで、データベースの変更がコミットされないためです。

ステップ 7B -- 以前の QMF NLF サンプル表の削除

QMF NLF の以前のリリースもインストールされている DB2 サブシステムに QMF 7 NLF をインストールする場合に、このステップを実行してください。それ以外の場合は、148ページの『ステップ 7C -- NLF サンプル表の作成』に進みます。

このステップでは、以前のバージョンのインストール時に作成されたサンプル表を削除します。QMF NLF サンプル表は QMF 7 NLF 用に変更されています。

準備

このステップで使用されるジョブは QMF710.SDSQSApN(DSQ1nDSJ) です。137ページの『ステップ 3 -- フォアグラウンドでの QMF NLF のインストール』で行った調整が十分でなかった場合は、インストール要件に適合するようにジョブ・ステートメントを変更します。必要な場合、ジョブのインストール・プロシージャのインストール・パラメーター値を変更します。

パラメーター名

値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部名 (QMF710)

DB2EXIT

DB2 出口ライブラリーの名前 (DSN710.SDSNEXIT)

DB2LOAD

DB2 プログラム・ライブラリーの名前 (DSN710.SDSNLOAD)

RGN ジョブ・ステップの領域サイズ (2048K)

QMF NLF の計画およびインストール

このジョブにその他の変更を加えてはなりません。

DB2 許可

ユーザーが Q でない場合には、次の照会を実行して必要な権限を得ます。

```
GRANT SYSADM TO authid
```

ここで、*authid* は 1 次許可 ID です。

実行

ライブラリー QMF710.SDSQSAPn 内のジョブ DSQ1nDSJ を実行します。完了メッセージについて SYSTERM を検討します。エラーが発生した場合は、エラー・メッセージについて SYSTSPRT または SYSPRINT を検査します。

ジョブの再実行

ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。ただし、ジョブが除去しようとする表がすでに除去されていた場合には、そのジョブが失敗することもあります。

ステップ 7C -- NLF サンプル表の作成

このステップでは、NLF サンプル表を作成します。

注: ネットワーク内の各ロケーションの QMF NLF ユーザーは、QMF NLF のインストール先のロケーションで作成されたすべてのサンプル表の使用を許可されています。

準備

このステップに関するジョブは QMF710.SDSQSAPn(DSQ1nIVS) です。ステップ 3 で行った調整が十分でなかった場合は、インストール要件に適合するようにジョブ・ステートメントを変更してください。必要な場合、ジョブのインストール・プロシージャのインストール・パラメーター値を変更します。

パラメーター名

値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部 (**QMF710**)

DB2EXIT

DB2 出口ライブラリーの名前 (**DSN710.SDSNEXIT**)

DB2LOAD

DB2 プログラム・ライブラリーの名前 (**DSN710.SDSNLOAD**)

RGN ジョブ・ステップの領域サイズ (**2048K**)

CDS, CDP

小数部で使用される小数点の句読記号を識別します。これは、DB2 のインストール時に指定された DECPOINT オプションと一致する必要があります。

- ピリオドの場合は、現行値をそのままにしておいてください。
- コンマの場合は、CDS を **6**、CDP を **7** に変更してください。

DECPOINT オプションの詳細については、*DB2 UDB (OS/390 版) インストールの手引き* を参照してください。

DB2 権限

ユーザーが Q である場合は、少なくとも、次の SQL ステートメントによって付与される DB2 権限が必要になります。

```
GRANT SELECT ON SYSIBM.SYSTABLES TO Q WITH GRANT OPTION
GRANT SELECT ON SYSIBM.SYSTABAUTH TO Q WITH GRANT OPTION
GRANT SELECT ON SYSIBM.SYSCOLUMNS TO Q WITH GRANT OPTION
```

ユーザーが Q でない場合には、次の照会を実行して必要な権限を得ます。

```
GRANT SYSADM TO authid
```

ここで、*authid* は 1 次許可 ID です。

実行

ライブラリー QMF710.SDSQSAPn 内のジョブ DSQ1nIVS を実行します。完了メッセージについて SYSTERM を検討します。エラーが発生した場合は、エラー・メッセージについて SYSTSPRT または SYSPRINT を検査します。

ジョブの再実行

ジョブが失敗した場合には、エラーを訂正してジョブを再実行することができます。

QMF NLF を別のデータベースにインストールする場合は、159ページの『ステップ 12 -- バッチ IVP 実行のための NLF バッチ・ジョブのセットアップ (オプション)』に進んでください。

これで、TSO または CICS に合うように NLF/QMF を調整する準備が整いました。

- QMF NLF の TSO 用調整については、次の節を参照してください。
- QMF NLF の CICS 用調整については、151ページの『ステップ 9 -- NLF/QMF の CICS 用調整』を参照してください。

ステップ 8 -- NLF/QMF の TSO 用調整

NLF 用の TSO ログオン・プロシージャを作成するためには、まず QMF 基本製品用の TSO ログオン・プロシージャのコピーを作成してください。

TSO ログオン・プロシージャに対する以下の変更を除いて、NLF/QMF を TSO 用に調整するための手順については、71ページの『第5章 QMF の TSO 用調整』で概要が示してあります。

- 次の NLF ライブラリーは、QMF 基本ライブラリーの前に連結する必要があります。
 - ADMGGMAP DD ステートメントに連結するためのステートメントは、次のとおりです。

```
//ADMGGMAP DD DSN=QMF710.DSQMAPn,DISP=SHR
```
 - ISPPPLIB DD ステートメントに連結するためのステートメントは、次のとおりです。

```
//ISPPPLIB DD DSN=QMF710.SDSQPLBn,DISP=SHR
```
 - ISPMLIB DD ステートメントに連結するためのステートメントは、次のとおりです。

```
//ISPMLIB DD DSN=QMF710.SDSQMLBn,DISP=SHR
```
 - SYSPROC DD ステートメントに連結するためのステートメントは、次のとおりです。

```
//SYSPROC DD DSN=QMF710.SDSQCLTn,DISP=SHR
```
 - SYSEXEC DD ステートメントに連結するためのステートメントは、次のとおりです。

```
//SYSEXEC DD DSN=QMF710.SDSQEXCn,DISP=SHR
```
 - DSQPNLn DD ステートメントに連結するためのステートメントは、次のとおりです。

```
//DSQPNLn DD DSN=QMF710.DSQPNLn,DISP=SHR
```
- ISPF を用いて QMF を開始するためのステートメントは、次のようになります。

```
ISPSTART PGM(DSQMFn) NEWAPPL(DSQn) PARM(DSQSSUBS=dbname,...)
```

ISPF マスター・アプリケーション・メニュー (Master Application Menu) は、次の図に示すように変更する必要があります (DSQQMFn は NLF プログラムです)。


```

%----- MASTER APPLICATION MENU -----
%SELECT APPLICATION ==>_OPT +
%
%                                +USERID -
%                                +TIME -
% 1 +SPF - SPF PROGRAM DEVELOPMENT FACILITY +TERMINAL -
% 2 +QMF - QMF QUERY MANAGEMENT FACILITY +PF KEYS -
% 3 +QMFn - QMF NATIONAL LANGUAGE FEATURE
%
%
%
%
%
%
% P +PARMS - SPECIFY TERMINAL PARAMETERS AND LIST/LOG DEFAULTS
% X +EXIT - TERMINATE USING LIST/LOG DEFAULTS
%
%+PRESS%END KEY+TO TERMINATE +
%
%)INIT
)PROC
  &SEL = TRANS( TRUNC (&OPT, '.')
                1, 'PANEL(ISR@PRIM) NEWAPPL'
                2, 'PGM(DSQMF) NEWAPPL(DSQE)'
                3, 'PGM(DSQMFn) NEWAPPL(DSQn)'
                /* */
                /* ADD OTHER APPLICATIONS HERE */
                /* */
                P, 'PANEL(ISPOPT)'
                X, 'EXIT'
                ' '
                *, '?' )
)END

```

図 30. NLF 用の ISPF マスター・アプリケーション・メニューでの QMF ダイアログ

- ISPF を用いずに QMF を開始するためのステートメントは、次のようになります。

```
DSQQMFn DSQSPLAN=planid,DSQSSUBS=dbname,...
```

ここで、DSQQMFn は NLF プログラムです。

ステップ 9 -- NLF/QMF の CICS 用調整

QMF 製品を 81 ページの『第 6 章 QMF の CICS 用調整』の説明に従って CICS 用に調整した後で、このステップを実行することができます。QMF 3.1 からの移行を行う場合は、154 ページの『ステップ 9Da -- CICS 管理テーブルの更新 (CICS V2 のみ)』を除くすべてのステップを実行する必要があります。

ステップ 9A -- DB2 RCT への NLF/QMF トランザクション ID の追加

トランザクションのデータベースのプラン ID および許可 ID が DB2 リソース管理テーブル (RCT) に指定されています。たとえば、“QMFn” というトランザクション ID および “DEPT1” という許可 ID を指定するためには、次のステートメントを追加します。

QMF NLF の計画およびインストール

DSNCRCT TYPE=ENTRY, TXID=QMFn, PLAN=QMF710, AUTH=DEPT1

QMF には、QMF710.SDSQSAPn(DSQ1nRCT) に置かれているサンプル RCT 項目が用意されています。

QMF トランザクションを DB2 に対して記述する情報を使用して RCT を更新した後で、RCT を再生成する必要があります。

ステップ 9B -- DFHEAI および DFHEAI0 を使用したリンク・エディット

QMF は、CICS コマンド・レベル・アプリケーション・プログラミング・インターフェースを使用して CICS のもとで稼働します。したがって、QMF プログラムを実行するためには、まず EXEC インターフェース・モジュール DFHEAI および DFHEAI0 を使用して QMF をリンク・エディットする必要があります。CICS インターフェース・モジュール DFHEAI および DFHEAI0 を組み込むためには、QMF サービスを適用するたびにこのステップを実行する必要があります。

サブステップ 9Ba -- CICS コマンド・インターフェース・モジュールを使用した QMF のリンク・エディット

このジョブでは、CICS コマンド・レベル・サポート・モジュール DFHEAI および DFHEAI0 を使用して QMF NLF モジュールをリンク・エディットします。

準備: このステップで使用されるジョブは QMF710.SDSQSAPn(DSQ1nLNK) です。インストール先システムに適合するようにジョブ・ステートメントを変更してください。必要な場合、ジョブのインストリーム・プロシージャのインストール・パラメーター値を変更します。

パラメーター名

値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部名 (**QMF710**)

REG ジョブ・ステップの領域サイズ (**4096**)

OUTC ジョブの出力クラス (*)

CLOAD

CICS ロード・ライブラリーの名前 (**CICS.LOADLIB**)

このジョブの完了後に、リストを調べ、すべてのモジュールが正常にリンク・エディットされたかを確認する。

注: いずれかのモジュールが PTF によって変更されている場合は、このジョブを再実行する必要があります。

サブステップ 9Bb -- QMF 提供の管理プログラムの変換、アセンブル、およびリンク・エディット

準備: このステップで使用されるジョブは QMF710.SDSQSAPn(DSQ1nGLK) です。インストール先システムに適合するようにジョブ・ステートメントを変更してください。必要な場合、ジョブのインストリーム・プロシーチャーのインストール・パラメーター値を変更します。

パラメーター名
値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部名 (**QMF710**)

REG ジョブ・ステップの領域サイズ (**4096**)

OUTC ジョブの出力クラス (*)

CLOAD

CICS ロード・ライブラリーの名前 (**CICS.LOADLIB**)

CMACS

CICS マクロ・ライブラリーの名前 (**CICS.MACLIB**)

SUFFIX

CICS ASM 変換プログラムの接尾部 (**1\$**)

ASMBLR

アセンブラーの名前 (**IEV90**)

WORK

作業ボリューム単位の名前 (**SYSDA**)

ステップ 9C -- ADMF データ・セットへの QMF/GDDM マップ・セットのロード

準備: このステップで使用されるジョブは QMF710.SDSQSAPn(DSQ1nADM) です。インストール先システムに適合するようにジョブ・ステートメントを変更してください。必要な場合、ジョブのインストリーム・プロシーチャーのインストール・パラメーター値を変更します。

パラメーター名
値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部名 (**QMF710**)

QMF NLF の計画およびインストール

REG ジョブ・ステップの領域サイズ (2048)

GDDM

GDDM ADMF データ・セットの名前 (GDDM.ADMF)

ステップ 9Da -- CICS 管理テーブルの更新 (CICS V2 のみ)

CICS のもとで NLF/QMF 機能を実行するためには、まず QMF 項目を次のように定義する必要があります。

FCT (ファイル管理テーブル): NLF/QMF ヘルプおよび画面定義が入っている QMF パネル・ファイルを記述します。メンバー DSQ1nFCT (ライブラリー QMF710.SDSQSAPn 内の) を CICS システム上の既存の FCT 項目に追加またはコピーしてください。

PCT (プログラム管理テーブル): この NLF/QMF の QMF トランザクション名を記述します。メンバー DSQ1nPCT (ライブラリー QMF710.SDSQSAPn 内の) を CICS システム上の既存の PCT 項目に追加またはコピーしてください。

PPT (処理プログラム・テーブル): NLF/QMF の定数およびメッセージが入っている QMF プログラムを記述します。メンバー DSQ1nPPT (ライブラリー QMF710.SDSQSAPn 内の) を CICS システム上の既存の PPT 項目に追加またはコピーしてください。

メンバーを CICS システムに組み込むかまたはコピーした後で、アSEMBルおよびリンク・エディットを行ってください。

ステップ 9Db -- CICS 管理テーブルの更新 (CICS ESA のみ)

CICS のもとで NLF/QMF 機能を実行するためには、まず QMF 項目を CICS システム定義ファイル (CSD) に定義する必要があります。

準備: このステップで使用されるジョブは QMF710.SDSQSAPn(DSQ1nCSD) です。インストール先システムに適合するようにジョブ・ステートメントを変更してください。必要な場合、ジョブのインストリーム・プロシーチャーのインストール・パラメーター値を変更します。

パラメーター名

値の記述 (括弧内はデフォルト)

QMFTPRE

QMF ターゲット・ライブラリーの接頭部名 (QMF710)

REG ジョブ・ステップの領域サイズ (2048)

OUTC ジョブの出力クラス (*)

CLOAD

CICS ロード・ライブラリーの名前 (**CICS.LOADLIB**)

CCSD CICS CSD データ・セットの名前 (**CICS.DFHCCSD**)

ステップ 9E -- CICS 領域のジョブ・ストリームの更新

QMF パネル・ファイルは、QMF を含んでいる CICS 領域を開始するために使用される既存の JCL に追加する必要があります。次のステートメントを追加してください。

```
//DSQPNLn DD DSN=QMF710.DSQPNLn,DISP=SHR
```

ここで、*n* は NLF 文字です。

ステップ 9F -- IVP の実行

103ページの『ステップ 33 (CICS の場合) -- IVP の実行』の指示に従って IVP を実行し、次の名前を変更してください。

- QMF320.DSQSAMPE を QMF710.SDSQSAP*n* に
- DSQ1EIVC を DSQ1*n*IVC に

ここで、*n* は NLF 文字です。

ステップ 10 -- QMF NLF の Workstation Database Server 用調整 (オプション)

Workstation Database Server のための QMF サポートはオプションです。Workstation Database Server を QMF NLF のアプリケーション・サーバーとして実行したい場合にのみ、このステップに示されている各ステップを実行する必要があります。

QMF NLF を OS/390 から Workstation Database Server にインストールする前に、QMF 基本製品を Workstation Database Server のデータベースにインストールするのに必要な各ステップに従ったことを検査する必要があります。QMF NLF のインストールには、アウトバウンド Workstation Database Server ID に SYSADM 権限が含まれていることが必要です。Workstation Database Server への QMF のインストールの詳細については、91ページの『第7章 QMF の ワークステーション・データ・サーバー用調整』を参照してください。

これらのステップでは、TSO バッチを実行し、システム・メッセージにあるステップの完了コードを検査します。完了メッセージは、指示されるように、SYSTSPRT または SYSTEMR の出力の中にあります。SYSPRINT には、IBM サポート用に診断情報が追加されています。

ステップ 10A -- Workstation Database Server での QMF NLF 制御表の作成

このステップでは、Workstation Database Server に QMF NLF コマンド同義語表およびプロファイル行を作成します。

1. QMF710.SDSQSAPE(DSQ1nDJ2) を編集する。
2. ジョブのインストリーム・プロシーチャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQ1TBJ4 PROC RGN='2048K',           Job-step region size
//           QMFTPRE='QMF710',         Prefix for QMF target libraries
//           DB2EXIT='DSN710.SDSNEXIT', Exit DB2 library name
//           DB2LOAD='DSN710.SDSNLOAD'  DB2 program library name
```

3. SYSTEM(DSN) の DSN を DB2 UDB (OS/390 版) サブシステム ID に変更する。
4. ジョブ QMF710.SDSQSAPE(DSQ1nDJ2) を実行依頼する。
5. リターン・コード 0 または 4 について検査する。完了メッセージについては、SYSTEM を検討する。

リターン・コードが 0 または 4 以外の場合は、先に進んではなりません。エラー・メッセージがないか SYSTSPRT または SYSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

ステップ 10B -- Workstation Database Server での QMF NLF サンプル表の作成

このステップでは、Workstation Database Server に QMF NLF サンプル表を作成します。

1. QMF710.SDSQSAPE(DSQ1nDJ4) を編集する。
2. ジョブのインストリーム・プロシーチャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQ1TBJ4 PROC RGN='2048K',           Job-step region size
//           QMFTPRE='QMF710',         Prefix for QMF target libraries
//           DB2EXIT='DSN710.SDSNEXIT', Exit DB2 library name
//           DB2LOAD='DSN710.SDSNLOAD'  DB2 program library name
```

3. SYSTEM(DSN) の DSN を DB2 UDB (OS/390 版) サブシステム ID に変更する。
4. ジョブ QMF710.SDSQSAPE(DSQ1nDJ4) を実行依頼する。
5. リターン・コード 0 または 4 について検査する。完了メッセージについては、SYSTEM を検討する。

リターン・コードが 0 または 4 以外の場合は、先に進んではなりません。エラー・メッセージがないか SYSTSPRT または SYSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

Workstation Database Server からの QMF NLF の削除

この節では、QMF NLF を Workstation Database Server から削除する方法について説明します。

Workstation Database Server からの QMF の削除: このステップは、QMF をすでに含んでいる Workstation Database Server に QMF を再インストールする場合にのみ実行する必要があります。

考慮事項: このステップでは、Workstation Database Server から QMF NLF コマンド同義語表およびシステム・プロファイル行を削除します。

1. QMF710.SDSQSAPE(DSQ1nDX1) を編集する。
2. ジョブのインストリーム・プロシージャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQ1TBJ4 PROC RGN='2048K',           Job-step region size
//                QMFTPRES='QMF710',     Prefix for QMF target libraries
//                DB2EXIT='DSN710.SDSNEXIT', Exit DB2 library name
//                DB2LOAD='DSN710.SDSNLOAD' DB2 program library name
```

3. SYSTEM(DSN) の DSN を DB2 UDB (OS/390 版) サブシステム ID に変更する。
4. ジョブ QMF710.SDSQSAPE(DSQ1nDX1) を実行依頼する。
5. リターン・コード 0 または 4 について検査する。完了メッセージについては、SYSTERM を検討する。

リターン・コードが 0 または 4 以外の場合は、先に進んではなりません。エラー・メッセージがないか SYSTSPRT または SYSSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

Workstation Database Server からの QMF NLF サンプル表の削除: このステップは、QMF NLF をすでに含んでいる Workstation Database Server に QMF NLF を再インストールする場合にのみ実行する必要があります。

このステップでは、Workstation Database Server からすべての QMF NLF サンプル表および表スペースを除去および作成します。

1. QMF710.SDSQSAPE(DSQ1nDX2) を編集する。
2. ジョブのインストリーム・プロシージャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQ1TBJ4 PROC RGN='2048K',           Job-step region size
//                QMFTPRES='QMF710',     Prefix for QMF target libraries
//                DB2EXIT='DSN710.SDSNEXIT', Exit DB2 library name
//                DB2LOAD='DSN710.SDSNLOAD' DB2 program library name
```

QMF NLF の計画およびインストール

3. SYSTEM(DSN) の DSN を DB2 UDB (OS/390 版) サブシステム ID に変更する。
4. ジョブ QMF710.SDSQSAPE(DSQ1nDX2) を実行依頼する。
5. リターン・コード 0 または 4 について検査する。完了メッセージについては、SYSTEM を検討する。

エラー・メッセージがないか SYSTSPRT または SYSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

ステップ 11 -- QMF NLF の DB2 (AS/400 版) サーバー用調整 (オプション)

QMF の DB2 (AS/400 版) Database Servers 用のサポートはオプションです。DB2 (AS/400 版) Database Server を、QMF NLF 用のアプリケーション・サーバーとして実行したい場合のみ、ステップ 11 で説明されている手順を行う必要があります。OS/390 から、QMF NLF を DB2 (AS/400 版) Database Server にインストールする前に、QMF 基本製品を DB2 (AS/400 版) Database Server データベース・サーバーにインストールするのに必要なステップを行ってあることを確認する必要があります。

TSO バッチを実行するこれらのすべてのステップでは、システム・メッセージにあるステップの完了コードを検査します。完了メッセージは、指示されるように、SYSTSPRT または SYSTEM の出力の中にあります。SYSPRINT には、IBM サポート用に診断情報が追加されています。

DB2 (AS/400 版) サーバーでの QMF NLF 制御表の更新の作成

1. QMF710.SDSQSAPE(DSQ1nAS2) を編集する。
2. ジョブのインストリーム・プロシージャのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。

```
//DSQ1nAS2 PROC RGN='2048K', Job-step region size
// QMFTPRE='QMF710', Prefix for QMF target libraries
// DB2EXIT='DSN710.SDSNEXIT', Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD' DB2 program library name
```

3. SYSTEM() で DB2 (OS/390 版) サブシステム ID に変更する。
4. ジョブのコメントを慎重に読んで、変更が必要であれば変更を行う。
5. ジョブ QMF710.SDSQSAPE(DSQ1nAS2) を実行依頼する。
6. リターン・コード 0 または 4 について検査する。完了メッセージについては、SYSTEM を検討する。リターン・コードが 0 または 4 以外の場合、先に進んではなりません。エラー・メッセージがないか SYSTSPRT または SYSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

DB2 (AS/400 版) サーバーでの QMF NLF サンプル表の作成

1. Edit QMF710.SDSQSAPE(DSQ1nAS4).
2. ジョブのインストリーム・プロシーチャーのインストール・パラメーターのデフォルト値を検査および変更する (必要な場合)。


```
//DSQ1nAS4 PROC RGN='2048K', Job-step region size
// QMFTPRE='QMF710', Prefix for QMF target libraries
// DB2EXIT='DSN710.SDSNEXIT', Exit DB2 library name
// DB2LOAD='DSN710.SDSNLOAD' DB2 program library name
```
3. SYSTEM() で DB2 (OS/390 版) サブシステム ID に変更する。
4. ジョブのコメントを慎重に読んで、変更が必要であれば変更を行う。
5. ジョブ QMF710.SDSQSAPE(DSQ1nAS4) を実行依頼する。
6. リターン・コード 0 または 4 について検査する。完了メッセージについては、SYSTEM を検討する。リターン・コードが 0 または 4 以外の場合には、先に進んではなりません。エラー・メッセージがないか SYSTSPRT または SYSPRINT を検査します。訂正アクションを行い、その後でこのジョブを再実行します。

ステップ 12 -- バッチ IVP 実行のための NLF バッチ・ジョブのセットアップ (オプション)

NLF の場合、TSO ログオン・プロシーチャーを 79ページの『ステップ 18 -- バッチ IVP を実行するための QMF バッチ・ジョブの設定 (オプション)』で説明されているように変更する必要があります。次のように、ISPSTART コマンドのプロシーチャーの末尾を変更します。

```
ISPSTART PGM(DSQMFn) NEWAPPL(DSQn) PARM(DSQSMODE=B,DSQSRUN=Q.DSQ1nBAT)
```

ステップ 13 -- QMF 対話モードでの IVP の実行

IVP の実行については、101ページの『ステップ 33 (TSO の場合) -- IVP の実行』および 103ページの『ステップ 33 (CICS の場合) -- IVP の実行』を参照してください。NLF IVP (DSQ1nIVP) (ライブラリー QMF710.SDSQSAPn 内にある) が NLF の検査に使用されます。このプロシーチャー (DSQ1nIVP) は、QMF 英語サンプル・ライブラリー (*prefix*.SDSQSAPE) から照会をインポートします。ここで、*prefix* は QMF データ・セットの接頭部です。

これらのプロシーチャーは、この接頭部が QMF710 であることを想定して作成されます。これがユーザーの接頭部でない場合は、それが DSQ1nIVP プロシーチャー内のどこに現れても、ユーザーの接頭部に一致するように QMF710 を変更してください。

```
IMPORT PROC FROM 'QMF710.SDSQSAPn(DSQ1nIVP)'
RUN PROC
```

ステップ 14 -- 各国語サンプル照会およびプロシーチャーのインストール

QMF NLF のインストールおよび検査後に、QMF NLF を使用してサンプル照会およびプロシーチャーの変換済みバージョンをインストールします。このインストールは、次の 2 つのステップで行います。

- 『ステップ 14A -- 既存のサンプル照会およびプロシーチャーの削除』
- 161ページの『ステップ 14B -- 各国語サンプル照会およびプロシーチャーのインストール』

ステップ 14A -- 既存のサンプル照会およびプロシーチャーの削除

ユーザーのロケーションに同じ言語 ID をもつ QMF NLF の前のリリースがインストールされていない場合は、このステップをとばして先に進んでください。

既存のサンプル照会およびプロシーチャーを削除するためには、変換済みの QMF コマンド (該当する場合) を使用して QMF プロシーチャー DSQ1nSQD (QMF 7 サンプル・ライブラリー QMF710.SDSQSAPn の) をインポートおよび実行してください。このプロシーチャー (DSQ1nSQD) は、QMF 英語サンプル・ライブラリー (*prefix* .SDSQSAPE) から照会をインポートします。ここで、*prefix* は QMF データ・セットの接頭部です。

これらのプロシーチャーは、この接頭部が QMF710 であることを想定して作成されます。これがユーザーの接頭部でない場合は、それが DSQ1nSQD プロシーチャー内のどこに現れても、ユーザーの接頭部に一致するように QMF710 を変更してください。

```
IMPORT PROC FROM 'QMF710.SDSQSAPn(DSQ1nSQD)'  
RUN PROC
```

このステップを実行すると、データベース状況表示パネルが表示されることがあります。そこで、何らかのアクションをとる必要はありません。

DB2 許可: ユーザーが Q である場合には、すでに必要な権限を付与されています。

ユーザーが Q でない場合には、次の照会を実行して必要な権限を得ます。

```
GRANT UPDATE ON Q.OBJECT_DIRECTORY TO authid  
GRANT UPDATE ON Q.OBJECT_REMARKS TO authid  
GRANT UPDATE ON Q.OBJECT_DATA TO authid
```

ここで、*authid* は 1 次許可 ID です。

このステップの再始動: ジョブが失敗した場合でも、次のステップに進むことができます。

ステップ 14B -- 各国語サンプル照会およびプロシーチャーのインストール

各国語サンプル照会およびプロシーチャーをインストールするためには、変換済みの QMF コマンド (該当する場合) を使用して QMF710.SDSQSAPn (DSQ1nSQI) の QMF プロシーチャーをインポートおよび実行してください。このプロシーチャー (DSQ1nSQI) は、QMF 英語サンプル・ライブラリー (*prefix* .SDSQSAPE) から照会をインポートします。ここで、*prefix* は QMF データ・セットの接頭部です。

これらのプロシーチャーは、この接頭部が QMF710 であることを想定して作成されます。これがユーザーの接頭部でない場合は、それが DSQ1nSQI プロシーチャー内のどこに現れても、ユーザーの接頭部に一致するように QMF710 を変更してください。

```
IMPORT PROC FROM 'QMF710.SDSQSAPn(DSQ1nSQI)'  
RUN PROC
```

ユーザーが Q でない場合は、ステップ 34 -- QMF アプリケーション照会およびアプリケーション・オブジェクトのインストール (TSO) を参照して、必要な GRANT 照会を実行する必要があります。

このステップでは、バッチ・モード IVP およびサンプル・アプリケーション・プロシーチャーもインストールします。

DB2 許可: ユーザーが “Q” である場合には、すでに必要な権限を付与されています。

ユーザーが Q でない場合には、次の照会を実行して必要な権限を得ます。

```
GRANT UPDATE ON Q.OBJECT_DIRECTORY TO authid  
GRANT UPDATE ON Q.OBJECT_REMARKS TO authid  
GRANT UPDATE ON Q.OBJECT_DATA TO authid
```

ここで、*authid* は 1 次許可 ID です。

このステップの再始動: このジョブの実行中に障害が生じた場合には、エラーを訂正してからプロシーチャー DSQ1nSQD を実行してください。このプロシーチャーにより、以前に作成されたサンプル照会を削除します。続いて、プロシーチャー DSQ1nSQI を再実行します。

ステップ 15 -- バッチ・モード IVP の実行 (オプション)

バッチ IVP の実行については、108ページの『ステップ 35 -- バッチ・モード IVP の実行 (オプション)』を参照してください。DSQQMFE の代わりに各国語プログラム DSQQMF_n を使用してバッチ IVP を開始します。このステップでは、QMF 7 バッチ IVP を使用します。

ステップ 16 -- ポストインストールの終結処理

インストール後の終結処理アクティビティーについては、109ページの『ステップ 36 -- インストール後の終結処理』を参照してください。

QMF NLF の前のリリースがインストールされていない場合は、このステップをとばして先に進んでください。

QMF NLF の前のリリースの各ライブラリーを削除することができます。デフォルトの接頭部をもつこれらのライブラリーは、次の図にリストされています。

考慮事項: QMF 7 データ・セットを削除することのないよう接頭部には特に注意を払ってください。

V2R2 Data Sets	V2R3 Data Sets	V2R4 Data Sets	V3R1 Data Sets
QMF220.DSQMACn	QMF230.DSQMACn	QMF240.DSQMACn	QMF310.DSQMACn
QMF220.DSQPMSn	QMF230.DSQPMSn	QMF240.DSQPMSn	QMF310.DSQPMSn
QMF220.DSQSAMPn	QMF230.DSQSAMPn	QMF240.DSQSAMPn	QMF310.DSQSAMPn
QMF220.DSQMAPn	QMF230.DSQMAPn	QMF240.DSQMAPn	QMF310.DSQMAPn
QMF220.DSQCLSTn	QMF230.DSQCLSTn	QMF240.DSQCLSTn	QMF310.DSQCLSTn
QMF220.DSQPLIBn	QMF230.DSQPLIBn	QMF240.DSQEXECn	QMF310.DSQEXECn
QMF220.DSQSLIBn	QMF230.DSQSLIBn	QMF240.DSQUSERn	QMF310.DSQUSERn
QMF220.DSQMLIBn	QMF230.DSQMLIBn	QMF240.DSQPLIBn	QMF310.DSQMLIBn
QMF220.DSQTLIBn	QMF230.DSQTLIBn	QMF240.DSQSLIBn	QMF310.DSQSLIBn
		QMF240.DSQMLIBn	QMF310.DSQMLIBn
		QMF240.DSQTLIBn	QMF310.DSQTLIBn

V3R1M1 Data Sets	V3R2 Data Sets	V3R3 Data Sets	V6R1 Data Sets
QMF311.DSQMACn	QMF320.DSQMACn	QMF330.DSQMACn	QMF610.ADSQMACn
QMF311.DSQPMSn	QMF320.DSQPMSn	QMF330.DSQPMSn	QMF610.ADSQPMSn
QMF311.DSQSAMPn	QMF320.DSQSAMPn	QMF330.DSQSAMPn	QMF610.SDSQSAPn
QMF311.DSQMAPn	QMF320.DSQMAPn	QMF330.DSQMAPn	QMF610.SDSQPLBn
QMF311.DSQCLSTn	QMF320.DSQCLSTn	QMF330.DSQCLSTn	QMF610.SDSQCLTn
QMF311.DSQEXECn	QMF320.DSQEXECn	QMF330.DSQEXECn	QMF610.SDSQMLBn
QMF311.DSQUSERn	QMF320.DSQUSERn	QMF330.DSQUSERn	QMF610.SDSQEXCn
QMF311.DSQPLIBn	QMF320.DSQPLIBn	QMF330.DSQPLIBn	QMF610.SDSQUSRn
QMF311.DSQSLIBn	QMF320.DSQSLIBn	QMF330.DSQSLIBn	QMF610.DSQMAPn
QMF311.DSQMLIBn	QMF320.DSQMLIBn	QMF330.DSQMLIBn	
QMF311.DSQTLIBn	QMF320.DSQTLIBn	QMF330.DSQTLIBn	

図 31. QMF NLF の前のリリースから削除されるライブラリー

ステップ 17 -- 永続ライブラリーの受け入れ

OS/390 システムで言語 *n* に関する QMF NLF のインストールを初めて行う場合は、このステップを実行してください。

このステップのジョブ名は DSQ1nJAC です。これにより、インストール先システムで使用されるプロシージャー DSQ1nJSM または SMP/E プロシージャーを呼び込みます。SMP/E ACCEPT ステップの実行については、112ページの『ステップ 37 -- 永続ライブラリーの受け入れ』を参照してください。

ステップ 18 -- CDS 間環境の作成

基本 QMF 7 および NLF に共通するモジュールに対して保守変更が行われなかった場合は、このステップをとばして先に進んでください。このステップにより、SMP/E は変更済みのモジュールを記憶することができます。

このステップには、SMP/E 環境で JCLIN データを更新するための SMP/E ジョブが含まれています。このジョブはメンバー DSQ1nCDS (ライブラリー

QMF NLF の計画およびインストール

QMF710.SDSQSAPn 内の)に入っています。このジョブに対する入力データは、メンバー DSQ1nJCL (ライブラリー QMF710.SDSQSAPn 内の)に入れられます。

第11章 リモート・サーバーでの QMF 7.1 パッケージのバインド

QMF バージョン 7 リリース 1 リクエスター・インストールがサーバーと通信できるためには、該当のサーバーに QMF 7.1 パッケージが存在している必要があります。サーバーで QMF 7.1 の新規インストールまたは移行インストールがすべて完了していれば、通信を開始することができ、それ以外のことを行なう必要はありません。ただし、QMF 3.3 またはそれ以降が含まれているサーバーで、移行が現行のオプションではない場合には、ユーザーは QMF710.SDSQSAPE データ・セットのにある DSQ1BPKG ジョブを実行することができます。このジョブは、QMF 7.1 パッケージを指定したサーバーにバインドします。(ただし、サーバーに QMF 3.3 またはそれ以降が検出される場合のみ。) DSQ1BPKG を読み取り、調整し、実行依頼してバインドを実行してください。エラー・メッセージがないかについてジョブ出力を検査した後、必要であればジョブを再実行します。

一例 ローカル DB2 (OS/390 版) サブシステムの DB2G を QMF 3.3 から QMF 7.1 に移行します。サブシステム DB2G の QMF ユーザーは、QMF 3.3 が含まれている DB2 (VM 版) サーバーの SQLV61A と定期的に通信しています。DB2 (VM 版) の DBA は、VM サーバーで QMF を 7.1 に移行することはできません。DB2G 内の QMF 7.1 インストールが SQLV61A 上の QMF と通信するためには、ジョブ DSQ1BPKG を実行して DB2 (VM 版) サーバーでパッケージをバインドする必要があります。

第2部 QMF (OS/390 版) の管理

第12章 必要な記憶域	177	CICS のもとでの QMF の開始の例	196
QMF 記憶域要件	177	QMF を SRPI からサーバーとして実行する ように設定	196
OS/390 の記憶域	177		
CICS/MVS 領域	178		
CICS/ESA 領域	178		
第13章 QMF の開始	179	第14章 開始手順のカスタマイズ	199
QMF を開始する前に	179	プログラム・パラメーターの要約	199
環境の確立	179	クイック・スタート	200
CICS と DB2 の接続	179	報告書記憶域のカスタマイズおよび報告書パ フォーマンス	203
TSO と DB2 の接続	179	DSQSBSTG (報告書データ用の記憶域の調 整).	203
クイック・スタート	180	各セッション用の仮想記憶域の正しい 量の選択	204
QMF を ISPF のもとで実行するように設定 メニュー・オプションからの QMF の開 始	181	各 CICS ユーザー用の仮想記憶域の正 しい量の選択	205
LIBDEF ステートメントの使用	183	CICS パフォーマンス・トレードオフ	205
ISPLLIB データ・セットの割り振り	185	DSQSRSTG (アプリケーションに使用され る予約記憶域の調整).	205
ISPSTART コマンドによる QMF の開始	185	DSQSRSTG の値が 0 の場合	206
ISPF でバッチ・モードによる QMF の開 始	185	DSQSBSTG の値が小さく、DSQSRSTG の値が大きい場合	207
ISPF のもとでの QMF の開始の例	186	DSQPILL (エクストラ記憶域の獲得)	207
QMF を TSO のもとで実行するように設定	187	非 CICS ユーザーに関する予備ファイ ルの割り振り	208
TSO ID の定義	187	予備ファイルに必要なスペースの見積 もり	209
ID のタイプの選択	188	非対話式 QMF セッションでの予備フ ァイルの使用	211
TSO のもとでの QMF	188	予備ファイルの問題の解決	212
DSQQMFE モジュールにより QMF を直接開 始	189	DSQSSPQN (CICS 予備記憶域の名前の指 定).	214
TSO CALL コマンドを使用した QMF の開 始	189	DSQSIROW (表示用に検索される報告書の 行数の制御).	214
TSO のバッチ環境での QMF の開始	190	DSQSIROW の値が小さい場合のパフ ォーマンス	215
TSO のもとでの QMF の開始の例	190	DSQSIROW 値が大きい場合のパフ ォーマンス	216
TSO EXEC の作成	191	セッション開始時点での QMF アクティビ ティーのトレース	217
プログラム・ロード・ライブラリーの検査	191	DSQSDBUG (トレース詳細レベルの設定)	217
TSO に関する考慮事項	191		
QMF データ・セットの検査	192		
GDDM データ・セットの検査	192		
QMF をバッチ・ジョブとしてネイティブ OS/390 で実行するように設定	193		
QMF を CICS のもとで実行するように設定	194		
CICS/DB2 接続機能の使用	195		
CICS のもとでの QMF	196		

DSQSDBQT (CICS のトレース・データ用 記憶域のタイプの指定)	218	クイック・スタート	241
DSQSDBQN (CICS のトレース・データ用 記憶域の名前の指定).	219	QMF へのユーザー・アクセスを可能にする ユーザー・プロファイルの作成	242
セッション中の初期アクティビティの制御	220	インストール先システムのためのプロファ イル構造の確立	243
DSQSDBNM (QMF 開始時の接続先のロケ ーションの指定)	220	新規のユーザー・プロファイルの Q.PROFILES 表への追加	243
DSQSMODE (対話式または非対話式 QMF セッションの指定)	221	固有のプロファイルを持たないユーザーに よる QMF の使用の防止	244
DSQSSUBS (QMF で使用される DB2 サ ブシステムの命名)	221	Q.PROFILES 制御表の読み取り	245
DSQSRUN (QMF の開始時に実行するプ ロシージャーの命名).	222	正しいプロファイルをユーザーの操作環境 に提供	251
初期プロシージャーの非対話式実行	223	ユーザー・プロファイルの更新	252
初期プロシージャーを使用した対話式 QMF 作業の実行	224	SET PROFILE コマンドの使用	252
初期プロシージャーに変数値を渡す	225	SQL UPDATE ステートメントの使用	253
DSQSPLAN (QMF アプリケーション・プ ランの命名).	229	SYSTEM プロファイルの更新.	254
DSQSPRID (TSO プロファイル・キーの指 定).	230	Q.PROFILES 制御表からのプロファイルの 削除	254
DSQSDBCS (2 バイト文字セット・データの 印刷の設定).	230	QMF アプリケーション・プランおよびパッ ッケージに対するアクセスの制御	255
REXX プログラム DSQSCMD を使用したデ フォルトの開始値の設定	231	アプリケーション・プランおよびパッケー ジに対するアクセスの付与.	255
第15章 QMF セッション制御機能	235	QMF アプリケーション・プランおよびパ ッケージに対するユーザー・アクセスの取 り消し	256
Q.SYSTEM_INI のインストールまたは除去	235	QMF オブジェクトおよびデータベース・オ ブジェクトに対するアクセスの制御.	256
デフォルトのシステム初期化プロシージャー のインポート	235	オブジェクトをアクセスするのに必要な DB2 特権	257
Q.SYSTEM_INI プロシージャーを実行する時 期	236	DB2 特権の付与と取り消し	258
Q.SYSTEM_INI の使用法	236	PUBLIC に対する付与	258
QMF と共に出荷される例	237	ユーザーに対する特権の付与	258
ユーザー・セッション・プロシージャーの 例	237	特定の特権の付与.	259
オブジェクトのリストを表示するプロシー ジャー	238	表特権の付与	259
セキュリティおよびセッション・プロシー ジャーの共用	239	視点特権の付与	259
診断の考慮事項	239	データベースを保守するための権限	262
第16章 エンド・ユーザーのための QMF サ ポートの確立	241	適切な特権の付与 : SAVE および IMPORT コマンド	262
QMF コード・ページに関する考慮事項	241	他のユーザーの付与の取り消し	264
Q AUTHID の役割	241	PUBLIC に対する付与の取り消し	264
		DB2 権限を付与できるユーザーが多過 ぎると起こり得る事態	265
		オブジェクトにアクセスするのに必要な SQL 特権	265
		QMF コマンドのために必要な SQL 特 権	266

指示照会および QBE 照会に必要な	ユーザー管理のデータ・セットの
SQL 特権 267	VSAM クラスタ 295
表編集機能に必要な SQL 特権 . . . 267	制御表の保守 295
SQL 特権の付与と取り消し 268	制御表のモニターと再編成 296
SQL GRANT ステートメントの使用 . . 269	索引使用の判別 297
SQL REVOKE ステートメントの使用 . . 270	バッファ・プールの切り替え 297
他のユーザーとの QMF オブジェクトの	DB2 カタログ表を使用する表および視点の保
共用 270	守 298
非コミット読み取りの許可 271	表と視点のリスト 298
オブジェクトの作成に関する標準の設定	表または視点の所有権の転送 298
271	表または視点のデータベースからの削除 . . . 299
ユーザーのデータベース・オブジェクト・リ	ローカルで定義された日付 / 時刻フォーマッ
ストのカスタマイズ 272	トのサポート 299
デフォルト・オブジェクト・リストの使用	フォーマットの指定 299
272	編集ルーチンを使用可能に 299
デフォルト・リストの変更 275	DXT エンド・ユーザー・ダイアログへのア
オブジェクト・リストの記憶域要件 . . . 277	クセス (ISPF のみ) 300
ユーザーがデータベース内で表を作成できる	EXTRACT コマンドのサポート 301
ようにする 277	リソースの割り振り 301
ユーザー用の表スペースの選択および割り	DXT データ・セットの割り振り 301
当て 279	CLIST の使用によるリソースの割り振り
表スペースのタイプの選択 282	および割り振り解除 302
ユーザーへの DB2 CREATETAB 権限の	割り振り CLIST の準備 302
付与 282	割り振り解除 CLIST の準備 306
ユーザーが図表をサポートできるようにする	ユーザー用文書編集インターフェースのカス
282	タマイズ 311
TSO および ISPF での図表サポート . . . 283	アプリケーションの変更 312
CICS での図表のサポート 284	文書インターフェース・マクロの名前変更 . . 312
制御表を使用した QMF オブジェクトの保守	Q.DSQAEDIS プロシージャをデータベ
284	ースに入れる 313
Q.OBJECT_DIRECTORY 表の読み取り . . . 285	Q への所有権の転送 313
Q.OBJECT_DATA 表の読み取り 286	データ・コンポーネントの変更 314
Q.OBJECT_REMARKS 表の読み取り . . . 288	メッセージ・コンポーネント 314
QMF 照会、書式、およびプロシージャ	DCF コンポーネント 315
のリスト 288	CLIST およびマクロの変更 316
QMF 照会、書式、およびプロシージャ	DSQAnD1P の変更 316
の表示 289	DSQABD1Q の変更 317
照会、書式、およびプロシージャの所有	LIBDEF をサポートするための
権の転送 289	DSQABDIP の変更 317
古くなった 照会、書式、およびプロシ	DSQABD1C の変更 318
ージャの削除 290	QMF EDIT コマンドのカスタマイズ 319
OS/390 データ・セット内の照会、書式、	NLF 環境での英語サポートのイネープリング
およびプロシージャのインポート . . . 291	グローバル変数を使用した通貨記号の定義 . . 322
QMF オブジェクト制御表用の表スペース	
の拡大 291	
DB2 サブシステムの保守 294	第17章 ユーザーによるオブジェクトの印刷 . . . 323
データ・セットの管理 294	クイック・スタート 323
DB2 管理のデータ・セット用の記憶域	
グループ 295	

オブジェクトの印刷	324	TSO およびネイティブ OS/390 バッチの印刷	
印刷に QMF または GDDM のいずれのサー		機能キーの同義語の定義	339
ビスを使用するか の決定	326	CICS の印刷機能キーの同義語の定義	340
GDDM サービスを使用して印刷を処理する	327	ユーザー・プロファイルを更新して GDDM	
プリンターのための GDDM ニックネーム		印刷を可能にする	341
の選択	327		
正しいタイプの GDDM 装置の選択	328	第18章 QMF コマンドのカスタマイズ	343
ニックネーム指定の作成	328	クイック・スタート	343
ファミリー 1 または 2 GDDM プリン		QMF が提供するデフォルトの同義語の使用	344
ターのニックネーム例	330	TSO における印刷報告書の表示 (DPRE)	345
ファミリー 3 GDDM プリンターのニ		DPRE の使用	345
ックネーム例	331	DPRE のカスタマイズ	346
ファミリー 4 GDDM プリンターのニ		コマンド同義語表の作成	347
ックネーム例 (TSO およびネイティブ		コマンド同義語定義を表へ入力	349
OS/390 バッチのみ)	331	動詞の選択	350
1 つの定義で複数のニックネームを定		VERB 列に関する規則	350
義	332	コマンド同義語動詞として基本 QMF	
ニックネーム定義の例	332	動詞を使用	351
GDDM デフォルト・モジュールをニック		目的語名の選択	351
ネームで更新	333	同義語定義の選択	352
外部デフォルトのファイル内でニックネ		線形プロシージャーを同義語定義で使	
ーム定義をテスト (TSO およびネイティブ		用	353
OS/390 バッチのみ)	334	変数を同義語定義で使用	354
TSO およびネイティブ OS/390 バッチに		情報を SYNONYM_DEFINITION 列に	
ニックネーム・ファイルを割り振る	334	入力	355
CICS でニックネームを使用	334	同義語のアクティブ化	356
ファミリー 2 ニックネームの物理装置		コマンド同義語表の保守の最小化	357
への結び付け	335	1 つの同義語表を全ユーザーに割り当てる	357
ファミリー 3 ニックネームの物理装置		同義語表の視点を個々のユーザーに割り当	
への結び付け	335	てる	358
QMF が GDDM ニックネームとインター		共用または私用で使用する同義語	358
フェースする方法	336	共用使用またはグループ使用のための	
TSO およびネイティブ OS/390 バッチの印刷		同義語	358
で QMF サービスを使用	337	許可表と対になった同義語	359
CICS で印刷を行うために QMF サービスを			
使用	337	第19章 QMF 機能キーのカスタマイズ	361
一時記憶域キューと一時データ・キューの		クイック・スタート	361
選択	337	カスタマイズしたいキーの選択	361
PRINT コマンドを使用して出力をキュー		フルスクリーン・パネルのデフォルト・キ	
に経路指定	338	ー	362
グローバル変数を使用して印刷のためのク		ウィンドウ・パネルのデフォルト・キー	363
ューを定義	339	機能キー表の作成	364
CICS 一時記憶域キューからの印刷	339	ユーザーの機能キー定義を表へ入力	366
CICS 一時記憶域キューからの報告書を見		コマンドと機能キーの結び付け	366
る	339	機能キーにラベルを付けて画面上に配置	368
		キー定義の例	369

フルスクリーン・パネルでキーの定義 を入力	369
ウィンドウ・パネルのキーの定義の 入力	370
ヘルプ・パネルまたはプロンプト・パ ネルへのキー定義の入力	370
カスタマイズしたいパネルの識別	371
フルスクリーン・パネル ID	371
ウィンドウ・パネル ID	372
コマンド・ウィンドウ	372
書式ウィンドウ	372
グローバル変数ウィンドウ	372
ヘルプおよびプロンプト・ウィンドウ	372
ロケーション・ウィンドウ	373
オブジェクト・リスト・ウィンドウ	373
指示照会ウィンドウ	373
新規機能キー定義のアクティブ化	374
機能キー表の検査と問題診断	375

第20章 QMF 書式のためのユーザー独自の

編集コードの作成	377
クイック・スタート	377
編集コードを選択する	378
DATE、TIME、および TIMESTAMP 情報の 処理	380
データをフォーマット設定するためのユーザ ー出口ルーチンの呼び出し	381
出口ルーチンとの情報の受け渡し	383
インターフェース制御ブロックのフィール ド	384
入力域の特徴を示すフィールド	386
入力域での U タイプ編集コードの表現 方法	386
入力域での V タイプ編集コードの表現 方法	387
出力域の特徴を示すフィールド	387
QMF 終了時に制御を出口ルーチンに渡す	387
編集ルーチンを HLASM またはアセンブラ ーで作成	388
TSO、SRPI、APPC およびネイティブ OS/390 用の編集ルーチンをアセンブラ ーで作成	388
アセンブラー編集ルーチンと TSO、SRPI、APPC およびネイティブ OS/390 との対話方法	388
プログラムのアセンブル	389

プログラムのリンク・エディット	389
アセンブルおよびリンク・エディット のステートメントの例 (TSO、SRPI、APPC、またはネイティ ブ OS/390)	390
プログラム例	390
編集ルーチンを CICS 用にアセンブラ ーで作成	390
アセンブラー編集ルーチンと CICS と の対話方法	391
プログラムの変換	392
プログラムのアセンブル	392
プログラムのリンク・エディット	393
変換、アセンブル、およびリンク・エ ディットのステートメント例 (CICS)	393
プログラム例	393
アセンブラー編集ルーチンと QMF との 対話方法	394
編集ルーチンを PL/I で作成	397
言語環境プログラム (LE) を使用せずに TSO、SRPI、APPC またはネイティブ OS/390 用に PL/I で編集ルーチンを作成	397
PL/I 編集ルーチンの、 TSO、SRPI、APPC、またはネイティブ OS/390 との対話方法	398
DSQUXDT および DSQUPLI のコンパ イル	399
プログラムのリンク・エディット	399
コンパイルおよびリンク・エディット のステートメント例 (TSO、SRPI、APPC、またはネイティ ブ OS/390)	399
プログラム例	400
言語環境プログラム (LE) を使用して TSO、SRPI、APPC、またはネイティブ OS/390 での編集ルーチンを PL/I で作成	400
PL/I 編集ルーチンが LE を使用した TSO、SRPI、APPC、またはネイティブ OS/390 と対話する方法	401
DSQUXDT のコンパイル	402
プログラムのリンク・エディット	402
コンパイルおよびリンク・エディット のステートメント例 (TSO、SRPI、APPC、またはネイティ ブ OS/390)	402
プログラム例	403

編集ルーチンを CICS 用に PL/I で作成	403	COBOL プログラムの変換	421
PL/I 編集ルーチンと CICS との対話方法	404	コンパイル	421
プログラムの変換	405	リンク・エディット	422
プログラムのコンパイル	406	変換、コンパイル、およびリンク・エディットのステートメント例 (CICS)	422
プログラムのリンク・エディット	406	CICS プログラム定義	422
変換、コンパイル、およびリンク・エディットのステートメント例 (CICS)	406	リテラル区切り文字：引用符またはアポストロフィ	423
CICS プログラム定義	407	プログラム例	423
プログラム例	407	COBOL 編集ルーチンと QMF との対話方法	423
PL/I 編集ルーチンと QMF との対話方法	407	法	423
編集ルーチンを COBOL で作成	412	2 バイト文字セット・データの処理	428
言語環境プログラム (LE) [®] を使用せずに TSO、SRPI、APPC、またはネイティブ OS/390 で編集ルーチンを COBOL で作成	412	DBCS データの編集コード	429
COBOL 編集ルーチンの、TSO、SRPI、APPC、およびネイティブ OS/390 との対話方法	413	編集ルーチンが受け取るもの	429
DSQUXDT のコンパイル	413	漢字列からのデータ	429
言語環境プログラム・ランタイム・ライブラリーの使用	414	文字列からのデータ	429
ランタイム・オプション・モジュールのアセンブル	414	編集ルーチンは必ず正しい結果を戻すこと	430
プログラムのリンク・エディット	414	ECSRSLT フィールドのオーバーフロー	430
コンパイルおよびリンク・エディットのステートメント例 (TSO、SRPI、APPC、またはネイティブ OS/390)	414	ー	430
プログラム例	416	報告書列の印刷	430
言語環境プログラム (LE) を使用して TSO、SRPI、APPC、またはネイティブ OS/390 で編集ルーチンを COBOL で作成	416	第21章 管理プログラム出口ルーチンを使用した QMF リソースの制御	433
LE を使用した COBOL 編集ルーチンの TSO、SRPI、APPC、およびネイティブ OS/390 との対話方法	417	クイック・スタート	433
DSQUXDT のコンパイル	418	IBM 提供の管理プログラム出口ルーチンの使用	434
プログラムのリンク・エディット	418	デフォルトの限界のアクティブ化	436
コンパイルおよびリンク・エディットのステートメント例 (TSO、SRPI、APPC、またはネイティブ OS/390)	418	管理プログラム出口ルーチンのリソースの制御方法	437
プログラム例	419	管理プログラムがリソース限界を知る方法	438
言語環境プログラム (LE) を使用して TSO、SRPI、APPC、またはネイティブ OS/390 で編集ルーチンを COBOL で作成	416	管理プログラムがリソース限界の到達を知る方法	440
LE を使用した COBOL 編集ルーチンの TSO、SRPI、APPC、およびネイティブ OS/390 との対話方法	417	リソース限界に到達すると起こること	440
DSQUXDT のコンパイル	418	ユーザー独自のリソース限界の定義	441
プログラムのリンク・エディット	418	ユーザー独自のリソース制御表の作成	443
コンパイルおよびリンク・エディットのステートメント例 (TSO、SRPI、APPC、またはネイティブ OS/390)	418	IBM 提供の管理プログラム出口ルーチンの変更またはユーザー独自のルーチン作成	445
プログラム例	419	管理プログラム出口ルーチンのプログラム・コンポーネント	446
言語環境プログラム (LE) を使用して TSO、SRPI、APPC、またはネイティブ OS/390 で編集ルーチンを COBOL で作成	416	TSO、SRPI、APPC、およびネイティブ OS/390 と管理プログラム出口ルーチンとの対話方法	447
LE を使用した COBOL 編集ルーチンの TSO、SRPI、APPC、およびネイティブ OS/390 との対話方法	417	CICS の管理プログラム出口ルーチンとの対話方法	448
DSQUXDT のコンパイル	418		
プログラムのリンク・エディット	418		
コンパイルおよびリンク・エディットのステートメント例 (TSO、SRPI、APPC、またはネイティブ OS/390)	418		
プログラム例	419		
言語環境プログラム (LE) を使用して TSO、SRPI、APPC、またはネイティブ OS/390 で編集ルーチンを COBOL で作成	416		
LE を使用した COBOL 編集ルーチンの TSO、SRPI、APPC、およびネイティブ OS/390 との対話方法	417		
DSQUXDT のコンパイル	418		
プログラムのリンク・エディット	418		
コンパイルおよびリンク・エディットのステートメント例 (TSO、SRPI、APPC、またはネイティブ OS/390)	418		
プログラム例	419		

QMF が管理プログラム出口ルーチンを呼び出す方法と時期	450	DB2 での 3 部分名による表または視点の指定	488
QMF が管理プログラムを呼び出す時点	450	3 部分名を使用した照会の指示	489
管理プログラム出口ルーチンへ入った		リモート作業単位に必要な接続の検査	489
ときに起こること	452	DB2 接続の検査	489
機能呼び出しのアドレス可能性の確立	457	DB2 (VM 版) 接続の検査	490
リソース制御情報を管理プログラム出口に渡す	458	DB2 (OS/390 版) 以外のロケーションへの	
DXEGOVA 制御ブロックの構造	458	QMF OS/390 ユーザーによるアクセスの準備	491
リソース制御表のアドレッシング	463	コマンド同義語表の作成	491
DXEXCBA 制御ブロックの構造	465	TSO 環境でのリモート・サーバー・コマンド同義語表の例	492
QMF セッション間のリソース制御情報の保管	474	DPRE コマンドをサポートする QMF の準備	493
ユーザー・アクティビティーの取り消し	475	その他のコマンドをサポートする QMF の準備	494
取り消されたアクティビティーに対するメッセージ	476	機能キー表の作成	494
CICS でのユーザー管理プログラム出口ルーチンの変換、アセンブル、リンク・エディット	478	QMF 管理プログラム制御表の更新	494
CICS の管理プログラム出口ルーチンの変換	478	各国語機能の QMF サーバーへのインストール	495
管理プログラム出口ルーチンのアセンブル	478	コード・ページ・サポート	495
管理プログラム出口ルーチンのリンク・エディット	479	APPLDATA 列使用上の制約	495
管理プログラム出口ルーチンの、TSO およびネイティブ OS/390 バッチでのアセンブルとリンク・エディット	480	一部の特特殊文字の使用回避	496
管理プログラム出口ルーチンのアセンブル	480	ユーザーがリモート・データベースにアクセスできるようにする	496
管理プログラム出口ルーチンのリンク・エディット	480	ユーザー・プロファイルの更新	496
DB2 管理プログラムの使用	481	現行 SQL 許可 ID へのアクセスの指定	496
リソースのモニター	481	ローカル・データベースへの接続	497
管理プログラム間の相違点	482	TSO、SRPI、APPC またはネイティブ OS/390 からの接続	497
最大プロセッサ時間の超過	482	CICS からの接続	497
DB2 管理プログラムの QMF への適用	483	リモート・データベースへの接続	497
RLST の選択	484	CICS におけるリモート・データベースでの実行	498
行を RLST に追加	484	ロケーション名の指定	498
第22章 リモート・データベース接続のカスタマイズ	485	DB2 (OS/390 版) の場合	499
クィック・スタート	485	Workstation Database Server の場合	499
必要なりモート・データベース接続の判別	486	DB2 (VM または VSE 版) の場合	499
リモート作業単位との接続	488	ユーザー・アクセスのためにデータのあるべき位置	499
DB2 間分散作業単位による接続 (DB2 (OS/390 版) のみ)	488	QMF オブジェクトでの処理	499
		表での処理	501
		SQL エラーの防止	501
		ユーザー ID の変換	502
		名前の変換	502
		DSQSPRID (TSO の場合)	502

各りモート QMF ロケーションからの QMF ユーザーの削除	503	初期化エラーの処理	532
自分のロケーションに管理者がアクセスでき るようにする	504	警告メッセージの処理	533
第23章 バッチ・プログラムとして実行する ための QMF のカスタマイズ.	505	印刷時の GDDM エラーの処理	534
クイック・スタート	505	印刷時の QMF エラーの処理	536
ユーザーが TSO でバッチ・モードを使用でき るようにする	506	表示エラーの処理	537
バッチ・モードでの操作権限	507	HEX 関数を使用する方法	537
RACF セキュリティの考慮事項	508	QMF 提供の 16 進およびビット編集コ ードを使用する方法	538
TSO SUBMIT コマンドによる OS/390 へ のジョブの送信	508	ユーザー作成編集ルーチンを用いた 2 進データの処理	538
TSO で QMF バッチ・ジョブ実行のため の JCL	509	パフォーマンスの問題の解決	538
JOB ステートメント	509	パフォーマンスを向上させるためのデ ータ・オブジェクトのリセット	538
EXEC ステートメント	510	ユーザーの報告書記憶域の増加	539
DD ステートメント	510	記憶域グループのポリューム・スペー スの拡大	540
PROFILE PREFIX ステートメント	511	CICS 領域のサイズの拡大	540
TSO または ISPF 使用によるフォアグラ ウンドでの QMF バッチの実行	512	REXX 関数パッケージの使用	541
プロシージャのデバッグ	513	診断援助機能を使用した問題の判別	541
QMF バッチ照会 / プロシージャ・アプリ ケーション (BATCH) の ISPF での使用	514	症状に応じた適切な診断援助機能の選択	541
アプリケーション使用権限の割り当て	514	QMF メッセージ・サポートを使用した問 題の診断	542
アプリケーションの使用	515	エラー・メッセージを出した QMF 機 能の判別	542
アプリケーションの開始	515	システム・エラー・メッセージの処理	543
プロンプト・パネルへの入力	516	SQL リターン・コードの処理	544
必須入力フィールド	516	QMF トレース機能の使用	544
オプションの入力フィールド	517	トレース・データ・セットの割り振り	544
バッチ・アプリケーションの変更	520	トレース機能の開始	545
適用可能な QMF コンポーネント	520	トレース出力の適切な詳細レベルの設 定	546
アプリケーションで可能な変更	521	モジュール・レベルでのトレース	549
アプリケーション変更の例	522	QMF トレース・データを表示する	549
ネイティブ OS/390 での QMF バッチの実行	525	QMF サービス・レベルの判別	551
ユーザーが CICS でバッチ・モードを使用でき るようにする	527	トレース機能の停止	552
端末からのバッチの実行	527	異常終了の診断	552
端末なしでのバッチの実行	528	OS/390 診断機能の使用	553
プロシージャのデバッグ	528	CICS 診断機能の使用	553
終了リターン・コード	529	TSO での QMF 割り込み機能の使用	555
第24章 障害追及と問題診断	531	割り込みの作成	555
クイック・スタート	531	割り込み作成後のトレース情報の表示	556
一般的な問題の障害追及	532	Q.ERROR_LOG 表からのエラー・ログ報 告書の使用	557
		IBM への問題報告	558
		ServiceLink を使用して既に報告されてい る問題の検索	558

第12章 必要な記憶域

この節は、各 QMF ユーザーが作成する可能性がある照会および報告書のための十分な記憶域があるかどうかを確認するために使用してください。ライブラリー・スペース、VSAM スペース、および DB2 データベース・スペースに関して前提となるハードウェアおよびソフトウェアならびに所要量については、1 ページの『第1部 QMF (OS/390 版) のインストール』を参照してください。

QMF 記憶域要件

QMF の使用を開始する前に、ユーザーが作成する QMF プログラムと QMF 報告書を収容するだけの十分な記憶域があるかどうかを確認する必要があります。

OS/390 領域は、下記のを収容できる十分な大きさが必要です。

すべての QMF モジュール： 2.8 MB の 31 ビット記憶域 (合計)

該当の環境には下記のを収容する必要があります。

ユーザーが照会を実行し、QMF 報告書データを保持するための記憶域： 0.5 MB ~ 1 MB 記憶域 (ユーザー 1 人当たり)

1 つの報告書について競合オプションを使用する場合、または照会の結果が大量のデータを戻すことになる場合は、1 MB を超える記憶域が必要になることがあります。これらの差を調整する方法の詳細な説明については、203ページの『DSQSBSTG (報告書データ用の記憶域の調整)』を参照してください。

両方の用途の記憶域を 16 MB より上に割り振ることができます。

OS/390 の記憶域

ユーザー記憶域について: QMF を実行するためには、0.5 ~ 1 MB が必要です。他のアプリケーションには、追加記憶域が必要です。たとえば、標準 TSO 環境で ISPF および GDDM を使用して実行する場合は、約 6 MB が必要になります。

必要な記憶域

QMF モジュールのほとんどは再入可能であり、EPLPA にロードできます。1 つの 52K モジュールについては、16 MB より下で 24 ビット・モードで実行する必要があります。このモジュールも再入可能であり、PLPA にロードできます。

CICS/MVS 領域

OS/390 のデフォルトの記憶域が実行のために割り振られている場合は、16 MB 未満の CICS 領域を指定すると、16 MB より上で 32 MB が使用可能になります。CICS 環境が QMF トランザクションを実行している場合は、16 MB 未満の領域を指定する必要があります。50 の QMF トランザクションを実行するためには、少なくとも 60 MB の使用可能な記憶域が 16 MB より上で必要です。16 MB を超える領域指定を行うと、2 つの割り振りが生じます。16 MB より下には、専用記憶域全体が割り振られます。これによって、CICS による処理に十分なローカル・システム・キュー域 (LSQA) が残されるわけではありません。領域サイズを変更するには、OS/390 システム出口の IEALIMIT または IEFUSI を使用します。16 MB より上の領域を 32 MB より大きい値に拡大するには、これらの出口の 1 つを使用します。

デフォルトの OS/390 領域を 16 MB より上にある 32 MB に限定して QMF を実行する場合は、約 26 の QMF トランザクションを実行することができます。

CICS/ESA 領域

CICS 3.1 では、動的記憶域 (DSA) は 16 MB より上および下に割り振ることができます。16 MB より上の DSA は、拡張 DSA (EDSA) と呼ばれます。DSA のサイズは、CICS システム初期化表パラメーターの DSASZE および EDSASZE で指定します。EDSASZE の CICS デフォルト値である 1536KB は小さ過ぎるため、QMF のユーザーをサポートできません。したがって、QMF を同時に使用するユーザーの数に応じて、EDSASZE を 16 ~ 50 MB の範囲まで拡大するようにお勧めします。QMF を同時に使用するユーザー 1 人につき 1 MB を 16 MB に加算するという式を使用するのが 1 つの方法です。本件の詳細については、*CICS/VSE* システム定義の手引き を参照してください。

第13章 QMF の開始

この章では、QMF を開始するさまざまな方法について説明しています。QMF の実行は、ISPF、TSO、CICS、ネイティブ OS/390 でバッチ・ジョブとして、あるいは QMF サーバーから開始することができます。

呼び出し可能インターフェースからの QMF の開始に関する情報については、*QMF アプリケーションの開発* を参照してください。

QMF を開始する前に

QMF を開始する前に、QMF を実行する環境を決定しておく必要があります。QMF の開始に使用する方法は、QMF を開始する環境によって異なります。

環境の確立

QMF を開始する前に、CICS および DB2 の環境、または QMF および DB2 の環境を確立しておく必要があります。

CICS と DB2 の接続

CICS 環境では、QMF は会話型トランザクションです。QMF のプログラムおよびトランザクション ID は、インストール中または QMF のカスタマイズ中に、処理プログラム・テーブル (PPT) およびプログラム管理テーブル (PCT) の中で CICS に対して定義されます。

QMF は CICS/DB2 接続機能を使用して、CICS 環境で DB2 のデータにアクセスします。CICS/DB2 接続機能は、リソース・マネージャー・インターフェースを使用して、DB2 のデータにアクセスし、データベース取り出しのつどタスク切り替えを必要とします。DB2 との CICS の接続の詳細については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

TSO と DB2 の接続

呼び出し接続機能 (CAF) によって、QMF と DB2 の間の接続が行われます。CAF インターフェースまたは DSNALI ロード・モジュールを確保して、使用可能にしておく必要があります。DB2 との QMF の接続の詳細については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

QMF の開始

クィック・スタート

表23 には、QMF の開始を設定できるように概要が示されています。

各例の中の *n* のシンボルは、各国語 ID (NLID) を表します。QMF を開始するときには使用したい各国語に対応する NLID を xviii ページの表1 から選択して、このシンボルを置き換えてください。たとえば、英語の QMF セッションを開始するには、QMFE を入力してください。

リストしている作業に関する詳細については、表の右端に記載してあるページを参照してください。

表 23. QMF を開始するためのオプション

作業	参照箇所
QMF を ISPF メニュー・オプションから開始するように設定するには、 既存の ISPF JCL に QMF を追加します。その上でユーザーが ISPF マスター・アプリケーション・メニューから QMF オプションを選択すると、QMF が開始されます。	182 ページ
QMF を ISPSTART コマンドの PGM 形式で開始するように設定するには、 次のように入力します。	185 ページ
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(...)	
QMF を ISPF でバッチ・モードで開始するように設定するには、 次のように入力します。	185 ページ
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(...DSQMODE=B, DSQRUN=aaa.bbb)	
CLIST (または EXEC) を使用しても、QMF をバッチ・モードで開始することができません。	
QMF を DSQMF モジュールで直接開始するように設定するには、 次のように入力します。	189 ページ
DSQMF DSQSBSTG=123456,DSQSIROW=0,DSQRUN=SAM.PROG1	
QMF を TSO CALL コマンドを使用して開始するように設定するには、 次のように入力します。	189 ページ
CALL 'QMF710.SDSQLOAD(DSQMF)' 'DSQMODE=I,DSQSSUBS=DB2T'	
QMF を TSO のバッチ環境で開始するように設定するには、 次のように入力します。	190 ページ
DSQMF ...DSQMODE=B,DSQRUN=aaa.bbb	
QMF をネイティブ OS/390 でバッチ・ジョブとして実行するように設定するには、 次のように入力します。	193 ページ
//RUN QMF EXEC PGM=DSQMF,PARM='DSQMODE=B,DSQRUN=aaa.bbb'	

表 23. QMF を開始するためのオプション (続き)

作業	参照箇所
QMF を消去画面から開始するには、次のように入力します。 QMFn	194 ページ
QMFn の後に、199ページの『第14章 開始手順のカスタマイズ』で説明している QMF プログラム・パラメーターの値を続けます。	
QMF を CICS アプリケーションから開始するには、次のように入力します。 EXEC CICS START TRANSID('QMFN') FROM('...') TERMID('NAME')	194 ページ
QMF プログラム・パラメーターの値は、単一引用符で囲み、FROM キーワードの後に続けます。CICS アプリケーションでは任意の QMF プログラム・パラメーターが使用できます。端末 ID (TERMID) は、対話式セッションでは必須ですが、非対話式セッションではオプションです。	
TSO EXEC を作成して QMF を開始するには、QMF からプログラム・モジュールおよびデータ・ファイルが使用できるようになっていること、および GDDM と DB2 (VM 版) または DB2 (VSE 版) の考慮事項が満たされていることを確認する必要があります。	191 ページ

QMF を ISPF のもとで実行するように設定

ユーザーに ISPF サービスを使用して QMF を開始させることができます。ISPF 環境に QMF リソースを定義する JCL を追加することができます。これには次の 3 つの方法があります。

- QMF を追加できる初期ダイアログが ISPF にある。
- QMF を直接開始する初期ダイアログで初期ダイアログを置き換える。
- QMF をプログラム・ダイアログとして開始するための CLIST を作成する。

上記のいずれかの方法を使用して、それ以外の方法を開始することができます。たとえば、CLIST から初期ダイアログを実行することができます。

JCL の使用: QMF プログラムの所在を指し示す JCL を使用する場合、その JCL は常に初期ダイアログ内になければなりません。

ISPF のもとで QMF を実行するには、ISPF SELECT サービスを使用して QMF プログラム・ダイアログを開始する必要があります。TSO 呼び出した場合は TSO コマンドを使用すると、予測できない結果になる可能性もあります。

制約事項:

1. QMF をコマンド・ダイアログとして実行することはできません。たとえば、次のステートメントは無効です。

```
ISPEXEC SELECT CMD(DSQMF) NEWAPPL(DSQE)
ISPSTART CMD(DSQMF) NEWAPPL(DSQE)
```


ヒント: QMF の開始に対しては、メニューによる直接操作の方が、CLIST による操作に比べてはるかに速くなる可能性があります。TSO ログオン・プロシージャによってユーザー・リソースをすべて割り振った場合は、メニュー・オプションに関して作成する CLIST には、割り振るリソースがありません。したがって、残される機能は QMF の開始の 1 つだけであり、これは CLIST なしで行うことができます。

メニューに複数のオプションを追加することができます。たとえば、ABC が試験的な DB2 サブシステムであるのに対して、DSN が実動サブシステムであるとします。この場合、メニューにオプションを 2 つ、つまり、各サブシステムごとに 1 つずつ追加することができます。各オプションにそれぞれ異なる CLIST を呼び出させる場合もあれば、サブシステムに対応する定位置パラメータを使用して 1 つの CLIST を作成する場合があります。メニューの)PROC セクションの追加行は、次のようになります。

```
2, 'PGM(DSQMF) NEWAPPL(DSQE) PASSLIB PARM(DSQSSUBS=DB2SSFDX)'
3, 'PGM(DSQMF) NEWAPPL(DSQE) PASSLIB PARM(DSQSSUBS=DB2SSFDY)'
```

LIBDEF ステートメントの使用

QMF プログラムおよび DB2 プログラムに対して ISPF LIBDEF サービスを使用するためには、まずプログラム・ライブラリーを "DSQLLIB" という固有の QMF DDNAME に割り振ってください。次に、ISPF LIBDEF ステートメントの LIBRARY オプションの ID 値として DDNAME "DSQLLIB" を指定してください。

たとえば、QMF および DB2 プロダクト・ライブラリーを割り振るには、次のような TSO 割り振りおよび ISPF LIBDEF ステートメントを指定します。

```
ALLOC FI(DSQLLIB) DA('QMF710.SDSQLOAD', 'DSN610.SDSNEXIT', 'DSN610.SDSNLOAD') SHR REUSE
LIBDEF ISPLLIB LIBRARY ID(DSQLLIB)
```

ISPF LIBDEF サービスを使用してプログラム・ライブラリーを割り振るには、次のような CLIST を書き込むことができます。

```

/*****
/* Allocate QMF and DB2 Programs to DSQLLIB */
/*****
ALLOC FI(DSQLLIB) SHR REUSE
      DA('QMF710.SDSQLOAD',          +
         'DSN610.SDSNEXIT',          +
         'DSN610.SDSNLOAD')
/*****
/* Allocate QMF libraries used for GDDM */
/*****
ALLOC FI(ADMGGMAP) DA('QMF710.DSQMAPE') SHR REUSE
ALLOC FI(ADMCFORM) DA('QMF710.DSQCFORM') SHR REUSE
```

QMF の開始

```
ALLOC FI(DSQCFRM) DA('QMF710.DSQCFRM') SHR REUSE
ALLOC FI(ADMGDF) DA('QMF710.CHARTLIB') SHR REUSE
/*****
/* Allocate QMF product datasets */
/*****
ALLOC FI(DSQPRINT) SYSOUT(Z) LRECL(133) RECFM(F B A) BLKSIZE(1330)
ALLOC FI(DSQPNLE) DA('QMF710.DSQPNLE') SHR REUSE
ALLOC FI(DSQDEBUG) SYSOUT(Z) LRECL(121) RECFM(F B A) BLKSIZE(1210)
ALLOC FI(DSQDUMP) SYSOUT(Z) LRECL(125) RECFM(V B A) BLKSIZE(1632)
ALLOC FI(DSQSPILL) NEW UNIT(SYSDA) SPACE(1,1) CYLINDERS
ALLOC FI(DSQEDIT) NEW UNIT(SYSDA)
/*****
/* Issue ISPF LIBDEF for QMF libraries used for ISPF */
/*****
ISPEXEC LIBDEF ISPLLIB LIBRARY ID(DSQLLIB)
ISPFEXE LIBDEF ISPPLIB DATASET ID('QMF710.SDSQPLBE')
ISPFEXE LIBDEF ISPSLIB DATASET ID('QMF710.SDSQSLBE')
ISPFEXE LIBDEF ISPMLIB DATASET ID('QMF710.SDSQMLBE')
/*****
/* Start QMF dialog using PASSLIB */
/*****
ISPEXEC SELECT PGM(DSQMFE) NEWAPPL(DSQE) PASSLIB
/*****
/* Free ISPF LIBDEF for QMF libraries used for ISPF */
/*****
ISPEXEC LIBDEF ISPLLIB LIBRARY ID( )
ISPEXEC LIBDEF ISPPLIB LIBRARY ID( )
ISPEXEC LIBDEF ISPSLIB LIBRARY ID( )
ISPEXEC LIBDEF ISPMLIB LIBRARY ID( )
FREE FI(DSQLLIB)

/*****
/* Free QMF product datasets */
/*****
FREE FI(DSQPRINT)
FREE FI(DSQPNLE)
FREE FI(DSQDEBUG)
FREE FI(DSQDUMP)
FREE FI(DSQSPILL)
FREE FI(DSQEDIT)
/*****
/* Free QMF libraries used for GDDM */
/*****
FREE FI(ADMGGMAP)
FREE FI(ADMCFORM)
FREE FI(DSQCFRM)
FREE FI(ADMGDF)
```

上記の CLIST は、ISPF がすでに実行されていて他の ISPF リソースがすでに割り振られていることを想定しています。

ISPLLIB データ・セットの割り振り

ISPLLIB を使用して QMF プログラム・ライブラリーを割り振ることができます。QMF ライブラリーを ISPLLIB に関する既存の割り振りに追加してください。たとえば、次のように指定します。

```
ALLOC DA('QMF710.SDSQLOAD',          +
         '..... other allocations ...' DDNAME(ISPLLIB)
```

ISPSTART コマンドによる QMF の開始

ユーザーが、外部の ISPF から ISPF ダイアログとして QMF を開始できるようにするための CLIST を開発した場合、このような CLIST を TSO READY モードで呼び出すと、次に行われるセッションに関してリソースを割り振った上で、QMF を開始することができます。これを行うには、READY モードで、あるいは CLIST (または EXEC) で、次のステートメントをコマンド行から入力します。

```
ISPSTART PGM(DSQMFE) NEWAPPL(DSQE) PARM(...)
```

ISPSTART は ISPF コマンドの 1 つです。この場合は、QMF を新規の ISPF アプリケーション DSQE として開始します。QMF プログラムの DSQMFE は、DSQE というアプリケーション ID で実行する必要があります。(言い換えれば、各国語 ID はそれぞれ同じものでなければならないということです。)

オプションの PARM オペランドは、パラメーター値を QMF プログラムの DSQMFE に渡すもので、次のようになります。

```
PARM(DSQSBSTG=256000,DSQSIROW=50,DSQSRUN=SAM.PROG1)
```

パラメーターの詳細については、199ページの『第14章 開始手順のカスタマイズ』を参照してください。

QMF は ISPSTART コマンドの PGM 形式を用いて開始することができます。

PGM は ISPSTART コマンドのオブジェクトです。PGM を用いて、QMF プログラムの DSQMFE を指定します。これを行うには、次のステートメントを TSO のコマンド行から入力するか、または EXEC のステートメントとして組み込みます。

```
ISPSTART PGM(DSQMFE) NEWAPPL(DSQE)
          PARM(DSQSBSTG=n1,...)
```

ISPF でバッチ・モードによる QMF の開始

QMF の実行をバッチ・モードで開始することができます。リソースおよび時間を節約するために、QMF をバッチ・モードで開始することができます。

QMF の開始

QMF の開始には、ISPF を使用しますが、CLIST は使用してもしなくても構いません。

- CLIST を使用しない場合

QMF の開始に CLIST を使用しない場合は、JCL の SYSTSIN データ・セットに次のステートメントを入れます。

```
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(...DSQSMODE=B,DSQSRUN=aaa.bbb)
```

- CLIST を使用する場合

QMF の開始を CLIST から行う場合は、JCL の SYSTSIN データ・セットに次のステートメントを入れます。

```
ISPSTART CMD(clist_name) NEWAPPL
```

ここで、*clist_name* は QMF を開始する CLIST の名前です。

上記の例のいずれでも、PARM は適正な操作モード (DSQSMODE=B) を確立し、実行するプロシージャー (DSQSRUN=aaa.bbb) を識別し、そのプロシージャーの変数を組み込むことができます。

PARM の後に続く省略記号 (...) は、DSQSMODE および DSQSRUN パラメータの必須値に加えて、ユーザーが組み込むことのできるオプション・パラメータ値を表します。図33 に示されているプロシージャーの名前には、所有者の許可 ID を含める必要があります。この例では、プロシージャーの名前が PROCA で、所有者のユーザー許可 ID が JONES であると想定しています。

```
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSMODE=B,DSQSRUN=JONES.PROCA)
```

図33. ユーザー名およびプロシージャー名を指定した ISPF でバッチ・モードによる QMF の開始

プロシージャーの実行後、QMF は終了し ISPF に制御を戻します。そこで ISPF は別のプロシージャーまたはコマンドを実行することができます。ISPF が終了すると、TSO は SYSTSIN にある次の TSO コマンドを実行します。SYSTSIN にあるコマンドがすべて実行されると、ジョブ・ステップは終了します。

ISPF のもとでの QMF の開始の例

QMF の開始および QMF へのパラメータの受け渡しの例をいくつか挙げておきます。

- ISPF を CLIST から開始し、QMF を初期ダイアログとして指定する場合：

```
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSIROW=150,DSQSRSTG=0)
```

このステートメントでは、DSQSIROW (報告書の最初の表示の前に取り出される行数) に 150 の値を渡し、DSQSRSTG (記憶域の予約量) に 0 の値を渡します。

- ISPF 内で作動する CLIST から開始する場合 :

```
ISPEXEC SELECT PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSSUBS=DB2SSFDX)
```

このステートメントでは、DB2 サブシステムの名前として DB2SSFDX を渡します。

- ISPF メニューから開始する。

```
)PROC
```

```
&SEL = TRANS( TRUNC (&OPT, '.')
              1, 'PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSDBCS=YES)'
              .
              .
              .
```

このコードでは、ユーザーがオプション 1 を選択すると、DSQSDBCS に関して必ず YES を渡します。

- CLIST から開始し、初期プロシージャを指定する場合 :

```
ISPSTART PGM(DSQMF) NEWAPPL(DSQE) PARM(DSQSRUN=Q.IPROC(&&&TABLE=Q.STAFF))
```

このステートメントでは DSQSRUN パラメーターを使用して、次のことを行ないます。

- QMF の開始時に実行される初期プロシージャ Q.IPROC を指定する。
- 変数 &TABLE に関して値 Q.STAFF をプロシージャに渡す。

DSQSRUN パラメーターを上記の例のように指定すると、その結果として次の QMF コマンドが出されます。

```
RUN Q.IPROC(&TABLE=Q.STAFF)
```

QMF を TSO のもとで実行するように設定

TSO では、ユーザーは以下の方法で QMF を開始することができます。

- CLIST または EXEC で DSQMF プログラムを使用
- TSO CALL コマンドを使用
- TSO のバッチ環境で

TSO ID の定義

TSO のもとで QMF を開始する際には、DB2 出口ルーチン DSN3@ATH を介して許可 ID を割り当てます。(IBM ではデフォルトの出口ルーチンも提供し

QMF の開始

ています。) ユーザーの TSO ログオン ID が指定されている場合は、このルーチンによって割り当て許可 ID のリストが戻されます。

ID のタイプの選択

次のように複数の許可 ID を割り当てることができます。

- 単一の SQL 許可 ID
- 単一の 1 次許可 ID および 1 つまたは複数の 2 次許可 ID

SQL 許可 ID は、1 次許可 ID と 2 次許可 ID のいずれかであることが必要です。1 次 ID と 2 次 ID は両方とも、ユーザーのセッションの期間中固定されています。

許可 ID は、文字数が 8 文字を超えない名前です。先頭文字は文字であることが必要ですが、他の 7 文字は文字でも数字でも構いません。これらの名前の規則については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

許可 ID はすべての DB2 特権のソースになります。ユーザーは、1 つまたは複数の割り当て許可 ID を用いて、QMF または SPUFI セッションを始めます。各許可 ID ごとに、あらゆる種類の DB2 特権を任意の数だけ所有することができます。たとえば、ユーザー A の許可 ID の 1 つが JONES であり、JONES には表 SMITH.TABLEA に対する SELECT 特権があります。したがって、ユーザー A も SMITH.TABLEA に対して SELECT 特権を持ち、この表に対して SELECT 照会を実行することができます。

TSO のもとでの QMF

デフォルトの出口ルーチン DSN3@ATH を変更しないで使用することにした場合は、次のようになります。

- ユーザーの 1 次許可 ID および SQL 許可 ID は、そのユーザーの TSO ログオン ID に一致する。
- 2 次許可 ID は割り当てられない。

ログオン・プロシージャでの

DD ステートメントの使用: ログオン・プロシージャでの DD ステートメントで、ユーザーにリソースを割り振ることができます。(QMF が ISPF ダイアログとして開始される場合にのみ使用可能な ISPF/PDF 編集プログラムでは、追加のリソースは必要ありません。)

新規の QMF ユーザーに、そのユーザーがログオンするときに呼び出される TSO ログオン・プロシージャを提供することができます。このカタログ式プロシージャは端末モニター・プログラム (TMP) を呼び出します。

TMP は、TSO セッションの間中、ユーザーと端末の間の基本インターフェースになります。インストール先システムが、IBM 提供の TMP ではなく、独自の TMP を使用している場合は、以下の説明の中に該当しないものがあり得ます。ユーザーが QMF の開始に使用する CLIST または EXEC を開発することができます。このような CLIST または EXEC の中では、必要なデータ・セットの多くを TSO ALLOCATION ステートメントによって割り振ることができます。特に、ユーザーに固有のデータ・セットを割り振ることができます。

次のステートメントを CLIST 内で使用して、固有のライブラリーをそのユーザーの CHART 書式に関して割り振ります。割り振られたライブラリーの名前は、ユーザーの TSO ログオン ID (変数 &SYSUID で表される) で始まります。

```
ALLOC DDNAME(DSQUCFRM) DSNAME('&SYSUID..CHARTLB.DATA') OLD
```

QMF セッションの終了後は、TSO FREE ステートメントを CLIST または EXEC で使用して、データ・セットの割り振り解除を行うこともできます。

DSQQMFE モジュールにより QMF を直接開始

1) READY モードでコマンド行から、あるいは 2) CLIST または EXEC で、DSQQMFE を入力することによって、TSO のもとでの QMF の実行を開始することができます。

```
DSQQMFE DSQSBSTG=123456,DSQSDBG=ALL,DSQSIROW=0,DSQSRUN=SAM.PROG1
```

この例では、DSQQMFE の後に続くパラメーター・ストリングはオプションです。

QMF を ISPF からは独立して、TSO で開始するときは、次のリターン・コードが有効です。

- 0 実行は正常に行われました
- 4 警告条件が発生しました
- 8 エラー条件が発生しました
- 16 重大エラーが発生しました

TSO CALL コマンドを使用した QMF の開始

TSO CALL コマンドを使用して QMF を開始することもできます。QMF ロード・ライブラリーの名前を指定し、次の例のように、データ・セット名の後に続けてオプションのプログラム・パラメーターを渡します。

```
CALL 'QMF710.SDSQLOAD(DSQMFE)' 'DSQSMODE=I,DSQSSUBS=DB2T'
```

QMF の開始

CALL コマンドの期間中、QMF ロード・ライブラリーが TASKLIB になります。ただし、DB2 ライブラリーおよび GDDM ライブラリーに対するアクセスを QMF に付与して、それらのプロダクトとのプログラム・インターフェースを LOAD できるようにする必要があります。DB2 ライブラリーおよび GDDM ライブラリーは、ほとんどの場合、TASKLIB の一部にはなっていません。DB2 ライブラリーおよび GDDM ライブラリーが使用不可の場合は、QMF はエラーを伴って終了します。

TSO CALL を ISPF から独立して使用して QMF を開始した場合のリターン・コードは、189ページの『DSQQMFE モジュールにより QMF を直接開始』の節に示したリターン・コードと同じものです。

TSO のバッチ環境での QMF の開始

ISPF サービスを使用しないで QMF を開始する場合は、JCL の SYSTSIN データ・セットに次のステートメントを入れます。

```
DSQQMFE ...DSQSMODE=B,DSQSRUN=aaa.bbb
```

ここで 'DSQSMODE=B' は、適正な操作モードを確立するものであり、'DSQSRUN=aaa.bbb' は、実行されるプロシージャーを識別するものです。プロシージャーには、プロシージャー名として変数を組み込むことができます。(これには所有者の許可 ID を含める必要があります。)

省略記号は、必須パラメーター DSQSMODE および DSQSRUN に加えて、ユーザーが組み込むことのできるオプション・パラメーター値を表しています。

TSO のもとでの QMF の開始の例

ISPF から独立して作動する QMF の開始およびそのような QMF へのパラメーターの受け渡しの例をいくつか次に挙げておきます。

- TSO READY モードから開始する場合：

```
DSQQMFE DSQSBSTG=50000,DSQSDBUG=NONE,DSQSMODE=B
```

このステートメントでは、L2 トレース (DSQSDBUG=NONE) をオンにし、DSQSBSTG (報告書用の最大記憶域) の値として 5000 を渡し、DSQSMODE (操作のモード) の値として B (バッチ) を渡します。

- CLIST から開始し、初期プロシージャーを指定する場合：

```
DSQQMFE DSQSRUN=Q.IPROC(&&TABLE=Q.STAFF)
```

このステートメントでは DSQSRUN パラメーターを使用して、次のことを行ないます。

- QMF の開始時に実行される初期プロシージャー Q.IPROC を指定する。
- 変数 &TABLE に関して値 Q.STAFF をプロシージャーに渡す。

DSQSRUN パラメーターを上記の例のように指定すると、その結果として次の QMF コマンドが出されます。

```
RUN Q.IPROC(&TABLE=Q.STAFF
```

TSO EXEC の作成

QMF を開始するために TSO EXEC を作成する場合は、プログラム・ロード・ライブラリーおよびデータ・セットが QMF で使用できるようにしておき、GDDM および DB2 に関する考慮事項に適合するようにしておくことが必要です。

プログラム・ロード・ライブラリーの検査

QMF の開始前に、ISPF、ISPF/PDF、QMF、DB2、および GDDM 用の DB2 のデータベースおよびロード・ライブラリーが、STEPLIB ステートメントから、または CLIST を介して使用可能であることが必要です。図34 にロード・ライブラリーをリストしてあります。

```
//*****
//*          PROGRAM LOAD LIBRARIES          *
//*****
//STEPLIB DD DSN=QMF710.SDSQLOAD,DISP=SHR      * QMF MODULES *
//          DD DSN=ISR.V4R1M0.ISRLOAD,DISP=SHR * PDF MODULES * Opt. for non-ISPF users
//          DD DSN=ISP.V4R1M0.ISPLOAD,DISP=SHR * ISPF MODULES * Opt. for non-ISPF users
//          DD DSN=DSN710.SDSNEXIT,DISP=SHR    * DB2 MODULES *
//          DD DSN=DSN710.SDSNLOAD,DISP=SHR    * DB2 MODULES *
//          DD DSN=GDDM230.SADMMOD,DISP=SHR    * GDDM MODULES *
```

図34. ISPF、ISPF/PDF、QMF、DB2、および GDDM 用のプログラム・ロード・ライブラリー

TSO に関する考慮事項

EXEC に関する TSO の検索順序については、インストール先システムで確立された DD 名を使用します。この検索順序は、TSO デフォルト・モジュールの IRXTSPRM と IRXISPRM、TSO EXECUTIL コマンド、および TSO ALTLIB コマンドにおける設定値の影響を受けます。192ページの図35 には TSO で使用されるデータ・セットをリストしてあります。インストール先システムでの REXX EXEC に関する検索順序がわからない場合は、SDSQEXCE を SYSEXEC と SYSPROC の両方に割り振ります。

QMF の開始

```
/******  
/*          DATASETS USED BY TSO          *  
/******  
//SYSPROC DD DSN=SYS2.CLIST,DISP=SHR          * CLIST Library  
//          DD DSN=QMF710.SDSQCLTE,DISP=SHR  
//SYSEXEC DD DSN=SYS2.EXEC,DISP=SHR  
//          DD DSN=QMF710.SDSQEXCE,DISP=SHR  
//SYSHelp DD DSN=SYS1.HELP,DISP=SHR  
//EDT     DD DSN=&EDIT,UNIT=SYSDA,SPACE=(1688,(40,12))
```

図 35. TSO で使用されるデータ・セット

QMF データ・セットの検査

下に挙げるデータ・セットのリストは、QMF で使用されるものです。これらのファイルは、DSQ で始まる DD 名に割り振られます。割り振りを変えたい場合は、呼び出し EXEC を変更する必要があります。

DSQPNLE

QMF パネル・ファイル

DSQUDUMP

QMF スナップ・ダンプ出力

DSQDEBUG

QMF トレース・ダンプ出力

DSQPRINT

印刷データ出力

DSQSPILL

予備データ・ファイル

DSQEDIT

編集転送ファイル

QMF710.SDSQLOAD

QMF ロード・ライブラリー

GDDM データ・セットの検査

GDDM データ・セットは、次の DD 名に割り振られます。

ADMGGMAP

QMF マップ・パネルの GDDM マップ・グループ

ADMCFORM

QMF 提供の図表書式

DSQUCFRM

ユーザー定義の ICUFORMS を保管する

図36 に QMF/GDDM データ・セットをリストしてあります。

```
//*****
//*          QMF/GDDM DATA SETS          *
//*****
//ADMGGMAP DD DSN=QMF710.DSQMAPE,DISP=SHR * GDDM Map Group
//ADMCFORM DD DSN=QMF710.DSQCHART,DISP=SHR * QMF-Supplied Chart Forms
//DSQUCFRM DD DSN=aaaaaaa,DISP=SHR      * Saves User-Defined ICUFORMS
//ADMCDATA DD DSN=xxxx,DISP=SHR
//ADMGDF   DD DSN=xxxx,DISP=SHR
//ADMSYMBL DD DSN=xxxx,DISP=SHR
```

図 36. QMF/GDDM データ・セット

QMF をバッチ・ジョブとしてネイティブ OS/390 で実行するように設定

QMF をネイティブ OS/390 でバッチ・ジョブとしてユーザーに開始させることができます。予備ファイルが存在する個所、パネルが保管される個所、パネル・ファイル名、および他の表と QMF オブジェクトの名前および所在を定義する JCL を作成する必要があります。これらのオブジェクトの使用法の詳細については、241ページの『第16章 エンド・ユーザーのための QMF サポートの確立』を参照してください。

QMF コマンドを発行する場合は、QMF の初期プロシージャの名前を指定する必要があります。194ページの図37 では、I=X であり、ここで X は QMF プロシージャ名です。

QMF が開始され、ついでプロシージャ X を実行します。プロシージャ X が完了すると、QMF は終了します。QMF リターン・コードがレジスター 15 に戻されますから、標準 JCL 条件コード・テストを使用して、JCL でそのリターン・コードをテストすることができます。

QMF の開始

```
//RUNQMF EXEC PGM=DSQQMFE,PARM='M=B,I=X,P=QMF710,S=DSN'  
//*****  
//* Program load libraries  
//*****  
//STEPLIB DD DSN=QMF710.SDSQLOAD,DISP=SHR  
// DD DSN=DSN710.SDSNEXIT,DISP=SHR  
// DD DSN=DSN710.SDSNLOAD,DISP=SHR  
// DD DSN=GDDM.GDDMLOAD,DISP=SHR  
//*****  
//* QMF/GDDM maps *  
//*****  
//ADMGGMAP DD DSN=QMF710.DSQMAPE,DISP=SHR  
//*****  
//* Datasets used by QMF *  
//*****  
//DSQPRINT DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330)  
//DSQDEBUG DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210)  
//DSQDUMP DD SYSOUT=*,DCB=(RECFM=VBA,LRECL=125,BLKSIZE=1632)  
//DSQSPILL DD DSN=&&SPILL,DISP=(NEW,DELETE),  
// UNIT=SYSDA,SPACE=(TRK,(100),RLSE),  
// DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)
```

図 37. ネイティブ OS/390 のバッチで QMF プロシージャーを実行するための JCL

考慮事項: TSO バッチによる実行に慣れている場合は、QMF プロシージャーで使用されるデータ・セット名は完全修飾名でなければならないことを承知しておく必要があります。TSO の接頭部および接尾部は、ネイティブ OS/390 では使用できません。

QMF を CICS のもとで実行するように設定

QMF を CICS 用として調整し終えたら、次のように QMF トランザクション (デフォルトは QMFE) を CICS 画面から開始します。

QMFE parameters

ここで、QMFE は QMF に関する PCT で定義されているトランザクション ID であり、parameters は所要のプログラム・パラメーターを表します。

また、CICS START コマンドを発行するためのアプリケーション・プログラムを書き、次の例のようにプログラム・パラメーターを指定することもできます。

```
EXEC CICS START TRANSID(QMFE) FROM (parameters) TERMID('id')
```

ここで parameters は QMF プログラム・パラメーターです。

端末 ID (TERMID) は対話式セッション (DSQSMODE = I の場合) では必須ですが、非対話式セッション (DSQSMODE = B の場合) ではオプションで

す。端末 ID によって、呼び出し側の CICS アプリケーションが実行される端末を指定する場合は、その CICS アプリケーションが終了すると、QMF セッションが開始されます。端末 ID を指定する場合は、該当の端末が存在し、しかも使用可能でなければなりません。また、その端末 ID が、START コマンドが出されるシステムのローカル端末かリモート端末のどちらかとして定義されるようにすることも必要です。

CICS/DB2 接続機能の使用

CICS のもとで QMF を開始する際には、DB2 出口ルーチン DSN3@SGN を介して DB2 サインオン処理を行います。(IBM ではデフォルトの出口ルーチンも提供しています。) CICS リソース管理テーブル (RCT) のトランザクションに関する AUTH 項目による指定に従って取得された ID が指定されている場合は、このルーチンによって割り当て許可 ID のリストが戻されます。

QMF は CICS 接続機能を使用して、CICS 環境で DB2 のデータにアクセスします。CICS 接続機能の接続については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。QMF 固有の情報はここで説明します。

トランザクション ID に関するプラン ID および許可 ID は、RCT の中で指定されます。次のようなステートメントを使用します。

```
DSNCRCT TYPE=ENTRY, TXID=QMFE, PLAN=QMF710, AUTH=DEPT1
DSNCRCT TYPE=ENTRY, TXID=QMFQ, PLAN=QMF710, AUTH=Q
```

QMFE トランザクションを呼び出すユーザーは、1 次許可 ID DEPT1 のもとで操作を行います。同様に、QMF 管理者は、QMFQ トランザクションを使用し、1 次許可 ID Q によって操作を行うことができます。RACF がシステムにインストールされている場合は、許可 ID は有効な RACF ID であることが必要です。トランザクション ID は、区画制御テーブル (PCT) でも定義されている必要があります。

QMF プログラムはインストール中にリンク・エディットおよびバインドされるので、CICS では追加のステップは必要ありません。

CICS 接続機能はリソース・マネージャー・インターフェースを使用して、DB2 のデータにアクセスします。データベースからの各取り出しのつどタスク切り替えがあります。すべてのユーザーにとって受け入れ可能な応答を維持するには、1 回の照会で取り出すことのできる行数を制限しなければならない場合があります。管理プログラムの使用による限界の設定の詳細については、433 ページの『第21章 管理プログラム出口ルーチンを使用した QMF リソースの制御』を参照してください。

QMF の開始

CICS のもとでの QMF

デフォルトの出口ルーチン DSN3@SGN を変更しないで使用することにした場合、1 次許可 ID および SQL 許可 ID は、CICS RCT のトランザクションに関する AUTH 項目で取得された ID と同じものになります。

CICS のもとでの QMF の開始の例

CICS のもとで実行中に QMF を開始する場合の例をいくつか次に挙げておきます。

- CICS の消去画面から開始する場合：

```
QMFE DSQSIROW=150,DSQSBSTG=500000,DSQSPILL=NO
```

このステートメントでは、DSQSIROW (画面表示の前に取り出される行数) の値として 150 を渡し、DSQSBSTG (報告書用の最大記憶域) の値として 500 000 を渡し、QMF 予備ファイルをオフにします (DSQSPILL=NO)。

- CICS の消去画面から開始し、初期プロシージャーを指定する場合：

```
QMFE DSQSRUN=Q.IPROC(&&TABLE=Q.STAFF)
```

このステートメントでは DSQSRUN パラメーターを使用して、次のことを行ないます。

- QMF の開始時に実行される初期プロシージャー Q.IPROC を指定する。
- 変数 &TABLE に関して値 Q.STAFF をプロシージャーに渡す。

DSQSRUN パラメーターを上記の例のように指定すると、その結果として次の QMF コマンドが出されます。

```
RUN Q.IPROC(&TABLE=Q.STAFF)
```

QMF を SRPI からサーバーとして実行するように設定

QMF を SRPI から実行するように設定することができます。この SRPI 方式の使用が必要になるのは、ユーザー・ワークステーションの 3270 端末エミュレーション画面のネットワーク定義だけが必要な場合です。

SRPI の基本要件は 3270 エミュレーションであり、これはコミュニケーション・マネージャー /2 で提供されます。SRPI はオペレーティング・システムのホスト側にも提供されています。TSO ユーザーは、*TSO/E Version 2 Guide to the Server-Requester Programming Interface* の資料も準備しておく必要があります。

SRPI 接続方式を使用してワークステーションを OS/390 システムに接続する方法の一部として、次のようなものがあります。

- IBM トークンリング・ネットワークをローカルで使用する。
- IBM 3174 制御装置に対して分散機能 (DFT) モードを使用する。
- 同期データ・リンク制御 (SDLC) をリモートで使用する。

接続で ACF/VTAM[®] を使用する場合は、VTAM ログモードを使用し、拡張データ・ストリーム (EDS) ビット・セットで定義しておく必要があります。

第14章 開始手順のカスタマイズ

この章では、ユーザーの QMF セッションのカスタマイズに役立てるために、プログラムにパラメーターを渡す場合に使用できるさまざまな方法について説明します。

呼び出し可能インターフェースまたは REXX EXEC でのパラメーターの受け渡しの説明については、*QMF アプリケーション開発の手引き* を参照してください。

プログラム・パラメーターの要約

次の表は、パラメーターの名前と短縮形および各パラメーターの適切な環境を示したものです。CICS でのみ使用されるパラメーターには短縮形がありません。

表 24. プログラム・パラメーター

名前	短縮形	環境	説明
DSQSBSTG	B	TSO、CICS	報告書のための最大記憶域
DSQSDBCS	K	TSO、CICS	非 DBCS 装置の DBCS サポート
DSQSDBNM	D	TSO、CICS	初期データベース・ロケーションの名前
DSQSDBQN	--	CICS	DSQDBQT で指定した記憶域の名前
DSQSDBQT	--	CICS	トレース記憶域のタイプ
DSQSDEBUG	T	TSO、CICS	トレース -- ALL または NONE
DSQSIROW	F	TSO、CICS	データベースから取り出される行
DSQSMODE	M	TSO、CICS	対話モードまたはバッチ・モード
DSQSPILL	L	TSO、CICS	予備ファイルの使用
DSQSPLAN	P	TSO	QMF アプリケーション・プランの名前
DSQSPRID	U	TSO	プロファイル・キー -- TSOID または PRIMEID
DSQSRSTG	R	TSO	予約記憶域の量
DSQSRUN	I	TSO、CICS	実行する QMF プロシージャの名前

開始手順のカスタマイズ

表 24. プログラム・パラメーター (続き)

名前	短縮形	環境	説明
DSQSSPQN	--	CICS	QMF 予備ファイルの名前
DSQSSUBS	S	TSO	DB2 サブシステムの名前

クイック・スタート

表25 には、プログラム・パラメーターを使用して、QMF セッションのさまざまな局面をカスタマイズする方法を示してあります。例で使用しているコマンド構文が適用されるのは、QMF を TSO で DSQQMFE モジュールによって開始する場合、および QMF を CICS で消去画面から開始する場合です。別の方法を使用して QMF を開始する場合は、179ページの『第13章 QMF の開始』に示してあるコマンド構文を参照してください。

各例の中の n のシンボルは、各国語の ID (NLID) を表します。QMF を開始するときに使用したい各国語に対応する NLID を xviiiページの表1 から選択して、このシンボルを置き換えてください。たとえば、英語の QMF セッションを開始するには、QMFE を入力してください。

リストしている作業に関する詳細については、表の右端に記載してあるページを参照してください。

表 25. パラメーターの受け渡し

作業	参照箇所
QMF 照会および報告書用として使用される記憶域の量に限界を設定する場合、CICS に関して 0 または 500 000 バイト以外の限界が必要であれば、DSQSBSTG パラメーターを使用します。たとえば、限界として 1 000 000 バイトを設定する場合は、次のようになります。 DSQQMFn B=1000000 QMFn B=1000000	203 ページ
QMF 照会および報告書用として使用される記憶域の量に限界を設定する場合、0 以外の限界が必要であれば、DSQSRSTG パラメーターを使用します。たとえば、限界として 1 000 000 バイトを設定する場合は、次のようになります。 DSQQMFn R=1000000	205 ページ

CICS ユーザーへの注

CICS は DSQSRSTG パラメーターを無視します。

表 25. パラメーターの受け渡し (続き)

作業	参照箇所
<p>報告書データ用の追加記憶域として一時記憶域 (予備ファイル) を使うためには、DSQSPILL パラメーターを使用する。たとえば、次のように入力します。</p> <p>DSQQMFn L=YES QMFn L=YES</p> <p>デフォルトは、CICS で実行する場合は、NO であり、それ以外の場合は、YES です。</p>	207 ページ
<p>CICS 予備ファイルの名前として DSQSnnnn (ここで、nnnn は CICS 端末 ID) 以外にも指定するには、DSQSSPQN パラメーターを使用します。たとえば、MYSPIILL という名前を指定する場合は、次のように入力します。</p> <p>QMFn DSQSSPQN=MYSPIILL</p>	214 ページ
<p>QMF が最初の報告書画面を表示する前に 100 以外の行数を QMF が検索できるようにするには、DSQSIROW パラメーターを使用する。たとえば、最初の画面を表示する前に QMF が 200 行検索できるようにするには、次のように入力する。</p> <p>DSQQMFn F=50 QMFn F=200</p>	214 ページ
<p>QMF のアクティビティーをトレース・データのログに記録するには (ユーザーのプロファイルが確立される前のアクティビティーも含む)、DSQSDEBUG パラメーターを使用する。たとえば、次のように入力します。</p> <p>DSQQMFn T=ALL QMFn T=ALL</p>	217 ページ
<p>CICS トレース・データ用として、一時データ (TD) キューではなく、一時記憶域 (TS) が必要であることを示すには、DSQSDBQT パラメーターを使用します。たとえば、次のように入力します。</p> <p>QMFn DSQSDBQT=TS</p>	218 ページ
<p>CICS トレース・データ用のキュー (一時記憶域か一時データかに関係なく) の名前として、DSQD 以外のものを指定するには、DSQSDBQN パラメーターを使用します。たとえば、一時記憶域キューに MYTRACE という名前を付ける場合は、次のように入力します。</p> <p>QMFn DSQSDBQN=MYTRACE</p>	219 ページ
<p>QMF 開始時の接続先のデータベース・ロケーションとしてデフォルト・ロケーション以外を指定するには、DSQSDBNM パラメーターを指定します。CICS の場合は、次のように指定します。</p> <p>DSQQMFn D=DBNAME QMFn D=DBNAME</p> <p>TSO の場合は、次のように指定します。</p> <p>DSQQMFn D=DBNAME</p>	220 ページ

開始手順のカスタマイズ

表 25. パラメーターの受け渡し (続き)

作業	参照箇所
ユーザー対話なしに QMF を実行するには (端末を使用する場合または使用しない場合)、 DSQSMODE パラメーターを使用し、さらに DSQSRUN パラメーターを使用して初期プロ シージャーを指定する。 DSQSDBNM パラメーターを使用して、必要とするデータベース・ ロケーションとの接続の確保を選択する必要がある場合もある。たとえば、Q というユーザ ー ID と STARTPROC という名前のプロシージャー例を使用して、なんらかの非対話式 QMF 作業を行う場合は、次のように入力します。 DSQQMFn M=B,D=DBNAME,I=STARTPROC.AUTHID QMFn M=B,D=DBNAME,I=STARTPROC.AUTHID	221 ページ
DSN 以外の DB2 サブシステムを使用するには、DSQSSUBS パラメーターを使用します。 DSQQMFn S=DSP	221 ページ

CICS ユーザーへの注

CICS は DSQSSUBS パラメーターを無視します。

QMF 開始時に初期プロシージャーを実行するには、DSQSRUN パラメーターを使用する。 222 ページ
たとえば、STARTPROC と呼ばれるプロシージャーを実行する場合は、次のように入力しま
す。

```
DSQQMFn I=STARTPROC
```

```
QMFn I=STARTPROC
```

QMF アプリケーション・プランの名前として、**QMF710** 以外のものを指定するには、 229 ページ
DSQSPLAN パラメーターを使用します。たとえば、次のように入力します。

```
DSQQMFn P=MYPLAN
```

CICS ユーザーへの注

CICS は DSQSPLAN パラメーターを無視します。

PRIMEID ではなく **TSOID** プロファイル・キーを使用するには、DSQSPRID パラメーター 230 ページ
を使用します。たとえば、次のように入力します。

```
DSQQMFn U=TSOID
```

表 25. パラメーターの受け渡し (続き)

作業	参照箇所
初期化プログラムを使用して、プログラム・パラメーターの値として、 QMF で設定されているデフォルト値以外の値を指定するには、 DSQSCMD パラメーターを使用する。たとえば、次のように入力します。 DSQQMFn DSQSCMD=NULL	231 ページ
DBCS データを非 DBCS 端末から印刷するには、 DSQSDBCS パラメーターを使用する。たとえば、次のように入力します。 DSQQMFn K=YES QMFn K=YES	230 ページ

報告書記憶域のカスタマイズおよび報告書パフォーマンス

ユーザーが、データベースからデータを取り出す **QMF** タスクを実行すると、そのデータは、仮想記憶域に保管されるデフォルトの報告書に戻されます。この節では、以下の事項のカスタマイズに役立つ **QMF** プログラム・パラメーターについて説明します。

- 報告書データ用として使用される記憶域の最大量
- 報告書用の仮想記憶域がいっぱいになったときに使用される予備記憶域
- **QMF** で最初の報告書画面が表示されるまでに検索されるデータの行数

DSQSBSTG (報告書データ用の記憶域の調整)

パラメーター名

DSQSBSTG

短縮形 B

有効な値

0 ~ 99 999 999 バイト

デフォルト

500 000 バイト (CICS の場合)、0 バイト (ISPF、TSO、SRPI、APPC、またはネイティブ OS/390 の場合)

DSQSBSTG の値は、報告書の生成に使用できる記憶域の上限 (バイト数) を **QMF** に示します。これは、0 から 99 999 999 までの値の範囲の正の整数です。DSQSBSTG に **QMF** が決定する最小値 (環境に応じて 15 ~ 32 KB) より小さい非ゼロ値を指定した場合は、この最小値まで拡大されます。

CICS を除くすべての場合で、DSQSBSTG の値が 0 であれば、このパラメーターは使用されずに、DSQSRSTG が使用されて記憶域が指定されます。ただし、DSQSBSTG と DSQSRSTG の両方を指定した場合は、DSQSBSTG が使用されます。

開始手順のカスタマイズ

DSQSRSTG の詳細については、205ページの『DSQSRSTG (アプリケーションに使用される予約記憶域の調整)』の説明を参照してください。

TSO、SRPI、APPC、ネイティブ OS/390、または ISPF の場合のデフォルトは 0 です。

CICS では、DSQSBSTG の値が 0 の場合は、QMF プログラムによって決定される記憶域の最少量が、報告書データ用として使用されます。値 0 を指定すると、大型の報告書を表示できる場合があります。CICS で実行される QMF トランザクションの大部分について、デフォルトをお勧めします。CICS の場合のデフォルトは 500 000 です。

各セッション用の仮想記憶域の正しい量の選択

QMF は複数のロード・モジュールで構成されています。メイン・モジュール (約 2.8 MB) は、16 MB より上で 31 ビット・モードで実行することができ、拡張ページ可能リンク・パック域 (EPLPA) に置くことができます。小さいサポート・モジュール (約 52 KB) は、16 MB より下で 24 ビット・モードで実行する必要があります。サポート・モジュールは、ページ可能リンク・パック域 (PLPA) に常駐することができます。EPLPA および PLPA を使用することによって、QMF を実行する OS/390 領域は、それぞれ QMF プログラムを共用することができます。

各 QMF 領域ごとに少なくとも 1.5 MB の仮想記憶域が必要です。追加記憶域があれば、QMF が仮想記憶域に保持できるデータ・レコードが増えるため、一般的にパフォーマンスの向上が得られます。

TSO のパフォーマンス・トレードオフ: DSQSPILL パラメーターを使用して、ユーザーに予備ファイル (仮想入出力域 (UNIT=SYSVIO) または他の DASD 記憶域) を提供することができます。予備ファイルがいっぱいになった場合でも、QMF は DSQSBSTG または DSQSRSTG パラメーターで指定されている量に応じて、仮想記憶域へのデータの取り出しを続行します。記憶域不足が生じた場合でも、ユーザーに通知が出されることはなく、QMF は報告書処理を完了することができます。したがって、十分なスペースを用意しておかなかった場合は、QMF が要求されたデータをすべて取り出すためにデータベースに戻る回数が多くなるため、たとえ予備ファイルを使用していても、パフォーマンスが低下することがあります。したがって、ユーザーが必要な QMF 作業を行うのに十分な大きさの仮想記憶域を確保できるようにしておくことをお勧めします。

また、管理プログラム出口ルーチンを使用して、データベースから取り出される行数を制限し、照会および報告書用として使用される仮想記憶域を減らすことを考慮する必要が生じる場合もあります。管理プログラム出口ルーチンの詳

細については、433ページの『第21章 管理プログラム出口ルーチンを使用した QMF リソースの制御』を参照してください。

各 CICS ユーザー用の仮想記憶域の正しい量の選択

照会を実行し報告書を生成する場合、一般的に、各 QMF トランザクションごとに少なくとも 1 MB の CICS 拡張動的記憶域が必要とされます。照会によっては、処理されるデータの量および報告書に対して使用される QMF のフォーマット設定オプションに応じて、使用する記憶域が増えるものも減るものもあります。24 KB を除けば、すべて拡張動的記憶域に割り振ることができます。たとえば、50 の QMF トランザクションをサポートするためには、少なくとも 50 MB の作業記憶域が必要であり、その内の 1.2 MB が 16 MB より下で動的記憶域に割り振られます。

QMF トランザクションは複数のロード・モジュールで構成されています。メイン・モジュール (約 2.8 MB) は、16 MB より上で 31 ビット・モードで実行することができ、EPLPA に置くことができます。小さいサポート・モジュール (約 52 KB) は、16 MB より下で 24 ビット・モードで実行する必要があります。このモジュールは、PLPA に常駐することができます。QMF トランザクションの実行を可能にする各 CICS 領域は、それぞれ QMF プログラムにアクセスできます。

QMF トランザクションが記憶域不足に陥った場合、CICS は記憶域が使用可能になるのを待ち、その後で該当の QMF トランザクションが処理を続行します。

CICS パフォーマンス・トレードオフ

DSQSPILL パラメーターを使用して、ユーザーに予備ファイルを提供することができます。予備ファイルがいっぱいになった場合は、記憶要求を満たす十分な大きさの記憶域が使用できる時まで、QMF トランザクションは延期されます。したがって、ユーザーが必要な QMF 作業を行うのに十分な大きさの仮想記憶域を確保できるようにしておくことをお勧めします。

また、管理プログラム出口ルーチンを使用して、データベースから取り出される行数を制限し、照会および報告書用として使用される仮想記憶域を減らすことを考慮する必要が生じる場合もあります。管理プログラム出口ルーチンの詳細については、433ページの『第21章 管理プログラム出口ルーチンを使用した QMF リソースの制御』を参照してください。

DSQSRSTG (アプリケーションに使用される予約記憶域の調整)

パラメーター名

DSQSRSTG

開始手順のカスタマイズ

短縮形 R

有効な値

0 ~ 99 999 999 バイト

デフォルト

0

DSQSRSTG パラメーターは、CICS を除くすべての場合にデフォルトです。

CICS ユーザーへの注

CICS は DSQSRSTG パラメーターを無視します。

ただし、報告書記憶域の指定の明示性を高める必要がある場合は、DSQSBSTG パラメーターを使用することができます。

このパラメーターの値は正の整数で、0 から 99 999 999 までの範囲の値であり、デフォルトは 0 です。値によって、他のプログラムおよび報告書の生成に影響を生じる可能性があります。

ユーザーによって 1 セッションで最初に報告書が生成された時点で、QMF は QMF アドレス・スペースで使用可能な記憶域の量を決定します。QMF 報告書のために確保される合計記憶域を入手するために使用される方法は、次のように、DSQSBSTG および DSQSRSTG の両方に依存します。

- DSQSBSTG が指定されていないか、または 0 に指定されている場合、QMF は DSQSRSTG の量を合計使用可能記憶域から引いて、QMF 報告書用として使用する最大量を決定します。残りの記憶域は、OS/390 システム・サービス、TSO コマンド、REXX、ISPF、および他の非 QMF ユーザー要求など、他のプログラムのために使用することができます。
- DSQSBSTG が指定されている場合は、その値を使用して、QMF 報告書用として獲得する記憶域を決定し、DSQSRSTG は使用されません。

DSQSRSTG の値が 0 の場合

DSQSBSTG と DSQSRSTG の両方も、値として 0 を指定することができます。この場合、DSQSRSTG パラメーターが使用され、他のシステム・サービス用には記憶域は予約されません。QMF セッション中に OS/390 システム・サービス、TSO コマンド、REXX、ISPF または他の非 QMF サービスのいずれも使用することのないユーザーの場合は、この値で十分と考えられます。ただし、ユーザーが OS/390 システム・サービスまたは TSO コマンドを使用し、DSQSRSTG=0 および DSQSBSTG=0 である場合は、失敗する危険があ

り、異常終了 (たとえば、abend878、abendb0a、abendb78、...) が起こる恐れがあります。その理由は、そのようなサービス用の記憶域を QMF がまったく予約していないためです。しかも、ほとんどの普通のユーザーも、インストール先定義の QMF コマンドを発行する際に、何気なく非 QMF プログラムを使用してしまう場合があります。このようなコマンドの場合は、一般的に、非 QMF プログラムを広範囲に使用する QMF アプリケーションによって実行されます。したがって、DSQSRSTG および DSQSBSTG の値の選択にあたっては、この点を考慮に入れることが必要です。

DSQSBSTG の値が小さく、DSQSRSTG の値が大きい場合

報告書処理用として最小記憶域を要求した場合は、ユーザーが報告書を取り扱う際のパフォーマンスに悪影響を生じる可能性があります。対応するデータ・オブジェクトに十分な記憶域がない場合は、QMF はデータの余分な行について予備ファイルを使用しなければなりません。予備ファイルで必要な入出力操作は、通常、パフォーマンスの低下を招きます。

DSQSPILL (エクストラ記憶域の獲得)

パラメーター名

DSQSPILL

短縮形 L

有効な値

YES または NO

デフォルト

CICS の場合は NO (予備ファイルを使用しない)、TSO の場合は YES (予備ファイルを使用する)

記憶域内に大量の報告書データがあると、他のプログラムの作動に影響を与える可能性があるため、QMF では予備ファイルの割り振りができるようにしてあります。

予備ファイルは、対話式 QMF セッションのパフォーマンスを向上させることができます。メモリー内のバッファーにデータを記憶することができるので、同一データの複数コピーのために、QMF がデータベースに戻る必要はありません。ユーザーが複数回表示して見る必要のあるデータを、データベースから複数回取り出す必要がありません。代わりに、予備ファイルを使用してそのデータを保管することができます。

CICS では、DSQSPILL パラメーターを YES に設定して、予備ファイルをアクティブ化します。

QMFn L=YES

開始手順のカスタマイズ

CICS 以外では、NO を指定しない限り、予備ファイルは自動的にアクティブ化されます。

```
DSQMFn L=NO
```

次の時点まで、データは予備ファイルに書き込まれます。

- RESET DATA コマンドを使用して、データ・オブジェクトをリセットする。
- 別の照会を実行することによってデータ・オブジェクトが置き換えられた。
- 照会が終了し (要求されたすべての行が検索され) データ・オブジェクトが完了した。
- 予備ファイルに定義された記憶域がいっぱいになった。予備ファイルは、CICS では DFHTEMP であり、CICS 以外では DSQSPILL である。
- CICS において、予備ファイルに取り出されたデータが 32 767 行 (CICS の一時記憶域キューに保持できる最大量) を超える。(各行に保持できるデータ量は 4K である。)

非 CICS ユーザーに関する予備ファイルの割り振り

ユーザーのログオン・プロシージャ、JCL、または CLIST の DD ステートメントによって、予備ファイルを割り振ることができます。この例をサンプル・プロシージャに示してあります。ここでは、DD ステートメント DSQSPILL によって予備ファイルを割り振っています。ステートメントは次のようになります。

```
//DSQSPILL DD DSN=&&SPILL,DISP=(NEW,DELETE),  
//          UNIT=SYSVIO,SPACE=(TRK,(1,9),RLSE),  
//          DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)
```

このステートメントは、次のことを行います。

- 該当のユーザーのセッションに固有の一時データ・セットとして、予備ファイルを割り振る。
- 予備ファイルを仮想入出力 (UNIT=SYSVIO) に割り振る。その代わりに、他の DASD 記憶域に予備ファイルを割り振ることもできます。
- 各ブロックごとに 1 レコードの、固定長レコードをもつ DSQSPILL ファイルを指定する。レコードは常に非ブロック化されていることが必要です。(1 ブロックは、OS/390 ページ (4096 バイト) のサイズです。)

このステートメントの SPACE オペランドでは、次のようにして、セッション中の予備ファイルの記憶域要件を最小化することができます。

- 予備ファイルが必要でないセッション中は、小さい 1 次エクステントでは、予備ファイルが保持する記憶域を単一のトラックに保留しておく。

- はるかに大きい 2 次エクステンツの使用は、予備ファイルが必要なときのみに限られる。
- RLSE キーワードによって、ユーザーのデータ・オブジェクトのリセット時に、QMF に 2 次エクステンツをすべて解放させる。これが行われるのは、たとえば、ユーザーが新規の報告書を開始する場合です。

CLIST 中で予備ファイルを割り振る場合は、次の例を使用します。

```
ATTR SPILL RECFM(F) LRECL(4096) BLKSIZE(4096)
ALLOC FILE(DSQSPILL) UNIT(SYSVIO) SPACE(1,19) RELEASE +
NEW DELETE USING(SPILL)
```

報告書が生成されるまで、ユーザーがこの割り振りを行うのを待った場合、予備ファイルはその報告書には使用されません。基礎 データ・オブジェクトが置き換えられた (たとえば、DISPLAY コマンドによって) 場合だけ、セッション中に予備ファイルが使用されます。

適切な予備ファイル・サイズの計算の説明については、『予備ファイルに必要なスペースの見積もり』を参照してください。

予備ファイルに必要なスペースの見積もり

予備ファイルに書き込まれたデータが設定限界を超えた (いっぱいまたは使用不可になった) 場合、QMF はデータを予備ファイルからは使用せずに、データベースから再度取り出し、仮想記憶域を使用してそのデータを保持します。CICS または TSO の DASD の記憶域を超過することがあり得ます。CICS では、予備ファイル用の一時記憶域は、各サイズがそれぞれ 4 KB のバッファ 32 767 個に制限されます。

QMF の記憶域要件を満たすためには、CICS 一時記憶域ファイル DFHTEMP または TSO DASD 記憶域に、並行 QMF ユーザーすべての個別予備ファイルを収容するだけでなく、他のすべてのトランザクションの補助一時記憶域に関する要件を満たすことができる大きさを十分に確保することが必要です。

以下の手順を使用して、個別予備ファイルに必要なスペースの量を計算します。その上で、並行 QMF ユーザーすべてを収容するために必要な個別予備ファイルの数に応じて、DFHTEMP を拡大します。

1. データ・オブジェクトに現れる可能性がある最大の表の 1 行の幅 (W) を計算する。このためには、フィールドの幅 (バイト数) を加算します (210 ページの表 26 を使用する)。計算例については、211 ページの表 27 を参照してください。
 - 1 つの個別表の行については、個々の行に入るデータとは無関係に、すべてが同じ幅とする。ただし、1 行の幅が 32,768 を超えることはできません。

開始手順のカスタマイズ

- 定義列は、予備ファイルに書き込まない。
2. **W が 4096 以下である場合は**、1 ページ当たりの行数 (R) を $R = 4096/W$ で計算し、その結果を小数点以下の切り捨てによって整数にする。
W が 4096 以下の場合は、QMF はページを増やさずに、できるだけ多くの行を 1 つのページに収容します。
 3. **W が 4096 より大きい場合は**、 $P = W/4096$ を使用して、1 行当たりのページ数 (P) を計算し、端数を切り上げてすぐ上の整数とします。
W が 4096 より大きいときは、QMF は 1 行の保持に必要な最小ページ数を使用し、列境界には無関係にページを増やすようにします。各行は、ページの先頭から始まります。
 4. **予備ファイルに必要なページ数の計算は**、W の値に応じて、次のように行います。
 - W が 4096 以下の場合は、表内の行数を R で割る ことによって、予備ファイルに必要なページ数を計算する。
 - W が 4096 を超えている場合は、表の行数に P を掛ける ことによって、予備ファイルに必要なページ数を計算する。

表 26. フィールド・タイプの長さ (予備ファイル・サイズの見積もりに使用)

フィールド・タイプ	フィールド長 (バイト数)
CHAR(n)	n+2
DATE	12
DECIMAL(n,m)	(n+1)/2+2, n は奇数。(n+2)/2+2, n は偶数
FLOAT(21)	10
FLOAT(53)	10
GRAPHIC(n)	n*2+2
INTEGER	6
SMALLINT	4
TIME	10
TIMESTAMP	28
VARCHAR(n)	n+4
LONG VARCHAR	(他のフィールドの長さによって異なる)
LONG VARGRAPHIC	(他のフィールドの長さによって異なる)
VARGRAPHIC(n)	n*2+4

行に LONG VARCHAR または LONG VARGRAPHIC フィールドが含まれる場合は、まず他のすべてのフィールドについてスペースを割り振ります。次

に、残りのスペースをフィールド数で割り、各 LONG VARCHAR または LONG VARCHARIC フィールドごとに、こうして求めた長さに切り捨てます。

表27 は、予備ファイルの計算の例を示したものです。

表 27. 予備ファイルの行の幅の計算例

行の内容	計算	幅に割り当てられる大きさ
2 - SMALLINT 列	$2 \times 4 =$	8 バイト
1 つの INTEGER 列		6 バイト
1 - DECIMAL(3,2) 列	$(3+1)/2+2 =$	4 バイト
1 つの DECIMAL(6,0) 列	$(6+2)/2+2 =$	6 バイト
1 つの FLOAT 列		10 バイト
1 つの CHAR(10) 列	$10 + 2 =$	12 バイト
1 つの VARCHAR(16) 列	$16 + 4 =$	20 バイト
行の合計幅		59 バイト

以下の計算の例では、予備ファイルのスペースを計算するための 2 つの方法を示しています。

R=4096/540 = 7 複数行 / バッファー の場合 :

$$\frac{600 \text{ 000 行}}{7} * \frac{1 \text{ トラック}}{10 \text{ ブロック}} * \frac{1 \text{ シリンダー}}{15 \text{ トラック}} = 571 \text{ シリンダー}$$

R=6000、2 バッファー / 行 の場合 :

$$6000 \text{ 行} * \frac{2 \text{ ブロック}}{10 \text{ ブロック}} / \text{行} * \frac{1 \text{ トラック}}{10 \text{ ブロック}} * \frac{1 \text{ シリンダー}}{15 \text{ トラック}} = 800 \text{ シリンダー}$$

非対話式 QMF セッションでの予備ファイルの使用

予備ファイルが対話式 QMF セッションでのパフォーマンスの向上に最も役立つのは、DSQSMODE パラメーターが I に設定されている場合です。QMF を非対話式に実行する (DSQSMODE パラメーターが B に設定されている) 場合は、報告書の生成にデータの受け渡しが複数回必要なときも、予備ファイルの使用によってパフォーマンスを向上させることができます。RUN QUERY コマンドに続いて SAVE DATA コマンドを実行する場合など、データ・オブジェクトを完了するために予備ファイルが必要になる場合もあります。

QMF をバッチで実行する場合は、行われる作業に基づいて、QMF プログラム・パラメーター DSQPILL(YES/NO) を設定する必要があります。当該ジョ

開始手順のカスタマイズ

ブで印刷用に大きいデータ・オブジェクトを生成する場合は、予備ファイルを割り振ると、パフォーマンスに悪影響をもたらす可能性があります。バッチでの実行時には、ほとんどの場合、DSQSPILL=NO を選択するのが最適の策です。

複数回にわたるデータの受け渡しが必要になるのは、次の場合です。

- 同じデータに関して、異なる書式で複数の報告書を印刷する必要がある場合。
- 報告書で PCT、CPCT、TCPCT、または TPCT 編集コードを使用する場合。
- 報告書の幅が印刷幅よりも大きいため、QMF がページの分割を行う必要がある場合。

非対話式 QMF セッションの詳細については、221ページの『DSQSMODE (対話式または非対話式 QMF セッションの指定)』を参照してください。

QMF 解説書 では、報告書のフォーマット設定に使用する QMF 書式のそれぞれについて説明し、書式の使用法の例が示してあります。

予備ファイルの問題の解決

ユーザーが使用できるように予備ファイルを作成すると、ユーザーが抱える記憶域の問題は解決できますが、その他の問題が生じる可能性があります。

DASD スペースに関する問題に直面したり、他のユーザーにとっての問題を生じたりする場合があります。

DASD ボリューム上のスペースが少な過ぎる問題: 多くのユーザーが QMF に関して同一のログオン・プロシージャーを使用していて、予備ファイルの問題に直面し、しかもこの共通のログオン・プロシージャーが予備ファイルをすべて特定の 1 つの DASD ボリュームに割り振っている場合は、このボリューム上のスペースが少な過ぎることが、ユーザーが直面している問題の原因になっていることがあります。そのような場合は、このログオン・プロシージャー中の予備ファイルの DD ステートメントを変更することによって、問題を解決できることがあります。この新規の DD ステートメントでは、特定のボリュームに対する現行の参照に代えて、非特定のボリューム参照を行うことになりません。

他のユーザーにとって予備ファイルの問題を生じる場合: あるユーザーが直面する予備ファイルの問題は、その予備ファイルの 2 次割り振りを増やすことによって解決できる場合があります。しかし、そのようにした場合は、他のユーザーにとっての予備ファイルの問題を生じることになる可能性があります。し

たがって、2 次割り振りを大幅に増やす必要がある場合は、該当のユーザーの予備ファイルを、他のユーザーの予備ファイルには使用しないボリュームに移動することを考えてください。

ユーザーは、それとは気付かずに、他のユーザーにとっての予備ファイルの問題の原因となっている場合があります。たとえば、大きい表の最下部までスクロールして、予備ファイルのオーバーフローが生じても、不完全なデータ条件を引き起こすようなことは何もしていない場合があります。これに該当するのは、その表が最初に表示された時点から別の表に置き換わる時点までの間に、ユーザーが特定のタイプのコマンドを発行しなかった場合です。この間は、他のユーザーが非常に必要としているスペースを、そのユーザーが不必要に保留していることになる可能性があります。

パフォーマンスの問題: 条件付きフォーマット設定または列定義 (これには、REXX を使用し、パフォーマンスに関する追加の考慮事項を伴う) を使用しない場合は、データベース内のデータに対するアクセスの結果がパフォーマンスを左右します。

最初にデータを取り出した後、QMF で使用できる記憶域が十分にある場合は、QMF はデータベースに再アクセスして、2 度目の行の入手をはかる必要はありません。

メモリーに制約があり、DSQSPILL ファイルが定義されている場合は、データを後で取り出すことができるように DSQSPILL にデータを書き込む時間が、処理時間の一部として組み込まれます。

パフォーマンスは次のようないくつかの事項に影響されます。

- DSQSIROW (取り出される初期行数) の値。主に影響が及ぶのは、報告書の初期表示だけです。
- データの受け渡しを複数回必要とすることを行うかどうか。(PCT など、特定の取扱コードでは、最初の報告書画面の表示前に、データをすべて読み取っておくことが必要です。) 主に影響が及ぶのは、報告書の初期表示だけです。
- 1 行のデータの保持に必要なメモリーの量。この影響は通常わずかです。
- 複数のデータのパスが必要とされる場合に、そのデータを再びデータベースから取り出すか (メモリーと DSQSPILL にすべてのデータが収められているとは限らない)、メモリーと DSQSPILL から取り出すか、または仮想メモリーだけから取り出すかどうか。
- 後方スクロールか前方スクロールか。FORWARD コマンドを連続するのが、パフォーマンス上は通常最適です。BACKWARD コマンドでは、応答セット

開始手順のカスタマイズ

の開始点でやり直しを必要とする場合があります。ただしこれらは、メモリーの量、どこまで後方スクロールするのか、報告書の複雑さ、およびその他の要因によって異なります。

メモリーが小さく、DSQSPILL 割り振りが不十分なときに応答セットが非常に大きいと、BACKWARD コマンドが使用されるたびに、応答セット全体が 1 行目から新規の現在行まで読み取られることがあります。

最高のパフォーマンスが得られるのは、データをすべて保持できる十分な大きさのメモリーがあり、DSQSPILL を使用しない場合です。

データの処理に使用されるリソースの合計量の節減にはなりません、最初の表示前に応答セット全体を仮想メモリーに収めてしまうことができれば (DSQSIROW が大きければ)、データベース・ロックは解除され、表示報告書全体のスクロールは最高速で行われます。これによって、最初の報告書画面の表示は遅くなります。ロックが解除されると、他のユーザーにとってパフォーマンスが向上するという効果を生じる場合があります。

DSQSSPQN (CICS 予備記憶域の名前の指定)

パラメーター名

DSQSSPQN

短縮形 (短縮形なし)

有効な値

キューに関する CICS 命名規則に従う任意の名前

デフォルト

DSQSnnnn (nnnn は CICS 端末 ID)

予備ファイルの使用を選択した場合は、CICS の一時記憶域キューの名前を指定して、QMF の予備データ用として使用することもできます。たとえば、MYDATA という名前を指定する場合は、次のようになります。

```
QMFn DSQSSPQN=MYDATA
```

非対話式 QMF セッションを CICS アプリケーション内から開始する場合に、CICS 端末 ID を指定しないことにすると、DSQSSPQN パラメーターをコーディングする必要があります。DSQSSPQN の値を明示的に指定する必要があり、もし指定しないと、QMF は開始されません。

DSQSIROW (表示用に検索される報告書の行数の制御)

パラメーター名

DSQSIROW

短縮形 F

有効な値

0 から 99 999 999 までの範囲にある任意の数

デフォルト

報告書の最初の画面の表示前に取り出される最小行数値である 100 行

DSQSIROW を使用して、報告書の最初の画面がユーザーに表示される前に、QMF がデータ・オブジェクト内に取り出す最大行数を指定します。

DSQSIROW が適用されるのは、次のようにして作成された新規のデータ・オブジェクトの初期ロードのみに限られます。

- SQL SELECT ステートメントを使用する照会の実行
- QMF DISPLAY コマンドによるデータベース表の表示

このパラメーターのための適切な値を決めるためには、209ページの『予備ファイルに必要なスペースの見積もり』のアルゴリズムのステップ 1 を使用して、ユーザーが照会する可能性の高い最大の表の行ブロックのサイズを見積もります。ブロックは、4096 バイトのバッファー 1 つに収まる行数です。

各行ブロックが取り出された後で、QMF は取り出された行の合計数を DSQSIROW の値と比較して、データの最初の画面を表示するかどうかを決定します。たとえば、インストール先のブロックが 62 行で、DSQSIROW を 50 に設定したものとします。QMF は 62 行のデータを検索し、62 と 50 とを比較した段階で行の検索を停止し、最初のデータ画面を表示します。

報告書フォーマット設定オプションの中には、パーセント (%) 取扱コードや ACROSS 報告書などのように、QMF が最初の画面を表示する前に、すべてのデータの取り出しを必要とするものもあります。QMF は、このような場合、DSQSIROW の値を無視します。これらのフォーマット設定オプションの詳細については、*QMF 解説書* を参照してください。

DSQSIROW の値が小さい場合のパフォーマンス

使用する DSQSIROW の値が小さ過ぎる場合、QMF は、データの最初の画面の表示前に、データ・オブジェクトを完了できないことがあります。データ・オブジェクトが完了していない場合は、データに 共用 ロックが掛かることになり、他のユーザーによる該当のデータの更新の試みが妨げられる可能性があります。DB2 では、EDM プールの保守を行って、リクエスターの要求に応じます。データ・オブジェクトが未完了の間、リクエスターは EDM リソースに関して他のすべてのリクエスターと競合することになります。

QMF 制御表またはシステム・カタログの一部がロックされた場合は、多くのユーザーに影響が及ぶ可能性があります。ロックの解除は、次のいずれかの方法で行うことができます。

開始手順のカスタマイズ

- BOTTOM コマンドを使用して、残りの行をデータ・オブジェクト内に取り込んでから、ロックを解除する。
- RESET DATA コマンドを使用して、ロックを解除し、要求されたデータがすべて取り込まれたかどうかに関係なく、データ・オブジェクトをクリアする。
- いずれかの SAVE コマンド (たとえば、SAVE DATA または SAVE FORM) を使用して、残りの行をデータ・オブジェクト内に取り込み、保管してから、ロックを解除する。

データ・オブジェクトを完了させるコマンドのリストについては、538ページの『パフォーマンスを向上させるためのデータ・オブジェクトのリセット』を参照してください。

非対話式セッションで (DSQSMODE パラメーターが B に設定されているとき) 最高のパフォーマンスを得るには、QMF によるデータの取り出しおよびフォーマット設定の間、オープン状態の読み取りロックの数を最小限に抑えなければならない場合を除いて、DSQSIROW の値としてゼロを使用します。非対話式 QMF セッションの詳細については、221ページの『DSQSMODE (対話式または非対話式 QMF セッションの指定)』を参照してください。

DSQSIROW は、画面に QMF が表示する行数を制限する目的で使用しないでください。小さい値を指定することはできますが、QMF は、非対話式セッションで、画面表示を満たすだけの十分な行数を検索します。

DSQSIROW 値が大きい場合のパフォーマンス

DSQSIROW に使用する値が大きすぎると、QMF は、最初のデータ画面の表示に長時間要することがあります。DSQSIROW を DSQSBSTG パラメーターの設定値より高い値に設定した場合は、たとえば、使用可能な記憶域が不十分で、ユーザーの要求には応じきれない旨を示すメッセージが、QMF によって表示されることがあります。

CICS では、該当の領域の記憶域がいっぱいの場合、QMF は仮想記憶域が使用可能になるのを待ってから、データベースに関する行の取り込みを完了します。TSO およびバッチでは、記憶域がいっぱいの場合、QMF は行の取り込みを停止するか、終了します。DSQSBSTG および DSQSIROW の値を計画する際には、CICS では、記憶域が使用可能になるのを待つために QMF がタイムアウトになる可能性があることに注意しなければなりません。

セッション開始時点での QMF アクティビティのトレース

QMF にはトレース機能が用意されているので、ユーザー・アクティビティのトレース、およびユーザーのセッション中に発生する可能性のあるエラーのトレースを行う場合に役立ちます。この節で説明するプログラム・パラメーターは、次の事項の制御に役立ちます。

- QMF アクティビティのトレースの詳細のレベル (ユーザーのプロファイルが確立される前のアクティビティを含む)
- トレース・データの保管個所

DSQSDEBUG (トレース詳細レベルの設定)

パラメーター名

DSQSDEBUG

短縮形 T

有効な値

ALL または NONE

デフォルト

NONE (トレース・データなし)

DSQSDEBUG を使用して、QMF アクティビティのトレースを行う必要がある詳細のレベルを指定します。NONE を指定した場合は、ALL という保管値を指定してプロファイルをロードしない限り、トレースは行われません。ALL を指定した場合は、ALL によってプロファイル値が上書きされ、ALL のままになります。

このパラメーターを用いて設定したトレースは、ユーザーが SET PROFILE (TRACE=value コマンドを発行して変更しない限り、また NONE の場合は、プロファイルがロードされるまで、有効です。有効なトレースの値の詳細については、546ページの『トレース出力の適切な詳細レベルの設定』を参照してください。

プログラム初期化エラー、およびユーザーのプロファイルが確立される前に発生する可能性があるその他のエラーを含めて、最も詳細なレベルまで QMF アクティビティをトレースする必要がある場合は、次のように DSQSDEBUG に ALL を設定します。

```
DSQQMFn T=ALL
```

```
QMFn T=ALL
```

CICS の場合で、値 ALL を使用する際に、選択する記憶域キューのタイプは、トレース出力を保持できる十分な大きさのものを確保することが必要で

開始手順のカスタマイズ

す。『DSQSDBQT (CICS のトレース・データ用記憶域のタイプの指定)』に、QMF の開始時に CICS キュー・タイプを指定する方法を説明してあります。

DSQSDEBUG に NONE を設定する場合は、トレース出力の詳細レベルは、QMF セッションの実行が次のように対話式であるか、非対話式であるかによって異なります。

- 対話式セッションと非対話式セッションのいずれの場合も、ユーザーのプロファイルが確立される前は、システム・エラーのトレースだけが初期化中に行われる。この初期トレースをオフにする唯一の方法は、トレース・データ用の記憶域を割り振らない、または定義しないことです。
- 非対話式セッションの場合、最も詳細なレベルまで、メッセージおよびコマンドがすべてトレースされる。

221ページの『DSQSMODE (対話式または非対話式 QMF セッションの指定)』に、対話式および非対話式セッションについて詳細に説明してあります。

QMF の開始後、SET PROFILE (TRACE=NONE コマンドを使用して、トレースをオフにすることができます。さらに、このコマンドを使用すれば、異なる QMF 機能を表すさまざまな値で NONE を置き換えることによって、トレースの詳細レベルをさらに特定して設定することもできます。詳細については、544ページの『QMF トレース機能の使用』を参照してください。

DSQSDBQT (CICS のトレース・データ用記憶域のタイプの指定)

パラメーター名

DSQSDBQT

短縮形 (短縮形なし)

有効な値

TD または TS

デフォルト

TD (一時データ・キュー)

DSQSDBQT を使用して、トレース・データ用として使用したい CICS 記憶域のタイプを指定します。トレース用として CICS の補助一時記憶域キューを使用する場合は、次のように値 TS を指定します。

```
QMFn DSQSDBQT=TS
```

メッセージ・レベルのトレースの場合は、一時記憶域 (TS) の使用をお勧めします。それ以外のタイプのトレース (ALL など) については、トレース出力が 32 767 行分のデータ (CICS の一時記憶域キューの限界) を超えると考えられる場合は、一時データ・キューの使用を考慮します。

DSQD という名前の一時データ・キューが、QMF のインストール中にシステムによって事前定義されています。DSQSDBQN パラメーターを使用して、一時データ・キューに DSQD 以外の名前を付ける場合は、そのキューを初めて使用する前に、CICS に対してそのキューを事前定義しておく必要があります。

QMF トレースをさらに詳細に指定する場合とトレース・データの表示に関する詳細については、544ページの『QMF トレース機能の使用』を参照してください。

DSQSDBQN (CICS のトレース・データ用記憶域の名前の指定)

パラメーター名

DSQSDBQN

短縮形 (短縮形なし)

有効な値

キューに関する CICS 命名規則に従う任意の名前

デフォルト

DSQD

DSQSDBQN では、トレース・データを保持する一時データ・キューまたは一時記憶域キューの名前を指定します。DSQD という名前の一時データ・キューが、CICS DCT でシステムによって事前定義されています。

DSQSDBQT で一時データ・キューを指定し、そのキューに DSQD 以外の名前を付けたい場合は、そのキューがまだ使用可能でなければ、CICS DCT でそのキューを定義します。

そのキューの名前は、DSQSDBQT で指定したキューのタイプに関する CICS の指定に適合する必要があります。TD キュー名は、1 ～ 4 文字の範囲の名前です。TS キュー名は、1 ～ 8 文字の範囲の名前です。

CICS に対して一時記憶域キューを事前定義しておく必要はありません。たとえば、次のステートメントは、QMF セッションに関してトレース・データを保持するために、MYTRACE という名前の一時記憶域キューを動的に割り振ります。

```
QMFn DSQSDBQN=MYTRACE,DSQSDBQT=TS
```

QMF はキュー内の単一のトレース項目に関して CICS ENQ および DEQ コマンドを発行するので、単一のキューを複数のユーザーで使用することができます。トレース・データがキューに書き込まれた後で表示して見る方法については、549ページの『QMF トレース・データを表示する』を参照してください。

セッション中の初期アクティビティーの制御

この節では、次のような初期 QMF アクティビティーの制御に役立つプログラム・パラメーターについて説明します。

- データベースとの接続のロケーションの指定
- 非対話式セッションの開始
- TSO で使用される DB2 サブシステムの命名
- 初期プロシージャの実行。これは、プロシージャに定義されている事前決定の作業量を実行した後に QMF を終了する。
- QMF アプリケーション・プランの命名
- TSO プロファイル・キーの指定

DSQSDBNM (QMF 開始時の接続先のロケーションの指定)

パラメーター名

DSQSDBNM

短縮形 D

有効な値

任意の有効なデータベース名

デフォルト

CICS の場合は、現在 CICS で使用されているデフォルトのデータベース。それ以外の場合は、使用されているサブシステムのデフォルトのデータベース。

DSQSDBNM を使用すれば、QMF セッションに関する初期接続先のロケーションを指定することができます。このロケーションは、リモート・データベースでも構いません。DSQSDBNM はすべての操作環境で指定できます。

リモート作業単位のための設定を行う場合: DSQSDBNM の値の最大長 (文字数) は、リモート作業単位接続を開始するアプリケーション・リクエストのタイプによって異なります。各リクエスト・タイプに応じた長さを表28 に示しています。

表 28. リクエストのタイプに基づいた DSQSDBNM の値の最大長

リクエストのタイプ	最大長
UDB (OS/390 版)	16
DB2 (VM 版) および DB2 (VSE 版)	18

DSQSMODE (対話式または非対話式 QMF セッションの指定)

パラメーター名

DSQSMODE

短縮形 M

有効な値

B (非対話式) または I (対話式)

デフォルト

I (呼び出し可能インターフェースを介して開始される場合は B)

ユーザーが行う必要のある照会および報告書作成のタスクによっては、QMF との対話を必要としない場合があります。たとえば、販売担当員の場合は、数日ごとに同じ QMF プロシージャを使用して、アカウント状況に関する一連の表を照会します。データは変わっても、データにアクセスするのに必要なプロシージャとタスクはいつも同じで変わりません。

QMF プログラム・パラメーターの DSQSMODE を使用すれば、QMF 作業を行うのに非対話式セッションを開始することによって、リソースおよび時間の節減をはかることができます。そうすれば、そのトランザクションが実行されている間、端末は空いているので他の作業を行うことができます。

非対話式セッションを開始する場合は、次のように値 B を使用します。

```
DSQQMFn M=B,I=STARTPROC
```

非対話式セッションでは QMF パネルは表示されないため、必要な QMF 作業を行う初期プロシージャを実行して、そのプログラムを終了する場合は、DSQSRUN (I) パラメーターを使用します。このパラメーターについては、222 ページの『DSQSRUN (QMF の開始時に実行するプロシージャの命名)』で詳細に説明します。

さらに、デフォルトのデータベース・ロケーションを使用したくない場合は、DSQSDBNM パラメーターを使用して、データベース接続に関する ID およびパスワードを指定します。

DSQSSUBS (QMF で使用される DB2 サブシステムの命名)**CICS ユーザーへの注**

CICS は DSQSSUBS パラメーターを無視します。

開始手順のカスタマイズ

パラメーター名

DSQSSUBS

短縮形 S

有効な値

QMF で使用する DB2 サブシステムの名前

デフォルト

DSN

このパラメーターの値は、QMF で使用する DB2 サブシステムの名前です。DSN (デフォルト) 以外の名前にする場合は、このパラメーターによってその名前を QMF に渡す 必要 があります。

名前は SYS1.PARMLIB のメンバー IEFSSNxx の中で見つかります。

データ共用環境での考慮事項

データ共用環境では、「総称」接続名として DB2 サブシステム名の代わりに、DB2 グループ接続機構名を使用することができます。

DSQSRUN (QMF の開始時に実行するプロシーチャーの命名)

パラメーター名

DSQSRUN

短縮形 I

有効な値

任意の有効なプロシーチャー名 (QMF 解説書 を参照)

デフォルト

初期プロシーチャーは実行されない

QMF の開始後すぐに実行させる QMF プロシーチャーの名前を渡すには、DSQSRUN パラメーターを使用します。非対話式セッションでは、行う必要のある QMF 作業を行ってから、そのプログラムを終了する場合に、このプロシーチャーを使用します。

たとえば、STARTPROC という名前の初期プロシーチャーを実行する場合は、次のように入力します。

```
DSQQMFn I=STARTPROC
```


他のユーザーも同じプロシージャを使用して QMF を開始する場合は、プロシージャの名前をその所有者の SQL 許可 ID で修飾します。たとえば、ユーザー JONES がプロシージャ STARTPROC の所有者である場合は、次のように入力します。

```
DSQQMFn I=JONES.STARTPROC
```

初期プロシージャの名前を渡すと、QMF は RUN PROC コマンドを発行し、そのコマンドが指定したプロシージャを実行します。

重要: QMF では、ユーザー ID およびプロシージャの構文でブランクを使用することは許されません。たとえば、次の例は QMF では認識されません。

```
DSQQMFn I=JONES. STARTPROC
```

途中で組み込みブランクがあるプロシージャ名を使用するには、次のように、名前を引用符で囲む必要があります。

```
DSQQMFn I=JONES.'START PROC'
```

DSQSRUN を使用すると、次の場合に役立ちます。

- 非対話式 QMF 作業を自動化して、通常は対話式実行で使用されるリソースを保存できるようにする場合。
- ユーザーが事前定義プロシージャの境界内で対話式 QMF 作業を実行してから、そのプロシージャで指定されている作業を終了することができるようにする場合。

初期プロシージャの非対話式実行

リソースを保存するには、DSQSMODE パラメーターの値として B を使用し、DSQSRUN パラメーターを使用してプロシージャに名前を付けることによって、プロシージャを非対話式に実行することができます。たとえば、毎月曜日の朝、在庫状況報告書を作成する必要があるものとします。毎日曜日の夜、INVENTORY と呼ばれる表の同じ列からデータを取り出す照会を実行する必要があります。その場合の照会は、次の照会に似たものとなります。この例では、この照会を INVENTORY_QUERY と呼びます。

```
SELECT * FROM INVENTORY
WHERE STOCK < 20
```

この照会を実行して状況報告書を印刷するのに使用するプロシージャは、次に挙げる CICS および TSO のプロシージャのようなものになります。例では、この QMF プロシージャを INVENTORY_PROC を呼ぶことにします。

開始手順のカスタマイズ

```
RUN QUERY INVENTORY_QUERY  
PRINT REPORT  
EXIT
```

プロシージャに EXIT コマンドが組み込まれるのは、QMF を非対話式に実行する場合は、QMF セッションを終了させるユーザーが存在しないからです。EXIT によって、QMF セッションが終了し、QMF が保留しているリソースが解放されます。非対話式に実行される初期プロシージャでは、必ず EXIT コマンドを使用してください。

報告書の作成に伴うタスクは変わらないため (変わるのはデータだけ)、DSQSRUN パラメーターを使用して、勤務時間外の日曜日の夜、次のように表 INVENTORY を照会し、報告書を印刷することができます。

```
QMFn I=INVENTORY_PROC,M=B
```

初期プロシージャを使用した対話式 QMF 作業の実行

対話式 QMF セッションで初期プロシージャを使用し、エンド・ユーザーのデータ・アクセス・タスクを事前定義して、必要なデータだけをエンド・ユーザーが容易にアクセスできるようにすることができます。たとえば、ある QMF エンド・ユーザーが、毎月曜日の朝、在庫状況報告書を作成する責任を負っているとします。ユーザーは在庫量が少ないことを示す値は分かっていますが、状況報告書の作成方法を正確には知らないことがあります。このような場合には、照会に変数を入れておけば、エンド・ユーザーは、少ない在庫量を示す値だけを入力すれば済むことになります。この照会は INVENTORY_QUERY と呼ばれます。

```
SELECT * FROM INVENTORY  
WHERE STOCK < &LOWSTOCK
```

印刷する前にユーザーがデータの表示を必要とすることがあるため、INVENTORY_PROC プロシージャには EXIT コマンドを組み込まなくても構いません。

```
RUN QUERY INVENTORY_QUERY
```

このような場合、DSQSMODE パラメーターを指定しないで DSQSRUN パラメーターを使用すれば、エンド・ユーザーに代わって対話式セッションを開始することができます。

```
QMFn I=INVENTORY_PROC
```

プロシージャ INVENTORY_PROC がプロンプトを出して、エンド・ユーザーに &LOWSTOCK 変数値の入力を指示します。初期プロシージャで変数をどの

ように使用するかについての例が追加されて、『初期プロシージャーに変数値を渡す』に記載してあります。 *QMF* 解説書 では、変数について詳細に説明しています。

ユーザーが値を入力すると、*QMF* はただちに報告書を表示します。そこでユーザーは報告書を見て、*QMF PRINT* コマンドを発行しそれを印刷することができます。

対話式セッションの場合は、報告書を見終わったらコマンド行に *EXIT* を入力するようにユーザーに指示してください。*EXIT* コマンドが発行されるまで、初期プロシージャーは繰り返し実行されます。したがって、報告書パネルで終了機能キーを押すと、初期プロシージャーが再実行されることになり、*QMF* ホーム・パネルが表示されることにはなりません。

さらに、*DSQSRUN* パラメーターを使用する際は、グローバル変数 *DSQEC_RERUN_IPROC* を必ず 0 に設定し、現行オブジェクトが *QMF* ホーム・パネルでないことを確認する必要があります。*QMF* アプリケーション開発の手引き には、このグローバル変数に関する詳細、ならびにユーザーが事定義のプロシージャーおよびアプリケーションに指定されている *QMF* アクティビティを実行する場合に役立つプロシージャーの作成方法に関する説明が記載されています。

初期プロシージャーに変数値を渡す

初期プロシージャーの名前を *DSQSRUN* パラメーターに指定するときは、そのプロシージャーに含まれる変数の値も与えることができます。*DSQSRUN* パラメーターには、プロシージャー名の直後に続けて、1 つまたは複数の変数とその値を指定することができます。

DSQSRUN に変数を指定するときは、次の規則に従ってください。

- この節の例に示されているように、変数パラメーター・リストは、前後を括弧で囲む。
- 変数名の前にアンパーサンドを指定し、ストリングを *variable_name= 値* のフォーマットにする。
- プロシージャー名と変数パラメーター・リストを合わせた文字数の合計が、必ず 98 文字以下になるようにする。
- 変数パラメーターの指定項目間は、1 つのコンマ、1 つまたは複数のブラケット、あるいは 1 つのコンマと複数のブラケットの組み合わせで分離する。

226ページの表29 に、環境および各環境で変数を使用する場合に必要なアンパーサンドの数をリストしてあります。

開始手順のカスタマイズ

表 29. プログラム変数の前に付ける必要があるアンバーサンドの数

環境	追加アンバー サンドの数	例
CICS	0	&variable= 値
SRPI	0	&variable= 値
APPC	0	&variable= 値
TSO (ISPF なし)	0	&variable= 値
TSO (ISPF 付き)	1	&&variable= 値
TSO (CLIST 使用 ISPF なし)	2	&&&variable= 値
TSO (CLIST 使用 ISPF 付き)	3	&&&&variable= 値

初期プロシージャの名前を指定すると、*QMF* はその初期プロシージャを実行する `RUN PROC` コマンドを発行します。プロシージャ内で変数を使用する場合、そのような変数に指定する値は、`RUN` コマンドで変数を渡す場合に使用する構文に適合する必要があります。この構文については、*QMF* 解説書を参照してください。

たとえば、組織内の従業員について、2 つの情報が必要になる頻度が高いものとしします。必要な情報の 1 つは従業員の名前ですが、もう 1 つはそのつど変わるものとしします。この場合、`NAME` 列を含み、他の列については変数を使用する照会を定義することができます。図 38 に照会およびプロシージャの例を示します。またこの図は、`DSQSRUN` パラメーター入力時に変数の値を渡す方法、および *QMF* が発行する `RUN PROC` コマンドも示しています。

照会 (名前は `JONES.QUERY2`)

```
SELECT NAME, &COL  
FROM Q.STAFF
```

プロシージャ (名前は `JONES.PROC2`)

```
RUN QUERY JONES.QUERY2 (&&COL=&COL
```

`DSQSRUN` パラメーター

```
QMFn I=JONES.PROC2(&COL=YEARS)
```

結果の `RUN` コマンド

```
RUN PROC JONES.PROC2 (&COL=YEARS)
```

図 38. `DSQSRUN` を使用した *QMF* 列名の受け渡し

図39 にも類似例を示してありますが、こちらの例では、1つの列名をプロシージャに渡すのではなく、複数の列名を渡す（従業員の名前、所属部門、および従業員の給与を戻す）ことができます。

照会 (名前は JONES.QUERY3)

```
SELECT &COLS
FROM Q.STAFF
```

プロシージャ (名前は JONES.PROC3)

```
RUN QUERY JONES.QUERY3 (&&COLS=&COLS
```

DSQSRUN パラメーター

```
QMFn I=JONES.PROC3(&COLS=((DEPT,NAME, SALARY))
```

結果の RUN コマンド

```
RUN PROC JONES.PROC3(&COLS=((DEPT,NAME,SALARY)))
```

図 39. DSQSRUN を使用した複数の QMF 列名の受け渡し

次の 4 つの例は、照会の WHERE キーワードの後ろに通常指定する情報の渡し方を示したものです。(WHERE キーワードの詳細については、QMF 解説書を参照してください。)

これらの例には文字ストリングが含まれていて、それらについては、RUN PROC コマンド処理時に QMF がどのように値を評価するかによって、特殊な構文が必要です。図に示してあるように、特殊文字 (コンマ、ブランク、括弧、引用符、アポストロフィまたは単一引用符、および等号) もこのストリングの中に入れることができます。

たとえば、組織内のすべての管理者の名前と従業員番号を知りたい場合は、228 ページの図40 に示すような照会を実行することができます。DSQSRUN パラメーターに文字ストリング MGR を渡すときは、必ずその値を単一引用符で囲みます。

開始手順のカスタマイズ

照会 (名前は JONES.QUERY4)

```
SELECT JOB, NAME, ID
FROM Q.STAFF
WHERE JOB=&JOB
```

プロシージャ (名前は JONES.PROC4)

```
RUN QUERY JONES.QUERY4 (&&JOB=&JOB
```

DSQSRUN パラメーター

```
QMFn I=JONES.PROC4(&JOB='MGR')
```

結果の RUN コマンド

```
RUN PROC JONES.PROC4 (&JOB='MGR')
```

図 40. DSQSRUN を使用した単一引用符内ストリングの渡し方

図41 に示すのは、コンマを含む変数値を渡す方法です。値 SAN JOSE, CA は、コンマを含んでいるため、単一引用符で囲まなければなりません。

照会 (名前は JONES.QUERY5)

```
SELECT *
FROM Q.APPLICANT
WHERE ADDRESS=&CITY
```

プロシージャ (名前は JONES.PROC5)

```
RUN QUERY JONES.QUERY5 (&&CITY=&CITY
```

DSQSRUN パラメーター

```
QMFn I=JONES.PROC5(&CITY='sAN JOSE,CA')
```

結果の RUN コマンド

```
RUN PROC JONES.PROC5 (&CITY='sAN JOSE,CA')
```

図 41. DSQSRUN を使用した文字ストリング内のコンマの受け渡し

229ページの図42 は、単一引用符を含む (たとえば、名前の中のアポストロフィ) 変数値の渡し方を示しています。DSQSRUN パラメーターでこのような値を渡すときは、必ずその値を単一引用符で囲み、アポストロフィに代えて、1 つではなく 2 つの単一引用符を使用します。

照会 (名前は JONES.QUERY6)

```
SELECT *
FROM Q.STAFF
WHERE NAME=&NAME
```

プロシージャ (名前は JONES.PROC6)

```
RUN QUERY JONES.QUERY6 (&&NAME=&NAME)
```

DSQSRUN パラメーター

```
QMFn I=JONES.PROC6(&NAME='O''BRIEN')
```

結果の RUN コマンド

```
RUN PROC JONES.PROC6 (&NAME='O''BRIEN')
```

図 42. DSQSRUN を使用し、文字列の一部としてアポストロフィを渡す方法

図 43 は、照会の 2 つの異なる部分で変数値を渡す方法を示しています。

照会 (JONES.QUERY7)

```
SELECT *
FROM Q.STAFF
WHERE DEPT IN &DEPT
AND JOB = &JOB
```

プロシージャ (名前は JONES.QUERY7)

```
RUN JONES.QUERY7 (&&DEPT=&V1 &&JOB=&V2)
```

DSQSRUN パラメーター

```
QMFn I=JONES.PROC7(&V1=(((10,38))) &V2='MGR')
```

結果の RUN コマンド

```
RUN PROC JONES.PROC7(&V1=(((10,38))) &V2='MGR')
```

図 43. DSQSRUN を使用した複数の変数パラメーターおよび値の受け渡し

DSQSPLAN (QMF アプリケーション・プランの命名)**CICS ユーザーへの注**

CICS は DSQSPLAN パラメーターを無視します。

パラメーター名

DSQSPLAN

短縮形 P**有効な値**

QMF アプリケーション・プランの名前

開始手順のカスタマイズ

デフォルト

QMF710

DSQSPLAN パラメーターを使用して、QMF アプリケーション・プランの名前を変更することができます。DSQSPLAN パラメーターのデフォルト値は QMF710 です。QMF アプリケーション・プランが QMF710 でない場合は、このパラメーターで正しい名前を QMF に渡す必要があります。

DSQSPRID (TSO プロファイル・キーの指定)

パラメーター名

DSQSPRID

短縮形 U

有効な値

PRIMEID または TSOID

デフォルト

PRIMEID

このパラメーターでは、次のものに TSO ログオン ID か DB2 1 次許可 ID のどちらを使用するかを指定します。

- Q.PROFILES からユーザーのプロファイル行を選択するとき使用する CREATOR の列の値
- ユーザーのプロファイル行の RESOURCE_GROUP の列がヌルまたはブランクの場合のユーザーのデフォルトのリソース・グループ

QMF のバージョン 3 リリース 1 より前のリリースでは、上に挙げたいずれの場合も TSO ログオン ID を使用しています。

バージョン 2 リリース 4 以降のリリースから移行し、TSO ログオン ID がそれらの 1 次許可 ID とは異なり、CREATOR の列の値を 1 次許可 ID で置き換えたくない場合、このパラメーターを使用することができます。

DSQSDBCS (2 バイト文字セット・データの印刷の設定)

パラメーター名

DSQSDBCS

短縮形 K

有効な値

YES または NO

デフォルト

NO

英大文字または日本語の NLF を使用する場合は、2 バイト文字セット (DBCS) データの印刷が必要になることがあります。DSQSDBCS プログラム・パラメーターを YES に設定すれば、非 DBCS 端末から DBCS データを印刷することができます。

たとえば、ユーザーが IBM 3279 表示端末を使用し、非数値列に DBCS データが含まれている表 (DBCSTABLE) を印刷する必要があるものとします。次のステートメントは、消去 CICS 画面から英大文字 NLF を開始し、ユーザーが PRINT DBCSTABLE (PRINTER=DBCSPRT のようなコマンドを使用して DBCSTABLE を印刷できるようにします。

```
QMFU K=YES
```

DBCSPRT プリンターに GDDM ニックネームを設定する方法の詳細については、323ページの『第17章 ユーザーによるオブジェクトの印刷』を参照してください。

REXX プログラム DSQSCMD を使用したデフォルトの開始値の設定

パラメーター名

DSQSCMD

短縮形 (短縮形なし)

有効な値

NULL または

デフォルト

DSQSCMDE

初期化プログラムを使用して、プログラム・パラメーターのデフォルト値を指定することができます。この目的のため、REXX プログラム DSQSCMD が準備されています。DSQSCMD は、QMF が提供するデフォルトのプログラム・パラメーター値を変更することができ、環境をまたがって実行することができます。

QMF の開始時に指定するパラメーター値によって、REXX プログラム DSQSCMD で設定した値は上書きされます。ワークステーション・セッションの開始時に指定するパラメーター値によって、DSQSCMD で設定した値は上書きされます。

DSQSCMD は、呼び出し可能インターフェースを使用して、QMF がアプリケーション・プログラムから開始される時点で、START コマンドの開始関数キーワードとしてのみ有効です。

開始手順のカスタマイズ

TSO の場合、QMF は REXX プログラム DSQSCMDE を呼び出して (図44 を参照)、プログラム・パラメーターの値を指定します。この IBM 提供のプログラムでは、デフォルト値を指定します。これらのデフォルト値を調整することによって、QMF 環境をインストール先システムに合わせて調整することができます。

DSQSCMDE でパラメーター値を指定する必要がない場合は、'NULL' を使用することができます。

汎用プログラミング・インターフェース

```
/*REXX -----*/
/* DSQSCMDE:                                           */
/*                                                     */
/*  COPYRIGHT: LICENSED MATERIALS - PROPERTY OF IBM   */
/*           5645-DB2, 5648-A70 (C) COPYRIGHT IBM CORP. */
/*           1989, 1998. (PUBLISHED)                  */
/*           ALL RIGHTS RESERVED.                     */
/*           US GOVERNMENT USERS RESTRICTED RIGHTS -  */
/*           USE, DUPLICATION OR DISCLOSURE RESTRICTED */
/*           BY GSA ADP SCHEDULE CONTRACT WITH IBM CORP. */
/*                                                     */
/*  STATUS: VERSION 7 RELEASE 1 LEVEL 0                */
/*                                                     */
/*  This REXX program returns default QMF program parameters. */
/*  Values returned by this program can be substituted with */
/*  values specified on the QMF START command.          */
/*                                                     */
/*-----*/
      Trace Off
/* Signal ON ERROR           Immediate exit upon any error condition */

PARSE UPPER SOURCE CSYS .
```

図44. REXX プログラム DSQSCMDE の例 (1/3)

```

/*-----*/
/* Customer should tailor the QMF environment by adjusting any */
/* of the following variable values. Each variable value is */
/* commented indicating the environment(s) in which it is */
/* effective. */
/* */
/* IMPORTANT: */
/* A value must be specified for each one of the following */
/* variables. Also each variable can only contain a single */
/* value and must NOT contain a blank. Use the term NULL */
/* instead of a blank value. */
/*-----*/
DSQADPAN = "1" /* CMS and TSO */
DSQALANG = "E" /* CMS and TSO */
DSQSBSTG = "NULL" /* CMS and TSO */
DSQSDBCS = "NO" /* CMS and TSO */
DSQSDBNM = "NULL" /* CMS and TSO */
DSQSDBG = "NONE" /* CMS and TSO */
DSQSIROW = "100" /* CMS and TSO */
DSQSMODE = "BATCH" /* CMS and TSO */
DSQSPILL = "NULL" /* CMS and TSO */
DSQSRSTG = "0" /* CMS and TSO */
DSQSRUN = "NULL" /* CMS and TSO

DSQSDCSS = "QMF710E" /* CMS only

DSQSPLAN = "QMF710" /* TSO only
DSQSPRID = "PRIMEID" /* TSO only
DSQSSUBS = "DSN" /* TSO only

/*-----*/
/* Return variables to the QMF start function. */
/* */
/* IMPORTANT: Sequence of variables in RETURN statement must NOT */
/* be altered. */
/*-----*/

```

図 44. REXX プログラム DSQSCMDE の例 (2/3)

開始手順のカスタマイズ

```
IF CSYS = CMS THEN DO
  RETURN DSQSMODE DSQSRUN DSQALANG DSQSIROW DSQSRSTG ,
         DSQSDBCS DSQSDBG DSQSDCSS DSQSBSTG DSQSPILL ,
         DSQSDBNM DSQADPAN
END
ELSE DO
  RETURN DSQSMODE DSQSRUN DSQALANG DSQSIROW DSQSRSTG ,
         DSQSDBCS DSQSDBG DSQSPLAN DSQSSUBS DSQSBSTG ,
         DSQSPRID DSQSPILL DSQSDBNM DSQADPAN
END

ERROR:          /* Immediate exit upon any error condition */
EXIT 12
```

図 44. REXX プログラム *DSQSCMDE* の例 (3/3)

└ 汎用プログラミング・インターフェースの終り _____

第15章 QMF セッション制御機能

このセッション制御機能は、QMF の開始時に特定の QMF プロシージャを実行することによって、QMF セッションを初期化する方法を提供します。この QMF プロシージャの名前は Q.SYSTEM_INI です。この機能によって、Q.SYSTEM_INI プロシージャは、ユーザーが QMF ホーム画面を見るのに先立って、そのユーザーが実行を許可されている任意の QMF コマンドまたは任意の保管照会を実行することができます。

Q.SYSTEM_INI のインストールまたは除去

Q.SYSTEM_INI プロシージャは、他の QMF プロシージャと同様にデータベースに作成し保管します。プロシージャの名前は SYSTEM_INI であり、許可 ID Q のもとで保管する必要があります。この QMF プロシージャは、すべての QMF ユーザー間で共用されます。プロシージャを共用可能にするために、SAVE コマンド・オプション "SHARE=YES" を指定してください。プロシージャについて説明するコメントを追加することもお勧めします。たとえば、次のように指定します。

```
SAVE PROC AS Q.SYSTEM_INI (SHARE=YES,COMMENT='QMF System Initialization Procedure')
```

デフォルトのシステム初期化プロシージャのインポート

QMF は、許可 ID が Q の下にデフォルトの QMF システム初期化プロシージャを正しくインポートし、すべての QMF ユーザー間で共用されるようにするプロシージャを提供します。

このプロシージャを実行する前に、QMF コマンド言語が英語に設定されていることを確認してください。そのためには、QMF グローバル変数 DSQEC_NLFCMD_LANG を 1 に設定します。プロシージャが実行されたら、QMF グローバル変数 DSQEC_NLFCMD_LANG を 0 に設定することによって、統轄言語に戻すことができます。

次のコマンドを発行してください。

```
IMPORT PROC FROM 'QMF710.SDSQSAPE(DSQ1EINI)'
```

次に、Run キーを押すか、RUN PROC コマンドを発行するかして、プロシージャを実行してください。DSQ1EINI PROC は、QMF デフォルトのシステム初期化プロシージャを 'QMF710.SDSQSAPE(DSQ0BINI)' という名前のデ

QMF セッション制御機能

ータ・セットからインポートします。SHARE=YES が指定された Q.SYSTEM_INI PROC としての PROC 例が、データベースに保管されていません。

DSQ1EINI を使用すれば、Q.SYSTEM_INI PROC のユーザー固有のバージョンをインポートすることができます。編集のために現行の Q.SYSTEM_INI PROC を OS/390 データ・セットにエクスポートしたいことがあります。あるいは、PROC を OS/390 データ・セットに最初から作成することもできます。いずれの場合においても、ユーザー独自のバージョンをインポートする準備ができれば、次のコマンドを入力してください。

```
IMPORT PROC FROM 'QMF710.SDSQSAPE(DSQ1EINI)'
```

この結果、DSQ1EINI PROC が表示されます。そこで PROC を編集して、ユーザーの OS/390 データ・セットにインポートされる OS/390 データ・セットの名前を変更してください。次に、Run キーを押すか、RUN PROC コマンドを発行するかして、プロシージャーを実行してください。DSQ1EINI PROC によって、PROC 画面に指定した OS/390 データ・セットがインポートされ、その名前が SHARE=YES を指定した Q.SYSTEM_INI であることが確認されます。このプロシージャーによって、すでに存在していた Q.SYSTEM_INI PROC は置き換えられます。

Q.SYSTEM_INI プロシージャーを実行する時期

Q.SYSTEM_INI プロシージャーは、DSQSRUN パラメーターによって指定した QMF 初期プロシージャーの直前、および QMF による初期化の完了直後に実行します。QMF プロシージャーに使用可能なすべての QMF 機能が、Q.SYSTEM_INI プロシージャーによっても使用可能です。

Q.SYSTEM_INI の使用法

QMF セッション・プロシージャー Q.SYSTEM_INI は、いくつかの QMF グローバル変数またはプロファイル値の設定のような簡単なプロシージャーにすることも、あるいは QMF に対する完全なフロントエンドのような複雑なプロシージャーにすることもできます。各ユーザーは、Q.SYSTEM_INI から所有するセッション・プロシージャーを呼び出すことができますが、固有のセッションで置き換えることはできません。

QMF と共に出荷される例

QMF と共に提供される Q.SYSTEM_INI プロシージャの例では、SHARE=YES をすべてのユーザーのデフォルトにします。

```
--
-- QUERY                D S Q O B I N I
-- MANAGEMENT          -----
-- FACILITY
--
-- Q M F    S Y S T E M    I N I T I A L I Z A T I O N    P R O C
-- -----  -----
--
-- FUNCTION: PROVIDE AN EXAMPLE QMF SYSTEM INITIALIZATION PROCEDURE
--            THAT CAN BE ADDED AFTER QMF INSTALLATION. YOU MAY MOD-
--            IFY OR REPLACE THIS PROCEDURE WITH YOUR OWN VERSION.
--
--            THE PROCEDURE MUST BE STORED IN THE DATABASE UNDER THE
--            NAME OF Q.SYSTEM_INI BEFORE IT WILL RUN AUTOMATICALLY.
--            -----
--
-- THE COMMAND BELOW IS AN EXAMPLE OF ESTABLISHING A NEW DEFAULT
-- FOR THE SHARE OPTION OF THE SAVE COMMAND THAT WILL APPLY TO ALL
-- QMF USERS. (REMOVE THE LEADING COMMENT SYMBOLS "--" TO ACTIVATE
-- IT.)
--
-- SET GLOBAL ( DSQEC_SHARE=1 -- MAKE SHARE=YES THE DEFAULT FOR ALL
```

注: QMF と共に出荷される実際の例は上記の例と異なる場合があります。

図 45. QMF と共に出荷される Q.SYSTEM_INI

Q.SYSTEM_INI は、メンバー DSQOBINI として QMF 製品内にあります。

ユーザー・セッション・プロシージャの例

セッション・プロシージャは別のプロシージャを呼び出すことができます。呼び出される側のプロシージャは、QMF ユーザーが作成、所有、および更新したユーザー・プロシージャにすることができます。各ユーザーが固有の SQLID を持っている場合は、異なるユーザーに同じ名前のプロシージャを使用することができます。それぞれのユーザーが QMF を開始する場合、QMF はそれら独自の SQLID の下に実行されます。QMF オブジェクトまたはデータベース・オブジェクトにアクセスしているときに、別のオブジェクト所有者が指定されていない限りは、SQLID はデフォルトのオブジェクト所有者になります。たとえば、QMF セッション・プロシージャ Q.SYSTEM_INI の場合、グローバル変数つまり会社全体に関するグローバル変数を設定してから、ユーザー・セッション・プロシージャを呼び出すことができます。次の例で、ユーザー・セッション・プロシージャの名前は USER_INI です。

QMF セッション制御機能

```
PROC                Q.SYSTEM_INI                LINE    1

-- This QMF procedure example shows how to setup QMF session defaults for
-- every QMF user and then calls a user procedure called USER_INI that will set
-- individual QMF session defaults
--
QMF SET GLOBAL (DSQEC_NLFCMD_LANG=1)  -- Process English Commands
QMF RESET PROC                        -- Hide Contents of this PROC
QMF SET PROFILE (WIDTH=80,LENGTH=66)  -- Set Default Report Page Size
QMF SET PROFILE (SPACE=COMMON)       -- Set Default Space for Save Data Command
QMF SET GLOBAL (DSQDC_LIST_ORDER=5D)  -- Object List Sorted by Date Modify
QMF SET GLOBAL (DSQEC_RESET_RPT=1)    -- Prompt for Report Completion
RUN USER_INI                          -- Run Users Session Procedure
QMF END                                -- Display QMF Home screen first
QMF SET GLOBAL (DSQEC_NLFCMD_LANG=0)  -- Return to Presiding Language
```

図 46. ユーザー定義プロシージャーを呼び出す Q.SYSTEM_INI の例

```
PROC                WILLIAMS.USER_INI          LINE
1
-- This QMF procedure example shows how to setup QMF session defaults for
-- A QMF user. The following settings replace any settings set by the
-- SYSTEM_INI proc.
--
QMF SET GLOBAL (DSQEC_NLFCMD_LANG=1)  -- Process English Commands
QMF RESET PROC                        -- Hide Contents of this PROC
QMF SET PROFILE (SPACE=MYSPEACE)      -- Store data in MYSPACE.
QMF SET PROFILE (PRINTER=MYROOM)      -- Print reports at My Printer
QMF SET GLOBAL (DSQDC_LIST_ORDER=3A)  -- Object List Sorted by Object Name
QMF SET GLOBAL (DSQEC_RESET_RPT=2)    -- Always ResetReports
QMF SET GLOBAL (DSQEC_SHARE = 1)      -- Always Share My QMF Objects
QMF SET GLOBAL (DSQEC_NLFCMD_LANG=0)  -- Return to Presiding Language
```

図 47. ユーザー・セッション・プロシージャーの例 : user.USER_INI

オブジェクトのリストを表示するプロシージャー

次に示すのは、QMF ホーム画面の代わりにオブジェクトのリストを表示する SYSTEM_INI プロシージャーの例です。


```

PROC                Q.SYSTEM_INI                LINE    1

-- This QMF procedure example shows how to set up QMF session defaults for
-- every QMF user to display a list of objects instead of the QMF Home
-- screen.
--
  QMF SET GLOBAL (DSQEC_NLFCMD_LANG=1)  -- Process English Commands
  QMF RESET PROC                        -- Hide Contents of this procedure
  QMF SET GLOBAL (DSQDC_LIST_ORDER=3A)  -- Object List sorted by object name
  QMF SET GLOBAL (DSQEC_NLFCMD_LANG=0)  -- Return to Presiding Language
  QMF LIST ALL                          -- LIST OBJECTS FOR ENGLISH

```

図 48. QMF ホーム画面ではなく Q.SYSTEM_INI を使用した、オブジェクトのリストの表示

セキュリティおよびセッション・プロシージャの共用

QMF セッション・プロシージャ Q.SYSTEM_INI および、このプロシージャによって使用または呼び出されるその他のオブジェクトは、QMF セッション中、その他の任意の QMF オブジェクトまたはデータベース・オブジェクトと同じセキュリティが適用されます。Q.SYSTEM_INI プロシージャは、QMF セッションが開始されるたびに QMF がその実行を試みることを除けば、他のプロシージャと同じです。このプロシージャが存在しない場合、QMF はその実行を試みません。

Q.SYSTEM_INI プロシージャが存在していて、ただし制限されているか、または共用されていない場合、結果は他の QMF プロシージャ・オブジェクトと同じです。QMF を開始した SQLID が "Q" の場合、このプロシージャは実行可能です。Q 以外の SQLID は、プロシージャ Q.SYSTEM_INI の実行を許可されていないことを示すメッセージを受け取ります。

診断の考慮事項

QMF セッション・プロシージャ Q.SYSTEM_INI は他の QMF プロシージャと同じ環境で実行します。既存の QMF プロシージャに使用されるすべての診断プロシージャを、Q.SYSTEM_INI プロシージャにも使用することができます。通常のプロシージャ実行の他に、QMF の開始時に DSQSRUN パラメーターに指定した QMF 始動プロシージャの前に、このプロシージャを実行することをお勧めします。DSQSRUN パラメーターに指定したプロシージャにセッション制御がある場合には、それらを Q.SYSTEM_INI プロシージャに移動することをお勧めします。

QMF L2 トレース・オプションを使用して、発行されるコマンドおよびメッセージを表示することができます。セッション・プロシージャのコマンドおよび

QMF セッション制御機能

びメッセージは、他のコマンドおよびメッセージと区別されます。QMF トレース・オプションの詳細については、544ページの『QMF トレース機能の使用』を参照してください。

第16章 エンド・ユーザーのための QMF サポートの確立

QMF を開始し、ホーム・パネルが表示されたら、エンド・ユーザーのためのサポートをカスタマイズする上で、QMF 機能を役立てることができます。この章では、エンド・ユーザーが QMF にアクセスしてそのデータベースのデータを扱えるようにするために QMF を設定する方法を説明します。

QMF コード・ページに関する考慮事項

QMF は、GDDM 製品によって提供されるサービスを介して端末画面から情報を受け取ったり端末画面に情報を表示したりします。GDDM 装置サポートの準備、QMF で使用するコード・ページの指定、または GDDM セッションのデフォルトの調整を行うには、*GDDM* システムのカスタマイズおよび管理を参照してください。

Q AUTHID の役割

QMF のインストールによって、SYSADM 権限が自動的にユーザー ID Q に付与されます。ユーザー Q が所有し管理する QMF リソースは、次のとおりです。

- すべての QMF 制御表
- サンプル照会
- QMF と一緒に出荷されるサンプル表 (サンプル表の説明は、*QMF* 解説書を参照)
- 272ページの『ユーザーのデータベース・オブジェクト・リストのカスタマイズ』の節で説明しているデータベース・オブジェクト・リストのデフォルトの視点

本書全般の説明および手順では、読者が QMF の管理のために Q ユーザー ID、または SYSADM 権限を伴う別の ID を使用することを前提としています。

クイック・スタート

ユーザーの QMF 環境の設定および保守にあたっては、表30 に示すステップを指針として使用してください。さらに詳しい説明が必要な場合は、表の右端に記載されているページを参照してください。

QMF サポートの確立

表 30. エンド・ユーザーのための QMF サポートの確立

作業	参照箇所
ユーザーが QMF プロファイルをもつようにする。このためには、ユーザーに Q.PROFILES 表の SYSTEM 行を使用できるようにするか、またはユーザーの SQL 許可 ID に基づいて Q.PROFILES に固有な行を挿入する。	242 ページ
QMF アプリケーション・プランおよびパッケージにアクセスできるようにする。このためには、SQL GRANT ステートメントを使用して EXECUTE 特権を付与する。	255 ページ
ユーザーが処理する必要のあるデータベースおよび QMF オブジェクトにアクセスできるようにする。このためには、表および視点については SQL GRANT ステートメントを使用し、QMF 照会、書式、およびプロシージャについては QMF SAVE コマンドの SHARE パラメーターを使用する。	256 ページ
ユーザーのデータベース・オブジェクト・リストをカスタマイズする。このためには、グローバル変数 DSQEC_TABS_LDB2、DSQEC_TABS_RDB2、DSQEC_ALIASES、DSQEC_COLS_LDB2、および DSQEC_COLS_RDB2 を使用する。	272 ページ
ISPF の文書編集インターフェースをカスタマイズする。このためには、IBM 提供のマクロを使用する。	311 ページ
ユーザーが表を作成できるようにする。このためには (必要に応じて)、私用表スペースを割り当てるか、または DB2 RESOURCE 権限を付与して、共用表スペースを割り当てる。	277 ページ
ユーザーに図表をサポートできるようにする。このためには、GDDM の対話式図表ユーティリティ (ICU) を使用する。	282 ページ
ユーザーの照会、書式、およびプロシージャを保守する。このためには、QMF オブジェクト制御表 (Q.OBJECT_DIRECTORY、Q.OBJECT_DATA、および Q.OBJECT_REMARKS) を更新し、再編成する。	284 ページ
必要に応じて、QMF オブジェクト制御表の表スペースを拡大する。このためには、DB2 DBS ユーティリティの UNLOAD および RELOAD コマンドを使用する。これらの表で定義されている索引および視点を再作成する。	291 ページ
ユーザーのデータベースの表および視点を保守する。このためには、DB2 システム表の更新および再編成を行う。	298 ページ

QMF へのユーザー・アクセスを可能にするユーザー・プロファイルの作成

すべての QMF ユーザーは、ユーザー・プロファイルにアクセスする必要があります。このユーザー・プロファイルは、特定のユーザーの個々の入力を QMF がどのように処理するかを決定します。プロファイルを使用して、ユーザー環境の特定の局面 (たとえば、プリンター出力の宛先、または端末入力の大文字変換の有無など) を制御します。

ユーザーの QMF セッションの各局面は、制御表 Q.PROFILES の列の値にマップします。表 Q.PROFILES の各行は、それぞれが個別のユーザー・プロファ

イルです。245ページの『Q.PROFILES 制御表の読み取り』では、表 Q.PROFILES について詳細に示し、指定できるプロファイル値について説明しています。

インストール先システムのためのプロファイル構造の確立

以下の方法のいずれかを使用して、ユーザーにプロファイルを提供します。

- ユーザーがデフォルトの QMF プロファイル (表 Q.PROFILES で CREATOR の列に値 SYSTEM が入っている行) を使用できるようにする。
Q.PROFILES 表は、この行に事前定義されたデフォルトのプロファイル値が指定された状態で出荷されます。この SYSTEM プロファイルで使用されるデフォルトについては、245ページの『Q.PROFILES 制御表の読み取り』で説明しています。現場の要求を満たす総称プロファイルを作成するために、これらの値を変更することができます。
- 『新規のユーザー・プロファイルの Q.PROFILES 表への追加』に示すように、該当のユーザー用として Q.PROFILES に固有の行を作成する。
Q.PROFILES の CREATOR 列にユーザーの 1 次許可 ID を設定し、他の列の値は個別の要件に応じてカスタマイズします。DSQSPRID の値に TSOID を指定して、TSO で QMF を開始する場合、CREATOR 列はユーザーの TSO ログオン ID になります。

インストール先の一部のユーザーのために固有のプロファイルを作成し、他のユーザーには SYSTEM のデフォルトのプロファイルを使用できるようにすることができます。また、セキュリティー上および追跡上の理由により、SYSTEM のプロファイルを削除して、固有のプロファイルをもたないユーザーによる QMF の使用を防止することもできます。

新規のユーザー・プロファイルの Q.PROFILES 表への追加

SQL INSERT 照会または QMF 表編集機能 (QMF 使用の手引き を参照) を使用して、新規のユーザー・プロファイルを表 Q.PROFILES に追加することができます。244ページの図49 は、JONES という名前の SQL 許可 ID を持つユーザー (基本 QMF すなわち英語版) と、SCHMIDT という SQL 許可 ID を持つユーザー (ドイツ語 NLF) のための固有のプロファイルを作成する SQL の例です。244ページの図49 に示すように、Q.PROFILES の TRANSLATION 列を使用して、英語環境と NLF 環境を区別します。

図49 に示してある値は、使用できるプロファイル値の例です。その他の有効なプロファイル値については、246ページの表31 を参照してください。

QMF サポートの確立

基本 QMF (英語)

```
INSERT INTO Q.PROFILES  
(CREATOR, LANGUAGE, SPACE, TRANSLATION,  
PFKEYS, SYNONYMS, RESOURCE_GROUP,  
ENVIRONMENT)  
VALUES ('JONES', 'PROMPTED', 'SAVEIT'  
'ENGLISH', 'PFKEYS', 'COMMAND_SYNONYMS'  
'NONPRIME', 'CICS')
```

ドイツ語 NLF

```
INSERT INTO Q.PROFILES  
(CREATOR, LANGUAGE, SPACE, TRANSLATION,  
PFKEYS, SYNONYMS, RESOURCE_GROUP,  
ENVIRONMENT)  
VALUES ('SCHMIDT', 'MENUE', 'STUT2BER'  
'DEUTSCH', 'DEUTASTEN'  
'COMMAND_SYNONYM_D', 'SCHICHT'  
'CICS')
```

図 49. ユーザー・プロファイルの作成

重要: Q.PROFILES に行を挿入するときに、必ず TRANSLATION の値を指定するか、または TRANSLATION のデフォルトにヌル値を指定します。そうすると、プロファイル行が自動的に無視されます。図49 に示すのは、指定できるプロファイル値のサブセットだけです。その他の値を指定する場合は、指針として245ページの『Q.PROFILES 制御表の読み取り』を使用します。

多くのユーザーを登録する場合は、標準プロファイルを記述し、一般的には登録する新規のユーザーごとに変わる値 (CREATOR 列の値など) の置換変数値を使用する、テンプレート照会を設定します。置換変数の使用に関する詳細については、*QMF 解説書* を参照してください。

NLF を使用する場合: 同じユーザーに対して、使用する各国語環境に応じて異なるプロファイルを設定することができます。ある国語について 1 組のプロファイル値を、別の国語については別の 1 組のプロファイル値を設定することができます。

固有のプロファイルを持たないユーザーによる QMF の使用の防止

複数のユーザーが共通のデフォルトの SYSTEM プロファイルのもとで QMF を使用している場合は、個々のリソースの使用を追跡するのが困難になる可能性があります。QMF の使用を、固有のプロファイルを持つユーザーに制限するには、Q.PROFILES の SYSTEM の行を削除します。245ページの図50 は、このような行を削除する SQL ステートメントを示しています。QMF 使用の手引き で説明されているように、表編集機能を使用することもできます。

基本 QMF (英語) ドイツ語 NLF

```
DELETE FROM Q.PROFILES
      DELETE FROM Q.PROFILES
WHERE CREATOR='sSYSTEM'
      WHERE CREATOR='sSYSTEM'
AND TRANSLATION='ENGLISH'
      AND TRANSLATION='DEUTSCH'
```

図 50. QMF の使用を、固有のプロファイルを持つユーザーに制限

重要: 基本 QMF 環境の場合も NLF 環境の場合も、Q.PROFILES から行を削除するときは TRANSLATION 値を指定する必要があります。指定しないと、意図した行以外の行も (別の各国語環境の行まで含めて) 削除されてしまうことがあります。また、必ず WHERE 文節を使用してください。そうしないと、Q.PROFILES のすべての行が削除されます。

Q.PROFILES から SYSTEM 行を削除したら、すべての QMF ユーザーについて固有のプロファイルを作成してください。作成しないと、ユーザーは、QMF を使用することができなくなります。固有のプロファイルを作成する例を 244 ページの図49 に示してあります。

Q.PROFILES 制御表の読み取り

246ページの表31 は、Q.PROFILES 制御表の列です。制御表の各列は、カスタマイズすることのできる、ユーザーの QMF セッションの局面を表しています。示してあるデフォルトは、英語 QMF 環境のためのものです。

NLF を使用する場合: デフォルト値は、英語環境と一部の NLF とでは異なる場合があります。たとえば、英語のデフォルトが UPPER であるからといって、すべての NLF のデフォルトが UPPER であると見なしてはなりません。ドイツ語 NLF の CASE フィールドのデフォルト値は MIXED です。他の NLF では異なっていることもあります。したがって、デフォルト値については、各 NLF ごとに、変換版の制御表 Q.PROFILE を参照してください。(n シンボルを、xviiiページの表1 の NLID で置き換えます。)

Q.PROFILES 制御表には、属性 UNIQUE および CLUSTER を備えた索引

QMF サポートの確立

Q.PROFILEX があります。キー付きの列は、CREATOR、TRANSLATION および ENVIRONMENT です。これら 3 つの列については、どの行とも同じ値をもつことはできません。

表 31. Q.PROFILES 表の構造

列の名前	データ・タイプと長さ	ヌルの使用	機能および使用できる値
CREATOR	CHAR (8)	不可	<p>機能 : プロファイルを所有する許可 ID (ユーザー) を指定する。</p> <p>値 : SYSTEM (デフォルト)、1 次許可 ID、または TSO ログオン ID (DSQSPRID が TSOID に設定されている場合)。SYSTEM 行は、英語および各 NLF ごとに Q.PROFILES と共に出荷される。固有なプロファイル行をもたないユーザーは、SYSTEM 行を使用することができる。</p>
CASE	CHAR (18)	可	<p>機能 : 端末入力を大文字に変換するかどうかを指定する。</p> <p>値 : UPPER (デフォルト)、STRING、または MIXED。これらの値の説明については、<i>QMF 解説書</i> を参照。NLF ユーザーの場合、CASE は別のデフォルトを持っていることがある。</p>
DECOPT	CHAR (18)	可	<p>機能 : 数字報告書列に QMF が入れる区切り記号を指定する。</p> <p>値 : PERIOD (デフォルト)、COMMA、および FRENCH。詳細については、<i>QMF 解説書</i> を参照してください。NLF ユーザーの場合、DECOPT は変換されて別のデフォルトを持っていることがある。</p>
CONFIRM	CHAR (18)	可	<p>機能 : 確認パネルの表示を制御する。</p> <p>値 : YES (デフォルト) は、データベースの変更前に確認パネルの表示を必要とする場合であり、NO は、その必要がない場合に指定する。</p>

表 31. Q.PROFILES 表の構造 (続き)

列の名前	データ・タイプと 長さ	ヌルの使用	機能および使用できる値
WIDTH	CHAR (18)	可	機能 : 1 ページ当たりの印刷列数を制御する。 値 : 22 から 999。 デフォルト = 132。
LENGTH	CHAR (18)	可	機能 : 1 ページ当たりの印刷行数を制御する。 値 : 1 ~ 999、またはページの制御の切れ目を希望しない場合は CONT。 デフォルト = 60。
LANGUAGE	CHAR (18)	可	機能 : RESET QUERY コマンドが発行された後で、新規の照会を作成する場合に QMF が使用する照会言語を制御する。 値 : SQL (デフォルト)、QBE (例示照会)、または PROMPTED (指示照会)。

QMF サポートの確立

表 31. Q.PROFILES 表の構造 (続き)

列の名前	データ・タイプと長さ	ヌルの使用	機能および使用できる値
SPACE	CHAR (50)	可	<p>機能 : SAVE DATA および IMPORT コマンドを使用して作成される表を保持する表スペースを指定する。</p> <p>DB2 パラレル・エディションの場合、この値は NODEGROUP 名を指す。ただし、QMF は、この値を TABLESPACE 名と呼ぶ。操作には影響しない。 DataJoiner は表スペースを利用しないので、DataJoiner 文脈では SPACE オプションの値は無視される。操作は、ブランク値が指定されている場合のように継続される。</p> <p>値 : SAVE DATA および IMPORT コマンドを使用して作成される表を保持する表スペースを指定する。</p> <p>DB2 パラレル・エディションの場合、この値は NODEGROUP 名を指す。ただし、QMF は、この値を TABLESPACE 名と呼ぶ。操作には影響しない。 DataJoiner は表スペースを使用しないので、DataJoiner 文脈では SPACE オプションの値は無視される。操作は、ブランク値が指定されている場合のように継続される。</p>

表 31. Q.PROFILES 表の構造 (続き)

列の名前	データ・タイプと 長さ	ヌルの使用	機能および使用できる値
TRACE	CHAR (18)	可	<p>機能：トレース出力の詳細レベルを制御する。</p> <p>値：ALL は、最も詳細なレベルまで、すべての機能をトレースする。機能コードと番号の文字ストリングは、個々の QMF 機能のトレース・レベルを指示する。NONE (デフォルト) は、通常レベルのトレースを禁止する。デフォルトは、DSQSMODE の値により異なる。たとえば、DSQSMODE が B のときは、トレース・レベルは L2 であり、それ以外の場合は、NONE である。QMF トレース機能の詳細については、544ページの『QMF トレース機能の使用』を参照。QMF の開始時にトレース値を指定する場合は、217ページの『DSQSDEBUG (トレース詳細レベルの設定)』を参照。NLF で変換される値は ALL と NONE だけである。</p>
PRINTER	CHAR (8)	可	<p>機能：プリンター出力の経路を制御する。</p> <p>値：ヌル (デフォルト) またはブランク値を使用して、印刷出力の宛先を CICS の一時記憶域キューまたは一時データ・キュー、あるいは DD 名が DSQPRINT のデータ・セットに指定する。出力を GDDM 定義プリンターへ向けるには、GDDM ニックネームを使用する。値の選択と指定については、323ページの『第17章 ユーザーによるオブジェクトの印刷』を参照。</p>

QMF サポートの確立

表 31. Q.PROFILES 表の構造 (続き)

列の名前	データ・タイプと長さ	ヌルの使用	機能および使用できる値
TRANSLATION	CHAR (18)	不可	<p>機能 : 英語環境または NLF 環境を指定する。</p> <p>値 : 英語 (デフォルト) または NLF の名前。 xviii ページの表1 の右側に、この列で使用する必要のある変換済みの名前が示してある。</p>
PFKEYS	VARCHAR (31)	可	<p>機能 : ユーザーのカスタマイズされた機能キー定義が保管されている表または視点 (ある場合) を指示する。</p> <p>値 : 有効な DB2 表名または視点名のいずれか。ブランクまたはヌル (デフォルト) の場合は、QMF デフォルトのキーが使用される。361 ページの『第19章 QMF 機能キーのカスタマイズ』に、この表の作成方法の説明がある。</p>
SYNONYMS	VARCHAR (31)	可	<p>機能 : ユーザーのカスタマイズされたコマンド定義が保管されている表または視点 (ある場合) を指示する。</p> <p>値 : 有効な DB2 表名または視点名のいずれか。ブランクまたはヌル (デフォルト) の場合、カスタマイズ定義は使用されない。343 ページの『第18章 QMF コマンドのカスタマイズ』に、この表の作成方法の説明がある。NLF ユーザーの場合、IBM 提供の表は Q.COMMAND_SYNONYM_n という名前が付いている。この場合の n は各国語 ID。</p>

表 31. Q.PROFILES 表の構造 (続き)

列の名前	データ・タイプと 長さ	ヌルの使用	機能および使用できる値
RESOURCE_GROUP	CHAR (16)	可	<p>機能 : 管理プログラム出口ルーチンがどのようにユーザーのリソースまたはコマンドを制限するかを制御する。</p> <p>値 : 任意の有効なリソース・グループ名。ブランクまたはヌル (デフォルト) の場合、QMF はここでユーザーの許可 ID の使用を試みるが、ユーザーのセッションは管理しない (ただし、許可 ID が有効なリソース・グループ名の場合を除く)。詳細については、433ページの『第21章 管理プログラム出口ルーチンを使用した QMF リソースの制御』を参照。</p>
MODEL	CHAR (8)	可	<p>機能 : データ・アクセスのモデルを指定する。</p> <p>値 : この列には、リレーショナル・データを示して、常に値 REL を使用する。</p>
ENVIRONMENT	CHAR (8)	可	<p>機能 : 操作環境を指定する。</p> <p>値 : OS/390 を介してプロファイルにアクセスする場合、この値は CICS、CICSMVS、または TSO のいずれか。プロファイルが DB2 に保管されているが、VM DB2 UDB for OS/390 アプリケーション・リクエストからアクセスされる場合、この値は CMS になる。</p>

注意: DSQPRINT からの出力が HOLD キューに向かうように割り振る場合、OUTPUT キューへの出力を印刷のために解放するには、次の TSO コマンドを発行する必要があります。

```
FREE DDNAME(DSQPRINT)
```

正しいプロファイルをユーザーの操作環境に提供

QMF は開始すると、Q.PROFILES 表の CREATOR、ENVIRONMENT、および TRANSLATION 列を検索して、QMF セッションの確立許可を受けているユーザーを判別します。ユーザーのプロファイルに正しい値を追加して、QMF がそれらの値を認識し、開始することを確認する必要があります。

QMF サポートの確立

QMF は、以下の順序で特定のプロファイル値を検索します。

1. CREATOR= ユーザー ID、ENVIRONMENT= 現行操作環境
2. CREATOR= ユーザー ID、ENVIRONMENT=CICS (CICS で実行の場合)
3. CREATOR= ユーザー ID、ENVIRONMENT=NULL
4. CREATOR=SYSTEM、ENVIRONMENT= 現行操作環境
5. CREATOR=SYSTEM、ENVIRONMENT=CICS (CICS で実行の場合)
6. CREATOR=SYSTEM、ENVIRONMENT=NULL

ユーザー ID は、QMF へのログオンを試みるユーザーの許可 ID です。DB2 は、この ID を使用して、ユーザーがデータベースを使用する許可を得ているかどうかを判別します。

現行操作環境 は、QMF が CICS、TSO、または CMS から開始される場合、それぞれ CICS、OS/390、TSO、または CMS となります。

QMF は、前述のリストのペアの 1 つに一致する CREATOR と ENVIRONMENT のための値を見つける必要があります。見つからない場合、QMF 初期化はエラーとなって、QMF ホーム・パネルを表示せずに終了します。

ユーザー・プロファイルの更新

ユーザー・プロファイルの値は、SET PROFILE コマンドまたは SQL UPDATE ステートメントのいずれかを使用することによって変更することができます。

SET PROFILE コマンドの使用

このコマンドを使用すれば、QMF コマンド行に最小限の入力を行うだけでコマンドの入力ができるので、SQL UPDATE ステートメントを使用するよりも作業が速くなります。

SET PROFILE で設定した値は、このユーザー・セッションが終了するまでの間だけ有効です。変更した値を保管するには、SAVE PROFILE コマンドを使用してください。SET PROFILE コマンドとそのパラメーターの詳細については、*QMF 解説書* を参照してください。

このコマンドを使用するために特別な SQL 特権は必要ないので、ユーザーは容易に自分のプロファイルを変更することができます。ただし、ユーザーは SET PROFILE を使用して、ユーザーの QMF セッションのカスタマイズに使用される可能性があるフィールドを更新することはできません。これらのフィールドは、PFKEYS、SYNONYMS、および RESOURCE_GROUP です。これら

の Q.PROFILES フィールドは、SQL UPDATE ステートメントまたは QMF 表編集プログラムを使用して更新できます。表編集機能については、*QMF 使用の手引き* で説明されています。

SQL UPDATE ステートメントの使用

SQL UPDATE ステートメントを使用すれば、SYNONYMS、PFKEYS、および RESOURCE_GROUP も含めて、Q.PROFILES 制御表のすべてのフィールドを更新することができます。これらの列の説明については、値を指定しなかった場合の結果も含めて、246ページの表31 を参照してください。

これらの列の値を選択する方法の詳細については、以下の項を参照してください。

- 343ページの『第18章 QMF コマンドのカスタマイズ』
- 361ページの『第19章 QMF 機能キーのカスタマイズ』
- 433ページの『第21章 管理プログラム出口ルーチンを使用した QMF リソースの制御』

既存のユーザー・プロファイルを更新する場合は、図51 に示すような SQL UPDATE 照会を使用してください。この例では、ユーザーのコマンド同義語を保管する表の名前を変更します。左側は、基本 (英語) QMF でのユーザー JONES の場合の照会の例で、右側は、ドイツ語 NLF でのユーザー SCHMIDT の場合の同じ照会の例です。

基本 QMF (英語)

ドイツ語 NLF

```
UPDATE Q.PROFILES
      UPDATE Q.PROFILES
SET SYNONYMS='COMMAND_SYNONYMS'
      SET SYNONYMS='GUMMOW.XYZ'
WHERE CREATOR='JONES' AND
      WHERE CREATOR='SCHMIDT' AND
TRANSLATION='ENGLISH'
      TRANSLATION='DEUTSCH'
```

図 51. Q.PROFILES 表で UPDATE 照会を使用したユーザー・プロファイルの更新

重要: Q.PROFILES 表に UPDATE、DELETE、および INSERT 照会を実行するときは、常に 照会に TRANSLATION 列を含めてください。含めない場合は、指定した変更がすべての 言語環境に適用されます。

SYSTEM プロファイルの更新

Q.PROFILES の SYSTEM 行に提供されているデフォルト値を変更することができます。ただし、ユーザーが SYSTEM 行に割り当てられている値とは異なる値を使用する必要がある場合、そのユーザーには固有のプロファイル行が必要です。

たとえば、システムに PRIME および NONPRIME という名前の 2 つのリソース・グループが定義されているものとします。PRIME は、Q.PROFILES の SYSTEM 行の RESOURCE_GROUP フィールドのデフォルト値であるとし、NONPRIME グループ内のユーザーについては、244ページの図49 の例で示しているように、固有のプロファイル行を指定して、正式に登録する必要があります。

Q.PROFILES 制御表からのプロファイルの削除

使用しなくなったユーザー・プロファイルは、Q.PROFILES 制御表から定期的に削除しなければならない場合があります。Q.PROFILES からユーザー・プロファイル削除するのは、そのユーザー・プロファイルの 1 次許可 ID または TSO ログオン ID によって作成されたオブジェクトが削除されているか、または他のユーザーに安全に移行されたかのどちらかが確認できている場合です。

- QMF 照会、書式、およびプロシージャーについてこれらの作業を実行する方法については、284ページの『制御表を使用した QMF オブジェクトの保守』を参照してください。
- データベースの表および視点に関する指示については、298ページの『DB2 カタログ表を使用する表および視点の保守』を参照してください。

ユーザー・プロファイルを削除する場合は、図52 に示す例のような照会を使用します。

基本 QMF (英語)

ドイツ語 NLF

```
DELETE FROM Q.PROFILES
```

```
DELETE FROM Q.PROFILES
```

```
WHERE CREATOR='JONES'
```

```
WHERE CREATOR='SCHMIDT'
```

```
AND TRANSLATION='ENGLISH'
```

```
AND TRANSLATION='DEUTSCH'
```

図 52. QMF ユーザー・プロファイルの削除

NLF を使用する場合: 1 つの NLF 環境でユーザー・プロファイルを削除したい場合は、TRANSLATION 列に値を指定します。TRANSLATION に値を指定しないと、QMF はすべての NLF 環境のプロファイルを削除します。

プロファイルが削除されたユーザーが私用の表スペースを持っていた場合、その表スペースに保管を必要とするものが何も入っていないければ、SQL DROP TABLE SPACE ステートメントを SQL 照会パネルで使用します。また、特定の QMF オブジェクトまたはデータベース・オブジェクトを削除したい場合も、SQL DROP TABLE ステートメントまたは QMF ERASE コマンドを使用できます。DB2 UDB (OS/390 版) SQL 解説書 で、DROP ステートメントについて説明しています。QMF 解説書 に ERASE コマンドの説明があります。

QMF アプリケーション・プランおよびパッケージに対するアクセスの制御

QMF ユーザーはすべて、QMF のインストール中に DB2 によって構築される QMF のアプリケーション・プランおよびパッケージにアクセスできる必要があります。このプランおよびパッケージによって、DB2 アプリケーション・プログラムとしての QMF の実行が可能になります。

アプリケーション・プランおよびパッケージに対するアクセスの付与

SQL GRANT 照会を用いて、PUBLIC または個別のユーザーに EXECUTE 特権を付与することによって、ユーザーが QMF を使用できるようにすることができます。たとえば、ユーザー JONES にアクセスを付与するには、次のようにします。

```
GRANT EXECUTE on plan QMF_PLAN
to JONES
```

QMF のアプリケーション・プランおよびパッケージに対するアクセスを個別に付与する場合は、新規のユーザーごとに SQL GRANT ステートメントを実行する必要があります。

個別ユーザーごとにアクセスを制約する場合は、DB2 の 1 次または 2 次許可 ID を選択し、アプリケーション・プランおよびパッケージの使用をそれらに制限します。細かな差が現れるのは、複数の 1 次許可 ID が同じ 2 次許可 ID を使用している場合です。プロファイルによる QMF への制限登録を使用する場合、QMF にアクセスできるのは、Q.PROFILES に行を持っている 1 次許可 ID だけになります。特定の許可 ID に対する EXECUTE 特権の付与に基づいて QMF に対するアクセスを制約する場合、そのような 許可 ID を 1 次または 2 次許可 ID として持っているだけで、QMF にアクセスできることになります。

QMF アプリケーション・プランおよびパッケージに対するユーザー・アクセスの取り消し

ユーザーの照会、書式、およびプロシージャーの処理後は、QMF のアプリケーション・プランおよびパッケージに対するユーザーのアクセスが個別に付与されている場合、そのようなアクセスを除去する必要があります。その場合に、次のような照会を実行することができます。

```
REVOKE EXECUTE on plan 'QMF_PLAN'  
FROM 'JONES'
```

```
REVOKE EXECUTE on package 'QMF_PACKAGE'  
FROM 'JONES'
```

QMF で使用されるすべてのパッケージに対する EXECUTE 権限を取り消します。

ユーザーの EXECUTE 特権が複数回にわたって付与されていた場合は、次のような照会を使用して、各付与ごとに個別に取り消しを行う必要があります。

```
REVOKE EXECUTE on plan 'QMF_PLAN'  
FROM 'JONES' by all
```

```
REVOKE EXECUTE on package 'QMF_PACKAGE'  
FROM 'JONES' by all
```

GRANT を取り消す場合は、SYSADM 権限が必要です。

除去されるユーザーが元は QMF 管理者で、QMF のアプリケーション・プランおよびパッケージに対するアクセスを他のユーザーに付与している場合は、この元管理者のアクセスを除去すると、そのような他のユーザーについてもアクセスが除去されることになります。

他のユーザーが元ユーザーと許可 ID を共用している場合は、該当の許可 ID からアプリケーション・プランおよびパッケージに対するアクセスを取り消すことがないようにします。取り消した場合は、その許可 ID を共用していたユーザーは QMF を使用できなくなってしまいます。

QMF オブジェクトおよびデータベース・オブジェクトに対するアクセスの制御

照会やプロシージャーなどの QMF オブジェクト、および表編集プログラムなどの機能を使用すると、ユーザーはデータベースの表に保管されているデータをアクセスし、操作することができます。このデータは重要なデータである可能性があり、特定のオブジェクトに対するユーザーのアクセスを制御しなければなりません。

- SQL GRANT および REVOKE ステートメントを QMF の SQL 照会パネルで使用すれば、表および視点に対するアクセスを制御することができます。これについては、268ページの『SQL 特権の付与と取り消し』で説明します。265ページの『オブジェクトにアクセスするのに必要な SQL 特権』には、特定の QMF コマンドまたはオブジェクトに対する機能を使用するために必要な特権についての説明があります。
- QMF SAVE コマンドの SHARE パラメーターを使用すれば、照会、書式、およびプロシージャに対するアクセスを制御することができます。これについては、270ページの『他のユーザーとの QMF オブジェクトの共用』で説明します。

オブジェクトにアクセスするのに必要な DB2 特権

QMF 照会、表編集機能、および QMF コマンドの実行に必要な DB2 特権は、基本 SQL ステートメントの実行に必要な特権です。基本 SQL ステートメントは、265ページの『オブジェクトにアクセスするのに必要な SQL 特権』にリストしてあります。

DB2 特権の配布は、次のように 2 ステップの処理で行われます。

1. ユーザーに一組の許可 ID を割り当てる。
2. 許可 ID に DB2 特権を割り当てる。

特権の割り当ておよび取り消しは、次のようにして行います。

- DB2 出口ルーチンによって許可 ID を割り当てる。
- SQL GRANT 照会によって DB2 特権を割り当てる。
- SQL REVOKE 照会によって前の付与を取り消す。

QMF セッションで実行するすべての照会で DB2 特権が必要なわけではありません。この特権を必要としない照会は、静的照会と呼ばれ、QMF コードに入っています。QMF はこのような照会を使用して、たとえば、所有する制御表を更新します。ユーザーが QMF の管理にまったく関与しない場合は、このような制御表に対する DB2 特権はまったく必要ありません。

動的照会 を実行する特権は、ユーザーだけから得られます。動的照会には、RUN コマンドを用いて実行する照会がすべて含まれます。また、ユーザーに代わって QMF が公式化する特定の照会も含まれます。たとえば、ユーザーは DISPLAY コマンドを発行して、表の内容を表示します。

QMF コマンド、指示照会と QBE 照会、および表編集機能に関して必要な DB2 特権は、265ページの『オブジェクトにアクセスするのに必要な SQL 特権』にリストしてある SQL のものと同じです。

DB2 特権の付与と取り消し

DB2 特権を付与するには、1 つまたは複数の DB2 特権を 1 つまたは複数の許可 ID に付与する GRANT 照会を実行します。たとえば、次の照会では、表 SMITH.TABLEA に対する SELECT および UPDATE 特権を許可 ID JONES および JOHNSON に付与します。

```
GRANT SELECT, UPDATE ON TABLE SMITH.TABLEA TO JONES, JOHNSON
```

DB2 特権の付与を取り消すには、REVOKE 照会を実行します。SQL 許可 ID の付与者である場合は、それらの付与をいつでも取り消すことができます。たとえば、ある QMF セッションで、ユーザーの現行 SQL 許可 ID が JONES であるものとします。JONES は、以前に表 SMITH.TABLEA に対する SELECT 特権を BAKER に付与しています。次の照会によって、この特権の付与は取り消されます。

```
REVOKE SELECT ON TABLE SMITH.TABLEA FROM BAKER
```

付与先の特権を取り消したにもかかわらず、その付与先が依然としてその特権を持っている場合、該当の付与先は、該当の特権を別のユーザーから付与されたこととなります。

PUBLIC に対する付与

付与は PUBLIC および個人に対して行うことができます。PUBLIC に対して、ある特権を付与した場合、その特権はローカル・ユーザーのすべてにとって使用可能になります。

分散データを使用可能にしている DB2 サブシステムのリモート・ユーザーおよびローカル・ユーザーにとって、あるオブジェクトを使用可能にする場合は、PUBLIC AT ALL LOCATIONS に権限を付与します。たとえば、次の照会では、表 Q.STAFF に対する SELECT 特権を付与します。

```
GRANT SELECT ON TABLE Q.STAFF TO PUBLIC  
GRANT SELECT ON TABLE Q.STAFF TO PUBLIC AT ALL LOCATIONS
```

Q.STAFF は QMF のサンプル表の 1 つです。上記の照会例、および他のサンプル表に関する類似の照会例は、QMF のインストール過程で実行されるので、全員がサンプル表に対する SELECT 特権を持つこととなります。

ユーザーに対する特権の付与

GRANT 照会を実行する特権は、付与元から、つまり該当のユーザーの現行 SQL 許可 ID から行われる必要があります。付与元は付与される特権すべてを持っていることが必要であり、各特権は GRANT オプション付きでなければなりません。たとえば、BAKER は、表 SMITH.TABLEA に対する SELECT および UPDATE 特権を JONES に付与したいものとします。この場合、

BAKER は同じ表に対して GRANT オプション付きの SELECT および UPDATE 特権を持っていないければなりません。

GRANT 照会には、式 WITH GRANT OPTION を組み込むことができます。上記の式が組み込まれていると、特権は GRANT オプション付きで付与されます。GRANT オプションが付いていなければ、ユーザーは他のユーザーに権限を付与できません。たとえば、次の照会では、SMITH.TABLEA に対する SELECT 特権を JONES および JOHNSON に付与します。照会の実行後、該当の特権を他のユーザーに付与できるのは、JOHNSON だけです。

```
GRANT SELECT ON TABLE SMITH.TABLEA TO JONES
GRANT SELECT ON TABLE SMITH.TABLEA TO JOHNSON WITH GRANT OPTION
```

DB2 特権を受ける場合は、SQL GRANT 照会による場合、SYSADM 権限による場合、または自らが作成オブジェクトの所有者である場合があります。どのような DB2 特権も、インストール SYSADM 権限の所有者から開始された一連の付与の結果であることがあります。インストール SYSADM 権限は、保有し得る DB2 権限のうち最も高いものです。DB2 のインストール時に、1 つまたは 2 つの許可 ID がこの権限を受け取ります。ユーザーは、この権限を使用して、それよりも下位の特権を他のユーザーに付与することができ、こうして付与された特権は、さらに次々と他のユーザーに付与されます。

特定の特権の付与

特定の特権を付与するためには、ユーザーの許可 ID の 1 つがそうする権限を持っていることが必要であり、この許可 ID がユーザーの現行 SQL 許可 ID であることが必要です。この許可 ID がユーザーの現行 SQL 許可 ID でない場合は、SET CURRENT SQLID 照会を実行します。

表特権の付与

表に関して最も普通に使用される特権は、SELECT、INSERT、UPDATE、および DELETE 特権です。ある表に対する SELECT 特権が付与されると、付与先は SELECT 照会または副照会で、その表からデータを選択することができます。ある表に対する INSERT、UPDATE、または DELETE 特権が付与されると、ユーザーはその表のデータを変更することができます。

特定の表の所有者である場合は、その表特権をすべて GRANT オプション付きで持つことになります。

視点特権の付与

視点アクセスは、機密データや、読み取り専用の場合、または作成の際に付与されます。

スクリーニング・ツールとしての視点: 表の代わりに視点を使用すれば、機密データを他の人から見えないようにすることができます。たとえば、人事情報が入っている表 SMITH.STAFF を基にして視点を作成するとします。表の各行は、それぞれ 1 人の従業員を表しています。各行ごとに、視点では該当の従業員の名前、所属部門、職種、および勤続年数を示すことにします。従業員の給与および手当は、表示しないものとします。

このような視点の作成は、次のような照会を用いて行います。

```
CREATE VIEW VIEWA AS
  SELECT NAME, DEPT, JOB, YRS
  FROM SMITH.STAFF
```

視点所有者と基本オブジェクト: 視点に関する特権の付与は、その視点の所有者から開始されます。本書では、視点の所有者は作成者であることにします。所有者が付与できる特権は、その視点の 基本オブジェクト に対して所有者が持っている権限によって異なります。基本オブジェクトとは、視点の定義照会の FROM 文節で名前を指定する表および視点のことです。たとえば、次の照会で作成される視点の基本オブジェクトは、表 SMITH.STAFF です。

```
CREATE VIEW VIEWA AS
  SELECT NAME, DEPT, JOB, YRS
  FROM SMITH.STAFF
```

視点特権と読み取り専用視点: 視点特権には、SELECT、INSERT、UPDATE、および DELETE があります。SELECT 特権を持っていれば、ユーザーは、表を SELECT 照会および副照会で使用する場合と同じように、視点を使用することができます。それ以外の特権を使用すれば、ユーザーは、該当の視点が表示する表のデータを変更することができます。

視点の所有者は、その視点に対して SELECT 特権を持っていますが、それ以外の特権は持っていない場合があります。視点の所有者が基本オブジェクトに対する特権を持っていない場合、SELECT 以外の特権はない場合があります。言い換えれば、視点が 読み取り専用 なので、他の特権は持たないこととなります。

定義照会が結合の場合、視点は読み取り専用になります。照会が結合以外でも、読み取り専用の視点として現れる場合があります。読み取り専用視点の詳細については、DB2 UDB (OS/390 版) SQL 解説書の CREATE VIEW 照会についての説明を参照してください。

視点を作成するための特権: 視点を作成する場合は、ユーザーの SQL 許可 ID が、その視点の基本オブジェクトのそれぞれに対して SELECT 特権を持っていることが必要です。それ以外の特権は必要ありません。

視点の所有者が 1 つまたは複数の基本オブジェクトに対する SELECT 特権を失った場合、該当の視点はシステムから除去されます。その視点を基本オブジェクトとして使用している視点もすべて除去され、それ以降についても同じことが該当します。

視点特権の付与: ユーザーが視点特権を GRANT オプション付きで持っている場合は、GRANT オプションを使用して、該当の特権を他のユーザーに付与することができます。特権の付与先は、基本オブジェクトに対する特権を持っている必要はありません。この事実によって、視点はデータを見えないようにする上で有用になります。ユーザーは、視点に対する SELECT 特権を付与されても、基本オブジェクトに対する特権がないため、該当の視点しか見ることができません。基本オブジェクトに対する SELECT 特権を必要とする場合、ユーザーは、該当の視点をう回して、基本オブジェクトを直接照会することができます。

視点の所有者の特権: 所有者は、通常、1 つまたは複数の表を作成した後、これらの表から 1 つまたは複数の視点を作成します。こうして作成した視点のそれぞれに関して、所有者には GRANT オプション付きの SELECT 特権があります。視点が読み取り専用でない場合、所有者は INSERT、UPDATE、および DELETE 特権も GRANT オプション付きで持っています。したがって、所有者はこれらの特権を他のユーザーに付与することができます。

他のタイプの基本オブジェクトを持つ視点: 表と視点の両方の所有者の場合は、基本オブジェクトに対して完全な特権のセットを GRANT オプション付きで持っています。基本オブジェクトに視点が含まれている場合、またはオブジェクトが視点の所有者によって所有されていない場合、所有者が基本オブジェクトに対して保有する特権は、大幅に異なることになる可能性があります。

この場合には、次の規則が適用されます。

- 視点の所有者は、その視点に対して常に SELECT 特権を持っています。視点の所有者が GRANT オプション付きでこの特権を持つのは、その視点の基本オブジェクトのそれぞれに対して GRANT オプション付きの SELECT 特権を持っている場合です。
- 視点の所有者がその視点に対して INSERT、UPDATE、または DELETE 特権を持つのは、次の 2 つが両方とも該当する場合です。
 - 視点を読み取り専用ではない。これは、該当の視点が単一の基本オブジェクトをもっていることを意味します。
 - 視点の所有者が該当の単一基本オブジェクトに対しても同じ特権を持っている。

データベースを保守するための権限

データベースの作成後に、ユーザーのいずれかにその保守をしてもらいたいとします。適切な DB2 権限を使用して、このようなユーザーにデータベースに対する DBADM 権限を付与することができます。この権限を付与されたユーザーは、以下に挙げるような保守タスクを実行することができます。

- データベースからの表スペースおよび表の作成と除去
- データベースの表に関する索引の作成と除去
- 表および索引を保守するためのユーティリティの実行

この権限の保有者は、データベース表の実際の所有者が誰であるかに関係なく、データベース表に対して完全な特権のセットを持つこととなります。たとえば、許可 ID JONES にデータベース DBASEA の保守ができるようにしたい場合は、次の照会を実行します。

```
GRANT DBADM ON DATABASE DBASEA TO JONES
```

上記の照会を実行できるのは、ユーザーの SQL 許可 ID が SYSADM 権限を持っているか、または該当のデータベースの所有者である場合です。

データベースに対する DBADM 権限には、次の特権も備わっています。

- CREATETS 特権。これは、データベースの表スペースを作成するためのものです。
- CREATETAB 特権。これは、データベース内に表を作成するためのものです。

データベースに対する DBADM 権限を付与できる場合は、下位の特権も同時に付与することができます。さらに、データベースに対する DBADM 権限を GRANT オプション付きで持っているユーザーであれば、誰でも同じことができます。たとえば、許可 ID JONES にデータベース DBASEA に対する下位特権を持たせたい場合は、次の照会を実行します。

```
GRANT DBADM ON DATABASE DBASEA TO JONES WITH GRANT OPTION
```

適切な特権の付与：SAVE および IMPORT コマンド

CICS では IMPORT の使用は控えめにします。同じアドレス・スペース内の他のユーザーのパフォーマンスに影響を与える可能性があるからです。また、QMF では OS QSAM サービスの GET/PUT も使用します。これによって、入出力操作中に、同じ CICS 領域内の他の QMF ユーザーをロックアウトすることができます。

QMF には、SAVE および IMPORT コマンドの結果の照会を実行する DB2 特権があることが必要です。この特権は、ユーザーが RUN コマンドによって照会を実行する場合と同様に、ユーザーから行われることが必要です。たとえ

ば、QMF が表に対して INSERT 照会を実行するには、まずユーザーが表に対する INSERT 特権を持つか、INSERT 特権を暗黙指定する権限を持つ必要があります。

必要な特権の判別: 必要な特権は、ユーザーが作成する表がユーザー用なのか、または他のユーザー用なのかによっていくぶん異なります。

ユーザーが作成する表が他のユーザー用である場合は、ユーザーの 1 次または 2 次許可 ID を修飾子 (オブジェクトの所有者) とする必要があります。表を別のユーザー用として作成する場合は、他の特権によって適切な CREATE 表照会が実行されても、INSERT 照会が実行されない場合があります。この問題の説明については、268ページの『SQL 特権の付与と取り消し』を参照してください。

表構造が作成された後、ユーザーが所有の表を作成すると、このユーザーは自動的に必要な INSERT 特権を持つことになります。必要なのは、CREATE TABLE 照会を実行する特権だけです。これを行うための最低限の特権は、選択した表スペース・オプションに応じて、次のように異なります。

明示オプション

ユーザーは、少なくとも、データベースに対する CREATETAB 特権、および表スペースを受け取る USE 特権を必要とする。

暗黙オプション

ユーザーは、少なくとも、データベースに対する CREATETAB 特権および CREATETS 特権を必要とする。

デフォルトのデータベース DSNDDB04 のユーザーの場合は、すでにこれらの特権の一部を持っていることがあります。DB2 のインストール過程で、デフォルトのデータベースに関する CREATETAB および CREATETS 特権が PUBLIC に付与されています。したがって、デフォルトのデータベースのユーザーが、暗黙的な表スペース・オプションのもとで操作する場合は、表を作成するための最低限の権限を自動的に持つことになります。これに対して、このユーザーが明示的な表スペース・オプションのもとで操作する場合は、USE 特権のみを付与することが必要です。

重要: データベースは DB2 デフォルトのデータベース (DSNDDB04) である場合があります。ただし、DB2 自体が排他的に使用するデータベースのいずれか (DSNDDB01、DSNDDB03、または DSNDDB05) であってはなりません。

必要な特権の付与: 次の照会の 1 つまたは複数を使用すれば、ユーザーに欠けている特権を付与することができます。

QMF サポートの確立

```
GRANT CREATETAB ON DATABASE &dbname TO &authid  
GRANT CREATETS ON DATABASE &dbname TO &authid  
GRANT USE OF TABLESPACE &dbname.&tbspname TO &authid
```

ここで、

&dbname

は、データベースの名前を指定します。

&authid

ユーザーの許可 ID を指定します。

&tbspname

受け取る表スペースの名前を指定します。

これらの値は、引用符で囲むことはできません。たとえば、データベース DATABSE2 に対する CREATETAB 特権を USERA に付与したい場合は、次の照会を実行します。

```
GRANT CREATETAB ON DATABASE DATABSE2 TO USERA
```

上記の照会を実行する権限があるのは、該当の照会によって付与される特権を持ち、しかも該当の特権を GRANT オプション付きで所有している場合です。SYSADM または SYSCTRL (DB2 2.3 の場合) 権限を持っている場合、あるいは DBADM、DBCTRL、または DBMAINT 権限を GRANT オプション付きで持っている場合は、これに該当します。

他のユーザーの付与の取り消し

SQL 許可 ID が SYSADM 権限を持っている場合は、他のユーザーの付与を取り消すことができます。これによって、特権の付与元が複数の場合でも、その特権を取り消す手段が得られることとなります。たとえば、BAKER が表 SMITH.TABLEA に対して SELECT 特権を持っているとします。JONES は、この特権を BAKER から除去したいと思っていますが、付与元がわかりません。JONES には SYSADM 権限があるので、次の照会を実行することができます。

```
REVOKE SELECT ON TABLE SMITH.TABLEA FROM BAKER BY ALL
```

BY ALL によって、該当の特権に関する付与がすべて除去されます。

PUBLIC に対する付与の取り消し

PUBLIC からの特権の付与の取り消しは、単一の許可 ID からの取り消しの場合と同様に行うことができます。ただし、この場合、同じ特権を別の付与元から取得しているユーザーから、この特権が除去されるわけではありません。

表の所有者から表特権を除去することはできません。また、暗黙のデータベース特権 (CREATETAB など) についても、たとえば、データベースに対する

DBADM 権限を持っているユーザーから、それを除去することはできません。REVOKE 照会を用いてできること、またはできないことの詳細については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。また、*DB2 UDB (OS/390 版) SQL 解説書* の REVOKE コマンドの説明も参照してください。

DB2 権限を付与できるユーザーが多過ぎると起こり得る事態

DB2 特権を取り消すと、意図したより多くのユーザーからこの特権を取り消す場合があります。これはカスケード効果として知られており、権限によっては他の権限の存在に依存するものがあるためです。たとえば、保持していた特権が単一の付与の場合、付与者がその特権を失うと、その特権は失われます。BAKER が SMITH.TABLEA に対して、GRANT オプション付きの SELECT 特権を持っているとします。BAKER は、この特権を JOHNSON と JONES に付与しました。JOHNSON と JONES にとっては、BAKER がこの特権に関する唯一の付与元です。REVOKE 照会によって、BAKER からその特権が除去されることになりました。照会の結果、この特権は JOHNSON と JONES からも除去されます。

特権の喪失が多くのユーザーに波及する可能性のあるのは、特に、特権を失ったユーザーが、他のユーザーに特権を付与していた場合です。この特権の喪失により、同様に他の喪失が生じる可能性があります。

- 視点の所有者は、基本オブジェクトの 1 つに対する SELECT 特権を失った場合、この視点を失います。失われた視点を基本オブジェクトとしている視点があれば、それらの視点も失われ、それ以降についても同様です。
- DB2 アプリケーション・プランは、そのバインドを行った許可 ID が、プログラムの操作に必要なプランの特権を失った場合は、無効になる可能性があります。たとえば、表に対する SELECT 特権がこれに該当する場合があります。この事態が発生すると、そのプログラムの実行は誰にもできません。

カスケード効果と付与の無効な取り消しの両方が発生する可能性が特に高いのは、多くのユーザーが DB2 特権を付与することができる場合です。

オブジェクトにアクセスするのに必要な SQL 特権

ユーザーがまたは視点で特定の SQL ステートメントを使用するには、必要な SQL 特権が付与されている必要があります。たとえば、ユーザー JONES が DISPLAY TABLE SALES_TOTALS と入力しても、表 SALES_TOTALS に対する SQL SELECT 特権がない場合、QMF は次のようなメッセージを表示します。

```
You lack the authorization needed for this DISPLAY command.
```

JONES がこのようなエラー・メッセージを受け取らないようにするには、このユーザーに SALES_TOTALS 表に対する SQL SELECT 特権を付与します。

QMF サポートの確立

ユーザーが QMF コマンドを実行するのか、指示照会または QBE 照会を実行するのか、あるいは表編集機能を使用するのかに応じて、必要な SQL 特権は異なります。

SELECT 照会が QMF によって、QMF 照会インターフェースの 1 つを介して、またはコマンド (DISPLAY TABLE または PRINT TABLE) の結果として発行されると、QMF はその照会に FOR FETCH ONLY を追加して、リモート・データへのアクセス時のパフォーマンス向上を図ります。したがって、QMF によって実行される SQL 照会には FOR FETCH ONLY を追加しないでください。

QMF コマンドのために必要な SQL 特権

表32 を使用して、ユーザーが使用する必要のある QMF コマンドを見つけ、ユーザーが処理する表または視点に対する必要な SQL 特権をユーザーに付与します。SQL GRANT ステートメントの例については、268ページの『SQL 特権の付与と取り消し』を参照してください。

表 32. QMF コマンドおよびそれに同等な SQL

QMF コマンド	コマンドが参照するオブジェクトで必要になる SQL 特権
DISPLAY 表 / 視点	SELECT
DRAW 表 / 視点	SELECT
EDIT TABLE 表 / 視点	必要な特権は表編集機能モードによって異なる。詳細は、267ページの『表編集機能に必要な SQL 特権』を参照。
EXPORT TABLE 表 / 視点	SELECT
IMPORT TABLE 表 / 視点	表がある場合は、SELECT、DELETE、および INSERT。コメントを組み込むには、表の所有権か、または表のデータベースに対する DBADM 権限のどちらかを持っている必要がある。表がない場合は、データベースに対する CREATETAB 特権か DBADM 権限、またはユーザー・プロファイルの SPACE フィールドで指定された表スペースに関する USE 特権を持っている必要がある。
PRINT 表 / 視点	SELECT
RUN 照会	照会で使用される任意の特権。
RUN プロシージャ	プロシージャ内のコマンドで使用される任意の特権。
SAVE DATA	表がある場合は、SELECT、DELETE、および INSERT。コメントを組み込むには、表の所有権か、または表のデータベースに対する DBADM 権限のどちらかを持っている必要がある。表がない場合は、データベースに対する CREATETAB 特権か DBADM 権限、またはユーザー・プロファイルの SPACE フィールドで指定された表スペースに関する USE 特権を持っている必要がある。

表 32. QMF コマンドおよびそれに同等な SQL (続き)

QMF コマンド	コマンドが参照するオブジェクトで必要になる SQL 特権
LIST 表 / 視点	SELECT

すべてのユーザーが SAVE コマンドを使用して新しい表を作成できるわけではありません。詳細については、277ページの『ユーザーがデータベース内で表を作成できるようにする』を参照してください。

SELECT、INSERT、UPDATE、または DELETE などの SQL 特権の詳細については、DB2 UDB (OS/390 版) SQL 解説書を参照してください。

指示照会および QBE 照会に必要な SQL 特権

表33 を使用して、ユーザーが必要とする照会のタイプを見つけ、照会を実行する表または視点に関する SQL 特権をユーザーに付与します。

表 33. QMF 照会のタイプおよびそれに対応する SQL

ユーザーが使用する照会のタイプ	必要な SQL 特権
PROMPTED	SELECT
QBE I.	INSERT
QBE P.	SELECT
QBE U.	UPDATE
QBE D.	DELETE

指示照会または QBE 照会の詳細については、QMF 使用の手引き を参照してください。

表編集機能に必要な SQL 特権

表34 を使用して、ユーザーが使用する必要のある表編集機能を見つけ、ユーザーが編集する必要のある表または視点に関する SQL 特権をユーザーに付与します。

表 34. 表編集機能コマンドおよびそれと対応する SQL

ユーザーが使用する表編集機能の機能	編集する表または視点に必要な SQL 特権
ADD	INSERT
SEARCH	SELECT
CHANGE	UPDATE
DELETE	DELETE

表編集機能の詳細については、QMF 使用の手引き を参照してください。

SQL 特権の付与と取り消し

ユーザーは、作成するオブジェクトを自動的に所有し、データベースに保管します (異なる所有者として表作成されていない限り)。オブジェクトの所有者は、所有するオブジェクトに関するすべての SQL 特権を自動的に所有しており、他のユーザーに対しこれらの特権を付与する (または取り消す) ことができます。DB2 管理者権限を持っていれば、データベース内のいずれのオブジェクトに関しても、SQL 特権の付与または取り消しを行うことができます。ユーザー Q はこの権限を持ち、QMF のインストール中に DB2 に対して事前定義されます。

所有していないオブジェクトの特権を付与するか、または取り消すときは、所有者の SQL 許可 ID でそのオブジェクトを修飾します。

```
JONES.ORDER_BACKLOG
```

SQL 許可 ID は暗黙の修飾子であっても構いません。照会には非修飾の表名、視点名、および索引名を含めることができます。QMF コマンドには非修飾の照会名、プロシージャー名、および書式名を含めることができます。これらの場合、ユーザーの SQL 許可 ID が暗黙の修飾子として使用されます。たとえば、ユーザーが JONES を現行 SQL 許可 ID として使用し、操作してとします。セッション中、そのユーザーが次のようなコマンドを出すとしてとします。

```
RUN QUERYA (FORM=FORMA
```

これによって、次のような SQL 照会が実行されます。

```
SELECT * FROM TABLEA
```

RUN コマンドでは、照会 JONES.QUERYA と書式 JONES.FORMA が参照されます。SELECT コマンドでは、表 JONES.TABLEA が参照されます。

非修飾名を持つ表、視点、索引、または別名を作成する場合、ユーザーの現行 SQL ID がそのオブジェクトの所有者になります。したがって、この ID には、該当のオブジェクトを作成するのに必要な特権が必要です。

修飾名を持つ表、視点、索引、または別名を作成する場合、修飾子は、ユーザーの 1 次または 2 次許可 ID であることが必要です。ただし、ユーザーの現行 SQL ID が少なくとも DBCTRL 権限を持っている場合、表または索引については、任意の修飾子を使用できます。また、SYSADM 権限を持っている場合は、視点または別名についても、任意の修飾子を使用することができます。

必要な特定の権限の詳細については、DB2 UDB (OS/390 版) 管理の手引きを参照してください。

SQL GRANT ステートメントの使用

SQL GRANT ステートメントは、SQL SELECT、UPDATE、INSERT、および DELETE 特権を付与するために使用します。たとえば、ユーザー JONES が次のコマンドを発行する必要があるとします。

```
EDIT TABLE ORDER_BACKLOG (MODE=CHANGE
```

この表の所有者である場合は、図53 のステートメントを使用して、変更モードで表 ORDER_BACKLOG を編集するのに必要な SQL UPDATE 特権を JONES に付与します。

```
GRANT UPDATE ON ORDER_BACKLOG TO JONES WITH GRANT OPTION
```

図53. 単一の QMF ユーザーに対する SQL 特権の付与

WITH GRANT OPTION は、JONES が ORDER_BACKLOG 表に関して付与された SQL 特権のどれでも他のユーザーに付与できることを示します。

GRANT 照会を頻繁に実行する必要がある場合は、照会の中で変更の頻度が高い部分 (UPDATE、ORDER_BACKLOG、および JONES など) に代えて QMF 変数を使用します。変数については、QMF 解説書で説明されています。また、照会が複数の場合は、QMF プロシージャーを使用して、この作業を行うことを考慮する必要もあります。QMF 使用の手引きにプロシージャーの作成方法が説明してあります。

すべての QMF ユーザーに対し SQL 特権を付与するには、キーワード PUBLIC を使用します。たとえば、図54 のステートメントを使用すれば、表 ORDER_BACKLOG に対する INSERT 権限をすべてのユーザーに付与し、それらのユーザーのそれぞれが他のユーザーに INSERT 権限を付与できるようにすることができます。

```
GRANT INSERT ON ORDER_BACKLOG TO PUBLIC WITH GRANT OPTION
```

図54. すべての QMF ユーザーへの SQL 権限の付与

GRANT ステートメントの詳細については、DB2 UDB (OS/390 版) SQL 解説書を参照してください。

重要: 1 つのデータベース・オブジェクトに対する INSERT、UPDATE、または DELETE 特権を複数のユーザーに付与した場合、複数のユーザーが該当のオブジェクトに同時にアクセスしようとする、リソースの競合が起こり、パフォーマンスまたはその他の問題の原因となることがあります。

QMF サポートの確立

ます。あるユーザーが QMF 初期化時に必要な表を編集していると、その表はロックされて、他のユーザーによる QMF の開始が妨げられることがあります。

SQL REVOKE ステートメントの使用

SQL REVOKE ステートメントを使用すると、特権が除去されます。

```
REVOKE UPDATE ON ORDER_BACKLOG FROM JONES
```

図 55. QMF ユーザーからの SQL 特権の取り消し

すべての QMF ユーザーから特権を取り消すには、PUBLIC キーワードを使用します。

DB2 特権には カスケード 構造があります。つまり、あるユーザーから特権を取り消すと、そのユーザーから付与を受けていたその他のユーザーからも、該当の特権が自動的に取り消されます。

REVOKE ステートメントの詳細については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

他のユーザーとの QMF オブジェクトの共用

QMF ユーザーであれば、誰もが QMF SAVE コマンドの SHARE パラメーターを使用することによって、QMF の照会、書式、およびプロシージャに対するアクセスを可能にすることができます。

オブジェクトを保管する際に SHARE=YES を指定すれば、他のユーザーが誰でもその照会を表示でき、その照会を置換または消去しない QMF コマンドで使用できるようにすることができます。たとえば、図56 のコマンドでは、現行照会を ORDER_QUERY として保管し、他のユーザーが誰でもその照会を表示および実行できるようにしています。

```
SAVE QUERY AS ORDER_QUERY (SHARE=YES
```

図 56. QMF オブジェクトの共用

デフォルトはグローバル変数 DSQEC_SHARE によって定義されます。詳細については、*QMF 解説書* を参照してください。

オブジェクトの所有者は、図57 に示すように、DISPLAY コマンドの後に続けて SAVE コマンドを使用して、オブジェクトの共用状況をいつでも変更することができます。

```
DISPLAY ORDER_QUERY
SAVE QUERY AS ORDER_QUERY (SHARE=NO)
```

図 57. QMF オブジェクトの共用状況の変更

SAVE コマンドの詳細については、*QMF 解説書* を参照してください。

非コミット読み取りの許可

ユーザーの QMF セッションが非コミット読み取りを行えるようにしたい場合は、Q.SYSTEM_INI プロシージャ内のグローバル変数 DSQEC_ISOLATION の値を指定することができます。

非コミット読み取りは分散環境において役立ちます。ただし、非コミット読み取りを許可することによって、存在しないデータを QMF 報告書に取り込む可能性が生じます。ユーザーの QMF 報告書に存在しないデータがないようにする必要がある場合は、非コミット読み取りを許可しないでください。

以下の値を取ることができます。

- 0 分離レベル UR、非コミット読み取り。
- 1 分離レベル CS、カーソル固定。これがデフォルトです。

QMF V6 については、値 0 は、以下の Database Server (SQL の文節使用をサポートするもの) と共に使用される場合のみ有効になります。

- DB2 (OS/390 版) V4 またはそれ以降
- DB2 (VM 版) V5 またはそれ以降

オブジェクトの作成に関する標準の設定

インストール先システム内のオブジェクトは、多くのユーザーの間で共用される場合があります。したがって、該当のオブジェクトが何であり、どのように使用するかを示す名前をオブジェクトに付けておく必要があります。照会、書式、プロシージャ、および表のユーザーが、他のユーザーのために、それらの目的を記述するコメントを付けるようにしてください。オブジェクトが表および視点の場合は、保守および管理の必要性がさらに高くなるため、作成に関する特別な指針の確立を考慮します。

QMF サポートの確立

SAVE コマンドを使用して QMF およびデータベース・オブジェクトのコメントを作成する方法については、*QMF 解説書* を参照してください。

ユーザーのデータベース・オブジェクト・リストのカスタマイズ

QMF ユーザーは、定期的にデータベースに保管したオブジェクトをリストしたり、表の用途または表内の列に入っているデータのタイプを示すコメントを表示したりする必要があります。このような機能を実行するのが、QMF の LIST および DESCRIBE コマンドです。

ユーザーが表に対して LIST または DESCRIBE コマンドを発行すると、QMF は、一連の DB2 カタログ表で定義されている視点を使用して、その表に関する情報を入手します。この視点の名前は、グローバル変数 DSQEC_TABS_LDB2 または DSQEC_TABS_RDB2 に保管されます。ユーザーが表内の列に対してそれらのコマンドを発行すると、QMF はグローバル変数 DSQEC_COLS_LDB2 または DSQEC_COLS_RDB2 を使用して、視点の名前を入手します。

QMF では、インストール時にロードされる一組のデフォルトの視点を用意しており、これらは、ユーザーが見ることを許可されている表および列情報だけを戻します。許可に関する処理には余分な時間とリソースが必要なので、QMF では、ユーザーが独自の視点を作成して、表リストおよび列情報をカスタマイズできるようにもしています。

デフォルト・オブジェクト・リストの使用

QMF は以下のデフォルトの視点を提供し、DB2 (MVS/ESA 版) データベースへのインストール時に、それらを自動的にユーザー Q に割り当てます。

```
Q.DSQEC_TABS_LDB2
Q.DSQEC_TABS_RDB2
Q.DSQEC_COLS_LDB2
Q.DSQEC_COLS_RDB2
Q.DSQEC_ALIASES
```

また、QMF はリモート作業単位環境で必要になる場合がある SQL デフォルトの視点も提供します。

```
Q.DSQEC_TABS_SQL
Q.DSQEC_COLS_SQL
```

視点 Q.DSQEC_TABS_LDB2 は、DB2 (MVS/ESA 版) および Workstation Database Server の現行ロケーションからの表および視点のリストだけを選択し

ます。図58 は、DB2 (MVS/ESA 版) に提供される視点を示します。

```
CREATE VIEW Q.DSQC_TABS_LDB2
  (OWNER,TNAME,TYPE,SUBTYPE,MODEL,RESTRICTED,REMARKS,
   CREATED,MODIFIED,LAST_USED,LABEL,LOCATION,OWNER_AT_LOCATION,
   NAME_AT_LOCATION)
AS SELECT DISTINCT
  CREATOR,NAME,'TABLE',TYPE,' ',' ',REMARKS,' ',' ',' ',
  LABEL,LOCATION,TBCREATOR,TBNAME
  FROM SYSIBM.SYSTABLES, SYSIBM.SYSTABAUTH
  WHERE CREATOR = TCREATOR AND NAME=TTNAME AND GRANTEETYPE = ' ' AND
  GRANTEE IN (USER,'PUBLIC',CURRENT SQLID,'PUBLIC*')
```

図58. LIST コマンドの表のリストを提供するデフォルトの視点

自分で作成した視点 (たとえば、QMFADM.LOCAL_DB2_TABLES) を使用し、デフォルトの視点を一時変更するには、次のようなコマンドを出します。

```
SET GLOBAL (DSQC_TABS_LDB2 = QMFADM.LOCAL_DB2_TABLES
```

視点 Q.DSQC_TABS_RDB2 は、3 部分名または LIST の LOCATION オプションによってアクセスしたりリモート DB2 ロケーションの表および視点のリストだけを選択します。ユーザーの現行ロケーションは DB2 でなければなりません。図59 は、この視点が提供する情報のタイプを示します。

```
CREATE VIEW Q.DSQC_TABS_RDB2
  (OWNER,TNAME,TYPE,SUBTYPE,MODEL,RESTRICTED,REMARKS,
   CREATED,MODIFIED,LAST_USED,LABEL,LOCATION,OWNER_AT_LOCATION,
   NAME_AT_LOCATION)
AS SELECT DISTINCT
  CREATOR,NAME,'TABLE',TYPE,' ',' ',REMARKS,' ',' ',' ',
  LABEL,LOCATION,TBCREATOR,TBNAME
  FROM SYSIBM.SYSTABLES, SYSIBM.SYSTABAUTH
  WHERE CREATOR = TCREATOR AND NAME=TTNAME AND GRANTEETYPE = ' ' AND
  GRANTEE IN (USER,CURRENT SQLID,'PUBLIC*')
```

図59. LIST コマンドの表のリストを提供するデフォルトの視点

自分で作成した視点 (たとえば、QMFADM.REMOTE_DB2_TABLES) を使用し、デフォルトの視点を一時変更するには、次のようなコマンドを発行します。

```
SET GLOBAL (DSQC_TABS_LDB2 = QMFADM.REMOTE_DB2_TABLES
```

リモート・ユーザーの場合: 関連するリモート・ロケーションで PUBLIC としてのみ定義されているオブジェクトにアクセスすることはできません。

QMF サポートの確立

視点 Q.DSQC_ALIASES は、表のリストに関する別名のリスト、あるいは DB2 (MVS/ESA 版) または DB2 Common Server 内の別名に関する列情報だけを選択します。図60 は、DB2 (MVS/ESA 版) に提供される視点を示します。

```
CREATE VIEW Q.DSQC_ALIASES
  (OWNER, TNAME, TYPE, SUBTYPE, MODEL, RESTRICTED, REMARKS,
   CREATED, MODIFIED, LAST_USED, LABEL, LOCATION, OWNER_AT_LOCATION,
   NAME_AT_LOCATION)
AS SELECT
  CREATOR, NAME, 'TABLE', TYPE, ' ', ' ', REMARKS, ' ', ' ', ' ',
  LABEL, LOCATION, TBCREATOR, TBNAME
FROM SYSIBM.SYSTABLES
WHERE CREATOR IN (USER, CURRENT SQLID) AND TYPE = 'A'
```

図60. LIST コマンドの別名のリストを提供するデフォルトの視点

自分で作成した視点 (たとえば、QMFADM.DB2_ALIASES) を使用し、デフォルトの視点を一時変更するには、次のようなコマンドを発行します。

```
SET GLOBAL (DSQC_ALIASES = QMFADM.DB2_ALIASES)
```

視点 Q.DSQC_COLS_LDB2 は、現行の DB2 (MVS/ESA 版) ロケーションまたは Workstation Database Server のロケーションに存在する表に関する列情報だけを選択します。図61 は、DB2 (MVS/ESA 版) に提供される視点を示します。

```
CREATE VIEW Q.DSQC_COLS_LDB2
  (OWNER, TNAME, CNAME, REMARKS, LABEL)
AS SELECT DISTINCT
  TBCREATOR, TBNAME, NAME, REMARKS, LABEL
FROM SYSIBM.SYSCOLUMNS, SYSIBM.SYSTABAUTH
WHERE TCREATOR = TBCREATOR AND TTNAME = TBNAME AND GRANTEETYPE = ' '
AND GRANTEE IN (USER, 'PUBLIC', CURRENT SQLID, 'PUBLIC*')
```

図61. DESCRIBE コマンドの列情報を提供するデフォルトの視点

自分で作成した視点 (たとえば、QMFADM.LOCAL_DB2_COLUMNS) を使用し、デフォルトの視点を一時変更するには、次のようなコマンドを発行します。

```
SET GLOBAL (DSQC_COLS_LDB2 = QMFADM.LOCAL_DB2_COLUMNS)
```

視点 Q.DSQC_COLS_RDB2 は、別の DB2 ロケーションにある表からの列情報だけを選択します。ユーザーの現行ロケーションは DB2 でなければなりません。275ページの図62 は、この視点が提供する情報のタイプを示します。

```

CREATE VIEW Q.DSQEC_COLS_RDB2
  (OWNER, TNAME, CNAME, REMARKS, LABEL)
AS SELECT DISTINCT
  TBCREATOR, TBNAME, NAME, REMARKS, LABEL
FROM SYSIBM.SYSCOLUMNS, SYSIBM.SYSTABAUTH
WHERE TCREATOR = TBCREATOR AND TTNAME = TBNAME AND GRANTEETYPE = ' '
  AND GRANTEE IN (USER,CURRENT SQLID,'PUBLIC*')

```

図 62. DESCRIBE コマンドの列情報を提供するデフォルトの視点

自分で作成した視点 (たとえば、QMFADM.REMOTE_DB2_COLUMNS) を使用し、デフォルトの視点を一時変更するには、次のようなコマンドを発行します。

```
SET GLOBAL (DSQEC_COLS_RDB2 = QMFADM.REMOTE_DB2_COLUMNS
```

リモート・ユーザーの場合: 関連するリモート・ロケーションで PUBLIC としてのみ定義されているオブジェクトにアクセスすることはできません。

QMF と一緒に出荷される視点は、SYSIBM.SYSTABAUTH が、ユーザーまたは PUBLIC に所定の表に対する許可を与える複数の項目を持つ場合は、複数の同一行を戻すことができます。QMF LIST または DESCRIBE コマンドで使用される場合、重複する OWNER および TNAME を持つ行 (表視点の場合)、または重複する OWNER、TNAME、および CNAME を持つ行 (列視点の場合) は無視されます。

デフォルト・リストの変更

DB2 は許可情報を表 SYSIBM.SYSTABAUTH および SYSIBM.SYSCOLAUTH から収集するので、表リストおよび列情報に関して QMF 提供のデフォルトの視点を使用すると、処理時間が増える可能性があります。このような許可検査によって得られる特別のセキュリティーを必要としない場合は、データベースに保管されるオブジェクトのリストを生成する独自の視点の作成を考慮してください。

独自の視点を作成する場合は、276ページの図63 に示すような照会を使用します。この照会では、視点内の重複する行が除去されるので、QMF に行を戻すまでに費やされる時間が DB2 では長くなりますが、データベースとユーザー・マシンとの間のデータ転送が少なくなり、パフォーマンスが向上します。カスタマイズした視点には、QMF で有効な任意の名前を付けることができます。QMF の命名規則については、*QMF 解説書* を参照してください。

QMF サポートの確立

```
CREATE VIEW Q.DATABASE_OBJECTS
(OWNER,TNAME,TYPE,SUBTYPE,MODEL, RESTRICTED, REMARKS,
  CREATED,MODIFIED, LAST_USED,LABEL,LOCATION,OWNER_AT_LOCATION,
  NAME_AT_LOCATION)
AS SELECT CREATOR,TNAME,
'TABLE',TABLETYPE,' ',' ',REMARKS,
' ',' ',' ',' ',TLABEL,' ',' ',' '
FROM SYSIBM.SYSTABLES
WHERE TNAME IN (SELECT TTNAME
FROM SYSIBM.SYSTABAUTH
WHERE TCREATOR = A.CREATOR
AND GRANTEETYPE = ' &'
AND GRANTEE IN (USER, 'PUBLIC'))
```

図 63. グローバル変数を使用したオブジェクト・リストのカスタマイズ

ユーザーが、特権を持っているけれども、結合を必要としない表のみを示す視点を作成したい場合は、SYSIBM.SYSTABAUTH だけから選択し、REMARKS または LABEL については値を戻さない視点を定義することができます。

それ以外の管理者の場合、デフォルトの QMF 視点に似ているけれども、表リストについては SYSIBM.SYSTABLES だけから、列リストについては SYSIBM.SYSCOLUMNS だけから選択する、別の視点を作成することを考慮してください。その上で、管理者は、グローバル変数 DSQEC_COLS_LDB2 または DSQEC_COLS_RDB2 でこの視点に名前を付け、データベース内の任意の列に関して記述情報にアクセスすることができます。

ユーザーの独自のリスト視点を作成する場合は、以下の規則に従います。

- 視点は、対応する QMF 提供の視点と同じ視点列名を持っていなければならない。代替視点の CREATE VIEW ステートメントに指定する列名の順序は任意です。
- 列はすべてデータ・タイプが CHAR または VARCHAR でなければならない。QMF は、その他のデータ・タイプを検出するとエラーを戻します。
- 視点内の列に関しては、下記の最大長を超えてはならない。
 - TNAME、CNAME、および NAME_AT_LOCATION の場合は 18 文字
 - REMARKS の場合 254 文字
 - LABEL の場合 30 文字
 - RESTRICTED の場合 1 文字
 - LOCATION の場合 16 文字
 - OWNER、TYPE、SUBTYPE、MODEL、および OWNER_AT_LOCATION の場合は 8 文字
- OWNER、TNAME、TYPE、および CNAME には必ず値を指定します。これらの列をヌルにすることはできません。

DSQEC_TABS_LDB2、DSQEC_TABS_RDB2、DSQEC_ALIASES、DSQEC_COLS_LDB2、および DSQEC_COLS_RDB2 は、ユーザーの QMF セッションの局面の制御に役立つ一連のグローバル変数の一部です。プロシージャーにおけるグローバル変数の使用方法に関する詳細については、*QMF 使用の手引き* を参照してください。グローバル変数のリスト、およびアプリケーションでのグローバル変数の使用については、*QMF アプリケーション開発の手引き* を参照してください。

オブジェクト・リストの記憶域要件

LIST コマンドの場合、オブジェクト・リストの行ごとに 2 組の記憶域要件があります。

- QMF 内部 RPT レコードの収集には、以下のような要件があります。
 - オブジェクト OWNER キー情報 (50 バイト)
 - REMARKS (最高 254 バイト)
 - LABEL 付き TABLE (最高 30 バイト)
 - ALIAS (42 バイト)
 - QUERY、PROC、および FORM のオブジェクト情報 (63 バイト)
- 表示データと制御情報を保持する記憶域には、130 バイトに加えて、REMARKS の実際のバイト数 (最高 254 バイト) および表に関連する LABEL の実際のバイト数 (最高 30 バイト) が必要です。

ユーザーがデータベース内で表を作成できるようにする

QMF ユーザーは、次のいずれかの方法を使用して、表を作成することができます。

- SQL CREATE TABLE ステートメント。
SQL CREATE TABLE ステートメントを、QMF SQL 照会パネルで入力するか、または保管照会から実行します。
- QMF DISPLAY TABLE (または DISPLAY *viewname*) コマンドと、その後続く SAVE DATA コマンド。

基礎表または基礎視点に対するすべての SQL 特権が必要です。SAVE DATA コマンドで指定した名前が既存の表の名前である場合、QMF は既存のデータ・オブジェクトを置換するか、または既存のデータ・オブジェクトに追加します。表属性が一致しない場合、SAVE コマンドはリジェクトされる可能性があります。SAVE DATA コマンドの詳細については、*QMF 解説書* を参照してください。

- QMF IMPORT TABLE または IMPORT VIEW コマンド。

QMF サポートの確立

インポートされる表または視点に対するすべての SQL 特権が必要です。ユーザーが **IMPORT** コマンドで指定した名前がすでに存在している表の名前である場合、QMF は既存の表内のデータを置換するか、または既存の表内のデータに追加します。表属性が一致しない場合、**IMPORT** コマンドはリジェクトされる可能性があります。**IMPORT** コマンドの詳細については、*QMF 解説書* およびオンライン・ヘルプを参照してください。

インストール先システムの要件によって、ユーザーに代わって表を作成したり、ユーザー自身が独自の表を作成できるようにしなければならない場合があります。両方の方法を 表35 に示してあります。

表 35. データベース内の表の作成

管理者がユーザーのために表を作成

ステップ 1

表スペースを作成し、最初に使用する前にそのスペースを **DB2 UDB (OS/390 版) 管理の手引き** を使用すれば、表スペースを作成する場合に割り当てる権限を決めるときに役立ちます。

ステップ 2

表を作成するには、SQL **CREATE TABLE** ステートメントを出すか、**QMF DISPLAY** コマンドに続けて **SAVE DATA** コマンドを発行するか、または **IMPORT TABLE** コマンドを発行する。表の作成の例については、*QMF 使用の手引き* を参照。

ステップ 3

DB2 パフォーマンスの向上を図るために、作成する表に 1 つまたは複数の索引を作成します。**CREATE INDEX** ステートメントの説明および表の論理設計の詳細については、*DB2 UDB (OS/390 版) SQL 解説書* を参照。

ユーザーが自分自身で表を作成

ステップ 1

DB2 UDB (OS/390 版) 管理の手引き を使用して、ユーザーに **DB2 CREATETS** 権限または **DB2 CREATETAB** 権限を付与する。表スペースを作成し (ユーザーに **CREATETAB** 権限だけを付与した場合)、最初に使用する前に、**DB2** に対してこれを定義します。

ステップ 2

SPACE フィールドに関して SQL **UPDATE** ステートメントを使用して、ユーザーの **QMF** プロファイル内に表スペースを割り当てます。プロファイルの更新については、252ページの『ユーザー・プロファイルの更新』で説明しています。**SYSTEM** プロファイルは、そのデフォルト値を変更したい場合には更新できます。

ステップ 3

表スペース内に独自の表を作成するユーザーに **CREATETAB** 権限を付与するか、または **CREATETS** 権限を割り当てて、ユーザーが独自に使用する表スペースを作成できるようにします。ユーザーは、自分が作成する表および表スペースに対して、自動的に SQL 特権をすべて持ちます。

表 35. データベース内の表の作成 (続き)

管理者がユーザーのために表を作成	ユーザーが自分自身で表を作成
<p>ステップ 4</p> <p>表にデータを入れる。DB2 LOAD ユーティリティ、QMF IMPORT コマンド (小さい表を転送する場合) またはその他の方法を使用します。DB2 UDB (OS/390 版) ユーティリティの手引きと解説書、に LOAD ユーティリティの使用方法が説明されている。QMF におけるオブジェクトのエクスポートとインポートについては、QMF 使用の手引き を参照。</p> <p>ステップ 5</p> <p>265ページの『オブジェクトにアクセスするのに必要な SQL 特権』で説明しているように、表に関する DB2 および SQL 特権を必要なユーザーに付与します。</p>	<p>ステップ 4</p> <p>SQL CREATE TABLE ステートメント、QMF SAVE DATA と IMPORT コマンド、および表の作成に関して現場で採用しているその他の指針について研修を行います。これらのコマンドの詳細については、QMF 解説書 を参照。</p> <p>ステップ 5</p> <p>新しい表を作成するためにユーザーが出す SAVE DATA または IMPORT コマンドのオブジェクトとなる表または視点に対する DB2 および SQL 特権を付与します。少なくとも SELECT 特権を付与しないと、QMF は新しい表を作成するためのデータを読み取ることができません。</p> <p>QMF 機能およびコマンドのための SQL 特権については、265ページの『オブジェクトにアクセスするのに必要な SQL 特権』以降に説明があります。</p>

CREATE TABLE、CREATE INDEX、および表の作成に関連する他の SQL ステートメントの詳細については、DB2 UDB (OS/390 版) SQL 解説書 を参照してください。

ユーザー用の表スペースの選択および割り当て

表スペースは、ユーザーに割り当てることも、ユーザーが作成することもできます。QMF ユーザーが CREATETAB 権限を持っている場合は、割り当てられた表スペース内に表を作成することができます。表スペースが所有されている場合は、権限が他のユーザーに割り当てられている場合を除き、所有者以外はその中に表を作成することはできません。表スペースに関するその他の指針については、DB2 UDB (OS/390 版) 管理の手引き を参照してください。

表スペースを作成する場合は、明示か暗黙かの 2 つのオプションのいずれかを選択する必要があります。

明示 このオプションを選択した場合は、ユーザーの SAVE および IMPORT コマンドで作成された表はすべて、SQL CREATE TABLESPACE コマ

QMF サポートの確立

ンドで作成された単一の表スペースに現れます。DB2 の用語では、この表スペースは『明示作成』表スペースです。たとえば、次のようになります。

```
UPDATE Q.PROFILES
  SET SPACE='DBASE1.TSPACE1'
  WHERE CREATOR='USERA' AND TRANSLATION='ENGLISH'
```

暗黙 このオプションを選択した場合は、ユーザーの SAVE および IMPORT コマンドで作成された各表は、それぞれ DB2 によってその表専用で作成された表スペースに入ります。DB2 の用語では、この表スペースは『暗黙作成』表スペースです。このような表スペースは、デフォルトの LOCKSIZE、BUFFERPOOL、STOGROUP、およびスペース属性を持ち、表名に由来する名前を持ちます。たとえば、次のようになります。

```
UPDATE Q.PROFILES
  SET SPACE='DATABASE DBACE1'
  WHERE CREATOR='USERA' AND TRANSLATION='ENGLISH'
```

デフォルト属性については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

表スペースの説明については、*DB2 UDB (OS/390 版) 管理の手引き 管理の手引き* を参照してください。

表スペースに関する 2 つのオプションのいずれを選択するかを決めるにあたっては、以下の要因を考慮する必要があります。

表サイズ

暗黙作成表スペースのデフォルト属性は、意図した表には適さない場合があります。スペース・パラメーター (PRIQTY および SECQTY) のデフォルト値は、小さいサンプル表や要約表向けです。ユーザーの表が大きい場合は、明示表スペース・オプションを選択する方が適している場合があります。

表スペースが小さ過ぎる場合、新しい表は表スペース内に残りますが、空です。したがって、表スペースを拡大してからでないと、SAVE または IMPORT コマンドを正常に実行することはできません。表スペースを拡大するプロシージャについては、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

保守 QMF の 明示表スペース・オプションを使用する場合は、セグメント表スペースを活用すれば、保守が単純化されます。暗黙作成表スペースの場合も、保守を単純化することができます。

たとえば、ユーザーがさまざまな一時表を作成し、後でそれらを消去する場合、単純表スペース（セグメント表スペースではなく）でこれらの表を作成し消去すると、デッド・スペースが急速に形成されるため、すぐに表スペースを再編成して、このデッド・スペースを除去しなければならなくなります。これに対して、セグメント表スペースで表を除去する場合、セグメントは除去がコミットされるとすぐに再使用が可能になります。表スペースの再編成を待つ必要はありません。暗黙作成表スペースは、その中に入っている表が消去されると、自動的に消去されます。

リソース競合

リソース競合を避けるには、セグメント表スペースを指定した明示表スペース・オプション、または暗黙表スペース・オプションのどちらかを使用します。

セグメント表スペースでは、ある表がロックされても、そのロックによって他の表のセグメントへのアクセスが妨げられることはありません。単一の単純表スペースに多くの表が入っている場合は、それぞれの表を複数のユーザーが使用すると、リソース競合を生じる可能性があります。表をセグメント表スペースまたは分割表スペースに入れれば、リソース競合を避けることができます。

保全性とセキュリティー

一部の DB2 特権には、ユーザーが必要としない特権ではあるけれども、それをユーザーに付与する必要が生じるものがあります。明示表スペース・オプションを選択した場合、このような追加特権は、選択したデータベース内での表の作成に限定することができます。暗黙表スペース・オプションを選択した場合は、データベースの表スペースを作成する特権をユーザーに付与する必要があり、この特権を `SAVE` および `IMPORT` コマンドで作成する表スペースに制限することはできません。

便宜

ユーザー作成の表では、明示作成表スペースがすでに使用可能になっています。この表スペースは、QMF のインストール過程で作成され、インストール検査プロシージャで使用されます。この表スペースの名前は `DSQDBDEF` で、データベースは `DSQDBDEF` です。この表スペースは、ユーザーの表を保持するのに十分な大きさです。

多くのユーザーがこの表スペースを使用するのは、表が主に読み取り専用の場合だけにする必要があります。

QMF サポートの確立

表スペースのタイプの選択

ユーザー用の表スペースは、次の 3 つのタイプから選択できます。

- 単純
- セグメント
- 区分

表スペースのタイプの詳細については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

ユーザーへの DB2 CREATETAB 権限の付与

データベースに表を作成する必要があるユーザーに、DB2 CREATETAB 権限を付与する必要があります。ユーザーに CREATETAB 権限を付与するには、図64 に示した SQL ステートメントを発行します。ここで、*userid1*、*userid2*、および *userid3* は SQL 許可 ID を表します。(図64 に示した SQL ステートメントは、DB2 (MVS/ESA 版) で使用するものです。)

```
GRANT CREATETAB on database DBASEA TO userid1, userid2, userid3, ...
```

図64. 複数のユーザーに CREATETAB 権限を付与するための SQL ステートメント

CREATETAB 権限を持つユーザーは、表スペース内に表を作成できます。CREATETS 権限を持つユーザーは、自分用の表スペースを作成できます。

あるユーザーに表を作成できるようにさせたいが、使用されるリソースの量については自分の管理下に置いておく必要がある場合は、CREATETS 権限を付与するのではなく、そのユーザー用の表スペースを割り当てます。この方法により、表スペースのサイズ、および使用されるリソースの量を制御することができます。

表スペースの作成に関する情報および DB2 権限レベルの検討については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

ユーザーが図表をサポートできるようにする

QMF は、GDDM-PGF 製品が提供している対話式図表ユーティリティー (ICU) を使用して図表を作成します。図表書式は、データを含まないさまざまなタイプの図表 (円グラフまたはヒストグラムなど) のテンプレートです。ユーザーが図表を作成すると、QMF はデータを図表フォーマットに関連付けます。次にユーザーが QMF DISPLAY CHART または EXPORT CHART コマンドを入

力すると、図表フォーマットとデータが組み合わされて、グラフィック・データ・ファイル (GDF) データが作成されます。

TSO および ISPF での図表サポート

ユーザーは、単一の報告書から散布図、円グラフ、および棒グラフなどのさまざまな図表書式を指定することができます。ユーザーは、IBM 提供の図表書式または独自の図表書式を作成できます。さらに、ユーザーは、新規に作成した図表書式について、保管するライブラリーがあれば、保管しておくことができます。

ユーザーの保管図表書式を保持するためのライブラリーは、次のようにして作成することができます。

1. 次のような DD ステートメントによって新規のライブラリーを作成する。

```
//DSQUCFRM DD DSN=aaaaaaaa,DISP=(NEW,CATLG),
//          UNIT=xxxx,VOL=SER=yyyy,
//          SPACE=(400,(200,50,25)),
//          DCB=(LRECL=400,BLKSIZE=400,RECFM=F)
```

DSN、UNIT、VOL、および SPACE パラメーターは指定しますが、DCB パラメーターは変更しないでください。

2. DD 名 DSQUCFRM を使用して、ユーザーの QMF セッションにライブラリーを割り振る。データ・セットの割り振りは、ユーザーの TSO ログオン・プロシージャによって行う場合や、ユーザーが QMF に到達するために呼び出す CLIST によって行う場合もあります。たとえば、次のように指定します。

```
ALLOC DSNAME(aaaaaaaa) DDNAME(DSQUCFRM) SHR
```

IBM 提供の図表書式は、ライブラリー QMF710.DSQCHART に入っています。このライブラリーを DD 名 ADMCFORM に割り振ります。ユーザーが指定した図表書式については、このライブラリーとユーザーのライブラリーの両方が検索されますが、新規のライブラリーが最初に検索されます。ユーザーが図表書式を保管する場合、必ず新規のライブラリーに保管され、QMF710.DSQCHART に入ることはありません。

このような仕組みになっているため、各ユーザーは、IBM 提供の図表書式と自ら保管した図表書式の両方にアクセスすることができます。また、IBM 提供の図表書式が置き換えられることも、これで防ぐことができます。

CICS での図表のサポート

QMF ユーザーは、GDDM の機能の 1 つである対話式図表ユーティリティー (ICU) により、報告書から図表を作成することができます。ユーザーは、単一の報告書から散布図、円グラフ、および棒グラフなどのさまざまな図表書式を指定することができます。ユーザーは、IBM 提供の図表書式または独自の図表書式を作成できます。さらに、ユーザーは、新規に作成した図表書式について、保管するライブラリーがあれば、保管しておくことができます。インストール後の図表サポートに関する GDDM 機能のロードについては、599ページの『QMF インストール後に図表機能を追加する』を参照してください。

QMF のインストール時に、IBM 提供の図表を保持するために、データ・セットが作成されます。このデータ・セットは、FCT または CSD ファイル項目によって、DSQUCFRM という名前で CICS に記述されます。このデータ・セットは、通常 CICS の始動時に CICS 領域に割り振られ、CICS ユーザー に使用可能になります。データ・セット DSQUCFRM は、QMF から ICS を使用するとき図表書式を保管するために使用されるデフォルトの図表ライブラリーです。拡張形式の ICU パネル・ディレクトリーを使用すれば、図表書式を他の図表ライブラリーに保管することができます。各図表ライブラリーは、それぞれ CICS に対して記述され、QMF を実行している CICS 領域によってアクセスされなければなりません。図表ライブラリーは、FCT またはファイル項目を使用して CSD データ・セットに記述します。拡張 ICU パネル・ディレクトリーの使用方法に関する説明については、*GDDM PGF 対話式図表ユーティリティー* を参照してください。

QMF では、ICU に加えて、図表エクスポート・コマンドを用意しています。このコマンドは、図表全体を図形データ・フォーマット (GDF) で保管する場合に使用します。図表をエクスポートすると、GDF データは GDDM ADMF ライブラリーに保管されます。GDDM の ICU 機能を使用しても、図表全体を GDF で保管することができます。

制御表を使用した QMF オブジェクトの保守

QMF の照会、書式、およびプロシージャーを保管する QMF 制御表を、定期的に圧縮および再編成する必要があります。QMF 制御表の定期保守には、新規の所有者へのオブジェクトの転送、あるいは表用の表スペースが既存の QMF オブジェクトを収容しきれなくなった場合の表スペースの拡大などのようなタスクが含まれる場合があります。

QMF の照会、書式、およびプロシージャーはすべて、次の 3 つの QMF 制御表のいずれかに保管されます。

- 表 Q.OBJECT_DIRECTORY。これは『Q.OBJECT_DIRECTORY 表の読み取り』に説明があります。
- 表 Q.OBJECT_DATA。これは 286ページの『Q.OBJECT_DATA 表の読み取り』に説明があります。
- 表 Q.OBJECT_REMARKS。これは288ページの『Q.OBJECT_REMARKS 表の読み取り』に説明があります。

上記の表について QMF オブジェクトのリスト、表示、または削除を定期的に行い、必要に応じてその再編成を行うことによって、QMF およびデータベースの実行を効率的に保ちます。また、ある所有者から別の所有者にオブジェクトを転送するために、これらの表の情報を使用する必要があることもあります。

Workstation Database Server のユーザー

DB2 Common Server には追加の制御表 Q.OBJECT_DATA2 があります。QMF ユーザーはこの表に対する INSERT 特権が必要です。データベース・アクティビティがすべて完了した時点で、この表にレコードが入ってはいけません。レコードが残っていると、すべて除去される可能性があります。

制御表のモニターまたは再編成を行うユーザーには、STATS および REORG 特権を割り当てる必要があります。

Q.OBJECT_DIRECTORY 表の読み取り

この表には、データベースにある QMF の照会、書式、およびプロシージャータごとに、それぞれ 1 行が入ります。表には、属性 UNIQUE および CLUSTER を備えた索引 Q.OBJECT_DIRECTORYX があります。キー付きの列は OWNER と NAME です。これらの列の値は、2 行に同じ値を持つことはできません。

Q.OBJECT_DIRECTORY 表は、表36 に示す構造を持っています。

表 36. Q.OBJECT_DIRECTORY 表の構造

列名	データ・タイプ	長さ (バイト単位)	ヌルの使用	機能 / 値
OWNER	CHAR	8	不可	オブジェクトの作成者の許可 ID を示す。
NAME	VARCHAR	18	不可	オブジェクトの名前を示す。

QMF サポートの確立

表 36. Q.OBJECT_DIRECTORY 表の構造 (続き)

列名	データ・タイプ	長さ (バイト単位)	ヌルの使用	機能 / 値
TYPE	CHAR	8	不可	オブジェクトのタイプ FORM、PROC、または QUERY を示す。
SUBTYPE	CHAR	8	可	TYPE が QUERY のときは SQL、QBE、または PROMPTED。TYPE が QUERY でないときはヌルまたはブランク。
OBJECTLEVEL	INTEGER	4	不可	QMF はこの数を使用して、Q.OBJECT_DATA 表の定義テキストからオブジェクトを再構築する。
RESTRICTED	CHAR	1	不可	オブジェクトを共有していない場合 (QMF SAVE コマンドの SHARE パラメーターを使用) は YES。オブジェクトを他のユーザーと共有している場合は NO。
MODEL	CHAR	8	可	この値は常に、QMF (OS/390 版) V7R1 の場合 REL で、リレーショナル・データを示す。
CREATED	TIMESTAMP		可	オブジェクトが作成された時点を表すタイム・スタンプ値を示す。この値は SAVE または IMPORT コマンドの後で記録されます。
MODIFIED	TIMESTAMP		可	オブジェクトが最後に変更された時点を表すタイム・スタンプ値を示します。この値は SAVE または IMPORT コマンドの後で記録されません。
LAST_USED	TIMESTAMP		可	オブジェクトが最後に使用された時点を表すタイム・スタンプ値を示します。値は 1 日に 1 回だけ更新されます。

Q.OBJECT_DATA 表の読み取り

この表には、データベースにある照会、書式、およびプロシージャごとに、それぞれ 1 行または複数行が入ります。各行には、これらのオブジェクトの 1 つに関する定義テキストの全部または一部が入ります。オブジェクトはこのテ

キストから再構成されますが、それは表 Q.OBJECT_DIRECTORY の OBJECTLEVEL 列のフォーマット番号に対応するテキストを組み合わせて行います。

Q.OBJECT_DATA 表には、属性 UNIQUE および CLUSTER が指定されている索引 Q.OBJECT_OBJDATAX があります。キー付きの列は OWNER、NAME、および SEQ です。

この表は 表37 に示す構造を持っています。

表 37. Q.OBJECT_DATA 表の構造

列名	データ・タイプ	長さ (バイト単位)	ヌルの使用	機能 / 値
OWNER	CHAR	8	不可	オブジェクトの作成者の許可 ID を示す。
NAME	VARCHAR	18	不可	オブジェクトの名前を示す。
TYPE	CHAR	8	不可	オブジェクトのタイプ FORM、PROC、または QUERY を示す。
SEQ	SMALLINT	2	不可	このテキストがオブジェクトのテキスト全体の中で占める順序を示す。たとえば、この行がオブジェクトの中の最初の行のテキストであれば、SEQ は 1 で、2 番目であれば SEQ は 2 で、以下同様になる。
APPLDATA	LONG VARCHAR FOR BIT DATA (注を参照)	3600 (注を参照)	可	オブジェクトを定義するテキストの全部または一部を含む。テキストは内部 QMF フォーマットで表示される。Q.OBJECT_DIRECTORY 内の OBJECTLEVEL 列によって、このフォーマットが定義される。 考慮事項: APPLDATA 列は、コード・ページ (CCSID) 変換を受けることが絶対にあってはなりません。

注: DataJoiner V1.2.1 および DB2 (AIX 版) パラレル・エディション V1.2 を使用する場合、APPLDATA のデータ・タイプと長さは、ビット・データを示す VARCHAR(3600) になります。これは V1 SQL データベースに関する永続的な制限です。

Workstation Database Server のユーザー

DB2 Common Server の場合、類似の表 Q.OBJECT_DATA2 があります。この表が必要なのは、SAVE または IMPORT コマンドの内部 QMF 処理の場合です。

Q.OBJECT_REMARKS 表の読み取り

この表には、データベースにある照会、書式、およびプロシージャごとに、それぞれ 1 行が入ります。この行には、オブジェクトが作成された際、または最後に置換された際に QMF SAVE コマンドを使用して入力されたコメントが入っています。(QMF 解説書の SAVE コマンドの説明を参照してください。)

Q.OBJECT_REMARKS 表には、属性 UNIQUE および CLUSTER が指定されている索引 Q.OBJECT_REMARKSX があります。キー付きの列は OWNER と NAME です。

この表は 表38 に示す構造を持っています。

表 38. Q.OBJECT_REMARKS 表の構造

列名	データ・タイプ	長さ (バイト単位)	ヌルの使用	機能 / 値
OWNER	CHAR	8	不可	オブジェクトを作成したユーザーの許可 ID を示す。
NAME	VARCHAR	18	不可	オブジェクトの名前を示す。
TYPE	CHAR	8	不可	オブジェクトのタイプ FORM、PROC、または QUERY を示す。
REMARKS	VARCHAR	254	可	オブジェクトが作成または置換された際にオブジェクトと共に保管されたコメントを含む。

QMF 照会、書式、およびプロシージャのリスト

QMF 環境の保守に役立てるために必要な情報を入手するには、QMF ユーザーがデータベースに保管した照会、書式、およびプロシージャをリストする必要があります。管理者権限を持っている場合は、289ページの図65 に示す照会を使用すれば、自分が所有者ではない QMF オブジェクトをリストすることができます。

```

SELECT D.NAME, D.TYPE, D.SUBTYPE, D.RESTRICTED, R.REMARKS
  FROM Q.OBJECT_DIRECTORY D,
       Q.OBJECT_REMARKS R
 WHERE D.OWNER = 'userid'
       AND D.OWNER = R.OWNER
       AND D.NAME = R.NAME
 ORDER BY D.TYPE, D.SUBTYPE, D.RESTRICTED

```

図 65. 特定のユーザーが所有している照会、書式、およびプロシーチャーのリスト

この照会では、タイプ (FORM、PROC、QUERY) 別にソートされ、TYPE が照会の場合は、さらにサブタイプ (SQL、QBE、または PROMPTED) 別にソートされたオブジェクトのリストが戻されます。userid に指定する値は、単一引用符で囲みます。各タイプのオブジェクトは、所有者によって共用されているかどうかに応じて、さらにソートされます。共用状況は、表 Q.OBJECT_DIRECTORY の RESTRICTED 列に反映されます。

QMF 照会、書式、およびプロシーチャーの表示

オブジェクトをリストしても REMARKS 列では情報が十分得られない場合は、次のいずれかの方式によってオブジェクトの表示を試みます。

- 以下の照会を実行してユーザーのオブジェクトを共用して、次に自分自身の ID からこれらのオブジェクトを表示する。

```

UPDATE Q.OBJECT_DIRECTORY
  SET RESTRICTED = 'N'
  WHERE OWNER = 'userid'

```

図 66. 別のユーザーのオブジェクトをすべてのユーザーで共用

userid に指定する値は、単一引用符で囲みます。

重要: この照会を実行するのは、ユーザーのオブジェクトのうちどれが制限され、どれが制限されていないかについて追跡する必要がない場合だけに限ります。この照会の実行後は、RESTRICTED を元どおり Y に設定し直すことができますが、その後は元から制限されていたオブジェクトがどれであったか示すことはできません。

- 表示したいオブジェクトごとに QMF DISPLAY コマンドを出す。

照会、書式、およびプロシーチャーの所有権の転送

ユーザー間で QMF オブジェクトを転送する場合、290ページの図67 に示す照会を使用します。これら 3 つすべての照会を実行してください。

QMF サポートの確立

重要: 最初に、転送するオブジェクトと同じ名前で保管されているオブジェクトが新規所有者にないことを確認します。そうしないと、QMF は既存のオブジェクトを転送するオブジェクトで置き換えます。

```
UPDATE Q.OBJECT_DIRECTORY      UPDATE Q.OBJECT_REMARKS      UPDATE Q.OBJECT_DATA
SET OWNER = 'newuserid'        SET OWNER = 'newuserid'      SET OWNER = 'newuserid'
WHERE OWNER = 'olduserid'      WHERE OWNER = 'olduserid'    WHERE OWNER = 'olduserid'
AND NAME IN namelist          AND NAME IN namelist        AND NAME IN namelist
```

図 67. QMF オブジェクトの他のユーザーへの転送

図67 に示されている照会で、namelist は転送されるオブジェクト名のリストです。このリストは括弧で囲み、各名前は単一引用符で囲んでコンマで区切る必要があります。たとえば、次のように指定します。

```
('QUERY1','QUERY2','FORMA','PROCB')
```

古い SQL 許可 ID で修飾されたオブジェクトを指名する照会またはプロシージャについては、修飾子の変更を忘れないで行ってください。たとえば、MYQUERY を BAXTER から JONES に転送する場合は、名前を BAXTER.MYQUERY から JONES.MYQUERY に変更します。

オブジェクトを転送した後、そのオブジェクトを共用する必要がある場合は、289ページの図66 に示されているような SQL 照会を使用して、RESTRICTED 列の値を Y に変更します。

古くなった 照会、書式、およびプロシージャの削除

特定のユーザーの QMF 照会、書式、およびプロシージャをすべて削除する場合は、図68 の SQL を使用します。必ず 3 つの照会をすべて実行することが必要です。これは、各オブジェクトの内部表現が 3 つの QMF 制御表 Q.OBJECT_DIRECTORY、Q.OBJECT_DATA、および Q.OBJECT_REMARKS にまたがっているためです。ユーザー ID 変数に指定する値は、単一引用符で囲みます。

表が適切に更新されていない場合は、予期できない結果が起こる可能性があります。

```
DELETE FROM Q.OBJECT_DIRECTORY  DELETE FROM Q.OBJECT_REMARKS  DELETE FROM Q.OBJECT_DATA
WHERE OWNER = 'olduserid'      WHERE OWNER = 'olduserid'    WHERE OWNER = 'olduserid'
```

図 68. QMF 制御表から不要なオブジェクトを削除

また、Q.OBJECT_DIRECTORY の日時によるソート機能を使用して、使用されなくなったオブジェクトを削除することもできます。最後に使用された日付が 06/01/95 より前のオブジェクトをすべて選択し、3 つの制御表から該当する行をすべて削除することができます。

OS/390 データ・セット内の照会、書式、およびプロシーチャーのインポート

ユーザーが OS/390 データ・セットにエクスポートされた QMF オブジェクトを持っている場合は、QMF IMPORT コマンドを使用してこれらを元に戻すことができます。

エクスポートされたオブジェクトが RACF 保護付きの場合、これらからオブジェクトをインポートするには、RACF 読み取りアクセスが必要です。このアクセスを取得するには、RACF 管理者に連絡してください。

QMF オブジェクト制御表用の表スペースの拡大

QMF オブジェクトが多くなり過ぎて、QMF オブジェクト制御表 Q.OBJECT_DIRECTORY、Q.OBJECT_DATA、および Q.OBJECT_REMARKS が入っている表スペースに収まりきれなくなる事態が定期的に生じる可能性があります。

表スペースを拡大する前に、そのスペース要件を決定する必要があります。見積もりの要因の 1 つになり得るものとして、現在使用されているスペースの量があります。

スペースが DB2 管理の場合、この情報は次のようにして入手することができます。

1. 表スペースの記憶域グループに対して STOSPACE ユーティリティーを実行する。
2. 次の照会を実行する。

```
SELECT SPACE
  FROM SYSIBM.SYSTABLEPART
 WHERE TSNAME='ttttttt' AND DBNAME='DSQDBCTL'
```

ここで、ttttttt は表スペース名です。結果 (SPACE) によって、現在表スペースに割り振られている記憶域の K バイト (KB) 数が与えられます。

スペースがユーザー管理の場合は、データ・セット名が分かっている場合、スペース情報に関して TSO LISTCAT コマンドを使用することができます。

QMF オブジェクト制御表用の表スペースを拡大するには、次のようにします。

QMF サポートの確立

1. 表スペースのイメージ・コピーを作成する。
これは、プロシージャーが正常に行われなかった場合の復元に使用できます。
2. 表スペース用の記憶域グループを作成する。
これを行うのは、表スペースがユーザー管理のデータ・セットであり、すでに使用可能な記憶域グループがない場合に限りです。
表スペースに使用されるデータ・セット管理のタイプを判別するには、次の照会を実行します。

```
SELECT STORTYPE
  FROM SYSIBM.SYSTABLEPART
 WHERE TSNAME='DSQTSCT3' AND DBNAME='DSQDBCTL'
```

これは、表スペース (DSQTSCT3) に関して、結果は 1 行で出されます。(表39 を参照してください。) 結果は、STORTYPE の値が E または I になります。

- E** 表スペースのデータ・セットがユーザー管理である (関連する記憶域グループがない) ことを示します。
- I** 表スペースのデータ・セットが DB2 管理であることを示します。

表 39. QMF オブジェクトを保管する制御表用の表スペース

表スペース 名前	内容	デフォルト サイズ
DSQTSCT1	Q.OBJECT_DIRECTORY 表	256 ページ
DSQTSCT2	Q.OBJECT_REMARKS 表	256 ページ
DSQTSCT3	Q.OBJECT_DATA	5120 ページ

表 40. DB2 パラレル・エディション VIR2 データベースを使用して QMF オブジェクトを保管する制御表のノード・グループ

NODEGROUP 名	用途	特性
DSQTSCTL	この表の別の個所に記述されている表を除くすべての QMF 制御表。	複数のノードに配布可能。成長への対応力は低い。
DSQTSOBJ	PROC、照会、および FORM の各オブジェクトを保管する QMF OBJECT 制御表。	複数のノードに配布可能。成長への対応力は高い。
DSQTSDEF	QMF プロファイル内で初期化されたデフォルトの SAVE DATA スペース。	複雑にならないように、単一ノードに制限するように定義する必要がある。
DSQTSAMP	QMF サンプル表。	複数のノードに配布する候補。

3. データベースを停止する。

コマンド `-STOP DATABASE(DSQDBCTL)` を使用します。

4. 表スペースの記述を変更する。

- 表スペースのデータ・セットがユーザー管理の場合は、以下の形の DB2 ステートメントを出す。

```
ALTER TABLESPACE DSQDBCTL.tttttt
  USING STOGROUP ssssss PRIQTY pppp SECQTY ssss
```

ここで tttttt は表スペース名です。このステートメントは、表スペースをユーザー管理から DB2 管理に変更し、管理用の記憶域グループ (ssssss) を指定します。数量 pppp および ssss は、拡大後の表スペースの新規の 1 次および 2 次割り振りサイズ (K バイト数) です。

- 表スペースのデータ・セットが DB2 管理の場合は、次のような DB2 ステートメントを実行する。

```
ALTER TABLESPACE DSQDBCTL.tttttt
  PRIQTY pppp SECQTY ssss
```

ここで tttttt は表スペース名です。pppp および ssss は、拡大後の表スペースの新規の 1 次および 2 次割り振りサイズ (K バイト数) です。

5. 表スペース・データを移動する。

表スペース記述を単に変更しただけでは、拡大は行われません。代わりに、表スペースを再度埋め込むことが必要です。

6. 次のステートメントでデータベースを開始する。

`-START DATABASE(DSQDBCTL)`

DB2 LOAD ユーティリティを使用して、表スペースを拡大することもできます。

表スペースの拡大の詳細については、*DB2 UDB (OS/390 版) ユーティリティの手引きと解説書* を参照してください。

注: QMF が以前にインストールされていない場合、QMF バージョン 7 が DB2 管理の表スペースのデータ・セットを作成します。

DB2 プラットフォームに関する注

特に断りがない限り、この節では DB2 (MVS/ESA 版) について説明します。

複数の表スペースで複数のデータベースを保守できます。

Workstation Database Server のユーザー

各サーバー (指定されたロケーション) は単一のデータベースです。このような単一のデータベース内で複数の表スペースを保守できます。

特殊な管理タスクをユーザーに割り当てて、そのユーザー独自の許可 ID のもとでそれを実行させる場合があります。このようなユーザーには、該当のタスクに必要な照会およびユーティリティを実行するのに十分な DB2 権限だけを付与する必要があります。たとえば、1 人で次のものが必要になる場合があります。

- 新しいユーザーの QMF プロファイルを挿入するための、表 Q.PROFILES に対する INSERT 特権
- 関連する表、索引、および表スペースを管理するための、特定のデータベースに対する DBADM 権限
- 表 Q.OBJECT のモニター、および再編成 (必要な場合) を行うための、これらの表のデータベースに対する STATS および REORG 特権

データ・セットの管理

表スペースおよび索引のデータ・セットは、ユーザー管理の場合もあれば、DB2 管理の場合もあります。表スペースおよび索引を拡大するために行わなければならないことは、これらのデータ・セットの管理方法によって決まります。表スペースの拡大については、291ページの『QMF オブジェクト制御表用の表スペースの拡大』で説明しています。

DB2 Common Server のユーザー

QMF 表スペースは、システム管理スペース (SMS) として定義されます。

DB2 管理のデータ・セット用の記憶域グループ

バージョン 3.2 より前のバージョンでは、DB2 が制御表の索引および表スペースのためのスペースを管理していました。したがって、各表スペースと各索引ごとに、記憶域グループを 1 つずつ使用する必要がありました。記憶域グループは、記憶域グループがサポートするオブジェクト用として、スペースが引き出される元の、名前の付いた DASD ボリュームの集合です。表41 に示すように、索引を持つ各制御表ごとに、索引と表スペースは共通の記憶域グループを共用します。

Workstation Database Server のユーザー

記憶域グループは適用されません。

表 41. 制御表の記憶域グループ

表	表スペース	記憶域グループ
Q.PROFILES	DSQTSPRO	DSQSGPRO
Q.ERROR_LOG	DSQTSLOG	DSQSGLOG
Q.OBJECT_DIRECTORY	DSQTSCT1	DSQSGCT1
Q.OBJECT_REMARKS	DSQTSCT2	DSQSGCT2
Q.OBJECT_DATA	DSQTSCT3	DSQSGCT3

ユーザー管理のデータ・セットの VSAM クラスター

制御表のデータ・セットを管理するには、各表スペースおよび各索引ごとに、VSAM クラスターが 1 つずつ必要です。VSAM ステートメントを使用してこれらのクラスターを定義し、結果のクラスターを、SQL CREATE 照会によって DB2 に結び付けます。クラスターとその DB2 オブジェクトとの間の結び付けは、クラスターの名前、およびクラスターがカタログされる ICF (統合カタログ機能) の名前で行われます。

制御表の保守

制御表の保守のほとんどは、QMF のもとで行うことはできません。QMF は制御表に依存して操作が行われるからです。保守照会は、バッチ・モードの TSO

QMF サポートの確立

で DSN プロセッサを介して発行するか、または DB2I の SPUIF 機能を介して対話式に発行することができます。

Workstation Database Server のユーザー

さらに、データベースのローカル・オペレーティング・システム環境から、DB2 コマンド行処理プログラムを使用することができます。

これらのトピックに関する説明については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

保守作業中は QMF を使用しないでください。このことを徹底するためには、制御表が入っている表スペースの 1 つに DB2-STOP DATABASE コマンドを適用します。これにより、他の制御表および索引で保守操作を行うことができます。次のようなことが可能です。

- バッチ・モードの TSO で作業している場合、DB2-STOP DATABASE を DSN への最初の入力として組み込む。
- DB2I を使用している場合は、DB2I コマンド・パネルから DB2-STOP DATABASE コマンドを発行する。

DB2-STOP DATABASE コマンドの説明については、*DB2 UDB (OS/390 版) ユーティリティの手引きおよび解説書* を参照してください。

制御表のモニターと再編成

DB2 システム・カタログを介して制御表の条件をモニターすることにより、保守問題の発生を予防する必要があります。詳細については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

RUNSTATS ユーティリティの実行: 制御表および索引に対して RUNSTATS ユーティリティを定期的に行って、特定の DB2 システム表に現行統計を追加します。その上で、これらの表の照会およびこれらの統計の検査を行って、再編成が必要かどうかを判断します。

再編成が必要な場合は、次のことを行います。

1. REORG ユーティリティを実行する。
2. RUNSTATS ユーティリティを再実行する。
3. 更新後のシステム表を再度照会して、再編成によって統計が向上したかどうかを調べる。

再編成が最も効果的に行われると、制御表および索引のスペース所要量を最小限に抑え、QMF 操作の効率を高めることができます。

DB2 UDB (OS/390 版) 管理の手引き では、最も効率的な検索パスを選択できるように、再編成後に最も重要なアプリケーションの再バインドを行うよう提案しています。このことは、こうした再編成を行うたびに、QMF アプリケーション・プランの再バインドを行う必要があることを意味しています。

索引使用の判別

Q.OBJECT_DATA に項目がほとんどないときに、QMF アプリケーション・プランのバインドが行われると、QMF 照会のパフォーマンスに影響が生じる可能性があります。こうした状況のもとでは、Q.OBJECT_DATA の索引が最適化プログラムによって使用されません。(最適化プログラムは、DB2 機能の 1 つで、表内の行にアクセスする最適の方法を判別するものです。) その代わりに、表スペースの走査が実行されるので、Q.OBJECT_DATA に多くの項目が入った時の、将来のパフォーマンスに影響を及ぼします。アプリケーション・プランを再バインドして、索引が使用されるようにする必要があります。

Q.OBJECT_DATA の索引が使用されているかどうかを判別するには、次の照会を実行してください。

```
SELECT BCREATOR, BNAME
FROM SYSIBM.SYSPLANDEP
WHERE DNAME='QMF710'
AND BTYPE='I'
```

この照会では、QMF アプリケーション・プランを決定する任意の索引の所有者 (BCREATOR) および名前 (BNAME) を選択します。QMF710 がデフォルトの計画名ですが、QMF のインストール時に使用した名前を選択してください。

結果が Q.OBJECT_OBJDATA (QMF V2R2 から移行した場合は Q.OBJECT_DATA) の項目を示していない場合は、次のことを行います。

1. 表スペース DSQDBCTL.DSQTSTCT3 に対して RUNSTATS を実行する。
2. QMF アプリケーション・プランを再バインドする。

バッファ・プールの切り替え

パフォーマンスの理由により、制御表が入る表スペース用または制御表索引用のバッファ・プールを変更しなければならない場合があります。たとえば、インストール先システムで QMF 指向が強い場合は、制御表索引および表スペース用のバッファ・プールを BP1 に切り替えて、BP1 を排他使用するように予約することができます。

QMF サポートの確立

バッファ・プールの変更には、ALTER TABLESPACE および ALTER INDEX 照会を用います。これらの照会およびそれらの照会を実行するのに必要な権限に関する説明については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。新しいバッファ・プール用として BP0、BP1、または BP2 は選択できますが、BP32K は選択できません。

ALTER TABLESPACE および ALTER INDEX 照会を用いて値を変更できるパラメーターが他にもあります。これらのうちで唯一、QMF の操作性を損なわずに変更できるのは DSETPASS パラメーターです。

DB2 カタログ表を使用する表および視点の保守

DBA 権限を持っている場合は、DB2 カタログ表にアクセスして、表および視点のリスト、表示、転送、または削除を行うことができます。このような DB2 カタログ表の使用に関する詳細については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

重要: システム・カタログ内の一部の表には、2 進データが入っている列があります。このような列は文字データ・タイプですが、文字データは含まれていません。これらの列からデータを取り出すと、矛盾した表示が行われる可能性があります。列の「文字」の中に、画面管理プログラムに予期しない信号を出すものがあるからです。システム・カタログ表の詳細については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

表と視点のリスト

図69 の照会は、TABLETYPE (T は表を示し、V は視点を示す)、TNAME (表名)、TABLE SPACENAME、および REMARKS の列を持つ表のリストを戻します。

```
SELECT TABLETYPE, TNAME, TABLE SPACE NAME, REMARKS
FROM SYSIBM.SYSTABLES
WHERE CREATOR = 'userid'
ORDER BY TABLETYPE, TNAME
```

図 69. 特定のユーザーが所有する DB2 表および視点のリスト

表または視点の所有権の転送

表または視点の所有権の転送は、お勧めできません。

表または視点のデータベースからの削除

表または視点をデータベースから削除するには、SQL DROP TABLE ステートメントまたは QMF ERASE コマンド 使用します。これを削除できるのは、表の作成者か、または DBA 権限を持っている場合だけです。

表 SYSIBM.SYSTABLES の表を定義する行を削除すると、その表に関連する視点、同義語、および索引もすべて削除されます。データベースから表を除去する前に、その表に依存する (たとえば、コマンド同義語および機能キーの定義に関して) ユーザーが他にいないことを確認してください。

表の消去の詳細については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

ローカルで定義された日付 / 時刻フォーマットのサポート

CICS ユーザーへの注

CICS では、ローカル定義の日付 / 時刻フォーマットはサポートされていません。

ローカル・フォーマットを定義するには、インストール先システムで 2 つのフォーマット設定ルーチンを作成します。そのうちの 1 つは DSNXVDTX という名前で、日付のフォーマット設定を行います。もう 1 つは DSNXVTMX という名前で、時刻のフォーマット設定を行います。これらのルーチンを作成するのは、DB2 管理タスクです。これをユーザーが行う必要がある場合は、*DB2 UDB (OS/390 版) 管理の手引き* の、ローカルで定義されたフォーマットに関する説明を参照してください。

フォーマットの指定

報告書を作成する場合、ユーザーはローカル・フォーマットをいずれのタイプのデータ (日付の TDL、時刻の TTL) についても指定できます。QMF は適切なルーチンを呼び出してフォーマット設定を行います。QMF が DSNXVTMX と DSNXVDTX の両方をロードできるようにしておく必要があります。

編集ルーチンを使用可能に

編集ルーチンは、これらのロード・ライブラリーをユーザーの JCL の STEPLIB 連結に入れることにより使用可能になります。DB2 プログラム・ライブラリーよりも前に、このライブラリーを検索することが必要です。プログ

QMF サポートの確立

ラム・ライブラリーが最初に検索されると、QMF は DB2 ライブラリーから IBM 提供の 2 つの「スタブ」をロードして使用します。これらのスタブは、ローカル・フォーマットが定義されていない場合に使用されることを意味し、フォーマット設定を行うものではありません。たとえば、フォーマット設定ルーチンがライブラリー XYZ.FORMAT に入っているものとします。図70 の STEPLIB ステートメントにこのライブラリーが正しく入っています。ここで、DB2 プログラム・ライブラリーは DSN230.SDSQLOAD です。

```
//STEPLIB DD DSN=ISP.V2R2M0.ISPLOAD,DISP=SHR
//        DD DSN=ISR.V2R2M0.ISRLOAD,DISP=SHR
//        DD DSN=QMF710.SDSQLOAD,DISP=SHR
//        DD DSN=XYZ.FORMAT,DISP=SHR          (local formatting library)
//        DD DSN=DSN230.DSNLOAD,DISP=SHR     (DB2 program library)
//        DD DSN=GDDM.OSPID.GDDMLOAD,DISP=SHR
```

図 70. 編集ルーチンを使用可能に

DXT エンド・ユーザー・ダイアログへのアクセス (ISPF のみ)

QMF の EXTRACT コマンドは、IBM のデータ抽出プログラム (DXT) エンド・ユーザー・ダイアログにアクセスします。これらのサービスを使用すれば、ユーザーは多くの異なるソースからデータを抽出し、そのデータを DB2 の表にロードすることができます。データ・ソースとして使用できるものには、IMSTM、VSAM、物理順次データ・セット、および他の DB2 システムからの表が含まれます。

EXTRACT コマンドのサポートを計画する場合は、次のことを確認する必要があります。

- バージョン 2 リリース 5 の DXT ダイアログがインストール先システムで稼働する。
- QMF EXTRACT コマンドを使用する可能性があるユーザーは、すべてが DXT ダイアログに登録されており、その使用について教育されている。

DXT の詳細については、625ページの『参考文献』の参考文献にリストされた DXT の資料を参照してください。

表にデータをロードする他の手段としては、データの順次ソースでは DB2 ローダーを使用する方法があります。DB2 ローダーに関する詳細については、DB2 UDB (OS/390 版) SQL 解説書を参照してください。

EXTRACT コマンドのサポート

EXTRACT コマンドのサポートするには、次のことを行う必要があります。

- そのコマンドが各ユーザーにデータ・セットを割り振る。
- ユーザーがコマンドを終了した後で、これらのデータ・セットの割り振りを解除する。

データ・セットは、すべてのユーザーに共通の DXT ライブラリー、または DXT に登録された個別のユーザー用に作成されたデータ・セットにも置けません。

これらのデータ・セットについては、*Data Extract: Planning and Administration Guide for Dialogs* で説明されています。DXT ダイアログ・ユーザーを登録する場合は、この資料を必ずお読みください。登録しない場合、この処理に関しては、以下の説明だけをお読みください。

リソースの割り振り

QMF でサポートできるのは、英語、漢字、および英大文字 (UCF) の DXT ダイアログです。必要な DXT データ・セットはそれぞれ異なりますが、いずれも ISPF LIBDEF ステートメントで割り振ることができます (LIBDEF の使用に関する詳細については、302ページの『CLIST の使用によるリソースの割り振りおよび割り振り解除』を参照してください)。

別の方式を選択して割り振りを行う場合は、CLIST の変更に関する以下のトピックはとばして構いません。CLIST が変更されなければ、選択した方式の妨げになることはありません。

DXT データ・セットの割り振り

302ページの表42 には、DXT バージョン 2 リリース 5 のいくつかのダイアログに必要なデータ・セットが示されています。この表は、データ・セットおよび関連する DD 名を識別しています。所定の DD 名のいずれの場合も、この表に表示してあるデータ・セットは、すでにその DD 名に関して割り振られているデータ・セットがあれば、それ以外に追加されるものです。

表に示してある名前は、DXT で提供されるデフォルトの名前です。これらのデータ・セットについては、インストール先システムで別の名前を使用することができます。

この表で、*n* はそれぞれ言語キーを示します。DXT ダイアログでは、言語キーは E (英語)、K (漢字)、および U (英大文字) です。

QMF サポートの確立

表 42. DXT バージョン 2.5 で必要なデータ・セット

DDNAME	デフォルト・データ・セット名
ISPLLIB	DXT250.DVRLOAD
ISPPLIB	DXT250.DVRPLIBn
ISPMLIB	DXT250.DVRMLIBn
ISPSLIB	userid.DXT250.DVRJEDIn DXT250.DVRSLIBn
ISPTLIB	userid.DXT250.DVRTLIBn DXT250.DVRTADMn
ISPTABL	userid.DXT250.DVRTLIBn
DVRDJEDI	userid.DXT250.DVRJEDIn または DXT250.DVRJEDIn (注を参照)
DVRDJEDO	userid.DXT250.DVRJEDIn
DVRDIMEX	userid.DXT250.DVRIMEXn
DVREUADD	DXT250.DVRTADMn
DVRSTABL	DXT250.DVRTLIBn (注を参照)

注: ライブラリー DXT250.DVRTLIBn は、インストール先システムで DXT ダイアログのオブジェクト共用機能を使用する場合だけ適用されます。

CLIST の使用によるリソースの割り振りおよび割り振り解除

重要: ISPF LIBDEF ステートメントについて精通していない場合は、これ以降を読む前に、対話式システム生産性向上機能 対話管理サービスの解説を参照してください。

必要なデータ・セットを割り振るには、次のいずれかを行います。

- ユーザーの TSO ログオン・プロシージャーに JCL を追加する。
- IBM 提供の 2 つの CLIST を使用する。

QMF は、EXTRACT コマンドを出す直前に一方の CLIST を呼び出し、コマンドが実行された直後にもう一方の CLIST を呼び出します。適切な変更を加えれば、最初の CLIST で追加のリソースを割り振り、2 番目の CLIST で追加のリソースを割り振り解除できます。304 ページの図 71 に、必要な割り振りを行う場合に使用できる CLIST を示します。ユーザーの TSO ログオン・プロシージャーに JCL を追加する方式よりも、この方式の方が優れています。この方式は、DXT データ・セットの割り振りを、それが使用可能な時だけに限ることができるからです。

割り振り CLIST の準備

この CLIST は、ライブラリー QMF710.SDSQCLTE のメンバー DSQABX1L です。QMF は、ユーザーが EXTRACT コマンドを出すと、常に ISPF

SELECT サービスを介してこの CLIST を呼び出します。割り振りに先立って、DSQABX1L に対して次の変更を加えなければならない場合があります。

1. PROC ステートメントを変更する。

元の PROC ステートメントは、次のとおりです。

```
PROC 0 DXTPRE(DXT250) LKEY(E) OBJSHR(NO)
```

QMF は CLIST にパラメーターを渡さないため、次の 3 つのキーワード・パラメーターの値が正しいことを確認しておく必要があります。

DXTPRE

DXT ライブラリーの接頭部を識別します。元の値 DXT250 は表42に示されています。

LKEY 言語環境を識別します。言語キー (301 ページ参照) が含まれています。元の値 E は英語環境を指定します。

OBJSHR

YES でも NO でも構いません。元の値 NO は、DXT オブジェクト共用が無効であることを示します。

2. 3 番目の実行可能ステートメントを除去する。

これはステートメント EXIT CODE(0) (CLIST 内の最後のステートメント) です。EXTRACT コマンドをサポートしないか、または別の方法で割り振りを行う場合は、CLIST が何も行わないことが、これで確実になります。

3. 必要に応じて、コードを変更する。

304ページの図71 に示すのは、CLIST がその LIBDEF ステートメントのデータ・セット名を生成する方法です。これらのデータ・セット名は、DXT V2R5 ダイアログではデフォルトです。必要に応じて、コードを変更して、インストール先システムで使用する名前を作成しますが、ロジックの変更または割り振りが正常に行われなかった場合のリターン・コードの変更を行ってはなりません。

QMF サポートの確立

```
PROC 0 DXTPRE(DXT240) LKEY(E) OBJSHR(NO) 00001000
CONTROL NOFLUSH NOPROMPT NOMSG 00002000
EXIT CODE(0) /* EXIT IF ALREADY ALLOCATED */ 00003000
/*****/ 00004000
/* */ 00005000
/* CLIST NAME: DSQABX1L */ 00006000
/* */ 00007000
/* DESCRIPTIVE NAME: DXT/END USER DIALOGS LIBRARY ALLOCATIONS */ 00008000
/* CLIST FOR THE QMF-DXT BRIDGE */ 00009000
/* */ 00010000
/* COPYRIGHT: 5645-DB2, 5648-A70 (C) COPYRIGHT IBM CORP. */ 00011000
/* 1982, 1998 */ 00012000
/* (Published) */ 00013000
/* LICENSED MATERIAL - PROGRAM PROPERTY OF IBM */ 00014000
/* ALL RIGHTS RESERVED */ 00015000
/* U.S. GOVERNMENT USERS RESTRICTED RIGHTS */ 00016000
/* - USE, DUPLICATION OR DISCLOSURE RESTRICTED BY */ 00017000
/* GSA ADP SCHEDULE CONTRACT WITH IBM CORP. */ 00018000
/* */ 00019000
/* STATUS: VERSION 7 RELEASE 1 LEVEL 0 */ 00020000
/* */ 00021000
/* FUNCTION: */ 00022000
/* THIS CLIST IS CALLED PRIOR TO CALLING THE DXT PRODUCT. THIS */ 00023000
/* CLIST ALLOWS THE USER TO ALLOCATE DXT LIBRARIES PRIOR TO */ 00024000
/* STARTING THE DXT PRODUCT. IF YOU ALLOCATED DXT LIBRARIES */ 00025000
/* PRIOR TO STARTING QMF YOU SHOULD NOT HAVE TO MODIFY THIS */ 00026000
/* CLIST. IN WHICH CASE THE CLIST SIMPLY EXITS WITH A ZERO */ 00027000
/* RETURN CODE. */ 00028000
/* */ 00029000
/* IF YOU DID NOT ALLOCATE DXT LIBRARIES PRIOR TO STARTING */ 00030000
/* THE QMF PRODUCT YOU WILL NEED TO ALLOCATE THEM USING THIS */ 00031000
/* CLIST. IF YOU ALLOCATE DXT LIBRARIES USING THIS CLIST YOU */ 00032000
/* WILL NEED TO CHANGE CLIST "DSQABX1F" WHICH IS EXECUTED */ 00033000
/* UPON COMPLETION OF THE DXT PRODUCT. */ 00034000
/* */ 00035000
/* INPUT: DXTPRE - CONTAINS THE PREFIX NAME FOR DXT LIBRARIES. */ 00036000
/* THIS PARAMETER IS NOT PASSED TO THIS CLIST */ 00037000
/* BY QMF. IT MUST BE SET IN THE PROC STATEMENT. */ 00038000
/* */ 00039000
/* LKEY - CONTAINS THE LANGUAGE KEY THAT IS USED AS */ 00040000
/* A SUFFIX FOR DXT LIBRARY NAMES. THE LKEY OF */ 00041000
/* "E" IS USED FOR ENGLISH. */ 00042000
/* THIS PARAMETER IS NOT PASSED TO THIS CLIST */ 00043000
/* BY QMF. IT MUST BE SET IN THE PROC STATEMENT. */ 00044000
/* */ 00045000
```

図 71. DXT データ・セットを割り振るための CLIST (DSQABX1L) (1/3)

```

/*          OBJSHR - ENTER VALUE "YES" OR "NO". IF YOU HAVE TAKEN */ 00046000
/*          ADVANTAGE OF THE DXT DIALOGS OBJECT SHARING */ 00047000
/*          CAPABILITY SPECIFY THE VALUE "YES". IF YOU */ 00048000
/*          ARE NOT USING OBJECT SHARING SPECIFY THE */ 00049000
/*          VALUE "NO". */ 00050000
/*          THIS PARAMETER IS NOT PASSED TO THIS CLIST */ 00051000
/*          BY QMF. IT MUST BE SET IN THE PROC STATEMENT.*/ 00052000
/*          */ 00053000
/*          OUTPUT: NONE */ 00054000
/*          */ 00055000
/*          EXIT CONDITIONS: NONE */ 00056000
/*          */ 00057000
/*          ABEND CODE: VALUE - NONE */ 00058000
/*          */ 00059000
/*          EXTERNAL REFERENCES: */ 00060000
/*          ROUTINES: NONE */ 00061000
/*          DATA AREAS: NONE */ 00062000
/*          */ 00063000
/*          CHANGE ACTIVITY: NONE */ 00064000
/*          */ 00065000
/*          */ 00066000
/*-END-OF-SPECIFICATION-***** */ 00067000
ISPEXEC CONTROL ERRORS RETURN /* RETURN ISPF ERRORS TO CLIST*/ 00068000
      IF &LASTCC ^= 0 THEN EXIT CODE(1) 00069000
/*          */ 00070000
/* ALLOCATE DXT LIBRARIES USING ISPF LIBDEF. DXT LIBRARIES ARE NAMED */ 00071000
/* "&DXTPRE.NAME" AND USER LIBRARIES ARE NAMED "&USERID.&DXPRE.NAME" */ 00072000
/*          */ 00073000
SET &USERID = &SYSUID /* SET CURRENT USERID */ 00074000
SET &LIBS = &STR('&DXTPRE..DVRLOAD') 00075000
ISPEXEC LIBDEF ISPLLIB DATASET ID(&STR(&LIBS)) 00076000
      IF &LASTCC ^= 0 THEN EXIT CODE(2) 00077000
SET &LIBS = &STR('&DXTPRE..DVRPLIB&LKEY. ') 00078000
ISPEXEC LIBDEF ISPLLIB DATASET ID(&STR(&LIBS)) 00079000
      IF &LASTCC ^= 0 THEN EXIT CODE(3) 00080000
SET &LIBS = &STR('&DXTPRE..DVRMLIB&LKEY. ') 00081000
ISPEXEC LIBDEF ISPLLIB DATASET ID(&STR(&LIBS)) 00082000
      IF &LASTCC ^= 0 THEN EXIT CODE(4) 00083000
SET &LIBS = &STR('&USERID..DXTPRE..DVRJEDI&LKEY. ') 00084000
SET &LIBS = &STR(&LIBS , '&DXTPRE..DVRSLIB&LKEY. ') 00085000
ISPEXEC LIBDEF ISPLLIB DATASET ID(&STR(&LIBS)) 00086000
      IF &LASTCC ^= 0 THEN EXIT CODE(5) 00087000
SET &LIBS = &STR('&USERID..DXTPRE..DVRTLIB&LKEY. ') 00088000
SET &LIBS = &STR(&LIBS , '&DXTPRE..DVRTADM&LKEY. ') 00089000
ISPEXEC LIBDEF ISPTLIB DATASET ID(&STR(&LIBS)) 00090000
      IF &LASTCC ^= 0 THEN EXIT CODE(6) 00091000
ISPEXEC LIBDEF ISPTABL DATASET ID('&USERID..DXTPRE..DVRTLIB&LKEY. ') 00092000
      IF &LASTCC ^= 0 THEN EXIT CODE(7) 00093000

```

図 71. DXT データ・セットを割り振るための CLIST (DSQABX1L) (2/3)

QMF サポートの確立

```
IF &OBJSHR = NO THEN +                                00094000
DO                                                       00095000
  ISPEXEC LIBDEF DVRDJEDI DATASET ID('&USERID..&DXTPRE..DVRJEDI&LKEY.')
```

00096000
IF &LASTCC ^= 0 THEN EXIT CODE(8) 00097000
END 00098000
ELSE + 00099000
DO 00100000
 ISPEXEC LIBDEF DVRDJEDI DATASET ID('&DXTPRE..DVRJEDI&LKEY.')

00101000
IF &LASTCC ^= 0 THEN EXIT CODE(8) 00102000
END 00103000
ISPEXEC LIBDEF DVRDJEDO DATASET ID('&USERID..&DXTPRE..DVRJEDI&LKEY.')

00104000
IF &LASTCC ^= 0 THEN EXIT CODE(9) 00105000
ISPEXEC LIBDEF DVRDIMEX DATASET ID('&USERID..&DXTPRE..DVRIMEX&LKEY.')

00106000
IF &LASTCC ^= 0 THEN EXIT CODE(10) 00107000
ISPEXEC LIBDEF DVREUADD DATASET ID('&DXTPRE..DVRTADM&LKEY.')

00108000
IF &LASTCC ^= 0 THEN EXIT CODE(11) 00109000
IF &OBJSHR = YES THEN + 00110000
DO 00111000
 ISPEXEC LIBDEF DVRSTABL DATASET ID('&DXTPRE..DVRTLIB&LKEY.')

00112000
IF &LASTCC ^= 0 THEN EXIT CODE(12) 00113000
END 00114000
EXIT CODE(0) 00115000

図 71. DXT データ・セットを割り振るための CLIST (DSQABX1L) (3/3)

割り振り解除 CLIST の準備

309ページの図72 の CLIST は、ライブラリー QMF710.SDSQCLTE のメンバー DSQABX1F です。QMF は、EXTRACT コマンドが実行された後で、ISPF SELECT サービスによってこの CLIST を呼び出します。割り振り解除に先立って、DSQABX1F に対して次の変更を加えなければならない場合があります。

1. PROC ステートメントを変更する。

元の PROC ステートメントは、次のとおりです。

```
PROC 0 QMFPRE(QMF710) LKEY(E) OBJSHR(NO)
```

QMF は CLIST にパラメーターを渡さないため、次の 3 つのキーワード・パラメーターの値が正しいことを確認しておく必要があります。

QMFPRE

CLIST の変数 &QMFPRE の値を設定します。この値は、バージョン 3 リリース 1 の多くのデータ・セット名内の最初の修飾子です。元の値 QMF710 は、QMF バージョン 7 のインストール先システムのデフォルトです。この修飾子がインストール先システムで異なる場合は、QMFPRE の値の変更を必要とする場合があります。これを行う必要があるかどうかは、次のステップで説明するデータ・セットの割り振り解除を行う方法によります。

LKEY 言語環境を識別します。割り振り CLIST で使用した LKEY の値と同じ値を使用します (303 ページ参照)。

OBJSHR

DXT オブジェクト共有が有効かどうかを示します。割り振り CLIST で使用した OBJSHR の値と同じ値を使用します。

2. 3 番目の実行可能ステートメントを除去する。

これはステートメント EXIT CODE(0) です。EXTRACT コマンドをサポートしないか、または別の方法で割り振りを行う場合は、CLIST が何も行わないことが、これで確実にあります。

3. 必要に応じて、ブランチ・ステートメントを変更し、コードを変更する。

PROLOG 内のコメントの後の 3 番目のステートメントは、ブランチ・ステートメント GOTO A です。このステートメントの後に 2 つのコード・ブロックのセクション A とセクション B があります。両方のブロックは、DXT データ・セットの割り振り解除を行います。その方法が異なります。セクション B を選択して割り振り解除を行う場合は、ブランチ・ステートメントを GOTO B に変更する必要があります。

セクション A は、割り振り CLIST が出した LIBDEF ステートメントをすべて無効にします。これは DXT データ・セットを割り振り解除しますが、ユーザーが EXTRACT コマンドの終了後に必要とするデータ・セットまで割り振り解除する可能性があります。このようなデータ・セットは、ユーザーが EXTRACT コマンドを出す前に、LIBDEF ステートメントによって割り振られたものです。EXTRACT コマンドより前に出された LIBDEF ステートメントで割り振られたデータ・セットをユーザーが必要とする場合は、ブランチ・ステートメントを変更して、セクション B が呼び出されるようにします。

この記述に該当するデータ・セットには、その DD 名の LIBDEF ステートメントが、この CLIST および対応する割り振り CLIST に現れていなければなりません。たとえば、次の LIBDEF ステートメントは、DD 名 ISPLLIB にすでに割り振られているライブラリーにパネル・ライブラリー ABC.XYZ を追加します。

```
ISPEXEC LIBDEF ISPLLIB DATASET(ABC.XYZ)
```

割り振り CLIST では、CLIST がその LIBDEF ステートメントを ISPLLIB に対して実行すると、この割り振りは消えます。これを復元するには、割り振り解除 CLIST 内で元の LIBDEF ステートメントを出し直す必要があります。復元したい割り振りがこれだけの場合は、セクション A で ISPLLIB に対する LIBDEF ステートメントを変更するだけで、セクション A を使用できます。

QMF サポートの確立

セクション B の LIBDEF ステートメントは、QMF ライブラリーを ISPF DD 名に再割り振りします。セクション B を選択した場合は、これらのステートメントを必要に応じて変更します。

LIBDEF ステートメントは、TSO ログオン・プロシージャまたは TSO ALLOCATE ステートメントによって割り振られたデータ・セットを割り振り解除できません。したがって、QMF セッションに必要なデータ・セットが、すべてこの方法で割り振られている場合は、いつでもセクション A を使用することができます。

```

PROC 0 QMFPRE(QMF710) LKEY(E) OBJSHR(NO)                                00001000
CONTROL NOFLUSH NOPROMPT NOMSG                                         00002000
EXIT CODE(0) /* EXIT IF ALREADY ALLOCATED */                             00003000
/*****                                                                    00004000
/*                                                                    */ 00005000
/* CLIST NAME: DSQABX1F */ 00006000
/*                                                                    */ 00007000
/* DESCRIPTIVE NAME: DXT/END USER DIALOGS LIBRARY FREE */ 00008000
/* CLIST FOR THE QMF-DXT BRIDGE */ 00009000
/*                                                                    */ 00010000
/* COPYRIGHT: LICENSED MATERIALS - PROPERTY OF IBM */ 00011000
/* 5675-DB2, 5697-F42 (C) COPYRIGHT IBM CORP. */ 00012000
/* 1982, 2000. (PUBLISHED) */ 00013000
/* ALL RIGHTS RESERVED. */ 00014000
/* US GOVERNMENT USERS RESTRICTED RIGHTS - */ 00015000
/* USE, DUPLICATION OR DISCLOSURE RESTRICTED */ 00016000
/* BY GSA ADP SCHEDULE CONTRACT WITH IBM CORP. */ 00017000
/*                                                                    */ 00018000
/* STATUS: VERSION 7 RELEASE 1 LEVEL 0 */ 00019000
/*                                                                    */ 00020000
/* FUNCTION: */ 00021000
/* THIS CLIST IS CALLED WHEN THE DXT PRODUCT HAS ENDED. THIS */ 00022000
/* CLIST IS USED TO FREE ANY ALLOCATIONS MADE BY THE CLIST */ 00023000
/* "DSQABX1L" AND REALLOCATE QMF LIBRARIES IF THE "LIBDEF" */ 00024000
/* FUNCTION WAS USED TO ALLOCATE DXT PRODUCT LIBRARIES. */ 00025000
/*                                                                    */ 00026000
/* IF YOU ALLOCATED ALL OF YOUR DXT LIBRARIES BEFORE YOU */ 00027000
/* STARTED QMF OR ISPF, YOU SHOULD NOT MODIFY THIS CLIST. */ 00028000
/* THE CLIST THAT IS DISTRIBUTED BY THE QMF PRODUCT EXITS */ 00029000
/* AND PERFORMS NO LIBRARY ALLOCATION. */ 00030000
/*                                                                    */ 00031000
/* IF YOU DID NOT ALLOCATE YOUR QMF LIBRARIES BY USING THE */ 00032000
/* "LIBDEF" FUNCTION, BUT DID ALLOCATE DXT LIBRARIES USING */ 00033000
/* "DSQABX1L", YOU WILL NEED TO FREE THE DXT LIBRARY */ 00034000
/* ALLOCATIONS. SEE SECTION "A" */ 00035000
/*                                                                    */ 00036000
/* IF YOU ALLOCATED QMF LIBRARIES USING "LIBDEF", YOU MUST */ 00037000
/* USE THIS CLIST TO RE-ALLOCATE THE QMF LIBRARIES BECAUSE */ 00038000
/* THEY WERE REPLACED BY DXT LIBRARY DEFINITIONS WHEN CLIST */ 00039000
/* "DSQABX1L" WAS EXECUTED. SEE SECTION "B" */ 00040000
/*                                                                    */ 00041000
/* INPUT QMFPRE - CONTAINS THE PREFIX NAME FOR QMF LIBRARIES. */ 00042000
/* THIS PARAMETER IS NOT PASSED TO THIS CLIST */ 00043000
/* BY QMF. IT MUST BE SET IN THE PROC STATEMENT. */ 00044000
/*                                                                    */ 00045000
/* LKEY - CONTAINS THE LANGUAGE KEY THAT IS USED AS */ 00046000

```

図 72. DXT データ・セットを割り振り解除する (DSQABX1F) (1/3)

QMF サポートの確立

```

/*          A SUFFIX FOR QMF LIBRARY NAMES. THE LKEY OF */ 00047000
/*          "E" IS USED FOR ENGLISH.                    */ 00048000
/*          THIS PARAMETER IS NOT PASSED TO THIS CLIST */ 00049000
/*          BY QMF. IT MUST BE SET IN THE PROC STATEMENT.*/ 00050000
/*          */                                           */ 00051000
/*          OBJSHR - ENTER VALUE "YES" OR "NO". IF YOU HAVE TAKEN */ 00052000
/*          ADVANTAGE OF THE DXT DIALOGS OBJECT SHARING */ 00053000
/*          CAPABILITY SPECIFY THE VALUE "YES". IF YOU */ 00054000
/*          ARE NOT USING OBJECT SHARING SPECIFY THE */ 00055000
/*          VALUE "NO".                                   */ 00056000
/*          THIS PARAMETER IS NOT PASSED TO THIS CLIST */ 00057000
/*          BY QMF. IT MUST BE SET IN THE PROC STATEMENT.*/ 00058000
/*          */                                           */ 00059000
/*          OUTPUT: NONE                                  */ 00060000
/*          */                                           */ 00061000
/*          EXIT CONDITIONS: NONE                        */ 00062000
/*          */                                           */ 00063000
/*          ABEND CODE: VALUE - NONE                    */ 00064000
/*          */                                           */ 00065000
/*          EXTERNAL REFERENCES:                        */ 00066000
/*          ROUTINES: NONE                              */ 00067000
/*          DATA AREAS: NONE                           */ 00068000
/*          */                                           */ 00069000
/*          CHANGE ACTIVITY:                            */ 00070000
/*          */                                           */ 00071000
/*          CHANGE DATE:                                */ 00072000
/*          */                                           */ 00073000
/*          */                                           */ 00074000
/*          */                                           */ 00075000
/*          */                                           */ 00076000
/*          */                                           */ 00077000
/*          */                                           */ 00078000
/*          */                                           */ 00079000
/*          */                                           */ 00080000
/*          */                                           */ 00081000
/*          */                                           */ 00082000
/*          */                                           */ 00083000
/*          */                                           */ 00084000
/*          */                                           */ 00085000
/*          */                                           */ 00086000
/*          */                                           */ 00087000
/*          */                                           */ 00088000
/*          */                                           */ 00089000
/*          */                                           */ 00090000
/*          */                                           */ 00091000
/*          */                                           */ 00092000

```

図 72. DXT データ・セットを割り振り解除する (DSQABX1F) (2/3)


```

/*****/ 00093000
/* SECTION B: ALLOCATE QMF LIBRARIES USING ISPF LIBDEF */ 00094000
/*****/ 00095000
B: SET &LIBS = &STR('&QMFPRE..SDSQLOAD') 00096000
   ISPEXEC LIBDEF ISPLLIB DATASET ID(&STR(&LIBS)) 00097000
   IF &LASTCC ^= 0 THEN EXIT CODE(2) 00098000
   SET &LIBS = &STR('&QMFPRE..SDSQPLB&LKEY.') 00099000
   ISPEXEC LIBDEF ISPLLIB DATASET ID(&STR(&LIBS)) 00100000
   IF &LASTCC ^= 0 THEN EXIT CODE(3) 00101000
   SET &LIBS = &STR('&QMFPRE..SDSQMLB&LKEY.') 00102000
   ISPEXEC LIBDEF ISPLLIB DATASET ID(&STR(&LIBS)) 00103000
   IF &LASTCC ^= 0 THEN EXIT CODE(4) 00104000
   SET &LIBS = &STR('&QMFPRE..SDSQSLBE') 00105000
   ISPEXEC LIBDEF ISPLLIB DATASET ID(&STR(&LIBS)) 00106000
   IF &LASTCC ^= 0 THEN EXIT CODE(5) 00107000
   ISPEXEC LIBDEF DVRDJEDI 00108000
   ISPEXEC LIBDEF DVRDJEDO 00109000
   ISPEXEC LIBDEF DVRDIMEX 00110000
   ISPEXEC LIBDEF DVREUADD 00111000
   IF &OBJSHR = YES THEN + 00112000
     ISPEXEC LIBDEF DVRSTABL 00113000
   EXIT CODE(0) 00114000

```

図 72. DXT データ・セットを割り振り解除する (DSQABXIF) (3/3)

ユーザー用文書編集インターフェースのカスタマイズ

文書インターフェースは、ISPF/PDF および PS/TSO 編集プログラム用の IBM 提供のマクロです。このマクロを使用すると、QMF の外側で操作しているユーザーが QMF セッションを開始できます。そのセッションでは、ユーザーは、編集集中の文書に QMF 報告書を挿入することができます。報告書は、編集セッションの開始前に作成することができます。さらに重要なのは、GETQMF マクロが開始された QMF セッション中の、このマクロが出された時点で、ユーザーは報告書を作成することができるということです。

ユーザーがこのマクロを使用できる前に、次のことを行っておく必要があります。

- 各ユーザーに適正な QMF リソースを確保させる。
リソースとは QMF ライブラリーのことです。TSO ログオン・プロシージャの例では、これらには次の形の名前が付けられています。

```
QMF710.DSQ...
```

必要なリソースの説明については、177ページの『第12章 必要な記憶域』を参照してください。

QMF サポートの確立

ISPF/PDF および PS/TSO 編集プログラムは、これらのリソースがなくても操作できますが、文書インターフェースは QMF セッションを正常に開始することはできません。

- 特定の文書インターフェース・コンポーネントを変更する。

このような変更の中には必須のものもあれば、オプションのものもあります。この節では、必須およびオプションの両方の変更について説明します。文書インターフェースを使用するには、*QMF 使用の手引き* も参照する必要があります。

NLF を使用する場合: 該当する NLF 版の文書インターフェースもカスタマイズすることができます。

アプリケーションの変更

アプリケーションの変更は、そのコンポーネントの 1 つまたは複数を変更することによって行います。変更できるコンポーネントは、特定の QMF ライブラリーのメンバーです。

- CLIST およびマクロは QMF710.SDSQCLTE のメンバーです。
- それ以外のコンポーネントは QMF710.SDSQSAPE のメンバーです。

文書インターフェース・マクロの名前変更

マクロ・コンポーネント DSQAED1P は、ユーザーが文書インターフェースを使用するために呼び出すマクロです。

このマクロを使用するには、次のことを行います。

- マクロのコピーの名前を変更する。変更後の望ましい名前は GETQMF です。これは本書および *QMF 使用の手引き* で、このマクロに使用している名前です。
- 名前を変更したコピーを QMF710.SDSQCLTE、つまり元のマクロが入っているライブラリーに入れる。

NLF を使用する場合: メインのマクロは、ライブラリー QMF710.DSQCLSTn のメンバー DSQA n DIP です。これは英語のメインのマクロと同様に、名前を変更しても、他のコンポーネントに影響が生じることはありません。ユーザーの JCL が英語環境と NLF 環境の両方をサポートする場合は、GETQMF 以外の名前を選択してください。たとえば、GETQMF n に変更してください。

Q.DSQAED1S プロシーチャーをデータベースに入れる

Q.DSQAED1S プロシーチャーは、ライブラリー QMF710.SDSQSAPE のメンバー DSQAED1S に入っています。このプロシーチャーをデータベースに入れる処理は、DB2 のバージョンによって異なります。

ユーザー Q の場合は、次の QMF コマンドを入力すれば、Q.DSQAED1S をデータベースに容易に入れることができます。

```
IMPORT PROC DSQAED1S FROM 'QMF710.SDSQSAPE(DSQAED1S)' (SHARE=YES
```

ユーザー Q ではないが、次のいずれかを持っている場合があります。

- SYSADM 権限
- SYSCTRL 権限
- 2 次許可 ID の 1 つとしての Q

次の QMF コマンドを照会パネルから入力することによって、同じように DSQAED1S をデータベースに容易に入れることができます。

```
SET CURRENT SQLID = 'Q'
IMPORT PROC DSQAED1S FROM 'QMF710.SDSQSAPE(DSQAED1S)' (SHARE=YES
```

Q 以外のユーザーで、SYSADM (または SYSCTRL) 権限も、ユーザーの 2 次許可 ID としての Q も持っていない場合は、『Q への所有権の転送』で説明するプロシーチャーを使用する必要があります。

NLF を使用する場合: ライブラリー QMF710.SDSQSAPn のメンバー DSQAnD1S の NLID を変更します。

Q への所有権の転送

ユーザー Q として QMF を使用できない場合でも、前項に示したコマンドを出すことはできます。ただし、まず最初にプロシーチャーの所有権を自分の許可 ID から Q に転送しておく必要があります。これは次のようにして行うことができます。

1. 次の照会を作成する。

```
UPDATE Q.&T
  SET OWNER = 'Q'
  WHERE NAME = &N AND OWNER = USER
```

2. 次のコマンドを実行する。

```
RUN QUERY ( &T=OBJECT_DIRECTORY, &N='DSQAED1S'
RUN QUERY ( &T=OBJECT_DATA, &N='DSQAED1S'
RUN QUERY ( &T=OBJECT_REMARKS, &N='DSQAED1S'
```

各コマンドはそれぞれ表 Q.OBJECT の 1 つを更新し、これらの表に対する UPDATE 特権を要求します。

QMF サポートの確立

照会が正常に実行されない場合、Q.DSQAED1S という名前のオブジェクトがすでにデータベース内に存在する可能性があります。このような場合は、そのオブジェクトの名前を変更するか、または削除してから、所有権の転送を再度試みます。以下の 2 つの照会の 1 つによって、オブジェクトの名前の変更または削除ができます。いずれの照会を選択した場合も、それに対して 3 つの RUN QUERY コマンドを実行する必要があります。

- オブジェクトの名前を変更するには、以下の照会を使用してオブジェクトの新規の名前で *newname* を置き換える。

```
UPDATE Q.&T
  SET NAME = 'newname'
  WHERE NAME = &N AND OWNER = 'Q'
```

- オブジェクトを削除するには、次の照会を使用する。

```
DELETE FROM Q.&T
  WHERE NAME = &N AND OWNER = 'Q'
```

データ・コンポーネントの変更

データ・コンポーネントは 5 つあり、すべてライブラリー

QMF710.SDSQSAPE 内に入っています。CLIST やマクロとは異なり、これらのコンポーネントにはロジックも実行可能コマンドも入っていません。その代わり、これらのコンポーネントにはメッセージやユーザーの報告書に現れる情報が入っています。

文書インターフェースは、これらのコンポーネントが単一のライブラリーに入っていることを前提としているため、以下のいずれかの方法で変更することができます。

- 変更後のコンポーネントを QMF710.SDSQSAPE に保存することができる。
この場合は、元のコンポーネントの名前を変更し、変更後のコンポーネントに元の名前を付けます。
- 変更後のコンポーネントを新規のライブラリーに入れることができる。
この場合は、他のすべてのデータ・コンポーネントを、古いライブラリーから新規のライブラリーにコピーする必要があります。

2 番目の方法を使用する場合は、マクロ DSQAED1P (316ページの『DSQAnD1P の変更』参照) に変更を加える必要があります。

メッセージ・コンポーネント

5 つのデータ・コンポーネントの 1 つには DSQAED0L という名前が付いています。このコンポーネントには次のものが入っています。

- ユーザーによる文書インターフェースの操作中に、ユーザーの画面に表示される可能性があるメッセージ
- 特定の QMF コマンドのキーワード

このコンポーネントは変更 しないで ください。

NLF を使用する場合: ライブラリー QMF710.DSQSAMPn のメンバー DSQAnD0L の NLID を変更します。

DCF コンポーネント

DCF (文書構成プログラム) は IBM ライセンス・テキスト処理システムの 1 つで、印刷文書の作成でコンピューターの使用をサポートするものです。

インストール先システムで DCF を使用する場合は、残りの 4 つの DCF コンポーネントを変更したい場合があります。DCF の詳細については、*Document Composition Facility: SCRIPT/VS Text Programmer's Guide* を参照してください。

ユーザーは、文書インターフェースに、現行文書が DCF によるフォーマット設定であることを示すことができます。これに対する応答として、文書インターフェースは、ユーザーの挿入報告書に DCF 制御ステートメントを追加します。これらのステートメントが現れる場合、それは DCF コンポーネントのいずれか 1 つの中のすべてのレコードで構成されています。コンポーネントの中のレコードは、一部または全部を変更することができます。コンポーネント、および提供されるものは、次のとおりです。

DSQABD01: 報告書の直前に挿入されるステートメントを提供します。IBM 提供のコンポーネントでは、次のとおりです。

```
.* QMF Document Interface heading control:
.SA
.RH SUP
.RF SUP
.HS 0
.FS 0
.TM 0.5I
.BM 0
.DC CONT OFF
.FO OFF
```

DSQABD02: 各ページ後書きの直後に挿入されるステートメントを提供します。IBM 提供のコンポーネントでは、提供されるのは単一のステートメントで、次のとおりです。

```
.* QMF Document Interface page footing control:
```

QMF サポートの確立

DSQABD03: 各ページ・ヘッダーの直前に挿入されるステートメントを提供します。IBM 提供のコンポーネントでは、次のとおりです。

```
.PA NOSTART  
.* QMF Document Interface page heading control:
```

DSQABD04: 報告書の終わりの直後に挿入されるステートメントを提供します。IBM 提供のコンポーネントでは、次のとおりです。

```
.* QMF Document Interface footing control:  
.RE  
.* QMF REPORT END
```

CLIST およびマクロの変更

すでに述べたように、これらのコンポーネントはすべてライブラリー QMF710.SDSQCLTE 内にあります。CLIST またはマクロを変更する場合は、元のマクロではなくコピーを変更し、それを別のライブラリーに入れます。新しいライブラリーの DD ステートメントは、ユーザーの JCL の SYSPROC に関するステートメントに現れる必要があります。まだ、そこにはない場合は、QMF710.SDSQCLTE のステートメントの前に挿入します。そうでない場合は、変更したコンポーネントではなく、元のコンポーネントが使用されます。たとえば、変更したコンポーネントをライブラリー XYZ.NEWCLIST に入れると、SYSPROC の DD ステートメントは以下のようになります。

```
//SYSPROC DD DSN=SYSUT2.CLIST,DISP=SHR  
//          DD DSN=XYZ.NEWCLIST,DISP=SHR  
//          DD DSN=QMF710.SDSQCLTE,DISP=SHR
```

DSQAnD1P の変更

これは GETQMF に名前変更したマクロです。このマクロに対して、以下のことも行えます。

- 次のステートメントを変更する。

```
SET &SAMPLIB = QMF710.DSQSAMP&LANGCHAR  
SET &BASELIB = QMF710.SDSQSAPE
```

&SAMPLIB

文書インターフェースのデータ・コンポーネントが入っているライブラリーを識別します。

&BASELIB

QMF サンプル・ライブラリーを識別します。

&LANGCHAR の値が E の場合、両方の変数は同じライブラリー QMF710.SDSQSAPE を指します。ライブラリーの名前が異なる場合、割り当てられた名前 &SAMPLIB および &BASELIB を変更します。

- 次のステートメントを変更する。

```
ALLOC FI(DSQPRINT) SYSOUT RECFM(F B A) LRECL(133) BLKSIZE(1330)
```

ユーザーは、対話式 QMF セッションで文書インターフェースを呼び出すことができます。これが行われると、文書インターフェースは DSQPRINT を再割り振りできます。このステートメントは、DSQPRINT をデフォルトに復元します。これを希望しない場合は、DSQPRINT を希望する値に復元するステートメントで、このステートメントを置き換えます。

DSQABD1Q の変更

この CLIST は、文書インターフェースで開始されたセッションのデータ・セットを割り振ります。必要と思われる変更をすべて CLIST コードに行います。たとえば、インストール先システムに特有のデータ・セットに関する割り振りの追加を必要とする場合があります。

このような割り振りの一部として、GDDM データ・セットがあります。文書インターフェース自体は、これらのデータ・セットを使用しませんが、ユーザーがこの割り振りを必要とする場合があります。

変数 &LANGCHAR の値は E です。この値は、たとえば、ライブラリーに、英大文字機能アプリケーションのコンポーネントとは異なる英語コンポーネントが入っていることを示します。

LIBDEF 割り振りをサポートするには、LIBDEF サービスをアクティブ化し、必要に応じてファイル名を調整します。

```

/*****@82*/
/* Remove the Following "GOTO NOLIBDEF" statement to allocate @82*/
/* ISPF libraries using the ISPF LIBDEF service. @82*/
/*****@82*/
GOTO NOLIBDEF
/*****@82*/
/* ALLOCATE QMF ISPF LIBRARIES USING LIBDEF @82*/
/*****@82*/
SET PNAME = 'QMF710.DSQPLIB&LANGCHAR' /* ISPF Panel Library */
SET MNAME = 'QMF710.DSQMLIB&LANGCHAR' /* ISPF Message Library */
SET SNAME = 'QMF710.DSQSLIB&LANGCHAR' /* ISPF Skeleton Library */
SET LNAME = 'QMF710.SDSQLOAD' /* QMF Modules */
ISPEXEC LIBDEF ISPLIB DATASET ID(&PNAME)

```

LIBDEF をサポートするための DSQABD1P の変更

LIBDEF 機能を使用して QMF ライブラリーを割り振った場合は、LIBDEF により割り振られたライブラリーの使用を解放するために DSQABD1P を変更します。DSQABD1P 内の下記のステートメントのコメントの指定を除いてください。

QMF サポートの確立

```
/* **** */
/* FREE ISPF LIBDEFs @82*/
/* You might or might not need to free libdefs here. */
/* If you do, then remove comments from LIBDEF statements. */
/* **** */
/* ISPEXEC LIBDEF ISPLLIB DATASET ID() */
/* ISPEXEC LIBDEF ISPLMLIB DATASET ID() */
/* ISPEXEC LIBDEF ISPLSIB DATASET ID() */
/* ISPEXEC LIBDEF ISPLLIB DATASET ID() */
/* FREE FI(DSQLLIB) */
```

DSQABD1C の変更

このコンポーネントは以下の方法で変更できます。

- 次のステートメントを変更する。

```
ALLOC FI(DSQPRINT) UNIT(SYSDA) SPACE(5,2) TRACKS +
RECFM(F B A) LRECL(&PRINTREC) BLKSIZE(&EVAL(&PRINTREC*10))
```

このステートメントは、ユーザーの報告書のデータ・セットを割り振ります。次に、ユーザーは **QMF PRINT** コマンドを介してこのデータ・セットを埋めることができます。ユーザーが作成する報告書が極端に大きい場合は、このステートメントの **SPACE** オペランドを変更しなければならないことがあります。

- 次のステートメントを変更する。

```
ISPEXEC SELECT PGM(DSQQMF&LANGCHAR)
               PARM(I=&PROCNAME)
               NEWAPPL(DSQ&LANGCHAR)
```

ステートメントが現在の書式の場合、DB2 のサブシステムの名前は **DSN** で、**QMF** のアプリケーション・プランの名前は **QMF710** でなければなりません。そうでない場合は、このステートメントの **PARM** オペランドに情報を追加する必要があります。たとえば、サブシステムおよびアプリケーション・プランに **ABC** および **QMFXXX** という名前を付けます。そうすると、変更後のステートメントは次のようになります。

```
ISPEXEC SELECT PGM(DSQQMF&LANGCHAR)
               PARM(I=&PROCNAME,S=ABC,P=QMFXXX)
               NEWAPPL(DSQ&LANGCHAR)
```

変更後のステートメントは、**QMF** のプログラム・パラメーターの 2 つのデフォルト値を上書きします。

プログラム・パラメーターの説明については、199ページの『第14章 開始手順のカスタマイズ』を参照してください。

QMF EDIT コマンドのカスタマイズ

EDIT コマンドを使用すると、編集プログラムで QMF の照会およびプロシージャを変更できます。編集プログラムの 1 つとして、ISPF/PDF が使用できます (QMF が ISPF のもとで開始されている場合)。

以下の手順では、ISPF のもとで稼働する CLIST によって呼び出すことのできる編集プログラムの使用を前提としています。EDIT TABLE コマンドは、表編集機能を呼び出すもので、テキスト編集プログラムを必要としません。

EDIT コマンドで編集プログラムを使用可能にするには、以下のようになります。

1. 編集プログラムを呼び出すための CLIST を作成し、編集したいデータ・セットの名前を定位置パラメーターとして渡します。たとえば、次のコマンドで QMF はデータ・セット USERA.XYZDATA.TEXT を編集するための CLIST である XYZEDIT を呼び出します。

```
XYZEDIT 'USERA.XYZDATA.TEXT'
```

2. 編集プログラムにアクセスできるユーザーに割り振られているコマンド・ライブラリーに、この CLIST を入れます。これを、データ・セット SYSPROC の連結の一部であるライブラリーに入れます。可能な選択の 1 つは QMF ライブラリー QMF710.SDSQCLTE で、これはすべての QMF ユーザーで使用可能になります。
3. 個々のユーザーに、編集するオブジェクトのデータ・セットの割り振りおよびカタログ化を行います。このデータ・セットは、ユーザーが EDIT コマンドで編集プログラムを呼び出すたびに埋め直されます。データ・セットに以下の特性を付与します。
 - 物理順次編成 (DSORG=PS)
 - 固定長、79 バイト・レコード (LRECL=79)
 - ブロック化因数 51 (BLKSIZE=4029)
4. 各ユーザーの JCL で、そのユーザー用としてステップ 3 でカタログ化したデータ・セットを割り振ります。DD 名 DSQEDIT でこれを割り振ります。データ・セットの後処理として DISP=OLD を書き込みます。
5. ユーザーには EDIT コマンドの指定方法についての参考となります。コマンドのフォーマットは、次のとおりです。

```
EDIT yyyy (EDITOR=xxxx)
```

ここで yyyy は、PROC か QUERY のいずれかで、xxxx は、編集プログラムを呼び出すために作成された CLIST の名前です。EDIT コマンドの詳細については、*QMF 解説書* を参照してください。

QMF サポートの確立

- 異なる ISPF アプリケーション ID の QMF SQL 照会または QMF プロシージャは、QMF EDIT コマンドで編集プログラム名として EXEC または CLIST を使用することにより編集できます。

SQL 照会または QMF プロシージャを編集するために、プログラム開発機能 (PDF) 編集プログラムを指定した場合、QMF は QMF アプリケーション ID の DSQE または DSQ n (n は NLF 文字) で PDF を実行します。さらに、QMF は機能キーおよびコマンド行の場所を QMF 製品に適合するように設定します。

別の機能キーのセットを使用したり、あるいは既存の PDF マクロまたは特殊 PDF 編集プログラム画面を持つ必要がある場合、PDF 編集プログラムを DSQ* 以外のアプリケーション ID で実行することによって、それらを使用することができます。これを行うには、2 つの小さい REXX プログラムまたは CLIST を実行します。最初のプログラムは、単に 2 番目のプログラムに実行を経路指定するだけで、次に 2 番目のプログラムが、希望の機能キーを使用するかまたは編集呼び出しマクロや固有の編集パネルなどのその他の特別の設定要件を使用して、希望の ISPF アプリケーション ID で実行される編集プログラムを呼び出します。

図73 の REXX プログラム例は、QMF の開始時に DDNAME(DSQEDIT) によって定義されている編集転送データ・セットを使用して、SQL 照会または QMF プロシージャを編集する方法を示しています。この例では、PDF アプリケーション ID ISP が使用されています。

Edit Program 1 (MYEDIT)

```
/* REXX   QMF Edit program 1           */
/*       Transfer to ISP application ID */
Address ISPEXEC "SELECT CMD(MYEDIT2) NEWAPPL(ISP)"
Exit 0
```

Edit Program 2 (MYEDIT2)

```
/* REXX   QMF Edit program 2           */
/*       Invoke PDF Editor using DDNAME */
Address ISPEXEC "LMINIT DATAID(EDT) DDNAME(DSQEDIT)"
Address ISPEXEC "EDIT  DATAID("EDT")"
Address ISPEXEC "LMFREE DATAID("EDT")"
Exit 0
```

図73. 編集転送データ・セットの使用による編集

REXX プログラムは、実行前に SYSPROC か SYSEXEC のどちらかの有効な連結に割り振られている必要があります。QMF から実行するには、QMF コマンド行に次の QMF EDIT コマンドを入力します。

EDIT QUERY (E=MYEDIT)

重要: プロシージャまたは照会を編集し、結果のオブジェクトが大き過ぎて QMF の作業域に収まりきらない場合、QMF はこのオブジェクトを切り捨て、エラー・メッセージを表示します。ただし、QMF は DD 名 QMFEDIT に関連したファイルにオブジェクト全体を保管します。オブジェクトを QMF に取り込むには、ユーザーは RESET DATA コマンドを出す必要があります。この情報は、保管オブジェクトのファイル名をも含めて、この条件に関連するエラー・メッセージのメッセージ・ヘルプで提供されます。

NLF 環境での英語サポートのイネープリング

いずれの NLF にも、QMF の変換された動詞、キーワード、メッセージ、およびパネルの完全セットがあります。グローバル変数 DSQEC_NLFCMD_LANG を使用すれば、ユーザーがコマンドを入力する言語を変更できます。

DSQEC_NLFCMD_LANG を 1 に設定すると、ユーザーはコマンドを英語でしか入力できません。

デフォルト値 0 では、ユーザーはコマンドおよびキーワードを、以下のコマンドを除いて、現行セッションの国語でしか入力できません。

```
SET
GET
INTERACT
MESSAGE
START
```

QMF では、上記のコマンドについては DSQEC_NLFCMD_LANG の設定には関係なく、英語または NLF のどちらでも入力できます。

NLF 環境で作業しているユーザーが書式オブジェクトを英語で保管できるようにするには、DSQEC_FORM_LANG 変数を使用します。SAVE, EXPORT, および IMPORT コマンドでは、保管する書式の言語を、LANGUAGE オプションによってユーザーが指定できます。このオプションの値は ENGLISH と SESSION であり、グローバル変数 DSQEC_FORM_LANG により制御されません。

DSQEC_FORM_LANG を 0 に設定すると、現行セッションの言語が保管書式の言語として使用されます。

デフォルト値は 1 で、この場合は保管書式の国語として英語が指定されます。

QMF サポートの確立

ユーザーが IMPORT または EXPORT コマンドで LANGUAGE キーワードを指定した場合は、その値によって変数 DSQEC_FORM_LANG の現行値が上書きされます。

QMF セッション時に表示される国語を変更するには、QMF ユーザーは現行の QMF セッションを終了し、別のセッションを開始する必要があります。QMF セッション内から国語を変更することはできません。

グローバル変数を使用した通貨記号の定義

キーボード上に示されていない通貨記号が必要な場合は、ロジックを持つプロシージャーで HEX (16 進) 値を使用して通貨記号を指定することができます。たとえば、次の PROC では、通貨記号が HEX '9F' に設定されます。

```
/* */  
"SET GLOBAL (DSQDC_CURRENCY =" '9F'X
```

後書きブランクが通貨記号のために必要な場合は、次のように通貨記号を単一引用符で囲むことができます。

```
SET GLOBAL (DSQDC_CURRENCY = 'FR '
```

コマンドをコマンド行または線形 PROC のいずれかで使用することができます。

第17章 ユーザーによるオブジェクトの印刷

QMF エンド・ユーザーは、データベースから検索するデータを頻繁に印刷する必要があります。このデータは、報告書、図表、データベース表、あるいは他の QMF またはデータベース・オブジェクトのフォーマットになります。

エンド・ユーザーのために印刷を設定する方法は、使用するプリンターのタイプ、および印刷する必要のある QMF オブジェクトに応じて異なります。この章は、効果的に印刷作業を進めるためには、QMF サービスを使用するのがよいか図形データ表示管理プログラム (GDDM) サービスを使用するのがよいかを決定するのに役立ちます。また、それらの方法を使用してオブジェクトを印刷する方法について説明します。

2 バイト文字セット (DBCS) データを印刷する必要がある場合、非 DBCS 端末からユーザーが DBCS データを印刷できるように QMF を始動する際に、DSQSDBCS プログラム・パラメーターを使用することができます。詳細については、230ページの『DSQSDBCS (2 バイト文字セット・データの印刷の設定)』を参照してください。

クイック・スタート

表43 は、QMF オブジェクトをプリンターまたはディスプレイに印刷する際の手引きとして使用してください。さらに各ステップについて詳しい説明が必要な場合は、表の右端に示しているページを参照してください。

印刷中にエラーを受け取った場合は、532ページの『一般的な問題の障害追及』を参照して問題を解決してください。

表 43. QMF オブジェクトの印刷

作業	参照箇所
QMF PRINT コマンドまたはコマンド同義語を使用して、QMF オブジェクトを印刷する。QMF がオブジェクトを印刷する方法は、印刷しようとするオブジェクトのタイプによって異なる。	324、339、および 340 ページ

ユーザーによるオブジェクトの印刷

表 43. QMF オブジェクトの印刷 (続き)

作業	参照箇所
QMF サービスまたは GDDM サービスのいずれかを選択して、印刷を処理する。または要求に合うように 2 つの方法を結合する。GDDM は、図形の表示をサポートする任意の装置に印刷することができる。TSO およびネイティブ OS/390 バッチ においては、QMF は DSQPRINT を使用して印刷する。CICS では、QMF は一時記憶域キューまたは一時データ・キューへ印刷する。QMF は米国規格協会 (ANSI) の紙送り制御文字コードをサポートする装置だけに印刷できる。	326 ページ
GDDM サービスを使用して印刷する。プリンターに GDDM ニックネームを定義し、そのニックネームで GDDM デフォルトのモジュール ADMADFC (CICS の場合) または ADMADFT (TSO およびネイティブ OS/390 バッチ の場合) を更新する。TSO およびネイティブ OS/390 バッチ については、DD 名 ADMDEFS を変更することもできる。CICS リソース定義表を更新し、CICS がニックネームを物理装置に結び付けられるようにする。	327 ページ
QMF サービスを使用して印刷する。TSO およびネイティブ OS/390 バッチ では、印刷のために QMF が使用するデータ・セットまたは出力クラスを指す DD ステートメントを使用して DSQPRINT を割り振る。	337 ページ
QMF サービスを使用して印刷する。CICS では、一時データ・キューを定義するか、あるいは一時記憶域キューを使用して出力を受信する。	337 ページ
ユーザーのプロファイル内の LENGTH および WIDTH 値を更新して、ページ・サイズを指定する。印刷するために GDDM サービスをアクティブ化するには、Q.PROFILES の PRINTER フィールドに有効なニックネームを指定する。	341 ページ

オブジェクトの印刷

QMF オブジェクトとデータベース・オブジェクトを印刷するための規則は、オブジェクトのタイプによって異なります。表44 は、各オブジェクトのための要件を要約したものです。

表 44. QMF オブジェクトおよびデータベース・オブジェクトのための印刷要件の要約

オブジェクトのタイプ	ニックネームの必要性	GDDM の制御取得	出力の経路指定先
図表	可	GDDM ICU は、PRINT コマンドが発行されるとき必ず制御を得る。	出力は GDDM によって制御される。詳細については、GDDM 3.1 は <i>GDDM システムのカスタマイズおよび管理</i> を、GDDM 2.3 の場合は <i>GDDM Installation and System Management for OS/390</i> を参照。

表 44. QMF オブジェクトおよびデータベース・オブジェクトのための印刷要件の要約 (続き)

オブジェクトのタイプ	ニックネームの必要性	GDDM の制御取得	出力の経路指定先
書式	可	PRINT コマンドが発行されると、必ず GDDM が制御を得ます。	出力は GDDM によって制御される。詳細については、GDDM 3.1 は <i>GDDM システムのカスタマイズおよび管理</i> を、GDDM 2.3 の場合は <i>GDDM Installation and System Management for OS/390</i> を参照。
QBE 照会	不可	ニックネームが PRINT コマンドで、またはプロファイル内で指定される場合のみ。	出力は、GDDM ニックネームまたは DD 名 DSQPRINT に関連付けられている装置へ送られる。
プロシージャー	不可	ニックネームが PRINT コマンドで、またはプロファイル内で指定される場合のみ。	出力は、GDDM ニックネームまたは DD 名 DSQPRINT に関連付けられている装置へ送られる。
プロファイル	不可	ニックネームが PRINT コマンドで、またはプロファイル内で指定される場合のみ。	出力は、GDDM ニックネームまたは DD 名 DSQPRINT に関連付けられている装置へ送られる。
指示照会	可	PRINT コマンドが発行されると、必ず GDDM が制御を得ます。	出力は GDDM によって制御される。詳細については、GDDM 3.1 は <i>GDDM システムのカスタマイズおよび管理</i> を、GDDM 2.3 の場合は <i>GDDM Installation and System Management for OS/390</i> を参照。
報告書	不可	ニックネームが PRINT コマンドで、またはプロファイル内で指定される場合のみ。	出力は、GDDM ニックネームまたは DD 名 DSQPRINT に関連付けられている装置へ送られる。
SQL 照会	不可	ニックネームが PRINT コマンドで、またはプロファイル内で指定される場合のみ。	出力は、GDDM ニックネームまたは DD 名 DSQPRINT に関連付けられている装置へ送られる。
表	不可	ニックネームが PRINT コマンドで、またはプロファイル内で指定される場合のみ。	出力は、GDDM ニックネームまたは DD 名 DSQPRINT に関連付けられている装置へ送られる。

印刷に QMF または GDDM のいずれのサービスを使用するか の決定

GDDM サービスまたは QMF サービスのいずれを使用して印刷するかは、印刷する必要のあるオブジェクトのタイプと、使用できるプリンターや他のリソースのタイプによって決まります。それぞれの要求にどちらの方法が適しているか、この節を読んで判別してください。

- 図表、書式、または指示照会を印刷する必要がある場合は、GDDM を使用する。

QMF は GDDM サービスを使用してこれらのオブジェクトを表示します。これらのオブジェクトを印刷するためには GDDM を使用しなければなりません。GDDM サービスを使用しない場合は、報告書、表、QBE と SQL 照会、プロシージャ、および QMF プロファイルの印刷しかできません。

- 出力の経路を指名プリンターに指定するようにインストール先システムを設定している場合は、印刷に GDDM サービスを使用する。

GDDM を使用すると、名前を物理装置に結び付けることができます。GDDM を使用しないで QMF サービスだけを使用する場合は、オブジェクトをプリンターに経路指定するために使用する記憶域キューのタイプと名前を指定して、オブジェクトを印刷する必要があります。

- CICS では、経路指定を自動的に処理する必要がある場合は (出力を経路指定するプログラムを作成しないで)、GDDM を使用するか、または QMF で使用する一時データ・キューを定義する。

GDDM は、CICS に定義された一時データ・キュー定義を使用して経路指定を行います。出力を保留するために一時データ・キューを使用する場合は、QMF が同様な経路指定を行います。

一時記憶域に印刷する場合は、一時記憶域キューをプリンターに送るプログラム、または印刷出力を CICS 提供のトランザクション CEBR を使用してオンラインで表示するプログラムを作成する必要があります。

- CICS では、32 767 を超える出力行を印刷する必要がある場合には、GDDM を使用するか、または QMF で使用する一時データ・キューを定義します。

一時記憶域キューは 32 767 を超えるデータ行を処理することはできません。

QMF と GDDM は、プリンター入力を非同期に処理します。すなわち、オブジェクトが実際に印刷される前に、QMF はオブジェクトが印刷されるというメッセージを戻すことがあります。

GDDM サービスを使用して印刷を処理する

重要: この節の説明は、GDDM 製品の出荷時に与えられている GDDM デフォルト値を使用する場合にだけ、適用されます。これらの値の変更についての詳細は、以下のいずれかの資料を参照してください。

- *GDDM Installation and System Management for OS/390* (GDDM 2.3)
- *GDDM システムのカスタマイズおよび管理* (GDDM 3.1)

QMF オブジェクトを印刷するために GDDM サービスを使用するには、以下のことを行う必要があります。

1. プリンターの GDDM ニックネームを選択する (『プリンターのための GDDM ニックネームの選択』を参照)。

ニックネームを使用すると、複雑なプリンターまたはディスプレイを事前定義できるので、エンド・ユーザーの作業を簡単にすることができます。ニックネームは、GDDM に対して報告書のフォーマット設定と配布の方法を示す、装置特性を定義します。ニックネームはローカル装置およびリモート装置の両方を定義することができます。

2. GDDM デフォルト・モジュールの ADMADFC (CICS の場合) または ADMADFT (TSO およびネイティブ OS/390 バッチ の場合) を、ニックネームの指定で更新します。
3. TSO およびネイティブ OS/390 バッチ については、DD 名 ADMDEFS を割り振ることができます。GDDM デフォルト・モジュールの更新については、333ページの『GDDM デフォルト・モジュールをニックネームで更新』に説明があります。DD 名 ADMDEFS の割り振りについては、334ページの『TSO およびネイティブ OS/390 バッチにニックネーム・ファイルを割り振る』に説明があります。
4. CICS リソース定義をニックネーム指定の値で更新する。これにより CICS は、それが管理する物理装置にニックネームを結び付けることができます。これについては、334ページの『CICS でニックネームを使用』に説明があります。
5. Q.PROFILES 表の該当ユーザー行の PRINTER フィールドを更新する。これについては、341ページの『ユーザー・プロファイルを更新して GDDM 印刷を可能にする』に説明があります。

プリンターのための GDDM ニックネームの選択

TSO およびネイティブ OS/390 バッチにおいて、QMF PRINT コマンドの PRINTER キーワードにプリンター名を入力すると、GDDM は最初に

ユーザーによるオブジェクトの印刷

ADMDEFS データ・セットを、次にデフォルト・モジュールである ADMADTC を検索して、出力の宛先および方法を定義するニックネームと一致させます。

CICS では、GDDM はデフォルト・モジュールの ADMADTC だけを検索しません。GDDM は、(端末を含めて) 通信できる装置をすべて認識するために、ニックネームを使用します。

正しいタイプの GDDM 装置の選択

使用するプリンターのニックネームは、装置のタイプによって異なります。

- **ファミリー 1 装置**は、GDDM-PCLK または GDDM-OS/2[®] リンクを使用するワークステーションに接続される補助装置を指定します。ファミリー 1 装置には、3270 データ・ストリーム端末などのディスプレイも含まれることがあります。
- **ファミリー 2 装置**は、IBM 3270 端末や待機プリンターなどを含みます。
- **ファミリー 3 装置**は、ANSI コードの紙送り制御文字をサポートするシステム・プリンターです。
- **ファミリー 4 装置**は、印刷出力を行うために ADMOPUT および ADMOPUJ ユーティリティ (TSO およびネイティブ OS/390 バッチのみ) を使用する必要のある、拡張機能プリンターです。これらのユーティリティは GDDM により提供されます。

この章では、ファミリー 1、2、3 の装置のニックネームを定義する方法を説明します。ファミリー 4 プリンターのニックネームの設定方法ならびに ADMOPUT および ADMOPUJ ユーティリティの使用の詳細については、GDDM 3.1 の場合は *GDDM システムのカスタマイズおよび管理* を、GDDM 2.3 の場合は *GDDM Installation and System Management for OS/390* を参照してください。これらの資料には、それぞれのタイプの GDDM 装置の詳細も載っています。

ニックネーム指定の作成

TSO およびネイティブ OS/390 バッチのニックネームを作成するには、ニックネームを DD 名 ADMDEFS データ・セットに追加します。GDDM は最初にこのデータ・セットを見ます。ニックネームが見つからない場合、GDDM は GDDM ADMMNICK 指定が定義される外部デフォルト・モジュールである ADMADFT を探します。

CICS のニックネームを作成するには、最初に GDDM ADMMNICK 指定を GDDM 外部デフォルト・モジュール ADMADFC に定義します。この指定は、

プリンターが処理できる 1 ページ当たりの行数、および CICS によるプリンターの管理方法などの装置特性を GDDM に示します。

ADMMNICK 指定は、図74 に示されているフォーマットを使用します。

```
ADMMNICK NAME=nickname,TOFAM=family_type,DEVTOK=device_token(,TONAME=name)
```

図74. ADMMNICK 指定を使用したニックネームの定義

TONAME は CICS のみで使用されます。

- NAME を使用し、QMF PRINT コマンドで使用するプリンターのニックネームを 1 ～ 8 文字で示します。たとえば、MYPRTR がニックネームの場合は、コマンド PRINT REPORT (PRINTER=MYPRTR を入力できます。NAME は単一名、コンマで区切った名前のリスト、あるいは前または後に ? を付けた名前 (出力を同じような名前を持つ複数のプリンターに送るためのワイルドカードとして使用する) でも構いません。
- TOFAM を使用して、使用している装置のタイプを示します。GDDM は装置の 4 つのファミリーを認識し、それぞれを異なる形で処理します。
- DEVTOK を使用して、有効な GDDM 装置トークンを示します。これは、装置とその印刷構成 (たとえば、1 インチ当たり 6 行で、60 行 x 85 列を印刷する 3820 プリンター) を固有に識別します。有効な装置トークンのリストは、以下の資料に載っています。

GDDM システムのカスタマイズおよび管理 (GDDM 3.1)

GDDM Installation and System Management for OS/390 (GDDM 2.3)

- CICS では、TONAME フィールドが TCT または DCT 内の項目を指し、CICS が GDDM とプリンターの間の通信を正しく管理できるようにします。TONAME を使用して、装置のタイプによって異なる値を持つプリンター定義名の 1 ～ 4 文字の名前を指します。
 - ニックネームがファミリー 1 または 2 プリンターを定義する場合、TONAME は、プリンターを CICS に定義する CICS 端末管理テーブル (TCT) 内の一致する項目を指します。一致する項目では、TRMIDNT フィールドは TONAME と同じ値を持ちます。

CICS システム定義 (CSD) ファイルを更新するために、CICS オンライン・リソース定義 (RDO) を使用してプリンターを CICS に定義する場合、TERMINAL 属性は TONAME と同じ値を持ちます。
 - ニックネームがファミリー 3 プリンターを定義する場合、TONAME は、プリンターを CICS に定義する CICS 宛先管理テーブル (DCT) 内の

ユーザーによるオブジェクトの印刷

一致する項目を指し、これによってプリンターが CICS に定義されます。一致する項目では、DESTID フィールドは TONAME と同じ値を持ちます。

固有のラベルを構文に追加できます。たとえば、GDDMPRT1 は次のニックネーム定義のラベルとして可能なものです。

```
GDDMPRT1 ADMMNICK NAME=MYPRINT,TOFAM=3,DEVTK=ADMKSYSP
```

ファミリー 1 または 2 GDDM プリンターのニックネーム例

ニックネーム GRAPHIC をファミリー 1 または 2 GDDM プリンターに定義するには、図75 と類似した ADMMNICK 指定を使用します。この指定は、ファミリー 2 GDDM プリンターのためのものです (ファミリー 1 GDDM プリンターには TOFAM=1 を使用します)。これは、装置トークン R87S を使用しますが、リモート接続される 3287 プリンターのトークンの例です。

```
ADMMNICK NAME=GRAPHIC,TOFAM=2,DEVTK=R87S,TONAME=GRAP
```

図 75. ADMMNICK 指定を使用したファミリー 2 プリンターのニックネームの定義

TSO およびネイティブ OS/390 バッチでニックネームを作成すると、QMF PRINT コマンドの実行結果として一時データ・セットが作成され、すでに存在するニックネームを指定します。このデータ・セットは userid.ADMPRINT.REQUEST.#nnnnn です。ここで nnnnn は順序番号です。次に、ADMOPUT ユーティリティを使用してデータ・セットを印刷できます。ADMOPUJ ユーティリティを使用して印刷ジョブを JES スプールに書き込むこともできます。

GDDM 印刷ユーティリティ (ADMOPUT または ADMOPUJ) のいずれかを使用し、GDDM ニックネームを使用して QMF オブジェクトを印刷する場合は、QMF 提供の GDDM マップ・グループを GDDM 印刷ユーティリティに対して使用可能にする必要があります。ADMGGMAP DD ステートメントには、マップ・グループを持つデータ・セットの名前 (QMF710.DSQMAPE) が含まれます。

```
//ADMGGMAP DD DSN=QMF710.DSQMAPE,DISP=SHR
```

このステートメントがないと、ファミリー 2 プリンターで書式を印刷しようとするとエラーが生じます。GDDM 印刷ユーティリティの詳細については、GDDM バージョン 2 リリース 3 を使用している場合は *GDDM Installation and System Management for OS/390 (GDDM 2.3)* を、また GDDM バージョン 3 リリース 1 を使用している場合は *GDDM システムのカスタマイズおよび管理* を参照してください。

重要: CICS では、ADMMNICK 指定を作成した後、TCT を更新して名前と物理装置を結び付けます。これは 335ページの図78 の例に示されています。ADMMNICK 指定の TONAME と、TCT の TRMIDNT は一致する値を持つ必要があります。

CICS RDO 機能を使用して CSD オンラインを更新することもできます。プリンターをこの方法で定義する場合は、CSD 内の TERMINAL 属性と ADMMNICK 指定内の TONAME の値が必ず一致するようにします。

ファミリー 3 GDDM プリンターのニックネーム例

ニックネーム 370PRINT をファミリー 3 GDDM プリンターに定義するには、図76 と類似した ADMMNICK 指定を使用します。

```
ADMMNICK NAME=370PRINT,TOFAM=3,DEVTOK=R87S,TONAME=370P
```

図76. ADMMNICK 指定を使用してファミリー 3 プリンターのニックネームを定義

TSO およびネイティブ OS/390 バッチでニックネームを作成した後、DD 名 ADMLIST が作成されます。次に、フォーマット設定したファイルを選択したプリンターに送ります。

CICS では、ADMMNICK 指定を作成した後、DCT を更新して名前と物理装置を結び付けます。これは 336ページの図80 の例に示されています。ADMMNICK 指定の TONAME と、DCT の DESTID は一致する値を持つ必要があります。

ファミリー 4 GDDM プリンターのニックネーム例 (TSO およびネイティブ OS/390 バッチのみ)

ニックネーム 3900PRNT をファミリー 4 GDDM プリンターに定義するには、図77 と類似した ADMMNICK 指定を使用します。

```
ADMMNICK NAME=3900PRNT,TOFAM=4,DEVTOK=R87S
```

図77. ADMMNICK 指定を使用してファミリー 4 プリンターのニックネームを定義

ニックネームを作成した後に、DD 名 ADMIMAGE が作成されます。このファイルは、JES を通して自動的に PSF/MVS ヘスプールすることができます。ファミリー 4 印刷の詳細については、GDDM システムのカスタマイズおよび管理を参照してください。

ユーザーによるオブジェクトの印刷

1 つの定義で複数のニックネームを定義

次に示したように、ニックネーム定義にワイルドカード ? を入れ、単一のニックネームで複数のプリンターのアドレスを定義できます。

```
ADMMNICK TOFAM=3,NAME=MYPRINT?,PROCOPT=((PRINTCTL,0))
```

ニックネーム MYPRINT? により、MYPRINT1、MYPRINT2、MYPRINTA などの名前を持つプリンターに印刷出力を経路指定できます。たとえば、次のように入力します。

```
PRINT REPORT (PRINTER=MYPRINT2
```

GDDM はニックネーム MYPRINT? のニックネーム定義を使用してデータ・セットを作成し、PRINT コマンドからの出力を DD 名 MYPRINT2 を持つデータ・セットに出します。

ニックネーム定義の例

この節では、ファミリー 1、2、または 3 の装置のために使用できるニックネームの例を示します。ファミリー 4 装置のニックネーム定義に関する例については、以下の資料を参照してください。

- *GDDM システムのカスタマイズおよび管理* (GDDM 3.1)
- *GDDM Installation and System Management for OS/390* (GDDM 2.3)
- **3800、3812、または 3820 プリンター、6 行 / インチ**：以下の定義を使用してニックネーム GDDMPRT1 をファミリー 3 プリンターに定義します。

```
GDDMPRT1 ADMMNICK TOFAM=3,DEVTOK=S3800N6,NAME=MYPRINT1
```

- **3800、3812、または 3820 プリンター、8 行 / インチ**：以下の定義を使用してニックネーム GDDMPRT2 をファミリー 3 プリンターに定義します。

```
GDDMPRT2 ADMMNICK TOFAM=3,DEVTOK=S3800N8,NAME=MYPRINT2
```

- **非 3800 システム・プリンター、132 桁、8 行 / インチ**：以下の定義を使用してニックネーム GDDMPRT3 をファミリー 3 プリンターに定義します。

```
GDDMPRT3 ADMMNICK TOFAM=3,DEVTOK=S1403W8,NAME=MYPRINT3
```

- **リモート接続の 3287 (図表の印刷に適切)**：以下の定義を使用して、ニックネーム GDDMPRT4 をファミリー 2 プリンターに定義します。

```
GDDMPRT4 ADMMNICK TOFAM=2,DEVTOK=R87,NAME=MYPRINT4
```

- **印刷制御オプションのない任意の宛先**：以下の定義を使用して、ニックネーム GDDMPRT5 をファミリー 3 プリンターに定義します。

```
GDDMPRT5 ADMMNICK TOFAM=3,PROCOPT=((PRINTCLTL,)),NAME=MYPRINT5
```

PROCOPT パラメーターは印刷制御 (PRINTCTL) キーワードを使用して処理オプションを指定するので、これによって多くの印刷制御オプションを指定す

ることができます。たとえば、PRINTCTL を使用して、印刷されるページ・ヘッダー、印刷部数、およびマージンの幅を指定することができます。この例では、ゼロが指定されているため、ページ・ヘッダーが抑止されます。

考慮事項: 印刷データ・セットが RECFM=F を持つ場合、GDDM 印刷はデータ・セットの DCB を RECFM=F から RECFM=V に変更します。

印刷制御オプションのリストおよびその使用方法については、GDDM システムのカスタマイズおよび管理 を参照してください。

- **GDDM-PCLK を使用する PC プリンター (DOS ユーザー用) :** ファミリー 1 プリンターにニックネーム PCPRINT を定義する場合は、次の定義を使用します。

```
GDDMPRT6 ADMMNICK TOFAM=1,FAM=0,NAME=PCPRINT,TONAME=*,ADMPCPRT
```

ここで、* はユーザーの現行装置またはデフォルト値を示します。

DOS に接続されているワークステーション・プリンターに印刷するには、GDDM-PCLK がワークステーションにインストールされている必要があります。

- **GDDM-OS/2 リンクを使用する PC プリンター (OS/2 ユーザー用) :** ファミリー 1 プリンターにニックネーム GDDMOS2P を定義するには、次の定義を使用します。

```
GDDMPRT7 ADMMNICK TOFAM=1,FAM=0,NAME=PMPRINT,TONAME=*,ADMPMOP
```

ただし、* は、ユーザーの現行装置またはデフォルト値を示します。

OS/2 に接続されているワークステーション・プリンターに印刷するには、GDDM-OS/2 リンクがワークステーションにインストールされている必要があります。

GDDM デフォルト・モジュールをニックネームで更新

CICS では、ADMMNICK ニックネーム指定は GDDM 外部デフォルト・モジュール ADMADFC 内に常駐しますが、これは GDDM 製品と一緒に提供されます。TSO およびネイティブ OS/390 バッチでは、外部デフォルト・モジュールは ADMADFT です。

デフォルト・モジュールには、GDDM 製品のデフォルト値も含まれます。モジュールは SADMSAM データ・セットのメンバーとして保管されます。

モジュールをニックネーム指定で更新するには、以下のようになります。

1. ソース・ファイルを編集してニックネームを追加します。

ユーザーによるオブジェクトの印刷

2. モジュールの ADMMDFT ステートメントの後ろに ADMMNICK 指定を入力する。
3. 変更されたデフォルト・モジュールを再アSEMBルしリンク・エディットします。

デフォルト・モジュールに関する詳細については、以下の資料を参照してください。

- *GDDM システムのカスタマイズおよび管理* (GDDM 3.1)
- *GDDM Installation and System Management for OS/390* (GDDM 2.3)

外部デフォルトのファイル内でニックネーム定義をテスト (TSO およびネイティブ OS/390 バッチのみ)

ニックネーム定義をテストするには、それらを 外部デフォルトのファイル 内に入れ、これらを使用して印刷し、正しく作動することが確認できるまで行います。次に、これらを 外部デフォルト・モジュール にアSEMBルします。

GDDM は外部デフォルト・モジュールを使用して、データ・セットよりも効率的に所定のニックネームを見つけます。CICS および TSO またはネイティブ OS/390 バッチを使用する計画がある場合は、外部デフォルト・モジュールも使用してください。

外部デフォルト・ファイルまたはモジュールを使用するとユーザーの JCL に影響を及ぼします。これは外部デフォルト・ファイルは DD ステートメントを必要としますが、外部デフォルト・モジュールは STEPLIB ライブラリーのメンバーでなければならないからです。JCL の変更に関して GDDM 管理者からアドバイスを得てください。

TSO およびネイティブ OS/390 バッチにニックネーム・ファイルを割り振る

TSO およびネイティブ OS/390 バッチの場合、ニックネーム・データ・セットの DD 名は ADMDEFS です。QMF セッションを開始するときに、それを割り振る必要があります。DD 名 ADMDEFS をユーザーのログオン・プロシージャに追加するには、次のようにします。

```
//ADMDEFS DD DSN=LOCAL.GDDM.NICKNAME,DISP=SHR
```

CICS でニックネームを使用

CICS では、ニックネームはユーザーのデフォルトの指定に組み込まれ、外部デフォルト・モジュール ADMADFC にアSEMBルされます。

ADMADFC モジュールを更新した後、CICS リソース定義を更新して、CICS がニックネームをそれが管理する物理装置に結び付けられるようにする必要があります。

ファミリー 2 ニックネームの物理装置への結び付け

QMF は報告書用に GDDM ニックネームの使用をサポートし、QMF 図表、書式、および指示照会の印刷にはニックネームを必要とします。VTAM および TCT 項目を使用して CICS に記述されたプリンターを持つ場合は、そのプリンターはキューに入っているものとして記述する必要があります (GDDM ファミリー 2 装置)。ファミリー 2 装置を使用するとき、TONAME の ADMMNICK 指定は、ファミリー 3 装置の CICS DCT 項目ではなく、TCT 項目を指します。

たとえば、次のニックネーム指定があるとします。

```
ADMMNICK NAME=GRAPHIC,TOFAM=2,DEVTOK=R87S,TONAME=GRAP
```

この場合、図78 に示されている例と類似したマクロを使用して、CICS TCT を更新することができます。

```
GRAP      DFHTCT TYPE=TERMINAL,
           ACCMETH=VTAM,
           TRMIDNT=GRAP,
           TRMTYPE=SCSPRT,
           . . .
           . . .
           . . .
```

図78. ファミリー 2 GDDM プリンターのニックネームを CICS に定義

ファミリー 3 ニックネームの物理装置への結び付け

ファミリー 3 装置を使用するには、図79 に示されるような GDDM ニックネーム表を設定します。

```
GDDMPRT  ADMMNICK TOFAM=3,   FAMILY (SYSTEM PRINTER)           X
           NAME=SYSPRT,     PRINTER NAME (NICKNAME)         X
           DEVTOK=S1403W6,  DEVICE TOKEN (1403)            X
           TONAME=SYSP      TONAME MUST MATCH CICS DCT ENTRY
```

図79. ファミリー 3 GDDM プリンターのニックネームを CICS に定義

GDDM Installation and System Management for OS/390 (GDDM 2.3) および *GDDM システムのカスタマイズおよび管理* (GDDM 3.1) で、ニックネームの

ユーザーによるオブジェクトの印刷

ユーザー・デフォルト指定への組み込み処理、およびユーザー・デフォルト指定の外部デフォルト・モジュール ADMADFC へのアセンブルについて説明しています。

TONAME パラメーターは、図80 に示されるように、CICS DCT 中に一致する項目を持たなければなりません。

```
* THE GDDM NICKNAME IS SYSPRT AND THE
* LONGEST RECORD THAT CAN BE PRINTED
* IS 256.
```

```
DFHDCT TYPE=SDSCI,DSCNAME=ADMSYSP,          X
        RECFORM=VARBLK,                      X
        RECSIZE=260,BLKSIZE=6050,TYPEFLE=OUTPUT
```

```
      .
* ENTRY FOR GDDM NICKNAME SYSPRT
SYSP   DFHDCT TYPE=EXTRA,DESTID=SYSP,DSCNAME=ADMSYSP,RSL=1
```

図80. TONAME 項目の CICS DCT

以下のように DD 名 ADMSYSP を CICS 始動 JCL に追加する必要もありません。

```
//ADMSYSP DD SYSOUT=A
```

QMF が GDDM ニックネームとインターフェースする方法

QMF は GDDM によって提供される標準インターフェースを介して GDDM ニックネームとインターフェースし、QMF が GDDM 印刷ファイルをオープンできるようにする呼び出しを発行します。

PRINT コマンドが開始されるとき、以下のデフォルト値が、DSOPEN 呼び出し上で QMF によって提供されます。

- 装置タイプは ファミリー 2 に設定されます。
- 装置トークンは * に設定されます。
- 処理オプションは指定されません (PROCOPT はゼロに設定)。
- 名前リストの中の唯一の項目はニックネームです。

印刷操作は、ASCPUT および FSFRCE GDDM サービスを使用して、一度に 1 ページずつ実施されます。印刷が完了すると、QMF は DSDROP ステートメントで印刷操作を終了します。

TSO およびネイティブ OS/390 バッチの印刷で QMF サービスを使用

DSQPRINT を使用して、報告書、表、SQL または QBE 照会、プロシージャ、またはユーザーのプロファイルを印刷することができます。

DSQPRINT は、報告書、表、SQL または QBE 照会、プロシージャ、またはプロファイルを印刷するために、プリンター名がコマンド行またはユーザー・プロファイルに指定されないときに、QMF が使用する特別なプリンター宛先です。DSQPRINT は、印刷のために QMF が使用するデータ・セットまたは出力クラスを指す DD ステートメントで割り振らなければなりません。この DD ステートメントは QMF 始動 EXEC、CLIST、または JCL の一部となります。

印刷出力をユーザー所有のデータ・セットに追加するには、DSQPRINT の割り振りを行う必要があります。そのためには、以下の JCL を使用します。

```
//DSQPRINT DD DSN=&SYSUID..PRINT.DATA,DISP=MOD
```

または、次の CLIST を使用します。

```
ALLOC DDNAME(DSQPRINT) SYSOUT(A) LRECL(133) RECFM(F B A) BLKSIZE(1330)
FREE DDNAME(DSQPRINT)
```

出力をプリンターに経路指定するには、次の構文を使用して DSQPRINT を割り振ります。

```
//DSQPRINT DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330)
```

ISPF を使用する場合: QMF 提供の DPRE (印刷報告書の表示) コマンド同義語を使用すれば、報告書を印刷しないで、指定された幅および長さの値の効果を見ることができます。これは DSQPRINT を使用している場合にだけ適用されます。DPRE の詳細については、345ページの『TSO における印刷報告書の表示 (DPRE)』および *QMF* 解説書を参照してください。

CICS で印刷を行うために QMF サービスを使用

印刷を処理するために QMF サービスを使用するには、使用したい記憶域のタイプを指定し、CICS にその記憶域の名前を提供します。

一時記憶域キューと一時データ・キューの選択

CICS 一時記憶域キューは、出力が 32 767 行に限定されます。これらはローカル印刷宛先にだけ経路指定されます。一時記憶域を使用する場合は、データを

ユーザーによるオブジェクトの印刷

このキューから一時データ・キューへ経路指定するプログラムを作成するか、または報告書を CICS 提供のトランザクション CEBR を使用してオンラインで表示する必要があります。

CICS 一時データ・キューは、CICS が開始される前の CICS DCT に関連する記憶域の量によってだけ制限されます。一時データ・キューは、区画内または区画外データ・キューとして定義することができます。一時データ・キューを使用して、データをデータ・セットまたは SYSOUT クラスに印刷することができます。いくつかの区画内データ・キューは、32 767 行に限定されます。

PRINT コマンドを使用して出力をキューに経路指定

QMF PRINT コマンドで、キューの名前とそのキューのために定義される記憶域のタイプの両方を指定できます。たとえば、報告書を XYZ という名前の一時記憶域キューに印刷するには、次のコマンドを入力します。

```
PRINT REPORT (QUEUET=TS,QUEUEN=XYZ
```

XYZ という名前の一時データ・キューから印刷するには、次のコマンドを入力します。一時データ・キューは、最初に使用される前に CICS に定義されていることを確認してください。

```
PRINT REPORT (QUEUET=TD,QUEUEN=XYZ
```

QUEUET および QUEUEN は、QUEUETYPE および QUEUENAME の省略形です。

QMF は ENQ ステートメントをキュー名に関して発行し、別のプログラムがそれを使用している場合にキューに書き込まれるのを阻止します。名前がすでに別のアプリケーションによってキューに入れられている場合、CICS は QMF に対しキューがその時点では使用不可であることを示します。SUSPEND (S) キーワードを使用し、キューが使用不可なときは何をするかを QMF に伝えます。値 YES (または Y) を使用し、キューが使用可能になるまで報告を保留し、使用可能になった後、書き込みを行います。たとえば、次のように指定します。

```
PRINT REPORT (QUEUET=TS,QUEUEN=XYZ,S=YES
```

値 NO はデフォルト値で、PRINT コマンドを取り消し、ユーザーにはメッセージを戻します。

グローバル変数を使用して印刷のためのキューを定義

PRINT コマンドで値を指定しない場合、QMF はグローバル変数 DSQAP_CICS_PQNAME および DSQAP_CICS_PQTYPE 内に保管されている値を使用します。

印刷のために一時記憶域キューを使用している場合には、グローバル変数 DSQAP_CICS_PQTYPE を TS に、また一時データ・キューを使用している場合は TD に設定します。TS がデフォルトです。

グローバル変数 DSQAP_CICS_PQNAME を使用して、一時記憶域キューまたは一時データ・キューの名前を定義します。一時データ・キューの名前は 1 ～ 4 バイトにできます。一時記憶域キューの名前は 1 ～ 8 バイトにできます。デフォルトの一時記憶域キューは DSQPnnnn です。ここで nnnn はユーザーの 4 バイトの CICS 端末 ID です。たとえば、DSQPA085 は有効な名前です。

CICS 一時記憶域キューからの印刷

ユーザーの環境を、印刷出力を一時記憶域キューに経路指定するように設定する場合は、出力をキューからプリンターに経路指定するトランザクションを作成する必要があります。QMF ユーザーは、CICS コマンドを使用して印刷トランザクションを開始することができます。同じ端末からの後続の印刷コマンドは、すべて同じキュー名を使用し、直前の報告書に追加されます。

CICS 一時記憶域キューからの報告書を見る

報告書を CICS 提供のトランザクション CEBR で見ることができます。

TSO およびネイティブ OS/390 バッチの印刷機能キーの同義語の定義

システムをカスタマイズして、QMF を終了することなくオブジェクトを印刷することができます。ローカル印刷ユーティリティーは、印刷のためのコマンド同義語を定義し、かつ印刷機能キーのカスタマイズを行った場合には、印刷機能キーを押すだけで使用することができます。

1. REXX EXEC または CLIST を作成して、現行オブジェクトをローカルに印刷します。以下に QMF 呼び出し可能インターフェースを使用した例があります。

```
/* PRTQMF REXX EXEC for local DSPRINT */
CALL DSQCIX "PRINT PROC (PRINTER=MYPRINT1"
DSPRINT '&SYSUID..MYPRINT1.DATA'
```

この例では、定義された MYPRINT1 ニックネームを持っており、印刷出力を MYPRINT1.DATA と呼ばれるデータ・セットに出すことを想定します。

ユーザーによるオブジェクトの印刷

印刷コマンドを避け、ローカル印刷のために単にオブジェクトをエクスポートしたい QMF ユーザーもあります。その場合、EXEC は次のようになります。

```
/* PRTQMF REXX EXEC for local print utilities called DSPRINT */  
CALL DSQCIX "EXPORT PROC TO MYPROC"  
DSPRINT '&SYSUID..MYPROC.PROC'
```

2. 印刷のための QMF コマンド同義語を作成します。次に、PRTQMF EXEC を実行するためのコマンド同義語 PRTQMF を作成する照会の例を示します。

```
INSERT INTO COMMAND_SYNONYMS (VERB, SYNONYM_DEFINITION, REMARKS)  
VALUES('PRTQMF','TSO PRTQMF','Print QMF Proc')
```

3. ここで、プロシージャ・パネル上の機能キーをカスタマイズして、このコマンド同義語を使用することができます。各パネルについてカスタマイズを行う必要があります。プロシージャ・パネル上の機能キー 4 をカスタマイズするための照会は次のようになります。

```
INSERT INTO PFKY_TABLE (PANEL,ENTRY_TYPE,NUMBER,PF_SETTING)  
VALUES('PROC','K',4,'PRTQMF')
```

この例では、ユーザーのプロファイルは、機能キー・カスタマイズ表の名前である、PFKY_TABLE に設定された PFKEYS 列値を持つことが想定されています。(照会を実行した後に、機能キーの変更を実施するには、QMF を再始動しなければなりません。)

CICS の印刷機能キーの同義語の定義

ユーザーが QMF を終了せずにオブジェクト (以下の例では報告書) を印刷できるようにカスタマイズできます。

印刷機能キーが押されたとき、ローカル印刷ユーティリティーを呼び出すために、次の技法があります。

1. PRT_QMF と呼ばれる QMF プロシージャを作成します。これはオブジェクトを一時記憶域に送り、その後オブジェクトを印刷するトランザクションを開始します。

```
PRINT REPORT (QUEUENAME=QMFREPT,QUEUETYPE=TS)  
CICS QMFP (FROM='QMFREPT')
```

2. 印刷のための QMF コマンド同義語を作成します。次に、PRTQMF EXEC を実行するためのコマンド同義語 PRTQMF を作成する照会の例を示します。

```
INSERT INTO COMMAND_SYNONYMS (VERB, SYNONYM_DEFINITION, REMARKS)  
VALUES('PRTQMF','RUN PRT_QMF','Print QMF Report')
```

3. ここで、報告書パネル上の機能キーをカスタマイズして、このコマンド同義語を使用することができます。各パネルごとに、キーをカスタマイズする必要があります。報告書パネル上の機能キー 4 をカスタマイズするための照会は次のようになります。

```
INSERT INTO PFKY_TABLE (PANEL,ENTRY_TYPE,NUMBER,PF_SETTING)
VALUES('REPORT','K',4,'PRTQMF')
```

この例では、ユーザーのプロファイルは、機能キー・カスタマイズ表の名前、PFKY_TABLE に設定された PFKEYS 列値を持つことが想定されています。(照会を実行した後に、機能キーの変更を実施するには、QMF を再始動しなければなりません。)

ユーザー・プロファイルを更新して GDDM 印刷を可能にする

QMF PRINT コマンドが入力されると、QMF は Q.PROFILES 表内のユーザーの行の LENGTH、WIDTH、および PRINTER の各フィールドを参照します。プロファイルのこれらのフィールドを使用して、ユーザーの出力のサイズおよび宛先を指定します。

印刷のための GDDM サービスをアクティブ化するには、プロファイルの PRINTER 列にデフォルトの GDDM プリンター・ニックネームを指定します。LENGTH および WIDTH に指定された値は、ADMMNICK 指定内の装置トークンによって指定された幅および長さと同じであることを確認してください。また、使用されるプリンター名が、CICS では、ADMADFC デフォルト・モジュール、TSO およびネイティブ OS/390 バッチでは ADMADFT デフォルト・モジュールの項目の 1 つと一致するようにします。一致しないと、QMF はエラー・メッセージを表示します。

プロファイルにプリンター名を指定しない場合、ユーザーがプリンター名を指定しないで、図表、書式、または指示照会を印刷しようとする、QMF はメッセージ Please supply a nickname for your printer を表示します。Enter キーを押すと、プリンター名のプロンプト指示が表示されます。ニックネーム・ファイルの項目と一致するプリンター名を入力するように指示してください。

ユーザー・プロファイルの PRINTER フィールドに GDDM ニックネームが含まれていない場合は、QMF サービスが印刷のために使用されます。PRINTER がブランクの場合でも、LENGTH および WIDTH にデフォルト値を指定できます。

ユーザーによるオブジェクトの印刷

プロファイル内にデフォルトの GDDM プリンター名を指定したけれども、印刷のために QMF サービスを使用したい場合は、**PRINTER** キーワードに空白値を指定し、ユーザーのプロファイル内の GDDM プリンターのニックネームを上書きします。

```
PRINT REPORT (PRINTER=' ')
```

第18章 QMF コマンドのカスタマイズ

QMF コマンド同義語は、ユーザー自身の用語を定義し、それらを QMF、CICS、または TSO 各コマンドに結び付けられるようにすることによって、QMF コマンドの基本セットのカスタマイズを支援します。同義語は、単に QMF、CICS、または TSO コマンドに対応するもう 1 つの語として働く場合と、複数のコマンドの働きをする語である場合とがあります。

コマンド同義語が作成された後は、QMF のエンド・ユーザーは QMF コマンドを通常入力するのと同じようにして、コマンド行に同義語を入力することができます。

クイック・スタート

コマンド同義語を作成するには、表45 のステップに従ってください。ステップについてさらに詳しい説明が必要な場合は、表の右端に示しているページを参照してください。

表 45. QMF コマンドの同義語の作成

作業	参照箇所
QMF で提供されるデフォルトの同義語を使用して印刷報告書を表示し、バッチ照会またはプロシージャーを実行し、報告書レイアウトをカスタマイズし、あるいは QMF を対話モードのまま ISPF にブリッジする。	344 ページ
コマンド同義語表を作成する。その中には列 VERB、OBJECT、および SYNONYM_DEFINITION が含まれる。この表は、ユーザーの選択した同義語をそれらが表すコマンドまたはプロシージャーと関連付ける。	347 ページ
同義語とその定義を表に入力する。VERB および OBJECT は同義語を保管する。SYNONYM_DEFINITION はその同義語を入力したときに実行されるコマンドまたはプロシージャーを示す。有効な動詞、目的語名、および同義語定義についての指針にしたがうこと。	349 ページ
ユーザーのために同義語をアクティブ化する。Q.PROFILES のユーザー行の SYNONYMS フィールドを同義語表の名前で更新する。次に、データベースに再接続して新規の同義語をアクティブ化するようにユーザーに指示する。	356 ページ
インストール先システムのコマンド同義語表の保守を最小化する。そのためには、すべてのユーザーのための同義語表を 1 つだけ作成するか、または同義語表について異なるタイプの視点を作成する。	357 ページ

QMF が提供するデフォルトの同義語の使用

QMF は、インストール時定義コマンドとして使用可能な 4 つのアプリケーションを提供します。インストール後、これらの同義語は Q.COMMAND_SYNONYMS 表に現れます。この表に対してアクセスするユーザーは、適切な同義語を QMF コマンドであるかのように入力し、これらのアプリケーションを呼び出すことができます。

Workstation Database Server のユーザー

インストール後、これらの同義語は表 Q.COMMAND_SYN_TSO に現れません。

印刷報告書の表示

同義語は DPRE です。ユーザーの現行報告書を印刷書式で表示します。DPRE のカスタマイズについては、345ページの『TSO における印刷報告書の表示 (DPRE)』を参照してください。

バッチ照会 / プロシージャ

同義語は BATCH です。照会またはプロシージャをバッチ・モードで実行できるようにします。このアプリケーションに関する詳細については、514ページの『QMF バッチ照会 / プロシージャ・アプリケーション (BATCH) の ISPF での使用』を参照してください。

書式のレイアウト

同義語は LAYOUT です。照会を実行せずに報告書を調整できるようにします。Layout コマンドの構文およびこのアプリケーションの使用法に関する例については、*QMF* 解説書を参照してください。

ISPF へのブリッジ

同義語は ISPF です。対話モード QMF を一時的に離れ、ISPF/PDF セッションに『ブリッジ』できるようにします。セッションが終了したら、QMF の ISPF コマンドが発行されたポイントへ戻ります。ISPF アプリケーションの詳細については、*QMF* 使用の手引き、ならびに *QMF* アプリケーション開発の手引き を参照してください。

ISPF に関する考慮事項: 同義語 ISPF は、QMF が ISPF のもとで開始されている場合にのみ有効です。QMF が ISPF のもとで開始されていない場合は、TSO ISPSTART を入力して ISPF にアクセスすることができます。

TSO における印刷報告書の表示 (DPRE)

印刷される報告書は、画面上に表示されるものとまったく同じではないことが分かります。たとえば、印刷される報告書には 1 つまたは複数のページ制御の切れ目が存在していても、表示される報告書は単一ページとして扱われます。

印刷される報告書と表示される報告書の違いは、主として見栄えの点です。表示項目および数値は、画面上と印刷されるページ上で違いはありません。しかし、違いが重要になる場合があります。(違いに関する詳細については、*QMF* 使用の手引きを参照してください。) このため、IBM は DPRE と呼ばれる QMF アプリケーションを提供し、報告書が印刷時と同じように表示されるようにしています。QMF のインストール後、このアプリケーションは、Q.COMMAND_SYNONYMS 表に保管されているコマンドを使用して呼び出すことができます。このアプリケーションはすべてのユーザーが共用することができます。

DPRE コンポーネントは以下のとおりです。

- データベース内の Q.DSQAER1P という名前のプロシージャー名
- QMF710.SDSQCLTE という名前のライブラリー内の DSQABR13 という名前の CLIST

DPRE の使用

DPRE を使用するには、報告書データを持つ DATA オブジェクト、および適切な書式を持つ FORM オブジェクトをロードして、次のコマンドを発行します。

DPRE

アプリケーションは次にプリンター出力を生成し、ISPF ブラウズ機能によってそれを表示します。ブラウズを終了すると、プリンター出力は消えます。

NLF を使用する場合: DPRE の変換済みコマンド同義語を発行し、印刷報告書を表示します。たとえば、DPRE の変換済みドイツ語コマンド同義語は AGB です。他の言語環境における DPRE の変換済みコマンド同義語については、Q.COMMAND_SYNONYM_n 制御表またはコマンドに関する変換済みオンライン・ヘルプを参照してください。

報告書パラメーター: ブラウズされる報告書の LENGTH パラメーターは、PROFILE から取り出されます。PROFILE に指定される WIDTH パラメーターは、132 (lrecl) 未満の場合に使用されます。そうでない場合は 132 (lrecl) の幅が使用されます。これは DSQPRINT に関する TSO 割り振りステートメン

QMF コマンドのカスタマイズ

トに指定された長さです。132 が小さ過ぎる場合は、DSQPRINT に関する TSO 割り振りステートメントを、より大きな幅に適合するように変更することができます。

パフォーマンスの考慮事項: QMF の設計は、報告書がユーザーのニーズに適合するまで FORM パネルの変更と REPORT の表示を繰り返すことによって、印刷される報告書を作成できるようにしています。DPRE を使用すれば、FORM パネルの変更と試験的な報告書のブラウズを交互に行うことができます。ただし、2 番目の作成方法は最初の方法に比べて費用がかさむので、リソースが重要な場合に、控えめに使用するようしてください。

大きな報告書の場合、報告書のすべての行が、報告書を表示する前に取り出されます。

エラーへの応答: DSQPRINT は、PRINTER=' ' が明示または暗黙のどちらかで指定されている QMF PRINT コマンドからの出力を受け取るデータ・セットの DD 名です。DPRE を実行すると、DSQPRINT はブラウズされる資料を保持するデータ・セットとして再定義されます。エラーで実行が停止された場合、この定義は有効のままになっている可能性があります。

DPRE のカスタマイズ

重要: ファイルに変更を行うときは、最初にその名前を変更し、元のファイルおよび変更済みファイルのバックアップ・コピーを必ず保管します。

DPRE の 2 つの区域を変更できます。

- BROWSE データ・セットの処理

アプリケーションは、DSQPRINT を作成される順次データ・セットとして再割り振りします。このデータ・セットには、ユーザーがブラウズする報告書の印刷書式が含まれます。このデータ・セットの名前およびその後処理を変更することができます。

- DPRE の機能キーの変更

DPRE の機能キーを変更するには、QMF PROC Q.DSQAER1P および QMF710.SDSQCLTE (DSQABR13) を編集する必要があります。たとえば、DPRE アプリケーション機能キー 12 を CURSOR から RETRIEVE へ変更するには、以下の両方を実行する必要があります。

- Q.DSQAER1P で、PF12CON 行の値を CURSOR から RETRIEVE へ変更します。
- CLIST DSQABR13 で、ZPF12 と ZPF24 の両方の値を、CURSOR から RETRIEVE へ変更します。

- DSQPRINT の再割り振り

報告書のブラウズが終了してから、DSQPRINT を、アプリケーションが呼び出される前の状態に再割り振りする必要があります。アプリケーション内の以下のステートメントがこれを行います。これらのステートメントはプロシージャ DSQAERIP にあります。

```
ADDRESS TSO "ATTR DSQDPRA LRECL(133) RECFM(F B A) BLKSIZE(1330)"
ADDRESS TSO "ALLOC DDNAME(DSQPRINT) SYSOUT(A) USING(DSQDPRA)"
```

ALLOC ステートメントを変更することができます。たとえば、DSQPRINT の出力クラスを A から C に変更できます。出力クラス C が機密の印刷出力を処理する場合に、インストール先システムのほとんどの QMF 報告書が機密である場合には、これを行うと便利です。変更された ALLOC ステートメントは次のようになります。

```
ADDRESS TSO "ALLOC DDNAME(DSQPRINT) SYSOUT(C) USING(DSQDPRA)"
```

コマンド同義語表の作成

QMF セッションが開始されると、QMF はユーザー・プロファイルの同義語フィールドに指定された名前を持つコマンド同義語表をロードします。コマンドを入力すると、QMF は最初に同義語表で一致するものを検査します。一致するものが見つからない場合、QMF はそのコマンドが基本 QMF コマンドであると見なします。コマンドの前に *QMF* の文字を入力すると、QMF は自動的にそのコマンドが基本 QMF コマンドであると見なし、同義語表で一致するものを検査しません。

以下のプロシージャにより、コマンド同義語表を作成します。次に、同義語の入力および定義について、349ページの『コマンド同義語定義を表へ入力』を参照してください。

1. 必要に応じて、コマンド同義語表を保持するための表スペースを獲得または追加します。279ページの『ユーザー用の表スペースの選択および割り当て』は、表スペースの選択または作成方法を示しています。使用可能な表スペースがない場合は、以下のような照会によって、表のためのスペースを作成します。

QMF コマンドのカスタマイズ

```
CREATE TABLESPACE DSQTSSN1
  IN DSQDBCTL
  USING STOGROUP DSQSGSYN
  PRIQTY 100
  SECQTY 20
  LOCKSIZE PAGE
  BUFFERPOOL BP0
  CLOSE NO
```

図 81. 表スペースの作成

この照会の実行により、表スペース DSQTSSN1 が作成されます。この表スペースの記憶域グループとデータベースは、Q.COMMAND_SYNONYMS が含まれる表スペースのものでもあります。

DSQDBCTL.DSQTSSYN を表スペースとして使用することもできます。Q.COMMAND_SYNONYMS 表は DSQDBCTL.DSQTSSYN に常駐します。

2. QMF SQL 照会パネルから、図82 に似た SQL CREATE TABLE ステートメントを実行して、表を作成します。ユーザーの表の名前で COMMAND_SYNONYMS を、またユーザーの表スペースの名前で TBSPACE1 を置き換えます。照会の他の部分は、次に示されているとおりに入力します。

```
CREATE TABLE COMMAND_SYNONYMS
( VERB          CHAR(18)          NOT NULL,
  OBJECT        VARCHAR(31),
  SYNONYM_DEFINITION VARCHAR(254) NOT NULL,
  REMARKS       VARCHAR(254))
IN TBSPACE1
```

図 82. コマンド同義語表の作成

VERB および OBJECT 列はユーザーの同義語を保管します。SYNONYM_DEFINITION 列は、同義語を入力したときに実行されるコマンドまたはプロシーチャーを保管します。

各列の順番は任意で、またコメントの列を追加して、各同義語がどんな機能を実行するかを識別することができます。

3. 図82 の照会で作成された COMMAND_SYNONYMS 表に関する次の例を使用し、コメントを DB2 システム・カタログに追加します。

```
COMMENT ON TABLE COMMAND_SYNONYMS IS 'SYNONYMS FOR R AND D'
```

句 SYNONYMS FOR R AND D が DB2 システム・カタログの REMARKS 列に現れます。

新規の表に関するコメントを DB2 システム・カタログに加える必要はありませんが、それを行う場合は、1 つのコメントは表に関するもの、その他は各列について記述するものにします。たとえば、COMMAND_SYNONYMS が、私用同義語と共用同義語を区別する AUTHID という名前の列をもつものとし、これを説明するコメントを加えるには、次の照会を実行します。

```
COMMENT ON COLUMN COMMAND_SYNONYMS.AUTHID
  IS 'PRIVATE SYNONYM: USE AUTH ID. PUBLIC SYNONYM: USE NULL'
```

後続の COMMENT ON 照会を実行することにより、現行のものを置き換えることができます。COMMENT ON 照会の詳細については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

4. 索引を作成し、QMF がコマンド同義語表を処理するとき、初期化時のパフォーマンスが最大になります。次に示すのと同じようなステートメントを使用します。

```
CREATE UNIQUE INDEX SYNONYMS_INDEX
  ON COMMAND_SYNONYMS (VERB,OBJECT)
```

VERB および OBJECT 列の両方に UNIQUE キーワードで索引を付け、同義語定義の重複を避けます。UNIQUE キーワードを使用しない場合には、QMF はこの表に同義語の重複を許可します。QMF は、表中で最初に検出した同義語を使用し、初期化を行った後に QMF ホーム・パネルに警告メッセージを表示します。

コマンド同義語定義を表へ入力

コマンド同義語表を作成した後、図83 と類似した SQL INSERT ステートメントを使用して、同義語を表に入力します。表編集機能を使用して表を更新することもできます。これは *QMF 使用の手引き* に説明されています。テキスト編集プログラムを使用して TSO ファイル内に同義語を作成し、次にこれらを DB2 LOAD ユーティリティで表に入力することもできます。このユーティリティに関する情報については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

```
INSERT INTO COMMAND_SYNONYMS (VERB,OBJECT,SYNONYM_DEFINITION)
  VALUES('COMPUTE', 'MONTHLY_SALES', 'RUN PROC JONES.SALES_FIGURES')
```

図 83. コマンド同義語定義の作成

QMF コマンドのカスタマイズ

これが 356ページの『同義語のアクティブ化』のプロシーチャーに従ってアクティブ化されると、同義語 COMPUTE MONTHLY_SALES は、ユーザー JONES が所有する SALES_FIGURES と呼ばれる QMF 線形プロシーチャーを実行します。

図84 の照会は、オブジェクト列に項目がない同義語の例を示しています。

```
INSERT INTO COMMAND_SYNONYMS (VERB,SYNONYM_DEFINITION)
VALUES('EXECUTE','RUN QUERY')
```

図 84. コマンド同義語定義の作成

アクティブ化されてから、同義語 EXECUTE は、現在 QMF 一時記憶域にある照会を実行します。

図83 および 図84 の同義語は、QMF が各同義語を正しく処理することを可能にする指針に従います。この節の残りの部分では、これらの指針について説明します。これらの指針は、QMF が表中の VERB、OBJECT、および SYNONYM_DEFINITION の列を正しく処理できるようにするために従う必要があります。

動詞の選択

どのコマンド同義語にも動詞が必要です。 目的語名のみがオプションです。

動詞は、SYNONYM_DEFINITION 列に保管される QMF RUN コマンド、CICS コマンド、または TSO コマンドに対するユーザー独自の語です。たとえば、ユーザーの企業で財務結果を戻すプロシーチャーだけを実行する財務分析者がいる場合、QMF 基本動詞 RUN の同義語 COMPUTE を作成することができます。

VERB 列に関する規則

同義語表の VERB 列の各項目について以下のことを確認してください。

- 長さは 1 ～ 18 文字。
- ブランクは含まない。
- 動詞 *QMF* を含まない (他の基本 QMF コマンドは許可される)。
- 最初の文字は、英字または国別文字でなければならない (英語では、国別文字は #、@、および \$)。

最初の文字の後の文字は、英字、国別文字、10 進数字または下線とすることができます。 その他の文字は使用できません。

以下の例はこれらの規則を示しています。QMF は、VERB 列に無効な項目がある行は無視して、警告メッセージを表示します。

有効な動詞：

無効な動詞：

COMPUTE

DO SALES (ブランクは二重引用符で囲んでいなければ無効)

DISPLAY

ADJ%AGE (% は許可されない)

PRINT PRINT_PRODUCTIVITY_TOTALS (18 文字を超えている)

コマンド同義語動詞として基本 QMF 動詞を使用

PRINT などの基本 QMF コマンドを同義語として使用できます。たとえば、印刷出力を GDDM 定義のプリンターへ自動的に経路指定する同義語を定義することもできます。

基本 QMF コマンドでもある同義語を定義する場合、基本 QMF コマンドを使用するときは、コマンドの前に *QMF* を付けてください。たとえば、同義語 *DISPLAY* が、QMF コマンドの *RUN PROC SALES_REPORT* を実行する同義語定義を表すとします。*SALES_REPORT* プロシージャは照会を実行し、報告書を GDDM 定義のプリンターで印刷します。*DISPLAY* の前に *QMF* の入力を忘れた場合は、必ずしも出したかったデータのフォーマットで印刷報告書が受け取れるとは限りません。動詞 - 目的語の形の同義語で基本動詞を使用すると、同じような影響があります。

いくつかの基本 QMF コマンドは、その後ろにパラメーターを指定する必要があります。たとえば、*IMPORT* コマンドの後ろには、*TABLE* などのオブジェクト・タイプが必要です。*IMPORT* などの動詞を、動詞 - 目的語のペア内で使用する場合は、誤って同義語を実行しないように、これらのパラメーターではない目的語名を選択してください。使用する他の基本コマンドに関しては、それがパラメーターを必要とするかどうかを、*QMF* 解説書の構文図で調べてください。

目的語名の選択

目的語名はコマンド同義語ではオプションです。しかし、目的語名を使用するときは、動詞と目的語名の *両方* を必ず指定してください。指定しないと、QMF は同義語表で一致するものを見つけることができません。OBJECT 列には、以下の規則に従って入力する必要があります。

- 長さは 1 ～ 18 文字。
- DB2 表の命名規則に適合しなければならない。

QMF コマンドのカスタマイズ

- 目的語名が空白またはその他の特殊文字を持つ場合は、二重引用符で囲まなければならない。(QMF もデータベース・マネージャーも、名前を処理するときに、二重引用符を除去する。)

以下の例は有効な目的語と無効な目的語を示しています。

有効な目的語：

無効な目的語：

PFKEYS

80CAT (最初の文字が数字であってはならない)

MONTH_2_REPORT

ADJ%AGE (% は許可されない)

“User x”.“Net Sales”

JANUARY_PRODUCTIVITY (18 文字を超えている)

『Net Sales』

JONES GROSS (空白には二重引用符が必要)

完全修飾表名を使用する場合: 目的語名は完全修飾表名と同じにすることができます。これは QMF 言語と整合性を持ちます。ただし、表以外の QMF 目的語は 3 部分名で参照することはできません。たとえば、以下の QMF コマンドの目的語名は、完全修飾表名を持ちます。

DISPLAY FORM.BACKUP

同義語定義の選択

同義語定義は、コマンド同義語が入力されたときに実行される QMF コマンドまたはプロシージャです。SYNONYM_DEFINITION 列には次のものを入れることができます。

- QMF プロシージャまたは照会を呼び出す RUN コマンド。たとえば、RUN PROC JONES.SALES_DATA は、コマンド同義語 COMPUTE MONTHLY_SALES の同義語定義です。
- CLIST を呼び出す TSO コマンド。
- 別の CICS トランザクションを開始させる CICS コマンド。

同義語定義は、その定義が QMF 線形プロシージャを実行する場合には、両方のタイプのコマンドを含むことができます。

コマンド同義語で実行される複雑なアプリケーションの開発については、*QMF アプリケーション開発の手引き* を参照してください。

線形プロシージャーを同義語定義で使用

線形プロシージャーは、QMF コマンドを順番に実行する QMF プロシージャーです。同義語定義は、いくつかの QMF コマンドの働きをする線形プロシージャーを含むことができます。たとえば、図85のプロシージャーは以下のタスクを実行します。

1. SALES_DATA と呼ばれる以下の照会を実行する。これは販売担当者番号 20 が担当する全カスタマーを示す報告書を作成します。

```
SELECT QUANTITY, CUSTNO
FROM Q.SALES
WHERE SALESREPNO = 20
```
2. 報告書を、QMF から TSO 仮想記憶域または CICS 一時キューに経路指定する。図85では、XYZ は一時記憶域キューの名前です。
3. CICS または TSO プロシージャーを実行し、報告書を仮想記憶域から事前定義印刷宛先へ経路指定する。図85では、RPTX はトランザクション名です。これは QMF と非同期的に実行され、出力を REPORTX という名前の宛先に経路指定します。

```
-- Procedure name: SALES_PROC
RUN QUERY SALES_DATA
PRINT REPORT (QUEUE=XYZ,QUEUETYPE=TS)
CMS RPTX (FROM=('REPORTX, XYZ'))
```

図 85. コマンド同義語を使用して実行するサンプル・プロシージャー

このプロシージャーを実行する同義語の定義は、図86 のようになります。

VERB	OBJECT	SYNONYM DEFINITION
SHOW	SALES	RUN PROC SALES_PROC

図 86. コマンド同義語を使用して線形プロシージャーを実行

NLF を使用する場合: 照会中の QMF コマンド、書式、およびプロシージャーに含まれるその他の目的語は、プロシージャーを呼び出すコマンド同義語が使用される前に、必ず変換されているようにします。また、これらのコンポーネントが、使用する NLF に適合することを確認してください。プロシージャーで DSQEC_NLFCMD_LANG 変数を 1 に設定しない限りコマンドは、コマンド同義語が使用される前に、必ず変換されているようにしてください。

DSQEC_NLFCMD_LANG 変数は、321ページの『NLF環境での英語サポートのイネープリング』で説明されています。

変数を同義語定義で使用

変数を同義語定義で使用し、定義内で指名された目的語 (照会など) に存在する類似名変数に関する値を渡すことができます。たとえば、図87 は、表名に値 Q.STAFF を渡す定義を示します。これは MYQUERY が実行されるときに評価されます。

VERB	OBJECT	SYNONYM DEFINITION
EXECUTE	-	RUN QUERY MYQUERY (&TABLENAME=Q.STAFF

図 87. 変数をコマンド同義語定義で使用

MYQUERY は次のようになります。

```
SELECT * FROM &TABLENAME
```

アンパーサンドは、同義語定義の変数名では二重になっていますが、QMF が RUN コマンドを実行するときは単一アンパーサンドになります。

変数 &ALL 以外のすべての変数には、同義語定義でアンパーサンドを 2 個使用してください。&ALL は、変数値を同義語定義の中に組み込むのではなく、同義語を入力するときに変数値を入力するための、特殊な QMF 変数です。同義語定義内で変数 &ALL を使用すると、その同義語の右側に入力した情報がすべて変数値として使用されます。&ALL 変数を使用して、同義語定義内の情報の所在を示すことができます。

図88 の同義語定義は、&ALL を使用して定義される同義語の例を示しています。

VERB	OBJECT	SYNONYM DEFINITION
SHOW_INFO	-	RUN QUERY STAFFQUERY (&ALL)

図 88. 変数 &ALL をコマンド同義語で使用

STAFFQUERY という名前の照会は、次のようになります。

```
SELECT * FROM Q.STAFF
WHERE DEPT=&DEPT and JOB=&EMPLOYEE_JOB
```

前の例において定義された SHOW_INFO 同義語をアクティブ化してから、次のステートメントを QMF コマンド行から入力することにより、部門 10 の全管理者に関する情報を表示できます。

```
SHOW_INFO &DEPT=10 &EMPLOYEE_JOB='MGR'
```

&ALL の規則: 変数 &ALL を同義語定義の中で使用するとき、以下の点を守ってください。

- 同義語定義では &ALL は 1 度だけ使用する。
- &ALL は、常に大文字で入力する。
- &ALL の後に数字または文字を続けることはできない。
- &ALL に置き換える値はすべて、QMF がコマンド全体を評価するときに構文上正しいものでなければならない。QMF コマンドの構文についての詳細については、*QMF 解説書* を参照してください。

ユーザーがコマンド同義語に続いて値を入力しないと、QMF は &ALL に対してヌル値で置換します。354ページの図88 で示す同義語定義では、ユーザーがコマンド行に SHOW_INFO だけを入力すると、QMF は、&DEPT 変数と &EMPLOYEE_JOB 変数の値を入力するようにユーザーにプロンプトで指示します。

情報を SYNONYM_DEFINITION 列に入力

同義語定義を同義語表に入力するときは、以下の規則に従ってください。

- 同義語定義内の変数は、前後を単一引用符で囲む。
変数の前後に単一引用符を付けておくと、ユーザーが照会の実行時にコマンドに引用符を追加する必要がなくなります。たとえば、以下の同義語定義内で &ALL に単一引用符を付けます。

```
RUN MYQUERY (&&NAMEVALUE='&ALL')
```

名前 O'BRIEN を検索する場合、'O'BRIEN' と入力する必要はありません。QMF がこれを行うからです。

- 基本動詞とキーワードは大文字で入力する。
同義語定義のリテラル情報は、大文字に変換されません。
- 所有者と、同義語を使用するユーザーの SQL 許可 ID とが異なる場合は、すべての目的語名を修飾する。

QMF は、指定された目的語名を含んでいる同義語を検索するときは、名前を修飾しないで検索します。たとえば、同義語定義の中にユーザー ID JONES が所有する MY_SALES という名前の照会が含まれている場合、そ

QMF コマンドのカスタマイズ

の同義語定義内の目的語名は JONES.MY_SALES と読めるようにしておく必要があります。そうでないと、そのコマンド同義語を使用できるのは JONES だけということになります。

- 区切り ID の外側にある英字については、大文字だけを使用する。

QMF がユーザー入力 (同義語) を大文字に変換し、同義語定義が小文字の場合、QMF はユーザーが入力した同義語に一致する同義語定義を検出できません。ユーザーの QMF プロファイルの CASE 値は、入力を大文字に変換するかどうかを制御します。CASE 値を変更するには、SET PROFILE コマンドを使用します。このコマンドについての説明は、*QMF 解説書* にあります。

同義語のアクティブ化

コマンド同義語表をユーザーのためにアクティブ化するには、以下のようになります。

1. ユーザー・プロファイルの SYNONYMS フィールドを適切なコマンド同義語表名で更新する。

たとえば、COMMAND_SYNONYMS 表を英語版のユーザー JONES に割り当て、表 GUMMOW.XYZ をドイツ語 NLF 環境のユーザー SCHMIDT に割り当てるには、図89 の照会を使用します。

基本 QMF (英語)

ドイツ語 NLF

UPDATE Q.PROFILES

```
UPDATE Q.PROFILES
```

SET SYNONYMS='COMMAND_SYNONYMS'

```
SET SYNONYMS='GUMMOW.XYZ'
```

WHERE CREATOR='JONES'

```
WHERE CREATOR='SCHMIDT'
```

AND TRANSLATION='ENGLISH'

```
AND TRANSLATION='DEUTSCH'
```

AND ENVIRONMENT='CMS'

```
AND ENVIRONMENT='CMS'
```

図89. ユーザーの QMF コマンド同義語のアクティブ化

重要: Q.PROFILES を更新するときは、必ず TRANSLATION に値を指定します。指定しないと、意図した行以外の行も変更してしまう場合があります。

356ページの図89 の照会は、QMF にすでに登録されているユーザーに適用されます。類似の照会を使用して SYSTEM プロファイルを更新することができます。新規のユーザーを登録する場合は、244ページの図49 に示されているものと同じような INSERT 照会を使用します。

- SQL SELECT 特権を PUBLIC に与え、割り当てられたユーザーが同義語にアクセスできるようにする。たとえば、次のように指定します。

```
GRANT SELECT ON COMMAND_SYNONYMS TO PUBLIC
```

表そのものではなく、同義語の視点を使用している場合は、SELECT をその視点だけに与え、資格のないユーザーが同義語にアクセスするのを阻止します。視点については、『コマンド同義語表の保守の最小化』で説明しています。

- 新規の同義語をアクティブ化するために、現行 QMF セッションを終了し、別のセッションを開始するようにユーザーに指示します。

コマンド同義語の省略形については、QMF コマンドと同じ規則があてはまりません。どの省略形も固有の QMF コマンドまたはコマンド同義語を示さなければなりません。たとえば、同義語 EXECUTE のための有効な最も短い省略形は EXE です。EX とだけ入力すると、QMF はコマンド同義語 EXECUTE と基本 QMF コマンド EXPORT とを区別することができません。QMF コマンドの適切な省略形は、*QMF 解説書* に記載してあります。

コマンド同義語表の保守の最小化

コマンド同義語表を、QMF のホーム・パネルが表示される前に、初期化します。QMF の初期化時間が長引くようになった場合、おそらく、コマンド同義語表を再編成する時期になっています。表の統計をモニターする場合は、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

ユーザーのコマンド同義語表を保守する時間を最小にするためには、1 つの同義語表をすべてのユーザーに割り当てるか、または、同じ表のさまざまな異なる視点を割り当てます。両方の方法について、この節で説明します。

1 つの同義語表を全ユーザーに割り当てる

個々のユーザーに作成するコマンド同義語表が多くなるに従って、それを保守する時間は増えていきます。保守時間を短くするための 1 つの方法は、コマンド同義語表を 1 つだけ作成し、それを全ユーザーに割り当てるというものです。358ページの図90 の照会は、COMMAND_SYNONYMS という名前の表を基本 (英語版) QMF を使用する全ユーザーに割り当てています。

QMF コマンドのカスタマイズ

```
UPDATE Q.PROFILES
SET SYNONYMS='Q.COMMAND_SYNONYMS'
WHERE TRANSLATION='ENGLISH' and ENVIRONMENT='CMS'
```

図 90. 単一のコマンド同義語表を全 QMF ユーザーに割り当てる

同義語表の視点を個々のユーザーに割り当てる

個々のユーザーに対して、その必要に応じた固有の同義語を使用可能にし、しかも表の保守にかかる時間を容認できる程度に抑えるには、1 つの同義語表からいくつかの視点を作成し、それらを個々のユーザーまたはユーザー・グループに割り当てる方法を使用します。作成できる視点のタイプには 3 種類あります。

共用または私用で使用する同義語

個々に使用される同義語の数が少ない場合には、各同義語について共用 (全ユーザー) または私用 (個々のユーザー) のフラグ処理をする視点の作成および割り当てを考慮します。

1. 表を作成するときに、同義語表に AUTHID 列を追加する。AUTHID 列をヌルにしておく、それは共用同義語であることを示し、AUTHID 列にユーザー ID を入れると、それは私用同義語を示します。同じ同義語に多数の項目を割り当て、それぞれの項目を異なるユーザーに割り当てることが可能です。
2. 図91 のような照会を使用して、同義語表の視点を作成する。この照会により、ユーザー (図で userid で示されている) は、表中のすべての共用同義語、および自分の SQL 許可 ID に対して私用に割り当てられた同義語を使用できます。

```
CREATE VIEW SYNVIEW (VERB,OBJECT,SYNONYM_DEFINITION)
AS SELECT VERB, OBJECT, SYNONYM_DEFINITION
FROM COMMAND_SYNONYMS
WHERE AUTHID='userid' OR AUTHID IS NULL
```

図 91. 同義語の使用を個人用および共用として制御する視点の作成

共用使用またはグループ使用のための同義語

大きなエンド・ユーザー・グループをサポートしている場合は、ある特定の同義語がある特定のユーザー・グループで使用されるようにフラグを立てた視点を作成し、割り当てる方法があります。

視点を作成するために使用される同義語表は、あるユーザー・グループに属する同義語 1 つにつき 1 行を、また、それぞれの共用同義語 1 つにつき 1 行

を使用した行からなります。AUTHID にはヌルを入れるか、またはそのユーザー・グループを他と区別する値を入れます。

1. 表にまだ AUTHID 列がない場合は、それを同義語表に追加する。
2. 図92 のような照会を使用して、同義語表の視点を作成する。この図の例は、共用ユーザー ID が DEPTD02 であるユーザー・グループのための視点を示しています。DEPTD02 グループの中のユーザーはだれでも、表の中の共用同義語すべてと、このグループに特に割り当てられている同義語を使用できます。

```
CREATE VIEW GROUPVIEW (VERB,OBJECT,SYNONYM_DEFINITION)
AS SELECT VERB, OBJECT, SYNONYM_DEFINITION
FROM COMMAND_SYNONYMS
WHERE AUTHID='DEPTD02' OR AUTHID IS NULL
```

図92. グループ用と共用の同義語を制御する視点の作成

許可表と対になった同義語

ある列に SQL 許可 ID を、もう 1 つの列にキー値を入れた別の表を新たに作成する方法もあります。特定の SQL 許可 ID のキー値がコマンド同義語表の行のキー値と一致する場合、そのユーザーはその行に入っている同義語を使用することができます。

図93 のような照会を使用すると、コマンド同義語を保守するこの方法を実施できます。この照会は、COMMAND_SYNONYMS という表に関する KEYVIEW という名前の視点を作成します。この視点には、COMMAND_SYNONYMS と補助表 KEYTABLE の間でキーが一致する同義語だけが入ります。

```
CREATE VIEW KEYVIEW (VERB,OBJECT,SYNONYM_DEFINITION)
AS SELECT VERB, OBJECT, SYNONYM_DEFINITION
FROM COMMAND_SYNONYMS
WHERE AUTHID IS NULL OR AUTHID IN
(SELECT KEYS FROM KEYTABLE WHERE USER=userid)
```

図93. 同義語の使用を制御するための追加の表を使用する視点の作成

QMF コマンドのカスタマイズ

第19章 QMF 機能キーのカスタマイズ

各 QMF パネルの機能キーのデフォルトの設定およびラベルは、エンド・ユーザーが一般的に行う QMF のいくつかのタスクを表しています。しかし、各インストール先には固有の要件があるため、QMF では画面上に表示されるラベルと、ユーザーがキーを押した時に実行される QMF コマンドの両方をカスタマイズする手段を提供しています。

クイック・スタート

表46 の各ステップに従って、ユーザーの機能キーをカスタマイズします。ステップについてさらに詳しい説明が必要な場合は、表の右端に示してあるページを参照してください。

表 46. QMF 機能キーのカスタマイズ

作業	参照箇所
カスタマイズしたいパネルおよび機能キーを選択する。表編集機能パネルおよびデータベース状況表示パネルを除き、すべてのパネルの機能キー設定を変更できます。キーのカスタマイズの際の柔軟性は、ユーザーが選択したパネルのタイプによって決まります。	361 ページ
カスタマイズした機能キーの定義を保持する表を作成する。少なくとも 4 列 (PANEL、ENTRY_TYPE、NUMBER および PF_SETTING) を組み込む。これらの列には、キーを押した時に発行される QMF コマンド、および画面上でキー番号の側に表示されるラベル・テキストに関する情報があります。	364 ページ
ユーザーがカスタマイズしたキーの定義を機能キー表に挿入する。定義を挿入するには、カスタマイズするパネルのパネル ID、キーを押した時に発行される QMF コマンド、画面でキー番号の次に表示されるテキスト、および画面のどこにキーが位置づけられるかを参照。	366 ページ
新規の機能キー定義をアクティブ化する。Q.PROFILES のユーザーの行の PFKEYS フィールドを、ユーザーが作成した機能キー表の名前で更新する。	374 ページ

カスタマイズしたいキーの選択

QMF 機能キーは、2 つのタイプのパネルで表示されます。1 次パネル (FORM.MAIN や REPORT のようなフルスクリーン・パネル) と、2 次パネル (ウィンドウ・ダイアログ・パネルとして現れるパネル) です。ヘルプ、指示、および指示照会パネルは、2 次パネルの例です。

QMF 機能キーのカスタマイズ

『フルスクリーン・パネルのデフォルト・キー』の表は、フルスクリーン・パネルおよびウィンドウ・パネルの両方のデフォルトの QMF 機能キー・ラベルとコマンドを示します。どの機能キーを変更するかを決定するのに使用してください。

表編集プログラム・パネルの機能キーをカスタマイズすることはできません。その他のパネルでは、QMF コマンドまたはインストール時定義のコマンドを選択して、ユーザーが変更する機能キー・ラベルと関連付けることができます。

フルスクリーン・パネルのデフォルト・キー

キー	実行されるコマンド
Backward	BACKWARD
Cancel	CANCEL
Change	CHANGE
Chart	DISPLAY CHART または SHOW CHART
Check	CHECK
Clear	CLEAR
Command	SHOW COMMAND
Comments	SWITCH COMMENTS
Delete	DELETE
Describe	DESCRIBE
Draw	DRAW
Edit Table	EDIT TABLE
End	END
Enlarge	ENLARGE
Form	DISPLAY FORM または SHOW FORM
Forward	FORWARD
Help	HELP
Insert	INSERT
Left	LEFT
List	LIST
Print	PRINT
Proc	DISPLAY PROC または SHOW PROC
Profile	DISPLAY PROFILE

キー	実行されるコマンド
Query	DISPLAY QUERY または SHOW QUERY
Reduce	REDUCE
Refresh	REFRESH
Report	DISPLAY REPORT または SHOW REPORT
Retrieve	RETRIEVE
Right	RIGHT
Run	RUN QUERY または RUN PROC
Save	SAVE PROFILE
Show	SHOW
Show Field	SHOW FIELD
Show SQL	SHOW SQL
Sort	SORT
Specify	SPECIFY
Specify View	SPECIFY VIEW

ウィンドウ・パネルのデフォルト・キー

キー	実行されるコマンド
Attribute	SPECIFY ATTRIBUTES
Backward	BACKWARD
Cancel	CANCEL
Clear	CLEAR
Command	SHOW COMMAND
Comments	SWITCH COMMENTS
Condition	SPECIFY CONDITION
Delete	DELETE
Describe	DESCRIBE
End	END
Exit	END
Forward	FORWARD
Help	HELP
Index	HELP INDEX
Keys	HELP KEYS

QMF 機能キーのカスタマイズ

キー	実行されるコマンド
List	LIST
Menu	HELP MENU
More Help	HELP MORE
Next Column	NEXT COLUMN
Next Definition	NEXT DEFINITION
Previous Column	PREVIOUS COLUMN
Previous Definition	PREVIOUS DEFINITION
Refresh	REFRESH
Show Entity	SHOW ENTITY
Show Field	SHOW FIELD
Show View	SHOW VIEW
Sort	SORT
Specify Attributes	SPECIFY ATTRIBUTES
Specify Condition	SPECIFY CONDITION
Switch	HELP SWITCH

グローバル変数リスト・パネルでは、RESET GLOBAL は Delete 機能キーが押されたときに実行されるコマンドです。

これらの機能キーに関連するコマンドの詳細については、*QMF 解説書* を参照してください。

機能キー表の作成

どの機能キーをカスタマイズしたいかを決定したら、以下のステップに従って、ユーザーがカスタマイズした機能キー定義を適切なパネルにリンクする表を作成します。

1. 365ページの図94 に示されているものと同じような SQL CREATE TABLE ステートメントを使用して、表を作成します。ユーザー独自の名前で MY_PFKEYS を、独自の表スペースで TBSPACE1 を置換してください。

```
CREATE TABLE MY_PFKEYS
(PANEL          CHAR(18)          NOT NULL,
ENTRY_TYPE     CHAR(1)           NOT NULL,
NUMBER         SMALLINT          NOT NULL,
PF_SETTING     VARCHAR(254),
REMARKS        VARCHAR(254))
IN TBSPACE1
```

図 94. 機能キー表の作成

表を保持するための DB スペースの獲得に関する詳細については、279ページの『ユーザー用の表スペースの選択および割り当て』を参照してください。新規の表スペース作成の情報については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

- 次に示すのと同じような SQL ステートメントを使用して、コメントを DB2 システム・カタログに追加します。

```
COMMENT ON TABLE MY_PFKEYS IS 'PF KEYS RESERVED FOR FINANCIAL ANALYSTS'
```

PF KEYS RESERVED FOR FINANCIAL ANALYSTS という句が、DB2 システム・カタログの **REMARKS** 列に現れます。システム・カタログへのコメントの追加の詳細については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

新規の表に関するコメントを DB2 システム・カタログに加える必要はありませんが、それを行う場合は、1 つのコメントは表に関するもの、その他は各列について記述するものにします。たとえば、MY_PFKEYS は、私用と共用の機能キーを区別する AUTHID という名前の列をもつものとし、これを説明するコメントを加えるには、次の照会を実行します。

```
COMMENT ON COLUMN MY_PFKEYS.AUTHID
IS 'PRIVATE PFKEY: USE AUTH ID. PUBLIC PFKEY: USE NULL'
```

後続の **COMMENT ON** 照会を実行することにより、現行のものを置き換えることができます。**COMMENT ON** 照会の詳細については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

- 次のステートメントと同じような SQL ステートメントを使用して、索引を作成します。

```
CREATE UNIQUE INDEX MY_PFKEYSX
ON MY_PFKEYS (PANEL, ENTRY_TYPE, NUMBER)
```

列 **PANEL**、**ENTRY_TYPE** および **NUMBER** を索引する **UNIQUE** キーワードを使用して、表に同一の行がないことを確認します。

QMF 機能キーのカスタマイズ

UNIQUE キーワードを使用しないを選択すると、QMF は重複するキー定義を許可します。QMF は、同一キーに対する複数のキー定義を検出した場合、ホーム・パネル上に警告メッセージを表示し、警告メッセージに関する情報をユーザーのトレース・データに書き込みます。ウィンドウ・パネルに対する複数のキー定義について、メッセージは出されません。QMF は最後に検出した定義を使用します。

ユーザーの機能キー定義を表に入力

SQL INSERT ステートメントまたは QMF 表編集機能を使用して、カスタマイズしたキー定義を機能キー表に挿入します。各機能キー定義は表で 2 行にわたります。

- 1 行は、ユーザーがキーを押したとき QMF が発行するコマンドを指定します。
- もう 1 行は、画面に表示されるラベル・テキストを指定します。

カスタマイズしたいキーごとに両方の行を入力してください。関連するラベルをもっていない機能キー・コマンドは、ユーザーの画面には表示されません。同様に、関連するコマンドをもっていないラベルは非アクティブです。

次の 2 つの節では、ユーザーが各行に入力する必要がある値について説明します。

コマンドと機能キーの結び付け

QMF パネルの各機能キーは、その機能キーを押したときに実行される QMF コマンドと結び付けられます。カスタマイズした機能キーも必ずこのように機能するようにするには、表に入力する 2 行の 1 つが 367 ページの表 47 に示す値を必ず持つようにします。

表 47. 機能キー表をカスタマイズする値

列	値	情報
PANEL	カスタマイズする QMF パネルの ID	<p>371ページの『フルスクリーン・パネル ID』は、フルスクリーン・パネルで使用する必要がある ID を示します。372ページの『ウィンドウ・パネル ID』は、特定のウィンドウ・パネルで使用する必要がある ID を示します。</p> <p>ウィンドウ・パネルの 1 つのクラスの各パネルで表示される同じキーのセットを定義する場合は、表の最下部に示されている クラス ID を使用します。たとえば、書式ウィンドウの Specify パネルをカスタマイズする場合、Specify パネルに書式クラスのその他のパネルと異なるキーを表示させたければ、パネル ID FOSPEC を使用します。そうでなければ、そのクラスのすべてのパネルの特徴を示すパネル ID FOXXXX を使用します。</p> <p>クラス ID を使用する変更は、そのクラス ID によりカスタマイズされたすべての パネルに適用されます。ヘルプおよび指示ウィンドウは、固有の ID のセットを持っていません。クラス ID を使用することによってのみカスタマイズすることができます。</p>
ENTRY_TYPE	K	K は、キーを押したときに QMF が発行するコマンドを、この行が定義することを示します。
NUMBER	カスタマイズする機能キーの番号	F5 の定義を変更する場合は、5 をこの列に入力します。

QMF 機能キーのカスタマイズ

表 47. 機能キー表をカスタマイズする値 (続き)

列	値	情報
F_SETTING	キーを押したときに実行されるコマンドのテキスト	<p>このコマンドは、表示されるパネルに適切なコマンドであることを確認してください。たとえば、ENLARGE コマンドは QBE 照会の QUERY パネルだけに適切なコマンドです。QMF は、ユーザーがキーを押すまで、コマンドがそのパネルに対して適切なコマンドかどうかを検査しないため、エンド・ユーザーが新規の機能キーを必要とする前に各キーを検査してください。</p> <p>コマンドは大文字で入力してください。これは、QMF が機能キーと関連付けられたコマンドを検索するとき、端末からの入力を大文字に変換しないためです。この値が小文字で、ユーザー・プロファイルの CASE フィールドに値 UPPER がある場合は、コマンドは実行されません。</p> <p>カスタマイズする各パネルに END または CANCEL に設定されたキーがあることを確認してください。これらのコマンドに定義されたキーがないと、ユーザーがパネルを終了できない場合があります。</p>

NLF を使用する場合: 基本となるコマンドが、正しい各国語変換を使用していることを確認します。さらに、各キーに対するラベル・テキストが、使用している NLF 言語で書かれていると便利です。

機能キーにラベルを付けて画面上に配置

各 QMF パネルの機能キーは、機能キー番号の隣にラベルがあります。画面にラベルが表示されることを確認するには、表に 2 番目の行を追加する必要があります。機能キー表のこの行の各列に以下の値があることを確認してください。

表 48. 機能キー表にラベルを付けるための値

列	値	情報
PANEL	カスタマイズするパネルの ID	これは、定義の最初の行で使用したのと同じ ID ですが、366ページの『コマンドと機能キーの結び付け』で説明されています。
ENTRY_TYPE	L	L は、この行が機能キーと関連付けられているラベルを定義していることを示します。

表 48. 機能キー表にラベルを付けるための値 (続き)

列	値	情報
NUMBER	フルスクリーン・パネルをカスタマイズしている場合、画面上にキーが表示される行の番号です。	ウィンドウまたはヘルプ・パネルをカスタマイズしている場合、NUMBER は機能キーの番号を表しています (366ページの『コマンドと機能キーの結び付け』で表に追加した最初の行と同じです)。たとえば、ホーム・パネルでは、F5 は行 1 に表示され、F12 は行 2 に表示されます。
F_SETTING	機能キー・ラベルのテキスト	フルスクリーン・パネルでは、QMF はユーザーがこの行に入力した通りに画面上に表示します。スペースの調整は行いません。たとえば、QMF ホーム・パネルをカスタマイズしている場合、カスタマイズしているキーもしていないキーも含めて、パネル上に表示されるすべてのキーを入力する必要があります。QMF は、ユーザーがカスタマイズを行わないキーのデフォルト・キー設定を自動的に埋めることはしません。例が 370ページの図95 に示されています。 ウィンドウ・パネルでは、この列にキーのラベルだけ入力する必要があります。例が 370ページの図96 および 371ページの図97 に示されています。

キー定義の例

この節の例は、各タイプの QMF パネルに対する完全な機能キー定義の入力方法の参考に使用してください。例では、フルスクリーン・パネル、ウィンドウ・パネルおよびヘルプ・パネルの更新方法を示しています。

示されている例は、371ページの『カスタマイズしたいパネルの識別』の表からのパネル ID を使用しています。これらの表を使用して、機能キー表の PANEL 列に適切な値を入手してください。

重要: カスタマイズされた各 2 次パネルには、ユーザーがパネルを終了するために CANCEL コマンドに設定したキーがあることを確認してください。

フルスクリーン・パネルでキーの定義を入力

370ページの図95 に示した SQL 照会を使用して、ホーム・パネル上の F2 を EDIT TABLE から IMPORT に変更します。ホーム・パネルをパネル ID HOME で識別し、F2 を押したときにそのコマンドが実行されるようにカスタマイズするため、番号 2 (表示された最初の照会) を示します。

QMF 機能キーのカスタマイズ

```
INSERT INTO MY_PFKEYS (PANEL,ENTRY_TYPE,NUMBER,F_SETTING)
VALUES('HOME','K',2,'IMPORT')
INSERT INTO MY_PFKEYS (PANEL,ENTRY_TYPE,NUMBER,F_SETTING)
VALUES('HOME','L',1,'Help      2=Import      3=End      4=Show      5=Chart      6=Query')
```

図 95. ホーム・パネルで QMF コマンドの機能キーを変更

QMF ホーム・パネルはここで F2 に関してインポートを表示します。

Type command on command line or use PF keys. For help, press PF1 or type HELP.

1=Help	2=Import	3=End	4=Show	5=Chart	6=Query
7=Retrieve	8=Edit Table	9=Form	10=Proc	11=Profile	12=Report

OK, cursor positioned.
COMMAND ==>

2 番目の照会で、各キーをユーザーがカスタマイズしなかった場合でも、F_SETTING 列にホーム・パネルの上側のキー行に表示される通り正確に入力してください。たとえば、2 番目の照会の F_SETTING 列にインポートだけを指定すると、ホーム・パネルは次のようになります。

Type command on command line or use PF keys. For help, press PF1 or type HELP.

インポート

7=Retrieve	8=Edit Table	9=Form	10=Proc	11=Profile	12=Report
------------	--------------	--------	---------	------------	-----------

OK, cursor positioned.
COMMAND ==>

ウィンドウ・パネルのキーの定義の入力

図96 の SQL 照会は、F3 キーを指示照会の表パネルに追加します。この機能キーは、CANCEL コマンドを実行し、CancelMe とラベルが付けられています。

```
INSERT INTO MY_PFKEYS (PANEL,ENTRY_TYPE,NUMBER,F_SETTING)
VALUES('QPTABL','K',3,'CANCEL')
INSERT INTO MY_PFKEYS (PANEL,ENTRY_TYPE,NUMBER,F_SETTING)
VALUES('QPTABL','L',3,'CancelMe')
```

図 96. 指示照会の Specify パネルの機能キーの変更

ヘルプ・パネルまたはプロンプト・パネルへのキー定義の入力

371ページの図97 の SQL 照会は、F13 キーをすべてのヘルプ・パネルに追加します。この機能キーは、CANCEL コマンドを実行し、CancelMe とラベルが付けられています。

```
INSERT INTO MY_PFKEYS (PANEL,ENTRY_TYPE,NUMBER,F_SETTING)
VALUES('HEXXXX', 'K', 13, 'CANCEL')
INSERT INTO MY_PFKEYS (PANEL,ENTRY_TYPE,NUMBER,F_SETTING)
VALUES('HEXXXX', 'L', 13, 'CancelMe')
```

図 97. ヘルプ・パネルまたは指定パネルの機能キーの変更

すべてのヘルプおよびプロンプト・パネルは、単一クラス ID を使用してカスタマイズします。あるクラスの 1 つのパネルに加えた変更は、そのクラス ID で定義されているすべてのパネル上に現れるので、1 つのヘルプまたはプロンプト・パネルに行う変更が、そのクラスのすべてのヘルプまたはプロンプト・パネルに対して適切であることを確認してください。

カスタマイズしたいパネルの識別

この節の表を使用して、機能キー表の PANEL 列にどの ID を入力するのかを決めてください。次のコマンドを使用して、グローバル変数 DSQDC_SHOW_PANID を 1 に設定すると、パネルの左上隅にパネル ID が表示されます。

```
SET GLOBAL (DSQDC_SHOW_PANID=1
```

フルスクリーン・パネル ID

図 98 に、QMF 英語ベース用のフルスクリーン・パネル ID がリストされています。QMF NLF の任意の有効なフルスクリーン・パネル ID のリストを表示するには、QMF NLF の任意のパネルから、QMF コマンド HELP DSQ22957 を入力します。各 QMF NLF の有効なフルスクリーン・パネル ID は、DSQ22957 メッセージの言語固有バージョンにリストされています。機能キー表の PANEL 列に、ここに示してあるとおりに、あるいはメッセージ・テキストに示されているとおりに、ID を入力してください。

PROMPTED QUERY	FORM.BREAK1	FORM.COLUMNS
SQL QUERY	FORM.BREAK2	FORM.CONDITIONS
QBE QUERY	FORM.BREAK3	FORM.DETAIL
PROC	FORM.BREAK4	FORM.FINAL
PROFILE	FORM.BREAK5	FORM.MAIN
REPORT	FORM.BREAK6	FORM.OPTIONS
GLOBALS	FORM.CALC	FORM.PAGE
HOME		

図 98. QMF 英語ベースのフルスクリーン・パネル ID

ウィンドウ・パネル ID

この節の表を使用して、ウィンドウ・パネル ID を参照してください。パネル ID が表示されるようにグローバル変数 DSQDC_SHOW_PANID を設定した場合は、画面に表示される場合、これらの表中に示されている各 ID には、4 文字が先頭に置かれているのがわかります。

表に名前が付けられていないウィンドウ・パネルは、固有のパネル ID を持っていないので、各表の最下部に示されているクラス ID を使用してカスタマイズしてください。すべてのクラス ID には、その中に文字ストリング XXXX があります。これらの文字は可変文字ではなく、実際の ID の一部です。

コマンド・ウィンドウ

パネル ID	表題または説明
COENTR	コマンド項目
COXXXX	コマンド・ウィンドウ・クラス

書式ウィンドウ

パネル ID	表題または説明
FOALIG	配置
FODFIN	定義
FOSPEC	指定
FOXXXX	書式ウィンドウ・クラス

グローバル変数ウィンドウ

パネル ID	表題または説明
GLADVA	追加変数
GLSHVA	表示変数
GLXXXX	グローバル変数ウィンドウ・クラス

ヘルプおよびプロンプト・ウィンドウ

パネル ID	表題または説明
HEXXXX	ヘルプ・ウィンドウ・クラス
PRXXXX	プロンプト・ウィンドウ・クラス

ロケーション・ウィンドウ

パネル ID	表題または説明
PLLOCA	ロケーション・ウィンドウ・リスト

オブジェクト・リスト・ウィンドウ

パネル ID	表題または説明
OBDESC	オブジェクト記述
OBLIAC	オブジェクト・リスト：処理
OBLIMU	オブジェクト・リスト：複数選択
OBLISI	オブジェクト・リスト：単一選択
OBSORT	オブジェクト・リスト・ソート
OBXXXX	オブジェクト・リスト・ウィンドウ・クラス

指示照会ウィンドウ

パネル ID	表題または説明
QPCDCH	条件コネクタ - 変更内容
QPCDIT	条件コネクタ
QPCOCH	列 - 変更内容
QPCODE	列の説明
QPCOFI	列要約機能項目
QPCOFU	列要約機能
QPCOLI	列名リスト
QPCOLU	列
QPDUCH	重複行 - 変更内容
QPDUPL	重複行
QPEXPR	式
QPJOCO	結合列
QPJOTA	結合表
QPROBE	行 - 間
QPROCH	行 - 変更内容 (左側)
QPROCT	行 - 含む
QPROC1	行 - 比較演算子 1
QPROC2	行 - 比較演算子 2
QPROEN	行 - 終了

QMF 機能キーのカスタマイズ

パネル ID	表題または説明
QPROEQ	行 - 等しい
QPROGQ	行 - より大きい、または等しい
QPROGR	行 - より大きい
QPROLQ	行 - より小さい、または等しい
QPROLS	行 - より小さい
QPROST	行 - 開始
QPROWS	行 (行条件)
QPSHFI	フィールド表示
QPSHSQ	SQL 表示
QPSOCH	ソート - 変更
QPSORT	ソート
QPSPEC	指定
QPTABL	表
QPXXXX	PQ ウィンドウ・クラス

新規機能キ一定義のアクティブ化

ユーザーがカスタマイズされた機能キ一定義を使用できるようにするには、以下のことを行います。

1. ユーザー・プロファイルの PFKEYS フィールドを、機能キ一定義表の名前で更新します。

たとえば、375ページの図99 のような照会を使用して、英語 QMF ユーザーの JONES に表 MY_PFKEYS を、またドイツ語 NLF ユーザーの SCHMIDT に表 MEIN_FKY を割り当てます。Q.PROFILES 表を更新する照会の列 TRANSLATION および ENVIRONMENT には、常に値を入れてください。

基本 QMF (英語)

ドイツ語 NLF

UPDATE Q.PROFILES

UPDATE Q.PROFILES

SET PFKEYS = 'MY_PFKEYS'

SET PFKEYS = 'MEIN_PFKY'

WHERE CREATOR='JONES'

WHERE CREATOR='sCHMIDT'

AND TRANSLATION = 'ENGLISH'

AND TRANSLATION = 'DEUTSCH'

AND ENVIRONMENT = 'CICS' (or 'TSO')

AND ENVIRONMENT = 'CICS' (or 'TSO')

図 99. カスタマイズされた機能キーをユーザーからアクセスが可能にする

2. 表にアクセスする必要があるユーザーに、SQL SELECT 特権を付与します。

すべてのユーザーが表を使用するために表が割り当てられるように、SELECT 特権を PUBLIC に付与します。たとえば、次のように指定します。

```
GRANT SELECT ON MY_PFKEYS TO PUBLIC
```

インストール先での機能キーの保守を最小化するために、表の視点を割り当てることができます。SELECT 特権を視点についてだけ付与して、ユーザーが使用する以外の目的で機能キーにアクセスできないようにします。

機能キー表の視点を割り当てる手順は、357ページの『コマンド同義語表の保守の最小化』で説明しているコマンド同義語表と同じです。その節で説明している方針を使用して、表または視点を個々のユーザーまたはユーザーのグループに割り当てるかどうかを決定してください。

3. 現行 QMF セッションを終了し、新規の機能キーをアクティブ化するために別のセッションを開始するようにユーザーに指示します。

機能キー表の検査と問題診断

機能キー表の名前を Q.PROFILES 項目に挿入して新規の機能キー定義をアクティブ化させれば、新規定義は検査を受けられる状態になります。新規定義は 2 つの条件中 1 つが満たされなければ、有効になりません。

- QMF をクローズして新規に QMF セッションを開始する。

QMF 機能キーのカスタマイズ

- QMF 内から、CONNECT TO *locname* コマンドを入力して再度 QMF に接続します。ここで、*locname* は QMF ホーム・パネルに表示されるロケーション名と同じです。

上記 2 つのアクションのいずれかを行った後 Warning messages have been generated というメッセージが表示されたら、QMF を終了し、QMF トレース・データ (DSQDEBUG) 出力を検査します。トレースは、問題を修正するのに使用できるメッセージを提供します。QMF に再接続した後、新規の機能キ一定義が見つからない場合は、Q.SYSTEM_INI プロシージャまたは他のユーザー制御フィーチャーによってメッセージ Warning messages have been generated が隠されている可能性があります。この場合は QMF を終了して、DSQDEBUG トレース出力を検査します。

QMF トレース・データにエラーがない場合、SHOW GLOBALS コマンドを発行してグローバル変数 DSQAP_PFKKEY_TABLE を検査します。このグローバル変数に新規に作成または変更した機能キー表の名前が含まれていない場合には、Q.PROFILES の行入力を検査します。

第20章 QMF 書式のためのユーザー独自の編集コードの作成

注: 本章には、汎用プログラミング・インターフェースとそれに関連する、指針としての情報が含まれます。

QMF 書式 は、データベースから戻されるデータのフォーマットをユーザーが制御するのに役立ちます。QMF 書式の MAIN および COLUMNS パネルの EDIT 列にある編集コードを使用して、報告書データをさまざまな方法でフォーマット設定します。たとえば、給与データを戻す列では 10 進数編集コードを使用します。この編集コードは、数値データに通貨記号を付けて 10 進数にフォーマット設定します。

QMF と共に提供されている編集コードが、インストール先の報告書編集の要件に合わない場合は、本章の情報を使用して FORM.MAIN および FORM.COLUMNS パネルの EDIT 列で使用するユーザー独自の編集コードを作成することができます。QMF 解説書 は QMF で提供される編集コードを示しています。

また、本章では、ユーザーの編集コードで記述されたデータをフォーマット設定する際の、アセンブラー、PL/I、または COBOL での編集出口ルーチンの書き方も示しています。QMF では、ユーザーの編集出口ルーチンへの標準インターフェースと、ユーザー独自のルーチンを書くための開始点となるサンプル編集出口プログラムの両方を提供しています。

CICS で実行される QMF では、31 ビット・アドレッシングが必要です。

本章の作業を開始する前に、報告書のフォーマット設定と編集コードについての QMF 機能を説明している QMF 解説書 の節を読み返してください。

クイック・スタート

表49 のステップを、ユーザー編集出口ルーチンを作成する指針として使用してください。各ステップに関するより詳細な情報が必要な場合は、表の右端に示されているページを参照してください。

QMF 書式のためのユーザー独自の編集コードの作成

表 49. ユーザー編集出力ルーチンの作成

作業	参照箇所
作成するルーチンで何を行いたいかを決め、そのルーチンを識別する編集コードを選択する。編集コードには、Uxxxx または Vxxxx を使用する。ここで、xxxx は 0 から 4 桁の英字であり、埋め込みブランクやヌル値は含まない。	378 ページ
出力ルーチンがデータをフォーマット設定するように要求する。IBM 提供のインターフェース制御ブロックのフィールドを使用する。	381 ページ
出力ルーチンからのパラメーターを受け入れ、フォーマット設定した結果を出力ルーチンに戻す。インターフェース制御ブロックで提供されている標準入出力フィールドを使用する。	383 ページ
QMF が終了したときに、ユーザーの編集出力ルーチンに制御が渡されるように要求する。インターフェース制御ブロックのフィールドの終了スイッチを設定することによってこれを行う。ルーチンが記憶域の解放などの終結処置アクティビティーを実行する必要がある場合、編集出力ルーチンに制御を渡すことがある。	387 ページ
編集出力ルーチンを HLASM やアセンブラーで作成する場合、まず、IBM 提供のサンプル・アセンブラー・プログラムから始めてください。プログラムの作成の終了後、プログラムの変換、アセンブル、およびリンク・エディットを行い、それを CICS に対して定義する。もしくは、プログラムを TSO 用にアセンブルし、リンク・エディットする。	388 ページ
ユーザーの編集出力ルーチンを PL/I で作成するために、IBM 提供のサンプル PL/I プログラムを開始する。プログラムの作成の終了後、プログラムの変換、コンパイル、およびリンク・エディットを行い、それを CICS に対して定義する。もしくは、プログラムを TSO 用にコンパイルし、リンク・エディットする。	397 ページ
編集出力ルーチンを IBM COBOL (MVS および VM 版)、COBOL/370 または VS COBOL II で作成する場合、IBM 提供のサンプル COBOL プログラムから始めてください。プログラムの作成の終了後、プログラムの変換、コンパイル、およびリンク・エディットを行い、それを CICS に対して定義する。もしくは、プログラムを TSO 用にコンパイルし、リンク・エディットする。	412 ページ

編集コードを選択する

ユーザー編集出力ルーチンが処理する Uxxxx もしくは Vxxxx 編集コードを作成します。U コードについては、編集ルーチンに渡されたデータは、ソース・データの内部データベース表現になります。V コードでは、数値データは文字ストリングに変換され、この文字ストリングが編集プログラムに渡されません。

どちらのコードも、文字または数値データのいずれかの処理を指示することができます。U および V は大文字でなければなりません。xxxx を端末から入力可能な 0 から 4 個の文字 (英字、数字、または特殊文字) で置き換えま

す。埋め込み空白やヌルは使用できません。以下に示す例は、いずれも有効な U タイプおよび V タイプの編集コードです。

```
U1    UAB42    V_1    VX%5
```

ソース・データが文字の場合は、いずれのタイプのコードでも同じように簡単に処理されます。フォーマット設定で算術演算が必要な場合は、数値ソースについては U コードを使用し、その他の場合は V コードを使用するように考慮してください。データ・タイプが拡張浮動小数点の場合、プログラミング言語がこれをサポートしていることを確認してください。たとえば、VS COBOL II は拡張浮動小数点データを処理できません。このような場合は、V コードを使用することをお勧めします。

数値データを含む V コードの場合、QMF はデータを文字フォーマットに変換してから、ユーザー編集ルーチンを呼び出します。変換された数値の長さは、表50 で示しているように、元のデータ・タイプによって決まります。

表 50. QMF がデータ・タイプに応じて数値データを変換する方法

元の数値データのデータ・ QMF が変換する長さ	
タイプ	
短精度整数	5
整数	11
10 進数	元のデータの精度に等しい (元のデータが偶数の場合は、奇数に繰り上げられます。)
浮動小数点	15 以上 (10 を基数とする指数により異なる)
拡張浮動小数点	30 以上 (10 を基数とする指数により異なる)

編集コードを数値データの処理または文字データの処理に限定する必要はありません。QMF と一緒に提供されているサンプルの編集ルーチンでは、1 つの編集コードで数値と文字データの両方を処理しています。

ユーザーのプロファイルの CASE フィールドの値が UPPER または STRING の場合は、QMF は端末から入力されるすべての入力を大文字に変換するため、編集コードを確認できない場合があります。編集コードが大文字と小文字の混在を受け入れるように作成されている場合には、混在が設定されている時にその編集コードを入力します。

DATE、TIME、および TIMESTAMP 情報の処理

インストール先システムが日付 / 時刻データ・タイプをサポートしている場合には、DATE、TIME、および TIMESTAMP のデータ・タイプで列をフォーマット設定することができます。これによって、ユーザーはローカルの日時 / 時刻の出力ルーチンを使用することができます。これらのデータ・タイプの詳細については、*QMF 使用の手引き* を参照してください。

編集ルーチンは、他のデータ・タイプの列のデータをフォーマット設定できるのと同じように、これらの列のデータをフォーマット設定することができます。1 つの違いは、フォーマット設定される値が制御ブロック・フィールド ECSINPT に現れる場合は、処理されるコードが U コードまたは V コードに関係なく、常に文字ストリングとして渡されるということです。ストリングのフォーマットについては、表51 で説明しています。

表 51. DATE、TIME、および TIMESTAMP データのフォーマット設定

データ・タイプ	ストリングの形式
DATE データ	<p>yyyy-mm-dd。ここで、</p> <p>yyyy 年を指定。常に、4 桁の数字。</p> <p>mm 月を指定 (01 は 1 月、... 12 は 12 月)。常に、2 桁の数字で、先行ゼロが可能。</p> <p>dd 日を指定。常に、2 桁の数字で、先行ゼロが可能。</p> <p>ダッシュ (-) は実際のダッシュを意味します。</p> <p>たとえば、1990-12-12 は、1990 年 12 月 12 日です。</p>
TIME データ	<p>hh.mm.ss。ここで、</p> <p>hh 時刻を指定 (00 から 23 の 24 時間表示による)。常に、2 桁の数字で、先行ゼロが可能。</p> <p>mm 分を指定。常に、2 桁の数字で、先行ゼロが可能。</p> <p>ss 秒を指定。常に、2 桁の数字で、先行ゼロが可能。</p> <p>ピリオドは実際のピリオドを意味します。</p> <p>たとえば、13.08.36 は、米国で一般に使用されている表記法では、1:08 P.M. 36 秒となります。</p>

表 51. DATE、TIME、および TIMESTAMP データのフォーマット設定 (続き)

データ・タイプ	ストリングの形式
TIMESTAMP データ	yyyy-mm-dd-hh.mm.ss.nnnnnn。ここで、 yyyy-mm-dd DATE データの場合と同様に日付を指定。 hh.mm.ss TIME データの場合と同様に時刻を指定。 nnnnnn 6 桁の数字で、百万分の一秒単位までカウント。 たとえば、1990-12-12-13.08.36.123456 は、米国で一般に使用されている表記法では、1990 年 12 月 12 日の 1:08 P.M. 36.123456 秒となります。

使用できるデータ・タイプについては、384ページの表52 の ECSINTYP フィールドを参照してください。

データをフォーマット設定するためのユーザー出口ルーチンの呼び出し

図100 は、QMF とユーザー編集出口ルーチンの作業が、ユーザーによって定義された編集コードを使用して、どのようにデータをフォーマット設定するか示しています。

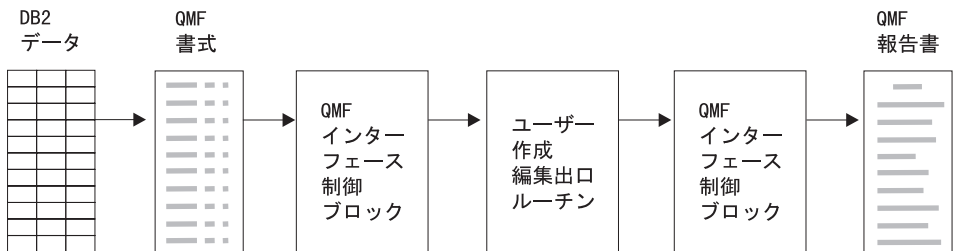


図 100. ユーザー編集ルーチンと QMF との関係

ユーザー独自のコードを FORM.MAIN または FORM.COLUMNS の列に入力すると、QMF はデータの特徴を 図100 に示される最初のインターフェイス制御ブロックに渡します。この特性は、384ページの『インターフェイス制御ブロックのフィールド』で説明している制御ブロックの特定のフィールドに常駐します。QMF はまた、フォーマット設定されるデータを入力域に、またフォーマット設定の結果を保持する出力域に渡します。

IBM は、サンプル編集出口ルーチンの 6 つの異なるバージョンを QMF710.SDSQSAPE で提供しています。

QMF 書式のためのユーザー独自の編集コードの作成

言語	TSO およびネイティブ OS/390 バッチ	CICS
COBOL	DSQUXDTC	DSQUXCTC
PL/I	DSQUXDTP	DSQUXCTP
アセンブラー	DSQUXDTA	DSQUXCTA

サンプル・プログラムは、2 つの編集コードをサポートします。

VSS ダッシュを社会保障番号または文字ストリングに追加します。

UDN プログラム内部の表を使用して、部門番号を部門名に変換します。

サンプル・プログラムには、ユーザーがより簡単に、ユーザー編集ルーチンの作業が分かるようにコメントが付けられています。ユーザーのプログラムを作成するためのテンプレートとして、サンプルを使用することができます。

QMF はユーザー編集ルーチンを、TSO、SRPI、APPC、およびネイティブ OS/390 (DSQUEDIT) では再配置可能モジュールとして、また、CICS (DSQUECIC) では再入可能モジュールとして提供します。これらは QMF ライブラリー QMF710.SDSQLOAD に置かれています。編集ルーチンを使用する準備ができたならば、QMF 提供のモジュールを削除または名前変更します。図 101 は、編集ルーチンの一般的な構造を示しています。

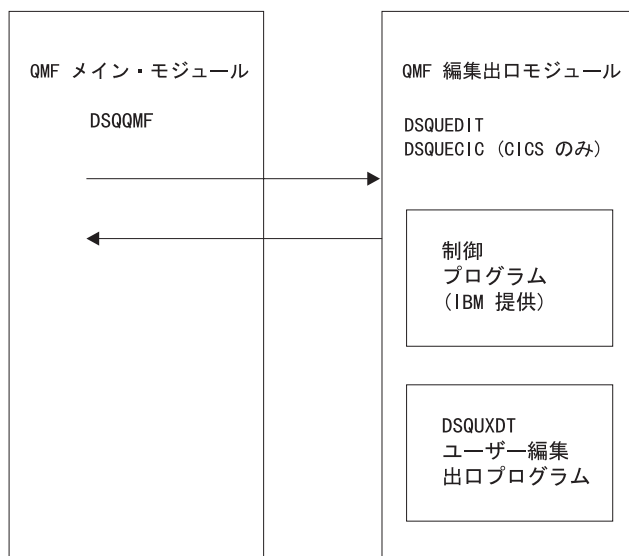


図 101. 編集ルーチンの一般的なプログラム構造

出力ルーチンとの情報の受け渡し

データベースから戻されたデータをフォーマット設定するために、QMF はユーザーの編集出力ルーチン呼び出し、インターフェース制御ブロックのフィールドを介して情報を渡します。また、入出力域を使用して出力ルーチンとの情報の受け渡しが行われます。その中には、フォーマット設定されるデータベースのデータとフォーマット設定された結果をどこに入れるかについての情報が含まれます。

フォーマット設定されるデータとしては、列値、組み込み関数の結果、定義された列、計算、あるいはヘッダー、後書き、最終合計行の変数で表される値があります。

フォーマット設定の制御を受け取ると、ユーザーの編集ルーチンは以下のリストのパラメーターを取り出します。

- インターフェース制御ブロック。
- ECSINPT の値。すなわち、フォーマット設定される入力域からのデータ。
- ECSRSLT の値。すなわち、フォーマット設定された結果を含む出力域。ECSRSLEN は、呼び出しのたびにこの出力域に実際に渡された記憶域の量を含んでいます。結果は、折り返し列にすることはできません。

重要: ECSRSLEN フィールドに示されている以上のメモリーを出力域で使わないでください。使用すると、QMF エラー DSQ60439 - User edit program memory overwrite (ユーザー編集プログラムのメモリー上書き) が出されます。

ユーザー編集プログラムの変更が必要になることがあります。このアプリケーション・エラーを訂正するためには、次のいずれかを行ってください。

- FORM の編集コードを報告書で要求される正しい長さに変更することによって、COLUMN の WIDTH を広げます。
- ECSRSLEN の長さを検査し、QMF に渡される結果をユーザーのプログラムで PAD (埋め込み) または TRUNCATE (切り捨て) するかどうか判別します。

ECSINPT、ECSRSLT、および ECSRSLEN は、384ページの表52 で説明しているインターフェース制御ブロックのフィールドです。

インターフェース制御ブロックのフィールド

インターフェース制御ブロックのフィールドを使用して、ユーザー出力ルーチンとの間で情報の受け渡しを行います。アセンブラー、PL/I、または COBOL で機能するインターフェース制御ブロックはそれぞれ別々のものですが、インターフェース制御ブロックのフィールドは、ユーザー編集出力ルーチンの作成に使われているプログラム言語に関係なく標準のものです。これらのフィールドは、表52 に示されています。特に記述のない限り、それぞれのフィールドはすべてのフォーマット設定の呼び出しに関連しています。

これらと同じフィールドが、QMF と一緒に出荷されたサンプル・プログラム (サポートされるそれぞれのプログラム言語ごとに 1 つ) に現れます。これらのフィールド名をユーザーのソース・プログラムに含めることができます。サンプル・プログラムは、QMF プロダクション・ディスクにあります。

表 52. QMF インターフェース制御ブロックのフィールド

名前	内容												
ECSDEOPT	PROFILE の DECOPT オプションにより決められた現行の小数点記号 (ピリオドまたはコンマ) を含んでいる。												
ECSECODE	ユーザー編集コードが入っている。												
ECSERRET	呼び出し時点では、ゼロが入っている。エラーを記録するには、これをゼロ以外のリターン・コードに設定する。以下のリストのいずれかの値を使用して、エラーのタイプを示す。 <table border="0"> <thead> <tr> <th>番号</th> <th>エラー</th> </tr> </thead> <tbody> <tr> <td>99101</td> <td>認識できない編集コード</td> </tr> <tr> <td>99102</td> <td>編集コードに不適切な入力データ・タイプ</td> </tr> <tr> <td>99103</td> <td>フォーマット設定される項目に無効な入力値</td> </tr> <tr> <td>99104</td> <td>フォーマット設定される項目が短過ぎる</td> </tr> <tr> <td>99105</td> <td>ECSRSLT の結果用にスペースが不十分 (結果が割り当てスペースに対し幅が広すぎる)</td> </tr> </tbody> </table> リストされているエラー・コード (および、その関連メッセージとヘルプ・パネル) は、エラーに固有。その他のコードの場合は、汎用のバックアップ・ヘルプ・パネルとともに、一般的なエラー・メッセージが表示される。	番号	エラー	99101	認識できない編集コード	99102	編集コードに不適切な入力データ・タイプ	99103	フォーマット設定される項目に無効な入力値	99104	フォーマット設定される項目が短過ぎる	99105	ECSRSLT の結果用にスペースが不十分 (結果が割り当てスペースに対し幅が広すぎる)
番号	エラー												
99101	認識できない編集コード												
99102	編集コードに不適切な入力データ・タイプ												
99103	フォーマット設定される項目に無効な入力値												
99104	フォーマット設定される項目が短過ぎる												
99105	ECSRSLT の結果用にスペースが不十分 (結果が割り当てスペースに対し幅が広すぎる)												
ECSFREQ	フォーマット設定の呼び出しでは E に、終了呼び出しでは T になる。												
ECSINLEN	フォーマット設定される値のバイトで表した長さが入っている。												
ECSINNUL	フォーマット設定される値がヌルの場合は、N になります。												

表 52. QMF インターフェース制御ブロックのフィールド (続き)

名前	内容
ECSINPRC	フォーマット設定される値の精度を含む。データ・タイプが DECIMAL の場合は U タイプ・コードのみに適用され、フォーマット設定される文字ストリングが数値データから来ている場合には、V タイプ・コードのみに適用される。
ECSINSCL	フォーマット設定される値の位取りを含む。データ・タイプが DECIMAL の場合は U タイプ・コードのみに適用され、フォーマット設定される文字ストリングが数値データから来ている場合には、V タイプ・コードのみに適用される。
ECSINSGN	変換される数値の符号 (ブランクまたは -) が入っている。フォーマット設定される文字ストリングが数値データから来ている場合には、V タイプ・コードのみに適用される。
ECSINTYP	データベース用語で、フォーマット設定される値が表現される方法を示す。すべてのタイプの編集コードに適用される。以下の値を取ることができる。 384 DATE データ・タイプ 388 TIME データ・タイプ 392 TIMESTAMP データ・タイプ 448 VARCHAR データ・タイプ 452 CHAR データ・タイプ 456 LONG VARCHAR データ・タイプ 464 VARGRAPHIC データ・タイプ 468 GRAPHIC データ・タイプ 472 LONG VARGRAPHIC データ・タイプ 480 FLOAT データ・タイプ 484 DECIMAL データ・タイプ 496 INTEGER データ・タイプ 500 SMALLINT データ・タイプ 940 拡張浮動小数点データ・タイプ 拡張浮動小数点データ・タイプは、データベース (または COBOL) ではサポートされない。AVERAGE および STDEV などの関数に制限される。拡張浮動小数点の値は、30 桁以上の精度を持っている。
ECSNAME	制御ブロックの名前である DXEECS が入っている。記憶域ダンプで、目印としての役割を果たす。
ECSRQMF	T に設定されると、終了呼び出しが要求される。
ECSRSLEN	バイトで表した出力域の長さが入っている (その値は、FORM の WIDTH 列から取られる)。
ECSTHSEP	PROFILE の DECOPT オプションにより決定される千単位区切り文字を含む (ブランクまたはコンマ)。

QMF 書式のためのユーザー独自の編集コードの作成

表 52. QMF インターフェース制御ブロックのフィールド (続き)

名前	内容
ECSUSERS	ある呼び出しから次の呼び出しへそのまま渡される情報を出力ルーチンが記録できる 256 バイトのスクラッチパッド域。編集ルーチンのロード後の最初の呼び出しでは、このフィールドには 2 進ゼロが入っている。

入力域の特徴を示すフィールド

制約事項: この節は、DATE、TIME、および TIMESTAMP 列からの値には適用されません。これらのタイプの値については、380ページの『DATE、TIME、および TIMESTAMP 情報の処理』を参照してください。

セッション中に、サブプログラム DSQUXDT はさまざまな編集コードの保守の必要が生じる場合があります。そのような場合は、ユーザーのルーチンはエグゼクティブ・ルーチンにするように考慮してください。これは、渡された編集コードの分析のみを行い、その後で実際にフォーマット設定を行う適切なルーチンを呼び出すというものです。このような設計にしておくと、新規のユーザー編集コードが考案された時に、ソース・コードの理解と変更が容易になります。

インターフェース制御ブロックのフィールドに加えて入力フィールドも、ユーザーの編集出力ルーチンはフォーマット設定されるデータについての情報を受け取ります。

フォーマット設定される値は ECSINPT のフィールドに現れます。それがどのように表現されるかは、次の 2 つの要因によって異なります。

- フォーマット設定される値が数値であるか、文字であるか。これは、ECSINTYP フィールドによって決まります。
- 編集コードが U コードであるか、V コードであるか。これは、ECSECODE フィールドによって決まります。

入力域での U タイプ編集コードの表現方法

数値は内部データベース・フォーマットで表現されます。たとえば、ECSINTYP が 496 (INTEGER データ・タイプ) の場合、値はフルワードの整数です。また、484 (DECIMAL データ・タイプ) の場合は、値は 10 進数フォーマットです。10 進数フォーマットの位取りと精度は、ECSINSCL および ECSINPRC フィールドにあります。長さ (バイト) は ECSINLEN フィールドにあります。

定義された列、計算、および合計値からの数値データは、拡張浮動小数点値として戻されますが、DB2 はこのデータ・タイプを明示的にはサポートしていません。長さ (16 バイト) は ECSINLEN フィールドにあります。

文字や漢字の値は、それぞれの内部文字ストリング・フォーマットで表現されます。唯一の例外として、可変長ストリング (たとえば、VARCHAR データ・タイプ) の場合にはストリングそのものだけが表現され、その前にある長さフィールドは現れません。文字値の場合はすべて、ストリングの長さ (バイト) は ECSINLEN にあります。

入力域での V タイプ編集コードの表現方法

数値は、数字ストリングで表現されます。長さは ECSINLEN フィールドにあります。必要に応じて、先行または後書きのゼロでストリングが埋められます。

ストリングには符号や小数点は含まれていません。その代わりに、符号はブランクまたは負符号として、ECSINSGN フィールドに現れ、小数点の位置は ECSINSCL フィールドにあります。たとえば、ECSINPT にあるストリングが 12345、ECSINSGN がブランク、かつ ECSINSCL が 3 に等しいとすると、表現される値は +12.345 となります。

文字または漢字の値は、それぞれの文字ストリングで表現されます。文字値の場合はすべて、ストリングの長さ (バイト) は ECSINLEN にあります。

出力域の特徴を示すフィールド

ECSRSLT フィールドは、そのフィールドを完全に埋める文字ストリングのフォーマットであり、フォーマット設定された出力を受け取ります。入力時には、このフィールドは常にブランクです。このフィールドの長さ (バイト) は ECSRSLEN フィールドにあります。QMF は編集ルーチンを呼び出す前に ECSRSLT をブランクにします。

QMF 終了時に制御を出力ルーチンに渡す

制御ブロックの ECSRQMF フィールドを使用して、QMF の終了時にはユーザー出力ルーチンが必ず制御を受け取るように指示します。編集出力ルーチンが最初に制御を受け取る時に、ECSRQMF 値を更新するようにしてください。

QMF 終了時にユーザー編集出力ルーチンが制御を受け取る際、ルーチンに渡されるパラメーターは制御ブロック、入力域、および出力域です。制御ブロックのみに、使用できる情報が含まれています。

編集ルーチンを HLASM またはアセンブラで作成

TSO、SRPI、APPC、CICS、およびネイティブ OS/390 用の編集ルーチンを、アセンブラで作成することができます。

TSO、SRPI、APPC およびネイティブ OS/390 用の編集ルーチンをアセンブラで作成

TSO、SRPI、APPC およびネイティブ OS/390 のアセンブラ用の QMF 編集出口インターフェースは、以下の部分から構成されています。

- インターフェース制御ブロック。DXEECSA として QMF と共に出荷。
- 制御プログラム。DSQUXIA として QMF と共に出荷。
- ユーザーの編集出口プログラム。名前は DSQUXDT。

図102 は、TSO、SRPI、APPC またはネイティブ OS/390 用のアセンブラ編集出口ルーチンのプログラム構造を示しています。

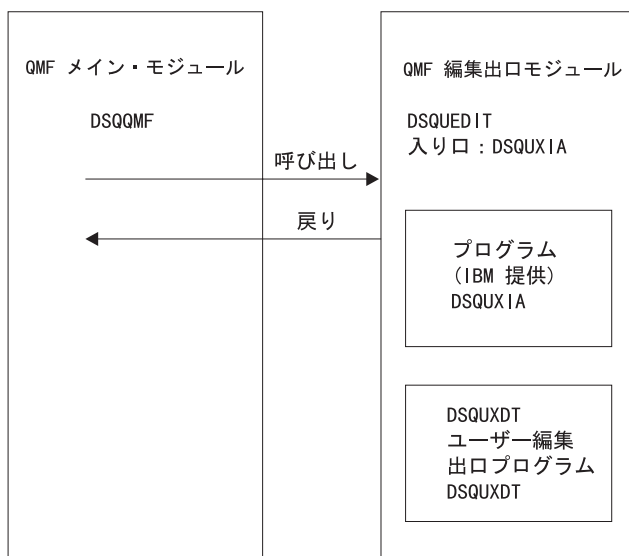


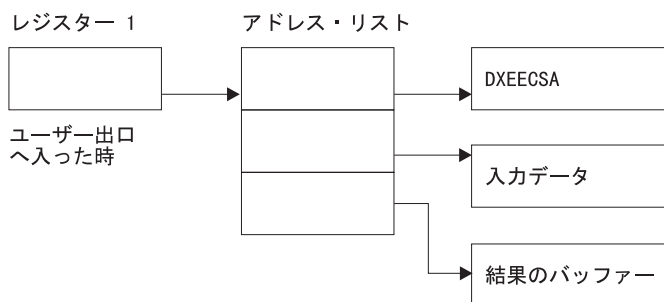
図102. TSO、SRPI、APPC、またはネイティブ OS/390 用のアセンブラ編集出口ルーチンのプログラム構造

アセンブラ編集ルーチンと TSO、SRPI、APPC およびネイティブ OS/390 との対話方法

ユーザー編集プログラムは、TSO、SRPI、APPC およびネイティブ OS/390 では、標準の CALL ステートメントを使用して、サブルーチンとして呼び出されます。ユーザーの編集出口プログラムへ入力の際には、以下の条件が存在します。

QMF 書式のためのユーザー独自の編集コードの作成

- レジスター 1 には、標準パラメーター・リストのアドレスが入っています。



- レジスター 13 には、標準 SAVE 域のアドレスが入っています。
- レジスター 14 には、呼び出し側 (QMF) の戻りアドレスが入っています。

DXEECS のアセンブラー DSECT が DXEECSA として QMF と共に出荷されており、ライブラリー QMF710.SDSQSAPE の中にあります。アセンブラー COPY ステートメントを使用して、この DSECT をユーザーのプログラムに組み込みます。

標準的な規則に従って、レジスターを呼び出し時の値に復元し、次にレジスター 14 のアドレスに戻すことによって、制御を QMF に戻します。

プログラムのアセンブル

アセンブル中は、TSO、SRPI、APPC、またはネイティブ OS/390 の QMF サンプル・ライブラリー QMF710.SDSQSURE にある QMF 編集出口インターフェース制御ブロック DXEECSA が、マクロ・ライブラリーで使用可能になっている必要があります。

プログラムのリンク・エディット

QMF モジュール・ライブラリー QMF710.SDSQLOAD にある IBM 提供の制御モジュール DSQUXIA を、ユーザー編集プログラム DSQUXDT に組み込むことにより、新規の QMF 編集出口モジュール DSQUEDIT を作成します。IBM 提供の制御モジュール DSQUXIA を入り口点として指定する必要があります。

モジュール DSQUEDIT は、24 ビットもしくは 31 ビットのアドレッシング・モードのいずれでも実行可能です。QMF は 31 ビット・アドレッシング・モードで実行しますが、出口モジュール DSQUEDIT が 24 ビット・アドレッシング・モードであれば、自動的に 24 ビット・アドレッシング・モードに切り替わります。

31 ビット・アドレッシング・モードをお勧めします。

QMF 書式のためのユーザー独自の編集コードの作成

アセンブルおよびリンク・エディットのステートメントの例 (TSO、SRPI、APPC、またはネイティブ OS/390)

以下のステートメントの例は、TSO、SRPI、APPC、またはネイティブ OS/390 における、アセンブルとリンク・エディットのジョブのものであります。

```
//sampasm JOB
//STEP1 EXEC PROC=ASMHCL
//* Provide Access to QMF Edit Macro DXEECSA
//C.SYSLIB DD DSN=QMF710.SDSQSRE,DISP=SHR
//C.SYSIN DD *
        .
        Your program or copy of QMF sample DSQUXDTA
        .
/*
/** Provide Access to QMF Interface Module
//L.QMFLOAD DD DSN=QMF710.SDSQLOAD,DISP=SHR
//L.SYSIN DD *
        INCLUDE QMFLOAD(DSQUXIA)
        ENTRY DSQUXIA
        MODE AMODE(31) RMODE(ANY)
        NAME DSQUEDIT(R)
/*
```

プログラム例

DSQUXDTA という名前の IBM 提供の編集出口プログラムの例が、QMF サンプル・ライブラリー QMF710.SDSQSAPE にあります。このプログラムにはコメントが数多く付いており、オンラインで見ると印刷することも可能です。また、ユーザーの要件に合わせて変更することもできます。このプログラムを使用する場合は、ユーザーのプログラム・ライブラリーにコピーし、名前を DSQUXDT に変更してください。

編集ルーチンを CICS 用にアセンブラで作成

CICS におけるアセンブラ用の QMF 編集出口インターフェースは、以下の部分から構成されています。

- インターフェース制御ブロック。DXEECSA として QMF と共に出荷。
- CICS プロローグ・マクロおよびエピローグ・マクロ。DFHEIENT および DFHEIRET として CICS と共に出荷。
- CICS コマンド・インターフェース・モジュール。DFHEAI および DFHEAIO として CICS と共に出荷。
- ユーザーの編集出口プログラム。名前は DSQUECIC。

391ページの図103 は、CICS のアセンブラ編集出口ルーチンのプログラム構造を示しています。

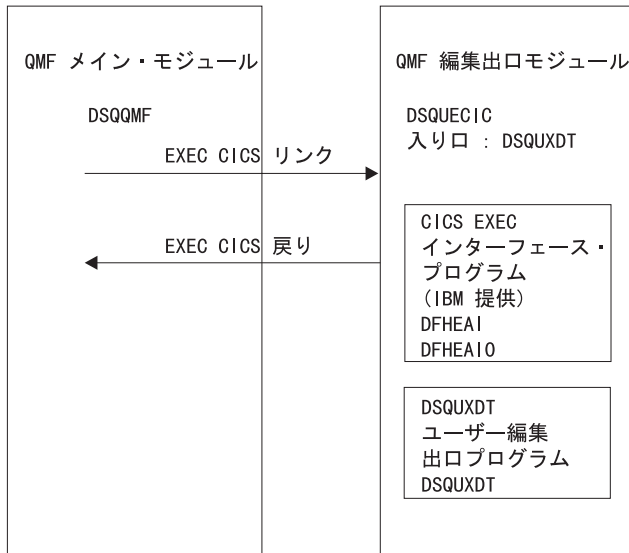


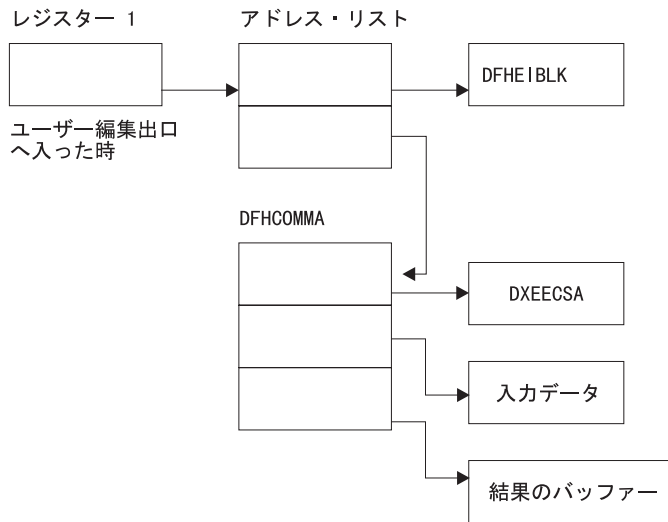
図 103. CICS のアセンブラー編集出口ルーチンのプログラム構造

アセンブラー編集ルーチンと CICS との対話方法

ユーザー編集プログラムは、標準の CICS LINK コマンド・インターフェースを使用して、呼び出されます。ユーザーのプログラムは、メインの QMF プログラムとは異なるプログラム・レベルで実行されています。ユーザーの編集出口プログラムへ入力の際には、以下の条件が存在します。

- レジスター 1 には、CICS 提供のマクロ DFHEIENT および DFHEIRET による処理に適している標準の CICS パラメーター・リストのアドレスが入っています。

QMF 書式のためのユーザー独自の編集コードの作成



- レジスター 13 には、CICS 提供のマクロ DFHEISTG で記述されているとおりの、標準の CICS 作業記憶域のアドレスが入っています。

DXEECS のアセンブラー DSECT が DXEECSA として QMF と共に出荷されており、ライブラリー QMF710.SDSQSAPE の中にあります。アセンブラー COPY ステートメントを使用して、この DSECT をユーザーのプログラムに組み込みます。

標準の CICS RETURN コマンドを使用して、QMF に制御を戻します。

プログラムの変換

アセンブラー用の CICS 変換プログラムを使用して、ユーザーのプログラムを変換する必要があります。ユーザーのプログラムを変換する際、通常、CICS は、アドレス可能度を設定する標準の CICS プロログ (DFHEIENT) を提供し、レジスターを標準の CICS 作業記憶域に保管し、さらに標準の CICS エピログ (DFHEIRET) を提供します。

標準の CICS RETURN コマンド、たとえば、EXEC CICS RETURN を使用して、QMF に制御を戻します。

プログラムのアセンブル

アセンブル中は、QMF サンプル・ライブラリー QMF710.SDSQUSRE にある QMF 編集出口インターフェース制御ブロック DXEECSA、および CICS マクロ・ライブラリーが使用可能になっている必要があります。

プログラムのリンク・エディット

EXEC CICS インターフェース制御モジュール DFHEAI および DFHEAI0 をユーザーの編集プログラム DSQUXCTA に組み込むことによって、新しい QMF 編集出口モジュール DSQUECIC を作成します。これらのインターフェース制御モジュールは、CICS 製品で配布される CICS モジュール・ライブラリーにあります。EXEC CICS モジュール DFHEAI は、編集出口モジュールの最初のモジュールであり、かつ、入り口点は DSQUECIC であることが必要です。

モジュール DSQUECIC は、31 ビット・アドレッシング・モードで実行可能でなければなりません。

変換、アセンブル、およびリンク・エディットのステートメント例 (CICS)

次に示すのは、CICS の変換、アセンブルおよびリンク・エディットのジョブのステートメントの例です。

```
//SAMPASM JOB ...
/* Add a parameter PROGLIB to procedure DFHEITAL
/*      PROGLIB=&PROGLIB,
//TRNCOMLK EXEC PROC=DFHEITAL,PROGLIB='QMF710.SDSQLOAD'
//TRN.SYSIN DD *
    .
    Your program or modified copy of QMF sample DSQUXCTA
    .
/*
/* Provide access to QMF Edit Macro DXEECSA
//ASM.SYSLIB DD DSN=QMF710.SDSQSRE,DISP=SHR
//LKED.SYSIN DD *
    INCLUDE SYSLIB(DFHEAI)
    INCLUDE CICSLOAD(DFHEAI0)
    ORDER DFHEAI,DFHEAI0
    ENTRY DSQUECIC
    MODE AMODE(31) RMODE(ANY)
    NAME DSQUECIC(R)
/*
```

プログラム例

DSQUXCTA という名前の IBM 提供の編集出口プログラムの例が、QMF サンプル・ライブラリー QMF710.SDSQSAPE にあります。このプログラムにはコメントが数多く付いており、オンラインで見ると印刷することも可能です。また、ユーザーの要件に合わせて変更することもできます。このプログラムを使用する場合は、ユーザーのプログラム・ライブラリーにコピーして、名前を DSQUECIC に変更してください。

QMF 書式のためのユーザー独自の編集コードの作成

アセンブラー編集ルーチンと QMF との対話方法

QMF とユーザー編集インターフェース DSQUEDIT との間のインターフェース制御ブロックは DXEECS です。これはユーザーの編集コードを含み、ソース・データおよび編集結果のターゲット・ロケーションを識別し、さらにユーザー編集ルーチンが使用するスクラッチパッド域を提供します。制御ブロックは、ユーザー編集ルーチンへの呼び出しの間も存続しています。スクラッチパッド域は、出口ルーチンの最初の呼び出しの後には、QMF により変更されることはありません。

395ページの図104 は、アセンブラーの DXEECS 制御ブロックを示しています。

QMF 書式のためのユーザー独自の編集コードの作成

```

***** 00001000
*
* CONTROL BLOCK NAME: DXEECS (ASSEMBLER VERSION) * 00002000
*
* FUNCTION: * 00003000
*
* THIS IS THE INTERFACE CONTROL BLOCK BETWEEN QMF AND * 00004000
* THE USER EDITING INTERFACE, DSQUEDIT (TSO/CMS), OR * 00005000
* DSQUECIC (CICS). * 00006000
*
* IT CONTAINS THE USER'S EDIT CODE, IDENTIFIES THE SOURCE * 00007000
* DATA AND THE TARGET LOCATION FOR THE EDITED RESULT * 00008000
* AND PROVIDES A SCRATCHPAD AREA FOR THE USER EDIT * 00009000
* ROUTINE'S USE. * 00010000
*
* THIS CONTROL BLOCK IS PERSISTENT BETWEEN CALLS TO THE * 00011000
* USER EDIT ROUTINE. * 00012000
*
* THE SCRATCHPAD AREA WILL NOT BE MODIFIED BY QMF AFTER * 00013000
* THE INITIAL INVOCATION OF THE EXIT ROUTINE. * 00014000
*
* STATUS: VERSION 7 RELEASE 1 LEVEL 0 * 00015000
*
* INNER CONTROL BLOCKS: NONE * 00016000
*
* CHANGE ACTIVITY: * 00017000
*
* CHANGE DATE: * 00018000
*
***** 00019000
*
* DXEECS DSECT * 00020000
ECSNAME DS CL8 -- CONTROL BLOCK IDENTIFICATION * 00021000
SPACE * 00022000
ECSEDCTL DS XL40 -- EDIT CONTROL * 00023000
ORG ECSEDCTL * 00024000
ECSFREQ DS CL1 ----- FUNCTION REQUEST * 00025000
ECSFEDIT EQU C'E' ----- EDIT FUNCTION * 00026000
ECSFTERM EQU C'T' ----- TERMINATE FUNCTION * 00027000
* (TO FREE RESOURCES... QMF * 00028000
* WILL CALL THE USER EDIT * 00029000
* ROUTINE FOR THIS FUNCTION * 00030000
* ONLY IF THE USER EDIT ROUTINE * 00031000
* HAS PREVIOUSLY REQUESTED IT. * 00032000
* SEE ECSRQMF BELOW.) * 00033000
ECSPAD10 DS CL3 ----- RESERVED FIELD * 00034000
ECSECODE DS CL5 ----- EDIT CODE FROM FORM OBJECT * 00035000

```

図 104. アセンブラー DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (1/3)

QMF 書式のためのユーザー独自の編集コードの作成

ECSPAD20 DS	CL3	-----	RESERVED FIELD	00049000
ECSDECPT DS	CL1	-----	SYMBOL FOR DECIMAL POINT	00050000
*			(AS DEFINED BY DECIMAL OPTION IN	00051000
*			CURRENT PROFILE OBJECT	00052000
ECSTHSEP DS	CL1	-----	SYMBOL FOR THOUSANDS SEPARATOR	00053000
*			(AS DEFINED BY DECIMAL OPTION IN	00054000
*			CURRENT PROFILE OBJECT	00055000
ECSPAD30 DS	CL6	-----	RESERVED FIELD	00056000
ECSQMF DS	CL20	-----	AREA RESERVED FOR QMF'S USE	00057000
	SPACE			00058000
ECSINDTA DS	XL16	--	DESCRIPTION OF THE INPUT DATA	00059000
	ORG	ECSINDTA		00060000
ECSINTYP DS	F	-----	DATA TYPE OF THE INPUT AS IT	00061000
*			EXISTS IN THE DATA BASE.	00062000
ECSFLT EQU	480	-----	FLOATING POINT DATA TYPE CODE	00063000
ECSDEC EQU	484	-----	DECIMAL DATA TYPE CODE	00064000
ECSINT EQU	496	-----	INTEGER DATA TYPE CODE	00065000
ECSSINT EQU	500	-----	SMALL INTEGER DATA TYPE CODE	00066000
ECSVCHR EQU	448	-----	VARCHAR DATA TYPE CODE	00067000
ECSFCHR EQU	452	-----	(FIXED) CHARACTER DATA TYPE CODE	00068000
ECSLCHR EQU	456	-----	LONG VARCHAR DATA TYPE CODE	00069000
ECSVG EQU	464	-----	VARGRAPHIC DATA TYPE CODE	00070000
ECSFG EQU	468	-----	(FIXED) GRAPHIC DATA TYPE CODE	00071000
ECSLG EQU	472	-----	LONG VARGRAPHIC DATA TYPE CODE	00072000
ECSDATE EQU	384	-----	DATE DATA TYPE CODE	00073000
ECSTIME EQU	388	-----	TIME DATA TYPE CODE	00074000
ECSTS EQU	392	-----	TIMESTAMP DATA TYPE CODE	00075000
ECSFLTX EQU	940	-----	EXTENDED FLOATING PT CODE	00076000
*				00077000
ECSINLEN DS	F	-----	LENGTH OF INPUT DATA	00078000
ECSINPRC DS	H	-----	PRECISION OF INPUT DATA IF IT IS	00079000
*			DECIMAL DATA TYPE (U-TYPE EDIT CODE)	00080000
*			OR IF IT WAS ANY NUMERIC DATA TYPE	00081000
*			(V-TYPE EDIT CODE)...	00082000
*			ZERO OTHERWISE	00083000
ECSINSCL DS	H	-----	SCALE OF INPUT DATA IF IT IS	00084000
*			DECIMAL DATA TYPE (U-TYPE EDIT CODE)	00085000
*			OR IF IT WAS ANY NUMERIC DATA TYPE	00086000
*			(V-TYPE EDIT CODE)...	00087000
*			ZERO OTHERWISE	00088000
ECSINSGN DS	CL1	-----	SIGN OF CONVERTED NUMERIC DATA	00089000
*			(V-TYPE EDIT CODE ONLY)...	00090000
ECSPLUS EQU	C' '	-----	POSITIVE SIGN	00091000
ECSMINUS EQU	C' -'	-----	NEGATIVE SIGN	00092000
*				00093000
ECSINNULL DS	CL1	-----	NULL INPUT DATA INDICATOR	00094000
ECSNULL EQU	C'N'	-----	INPUT DATA IS NULL	00095000
*				00096000

図 104. アセンブラー DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (2/3)

QMF 書式のためのユーザー独自の編集コードの作成

ECSPAD40	DS	CL10	----- RESERVED FIELD	00097000
		SPACE		00098000
ECSRSDTA	DS	XL16	-- DESCRIPTION OF THE RESULT BUFFER	00099000
	ORG	ECSRSDTA		00100000
ECSRSLN	DS	F	----- LENGTH OF RESULT AREA	00101000
*			(EQUIVALENT TO COLUMN WIDTH IN THE	00102000
*			FORM OBJECT	00103000
ECSPAD50	DS	CL12	----- RESERVED FIELD	00104000
		SPACE		00105000
ECSUCTL	DS	XL16	-- USER CONTROL AREA	00106000
	ORG	ECSUCTL		00107000
ECSERRET	DS	F	----- EDIT ROUTINE ERROR RETURN CODE	00108000
ECSERR01	EQU	99101	----- UNRECOGNIZED EDIT CODE	00109000
ECSERR02	EQU	99102	----- IMPROPER INPUT DATA TYPE	00110000
ECSERR03	EQU	99103	----- INVALID INPUT DATA VALUE	00111000
ECSERR04	EQU	99104	----- INPUT DATA LENGTH IS TOO SHORT	00112000
ECSERR05	EQU	99105	----- RESULT BUFF LENGTH IS TOO SHORT	00113000
*				00114000
ECSRQMF	DS	CL1	----- REQUEST FOR QMF	00115000
ECSRTERM	EQU	C'T'	----- REQUEST INVOCATION FOR	00116000
*			TERMINATION FUNCTION	00117000
*				00118000
ECSPAD60	DS	CL11	----- RESERVED FIELD	00119000
		SPACE		00120000
ECSUSERS	DS	CL256	-- USER SCRATCH PAD AREA	00121000
		SPACE 2		00122000
ECSINPT	DSECT		-- EDIT ROUTINE INPUT DATA	00123000
ECSINPTC	DS	CL32767	----- CHARACTER STRING	00124000
	ORG	ECSINPTC		00125000
ECSINSIN	DS	H	----- SMALL INTEGER	00126000
	ORG	ECSINPTC		00127000
ECSININT	DS	F	----- INTEGER	00128000
	ORG	ECSINPTC		00129000
ECSINFLT	DS	D	----- FLOATING POINT	00130000
		SPACE 2		00131000
ECSRSLT	DSECT		-- EDIT ROUTINE RESULT BUFFER	00132000
ECSRSLTC	DS	CL32767	----- CHARACTER STRING	00133000

図 104. アセンブラー DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (3/3)

編集ルーチンを PL/I で作成

TSO または CICS 用の編集ルーチンを PL/I で作成することができます。

言語環境プログラム (LE) を使用せずに TSO、SRPI、APPC またはネイティブ OS/390 用に PL/I で編集ルーチンを作成

TSO、SRPI、APPC およびネイティブ OS/390 の PL/I 用の QMF 編集出口インターフェースは、以下の部分から構成されています。

- インターフェース制御ブロック。DXEECSP として QMF と共に出荷。
- 制御プログラム。DSQUXIP として QMF と共に出荷。

QMF 書式のためのユーザー独自の編集コードの作成

- 制御プログラム。DSQUPLI として QMF と共に出荷。
- ユーザーの編集出口プログラム。名前は DSQUXDT。

図105 は、TSO、SRPI、APPC、またはネイティブ OS/390 用の PL/I 編集出口ルーチンのプログラム構造を示しています。

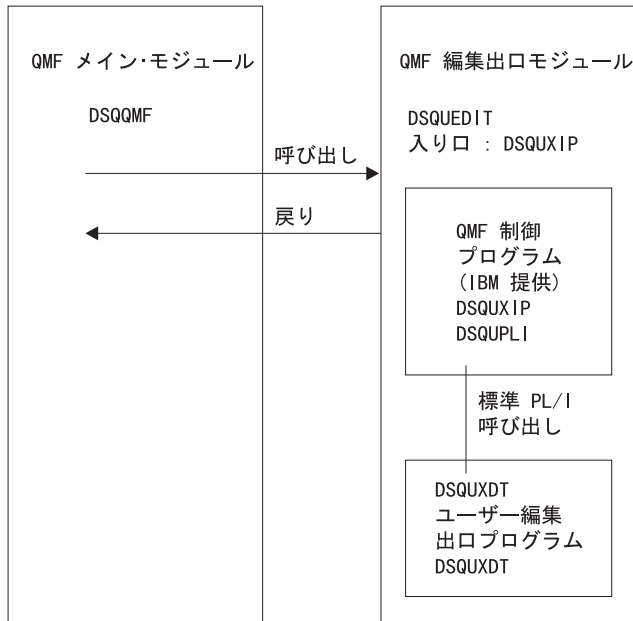


図105. LE を使用しない、TSO、SRPI、APPC、またはネイティブ OS/390 用の PL/I 編集出口ルーチンのプログラム構造

PL/I 編集ルーチンの、TSO、SRPI、APPC、またはネイティブ OS/390 との対話方法

ユーザー編集プログラムは、標準の PL/I CALL ステートメントを使用して、PL/I 外部プロシージャーとして呼び出されます。以下のパラメーターが、示された順序で提供されます。

1. DXEECS
2. 入力データ
3. 出力データ

次の例は、パラメーターを指定するプロシージャー・ステートメントの例です。

```
DSQUXDT:
  PROCEDURE(DXEECSF,ECSINPTF,ECSRSLTF) OPTIONS(REENTRANT);
```


PL/I データ構造は、QMF と共に DXEECSP として出荷され、ライブラリー QMF710.SDSQSAPE の中にあります。このデータ構造をユーザーのプログラムに組み込みます。

標準の RETURN ステートメントを使用して、QMF に制御を戻します。

DSQUXDT および DSQUPLI のコンパイル

コンパイル中は、QMF サンプル・ライブラリー QMF710.SDSQUSRE にある QMF 編集出口インターフェース制御ブロック DXEECSP がマクロ・ライブラリーで使用可能になっている必要があります。

両方のプログラムを、STAE または SPIE マクロなしでコンパイルします。そのためには、次のステートメントを PL/I プログラムに加えます。

```
DCL PLIXOPT CHAR(15) VAR INIT('NOSTAE,NOSPIE') STATIC EXTERNAL;
```

DSQUPLI を MAIN オプションを指定してコンパイルします。ユーザーの編集出口プログラム DSQUXDT は、MAIN を指定してはなりません。

プログラムのリンク・エディット

QMF モジュール・ライブラリー QMF710.SDSQLOAD にある IBM 提供の制御モジュール DSQUXIP および DSQUPLI を、ユーザー編集プログラム DSQUXDT に組み込むことにより、新規の QMF 編集出口モジュール DSQUEDIT を作成します。DSQUXIC を入り口点として指定する必要があります。

モジュール DSQUEDIT は、24 ビットもしくは 31 ビットのアドレッシング・モードのいずれでも実行可能です。QMF は 31 ビット・アドレッシング・モードで実行しますが、出口モジュール DSQUEDIT が 24 ビット・アドレッシング・モードであれば、自動的に 24 ビット・アドレッシング・モードに切り替わります。

31 ビット・アドレッシング・モードをお勧めします。

コンパイルおよびリンク・エディットのステートメント例 (TSO、SRPI、APPC、またはネイティブ OS/390)

以下に示すのは、TSO、SRPI、APPC、またはネイティブ OS/390 のジョブをアセンブルし、リンク・エディットするステートメントの例です。

```
//samPLI      JOB  
//STEP1      EXEC IEL1CL  
//* Provide Access to QMF Edit Macro DXEECSP  
//PLI.SYSLIB  DD  DSN=QMF710.SDSQUSRE,DISP=SHR  
//PLI.SYSIN   DD  *
```

QMF 書式のためのユーザー独自の編集コードの作成

```
                Your program or copy of QMF sample DSQUXDTP
/*
  /* Provide Access to QMF Interface Module
//LKED.QMFLOAD DD DSN=QMF710.SDSQLOAD,DISP=SHR
//LKED.SYSIN   DD *
                INCLUDE QMFLOAD(DSQUXIP)
                INCLUDE QMFLOAD(DSQUPLI)
                ENTRY DSQUXIP
                MODE  AMODE(31) RMODE(ANY)
                NAME  DSQUEDIT(R)
/*
```

プログラム例

DSQUXDTP という名前の IBM 提供の編集出口プログラムの例が、QMF サンプル・ライブラリー QMF710.SDSQSAPE にあります。このプログラムにはコメントが数多く付いており、オンラインで見ること印刷することも可能です。また、ユーザーの要件に合わせて変更することもできます。このプログラム例を使用する場合は、ユーザーのプログラム・ライブラリーにコピーし、名前を DSQUXDT に変更してください。

言語環境プログラム (LE) を使用して TSO、SRPI、APPC、またはネイティブ OS/390 での編集ルーチンを PL/I で作成

LE を使用した TSO、SRPI、APPC、またはネイティブ OS/390 での PL/I 用の QMF 編集出口インターフェースは、以下の部分から構成されています。

- インターフェース制御ブロック。DXEECSPP として QMF と共に出荷。
- 制御プログラム。DSQUXILE として QMF と共に出荷。
- 動的ロード LE 事前初期化サービス・プログラム。名前は CEEPIPI。
- ユーザーの編集出口プログラム。名前は DSQUXDT。

401ページの図106 は、TSO、SRPI、APPC、またはネイティブ OS/390 での PL/I 編集出口ルーチンのプログラム構造を示しています。

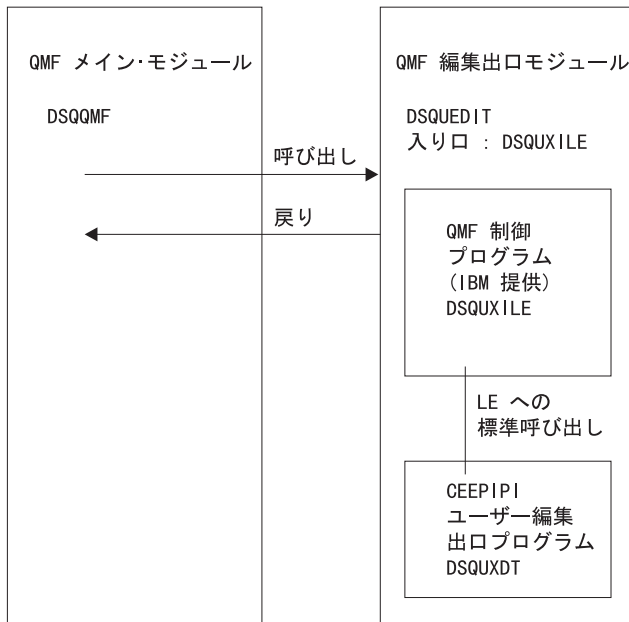


図 106. LE を使用した、TSO、SRPI、APPC、またはネイティブ OS/390 での PL/I 編集出口ルーチンのプログラム構造

PL/I 編集ルーチンが LE を使用した TSO、SRPI、APPC、またはネイティブ OS/390 と対話する方法

ユーザー編集プログラムは、LE サブルーチンとして呼び出されます。以下のパラメーターが、示された順序で提供されます。

1. DXEECS
2. 入力データ
3. 出力データ

次の例は、パラメーターを指定するプロシージャー・ステートメントの例です。

```
DSQUXDT:
PROCEDURE(DXEECSF,ECSINPTF,ECSRSLTF) OPTIONS(REENTRANT);
```

PL/I データ構造は、QMF と共に DXEECSF として出荷され、ライブラリー QMF710.SDSQSAPE の中にあります。このデータ構造をユーザーのプログラムに組み込みます。

標準の RETURN ステートメントを使用して、QMF に制御を戻します。

QMF 書式のためのユーザー独自の編集コードの作成

DSQUXDT のコンパイル

コンパイル中は、QMF サンプル・ライブラリー QMF710.SDSQUSRE にある QMF 編集出口インターフェース制御ブロック DXEECSF がマクロ・ライブラリーで使用可能になっている必要があります。

プログラムを、STAE または SPIE マクロなしでコンパイルします。そのためには、以下のステートメントを PL/I プログラムに加えます。

```
DCL PLIXOPT CHAR(15) VAR INIT('NOSTAE,NOSPIE') STATIC EXTERNAL;
```

DSQUPLI を MAIN オプションを指定してコンパイルします。ユーザーの編集出口プログラム DSQUXDT は、MAIN を指定してはなりません。

プログラムのリンク・エディット

QMF モジュール・ライブラリー QMF710.SDSQLOAD にある IBM 提供の制御モジュール DSQUXILE を、ユーザー編集プログラム DSQUXDT に組み込むことにより、新規の QMF 編集出口モジュール DSQUEDIT を作成します。モジュール DSQUXILE を入り口点として指定する必要があります。

モジュール DSQUEDIT は、24 ビットもしくは 31 ビットのアドレッシング・モードのいずれでも実行可能です。QMF は 31 ビット・アドレッシング・モードで実行しますが、出口モジュール DSQUEDIT が 24 ビット・アドレッシング・モードであれば、自動的に 24 ビット・アドレッシング・モードに切り替わります。

31 ビット・アドレッシング・モードをお勧めします。

コンパイルおよびリンク・エディットのステートメント例 (TSO、SRPI、APPC、またはネイティブ OS/390)

次に示すのは、ジョブを、TSO、SRPI、APPC、またはネイティブ OS/390 でアセンブルしリンク・エディットするステートメントの例です。

```
//samPLI      JOB
//STEP1       EXEC PLIXCL
//* Provide Access to QMF Edit Macro DXEECSF
//PLI.SYSLIB   DD DSN=QMF710.SDSQUSRE,DISP=SHR
//PLI.SYSIN    DD *
.
                Your program or copy of QMF sample DSQUXDTP
.
/*
//* Provide Access to QMF & LE Interface Module
//LKED.QMFLOAD DD DSN=QMF710.SDSQLOAD,DISP=SHR
//LKED.SYSLIB  DD DSN=SYS1.SCEELKED,DISP=SHR
//LKED.SYSIN   DD *
                INCLUDE QMFLOAD(DSQUXILE)
```

```
ENTRY DSQUXILE  
MODE AMODE(31) RMODE(ANY)  
NAME DSQUEDIT(R)  
  
/*
```

プログラム例

DSQUXDTP という名前の IBM 提供の編集出口プログラムの例が、QMF サンプル・ライブラリー QMF710.SDSQSAPE にあります。このプログラムにはコメントが数多く付いており、オンラインで見ると印刷することも可能です。また、ユーザーの要件に合わせて変更することもできます。このプログラム例を使用する場合は、ユーザーのプログラム・ライブラリーにコピーし、名前を DSQUXDT に変更してください。

編集ルーチンを CICS 用に PL/I で作成

CICS における PL/I 用の QMF 編集出口インターフェースは、以下の部分から構成されています。

- インターフェース制御ブロック。DXEECSP として QMF と共に出荷。
- CICS コマンド・インターフェース・モジュール。DFHPL1OI として CICS と共に出荷。
- ユーザーの編集出口プログラム。名前は DSQUECIC。

404ページの図107 は、CICS の PL/I 編集出口ルーチンのプログラム構造を示しています。

QMF 書式のためのユーザー独自の編集コードの作成

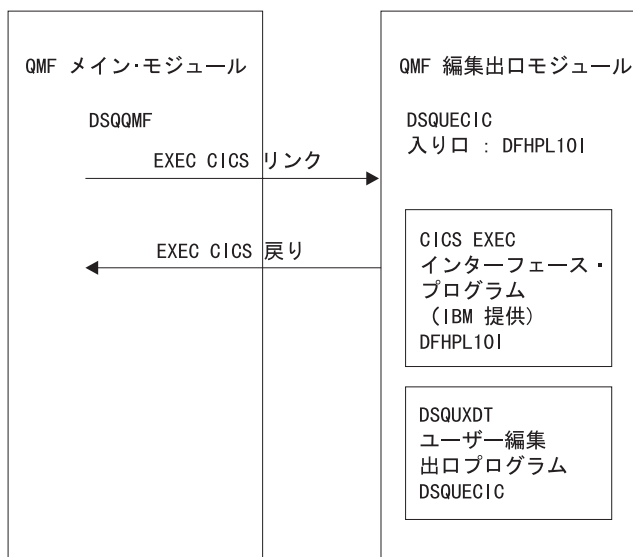
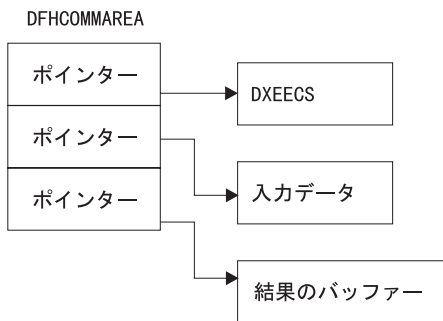


図 107. CICS の PL/I 編集出口ルーチンのプログラム構造

PL/I 編集ルーチンと CICS との対話方法

ユーザー編集プログラムは、標準の CICS LINK コマンド・インターフェースを使用して、呼び出されます。ユーザーのプログラムは、メインの QMF プログラムとは異なるプログラム・レベルで実行されています。ユーザーの編集プログラムは、PL/I 用の CICS 変換プログラムを使用して、変換する必要があります。

次の図に示すように、CICS 連絡域である DFHCOMMAREA を使用して、ユーザー編集ルーチンのプログラム・パラメーター、DXEECS、入力データ、および出力データにアドレスが渡されます。



QMF 書式のためのユーザー独自の編集コードの作成

変換後、CICS 変換プログラムは、CICS 環境ブロック DFHEIBLK を記述するプロシージャ・ステートメントを提供します。次の例のように、CICS コミュニケーション・ブロック DFHCOMMAREA を指すポインターを提供します。

```
DSQUECIC:
  PROCEDURE(DFHCOMP) OPTIONS(REENTRANT,MAIN);
```

QMF は、CICS 連絡域 DFHCOMMAREA で、ユーザー編集ルーチンの制御ブロックである DXEECS、入力データ、および出力データにアドレスを渡します。次のように、PL/I プログラムで DFHCOMMAREA のユーザー独自の記述を用意します。

```
/******  
/* CICS DFHCOMM AREA DESCRIPTION OF EDIT EXIT PARAMETERS */  
/******  
DECLARE  
  DFHCOMP PTR;  
DECLARE  
  1 DFHCOMM BASED(DFHCOMP),  
    02 DFHCOMM_ECSPTR PTR,  
    02 DFHCOMM_INPTR PTR,  
    02 DFHCOMM_OUTPTR PTR;
```

ユーザー編集ルーチンの制御ブロック DXEECS、入力データ域 ECSINPT、および結果のデータ域 ECSRSLT にアドレス可能度を与えるには、次の例のように、これらのデータ域のアドレスを DFHCOMMAREA にある値に設定します。

```
ECSPTR   = DFHCOMM_ECSPTR;    /* ADDRESS OF DXEECS:  
                               EDIT CODE SPECIFICATIONS */  
ECSINPT  = DFHCOMM_INPTR;    /* ADDRESS OF INPUT DATA */  
ECSRSLTP = DFHCOMM_OUTPTR;   /* ADDRESS OF RESULT AREA */
```

PL/I データ構造は、QMF と共に DXEECS として出荷され、ライブラリー QMF710.SDSQSAPE の中にあります。この構造をユーザーのプログラムに組み込みます。

次のように、標準の CICS RETURN コマンドを使用して、QMF に制御を戻します。

```
EXEC CICS RETURN;
```

プログラムの変換

PL/I 用の CICS 変換プログラムを使用して、ユーザーのプログラムを変換します。変換中に CICS は通常、CICS 環境制御ブロック EIB の入力パラメーターおよびデータ構造定義を提供します。

QMF 書式のためのユーザー独自の編集コードの作成

プログラムのコンパイル

コンパイル中は、QMF サンプル・ライブラリー QMF710.SDSQUSRE にある QMF 編集出口インターフェース制御ブロック DXEECSPL がマクロ・ライブラリーで使用可能になっている必要があります。

ユーザーのプログラムは、STAE または SPIE マクロなしでコンパイルする必要があります。これを行うには、ユーザーの PL/I プログラムに次のステートメントを追加します。

```
DCL PLIXOPT CHAR(15) VAR INIT('NOSTAE,NOSPIE') STATIC EXTERNAL;
```

PL/I コンパイラー・オプション SYSTEM(CICS) を指定します

プログラムのリンク・エディット

EXEC CICS インターフェース制御モジュール DFHPL1OI とユーザーの編集プログラムを組み込むことによって、新しい QMF 編集出口モジュール DSQUEECIC を作成します。このインターフェース制御モジュールは、CICS 製品で配布される CICS モジュール・ライブラリーにあります。リンク・エディットに必要な PL/I ライブラリーを必ず割り当ててください。

モジュール DSQUEECIC は、31 ビット・アドレッシング・モードで実行可能でなければなりません。

変換、コンパイル、およびリンク・エディットのステートメント例 (CICS)

次に示すのは、CICS の変換、コンパイルおよびリンク・エディットのジョブのステートメントの例です。

```
//SAMPLI JOB ...
/** Add a parameter PROGLIB to procedure DFHEITPL
/** PROGLIB=&PROGLIB,
//TRNCOMLK EXEC PROC=DFHEITPL,PROGLIB='QMF710.SDSQLOAD'
//TRN.SYSIN DD *
        .
        Your program or modified copy of QMF sample DSQUXCTP
        .
/**
/** Provide access to QMF Edit Macro DXEECSPL
//PLI.SYSLIB DD DSN=QMF710.SDSQUSRE,DISP=SHR
//LKED.SYSIN DD *
        REPLACE PLISTART
        INCLUDE CICSLOAD(DFHPL1OI)
        REPLACE PLISTART
        ORDER DFHPL1OI
        ENTRY DFHPL1OI
        MODE AMODE(31),RMODE(ANY)
        NAME DSQUEECIC(R)
/**
```


CICS プログラム定義

QMF がインストールされると、QMF 編集出口プログラムがアセンブラーのプログラム言語でインストールされます。PL/I 編集出口プログラムを使用するためには、CICS プログラム管理テーブル (PCT) マクロまたはオンライン・リソース定義 (RDO) を使用して、モジュール DSQUECIC のプログラム言語を PL/I に変更する必要があります。

プログラム例

DSQUXCTP という名前の IBM 提供の編集出口プログラムの例が、QMF サンプル・ライブラリー QMF710.SDSQSAPE にあります。このプログラムにはコメントが数多く付いており、オンラインで見ると印刷することも可能です。また、ユーザーの要件に合わせて変更することもできます。このプログラムを使用する場合は、ユーザーのプログラム・ライブラリーにコピーして、名前を DSQUECIC に変更してください。

PL/I 編集ルーチンと QMF との対話方法

QMF とユーザー編集インターフェース DSQUEDIT との間のインターフェース制御ブロックは DXE ECS です。これはユーザーの編集コードを含み、ソース・データおよび編集結果のターゲット・ロケーションを識別し、さらにユーザー編集ルーチンが使用するスクラッチパッド域を提供します。制御ブロックは、ユーザー編集ルーチンへの呼び出しの間も存続しています。スクラッチパッド域は、出口ルーチンの最初の呼び出しの後には、QMF により変更されることはありません。

408ページの図108 は、アセンブラーの DXE ECS 制御ブロックを示しています。

QMF 書式のためのユーザー独自の編集コードの作成

```

/*****/ 00001000
/* */ 00002000
/* CONTROL BLOCK NAME: DXEECS (PLI VERSION) */ 00003000
/* */ 00004000
/* FUNCTION: */ 00005000
/* */ 00006000
/* THIS IS THE INTERFACE CONTROL BLOCK BETWEEN QMF AND */ 00007000
/* THE USER EDITING ROUTINE INTERFACE, DSQUEDIT (TSO/CMS), */ 00008000
/* OR DSQUEVIC (CICS). */ 00009000
/* */ 00010000
/* IT CONTAINS THE USER'S EDIT CODE, IDENTIFIES THE SOURCE */ 00011000
/* DATA AND THE TARGET LOCATION FOR THE EDITED RESULT */ 00012000
/* AND PROVIDES A SCRATCHPAD AREA FOR THE USER EDIT */ 00013000
/* ROUTINE'S USE. */ 00014000
/* */ 00015000
/* THIS CONTROL BLOCK IS PERSISTENT BETWEEN CALLS TO THE */ 00016000
/* USER EDIT ROUTINE. */ 00017000
/* */ 00018000
/* THE SCRATCHPAD AREA WILL NOT BE MODIFIED BY QMF AFTER */ 00019000
/* THE INITIAL INVOCATION OF THE EXIT ROUTINE. */ 00020000
/* */ 00021000
/* */ 00022000
/* STATUS: VERSION 7 RELEASE 1 LEVEL 0 */ 00023000
/* */ 00024000
/* INNER CONTROL BLOCKS: NONE */ 00025000
/* */ 00026000
/* CHANGE ACTIVITY: */ 00027000
/* */ 00028000
/* CHANGE DATE: */ 00029000
/* */ 00030000
/*****/ 00031000
00032000
DECLARE 00033000
  1 DXEECS BASED(ECSPTR), /* EDIT ROUTINE INFORMATION */ 00034000
    3 ECSNAME CHARACTER(8), /* CONTROL BLOCK IDENTIFICATION */ 00035000
                                00036000
  3 ECSEDCTL, /* EDIT CONTROL */ 00037000
    5 ECSFREQ CHARACTER(1), /* FUNCTION REQUEST */ 00038000
                                (CODES ARE DEFINED BELOW) */ 00039000
    5 ECSPAD10 CHARACTER(3), /* RESERVED FIELD */ 00040000
    5 ECSECODE CHARACTER(5), /* EDIT CODE FROM FORM OBJECT */ 00041000
    5 ECSPAD20 CHARACTER(3), /* RESERVED FIELD */ 00042000
    5 ECSDECP CHARACTER(1), /* SYMBOL FOR DECIMAL POINT */ 00043000
                                (AS DEFINED BY DECIMAL OPTION */ 00044000
                                IN CURRENT PROFILE OBJECT) */ 00045000
  5 ECSTHSEP CHARACTER(1), /* SYMBOL FOR THOUSANDS SEPARATOR */ 00046000
                                (AS DEFINED BY DECIMAL OPTION */ 00047000
                                IN CURRENT PROFILE OBJECT) */ 00048000
  5 ECSPAD30 CHARACTER(6), /* RESERVED FIELD */ 00049000

```

図 108. PLI DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (1/4)

QMF 書式のためのユーザー独自の編集コードの作成

```

5 ECSQMF CHARACTER(20), /* AREA RESERVED FOR QMF'S USE */ 00050000
                                00051000
3 ECSINDTA, /* DESCRIPTION OF THE INPUT DATA*/ 00052000
5 ECSINTYP FIXED BINARY(31), /* DATA TYPE OF THE INPUT AS 00053000
                                IT EXISTS IN THE DATA BASE 00054000
                                (SEE CODES DEFINED BELOW) */ 00055000
5 ECSINLEN FIXED BINARY(31), /* LENGTH OF INPUT DATA */ 00056000
5 ECSINPRC FIXED BINARY(15), /* PRECISION OF INPUT DATA IF 00057000
                                IS IT DECIMAL DATA TYPE 00058000
                                (U-TYPE EDIT CODE) OR 00059000
                                IF IT WAS ANY NUMERIC 00060000
                                DATA TYPE (V-TYPE EDIT 00061000
                                CODE)... 00062000
                                ZERO OTHERWISE */ 00063000
5 ECSINSCL FIXED BINARY(15), /* SCALE OF INPUT DATA IF 00064000
                                IS IT DECIMAL DATA TYPE 00065000
                                (U-TYPE EDIT CODE) OR 00066000
                                IF IT WAS ANY NUMERIC 00067000
                                DATA TYPE (V-TYPE EDIT 00068000
                                CODE)... 00069000
                                ZERO OTHERWISE */ 00070000
5 ECSINSGN CHARACTER(1), /* SIGN (V-TYPE EDIT ONLY) 00071000
                                SEE VALUES DEFINED 00072000
                                BELOW */ 00073000
5 ECSINNULL CHARACTER(1), /* NULL INPUT DATA INDICATOR 00074000
                                SEE VALUE DEFINED 00075000
                                BELOW */ 00076000
5 ECSPAD40 CHARACTER(10), /* RESERVED FIELD */ 00077000
3 ECSRSDTA, /* DESCRIPTION OF THE RESULT 00078000
                                BUFFER */ 00079000
5 ECSRSLEN FIXED BINARY(31), /* LENGTH (EQUIVALENT TO 00080000
                                COLUMN WIDTH IN THE 00081000
                                FORM OBJECT) */ 00082000
5 ECSPAD50 CHARACTER(12), /* RESERVED FIELD */ 00083000
                                00084000
3 ECSUCTL, /* USER CONTROL AREA */ 00085000
5 ECSERRET FIXED BINARY(31), /* EDIT ROUTINE ERROR RETURN CODE 00086000
                                (SEE CODES DEFINED BELOW) */ 00087000
5 ECSRQMF CHARACTER(1), /* REQUEST FOR QMF 00088000
                                (SEE CODE(S) DEFINED BELOW */ 00089000
5 ECSPAD60 CHARACTER(11), /* RESERVED FIELD */ 00090000
                                00091000
3 ECSUSERS CHARACTER(256); /* USER SCRATCH PAD AREA */ 00092000

```

図 108. PL/I DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (2/4)

QMF 書式のためのユーザー独自の編集コードの作成

```

                                00093000
DECLARE                               /* INPUT DATA PARAMETER... */ 00094000
  ECSINPT CHARACTER(32767)           /* CHARACTER INPUT DATA */ 00095000
    BASED(ECSINPT),                 00096000
  ECSINSIN FIXED BINARY(15)         /* SMALL INTEGER INPUT DATA */ 00097000
    BASED(ECSINPT),                 00098000
  ECSININT FIXED BINARY(31)         /* INTEGER INPUT DATA */ 00099000
    BASED(ECSINPT),                 00100000
  ECSINFLT FLOAT BINARY(53)         /* FLOATING POINT INPUT DATA */ 00101000
    BASED(ECSINPT);                 00102000
                                    00103000
DECLARE                               /* RESULT BUFFER PARAMETER... */ 00104000
  ECSRSLT CHARACTER(32767)           /* EDIT ROUTINE RESULT BUFFER */ 00105000
    BASED(ECSRSLTP);                 00106000
                                    00107000
DECLARE                               00108000
  (ECSPTR,                           /* MUST CONTAIN DXEECS ADDRESS */ 00109000
   ECSINPT,                           /* MUST CONTAIN ECSINPT ADDRESS */ 00110000
   ECSRSLTP                             /* MUST CONTAIN ECSRSLT ADDRESS */ 00111000
  ) POINTER;                           00112000
                                    00113000
                                    00114000
DECLARE (                               /* DATA TYPE CONSTANTS: */ 00115000
  (SEE ECSINTYP ABOVE)                /* */ 00116000
  ECSINT INITIAL(496),                 /* INTEGER */ 00117000
  ECSSINT INITIAL(500),                 /* SMALL INTEGER */ 00118000
  ECSFLT INITIAL(480),                 /* FLOATING POINT */ 00119000
  ECSVCHR INITIAL(448),                 /* VARYING CHARACTER */ 00120000
  ECSFCHR INITIAL(452),                 /* FIXED CHARACTER */ 00121000
  ECSLCHR INITIAL(456),                 /* VERY LONG CHARACTER */ 00122000
  ECSVG INITIAL(464),                 /* VARYING GRAPHIC */ 00123000
  ECSFG INITIAL(468),                 /* FIXED GRAPHIC */ 00124000
  ECSLG INITIAL(472),                 /* VERY LONG GRAPHIC */ 00125000
  ECSDEC INITIAL(484),                 /* DECIMAL */ 00126000
  ECSDATE INITIAL(384),                 /* DATE */ 00127000
  ECSTIME INITIAL(388),                 /* TIME */ 00128000
  ECSTS INITIAL(392),                 /* TIMESTAMP */ 00129000
  ECSFLTIX INITIAL(940)                 /* EXTENDED FLOATING POINT */ 00130000
  ) FIXED BINARY(31) STATIC;           00131000
                                    00132000
                                    00133000
DECLARE (                               /* FUNCTION REQUEST CONSTANTS */ 00134000

```

図 108. PLI DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (3/4)

QMF 書式のためのユーザー独自の編集コードの作成

```

                                (SEE ECSFREQ ABOVE)          */ 00135000
ECSFEDIT INITIAL('E'),          /* EDIT                      */ 00136000
ECSFTERM INITIAL('T')          /* TERMINATE                  */ 00137000
                                (TO FREE RESOURCES...
                                QMF WILL CALL THE USER
                                EDIT ROUTINE FOR THIS
                                FUNCTION ONLY IF THE
                                USER EDIT ROUTINE HAS
                                PREVIOUSLY REQUESTED
                                IT.)                          */ 00144000
                                ) CHARACTER(1) STATIC;
                                                                00145000
                                                                00146000
                                                                00147000
                                                                00148000
DECLARE (                      /* PLUS/MINUS SIGN CONSTANTS */ 00149000
                                (SEE ECSINSGN ABOVE)          */ 00149000
                                ECSPLUS INITIAL(' '),         /* INPUT DATA IS POSITIVE   */ 00150000
                                ECSMINUS INITIAL('-'),         /* INPUT DATA IS NEGATIVE   */ 00151000
                                ) CHARACTER(1) STATIC;
                                                                00152000
                                                                00153000
                                                                00154000
                                                                00155000
DECLARE (                      /* NULL INDICATION CONSTANT  */ 00155000
                                (SEE ECSINNUL ABOVE)          */ 00156000
                                ECSNULL INITIAL('N')          /* INPUT DATA IS NULL       */ 00157000
                                ) CHARACTER(1) STATIC;
                                                                00158000
                                                                00159000
                                                                00160000
                                                                00161000
DECLARE (                      /* REQUEST-FOR-QMF CONSTANTS */ 00161000
                                (SEE ECSRQMF ABOVE)            */ 00162000
                                ECSRTERM INITIAL('T')         /* REQUEST QMF TO INVOKE     */ 00163000
                                                                00164000
                                                                00165000
                                                                00166000
                                                                00167000
                                                                00168000
                                                                00169000
DECLARE (                      /* QMF-DEFINED ERROR RETURN  */ 00169000
                                CONSTANTS                      */ 00170000
                                (SEE ECSERRET ABOVE)          */ 00171000
                                ECSERR01 INITIAL(99101),      /* UNRECOGNIZED EDIT CODE   */ 00172000
                                ECSERR02 INITIAL(99102),      /* IMPROPER INPUT DATA TYPE */ 00173000
                                                                00174000
                                                                00175000
                                                                00176000
                                                                00177000
                                                                00178000
                                                                00179000
                                                                00180000
                                                                00181000
                                ECSERR03 INITIAL(99103),      /* REQUESTED EDIT EDIT CODE */ 00174000
                                                                00175000
                                                                00176000
                                                                00177000
                                                                00178000
                                                                00179000
                                                                00180000
                                                                00181000
                                ECSERR04 INITIAL(99104),      /* INVALID INPUT DATA VALUE */ 00175000
                                                                00176000
                                                                00177000
                                                                00178000
                                                                00179000
                                                                00180000
                                                                00181000
                                ECSERR05 INITIAL(99105),      /* RECEIVED                  */ 00176000
                                                                00177000
                                                                00178000
                                                                00179000
                                                                00180000
                                                                00181000
                                ECSERR05 INITIAL(99105),      /* LENGTH OF INPUT DATA IS  */ 00177000
                                                                00178000
                                                                00179000
                                                                00180000
                                                                00181000
                                ECSERR05 INITIAL(99105),      /* SHORT                     */ 00178000
                                                                00179000
                                                                00180000
                                                                00181000
                                ECSERR05 INITIAL(99105),      /* LENGTH OF RESULT BUFFER  */ 00179000
                                                                00180000
                                                                00181000
                                ECSERR05 INITIAL(99105),      /* TOO SHORT                 */ 00180000
                                                                00181000
                                ) FIXED BINARY(31) STATIC;
                                                                00181000

```

図 108. PL1 DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (4/4)

編集ルーチンを COBOL で作成

TSO、SRPI、APPC、ネイティブ OS/390、または CICS 用の編集ルーチンを COBOL で作成することができます。

本節では、特に断りのない限り、COBOL とは VS COBOL II、COBOL/370、および COBOL (OS/390 用) を指します。

言語環境プログラム (LE)[®] を使用せずに TSO、SRPI、APPC、またはネイティブ OS/390 で編集ルーチンを COBOL で作成

TSO、SRPI、APPC およびネイティブ OS/390 での COBOL の QMF 編集出口インターフェースは、以下の部分から構成されています。

- インターフェース制御ブロック。DXEECSO として QMF と共に出荷。
- 制御プログラム。DSQUXIC として QMF と共に出荷。
- ユーザーの編集出口プログラム。名前は DSQUXDT。

図109 は、TSO、SRPI、APPC、またはネイティブ OS/390 での COBOL 編集出口ルーチンのプログラム構造を示しています。

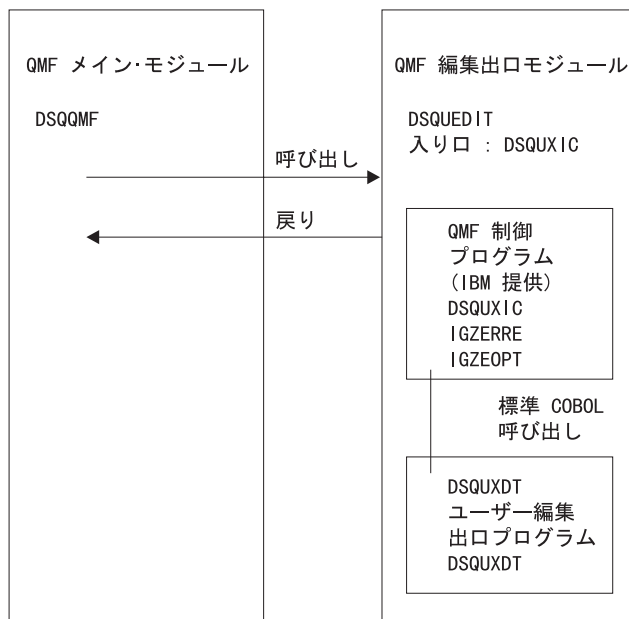


図109. TSO、SRPI、APPC、またはネイティブ OS/390 での COBOL 編集出口ルーチンのプログラム構造

COBOL 編集ルーチンの、TSO、SRPI、APPC、およびネイティブ OS/390 との対話方法

ユーザー編集プログラムは、標準の COBOL CALL ステートメントを使用して、COBOL サブプログラムとして呼び出されます。以下のパラメーターが、示された順序で提供されます。

1. DXEECS
2. 入力データ
3. 出力データ

次の例は、パラメーターを指定するプロシージャ・ステートメントの例です。

```
PROCEDURE DIVISION  
    USING DXEECS, ECSINPT, ECSRSLT.
```

標準の サブプログラム GOBACK ステートメントを使用して、QMF に制御を戻します。

DSQUXDT のコンパイル

DSQUXDT (ユーザーが作成した編集出口プログラム) をコンパイルします。コンパイル中は、QMF サンプル・ライブラリー QMF710.SDSQUSRE にある QMF 編集出口インターフェース制御ブロック DXEECS がマクロ・ライブラリーで使用可能になっている必要があります。

COBOL コンパイラー・オプションを次のように選択します。

COBOL II

コンパイラー・オプション RENT、RES、NODYNAM、OBJECT、および LIB を指定します。

COBOL/370 または IBM COBOL (OS/390 および VM 用)

コンパイラー・オプション OBJECT、LIB、RENT、および NODYNAM を指定します。

QMF は、引用符をリテラル区切り文字として、ユーザー編集ルーチン制御ブロック DXEECS を配布します。IBM より配布される DXEECS 制御ブロックを使用する場合は、QUOTE コンパイラー・オプションが必要になります。

DSQUXDT のコンパイル後、結果のロード・モジュールを QMF710.SDSQLOAD ライブラリーに入れます。

QMF 書式のためのユーザー独自の編集コードの作成

言語環境プログラム・ランタイム・ライブラリーの使用

QMF ユーザー編集出口プログラムで言語環境プログラムのランタイム・ライブラリーを使用する場合、以下の点を考慮してください。

- QMF は新規のコンパイルを必要としません。
- LE ランタイム・ライブラリーと共に使用する QMF ユーザー編集出口プログラムには、LINK EDIT が必要です。
- QMF アセンブラー・ドライバー DSQUXIC が IGZERRE を呼び出します。詳細については、IBM COBOL の資料を参照してください。

ランタイム・オプション・モジュールのアセンブル

ランタイム・オプション・マクロ IGZOPT をアセンブルする際、COBOL ランタイム・オプション STAE=NO を指定する必要があります。(言語環境プログラムのオプション・モジュールの場合は、STAE=NO に代えて TRAP=OFF を使用します。) 結果として得られるオブジェクト・モジュール IGZEOPT を QMF 編集出口モジュール DSQUEDIT に組み込みます。

プログラムのリンク・エディット

QMF モジュール・ライブラリー QMF710.SDSQLOAD にある IBM 提供の制御モジュール DSQUXIC を、ユーザー編集プログラム DSQUXDT に組み込むことにより、新しい QMF 編集出口モジュール DSQUEDIT を作成します。DSQUXIC を入り口点として指定する必要があります。

モジュール DSQUEDIT は、24 ビットもしくは 31 ビットのアドレッシング・モードのいずれでも実行可能です。QMF は 31 ビット・アドレッシング・モードで実行しますが、出口モジュール DSQUEDIT が 24 ビット・アドレッシング・モードであれば、自動的に 24 ビット・アドレッシング・モードに切り替わります。

注: 31 ビットのアドレッシング・モードを推奨します。

コンパイルおよびリンク・エディットのステートメント例 (TSO、SRPI、APPC、またはネイティブ OS/390)

以下に示すのは、TSO、SRPI、APPC、またはネイティブ OS/390 でのジョブをコンパイルおよびリンク・エディットするステートメントの例です。

COBOL II 用:

```
//samCOBOL JOB
//* Assemble run time option macro
//STEP1 EXEC PGM=IEV90,PARM='DECK,NOLOAD'
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT2 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
```


QMF 書式のためのユーザー独自の編集コードの作成

```
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYS PUNCH DD DSN=&&TEMPOBJ(IGZEOPT),DISP=(,PASS),UNIT=SYSDA,
// SPACE=(TRK,(1,1,1)),DCB=(BLKSIZE=3120,LRECL=80,DSORG=PO)
/* Provide Access to Cobol run time option macro
//SYSLIB DD DSN=SYS1.MACLIB,DISP=SHR
//SYSIN DD *
        IGZOPT SYSTYPE=OS,STAE=NO
        END
/*
//STEP2      EXEC PROC=COB2UCL
/* Provide Access to QMF Edit Macro DXEECS
//COB2.SYSLIB DD DSN=QMF710.SDSQUSRE,DISP=SHR
//COB2.SYSIN  DD *
        .
        Your program or copy of QMF sample DSQUXDT
        .
/*
/* Provide Access to QMF Interface Module
//LKED.QMFLOAD DD DSN=QMF710.SDSQLOAD,DISP=SHR
/* Make sure COBOL library is concatenated after &&TEMPOBJ
//LKED.SYSLIB  DD DSN=&&TEMPOBJ,DISP=(OLD,PASS)
                DD DSN=COB2LIB,DISP=(OLD,PASS)
//LKED.SYSIN   DD *
                INCLUDE QMFLOAD(DSQUXIC)
                INCLUDE SYSLIB(IGZEOPT)
                INCLUDE SYSLIB(IGZERRE)
                ENTRY DSQUXIC
                MODE  AMODE(31) RMODE(ANY)
                NAME  DSQUEDIT(R)
/*
```

COBOL/370 または IBM COBOL (OS/390 および VM 用) 用:

```
//samCOBOL JOB
/* Assemble run time option macro
//STEP1 EXEC PGM=IEV90,PARM='DECK,NOLOAD'
//SYS PPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT2 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYS PUNCH DD DSN=&&TEMPOBJ(IGZEOPT),DISP=(,PASS),UNIT=SYSDA,
// SPACE=(TRK,(1,1,1)),DCB=(BLKSIZE=3120,LRECL=80,DSORG=PO)
/* Provide Access to Cobol run time option macro
//SYSLIB DD DSN=SYS1.MACLIB,DISP=SHR
//SYSIN DD *
        IGZOPT SYSTYPE=OS,STAE=NO
        END
/*
//STEP2      EXEC PROC=IGYWCL
/* Provide Access to QMF Edit Macro DXEECS
//COBOL.SYSLIB DD DSN=QMF710.SDSQUSRE,DISP=SHR
//COBOL.SYSIN  DD *
        .
        Your program or copy of QMF sample DSQUXDT
        .
```

QMF 書式のためのユーザー独自の編集コードの作成

```
/*
/** Provide Access to QMF Interface Module
//LKED.QMFLOAD DD DSN=QMF710.SDSQLOAD,DISP=SHR
//LKED.SYSLIB DD ...
DD DSN=##TEMPOBJ,DISP=(OLD,PASS)
//LKED.SYSIN DD *
INCLUDE QMFLOAD(DSQUXIC)
INCLUDE SYSLIB(IGZEOPT)
INCLUDE SYSLIB(IGZERRE)
ENTRY DSQUXIC
MODE AMODE(31) RMODE(ANY)
NAME DSQUEDIT(R)
/*
```

プログラム例

DSQUXDTC という名前の IBM 提供の編集出口プログラムの例が、QMF サンプル・ライブラリー QMF710.SDSQSAPE にあります。このプログラムにはコメントが数多く付いており、オンラインで見ると印刷することも可能です。また、ユーザーの要件に合わせて変更することもできます。このプログラムを使用する場合は、ユーザーのプログラム・ライブラリーにコピーし、名前を DSQUXDTC に変更してください。

言語環境プログラム (LE) を使用して TSO、SRPI、APPC、またはネイティブ OS/390 で編集ルーチンを COBOL で作成

TSO、SRPI、APPC およびネイティブ OS/390 での COBOL の QMF 編集出口インターフェースは、以下の部分から構成されています。

- インターフェース制御ブロック。DXEECSK として QMF と共に出荷。
- 制御プログラム。DSQUXILE として QMF と共に出荷。
- ユーザーの編集出口プログラム。名前は DSQUXDTC。
- LE 事前初期化サービス・プログラム。名前は CEEPIPI。

417ページの図110 は、TSO またはネイティブ OS/390 バッチ の COBOL 編集出口ルーチンのプログラム構造を示しています。

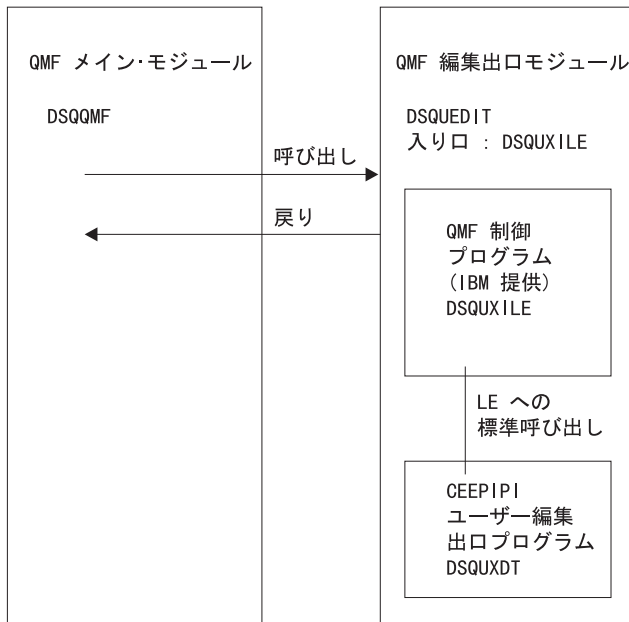


図 110. LE を使用した TSO、SRPI、APPC、またはネイティブ OS/390 での COBOL 編集出口ルーチンのプログラム構造

LE を使用した COBOL 編集ルーチンの TSO、SRPI、APPC、およびネイティブ OS/390 との対話方法

ユーザー編集プログラムは、LE サブルーチンとして呼び出されます。以下のパラメーターが、示された順序で提供されます。

1. DXEECS
2. 入力データ
3. 出力データ

次の例は、パラメーターを指定するプロシージャ・ステートメントの例です。

```
PROCEDURE DIVISION
  USING DXEECS, ECSINPT, ECSRSLT.
```

COBOL データ構造は、QMF と共に DXEECS として出荷され、ライブラリー QMF710.SDSQSAPE の中にあります。このデータ構造をユーザーのプログラムに組み込みます。

標準サブプログラム GOBACK ステートメントを使用して制御を QMF に返します。

QMF 書式のためのユーザー独自の編集コードの作成

DSQUXDT のコンパイル

コンパイル中は、QMF サンプル・ライブラリー QMF710.SDSQSUSRE にある QMF 編集出口インターフェース制御ブロック DXEECSK がマクロ・ライブラリーで使用可能になっている必要があります。

次のコンパイル・オプションを使用してプログラムをコンパイルします。

```
OBJECT, LIB, RENT, RES, and NODYNAM.
```

QMF は、引用符をリテラル区切り文字として、ユーザー編集ルーチン制御ブロック DXEECSK を配布します。IBM より配布される DXEECSK 制御ブロックを使用する場合は、QUOTE コンパイラー・オプションが必要になります。

プログラムのリンク・エディット

QMF モジュール・ライブラリー QMF710.SDSQLOAD にある IBM 提供の制御 QMF モジュール DSQUXILE を、ユーザー編集プログラム DSQUXDT に組み込むことにより、新規の QMF 編集出口モジュール DSQUEDIT を作成します。

モジュール DSQUXILE を入り口点として指定する必要があります。

モジュール DSQUEDIT は、24 ビットもしくは 31 ビットのアドレッシング・モードのいずれでも実行可能です。QMF は 31 ビット・アドレッシング・モードで実行しますが、出口モジュール DSQUEDIT が 24 ビット・アドレッシング・モードであれば、自動的に 24 ビット・アドレッシング・モードに切り替わります。

注: 31 ビットのアドレッシング・モードを推奨します。

コンパイルおよびリンク・エディットのステートメント例 (TSO、SRPI、APPC、またはネイティブ MVS)

次に示すのは、TSO、SRPI、APPC、またはネイティブ MVS のジョブをコンパイルしリンク・エディットするステートメントの例です。

```
//samCOBOL JOB
//STEP1 EXEC PROC=IGYWCL
//* Provide Access to QMF Edit Macro DXEECSK
//COBOL.SYSLIB DD DSN=QMF710.SDSQSUSRE,DISP=SHR
//COBOL.SYSIN DD *
```

QMF サンプル DSQUXDTC のユーザー・プログラムまたはコピーは、次のとおりです。

```
/*
//* Provide Access to QMF Interface Module
//LKED.QMFLOAD DD DSN=QMF710.SDSQLOAD,DISP=SHR
```

```
//LKED.SYSLIB DD ...  
// DD DSN=&&TEMPOBJ,DISP=(OLD,PASS)  
// DD DSN=SYS1.SCEELKED,DISP=SHR  
//LKED.SYSIN DD *  
        INCLUDE QMFLOAD(DSQUXILE)  
        ENTRY DSQUXILE  
        MODE AMODE(31) RMODE(ANY)  
        NAME DSQUEDIT(R)  
  
/*
```

プログラム例

DSQUXDTC という名前の IBM 提供の編集出口プログラムの例が、QMF サンプル・ライブラリー QMF710.SDSQSAPE にあります。このプログラムにはコメントが数多く付いており、オンラインで見ると印刷することも可能です。また、ユーザーの要件に合わせて変更することもできます。このプログラム例を使用する場合は、ユーザーのプログラム・ライブラリーにコピーし、名前を DSQUXDT に変更してください。

編集ルーチンを CICS 用に COBOL で作成

CICS における COBOL 用の編集出口インターフェースは、以下の部分から構成されています。

- インターフェース制御ブロック。DXEEECSC として QMF と共に出荷。
- CICS コマンド・インターフェース・モジュール。DFHECI として CICS と共に出荷。
- ユーザーの編集出口プログラム。名前は DSQUEECIC。

420ページの図111 は、CICS の COBOL 編集出口ルーチンのプログラム構造を示しています。

QMF 書式のためのユーザー独自の編集コードの作成

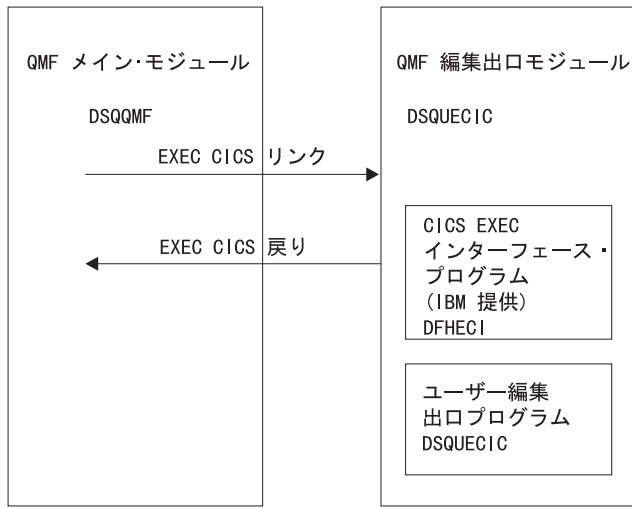
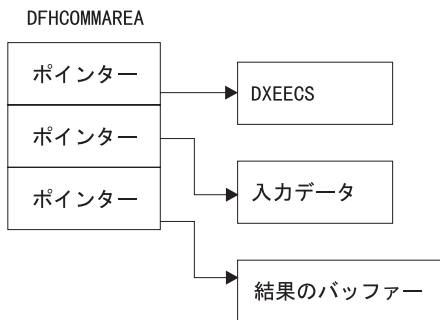


図 111. CICS の COBOL 編集出口ルーチンのプログラム構造

COBOL 編集ルーチンと CICS との対話方法

ユーザー編集プログラムは、標準の CICS LINK コマンド・インターフェースを使用して、呼び出されます。ユーザーのプログラムは、メインの QMF プログラムとは異なるプログラム・レベルで実行されています。ユーザーの編集プログラムは、COBOL 用の CICS 変換プログラムを使用して、変換する必要があります。次の図に示すように、CICS 連絡域である DFHCOMMAREA を使用して、ユーザー編集ルーチンのプログラム・パラメーター、DXEECS、入力データ、および出力データにアドレスが渡されます。



変換後、CICS 変換プログラムは、次の例のように、CICS 環境ブロック DFHEIBLK および CICS コミュニケーション・ブロック DFHCOMMAREA を記述するプロシージャ・ステートメントを提供します。

```
PROCEDURE DIVISION USING DFHEIBLK DFHCOMMAREA.
```

QMF 書式のためのユーザー独自の編集コードの作成

QMF は、CICS 連絡域 DFHCOMMAREA で、ユーザー編集ルーチンの制御ブロックである DXEECS、入力データ、および出力データにアドレスを渡します。次のように、COBOL プログラム・リンケージ・セクションで DFHCOMMAREA のユーザー独自の記述を提供してください。用意します。

```
LINKAGE SECTION.
```

```
01 DFHCOMMAREA.  
02 ECSADR POINTER.  
02 ECSINADR POINTER.  
02 ECSRLADR POINTER.
```

ユーザー編集ルーチンの制御ブロック DXEECS、入力データ域 ECSINPT、および結果のデータ域 ECSRSLT にアドレス可能性を与えるには、次の例のように、これらのデータ域のアドレスを DFHCOMMAREA にある値に設定します。

```
SETUP SECTION.
```

```
SET ADDRESS OF DXEECS TO ECSADR.  
SET ADDRESS OF ECSINPT TO ECSINADR.  
SET ADDRESS OF ECSRSLT TO ECSRLADR.
```

COBOL サンプル集は、QMF と共に DXEECS として出荷され、ライブラリー QMF710.SDSQSAPE の中にあります。このサンプル集をユーザーのプログラムに組み込みます。

次のように、標準の CICS RETURN コマンドを使用して、QMF に制御を戻します。

```
EXEC CICS  
RETURN  
END-EXEC.
```

COBOL プログラムの変換

ユーザーのプログラムを、COBOL 用の CICS 変換プログラムを使用して変換します。プログラムを変換する際、通常、CICS は標準プロシージャおよびリンケージ・セクションを使用します。前記のリンケージ・セクションの例で指定した構造を与えることにより、標準の CICS 連絡域 DFHCOMMAREA を置き換えます。

コンパイル

コンパイル中は、QMF サンプル・ライブラリー QMF710.SDSQUSRE にある QMF 編集出口インターフェース制御ブロック DXEECS がマクロ・ライブラリーで使用可能になっている必要があります。

QMF 書式のためのユーザー独自の編集コードの作成

COBOL コンパイラー・オプション RENT、RES、および NODYNAM、さらにランタイム・オプション NOSTAE および NORTEREUS を指定します。

QMF は、引用符をリテラル区切り文字として、ユーザー編集ルーチン制御ブロック DXEECSK を配布します。IBM より配布される DXEECSK 制御ブロックを使用する場合は、QUOTE コンパイラー・オプションが必要になります。

リンク・エディット

EXEC CICS インターフェース制御モジュール DFHECI をユーザーの編集出口プログラム DSQUXCTC に組み込むことによって、新規の QMF 編集出口モジュール DSQUECIC を作成します。このインターフェース制御モジュールは、CICS 製品で配布される CICS モジュール・ライブラリーにあります。DFHECI は、編集出口モジュールの最初のモジュールであり、かつ、入り口点は DSQUECIC であることが必要です。リンク・エディットに必要な COBOL ライブラリーを必ず割り当ててください。

モジュール DSQUECIC は、31 ビット・アドレッシング・モードで実行可能でなければなりません。

変換、コンパイル、およびリンク・エディットのステートメント例 (CICS)

次に示すのは、CICS の変換、コンパイルおよびリンク・エディットのジョブのステートメントの例です。

```
//SAMCOBOL JOB ...
/** Add a parameter PROGLIB to procedure DFHEITVL
/**     PROGLIB=&PROGLIB,
//TRNCOMLK EXEC PROC=DFHEITVL,PROGLIB='QMF710.SDSQLOAD',
//     PARM.TRN='QUOTE',
//     PARM.COB='RENT,RES,NODYNAM,OBJECT,LIB,LIST,MAP,QUOTE'
//TRN.SYSIN DD *
        .
        Your program or modified copy of QMF sample DSQUXCTC
        .
/**
/** Provide access to QMF Edit Macro DXEECSK
//COB.SYSLIB DD DSN=QMF710.SDSQSRE,DISP=SHR
//LKED.SYSIN DD *
        INCLUDE SYSLIB(DFHECI)
        ORDER DFHECI
        ENTRY DSQUECIC
        MODE AMODE(31) RMODE(ANY)
        NAME     DSQUECIC(R)
/**
```

CICS プログラム定義

QMF がインストールされると、QMF 編集出口プログラムがアSEMBラーのプログラム言語でインストールされます。COBOL 編集出口プログラムを使用す

るためには、CICS プログラム管理テーブル (PCT) マクロまたはオンライン・リソース定義 (RDO) を使用して、モジュール DSQUECIC のプログラム言語を COBOL に変更する必要があります。

リテラル区切り文字：引用符またはアポストロフィ

COBOL プログラムでは、リテラルを区切るには、引用符 (") またはアポストロフィ (') のいずれかを使用する必要があります。選択した区切り文字を、QUOTE または APOST と指定することにより、CICS 変換処理および COBOL コンパイラーに指示します。COBOL コンパイラーに有効な APOST または QUOTE オプションが、CICS 変換プログラムのものと一致していることを確認してください。

QMF で配布されている編集制御ブロック DXEECS およびサンプル COBOL プログラム DSQUXCTC では、リテラルを区切るために引用符 (") を使用しています。インストール先システムまたはプログラムで、代わりにアポストロフィ (') を使用している場合は、DXEECS を変更するか、あるいは構造をユーザーのプログラムにコピーし、引用符をアポストロフィに変更する必要があります。

プログラム例

DSQUXCTC という名前の IBM 提供の編集出口プログラムの例が、QMF サンプル・ライブラリー QMF710.SDSQSAPE にあります。このプログラムにはコメントが数多く付いており、オンラインで見ると印刷することも可能です。また、ユーザーの要件に合わせて変更することもできます。このプログラムを使用する場合は、ユーザーのプログラム・ライブラリーにコピーして、名前を DSQUECIC に変更してください。

COBOL 編集ルーチンと QMF との対話方法

QMF とユーザー編集インターフェース DSQUEDIT との間のインターフェース制御ブロックは DXEECS です。これはユーザーの編集コードを含み、ユーザー編集ルーチンが使用するためのスクラッチ・パッド域を用意します。制御ブロックは、ユーザー編集ルーチンへの呼び出しの間も存続しています。スクラッチパッド域は、出口ルーチンの最初の呼び出しの後には、QMF により変更されることはありません。

424ページの図112 は、アセンブラの DXEECS 制御ブロックを示しています。

QMF 書式のためのユーザー独自の編集コードの作成

```

***** 00001000
*      * 00002000
* CONTROL BLOCK NAME: DXEECS (COBOL VERSION) * 00003000
*      * 00004000
* FUNCTION: * 00005000
*      * 00006000
* THIS IS THE INTERFACE CONTROL BLOCK BETWEEN QMF AND * 00007000
* THE USER EDITING INTERFACE, DSQUEDIT (TSO/CMS), OR * 00008000
* DSQUEECIC (CICS). * 00009000
*      * 00010000
* IT CONTAINS THE USER'S EDIT CODE, IDENTIFIES THE SOURCE * 00011000
* DATA AND THE TARGET LOCATION FOR THE EDITED RESULT * 00012000
* AND PROVIDES A SCRATCHPAD AREA FOR THE USER EDIT * 00013000
* ROUTINE'S USE. * 00014000
*      * 00015000
* THIS CONTROL BLOCK IS PERSISTENT BETWEEN CALLS TO THE * 00016000
* USER EDIT ROUTINE. * 00017000
*      * 00018000
* THE SCRATCHPAD AREA WILL NOT BE MODIFIED BY QMF AFTER * 00019000
* THE INITIAL INVOCATION OF THE EXIT ROUTINE. * 00020000
*      * 00021000
*      * 00022000
* NOTE: THIS FILE IS DESIGNED TO BE COPIED INTO THE LINKAGE * 00023000
* SECTION OF THE USER EDIT ROUTINE. * 00024000
*      * 00025000
*      * 00026000
* STATUS: VERSION 7 RELEASE 1 LEVEL 0 * 00027000
*      * 00028000
* INNER CONTROL BLOCKS: NONE * 00029000
*      * 00030000
* CHANGE ACTIVITY: NEW CONTROL BLOCK * 00031000
*      * 00032000
* CHANGE DATE: * 00033000
*      * 00034000
***** 00035000
00036000
01 DXEECS. 00037000
02 ECSNAME PICTURE X(8). 00038000
* -- CONTROL BLOCK IDENTIFICATION 00039000
00040000
02 ECSEDCTL. 00041000
* -- EDIT CONTROL 00042000
00043000
03 ECSFREQ PICTURE X(1). 00044000
* -- FUNCTION REQUEST 00045000

```

図 112. COBOL DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (1/5)

QMF 書式のためのユーザー独自の編集コードの作成

```

88 ECS-EDIT-FUNCTION      VALUE "E".                00046000
88 ECS-TERMINATE-FUNCTION VALUE "T".                00047000
*
*      ---- TERMINATE FUNCTION TO FREE RESOURCES. 00048000
*      QMF WILL CALL THE USER EDIT ROUTINE      00049000
*      FOR THIS FUNCTION ONLY IF THE USER       00050000
*      EDIT ROUTINE HAS PREVIOUSLY REQUESTED    00051000
*      IT. (SEE ECSRQMF BELOW.)                 00052000
03 ECSPAD10      PICTURE X(3).                    00053000
*
*      -- RESERVED FIELD                        00054000
03 ECSECODE      PICTURE X(5).                    00055000
*
*      -- EDIT CODE FROM FORM OBJECT           00056000
03 ECSPAD20      PICTURE X(3).                    00057000
*
*      -- RESERVED FIELD                        00058000
03 ECSDECPT      PICTURE X(1).                    00059000
*
*      -- SYMBOL FOR DECIMAL POINT             00060000
*      -- (AS DEFINED BY DECIMAL OPTION IN     00061000
*      -- CURRENT PROFILE OBJECT               00062000
03 ECSTHSEP      PICTURE X(1).                    00063000
*
*      -- SYMBOL FOR THOUSANDS SEPARATOR       00064000
*      -- (AS DEFINED BY DECIMAL OPTION IN     00065000
*      -- CURRENT PROFILE OBJECT               00066000
03 ECSPAD30      PICTURE X(6).                    00067000
*
*      -- RESERVED FIELD                        00068000
03 ECSQMF        PICTURE X(20).                   00069000
*
*      -- AREA RESERVED FOR QMF'S USE          00070000
*
*
02 ECSINDTA.    00071000
*
*      -- DESCRIPTION OF THE INPUT DATA       00072000
*
*
03 ECSINTYP      PICTURE S9(9) COMPUTATIONAL.     00073000
*
*      -- DATA TYPE OF THE INPUT AS IT       00074000
*      -- EXISTS IN THE DATA BASE.           00075000
*
*
88 ECS-FLOATING-POINT  VALUE IS +480.            00076000
88 ECS-DECIMAL         VALUE IS +484.            00077000
88 ECS-INTEGER         VALUE IS +496.            00078000
88 ECS-SMALL-INTEGER   VALUE IS +500.            00079000
88 ECS-VARCHAR         VALUE IS +448.            00080000
88 ECS-FIXED-CHAR     VALUE IS +452.            00081000
88 ECS-LONG-VARCHAR   VALUE IS +456.            00082000
88 ECS-VARG           VALUE IS +464.            00083000
88 ECS-FIXED-G        VALUE IS +468.            00084000
88 ECS-LONG-VARG     VALUE IS +472.            00085000
88 ECS-DATE           VALUE IS +384.            00086000
88 ECS-TIME           VALUE IS +388.            00087000
88 ECS-TIMESTAMP      VALUE IS +392.            00088000
88 ECS-EXT-FLOATING-POINT VALUE IS +940.        00089000
03 ECSINLEN      PICTURE S9(5) USAGE IS COMPUTATIONAL. 00090000
*
*      -- LENGTH OF INPUT DATA               00091000
03 ECSINPRC      PICTURE S9(2) USAGE IS COMPUTATIONAL. 00092000

```

図 112. COBOL DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (2/5)

QMF 書式のためのユーザー独自の編集コードの作成

```

*          -- PRECISION OF INPUT DATA IF IT IS          00095000
*          -- DECIMAL DATA TYPE (U-TYPE EDIT CODE)      00096000
*          -- OR IF IT WAS ANY NUMERIC DATA TYPE        00097000
*          -- (V-TYPE EDIT CODE)...                      00098000
*          -- ZERO OTHERWISE.                            00099000
03 ECSINSCL PICTURE S9(2) USAGE IS COMPUTATIONAL.        00100000
*          -- SCALE OF INPUT DATA IF IT IS              00101000
*          -- DECIMAL DATA TYPE (U-TYPE EDIT CODE)      00102000
*          -- OR IF IT WAS ANY NUMERIC DATA TYPE        00103000
*          -- (V-TYPE EDIT CODE)...                      00104000
*          -- ZERO OTHERWISE.                            00105000
03 ECSINSGN PICTURE X(1).                                00106000
*          -- SIGN OF CONVERTED NUMERIC DATA            00107000
*          -- (V-TYPE EDIT CODE ONLY)...                 00108000
          88 ECS-POSITIVE VALUE " ".                    00109000
          88 ECS-NEGATIVE VALUE "-".                   00110000
          00111000
03 ECSINNUL PICTURE X(1).                                00112000
*          -- NULL INPUT DATA INDICATOR                 00113000
          88 ECS-NULL-DATA VALUE "N".                  00114000
          00115000
03 ECSPAD40 PICTURE X(10).                               00116000
*          -- RESERVED FIELD                             00117000
          00118000
02 ECSRSDTA.                                            00119000
*          -- DESCRIPTION OF THE RESULT BUFFER           00120000
          00121000
          03 ECSRSLEN PICTURE S9(5) USAGE IS COMPUTATIONAL. 00122000
*          -- LENGTH OF RESULT AREA                     00123000
*          -- (EQUIVALENT TO COLUMN WIDTH IN THE        00124000
*          -- FORM OBJECT                                 00125000
03 ECSPAD50 PICTURE X(12).                               00126000
*          -- RESERVED FIELD                             00127000
          00128000
02 ECSUCTL.                                            00129000
*          -- USER CONTROL AREA                         00130000
          00131000
          03 ECSERRET PICTURE S9(9) USAGE IS COMPUTATIONAL. 00132000
*          -- EDIT ROUTINE ERROR RETURN CODE             00133000
*          -- (SEE QMF-DEFINED ERROR CODES BELOW).      00134000
03 ECSRQMF PICTURE X(1).                                00135000
*          -- REQUEST FOR QMF                           00136000
*          -- (SEE CODE(S) DEFINED BELOW.)              00137000
03 ECSPAD60 PICTURE X(11).                              00138000
*          -- RESERVED FIELD                             00139000
          00140000
02 ECSUSERS.                                           00141000

```

図 112. COBOL DXECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (3/5)

QMF 書式のためのユーザー独自の編集コードの作成

```

*                -- USER SCRATCH PAD AREA                                00142000
                                                                00143000
03 ECSUSERS-ARRAY                                         00144000
    PICTURE X(1)                                          00145000
    OCCURS 256 TIMES.                                    00146000
                                                                00147000
                                                                00148000
                                                                00149000
*****          -- EDIT ROUTINE INPUT DATA
01 ECSINPT.                                               00150000
02 ECSINPTC        PICTURE X(32767).                     00151000
02 ECSINPT-ARRAY  REDEFINES ECSINPTC                    00152000
    PICTURE X(1)                                          00153000
    OCCURS 32767 TIMES.                                   00154000
02 ECSINPT-INTEG- OVL                                     00155000
    REDEFINES ECSINPTC.                                   00156000
03 ECSINPT-INTEG  00157000
    PICTURE S9(9)                                         00158000
    USAGE IS COMPUTATIONAL.                              00159000
03 FILLER          PICTURE X(1)                           00160000
    OCCURS 32763 TIMES.                                   00161000
02 ECSINPT-SMALL- OVL                                     00162000
    REDEFINES ECSINPTC.                                   00163000
03 ECSINPT-SMALL- OVL                                     00164000
    PICTURE S9(4)                                         00165000
    USAGE IS COMPUTATIONAL.                              00166000
03 FILLER          PICTURE X(1)                           00167000
    OCCURS 32765 TIMES.                                   00168000
02 ECSINPT-FLOAT- OVL                                     00169000
    REDEFINES ECSINPTC.                                   00170000
03 ECSINPT-FLOAT- OVL                                     00171000
    USAGE IS COMPUTATIONAL-2.                             00172000
03 FILLER          PICTURE X(1)                           00173000
    OCCURS 32759 TIMES.                                   00174000
                                                                00175000
                                                                00176000
*****          -- EDIT ROUTINE RESULT BUFFER
01 ECSRSLT.                                               00177000
02 ECSRSLT-ARRAY  PICTURE X(1)                           00178000
    OCCURS 1 TO 32767 TIMES                               00179000
    DEPENDING ON ECSRSLLEN.                              00180000
                                                                00181000
                                                                00182000

```

図 112. COBOL DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (4/5)

QMF 書式のためのユーザー独自の編集コードの作成

```
***** 00183000
* * 00184000
* THE DATA DEFINITIONS BELOW ARE FOR DOCUMENTATION * 00185000
* PURPOSES ONLY SINCE COBOL DOES NOT ALLOW LINKAGE * 00186000
* SECTION DATA DEFINITIONS TO HAVE VALUE CLAUSES * 00187000
* * 00188000
***** 00189000
00190000
***** -- QMF-DEFINED VALUES FOR ECSERRET 00191000
* (SEE ABOVE). 00192000
*77 ECS-UNKNOWN-EDIT-CODE 00193000
* PICTURE S9(9) VALUE IS +99101 00194000
* USAGE IS COMPUTATIONAL. 00195000
*77 ECS-IMPROPER-DATA-TYPE 00196000
* PICTURE S9(9) VALUE IS +99102 00197000
* USAGE IS COMPUTATIONAL. 00198000
*77 ECS-INVALID-DATA-VALUE 00199000
* PICTURE S9(9) VALUE IS +99103 00200000
* USAGE IS COMPUTATIONAL. 00201000
*77 ECS-INPUT-DATA-TOO-SHORT 00202000
* PICTURE S9(9) VALUE IS +99104 00203000
* USAGE IS COMPUTATIONAL. 00204000
*77 ECS-RESULT-BUFFER-TOO-SHORT 00205000
* PICTURE S9(9) VALUE IS +99105 00206000
* USAGE IS COMPUTATIONAL. 00207000
00208000
00209000
***** -- POSSIBLE REQUEST-FOR-QMF CODES 00210000
* (SEE ECSRQMF ABOVE). 00211000
*77 ECS-CALL-FOR-TERMINATE 00212000
* PICTURE X(1) VALUE IS "T". 00213000
```

図 112. COBOL DXEECS 制御ブロックのユーザー編集ルーチンのフィールド定義 (5/5)

2 バイト文字セット・データの処理

2 バイト文字セット (DBCS) データは、文字の列または図形データ・タイプの列 (GRAPHIC、VARGRAPHIC、および LONG VARGRAPHIC) で表示できません。このタイプのデータを処理するルーチンを設計する必要がある場合は、この節をお読みください。

日本語 DBCS で表現される文字は、ラテン文字およびカタカナ文字です。ラテン文字は以下の特性があります。

- 文字の最初の (左端) バイトは、値 X'42' を持ちます。
- 文字の 2 番目のバイトは、EBCDIC と同等の値を持ちます。

カタカナ文字には以下の特性があります。

- 文字の最初のバイトは X'43' を含みます。
- 2 番目のバイトは、EBCDIC と同等の値を持ちます。

DBCS データの編集コード

Uxxxx もしくは Vxxxx のいずれかの編集コードが DBCS データに使用できません。編集ルーチンが受け取るデータは同じです。

編集ルーチンが受け取るもの

フォーマット設定されるデータは ECSINPT フィールドにあり、そのデータの長さは ECSINLEN にバイトで入っています。ECSINPT にあるものは、データがどこから生じたかによってある程度異なります。さらに正確には、そのデータを含む列が文字列か、または図形データ・タイプかによって異なります。

漢字列からのデータ

フォーマット設定されるデータが漢字データ・タイプの列からのものである場合は、ECSINPT のテキストは、このデータの前にシフト文字が 1 つ付き、さらに後ろにもシフト文字が 1 つ付きます。いずれのシフト文字も 1 バイトです。DBCS 端末では、シフト文字は DBCS 文字の始まりと終わりを示しています。

So は DBCS スtringの開始を示すシフト文字であり、Si はその終了を示すシフト文字です。So の値は X'0E' です。Si の値は X'0F' です。

シフト文字は ECSINLEN に記録されるデータ長に含まれます。したがって、ECSINLEN 内に現れる長さは、常に実際のデータの長さより 2 バイト大きくなります。データは DBCS 文字であると推定されるため、その長さ (バイト) は常に偶数です。

文字列からのデータ

処理されるデータが文字の列からのものである場合は、ECSINPT 内のデータは列データの単なるコピーです。図形列からのデータとは異なり、このデータは DBCS 文字に加えて 1 バイト文字およびシフト文字をもつことができます。DBCS 文字を見つけるには、DBCS スtringを囲んでいる So および Si 文字を探す必要があります。ECSINPT 内に So または Si 文字がない場合、このスtringには DBCS データはありません。たとえば、ECSINPT に次のようなスtringが入っています。

```
ccccSodededededededeSi ccSodededededeSi
```

ここで、c、d、および e は任意のバイトを表し、So および Si はシフト・バイトを表しています。シフト・バイトの配置から、すべての c が 1 バイト文字を表しており、すべての de が DBCS 文字を表していることが分かります。

QMF 書式のためのユーザー独自の編集コードの作成

1 バイト文字はラテン文字、アラビア数字、および正符号や括弧などの特殊文字を表すことができます。日本語の DBCS の場合、カタカナ文字も表すことができます。小文字のラテン文字を意味する一部のバイトにはカタカナ記号として表示されるものもあります。小文字の英字を含む列とカタカナを含む列を識別する編集コードを考える必要があるかもしれません。

編集ルーチンは必ず正しい結果を戻すこと

結果を ECSRSLT フィールドに戻しますが、このとき未使用のバイトの末尾ブランクを伴います。結果はユーザーの画面上で読み取れるようにします。このことは、結果として得られる DBCS および EBCDIC 文字は適切な表現でなければならないこと、さらに DBCS 文字はいずれもその始まりと終わりに So と Si 文字が付いていなければならないことを意味しています。

ECSRSLT フィールドのオーバーフロー

ECSRSLT フィールドがオーバーフローしないように気を付けてください。このフィールドの長さは ECSRSLEN フィールドにあります。結果が入りきらない場合には、右側を切り捨てます。切り捨てられる結果で表示している最後の文字が DBCS 文字の場合は、必ず、右端のバイトを残しておき、その文字に Si 文字を続けるようにしてください。

報告書列の印刷

QMF は、ECSRSLT フィールドを対応する報告書列にコピーします。その結果は、報告書列の幅と完全に同じ幅です。データに ALIGNMENT を指定していないと、データは入力したとおりの位置合わせになります。

報告書を出力する装置が戻されたものをどのように表現するかは、装置の特性によって異なります。一部の端末では、以下のような規則が適用されます。

- 報告書が画面に表示される場合、結果に埋め込まれた Si および So 文字も端末上に表示されます。
- Si および So 文字は、ブランクまたは特殊記号として表示されます。特殊記号は Si 用に 1 つ、So 用に 1 つあります。
- ユーザーが特別な組み合わせのキーを押さない限り、特殊記号の代わりにブランクが表示されます。

その他の装置の場合は、この規則は多少異なります。

オンライン・ヘルプでは DBCS 文字の使用に関する指示は、照会および QMF コマンドでは特定の DBCS 文字を使用しないように勧めています。編集ルー

チンにより戻されるフォーマット設定されたデータには、この制約事項は適用されません。適正な DBCS 文字はすべて、ECSRSLT フィールドに戻されます。

第21章 管理プログラム出口ルーチンを使用した QMF リソースの制御

注: 本章には、汎用プログラミング・インターフェースとそれに関連する、指針としての情報が含まれます。

管理プログラム出口ルーチンにより、エンド・ユーザー・アクティビティーを制限し、またインストール先システムでのコンピューター・リソースの使用を制御できます。IBM は QMF で管理プログラム出口ルーチンを提供しています。ここでは、QMF コマンドの実行に費やされる時間、あるいは、データベースからユーザーが検索可能な行数に対して、デフォルトの限界値が設定されています。このデフォルトの出口ルーチンを使用したり、あるいはアセンブラーを使用してルーチンを変更したり、ユーザー独自のものを作成することもできます。

QMF 管理プログラムと一緒に DB2 管理プログラムを使用して、動的に SELECT、INSERT、UPDATE、および DELETE 照会を実行するときに費やされるプロセッサ時間をモニターすることができます。DB2 管理プログラムは独立して使用することも可能です。

また、QMF OS/390 高性能オプション / マネージャー (HPO/ マネージャー) を使用して、QMF セッション・アクティビティーの管理と制御を行うことができます。HPO/ マネージャーには、QMF セッション・アクティビティーとのリアルタイム・ユーザー・インターフェースならびに実行前に照会のリソース使用を見積もる照会分析機能があります。HPO/ マネージャーは、QMF 管理プログラムを一時変更します。HPO 機能の詳細については、*QMF High Performance Option User's Guide for OS/390* を参照してください。

クイック・スタート

表53 のステップを、管理プログラム出口ルーチンの設定および使用上のガイドとして使用してください。各ステップに関するより詳細な情報が必要な場合は、表の右端に示されているページを参照してください。

管理プログラム出口ルーチンを使用した QMF リソースの制御

表 53. 管理プログラム出口ルーチンの使用

作業	参照箇所
データベースから検索された行数が 25 000 に達したときにユーザーにプロンプトを出し、さらに行数が 100 000 に達するとデータ検索を取り消す場合、Q.RESOURCE_VIEW の INTVAL フィールドを 0 (ただし、RESOURCE_GROUP=SYSTEM および RESOURCE_OPTION=SCOPE) に設定することにより、管理プログラムをオンにします。次に、ユーザーのプロファイルの RESOURCE_GROUP フィールドを SYSTEM に更新し、データベースを再接続します。	436 ページ
プロセッサ時間が 6 分 経過したときにユーザーにプロンプトを出し、さらにプロセッサ時間が 24 分 経過するとデータを取り消す場合、Q.RESOURCE_VIEW の INTVAL フィールドを 0 (ただし、RESOURCE_GROUP=SYSTEM および RESOURCE_OPTION=SCOPE) に設定することにより、管理プログラムをオンにします。次に、ユーザーのプロファイルの RESOURCE_GROUP フィールドを SYSTEM に更新し、データベースを再接続します。このプロンプトのオプションは CICS では使用できません。	436 ページ
プロセッサ時間のデフォルト値 6 分 または 24 分 以外の時間制限、あるいはデータベース行制限のデフォルト値 25 000 および 100 000 以外の行制限を使用して管理プログラム出口ルーチンを設定する場合、Q.RESOURCE_TABLE に新しい行を追加して、ユーザーに警告を与え (オプション)、データ検索を取り消すためのポイントを定義します。管理プログラムをオンにして、ユーザーのプロファイルをステップ 434 で説明しているように更新します。この時間オプションは CICS では使用できません。	441 ページ
データベース検索の行数以外のアクティビティを制限する場合は、アセンブラーを使用して IBM 提供の管理プログラム出口ルーチンを変更するか、またはユーザー独自の出口ルーチンを作成します。	445 ページ
CICS において、 IBM 提供の管理プログラム出口ルーチンを変更するか、またはユーザー独自の出口ルーチンを作成する場合、そのルーチンを変換、アセンブル、およびリンク・エディットします。	478 ページ
TSO およびネイティブ OS/390 バッチ において、 IBM 提供の管理プログラム出口ルーチンを変更するか、またはユーザー独自の出口ルーチンを作成する場合、そのルーチンをアセンブルし、リンク・エディットします。	480 ページ
照会の SELECT 、 INSERT 、 UPDATE 、および DELETE を動的に実行するときに費やされるプロセッサ時間をモニターする場合、QMF 管理プログラムと一緒に DB2 管理プログラムを使用します。最大プロセッサ時間をリソース限定表 (RLST) に設定します。	481 ページ

IBM 提供の管理プログラム出口ルーチンの使用

CICS 用に提供される管理プログラム出口ルーチン (DSQUEGV3) は、データベースから検索できる行数を制御します。TSO およびネイティブ OS/390 バッチ 用に提供される管理プログラム出口ルーチン (DSQUEGV1) は、データベースから検索できる行数を制御します。管理プログラム出口ルーチンは出荷時に、以下のような行数についての事前定義値を 2 つ持っています。

管理プログラム出力ルーチンを使用した QMF リソースの制御

- 行プロンプト値は、検索した行数が 25 000 に達した時点で、図 113 に示すメッセージを表示し、ユーザーに警告を与えます。

```
DSQUn00 QMF governor prompt:  
Command has fetched 25000 rows of data.  
  
==> To continue QMF command press the "ENTER" key.  
==> To cancel QMF command type "CANCEL" then press the "ENTER" key  
==> To turn off prompting type "NOPROMPT" then press the "ENTER" key
```

図 113. リソース限界に近づいたときに表示されるメッセージ。図の中のシンボル n は、xviii ページの表 1 からの NLID を表しています。

重要: 取り消しのプロンプトが表示されても、データベースのアクティビティは中断されません。DB2 は行の取り出しを継続し、プロセッサ時間も使い続けます。

- 行限界値は、図 113 のメッセージに応答してユーザーが Enter キーを押す場合に、100 000 行が検索されたときにデータ検索を取り消します。IBM 提供の管理プログラムがデータ検索を取り消すと、図 114 に示すメッセージが表示されます。

```
Row limit exceeded! Your command canceled by QMF governor.
```

図 114. リソース限界を超えたときに表示されるメッセージ

プロシージャを実行しているとき、図 114 のメッセージではなく、ユーザーのプロシージャが取り消された旨のメッセージを受け取る場合があります。たとえば、報告書の完了を必要とするコマンド (ERASE など) がユーザーのプロシージャに含まれている場合には、図 115 に示すようなメッセージを受け取ります。

```
Procedure canceled.
```

図 115. プロシージャが取り消されたときに表示されるメッセージ

243 ページの『インストール先システムのためのプロファイル構造の確立』で説明されている SYSTEM プロファイルを使用しているユーザーは、25 000 および 100 000 のデフォルト値を使用するようにすでに設定済みです。固有のプロファイルをもつユーザーのためにデフォルト値をアクティブ化するには、436 ページの『デフォルトの限界のアクティブ化』を参照してください。

管理プログラム出口ルーチンを使用した QMF リソースの制御

TSO およびネイティブ OS/390 バッチ では、QMF コマンドを実行する際に費やされる時間に関する、さらに 2 つの事前定義値 (時間制限値および時間指示値) を持っています。

- 時間プロンプト値は、サイクルに関するプロセッサ時間が 6 分に達した時点で、図116 に示すメッセージを表示し、ユーザーに警告を与えます。

```
DSQUn00 QMF governor prompt:
Command has executed for 6 minutes

==> To continue QMF command press the "ENTER" key.
==> To cancel QMF command type "CANCEL" then press the "ENTER" key
==> To turn off prompting type "NOPROMPT" then press the "ENTER" key
```

図116. リソース限界に近づいたときに表示されるメッセージ. 図の中のシンボル n は、xviiiページの表1 からの NLID を表しています。

- 時間制限値は、サイクルの間にプロセッサ時間が 24 分経過すると、コマンドを取り消します。

ユーザーが警告を受ける時点およびデータ検索を取り消される時点について、ユーザー独自の限界値を定義したい場合には、441ページの『ユーザー独自のリソース限界の定義』を参照してください。

デフォルトの限界のアクティブ化

データベースの検索行数が 25 000 に達するとユーザーに警告を与え、100 000 の時点で QMF アクティビティーを取り消すには、以下の手順を行ってください。

1. SQL 照会パネルから、図117 に示す照会を行います。

```
UPDATE Q.RESOURCE_VIEW
SET INTVAL=0
WHERE RESOURCE_OPTION='SCOPE' AND
      RESOURCE_GROUP='SYSTEM'
```

図117. IBM 提供の管理プログラムのデフォルト値をアクティブ化

2. ユーザーのプロファイルの RESOURCE_GROUP フィールドの SYSTEM 値を設定します。たとえば、437ページの図118 の UPDATE ステートメントでは、ユーザー JONES (英語版 QMF を使用) およびユーザー SCHMIDT (ドイツ語版 QMF 使用) のデフォルト値をアクティブ化しています。

管理プログラム出口ルーチンを使用した QMF リソースの制御

重要: TRANSLATION 列の値を常に指定するか、または Q.PROFILES 内の行を予定以上に変更することもできます。

基本 QMF (英語)

ドイツ語 NLF

UPDATE Q.PROFILES

UPDATE Q.PROFILES

SET RESOURCE_GROUP = 'SYSTEM'

SET RESOURCE_GROUP = 'SYSTEM'

WHERE CREATOR='JONES' AND

WHERE CREATOR='SCHMIDT' AND

TRANSLATION='ENGLISH'

TRANSLATION='DEUTSCH'

図 118. ユーザー・リソース・グループの更新

重要: QMF を開始する際、DSQSPRID パラメーター値が TSOID であれば、リソース・グループ名はユーザー ID になります。

Q.PROFILES 表で新規にユーザー・プロファイルを作成する方法の詳細については、242ページの『QMF へのユーザー・アクセスを可能にするユーザー・プロファイルの作成』を参照してください。

3. 現行の QMF セッションを終了し、別のセッションを開始して新しい値をアクティブ化するように、ユーザーに指示します。

『管理プログラム出口ルーチンのリソースの制御方法』では、管理プログラムがどのように Q.RESOURCE_VIEW および Q.PROFILES 表の情報を使用して、リソースを制御しているのかを説明しています。

行の限界を、デフォルト値の 25 000 および 100 000 以外に定義したい場合には、『管理プログラム出口ルーチンのリソースの制御方法』を参照してください。その後、441ページの『ユーザー独自のリソース限界の定義』の手順を参照してください。

管理プログラム出口ルーチンのリソースの制御方法

管理プログラムは、リソースを制御するために 2 種類のリソースを使用します。

- リソース限界に関してユーザー用に設定する情報。 Q.RESOURCE_TABLE と呼ばれるリソース制御表に定義されています。
- ユーザーのセッションの状態に関する情報。ユーザーのリソース・グループに定義されたリソース限界に、ユーザーのアクティビティーがどの程度近づ

管理プログラム出口ルーチンを使用した QMF リソースの制御

いてきているかを、管理プログラムに伝えます。この情報は、IBM 提供の制御ブロック DXEGOVA および DXEXCBA にある管理プログラム出口ルーチンに渡されます。

管理プログラムがリソース限界を知る方法

IBM 提供の Q.RESOURCE_TABLE の各行には、以下のものが含まれています。

- リソース・グループの名前 (RESOURCE_GROUP)。これは、同じ方法で管理する 1 人または複数のユーザーのアクティビティーの特徴を示すものです。
- リソースの名前 (RESOURCE_OPTION)。RESOURCE_GROUP に名前のあるユーザーのグループに制限するリソースです。
- リソース・オプションの限界を定義する値 (INTVAL、FLOATVAL、または CHARVAL)。リソース・オプションとしては、整数値、浮動小数点値、または文字値が可能です。

445ページの表54 は、IBM から出荷された状態の Q.RESOURCE_TABLE の構造を示しています。Q.RESOURCE_TABLE には Q.RESOURCE_INDEX 索引があります。キー付きの列は、RESOURCE_GROUP と RESOURCE_OPTION です。

QMF の旧リリースから移行する場合: QMF の旧リリースには、
Q.RESOURCE_INDEX は含まれていません。

Q.RESOURCE_TABLE は SYSTEM と呼ばれる事前定義のリソースと一緒に IBM から出荷されます。SYSTEM リソース・グループは、図119 に示すように、3 つの事前定義されたリソース・オプションを持っています。このグループは TSO およびネイティブ OS/390 バッチ の場合には、さらに 3 つの時間オプションを持っています。CHARVAL 列を使用して、示されているように各行に定義されている限界を指示します。

リソース グループ	リソース オプション	整数値	浮動小数点値	文字値
SYSTEM	SCOPE	-	-	INDICATE WHETHER GOVERNOR IS ACTIVE
SYSTEM	ROWLIMIT	100000	-	CANCEL AFTER FETCHING 100000 ROWS
SYSTEM	ROWPROMPT	25000	-	PROMPT USER AFTER FETCHING 25000 ROWS
SYSTEM	TIMELIMIT	1440	-	CANCEL AFTER 24 MINUTES CPU
SYSTEM	TIMEPROMPT	360	-	PROMPT USER AFTER 6 MINUTES CPU
SYSTEM	TIMECHECK	900	-	15 MINUTES INTERVAL BETWEEN TIME CHECK

図 119. IBM 提供の管理プログラム出口のデフォルトのリソース・グループとオプション

SCOPE = 0

特定リソース・グループの管理をアクティブ化します。

ヌルを含む SCOPE に対する非ゼロ値は、リソース・グループの管理を非アクティブ化します。

ROWLIMIT = 100000

ユーザーが警告を受けたとき継続することを決定した場合、管理プログラム出口ルーチンは、100 000 行検索された時点でデータ検索アクティビティを取り消します。(検索は FETCH の場合のみ。) ROWLIMIT はバッファー・サイズによって異なります。したがって、バッファーに 100 000 で割りきれない行数が入っている場合は、100 000 行以上検索することもあり得ます。

ROWPROMPT = 25000

データベースが 25 000 行以上検索されると、ユーザーに警告を与えます。

TSO およびネイティブ OS/390 バッチ の場合、さらに 3 種類のオプションがあります。

TIMELIMIT = 1440

警告を受けたときに、ユーザーが継続することにした場合には、管理プログラム出口ルーチンは、プロセッサ時間が 24 分経過した時点でコマンドを取り消します。TIMELIMIT は TIMECHECK 間隔ごとに検査されます。したがって、TIMECHECK 間隔が 24 で割りきれない間隔に設定された場合には、プロセッサ時間は 24 分以上経過することがあります。TIMELIMIT は TIMECHECK 間隔が処理された後に評価されます。

プロセッサ時間: プロセッサ時間 とは、ジョブ・ステップ時間に SBR (サービス要求ブロック) 時間を加えたものを指します。

TIMEPROMPT = 360

プロセッサ時間が 6 分経過した時点でユーザーに警告を与えます。TIMECHECK 間隔が処理された後で評価されます。

TIMECHECK = 900

時間の検査、プロンプト、または取り消し間の実際の時間を 15 分と指定します。

IBM は、Q.RESOURCE_VIEW と呼ばれるこの表の視点も提供しています。これには、Q.RESOURCE_TABLE の 5 つの列がすべて含まれています。QMF

管理プログラム出口ルーチンを使用した QMF リソースの制御

が管理プログラム出口ルーチンを呼び出すたびに、QMF は Q.RESOURCE_VIEW に保管されているリソース制御情報をそのルーチンに渡します。管理プログラム出口ルーチンは、このリソース情報を使用してユーザーがいつリソースの限界に達するかを判別するのに役立ってます。

管理プログラムがリソース限界の到達を知る方法

管理プログラム出口ルーチンの呼び出し時に、QMF は Q.RESOURCE_VIEW を照会します。これは、ユーザーの属するリソース・グループのリソース制御表で定義されているリソースの限界を示すものです。リソース・グループを判別するには、QMF は Q.PROFILES 表のユーザー行の RESOURCE_GROUP フィールドの値を検査して、Q.RESOURCE_VIEW で一致する値を検査します。

QMF は 2 つの制御ブロック DXEGOVA と DXEXCBA を使用して、情報を管理プログラム出口ルーチンに渡します。DXEGOVA 制御ブロックは、各ユーザーに設定した限界に関する Q.RESOURCE_VIEW からの情報を持っています。DXEXCBA 制御ブロックは、ユーザーが現行の QMF セッションで行っているアクティビティーに関する情報を含んでいます。これは、管理プログラムにユーザーがリソースの限界にどの程度近づいているかを知らせるものです。

図120 は、管理プログラムがリソースの使用を制限する方法を示しています。

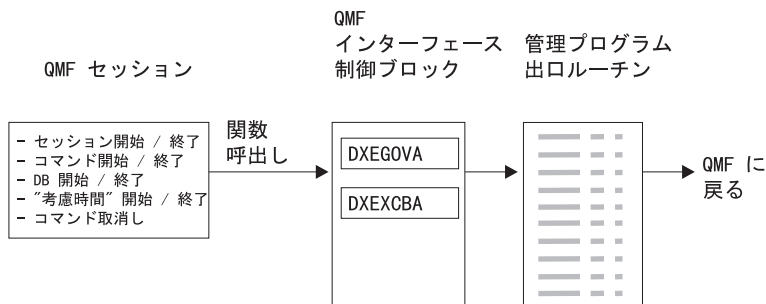


図120. 管理プログラム出口ルーチンと QMF の関係

QMF は 図120 に示すように、管理プログラム出口ルーチンを、QMF セッション内のいろいろな時点で呼び出します。このような呼び出しは、機能呼び出しと呼ばれています。機能呼び出しに関する詳細については、450ページの『QMF が管理プログラムを呼び出す時点』を参照してください。

リソース限界に到達すると起こること

QMF が管理プログラム出口ルーチンに渡すリソース制御情報が、リソース限界に達したことを示すとき、IBM 提供の管理プログラム出口ルーチンは QMF

取り消しサービスを呼び出し、ユーザーが実行しようとする QMF アクティビティーを取り消し、435ページの図114 のメッセージが表示されます。

『デフォルトの限界のアクティブ化』で述べた行数の限界にデフォルト値を使用する場合は、IBM 提供の管理プログラム出口ルーチンは 435ページの図113 で示すように、アクティビティーを取り消す前に警告も表示します。検索行数にデフォルト値を使用しない場合に、この警告をアクティブ化する方法については、『ユーザー独自のリソース限界の定義』を参照してください。

IBM 提供の管理プログラム出口ルーチンは、QMF に制御を戻すときに行数のカウントをリセットし、管理プログラムの呼び出しごとに行数が累積されないようにしています。

ユーザー独自のリソース限界の定義

この節では、新規のリソース・グループを作成する方法について説明します。このリソースとは、データベースから検索される行数のことです。行数以外のリソースの限界を定義する場合は、アセンブラーを使用して IBM 提供の管理プログラム出口ルーチンを変更するか、またはユーザー独自の出口ルーチンを作成する必要があります。使用できる機能の詳細については、445ページの『IBM 提供の管理プログラム出口ルーチンの変更またはユーザー独自のルーチン作成』を参照してください。

以下の手順を使用して、リソース制御表に新しいリソース・グループを追加します。この手順では、GROUP1 という名前のリソース・グループを追加します。これは、行数が 10 000 に達したときに管理プログラムは GROUP1 のユーザーにプロンプトを出し、行数が 15 000 に達したとき、または (TSO およびネイティブ OS/390 バッチ の場合) 1000 秒が経過したときに、ユーザーのアクティビティーを取り消すというものです。また、この手順では、ユーザーをリソース・グループに追加する方法についての例も示しています。

1. 図121 の照会を実行して、ユーザーがリソースの限界に近づいたときに警告を受ける行数を設定します。

行数の限界に近づいてもユーザーに警告を与えたくない場合は、442ページのステップ 2 にスキップしてください。

```
INSERT INTO Q.RESOURCE_VIEW (RESOURCE_GROUP,RESOURCE_OPTION,INTVAL)
VALUES('GROUP1','ROWPROMPT',10000)
```

図 121. 行限界のプロンプトのアクティブ化

管理プログラム出口ルーチンを使用した QMF リソースの制御

2. 図122 の照会を実行して、管理プログラムがユーザーのアクティビティーを取り消す行数を設定します。

```
INSERT INTO Q.RESOURCE_VIEW (RESOURCE_GROUP,RESOURCE_OPTION,INTVAL)
VALUES('GROUP1','ROWLIMIT',15000)
```

図 122. ユーザーが行限界に達したときアクティビティー取り消しをアクティブ化

3. 図123 を実行して、リソースの限界に近づいたときに警告を受けるまでのプロセッサ経過時間を設定します。

経過時間の限界に近づいてもユーザーに警告を与えたくない場合は、ステップ 4 にスキップしてください。

```
INSERT INTO Q.RESOURCE_VIEW (RESOURCE_GROUP,RESOURCE_OPTION,INTVAL)
VALUES('GROUP1','TIMEPROMPT',300)
```

図 123. 時間限界のプロンプトのアクティブ化

4. 図124 の照会を実行し、管理プログラムがユーザーのアクティビティーを取り消すまでのプロセッサ経過時間を設定します。

```
INSERT INTO Q.RESOURCE_VIEW (RESOURCE_GROUP,RESOURCE_OPTION,INTVAL)
VALUES('GROUP1','TIMELIMIT',1000)
```

図 124. ユーザーが時間限界に達したときアクティビティー取り消しをアクティブ化

5. 図125 の照会を実行し、管理プログラムがユーザーのアクティビティーを検査する実際の時間間隔を設定します。

```
INSERT INTO Q.RESOURCE_VIEW (RESOURCE_GROUP,RESOURCE_OPTION,INTVAL)
VALUES('GROUP1','TIMECHECK',800)
```

図 125. 時間間隔検査のアクティブ化

6. 図126 に示されている照会を実行して、GROUP1 リソース・グループの管理プログラムをオンにしてください。

```
INSERT INTO Q.RESOURCE_VIEW (RESOURCE_GROUP,RESOURCE_OPTION,INTVAL)
VALUES('GROUP1','SCOPE',0)
```

図 126. 特定のリソース・グループに対する管理プログラムの開始

管理プログラム出口ルーチンを使用した QMF リソースの制御

SCOPE は管理をアクティブ化あるいは非アクティブ化させるリソース・オプションです。Q.RESOURCE_TABLE 内の各リソース・グループは、SCOPE と呼ばれる RESOURCE_OPTION を、また SCOPE は対応するゼロの INTVAL をもつ必要があります。そうでないとリソース・グループは管理されません。管理を非アクティブ化する場合は、INTVAL を 1 に設定します。

7. 図127 と同様の照会を実行して、ユーザー JONES を英語版 QMF 環境のリソース・グループ GROUP1 に追加します。

```
UPDATE Q.PROFILES
SET RESOURCE_GROUP='GROUP1'
WHERE CREATOR='JONES' AND
TRANSLATION='ENGLISH'
```

図 127. ユーザー・リソース・グループの更新

NLF を使用する場合: 同様の照会を使用して、NLF 環境におけるユーザーのプロファイルを更新しますが、xviiiページの表1からの TRANSLATION 値を使用します。

8. プロファイルを更新したユーザーに対して、現行の QMF セッションを終了させ、新規の値をアクティブ化するために別のセッションを開始するように指示します。

ユーザー独自のリソース制御表の作成

ユーザーは自分用の表を作成したり、Q.RESOURCE_TABLE の名前変更をすることができます。また、Q.RESOURCE_VIEW がユーザーの作成した表で定義されている視点であり、また、この表に 445ページの表54 で示されたすべての列が含まれている場合には、ユーザーが作成した表に追加の列を組み込むことも可能です。

444ページの図128 は、MY_RESOURCES という表の作成に使える SQL ステートメントの例を示しています。照会の中で、ユーザーの表、列、および表スペースの名前と置き換えてください。新規の表を作成する前に、データベースから Q.RESOURCE_TABLE を消去することを確認してください。これは、Q.RESOURCE_VIEW がこの表に定義されているからです。

```
DROP TABLE Q.RESOURCE_TABLE
```

Q.RESOURCE_TABLE を除去するとデータベースから Q.RESOURCE_VIEW も除去されるため、444ページの図128 および 444ページの図129 に示されているように、表も視点も両方とも再作成する必要があります。

管理プログラム出口ルーチンを使用した QMF リソースの制御

視点を消去すると、自動的に QMF アプリケーション・プランが無効化されま
す。このため、リソース表および視点の消去と再作成を行う場合は、QMF の
外で作業を行うようにします。QMF が非アクティブである時点を選び、DB2
の DB2I 機能を使用します。DB2I により、対話式に作業を実行することがで
きます。

IBM 提供の表スペースを使用しない場合は、ユーザーが自分用の表スペースを
作成する必要があります。QMF 許可プランを明示的に再バインドする場合
は、BIND 特権もそのプランで必要になります。ユーザーのそれぞれの SQL
コマンドで必要になる権限についての情報は、DB2 UDB (OS/390 版) SQL 解
説書に記載されています。

```
CREATE TABLE MY_RESOURCES
  (GROUP_NAME CHAR(16) NOT NULL,
   CONSTRAINT CHAR(16) NOT NULL,
   INTEGER INTEGER,
   FLOAT_VALUE FLOAT,
   CHARACTER VARCHAR(80))
IN TBSPACE1
```

図 128. リソース制御表の作成または Q.RESOURCE_TABLE の名前変更

Q.RESOURCE_TABLE 以外の表を使用したり、Q.RESOURCE_TABLE に別の
名前を付けることに決めた場合は、必ず、Q.RESOURCE_VIEW を再作成して
ください。その理由は、QMF は表ではなく 視点 を照会して、管理プログラ
ム出口ルーチンに渡すリソース制御情報を得るからです。

Q.RESOURCE_TABLE を作成した後、Q.RESOURCE_VIEW を再作成するに
は、Q.RESOURCE_VIEW に関する SELECT 特権を PUBLIC に与えます。そ
の後、SPUFI を使用して、新規の視点をテストします。最後に、QMF 許可プ
ランを再バインドします。

図129 は、Q.RESOURCE_VIEW を新規の表 MY_RESOURCES の視点として
再定義する方法を示しています。ユーザー独自の表および列名を、図に示した
ものと置き換えます。

```
CREATE VIEW Q.RESOURCE_VIEW
  (RESOURCE_GROUP, RESOURCE_OPTION, INTVAL, FLOATVAL, CHARVAL)
  AS SELECT GROUPNAME, CONSTRAINT, INTEGER, FLOAT_VALUE, CHARACTER
  FROM MY_RESOURCES
```

図 129. Q.RESOURCE_VIEW の再定義

表 54. Q.RESOURCE_TABLE 表の構造

列名	データ・ タイプ	長さ (バイト単位)	ヌルの使用	機能 / 値
RESOURCE_GROUP	CHAR	16	不可	リソース・グループの名前を含む。 Q.PROFILES のユーザー行の RESOURCE_GROUP フィールドを更新し、そのユーザーの管理をアクティブ化する。
RESOURCE_OPTION	CHAR	16	不可	モニターするリソースの名前。
INTVAL	INTEGER		可	整数値をもつリソース・オプションの リソース限界を反映する。たとえば、 データベースの検索行数は整数値のリ ソース。
FLOATVAL	FLOAT		可	浮動小数点値のリソース・オプション のリソース限界を表す。 FLOATVAL は IBM 提供の管理プログラムについ てはヌルです。
CHARVAL	VARCHAR	80	可	文字値のリソース・オプションのリソ ース限界を表す。たとえば、 DAY_OF_WEEK リソース・オプショ ンを確立し、MONDAY を CHARVAL に割り当て、QMF ユーザーが月曜日 にだけ QMF をログオンできるように する。CHARVAL は、IBM 提供の管 理プログラムの場合は、コメント列と して使用。

IBM 提供の管理プログラム出口ルーチンの変更またはユーザー独自のルーチン作成

データベースから戻される行数あるいは経過したプロセッサ時間以外のリソースを管理することに決めた場合は、以下のようにして、IBM 提供の管理プログラム出口ルーチンを変更するか、またはユーザー独自の出口ルーチンを作成する必要があります。

1. QMF が出口ルーチン呼び出す時点での、ルーチンのアドレス可能性を確立します。450ページの『QMF が管理プログラム出口ルーチン呼び出す方法と時期』で、このステップについて説明しています。
2. リソース制御情報を管理プログラム出口ルーチンに渡し、この情報を保管します。458ページの『リソース制御情報を管理プログラム出口に渡す』で、このステップについて説明しています。

管理プログラム出口ルーチンを使用した QMF リソースの制御

3. アクティビティーを取り消すための QMF 取り消しサービスのアドレス可能性を確立する。475ページの『ユーザー・アクティビティーの取り消し』で、このステップについて説明しています。
4. 取り消されたアクティビティーに対するメッセージを提供する QMF メッセージ・サービスへのアドレス可能性を確立します。476ページの『取り消されたアクティビティーに対するメッセージ』で、このステップについて説明しています。
5. CICS の場合、IBM 提供の管理プログラム出口ルーチンを変更したか、ユーザー独自の出口ルーチンを作成したかに関係なく、管理プログラム出口ルーチンを変換し、アセンブルし、リンク・エディットします。478ページの『CICS でのユーザー管理プログラム出口ルーチンの変換、アセンブル、リンク・エディット』で、このステップについて説明しています。
6. TSO およびネイティブ OS/390 バッチ の場合、IBM 提供の管理プログラム出口ルーチンを変更したか、ユーザー独自の出口ルーチンを作成したかに関係なく、管理プログラム出口ルーチンをアセンブルし、リンク・エディットします。480ページの『管理プログラム出口ルーチンの、TSO およびネイティブ OS/390 バッチでのアセンブルとリンク・エディット』で、このステップについて説明しています。

管理プログラム出口ルーチンのプログラム・コンポーネント

管理プログラム出口ルーチンの変更または作成を始める前に、管理プログラム出口ルーチンのコンポーネントの名前とその使用目的を知っておく必要があります。

表55 は、これらの名前を示しています。名前はインストールした言語 (英語または NLF) によって変わります。次の表にある名前の中の n のシンボルを、使用中の NLF と一致する NLID (xviiiページの表1 から) と置き換えてください。コンポーネントの名前の中で、1 は TSO およびネイティブ OS/390 バッチ を表し、3 は CICS を表しています。

表 55. IBM 提供の管理プログラム・コンポーネント

メンバー名	ライブラリー	機能
TSO およびネイティブ OS/390 バッチ		
DSQUnGV1	QMF710.SDSQLOAD	TSO およびネイティブ OS/390 バッチ 用のロード・モジュール
DSQUnGV1	QMF710.SDSQUSRn	TSO およびネイティブ OS/390 バッチ の管理プログラム出口ルーチンのソース・コード

管理プログラム出口ルーチンを使用した QMF リソースの制御

表 55. IBM 提供の管理プログラム・コンポーネント (続き)

DXEUnGV1	QMF710.SDSQUSRn	TSO およびネイティブ OS/390 バッチ の管理プログラムのプロンプトと取り消しメッセージのテキストおよび関連の定義
CICS		
DSQUnGV3	QMF710.SDSQLOAD	CICS 用のロード・モジュール
DSQUnGV3	QMF710.SDSQUSRn	CICS 用の管理プログラム出口ルーチンのソース・コード
DXEUnGV3	QMF710.SDSQUSRn	CICS での管理プログラム取り消しメッセージのテキストと関連の定義
DXEUnGM	QMF710.SDSQUSRn	CICS での管理プログラムのプロンプトの BMS マップ
共通		
DXEGOVA	QMF710.SDSQUSRn	DXEGOVA 制御ブロックの DSECT
DXEXCBA	QMF710.SDSQUSRn	DXEXCBA 制御ブロックの DSECT

これらのメンバーは、QMF プロダクション・ディスクにあります。

NLF を使用する場合: 各言語環境用のモジュール DSQUn GVx の複数の異なるバージョンを使用することにより、英語版の QMF セッションのほか、NLF セッションでもリソースを管理することができます。たとえば、英語版とドイツ語版の両方をインストールしている場合、TSO およびネイティブ OS/390 バッチ の英語版についてはモジュール DSQUEGV1 を使用し、TSO およびネイティブ OS/390 バッチ のドイツ語版についてはモジュール DSQUDGV1 を使用します。

リソース制御表 (Q.RESOURCE_TABLE または自分自身で作成したもの) と Q.RESOURCE_VIEW を、言語環境間で共用することができます。これは、Q.PROFILES 表に英語のプロファイルと NLF のプロファイルを含むことができるのと同じです。

TSO、SRPI、APPC、およびネイティブ OS/390 と管理プログラム出口ルーチンとの対話方法

ユーザーのセッション開始時に、QMF は LOAD コマンドを発行して、管理プログラムをユーザーの仮想記憶域に入れます。パフォーマンス上の理由により、QMF と管理プログラム出口ルーチン間では、アSEMBラー呼び出しインターフェースが使用されます。管理プログラム出口ルーチンは、速いパフォー

管理プログラム出口ルーチンを使用した QMF リソースの制御

マンスを備えている必要があります。その理由は、制御しようとするリソースによっては、データベースから検索される行ごとに毎回出口ルーチンが呼び出される可能性があるからです。

本章においては、ロード・モジュール・ライブラリー QMF710.SDSQLOAD は、ユーザーの STEPLIB データ・セットに連結されているライブラリーに入っているものと想定しています。このモジュールのアセンブルとリンク・エディットについては、480ページの『管理プログラム出口ルーチンの、TSO およびネイティブ OS/390 バッチでのアセンブルとリンク・エディット』で説明しています。

図130 は、管理プログラム出口ルーチンのプログラム構造を示しています。

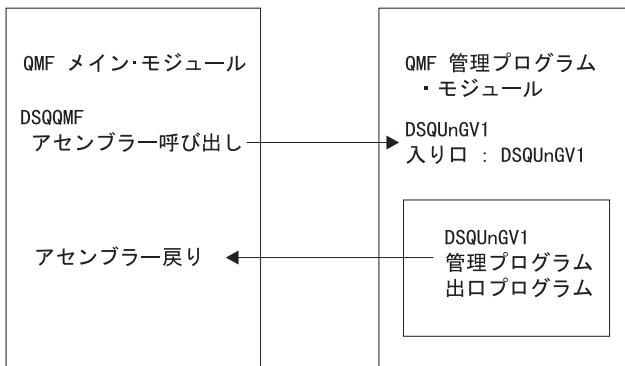


図130. QMF と管理プログラム出口ルーチンを対話させる TSO、SRPI、APPC、またはネイティブ OS/390 の処理

CICS の管理プログラム出口ルーチンとの対話方法

ユーザーのセッション開始時に、QMF は EXEC CICS LOAD コマンドを発行して、管理プログラムをユーザーの仮想記憶域に入れます。パフォーマンス上の理由により、QMF と管理プログラム出口ルーチン間では、アセンブラー呼び出しインターフェースが使用されます。管理プログラム出口ルーチンは、速いパフォーマンスを備えている必要があります。その理由は、制御しようとするリソースによっては、データベースから検索される行ごとに毎回出口ルーチンが呼び出される可能性があるからです。このモジュールのアセンブルとリンク・エディットについては、478ページの『CICS でのユーザー管理プログラム出口ルーチンの変換、アセンブル、リンク・エディット』で説明しています。

CICS 制御ブロックの管理プログラム出口ルーチンへのインターフェースは、以下の部分で構成されます。

管理プログラム出口ルーチンを使用した QMF リソースの制御

- インターフェース制御ブロック DXEXCBA および DXEGOVA。QMF と共に出荷。
- CICS 提供のプロローグおよびエピローグ・マクロ DFHEIENT および DFHEIRET。CICS と共に出荷。
- コマンド・インターフェース・モジュール DFHEAI および DFHEAI0。CICS と共に出荷。
- 管理プログラム出口プログラム。名前は DSQUn GV3。

図131 は、管理プログラム出口ルーチンのプログラム構造を示しています。

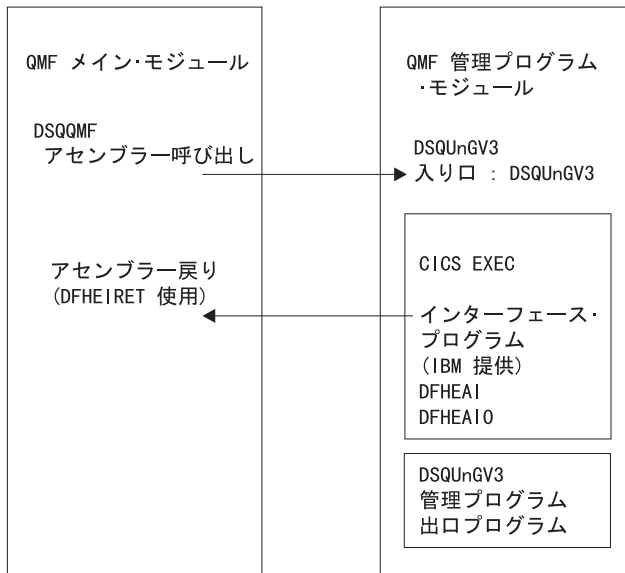


図 131. QMF と管理プログラム出口ルーチンを対話させる CICS 処理

管理プログラム出口ルーチンは、QMF メイン・プログラムと同じプログラム・レベルで実行されます。

管理プログラム出口ルーチンの入り口点は、DSQUn GV3 です。QMF は、管理プログラム出口ルーチンを呼び出すとき、必ず、EXEC CICS LOAD コマンドの結果として CICS が戻すアドレスにブランチします。

ロードが失敗したり、モジュールが 31 ビット・アドレッシング・モードをサポートしていない場合、QMF は、警告メッセージを出し、管理プログラム出口ルーチンを使用不可にし、さらに管理プログラムなしでセッションを継続します。このモジュールのアセンブルとリンク・エディットについては、478ページの

管理プログラム出口ルーチンを使用した QMF リソースの制御

ジの『CICS でのユーザー管理プログラム出口ルーチンの変換、アセンブル、リンク・エディット』で説明しています。

QMF が管理プログラム出口ルーチンを呼び出す方法と時期

QMF は、管理プログラム出口ルーチンに対して標準のアセンブラー CALL ステートメントを発行します。機能呼び出し という用語は、QMF セッション中にこれらの CALL ステートメントが出される時点を述べています。

QMF が管理プログラムを呼び出す時点

管理プログラム出口ルーチンに対する機能呼び出しは、特定のタイプの QMF アクティビティーに先行するか、もしくはその後続くかのいずれかです。たとえば、QMF はコマンドを実行する前または後に制御を管理プログラム出口ルーチンに渡します。

QMF は管理プログラムを呼び出すと、常に DSQUn GVx という名前の入り口点にブランチします。したがって、入り口点を使用して、出口のタイプを判別することはできません。その代わりに、制御ブロック・フィールド GOVFNCT を使用します。その値は、出口のタイプを識別する正の整数です。

• QMF セッションの開始時と終了時

QMF は、QMF セッションの初期化の間に管理プログラム出口ルーチンを呼び出しますが、これは管理プログラム出口ルーチンがユーザーの仮想記憶にロードされた後です。管理プログラムは、QMF の Q.RESOURCE_VIEW 照会から渡される行に含まれているリソース制御情報を使用し、セッション用に自身を初期化します。

管理プログラム出口ルーチンは、セッションが終了する直前に呼び出されます。このとき、ユーザー・セッションのアクティビティーを中止するために必要なすべてを実行することができます。たとえば、仮想記憶域を解放することができます。

• データベースへの新規接続が行われた後

ユーザーが CONNECT コマンドを出すと、Q.PROFILES 表およびリソース制御表が再度、初期化されます。管理プログラムが、呼ばれますが、これは異なる CONNECT ID が使用された場合にリソース制御値が変更される場合があるからです。完了していないデータベース操作はすべて、接続が行われる前に完了します。

管理プログラム出口ルーチンは、データベースへの接続を取り消すことはできませんが、ユーザー独自のルーチンでステートメントを書いてユーザーのセッションを次のアクティビティーで取り消すことができます。これは、管

理プログラムに渡されたリソース情報がそのユーザーが QMF の使用を許可されていないことを示している場合です。

• コマンド実行の前後

QMF は、すべてのコマンドの実行前と実行後に管理プログラムを呼び出します。コマンド完了に関する呼び出しの前に、コマンド開始に関する呼び出しが複数存在する可能性があります。たとえば、RUN PROC コマンドは、RUN QUERY コマンドがプロシージャに組み込まれている場合には、2 つの「開始コマンド」呼び出しと 2 つの「終了コマンド」呼び出しを生じます。

• データベース・アクティビティーの開始前および終了後

QMF は PREPARE、OPEN および FETCH などのさまざまなデータベース操作を開始する直前に管理プログラムを呼び出します。QMF はまた、データベース・アクティビティーの完了時にも管理プログラムを呼び出します。

QMF がデータを検索する場合、最小サイズ 4K のバッファーに収まる最大行数に合わせます。QMF は、最初の行を検索してバッファーに入れるときに 1 回、バッファーが全部埋まったか表の終わりに達したかいずれか先に起こった時点で 1 回、管理プログラムを呼び出します。

また、SQL、QBE、またはプロンプトが出された照会が RUN QUERY を使用して実行依頼されたとき、あるいは QMF がコマンドで開始された照会を実行しているときにも、QMF は管理プログラムを呼び出します。たとえば、SAVE DATA コマンドは DELETE、CREATE、および INSERT 照会を生じる場合があります。管理プログラムは、これらの操作の前後に呼び出されます。コマンドを入力したときに未完了のデータ・オブジェクトがあった場合、データ・オブジェクトの完了処理の間に、データベース・アクティビティーに対する管理プログラムの呼び出しが行われる場合もあります。データ・オブジェクトの完了処理に関連する問題の取り扱いの詳細については、538ページの『パフォーマンスの問題の解決』を参照してください。

以下の QMF コマンドは、常にデータベース・アクティビティーを強制します。

- DISPLAY 表コマンド
- 表編集プログラムの EDIT TABLE コマンド
- 表の ERASE コマンド
- EXPORT TABLE コマンド
- 表への IMPORT コマンド
- 表または視点の PRINT コマンド
- 照会の RUN コマンド

管理プログラム出口ルーチンを使用した QMF リソースの制御

- SAVE DATA コマンド (これは暗黙的 CREATE TABLE 照会を強制)
- 報告書が表示されるときに、データ取り出しをもたらすスクロール・コマンド
- データ検索操作 (取り出し操作)
- **ユーザー選択の前後**

セッション中のさまざまな時点で、QMF はユーザーが決定を行うのを待ちます。QMF が待ちに費やす時間は **考慮時間** と呼ばれます。

QMF は、考慮時間をとる必要のある操作を実行する前に管理プログラムを呼び出します。これは、ユーザー入力の選択でパネルを表示するような場合です。ユーザーが応答を入力し、考慮時間が終了するとすぐに、QMF は管理プログラムを呼び出します。

以下のいずれのアクティビティーが行われた場合に考慮時間をとる必要が生じます。

 - コマンドとコマンドの実行の間での QMF パネルの表示
 - ヘルプ・パネルの表示
 - 確認プロンプト・パネルの表示。たとえば、ユーザーが SAVE コマンドでオブジェクトを置き換えて、何かを消去するような場合。
 - コマンド・プロンプト・パネルの表示。たとえば、ユーザーが DISPLAY ? を入力した場合。
 - LIST プロンプト・パネルの表示
 - ICU および EXTRACT パネルの表示
 - EDIT PROC および EDIT QUERY 機能の実行
- **異常終了の開始**

QMF は、異常終了を開始する直前に管理プログラムを呼び出します。管理プログラムは、異常終了処理の開始前に、必要な終結処置を実行することができます。これらのアクションは、セッション終了の間のものと似ている場合があります。

IBM 提供の管理プログラム出口ルーチンでは、QMF は DXEGOVA 制御ブロックの GOVFNCT フィールドを使用して、機能呼び出しのタイプについての情報を渡します。この制御ブロックのフィールドについては、459ページの表56で説明しています。機能呼び出しの各タイプには、GOVFNCT フィールドの特別な値があります。これらの値は、457ページの図137に示されています。

管理プログラム出口ルーチンへ入ったときに起こること

QMF は、入り口点 DSQUnGV1 または DSQUn GV3 のアドレスへブランチすることにより、管理プログラム出口ルーチンを呼び出します。

CICS 入り口点 DSQUnGV3 へのブランチ: CICS の管理プログラム出口ルーチンへ入るには、以下のように、標準の CICS リンケージ規則に従います。

- レジスター 1 には、CICS 提供のマクロ DFHEIENT および DFHEIRET による処理に適している CICS パラメーター・リストが入っています。図132 は、管理プログラムに対する呼び出し時のレジスター 1 の内容を示します。DFHEIBLK は CICS 連絡域のアドレスです。DFHCOMMA には、2 つのポインターが入っています。1 つは、DXEXCBA 制御ブロックを指すポインターで、もう 1 つは DXEGOVA 制御ブロックを指すポインターです。

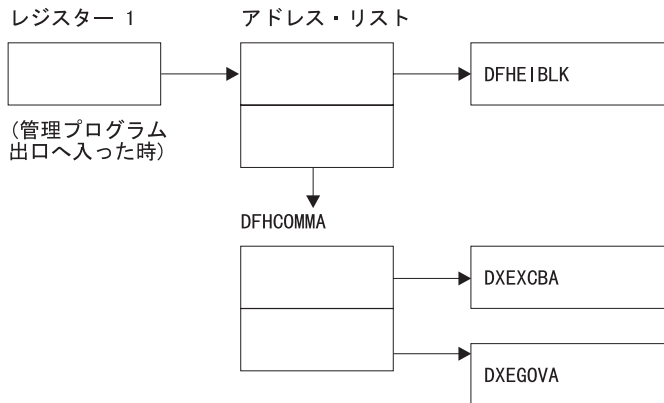


図 132. 管理プログラム出口ルーチンの呼び出し時のレジスター 1 の内容

- レジスター 13 には、CICS DSECT (DFHEISTG) で記述されているとおりの、標準の CICS 作業記憶域のアドレスが入っています。
- レジスター 14 には戻りアドレスが入っています。

管理プログラムは、QMF と同じプログラム・レベルで実行されるため、環境を変更する EXEC CICS コマンド (たとえば、CICS HANDLE CONDITION) を使用する場合には、注意してください。CICS HANDLE CONDITION を使用する必要がある場合は、EXEC CICS PUSH および EXEC CICS POP を使用してこれらを保管し、復元します。

454ページの図133 に示すものと類似のコードで、管理プログラムを開始します。

管理プログラム出口ルーチンを使用した QMF リソースの制御

```

DSQUEGV3 TITLE 'QMF GOVERNOR EXIT ROUTINE'
DFHEISTG DSECT
DSQUEGV3 DFHEIENT CODEREG=(12),DATAREG=(13),EIBREG=(10)
          B      FDENTRY          BRANCH AROUND CONSTANTS
*
MODNAME  DC    C'DSQUEGV3'      MODULE NAME
          DC    C' '
          DC    C'&SYSDATE '    DATE OF ASSEMBLY
          DC    C'&SYSTEMTIME '  TIME OF ASSEMBLY
          DS    OH
*
FDENTRY  DS    OH
          L     R01,4(R01)      GET ADDRESS OF DFHCOMMA
          L     XCBPTR,8(R01)   GET ADDRESS OF QMF EXIT CTL BLK
          L     GOVPTR,12(R01)  GET ADDRESS OF QMF GOV CTL BLK
          USING DXEXCBA,XCBPTR
          USING DXEGOVA,GOVPTR
          LA    WORKPTR,GOVUSERS GET ADDRESS OF GOVERNOR WORK AREA
          USING WORK,WORKPTR
*
          :
          :
          :
          GOVPTR EQU R03        PTR TO DXEGOV CONTROL BLOCK
          XCBPTR EQU R02        PTR TO DXEXCB CONTROL BLOCK
          WORKPTR EQU R04       PTR TO GOVERNOR SCRATCH PAD AREA

```

図 133. 管理プログラムの開始時点でのサンプル・コード (CICS の場合)

図133 のコードは、最初に、仮想記憶域のダンプでの目印としての役割を果たす定数ブロックのあたりにブランチします。定数は、入り口点および QMF の適合バージョンを指しています。また、コードがアSEMBルされた日時も示しています。

コードは、プログラム DXEXCB、DXEGOV、および GOVUSERS という名前のスクラッチパッド域の基底レジスターを設定します。スクラッチパッド域は、管理プログラムへの呼び出しと呼び出しの間、QMF により保存されています。WORK という名の DSECT が、IBM 提供の管理プログラムのコードにあるこのスクラッチパッド域を記述しています。

処理が完了すると、管理プログラムは CICS マクロ DFHEIRET で指定されているとおり、標準の CICS 戻りを使用して、QMF に制御を戻します。

考慮事項: コマンド EXEC CICS RETURN は使用しないでください。これは、QMF リソースを解放しないで QMF を終了します。

管理プログラムは、図134 と類似したコードで終了します。


```

:
*
XR    R15,R15          ZERO RETURN CODE
      DFHEIRET RCREG=15
*

```

図 134. 管理プログラムの終了コード

TSO、SRPI、APPC、またはネイティブ OS/390 の入り口点 DSQUnGV1 へのブランチ: 標準の IBM リンケージ規則を使用して、DSQUEGV1 という名前の入り口点を呼び出します。これは、どのタイプの呼び出しが行われる場合でも同じです。呼び出し時に、QMF は以下の情報を渡します。

- レジスタ 1 には、パラメーター・リストのアドレスが入っています。パラメーター・リストには 2 つのフルワードのアドレスが含まれています。1 つは制御ブロック DXEXCBA のアドレスであり、もう 1 つは制御ブロック DXEGOVA のアドレスです。

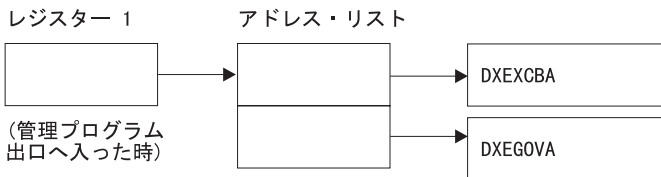


図 135. 管理プログラム出口ルーチンの呼び出し時のレジスタ 1 の内容

- レジスタ 13 には、SAVE 域のアドレスが入っています。
- レジスタ 14 には、呼び出しからの戻りアドレスが入っています。
- レジスタ 15 には、入り口点のアドレスが入っています。これは常に、DSQUEGV1 です。

管理プログラムが呼び出された後、456ページの図136 に示すようなコードで、管理プログラムが開始されることがあります。サンプル・コードは IBM 提供の管理プログラムからのものです。

管理プログラム出口ルーチンを使用した QMF リソースの制御

```

DSQUEGV1 CSECT
    USING *,R15
    B    FENTRY          BRANCH AROUND CONSTANTS
    DC   C'DSQUEGV1'    MODULE NAME
    DC   C' '
    DC   C'&SYSDATE '    DATE OF ASSEMBLY
    DC   C'&SYSTIME '    TIME OF ASSEMBLY
    DS   0H

*
FENTRY  STM  R14,R12,12(R13)  SAVE THE REGISTERS
        BALR R12,0           INITIALIZE BASE REGISTER
        DROP R15
        LA  R02,MAINSV       CHAIN THE SAVE AREAS
        ST  R02,8(R13)
        ST  R13,MAINSV+4
        LR  R13,R02

*
        L   R01,4(R01)       GET ADDRESS OF DFHCOMMA
        L   XCBPTR,0(R01)    GET ADDRESS OF QMF EXIT CTL BLK
        L   GOVPTR,4(R01)    GET ADDRESS OF QMF GOV CTL BLK
        USING DXEXCBA,XCBPTR
        USING DXEGOVA,GOVPTR
        LA  WORKPTR,GOVUSERS SCRATCH PAD ADDRESS
        USING WORK,WORKPTR

:
:
MAINSV  DS   18F           SAVE AREA
XCBPTR  EQU  R02           PTR TO DXEXCBA CONTROL BLOCK
GOVPTR  EQU  R03           PTR TO DXEGOVA CONTROL BLOCK
WORKPTR EQU  R04           PTR TO SCRATCH_PAD AREA
    
```

図 136. 管理プログラムの開始時点でのサンプル・コード (TSO、SRPI、APPC、またはネイティブ OS/390 の場合)

図136 のコードは、最初に、仮想記憶域のダンプでの目印としての役割を果たす定数ブロックのあたりにブランチします。定数は、入り口点および QMF の適合バージョンを指しています。また、コードがアセンブルされた日時も示しています。

コードは、プログラム DXEXCBA、DXEGOV、および GOVUSERS という名前のスクラッチパッド域の基底レジスターを設定します。スクラッチパッド域は、管理プログラムへの呼び出しと呼び出しの間、QMF により保存されています。WORK という名の DSECT が、IBM 提供の管理プログラムのコードにあるこのスクラッチパッド域を記述しています。

呼び出しを処理した後、管理プログラムは制御を標準的な方法で QMF に戻します。すなわち、標準のエピローグとプロローグを使用する必要があります。IBM 提供の管理プログラムでは、次のコードがこれを行います。

管理プログラム出口ルーチンを使用した QMF リソースの制御

L	R13,4(R13)	RESTORE CALLER'S SAVE AREA ADDRESS
LM	R14,R12,12(R13)	RESTORE CALLER'S REGISTERS
XR	R15,R15	ZERO RETURN CODE
BR	R14	RETURN TO CALLER

機能呼び出しのアドレス可能性の確立

QMF が管理プログラムを呼び出すときには、常に DSQU *n* GV1 または DSQU *n* GV3 という名前の入り口点にブランチするため、入り口点を使って機能呼び出しのタイプを判別することはできません。代わりに、DXEGOVA 制御ブロックの GOVFUNCT フィールドを使用してください。

IBM 提供の管理プログラム出口ルーチンの GOVFUNCT は、機能呼び出しのタイプを識別する文字値を持っています。この文字値は、順に 1 から 10 の 1 バイトの 2 進整数の値を持ちます。たとえば、QMF セッションの開始に関する機能呼び出しでは、GOVFUNCT の値は GOVINIT で、これは X'1' の数値と等価です。

機能呼び出しの各タイプのための文字値および数値の両方を、図137 に示します。(それぞれの機能呼び出しで生じるアクティビティーについての詳細情報が必要な場合は、450ページの『QMF が管理プログラムを呼び出す時点』を参照してください。)

GOVINIT	EQU	1	-----	INITIALIZATION OF SESSION
GOVTERM	EQU	2	-----	TERMINATION OF SESSION
GOVSCMD	EQU	3	-----	START COMMAND
GOVECMD	EQU	4	-----	END COMMAND
GOVCONN	EQU	5	-----	CONNECT COMMAND
GOVSDBAS	EQU	6	-----	START DATA BASE
GOVEDBAS	EQU	7	-----	END DATA BASE
GOVSACTV	EQU	8	-----	SUSPEND QMF ACTIVITY
GOVRACTV	EQU	9	-----	RESUME QMF ACTIVITY
GOVABEND	EQU	10	-----	QMF ABEND OPERATION

図 137. DXEGOVA の GOVFUNCT フィールドの文字値と数値

CICS で実行している場合は、GOVABEND は呼び出されません。

ユーザー独自の出口ルーチンのパフォーマンスを向上させるには、IBM 提供の管理プログラムが使用している規則に従い、GOVFUNCT の値をブランチ・テーブルを使用して 2 進数の値に置き換えるようにします。QMF は、それぞれの機能呼び出しごとに、ブランチ・テーブルを使用してブランチすべきアドレスを見つけます。

管理プログラム出口ルーチンを使用した QMF リソースの制御

図138 は、IBM 提供の管理プログラムの場合の、ブランチ・アドレスを識別するコードの例を示しています。

	XR	R07,R07	ZERO REGISTER 7
	IC	R07,GOVFUNCT	IDENTIFY EXIT TYPE
	SLL	R07,2	DETERMINE BRANCH TABLE OFFSET
	LA	R15,FUNBTAB(R07)	GET BRANCH TABLE ADDRESS
	L	R15,0(R15)	GET BRANCHING ADDRESS
	BALR	R14,R15	BRANCH TO THE APPROPRIATE CODE
		. . .	
		. . .	
		. . .	
		. . .	
FUNBTAB	DS	0F	
	DC	A(BYPASS)	VALUE "0" - UNUSED
	DC	A(INIT)	VALUE "1" - QMF INITIALIZATION
		. . .	
		. . .	
		. . .	
	DC	A(SUSPEND)	VALUE "10" - QMF ABEND IN PROCESS

図 138. 機能呼び出しのタイプの識別と該当するアドレスへのブランチ

リソース制御情報を管理プログラム出口に渡す

まだこれを行っていない場合には、この後に続く節をお読みください。リソース制御情報を管理プログラムが使用できるフォーマットで設定する方法を説明しています。

- 437ページの『管理プログラム出口ルーチンのリソースの制御方法』
- 441ページの『ユーザー独自のリソース限界の定義』

QMF は、DXEGOVA および DXEXCBA という名前の 2 つの制御ブロックを使用して、リソース制御情報を渡します。これらは、462ページの図139 および 472ページの図141 に示されています。これらのアドレスが、機能呼び出しのたびごとに管理プログラムに渡されます。DSECT DXEXCBA (DXEXCBA として出荷される) および DSECT DXEGOVA (DXEGOVA として出荷される) は、SDSQUSRE MACLIB 内にあります。アセンブラー COPY ステートメントを使用して、これらの DSECT をユーザーのプログラムに組み込みます。

DXEGOVA 制御ブロックの構造

DXEGOVA 制御ブロックは、ユーザーのリソースの制約に関する情報を管理プログラム出口ルーチンに渡します。この情報は、RESOURCE_VIEW と呼ばれるリソース制御の視点にあります。この視点の使用方法に関する詳細については、438ページの『管理プログラムがリソース限界を知る方法』を参照してください。

管理プログラム出口ルーチンを使用した QMF リソースの制御

表56 は、DXEGOVA 制御ブロックの各フィールド名とそのデータ・タイプ、および目的について示しています。それぞれのデータ・タイプは、DSECT のフィールドを定義する DS ステートメントに現れる順番にリストされています。たとえば、GOVOROWS フルワードの場合、F という文字はこのフィールドにフルワード整数が含まれていることを表しています。GOVOROWS の DS ステートメントは、GOVOROWS DS F として表されます。

制御ブロックのレイアウトおよびそれに含まれる情報は、QMF サポートに関してはすべての操作環境において同じです。したがって、制御ブロックに示される一部の情報は VM/ESA[®] または VSE/ESA[®] 操作環境でだけ使用されるため、OS/390 環境の QMF には適用されない場合があります。

表 56. 管理プログラムへの DXEGOVA インターフェース制御ブロックのフィールド

フィールド	データ・タイプ	目的
GOVCADDR	A	アクティビティーを取り消すためにブランチするアドレスを含む。このフィールドで使用するコードは、475ページの『ユーザー・アクティビティーの取り消し』に表されている。
GOVFNCT	XL1	機能呼び出しのタイプを示す。取り得る値は次のとおり。 <ul style="list-style-type: none"> • GOVINIT (セッションの初期化); GOVTERM (セッション終了) • GOVSCMD (開始コマンド); GOVECMD (終了コマンド) • GOVCONN (接続コマンド) • GOVSDBAS (データベース検索操作の開始); GOVEDBAS (データベース検索操作の終了) • GOVSACTV (ユーザー考慮時間による QMF アクティビティーの延期); GOVRACTV (QMF アクティビティーの再開) • GOVABEND (異常終了の開始) <p>このフィールドで使用するコードは、457ページの『機能呼び出しのアドレス可能性の確立』に記載されています。</p>
GOVGROUP	CL16	ユーザーのリソース・グループの名前を含む。この値は、QMF のセッション中、変わらない。リソース・グループに関する詳細については 438ページの『管理プログラムがリソース限界を知る方法』をお読みください。
GOVNAME	CL8	制御ブロックの名前を含む (DXEGOVA)。この値はセッション中は変更されない。これは、仮想記憶域のダンプでの目印としての役割を果たす。
GOVOROWS	F	リソース制御表のユーザーのリソース・グループの行数を含む。この値は、セッション中も変わらず、ゼロも可能。

管理プログラム出口ルーチンを使用した QMF リソースの制御

表 56. 管理プログラムへの DXEGOVA インターフェース制御ブロックのフィールド (続き)

フィールド	データ・タイプ	目的
GOVRESCT	10XL128	<p>リソース制御表からの情報を含む。この情報は、10 個の連続する記憶域のブロックに分割され、これは DSECT GOVRESCT のように構造化される。1 つのブロックには、QMF リソース制御表のユーザー・リソース・グループの 1 つの行に関する情報を含む。</p> <ul style="list-style-type: none">リソース・グループが 10 行未満の場合は、未使用のブロックはフィールドの終わりにある。リソース・グループが 10 行を超える場合は、(GOVRESCT DSECT の) GOVNEXTR という名前のフィールドを使用し、追加行にアクセスする。 <p>すべてのブロックは、463ページの『リソース制御表のアドレッシング』で説明のように、チェーンの一部。このフィールドの値はセッション中、変わらない。</p>

管理プログラム出口ルーチンを使用した QMF リソースの制御

表 56. 管理プログラムへの DXEGOVA インターフェース制御ブロックのフィールド (続き)

フィールド	データ・タイプ	目的
GOVRESCT	DSECT	リソース制御表のユーザーの 1 つの行に関する情報を含む記憶域ブロックを記述する。このようなブロックはすべて、463 ページの『リソース制御表のアドレッシング』で説明しているチェーンで互いにリンクされている。以下のフィールドがブロック内にある。 GOVOPTN(CL16) リソース制御表の RESOURCE_OPTION 列の値を含む。チェーンの中のブロックは、このフィールドの内容でアルファベット順に並んでいる。 GOVNULLI(H) INTVAL 列のヌル標識 GOVINTVL(F) INTVAL 列の値 GOVNULLF(H) FLOATVAL 列のヌル標識 GOVFLOAT(D) FLOATVAL 列の値 GOVNULLC(H) CHARVAL 列のヌル標識 GOVCHLEN(H) CHARVAL 列のデータの長さ GOVCHAR(CL80) CHARVAL 列の値 GOVNEXTR(A) 次のリソース表の列のデータ・ブロックを指す。この行が最終行の場合は、ゼロが入っている。 対応する列値がヌルでない場合は、構造のヌル標識はすべてゼロ。列値がヌルの場合は、標識はゼロではない。
GOVSQLCA	A	SQL 連絡域 (SQLCA) のアドレス。これは、リソース制御視点 (Q.RESOURCE_VIEW) の SQL SELECT 照会に関する情報を保持する。
GOVSQLRC	F	リソース制御視点 (Q.RESOURCE_VIEW) の SQL SELECT 照会からのリターン・コード。これが非ゼロの場合は、その照会は失敗し、管理プログラムに渡される行はない。
GOVUSERS	CL2048	スクラッチパッド域。セッション呼び出し間で保存される。QMF はこの値を変更しない。

462 ページの図 139 は、DXEGOVA 制御ブロックの構造を示す。

管理プログラム出口ルーチンを使用した QMF リソースの制御

```

***** 00001000
*
* CONTROL BLOCK NAME: DXEGOVA * 00002000
*
* FUNCTION: * 00003000
*
* THIS IS THE INTERFACE CONTROL BLOCK BETWEEN QMF AND * 00004000
* THE GOVERNOR EXIT ROUTINE. * 00005000
*
* STATUS: VERSION 7 RELEASE 1 LEVEL 0 * 00006000
*
* INNER CONTROL BLOCKS: NONE * 00007000
*
* CHANGE ACTIVITY: NA * 00008000
*
* CHANGE DATE: NA * 00009000
*
***** 00010000
*
DXEGOVA DSECT 00011000
DS 0D 00012000
GOVNAME DS CL8 -- CONTROL BLOCK IDENTIFICATION 00013000
SPACE 00014000
GOVEXCTL DS XL72 -- EXIT CONTROL 00015000
ORG GOVEXCTL 00016000
GOVFUNCT DS XL1 ----- FUNCTION CODE 00017000
GOVINIT EQU 1 ----- INITIALIZATION OF SESSION 00018000
GOVTERM EQU 2 ----- TERMINATION OF SESSION 00019000
GOVSCMD EQU 3 ----- START COMMAND 00020000
GOVECMD EQU 4 ----- END COMMAND 00021000
GOVCONN EQU 5 ----- CONNECT COMMAND 00022000
GOVSDBAS EQU 6 ----- START DATA BASE 00023000
GOVEDBAS EQU 7 ----- END DATA BASE 00024000
GOVSACTV EQU 8 ----- SUSPEND QMF ACTIVITY 00025000
GOVRACTV EQU 9 ----- RESUME QMF ACTIVITY 00026000
GOVABEND EQU 10 ----- QMF ABEND OPERATION 00027000
GOVPAD10 DS CL7 ----- RESERVED FIELD 00028000
SPACE 00029000
GOVCADDR DS A ----- ADDR TO BRANCH TO FOR CANCELLATION 00030000
SPACE 00031000
GOVOROWS DS F ----- NUMBER OF OPTION ROWS RETRIEVED 00032000
SPACE 00033000
GOVSQLRC DS F ----- RESOURCE TABLE SQL RETURN CODE 00034000
SPACE 00035000
GOVSQLCA DS A ----- ADDRESS OF SQLCA FOR ERROR CONDITION 00036000
SPACE 00037000
GOVGROUP DS CL16 ----- GROUP NAME 00038000
GOVPAD20 DS CL32 ----- RESERVED FIELD 00039000

```

図 139. DXEGOVA 制御ブロック (1/2)

管理プログラム出口ルーチンを使用した QMF リソースの制御

	SPACE			00049000
GOVUCTL	DS	XL304	-- USER CONTROL AREA	00050000
	ORG	GOVUCTL		00051000
GOVUSERS	DS	CL2048	----- USER SCRATCH PAD AREA	00052000
GOVPAD30	DS	CL48	----- RESERVED FIELD	00053000
	SPACE			00054000
	DS	0D		00055000
GOVRESC	DS	10XL128	-- RESOURCE CONTROL TABLE	00056000
	ORG	GOVRESC		00057000
GOVRESCT	DSECT		-- RESOURCE CONTROL TABLE MAPPING	00058000
	DS	0D		00059000
GOVOPTN	DS	CL16	----- RESOURCE OPTION	00060000
GOVNULLI	DS	H	----- INTEGER NULL INDICATOR	00061000
GOVPAD40	DS	CL2	----- RESERVED FIELD	00062000
GOVINTVL	DS	F	----- INTEGER OPTION REPRESENTATION	00063000
GOVNULLF	DS	H	----- FLOATING POINT NULL INDICATOR	00064000
GOVPAD50	DS	CL6	----- RESERVED FIELD	00065000
GOVFLOAT	DS	D	----- FLOATING POINT OPTION REPRESENTATION	00066000
GOVNULLC	DS	H	----- CHARACTER NULL INDICATOR	00067000
GOVCHLEN	DS	H	----- LENGTH OF THE CHARACTER OPTION	00068000
GOVCHAR	DS	CL80	----- CHARACTER OPTION REPRESENTATION	00069000
GOVNEXTR	DS	A	----- POINTER TO NEXT RESOURCE CONTROL ROW	00070000

図 139. DXEGOVA 制御ブロック (2/2)

リソース制御表のアドレッシング

DXEGOVA 制御ブロックの GOVGROUP フィールドは、リソース制御表で定義されている視点である Q.RESOURCE_VIEW の RESOURCE_GROUP 列の値を持っています。

ユーザーのリソース・オプションに関するすべての情報が、ブロックに保管されます。モニターすることになっているユーザーのリソース・オプションごとに、1 つのブロックがあります。

最初のブロックは最初のリソース・オプションを定義し、DXEGOVA 制御ブロックに DSECT GOVRESCT として保管されます。この DSECT は、図139 の最後の部分に示されています。この DSECT のアドレスは、DXEGOVA の GOVRESC フィールドに定義されています。ユーザー独自のルーチンで、GOVRESCT DSECT を使用して GOVRESC フィールドに対するアドレス可能性を確立することができます。

DSECT の負のハーフワード整数は、Q.RESOURCE_VIEW に INTVAL、CHARVAL、または FLOATVAL として入力されたヌル値を表しています。ゼロまたは正のハーフワード整数は、Q.RESOURCE_VIEW の列の値を示しています。

管理プログラム出口ルーチンを使用した QMF リソースの制御

リソース制御情報を保管するブロックはチェーンを形成しています。このチェーンの中では、1 つのブロックの中のポインターが、チェーンの中の次のブロック (次のリソース・オプション) の始まりを指しています。たとえば、図139 の GOVRESCT DSECT の GOVNEXTR DS ステートメントは、リソース制御情報のチェーンの中の次のブロックのアドレスを含んでいます。チェーン内の各ブロックには GOVNEXTR DS ステートメントがあります。最終ブロックでは、GOVNEXTR DS ステートメントはゼロを含んでおり、ユーザーのリソース制御情報の終わりを示しています。

465ページの図140 は、リソース制御情報のブロックを処理する IBM 提供の管理プログラムのコードの一部を示したものです。このコードでは、GOVRESCT が GOVRESCT DSECT を指しています。

管理プログラム出口ルーチンを使用した QMF リソースの制御

```

L      R08,GOVOROWS      GET NUMBER OF RESOURCE TABLE ROWS
LTR    R08,R08           ANY RESOURCE TABLE ROWS?
BZ     ENDRESST          NO, SKIP RESOURCE INITIALIZATION
LA     R05,GOVRESC       GET ADDRESS OF 1ST RESOURCE ROW
USING  GOVRESC,R05      BASE RESOURCE RECORD ENTRY
LOOK4RES DS  0H         MAIN LOOP THRU RESOURCE ROWS
LTR    R05,R05          ANY MORE RESOURCE TABLE ROWS?
BZ     ENDRESST          NO, END RESOURCE INITIALIZATION
:
:
:
L      R05,GOVNEXTR      GET ADDRESS ON NEXT RESOURCE ROW
B      LOOK4RES          BEGIN NEXT ITERATION
ENDRESST DS  0H         -- BRANCH HERE WHEN FINISHED READING ALL ROWS
:
:
:
:
DXEGOVA DSECT
:
:
:
:
GOVRESC DS  10XL128      -- RESOURCE CONTROL TABLE
      ORG  GOVRESC
GOVRESCT DSECT          -- DSECT FOR RESOURCE ROW
:
:
:
:
GOVNEXTR DS  A          -- POINTER TO NEXT RESOURCE ROW
:
:
:
:

```

図 140. リソースの初期化

DXEXCBA 制御ブロックの構造

DXEXCBA 制御ブロックは、管理プログラムに入ると、QMF セッションの状態に関する情報を管理プログラム出口ルーチンに渡します。管理プログラムは、この情報とリソースの限界に関する情報 (DXEGOVA に含まれている) とを合わせて、リソース限界を超過する時点を判断し、ユーザーのアクティビティを取り消す時点を判別します。

たとえば、ユーザー JONES が EDIT TABLE コマンドを使用できないように、リソース・オプションを定義することができます。定義した後、DXEXCBA 制御ブロックの XCBQRYP フィールドに EDIT TABLE コマンド

管理プログラム出口ルーチンを使用した QMF リソースの制御

が指定されている場合、管理プログラム出口が QMF 取り消しサービスを呼び出してコマンドを取り消すことができるように、管理プログラム出口ルーチンを作成することができます。

表57 は、制御ブロックの各フィールド名とそのデータ・タイプ、および目的について示しています。それぞれのデータ・タイプは、DSECT のフィールドを定義する DS ステートメントに現れる順番にリストされています。

制御ブロックのレイアウトおよびそれに含まれる情報は、QMF サポートに関してはすべての操作環境において同じです。したがって、制御ブロックに示される一部の情報は VM/ESA または VSE/ESA 操作環境でだけ使用されるため、OS/390 環境の QMF には適用されない場合があります。

表 57. 管理プログラムへの DXEXCBA インターフェース制御ブロックのフィールド

フィールド	データ・タイプ	目的
XCBACTIV	CL1	データベース・アクティビティーの現行タイプを示している。行が現行データ・オブジェクトについて検索される場合にのみ適用される。行が IMPORT コマンドについて検索される場合には、適用されない。取り得る値は次のとおり。 1 OPEN 実行中 2 FETCH 実行中 3 PREPARE 実行中 4 DESCRIBE 実行中 5 CLOSE 実行中 このフィールドは、データベース・アクティビティーのタイプが変更されるごとく変わる。管理プログラムがタイマーの結果として、非同期的に制御を受け取ったときに、この値を使用することができる。
XCBAIACT	CL1	現行のコマンドが対話式に実行しているか否かを示す。 1 対話式 0 非対話式 (バッチ) 対話式コマンドは、プロンプトと状況パネルを表示する。このフィールドは、コマンド開始の任意の機能呼び出しで値を変更する。コマンドが完了するとゼロにリセットされる。
XCBAUTH	CL8	ユーザーの SQL 許可 ID を含んでいる。

管理プログラム出口ルーチンを使用した QMF リソースの制御

表 57. 管理プログラムへの DXEXCBA インターフェース制御ブロックのフィールド (続き)

フィールド	データ・タイプ	目的
XCBCAN	CL1	ユーザーまたは管理プログラムが現行コマンドの取り消しを要求したかどうかを示す。取り消しが要求された場合は、このフィールドは 1 に設定される。ゼロは、取り消しの要求がなかったことを示す。取り消しが要求された時点で、値が変更される。このフィールドは、コマンド終了の機能呼び出しの前にゼロにリセットされる。
XCBCLOC	CL18	現行ロケーションの名前が含まれる。
XCBCMDL	F	実行するコマンドを含むストリングの長さが含まれる。これは、XCBCMDP フィールドにより指されたストリングである。XCBCMDL が値を変更すると、このフィールドの値が変わる。
XCBCMDP	A	<p>実行するコマンドを含むストリングを指す。このフィールドは、コマンド開始の機能呼び出しの前のある点で QMF がコマンドの妥当性検査を行うときにリセットされる。</p> <p>このフィールドは、コマンドが完了したときに、機能呼び出しの前にゼロにリセットされる。コマンド同義語が実行中の場合は、ここに現れる。</p>
XCBCVERB	CL18	現行コマンドの動詞を保持している。このフィールドは、コマンドの開始の機能呼び出しで値を変更する。呼び出し間では、値は変わらない。
XCBDBMG	CL1	データベース・マネージャーを識別する。この値は、DB2 (VM または VSE 版) については 1 に、DB2 (OS/390 版) については 2 に設定される。
XCBEMODE	CL1	<p>QMF セッションの現行モードを示している。</p> <p>1 対話式</p> <p>2 非対話式 (バッチまたはサーバー)</p> <p>この値はセッション中は変更されない。非対話式セッションの開始の詳細については、221ページの『DSQSMODE (対話式または非対話式 QMF セッションの指定)』を参照。</p>
XCBERRET	F	デフォルトの取り消しメッセージで使用されるリターン・コードが含まれる。このメッセージの詳細については、476ページの『取り消されたアクティビティに対するメッセージ』を参照。
XCBINCI (ISPF のみ)	CL1	現行コマンドがコマンド・インターフェースを介して実行されるかどうかについて示している。実行される場合はこのフィールドは 1 に、されない場合は 2 に設定される。コマンド・インターフェースの詳細については、QMF アプリケーション開発の手引きを参照。
XCBINPRC	CL1	コマンドがどこで実行されているかを管理プログラムに伝える。1 はプロシージャーまたは LIST コマンド内で実行されていることを、0 は別の方法で実行されていることを示す。

管理プログラム出口ルーチンを使用した QMF リソースの制御

表 57. 管理プログラムへの DXEXCBA インターフェース制御ブロックのフィールド (続き)

フィールド	データ・タイプ	目的
XCBKPARM	CL1	DSQSDBCS プログラム・パラメーターの設定方法を管理プログラムに伝える。この値はセッション中、変わらない。取り得る値は、0 がラテン文字、1 が 2 バイト文字セット (DBCS) データ。このパラメーターの詳細については、230ページの『DSQSDBCS (2 バイト文字セット・データの印刷の設定)』を参照。
XCBLOGM	CL1	QMF が QMF トレース・データ・セットのメッセージをログに記録するかどうかを示す。メッセージをログに記録する場合は 1 を、ログに記録しない場合は 0 を使用する。メッセージ・ログについては、476ページの『取り消されたアクティビティーに対するメッセージ』に説明がある。QMF トレース機能の使い方については、544ページの『QMF トレース機能の使用』に説明がある。
XCBMGTXT	CL78	メッセージのテキストが入っている。メッセージは、QMF トレース・データに記録されるか、画面に表示、またはその両方が行われる。このフィールドの使用法の詳細については、476ページの『取り消されたアクティビティーに対するメッセージ』を参照。
XCBMSGNO (ISPF のみ)	CL8	ISPF メッセージ定義のメッセージ ID を含んでいる。これは、メッセージを DSQDEBUG データ・セットにログを記録するか、画面に表示するか、もしくはその両方を行うために使用することができる。その使用法の詳細については、476ページの『取り消されたアクティビティーに対するメッセージ』を参照。
XCBNAME	CL8	制御ブロック (DXEXCBA) を含む。これは、仮想記憶域のダンプでの目印としての役割を果たします。この値はセッション中は変更されない。
XCBNLANG	CL1	使用されている NLF を識別する (使用される NLID については、xviiiページの表1 を参照。) 値は、セッション中は変更されない。
XCBPANEL (ISPF のみ)	CL8	取り消しメッセージのメッセージ・ヘルプ・パネルのパネル ID を含む。XCBPANEL の使用法の詳細については、476ページの『取り消されたアクティビティーに対するメッセージ』を参照。
XCBPLAN	CL8	QMF のアプリケーション・プラン ID が含まれている。この値はセッション中、変わらない。このフィールドは、CICS には適用されない。

管理プログラム出口ルーチンを使用した QMF リソースの制御

表 57. 管理プログラムへの DXEXCBA インターフェース制御ブロックのフィールド (続き)

フィールド	データ・タイプ	目的
XCBQCE	F	DBMS から戻された、SQLCA の SQLDERRD(4) フィールドの値の 10 進数に対応するものが入っている。この 10 進数の整数部分は、データベース状況 (『相対費用見積もり』) パネル内に現れる。値は、コマンドの実行を終えると、機能呼び出し時にゼロに設定される。このフィールドは、操作がデータ検索照会ではない場合は、ゼロになる。照会費用見積もりは、DB2 パラレル・エディション V1.2 または DataJoiner v1.2.1.1からは使用できない。これらの環境では、値は 1 に設定される。
XCBQERR	CL1	前の機能呼び出し以降に QMF エラーが発生したかどうかを伝える。0 はエラーが発生しなかったことを示し、1 はエラーが発生したことを示す。
XCBQMF	CL10	QMF の現行のリリースを識別する。この値は QMF V7R1 で、セッション中は変更されない。
XCBQRYP	A	<p>QMF が実行のためにデータベースに渡す照会のコピーのアドレスが含まれている。管理プログラムは、データベース・アクティビティの開始を呼び出す時に (データ検索の前)、照会を検査し、アクティビティを取り消すかどうかを決める。アドレスは、セッションのはじめ、あるいはデータ・オブジェクトがリセットされるか、一時記憶域にインポートされたときのいずれかで、ゼロに設定される。</p> <p>このフィールドが情報を持っているのは、次のいずれかのコマンドによって、データ検索が要求された場合に限られる。OS/390 DB2 システム表または QMF 制御表についての照会に対してはなにも情報が提供されない。</p> <p>DISPLAY TABLE EDIT TABLE</p> <p>ERASE TABLE EXPORT TABLE</p> <p>IMPORT TABLE PRINT TABLE</p> <p>RUN QUERY SAVE DATA</p>
XCBREFR	CL1	<p>QMF が管理プログラムから戻った後で、画面を最新表示するかどうかを示す。1 は最新表示を行う、0 は行わないことを示している。</p> <p>ユーザーの管理プログラムが画面情報を表示する場合は、このフィールドを 1 に設定する。</p>

管理プログラム出口ルーチンを使用した QMF リソースの制御

表 57. 管理プログラムへの DXEXCBA インターフェース制御ブロックのフィールド (続き)

フィールド	データ・タイプ	目的
XCBRELN	CL2	QMF のリリース・レベルを識別する。QMF (OS/390版) V7R1 では、これは 11 になる。11. この値はセッション中、変わらない。
XCBRGRP	CL16	ユーザーのリソース・グループの名前を含む。この値はセッション中は変更されない。
XCBROWSF	F	<p>データ・オブジェクト内に検索される行数を反映する。最初はゼロであり、その後、行が検索されるごとに、このフィールドの値が変わる。データがデータベースから検索され、順次ファイル、CICS 一時記憶域、あるいは CICS 一時データ・キューから検索されても、すべてのデータ検索がカウントされる。</p> <p>QMF はこのフィールドをリセットしないが、管理プログラムはリセットできる。たとえば、ユーザーの管理プログラム出口ルーチンが検索されるデータベース行の数をモニターする場合、データ検索を開始したコマンドの終了に関する機能呼び出し時にこのフィールドをゼロに設定できる。</p>
XCBSYST	CL1	<p>現行のオペレーティング・システムを識別する。セッション中は値は変わらない。通常、この値は、TSO またはネイティブ OS/390 バッチ を示す 3 に設定されている。取り得る値は次のとおり。</p> <ul style="list-style-type: none"> 1 CMS (VM/SP) 3 TSO またはネイティブ OS/390 バッチ (MVS/XA™ または MVS/ESA) 4 CMS (VM/XA または VM/ESA) 5 CICS (VSE/ESA、MVS/ESA、または MVS/XA) <p>QMF (OS/390 版) V6 の場合には他の値も有効な理由については、251ページの『正しいプロファイルをユーザーの操作環境に提供』を参照。</p>
XCBTRACE	CL1	<p>ユーザー出口アクティビティーがトレースされる詳細のレベルの値を含む。取り得る値は 0 (最低の詳細度)、1 または 2 (最高の詳細度)。管理プログラムのこの値の使用については、476ページの『取り消されたアクティビティーに対するメッセージ』で説明する。</p> <p>セッション開始時に、ユーザーの QMF プロファイルからの TRACE フィールドの値をここで使用する。その後は、ユーザーが TRACE オプションの値を変更した場合のみ、値が変わる。トレースに関する詳細については、544ページの『QMF トレース機能の使用』を参照。</p>

表 57. 管理プログラムへの DXEXCBA インターフェース制御ブロックのフィールド (続き)

フィールド	データ・タイプ	目的
XCBUSER	CL8	ユーザーの TSO ログオン ID (TSO または SRPI の場合)、JOB ステートメントのユーザー・パラメーター (ネイティブ OS/390 バッチの場合)、または APPC 接続 で送信されるユーザー ID (APPC の場合) を含んでいる。このフィールドは、CICS では使用されず、CICS ではブランクになっている。
XCBUSERS	CL2048	ユーザーがある呼び出しから次の呼び出しまで管理プログラムに保管させたい結果を保管することのできるスクラッチパッド域。最初はブランクに設定されている。QMF はこの値を変更しない。

472ページの図141 は、DXEXCBA 制御ブロックの構造を示す。

管理プログラム出口ルーチンを使用した QMF リソースの制御

```

***** 00001000
* 00002000
* CONTROL BLOCK NAME: DXEXCBA * 00003000
* * 00004000
* FUNCTION: * 00005000
* * 00006000
* THIS IS THE INTERFACE CONTROL BLOCK BETWEEN QMF AND * 00007000
* EXIT ROUTINES. * 00008000
* * 00009000
* STATUS: VERSION 7 RELEASE 1 LEVEL 0 * 00010000
* * 00011000
* INNER CONTROL BLOCKS: NONE * 00012000
* * 00013000
* CHANGE ACTIVITY: * 00014000
* * 00015000
* * 00016000
***** 00017000
* 00018000
DXEXCBA DSECT 00019000
      DS 0D 00020000
XCBNAME DS CL8 -- CONTROL BLOCK IDENTIFICATION 00021000
      SPACE 00022000
XCBEXCTL DS XL190 -- EXIT CONTROL 00023000
      ORG XCBEXCTL 00024000
XCBAUTH DS CL8 ----- AUTHORIZATION ID 00025000
XCBUSER DS CL8 ----- USER ID 00026000
XCBPLAN DS CL8 ----- PLAN ID 00027000
      SPACE 00028000
XCBQMF DS CL10 ----- CURRENT VERSION/RELEASE 00029000
      SPACE 00030000
XCBRELN DS CL2 ----- QMF RELEASE LEVEL 00031000
      SPACE 00032000
XCBTRACE DS CL1 ----- QMF EXIT TRACE LEVEL 00033000
XCBTOFF EQU C'0' ----- NO TRACING 00034000
XCBTPART EQU C'1' ----- PARTIAL TRACING 00035000
XCBTFULL EQU C'2' ----- FULL TRACING 00036000
      SPACE 00037000
XCBSYST DS CL1 ----- OPERATING SYSTEM 00038000
XCBSYSTX EQU C'3' ----- MVS/ESA or XA (TSO,APPC, native) 00039000
XCBSYSTV EQU C'4' ----- CMS/VM/ESA 00040000
XCBSYSTY EQU C'5' ----- CICS (OS/390 or VSE) 00041000
      SPACE 00042000
XCBPAD10 DS CL4 ----- RESERVED FIELD 00043000
      SPACE 00044000
XCBNLANG DS CL1 ----- CURRENT NATIONAL LANGUAGE 00045000
      SPACE 00046000
XCBKPARM DS CL1 ----- SETTING OF K PARAMETER 00047000
XCBKPARN EQU C'0' ----- LATIN 00048000

```

図 141. DXEXCBA 制御ブロック (1/3)

管理プログラム出口ルーチンを使用した QMF リソースの制御

XCBKPARY	EQU	C'1'	-----	DBCS	00049000
	SPACE				00050000
XCBDBMG	DS	CL1	-----	DATA BASE MANAGER	00051000
XCBDBMGS	EQU	C'1'	-----	DB2 FOR VM/VSE	00052000
XCBDBMGD	EQU	C'2'	-----	DB2 FOR OS/390	00053000
XCBDBMGW	EQU	C'3'	-----	WORKSTATION DB2	00054000
	SPACE				00055000
XCBEMODE	DS	CL1	-----	CURRENT EXECUTION MODE	00056000
XCBIACTV	EQU	C'1'	-----	INTERACTIVE MODE	00057000
XCBBATCH	EQU	C'2'	-----	BATCH MODE	00058000
	SPACE				00059000
XCBIAICT	DS	CL1	-----	CURRENT INTERACT MODE	00060000
XCBIAIACY	EQU	C'1'	-----	INTERACTIVE EXECUTION	00061000
XCBIAIACN	EQU	C'0'	-----	NOT INTERACTIVE EXECUTION	00062000
	SPACE				00063000
XCBINCI	DS	CL1	-----	CURRENT COMMAND INTERFACE STATE	00064000
XCBINCIY	EQU	C'1'	-----	COMMAND INTERFACE ACTIVE	00065000
XCBINCIN	EQU	C'0'	-----	COMMAND INTERFACE NOT ACTIVE	00066000
	SPACE				00067000
XCBINPRC	DS	CL1	-----	PROCEDURE OR LIST CMD EXEC STATE	00068000
XCBPRCY	EQU	C'1'	-----	RUNNING A PROCEDURE OR LIST CMD	00069000
XCBPRCN	EQU	C'0'	-----	NOT RUNNING PROCEDURE OR LIST CMD	00070000
	SPACE				00071000
XCBCVERB	DS	CL18	-----	CURRENT COMMAND VERB	00072000
	SPACE				00073000
XCBCAN	DS	CL1	-----	CANCEL CURRENT COMMAND INDICATOR	00074000
XCBCANN	EQU	C'0'	-----	NO CANCELLATION	00075000
XCBCANY	EQU	C'1'	-----	CANCELLATION IN PROGRESS	00076000
	SPACE				00077000
XCBACTIV	DS	CL1	-----	TYPE OF DATA BASE ACTIVITY	00078000
XCBOPEN	EQU	C'1'	-----	OPEN	00079000
XCBFETCH	EQU	C'2'	-----	FETCH	00080000
XCBPREP	EQU	C'3'	-----	PREPARE	00081000
XCBDESCR	EQU	C'4'	-----	DESCRIBE	00082000
XCBCLOSE	EQU	C'5'	-----	CLOSE	00083000
XCBEXEC	EQU	C'6'	-----	EXECUTE	00084000
XCBEXECI	EQU	C'7'	-----	EXECUTE IMMEDIATE	00085000
XCBPAD20	DS	CL9	-----	RESERVED FIELD	00086000
	SPACE				00087000
XCBGRGP	DS	CL16	-----	RESOURCE GROUP NAME	00088000
XCBPAD30	DS	CL22	-----	RESERVED FIELD	00089000
	SPACE				00090000
XCBCMDP	DS	A	-----	POINTER TO ORIGINAL COMMAND STRING	00091000
*			-----	WILL NOT CONTAIN PROMPT VALUES	00092000
	SPACE				00093000
XCBCMDL	DS	F	-----	ORIGINAL COMMAND STRING LENGTH	00094000
	SPACE				00095000
XCBQCE	DS	F	-----	QUERY COST ESTIMATE VALUE	00096000

図 141. DXEXCBA 制御ブロック (2/3)

管理プログラム出口ルーチンを使用した QMF リソースの制御

	SPACE			00097000
XCBROWSF	DS	F	----- DATA BASE ROWS FETCHED FROM SOURCE	00098000
*			----- SET BY QMF; EXIT MAY RESET	00099000
	SPACE			00100000
XCBQERR	DS	CL1	----- QMF ERROR INDICATOR	00101000
XCBQERRN	EQU	C'0'	----- NO QMF ERROR DETECTED	00102000
XCBQERRY	EQU	C'1'	----- QMF ERROR DETECTED	00103000
XCBCLOC	DS	CL18	----- CURRENT LOCATION NAME	00104000
XCBPAD40	DS	CL41	----- RESERVED FIELD	00105000
	SPACE			00106000
XCBQRYP	DS	A	----- POINTER TO SQL QUERY	00107000
*			----- QUERY LENGTH IS FIRST HALFWORD	00108000
	SPACE			00109000
XCBUCTL	DS	XL432	-- USER CONTROL AREA	00110000
	ORG	XCBUCTL		00111000
XCBERRRET	DS	F	----- EXIT ERROR RETURN CODE	00112000
XCBMGTX	DS	CL78	----- EXIT ERROR MESSAGE TEXT	00113000
XCBMSGNO	DS	CL8	----- ISPF MESSAGE NUMBER	00114000
XCBPANEL	DS	CL8	----- ISPF MESSAGE HELP PANEL	00115000
XCBLOGM	DS	CL1	----- LOG MESSAGE INDICATOR	00116000
XCBLOGMN	EQU	C'0'	----- QMF SHOULD NOT LOG MESSAGE	00117000
XCBLOGMY	EQU	C'1'	----- QMF SHOULD LOG MESSAGE	00118000
XCBREFR	DS	CL1	----- REFRESH SCREEN INDICATOR	00119000
XCBREFRN	EQU	C'0'	----- QMF DOES NOT HAVE TO REFRESH SCR	00120000
XCBREFRY	EQU	C'1'	----- QMF SHOULD REFRESH SCREEN	00121000
XCBPAD50	DS	CL28	----- RESERVED FIELD	00122000
	SPACE			00123000
XCBUSERS	DS	CL2048	-- USER SCRATCH PAD AREA	00124000
XCBPAD60	DS	CL48	----- RESERVED FIELD	00125000

図 141. DXEXCBA 制御ブロック (3/3)

QMF セッション間のリソース制御情報の保管

セッションの最初の呼び出しで管理プログラムに渡される情報を、後続の管理プログラム・ルーチンに対する呼び出しで使用することができます。

DXEGOVA および DXEXCBA 制御ブロック内に用意されている 2048 バイトのスクラッチパッド域を使用し、リソース制御情報を保持するために必要な記憶域を獲得することができます。これらのフィールドには、保管する必要のあるものは何でも入れることができます。この情報は、管理プログラムに対する 1 つの呼び出しから、次の呼び出しまで存続しています (ただし、CONNECT 呼び出しがその情報を変更しない場合)。

IBM 提供の管理プログラムは 475 ページの図 142 に示したコードを使用して、GOVUSERS を指します。これは、DXEGOVA 制御ブロックのスクラッチパッド域です。以下の例の GOVUSERS を XCBUSERS で置き換えることにより、類似のコードを使用して DXEXCBA 制御ブロック内の XCBUSERS スクラッチパッド域を指せます。

管理プログラム出口ルーチンを使用した QMF リソースの制御

```
LA    WORKPTR,GOVUSERS
USING WORK,WORKPTR
```

図 142. 管理プログラム・スクラッチパッド域へのアドレス可能性の確立

図142 では、WORK は DSECT の名前であり、WORKPTR は汎用レジスター 4 と同等です。WORK DSECT は、スクラッチパッド域の情報を保管するフィールドの定義を含んでいます。

また、管理プログラムは、GETMAIN マクロを発行して、必要な記憶域を取得する場合があります。

ユーザー・アクティビティーの取り消し

ユーザーは、そのリソース限界に達すると、QMF 取り消しサービスを呼び出してユーザー・アクティビティーを取り消すことができます。たとえば、管理プログラム出口ルーチンは、以下のものを取り消すことができます。

- QMF セッション開始時における、機能呼び出し中の QMF セッション
- さまざまな関数を呼び出し中の現行コマンド、およびデータベース・アクティビティーを開始させるコマンド
- タイマーがアクティブのときの非同期コマンド (TSO、SRPI、APPC、またはネイティブ OS/390 の場合)

最初の 2 つのアクティビティーのいずれかを取り消すためのコードは、ソース・プログラム DSQUn GV1 または DSQUn GV3 にあります。管理プログラムに QMF 取り消しサービスを呼び出させてアクティビティーを取り消すには、DXEGOVA 制御ブロックの GOVCADDR という名のフィールドに現れるアドレスにジャンプします。図143 は、QMF 取り消しサービスへのアドレス可能性を確立するステートメントを示したものです。これらのステートメントを使用して管理プログラム出口ルーチンから QMF へ制御を渡す前に、レジスター 13 が管理プログラムの保管域を指していることを確認してください。これは、制御が戻ったときに、QMF が管理プログラムの状態を復元できるようにするためです。

```
L R15,GOVCADDR
BALR R14,R15
```

図 143. QMF 取り消しサービスの呼び出し

取り消しルーチンは、レジスター 14 によって指されるポイントに制御を戻します (この場合、BALR コマンドに続くコマンド)。レジスター 15 は、QMF

管理プログラム出口ルーチンを使用した QMF リソースの制御

が取り消し要求を受け入れた場合は 0 のリターン・コードを、また、QMF が非アクティブであるとき管理プログラムが取り消しを要求した場合は 100 のリターン・コードを含んでいます。

TSO、SRPI、APPC、またはネイティブ OS/390 において、非同期処理を使用して QMF コマンドを取り消すために、IBM 提供の管理プログラムはタイマー・マクロを使用しています。これは、制御をタイマー・ルーチンに戻します。タイマー・ルーチンは現行コマンドを取り消すかどうかをテストします。コマンドが取り消される場合は、取り消しを実行します。テストは、プロセッサ時間および現行データ・オブジェクトの検索行数に基づいています。また、テストは、取り消しのプロンプトに対するユーザーの応答に基づくこともできます。

タイマー・ルーチンは、IBM 提供の管理プログラムのソース・コード内の TIMEX という名前の CSECT です。ソース・コードは、ライブラリー QMF710.SDSQUSRE のメンバー DSQUn GV1 です。

非同期取り消し呼び出しを行うことは、PA1 を押すのと非常によく似ています。取り消しは即時に行われないこともあり、不可能な場合もあります。取り消しが生ずる前に、制御を管理プログラムに戻すことができます。

取り消されたアクティビティーに対するメッセージ

DXEXCBA 制御ブロックの以下のフィールドを使用することにより、QMF メッセージ・サービスを使用して、コマンドが取り消された後にユーザーにメッセージを表示することができます。

XCBMGTXT

メッセージ・テキストが含まれています。

XCBERRET

エラー・リターン・コードが含まれています。

XCBMSGNO

QMF が TSO の ISPF のもとで呼び出された場合には、ISPF メッセージ定義のメッセージ ID を含みます。

XCBPANEL

QMF が TSO の ISPF のもとで呼び出された場合には、ISPF メッセージ・ヘルプ・パネルのパネル ID を含みます。

管理プログラムへ入ったとき、XCBMGTXT はブランクを含み、また XCBERRET は 2 進数ゼロを含みます。XCBERRET の値は、画面にどんなメッセージが表示されるかを決定します。

管理プログラム出口ルーチンを使用した QMF リソースの制御

- OK, command canceled というメッセージを使用する場合は、XCBERRET はゼロのままにしておきます。
- A governor exit cancel occurred with return code xxxxx というメッセージを使用する場合は、XCBERRET にゼロ以外の値を使用します。この値が、xxxxx の代わりにメッセージに表示されます。

QMF 初期化が管理プログラム出口により取り消された場合は、XCBMGTXT および XCBERRET に先行するメッセージが、画面ではなく、ユーザーのトレース・データに現れます。

XCBLGGM を 1 に設定して、ユーザー独自の管理プログラム出口ルーチンにおけるすべての機能呼び出しのトレース・データでのメッセージを記録します。XCBERRET の値がゼロでない場合は、DXEXCBA 制御ブロックのXCBLGGM フィールドを値 1 に設定することにより、IBM 提供の管理プログラムは取り消しメッセージをユーザーのトレース・データに記録します。

ISPF メッセージ定義は長いメッセージ・テキストを含むことができ、またパネル ID を指定することができます。メッセージにおよびヘルプの指定パネルに、長いテキストを使用する場合は、XCBSMSGNO にメッセージ定義のメッセージ ID で埋め、XCBMGTXT および XCBPANEL はブランクのままにしておきます。メッセージ定義で HELP パネルが指定されなかった場合は、ユーザーはメッセージ・ヘルプは受け取りません。

メッセージ定義内の長メッセージ仕様を上書きするには、新規のメッセージ・テキストを XCBMGTXT 内に入れます。パネル仕様を上書きするには、新規のパネル ID を XCBPANEL 内に入れます。パネル ID を XCBPANEL に入れると、メッセージ定義でパネルの指定が無くても、メッセージ・ヘルプの準備もします。

関連する ISPF メッセージ定義が存在しない場合は、XCBSMSGNO をブランクにしておきます。その上で、メッセージ・テキストを XCBMGTXT に入れ、HELP パネル ID があれば XCBPANEL に入れます。この場合、XCBPANEL をブランクのままにしておく、ユーザーにはメッセージ・ヘルプは提供されません。

トレース機能は、DXEXCBA 制御ブロックの XCBTRACE フィールドの値により決まる詳細レベルで、メッセージを DSQDEBBUG データ・セットに書き込みます。メッセージをログに記録したくない場合は、XCBTRACE にゼロの値を使用します (ただし、トレース・データ・セットを割り当てない限り、初期化エラーはログに記録されます)。値 1 または 2 をトレース・オプションの U

管理プログラム出口ルーチンを使用した QMF リソースの制御

設定に使用し、トレース出力を入手します。QMF トレース機能の使い方の詳細については、544ページの『QMF トレース機能の使用』を参照してください。

QMF が TSO の ISPF のもとで呼び出された場合には、管理プログラムはメッセージを ISPF ログ・ファイルにも記録することができます。管理プログラムは、*ISPF MVS 対話管理サービスの解説* の資料で説明されている ISPF LOG サービスを用いてこの記録を行うことができます。

QMF アプリケーションからコマンドをバッチまたは非対話式に実行している場合には、メッセージは画面に表示されません。

IBM 提供の管理プログラムは終了の機能呼び出しについては、メッセージをログに記録しません。

CICS でのユーザー管理プログラム出口ルーチンの変換、アセンブル、リンク・エディット

IBM 提供の管理プログラム出口ルーチンを変更する場合でも、ユーザー独自のルーチンを作成する場合でも、ルーチンの変換、アセンブル、リンク・エディットが必要です。この節で示されているサンプルのリンク・エディットのステートメントを参照してください。

CICS の管理プログラム出口ルーチンの変換

アセンブラー用の CICS 変換プログラムを使用して、プログラムを変換します。ユーザーのプログラムを変換する際、CICS はアドレス可能性を確立する標準の CICS プロローグ (DFHEIENT) を提供し、レジスターを標準の CICS 作業記憶域に保管します。標準のプロローグは、標準の CICS エピローグ (DFHEIRET) も備えています。

管理プログラム出口ルーチンのアセンブル

QMF が管理プログラムでサポートしているのはアセンブラー言語のみです。アセンブラー言語は、たとえば、IBM 提供の管理プログラムのコーディングに使用されている言語です。管理プログラムは、HLASM またはアセンブラー H 用に作られています。このプログラムは、QMF710.SDSQUSRE ライブラリーの特定のメンバーを印刷すれば、見ることができます。詳細については、446ページの『管理プログラム出口ルーチンのプログラム・コンポーネント』を参照してください。

管理プログラム出口ルーチンのリンク・エディット

管理プログラムのロード・モジュールを、全 QMF ユーザーが使用可能なライブラリーに入れます。ライブラリーとしては、QMF 自体のロード・モジュールが入っている QMF710.SDSQLOAD ライブラリーをお勧めします。このライブラリーは、CICS の DFHRPL と連結しておく必要があります。

モジュール DSQU n GV3 を指定します。これは、IBM 提供のモジュールの名前です。ユーザー独自の管理プログラム・モジュールを QMF710.SDSQLOAD ライブラリーに入れると、このモジュールがそのライブラリーのメンバーであるため、IBM 提供のモジュールと置き換わります。

IBM 提供のモジュールとの置き換えを避けるためには、そのモジュールを名前変更するか、他のライブラリーに移します。あるいは、ユーザー独自の管理プログラムのモジュールを、DFHRPL の別のライブラリーに入れることも可能です。この最後の選択肢の場合、ユーザーのモジュールのある新しいライブラリーが連結シーケンスで、QMF710.SDSQLOAD よりも、必ず、前にくるようにしてください。そうしないと、QMF は IBM 提供のモジュールを呼び出してしまいます。

このモジュールの入り口点は、必ず、DSQU n GV3 でなければなりません。よくあることですが、ソース・コードがこのラベルの CSECT ステートメントで始まっている場合は、ほかに何もすることはありません。そうでない場合は、アセンブラー・コードの END ステートメントにエントリー名を指定してください。あるいは、リンクエッジ・エディターの入力で ENTRY ステートメントにエントリー名を付けてください。

リンク・エディットの際、CICS コマンド・インターフェース制御モジュールの DFHEAI および DFHEAI0 を組み込む必要があります。また、この制御モジュールは、管理プログラムのロード・モジュールのはじめに置く必要があります。CICS では、管理プログラムは AMODE(31) および RMODE(ANY) で実行する必要があります。

```
INCLUDE SYSLIB(DFHEAI)
INCLUDE SYSLIB(DFHEAI0)
ORDER DFHEAI,DFHEAI0
ENTRY DSQUEGV3
MODE AMODE(31),RMODE(ANY)
NAME DSQUEGV3(R)
```

管理プログラム出口ルーチンの、TSO およびネイティブ OS/390 バッチでのアセンブルとリンク・エディット

IBM 提供の管理プログラム出口ルーチンを変更する場合でも、ユーザー独自のルーチンを作成する場合でも、ルーチンの変換、アセンブル、リンク・エディットが必要です。この節で示されているサンプルのリンク・エディットのステートメントを参照してください。

管理プログラム出口ルーチンのアセンブル

QMF が管理プログラムでサポートしているのはアセンブラー言語のみです。アセンブラー言語は、たとえば、IBM 提供の管理プログラムのコーディングに使用されている言語です。管理プログラムは、HLASM またはアセンブラー H 用に作られています。このプログラムは、QMF710.SDSQUSRE ライブラリーの特定のメンバーを印刷すれば、見ることができます。詳細については、446ページの『管理プログラム出口ルーチンのプログラム・コンポーネント』を参照してください。

管理プログラム出口ルーチンのリンク・エディット

管理プログラムのロード・モジュールを、全 QMF ユーザーが使用可能なライブラリーに入れます。ライブラリーとしては、QMF 自体のロード・モジュールが入っている QMF710.SDSQLOAD ライブラリーをお勧めします。このライブラリーは、STEPLIB の連結の一部でも構いません。

モジュール DSQU n GV1 を指定します。これは、IBM 提供のモジュールの名前です。ユーザー独自の管理プログラム・モジュールを QMF710.SDSQLOAD ライブラリーに入れると、このモジュールがそのライブラリーのメンバーであるため、IBM 提供のモジュールと置き換わります。

IBM 提供のモジュールとの置き換えを避けるためには、そのモジュールを名前変更するか、他のライブラリーに移します。あるいは、ユーザー独自の管理プログラムのモジュールを、STEPLIB の別のライブラリーに入れることが可能です。ユーザーのモジュールを別のライブラリーに入れる場合、ユーザーのモジュールのある新規のライブラリーが連結シーケンスで、必ず、QMF710.SDSQLOAD よりも前にくるようにしてください。そうしないと、QMF は IBM 提供のモジュールを呼び出してしまいます。

このモジュールの入り口点は、必ず、DSQU n GV1 でなければなりません。ソース・コードが、DSQU n GV1 ラベルの CSECT ステートメントで始まっている場合は、ほかに何もすることはありません。ソース・コードが DSQU n GV1 ラベルで始まっていない場合は、アセンブラー・コードの END ステート

管理プログラム出口ルーチンを使用した QMF リソースの制御

メントにエンタリー名を指定してください。あるいは、リンケージ・エディターの入力で ENTRY ステートメントにエンタリー名を付けてください。

ユーザー独自のルーチンは、31 ビットまたは 24 ビットのいずれのアドレッシング・モードでも実行可能です。ユーザーのルーチンが 24 ビット・アドレッシング・モードが必要な OS/390 サービス (TPUT など) を要求する場合は、QMF は、31 ビット・モードで稼働している QMF から 24 ビット・モードで稼働している管理プログラムへ転送し、また、31 ビット・モードの QMF へ戻る処理を行います。

```
ENTRY DSQUEGV1
MODE AMODE(31),RMODE(ANY)
NAME DSQUEGV1(R)
```

QMF が提供している管理プログラム (DSQUEGV1) は、AMODE(24) および RMODE(24) で実行する必要があります。

```
ENTRY DSQUEGV1
MODE AMODE(24),RMODE(24)
NAME DSQUEGV1(R)
```

DB2 管理プログラムの使用

DB2 は、独自の管理プログラムを持っており、QMF 管理プログラムとは独立して稼働します。この節では、DB2 管理プログラムでは何ができるか、また、追加リソースの制御に対してどのように使用できるかについて説明します。

DB2 管理プログラムの詳細については、*DB2 UDB (OS/390 版) 管理の手引き*のリソース使用効率の向上に関する節をお読みください。DB2 の資料では、この管理プログラムは一般に、*リソース限定機能*と呼ばれています。DB2 管理プログラムを用いると、データベースに対するすべてのアクセスと分散アクセスを制御することができます。

リソースのモニター

DB2 管理プログラムは、特定の照会を実行するときに費やされるプロセッサ時間をモニターしています。モニターする照会は、動的に実行される SELECT、INSERT、UPDATE、および DELETE の照会です。QMF セッションでは、以下のようにして実行されるすべての照会が含まれます。

QMF RUN コマンドの使用

実行される照会は、SQL、QBE、または指示照会場合があります。QBE および指示照会の場合、管理プログラムは対応する SQL 照会をモニターします。

他の QMF コマンドの使用

他のコマンドのサポートでは、QMF はユーザーに代わって SQL 照会を作成し、実行します。たとえば、これらの照会の中には、DISPLAY 表コマンドに回答して、QMF が実行する SELECT 照会があります。

表編集機能の実行

表編集機能のサポートでは、QMF はユーザーに代わって、SQL 照会を作成し、実行します。たとえば、これらの照会の中には、SEARCH コマンドに回答して、QMF が実行する SELECT 照会があります。

それぞれの照会の詳細については、265ページの『オブジェクトにアクセスするのに必要な SQL 特権』を参照してください。

管理プログラム間の相違点

QMF 管理プログラムの操作を、DB2 管理プログラムで補うことができます。その前に、2 つの管理プログラムの違いを理解しておいてください。

- DB2 管理プログラムのモニターは、前の節で説明した照会のタイプに限定されます。たとえば、CREATE または DROP 照会を実行する際に費やされるプロセッサ時間のモニターは行いません。
- DB2 管理プログラムのモニターは、プロセッサ時間に限定されます。QMF 管理プログラムが行うような、行の取り出しをカウントすることはありません。
- DB2 管理プログラムの場合のプロセッサ時間とは、DB2 が費やす時間だけを含みます。これに対して、QMF 管理プログラムでは、コマンドの実行上 QMF が費やす時間、たとえば、予備ファイルの処理や SELECT 照会における実行結果の先頭ページの表示など、が含まれています。
- ユーザーが SELECT 照会を実行すると、DB2 管理プログラムは照会を実行するのに DB2 が費やした時間はすべてモニターしています。これには、PREPARE ステートメントから始まり、行の取り出しから、カーソルのクローズまで続きます。QMF 管理プログラムは、結果の先頭ページが表示されれば、モニターを終了します。引き続いて、行の取り出しを行っても、取り出しを生じさせたスクロール・コマンドの一部として扱われます。
- DB2 管理プログラムは、取り消しプロンプトの準備は行いません。所定の QMF セッションに対する唯一の制御パラメーターは、最大プロセッサ時間です。

最大プロセッサ時間の超過

照会が最大プロセッサ時間を超過すると、DB2 管理プログラムは照会を終了し、SQL エラー・コード -905 を戻します。これによって、QMF は管理プロ

管理プログラム出口ルーチンを使用した QMF リソースの制御

グラムが照会を取り消したことを知ります。QMF がこの情報をどのように処理するかについては、管理プログラムが QMF セッションのどこでその照会を取り消したかによって異なります。

QMF 初期化の間

ユーザー・セッションが開始されると、QMF は DB2 管理プログラムがモニターしているいくつかの照会を実行します。これらのいずれかの照会が取り消された場合、QMF はセッションを終了します。セッションを終了する前に、QMF はユーザーの DSQDEBUG データ・セットに説明のレコードを書き込みます。

セッションの終了は、QMF セッションが許可されていない時点で発生させることも可能です。この制限を強制的に実施する場合には、そのような許可されていない時点で QMF を使おうとする人に対しては、最大プロセッサ時間であるゼロを割り当てておきます。これによって、モニターされている照会はいずれも取り消されてしまいます。

QMF 初期化の後

初期化の後には、QMF は照会の取り消しを照会実行中のその他のエラーの処理と同じように扱います。たとえば、管理プログラムが RUN コマンドが出された INSERT 照会を取り消すとします。ここで、もし挿入があれば取り消され、照会パネルがエラー・メッセージと共に表示されます。そこでユーザーがメッセージ・ヘルプを要求すると、管理プログラムのアクションを説明するパネルが表示されます。

今度は、ユーザーが報告書をスクロールしている最中に取り消しが効力を生じたとします。この場合は、行取り出しが取り消しにつながる可能性があります。取り消しが生じると、データ・オブジェクトは未完了のままになります。その理由は、DB2 がカーソルをクローズしてしまうため、データ・オブジェクトが完了できないことによります。

DB2 管理プログラムの QMF への適用

DB2 管理プログラムが QMF セッションの管理を行えるようになるためには、まず入力が必要とします。この場合の入力とは、最大プロセッサ時間です。DB2 管理プログラムは、この入力を **リソース限定表** の行から取得します。DB2 の用語では、この表は **RLST** のことです。このような表は、適切な DB2 権限 (INSERT、UPDATE など) を持つ人なら誰でも変更可能です。1 つまたは複数の RLST に行を追加することによって、QMF ユーザーのために DB2 管理プログラムを制御することができます。

RLST の選択

QMF をインストールする DB2 サブシステムについて考えてみます。サブシステムは開始すると、特定の RLST と関連付けられます。次に、この RLST は DB2 管理プログラムに、QMF セッションを開始するユーザーを含むすべてのサブシステム・ユーザーのための入力を用意します。

異なる RLST を同じ DB2 サブシステムに、別の時点で関連付けることができます。たとえば、インストール先では、時間帯が異なれば RLST も異なるものを使用する可能性があります。ある時間帯の RLST では、その時間帯に QMF を使用できないようにします。QMF を開始しようとしても、初期化中に QMF は終了してしまい、DSQDEBUG データ・セットにメッセージが現れます。

行を RLST に追加

ユーザー（または、適切な DB2 権限を持つ人）は、QMF ユーザー用の RLST に行を追加することができます。行には以下のものが含まれています。

- 許可 ID
- DB2 アプリケーション・プランの名前
- 最大プロセッサ時間の値
- 要求が出されたサイトの LU 名 (DB2 2.2 以降)

たとえば、数人の個人ユーザーの行、およびその他のすべての人に適用される行を追加する場合があります。数人の個人ユーザーの行には、基本許可 ID と QMF アプリケーション・プラン名が入っています。その他のユーザーの行には、QMF 計画名と許可 ID にはブランクが入っています。

RLST で可能なこと、不可能なこと、さらに表の構造の詳細については、DB2 管理者に問い合わせてください。それぞれの RLST には、あらかじめ指定された名前とデータ・タイプを持つ必須の列がありますが、インストール先システムでさらに列を追加していることもあります。これらの表に関する一般的な説明については、DB2 UDB (OS/390 版) 管理の手引き を参照してください。

第22章 リモート・データベース接続のカスタマイズ

リモート・データベース接続をカスタマイズし、ユーザーがリモート・システム上のリレーショナル・データを照会して、ローカル・システム上のデータを表す報告書または図表を作成できるようにすることができます。分散ネットワーク内のどの DB2 データベースとも、接続を確立することができます。(QMF ユーザーは、OS/400® のロケーションには接続できません。) この接続は、QMF の初期化中に、あるいは QMF セッションから、確立することができます。同種 (たとえば DB2 (OS/390 版) から DB2 (OS/390 版)) および異種 (たとえば DB2 (OS/390 版) から DB2 (VM または VSE 版)) のロケーションの間で接続を確立することができます。

QMF を使用すると、DB2 (OS/390 版) と DB2 (VM および VSE 版) の両方でインプリメントされた分散データベース・ソリューションを介してリモート・データにアクセスできるようになります。このソリューションは、分散リレーショナル・データベース体系 (DRDA) に基づいています。DRDA はプロトコルとフォーマットのオープン・セットであり、これにより、同種または異種のリレーショナル・データベース管理システム (RDBMS) に属するローカルおよびリモート・データに容易にアクセスできるようになります。

ユーザーがリモート・ロケーションに接続されると、ユーザーのアプリケーション内の SQL ステートメントはすべて、処理のためにそのデータベースに送られます。ユーザーは、リモート作業単位接続のないデータやオブジェクトにアクセスするのと同じ方法で、そのロケーションのデータベースにあるデータや QMF オブジェクトにアクセスすることができます。QMF は実行中のシステムと同じシステムにあるプログラムを引き続き使用します。このタイプの分散接続を、リモート作業単位 と呼びます。(図式ダイアグラムについては、487ページの図144を参照してください。)

クイック・スタート

表58 は、ユーザーのリモート・データベース接続をカスタマイズする方法の概要を示します。

リストしている作業に関する詳細については、表の右端に示しているページを参照してください。

リモート・データベース接続のカスタマイズ

表 58. リモート・データベース接続のカスタマイズ

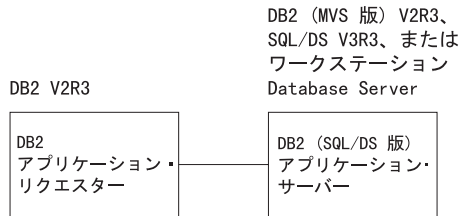
作業	参照箇所
ユーザーに必要なデータベース接続のタイプの判別。これは、ユーザーがアクセスする必要があるデータベースに応じて行う。	486 ページ
リモート作業単位に必要な接続を確認する。	489 ページ
ユーザーがアクセスするリモート・ロケーションを準備する。	491 ページ
ユーザーがリモート・データベースにアクセスできるようにする。このために、ユーザー・プロファイルおよび使用されるデータベースのロケーションを準備する。	496 ページ
他の管理者から自分のロケーションへのアクセスを可能にする。このために、他のユーザーのロケーションから自分のロケーションへのリモート作業単位を設定する。	504 ページ

必要なリモート・データベース接続の判別

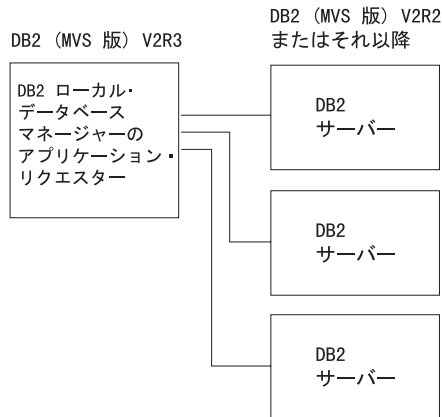
DB2 に対するデータベース接続には、2 つのタイプの設定が可能です。DB2 (OS/390 版) または DB2 (VM または VSE 版) へのリモート作業単位接続を使用するか、あるいは DB2 から DB2 への分散作業単位接続を使用することができます。

リモート作業単位および DB2 のみの分散作業単位は一緒に使用することができます。(概念図については、487ページの図144 を参照してください。)

リモート作業単位



DB2 のみの分散作業単位



リモート作業単位および DB2 のみの分散作業単位

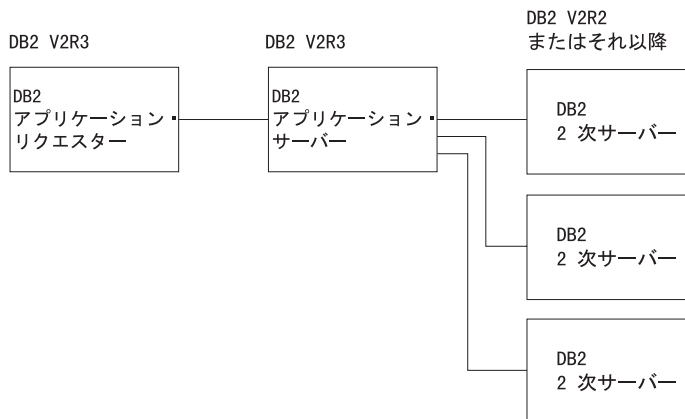


図 144. QMF/OS/390 からリモート作業単位と分散作業単位を使用する分散接続タイプ

リモート・データベース接続のカスタマイズ

注:

1. アプリケーション・リクエスターは、分散接続の QMF 側を処理するデータベース管理システム (DBMS) のコードです。QMF が接続されているローカル DB2 サブシステムは QMF のアプリケーション・リクエスターとして知られています。その理由は、DB2 のアプリケーション・リクエスターがローカル・データベース・マネージャー内にインストールされているためです。
2. アプリケーション・サーバーは、アプリケーション・リクエスターからの要求をサポートする DBMS コードであり、QMF では現行ロケーションと呼ばれています。
3. Database Server はアプリケーション・サーバーから送信される要求をサポートし、分散リレーショナル・データベース (DRDA) のデータベース・サポート・プロトコルを使用して、DRDA データベース間の整合性を保ちます。
4. 2 次サーバーは Database Server と似ていますが、DRDA データベース・サポート・プロトコルを使用しない点だけが異なります。

詳細については、*DRDA 接続の手引き* を参照してください。

リモート作業単位との接続

リモート作業単位を使用すれば、リモート DB2 (OS/390 版) のロケーションまたはリモート DB2 (VM または VSE 版) のロケーションにあるリレーショナル・データに接続し、アクセスすることができます。リモート・ロケーションは *サーバー* と呼ばれます。サーバーに接続されたとき、リモート作業単位の接続なしでデータおよびオブジェクトにアクセスするのと同じように、データおよび QMF オブジェクトにアクセスすることができます。

DB2 間分散作業単位による接続 (DB2 (OS/390 版) のみ)

分散作業単位を使用して、DB2 から DB2 への接続を設定することができます。分散作業単位があると、アプリケーション・プログラムは別のロケーションからデータにアクセスする場合に、異なるデータベースに接続する必要があります。代わりに、アプリケーションは照会または QMF コマンドの 3 部分名の中で、他のロケーションを指定します。

DB2 での 3 部分名による表または視点の指定

分散データ・サポートのある DB2 サブシステムに接続すると、3 部分名を使用して表または視点を指定することができます。QMF は単一の DB2 ロケーションに接続されたままになり、このロケーションが 3 部分名 (またはそれらの

別名) を使用するすべての SQL ステートメントを、3 部分名で参照されている DB2 ロケーションに送信します。その後、送信されたロケーションで、SQL ステートメントを処理します。

制約事項: 3 部分名を使用して、表または視点を参照する場合には、次の制限が適用されます。

- リモート DB2 サーバーからは、3 部分名を使用してローカル DB2 オブジェクトを参照することはできません。
- DB2 (VM または VSE 版) が現行ロケーションのときは、次のとおりです。
 - 3 部分名にあるロケーションは、現行ロケーション の名前 (QMF セッションが現在接続されているアプリケーション・サーバーの名前) と一致している必要があります。
 - QMF コマンド、指示照会、および QBE では、3 部分名をサポートしていません。
 - Workstation Database Server は、3 部分名をサポートしていません。

3 部分名を使用した照会の指示

指定されたロケーションへの接続を確立することが、リモート作業単位サポートの大部分を構成します。この接続が行われると、QMF はリモート作業単位サポート以前とほぼ同様に機能します。したがって、3 部分名のサポートもまだ提供されています。現行ロケーションが 3 部分名をサポートする DB2 ロケーションである場合、3 部分名を使用する SQL ステートメントは、別の DB2 ロケーションへ照会を宛先指示できます。

DB2 ネットワークは、DB2 V2R2、V2R3、および V3R1 サブシステムで構成できます。これらのいずれのサブシステムに対しても、3 部分名を使用して SQL ステートメントを実行することができます。ただし、リモート作業単位接続を確立できるのは、V2R3 またはそれ以降の DB2 サブシステムに限られます。

リモート作業単位に必要な接続の検査

ユーザーがリモート・システムにアクセスできるためには、ローカル・システムとリモート・システムとの間の接続が確立されていなければなりません。

DB2 接続の検査

DB2 からリモート・システムに接続するためには、リモート・システムが定義されていることを確認する必要があります。

リモート・データベース接続のカスタマイズ

DB2 アプリケーションがリモート・システムからのデータを要求すると、DB2 はリモート・システムを見つけるために、DB2 通信データベースを検索します。以下の項目が、DB2 で使用可能かどうかを検査する必要があります。

- 論理装置 (LU) 名およびトランザクション・プログラム名 (TPN)。VTAM はそれぞれのサーバーの LU 名を持っていることが必要です。
- リモート・サイトで必要なネットワーク・セキュリティー情報。
- リモート・サイトとの通信に使用するセッション限度およびモード名。

分散データベース要求を処理する DB2 アプリケーション・サーバーの場合は、以下の情報を検査してください。

- アプリケーション・サーバーがローカル通信サブシステムに定義されていること。
- 可能性のある 2 次サーバーの宛先がそれぞれ定義されていること。
- 必要なセキュリティーが整っていること。

リモート作業単位に必要な DB2 接続の詳細については、*DRDA 接続の手引き*を参照してください。

DB2 (VM 版) 接続の検査

DB2 (VM 版) からリモート・システムに接続するためには、リモート・システムが定義されている (それらの LU 名は登録されている) ことを確認する必要があります。

DB2 (VM 版) アプリケーションがリモート・システムからデータを要求すると、DB2 は CMS 通信ディレクトリーを検索して、該当するリモート・システムに関する情報を見つけます。以下の項目が、DB2 で使用可能かどうかを検査する必要があります。

- ゲートウェイ名 -- ローカル論理装置 (LU) 名
- リモート LU 名
- リモート・トランザクション・プログラム名 (TPN)
- アプリケーション・サーバーが必要とする会話セキュリティー・レベル
- アプリケーション・サーバーでアプリケーション・リクエスターを識別するユーザー ID
- アプリケーション・サーバーでアプリケーション・リクエスターを許可するパスワード
- アプリケーション・サーバーと通信するために使用するセッション特性を記述するモード名

- リモート・データベース名 (DBNAME)

分散データベース要求を処理する DB2 アプリケーション・サーバーの場合は、以下の情報を検査してください。

- アプリケーション・サーバーがローカル通信サブシステムに定義されていること。
- 必要なセキュリティーが整っていること。

リモート作業単位に必要な DB2 (VM 版) 接続の詳細については、*DRDA 接続の手引き* を参照してください。

DB2 (OS/390 版) 以外のロケーションへの QMF OS/390 ユーザーによるアクセスの準備

ユーザーがリモート・ロケーションにアクセスするには、以下に示すことが必要です。

- QMF 制御表をインストールする。
- QMF データベース・パッケージをリモート・ロケーションにバインドする。
- コマンド同義語表を作成する。
- QMF が DPRE コマンドをサポートするように準備する。
- QMF がその他のコマンドをサポートするように準備する。
- 必要ならば、機能キー表を作成する。
- 必要ならば、QMF 管理プログラム制御表を更新する。
- 必要ならば、QMF サーバーに各国語機能をインストールする。

命名規則のヒント: ロケーションに保管されるオブジェクトについて整合性のある命名規則を開発してください。そうすれば、ユーザーは特定のオブジェクトはどれでも、そのロケーションが分かれます。説明したような命名規則を確立し、使用すると次のことが容易に行えます。

- 所定のロケーションの同一アプリケーションの全オブジェクトをリストする。
- すべてのロケーションの同一アプリケーションの全オブジェクトをリストする。

コマンド同義語表の作成

ユーザーにリモートから作動するコマンドを提供するには、リモート・ロケーションにコマンド同義語表を作成する必要があります。コマンドは、ユーザーがログオンした環境で実行します。このため、TSO でログオンしてから、リモ

リモート・データベース接続のカスタマイズ

ート DB2 (VM または VSE 版) データベースに接続したユーザーの場合は、TSO コマンドとして定義されている QMF コマンドのみを使用することができます。

OS/390 以外のロケーションに、Q.COMMAND_SYN_TSO 表を作成し、TSO ユーザーがそのロケーションに接続された場合に使用できるようにします。

この表は、他のコマンド同義語表を作成するのと同じ方法で作成します。Q.PROFILES 表に同義語表の名前を確実に組み込んでください。Q.PROFILES 表の使用上の詳細については、245ページの『Q.PROFILES 制御表の読み取り』を参照してください。

TSO 環境でのリモート・サーバー・コマンド同義語表の例

QMF がワークステーションの Database Server にインストールされている場合、この同義語表は、ユーザー用に提供されます。QMF が DB2 (VM または VSE 版) サーバーにインストールされている場合は、同義語表は提供されません。

493ページの図145 は、TSO コマンド環境をサポートするために、DB2 (VM または VSE 版) リモート Database Server にコマンド同義語表を定義する方法を示しています。

```

CREATE TABLE Q.COMMAND_SYN_TSO
  ("VERB"          CHAR(18)          NOT NULL,
   "OBJECT"        VARCHAR(31),
   "SYNONYM_DEFINITION" VARCHAR(254) NOT NULL,
   "REMARKS"       VARCHAR(254))
  IN DSQTSYN;
COMMENT ON TABLE Q.COMMAND_SYN_TSO IS
  'QMF TSO COMMAND SYNONYM TABLE';
COMMENT ON COLUMN Q.COMMAND_SYN_TSO.VERB IS
  'NAME OF THE VERB';
COMMENT ON COLUMN Q.COMMAND_SYN_TSO.OBJECT IS
  'NAME OF THE OBJECT';
COMMENT ON COLUMN Q.COMMAND_SYN_TSO.SYNONYM_DEFINITION IS
  'DEFINITION OF SYNONYM';
COMMENT ON COLUMN Q.COMMAND_SYN_TSO.REMARKS IS
  'OPTIONAL COMMENTS ABOUT SYNONYM';

CREATE UNIQUE INDEX Q.COMMAND_SYN_TSOX ON Q.COMMAND_SYN_CMS
  ("VERB" ASC, "OBJECT" ASC);

GRANT SELECT ON Q.COMMAND_SYN_TSO TO PUBLIC;
INSERT INTO Q.COMMAND_SYN_TSO
  VALUES('ISPF',NULL,'CMS DSQAEZ1P P('&ALL')',
         'QMF DISPLAY PRINTED REPORT APPLICATION');
INSERT INTO Q.COMMAND_SYN_TSO
  VALUES('ISPF',NULL,'CMS DSQAEZ1P P('&ALL')',
         'QMF ISPF BRIDGE APPLICATION');
INSERT INTO Q.COMMAND_SYN_TSO
  VALUES('BATCH',NULL,'CMS DSQABB11 PNAME(DXYEABMP)',
         'QMF BATCH APPLICATION');
INSERT INTO Q.COMMAND_SYN_TSO
  VALUES('LAYOUT',NULL,'TSO DSQAEL0A',
         'QMF LAYOUT APPLICATION');

```

図 145. コマンド同義語表の例

DPRE コマンドをサポートする QMF の準備

リモート・ユーザーが DPRE コマンドをリモート・ロケーションで出せるようにするためには、TSO プロシージャを DB2 (OS/390 版) 以外の DRDA アプリケーション・サーバーにコピーするか、または CMS プロシージャを DB2 (VM 版) 以外の DRDA アプリケーション・サーバーにコピーする必要があります。CMS 用の DPRE コマンド・プロシージャは Q.DSQAER2P であり、TSO 用は DSQAER1P です。

プロシージャをコピーするには、以下のように行います。

1. 1 つのロケーションに接続する。
2. プロシージャを表示する。
3. 別のロケーションに接続する。
4. プロシージャを保管する。

その他のコマンドをサポートする QMF の準備

現行ロケーションでオブジェクトを使用するコマンドに QMF コマンド同義語を定義するために、現行ロケーションにコマンド同義語表をもっている必要があります。以下のタイプのコマンドは、現行ロケーションでオブジェクトを使用します。

- QMF オブジェクトを参照するコマンド (CONVERT QUERY)
- 表から情報を読み取るコマンド (DRAW)
- 表に変更を加えるコマンド (EDIT TABLE)
- そのロケーションで使用許可されている表をリストするコマンド (LIST TABLES)
- リストを表示するコマンド (LIST TABLES)

一部のコマンドは QMF を実行中のロケーションからプログラムまたはファイルを使用し、リモート・ロケーションにコマンド同義語表が必要となります。リモート・ロケーションで実行されるコマンドのタイプは次のとおりです。

- システムに特有なコマンド
- IMPORT/EXPORT コマンド

このため、ユーザーはそれぞれのロケーションごとにコマンド同義語表が必要となるため、Q.COMMAND_SYNONYMS からのすべてのコマンド（およびこれらによって参照されるオブジェクト）を、コマンドが実行されるデータベースに追加する必要があります。

機能キー表の作成

ローカル・ユーザー用に機能キーをカスタマイズした場合には、機能キー表をリモート・ロケーションにコピーしなければなりません。

表を提供するには、次のようにして行います。

1. 表を表示する。
2. 別のデータベースに接続する。
3. SAVE DATA コマンドをタイプする。
4. 索引を作成する。
5. SELECT 権限を付与する。

QMF 管理プログラム制御表の更新

必要があれば、リモート・ロケーションで QMF 管理プログラム制御表を更新してください。リモート・ロケーションでの管理プログラムの制限は、リモート作業中に使用される制限です。理由は、これらがアクセスされるリソースのためです。

各国語機能の QMF サーバーへのインストール

NLF を使用している場合には、各国語機能はリクエスターとサーバーの両方に必要なため、各国語機能を QMF リモート・ロケーションに追加してください。ユーザーが各国語機能にアクセスするためには、次のことが必要になります。

1. NLF 用に Q.PROFILES を用意する。
2. NLF 用にコマンド同義語表を作成する。

コード・ページ・サポート

DB2 (OS/390 版) と DB2 (VM および VSE 版) は両方とも、異なるシステム上において異なるコード・ページを使用するアプリケーション・リクエスターとサーバーとの間の文字変換を処理することができます。

異種データベースからくる文字ストリングを処理するために、アプリケーション・リクエスターとアプリケーション・サーバーの両方に、CCSID 変換規則を設定する必要があります。適切な CCSID 変換ペアをサーバーに定義して、アプリケーション・サーバーがアプリケーション・リクエスターから送信された文字ストリングを認識できるようにします。

変換の判断は、ストリング全体 (たとえば、CHAR または VARCHAR 列からのデータ) を変換する必要があるかどうかによって異なります。

CHAR または VARCHAR 列については、

- 列が FOR BIT DATA と定義されている場合は、その内容は変換されません。
- 列が FOR SBCS DATA とまたは FOR MIXED DATA と定義されている場合は、その内容は変換されます。

FOR BIT DATA、FOR SBCS DATA、および FOR MIXED DATA パラメータの詳細については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

APPLDATA 列使用上の制約

QMF は (Q.OBJECT_DATA の) VARCHAR 列 APPLDATA を使用して、そのプロシージャ、照会、および書式の定義を保持しています。QMF オブジェクトの定義には、変換してはならない一部のデータが含まれています。そのため、APPLDATA 列は変換可能文字データを含んでいるとは分類されず、FOR BIT DATA として定義されます。

リモート・データベース接続のカスタマイズ

一部の特殊文字の使用回避

QMF の実行とデータ・アクセスが可能なシステム間で変換が必要な文字の数は、アプリケーション・リクエスターおよびサーバーのロケーションで使用されるコード・ページによって異なります。変換が必要な文字のリストには、リクエスターのコード・ページとサーバーのロケーションのコード・ページ間で変換が必要なすべての文字の他に、否定符号 (¬) および縦線 (|) が含まれています。

QMF オブジェクト内の文字は変換されないため、アプリケーション・リクエスターおよびアプリケーション・サーバー内のコード・ページが異なる場合は、¬ および | の使用は避けてください。

ユーザーがリモート・データベースにアクセスできるようにする

QMF 6 は、異なるデータベース間でのリモート作業単位アクセスをサポートします。ユーザーは、異なる DB2 (VM または VSE 版) データベース間、異なる DB2 (OS/390 版) データベース間、あるいは DB2 (VM または VSE 版) データベースと DB2 (OS/390 版) データベースの間で移動することができます。

ユーザー・プロファイルの更新

ユーザーがリモート・ワークステーションにアクセスする必要がある場合は、ユーザーの Q.PROFILES 表を更新しなければなりません。Q.PROFILES 表を、242ページの『QMF へのユーザー・アクセスを可能にするユーザー・プロファイルの作成』の説明に従って更新してください。

プロファイル値には、QMF セッションの属性 (たとえば、照会タイプおよび LANGUAGE パラメーター) と、現行ロケーションに関連するもの (たとえば、SPACE パラメーター) とがあります。

特定のユーザーのために (CMS、TSO、SRPI、APPC、ネイティブ OS/390、または CICS からのアクセス用)、単一のプロファイル表に別々の行を設定することができます。これを行うには、ENVIRONMENT 列を用いて QMF 操作環境に適用される値を指定します。この場合、指定する値は、プロファイルが保管されているロケーションに固有のものである必要があります。

現行 SQL 許可 ID へのアクセスの指定

ユーザーの CURRENT SQLID は、異なるロケーションに接続した後は無効です。したがって、単一の QMF セッションから複数の DB2 アプリケーション・サーバーに対し同じ CURRENT SQLID を使用しなければならない場合

は、各サーバーへ接続した後に、CURRENT SQLID をリセットしなければならないことがあります。詳細については、*QMF 解説書* を参照してください。

Workstation Database Server のユーザー

CURRENT SQLID は DB2 Common Server では使用できません。

ローカル・データベースへの接続

QMF が、TSO、SRPI、APPC、ネイティブ OS/390、または CICS のもとで、データベースに接続されていることを検査する必要があります。

TSO、SRPI、APPC またはネイティブ OS/390 からの接続

QMF は呼び出し接続機能 (CAF) を介して DB2 にブリッジされます。開始プログラム・パラメーター DSQSSUBS は、QMF が CAF CONNECT を実行する際に使用する DB2 サブシステム ID (『ローカル DB2』) を指定します。DSQSSUBS パラメーターは、221ページの『DSQSSUBS (QMF で使用される DB2 サブシステムの命名)』でも説明されています。

CICS からの接続

QMF は CICS/DB2 接続機能を介して DB2 にブリッジされます。CICS トランザクションと DB2 の関係は、リソース管理テーブルに記述されます。この表は、接続先の DB2 サブシステム (「ローカル DB2」) を判別します。

リモート・データベースへの接続

QMF セッションからリモート・ロケーションへの、さまざまな接続方式をユーザーに提供することができます。以下の方式のうち、1 つまたは複数を設定することができます。

- プログラム・パラメーター DSQSDBNM の使用

このパラメーターは、QMF セッションの初期化時にリモート・ロケーションに接続する場合に使用してください。このパラメーターは、アプリケーション・リクエスターが DB2 (OS/390 版) であり、アプリケーション・サーバーが DB2 (VM または VSE 版) の場合に、(始動時に) 必要になります。DSQSDBNM パラメーターの詳細については、220ページの『DSQSDBNM (QMF 開始時の接続先のロケーションの指定)』を参照してください。

- QMF CONNECT コマンドの使用

このコマンドを使用して、QMF セッション中にリモート・ロケーションに接続します。

リモート・データベース接続のカスタマイズ

このコマンドにより、QMF セッション中に、分散ネットワーク内の別のロケーションに接続することができます。

このコマンドは、次のものから出すことができます。

- 呼び出し可能インターフェースまたはコマンド・インターフェース
- コマンド行から
- プロシージャ (線形プロシージャまたはロジックを持つプロシージャ) 内から

QMF CONNECT コマンド、コマンド・パラメーター、およびコマンド使用上の考慮事項の詳細については、*QMF 解説書* を参照してください。

プロシージャおよび呼び出し可能インターフェース、あるいはコマンド・インターフェースの詳細については、*QMF アプリケーション開発の手引き* を参照してください。

- また、接続を確立するプロシージャを作成し、そのプロシージャを実行するためにコマンドをコマンド同義語表に追加することも可能です。これによって、ユーザーはそのコマンドを入力して、リモート・データベースに接続することができます。

CICS におけるリモート・データベースでの実行

DB2 バージョン 2 リリース 3 で実行しているときに、QMF が CICS で実行中であり、かつ現行ロケーションがローカル DB2 ではない場合には、QMF からのデータベースへの参照は、すべて読み取り専用でなければなりません。そのため、データベースを変更しようとする、すべて拒否されます。そのような例としては、以下のようなものがあります。

- 変更を引き起こすコマンド (たとえば、SAVE *object*)
- 変更を引き起こす SQL または QBE 照会 (たとえば、INSERT、および I)
- Q.ERROR_LOG へのエラー・ログ

データベースを変更しようとする、

- コマンドまたは照会を出した場合には、データが読み取り専用であることを示すメッセージを受け取ります。
- ログに記録する必要があるエラーが生じた場合、DSQDEBUG データ・セットにのみログが記録され、Q.ERROR_LOG にはログが記録されません。

ロケーション名の指定

QMF は SQL を使用して、リレーショナル・データベースにアクセスします。リモート作業単位では、アプリケーション・リクエスターが CONNECT 要求を取り上げ、リモート・データベース管理システムと接続を確立します。分散

作業単位では、アプリケーション・リクエスターが SQL 要求を“受け取り”、それを適切な分散作業単位サーバーに経路指定します。

QMF では、ロケーション名 という用語を使用して、接続先の DRDA アプリケーション・サーバーを表します。ロケーション名を使用して、リモート・データベースに接続することができます。このロケーション名を使用して、表名を修飾することもできます。たとえば、SAN_JOSE.JONES.TABLE5 という名前の SQL 表は、ロケーション名が SAN_JOSE というデータベース管理システム (DBMS) によって管理されます。

DB2 (OS/390 版) の場合

ロケーション名はサブシステム全体を参照します。DB2 サブシステムをアクセス可能なサーバーが、通信データベースで定義されます。

3 部分名を使用している場合: リモート作業単位と分散作業単位の両方を使用するときは、3 部分名を使用してアクセスできるロケーションは、DB2 ロケーションである現行アプリケーション・サーバーからアクセスすることができます。

Workstation Database Server の場合

ロケーション名は単一のデータベースを参照します。

DB2 (VM または VSE 版) の場合

ロケーション名は DB2 データベース・マシン全体を参照し、CMS 通信ディレクトリーにカタログされます。

ユーザー・アクセスのためにデータのあるべき位置

DISPLAY TABLE 表名 などのデータをアクセスするコマンドおよび照会は、現行ロケーションへ宛先指示されます。ただし、現行ロケーションが DB2 で表名 が現行ロケーション以外の DB2 サブシステムを参照している 3 部分名 (またはその名前の別名) の場合は除きます。

QMF オブジェクトでの処理

データベースから検索したり、データベースに保管する QMF オブジェクト (照会、プロシージャ、および書式) は、現行ロケーション、すなわち接続されているロケーションに存在している必要があります。このことは、リモート作業単位サポートのない QMF の場合も同様であり、オブジェクトは 3 部分名を使わずにアクセスされる表と同じロケーションにあります。

リモート・データベース接続のカスタマイズ

データ、プロシージャ、照会、および書式を現行ロケーションに保管する際に、十分な記憶域が確保できるようにするために、現行ロケーションで次のことを行うようにしてください。すなわち、デフォルトの表スペース (DB2 の OS/390 版のロケーションに接続している場合) またはデフォルトの DB スペース (DB2 の VM 版または VSE 版のロケーションに接続している場合)、および QMF オブジェクト表 (Q.OBJECT_DIRECTORY、Q.OBJECT_DATA、および Q.OBJECT_REMARKS) を日常的にモニターします。

データベースに保管されている QMF オブジェクトの中には、QMF に呼び出されなければならないプログラムを参照しているものもあります。これらのプログラムには次のものがあります。

- ユーザー編集ルーチン
- ロジックおよび表計算を持つプロシージャをサポートしている REXX プログラム
- ローカルの日付・時刻編集コードのサポートで呼び出されるローカル日時ルーチン
- CMS、CICS、または TSO コマンドを含んでいる QMF プロシージャ
これらのプロシージャを作成する際には、CMS コマンドを TSO 環境で実行しようとしたり、TSO プロシージャを CMS 環境で実行したりすることのないようにする必要があります。
- プログラムが、呼び出し可能インターフェースまたはコマンド・インターフェースを介して、QMF コマンドを実行できるようにする EXEC または CLIST

これらのプログラムは、QMF が実行中のシステムと同じシステム (ログオンするシステム) にある必要があります。その理由は、これらのプログラムには、オペレーティング・システムのコマンドを含んでいる可能性があり、そのシステムでは正常に稼動しないか、あるいは期待どおりの結果を伴わない恐れがあるためです。その結果、そのシステムはデータベース (すなわち、QMF オブジェクト) のあるシステムとは異なったものになる可能性があります。この例については、*QMF* 使用の手引き および *QMF* アプリケーション開発の手引きを参照してください。

アプリケーション・リクエスターとアプリケーション・サーバーがコード・ページの異なるシステム上にある場合は、QMF オブジェクトを使用しても構いません。重要な制約事項については、495ページの『コード・ページ・サポート』を参照してください。

表での処理

データ定義ステートメントでは、3 部分名を使用できません。ただし、リモート作業単位のあるロケーションに最初に接続する場合には、そのロケーションで CREATE や GRANT などのデータ定義ステートメントを出すことができます。GRANT 文節 PUBLIC AT ALL LOCATIONS を使用することで、現行サーバーに常駐する表の特権を他のロケーションのユーザーに付与することができます。

SQL エラーの防止

SQL のエラーは、一般に、操作環境と同種の DBMS でサポートされている SQL と、異種 DBMS に接続する場合に使用する必要のある SQL との相違に関連しています。このようなエラーを避けるためには、アプリケーション・サーバーでサポートされている SQL を使用することを忘れないでください。

たとえば、QMF が OS/390 で実行中で、現行ロケーションが DB2 (VM 版) の場合、SQL 照会で使用する構文は、DB2 (VM 版) でサポートされている構文でなければなりません。CREATE TABLE コマンドの IN 文節を使用する場合には、次の (DB2 (VM 版)) 構文が受け入れられます。

```
CREATE TABLE ... IN DSP3
```

DB2 (VM 版) の次の対応する構文は受け入れられません。

```
CREATE TABLE ... IN DATABASE DSQ3
```

SQL ステートメント完了情報 (SQLCODE) には、現行ロケーションから戻された情報が入っています。OS/390 で QMF を使用すると、SQL 構文および DB2 から戻される情報をよく使い慣れていることになります。

QMF 照会の移植性が必要であれば、SAA でサポートされる SQL 構文を使用してください。これによって、DRDA アプリケーション間での移植性がかなり保たれます。

データベース管理上の以下の課題が QMF に影響を与えることがあります。

- 一部のアプリケーション・リクエスター環境では、アプリケーションに使用できる SQL ステートメントが制限されることがあります。たとえば、CICS で実行中の DB2 アプリケーション・リクエスターは、リモート・アプリケーション・サーバーでリソースを更新することはできません。
- アプリケーション・プログラムは、ネットワーク内で表、視点、および QMF オブジェクトがある個所を知っておく必要があります。リモート作業単位は、単一の作業単位内の単一のデータベース管理システムに限られるためです。

リモート・データベース接続のカスタマイズ

制約事項: 別のロケーションに保管されている 2 つの表に結合を発行することはできません。

- QMF が稼働中のロケーションにあるユーザー編集ルーチンおよび管理プログラム出口モジュールが、QMF セッションで使用されます。
- あるロケーションに接続した後で、プロファイル、リソース管理テーブル、同義語、および機能キーがそのロケーションの値に初期化されます。

ユーザー ID の変換

ユーザーを別のロケーションに接続すると、データベース・ユーザーのユーザー ID が別のユーザー ID に変換されることがあるため、データベース管理者は変換されたユーザー ID を定義する必要がある場合があります。

名前の変換

1 つのロケーション (DB2 (OS/390 版) または DB2 (VM または VSE 版)) に接続すると、1 次ユーザー ID が変換されることがあります。変換がどのように設定されるかについては、DBMS の資料を参照するか、データベース管理者に問い合わせてください。

名前変換の必要性: 1 文字から 8 文字までのユーザー ID は、特定のオペレーティング・システム内では固有でなければなりません。システム・ネットワーク体系 (SNA) のネットワークでは固有ではなくても構いません。

命名上の矛盾を避けるために、分散データベース・システムでは、以下の 名前変換 の方式をサポートしています。

アウトバウンド名前変換

アプリケーション・リクエスターは、エンド・ユーザーの名前を SNA ネットワークの宛先に送信する前に変換することができます。

インバウンド名前変換

アプリケーション・サーバーは、エンド・ユーザーの名前を SNA パートナーから受信する前に、変換することができます。

DSQSPRID (TSO の場合)

DB2 のロケーションに接続する場合、基本のデータベース・システムが強制的に名前変換を行う可能性があります。これによって、システムの 1 次許可 ID が影響を受けます。したがって、接続先のロケーションで登録が制限されている場合は、接続されたシステムの 1 次許可 ID が、そのロケーションの Q.PROFILES 表の 'CREATOR =' 値と対応していることを確認してください。

ほとんどの場合、1 次許可 ID は Q.PROFILES のプロファイル行を選択するために使用した CREATOR 列の値です。ただし、DSQSPRID パラメーター値を TSOID にして QMF を開始する場合には、CREATOR 列は TSO ログオン ID でなければなりません。

DSQSPRID に使用する値: TSO から QMF を使用してリモート・ロケーションに接続する場合、Q.PROFILES からの行の読み取りには、TSO ログオン ID ではなく、1 次許可 ID を使用します。DSQSPRID プログラム・パラメーター値には PRIMEID を使用します。(PRIMEID は、QMF の開始時に DSQSPRID プログラム・パラメーター値を指定しない場合のデフォルト値です)。その後、変換が生じると、QMF はリモート・ロケーションでの 1 次許可 ID に基づいて正しいプロファイル行を選択します。

TSOID の DSQSPRID 値を使用すると、QMF は (QMF が稼働中のシステム) TSO ログオン ID を使用して Q.PROFILES を読み取ります。TSOID を使用しなければならない場合には、CREATOR 列の値が TSO ログオン ID に設定されたプロファイル行が必要です。(プログラム・パラメーター DSQSPRID の詳細については、230ページの『DSQSPRID (TSO プロファイル・キーの指定)』を参照してください。

制御表 Q.PROFILES の詳細については、246ページの表31 を参照してください。

重要: TSOID の DSQSPRID 値は、主に 1 次許可 ID が TSO ログオン ID と異なるユーザーの、以前の QMF リリースからの移行用に使用されません。DSQSPRID の推奨値は、PRIMEID です。

各リモート QMF ロケーションからの QMF ユーザーの削除

ローカル QMF からユーザーのオブジェクトを削除した後、そのユーザーがアクセスする各リモート・ロケーションからそのユーザーのオブジェクトを削除する必要があります。他のリモート・ロケーションの QMF 管理者である場合は、ローカル・オブジェクトを削除したのと同じ方法でリモート・オブジェクトを削除することができます。リモート・ロケーションの QMF 管理者ではない場合、他の管理者にそのユーザー ID のオブジェクトを削除するよう要求する必要があります。

自分のロケーションに管理者がアクセスできるようにする

各 QMF ロケーションごとに異なる QMF 管理者がいる場合には、他の管理者が自分のロケーションにアクセスして、その管理者のロケーションからリモート作業単位を設定できるようにしなければなりません。

次のことを行う必要があります。

- 管理者に対し、QMF リクエスター制御表の選択権限を付与する。
- QMF リクエスターに管理プログラム・ルーチンをインストールし、リモート・ユーザーがすべてのリソースを使用することがないようにする。

第23章 バッチ・プログラムとして実行するための QMF のカスタマイズ

QMF 管理者として、バッチ・モード・ユーザーに助言と援助を与えなければならないことがあります。また、自分のプロシーチャーをバッチ・モードで実行したい場合もできます。この章では、TSO、ISPF、ネイティブ OS/390、または CICS での QMF バッチ・モードの使用方法を説明します。ISPF の場合、QMF バッチ機能は TSO 端末モニター・プログラム (TMP) で QMF を実行します。

ユーザーがバッチ・モードを使用できるようにするには、ユーザーに適切な権限を与える必要があります。そうすれば、ユーザーはバッチ・モードを使用して、セッションとは独立してプロシーチャーを実行することができ、プロシーチャーの実行中にコマンドを対話式に発行することができます。バッチ・プロシーチャーはすぐには実行されないことがあります。ユーザーの QMF セッションが終了するのを待ってから実行することになります。

管理者およびユーザーは、実行してデータベースに保管するプロシーチャーを作成することができます。プロシーチャーでは、照会や他のプロシーチャーを呼び出したり、他の大部分のコマンドを実行することができます。バッチ・プロシーチャーの作成の詳細については、*QMF 使用の手引き* を参照してください。

また、QMF には、TSO でのバッチ・ジョブの実行を単純化するための QMF BATCH アプリケーションも用意されています。このアプリケーションの詳細については、514ページの『QMF バッチ照会 / プロシーチャー・アプリケーション (BATCH) の ISPF での使用』を参照してください。

NLF を使用する場合: 複数の言語をインストールしているユーザーは、対話式セッションの場合と同様に、バッチ QMF セッション用に言語環境を選択することができます。

クイック・スタート

表59 は、バッチ処理プログラムのカスタマイズの方法について要約していません。

バッチ・プログラムとして実行するためのカスタマイズ

リストしている作業に関する詳細については、表の右端に示しているページを参照してください。

表 59. バッチ処理プログラムのカスタマイズ

作業	参照箇所
ユーザーが TSO バッチ・モードで実行できるようにする。このためには、RACF が有効な場合はユーザーのログオン ID および DB2 許可 ID を許可するか、または PROFILE PREFIX ステートメントを使用してそれらの ID を許可する。	506 ページ
TSO バッチでジョブを実行するように送信する。このためには、スプールを使用して、指定されたバッチ・マシンにバッチ・ジョブを処理させる必要があります。	508 ページ
JCL を使用して TSO でバッチを実行する。このためには、次のようなステートメントでジョブを開始する必要があります。 <code>//BATCH JOB USER=LMN,PASSWORD=ABC,NOTIFY=LMN</code>	509 ページ
バッチ・プロシージャーを TSO でデバッグする。このためには、541ページの『診断援助機能を使用した問題の判別』で説明している L1 および L2 トレース・コードを使用します。	513 ページ
QMF BATCH アプリケーションを TSO で使用する。このためには、ユーザーが TSO FIB コマンドを使用する権限を持っているか確認する必要があります。	514 ページ
QMF をネイティブ OS/390 で実行する。ここで用意されている JCL を使用します。	525 ページ
ユーザーが CICS でバッチ・モードを実行できるようにする。このためには、 Q.PROFILES 表を使用してユーザーの QMF アカウントを許可します。	527 ページ
バッチ・プロシージャーを CICS でデバッグする。このためには、 QMF コマンドとメッセージ・トレース機能を使用し、一時データ・キューに情報を経路指定します。	513 ページ

ユーザーが **TSO** でバッチ・モードを使用できるようにする

この節では、**TSO** からバックグラウンド・ジョブを実行する方法を知ることが前提となります。フォアグラウンドで開始されたバックグラウンド・ジョブの実行の詳細については、**TSO 拡張機能** コマンド言語解説書を参照してください。

ユーザーが **RUN** コマンドでプロシージャーを実行した場合には、プロシージャーまたはセッションを取り消す以外には、**QMF** コマンドを実行することはできません。そのため、**RUN** コマンドを使用してプロシージャーを実行すると、かなりのセッション時間を占有してしまう恐れがあります。

その代わりに、適切な **TSO** 権限があれば、ユーザーはプロシージャーをバッチ・モードで実行することができます。このモードでは、プロシージャーはユーザーのセッションからは独立して実行されるため、ユーザーはコマンドを出し続けることができます。

適切な権限のある TSO ユーザーは、ジョブをバックグラウンドに実行依頼することができます。このためユーザーは、これらのジョブの実行を待たずに、セッションを継続することができます。そのようなジョブでは、QMF プロシージャーをバッチ・モードで実行することができます。したがって、バックグラウンド・ジョブの準備と実行は、バッチ・モードでの QMF プロシージャーの実行には不可欠な部分です。

バッチ・モードでの操作権限

バッチ・ジョブを実行依頼するためには、どの QMF 権限および DB2 権限が必要かを知っている必要があります。

ジョブが実行されているログオン ID および DB2 1 次許可 ID を判別します。

- インストール先システムで RACF を使用している場合は、ログオン ID は JOB ステートメントの USER パラメーターの値です。DB2 1 次許可 ID は、ログオン ID に対応するものです。
- インストール先システムで RACF を使用していない場合には、ログオン ID および 1 次許可 ID は 511 ページの『PROFILE PREFIX ステートメント』に説明されているように決められます。

ログオン ID と許可 ID は、QMF を対話式で使用する場合と同じ役割を果たします。その結果、プロシージャーは以下の条件が満たされている場合のみ実行されます。

- バッチ実行のログオン ID を使用して、QMF を対話式に操作できる。
- ログオン ID に対応する許可 ID が実行プロシージャーを持っている、もしくはそのプロシージャーを共有している。

プロシージャーのコマンドを実行する際、許可 ID は対話式に機能します。ただし、対話式で実行できるコマンドがすべてバッチ・モードで実行できるわけではありません。バッチ環境に適しているコマンドの詳細については、*QMF 使用の手引き* を参照してください。

QMF を対話式に使用する権限を持っているユーザー、およびジョブをバックグラウンドで実行することができるユーザーは、それをバッチ・モードでも使用することができますが、権限を持っていないユーザーはそれをバッチ・モードで使用することはできません。

RACF セキュリティーの考慮事項

RACF がセキュリティの一部であれば、あるユーザーが他のユーザーのログオン ID のもとでジョブを実行するのを防止することができます。そのようなジョブを実行するユーザーは、自分が見ることを許可されていないデータも含めて、他のユーザーがアクセスできるすべての DB2 データにアクセスすることができます。

TSO SUBMIT コマンドによる OS/390 へのジョブの送信

管理者もしくはユーザーは、実行するプロシージャーを作成し、それをデータベースに保管する必要があります。プロシージャーには、照会を出したり、他のプロシージャーを実行するもの、あるいは他の大部分のコマンドを実行するものもあります。QMF の TSO コマンドを介して、プロシージャーで CLIST やオンライン・プログラムを呼び出すものもあります。バッチ用プロシージャーの作成の詳細については、*QMF 使用の手引き* を参照してください。

プロシージャーを保管した後、管理者またはユーザーはそのプロシージャーを実行するジョブの JCL ファイルを作成する必要があります。このジョブの JCL は、バッチ操作の TSO を呼び出します。この JCL は、TSO が実行するステートメントの入ったデータ・セットを含めて、TSO および QMF が必要とするリソースを割り当てるものでなければなりません。そのステートメントのうちの 1 つによって、QMF セッションを開始する必要があります。

JCL 一般については、509ページの『TSO で QMF バッチ・ジョブ実行のための JCL』を参照してください。バッチ・モードの QMF セッションの開始の詳細については、179ページの『第13章 QMF の開始』を参照してください。

TSO SUBMIT コマンドを使用して、ジョブをバックグラウンドに実行依頼します。SUBMIT は、FIB (フォアグラウンド開始のバックグラウンド) コマンドの 1 つであり、これによってユーザーはバックグラウンド・ジョブの実行、モニター、および操作を行います。FIB コマンドを出すためには、適切な TSO 権限が必要です。(その権限を与えるのは、TSO の管理タスクです。) FIB コマンドとその使用法の詳細については、*TSO 拡張機能コマンド言語解説書* を参照してください。

SUBMIT コマンドは、次の場合に実行することができます。

- ユーザーの QMF セッション中に、QMF の TSO コマンドを使用
- TSO READY モード、またはジョブの JCL に合わせた CLIST

パラメーターに基づいて合わせる事が可能であり、呼び出された CLIST にその値が渡されます。

バッチ・プログラムとして実行するためのカスタマイズ

プロシージャーの実行中にエラーが起これると、以下のような結果になります。

- プロシージャーの終了
- 非コミットの DB2 回復単位のバックアウト

JOB ステートメントで、ジョブの終了時にユーザーへメッセージを送信するように指定することができます。メッセージはユーザーの画面に表示されます。ユーザーはメッセージを受け取るために、QMF セッションを終了する必要はありません。

実行が終了すると、ユーザーは印刷出力でエラーを調べることができます。適切な JCL を使用すれば、ユーザーはこの出力をデータ・セットに経路指定し、編集プログラムで調べることができます。このデータ・セットの 1 つに、確認メッセージとエラー・メッセージのレコードを入れ、必要であれば、実行した QMF コマンドのレコードを入れることも可能です。このデータ・セットとその内容の制御方法の詳細については、513ページの『プロシージャーのデバッグ』を参照してください。

TSO で QMF バッチ・ジョブ実行のための JCL

QMF はバッチ・モードの TSO を介してバッチ・モードで実行されるため、バッチ・ジョブの JCL は TSO ログオンの JCL に非常によく似ています。この節では、バッチ・モードで使用できる JCL ステートメントについて説明します。

JOB ステートメント

次に示すように、JCL は JOB ステートメントで開始します。

```
//BATCH JOB USER=LMN,PASSWORD=ABC,NOTIFY=LMN
```

ここに示したステートメントには、アカウント情報もユーザー名も入っていないため、すべてのインストール先システムに適しているわけではありません。ここで示しているオペランドでは、次の指定を行っています。

- ログオン ID は LMN です。
- ログオン・パスワードは ABC です。
- 端末メッセージがジョブ終了時にユーザー LMN に送信されます。

その他のオペランドも指定することができます。その他のオペランドとしては、詳細レベルおよび JCL とシステム・メッセージの経路指定を制御する MSGLEVEL や MSGCLASS などがあります。

考慮事項: RACF を使用していない場合は、PASSWORD パラメーターは無視され、機密漏れの危険が生じます。

バッチ・プログラムとして実行するためのカスタマイズ

EXEC ステートメント

JOB ステップに EXEC ステートメントを使用して、次のようにバッチ・モード QMF を実行することができます。

```
//SAMPLE EXEC PGM=IKJEFT01,TIME=1440,DYNAMNBR=30,REGION=3072K
```

このステートメントでは、

- TSO を呼び出します (PGM=IKJEFT01)。
- 許される、適切な数の的割り振りを指定します (DYNAMNBR=30)。
- QMF に十分な大きさの領域を指定します (REGION=3072K)。

DD ステートメント

QMF を対話式で実行する場合とバッチ・モードで実行する場合の両方に、同じ DD ステートメントを使用することができます。SYSPRINT、SYSTEMR、および SYSIN のステートメントは除去する必要があります。

1 つまたは複数の SYSOUT DD ステートメントに、HOLD=YES のオペランドを追加すれば、その出力を TSO の OUTPUT コマンドで操作することができます。(これは別の FIB コマンドです。) OUTPUT コマンドを使用して、SYSOUT DD ステートメントの出力を画面に経路指定することができます。

また、SYSTSPRT および SYSTSIN データ・セットの DD ステートメントも必要になります。

SYSTSPRT: このデータ・セットには、TSO および ISPF からのメッセージ出力が入っています。このデータ・セットは、次のように指定します。

```
//SYSTSPRT DD SYSOUT=A
```

SYSTSIN: SYSTSIN には、ジョブ・ステップ中に実行される TSO ステートメントが入っています。これらのステートメントを JCL に入れるには、次のように指定します。

```
//SYSTSIN DD *  
EXEC CLISTA  
PROFILE PREFIX(LMN)  
ISPSTART PGM(DSQMFE) NEWAPPL(DSQE) PARM(DSQSMODE=B,DSQSRUN=LMN.PROCA)  
  
:  
  
/*
```

図 146. SYSTSIN からの TSO ステートメントの追加

TSO はこれらのステートメントを、以下のように SYSTSIN にある順番で実行します。

- 最初のステートメントが CLISTA という名前の CLIST を実行し、これによって QMF ライブラリーの割り振りを行います。
- 2 番目のステートメントは、ユーザーの DS 名接頭部を LMN に設定します。
- ISPSTART ステートメントは、ISPF でバッチ・モード QMF を呼び出し、プロシージャー LMN.PROCA を実行します。

バッチ・モードでの ISPSTART コマンドの使用の詳細については、185ページの『ISPF でバッチ・モードによる QMF の開始』を参照してください。

PROFILE PREFIX ステートメント

510ページの図146 における PROFILE PREFIX ステートメントは、ユーザーの DS 名接頭部を LMN に設定しています。この例では、これをユーザーのログオン ID と想定しています。

ステートメントを置く位置: PROFILE PREFIX ステートメントは、QMF を開始させる最初の ISPSTART ステートメントの前に置きます。QMF の内部で PROFILE PREFIX ステートメントを出しても無効です。

PROFILE PREFIX によるプロファイルの変更: QMF SET PROFILE コマンドは、それ自体では、ユーザーの QMF プロファイルを永続的に変更することはありません。それに対して、PROFILE PREFIX ステートメントは、インストール先システムの設定次第で、ユーザーの TSO プロファイルを永続的に変更することができます。これを行った場合、ユーザーは後で DS 名接頭部を復元したい場合もあると思われます。接頭部設定の初期値は、ISPF システム変数 ZPREFIX に入っています。

PROFILE PREFIX の有効化: PROFILE PREFIX ステートメントを有効にするためには、DSQSPRID パラメーターは TSOID に設定しておく必要があります。以下の理由から、同様のステートメント (ユーザーの接頭部をユーザーのログオン ID に設定するようなステートメント) が、バッチ・モードで QMF を実行する他のジョブにも必要な場合があります。

- RACF を使用していない場合のユーザー識別
RACF を使用していないインストール先システムの場合、QMF はユーザーのログオン ID がユーザーの DS 名接頭部と等しいものと見なします。この接頭部がヌルの場合には、QMF は ログオン ID を BATCH と見なしま

バッチ・プログラムとして実行するためのカスタマイズ

す。このように、DS 名接頭部をユーザー・ログオン ID に設定することにより、PROFILE PREFIX ステートメントが QMF にユーザーのログオン ID を与えます。

この場合、DB2 がユーザーに割り当てる 1 次許可 ID は、DB2 インストール・パラメーターの UNKNOWN AUTHID で指定された値です。ログオン ID は、DSQDEBUEG データ・セットに記録されるトレース出力で使用されます。DSQSPRID パラメーターの設定次第で、1 次許可 ID もしくはログオン ID のいずれかが、プロファイルからの読み取り、およびデフォルトのリソース・グループの割り当てに使用されます。このパラメーターの説明については、199ページの『第14章 開始手順のカスタマイズ』を参照してください。

- データ・セット名に伴う問題の回避

QMF プロシージャが QMF IMPORT/EXPORT コマンドで、完全修飾された書式と不完全な書式の両方のデータ・セット名を使用していると、問題を生じることがあります。たとえば、ログオン ID LMN のもとで実行するプロシージャが、次の 2 つのコマンドを出すとします。

```
EXPORT QUERY TO 'LMN.QUERYX.QUERY'  
IMPORT QUERY FROM QUERYX
```

EXPORT コマンドはログオン ID (LMN) をエクスポート・ファイル名の最初の修飾子に使用しています。その後、IMPORT がこのファイルをインポートします。

ユーザーの DS 名接頭部が LMN ではなく ABC であれば、IMPORT ステートメントで参照されるファイルは、LMN.QUERYX.QUERY ではなく、ABC.QUERYX.QUERY となります。これは、この例の IMPORT コマンドのように名前が完全に修飾されていない場合には、データ・セット名の最初の修飾子には接頭部が使用されるためです。

そのため、このプロシージャでは、前にエクスポートしたファイルを見つけることができません。PROFILE PREFIX ステートメントは、DS 名接頭部をユーザーのログオン ID (この場合は、'LMN') に設定することによって、この問題を回避しています。

TSO または ISPF 使用によるフォアグラウンドでの QMF バッチの実行

QMF をフォアグラウンドで、バッチ・モードで開始させるためには、179ページの『第13章 QMF の開始』で説明されている QMF の開始方法のいずれも使用することができます。たとえば、TSO READY モードから、次のステートメントを出して、QMF を CLIST から開始させることができます。

バッチ・プログラムとして実行するためのカスタマイズ

```
ISPSTART CMD(clist_name) NEWAPPL
```

ここで、clist_name は、QMF を開始させる CLIST の名前です。この CLIST は、次の書式のステートメントを持っている必要があります。

```
ISPEXEC SELECT PGM(DSQMFE) NEWAPPL(DSQE)  
                PARM(...DSQSMODE=B,DSQSRUN=aaa.bbb)
```

ここで、ISPSTART ステートメントは、バックグラウンドではなく、フォアグラウンドで実行されます。CLIST の終了するのを待っている間、TSO で他のことを行うことはできません。

CLIST が実際に終了すると、TSO READY モードに戻ります。TSO コンソール、ログ、およびリスト・ファイルに対して、永続後処理パラメーターを指定する前にプロシージャが終了すると、CLIST の終了前に、ISPF 後処理プロンプト・パネルが表示される場合があります。このパネルが表示されないようにするには、これらのファイルに対して永続後処理パラメーターを指定しておきます。それぞれに対して、D (「削除」の指定) の値を指定すれば、たいいていの場合、十分です。この後処理の指定方法が分からない 場合には、ISPF の専門家に尋ねるか、ISPF ヘルプを使用してください。

プロシージャのデバッグ

トレース・コードと HELP コマンドを使用して、バッチ・モードのプロシージャを診断することができます。実際に、バッチ・モードで実行するプロシージャには、L2 トレースがデフォルトになっています。トレースの設定を変更するには、プロシージャで SET コマンドが必要になります。たとえば、L2 の代わりに L1 トレースを指定する場合は、プロシージャの先頭に次のステートメントを追加します。

```
SET PROFILE (TRACE=L1
```

L1 もしくは L2 トレースのいずれの場合も、ログは DSQDEBUG データ・セットに生成されます。このログ内に一連のメッセージ・レコードが入り、プロシージャの実行中に QMF が出したメッセージごとに、1 つのレコードが書かれます。

また、L2 トレースが有効な場合には、ログには、プロシージャ (およびその従属部分) が実行した QMF コマンドごとに、それぞれ 1 つのレコードも書かれます。

バッチ・プログラムとして実行するためのカスタマイズ

プロシージャがまだ終わるべきでないときに終了してしまった場合は、エラー・メッセージが DSQDEBUG データ・セットに書き込まれます。HELP コマンドを使用して、該当するメッセージ・ヘルプ・パネルを表示することができます。

対話式セッションの問題診断の詳細については、541ページの『診断援助機能を使用した問題の判別』を参照してください。

QMF バッチ照会 / プロシージャ・アプリケーション (BATCH) の ISPF での使用

QMF バッチ照会 / プロシージャ・アプリケーションは、バッチ・モードで照会やプロシージャを実行するために必要な知識と手間が最小限で済むように考えられたものです。このアプリケーションを使用するには、QMF を ISPF のもとで開始する必要があります。

NLF を使用する場合: ユーザーに変換された同義語を割り当てる必要があります。これによって、ユーザーは BATCH に対応するコマンド同義語を出すことになります。同義語の割り当ての詳細については、343ページの『第18章 QMF コマンドのカスタマイズ』を参照してください。

アプリケーション使用権限の割り当て

このアプリケーションは、共用プロシージャを実行することによって開始させることができるため、QMF ユーザーは誰でも使用することができます。このアプリケーションでは、ユーザーのバッチ・ジョブ用のプロシージャと JCL を作成します。ただし、ユーザーが TSO FIB (フォアグラウンド開始のバックグラウンド) コマンドの使用権限を持っていない場合は、ジョブの実行依頼を行うことはできません。この権限をユーザーに与えるのは、TSO 管理タスクです。

バッチ・ジョブはユーザーの TSO ログオン ID のもとで実行されるため、バッチ・プロシージャが発行するコマンドはユーザーの許可 ID のもとで実行されます。バッチ・ジョブのユーザーおよび対話式でジョブを行うユーザーに、次のように同じ規則が適用されます。

- ユーザーが実行する照会、プロシージャ、または書式を所有していない場合は、それぞれの所有者が共有している必要があります。
- 照会で参照されるいずれの表に対しても (検索照会を前提として)、ユーザーは SQL SELECT 特権を持っている必要があります。

- 照会やプロシージャの結果を新規の表に保管する場合には、ユーザーの SAVE コマンドは“拡張”されたものである必要があります。(277ページの『ユーザーがデータベース内で表を作成できるようにする』を参照してください。)

アプリケーションの使用

ユーザーはアプリケーションを開始する前に、実行可能な照会やプロシージャを持っており、さらに、必要があれば、報告書をフォーマット設定する書式も持ってなければなりません。これらのオブジェクトは、それぞれデータベースまたは一時記憶域のいずれにあっても構いません。オブジェクトがデータベースにある場合は、共用されているものであれば、他人が所有しているものでも構いません。

ユーザーが適切なフィールドに入力して ENTER キーを押すと、アプリケーションはバッチ・ジョブを構成し、バックグラウンドに実行依頼します。ジョブが実行されると、JOB ステートメントの NOTIFY パラメーターにより、適切なメッセージがユーザーの端末に送られます。

プロンプト・パネルが表示されている間、ユーザーは以下のことが行えます。

- ヘルプ機能キーを押して、アプリケーションのヘルプ・パネルを表示する。
- 終了機能キーを押して、アプリケーションを終了する。

(機能キーの設定は、プロンプト・パネルの下部に表示されています。)

NLF を使用する場合: BATCH に対応するコマンド同義語を出して、バッチ・モードの照会またはプロシージャを実行します。たとえば、BATCH に対するドイツ語に変換されたコマンド同義語は STAPEL です。他の言語環境での BATCH に対して変換されたコマンド同義語については、`Q.COMMAND_SYNONYM_n` 制御表を参照してください。

アプリケーションの開始

アプリケーションは、ユーザーが QMF のもとで操作している間に開始する必要があります。アプリケーションが開始すると、ユーザー用にバッチ・ジョブを準備し、バックグラウンドに実行依頼します。ジョブは、ユーザーがプロンプト・パネルに入力した情報をもとにして準備されます。ジョブでは、ユーザーの選択した単一の照会またはプロシージャが実行されます。バッチ・ジョブで照会が選択された想定した場合、ジョブではさらに以下のことも行われます。

バッチ・プログラムとして実行するためのカスタマイズ

- ・ 照会の実行により作成されたデータ・オブジェクトの保管
- ・ ユーザーの選択した書式による報告書オブジェクトのフォーマット設定
- ・ 報告書の印刷
- ・ 永続データ・セットへの報告書の書き込み
- ・ 1人または複数のユーザーへの報告書の送信

アプリケーションを使用する利点はそのプロンプト・パネルにあり、ユーザーはプロンプト・パネルでジョブが何をすべきかを指示するだけで、どのように行うかの詳細はアプリケーションに任せることができます。ユーザーには、JCL あるいは QMF プロシージャーに関する知識は必要ありません。

バッチ・アプリケーションを使用するには、次のように入力します。

BATCH

これによって、図147 に示すプロンプト・パネルが表示されます。

プロンプト・パネルへの入力

機能キー 1 を押すことによって、ユーザーはプロンプト・パネルへの入力についてヘルプの表示を得ることができます。これによって、3つのヘルプ・パネルの最初のパネルが表示されます。

QMF BATCH	QUERY/PROC BATCH PROMPT
OBJECT NAME ==>	Name of query or procedure
Current OBJECT ==> NO	Use object in temporary storage?
QUERY or PROC ==> QUERY	
PROC arguments ==>	
FORM NAME ==>	Form to be used with query
Current FORM ==> NO	Use form in temporary storage?
BATCH NAME ==>	Name of QMF batch execution proc
DB2 SUBSYSTEM ==>	DB2 PLAN ==>
LOGON PASSWORD ==>	TSO logon password
LOGGING ==> YES	Log messages and commands?
SAVE DATA ==>	Name of data to be saved
REPORT DATASET ==>	
NEW DATASET ==>	Is the data set new?
VOLUME ==>	Optional if NEW or uncataloged
REPORT WIDTH ==> 133	Width of report line
VIEW REPORT ==> YES	Should report be printed?
OUTPUT CLASS ==> A	Class for PRINT and TRACE
DISTRIBUTION	Enter userids and nodes to send report.
USERID ==>	NODE ==>
==>	==>

PF1=Help PF3=End Enter=Process batch request

図 147. QMF バッチ・プロンプト・パネル

必須入力フィールド

バッチ・プロンプト・パネルの一部のフィールドは、必須フィールドです。必須フィールドに値を入力せずに ENTER キーを押すと、ユーザーに値を入力す

バッチ・プログラムとして実行するためのカスタマイズ

るようにとのプロンプトでメッセージが表示されます。続いて、カーソルが入力を必要とするフィールドに位置付けられます。表60 に、必須フィールドの説明があります。

表 60. BATCH アプリケーションの必須入力フィールド

フィールド	説明
OBJECT NAME	バッチ・モードで実行する照会またはプロシージャの名前を示す値が必要。現在、照会またはプロシージャが一時記憶域にある場合は、この名前データベースに保管される。名前が既存のオブジェクトの名前である場合には、新規のオブジェクトによって古いオブジェクトが置き換えられる。(名前は修飾されていないものでなければならない。) オブジェクトがデータベースにある場合は、それが保管されたときの名前を入力する。(オブジェクトが誰か他の人に所有され、共用されている場合には、名前を修飾する必要がある。) プロファイル設定として CONFIRM=NO を使用して、オブジェクトを保管する。
QUERY or PROC	バッチで実行するオブジェクトのタイプ。QUERY または PROC のどちらかでなければならない。
BATCH NAME	バッチ・モードで実行する QMF プロシージャの名前を示す値が必要。(名前は修飾されていないもの。) 複数の照会を実行依頼する場合は、それぞれの照会の BATCH NAME フィールドを変更する必要がある。変更しないと、新規のバッチ・ジョブが古いジョブを置き換える。このプロシージャには、ユーザーの入力に応じて、適切な QMF コマンドが入っている。QUERY または PROC フィールドで指定されたユーザーの照会またはプロシージャは、このプロシージャから実行される。プロシージャは、SHARE=YES キーワード・オプションによって保管される。バッチ・マシンによって実行できるものでなければならない。プロファイル設定として CONFIRM=NO を使用して、このプロシージャを保管する。

オプションの入力フィールド

518ページの表61 に、パネル上の残りの (オプション) 入力フィールドの説明があります。YES または NO の値が期待されている個所には、通常はデフォルトの YES または NO が画面に表示されます。ユーザーが YES/NO フィールドをブランクにすると、ユーザーに対してプロンプトで入力を指示するメッセージが出されます。

バッチ・プログラムとして実行するためのカスタマイズ

表 61. BATCH アプリケーションのオプション入力フィールド

フィールド	説明
Current OBJECT	現在、バッチ照会またはプロシージャが一時記憶域にある場合は、YES を入力する。すると、その照会またはプロシージャは後でバッチで実行するように保管される。照会またはプロシージャがデータベースにある場合は、NO を入力する。デフォルト値は NO。
OBJECT NAME	フィールドで指定されている REXX プロシージャに対する引き数。
FORM NAME	書式を使用してバッチ照会を実行するには、ユーザーはこのフィールドに書式の名前を指定する必要がある。使用する書式によって、 <ul style="list-style-type: none">• デフォルトの書式の場合、フィールドを空のままにしておく。• データベースにある場合、書式はこの名前でも保管される。書式が誰か他の人に所有され、共用されている場合には、名前を修飾する必要がある。• 現行の書式の場合、保管できる名前を指定する。書式はそれぞれの許可 ID で保管されるため、名前は修飾しないで指定する必要がある。 この書式は、プロファイルの設定として CONFIRM=NO を用いて保管される。 既存の名前を入力すると、新規の書式によって古い書式が置き換えられる。
Current FORM	バッチ書式が現行の書式である場合には、このフィールドに YES を入力する。すると、書式は後で使用するために保管される。書式がデータベースにある場合は、NO を入力する。このフィールドのデフォルト値は NO。
DB2 SUBSYSTEM	QMF が使用する DB2 サブシステムの名前を入力する。その値は、プログラム・パラメーター DSQSUBS の値と同じ。
DB2 PLAN	QMF アプリケーション・プランの名前を入力する。その値は、DSQSPLAN の値と同じ。
LOGON PASSWORD	ログオン・パスワードを入力する。これは画面には表示されない。

バッチ・プログラムとして実行するためのカスタマイズ

表 61. BATCH アプリケーションのオプション入力フィールド (続き)

フィールド	説明
LOGGING	このフィールドのデフォルトの値は YES。これは、バッチ・モードのデフォルトのトレース・レベルが L2 であることを意味し、メッセージおよびコマンドをトレースする。L2 レベルのトレースを行いたくない場合は、NO を指定する必要がある。バッチ・プロシージャのトレースは、SET PROFILE (TRACE=NONE コマンドの後までは続かない。その部分は、したがって生成されたユーザー・プロシージャ内にある。
SAVE DATA	照会またはプロシージャの実行結果のデータを保管したい場合には、このフィールドに値を指定する必要がある。DATA は、この名前と CONFIRM=NO キーワード・オプションを使用して、新規の表として保管される。
REPORT DATASET	報告書を永続データ・セットに書き込みたい場合には、データ・セット名をここに入力。名前は完全に修飾されている必要がある。データ・セットを書き込まない場合は、このフィールドは空のままにしておく。 このデータ・セット名は OS/390 JCL に渡されるので、OS/390 の命名規則に適合している必要がある。完全修飾名にピリオド、@、#、\$ 以外の特殊文字が入っていないければ、引用符は不要。引用符が使用されると、OS/390 は特殊文字が使用されているものと見なし、データ・セットをシステムにカタログしない。
NEW DATASET	REPORT DATASET に入力した場合は、このフィールドも入力が必要。このデータ・セットが現在存在していないことを示す場合は、YES と入力。現在存在していることを示す場合は、NO と入力。
VOLUME	NEW DATASET フィールドに YES と入力した場合は、オプションでこのフィールドの入力が可能。新規データ・セットが入るボリュームの通し番号を入力。ボリュームは、インストール先システムで定義されているような、SYSDA クラスで使用できる装置である必要がある。
REPORT WIDTH	NEW DATASET フィールドに YES と入力した場合は、このフィールドの入力が必須。この値が、新規データ・セットの論理レコード長 (LRECL) になる。報告書の幅が LRECL 以下の場合、デフォルト値 133 を使用する。

バッチ・プログラムとして実行するためのカスタマイズ

表 61. BATCH アプリケーションのオプション入力フィールド (続き)

フィールド	説明
VIEW REPORT	このフィールドは YES または NO が入っている必要がある。YES はジョブの印刷を、NO はジョブを印刷しないことを示す。
OUTPUT CLASS	ジョブからの印刷出力の出力クラスを入力。印刷出力には、以下のものが含まれる。 <ul style="list-style-type: none">・ システム・メッセージ・ 印刷する場合、報告書 (DSQPRINT)・ トレース出力 (DSQDEBUG)・ 生成された場合、異常終了ダンプ (DSQDUMP) インストール先システムでその準備がされている場合には、ユーザーの端末に経路指定するために印刷出力を保留する出力クラスを選択することができる。
DISTRIBUTION USERID および NODE	ユーザーが結果の報告書を他のユーザーに送信したい場合、これらのフィールドにユーザー ID とノードを入力する必要がある。このフィールドを使用する場合、報告書出力のデータ・セットを REPORT DATASET フィールドに指定する必要がある。 USERID フィールドの 1 つにユーザーのログオン ID を、そして対応する NODE フィールドにユーザーのノードを同一行で入力する。この方法で、報告書の受取人を 2 人まで指定することができる。報告書は TSO TRANSMIT コマンドを用いて送信される。ノード情報が NAMES.TEXTLIST データ・セットにある場合には、そのユーザーの NODE フィールドは入力する必要はない。指定するノード ID は、このファイルの名前リスト全体に対応していてもよく、その場合は 2 人だけではなく、それ以上の人に報告書を送信することができる。

バッチ・アプリケーションの変更

バッチ・アプリケーションは、そのコンポーネントに変更を加えるか、あるいはカスタマイズされたアプリケーション用に新規コンポーネントを作成することによって、変更することができます。保守を行っても変更内容が失われるリスクがないようにするために、新規コンポーネントの作成の方をお勧めします。

適用可能な QMF コンポーネント

バッチ・アプリケーションを変更する場合は、QMF ライブラリーにある以下のコンポーネントについて認識しておく必要があります。

- QMF710.SDSQCLTE ライブラリーにおける CLIST の DSQABB11 および DSQABB12
BATCH コマンドでバッチ・アプリケーションを呼び出すと、実際には DSQABB11 を呼び出していることとなります。この CLIST の目的は、ISPF SELECT を通じて、新規アプリケーションとして DSQABB12 を呼び出すことです。アプリケーションのロジックの大部分は、DSQABB12 にあります。
- QMF710.SDSQMLBE ライブラリーのメンバー DSQBE00、DSQBE01、および DSQBE02 における ISPF メッセージの定義。
これらのメッセージは、アプリケーションの終了後、ユーザーの画面に表示されます。アプリケーションでは、QMF MESSAGE コマンドを使用して、これらのメッセージを生成します。
- QMF710.SDSQPLBE ライブラリーでの、以下のような目的を持つ各種 ISPF パネルの定義。
 - DXYEABMP はアプリケーションのプロンプト・パネル。
 - DXYEABM1、DXYEABM2、および DXYEABM3 はプロンプト・パネル用のヘルプ・パネル。
 - DXYEAB12、DXYEAB13、DXYEAB14、および DXYEAB15 ではアプリケーションのエラー・メッセージ用にメッセージ・ヘルプを用意。
- QMF710.SDSQSLBE ライブラリーにおける、ファイルを合わせるためのいくつかのモデル。
 - DSQABB1J はバッチ・ジョブの JCL をモデル化。バッチ・モードでの照会実行プロシージャーをモデル化したもの。
 - DSQABB1P および DSQABB1S は QMF プロシージャーをモデル化。ジョブの JCL 実行依頼プロシージャーをモデル化したもの。

アプリケーションで可能な変更

アプリケーションに対して、以下のような変更を行うことができます。

- ユーザーが DB2 サブシステムを選択できるようにする。
モデル・ファイル DSQABB1J の中に、バッチ・モードの QMF を呼び出す ISPSTART ステートメントがあります。このステートメントでは、QMF の DSQSSUBS パラメーターに値を与えていません。その結果、QMF が実行される DB2 サブシステムは、DSN という名前を持っていると見なされます。これとは異なる名前の DB2 サブシステムで QMF を実行したい場合には、ISPSTART コマンドの PARM オペランドに DSQSSUBS=xxx を追加します (ここで、xxx は適切なサブシステム名)。

バッチ・プログラムとして実行するためのカスタマイズ

- ユーザーが、印刷する報告書に GDDM ニックネームを指定できるようにする。
- インストール先システムの規則を強化するために、さらにロジックを追加する。
たとえば、ユーザーが報告書出力用に新規のデータ・セットを作成する際、受け入れ可能なボリューム・リストをユーザーに提供することができます。
- アプリケーションが生成した JCL を、インストール先システムに適合するよう変更する。

次のようなことが可能です。

- JOB ステートメントへのアカウント情報の追加。
- SYSTSIN データ・セットの ISPSTART ステートメントにある QMF アプリケーション・プランの名前の変更。

さらに、以下のような追加の変更も可能です。

- プロンプト・パネルへのフィールド (1 つまたは複数) の追加
- プロンプト・パネルのヘルプ・パネル変更
- 新規エラー・メッセージの DSQBE00、DSQBE01、または DSQBE02 への追加
- DSQABB12 のロジックの一部変更

重要: バッチ・アプリケーションを呼び出すユーザーは、userid.DSQ1EBFT.PROC という名のデータ・セット (ここで、userid はユーザーの TSO ログオン ID) を保守してはなりません。そのようなデータ・セットを保守すると、QMF バッチ・アプリケーションが正しく実行されない場合があります。

アプリケーション変更の例

次の例は、BATCH アプリケーションを変更する 1 つの方法を示しています。

すべてのユーザーが同じ PROFILE PREFIX を持つバッチ・アプリケーションを変更します。すべてのユーザーが固有のユーザー ID を持っているものと仮定しています。&SYSUID および &ZUSER を使用して、データ・セット名にユーザー ID を追加します。

DSQABB1S SKELETON に 3 つの変更を行う必要があります。523ページの図 148 は、必要な変更を示しています。古い行はコメント行にしています。そのすぐ後に、置き換えられる新規の行が続いています。

```

)CM -----
)CM FILE: DSQABB1S
)CM DESCRIPTION: THIS SKELETON CREATES DSQABB1S, THE PROC WHICH
)CM                SAVES THE CURRENT FORM (IF SPECIFIED)
)CM                IMPORTS AND SAVES THE PROC WHICH RUNS THE QUERY
)CM                SENDS THE QMF INVOCATION JOB TO OS/390 BATCH
)CM                RESETS THE PROC ITEM
)CM                FREES ISPF FILE USED FOR FILE TAILORING
)CM                DISPLAYS THE QUERY PANEL
)CM -----

)SEL &FAN = &YES
&SAVE &FORM &AS &FNAME (&SHARE=&YES, &CONFIRM=&NO
)ENDSEL

)CM &IMPORT &PROC &FROM '&ZPREFIX..DSQ1EBFT.&PROC.' (&MEMBER = DSQABB1P
&IMPORT &PROC &FROM '&ZPREFIX..&ZUSER..DSQ1EBFT.&PROC.' (&MEMBER = DSQABB1P
&SAVE &PROC &AS &PNAME (&CONFIRM=&NO
)CM TSO SUBMIT '&ZPREFIX..DSQ1EBFT.&PROC.(DSQABB1J)'
TSO SUBMIT '&ZPREFIX..&ZUSER..DSQ1EBFT.&PROC.(DSQABB1J)'

TSO FREE FILE(ISPF FILE) DELETE
&RESET &PROC
)CM &IMPORT &PROC &FROM DSQABB
&IMPORT &PROC &FROM &ZUSER..DSQABB

)SEL &ITM = &QUERY
&DISPLAY &QUERY
)ENDSEL

```

図 148. DSQABB1S SKELETON の変更

524ページの図149 のコメントにあるように、DSQABB12 CLIST に 5 つの変更を行います。

バッチ・プログラムとして実行するためのカスタマイズ

```

/*****/ 00088000
/* ALLOCATE USERID.DSQ1EBFT.PROC TO BE USED FOR ISPF */ 00089000
/* FILE TAILORING OUTPUT. */ 00090000
/*****/ 00091000
FREE FILE(ISPFILE) 00092000
/* ALLOC DDNAME(ISPFILE) DSNAME(DSQ1EBFT.&PROC) OLD 00093000
ALLOC DDNAME(ISPFILE) DSNAME(&SYSUID..DSQ1EBFT.&PROC) OLD 00093000
IF &LASTCC = 0 THEN + 00094000
DO 00095000
FREE ATTRLIST(ATTRPDS) 00096000
ATTR ATTRPDS LRECL(80) RECFM(F B) BLKSIZE(800) DSORG(PO) 00097000
/* ALLOC DDNAME(ISPFILE) DSNAME(DSQ1EBFT.&PROC) NEW SPACE(5,2) + 00098000
/* TRACKS DIR(10) USING(ATTRPDS) CATALOG 00099000
ALLOC DDNAME(ISPFILE) DSNAME(&SYSUID..DSQ1EBFT.&PROC) NEW + 00098000
SPACE(5,2) TRACKS DIR(10) USING(ATTRPDS) CATALOG 00099000
END 00100000
IF &RC = 8 THEN + 00101000
DO 00102000
:
:

/*****/ 00203000
/*EXPORT CURRENT CONTENTS OF PROC PANEL */ 00204000
/*****/ 00205000
ISPEXEC SELECT PGM(DSQCCI) + 00206000
/* PARM( &EXPORT &PROC &TO DSQABB (&CONFIRM = &NO ) 00207000
PARM( &EXPORT &PROC &TO &SYSUID..DSQABB (&CONFIRM = &NO ) 00207000
IF &LASTCC = 0 THEN DO 00208000
ISPEXEC SELECT PGM(DSQCCI) + 00209000
PARM(SET GLOBAL (DSQEC_NLF_CMD_LANG = &LOCLANG )) 00210000
SET &MSG = &DSQB.023 00211000
ISPEXEC SELECT PGM(DSQCCI) PARM( &MESSAGE &MSG ) 00212000
SET &RCDE = 8 00213000
GOTO CLEANUP 00214000
END 00215000
:
:

```

図 149. DSQABB12 CLIST の変更 (1/2)

バッチ・プログラムとして実行するためのカスタマイズ

```

/*****/
/* IMPORT AND RUN FILE TAILORED SKELETON */
/*****/
ISPEXEC SELECT PGM(DSQCCI) +
/* PARM( &IMPORT &PROC &FROM DSQ1EBFT (&MEMBER = DSQABB1S )
  PARM( &IMPORT &PROC &FROM &SYSUID..DSQ1EBFT (&MEMBER = DSQABB1S )
  IF &LASTCC ^= 0 THEN +
:
:
CLEANUP: FREE FILE(ISPFILE) DELETE
DONE: SET &ZPLACE = &SAVEPLC
      SET &ZPFCTL = &SAVEPFC
      SET &ZPF01 = &STR(&SAVEPF01)
      SET &ZPF13 = &STR(&SAVEPF13)
      SET &ZPF03 = &STR(&SAVEPF03)
      SET &ZPF15 = &STR(&SAVEPF15)
      SET &ZPF10 = &STR(&SAVEPF10)
      SET &ZPF22 = &STR(&SAVEPF22)
      SET &ZPF11 = &STR(&SAVEPF11)
      SET &ZPF23 = &STR(&SAVEPF23)
      ISPEXEC VPUT (ZPLACE ZPFCTL ZPF01 ZPF13) PROFILE
      ISPEXEC VPUT (ZPF03 ZPF15 ZPF10 ZPF22 ZPF11 ZPF23) PROFILE
/*
DELETE DSQABB.&PROC
DELETE &SYSUID..DSQABB.&PROC
EXIT CODE(&RCDE)
```

図 149. DSQABB12 CLIST の変更 (2/2)

ネイティブ OS/390 での QMF バッチの実行

TSO および ISPF で QMF バッチを実行する他に、ネイティブ OS/390 バッチ・ジョブとして QMF を実行することができます。526ページの図150 に示す JCL を使用して、ネイティブ OS/390 のバッチ・ジョブとして QMF を実行することができます。

バッチ・プログラムとして実行するためのカスタマイズ

```

/*****/
//QMFBAT JOB 00299000
//S1 EXEC PGM=DSQQMFE,PARM='M=B,I=yourQMFproc' 00300000
//* 00301000
//* Program libraries required when running in batch 00302000
//* 00303000
//STEPLIB DD DSN=QMF710.SDSQLOAD,DISP=SHR 00304000
// DD DSN=DSN.SDSNEXIT,DISP=SHR 00305000
// DD DSN=DSN.SDSNLOAD,DISP=SHR 00306000
// DD DSN=GDDM.ADMLOAD,DISP=SHR 00307000
//* 00308000
//* QMF/GDDM maps are required when running in batch 00309000
//* 00310000
//ADMGGMAP DD DSN=QMF710.DSQMAPE,DISP=SHR 00311000
//* 00312000
//* 00313000
//* 00314000
//* Datasets used by QMR 00315000
//* 00316000
//DSQPRINT DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=1330) 00317000
//DSQDEBUG DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210) 00318000
//DSQDUMP DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=125,BLKSIZE=1632) 00319000
//DSQSPILL DD DSN=DSN.SDSQSPILL,DISP=(NEW,DELETE), 00320000
// UNIT=SYSDA,SPACE=(TRK,(100),RLSE), 00321000
// DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096) 00322000
//* 00323000
/*****/ 00324000
```

図 150. QMF をネイティブ OS/390 のバッチ・ジョブとして実行するための JCL

QMF をネイティブ OS/390 で実行する場合には、以下のことを忘れないようにしてください。

- TSO は利用できない。
- 一部の QMF 機能には TSO または ISPF が必要であり、ネイティブ OS/390 で QMF を実行する場合はそれらの関数は稼動しない。
- ファイルのエクスポートまたはインポートでは、完全修飾されたデータ・セット名を使用する必要がある。デフォルトのユーザー ID の接尾部は使用できない。
- ロジックを持つプロシージャ (REXX PROCs) は使用できない。TSO 以外のアドレス・スペースにおいて REXX で QMF を実行するためには、527ページの図151 に示すように IRXJCL を使用する必要がある。

527ページの図151 に示す REXX プログラムでは、QMF 呼び出し可能インターフェイスを使用して、QMF を開始させ、QMF コマンドをバッチ・モードで実行します。


```
//QMF BATCH JOB REGION=8M,
// MSGCLASS=H,TIME=(2,30),USER=&SYSUID,NOTIFY=&SYSUID,CLASS=A
//ROBQMF1 EXEC PGM=IRXJCL
//STEPLIB DD DSN=DSN.DB2A.SDSNLOAD,DISP=SHR
// DD DSN=DSN.DB2A.SDSNEXIT,DISP=SHR
// DD DSN=QMFDEV.QMF710.SDSQLOAD,DISP=SHR
//ADMGGMAP DD DSN=QMFDEV.QMF710.DSQMAPE,DISP=SHR
//SYSEXEC DD DSN=ROBIN.QMF710.SDSQEXCE,DISP=SHR
//DSQPRINT DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=137,BLKSIZE=1330)
//DSQDEBUG DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210)
//DSQDUMP DD SYSOUT=A,DCB=(RECFM=VBA,LRECL=125,BLKSIZE=1632)
//SYSUDUMP DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//DSQSPILL DD DSN=&&SPILL,DISP=(NEW,DELETE),
// UNIT=VIO,SPACE=(CYL,(1,1),RLSE),
// DCB=(RECFM=F,LRECL=4096,BLKSIZE=4096)
//SYSTSIN DD *
/* REXX */
CALL DSQCIX "START (DSQSMODE=BATCH)"
SAY DSQ_MESSAGE_ID DSQ_MESSAGE TEXT
IF DSQ_RETURN_CODE = DSQ_SEVERE THEN EXIT DSQ_RETURN_CODE
CALL DSQCIX "RUN PROC REXXPP"
SAY DSQ_MESSAGE_ID DSQ_MESSAGE TEXT
IF DSQ_RETURN_CODE = DSQ_SEVERE THEN EXIT DSQ_RETURN_CODE
CALL DSQCIX "EXIT"
SAY DSQ_MESSAGE_ID DSQ_MESSAGE TEXT
EXIT DSQ_RETURN_CODE
/*
```

図 151. QMF をバッチ・モードで開始、実行させる REXX プログラム

ユーザーが CICS でバッチ・モードを使用できるようにする

CICS では、QMF は会話型トランザクションとして、対話式に実行されます。QMF が必要とするリソースはすべて、ユーザーのセッションを通じて使用可能です。リソースを保護するために、報告書の生成に使用できる QMF プロシージャーを実行するとよいでしょう。このプロシージャーは非対話式に実行できます。

QMF トランザクションは、端末から実行することも、端末なしのトランザクションとしても実行することができます。

端末からのバッチの実行

QMF を端末から実行して、報告書を作成することができます。たとえば、528ページの図152 のプロシージャーを書いて、CICS 補助記憶域に報告書を作成することができます。(QMF プロシージャーの「--」で始まる行を、QMF はコメント行として扱います。)

バッチ・プログラムとして実行するためのカスタマイズ

```
-- Procedure name: STATRPT1_PROC
--
-- Example QMF procedure to create an auxiliary CICS
-- temporary storage queue named STATRPT1
--
  RUN  QUERY STATRPT1_QUERY (FORM=STATRPT1_FORM)
  PRINT REPORT (QUEUENAME=STATRPT1,QUEUETYPE=TS)
--
-- End of procedure
```

図 152. CICS 補助記憶域に報告書を作成

ここで説明した QMF トランザクションをバッチ・モードで実行します。

```
QMFE M=BATCH,I=STATRPT1_PROC
```

QMF は、画面には何も表示せずにこのトランザクションを実行します。プロシージャが正常終了すると、報告書が CICS 記憶域キュー STATRPT1 に作成されます。そこで、CICS 提供のトランザクション CEBR を使用して、報告書を表示することができます。

```
CEBR STATRPT1
```

端末なしでのバッチの実行

QMF トランザクションは、端末なしでも実行することができます。バッチ・ジョブ実行のための端末は、QMF がトランザクションを完了するまでロックされます。端末を用いずに QMF を実行するには、EXEC CICS START コマンドを使用することができます。次の例では、QMF プロシージャ STATRPT1_PROC を実行します。

```
EXEC CICS START TRANSID(QMFE) FROM(M=BATCH,I=STATRPT1_PROC)
```

このトランザクションが完了した後で、CICS 提供のトランザクション CEBR を使用して、CICS 記憶域キュー STATRPT1 をブラウズすることができます。

プロシージャのデバッグ

QMF は、コマンド・レベルおよびメッセージ・レベルのトレース機能を備えています。この機能は、QMF プロシージャをバッチ・モードで実行すると問題が生じるような場合に役立ちます。QMF コマンド・レベルおよびメッセージ・レベルのトレースは、QMF をバッチ・モードで実行すると、自動的にアクティブになります。このメッセージ・トレースを CICS 一時記憶域または一時データ・キューに経路指定することができます。経路指定の方法の詳細については、219ページの『DSQSDBQN (CICS のトレース・データ用記憶域の名

前の指定)』および 218ページの『DSQSDBQT (CICS のトレース・データ用記憶域のタイプの指定)』を参照してください。

たとえば、前のプロシージャーを実行し、コマンドおよびメッセージ・トレースを QMFMSG という名前で CICS 補助記憶域に送るには、次のような CICS START コマンドを出します。

```
EXEC CICS START TRANSID(QMFE)
      FROM(M=BATCH,I=STATRPT1_PROC,DSQSDBQN=QMFMSG,DSQSDBQT=TS)
```

複数の QMF トランザクションが、同じトレース域にメッセージを出すことも可能です。一方で QMF が、トレース項目を書き出している間に、QMF は CICS ENQ コマンドをキュー名に出します。それぞれの項目には、そのトレース項目を作成した QMF トランザクションの端末 ID とタスク ID が付けられています。

QMF トレースを CICS 補助記憶域に経路指定する場合、コンポーネント・レベル全体のトレースを設定しないようにしてください。そのように設定すると、一時記憶域がすぐにいっぱいになるためです。メッセージ・レベル以外のトレースを行う場合には、一時データ (デフォルト) を使用するようお勧めします。

終了リターン・コード

QMF の終了リターン・コードは次のとおりです。

- 0 正常終了
- 8 異常終了

バッチ・プログラムとして実行するためのカスタマイズ

第24章 障害追及と問題診断

この章は、ユーザーが QMF を使用中に問題が発生した場合に、問題の解決を支援するのに使用してください。532ページの『一般的な問題の障害追及』では一般的な問題の解決方法を示し、541ページの『診断援助機能を使用した問題の判別』ではさらに複雑な問題の解決に役立つ診断援助機能について説明します。

クイック・スタート

表62 に示されているステップは、一般的なエラーの障害追及やより複雑な問題の診断を行う際の手引きとして使用してください。ステップについてさらに詳しい説明が必要な場合は、表の右端に示しているページを参照してください。

表 62. 一般的なエラーの障害追及および問題の診断

問題の診断ステップ	参照箇所
印刷中に GDDM エラー または QMF エラー が発生した場合は、プリンター名を指定しなかったか、名前の定義や装置の割り振りが誤っていたか、入出力エラーが発生したものと考えられる。	534 ページ
QMF ホーム・パネル に警告メッセージが表示された場合は、QMF が、初期化中に表やルーチンの読み取りまたはロードを行おうとして、エラーを検出したものと考えられる。	533 ページ
報告書表示が矛盾していると思われる場合には、報告書を表示する前に、(報告書を生成する表の) 未加工の 2 進データを文字データへ変換する必要があったということが考えられる。	537 ページ
応答が遅い場合には、データ・オブジェクトをリセットするか、記憶域を増やす必要があるものと考えられる。また、REXX の検索順序や QMF から REXX に何度も移動する処理が原因で、REXX に問題が起こっている場合もある。	538 ページ
問題が上記のいずれにも該当しない場合は、 QMF 、 TSO 、 ISPF 、 SRPI 、 APPC 、 OS/390 、 CICS のうちのどの診断援助機能が さらに問題を診断するのに役立つかを判別する。	541 ページ
QMF メッセージ・サポート を使用して問題を判別するには、ヘルプ・パネル上のメッセージ番号を用いて、それを出した QMF 機能などのエラーに関するさらに詳しい情報を判別する。	542 ページ

障害追及と問題診断

表 62. 一般的なエラーの障害追及および問題の診断 (続き)

問題の診断ステップ	参照箇所
QMF トレース機能を使用して問題を判別するには、DSQSDEBUG プログラム・パラメータを設定するか、ユーザーのプロファイルを表示して TRACE オプションの値を変更するか、または SET PROFILE (TRACE=value コマンドを使用して、トレースをオンにする。DSQSDEBUG パラメータの詳細レベルは、ALL または NONE である。トレースを実行する直前に、トレース・レベルを選択して指定することもできる。たとえば、コマンド行に SET T=C2D2L2 と入力することができる。	544 ページ
OS/390 診断機能を使用して問題を判別するには、本書の説明を用いて、OS/390 ダンプから QMF を識別する。	553 ページ
CICS 診断機能を使用して問題を判別するには、本書の記載事項および <i>CICS Problem Determination Guide</i> を参照して、CICS トランザクション・ダンプ中の QMF を調べる。	553 ページ
TSO の QMF 割り込み機能を使用して問題を判別するには、割り込みの作成と読み取りを行う。	555 ページ
Q.ERROR_LOG 表からの報告書を使用して問題を判別するには、エラーが発生した SQL 許可 ID およびエラーのおよその日時を指定して、表に対する SELECT 照会を実行する。	557 ページ
IBM に問題を報告するには、IBM の ServiceLink 機能がある場合にはその機能を使用するか、IBM サポート担当員に連絡する。	558 ページ

一般的な問題の障害追及

この節は、初期化時のエラー、印刷時のエラー、画面の警告メッセージ、矛盾した報告書表示、および応答時間の遅さやその他のパフォーマンスの問題を解決する方法を判別するのに役立ちます。

初期化エラーの処理

QMF を開始できない場合には、次に示すように一般的な修正方法がいくつかあります。

- ショップのすべての QMF ユーザーが QMF を開始できないのか、1 人のユーザーだけが開始できないのかを判別する。
- 端末画面上にメッセージが表示されていないかを調べ、さらに QMF メッセージおよびコードの DSQDEBUEG ファイル・メッセージの説明を探す。
- 画面に何も表示されていない、さらに DSQDEBUEG にも何もない場合には、SPUFI に入って SELECT * FROM Q.ERROR_LOG を発行し、QMF にアクセスしようとした際の項目がないかどうかを調べる。
- QMF は、QMF の初期化時に DB2 と GDDM を初期化する。DSN (DB2) または ADM (GDDM) エラー・メッセージが出ていた場合には、該当する製品のメッセージおよびコードの資料でそれらのメッセージを調べる。

DB2 データベースが初期化されて正しく作動しているかどうかを確認する。すべてのユーザーが始動時に ADMxxxx というタイプのメッセージを受け取っている場合には、GDDM IVP を実行して、基本 GDDM 製品が正しく作動しているかどうかを確認する。

- 最後に、ユーザーは、画面に表示されるメッセージはこの他にはないか、あるいは DSQDEBUG や Q.ERROR_LOG にはこの他に情報はないかどうかを確認する必要がある。他にメッセージが見つからなければ、ユーザーに TSO コマンドの PROFILE MSGID WTPMSG を実行して QMF の再始動を試みてもらう。

警告メッセージの処理

QMF の初期化中 (または CONNECT コマンドを発行した後) にエラーが発生すると、QMF ホーム・パネルに次のメッセージが表示される。

Warning messages have been generated

この種類のメッセージの原因となったエラーでは、QMF は停止しない。このメッセージは、以下のいずれかのロードまたは読み取り時に QMF で問題が生じていることを示している。

- コマンド同義語表
- 機能キー定義表
- リソース管理テーブル (管理プログラム出口ルーチン用)
- ユーザー編集出口ルーチン
- 管理プログラム出口ルーチン
- モジュール・レベル・トレース管理

コマンド同義語、機能キー、およびリソース管理テーブルについては、以下の点を確認する。

- ユーザーがその表に対して SQL SELECT 特権を持っていること。このことが問題の原因である場合は、265ページの『オブジェクトにアクセスするのに必要な SQL 特権』の指示に応じて SQL GRANT ステートメントを発行する。
- 表が適切な構造に従って構成されていること。
 - コマンド同義語表の構造については、343ページの『第18章 QMF コマンドのカスタマイズ』に説明がある。
 - 機能キー表の構造については、361ページの『第19章 QMF 機能キーのカスタマイズ』に説明がある。
 - リソース管理テーブルの構造については、445ページの表54に説明がある。

障害追及と問題診断

- 表のすべての行に、有効なデータが含まれている。これが問題であると考えられる場合には、以下の項を参照。
 - 有効なコマンド同義語定義については、349ページの『コマンド同義語定義を表へ入力』参照。
 - 有効な機能キー定義については、366ページの『ユーザーの機能キー定義を表へ入力』参照。
 - 有効なリソース管理情報については、445ページの表54参照。
- 表のすべての行が固有であること。

エラーに関する詳細な情報が、ユーザーのトレース・データにログが記録されている。TSO、SRPI、APPC、およびネイティブ OS/390 では、トレース・データは DSQDEBQT に保管される。CICS では、QMF セッションの開始時に DSQSDBQT または DSQSDBQN プログラム・パラメーターを使用してタイプや名前を変更していない限り、トレース・データは DSQD という名前の一時データ・キューに保管される。

トレース・データ内の情報を表示するには、最初にヘルプ・キーを押してメッセージ番号が入っているパネルを表示する。次に、549ページの『QMF トレース・データを表示する』の指示に従って、ユーザーのトレース・データをブラウズまたは印刷する。エラーに関する情報を見るには、トレース・データを調べてメッセージ番号の数値部分を探す。

印刷時の GDDM エラーの処理

印刷中に GDDM エラーが発生した場合は、QMF は次のようなメッセージを表示します。

```
GDDM error using nnnnnnnn. See message help for details.
```

メッセージ内の文字ストリング nnnnnnnn は、GDDM プリンターのニックネームを表します。ヘルプ・キーを押すと、エラーの説明を含むヘルプ・パネルが表示されます。この節では、いくつかの一般的なエラーとその修正方法について説明します。

DSQ50623

```
GDDM error. ADM0307 E FILE 'ADMPRINT.REQU--QUEUE' NOT  
FOUND. Severity 8. Function DSOPEN. *** CMD=PRINT
```

このようなメッセージが表示された場合には、QMF はユーザーが指定したプリンター名のニックネーム定義を見つけることができません。プリンター名のニックネーム定義を設定するか、すでに定義されているニ

ックネームを指定する必要があります。詳細については、327ページの『プリンターのための GDDM ニックネームの選択』を参照してください。

DSQ50623

GDDM error. ADM0314 E UNABLE TO OPEN 'MYPRINT'. DD STATEMENT MISSING. Severity 8. Function DSOPEN. * CMD=PRINT**

このようなメッセージが表示された場合には、QMF は出力の DD ステートメントを見つけることができませんでした。QMF 始動 EXEC、CLIST、または JCL に DD ステートメントを用意して、そのニックネームからの出力で何を行うかを指定する必要があります。JCL の例が、336ページの図80 にあります。

DSQ50623

GDDM error. ADM0482 E DEVICE NAME LIST '31E' IS INVALID FOR FAMILY 1. Severity 8. Function DSOPEN. * CMD=PRINT**

このようなメッセージが表示された場合には、ニックネーム定義は誤りです。指定した装置トークンが、ニックネームを作成した GDDM プリンターのタイプに対して有効なトークンではありません。GDDM プリンターに ADMMNICK 指定を作成する方法については、327ページの『プリンターのための GDDM ニックネームの選択』を参照してください。GDDM プリンターの各ファミリーに有効な装置トークンのリストについては、3.1 の場合は、*GDDM システムのカスタマイズおよび管理*、2.3 の場合は *GDDM Installation and System Management for OS/390* を参照してください。

DSQ50631

GDDM error. ADM0904 E ALPHANUMERIC FIELDS ARE NOT SUPPORTED FOR THIS DEVICE. Severity 8. Function ASDFLD. * CMD=PRINT**

このようなメッセージが表示された場合には、ユーザーが印刷しようとしている出力が、GDDM ニックネームで定義されているプリンターのタイプに対して無効です。QMF 図表などの一定のタイプの出力は、特定のファミリーの GDDM プリンターでしか印刷することはできません。このタイプの出力をどのファミリーのプリンターが処理できるかについては、3.1 の場合は *GDDM システムのカスタマイズおよび管理*、2.3 の場合は *GDDM Installation and System Management for OS/390* を参照してください。

DSQ90551

GDDM error. ADM0055 E SPINIT, AT '82F810C2'X ADM0050 E DEFAULTS ERROR. INVALID SYNTAX OR VALUE AT '...JIP,ADMMNICK'

QMF の開始時に、このようなメッセージが表示されることがあります。このメッセージは、ニックネームの ADMMNICK 指定のどこかに構文エラーがあったことを示しています。ADMMNICK 指定の構文の例については、327ページの『プリンターのための GDDM ニックネームの選択』を参照してください。構文エラーの修正を終えたら、ADMADFC GDDM のデフォルト・モジュールを再ロードしてください。

DSQ50633

GDDM error ADM0327 E 'TD WRITEQ' ERROR CODE '08000000'X, ON 'sYSP'. Severity 8. Function FSFRCE. * CMD=PRINT**

このようなメッセージは、QMF が印刷の出力先にしようとしている一時記憶域または一時データ・キュー (SYSP) がクローズされているか、始動 JCL に DD ステートメントがないことを示しています。この問題を解決する (JCL を変更して CICS を再始動するか、キューをオープンする) には、CICS 管理者に連絡してください。

印刷時の QMF エラーの処理

以下の情報は、印刷時に発生したエラーを解決する際に役立ちます。

起こったこと	意味	処理
コマンド行または機能キーから PRINT コマンドを出すと、次のようなメッセージが表示される。 GDDM printer nickname is required for PRINTER.	印刷しようとしているオブジェクトにはプリンター名が必要になるが、プロファイルにデフォルトのプリンター名がない。	ENTER キーを再度押すとプロンプト・パネルが表示されるので、そこでプリンター名とその他の印刷パラメーターを入力することができる。プロファイル内にデフォルトのプリンター名を設定して、プロンプトを表示しないようにすることもできる。
複数の PRINT コマンドを出しても、最後のオブジェクトだけが印刷される。	出力データ・セットに MOD のファイル属性指定がないため、各 PRINT の操作によってデータ・セットが再オープンされ、前の内容が上書きされている。	出力データ・セットのファイル属性指定を MOD に変更する。領域化されたデータ・セットのメンバーには、ファイル属性指定として MOD を使用することはできない。

起こったこと	意味	処理
QMF オブジェクトを印刷すると、印刷出力またはデータ・セットに予期しない制御文字が現れる。	使用している装置トークンまたは PROCOPT が、実際に印刷している装置と一致していない。	正しい装置トークンを指定するか、以下のいずれかの手法を使用して制御文字数を最少に減らす。 <ul style="list-style-type: none"> • 報告書、表、SQL または QBE 照会、プロシージャー、プロファイルについては、PRINTER=' ' を指定して GDDM の印刷をう回する。 • その他のオブジェクトについては、装置トークンを指定しない PROCOPT=((PRINTCTL,0)) を使用する。
報告書、表、SQL または QBE 照会、プロシージャー、またはプロファイルの印刷時に、次のメッセージが表示される。 File DSQPRINT did not open.	プロファイルにデフォルトのプリンター名がなく、現在割り振られている DSQPRINT データ・セットまたはシステム出力もない。	印刷コマンドを出す前に、DSQPRINT を割り振る。

注意：DSQDEBUG からの出力を HOLD キューに入れるよう割り振る場合には、OUTPUT キューへの出力を解放するために次の TSO コマンドを出す必要がある。

```
FREE DDNAME(DSQDEBUG)
```

表示エラーの処理

ユーザーが報告書を表示しようとしたときに報告書に表示制御文字がいくつか入っている場合には、報告書作成の基本となった表の 1 つまたは複数の列に (文字データではなく) 2 進データが入っています。QMF には、それらの制御文字の処理方法として次の 3 つの方法が用意されています。

- HEX 関数を使用する方法
- QMF 書式で QMF 提供の 16 進およびビット編集コードを使用する方法
- ユーザー作成編集ルーチンを用いて 2 進データを処理する方法

HEX 関数を使用する方法

HEX 関数は、引き数を対応する文字ストリングに変換する SQL スカラー関数です。結果のストリングは、引き数を 16 進表記で表した値となります。たとえば、関数の引き数が ABC の場合には、16 進表記のストリング C1C2C3 が生成されます。

ユーザーに、2 進データを含んでいる可能性のある列の前の照会で、HEX というワードを使用するように指示してください。たとえば、次のステートメントは表 SMITH.TABLEA の列 A の 2 進データの変換を行います。

```
SELECT HEX(A) FROM SMITH.TABLEA
```

QMF 提供の 16 進およびビット編集コードを使用する方法

文字データ用の 2 つの編集コード (およびその折り返し版) によって、QMF は文字の列に 2 進データを表示することができます。2 つの編集コードとは、X と XW (16 進表示の場合) および B と BW (ビット表示の場合) です。これらの編集コードの使用の詳細については、*QMF 解説書* を参照してください。

ユーザー作成編集ルーチンを用いた 2 進データの処理

HEX 関数や 16 進およびビット編集コードの使用は、2 進データを処理するのに有効な方法です。たとえば、各ビットが 1 つのデータ項目を表し、その値は自然言語形式で表示するものとします。5 番目のビットが 16 進値ではなく性別を表すとした場合には、ユーザーの編集コード・ルーチンによって Male または Female という値を表示させることができます。

ユーザー独自の編集コードを作成して COBOL、PL/I、またはアセンブラーで編集出口ルーチンを書くことによって、2 進データを必要な文字ストリングに変換することができます。場合によっては、新しく作成した編集コードを使用するユーザーのために、いくつかの QMF 書式を事前定義することも考慮してください。詳細については、377ページの『第20章 QMF 書式のためのユーザー独自の編集コードの作成』を参照してください。

パフォーマンスの問題の解決

ユーザーが照会の実行や報告書のフォーマット設定を行ったときに応答が遅い (パフォーマンスが低い) と感じた場合には、QMF が、1 つのコマンドの中で要求されたデータベース行をすべて検索してから次のコマンドを開始しようとしていることに問題がある場合があります。あるいは、要求されたすべての行を検索するのに十分なだけの仮想記憶域がユーザーにないことも考えられます。この節では、各種の問題の解決に役立つ方法を説明します。

パフォーマンスを向上させるためのデータ・オブジェクトのリセット

報告書の一部を表示した後、ユーザーが UPDATE 照会を実行しようとして、照会の結果が戻るまでに非常に長い時間待たされるものとします。QMF は 1 つのデータベース・タスクを完了してから別のタスクを開始するので、QMF は、UPDATE 照会を実行する前に報告書を完了しようとしている (残りの行を検索して DATA オブジェクトに入れようとしている) ものと思われます。以下のコマンドについては、QMF は報告書を完了してからでなければコマンドを実行することができません。

CONNECT	REFRESH (データベース・オブジェクト・リストの)
DISPLAY 表名 または QMF オブジェクト (データベースから)	RESET QUERY (LANGUAGE=PROMPTED が指定され、 照会を変更するもの)
DRAW 表名	RUN (データベース内のオブジェクト)
EDIT TABLEERASE	RUN QUERY
EXPORT (データベースから)	SAVE (データ、書式、プロシージャ、 またはプロファイル)
IMPORT (データベースへ)	
LIST	
PRINT (データベースから)	

ユーザーが、DB2 のオブジェクトを除去、作成、または変更する照会を実行すると、DATA オブジェクトが強制的に完了させられます。この処理の間、QMF は DATA オブジェクトを完了するまではキーボードの読み取りを行わないため、キーボードもロック状態になったように見えます。

グローバル変数 DSQEC_RESET_RPT の設定によっては、QMF が報告書を完了する前にこれらのコマンドを実行しようとした場合に起こるパフォーマンスの問題を避けるために、ユーザーに対して、報告書の必要な部分を表示し終わるとすぐに RESET DATA コマンドを発行するように、指示する必要がある場合があります。詳細については、*QMF 解説書* を参照してください。

記憶域が不足した状態で一定のコマンドを実行しようとしたユーザーは、incomplete DATA (DATA が未完了) というプロンプトを受け取ります。これは、QMF に現在の DATA オブジェクトを強制的に「完了」させるコマンドが原因で出されるものです。(DATA オブジェクトは、そのすべての行が QMF によって取り出され、DSQPILL ファイルへコピーした部分以外ほどの部分も廃棄されていないという時点で、完了します。) この問題を解決するために、ユーザーが DATA オブジェクトをリセットするかコマンドを撤回するかのいずれかを選択できるプロンプトが出されます。

QMF は、記憶域が不足している状態でシステム・エラーを検出すると、ユーザーの現在の DATA オブジェクトをリセットすることがあります。

ユーザーの報告書記憶域の増加

ユーザーは、大きな報告書を収容するのに十分な仮想記憶域がない場合にも、パフォーマンスが低くなるという経験をすることがあります。たとえば、DSQSRSTG を (TSO、SPRI、APPC、またはネイティブ OS/390 用に) 非常に高い値に設定してあるか、DSQSBSTG パラメーターを非常に低い値に設定してある場合に、ユーザーが何十万行にもものぼる検索を行う照会を実行すると、QMF は非常に少量のデータしかユーザー・メモリーに入れておくことができ

ません。ユーザーは、複雑な報告書のフォーマット設定や報告書のスクロールを行おうとしたときに、応答が遅い（パフォーマンスが低い）と感じます。

報告書のパフォーマンスを最大にするためには、DSQSBSTG または DSQSRSTG パラメーターを使用して、ユーザーに適切な量の仮想記憶域を指定するようにしてください。これらのパラメーターについては、203ページの『DSQSBSTG（報告書データ用の記憶域の調整）』で説明されています。パフォーマンスを最大にするためには、ユーザーが使用されると思われる報告書のうち、最大の報告書に対応できる値を使用してください。

また、ユーザーに予備ファイルを定義することもできます。これについては、207ページの『DSQSPILL（エクストラ記憶域の獲得）』に説明があります。ただし、QMF の操作に基本仮想記憶域を使用すれば、パフォーマンスは向上します。予備ファイルを利用していて仮想記憶域がほとんどないユーザーは、大規模な報告書ではパフォーマンスが低下するのに気付くことがあります。CICS では、予備ファイルには 4K サイズごとに最大 32 767 行が入るため、DSQSBSTG の設定は大きい方が QMF は確実に報告書を完了します。

予備ファイルがあっても、ユーザーはデータが完了していない状態を検出することがあります。これが頻繁に起こるようであれば、別の問題があるかどうかを調べる必要があると思われます。

QMF のパフォーマンスは、(SCROLL BACKWARD コマンドの結果として) QMF がデータ行を必要とし、そのデータが予備ファイルや仮想記憶域にないために、データ・カーソルが再オープンされて行が再びデータベースから検索されるような場合にも、低下することがあります。

記憶域グループのボリューム・スペースの拡大

問題の原因が、制御表記憶域グループのボリューム上に使用可能なスペースが不足していることにある場合には、DB2 ALTER STOGROUP 照会によって、この記憶域グループにボリュームを追加します。この照会の説明については、*DB2 UDB (OS/390 版) SQL 解説書* を参照してください。

CICS 領域のサイズの拡大

QMF トランザクションが CICS 領域内の仮想記憶域を使いきってしまうと、トランザクションは記憶域が使用可能になるのを待って、タイムアウト状態になる場合があります。CICS 領域のサイズは、177ページの『QMF 記憶域要件』に記載されている推奨値に従って指定するようにしてください。これらの推奨値は、インストールされている他の製品で必要な記憶域とは別に必要になるサイズです。

REXX 関数パッケージの使用

報告書で REXX を使用するとき、報告書作成プログラムが QMF と REXX との間で何度も切り替わる場合には、応答時間が長くなることがあります。入出力の時間を短縮するには、REXX 関数パッケージ内にユーザーの REXX プロシージャを入れることができます。

REXX パフォーマンスの詳細については、*IBM Compiler and Library for REXX/370: User's Guide and Reference* を参照してください。

診断援助機能を使用した問題の判別

532ページの『一般的な問題の障害追及』で説明した障害追及の手法を用いて問題を解決できなかった場合、この節を使用して、どの QMF および CMS 診断援助機能が問題の判別に役立つかを判断してください。

症状に応じた適切な診断援助機能の選択

表63は、発生している症状にはどの診断援助機能が必要かを判別するのに役立ちます。診断援助機能が表の一番上に横軸として、また症状が表の左側に縦軸としてリストされています。たとえば、管理プログラム出口ルーチンの使用中に問題が発生した場合には、QMF トレース機能、CICS または TSO の状況に関する情報、および QMF のメッセージおよびヘルプを使用して問題を判別することができます。

表 63. 問題のタイプおよびそのタイプに最適な診断援助機能

	QMF メッセージ 番号	QMF トレース	CICS または OS/390 ダンプ	CICS または OS/390 の 状況情報	ヘルプ・ メッセージ	QMF 以外の メッセージ 番号	エラー・ ログ出力
異常終了	x		x	x			
バッチ・セッション	x	x		x		x	x
呼び出し可能インターフェイス	x	x	x	x		x	
表示パネル	x	x			x	x	x
文書インターフェイス	x	x			x	x	x
エラー・メッセージ	x	x			x	x	x
管理プログラム出口ルーチン	x	x	x	x	x	x	
正しくない出力	x	x			x	x	x

障害追及と問題診断

表 63. 問題のタイプおよびそのタイプに最適な診断援助機能 (続き)

	QMF メッセージ 番号	QMF トレース	CICS または OS/390 ダンプ	CICS または OS/390 の 状況情報	ヘルプ・ メッセージ	QMF 以外の メッセージ 番号	エラー・ ログ出力
初期化	X	X		X	X	X	X
インストール	X				X	X	X
割り込み機能	X	X			X	X	X
ループ		X		X		X	X
パフォーマンス	X	X		X		X	X
印刷	X	X		X	X	X	X
QMF コマンド	X	X			X	X	X
SQL エラー・コード	X	X			X	X	X
終了	X	X		X	X	X	X
ユーザー編集ルーチン	X	X	X	X		X	

QMF メッセージ・サポートを使用した問題の診断

QMF はユーザーのセッション中にさまざまなメッセージを出して、QMF がユーザーの要求を正常に完了したのかあるいはエラーが発生したのかを知らせます。QMF メッセージにはすべて DSQnnnnn という形のメッセージ番号 (nnnnn は 5 桁の数字) が付いています。これらの番号は QMF メッセージおよびコード にリストされており、この資料には問題の解決方法に関するさらに詳しい説明が載っています。

メッセージ番号およびエラーに関するさらに詳しい情報を得るには、ヘルプ・キーを押してメッセージ・ヘルプ・パネルを表示してください。各ヘルプ・パネルにはパネル番号が付いています。問題を IBM に報告する場合、この番号を必要とする場合があります。番号が必ず表示されるように、次のようにグローバル変数 DSQDC_SHOW_PANID を 1 に設定しておいてください。

```
SET GLOBAL (DSQDC_SHOW_PANID=1
```

エラー・メッセージを出した QMF 機能の判別

DSQ で始まる QMF メッセージ番号を使用して、どの QMF コンポーネントがメッセージを出したかを判別することができます。この情報は、問題を特定の QMF 機能に絞り込むのに役立ちます。

QMF 機能とそれに関連するメッセージ番号の範囲を、543ページの表64 に示します。トレース ID は、546ページの『トレース出力の適切な詳細レベルの設

定』に説明があるとおり、各機能について QMF のアクティビティをトレースする場合に使用する ID と同じです。

表 64. QMF 機能およびその機能が発行するメッセージ番号

機能	トレース ID	メッセージ番号
データベース・サービス	I	DSQ10000 - DSQ19999 DSQ30000 - DSQ39999
ダイアログ・コマンド処理	D	DSQ20000 - DSQ29999
表示サービス	E	DSQ40000 - DSQ49999
共通サービスおよびシステム・インターフェース	C	DSQ50000 - DSQ59999
報告書フォーマット設定	F	DSQ60000 - DSQ69999
図表作成	P	DSQ70000 - DSQ79999
フルスクリーン・ウィンドウ	G	DSQ80000 - DSQ89999

表64 に示したメッセージ番号のほかに、QMF の初期化中に以下の範囲のメッセージ番号が生成されることがあります。

DSQI0001 - DSQI0100
DSQ90000 - DSQ99999

システム・エラー・メッセージの処理

システム・エラーは、システムの問題、リソースの問題、または予期しない状態を表します。これらの原因は、QMF、データベース・マネージャー、または何かほかのソフトウェア・コンポーネントに問題があるためと思われます。システム・エラーは、次のメッセージによって示されます。

Sorry, a system error occurred. Your command may not have been executed.

ヘルプ・キーを押してメッセージの詳細を表示するか、*QMF* メッセージおよびコード を参照してください。

データベースに対するコミットされていない変更内容はすべて、システムの問題によって QMF が停止した時点でロールバックされます。システム問題に関するエラー情報は、トレース・データに書き込まれます。これは、QMF を停止するシステム問題に関する唯一の情報リソースです。トレース・データの表示を指示する方法については、549ページの『QMF トレース・データを表示する』を参照してください。Q.ERROR_LOG 表に入るシステム・エラーの情報は、データベースがまだ実行中に発生したシステム・エラーに関する情報のみとなります。

SQL リターン・コードの処理

場合によっては、QMF が表示するメッセージが SQL リターン・コードにマップされることがあります。たとえば、QMF メッセージ DSQ10422 を受け取ったものとして扱います。このメッセージは、SQL リターン・コード -30060 にマップされます (このリターン・コードのテキストは次のとおりです)。

```
YOU ARE NOT AUTHORIZED TO ACCESS specific location.
```

メッセージの中の `specific location` はトークンと呼ばれるプレースホルダーで、実際のデータベースの値を表します。トークンは、QMF が DB2 から受け取る SQL 連絡域 (SQLCA) に入っています。

トークンの値を見つけるには、次のようにしてください。

1. 『QMF トレース機能の使用』の節で説明されている手順を用いて、QMF I2 または ALL トレースを実行します。なお、トレース・データはオンラインのままにして、検索しやすい状態にしておきます。
2. SQL リターン・コードを 16 進数に変換します。たとえば、SQL リターン・コード -30060 は 16 進表記では -746C です。
3. トレース・データの 16 進数番号を見つける。16 進数は、SQLCA というトレース・ブロックに入っています。
4. トレースの右側 (強調フィールド) をブラウザして、トークンを集めます。トークンの SQLCA マッピングについては、*IBM DB2 解説書* を参照してください。
5. 正しいトークンを見つけたら、*DB2 UDB (OS/390 版) 管理の手引き* の操作およびリカバリーの情報を参照し、SQL リターン・コードの原因となった問題を解決します。

QMF トレース機能の使用

QMF には、ユーザーのセッション中に QMF のアクティビティをトレースする機能があります。その機能からのトレース出力は、正しくない出力や脱落している出力、パフォーマンス上の問題、またはループなどのエラーを分析するのに役立ちます。この節では、トレース出力のための記憶域の割り振り方法、機能の開始方法とトレース詳細のレベルの判別方法、および診断するためのトレース・データの表示方法について説明します。

トレース・データ・セットの割り振り

本書の一部のプロシージャでは、QMF が DSQDEBUG データ・セットに記録するトレース情報に加えて、異常終了情報も使用しています。

TSO、SRPI、APPC、またはネイティブ OS/390 用の割り振り: トレース情報は DSQDEBUG データ・セットに記録され、異常終了ダンプ情報は DSQDUMP および SYSUDUMP データ・セットに入ります。QMF セッションを開始する前に これらのデータ・セットが割り振られていることを確認してください。データ・セットは、操作を行おうとするユーザー ID に対して、LOGON プロシージャによって自動的に割り振られます。

QMF セッションの前にこれらのデータ・セットが自動的に割り振られるかどうか分からない場合は、TSO 管理機能で確認してください。割り振りが行われていない場合には、QMF を呼び出して診断セッションを行う前に、次の TSO ステートメントを発行するようにしてください。

```
ATTR DEBUG RECFM( F B A) LRECL(121)
ATTR DUMP RECFM( F B A) LRECL(125)
ALLOC DDNAME(DSQDEBUG) SYSOUT(A) USING(DEBUG)
ALLOC DDNAME(DSQDUMP) SYSOUT(A) USING(DUMP)
ALLOC DDNAME(SYSUDUMP) SYSOUT(A)
```

図 153. TSO 用のデータ・セットの割り振り

CICS 用の割り振り: トレースは DSQDEBUG データ・セットに記録されます。このデータ・セットは、CICS の始動 JCL で割り振る必要があります。トレースは、同じ CICS アドレス・スペース内のすべてのユーザー間で共用することができます。

トレース機能の開始

1. DSQDEBUG の *ddname* を用いて、データ・セットを割り振ります。

トレース機能は、トレースの結果を DSQDEBUG データ・セット (印刷または表示が可能) に書き込みます。このデータ・セットは、トレース用のみ使用されるものです。

2. トレース・オプションを決定します。

これらのオプションによって、トレースの内容および詳細レベルを制御します。トレース・オプションの選択の詳細については、546ページの『トレース出力の適切な詳細レベルの設定』を参照してください。

217ページの『DSQSDEBUG (トレース詳細レベルの設定)』で説明してあり、QMF の開始時には DSQSDEBUG プログラム・パラメーターには値 ALL を指定してください。この値は、QMF のアクティビティーを QMF の初期化中に発生する可能性のあるプログラム障害も含めて、最も詳細なレベルまでトレースします。

出力が 32 767 行を超える場合には、それを入れるための一時データ・キューを使用する必要があります。218ページの『DSQSDBQT (CICS のトレース・データ用記憶域のタイプの指定)』および219ページの『DSQSDBN (CICS のトレース・データ用記憶域の名前の指定)』に、トレース出力を入れるためのキューのタイプと名前の指定方法が説明してあります。

3. これらのオプションを QMF トレースに指定します。

QMF セッション中、トレース・オプションの一部のセットは常に有効になっています。現行のトレース・オプションを指定変更するには、次のようにいくつかの方法があります。

- ユーザーに、次の QMF コマンドを入力するよう指示する。

```
SET PROFILE (T=value
```

ここで value は、ALL か、トレース出力での QMF 機能とその詳細レベルを示すストリングです。詳細レベルについては、『トレース出力の適切な詳細レベルの設定』で説明します。

- ユーザーのプロファイルの TRACE フィールドに SQL UPDATE ステートメントを使用する。これは前の方法と同じ働きをします。ユーザーに、データベースに再接続して新しい値で初期化するように指示してください。たとえば、パスワード MYPW を持つユーザー JONES は、次のように入力することができます。

```
CONNECT JONES (PA=MYPW
```

- DB2 CONNECT 権限を持っていないユーザーは、現在の QMF セッションを終了し、別のセッションを開始して値を初期化することができる。

4. QMF 初期化中に警告またはシステム・エラーが発生した場合には、トレース・データ・セットにアクセスします。

DSQDEBUB を調べると、エラーの理由を判別するのに役立ちます。

DSQDEBUB データ・セットのアクセス方法の詳細については、549ページの『QMF トレース・データを表示する』を参照してください。

5. トレース出力を解読します。

DSQDEBUB ファイルは、表示または印刷して分析することができます。トレース出力の印刷または表示の詳細については、549ページの『QMF トレース・データを表示する』を参照してください。

トレース出力の適切な詳細レベルの設定

すべての QMF 機能を最も詳細なレベルまでトレースしたい場合には、トレースの際に値 ALL を使用してください。

個々の QMF 機能をトレースしたい場合には、トレースしたい QMF 機能を示す英字と各機能のトレース・データをどの詳細レベルでトレースするかを示す数字を組み合わせた文字ストリングを用いて、Q.PROFILES の TRACE 列を更新してください。各文字と番号を、ペアにする必要があります。

値 1 (機能を中程度の詳細レベルでトレースする)

値 2 (機能を最高の詳細レベルでトレースする)

文字ストリングで指定した機能だけがトレースされます。各 QMF 機能を表す英字を、次の表に示します。

トレース ID

QMF 機能

A	アプリケーション・サポート・サービス
C	共通サービスおよびシステム・インターフェース
D	ダイアログ・コマンド処理
E	指示照会、QBE、表編集機能、グローバル変数リスト、およびデータベース・オブジェクト・リストなどの、QMF の各部分の表示サービス
F	報告書フォーマット設定
G	QBE、指示照会、および表編集機能フルスクリーン・ウィンドウ
I	データベース・サービス
L	メッセージおよびコマンドのロギング
P	図表作成 (対話式図表ユーティリティー)
R	記憶管理機能
U	ユーザー編集出口ルーチンや管理プログラム出口ルーチンなどの、ユーザー出口

たとえば、メッセージおよびコマンド・ログ記録を最も詳細なレベルまで、アプリケーション・サポート・サービスを中程度のレベルまで、そして共通サービスとシステム・インターフェースを最も詳細なレベルまで、トレースしたい場合は、次のコマンドを使用します。

```
SET PROFILE (T=L2A1C2
```

QMF セッション中にユーザーのアクティビティーを細かく記録するには、L1 および L2 のトレース・レコードを使用します。値 L1 を使用すると、QMF によって出されたすべてのメッセージをレコードに書き込みます。L2 を使用すると、L1 のすべてのレコードのほかに、QMF コマンドの実行について記述する追加のレコードの書き込みも行います。ユーザーが発行した各コマンド

障害追及と問題診断

と、QMF がそのコマンドにどのように応答したかをログに記録するには、L2 トレース・レコードを使用してください。図154 は、ユーザーが表にない列を指定したためにエラーとなった RUN QUERY コマンドの例を示しています。

```
-----
***** 93/12/15 20:39 *****
USERID: KRIS
AUTHORIZATION-ID: KRIS
COMMAND TEXT:
RUN QUERY
-----
***** 93/12/15 20:39 *****
USERID: KRIS
AUTHORIZATION-ID: KRIS
MESSAGE NUMBER: DSQ12405
MESSAGE TEXT:
Column name DATE is not in table STAFF.
&01: DATE
&02: STAFF
&09: -205
-----
```

図154. L2 トレース・コードを使用して、ユーザーのコマンドおよびメッセージをトレースする

DSQDEBUG データ・セットには、メッセージが出された順序で入っています。コマンドが組み込まれるときは、それらのコマンドも発生順になり、メッセージと組み合わせられます。メッセージは、データ・セットの中でそのメッセージの直前に出てくるコマンドと関連しています。

QMF メッセージには、表や列の名前などの変化するメッセージ部分を表す変数が含まれています。トレース・データを使用すると、ユーザーが変数の含まれているメッセージを解読するのに役立ちます。たとえば、図154 に示されているメッセージは、QMF メッセージおよびコード で次のように表示されます。

```
Column &01 is not in table &02.
```

図154 の下半分は、メッセージの &01 の値が DATE で、&02 の値が STAFF であることを示しています。メッセージにこれらの値を代入することによって、ユーザーの問題解決に役立ちます。

これらの変数は、エラー・メッセージに関連したヘルプ・パネルの定義にも現れることがあります。トレース・データの変数値をヘルプ・コマンドと組み合わせ、メッセージ・ヘルプ・パネルを再構成してください。

モジュール・レベルでのトレース

重要: モジュール・レベルでのトレースは、IBM サービス・レベル 2 のガイドダンスのもとでのみ実行してください。

SET PROFILE コマンドおよびモジュール DSQUTRAC を使用して、特定のモジュールのトレースをオンにすることができます。たとえば、行管理機能または合計管理機能をトレースしないで、フォーマッター・プログラム・バッファ管理プログラムだけをトレースすることができます。モジュール・レベルでのトレースのための値は、以下のとおりです。

- 値 3 (コンポーネント内の特定のプログラムの詳細トレースを行い、コンポーネント内のその他のすべてのプログラムの入り口と出口をトレースする)
- 値 4 (1 つのモジュールのみをトレースする)

モジュール・レベルのトレースを作成するには、トレースしたいモジュールを DSQUTRAC モジュール内にリストしてください。次に、そのモジュールをアセンブルおよびリンク・エディットします。モジュールが作成されたら、そのモジュールを使用可能にする必要があります。その後、次のコマンドを実行することができます。

```
SET PROFILE (TRACE F4
```

QMF トレース・データを表示する

DSQDEBUG には、トレース機能によって記録された情報が入ります。トレースを使用する場合には、QMF を開始する前に このデータ・セットが割り振られていなければなりません。このデータ・セットは、印刷または表示できるように割り振ることができます。

CICS では、ユーザーの数およびそのユーザーのセッションをトレースする詳細レベルによっては、トレース・データが非常に長くなることがあります。

TSO での印刷または表示: DSQDEBUG データ・セットは、TSO 環境ではユーザーの LOGON プロファイルによって自動的に割り振られる場合があります。その場合でも、元の割り振りが各自の必要を満たしていなければ (たとえば、元の割り振りでは DSQDEBUG を PRINT ファイルとして定義しているが、実際にはそれを表示したい場合など)、再割り振りすることができます。

印刷用に割り振り (または再割り振り) するには、DSQDEBUG を PRINT ファイルとして定義する次のステートメントを出してください。

```
FREE FILE(DSQDEBUG)
ATTR DEBUG RECFM( F B A) LRECL(121)
ALLOC DDNAME(DSQDEBUG) SYSOUT(A) USING(DEBUG)
```


障害追及と問題診断

割り振りには、最初のバイトが ANSI 紙送り制御文字の 121 文字固定長レコードが含まれています。トレース情報では、ANSI 制御文字を含まない 120 文字が 1 行にフォーマット設定されます。

注意: DSQDEBUG からの出力を HOLD キューに入れるよう割り振る場合には、OUTPUT キューへの出力を解放するために次の TSO コマンドを出す必要があります。

```
FREE DDNAME(DSQDEBUG)
```

また、次に示すステートメントを出すことによって、オンライン編集プログラムを使用して表示可能な順次データ・セットとして DSQDEBUG を割り振る(または再割り振りする) こともできます。データ・セットは、最初のバイトが ANSI 紙送り制御文字である 81 文字の固定長レコードで構成されます。トレース情報は、ANSI 制御文字を含まない 80 文字が 1 行にフォーマット設定されます。

```
FREE FILE(DSQDEBUG)
ATTR DEBUG RECFM( F B A) LRECL(81)
ALLOC DDNAME(DSQDEBUG) DSNNAME(DEBUG.LIST) NEW KEEP
```

CICS での印刷または表示: トレースは DSQDEBUG データ・セットに記録されます。このデータ・セットは、CICS 始動 JCL において割り振ってください。

QMF の初期化中に警告またはシステム・エラーが発生した場合、QMF のトレース・データ・セットを調べてエラーの原因を理解する必要があります。

CICS では、トレース・データ・セットは余分な領域のデータ・セットとして記述されています。トレース・データ・セットは、CICS 表の中には、551ページの図155 に示すように DCT TYPE=SDSCI コマンドおよび DCT TYPE=EXTRA コマンドによって記述されます。


```

* TRACE DATA SET
  DFHDCT TYPE=SDSCI,DSCNAME=DSQDEBUG,
    RECFORM=VARBLK,
    RECSIZE=121,
    BLKSIZE=6050,
    TYPEFILE=OUTPUT
*
*
  TITLE 'DSQDCT - CICS DESTINATION CONTROL TABLE'
*
* TRACE DATA SET
*
DSQD  DFHDCT TYPE=EXTRA,DESTID=DSQD,DSCNAME=DSQDEBUG,RSL=1

```

図 155. CICS 環境でのトレース・データ・セットの記述

単一の CICS 領域内のすべての QMF ユーザーからの QMF トレース・データが、単一のトレース・データ・セットに書き込まれます。各トレース項目には、それを記録したユーザーの端末 ID が入れられます。

CICS 領域がアクティブの間にトレース・データ・セットを調べるには、CICS キュー ID の DSQD を用いてトレース・データ・セットをクローズしなければなりません。この ID は、CICS 提供トランザクション CEMT の使用中に使用することができます。トレース・データ・セットをクローズすると、トレース・データ・セットの印刷または表示ができます。

トレース・データ・セットがクローズされているときは、CICS ユーザーが他のレコードを書き込むことはありません。この状態では、QMF はトレース・レコードを記録せずに操作を続行することになります。QMF トレースを再び使用可能にするには、CICS 提供トランザクション CEMT を使用して、CICS キュー ID の DSQD を用いてトレース・データ・セットをオープンしてください。

QMF サービス・レベルの判別

サービス・レベル情報は、次の場合に表示されます。

- 呼び出しで (または、Q.PROFILES から) T=ALL が指定されている場合
- コマンドとして SET (TRACE ALL が指定されていた場合
- 異常終了が発生した場合

QMF サービス・レベルは、以下の手順で判別することができます。

1. SET PROFILE コマンド (T=ALL を入力する。
2. SET PROFILE コマンド (T=NONE を入力する。
3. QMF を終了する。

障害追及と問題診断

4. DSQDEBUG ファイルを調べる。

結果のトレースには、プログラムとそのバージョン、日付、および時刻が示されます。次のトレース例にあるように、モジュールにプログラム一時修正 (PTF) が適用されている場合には、プログラム診断依頼書 (APAR) 番号も示されます。

```
** DSQFQWRM: ENTERED FROM DSQFMCTL ***  
V7R1.00 00/01/30 12:00 PNxxxxx
```

APAR PNxxxxx は、サービスが適用された最新の APAR です。

トレース機能の停止

トレース機能を用いて詳細な診断情報を収集すると、多くの場合トレース・データの記憶域キューはすぐに満杯になってしまうため、適当な時点でトレース機能を停止する必要があります。

トレースをオフにするには、QMF 内から次のコマンドを発行します。

```
SET PROFILE (T=NONE
```

QMF セッションが終了するまでトレースをオンのままにした場合、次に QMF を開始するときにはデフォルトの設定によってトレースは NONE に設定されます。QMF の開始時には、217ページの『DSQSDEBUG (トレース詳細レベルの設定)』で説明されているプログラム・パラメーター DSQSDEBUG がこのトレースを制御します。

異常終了の診断

各自の環境で使用可能な TSO、OS/390、または CICS 機能の診断機能を用いて、異常終了を診断する必要が生じることがあります。(CICS では、異常終了情報は DFHDMPx データ・セットに記録されます。このデータ・セットは、CICS の始動 JCL で割り振る必要があります。) 大部分の QMF プログラムには、診断出力で識別するのに役立つスタンプが含まれています。553ページの図156 にその例を示します。

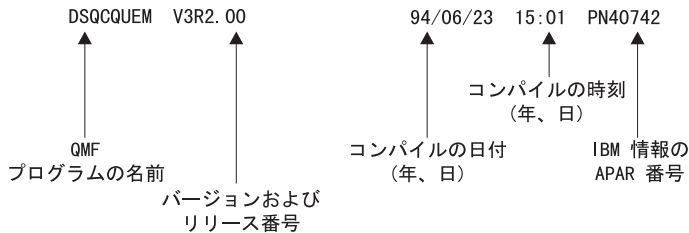


図 156. QMF プログラムを識別するスタンプの例

OS/390 診断機能の使用

OS/390 での異常終了を診断するには、*MVS Diagnosis: Tools and Service Aids* に記載されている手順を使用するか、QMF 異常終了ハンドラーが使用可能である必要があります。

QMF は開始時に、異常終了ハンドラーを確立します。QMF に障害が起こると異常終了ハンドラーが制御を獲得し、エラーを記録して、環境の終結処置を行います。完了後に、異常終了ハンドラーはオペレーティング・システムに制御を戻し、オペレーティング・システムが異常終了処理を継続できるようにします。

ユーザー編集コードの処理中または管理プログラムの実行中に異常終了が発生すると、問題診断を支援するために、ダンプ内に追加の区域が加えられます。

ユーザー編集コードの場合には、DXEPCS、入力域、および結果域が出力に追加されます。

管理プログラムの場合には、DXEXCBA および DXEGOV が出力に追加されません。

CICS 診断機能の使用

QMF での異常終了を診断するには、*CICS Problem Determination Guide* に記載されている手順を使用する必要があります。別のプログラムが QMF を異常終了させた可能性を考えると、トランザクションの CICS ダンプの中の情報を調べる必要が出てくるので、その際にこの手順が役立つためです。トランザクション・ダンプには、異常終了の発生時に CICS 領域で実行中だったプログラムの詳細なアクティビティの記録が示されています。

異常終了の原因となったプログラムは、QMF である場合も別のプログラムである場合もあります。この章で説明されている QMF 診断機能にエラーの原因に関する十分な情報が含まれていない場合には、CICS 実行診断機能 (CEDF) を使用すると QMF 異常終了の診断に役立つことがあります。

CICS 診断出力での QMF の識別: CICS 診断機能を使用して QMF での異常終了を診断する場合、以下の情報が CICS 出力の中から QMF プログラムを識別するのに役立ちます。

- QMF プログラム名は、接頭部 DSQ で始まっている。
- QMF はアセンブラ言語プログラムなので、CICS LINK ステートメントではなく、標準のアセンブラ呼び出しを出す。
- QMF は CICS で実行する場合、すべてのシステム・サービスに対して標準の EXEC CICS ステートメントを出す。
- QMF は GDDM 製品に対して、内部呼び出しインターフェースを使用する。
- QMF はデータベースに対して、標準 EXEC SQL ステートメントを出す。
- QMF は EXEC CICS ABEND コマンドは出さない。

CICS 異常終了メッセージの表示の定義: QMF が異常終了した場合やオペレーターがトランザクションを取り消した場合など、一部の状況では、CICS がユーザーの端末装置に異常終了を伝えるメッセージを送信することがあります。QMF は GDDM を使用して表示サービスを提供するフルスクリーン・アプリケーションであるので、どのようにして異常終了メッセージを表示したいかを CICS に定義する必要があります。

CICS オンライン・リソース定義 (RDO) を使用して、CICS TYPETERM 定義に CICS エラー・メッセージの診断表示属性を設定してください。

TYPETERM は部分端末定義で、1 つの定義で多くの端末表示を簡単に定義できるようにするものです。555ページの図157 に、使用できる診断表示属性の例を示します。

555ページの図157 に示す定義を使用すると、画面最下部の QMF メッセージ行の下にメッセージを表示します。メッセージは下線付きの赤い文字で表示され、画面表示の他の部分よりも高輝度になっています。この定義は、ユーザーが一定時間内に入力を行わない場合に QMF トランザクションをタイムアウトになるように定義した場合にも役立ちます。このタイプのトランザクション・タイムアウトの場合には、QMF 表示が画面にそのまま残るため、メッセージが読み取り可能なのは画面の最下部だけになるからです。

DIAGNOSTIC DISPLAY					
ERRLastline	: Yes	No		Yes	
ERRIntensify	: Yes	No		Yes	
ERRColor	: Red	NO		Blue	Red Pink Green
				Turquoise	Yellow NEutral
ERRHilight	: Underline	No		Blink	Reverse Underline

図 157. CICS 診断表示用の TYPETERM 指定

TSO での QMF 割り込み機能の使用

TSO では、QMF コマンドが非アクティブでも QMF 割り込みハンドラーをアクティブ化することができます。QMF に割り込みをかけるには、PA1 キーを押してください。

QMF プロシージャ・パネルを表示するには、画面を最新表示にする必要があります。それには PA2 キーを押してください。

問題に関する情報を収集するには、QMF 割り込み機能を使用します。割り込み機能を使用すると、異常終了ダンプを作成するか、あるいは、トレース情報を表示するか DSQDEBUG データ・セットへ書き込ませることができます。

割り込み機能は、問題診断のオブジェクトとなっているユーザーのログオン ID のもとで使用することができます。ただし、発生した際にその場にいたのでない限り、まず問題を再現する必要があります。

割り込みの作成

割り込み機能を使用する場合の最初のステップは、アテンション割り込みの作成です。大部分のシステム構成では、ATTN キーを押すか、リセット・キーと PA1 キーを一緒に押すことによって、アテンション割り込みを作成することができます。これらのキーが機能しない場合には、現在のシステム構成に合わせて適宜資料を参照して、割り込みの作成についての詳しい情報を入手してください。

割り込み機能は、次のメッセージを表示して応答します。

DSQ50546 QMF command interrupted! Clear screen and press enter.

図 158. QMF 割り込みハンドラー・プロンプト 1

割り込み作成後のトレース情報の表示

割り込みメッセージが表示された後、メッセージの指示に従って CLEAR キーおよび ENTER キーを押してください。すると、次のメッセージが表示されます。

```
DSQ50547 QMF command interrupted!   Do one of the following:
==> To continue QMF command,        type 'CONT'.
==> To cancel QMF command,          type 'CANCEL'.
==> To enter QMF debug,              type 'DEBUG'.
```

図 159. QMF 割り込みハンドラーのプロンプト 2

CONT、CANCEL、DEBUG のうち選択したものをタイプしてから、ENTER キーを押してください。

- 割り込みがなかったかのように、割り込みを行う前の位置に制御を戻す場合には、CONT を入力します。
- 割り込み時点で実行中のコマンドがあったときにはそれを停止させるという場合には、CANCEL を入力します。そうすると、キーボードのロックが解除され、QMF は次のコマンドを待機します。必ずコマンドを取り消すことができるとは限らないことに注意してください。
- 図160 に示すような診断情報を得たい場合には、DEBUG を入力します。

```
-- OK, QMF debug entered. QMF CSECT trace is:
   DSQDSUPV -> DSQDSUPX -> DSQEADAP -> DSQEMAIN -> DSQEINPT -> ENDTRACE
==> To continue QMF command,        type 'CONT'
==> To cancel QMF command,          type 'CANCEL'
==> To abnormally terminate QMF,    type 'ABEND'
==> To set QMF trace,               type 'TRACEALL' or 'TRACENONE'
```

図 160. 割り込み画面に DEBUG を入力したとき出される診断情報

この例の 2 行目にあるトレース情報は、割り込みの時点で制御は CSECT DSQEINPT にあり、その制御は CSECT の DSQDSUPV、DSQDSUPX、DSQEADAP、DSQEMAIN を順番に通ってこの CSECT に達していたことを示しています。

図160 のデバッグ・パネルに応答するには、次の説明に従って、CONT、CANCEL、ABEND、TRACEALL、または TRACENONE のいずれかを入力してください。入力したら、ENTER キーを押します。

- 割り込みがなかったかのように、割り込みを行う前の位置に制御を戻す場合には、CONT を入力します。

- 割り込み時点で実行中のコマンドがあったときにはそれを停止させるという場合には、CANCEL を入力します。そうすると、キーボードのロックが解除され、QMF は次のコマンドを待機します。ただし、必ずコマンドを取り消せるとは限らないので注意してください。
- QMF を異常終了させて異常終了ダンプを作成する場合には、ABEND を入力します (そのセッションで DSQDUMP データ・セットが割り振られていた場合)。
- QMF が最も詳細なレベルのトレース出力を DSQDEBUG データ・セットに追加し始めるようにする場合には、TRACEALL を入力します。制御は割り込み時点の位置に戻ります。
- QMF がトレース出力を DSQDEBUG データ・セットに追加するのを停止する場合には、TRACENONE を入力します。制御は割り込み時点の位置に戻ります。

Q.ERROR_LOG 表からのエラー・ログ報告書の使用

Q.ERROR_LOG 表は、リソースの問題およびソフトウェアの障害によって発生した可能性のある問題に関する情報を記録する、QMF 制御表です。表の構成を、表65 に示します。

表 65. Q.ERROR_LOG 表の構成

列名	データ・タイプ	長さ (バイト単位)	ヌルの使用	機能 / 値
DATESTAMP	CHAR	8	不可	エラーの発生した日付。フォーマットは <code>yyyymmdd</code> 。
TIMESTAMP	CHAR	5	不可	エラーが発生した時刻。フォーマットは <code>hh:mm</code> で、 <code>hh</code> は時、 <code>mm</code> は分。
USERID	CHAR	8	不可	ログオン ID、または CICS ではエラーを経験したユーザーの端末装置 ID。
MSG_NO	CHAR	8	不可	エラーと一緒に発行された QMF メッセージ番号。
MSGTEXT	VARCHAR	254	不可	メッセージのテキスト。SQL エラーでは、SQLCA からのデータがこの列に入ることがある。

長いエラー・メッセージは、それを表すのに表の複数の行を必要とすることがあります。その場合には、MSGTEXT 列を除く各列の値は繰り返し出されま

障害追及と問題診断

す。MSGTEXT 列には、各行にメッセージの一部が入ります。メッセージの部分には先頭に 1)、2)、3) などの番号が付けられて、メッセージ内のどの部分にあるものかが示されます。

問題の診断の参考にするために、Q.ERROR_LOG 表を照会してエラーに関する情報を取得することができます。照会を行うには、問題を経験したユーザーの端末装置 ID および問題が発生したおよその時刻を知っている必要があります。図161 は、照会のフォーマットを示しています。

```
SELECT TIMESTAMP, MSG_NO, MSGTEXT
  FROM Q.ERROR_LOG
 WHERE USERID = 'terminal_id' (for CICS)
  WHERE USERID = 'user_id'    (for other than CICS)
    AND DATESTAMP = 'date'
    AND TIMESTAMP BETWEEN 'time1' AND 'time2'
 ORDER BY TIMESTAMP, MSG_NO, MSGTEXT
```

図 161. 問題情報を得るためのエラー・ログの照会

指定する日付と時間には、有効なフォーマットを使用するようにしてください。これらのフォーマットは557ページの表65 に示されています。

IBM への問題報告

問題を IBM に報告する前に、IBM のソフトウェア・サポート機能 (SSF) を調べて、その問題がすでに報告されていないかどうかを確認してください。報告済みの多くの問題については、IBM サポート担当者がプログラム診断依頼書 (APAR) を用意しており、それには問題の解決方法に関する有用な情報が含まれています。

ServiceLink またはその他の機能を介して SSF にアクセスする場合は、『*ServiceLink* を使用して既に報告されている問題の検索』を読んで、問題をうまく見つけるにはどのように検索キーワードのストリングを指定したらよいのかその方法を参照してください。*ServiceLink* にアクセスできない場合には、直接 561ページの『IBM サポート・センターと共に作業する』に進んでください。

ServiceLink を使用して既に報告されている問題の検索

問題を記述する検索語のストリングを構成して、SSF を検索してください。QMF OS/390 6 の検索語の各ストリングは、コンポーネント ID の 566872101 と、問題を経験した QMF 各国語環境に対応するリリース番号 (559ページの表 66 に示してある番号) で始まります。

表 66. QMF 基本製品および NLF のリリース番号

NLF	ID
ブラジル・ポルトガル語	65A
デンマーク語	654
英語	610
フランス語	655
ドイツ語	656
イタリア語	657
日本語	658
韓国語	659
中国語 (簡体字)	653
スペイン語	65B
スウェーデン語	65C
スイス・フランス語	65D
スイス・ドイツ語	65E
英大文字	651

560ページの図162 のフローチャートは、問題の各特性を判別するために、検索キーワードをどのように構築するかを示しています。

障害追及と問題診断

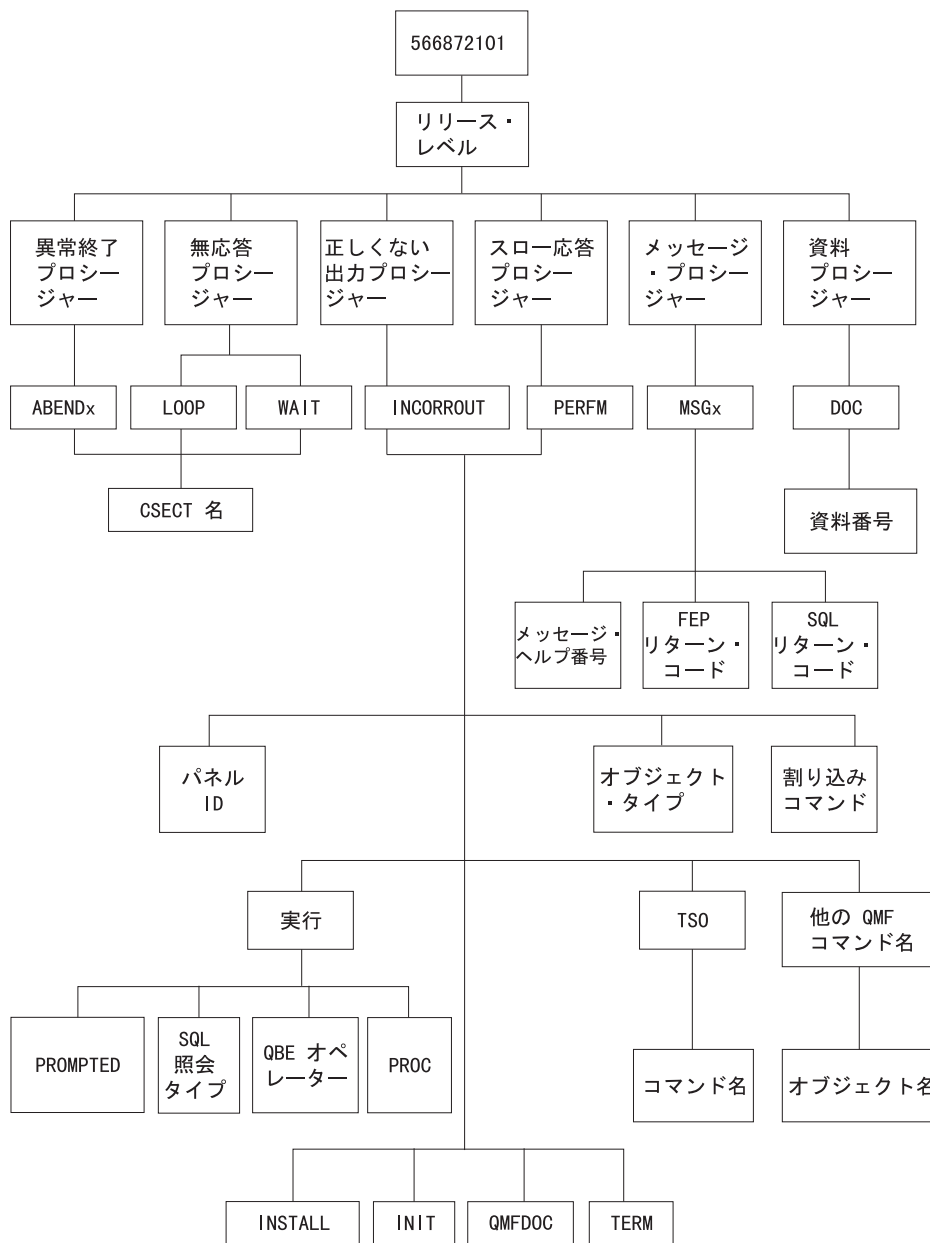


図 162. キーワード・タイプの図表。(この表を上から下へとたどって、キーワードを決定してください。)

たとえば検索している問題が、ユーザーが英語の QMF セッションの実行中に、DSQFDTBL 制御セクション (CSECT) で発生した異常終了タイプ 0C4 の問題である場合には、次の検索句を使用してください。

CSECT 名を見つけるには、ABEND CSECT NAME というヘッダーのトレース出力のセクションを見ます。CSECT 名にはアスタリスクが付けられています。QMF トレース機能の使用方法に関する詳細については、544ページの『QMF トレース機能の使用』を参照してください。

すでに分かっている QMF 問題を見つけるために SSF を検索する詳細な方法については、*ServiceLink User's Guide* を参照してください。

IBM サポート・センターと共に作業する

問題の診断がうまく進まず、この章で説明した診断援助機能も使用したという場合には、IBM サポート担当者に連絡して問題を報告してください。

問題の診断を支援するために、サポートの担当者から問題に関するさらに詳細な情報を要求する場合があります。たとえば、QMF での異常終了を報告するために連絡したとします。

この場合、エラーを引き起こしたかもしれないと疑っている問題の CSECT に関するいくつかの情報を提供する必要があります。多くの場合、このような種類の情報は 544ページの『QMF トレース機能の使用』で説明してあるトレース機能を使用して得ることができます。IBM 担当者は、541ページの表63 で示しているその他の診断エイドが生成した文書を必要とする場合もあります。この文書は、担当員が問題を再現するのに役立ちます。

付録A. 問題が発生した場合

インストール・プロセス時に、メッセージが表示されることがあります。表示されるメッセージのいくつかは、無視しても構わない通知メッセージです。その他のメッセージは、訂正アクションを必要とする警告メッセージまたはエラー・メッセージです。この節では、インストール時に発生する最も一般的なエラーのいくつかについて説明します。このリストは、QMF または他の製品に関するメッセージおよびコードの解説書の代わりになるものではありません。表示されたメッセージがこのリストに記載されていない場合は、該当するメッセージおよびコード解説書を調べてください。

表示される可能性のあるエラー・メッセージ

次のようなエラー・メッセージを 1 つ以上受け取ることがあります。

ABENDASRA

- QMF 始動時。
 - GDDM リンク・エディットが正常に実行されたか確認する。
 - GDDM IVP が領域ベースで正常に行われているか確認する。
 - QMF で正しくリンク・エディットされているか確認する。
 - 領域が QMF バージョン 7 LOADLIB とマップ・グループを割り振るようにする。
- モジュール DSQQMFE CSECT ADM において。

この問題は通常、GDDM 障害です。GDDM が CICS に対して正しくインストールおよび調整されているかを確認する。GDDM が CICS と同じ CSIゾーンに配置されているかを確認する。
- モジュール DSQQMFE CSECT DSQEGINT において。

GDDM が CICS に対してカスタマイズされ、かつ PPT 項目が GDDM モジュール ADMASPLC 用に存在しているかを確認する。
- モジュール DSQQMFE CSECT DSQIELI において。

PPT 項目が DB2 UDB (OS/390 版) インターフェース・モジュール DSQIELI 用に存在しているかを確認する。
- モジュール DSQCBST CSECT DSQCMCVP において。

QMF サービスが適用された後で、QMF CODE が索引ライブラリー内にある場合に OS/390 LLA REFRESH が行われたかを確認する。

問題が発生した場合

- R15 に FFFFFFFE を含んだ ABEND0C1 が表示された場合。
DSQ1ELNK を再実行する (特に、QMF 保守の設定後)。
- QMF の終了時。
管理プログラムが正しくリンクされているか検査する。ジョブ DSQ1EGLK を検討する。
- ABEND0C4 と DFHSM0102 が一緒に表示された場合。
このエラーは、照会の実行時またはヘルプ機能キーを押したときに発生します。DSQPNLE 用の FCT に RECFM=V が含まれていることを確認する。
- HELP または RUN コマンドの発行時。
QMF データ・セット DSQPNLE (ヘルプ・テキストおよび他の画面テキストを含む) が、正しくインストールされなかったか、または CICS 領域を開始したジョブに割り振られなかった。
 - FCT 項目が 86ページの『FCT (ファイル管理テーブル)』の説明のように正しく定義されているか確かめる。
 - DSQPNLE の DD ステートメントが CICS 領域を開始するジョブ・ストリームに存在しているか確かめる。DD ステートメントについては、89ページの『ステップ 24 -- CICS 始動ジョブ・ストリームの更新』で説明しています。

DSQPNLE データ・セットに関連づけられているコンソール・エラー・メッセージを探する必要もあります。

AEY9 ABEND

DB2 UDB (OS/390 版) 接続機能は、CICS 領域ではアクティブになりません。DSNC トランザクションを使用して接続機能を開始してください。

AZTS ABEND

GDDM が IOSYNCH=YES を指定して実行されているか確認する。

DSNT302I

名前 profilex が無効。DSQ1TBJ2 で生成される通常メッセージ。このメッセージは無視してください。

DSQ10297

サブシステム ID が無効。このエラーは、ISPF の始動時または呼び出し可能インターフェースの使用時に起こる可能性があります。ISPF 始動パラメーターを調べて、s=xxxx または DSQSSUBS=xxxx であることを確かめる。詳細については、76ページの『ISPF を使用した QMF の開始』を参照してください。

DSQ10493

このメッセージは、データベース許可エラーを示します。DB2 UDB (OS/390 版) リソース管理テーブル (RCT) に、QMF を開始するために使用するトランザクション ID の項目が含まれているか確かめてください。たとえば、QMF を開始するために CICS トランザクション ID QMFE を使用する場合には、次の項目をコーディングしてください。

```
DSNCRCT TYPE=ENTRY, TXID=QMFE, PLAN=QMF710, AUTH=DEPT1
```

この例では、許可 ID は DEPT1 であり、プラン ID は QMF710 です。

DSQ36805

SQLCODE 805。このエラーは始動時に起こります。SQLCODE 805 から戻されるすべてのトークンを記録し、DB2 UDB (OS/390 版) メッセージおよびコード、の SQLCODE 805 に関する指示に従ってください。

DSQI004I

GDDM エラー。

DSQI0026

このメッセージは通常、始動時に出されます。QMFE トランザクションがクリア画面から入力されているか確認してください。

G050 ABEND

CICS に合わせてユーザーが調整した GDDM のリリース・レベルが、CICS 領域を開始するためにジョブ・ストリームで使用している GDDM のリリース・レベルと一致しているか確かめてください。

IDC3012I

```
Entry QMFCAT.DSNDBC.DSQDBCTL.PROFILEX.I0001.A001.
```

IDC3009I

**VSAM カタログのリターン・コードは 8 - 理由コードは IGGOCLAS3-42.

IDC0551I

** 項目 QMFCAT.DSNDBC.DSQDBCTL.PROFILEX.I001 A001 は削除されない。

これらは、DSQ1VSTP の実行時に VSAM クラスターの削除および除去の際に出される通常メッセージです。これらのメッセージは無視してください。

問題が発生した場合

IEW0342

ライブラリーにモジュール xxxxxxxx は入っていない。

QMF が、まだ存在していないモジュールを置き換えようとしています。このメッセージは、ロード・モジュールがリンク・エディットされるたびに表示されます。

IEW0461

この警告メッセージが表示されるのは、次のいずれかの理由によります。

- 印刷される記号が、未解決の外部参照である。
- NCAL が指定されている。
- 参照が、制限付き NO=CALL または NEVERCALL に応じてマーク付けされている。

このメッセージは、3 つのロード・モジュール (DSQUXIA、DSQUXIC、および DSQUXIP) に関して出されます。これらのモジュールは、サンプル・アセンブラー、COBOL、および PL/I ユーザー出口です。これらのメッセージは無視してください。

DSQ22843

GDDM が IOSYNCH=YES を指定して実行されているか確認する。

QMF IVP が失敗してメッセージ A GDDM graphics printer nickname is required for printer が表示された場合には、GDDM ニックネーム定義にエラーがあります。

QMF IVP には、照会を印刷するためのステップが組み込まれています。このステップでは、GDDM ニックネームが必要になります。インストール先システムで GDDM ニックネームを使用する場合は、IVP プロシージャの PRINT QUERY ステートメントを PRINT QUERY (PRINTER = *gddmnickname* に変更してください。GDDM プリンターのニックネームを作成するプロシージャについては、323ページの『第17章 ユーザーによるオブジェクトの印刷』に述べてあります。インストール先システムで GDDM ニックネームを使用しない場合は、IVP プロシージャの PRINT を PRINT PROFILE で置き換えてください。QMF は、ニックネームを使用せずにプロファイルを印刷します。

警告メッセージ

QMF の開始後に出される警告メッセージの原因としては、次のものが考えられます。

- TSO と同じ AUTHID

TSO と CICS で同じデータベース AUTHID を使用する場合は、TSO コマンドを含む QMF コマンド同義語表を使用することができます。この警告は QMF の実行には影響を及ぼしませんが、このようなコマンド同義語は CICS セッション中には使用できません。

CICS セッションに固有のプロファイルを割り振り、警告メッセージを除去するには、89ページの『ステップ 23 -- QMF プロファイルの調整』の説明を参照してください。

- その他の原因

警告メッセージが出された場合、その警告の原因が QMF トレース・データ・セット DSQDEBUG に書き込まれます。DD 名 DSQDEBUG は、CICS 領域を開始したジョブ・ストリームに記述されます。

エラー・メッセージが表示されなかった場合

場合によっては、問題が生じていてもエラー・メッセージが表示されないことがあります。このようなエラーの最も一般的なタイプは、出力の誤りです。たとえば、QMF ホーム・パネルではバージョン 7 リリース 1 を読み取る代わりに、別のリリースを指します。この場合、ADMGGMAP DD 名が QMF710.DSQMAPn データ・セットを指すようにしてください。一般的な障害追及および特に出力の誤りに関する詳細については、531ページの『第24章 障害追及と問題診断』を参照してください。

QMF トレース・データ・セット DSQDEBUG へのアクセス

QMF の初期化中に警告またはシステム・エラーが発生した場合、QMF のトレース・データ・セットを調べてエラーの原因を理解する必要があります。CICS では、トレース・データ・セットは余分に区画されたデータ・セットとして記述されます。トレース・データ・セットは、568ページの図163 に示すように、DCT TYPE=SDSCI および DCT TYPE=EXTRA によって CICS 表に記述されます。

問題が発生した場合

```
TITLE 'DSQDCTSD - QMF SDSCI ENTRIES'  
* TRACE DATA SET  
  DFHDCT TYPE=SDSCI,DSCNAME=DSQDEBUG,  
    RECFORM=VARBLK,  
    RECSIZE=121,  
    BLKSIZE=6050,  
    TYPEFILE=OUTPUT  
  
*  
  TITLE 'DSQDCT - CICS DESTINATION CONTROL TABLE'  
  
*  
* TRACE DATA SET  
*  
DSQD  DFHDCT TYPE=EXTRA,DESTID=DSQD,DSCNAME=DSQDEBUG,RSL=1
```

図 163. トレース・データ・セットの (CICS 表の) 記述

単一の CICS 領域内のすべての QMF ユーザーからの QMF トレース・データが、単一のトレース・データ・セットに書き込まれます。各トレース項目には、それを記録したユーザーの端末 ID が入れられます。

CICS 領域がアクティブの間にトレース・データ・セットを調べるには、CICS キュー ID の DSQD を用いてトレース・データ・セットをクローズしなければなりません。これは、CICS 提供トランザクション CEMT を使用して実行することができます。トレース・データ・セットがクローズされると、TSO の ISPF からこのデータ・セットを印刷またはブラウズすることができます。トレース・データ・セットがクローズされているときは、CICS ユーザーはほかのレコードの書き込みを行うことはできません。QMF は、トレース・レコードを記録せずにこの状態で作動し続けます。QMF のトレースを使用可能に戻すには、CICS 提供のトランザクション CEMT を使用して、CICS キュー ID の DSQD を使用しているトレース・データ・セットをオープンします。

付録B. DB2 に常駐している QMF オブジェクト

以下の表は、DBA (データベース管理者) に、データベースに常駐する QMF V7 オブジェクトを示すものです。これらの表は、DB2 サブシステムで QMF V7 を実行するのに必要なすべてのデータベース・オブジェクトを要約するためのものです。これらの表は、本書で概要を示したインストール・ジョブを置き換えるためのものではなく、データベース・オブジェクトを回復する必要がある場合の単なる指針として示されているものです。

考慮事項: この節の情報は、Workstation Database Server に保管されるものではなく、DB2 (OS/390 版) Database Server に保管されるものを示します。

汎用 QMF データベース・オブジェクト

表67 は、汎用 QMF データベース・オブジェクトを示したものです。

表 67. QMF データベース・オブジェクト

オブジェクト名	オブジェクトのタイプ	DDL メンバー	コメント
DSQSGCTL	STOGROUP	DSQ1VSTB	データベース DSQDBCTL は、この STOGROUP に常駐する。
DSQDBCTL	DATABASE	DSQ1TBLB	すべての QMF 制御表は、このデータベースに常駐する。
DSQSGDEF	STOGROUP	DSQ1STGC	IVP で使用され、さらにデフォルトの SAVE DATA データベース / 表スペース・パラメーターの STOGROUP としても使用される。(デフォルトの Q.PROFILES SPACE パラメーター。)
DSQDBDEF	DATABASE	DSQ1STGC	IVP で使用され、またデフォルトの SAVE DATA データベース・パラメーターの DATABASE としても使用される。(デフォルトの Q.PROFILES SPACE パラメーター。)

DB2 に常駐している QMF オブジェクト

表 67. QMF データベース・オブジェクト (続き)

オブジェクト名	オブジェクトの タイプ	DDL メンバー	コメント
DSQTSDEF	TABLE SPACE	DSQ1STGC	IVP で使用され、またデフォルトの SAVE DATA 表スペース・パラメーター の TABLE SPACE としても使用される。 (デフォルトの Q.PROFILES SPACE パラ メーター。)

VSAM クラスタ

表 68 に、QMF 出荷時の VSAM クラスタを示します。

表 68. VSAM クラスタ

クラスタ名	クラスタが必要となるオブジェクト
QMFDSN.DSNDBC.DSQDBCTL.DSQTSTCT1.I0001.A001	DSQTSTCT1
QMFDSN.DSNDBC.DSQDBCTL.DSQTSTCT2.I0001.A001	DSQTSTCT2
QMFDSN.DSNDBC.DSQDBCTL.DSQTSTCT3.I0001.A001	DSQTSTCT3
QMFDSN.DSNDBC.DSQDBCTL.DSQTSPRO.I0001.A001	DSQTSPRO
QMFDSN.DSNDBC.DSQDBCTL.DSQTSLLOG.I0001.A001	DSQTSLLOG
QMFDSN.DSNDBC.DSQDBCTL.DSQTSGOV.I0001.A001	DSQTSGOV
QMFDSN.DSNDBC.DSQDBCTL.DSQTSSYN.I0001.A001	DSQTSSYN
QMFDSN.DSNDBC.DSQDBCTL.OBJECTRD.I0001.A001	Q.OBJECT_DIRECTORYX
QMFDSN.DSNDBC.DSQDBCTL.OBJECTRR.I0001.A001	Q.OBJECT_REMARKSX
QMFDSN.DSNDBC.DSQDBCTL.OBJECTRO.I0001.A001	Q.OBJECT_OBJDATA
QMFDSN.DSNDBC.DSQDBCTL.PROFILEX.I0001.A001	Q.PROFILEX
QMFDSN.DSNDBC.DSQDBCTL.COMMANDR.I0001.A001	Q.COMMAND_SYNONYMSX

QMF 制御表

表 69 に、QMF 出荷時の制御表を示します。

表 69. QMF の制御表

表	索引	視点	表スペース	DDL メンバー
Q.OBJECT_DIRECTORY	Q.OBJECT_DIRECTORYX	なし	DSQTSTCT1	DSQ1TBLI
Q.OBJECT_REMARKS	Q.OBJECT_REMARKSX	なし	DSQTSTCT2	DSQ1TBLI
Q.OBJECT_DATA	Q.OBJECT_OBJDATA	なし	DSQTSTCT3	DSQ1TBLI
Q.PROFILES	Q.PROFILEX	なし	DSQTSPRO	DSQ1TBLU
Q.ERROR_LOG	なし	なし	DSQTSLLOG	DSQ1TBLE
Q.COMMAND_SYNONYMS	Q.COMMAND_SYNONYMSX	なし	DSQTSSYN	DSQ1TBLN

表 69. QMF の制御表 (続き)

表	索引	視点	表スペース	DDL メンバー
Q.RESOURCE_TABLE	Q.RESOURCE_INDEX	Q.RESOURCE_VIEW	DSQTSGOV	DSQ1TBLG
Q.DSQ_RESERVED	なし	なし	DSQTSRDO	DSQ1TBLH DSQ1TBDB

デフォルトのリスト視点

デフォルトのリスト視点をカスタマイズしている可能性があります。表70 に、QMF 出荷時のデフォルトの視点を示します。

表 70. デフォルトのリスト視点

視点名	DDL メンバー
Q.DSQEC_TABS_LDB2	DSQ1BVDB
Q.DSQEC_COLS_LDB2	DSQ1BVDB
Q.DSQEC_QMFOBJS	DSQ1BVDB
Q.DSQEC_TABS_RDB2	DSQ1BVDB
Q.DSQEC_COLS_RDB2	DSQ1BVDB
Q.DSQEC_ALIASES	DSQ1BVDB

QMF プラン

表71 に、QMF 出荷時のプランを示します。

表 71. QMF プラン

プラン名	バインド・ジョブ	コメント
QMF710	DSQ1BINR	汎用 QMF プラン
DSQIN610	DSQ1BSQL	インストール・ジョブに限り使用される QMF プラン

QMF パッケージ

表72 に、QMF 出荷時のパッケージを示します。

表 72. QMF パッケージ

パッケージ名	バインド・ジョブ
DSQB*	DSQ1BINJ

注: 特定の DDL メンバーを実行するジョブの場合は、索引を参照してください。

DB2 に常駐している QMF オブジェクト

NLF パーツ

NLF を使用している場合の NLF のパーツでは、すべての n シンボルをインストールした NLF に一致する 1 文字の各国語 ID (NLID) で置換します。

表73 に、QMF 出荷時の NLF オブジェクトを示します。

表 73. QMF NLF オブジェクト

オブジェクト名	オブジェクトのタイプ	DDL メンバー	コメント
DSQSGSY n	STOGROUP	DSQ1 n SYS	これは Q.COMMAND_SYNONYMX_ n 索引の STOGROUP です。

表74 に、QMF 出荷時の NLF 表を示します。

表 74. QMF NLF 表

表	索引	SDSQSAP n の DDL メンバー
Q.OBJECT_SYNONYM_ n	Q.COMMAND_SYNONYMX_ n	DSQ1 n SYC

QMF サンプル表

表75 に、サンプル表を示します。

表 75. サンプル表

表	含まれている情報
Q.ORG	会社組織
Q.STAFF	会社従業員
Q.APPLICANT	新規採用候補者
Q.PRODUCTS	会社の製品
Q.SALES	販売および歩合
Q.PROJECT	請け負ったプロジェクト (部門別)
Q.INTERVIEW	新規採用の面接
Q.SUPPLIER	取引先情報
Q.PARTS	製品の部品データ

付録C. フォールバック

フォールバックとは、ユーザーを QMF の前のリリースへ戻す (移行する) 過程のことです。終結処置 は、OS/390 から前のリリースを削除する過程のことです。終結処置については 109ページの『ステップ 36 -- インストール後の終結処置』で説明しているのです、ここでは説明しません。

フォールバックは、QMF の 2 つのバージョンを同じ DB2 サブシステムから実行している場合を除いて、必要ありません。

前のプロファイルの再確立

QMF バージョン 2.4 にはない ENVIRONMENT 列は、バージョン 7 のプロファイルには影響しません。バージョン 2.2 以降に追加されたすべての列についても、同じことが言えます。

注: ログオン ID が 1 次許可 ID とは異なっていて、CREATOR の値が 1 次許可 ID を使用するように更新されていた場合には、フォールバックの一環として、それらをログオン ID に復元しなければなりません。

前のリリースでの QMF バージョン 7 のオブジェクトの使用

これは大幅に制限されています。フォールバックの余地はありますが、この付録でこれまでに説明してきた互換性の規則をユーザーによく理解してもらってください。

QMF の前のリリースにフォールバックする場合、QMF バージョン 7 で作成された一部のオブジェクトは前の環境では使用することができません。フォールバックの可能性を考える場合には、この点を考慮してください。以下に、一部のバージョン 7 のオブジェクトを前のリリースで使用する場合に適用される制約をリストします。

- 書式

バージョン 7 から保管またはエクスポートされ、QMF の前のリリースで表示またはインポートされる書式オブジェクトは、正常に実行されると考えることができます。ただし、バージョン 7 から保管またはエクスポートされた書式オブジェクトは、バージョン 2.4 またはそれより前のバージョンでは使用できません。

制御の切れ目フィールド番号 (またはヘッダー・レコード内のオブジェクト・レベル) を使用している、バージョン 7 からエクスポートされた書式は、前のアプリケーションで使用できるようにするには、書式アプリケーション移行援助プログラムを必要とします。

- 照会

前のリリースへのフォールバックについて、バージョン 7 の照会に、いくつかの制約事項が適用されます。

- SQL 照会: バージョン 7 から SQL 照会をエクスポートして前のリリースにインポートすることができ、その結果それらは正常に実行されます。ただし、バージョン 7 で保管された SQL 照会は、バージョン 2.4 またはそれより前のバージョンでは使用できません。
- 指示照会: バージョン 7 の指示照会を前のリリースで表示およびインポートすることはできます。ただし、それらの照会に変数、または以前の 55 文字または 65 文字の制限を超える式が含まれていない場合です。
- QBE 照会: QBE (例示照会) により作成され、バージョン 7 で保管またはエクスポートされた照会は、前のリリースで表示またはインポートすることができ、正常に実行されます。

- プロシージャ

バージョン 7 からエクスポートされたプロシージャ・オブジェクトは、前のリリースにインポートでき、新規の QMF コマンドまたはコマンド構文が使用されていないければ、実行することができます。バージョン 7 で保管されたプロシージャ・オブジェクトは、まず最初にバージョン 7 からエクスポートした後でそれらを前のリリースにインポートするといったことをしない限り、前のリリースでは表示することはできません。ロジックを持つプロシージャつまり REXX ロジックが入っているプロシージャは、バージョン 3 より前のリリースに表示またはインポートすることはできません。

- 前のリリースでは実行できない QMF コマンドが入っているプロシージャまたはアプリケーション

これらのコマンドは、さまざまな理由で実行に失敗することがあります。詳細については、575ページの『前のリリースでの バージョン 7 QMF コマンドの使用』を参照してください。

- 呼び出し可能インターフェースを呼び出すアプリケーション

アプリケーションは、QMF を呼び出すために CLIST およびプログラム内で呼び出し可能インターフェースを呼び出します。呼び出し可能インターフェースはバージョン 2.4 で採用されたため、QMF のそれより前のバージョンで実行されるアプリケーションは使用できません。

QMF の前のリリースと QMF バージョン 7 との相違点に関する詳細な情報を必要とする場合があります。その場合は、*QMF 解説書* の 2 つのリリースを比較してください。

前のリリースでの バージョン 7 QMF コマンドの使用

バージョン 7 のプロシージャおよびアプリケーションは、前のリリースでは実行できないコマンドを含んでいるために、QMF の前のリリースでは正しく実行されないことがあります。一部のコマンドでは、次のような場合があります。

- 前のリリースには存在しない。
- 前のリリースでは異なる作動をするオプションが含まれている。たとえば、DRAW コマンドは前と同じ構文ですが、現在では異なった結果を出します。すべてのキーワードに二重引用符が付いているため、ユーザーが引用符を追加する必要がなく、二重引用符を提供するために使用していたツールは必要なくなりました。

31 桁 10 進数のサポート

QMF バージョン 3.1 (またはそれ以降) を使用していて、さらに DB2 バージョン 2.3 (またはそれ以降) で操作している場合には、31 桁 10 進数のサポートがあります。一部の式では、15 桁から 31 桁の 10 進形式に移行する場合に、異常な結果または受け入れ不能な結果が生成されることがあります。

付録D. QMF OS/390 リリース間の移行

バージョン 7 をインストールする前にインストール先で QMF の前のリリースを使用していた場合は、ユーザーが QMF の前のリリースをそのまま使用していることがあります。その場合には、それらのユーザーが新しいリリースを操作できるように支援する必要があります。その際には以下のステップが必要になります。

- ユーザーに QMF バージョン 7 アプリケーション・プランへのアクセス権を付与する。
- ユーザーに適切な QMF プロファイルを提供する。
- 前に作成されたオブジェクト (たとえば、照会および書式) を、新規のリリースのもとでの QMF セッションで使用できるようにする。

移行とは何か

移行とは、上記の節で説明したステップを実行する過程のことです。移行は、以下の QMF のリリースから移る場合には基本的に同じです。

- バージョン 2 リリース 2
- バージョン 2 リリース 3
- バージョン 2 リリース 4
- バージョン 3 リリース 1
- バージョン 3 リリース 1 モディフィケーション・レベル 1
- バージョン 3 リリース 2

この付録では、QMF バージョン 7 が、本書に記載されている指示に従ってインストールされているものと想定しています。それ以外の場合、あるいは設定の一部が変更されている場合には、説明の一部が適用できないことがあります。

移行の説明では用語を短縮して使用することがあります。たとえば、‘V7R1’ は ‘バージョン 7’ を意味します。同様に、V2R2、V2R3、V2R4、V3R1 はそれぞれ前のリリースを表します。それらを適切な QMF リリースに結び付けるために、たとえば次に示す例のように、これらの語を他の語の前に付けることがあります。

- V3R1 ユーザー。QMF バージョン 3 リリース 1 を使用している人を指す。
- V2R4 書式。QMF バージョン 2 リリース 4 で作成された書式を指す。

QMF OS/390 リリース間の移行

- V2R2 データベース。QMF バージョン 2 リリース 2 がインストールされている DB2 データベースを指す。

フォールバックとは何か

フォールバック は、ユーザーを QMF バージョン 7 からそれより前のリリースに移行する過程です。フォールバックのみに関心がある場合は、592ページの『フォールバック』に進んでください。 終結処置については 109ページの『ステップ 36 -- インストール後の終結処置』で説明しているので、ここでは説明しません。

バージョン 3 (およびそれ以降) の書式と QMF の前のバージョンの書式では、内部表現が異なっています。前のいずれかの書式をバージョン 3 (またはそれ以降) の書式に変換すると、QMF の前のバージョンでは使用できなくなります。

QMF バージョン 2.4 (またはそれより前) の書式を処理するために開発されたアプリケーションも、ヘッダー・レコード内のオブジェクト・レベルまたは制御の切れ目フィールド番号を参照する場合には、作動しないことがあります。

バージョン 2 またはそれより前の書式からバージョン 7 の書式へユーザーが間違っただけで変換しないように (QMF バージョン 7 のもとで実行中に古い書式を置き換えることによってユーザーが行う誤り)、あるバージョンが今後は必要でないということが明らかになるまで、前のバージョンとバージョン 7 の両方を異なる名前でも保管することができます。たとえば、次のコマンドは作業域の書式をバージョン 7 の書式に変換します。

```
SAVE FORM AS FORM1  
EXPORT FORM TO FORM2
```

インストール先システムが QMF の前のリリースにフォールバックする場合には、QMF のバージョン 7 の使用を取りやめる前に、前のリリースのもとでバージョン 7 のオブジェクトを急いで再作成するようユーザーに指示してください。

注: REXX は CICS ではサポートされていません。

QMF バージョン 7 アプリケーション・プランおよびパッケージへのアクセス権の付与

この手順は、QMF の前のリリースすべてについて同一です。認可が引き続き有効な場合には、この節はとばしてください。

QMF インストール時に、QMF アプリケーション・プラン (QMF710) へのアクセスが PUBLIC に対してまたは移行中のユーザーに対して認可されなかった場合には、次の照会を実行します。(GRANT オプションで、QMF プランおよびパッケージに対する EXECUTE 特権が必要です。)

```
GRANT EXECUTE ON PLAN QMF710 to authid
```

ここで、*authid* は移行中のユーザーの許可 ID です。*authid* に PUBLIC を指定すると、プランおよびパッケージに対する EXECUTE 権限をすべての人に与えることになります。

このサブジェクトの詳細については、255ページの『QMF アプリケーション・プランおよびパッケージに対するアクセスの制御』を参照してください。

DB2 サブシステムと移行

ユーザーを移行させる場合には、QMF の旧バージョンと新バージョンが同じ DB2 サブシステムにある場合と 2 つの異なるサブシステムにある場合があります。

- QMF の 2 つのリリースが同じ DB2 サブシステム上にある場合は、『同じ DB2 サブシステム上にある QMF の移行』をお読みください。
- QMF の 2 つのリリースが同じ DB2 サブシステム上にはない場合は、582ページの『異なる DB2 サブシステム間での QMF の移行』をお読みください。

同じ DB2 サブシステム上にある QMF の移行

QMF の両方のリリースが同じ DB2 サブシステム上にある場合は、この節をお読みください。

QMF プロファイルの提供

QMF セッションの開始時に、ユーザーの QMF プロファイルが Q.PROFILES 表のいくつかの行から取られます。QMF の両方のリリースが同じ DB2 サブシステム上にある場合には、2 つのリリースは同じ Q.PROFILES 表を使用します。

ユーザーの 1 次許可 ID が TSO のログオン ID と異なっている場合には、QMF の開始時に DSQSPRID パラメーターの値が TSOID になっている必要があります。そうでない場合には、Q.PROFILES に、CREATOR を 1 次許可 ID に設定したユーザー行を挿入してください。

QMF バージョン 3.1.1 ユーザーの場合: バージョン 3.1.1 から移行している場合には、Q.PROFILES 表に新しい列は加わっていません。

QMF バージョン 3.1 ユーザーの場合: バージョン 3.1 から移行している場合には、Q.PROFILES 表に新しい列は加わっていません。

QMF バージョン 2.4 ユーザーの場合: Q.PROFILES 表に ENVIRONMENT という列が 1 つ加わっており、ここには新規のリリースだけに適用されるプロファイル・パラメーターが入っています。この列には、表76 に示すように、入力項目に対するデフォルト値が 1 つあります。新規の列は、バージョン 2.4 の実行には何も影響せず、またバージョン 3 のリリースでは表示されません。

重要: QMF バージョン 2.4 のユーザーが APAR PL77029 の修正をインストールした場合を除いて、新規の列は表示されません。

QMF バージョン 2.2 およびバージョン 2.3 ユーザーの場合: Q.PROFILES 表には MODEL と ENVIRONMENT という 2 つの列が加わっており、ここには新規のリリースだけに適用されるプロファイル・パラメーターが入っています。これらの列には、表76 に示すように、入力項目に対するデフォルト値が 1 つあります。2 つの新規の列は QMF バージョン 2.2 およびバージョン 2.3 の実行には何も影響を与えず、これらのリリースでは新規の列は表示されません。

重要: ユーザーが APAR PL77028 の修正をインストールした場合を除いて、新規の列は表示されません。

既存の行に追加された MODEL および ENVIRONMENT 列には、NULLS が入ります。SET または SAVE PROFILE によって、ユーザーは MODEL 列だけを提供することもできます。ENVIRONMENT 列は割り当てる必要があります。

表 76. バージョン 2.3 のプロファイル表に追加された列

列の名前	目的
MODEL	各ユーザーのデータの概念視点 (指示照会がデータにアクセスするために使用する視点) を識別します。デフォルトの値は REL です。この列の詳細については、付録F. QMF 制御表および QMF で使用される表スペースを参照してください。
ENVIRONMENT	QMF セッションの実行環境を識別する。このパラメーターの詳細については、『付録F. QMF 制御表および QMF で使用される表スペース』を参照してください。

Q.OBJECT_DIRECTORY の移行

バージョン 3.2 では、Q.OBJECT_DIRECTORY 表に新規の列が 3 つ加わりました。QMF の古いリリースから移行している場合には、これらの 3 つの列は前の表には何の影響も与えません。表 77 に、3 つの新規の列を示します。

表 77. バージョン 3.2 の Q.OBJECT_DIRECTORY 表に追加された列

列名	データ・タイプ	長さ (バイト単位)	ヌルの使用	機能 / 値
CREATED	TIMESTAMP		可	オブジェクトが作成された時点を表すタイム・スタンプ値を示す。この値は SAVE または IMPORT コマンドの後に記録されます。
MODIFIED	TIMESTAMP		可	オブジェクトが最後に変更された時点を表すタイム・スタンプ値を示す。この値は SAVE または IMPORT コマンドの後に記録されます。
LAST_USED	DATE		可	オブジェクトが最後に使用された時点を表すタイム・スタンプ値を示す。値は 1 日に 1 回だけ更新されます。

バージョン 2.4 からの ISPF での DPRE の移行

QMF バージョン 2.4 から QMF バージョン 7 へ移行するとき、QMF バージョン 2.4 とバージョン 7 の両方を同じデータベース上で実行する場合は、QMF の両方のバージョンに、DPRE の QMF バージョン 2.4 を使用する必要があります。QMF バージョン 7 をインストールした後で QMF バージョン 2.4 を用意するには、次の手順を行ってください。

1. DPRE をバージョン 7 のレベルにアップグレードした時点で使用できるよう、DSQABR13 の QMF バージョン 7 バージョンの名前変更または保管を行います。
2. 以下の CLIST を、QMF バージョン 2.4 のライブラリー QMF240.DSQCSTE から QMF バージョン 7 のライブラリー QMF710.SDSQCLTE に移動します。
 DSQABR11
 DSQABR12
 DSQABR13
3. Q という許可 ID を持つ QMF バージョン 2.4 セッションから、QMF プロシージャ Q.DSQAERIP のバージョン 2.4 を IMPORT (インポート) して保管します。この操作を行うには、次の QMF コマンドを発行します。

QMF OS/390 リリース間の移行

```
IMPORT PROC Q.DSQAER1P FROM 'QMF240.DSQSAMPE(DSQAER1P)'
```

QMF バージョン 2.4 を使用しなくなった場合には、次の手順で、DPRE をバージョン 7 レベルに復元してください。

1. 上記のステップ 1 で保管または名前変更した DSQABR13 の QMF バージョン 7 を、QMF バージョン 7 ライブラリー QMF710.SDSQCLTE に復元します。
2. Q という許可 ID を持つ QMF バージョン 7 セッションから、QMF プロシージャ Q.DSQAER1P のバージョン 7 を IMPORT (インポート) して保管します。この操作を行うには、次の QMF コマンドを発行します。

```
IMPORT PROC Q.DSQAER1P FROM 'QMF710.SDSQSAPE(DSQAER1P)'
```

QMF バージョン 7 のもとで、前のリリースからのオブジェクトを使用可能にする

QMF の両方のリリースが同じ DB2 サブシステム上にある場合には、すべての DB2 オブジェクト (たとえば、表および視点) は、前のリリースで使用できれば、QMF バージョン 7 で使用することができます。照会、書式、およびプロシージャもすべて使用できますが、QMF バージョン 7 のもとでは一部は使用できないことがあります。これについては 585 ページの『QMF オブジェクトの移行』で説明します。

異なる DB2 サブシステム間での QMF の移行

この節では、QMF の 2 つのリリースが異なる DB2 サブシステム上にある場合の移行方法について説明しています。

DB2 サブシステムが異なる場合、QMF の前のリリースのデータベースにある QMF オブジェクトはバージョン 7 ユーザーは使用できないために、移行は複雑になります。また、バージョン 7 のデータベース内のオブジェクトも、QMF の前のリリースのユーザーは使用できません。

QMF に必要な表および視点は、新しいサブシステムで使用できるようにする必要があります。

QMF プロファイルの提供

QMF バージョン 7 が異なる DB2 サブシステム内にあるときは、インストール・プロセスは新規の Q.PROFILES 表を作成します。

QMF バージョン 3.1.1 ユーザーの場合: バージョン 3.1.1 から移行している場合には、Q.PROFILES 表に新しい列は加わっていません。

QMF バージョン 3.1 ユーザーの場合: バージョン 3.1 から移行している場合には、Q.PROFILES 表に新しい列は加わっていません。

QMF バージョン 2.4 ユーザーの場合: 新規の Q.PROFILES 表には、追加の列として ENVIRONMENT が加わっています。

QMF バージョン 2.2 およびバージョン 2.3 ユーザーの場合: 新規の Q.PROFILES 表には、追加の列として MODEL と ENVIRONMENT の 2 つが加わっています。

QMF バージョン 2.4 、バージョン 2.3 、またはバージョン 2.2 ユーザーの場合: 新規に作成される表には、単一の SYSTEM 行が含まれます。列に割り当てられている値を表78 に示します。これらの列およびその意味の詳細については、246ページの表31 を参照してください。

表 78. インストール先システム提供の SYSTEM 行の値

列	値
CREATOR	SYSTEM
CASE	UPPER
DECOPT	PERIOD
CONFIRM	YES
WIDTH	132
LENGTH	60
LANGUAGE	SQL
SPACE	DSQDBDEF.DSQTSDEF
TRACE	NONE
PRINTER	ブランク
TRANSLATION	ENGLISH
PFKEYS	ゼロ長のストリング
SYNONYMS	Q.COMMAND_SYNONYMS
RESOURCE_GROUP	SYSTEM
MODEL	REL
ENVIRONMENT	NULL (ヌル)

CICS がインストールされている場合には、追加の SYSTEM 行があり、ここでは SYNONYMS がヌルに、ENVIRONMENT が CICS に設定されています。

表に SYSTEM 行だけがある場合、ユーザーは、この行で提供される QMF プロファイルを用いてバージョン 7 のセッションを開始します。このプロファイルは、前の QMF リリースにあるプロファイルとは異なることがあります。一連の INSERT 照会を使用すると前のプロファイルを再作成することができませんが、SET または SAVE PROFILE を用いてユーザー各自で再作成を行うこともできます。

ただしユーザーは、SET または SAVE PROFILE を使用して PFKEYS、SYNONYMS、RESOURCE_GROUP の各パラメーターの値を変更することはできません。これらを変更するには、Q.PROFILES 表で UPDATE 照会を行う必要があります。その例については、374ページの『新規機能キー定義のアクティブ化』を参照してください。

PFKEYS、SYNONYMS、RESOURCE_GROUP の各パラメーターは、QMF 環境のカスタマイズで中心的な役割を果たします。それぞれの簡単な説明については、580ページの表76 を参照してください。

Q.OBJECT_DIRECTORY の移行

バージョン 3.2 では、Q.OBJECT_DIRECTORY 表に新規の列が 3 つ加わりました。QMF の古いリリースから移行している場合には、これらの 3 つの列は前の表には何の影響も与えません。表79 に、3 つの新規の列を示します。

表 79. バージョン 3.2 の Q.OBJECT_DIRECTORY 表に追加された列

列名	データ・タイプ	長さ (バイト単位)	ヌルの使用	機能 / 値
CREATED	TIMESTAMP		可	オブジェクトが作成された時点を表すタイム・スタンプ値を示す。この値は SAVE または IMPORT コマンドの後に記録されます。
MODIFIED	TIMESTAMP		可	オブジェクトが最後に変更された時点を表すタイム・スタンプ値を示す。この値は SAVE または IMPORT コマンドの後に記録されます。
LAST_USED	DATE		可	オブジェクトが最後に使用された時点を表すタイム・スタンプ値を示す。値は 1 日に 1 回だけ更新されます。

QMF バージョン 7 のもとで、前のリリースからのオブジェクトを使用可能にする

DB2 表および QMF オブジェクトは、QMF の前のリリースのもとでサブシステムからエクスポートし、その後、QMF バージョン 7 のもとでインポートすることができます。

DB2 表を移行するには、適切な DB2 権限を持つユーザーが以下のことを行ってください。

1. DB2 提供のアプリケーション・プログラム DSNTIAUL を使用して、表をアンロードします。このプログラムの詳細については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。
2. アンロードした表を、DB2 ローダーを使用してバージョン 7 の DB2 サブシステムにロードします。ローダーの使用法の詳細については、*DB2 UDB (OS/390 版) 管理の手引き* を参照してください。

DXT がインストールされている場合には、表のアンロードおよびロードに DXT を使用することもできます。DXT は、IBM ライセンス・プログラム製品です。DXT の詳細については、*Data Extract : General Information* を参照してください。

QMF の 2 つのバージョンが異なる OS/390 システム上にある場合には、使用可能なネットワーク機能を用いて、エクスポートしたオブジェクトおよびアンロードした表を QMF バージョン 7 が入っているシステムに送信してください。

QMF 照会、書式、プロシージャ、およびアプリケーションを移行する場合には、必ず次の節（『QMF オブジェクトの移行』）を読んでください。

視点および同義語

QMF を使用してデータベースから表をエクスポートし、別のデータベースにその表をインポートする場合には、表に視点、索引、同義語、および権限があるのであれば、新規のデータベースでそれらを作成する必要があります。

QMF オブジェクトの移行

この節では、QMF オブジェクトを移行する場合の考慮事項を説明します。QMF の前のリリースで作成されたオブジェクトの大部分は、QMF バージョン 7 でも使用することができます。（QMF の前のリリースに戻る移行については、592ページの『フォールバック』を参照してください。）

照会および書式

QMF の前のリリースで作成されたすべての照会および書式は、QMF バージョン 7 でも使用することができます。

プロシージャ

バージョン 2.4 で保管またはエクスポートされたプロシージャ・オブジェクトは、バージョン 7 で表示またはインポートすることができます。また、コマンドとオプションの省略形が使用されている場合には、それらがバージョン 7 でも有効であれば、そのプロシージャ・オブジェクトを実行することもできます。ISPF を必要とするコマンドまたはアプリケーションが入っているバージョン 2.4 のプロシージャは、QMF バージョン 7 が ISPF ダイアログとして開始される場合にだけ、実行されます。英語で書かれ、QMF バージョン 2.4 で保管またはエクスポートされたプロシージャは、コマンド言語グローバル変数が英語のコマンドを受け入れるように設定されていれば、変更なしでバージョン 7 NLF セッション (英語が統轄言語ではない QMF セッション) でインポートおよび実行することができます。

前のリリースからのプロシージャの一部は、インストール先定義コマンドでも動詞として使用される動詞を含むコマンドを出している場合には、正しく機能しません。QMF バージョン 7 でこのような状態が生じないようにするため、ユーザーはすべてのコマンドの前に QMF を付けることができます。これにより、これらのコマンドはインストール先定義コマンドではなく、標準の QMF コマンドとして認識されます。その結果、これらのプロシージャは QMF バージョン 7 (または QMF バージョン 2 またはバージョン 3 の任意のリリース) で実行することができます。インストール先定義コマンドの詳細については、343ページの『第18章 QMF コマンドのカスタマイズ』を参照してください。

アプリケーションの移行

ISPF を必要とするコマンドが含まれているバージョン 2.4 のアプリケーションは、バージョン 7 が ISPF ダイアログとして開始される場合だけ実行されます。英語で書かれ、バージョン 2.4 で実行されるコマンドを出すアプリケーションは、コマンド言語グローバル変数が英語のコマンドを受け入れるように設定されていれば、変更なしでバージョン 7 NLF セッション (英語が統轄言語ではない QMF セッション) で実行することができます。

呼び出し可能インターフェースに関する考慮事項

QMF バージョン 7 より前にリンク・エディットされており、かつ呼び出し可能インターフェースを使用する QMF アプリケーションで LIBDEF 機能を使

用したい場合には、QMF バージョン 7 インターフェース・モジュールを使用してユーザーのアプリケーションを再度リンク・エディットする必要があります。

書式アプリケーション移行援助プログラム

QMF の前のバージョン用に作成されたアプリケーションが制御の切れ目フィールド ID を参照している場合には、それらのアプリケーションを QMF バージョン 7 で使用するには書式アプリケーション移行援助プログラムを使用しなければならないことがあります。

書式アプリケーション移行援助プログラムを使用すると、バージョン 2.4 の制御の切れ目番号を含んだアプリケーションを QMF バージョン 7 で使用できるようになります。ただし、書式アプリケーション移行援助プログラムを使用しても、QMF バージョン 3 の FORM (書式) をエクスポートしてそれらを QMF バージョン 2.4 またはそれより前のリリースで使用することはできません。

QMF バージョン 7 と共に出荷されるこの援助プログラムは、バージョン 7 の書式の制御の切れ目番号を前のバージョンで使用されていた番号に変更します。制御の切れ目フィールド ID を変更するだけでアプリケーションは正しく作動しなくなるため、ヘッダー・レコード内のオブジェクト・レベル・フィールドは 3 から 4 に変更されています。ヘッダー・レコードの詳細については *QMF アプリケーション開発の手引き* を参照してください。

ユーザーまたはアプリケーションが EXPORT FORM コマンドを出すと書式アプリケーション移行援助プログラムが実行されます。このコマンドは、援助プログラムに関するコマンド同義語を自動的に開始します。これには REXX EXEC が必要であるため、REXX 解釈プログラムが使用可能でなければならず、また QMF が ISPF ダイアログ機能として作動していなければなりません。

アプリケーション開発担当者に、以下のことを推奨してください。

1. 古い番号付け構造を持つ制御の切れ目フィールドを参照しているアプリケーションを、QMF バージョン 3 で使用されている新しいフィールド番号に変更する。
2. すべてのアプリケーションの変更を完了したら、移行援助プログラムのコマンド同義語をコマンド同義語表から除去する。

QMF OS/390 リリース間の移行

移行援助プログラムを設定するには、QMF コマンド同義語表 Q.COMMAND_SYNONYMS および移行援助プログラムが必要なアプリケーションを使用するその他の同義語表に対して、次の SQL INSERT 照会を実行してください。

```
INSERT INTO Q.COMMAND_SYNONYMS (VERB, OBJECT, SYNONYM_DEFINITION, REMARKS)
VALUES ('EXPORT', 'FORM', 'TSO DSQAEF0A', 'Version 7 Form Migration Aid')
```

アプリケーションを新しい外部化書式に適合するよう調整できたらすぐに、この項目はコマンド同義語表から削除してください。

この項目は、次のような照会を用いて削除します。

```
DELETE FROM Q.COMMAND_SYNONYMS
WHERE VERB='EXPORT'
AND OBJECT='FORM'
AND SYNONYM_DEFINITION='TSO DSQAEF0A'
```

上記の INSERT 照会および DELETE 照会では、EXPORT、FORM、DSQAEF0A の各キーワードには NLS 変換が必要なので、この照会は適宜変換する必要があります。

OS/390 の ISPF のもとでの QMF の実行

QMF バージョン 7 を使用して、リソースを QMF セッションに割り振る方法およびその開始方法を検討します。ISPF LIBDEF を用いて同じ ISPF セッションから QMF バージョン 7 および QMF の前のバージョンを実行することができます。ただし、複数の QMF セッションを同時に実行したり、複数の QMF ライブラリー・セットを同時に割り振ったりすることはできません。

複数の DB2 ライブラリーを DSQLLIB に割り振ることにより、ISPF セッションを終了せずに種々のバージョンのデータベースを使用することもできます。この場合、ISPF LIBDEF を使用して、DSQLLIB に割り振られたライブラリーを ISPF で使用できるようにします。

ISPF のもとで実行中の QMF のこのバージョンを使用して、QMF はまず、DSQLLIB からのプログラムを探します。プログラムが DSQLLIB 内で見つからないかまたは DSQLLIB が割り振られていない場合は、QMF は、前のリリースで保持しているプログラムを探します。

移行に関するその他の考慮事項

この節では、QMF で使用されている環境に関する特別な考慮事項を含めて、QMF での移行に関するその他の考慮事項を説明します。

31 桁 10 進数のサポート

QMF バージョン 3.1 (またはそれ以降) を使用していて、さらに DB2 バージョン 2.3 (またはそれ以降) で操作している場合には、31 桁の 10 進数のサポートがあります。QMF の前のバージョンまたはデータベースの前のバージョンから移行する場合には、どの表が 31 桁 10 進数サポートの影響を受けているかを判別する必要がある場合があります。次に示す照会は、DB2 表の影響を受ける可能性のあるユーザー表のリストを検索します。

```
SELECT DISTINCT(TBNAME)
FROM SYSIBM.SYSCOLUMNS
WHERE COLTYPE IN ('INTEGER', 'SMALLINT', 'FLOAT', 'DECIMAL', 'DATE', 'TIME',
                  'TIMESTAMP')
ORDER BY TBNAME
```

管理プログラム

ユーザーがバージョン 2.4 の IBM 提供管理プログラムを QMF バージョン 3 で使用したいときに、バージョン 2.4 の管理プログラムが XCBPANEL を DXYEMU00 か DXYEMU01 に設定しているという場合には、ユーザーは次のいずれか 1 つを行う必要があります。

- ユーザーがバージョン 2.4 の管理プログラムを使用して、ISPF なしに QMF バージョン 3 製品を実行したい場合には、ユーザーは XCBPANEL を設定しているステートメントを管理プログラム出口から除去する必要があります。(QMF バージョン 3 には、デフォルトの設定として ISPF パネルの代わりにウィンドウ・ヘルプ・パネルが用意されています。)
- ユーザーが QMF バージョン 2.4 と共に出荷される IBM 提供の管理プログラムを変更なしに使用したい場合には、ユーザーは QMF バージョン 3 と共に出荷される QMF 管理プログラムを使用して、ISPF パネル DXYEMU00 および DXYEMU01 を QMF バージョン 3 の ISPF パネル・ライブラリーに加える必要があります。

CICS での管理プログラム

QMF バージョン 3.1 管理プログラムの変更バージョンまたは置換バージョンを使用している場合には、インターフェースを変更する必要があります。バージョン 3.1.1 およびそれ以降の管理プログラム・インターフェースでは、標準 CICS パラメーターの渡し方が、最初および 2 番目のパラメーターを使用する方法から DFHCOMMA 区域に入れて管理プログラムに渡す方法に変更されました。

IBM 提供の管理プログラムの使用を予定している場合には、その管理プログラムを新規のバージョンで置き換えてください。

QMF OS/390 リリース間の移行

IBM 提供の管理プログラムを変更した場合、または書き換えた場合には、DXEGOVA および DXEXCBA 制御情報のアドレスにアクセスする方法を、少し変更する必要があります。(詳細については、458ページの『リソース制御情報を管理プログラム出口に渡す』を参照してください。) その場合も、管理プログラムは以前と同じように機能し、内容も変化しません。

CICS でのユーザー編集ルーチン

QMF バージョン 3.1 APAR PN07713 に文書化されているユーザー編集ルーチンは、基本 QMF バージョン 3.1.1 およびそれ以降の製品の一部分です。PN07713 の指示に従ってユーザー編集出口モジュール DSQUECIC を作成した場合には、そのモジュールは QMF バージョン 3.1.1 またはそれ以降でも変更なしで使用することができます。

TSO およびネイティブ OS/390 バッチ でのユーザー編集ルーチン

QMF バージョン 7 では、ユーザー編集コードを再リンクする必要があります。ユーザー編集コードの再リンクの詳細については、377ページの『第20章 QMF 書式のためのユーザー独自の編集コードの作成』を参照してください。

CICS での呼び出し可能インターフェース

QMF バージョン 3.1 またはそれ以降から移行している場合には、QMF 提供の機能呼び出しと主となる QMF プログラムとの間のインターフェースが、CALL インターフェースから EXEC CICS LINK インターフェースに変更されているので注意してください。新規のインターフェースでは、ユーザー・プログラムと QMF 製品とがより分離して取り扱えるようになっています。インターフェースが変更されたため、呼び出し可能インターフェースを使用しているプログラムは再リンク・エディットする必要があります。

CICS での印刷

QMF バージョン 3.1 では、PRINT コマンドにプリンターが指定されていない場合には、出力は CICS 補助一時記憶域に入れられて、前の報告書を置き換えます。バージョン 3.2 およびそれ以降では、報告書は置き換えられません。CICS 補助一時記憶域がすでに存在している場合には、QMF は報告書を既存の一時記憶域キューに追加します。

OS/390 上の CICS でのエクスポート / インポート・サポート

QMF バージョン 3.1.1 の CICS/MVS 環境でのエクスポート / インポート・サポートは、TSO ファイル・サポートを使用しています。このレベルのサポートは、CICS 環境での実行時には推奨されません。実際、あるエラー条件によっては CICS 領域全体が異常終了することがあります。なお、QMF バージョ

ン 3.2 およびそれ以降では、この問題は訂正されています。すなわち、TSO ファイル・システム・サポートが、CICS 一時データのサポートで置き換えられました。

CICS で QMF を実行するときに、QMF EXPORT または IMPORT コマンドを使用する予定があり、さらに CICS 記憶域保護 (SIT STGPROT=YES) を使用中である場合には、QMF モジュール DSQCBST の実行キーを CICS (EXECkey=CICS) に設定する必要があります。こうしておく、IGG019II 状態で異常終了 (ABENDASRA または ABENDOC4) するのを避けることができます。

移行の考慮事項およびサポート

QMF には、推奨されている CICS 一時記憶域または一時データ・キューを使用するか、一時的に TSO データ・セットを使用するかを選択することができますようにする移行機能が用意されています。QMF バージョン 7 がインストールされている場合には、CICS 一時記憶域および一時データ・キューのデフォルト使用はアクティブです。TSO データ・セットを使用しない場合には、移行に関する考慮事項はありません。

TSO データ・セットを使用したい場合には、QMF エクスポート / インポート制御モジュール DSQCTLXI を使用不可にする必要があります。それには、CICS 提供の CEMT トランザクションを使用してください。たとえば、次のように指定します。

```
CEMT SET PROGRAM(DSQCTLXI) DISABLE
```

DSQCTLXI は、CICS CSD または PCT 表からそれ自身を除去することによっても使用不可にすることができます。DSQCTLXI を使用不可にしておけば、CICS で実行中のすべての QMF セッションが、エクスポートおよびインポート・コマンドに TSO データ・セット・サポートを使用するようになります。

CICS 一時記憶域または一時データ・キューのサポートを使用不可にした後で、CEMT を使用することによって、または CICS CSD または PCT 表からプログラム項目が除去されている場合はそれを追加することによって、そのサポートを再びアクティブ化することができます。CEMT を使用するには、次のコマンドを入力してください。

```
CEMT SET PROGRAM(DSQCTLXI) ENABLE
```

バージョン 2 からの移行

QMF バージョン 3 には、新しい DD ステートメントおよびデータ・セット DSQPNLE が含まれており、ISPTLIB はもう使用されません。

QMF OS/390 リリース間の移行

メッセージ・ツールは製品と共には出荷されなくなりました。メッセージ・ヘルプ・パネルを表示するには、QMF コマンド行で次のコマンドを発行してください。

```
HELP msgno
```

メッセージの詳細については、*QMF* メッセージおよびコード を参照してください。

機能キーをカスタマイズするための *QMF* デモ用アプリケーションは、*QMF* と共には出荷されなくなりました。

QMF ISPF パネル・ライブラリー・データ・セット SDSQPLBE (旧名 DSQPLIBE) のメンバー・サイズは、*QMF* が ISPF パネルから GDDM マップ・パネルに変換されたために、約 2500 メンバーから約 70 メンバーに縮小されました。

QMF バージョン 2 で使用可能な以下の視点は、バージョン 3 には存在しません。

- Q.AUTH_LIST
- Q.COLUMN_LIST
- Q.TABLE_LIST
- Q.QUERY_LIST
- Q.PROC_LIST
- Q.FORM_LIST
- Q.AMFTABLE_LIST

バージョン 3 でこれらのオブジェクト・リストを入手する方法については、272ページの『ユーザーのデータベース・オブジェクト・リストのカスタマイズ』を参照してください。

フォールバック

フォールバックとは、ユーザーを *QMF* の前のリリースへ戻す (移行する) 過程のことです。終結処置 は、OS/390 から前のリリースを削除する過程のことです。終結処置については 109ページの『ステップ 36 -- インストール後の終結処置』で説明しているので、ここでは説明しません。

フォールバックは、*QMF* の 2 つのバージョンを同じ DB2 サブシステムから実行している場合を除いて、必要ありません。

前のプロファイルの再確立

QMF バージョン 2.4 にはない ENVIRONMENT 列は、バージョン 7 のプロファイルには影響しません。バージョン 2.2 以降に追加されたすべての列についても、同じことが言えます。

注: ログオン ID が 1 次許可 ID とは異なっていて、CREATOR の値が 1 次許可 ID を使用するように更新されていた場合には、フォールバックの一環として、それらをログオン ID に復元しなければなりません。

前のリリースでの QMF バージョン 7 のオブジェクトの使用

これは大幅に制限されています。フォールバックの余地はありますが、この付録でこれまでに説明してきた互換性の規則をユーザーによく理解してもらってください。これらの規則をまだ読んでいない場合には、585ページの『QMF オブジェクトの移行』を参照してください。

QMF の前のリリースにフォールバックする場合、QMF バージョン 7 で作成された一部のオブジェクトは前の環境では使用することができません。フォールバックの可能性を考える場合には、この点を考慮してください。以下に、一部のバージョン 7 のオブジェクトを前のリリースで使用する場合に適用される制約をリストします。

- 書式

バージョン 7 から保管またはエクスポートされ、QMF の前のリリースで表示またはインポートされる書式オブジェクトは、正常に実行されることができません。ただし、バージョン 7 から保管またはエクスポートされた書式オブジェクトは、バージョン 2.4 またはそれより前のバージョンでは使用できません。

制御の切れ目フィールド番号 (またはヘッダー・レコードのオブジェクト・レベル) を使用している、バージョン 7 からエクスポートされた書式は、前のアプリケーションで使用する前に、587ページの『書式アプリケーション移行援助プログラム』で説明されているように、書式アプリケーション移行援助プログラムを必要とします。

- 照会

前のリリースへのフォールバックについて、バージョン 7 の照会に、いくつかの制約事項が適用されます。

- SQL 照会: バージョン 7 から SQL 照会をエクスポートして前のリリースにインポートすることができ、その結果それらは正常に実行されます。ただし、バージョン 7 で保管された SQL 照会は、バージョン 2.4 以前では使用できません。

- 指示照会: バージョン 7 の指示照会を前のリリースで表示およびインポートすることはできません。ただし、それらの照会に変数、または前の 55 文字または 65 文字の制限を超える式が含まれていない場合です。
- QBE 照会: QBE (例示照会) により作成され、バージョン 7 で保管またはエクスポートされた照会は、前のリリースで表示またはインポートすることができ、正常に実行されます。
- プロシージャー
バージョン 7 からエクスポートされたプロシージャー・オブジェクトは、前のリリースにインポートでき、新規の QMF コマンドまたはコマンド構文が使用されていないければ、実行することができます。バージョン 7 で保管されたプロシージャー・オブジェクトは、まず最初にバージョン 7 からエクスポートした後でそれらを前のリリースにインポートしない限り、前のリリースでは表示することができません。ロジックを持つプロシージャーつまり REXX ロジックが入っているプロシージャーは、バージョン 3 より前のリリースで表示またはインポートすることはできません。
- 前のリリースでは実行できない QMF コマンドが入っているプロシージャーまたはアプリケーション
これらのコマンドは、さまざまな理由で実行に失敗することがあります。詳細については、『前のリリースでの バージョン 7 QMF コマンドの使用』を参照してください。
- 呼び出し可能インターフェースを呼び出すアプリケーション
アプリケーションは、QMF を呼び出すために CLIST およびプログラム内で呼び出し可能インターフェースを呼び出します。呼び出し可能インターフェースはバージョン 2.4 で採用されたため、それより前の QMF のバージョンで実行されるアプリケーションは使用できません。

前の QMF リリースと QMF バージョン 7 との相違点について詳細な情報が必要な場合は、*QMF 解説書* で 2 つのリリースを比較してください。

前のリリースでの バージョン 7 QMF コマンドの使用

バージョン 7 のプロシージャーおよびアプリケーションは、前のリリースでは実行できないコマンドを含んでいるために、QMF の前のリリースでは正しく実行されないことがあります。一部のコマンドでは、次のような場合があります。

- 前のリリースには存在しない。
- 前のリリースでは異なる作動をするオプションが含まれている。たとえば、DRAW コマンドは前と同じ構文ですが、現在では異なる結果を出します。

現在ではすべてのキーワードに二重引用符が付いているため、ユーザーが引用符を追加する必要がなく、二重引用符を付けるために使用していたツールは必要なくなっています。

31 桁 10 進数のサポート

QMF バージョン 3.1 (またはそれ以降) を使用していて、さらに DB2 バージョン 2.3 (またはそれ以降) で操作している場合には、31 桁の 10 進数のサポートがあります。一部の式では、15 桁から 31 桁の 10 進形式に移行する場合に、異常な結果または受け入れ不能な結果が生成されることがあります。

付録E. QMF および GDDM プログラムの CICS への定義のしかた

本書では、QMF プログラムを CICS に対して定義したり、GDDM 定義をロードするのに必要なジョブ、および QMF パネルの図表フォーマットを提供しています。デフォルトのインストールを変更する必要がある場合は、この節を使用して QMF プログラムがどのように定義されるか、また QMF インストール中に GDDM 定義がどのようにロードされるかを見つけてください。

QMF プログラムの CICS/MVS への定義のしかた

QMF のインストール中に、デフォルトのトランザクション ID である QMF n が QMF に対して定義されます。この場合、 n は、 xviii ページの表1 の各国語 ID です。トランザクション ID は、CICS プログラム管理テーブル (PCT) またはシステム定義 (CSD) ファイルのいずれかで定義されます。必要があれば、このデフォルトのトランザクション ID を次の目的で変更することができます。

- CSD を更新する場合は、*CICS/MVS Resource Definition (Online)* を参照してください。
- PCT を更新する場合は、*CICS/MVS Resource Definition (Macro)* を参照してください。

常駐の QMF プログラム

QMF インストール中に、以下のプログラムが CICS における常駐プログラムとして定義されます。

DSQQMF
DSQQMF n
DSQCBST
DSQC n LTT
DSQC n BLT

CICS/MVS は、RMODE(ANY) プログラムを永続常駐プログラムとして扱います。これは、16 Mb 境界より上に大きな仮想記憶域が使用可能なためです。常駐プログラムとして定義されたプログラムは、CICS システム初期化中にロードされます。常駐しないプログラムは、プログラムへの最初の参照時にロードされます。

QMF および GDDM プログラムの CICS への定義のしかた

最初に開始する QMF トランザクションによって、特定の GDDM プログラムがロードされます。詳細については、『非常駐 GDDM プログラムが QMF に与える影響』を参照してください。

非常駐プログラムがパフォーマンスに与える影響

何人かのユーザーが QMF を使用する場合、常駐記憶域から QMF プログラムを除去すると QMF および CICS のパフォーマンスに影響する場合があります。QMF は、ユーザーがプログラムを開始するたびにロードされるからです。ただし、インストール先システムで常駐記憶域からこれらのプログラムを除去する必要がある場合には、QMF プログラムの定義を常駐から非常駐に変更してください。

CEDA DEFINE PROGRAM コマンドで RESIDENT=NO と指定して、対話式に CSD のプログラム定義を変更することも、DFHPPT TYPE=ENTRY マクロで RES=NO と指定して、プログラム処理テーブル (PPT) の値を変更することもできます。CSD 中の QMF グループの PPT 項目に関する詳細については、*Installing and Managing QMF on VSE/ESA* を参照してください。

非常駐プログラムのパフォーマンスの考慮点については、*CICS/MVS Performance Guide* を参照してください。

QMF 初期化中の GDDM 定義のロードのしかた

OS/390 は、QMF 画面の印刷と表示に GDDM サービスを使用します。VSAM パネル・ファイル DSQPNLn には QMF 画面のテキストが入っており、QMF のインストール時に CICS に対して記述されます。QMF は、散布図、円グラフ、ヒストグラムなどのさまざまな図表の作成に GDDM-PGF 製品も使用します。

非常駐 GDDM プログラムが QMF に与える影響

GDDM プログラムは、常駐としては事前定義されていません。GDDM を CICS のために調整する時に、GDDM プログラムを常駐にすることを考慮します。これは、QMF の図表機能を使用するか否かにかかわらず、特定の GDDM プログラムは QMF が開始される時にロードされるからです。どのプログラムを常駐にするかを定める方法の詳細については、*CICS/MVS Performance Guide* を参照してください。GDDM の CICS のための調整については、以下の資料を参照してください。

GDDM Installation and System Management for OS/390 (GDDM 2.3)
GDDM システムのカスタマイズおよび管理 (GDDM 3.1)

QMF インストール後に図表機能を追加する

QMF のインストール後に GDDM-PGF をインストールする場合は、サブライブラリーに製品を復元するのではなく、GDDM-PGF をすべてインストールして CICS のために調整しなければなりません。

GDDM 3.1 を使用する場合は、GDDM-PGF 2.1.2 をインストールする必要があります。

GDDM 2.3 を使用する場合は、GDDM-PGF 2.1.1 をインストールする必要があります。

GDDM-PGF をインストールして調整した後に、GDDM を CICS のために調整中に GDDM により事前定義されている CICS ADMC トランザクションを実行してインストールを確認することができます。図表フォーマットをこれ以上カスタマイズする必要はありません。これらのフォーマットは、QMF のインストール中にユーザーのために定義されています。

リソースの使用を制御するためのトランザクション経路指定の使用

ユーザーのシステムの高速トランザクションを、余分のリソースを消費する可能性のある潜在的な長時間実行の QMF 照会から保護するために、連続領域操作または相互システム間通信を使用して、QMF トランザクションの実行を単一領域に隔離することを検討してください。1 つの CICS 端末所有領域を定義し、複数のトランザクション ID または動的経路指定出口を使用して、QMF トランザクション要求を別の領域に経路指定します。両方とも、*CICS/OS390 相互通信の手引き* の資料に説明があります。

QMF が CICS 領域で一領域を使用する方法に関する詳細については、203ページの『報告書記憶域のカスタマイズおよび報告書パフォーマンス』を参照してください。

付録F. QMF 制御表および QMF で使用される表スペース

QMF では、QMF ユーザーおよびユーザーが作成するオブジェクトを管理するために、表80 に示されている制御表を使用します。表スペースのサイズはページ単位で示されています。各ページは 4096 バイトのブロックです。表の構造および、QMF がこれをどのように使用するかに関する詳細が必要な場合は、表の右端にリストされているページを参照してください。

表 80. QMF 制御表および QMF が使用する表スペースのリスト

制御表名	表スペース	表スペースの サイズ (1K 単位)	表の内容	詳細な情報
Q.PROFILES	DSQTSPRO	100 - 1 次 20 - 2 次	QMF プロファイルが含まれる。これには、QMF セッション中のリソースおよびデータに対する各ユーザーのアクセスに関する情報が含まれています。	242 ページから 256 ページ
Q.OBJECT_DIRECTORY	DSQTSCT1	200 - 1 次 20 - 2 次	データベース内のすべての QMF 照会、書式、およびプロシージャーに関する一般的な情報が含まれていません。	285 ページ
Q.OBJECT_DATA	DSQTSCT3	5000 - 1 次 200 - 2 次	内部 QMF フォーマットで表した照会、書式、およびプロシージャーが入っています。	286 ページ
Q.OBJECT_REMARKS	DSQTSCT2	200 - 1 次 20 - 2 次	照会、書式、およびプロシージャーの作成 (または置換) 時に保管されたコメントが含まれています。	288 ページ
Q.COMMAND_SYNONYMS	DSQTSSYN	100 - 1 次 20 - 2 次	コマンド同義語に関する情報が含まれています。	343 ページ
Q.RESOURCE_TABLE	DSQTSGOV	100 - 1 次 20 - 2 次	管理プログラム出口ルーチンに渡されるリソース制御情報が含まれています。	433 ページ

QMF 制御表および QMF で使用される表スペース

表 80. QMF 制御表および QMF が使用する表スペースのリスト (続き)

制御表名	表スペース	表スペースの サイズ (1K 単位)	表の内容	詳細な情報
Q.ERROR_LOG	DSQTSLOG	100 - 1 次 20 - 2 次	システム・エラー、リソ ス・エラー、および「予 期しない状態」のエラ ーに関する情報が含ま れています。この情報 は、エラー・メッセ ージより詳細な情報 を含んでいます。	557 ページ
Q.DSQ_RESERVED	DSQTSRDO	100 - 1 次 20 - 2 次	初期化中に QMF が使 用する情報を含んで います。 重要 この表は変更 しないでください。	この表について は、本書では説明 していません。

制御表はすべて、DSQDBCTL データベースに入っています。

QMF 制御表について上記の表に示されている表スペース以外にも、QMF は QMF サンプル表のために表スペース DSQ1STBT を使用し、QMF SAVE DATA または IMPORT TABLE コマンドからのデータを保管するために DSQTSDEF を使用します。これらの表スペースは両方とも、デフォルトのサイズは 128 ページです。

QMF サンプル表および、SAVE DATA コマンドまたは IMPORT TABLE コマンドに関する詳細については、[QMF 使用の手引き](#) を参照してください。

付録G. 特記事項

本書において、日本では発表されていない IBM 製品 (機械およびプログラム)、プログラミングまたはサービスについて言及または説明する場合があります。しかし、このことは、弊社がこのような IBM 製品、プログラミングまたはサービスを、日本で発表する意図があることを必ずしも示すものではありません。本書で IBM ライセンス・プログラムまたは他の IBM 製品に言及している部分があっても、このことは当該プログラムまたは製品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 の有効な知的所有権またはその他の法的に保護された権利を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM によって明示的に指定されたものを除き、他社の製品と組み合わせた場合の操作の評価と検証はお客様の責任で行っていただきます。

IBM は、本書で解説されている主題について特許権 (特許出願を含む)、商標権、または著作権を所有している場合があります。本書の提供は、これらの特許権、商標権、および著作権について、本書で明示されている場合を除き、実施権、使用権等を許諾することを意味するものではありません。実施権、使用権等の許諾については、下記の宛先に書面にてご照会ください。

〒106-0032 東京都港区六本木 3 丁目 2-31
AP 事業所
IBM World Trade Asia Corporation
Intellectual Property Law & Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。

IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Corporation
J74/G4
555 Bailey Avenue
P.O. Box 49023
San Jose, CA 95161-9023
U.S.A.

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。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストはおこなっておりません。また、IBM 以外の製品に関するパフォーマンスの正確性、互換性、またはその他の要求は確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権表示

著作権使用許諾:

本書には、OS/2 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。サンプル・ソース・コードのすべての部分、またはすべての派生した創作物には、次のように、著作権表示を入れていただく必要があります。「(C) (お客様の会社名) (西暦年). All rights reserved.」

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は現れない場合があります。

商標

次の用語は、IBM Corporation の商標です。

ACF/VTAM	IBMLink
Advanced Peer-to-Peer Networking	IMS
AIX	Language Environment
AIX/6000	MVS
AS/400	MVS/ESA
C/370	MVS/XA
CICS	OfficeVision/VM
CICS/ESA	OS/2
CICS/MVS	OS/390
CICS/VSE	PL/I
COBOL/370	QMF
DATABASE 2	RACF
DataJoiner	S/390
DB2	SQL/DS
DB2 Universal Database	Virtual Machine/Enterprise Systems Architecture
Distributed Relational Database Architecture	Visual Basic
DRDA	VM/XA
DXT	VM/ESA
GDDM	VSE/ESA
IBM	VTAM

Java、またはすべての Java ベースの商標およびロゴ、および Solaris は、米国、およびその他の国における Sun Microsystems, Inc.の商標です。

Lotus および 1-2-3 は米国、およびその他の国における Lotus Development Corporation の商標です。

Microsoft、Windows、および Windows NT は、Microsoft Corporation の商標または登録商標です。

アスタリスクを 2 つ (**) つけて示す他の会社名、製品名、サービス名は、他社の商標またはサービス・マークです。

用語集

この用語集は、QMF ライブラリーの中で使用されている用語を定義したものです。探している用語が見つからない場合は、本書の索引または *IBM コンピューティング辞典* を参照してください。

異常終了 (abend). タスクの異常終了。

ABENDx. 異常終了問題のキーワード。

拡張対等通信ネットワーク機能 (Advanced Peer-to-Peer Networking). ネットワークを形成する複数のコンピューターがそれぞれ対等のものとして動的に通信できるようにする、分散ネットワークおよびセッション制御のアーキテクチャー。拡張プログラム間通信 (*Advanced Program-to-Program Communication (APPC)*) と比較。相互接続した複数のシステムが互いに通信でき、プログラムの処理を共用できるようにする SNA 同期データ・リンク制御 LU 6.2 プロトコルを具現化したもの。

拡張プログラム間通信 (APPC, Advanced Program-to-Program Communication). 相互に接続された複数のシステムが互いに通信でき、プログラムの処理を共用できるようにする SNA 同期データ・リンク制御 LU 6.2 プロトコルを具現化したもの。

総計関数 (aggregation function). 列中のデータを総計する関数の任意のグループ。書式パネル上で取扱コード AVERAGE、CALC、COUNT、FIRST、LAST、MAXIMUM、MINIMUM、STDEV、SUM、CSUM、PCT、CPCT、TPCT、TCPCT を使って要求される。

総計変数 (aggregation variable). FORM.BREAK、FORM.CALC、FORM.DETAIL、または FORM.FINAL パネルのいずれかを使って報告書の中に入れられる総計関数。その値は、作成された報告書に、切れ目後書き、明細ブロック・テキスト、または最終テキストの一部として現われる。

別名 (alias). DB2 UDB for OS/390 で、同じ、またはリモートの DB2 UDB for OS/390 サブシステム中の表あるいは視点を参照する SQL ステートメント中で使用できる代替名。OS/2 で、オブジェクト、データベース、または LU などのネットワーク・リソースを識別するために使用される代替名。QMF で、ローカルまたはリモート DB2 UDB for OS/390 サブシステムで保管されている QMF 表または視点にアクセスするために使用されるローカルに定義された名前。

APAR. プログラム診断依頼書 (Authorized Program Analysis Report)。

APPC. 拡張プログラム間通信 (Advanced program-to-program communication)。

アプリケーション (application). QMF ライセンス・プログラムを変更せずに QMF の機能を拡張する、QMF ユーザー作成のプログラム。QMF プロシージャー、インストール先定義コマンド、あるいは EXEC を呼び出す CMS または CLIST を呼び出す TSO のコマンドのための RUN コマンドを発行することによって QMF セッションから開始される。

用語集

アプリケーション・リクエスター (application requester). (1) アプリケーション・プロセスからデータベース要求を受け入れ、それをアプリケーション・サーバーに渡す機能。(2) DRDA において、リモート・リレーショナル・データベース管理システムへの送信元。

アプリケーション・リクエスターは、分散接続の QMF 終端を扱う DBMS コードである。QMF が接続するローカル DB2 UDB for OS/390 サブシステムは、QMF に対するアプリケーション・リクエスターとして知られる。DB2 UDB for OS/390 のアプリケーション・リクエスターはローカル・データベース・マネージャー内でインストールされるからである。したがって、DB2 UDB for OS/390 サブシステム全体(データを含む)がアプリケーション・リクエスターに関連しているが、SQL ステートメントは現行セッションで処理される。このサブシステムは、「ローカル DB2 UDB for OS/390」と呼ばれる。

DB2 (VM および VSE 版) では、アプリケーション・リクエスターは QMF と同じ仮想計算機で実行される。すなわち、どのデータベースも本来は、DB2 (VM および VSE 版) アプリケーション・リクエスターに関連付けられていない。

アプリケーション・サーバー (application server). アプリケーション・リクエスターからの要求のターゲット。(1) アプリケーション・プロセスが接続されるローカルまたはリモートのデータベース・マネージャー。アプリケーション・サーバーは、所要のデータをもっているシステムで実行される。(2) DRDA では、アプリケーション・リクエスターからの要求のターゲット。DB2 UDB for OS/390 では、アプリケーション・サーバーは、完全な DB2 UDB for OS/390 サブシステムの一部である。

DB2 (VM および VSE 版) では、アプリケーション・サーバーは、DB2 (VM および VSE 版) データベース・マシンの一部である。

アプリケーション・サポート・コマンド (application-support command). アプリケーション・プログラム内で使用され、アプリケーション・プログラムと QMF 間で情報を交換できるようにする、QMF コマンドの 1 つ。この種のコマンドには、INTERACT、MESSAGE、STATE、および QMF がある。

区域分離記号 (area separator). 表示される報告書の固定域とその他の区域を区分するバリア。

引き数 (argument). 独立変数。

基本 QMF 環境 (base QMF environment). QMF のインストール時に設定される QMF の英語環境。他の言語環境は、インストール後に設定される。

バッチ QMF セッション (batch QMF session). バックグラウンドで実行する QMF セッション。指定された QMF プロシージャが呼び出されたときに開始し、そのプロシージャが終了したときに終了する。バックグラウンド QMF セッション時には、ユーザー対話もパネル表示対話も行えない。

バインド (bind). DRDA で、アプリケーション・プログラム内の SQL ステートメントを、アプリケーション・サポート・プロトコル (およびデータベース・サポート・プロトコル) フロー上でデータベース管理システムに認識させるプロセス。バインドの際に、プリコンパイラーまたはプリプロセッサからの出力は、パッケージと呼ばれる制御構造に変換される。さらに、参照されたデータへのアクセス・パスが選択され、一部の許可検査が実行される。(オプションで、DB2 UDB for OS/390 では、出力がアプリケーション・プランとなることがある。)

組み込み関数 (built-in function). スカラー関数または列関数の総称。「関数」と呼ばれることもある。

計算変数 (calculation variable). CALCid は、ユーザー定義の計算値が入る書式用の特殊変数である。CALCid は FORM.CALC パネルで定義される。

呼び出し可能インターフェース (callable interface). QMF サービスへのアクセスを可能にするプログラミング・インターフェース。アプリケーションは、QMF セッション外で実行している場合でも、これらのサービスにアクセスできる。コマンド・インターフェース (*command interface*) と対比。

図表 (chart). 報告書の情報のグラフィック表示。

CICS. 顧客情報管理システム (Customer Information Control System)。

クライアント (client). サーバーから共用サービスを受ける機能単位。

CMS. 会話型モニター・システム (Conversational Monitor System)。

列 (column). 表データの垂直方向の集合。特定のデータ・タイプ (たとえば、文字または数値) と名前をもつ。列内の値は、すべて同じデータ特性をもっている。

列関数 (column function). 列内のすべての値に 1 回ずつ適用され、結果として単一の値を戻す操作であり、関数名の後に、括弧で囲んだ 1 つまたは複数の引き数を続けた形で表すもの。

列ヘッダー (column heading). ユーザーが書式に指定できる列名の代替。列名やラベルとは異なり、データベースには保管されない。

列ラベル (column label). データベースに保管される、データの列の代替記述子。使用すると、列ラベルは書式にデフォルトとして現われるが、ユーザーはそれを変更することができる。

列の折り返し (column wrapping). 1 つの列が数行にまたがることを可能にする報告書内の書式設定の値。長さが列幅を超える値を列に入れるときに、しばしば使用される。

コマンド・インターフェース (command interface). QMF コマンドを実行するためのインターフェース。QMF コマンドは、アクティブ QMF セッションからしか出せない。呼び出し可能インターフェース (*callable interface*) と対比。

コマンド同義語 (command synonym). インストール先定義コマンドの動詞または動詞 / 目的語の部分。ユーザーは、コマンドの代わりにコマンド同義語を入力し、その後他の必要情報を続ける。

コマンド同義語表 (command synonym table). それぞれの行にインストール先定義コマンドを記述する表。各ユーザーに、これらの表の 1 つを割り当てることができる。

コミット (commit). データに永続的な変更を加える処理。コミットされると、データ・ロックは解除され、コミットされたばかりのデータを、他のアプリケーションが使用できるようになる。ロールバック (*rollback*) も参照。

連結 (concatenation). 2 番目のストリングを 1 番目のストリングに付加して、2 つのストリングを 1 つに結合すること。

用語集

接続性 (connectivity). 異なるシステムが互いに通信できるようにすること。たとえば、DB2 UDB for OS/390 アプリケーション・リクエスターと DB2 (VM および VSE 版) アプリケーション・サーバー間の接続性によって、DB2 UDB for OS/390 ユーザーは、DB2 (VM および VSE 版) データベースのデータを要求できる。

会話 (conversation). LU 6.2 セッション上で、トランザクションを処理する一方で相互に通信することを可能にする 2 つのプログラム間の論理接続。

相関名 (correlation name). SELECT 照会の FROM 文節で指定された表名の別名。列名と連結させると、列が属する表を識別できる。

CP. VM 用制御プログラム (Control Program)。

CSECT. 制御セクション (Control section)。

現行ロケーション (current location). QMF セッションが現在接続されているアプリケーション・サーバー。CONNECT などの接続タイプのステートメント (これらはアプリケーション・リクエスターによって処理される) を除いて、このサーバーはすべての SQL ステートメントを処理する。QMF を初期化する場合、現行ロケーションは DSQSDBNM 始動プログラム・パラメーターにより指示される。(そのパラメーターが指定されていない場合は、ローカル DB2 UDB for OS/390 サブシステム)

現行オブジェクト (current object). 一時記憶域にあって、現在表示されているオブジェクト。保管オブジェクト (*saved object*) と対比。

顧客情報管理システム (Customer Information Control System (CICS)). リモート端末で入力されるトランザクションをユーザー作成アプリケーション・プログラムによって並行して処理できるようにする IBM ライセンス・プログラム。これには、データベースの構築、使用、維持管理の機能が含まれる。

DATA. 検索照会で戻された情報を収容する一時記憶域内のオブジェクト。表に含まれ、報告書でフォーマット設定された英数字で表現される情報。

データベース (database). 複数のユーザーの求めに応じてデータを受け入れ、保管し、提供するための所定の構造をもつデータの集合。DB2 UDB for OS/390 において、表スペースおよび索引スペースが入っている作成済みオブジェクト。DB2 (VM および VSE 版) では、システムで維持管理される表、索引、サポート情報 (制御情報およびデータ回復情報など) の集合。OS/2 では、表、視点、索引など、情報の集合。

データベース管理者 (database administrator). データベースの内容とデータベースへのアクセスを管理する担当者。

データベース管理システム (database management system (DBMS)). データベースを定義、作成、操作、制御、管理、使用するためのコンピューター・ベースのシステム。データベース管理システムはまた、データの保全性を保護するためのトランザクション管理およびデータ管理機能をもつ。

データベース・マネージャー (database manager). データベースを作成し維持管理するため、またデータベースをアクセスする必要のあるプログラムと通信するために使用されるプログラム。

データベース・サーバー (database server). (1) DRDA において、アプリケーション・サーバーから受け取った要求のターゲット。(2) OS/2 において、そのローカル・データベースのデータベース・サービスをデータベースのクライアントに提供するワークステーション。

日付 (date). 日、月、および年 (3 部分から構成される値) を指す。

日付 / 時刻デフォルト・フォーマット (date/time default formats). データベース・マネージャーのインストール・オプションで指定される日付 / 時刻フォーマット。そのフォーマットには、EUR、ISO、JIS、USA、または LOC (LOCAL) がある。

日付 / 時刻データ (date/time data). データ・タイプが DATE、TIME、または TIMESTAMP である、表の列内のデータ。

DB2 UDB for OS/390. DB2 ユニバーサル・データベース (OS/390 版) (IBM のリレーショナル・データベース管理システム)。

DB2 (AIX 用). AIX 用の DATABASE2。QMF のリレーショナル・データのデータベース・マネージャー。

DBCS. 2 バイト文字 セット (double-byte character set)。

DBMS. データベース管理システム (Database management System (DBMS))。

デフォルト書式 (default form). 照会が実行されるときに、QMF が作成する書式。保管されている書式が照会で実行されるときは、デフォルト書式は作成されない。

宛先管理テーブル (destination control table - DCT). CICS で、それぞれの一時データ・キューの宛先を収めているテーブル。

明細ブロック・テキスト (detail block text). 個々のデータ行に対応している、報告書の本文中のテキスト。

明細ヘッダー・テキスト (detail heading text). 報告書のヘッダーのテキスト。ヘッダーを印刷するかどうかは、FORM.DETAIL で指定する。

ダイアログ・パネル (dialog panel). 指示照会基本パネルの一部を重ね書きし、照会の作成を援助するダイアログを拡張するパネル。

分散データ (distributed data). ネットワーク内の複数のシステムに保管され、リモート・ユーザーおよびアプリケーション・プログラムで使用できるデータ。

分散データベース (distributed database). ユーザーからは論理的な全体として見ることができ、ローカルにアクセスできるが、実際は複数のロケーションにあるデータベースから構成されているデータベースの 1 つ。

分散リレーショナル・データベース (distributed relational database). すべてのデータがリレーショナル・モデルに従って保管されている分散データベースの 1 つ。

用語集

分散リレーショナル・データベース体系 (Distributed Relational Database Architecture). IBM とベンダーのリレーショナル・データベース・プロダクトで使用される分散リレーショナル・データベース処理の接続プロトコル。

分散作業単位 (distributed unit of work). 分散リレーショナル・データにアクセスする方式。単一の作業単位内で、ユーザーまたはアプリケーションが SQL ステートメントを複数のロケーションに実行依頼できる (ただし、SQL ステートメントと RDBMS は 1 対 1 に対応する)。

DB2 UDB for OS/390 では V2R2 において、QMF がサポートするシステム指示アクセスと呼ばれる、制限された形の分散作業単位サポートが採用されている。

DOC. 文章問題のキーワード。

2 バイト文字 (double-byte character). 2 個の文字バイトを必要とするエンティティ。

2 バイト文字セット (double-byte character set (DBCS)). 各文字が 2 バイトで表現される文字セット。日本語、中国語、韓国語など、256 個のコード・ポイントでは表現できないほど多くの記号をもつ言語には、2 バイト文字セットが必要である。各文字が 2 バイトを必要とするため、DBCS 文字のタイプ、表示、印刷には、DBCS をサポートするハードウェアとプログラムが必要となる。1 バイト文字セット (*single-byte character set*) と対比。

DRDA. 分散リレーショナル・データベース体系 (Distributed Relational Database Architecture (DRDA))。

期間 (duration). 数値の後に次の 7 つのキーワード、すなわち、YEARS、MONTHS、DAYS、HOURS、MINUTES、SECONDS、MICROSECONDS の 1 つを続けて表される時間の量。

EBCDIC. 拡張 2 進化 10 進コード (Extended Binary-Coded Decimal Interchange Code)。

確認域 (echo area). 指示照会を作成する指示照会基本パネルの一部。

ヨーロッパ・フォーマット (EUR (European) format). 以下のように日付 / 時刻値を表すフォーマット。

- 日付: dd.mm.yyyy
- 時刻: hh.mm.ss

拡張構文 (extended syntax). QMF 呼び出し可能インターフェースで使用する QMF コマンド構文。この構文は、呼び出し可能インターフェース・アプリケーションで獲得され、QMF と共用される記憶域に保管される変数を定義する。

例示エレメント (example element). QBE 照会において、計算または条件の中で使用される値のシンボル。

例示表 (example table). QBE 照会の枠組み。

固定域 (fixed area). 固定列を入れる報告書の部分。

固定列 (fixed columns). ユーザーが水平方向にスクロールしても、その位置から移動しない報告書の列。複数ページの印刷報告書では、これらの列は各ページの左側に繰り返される。

フォーム (form). 報告書または図表の、印刷または表示のための仕様が入っているオブジェクト。一時記憶域にある書式は、FORM という名前をもつ。

機能キー表 (function key table). キーを記述するテキストが付いている 1 つまたは複数の QMF パネルの機能キー定義が入っている表。各ユーザーに、これらの表の 1 つを割り当てることができる。

ゲートウェイ (gateway). 異なるネットワーク・アーキテクチャーをもつ 2 つのコンピューター・ネットワークを接続する機能単位。同一または類似のアーキテクチャーでネットワークまたはシステムを接続するブリッジと異なり、ゲートウェイは、異なるアーキテクチャーのネットワークまたはシステムを接続する。

GDDM. 図形データ表示管理プログラム (Graphical Data Display Manager)。

グローバル変数 (global variable). 1 回セットすると、1 つの QMF セッションの間使用できる変数。グローバル変数はプロシージャー、照会、または書式で使用できる。ランタイム変数 (*run-time variable*) と対比。

図形データ表示管理プログラム (Graphical Data Display Manager). ピクチャーが、グラフィック・プリミティブに対応する機能ルーチンを通してプロシージャーにのっとり定義および表示できるようにするルーチンのグループ。

グループ化行 (grouped row). G 関数または組み込み関数のいずれかで合計される QBE ターゲット表または例示表のデータ行。

HELP. エラー・メッセージ、QMF パネル、または QMF コマンドとそのオプションについての追加情報。

ホスト (host). ネットワークでのサービスをワークステーションに提供するメインフレームまたは中間サイズのプロセッサ。

HTML. ハイパーテキスト・マークアップ言語 (Hypertext Markup Language)。WWW で表示される文書用の標準化されたマークアップ言語。

ICU. 対話式図表ユーティリティ (Interactive Chart Utility)。

INCORROUT. 正しくない出力のキーワード。

索引 (index). 特定のキーをもつレコードに迅速にアクセスできるようにする、表中のレコード位置に関するデータの集合。

初期プロシージャー (initial procedure). QMF の呼び出し直後に実行される QMF 開始コマンドの DSQSRUN パラメーターで指定される QMF プロシージャーの 1 つ。

初期化プログラム (initialization program). QMF プログラム・パラメーターを設定するプログラム。このプログラムは、呼び出し可能インターフェースの DSQSCMD で指定される。対話式 QMF のデフォルト・プログラムは DSQSCMD n である。ここで、 n は主要言語の修飾子である (英語の場合は 'E')。

用語集

インストール先定義コマンド (installation-defined command). インストール先で作成されたコマンド。 QMF は、このコマンドを QMF 自身のコマンドの 1 つとして、またはそのコマンドの組み合わせとして処理する。

インストール先定義フォーマット (installation-defined format). インストール先で定義 (または作成) される日付 / 時刻フォーマット。 LOCAL フォーマットとも呼ばれる。

対話式実行 (interactive execution). コマンドの実行の間にユーザーと QMF との間に発生すべきすべてのダイアログが実際に行われる、QMF の実行。

対話式セッション (interactive session). ユーザーと QMF が対話できる任意の QMF セッション。 QMF INTERACT コマンドを使用すれば、ある対話式セッションから別の対話式セッションを開始できる。

対話式スイッチ (interactive switch). オンになっていると、アプリケーション・プログラムで QMF コマンドを対話的に実行できるようにする概念上のスイッチ。

呼び出し CLIST または EXEC (invocation CLIST or EXEC). QMF を呼び出す (開始する) プログラム。

ISO (国際標準化機構) フォーマット (International Standards organization format). 以下のように日付 / 時刻値を表すフォーマット。

- 日付: yyyy-mm-dd
- 時刻: hh.mm.ss

ISPF. 対話式システム生産性機能 (Interactive System Productivity Facility)。

IXF. 統合交換フォーマット (Integration Exchange Format)。各種のソフトウェア・プロダクト間で表データを転送するためのプロトコル。

JCL. OS/390 用のジョブ制御言語。

ジョブ制御 (job control). VSE で、各ジョブまたはジョブ・ステップの実行準備のために記憶域に呼び出されるプログラム。その機能の例としては、入出力装置を記号名に割り当てたり、スイッチをプログラムで使用するために設定したり、制御ステートメントをログ (または印刷) したり、各ジョブ・ステップの第 1 段階を取り出すことなどがある。

JIS (日本工業規格) フォーマット (Japanese Industrial Standard format). 以下のように日付 / 時刻値を表すフォーマット。

- 日付: yyyy-mm-dd
- 時刻: hh:mm:ss

結合 (join). リレーショナル操作の 1 つで、同じデータ・タイプの値をもつ列を突き合わせることによって、複数の表からデータを検索することができるようにするもの。

キーワード・パラメーター (keyword parameter). キーワードと割り当て値から構成される QMF コマンドの 1 要素。

同種 (like). 複数の類似または同一の IBM 操作環境を表現する語。たとえば、同種分散は、互換性のあるサーバー属性レベルによる 2 つの DB2 UDB for OS/390 間の分散である。異種 (*unlike*) と対比。

リテラル (literal). プログラミング言語において、1 つの値を直接表す字句単位。文字自体によって値が与えられている文字ストリング。

線形プロシージャー (linear procedure). REXX コメントで始まっていない プロシージャー。線形プロシージャーには、QMF コマンド、コメント、ブランク行、RUN コマンド、置換変数を入れることができる。ロジックを持つプロシージャー (*procedure with logic*) も参照。

線形構文 (linear syntax). あるプログラムまたはプロシージャーの 1 ステートメントに入力するか、または QMF コマンド行に入力できる QMF のコマンド構文。

行の折り返し (line wrapping). 報告書内の表の行が複数行を占めることができるようにする、表中の行のフォーマット設定。列名の行と列値の各行が、報告書の行の長さに必要なだけの行数に分割される。

ローカル (local). ユーザーのプロセッサに所在するリレーショナル・データベース、データ、またはファイルを形容する用語。ローカル *DB2 UDB for OS/390 (local DB2 UDB for OS/390)* も参照。リモート (*remote*) と対比。

ローカル・エリア・ネットワーク (local area network (LAN)). (1) ローカル・リソース共用のために接続された複数のプロセッサ。 (2) 限定された地域 (たとえば、単独のオフィス・ビル、倉庫、キャンパスなど) 内のネットワーク。

ローカル・データ (local data). データをアクセスしようとするサブシステムが維持管理するデータ。リモート・データ (*remote data*) と対比。

ローカル DB2 UDB for OS/390 (local DB2 UDB for OS/390). DB2 UDB for OS/390 と併用する場合は、アプリケーション・リクエスターは QMF と同じ MVS システムで実行される DB2 UDB for OS/390 サブシステムの一部である。したがって、DB2 UDB for OS/390 サブシステム全体 (データを含む) がアプリケーション・リクエスターに関連するが、SQL ステートメントは現行ロケーションで処理される。このサブシステムは、QMF プランがバインドされる個所である。

QMF が TSO で実行するときには、このサブシステムは、DSQSSUBS 始動プログラム・パラメーターを使用して指定される。QMF が CICS で実行するときには、このサブシステムはリソース管理テーブル (RCT) で識別される。ローカル DB2 UDB for OS/390 は、CICS 領域で開始された DB2 UDB for OS/390 のサブシステム ID である。

ロケーション (location). 分散リレーショナル・データベース・システムの特定のリレーショナル・データベース管理システム。各 DB2 UDB for OS/390 サブシステムは 1 つのロケーションと見なされる。

論理装置 (logical unit - LU). エンド・ユーザーが他のエンド・ユーザーと通信するために SNA ネットワークにアクセスし、これを通してエンド・ユーザーがシステム・サービス制御点から提供される機能にアクセスするポート。

論理装置タイプ 6.2 (Logical Unit type 6.2 (LU 6.2)). 分散処理環境におけるプログラム間の一般的な通信をサポートする SNA 論理装置タイプ。

用語集

LU. 論理装置 (Logical Unit)。

LU 6.2. 論理装置タイプ 6.2。

LOOP. 無限ループの問題のキーワード。

MSGx. メッセージ問題のキーワード。

多重仮想記憶 (Multiple Virtual Storage). MVS/ESA プロダクトを意味する。

MVS/ESA. 多重仮想記憶 / エンタープライズ・システム体系 (IBM のオペレーティング・システム)。

NCP. ネットワーク制御プログラム。

ネットワーク制御プログラム (Network Control Program (NCP)). 単一ドメイン、複数ドメイン、相互接続ネットワーク機能に対する通信コントローラー・サポートを提供する IBM のライセンス・プログラム。

NLF. 各国語機能 (National Language Feature)。米国英語以外の言語を 1 つ選択できるようにする、QMF とともに使用できるいくつかのオプション・フィーチャーの 1 つ。

NLS. 各国語サポート (National Language Support)。

ノード (node). SNA で、リンクの終点またはネットワークの複数のリンクに共通の接続点。ノードは、ホスト・プロセッサ、通信コントローラー、クラスター・コントローラー、または端末に分散できる。ノードは、ルーティングや他の機能の点でさまざまである。

ヌル (null). ある行のある列に値がないときに使用される特殊な値。ヌル は、ゼロと同じではない。

ヌル値 (null value). *ヌル (null)* を参照。

オブジェクト (object). QMF 照会、書式、プロシージャ、プロファイル、報告書、図表、データ、または表。報告書、図表、およびデータ・オブジェクトは、一時記憶域だけに存在し、データベースに保管できない。表オブジェクトは、データベースにだけ存在する。

オブジェクト名 (object name). QMF ユーザーが所有するオブジェクトを識別する文字ストリング。文字ストリングは最大長が 18 バイトで、英字で始まっていなければならない。用語「オブジェクト名」には、「所有者名」接頭部は含まれない。ユーザーは許可を得た場合にだけ、他のユーザーのオブジェクトにアクセスできる。

オブジェクト・パネル (object panel). 1 つの QMF コマンドの実行後、別の QMF コマンドの実行前に、オンラインで表示できる QMF パネル。そのようなパネルには、ホーム・パネル、報告書パネル、図表パネル、および QMF オブジェクトを表示するすべてのパネルがある。リスト・パネル、ヘルプ・パネル、プロンプト・パネル、および状況パネルは含まれない。

オンライン実行 (online execution). オブジェクト・パネルからの、または機能キーを押すことによるコマンドの実行。

所有者名 (owner name). 特定のオブジェクトを作成したユーザーの許可 ID。

パッケージ (package). アプリケーション・プログラムの SQL ステートメントがリレーショナル・データベース管理システムにバインドされる場合につくられる制御構造。データベース管理システムは、この制御構造を使用して、ステートメントの実行時に見出される SQL ステートメントを処理する。

パネル (panel). ウィンドウに表示するためにグループ分けした情報の特定の配列。パネルには、通知用のテキスト、入力フィールド、ユーザーが選択できるオプション、またはそれらの組み合わせを入れることができる。

パラメーター (parameter). QMF コマンドの 1 要素。この用語は QMF 資料では キーワード・パラメーター と定位置パラメーター の総称として用いられる。

パートナー論理装置 (partner logical unit). SNA において、セッションのリモート・システム。

PERFM. パフォーマンス問題のキーワード。

永続記憶域 (permanent storage). すべての表および QMF オブジェクトが保管されているデータベース。

プラン (plan). 複数のプログラムの SQL ステートメントがバインド時に一緒に集められてプランが作られる、パッケージの 1 つの形式。

定位置パラメーター (positional parameter). コマンド内の決まった位置に指定しなければならない QMF コマンドの 1 要素。

基本パネル (primary panel). ユーザーの照会を入れる指示照会のメイン・パネル

1 次 QMF セッション (primary QMF session). QMF の外部から開始される対話式セッション。このセッションの中で、他のセッションを開始するときは、INTERACT コマンドを使用する。

プロシージャー (procedure). QMF コマンドが入っているオブジェクト。プロシージャーは単一の RUN コマンドで実行できる。一時記憶域にあるプロシージャーは、PROC という名前をもつ。線形プロシージャー (*linear procedure*) およびロジックを持つプロシージャー (*procedure with logic*) も参照。

プロシージャー終了スイッチ (procedure termination switch). QMF MESSAGE コマンドでオンにできる概念上のスイッチ。オンの状態になっていると、制御が戻る QMF プロシージャーはすべて直ちに終了する。

ロジックを持つプロシージャー (procedure with logic). REXX のコメントで始まる QMF プロシージャー。ロジックを持つプロシージャーでは、条件つきロジックの実行、計算、ストリングの作成、およびホスト環境へのコマンドの戻しが可能である。線形プロシージャー (*linear procedure*) も参照。

プロファイル (profile). ユーザーのセッションの特性情報が入っているオブジェクト。保管プロファイルとは、永続記憶域に保管されているプロファイルのことである。一時記憶域にあるプロファイルは、PROFILE という名前をもつ。ユーザーは、それぞれプロファイルを 1 つしかもつことができない。

用語集

プロンプト・パネル (prompt panel). 不完全なまたは正しくない QMF コマンドが出された後で表示されるパネル。

指示照会 (Prompted Query). 1 組のダイアログ・パネルに対するユーザーの応答に従って作成される照会。

プロトコル (protocol). 通信をおこなう場合に順守する必要がある、通信システムの機能を律する規則。

PSW. プログラム状況ワード (Program status word)。

PTF. プログラム一時修正 (Program temporary fix)。

例示照会 QBE (Query-By-Example). 照会を図形的に作成するときに使用する言語。詳しくは、*QMF 使用の手引き* を参照。

QMF 管理権限 (QMF administrative authority). 最低でも、Q.PROFILES 制御表に対する挿入または削除の権限。

QMF 管理者 (QMF administrator). QMF 管理権限を持つ QMF ユーザー。

QMF コマンド (QMF command). QMF 言語の一部である任意のコマンドを指す。インストール先定義コマンドを含まない。

QMF セッション (QMF session). ユーザーが QMF を呼び出した時点から、EXIT コマンドを出すまでの、ユーザーと QMF 間で行なうすべての対話。

修飾子 (qualifier). QMF オブジェクトについて用いられるときは、所有者を識別する名前の部分。TSO データ・セットについて用いられるときは、それ以外の名前の部分とピリオドで区切られている名前の部分。たとえば、'TCK'、'XYZ'、および 'QUERY' は、すべてデータ・セット名 'TCK.XYZ.QUERY' の修飾子である。

照会 (query). データの照会または操作を実行する SQL ステートメントまたは QBE ステートメント、もしくはプロンプト指示によって作成されるステートメント。保管照会とは、データベースに保管されている SQL 照会、QBE 照会、または指示照会のことである。一時記憶域にある照会は、QUERY という名前をもつ。

RDBMS. リレーショナル・データベース管理システム (Relational database management system)。

リレーショナル・データベース (relational database). ユーザーが表の集合として認識するデータベース。

リレーショナル・データベース管理システム (relational database management system (RDBMS)). リレーショナル・データベースを定義、作成、操作、制御、管理、使用するためのコンピューター・ベースのシステム。

リモート (remote). ローカル・リレーショナル DBMS 以外のリレーショナル DBMS を指す語。

リモート・データ (remote data). データをアクセスしようとするサブシステム以外のサブシステムが維持管理するデータ。ローカル・データ (*local data*) と対比。

リモート・データ・アクセス (remote data access). リモート・ロケーションからデータを取り出すための方法。QMF で使用する 2 つのリモート・データ・アクセス機能は、リモート作業単位 と DB2 UDB for OS/390 専用分散作業単位 (システム指示アクセス と呼ばれる) である。

リモート作業単位 (remote unit of work). (1) アプリケーションがリレーショナル・データベースとは異なるシステム上にあり、単独アプリケーション・サーバーが単一の論理作業単位内ですべてのリモート作業単位要求にこたえるという、SQL 分散処理の形。(2) SQL ステートメントのリモートでの準備と実行を可能にする作業単位。

報告書 (report). データを検索するための照会が出されたとき、または表か視点に対する DISPLAY コマンドが入力されたときに作成されるフォーマット設定されたデータ。

REXX. 再構造化拡張実行プログラム (Restructured extended executor)。

ロールバック (rollback). アプリケーションまたはユーザーが行った、コミットされていないデータベース変更を除去するプロセス。ロールバックが起こると、ロックは解除され、変更が加えられたリソースの状態は、前回コミット、ロールバック、または開始されたときの状態に戻される。コミット (*commit*) も参照。

行 (row). 表データの水平方向の集合。

行演算子域 (row operator area). QBE ターゲット表または例示表の最左端の列。

ランタイム変数 (run-time variable). プロシージャまたは照会が実行される時にユーザーがその値を指定するプロシージャまたは照会の変数。ランタイム変数の値は、現行のプロシージャまたは照会でしか使用できない。グローバル変数 (*global variable*) と対比。

サンプル表 (sample tables). QMF に添えて出荷される表。サンプル表のデータは、QMF を初めて体験するユーザーがプロダクト学習のために使用できる。

保管オブジェクト (saved object). データベースに保管されているオブジェクト。現行オブジェクト (*current object*) と対比。

SBCS. 1 バイト文字セット (single-byte character set)。

スカラー (scalar). 列、リテラルの値、または他のスカラーを含む式の値。

スカラー関数 (scalar function). 1 つの値を別の値から作成し、関数名の後ろに括弧で囲んだ引き数リストを付けた形で表す操作。

画面 (screen). ユーザーに情報を表示するディスプレイの物理的な表示面。

スクロール可能域 (scrollable area). 上、下、左、右に移動可能な表示オブジェクトの部分。

用語集

サーバー (server). 共用サービスをネットワーク上でワークステーションに提供する機能単位。

セッション (session). ユーザーがログオンしてからログオフするまでの、ユーザーと QMF 間のすべての対話。

1 バイト文字 (single-byte character). 内部表示が 1 バイトから構成される文字。英字は 1 バイト文字の例である。

SNA. システム・ネットワーク体系 (Systems Network Architecture (SNA))。

SNAP ダンプ (SNAP dump). 異常終了時に QMF が生成する 1 つまたは複数の記憶域の内容の動的ダンプ。

ソート優先順位 (sort priority). 検索されたある列のソート値が別の検索列の値のソートを決定する、検索照会の仕様の 1 つ。

SQL. 構造化照会言語 (Structured Query Language)。

SQLCA. 構造化照会言語連絡域 (Structured Query Language Communication Area)。

SSF. ソフトウェア・サポート機能 (Software Support Facility)。現行のすべての APAR および PTF に関する情報の保管および検索を可能にする IBM のオンライン・データベースの 1 つ。

保管オブジェクト (stored object). 永続記憶域に保管されているオブジェクト。現行オブジェクト (*current object*) と対比。

ストリング (string). 1 組の連続した同じタイプの項目。たとえば、文字ストリング。

構造化照会言語 (Structured Query Language (SQL)). DB2 UDB for OS/390 および DB2 (VSE または VM 版) と通信を行なうために使用する言語。記述文で照会を書くために使用される。

副照会 (subquery). 別の照会 (主照会またはより高位の副照会) の WHERE または HAVING 文節の中に現れる完結した SQL 照会。

置換変数 (substitution variable). (1) 値が、グローバル変数またはランタイム変数のいずれかで指定される、プロシーチャー内または照会内の変数。 (2) 値が、グローバル変数で指定される書式の変数。

サブストリング (substring). SUBSTR 関数で開始位置と長さを指定する、ストリングの一部。

システム・ログ (System Log (SYSLOG)). オペレーターとの間での、ジョブ関連情報、操作上のデータ、通常でない状況発生の記述、コマンド、およびオペレーターとの間のメッセージを保管できるデータ・セットまたはファイル。

システム・ネットワーク体系 (Systems Network Architecture). ネットワークの構成および操作を通し、またそれを制御することによって情報単位を伝送するための論理構造、フォーマット、プロトコル、および操作手順の記述。

表 (table). リレーショナル・データベース・マネージャーの制御下にあるの名前の付いたデータの集合。表は、固定数の行と列から構成される。

表編集プログラム (Table Editor). 許可ユーザーが照会を作成しなくてもデータベースに変更を加えることができるようにする QMF の対話式編集プログラム。

表名域 (table name area). QBE 例示表の最左端の列。

表データ (tabular data). 列のデータ。データの内容および書式は、FORM.MAIN および FORM.COLUMNS で指定される。

ターゲット表 (target table). 例示エレメントを使用して、列を組み合わせたたり、行を組み合わせたたり、報告書に定数値を組み込んだりする空の表。

一時記憶域 (temporary storage). 現在使用している照会、書式、プロシーチャー、プロファイル、報告書、図表、およびデータ・オブジェクトを保管する区域。データ・オブジェクト以外はすべて表示できる。

一時記憶域キュー (temporary storage queue). CICS において、QMF とアプリケーションまたはシステム・サービスの間のオブジェクト転送に用いられる一時記憶域。

時刻 (time). 時刻を時、分、必要に応じて秒 (2 つまたは 3 つの部分値) で指定する。

スレッド (thread). アプリケーションの接続の記述、その進行のトレース、リソース機能処理能力の提供、および DB2 UDB for OS/390 リソースとサービスへのアクセス可能性の限度の決定を行なう DB2 UDB for OS/390 の構造。ほとんどの DB2 UDB for OS/390 機能は、スレッド構造のもとで実行される。

3 部分名 (three-part name). ロケーション名、所有者 ID、およびオブジェクト名から構成される表または視点の完全修飾名。アプリケーション・サーバー (つまり DB2 UDB for OS/390) でサポートされる場合は、3 部分名を SQL ステートメントで使って、指定されたロケーションの指定された表または視点を検索または更新することができる。

タイム・スタンプ (timestamp). 日付および時刻、必要に応じてマイクロ秒 (6 または 7 部分値)。

TP. トランザクション・プログラム (Transaction Program)。

TPN. トランザクション・プログラム名 (Transaction program name)。

トランザクション (transaction). 「作業単位の開始」から「コミット」または「ロールバック」までに発生する作業。

トランザクション・プログラム (transaction program). SNA ネットワークでトランザクションを処理するプログラム。2 種類のトランザクション・プログラムがある。アプリケーション・トランザクション・プログラムとサービス・トランザクション・プログラムである。

用語集

トランザクション・プログラム名 (transaction program name). LU 6.2 会話に参加する各プログラムがそれによって識別される名前。通常、接続のイニシエーターは、他の LU で接続を希望するプログラムの名前を識別する。LU 名とともに使用される場合、ネットワークでの特定のトランザクション・プログラムを識別する。

一時データ・キュー (transient data queue). CICS において、宛先管理テーブル (DCT) で定義されている名前をもつ記憶域。ここには、後続の内部処理または外部処理に備えてオブジェクトが保管される。

TSO. タイム・シェアリング・オプション (Time Sharing Option)。

2 フェーズ・コミット (two-phase commit). 参加しているリレーショナル・データベース管理システムが 1 作業単位を矛盾なくコミットまたはロールバックできるようにするために分散作業単位で使用されるプロトコル。

作業単位 (unit of work). (1) アプリケーション・プロセス内の回復可能な操作シーケンス。どの時点でも、アプリケーション・プロセスは、単一の操作単位であるが、アプリケーション・プロセスの寿命には、コミット操作またはロールバック操作の結果として多くの作業単位が関係する場合がある。(2) DRDA において、データベース・マネージャーが単一のエンティティとして扱う一連の SQL コマンド群。データベース・マネージャーは、ある作業単位の間に行われたすべてのデータ変更が実行されたか、あるいはデータ変更が 1 つも行われなかったかのどちらかを確認することによって、データの整合性を確保する。

異種 (unlike). 複数の異なる IBM 操作環境を指す語。たとえば、異種分散は DB2 (VM および VSE 版) と DB2 UDB for OS/390 間の分散である。同種 (*like*) と対比。

無名列 (unnamed column). 例示表に追加される空の列。無名列は、ターゲット表と同様に、列を組み合わせたり、行を組み合わせたり、あるいは定数値を報告書に組み込んだりするのに使用される。

USA (米国) フォーマット (United States of America format). 以下のように日付 / 時刻値を表すフォーマット。

- 日付: mm/dd/yyyy
- 時刻: hh:mm xM

値 (value). 表で行と列を割り当てられているデータ・エレメント。

バリエーション (variation). FORM.DETAIL パネルに指定されて、条件に応じて、1 つの報告書または報告書の部分のフォーマットの設定に使用できる、データ・フォーマット設定定義。

視点 (view). 1 つまたは複数の表にあるデータの代替表示。視点には、視点が定義されるもとになっている 1 つの表または複数の表に入っているすべての列、またはいくつかの列を組み込むことができる。(2) 照会用に検索されるデータの範囲を定義する 1 つまたは複数のエンティティ。

拡張仮想記憶 (Virtual Storage Extended). ディスク・オペレーティング・システム / 仮想記憶 (DOS/VS) の拡張版であるオペレーティング・システム。VSE は、(1) VSE/ 拡張機能サポート、および

(2) ユーザーのデータ処理ニーズを満たす必要のある IBM 提供およびユーザー作成のプログラムから成り立っている。VSE と VSE が制御するハードウェアは、両方で完全なコンピューター・システムを構成する。

VM. 仮想計算機 (Virtual Machine) (IBM のオペレーティング・システム)。VM/ESA 環境の総称。

VSE. 拡張仮想記憶 (Virtual Storage Extended) (IBM のオペレーティング・システム)。VSE/ESA 環境を総称的指す語。

WAIT. 無限待ち状態問題のキーワード。

ウィンドウ (window). パネルの全部または一部が表示される画面の長方形の部分。ウィンドウは、画面のサイズと等しくすることも、それ以下にすることもできる。

ワークステーション・データベース・サーバー (Workstation Database Server). UNIX および Intel のプラットフォーム上の IBM ファミリーの DRDA データベース・プロダクト。(たとえば、DB2 ユニバーサル・データベース (UDB)、DB2 コモン・サーバー、DB2 パラレル・エディション、および DataJoiner。)

折り返し (wrapping). 列の折り返し (*column wrapping*) および行の折り返し (*line wrapping*) を参照。

参考文献

以下の資料リストは、個々のライブラリーの全資料を示しているものではありません。これらの資料の注文、または個々のライブラリーの詳細については、IBM 担当員にお問い合わせください。

QMF 資料のリストについては、xiiiページの『QMF ライブラリー』を参照してください。

APPC の資料

Communicating with APPC and CPI-C: A Technical Overview
Networking with APPC: An Overview

CICS の資料

CICS Transaction Server for OS390

CICS/OS390 User's Handbook
CICS/OS390 アプリケーション・プログラミング解説書
CICS/OS390 アプリケーション・プログラミングの手引き
CICS/OS390 DB2 の手引き
CICS/OS390 資源定義 (マクロ)
CICS/OS390 資源定義 (オンライン)
CICS/OS390 Problem Determination Guide
CICS/OS390 システム定義の手引き
CICS/OS390 相互通信の手引き
CICS/OS390 Performance Tuning Handbook

CICS for VSE

- *CICS for VSE/ESA User's Handbook*
- *CICS for VSE/ESA 適用業務プログラミング 解説書*
- *CICS for VSE/ESA 適用業務プログラミングの手引き*
- *CICS for VSE/ESA 資源定義 (マクロ)*
- *CICS VSE/ESA 資源定義 (オンライン)*
- *CICS for VSE/ESA Problem Determination Guide*
- *CICS VSE/ESA システム定義の手引き*
- *CICS VSE/ESA 相互通信の手引き*

- *CICS for VSE/ESA Performance Tuning Handbook*

COBOL の資料

- COBOL II 適用業務プログラミングの手引き (VSE)*
- COBOL/VSE 言語解説書*
- COBOL/VSE プログラミングの手引き*

DATABASE 2 の資料

DB2 UDB (OS/390 版)

- DB2 UDB (OS/390 版) インストレーションの手引き*
- DB2 UDB (OS/390 版) 管理の手引き*
- DB2 UDB (OS/390 版) SQL 解説書*
- DB2 UDB (OS/390 版) コマンド解説書*
- DB2 UDB (OS/390 版) アプリケーション・プログラミングおよび SQL の手引き*
- DB2 UDB (OS/390 版) メッセージおよびコード*
- DB2 UDB (OS/390 版) ユーティリティーの手引きおよび解説書*
- DB2 UDB for OS390 Call Level Interface Guide and Reference*
- DB2 UDB (OS/390 版) リモート DRDA リクエスターおよびサーバー解説書*

DB2 (VSE および VM 版)

- DB2 Server for VM Installation Guide*
- DB2 Server for VSE Installation Guide*
- DB2 サーバー (VSE および VM 版) データベース管理*
- DB2 サーバー (VM 版) システム管理*
- DB2 サーバー (VSE 版) システム管理*
- DB2 サーバー (VSE および VM 版) オペレーション*
- DB2 サーバー (VSE および VM 版) SQL リファレンス*
- DB2 サーバー (VSE および VM 版) アプリケーション・プログラミング*
- DB2 サーバー (VSE および VM 版) ISQL ガイドおよびリファレンス*
- DB2 サーバー (VSE および VM 版) データベース・サービス・ユーティリティー*
- DB2 サーバー (VM 版) メッセージおよびコード*
- DB2 サーバー (VSE 版) メッセージおよびコード*
- DB2 Server for VSE and VM Diagnostic Guide and Reference*
- DB2 サーバー (VSE および VM 版) パフォーマンス・チューニング・ハンドブック*

DB2 (AS/400 版)

DB2 (OS/400 用) SQL 解説書

DB2 (OS/400 用) SQL プログラミング

パラレル・エディション

DB2 パラレル・エディション 管理の手引きおよび解説書

DB2 ユニバーサル・データベース

DB2 ユニバーサル・データベース コマンド解説書

DB2 ユニバーサル・データベース SQL 解説書

Universal Database Message Reference

DataJoiner

DataJoiner Application Programming and SQL Reference Supplement

DCF の資料

DCF and DLF General Information

DRDA の資料

DRDA Every Manager's Guide

DRDA 接続の手引き

DXT の資料

DXT Guide to Dialogs

Data Extract: Planning and Administration Guide for Dialogs

Data Extract: User's Guide

Learning to Use DXT

図形データ表示管理プログラム (GDDM) の資料

GDDM 概説書

GDDM Base Programming Reference

GDDM Base Programming Guide

GDDM 使用者の手引き

GDDM 導入およびシステム管理 (VSE)

GDDM メッセージ

HLASM の資料

IBM High-Level Assembler Programmer's Guide for OS/390, VM and VSE
IBM High-Level Assembler Language Reference for OS/390, VM and VSE

ISPF/PDF の資料

OS/390

ISPF (OS/390 版) 計画とカスタマイズ
ISPF (OS/390 版) ダイアログ・タグ言語 ガイドとリファレンス
Interactive System Productivity Facility for OS/390 Dialog Management Services and Examples

VM

ISPF for VM Dialog Management Services and Examples

OS/390 の資料

ユーティリティー

OS/390 Administration: Utilities
OS/390 Extended Architecture Utilities

JCL

OS/390 Extended Architecture JCL Reference
OS/390 Extended Architecture JCL User's Guide
OS/390 JCL Reference
OS/390 JCL Users Guide

ページング可能リンク・パック域 (PLPA)

OS/390 Extended Architecture Initialization and Tuning
OS/390 SPL: Initialization and Tuning

VSAM

OS/390 VSAM Administration Guide
OS/390 VSAM Catalog Administration Access Method Services

TSO

OS/390 TSO/E 入門
OS/390 TSO/E ユーザーズ・ガイド

SMP/E

OS/390 SMP/E メッセージおよびコード
OS/390 SMP/E 入門書
OS/390 SMP/E 解説書
OS/390 SMP/E 使用者の手引き

PL/I の資料

PL/I VSE/ESA 言語解説書

PL/I VSE/ESA プログラミングの手引き

REXX の資料**OS/390 環境**

REXX/370 コンパイラー 使用者の手引きと解説

TSO Extensions REXX/MVS Reference

VM 環境

Procedures Language VM/REXX Reference

Procedures Language VM/REXX User's Guide

ServiceLink の資料

ServiceLink User's Guide

VM の資料

Virtual Machine Planning Guide and Reference

Virtual Machine CMS Command and Macro Reference

VSE の資料

VSE Planning Guide

VSE Guide to System Functions

VSE System Utilities

VSE Guide for Solving Problems



プログラム番号: 5675-DB2

Printed in Japan

GC88-8623-00



日本アイ・ビー・エム株式会社

〒106-8711 東京都港区六本木3-2-12

Spine information:



照会報告書作成プログラム QMF (OS/390 版) V7R1 インストールおよび管理の手引き

バージョン 7